

Pasantía
Software de gestión de documentos archivo para el Fondo para la educación superior de Tocancipá (FOES)

Presentado por:

Rafael Alejandro Rozo Vega

**UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA
ESCUELA DE CIENCIAS BASICAS TECNOLOGIA E INGENIERIA
PROGRAMA INGENIERIA DE SISTEMAS
COLOMBIA**

Tabla de Contenidos

Capítulo 0 Resumen	9
Capítulo 0 Abstract	10
Capítulo 1 Introducción	11
Capítulo 2 Objetivos	12
2.1 Objetivos Generales.....	12
2.2 Objetivos Específicos.....	12
Capítulo 3 Justificación	13
Capítulo 4 Metodología	14
4.1 Tipo de Metodología.....	14
4.2 Metodología de Investigación.....	14
Capítulo 5 Marco Conceptual y Teórico	15
5.1 Introducción a PHP	15
5.1.1 Características.....	15
5.2.1 Programación Básica	16
5.2.2 Aspectos generales.....	16
5.2.3 Estructuras básicas.....	17
5.3 Programación Avanzada	18
5.3.1 ARCHIVOS.....	18
5.3.2 BASES DE DATOS.....	20
5.3.3 EXCEPCIONES.....	21
Capítulo 6 Modelo de Requisitos	22
6.1. Descripción del Problema	22
6.2. Determinación de requerimientos	22
6.2.1. Requerimientos Funcionales.....	23
6.2.2. Requerimientos no funcionales.....	24
6.3. Modelo de Caso de Uso	25
6.3.1 Actores.....	25
6.3.2 Casos de uso.....	26
6.4. Modelo de Interfaces	27
6.5. Actores y Caso de Uso	28
6.5.1 Actores.....	28

6.5.2	Casos de uso.....	29
6.6.	Modelo de Dominio del Problema.....	49
6.6.1	Identificación de clases.....	49
6.6.2	Identificación de asociaciones.....	50
6.6.3	Identificación de atributos.....	51
6.6.4	Diccionario de clases.....	52
6.6.5	Identificación de Módulos.....	53
Capítulo 7	Modelo de Análisis.....	54
7.1.	Arquitectura de Clases.....	54
7.1.1	Clases con estereotipos.....	54
7.1.2	Clases para casos de uso.....	54
7.2.	Identificación de Clases según Estereotipos.....	55
7.2.1	Borde.....	55
7.2.2	Entidad.....	59
7.2.3	Control.....	60
7.3.	Clases según Casos de Uso.....	62
7.3.1	Validar funcionario.....	62
7.3.2	Estudiantes foest.....	63
7.3.3	Registrar funcionario.....	63
7.3.4	Registrar estudiantes.....	63
7.3.5	Registrar beneficio1.....	63
7.3.6	Registrar beneficio2.....	64
7.3.7	Registrar algún otro beneficio.....	64
7.3.8	Registrar antigüedad Sisbén.....	64
7.3.9	Registrar tipos de poblacion.....	64
7.3.10	Registrar universidades.....	65
7.3.11	Registrar carreras.....	65
7.3.12	Registrar ciudad.....	65
7.3.13	Certificado estudiante.....	65
7.4.	Diagramas de Secuencia.....	66
7.4.1	Registrar funcionario.....	66
7.4.2	Registrar estudiante.....	68
7.4.3	Registrar Beneficio1.....	69
7.4.4	Registrar Beneficio2.....	70
7.4.5	Registrar Algunotrobeneficio.....	72
7.4.6	Registrar AntigüedadSisbén.....	74
7.4.7	Registrar tipos de poblacion.....	75
7.4.8	Registrar universidades.....	77
7.4.9	Registrar carreras.....	78
7.4.10	Registrar ciudad.....	80
7.4.11	Certificado estudiante.....	81
7.5.	Casos de Uso para el Sistema.....	82

7.5.1	Validar funcionario.....	82
7.5.2	Estudiantes foest.....	83
7.5.3	Registrar funcionario.....	87
7.5.4	Registrar estudiantes.....	89
7.5.5	Registrar beneficio1.....	91
7.5.6	Registrar beneficio2.....	92
7.5.7	Registrar algún otro beneficio.....	94
7.5.8	Registrar antigüedad Sisbén.....	96
7.5.9	Registrar tipos de poblacion.....	98
7.5.10	Registrar universidades.....	99
7.5.11	Registrar carreras.....	101
7.5.12	Registrar ciudad.....	103
7.5.13	Certificado estudiante.....	105
7.6.	Diccionario de Clases.....	106
7.6.1	Interface Usuario.....	106
7.6.2	Ingreso.....	106
7.6.3	Principal.....	107
7.6.4	Registro.....	107
Capítulo 8	Modelo de Diseño.....	112
8.1.	Diseño de Objetos.....	112
8.1.1	Tarjetas de clases.....	112
8.1.2	Responsabilidades.....	132
8.1.3	Colaboraciones.....	161
8.1.4	Jerarquías.....	188
8.1.5	Contratos.....	216
8.1.6	Subsistemas.....	246
8.1.7	Atributos.....	251
8.1.8	Algoritmos.....	268
8.2.	Diseño de Sistema.....	295
8.2.1	Interfaces gráficas.....	295
8.2.2	Bases de datos.....	296
8.3.	Diagrama de Secuencia del Diseño.....	300
8.3.1	Registrar Usuario Funcionario: Crear Registro Usuario funcionario.....	300
8.3.2	Registrar Usuario Funcionario: Actualizar Registro Usuario funcionario.....	300
8.3.3	Registrar Usuario Funcionario: Eliminar Registro Usuario funcionario.....	301
8.3.4	Registrar Usuario Estudiante: Crear Registro Usuario estudiante.....	301
8.3.5	Registrar Usuario Estudiante: Actualizar Registro Usuario estudiante.....	302
8.3.6	Registrar Usuario Estudiante: Eliminar Registro Usuario estudiante.....	302
8.3.7	Registrar beneficio 1: Crear Registro beneficio 1.....	303
8.3.8	Registrar beneficio 1: Actualizar Registro beneficio 1.....	303
8.3.9	Registrar beneficio 1: Eliminar Registro beneficio 1.....	304
8.3.10	Registrar beneficio 2: Crear Registro beneficio 2.....	304
8.3.11	Registrar beneficio 2: Actualizar Registro beneficio 2.....	305
8.3.12	Registrar beneficio 2: Eliminar Registro beneficio 2.....	305

8.3.13	Registrar algún otro beneficio: Crear Registro algún otro beneficio	306
8.3.14	Registrar algún otro beneficio: Actualizar Registro algún otro beneficio	306
8.3.15	Registrar algún otro beneficio: Eliminar Registro algún otro beneficio	307
8.3.16	Registrar antigüedad Sisbén: Crear Registro antigüedad Sisbén	307
8.3.17	Registrar antigüedad Sisbén: Actualizar Registro antigüedad Sisbén.....	308
8.3.18	Registrar antigüedad Sisbén: Eliminar Registro antigüedad Sisbén.....	308
8.3.19	Registrar tipos de poblacion: Crear Registro tipos de poblacion.....	309
8.3.20	Registrar tipos de poblacion: Actualizar Registro tipos de poblacion.....	309
8.3.21	Registrar tipos de poblacion: Eliminar Registro tipos de poblacion.....	310
8.3.22	Registrar universidades: Crear Registro universidades.....	310
8.3.23	Registrar universidades: Actualizar Registro universidades.....	311
8.3.24	Registrar universidades: Eliminar Registro universidades.....	311
8.3.25	Registrar carreras: Crear Registro carreras.....	312
8.3.26	Registrar carreras: Actualizar Registro carreras.....	312
8.3.27	Registrar carreras: Eliminar Registro carreras.....	313
8.3.28	Registrar ciudad: Crear Registro ciudad.....	313
8.3.29	Registrar ciudad: Actualizar Registro ciudad.....	314
8.3.30	Registrar ciudad: Eliminar Registro ciudad.....	314
8.3.31	Certificado: pedir certificado.....	315
8.3.32	Estadísticas: Estadísticas.....	315

Capítulo 9: Modelo de Implementación Codificación..... 316

9.1. Programación en PHP..... 316

9.1.1	Login.....	316
9.1.2	Conexion.....	318
9.1.3	Validación usuario.....	318
9.1.4	Archivo de bloqueo.....	319
9.1.5	Hoja de estilos.....	320
9.1.6	Página principal.....	322
9.1.7	Menú opciones.....	323
9.1.8	Nueva base de datos.....	324
9.1.9	Enlazador para nuevos periodos.....	327
9.1.10	Nuevo periodo.....	328
9.1.11	Pantalla creación de registro por periodos.....	339
9.1.12	Pantalla pedir documento para modificar (Cuando se crea un periodo nuevo)	363
9.1.13	Pantalla para modificar datos de usuario (Cuando se crea un periodo nuevo)	365
9.1.14	Pantalla estadísticas para nuevos periodos.....	395
9.1.15	Archivo que toma las variables y las envía a la base de datos (para periodos nuevos)	442
9.1.16	Archivo que manipula un registro de un periodo, para actualizarlo.....	446
9.1.17	Pantalla donde se pide el documento del beneficiario para la creación de un nuevo registro....	449
9.1.18	Menú periodo.....	451
9.1.19	FUID.....	451
9.1.20	Registrar Fuid.....	455

9.1.21	Archivo que toma los datos de registro de la pantalla registrar fuid y los pasa a la base de datos	457
9.1.22	Pantalla modificar beneficiario fuid	458
9.1.23	Archivo que lleva los datos de la pantalla modificar a la base de datos	461
9.1.24	Pantalla beneficios 1 registrados	461
9.1.25	Pantalla registro beneficio1	465
9.1.26	Archivo que lleva los datos de registro beneficio 1 a la base de datos	467
9.1.27	Pantalla modificar beneficio 1	467
9.1.28	Archivo que lleva los datos de modificar beneficio 1 a la base de datos	469
9.1.29	Archivo que borra datos de beneficio 1 de la base de datos	470
9.1.30	Pantalla beneficios 2 registrados	470
9.1.31	Pantalla registro beneficio 2	474
9.1.32	Archivo que lleva los datos de registro beneficio 2 a la base de datos	475
9.1.33	Pantalla modificar beneficio 2	476
9.1.34	Archivo que lleva los datos de modificar beneficio 2 a la base de datos	478
9.1.35	Archivo que borra datos de beneficio 2 de la base de datos	479
9.1.36	Pantalla otros beneficios registrados	479
9.1.37	Pantalla registro otros beneficios	482
9.1.38	Archivo que lleva los datos de registro otros beneficios a la base de datos	484
9.1.39	Pantalla modificar otros beneficios	485
9.1.40	Archivo que lleva los datos de modificar otros beneficios a la base de datos	487
9.1.41	Archivo que borra datos de otros beneficios de la base de datos	487
9.1.42	Pantalla antigüedad Sisbén	488
9.1.43	Pantalla registro antigüedad Sisbén	491
9.1.44	Archivo que lleva los datos de registro antigüedad Sisbén a la base de datos	493
9.1.45	Pantalla modificar antigüedad Sisbén	494
9.1.46	Archivo que lleva los datos de modificar antigüedad Sisbén a la base de datos	496
9.1.47	Archivo que borra datos de antigüedad Sisbén de la base de datos	496
9.1.48	Pantalla tipos de poblacion	497
9.1.49	Pantalla registro tipos de poblacion	500
9.1.50	Archivo que lleva los datos de registro tipos de poblacion a la base de datos	502
9.1.51	Pantalla modificar tipos de poblacion	502
9.1.52	Archivo que lleva los datos de modificar tipos de poblacion a la base de datos	505
9.1.53	Archivo que borra datos de tipos de poblacion de la base de datos	505
9.1.54	Pantalla universidades registradas	506
9.1.55	Pantalla registro universidades	509
9.1.56	Archivo que lleva los datos de registro universidades a la base de datos	511
9.1.57	Pantalla modificar universidades	512
9.1.58	Archivo que lleva los datos de modificar universidades a la base de datos	514
9.1.59	Archivo que borra datos de universidades de la base de datos	514
9.1.60	Pantalla carreras registradas	515
9.1.61	Pantalla registro carreras	518
9.1.62	Archivo que lleva los datos de registro carreras a la base de datos	520
9.1.63	Pantalla modificar carreras	521
9.1.64	Archivo que lleva los datos de modificar carreras a la base de datos	523

9.1.65	Archivo que borra datos de carreras de la base de datos.....	523
9.1.66	Pantalla ciudad registradas.....	524
9.1.67	Pantalla registro ciudad.....	527
9.1.68	Archivo que lleva los datos de registro ciudad a la base de datos.....	529
9.1.69	Pantalla modificar ciudad.....	529
9.1.70	Archivo que lleva los datos de modificar ciudad a la base de datos.....	531
9.1.71	Archivo que borra datos de ciudad de la base de datos.....	532
Capítulo 10: Modelo de Pruebas.....		533
10.1.	Pruebas de caja negra.....	533
10.1.1	Prueba de validación.....	533
10.1.2	Prueba de creación de funcionario.....	535
10.1.3	Prueba de cambio de contraseña de funcionario.....	537
10.1.4	Prueba de eliminación de funcionario.....	540
10.1.5	Prueba de creación de un nuevo periodo.....	541
10.1.6	Prueba de registro de un nuevo beneficiario del periodo.....	542
10.1.7	Prueba de una exportación de un periodo.....	545
10.1.8	Prueba de modificación de un beneficiario.....	546
10.1.9	Prueba de estadísticas.....	549
10.1.10	Prueba de exportación de estadísticas.....	550
10.1.11	Prueba de creación de beneficiario en fuid.....	552
10.1.12	Prueba de exportación de fuid.....	553
10.1.13	Prueba de modificación de beneficiario fuid.....	555
10.1.14	Prueba de creación de beneficio 1.....	556
10.1.15	Prueba de modificación beneficio 1.....	558
10.1.16	Prueba de eliminación beneficio 1.....	559
10.1.17	Prueba de creación de beneficio 2.....	560
10.1.18	Prueba de modificación beneficio 2.....	561
10.1.19	Prueba de eliminación beneficio 2.....	564
10.1.20	Prueba de creación de algún otro beneficio.....	565
10.1.21	Prueba de modificación algún otro beneficio.....	567
10.1.22	Prueba de eliminación algún otro beneficio.....	568
10.1.23	Prueba de creación antigüedad Sisbén.....	569
10.1.24	Prueba de modificación antigüedad Sisbén.....	570
10.1.25	Prueba de eliminación antigüedad Sisbén.....	572
10.1.26	Prueba de creación poblacion especial.....	573
10.1.27	Prueba de modificación poblacion especial.....	575
10.1.28	Prueba de eliminación poblacion especial.....	577
10.1.29	Prueba de creación de registro de universidad.....	578
10.1.30	Prueba de modificación de registro de universidad.....	580
10.1.31	Prueba de eliminación de registro de universidad.....	581
10.1.32	Prueba de creación de registro de carreras.....	582
10.1.33	Prueba de modificación de registro de carreras.....	584
10.1.34	Prueba de eliminación de registro de carreras.....	585

10.1.35 Prueba de creación de registro de ciudad.....	586
10.1.36 Prueba de modificación de registro de ciudad.....	588
10.1.37 Prueba de eliminación de registro de ciudad.....	590

Capítulo 11 Conclusiones.....	591
--------------------------------------	------------

Capítulo 12 Referencias Bibliográficas.....	592
--	------------



Capítulo 0 Resumen

El manejo de documentos en una organización es importante para su correcto funcionamiento, pero algunas empresas tienen demasiados documentos o archivos, lo cual dificulta la búsqueda de los mismos, y esto ocasiona la creación de duplicados, la pérdida de documentos, entre otros.

Existen softwares para la gestión de documentos, pero en su mayoría son costosos, o difíciles de manejar, lo cual no beneficia en mucho a microempresas o empresas que hasta ahora están iniciando, tampoco benefician a inventarios pequeños de oficinas las cuales quieren llevar sus propios inventarios, independiente a la organización.

En el siguiente documento, se encontrará el resultado de las pasantías desarrolladas en la Alcaldía de Tocancipá, la cual consistió en el desarrollo de Software de gestión de documentos archivo para el Fondo para la educación superior de Tocancipá (FOES), este se desarrolló con el lenguaje PHP, este software, permite la creación de nuevos periodos, el registro de usuarios (funcionarios), Beneficiarios, universidades, y otros.

PALABRAS CLAVES: PHP; Bases de datos; Software; Codificación; FOEST; Registro.

Capítulo 0 Abstract

The handling of documents in an organization is important for its proper functioning, but some companies have too many documents or files, which makes it difficult to search for them, and this causes the creation of duplicates, the loss of documents, among others.

There are softwares for the management of documents, but they are mostly expensive, or difficult to manage, which does not benefit much to microenterprises or companies that until now are beginning, nor benefit small inventories of offices which want to carry their own inventories , independent to the organization.

In the following document, you will find the result of the internships developed in the Mayor's Office of Tocancipá, which consisted in the development of File Management Software for the Higher Education Fund of Tocancipá (FOES), this was developed with the PHP language, this software, allows the creation of new periods, the registration of users (officials), Beneficiaries, universities, and others.

KEY WORDS: PHP; Databases; Software; Coding; FOEST; Registry.

Capítulo 1 Introducción

El manejo de documentos en una organización es importante para su correcto funcionamiento, pero algunas empresas tienen demasiados documentos o archivos, lo cual dificulta la búsqueda de los mismos, y esto ocasiona la creación de duplicados, la pérdida de documentos, entre otros. Existen softwares para la gestión de documentos, pero en su mayoría son costosos, o difíciles de manejar, lo cual no beneficia en mucho a microempresas o empresas que hasta ahora están iniciando, tampoco benefician a inventarios pequeños de oficinas las cuales quieren llevar sus propios inventarios, independiente a la organización

Capítulo 2 Objetivos

2.1 Objetivo General

- Desarrollar una aplicación software de gestión de documentos para el FOES de Tocancipá.

2.2 Objetivos Específicos

- Identificar las necesidades, requerimientos funcionales y no funcionales del software.
- Construir el modelo de la estructura del software en UML.
- Aplicar pruebas de caja negra al software.
- Implementar el software desarrollado.

Capítulo 3 Justificación

La secretaria de educación del municipio de Tocancipá se encuentra vinculada al Fondo para la Educación Superior (FOES) en el cual se ofrecen servicios y ayudas como subsidios en universidades, también cuenta con ayudas en instituciones de bachillerato, allí acuden varios jóvenes, para ello la secretaria de educación cuenta con un archivo el cual cuenta con más de 2000 expedientes, los cuales son difíciles de hallar, porque se encuentran en posiciones incorrectas o porque han sido prestadas.

Este archivo, es organizado de manera manual y no tiene un control adecuado cuando se prestan o se consultan los documentos o expedientes, ya que no hay una sola persona encargada, y cualquier empleado puede acceder a ellos, ocasionando pérdidas y desordenes en el archivo tanto físico como digital, también generando expedientes duplicados.

La información que se encuentra allí consignada es de alto valor, ya que han invertido dinero, brindando auxilios a los jóvenes en sus estudios de educación superior.

Es por ello que se desarrollara un software fácil de manejar, para la gestión de archivos, en el cual podremos encontrar los datos y la ubicación de un determinado documento o expediente en el archivo físico o digital, conocer la cantidad de versiones que tiene un documento, editar datos, eliminar registro, crear registros de las personas que utilizan el archivo, ya sean para consulta o préstamo, dando a conocer los plazos de devolución, también ayudara a organizar el archivo, ya que indicara que expedientes o documentos hacen falta y donde se pueden encontrar.

El software permitirá el uso de filtros, para una búsqueda rápida y sencilla, los filtros que se usarán serán usuario e identificación, además permitirá crear y restaurar copias de seguridad.

Capítulo 4 Metodología

4.1 Tipo de Metodología

El tipo de metodología es la inductiva en donde se toman datos generales como válidos.

4.2 Metodología de Investigación

La utilización de la técnica de investigación descriptiva se utiliza para obtener datos a través del instrumento de recolección como la observación directa.

Observación directa (realizada por Rafael Alejandro Rozo Vega)

Los funcionarios de la oficina del FOEST, realiza tramites todos los días, tanto recibiendo peticiones o solicitudes de estudiantes como con otras oficinas y secretarias, el FOEST cuenta con un sistema de registro desarrollado en un archivo Excel el cual se encuentra alojado en un computador, de allí se comparte a otras estaciones de trabajo, el archivo puede ser visto en todas las estaciones al mismo tiempo, pero solo se puede modificar en una estación a la vez, este archivo contiene información de estudiantes, como su nombre, apellidos, tipo de documento, numero de documento, teléfono, sexo, si pertenece a una población especial, nombre la universidad, semestre, horarios, si es beneficiario a subsidio de transporte, entre otros.

Cada inicio de semestre crea un nuevo registro en Excel, en el cual ingresan los datos de los estudiantes que son beneficiarios del subsidio y/o convenios, esto dificulta un poco las consultas de los funcionarios, ya que cuando quieren saber cuántas veces ha recibido el beneficio, deben revisar archivo por archivo

Actualmente se encuentran realizando actualizaciones a sus formularios y registros a Excel, lo cual ocasionara un cambio significativo en su sistema de registro.

El archivo físico está organizado con un numero de carpeta por estudiante, al igual que el digital el cual se encuentra en un solo computador, tanto el físico como el digital manejan el mismo número por estudiante.

Capítulo 5 Marco Conceptual y Teórico

5.1 Introducción a PHP

PHP es un potente lenguaje, y su intérprete, bien como módulo del servidor web o bien como binario CGI, puede acceder a ficheros, ejecutar comandos o abrir conexiones de red desde el servidor. Estas propiedades hacen que, por omisión, sea inseguro todo lo que se ejecute en un servidor web. PHP está diseñado específicamente para ser un lenguaje más seguro para escribir aplicaciones CGI que Perl o C. Partiendo de un correcto ajuste de opciones de configuración para tiempo de ejecución y en tiempo de compilación, y el uso de prácticas de programación apropiadas, pueden proporcionarle la combinación de libertad y de seguridad que necesita.

Dado que hay muchas vías para ejecutar PHP, existen muchas opciones de configuración para controlar su comportamiento. Al haber una extensa selección de opciones se garantiza poder usar PHP para un gran número de propósitos, pero a la vez significa que existen combinaciones que conllevan una configuración menos segura.

La flexibilidad de configuración de PHP rivaliza igualmente con la flexibilidad de su código. PHP puede ser usado para construir completas aplicaciones de servidor, con toda la potencia de un usuario de consola, o se puede usar sólo desde el lado del servidor implicando un menor riesgo dentro de un entorno controlado. El cómo construir ese entorno, y cómo de seguro es, depende del desarrollador PHP.

5.1.1 Características

- **Velocidad:** PHP no solo es rápido al ser ejecutado, sino que no genera retrasos en la máquina, por esto no requiere grandes recursos del sistema. PHP se integra muy bien junto a otras aplicaciones, especialmente bajo ambientes Unix.
- **Estabilidad:** PHP utiliza su propio sistema de administración de recursos y posee de un sofisticado método de manejo de variables, conformando un sistema robusto y estable.

- **Seguridad:** PHP maneja distintos niveles de seguridad, estos pueden ser configurados desde el archivo .ini

- **Simplicidad:** Usuarios con experiencia en C y C++ podrán utilizar PHP rápidamente. Además, PHP dispone de una amplia gama de librerías, y permite la posibilidad de agregarle extensiones. Esto le permite su aplicación en múltiples áreas, tales como encriptado, gráficos, XML y otras.

5.2 Programación Básica

5.2.1 Aspectos generales

- **Comentarios**

En cualquier lenguaje deben distinguirse el código y los comentarios, en PHP existen dos tipos distintos para la especificación de comentarios:

- **Comentario de una sola línea** es utilizado para las notas rápidas sobre el código complejo o para desactivar temporalmente una línea de código PHP. Es necesario añadir // o # antes del código.

Ejemplo:

```
//Mi comentario  
#define ('TABLE_HEADING_WRITABLE', 'Writable');
```

- **Comentario de líneas múltiples** es utilizado para comentar los bloques de código o escribir comentarios de múltiples líneas. Es necesario añadir /* antes y */ después del código.

Ejemplo:

```
/*$Id$  
osCommerce, Open Source E-Commerce Solutions  
http://www.oscommerce.com  
Copyright (c) 2010 osCommerce  
Released under the GNU General Public License  
*/
```


- **Palabras reservadas**

Estas palabras tienen un significado especial en PHP. Algunas representan cosas que se parecen a funciones, otras se parecen a constantes, etc.; pero no lo son en realidad: son constructores del lenguaje. No se puede usar ninguna de las siguientes palabras como constantes, nombres de clases, nombres de funciones o métodos. Se pueden usar como nombres de variables, pero podría dar lugar a confusiones.

En PHP existen 67 palabras.

- **Identificadores:**

Dentro de las palabras que un usuario puede definir se encuentran los identificadores, las cuales sirven para relacionarse con las estructuras del programa, los identificadores guardan valores de algún tipo, también son conocidos como variables.

- **Oraciones**

Corresponden a una línea de ejecución, y tiene como carácter final “;”.

5.2.2 Estructuras básicas

Variables

Se representan por medio de un nombre seleccionado dentro de los posibles identificadores.

Declaraciones

Consiste en relacionar una variable con el tipo de dato que va a guardar.

Constantes

Como parte de las declaraciones de las variables, Java permite distintos tipos de modificadores, que afectan los diversos factores de las variables.

Arreglos

Es una estructura presente en la gran mayoría de los lenguajes, aunque es similar a los demás lenguajes su manejo es diferente, se debe a que los arreglos se manipulan por referencia, al igual que los objetos.

Cadenas

Como en los demás lenguajes, las cadenas son como arreglos de caracteres.

Asignación

Las variables de tipos primitivos se utilizan para guardar valores de sus respectivos tipos, es la expresión más importante que hay es la de asignación de valores.

Operadores

En Java todos los tipos son valores con signo.

Control

Son expresiones que utilizan los lenguajes para controlar el flujo de la lógica del programa. Estas expresiones son bastante estandarizadas en los lenguajes modernos, aunque varían en ciertos detalles.

5.3 Programación Avanzada

5.3.1.ARCHIVOS

Los archivos en PHP se enuncian mediante variables, con ellas se pueden crear archivos, revisar y modificar el contenido.

- **Código para crear y modificar un archivo**

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<?php
```

```
$nombre_archivo = "logs.txt";

if(file_exists($nombre_archivo)
{
    $mensaje = "El Archivo $nombre_archivo se ha modificado";
}
Else
{
    $mensaje = "El Archivo $nombre_archivo se ha creado";
}

if ($archivo = fopen($nombre_archivo, "a"))
{
    if(fwrite($archivo, date("d m Y H:m:s"). " ". $mensaje. "\n"))
    {
        echo "Se ha ejecutado correctamente";
    }
    else
    {
        echo "Ha habido un problema al crear el archivo";
    }
    fclose($archivo);
}
?>
```

- **Código para ver el contenido de un archivo**

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<?php

    $nombre_archivo = "logs.txt"; //variable con el nombre del archivo que vamos a crear

    if(file_exists($nombre_archivo))
    {
        echo nl2br(file_get_contents($nombre_archivo));
    }
    else
    {
        $mensaje = "El archivo no existe";
    }
}
?>
```

5.3.2 BASES DE DATOS

Acceder a una base de datos es relativamente sencillo, hay que crear una variable la cual obtendrá la conexión, pero también a este código se le añade un código para que muestre un error en caso de que la conexión con la base de datos no se pueda realizar

```
$conexion = new mysqli ('servidor', 'usuario', 'contraseña', 'nombre base de datos') or die("No se conectó a la base de datos");
```

Consultas

```
$consulta = "SELECT * FROM alumnos";  
$resultado = mysqli_query ($conexion, $consulta);
```

Mostrar datos de una base de datos

```
echo "<table borde='2'>";  
echo "<tr>";  
echo "<th>Nombre</th>";  
echo "<th>Edad</th>";  
echo "</tr>";  
while ($columna = mysqli_fetch_array( $resultado))  
{  
    echo "<tr>";  
    echo "<td> ". $columna['nombre']. "</td><td> ". $columna['edad']. "</td>";  
    echo "</tr>";  
}  
echo "</table>";
```

Cerrar conexión

```
mysqli_close($conexion);
```

Crear registro

```
$conexion = mysqli_query($conexion, "insert into alumno(nombre)VALUES('$nombre')");
```

Modificar registro

```
$conexion = mysqli_query($conexion, "update alumnos set nombre='".$nombre."' where ID='".$ID."'");
```

Eliminar registro

```
$conexion = mysqli_query($conexion, "delete from alumno where alumno. nombre = '$nombre');"
```

5.3.3 EXCEPCIONES

Es un aspecto de suma importancia el manejo de excepciones, cuando ocurre un error en el programa, el programa lanza una excepción. El manejo de excepciones se requiere para cualquier código que pueda resultar en estados inconsistentes, en particular en situaciones en donde se trate de acceder a entidades externas al programa.



Capítulo 6 Modelo de Requisitos

6.1. Descripción del Problema

El archivo de registro en Excel utilizado en el FOEST genera problemas entre los funcionarios, especialmente cuando son épocas de matrículas en las universidades ya que es en ese momento es cuando más estudiantes se encuentran realizando trámites ante el FOEST y este archivo solo se puede modificar en una estación de trabajo a la vez.

A demás este archivo puede sufrir modificaciones no deseadas ya que el usuario puede estar trabajando en algo más y sin darse cuenta puede estar modificando el archivo, ocasionando perdida de información importante y ocasionando demoras en los demás procesos.

El uso de este tipo de archivos no permite la creación de informes, certificaciones y estadísticas de manera inmediata, lo cual lleva al usuario abrir otro programa y realizarlo manualmente.

El control de préstamos del archivo, a otras dependencias u oficinas no es el adecuado por qué se hace a memoria o anotando en cualquier papel, lo cual puede ocasionar la perdida de algún documento importante.

6.2. determinación de requerimientos

1. ¿Cuál es el proceso básico del FOEST?

- **¿Cuál es la finalidad del FOES?**

Es el Fondo para la Educación Superior de Tocancipá, el cual Brinda subsidios y auxilios en el pago de matrícula y transporte a la población del municipio de Tocancipá, la cual se debe encontrar inscrita a una universidad la cual debe tener código SNIES, a cambio de prestar horas de servicio social, los estudiantes que no cumplen con estas horas de servicio social, que tengan un promedio inferior a 3.7 o que pierdan una materia, perderán las diferentes ayudas en el siguiente semestre.

- **¿Cuánto tiempo tarda la actividad?**

Tiene varios tiempos de duración los cuales dependen de las tareas, ya que en consultas puede tardar pocos minutos, pero en registros y actualizaciones duran más tiempo y las estadísticas pueden demorar horas o días.

- **¿Con cuanta frecuencia se realiza?**

De forma continua.

- **¿Para qué se utiliza la información resultante?**

Para realizar el seguimiento y control de los estudiantes inscritos, ya que cada uno de ellos recibe un beneficio monetario, el cual se debe devolver con horas sociales, también para el desarrollo de estadísticas, certificados e informes.

2. ¿Qué datos utiliza o produce este proceso?

El FOEST utiliza datos personales de los estudiantes, los cuales son más de 2000 que están realizando una carrera o que están a punto de iniciarla, pero no involucra a todos las instituciones de educación superior, sino a las que tiene un convenio vigente firmado con la alcaldía de Tocancipá, respecto a los estudiantes que se encuentran inscritas en dichas universidades deben estar registradas en el Sisbén del municipio con una antigüedad de 3 años.

3. ¿Qué frecuencia y volumen del proceso existe?

Su frecuencia es continua y el volumen de información manejada aumenta a diario, en especial cada vez que se abren matriculas en las diferentes universidades.

4. ¿Qué controles utiliza para su realización?

El FOEST lleva un seguimiento de todos los estudiantes inscritos, de esta manera se puede determinar cuándo un estudiante cumple con los requisitos para continuar con el proceso, o se suspende del programa.

6.2.1. Requerimientos funcionales

Interfaz grafica

- Los campos de: identificación, teléfono, puntajes, porcentajes, montos y promedios aceptaran valores numéricos únicamente.
- Los campos de: nombres, apellidos y ciudades aceptaran valores alfabéticos únicamente.

- Los campos de: beneficios, tipo de documento, edad, población especial, genero, antigüedad del Sisbén, universidad, nivel, carrera, modalidad, semestre, materias perdidas, dependencias, ciudad y días viajados y meses a girar; serán listas de selección.
- El campo de fecha será una lista de selección tipo Date.
- Los demás campos aceptaran valores alfabéticos, numéricos y especiales.

Seguridad

- El sistema controlará el acceso y lo permitirá solamente a usuarios autorizados. Los usuarios deben ingresar al sistema con un nombre de usuario y contraseña.

6.2.2. Requerimientos no funcionales

Eficiencia

- Toda la funcionalidad de la aplicación debe responder al usuario en menos de 10 segundos.
- Los datos modificados en la base de datos deben ser actualizados para todos los usuarios que acceden en menos de 5 segundos.

Usabilidad

- La aplicación debe proporcionar mensajes informativos al usuario (modificación de datos, nuevos registros eliminación de registro, etc.).
- La aplicación debe ajustarse al tamaño de pantalla de los diferentes dispositivos.

Otros

- La aplicación es compatible con todos los sistemas operativos ya que es una aplicación web.
- La aplicación consumirá menos de 50Mb de memoria RAM en los dispositivos.
- La aplicación con su respetiva base de datos no pesara más de 100MB.
- La interfaz de usuario será implementada para navegadores web.

6.3. Modelo de Caso de Uso

6.3.1 Actores



6.3.2 Casos de uso



6.3. Modelo de Interfaces

En esta etapa inicial del proyecto no es muy importante el diseño de las pantallas.

Pantallas:

P1: Pantalla ingreso.

P2: pantalla principal.

P3: Pantalla estudiantes registrados.

P4: Pantalla registro estudiantes.

P5: Pantalla modificación estudiantes

P6: Pantalla beneficio 1.

P7: Pantalla registro beneficio 1.

P8: Pantalla modificación beneficio 1.

P9: Pantalla beneficio 2.

P10: Pantalla registro beneficio 2.

P11: Pantalla modificación beneficio 2.

P12: Pantalla algún otro beneficio.

P13: Pantalla registro algún otro beneficio.

P14: Pantalla modificación algún otro beneficio.

P15: Pantalla antigüedad Sisbén.

P16: Pantalla registro antigüedad Sisbén.

P17: Pantalla modificación antigüedad Sisbén.

P18: Pantalla tipos de población.

P19: Pantalla registro tipos de población.

P20: Pantalla modificación tipos de población.

P21: Pantalla universidades.

P22: Pantalla registro universidades.

P23: Pantalla modificación universidades.

P24: Pantalla carreras.

P25: Pantalla registro carreras.

P26: Pantalla modificación carreras.

P27: Pantalla ciudad.

P28: Pantalla registro ciudad.

P29: Pantalla modificación ciudad.

P30: Pantalla estadística.

P31: Pantalla certificados.

P32: Pantalla crear funcionarios.

P33: Pantalla funcionarios registrados.

P34: Pantalla documentos.

P35: Pantalla salir.

6.5. Actores y Caso de Uso

6.5.1 Actores

Actor	Usuario.
Casos de uso	Validar usuario, registrar usuario, registrar estudiante, consultar estudiantes, certificado estudiante, registrar universidad.
Tipo	Primario.
Descripción	Es el actor principal y representa a los usuarios que hagan uso del software de gestión de documentos archivo para el fondo para la educación superior Tocancipá (FOEST).

Actor	Bases de datos foest.
Casos de uso	Validar usuario, registrar usuario, Estudiantes foest, universidad.
Tipo	Secundario.
Descripción	Es un actor secundario y representa la base de datos donde se almacena toda la información que está relacionada con los usuarios y estudiantes.

6.5.2 Casos de uso

Caso de uso	Validar funcionario.
Actores	Usuario, base de datos foest.
Tipo	Inclusión.
Propósito	Valida a los usuarios ya registrados para el uso del sistema.
Resumen	Este caso inicia cada vez que el usuario entra a su cuenta mediante un login y password, este lo verifica con el respectivo registro, para autorizar el ingreso al sistema.
Precondiciones	Se requiere que el usuario se encuentre registrado antes.
Flujo principal	Se presenta en la página ingreso (p1), el usuario puede seleccionar la siguiente acción: ingresar. La actividad ingresar: valida la opción con un login y password, ingresados por el usuario(e1), una vez realizado el proceso de validación se continua con el caso de uso ofrecer productos
Subflujos	Ninguno.
Excepciones	e1. No se realizó la validación el login y/o password incorrectos. Se solicita al usuario registrarse o ingresar los datos correctos.

Caso de uso	Estudiantes foest.
Actores	Usuario.
Tipo	Inclusión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	El usuario, tiene la capacidad de utilizar las opciones que ofrece el sistema
Precondiciones	Se requiere que el usuario se halla validado correctamente antes.

Flujo principal	<p>Se presenta en la página de estudiantes registrados (p2), el usuario puede observar los estudiantes registrado, y puede seleccionar las siguientes acciones: Beneficio 1, Beneficio 2, algún otro beneficio, antigüedad Sisbén, tipos de población, universidades, Carreras, Ciudad y salir.</p> <ul style="list-style-type: none"> - La actividad beneficio 1: ejecuta el caso de uso registrar beneficio 1. - La actividad beneficio 2: ejecuta el caso de uso registrar beneficio 2. - La actividad algún otro beneficio: ejecuta el caso de uso registrar algún otro beneficio. - La actividad antigüedad Sisbén: ejecuta el caso de uso registrar antigüedad Sisbén. - La actividad tipos de población: ejecuta el caso de uso registrar tipos de población. - La actividad universidades: ejecuta el caso de uso registrar universidades. - La actividad carreras: ejecuta el caso de uso registrar carreras. - La actividad ciudad: ejecuta el caso de uso registrar ciudad. - La actividad salir sacara al usuario del sistema.
Subflujos	Ninguno.
Excepciones	Ninguno.

Caso de uso	Registrar funcionario.
Actores	Usuario, base de datos foest.
Tipo	Básico.
Propósito	Permite registrar usuarios para que puedan hacer uso del sistema.
Resumen	El usuario inicia este caso de uso. Ofrece funcionalidad para crear, modificar y eliminar el registro de usuario con el software de gestión de documentos archivo para el fondo para la educación superior Tocancipá (FOEST).

Precondiciones	Todas las actividades que se realicen requieren iniciar el caso de uso validar usuario.
Flujo principal	Se presenta cuando se ejecuta el caso de uso validar usuario, dependiendo de la opción que realice el usuario
Subflujos	<p>S1. Registro funcionario.</p> <p>Se ingresa al usuario a una pantalla crear funcionario (p32), la cual contiene los campos para realizar el registro, el usuario los debe llenar.</p> <p>Los campos son los siguiente: nombre, apellido, password, correo electrónico y una entrada de confirmación de password, para asegurar que la contraseña este bien ingresada.</p> <p>El usuario puede decidir si registrarse o salir.</p> <p>Si el usuario selecciona registrarse, el sistema genera un nuevo registro (e1, e2, e3, e4), continua con el subflujo administrar (s3).</p> <p>La actividad salir sacará al usuario del sistema y la información ingresada se perderá.</p> <p>S2. Obtener información de registro.</p> <p>El sistema obtiene de la base de datos el registro, continua con el subflujo administrar (s3).</p> <p>S3. Administrar.</p>

	<p>Presenta la pantalla funcionarios registrados (p33), con la información del registro del usuario, allí el usuario puede realizar las siguientes actividades. modificar, eliminar y salir.</p> <p>Si se selecciona la opción de “modificar” se ejecuta el subflujo actualizar registro (s4).</p> <p>Si se selecciona “eliminar” se ejecuta el subflujo eliminar registro(s5).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p> <p>S4. Modificar registro.</p> <p>Se actualiza el registro con la información guardada (e1, e3, e4), continua con el sufijo administrar (s3).</p> <p>S5. Eliminar registro.</p> <p>Se elimina el registro del usuario, continua con el subflujo registro usuario(s1).</p>
<p>Excepciones</p>	<p>E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.</p> <p>E2. Ya existe: ya existe un usuario con el mismo login, se solicita al usuario que cambie el login.</p> <p>E3. Login incorrecto: el login no es válido, se solicita al usuario que corrija el login.</p> <p>E4. Password incorrecto: el password no es válido, se solicita al usuario que corrija el password.</p>

Caso de uso	Registrar estudiantes.
Actores	Usuario, base de datos foest.
Tipo	Básico.
Propósito	Permite registrar los estudiantes que son beneficiarios del FOEST.
Resumen	El usuario inicia este caso de uso. Ofrece funcionalidad para crear, modificar y eliminar el registro de los estudiantes inscritos en el FOEST con el software de gestión de documentos archivo para el fondo para la educación superior Tocancipá (FOEST).
Precondiciones	Todas las actividades que se realicen requieren iniciar el caso de uso validar usuario.
Flujo principal	Se presenta cuando se ejecuta el caso de uso validar usuario, dependiendo de la opción que realice el usuario
Subflujos	<p>S1. Registro estudiante.</p> <p>Se ingresa al usuario a una pantalla registro estudiantes (p4), la cual contiene los campos para realizar el registro, el usuario los debe llenar.</p> <p>Los campos son los siguiente: N° carpeta, fecha de radicación, beneficio 1, beneficio 2, 1°vez, renovación, reintegro, primer apellido, segundo apellido, primer nombre, segundo nombre, tipo de documento, numero de documento Expedido en, edad, población especial, genero, teléfono 1; teléfono 2, dirección, correo electrónico, antigüedad Sisbén en años, puntaje del Sisbén, recibe algún otro beneficio, universidad sede nivel carrera, modalidad, semestre, valor del semestre, porcentaje a aplicar monto aprobado, otro beneficio, materias perdidas promedio 2018_1 y 2017, horas sociales cumplidas, dependencia donde realizo horas sociales, Horario, ciudad a la que viaja, días que viaja, valor por 2 recorridos valor semanal, valor mensual bruto , valor a pagar por mes, meses a girar y total semestre; también los datos del representante legal si es un menor de edad.</p>

	<p>El usuario puede decidir si registrar o salir.</p> <p>Si el usuario selecciona registrar, el sistema genera un nuevo registro (e1, e2, e3, e4), continua con el subflujo administrar (s2).</p> <p>La actividad salir sacará al usuario del sistema y la información ingresada se perderá.</p> <p>S2. Administrar.</p> <p>Presenta la pantalla estudiantes registrados (p3), con la información del registro de los estudiantes beneficiarios, allí el usuario puede realizar las siguientes actividades. modificar, eliminar y salir.</p> <p>Si se selecciona la opción de “modificar” se ejecuta el subflujo actualizar registro (s3).</p> <p>Si se selecciona “eliminar” se ejecuta el subflujo eliminar registro(s4).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p> <p>S3. Modificar registro.</p> <p>Se actualiza el registro con la información guardada (e1, e3, e4), continua con el sufijo administrar (s2).</p> <p>S4. Eliminar registro.</p> <p>Se elimina el registro del usuario, continua con el subflujo registro usuario(s2).</p>
Excepciones	<p>E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.</p>

	<p>E2. Ya existe: ya existe un usuario con el mismo login, se solicita al usuario que cambie el login.</p> <p>E3. Login incorrecto: el login no es válido, se solicita al usuario que corrija el login.</p> <p>E4. Password incorrecto: el password no es válido, se solicita al usuario que corrija el password.</p>
--	---

Caso de uso	Registrar beneficio 1.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	Permite al usuario registrar un beneficio nuevo en caso de que no exista el beneficio con el cual cuenta un estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.
Flujo principal	Se continua con el subflujo de beneficio 1 (s1) de lo contrario, si crea un nuevo beneficio, se continua con el subflujo nuevo beneficio (s2).
Subflujos	<p>S1. Beneficio 1.</p> <p>Se presenta la pantalla beneficio 1 (p6), en la cual aparece: nombre del beneficio.</p> <p>El usuario puede seleccionar las siguientes actividades: modificar, eliminar y salir.</p> <p>Si se selecciona “modificar”, se continua con el subflujo modificar beneficio (s3).</p> <p>Si se selecciona “eliminar”, se continua con el subflujo eliminar (s4).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p>

	<p>S3. modificar beneficio.</p> <p>Se actualiza el registro del beneficio con la información guardada (e1), continua con el sufijo beneficio (s1).</p> <p>S4. Eliminar.</p> <p>Se elimina el registro del beneficio, continua con el subflujo beneficio 1(s1).</p> <p>S2. Nuevo Beneficio 1.</p> <p>Se presenta una pantalla “registro beneficio 1” (p6). La pantalla incluye el siguiente espacio: nombre del beneficio.</p> <p>El usuario puede seleccionar las siguientes actividades: registrar y salir.</p> <p>Si se selecciona “registrar”, el sistema verifica la información (e1) y se continua con el subflujo beneficio 1 (s1).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p>
Excepciones	E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.

Caso de uso	Registrar beneficio 2.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.

Resumen	Permite al usuario registrar un beneficio nuevo en caso de que no exista el beneficio con el cual cuenta un estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.
Flujo principal	Se continua con el subflujo de beneficio 2 (s1) de lo contrario, si crea un nuevo beneficio, se continua con el subflujo nuevo beneficio (s2).
Subflujos	<p>S1. Beneficio 2.</p> <p>Se presenta la pantalla beneficio 2 (p9), en la cual aparece: nombre del beneficio. El usuario puede seleccionar las siguientes actividades: modificar, eliminar y salir. Si se selecciona “modificar”, se continua con el subflujo modificar beneficio (s3). Si se selecciona “eliminar”, se continua con el subflujo eliminar (s4). Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p> <p>S3. modificar beneficio.</p> <p>Se actualiza el registro del beneficio con la información guardada (e1), continua con el sufijo beneficio (s1).</p> <p>S4. Eliminar.</p> <p>Se elimina el registro del beneficio, continua con el subflujo beneficio 2(s1).</p>

	<p>S2. Nuevo Beneficio 2.</p> <p>Se presenta una pantalla “registro beneficio 2” (p10). La pantalla incluye el siguiente espacio: nombre del beneficio.</p> <p>El usuario puede seleccionar las siguientes actividades: registrar y salir.</p> <p>Si se selecciona “registrar”, el sistema verifica la información (e1) y se continua con el subflujo beneficio 2 (s1).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p>
Excepciones	E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.

Caso de uso	Registrar algún otro beneficio.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	Permite al usuario registrar un beneficio nuevo en caso de que no exista el beneficio con el cual cuenta un estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.
Flujo principal	Se continua con el subflujo de algún otro beneficio (s1) de lo contrario, si crea un nuevo beneficio, se continua con el subflujo nuevo beneficio (s2).
Subflujos	<p>S1. Algún otro Beneficio.</p> <p>Se presenta la pantalla algún otro beneficio (p12), en la cual aparece: nombre del beneficio.</p> <p>El usuario puede seleccionar las siguientes actividades: modificar, eliminar y salir.</p>

	<p>Si se selecciona “modificar”, se continua con el subflujo modificar beneficio (s3).</p> <p>Si se selecciona “eliminar”, se continua con el subflujo eliminar (s4).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p> <p>S3. modificar beneficio.</p> <p>Se actualiza el registro del beneficio con la información guardada (e1), continua con el sufijo algún otro beneficio (s1).</p> <p>S4. Eliminar.</p> <p>Se elimina el registro del beneficio, continua con el subflujo algún otro beneficio(s1).</p> <p>S2. Nuevo Algún otro beneficio.</p> <p>Se presenta una pantalla “registro algún otro beneficio” (p13). La pantalla incluye el siguiente espacio: nombre del beneficio.</p> <p>El usuario puede seleccionar las siguientes actividades: registrar y salir.</p> <p>Si se selecciona “registrar”, el sistema verifica la información (e1) y se continua con el subflujo algún otro beneficio (s1).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p>
Excepciones	<p>E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.</p>

Caso de uso	Registrar antigüedad Sisbén.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	Permite al usuario registrar una antigüedad nueva en caso de que no exista la antigüedad con el cual cuenta un estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.
Flujo principal	Se continua con el subflujo de antigüedad Sisbén (s1) de lo contrario, si crea una nueva antigüedad, se continua con el subflujo nueva antigüedad (s2).
Subflujos	<p>S1. Antigüedad Sisbén.</p> <p>Se presenta la pantalla antigüedad Sisbén (p15), en la cual aparece: el año de antigüedad.</p> <p>El usuario puede seleccionar las siguientes actividades: modificar, eliminar y salir.</p> <p>Si se selecciona “modificar”, se continua con el subflujo modificar antigüedad (s3).</p> <p>Si se selecciona “eliminar”, se continua con el subflujo eliminar (s4).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p> <p>S3. modificar antigüedad.</p> <p>Se actualiza el registro de la antigüedad con la información guardada (e1), continua con el sufijo antigüedad Sisbén (s1).</p>

	<p>S4. Eliminar.</p> <p>Se elimina el registro de la antigüedad, continua con el subflujo antigüedad Sisbén(s1).</p> <p>S2. Nueva antigüedad.</p> <p>Se presenta una pantalla “registro antigüedad Sisbén” (p16). La pantalla incluye el siguiente espacio: año.</p> <p>El usuario puede seleccionar las siguientes actividades: registrar y salir.</p> <p>Si se selecciona “registrar”, el sistema verifica la información (e1) y se continua con el subflujo antigüedad Sisbén (s1).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p>
Excepciones	E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.

Caso de uso	Registrar tipos de población.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	Permite al usuario registrar diferentes tipos de población en caso de que no exista el tipo de población al cual pertenece el estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.
Flujo principal	Se continua con el subflujo de tipo de población (s1) de lo contrario, si crea una nueva población, se continua con el subflujo nueva población (s2).

Subflujos**S1. Tipo de población.**

Se presenta la pantalla tipos de población (p18), en la cual aparece: el nombre de la población.

El usuario puede seleccionar las siguientes actividades: modificar, eliminar y salir.

Si se selecciona “modificar”, se continua con el subflujo modificar población (s3).

Si se selecciona “eliminar”, se continua con el subflujo eliminar (s4).

Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.

S3. modificar población.

Se actualiza el registro de la población con la información guardada (e1), continua con el sufijo tipo de población (s1).

S4. Eliminar.

Se elimina el registro de la población, continua con el subflujo tipo de población (s1).

S2. Nueva población.

Se presenta una pantalla “registro tipos de población” (p19). La pantalla incluye el siguiente espacio: el nombre de la población.

El usuario puede seleccionar las siguientes actividades: registrar y salir.

	<p>Si se selecciona “registrar”, el sistema verifica la información (e1) y se continua con el subflujo tipo de población (s1).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p>
Excepciones	E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.

Caso de uso	Registrar universidades.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	Permite al usuario registrar diferentes universidades en caso de que no esté registrada la universidad a la cual pertenece el estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.
Flujo principal	Se continua con el subflujo de universidades (s1) de lo contrario, si crea una nueva universidad, se continua con el subflujo nueva universidad (s2).
Subflujos	<p>S1. Universidades.</p> <p>Se presenta la pantalla universidades (p21), en la cual aparece: el nombre de la universidad y el porcentaje a aplicar a esa universidad.</p> <p>El usuario puede seleccionar las siguientes actividades: modificar, eliminar y salir.</p> <p>Si se selecciona “modificar”, se continua con el subflujo modificar universidad (s3).</p> <p>Si se selecciona “eliminar”, se continua con el subflujo eliminar (s4).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p>

	<p>S3. modificar universidad.</p> <p>Se actualiza el registro de la universidad con la información guardada (e1), continua con el sufijo universidades (s1).</p> <p>S4. Eliminar.</p> <p>Se elimina el registro de la universidad, continua con el subflujo universidades(s1).</p> <p>S2. Nueva universidad.</p> <p>Se presenta una pantalla “registro universidades” (p22). La pantalla incluye el siguiente espacio: el nombre de la universidad y el porcentaje a aplicar a esa universidad.</p> <p>El usuario puede seleccionar las siguientes actividades: registrar y salir.</p> <p>Si se selecciona “registrar”, el sistema verifica la información (e1) y se continua con el subflujo universidades (s1).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p>
Excepciones	E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.

Caso de uso	Registrar carreras.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.

Resumen	Permite al usuario registrar diferentes carreras en caso de que no esté registrada la carrera a la cual pertenece el estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.
Flujo principal	Se continua con el subflujo de carreras (s1) de lo contrario, si crea una nueva carrera, se continua con el subflujo nueva población (s2).
Subflujos	<p>S1. Carreras.</p> <p>Se presenta la pantalla carreras (p24), en la cual aparece: el nombre de la carrera. El usuario puede seleccionar las siguientes actividades: modificar, eliminar y salir. Si se selecciona “modificar”, se continua con el subflujo modificar carrera (s3). Si se selecciona “eliminar”, se continua con el subflujo eliminar (s4). Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p> <p>S3. modificar carrera.</p> <p>Se actualiza el registro de la carrera con la información guardada (e1), continua con el sufijo carreras (s1).</p> <p>S4. Eliminar.</p> <p>Se elimina el registro de la carrera, continua con el subflujo carreras(s1).</p>

	<p>S2. Nueva carrera.</p> <p>Se presenta una pantalla “registro carreras” (p25). La pantalla incluye el siguiente espacio: el nombre de la carrera.</p> <p>El usuario puede seleccionar las siguientes actividades: registrar y salir.</p> <p>Si se selecciona “registrar”, el sistema verifica la información (e1) y se continua con el subflujo carreras (s1).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p>
Excepciones	E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.

Caso de uso	Registrar ciudad.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	Permite al usuario registrar diferentes ciudades en caso de que no esté registrada la ciudad a la cual viaja el estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.
Flujo principal	Se continua con el subflujo de ciudades (s1) de lo contrario, si crea un nuevo registro para la ciudad, se continua con el subflujo nueva ciudad (s2).
Subflujos	<p>S1. Ciudades.</p> <p>Se presenta la pantalla ciudad (p27), en la cual aparece: el nombre de la ciudad.</p> <p>El usuario puede seleccionar las siguientes actividades: modificar, eliminar y salir.</p>

	<p>Si se selecciona “modificar”, se continua con el subflujo modificar ciudad (s3).</p> <p>Si se selecciona “eliminar”, se continua con el subflujo eliminar (s4).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p> <p>S3. modificar ciudad.</p> <p>Se actualiza el registro de la ciudad con la información guardada (e1), continua con el sufijo Ciudades (s1).</p> <p>S4. Eliminar.</p> <p>Se elimina el registro de la ciudad, continua con el subflujo Ciudades(s1).</p> <p>S2. Nueva carrera.</p> <p>Se presenta una pantalla “registro ciudad” (p28). La pantalla incluye el siguiente espacio: el nombre de la ciudad.</p> <p>El usuario puede seleccionar las siguientes actividades: registrar y salir.</p> <p>Si se selecciona “registrar”, el sistema verifica la información (e1) y se continua con el subflujo Ciudad (s1).</p> <p>Si se selecciona “salir” sacara al usuario del sistema y si no se ha guardado la información esta se perderá.</p>
Excepciones	<p>E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.</p>

Caso de uso	Certificado Estudiante.
Actores	Usuario, base de datos foest
Tipo	Básico.
Propósito	Permite al usuario hacer estadísticas y certificados.
Resumen	El usuario inicia este caso de uso. Ofrece funcionalidad para crear, obtener, modificar y eliminar estadísticas y certificados.
Precondiciones	Se requiere que el usuario se halla validado correctamente.
Flujo principal	Se ejecuta el caso de uso validar usuario, dependiendo de la opción seleccionada por el usuario, se continuará con el respectivo subflujo.
Subflujos	<p>S1. Solicitar documentos.</p> <p>Se despliega la pantalla Documentos (p34).</p> <p>El usuario puede seleccionar las siguientes opciones: certificados y estadísticas.</p> <p>Si se selecciona “certificados”, se continua con subflujo certificados (s2).</p> <p>Si se selecciona “estadísticas”, se continua con subflujo estadísticas (s3).</p> <p>Si se selecciona “salir” sacara al usuario del sistema.</p> <p>S2. certificados.</p> <p>Se despliega la pantalla certificados (p31). La cual debe llenarse con la cedula o tarjeta de identidad del estudiante.</p> <p>El usuario puede seleccionar las siguientes opciones: solicitar y salir.</p> <p>Si se selecciona “solicitar”, el sistema imprimirá un PDF con el certificado (e1).</p> <p>Si se selecciona “salir” sacara al usuario del sistema.</p>

	<p>S3. estadísticas</p> <p>Se despliega la pantalla estadística (p30).</p> <p>El usuario puede seleccionar las siguientes opciones: solicitar y salir.</p> <p>Si se selecciona “solicitar”, el sistema dará a escoger entre diversas opciones.</p> <p>Si se selecciona “salir” sacara al usuario del sistema.</p>
Excepciones	<p>E1. El documento no pertenece a ningún estudiante inscrito o el espacio este vacío.</p>

6.6. Modelo de Dominio del Problema

6.6.1 Identificación de clases

Las clases identificadas son las siguientes:

algunotrobeneficio

antigu_sisben

beneficio1

beneficio2

carreras

ciudadviaja

funcionario

poblacion

universidad

usuariosfoest

6.6.2 identificación de asociaciones

Asociaciones identificadas para relacionar clases.

Un estudiante requiere ser registrado.

Un estudiante se encuentra inscrito en una universidad.

Un estudiante se encuentra inscrito a una carrera.

Un estudiante puede tener beneficios.

Un estudiante tiene carpeta.

Un estudiante tiene un horario.

Un estudiante tiene que cumplir con horas sociales.

Un funcionario registra estudiantes.

Un funcionario usa el sistema.

Un funcionario saca estadísticas.

Un funcionario saca certificados.

Asociaciones identificadas con roles para relacionar clases.

Un estudiante requiere ser registrado.

Un estudiante se encuentra inscrito en una universidad.

Un estudiante se encuentra inscrito a una carrera.

Un estudiante puede tener beneficios.

Un estudiante puede ingresar por primera vez.

Un estudiante puede ser renovado.

Un estudiante puede ser reintegrado.

Un estudiante tiene carpeta.

Un estudiante tiene un horario.

Un estudiante tiene que cumplir con horas sociales.

Un funcionario registra estudiantes.

Un funcionario usa el sistema.

Asociaciones identificadas con roles y multiplicidad para relacionar clases.

Un estudiante pertenece a una universidad.

Un estudiante tiene una carrera.

Un estudiante puede tener varios beneficios.

Un estudiante puede tener un pago por beneficio.

Un funcionario atiende a varios estudiantes.

Un funcionario saca varios certificados.

6.5.3 Identificación de atributos

clases	atributos
algunotrobeneficio	ID, Nombre.
antigu_sisben	ID, Año.
beneficio1	ID, Nombre.
beneficio2	ID, Nombre.
Carreras	ID, Nombre.
Ciudadviaja	ID, Nombre.
funcionario	ID, Nombre, apellido, correo, login, password.
Población	ID, poblacion.
universidad	ID, Nombre, porcentaje.
Usuariosfoest	ID, carpeta, radicación, beneficio1, benefici 2, 1vez, renovación, reintegro, primerapellido, segundoapellido, primernombre, segundonombre, tipodocumento,

	<p>numerodocumento, Expedido, edad, población, genero, teléfono1; teléfono2, dirección, correo electrónico, antigüedadSisbén, puntajeisbén, algunotrobeneficio, universidad, sede, nivel, carrera, modalidad, semestre, valorsemestre, porcentaje, otrobeneficio, materiasperdidas promedio2018_1, promedio2017, horassocialescumplidas, dependencia, Horario, ciudadviaja, díasviaja, valor2recorridos, valorsemanal, valormensual, valorapagar, mesesgirar y totalsemestre; también los datos del representante legal si es un menor de edad.</p>
--	---

6.6.4 Diccionario de clases

Algunotrobeneficio: son beneficios otorgados por otra entidad

antigu_sisben: es el año de antigüedad de residencia de los estudiantes o aspirantes.

beneficio1: son los beneficios otorgados por el FOEST en cuanto a educación.

beneficio2: son los beneficios otorgados por el FOEST en cuanto a transporte.

Carreras: son las carreras que ofrecen las diversas universidades.

Ciudadviaja: son las ciudades a las cuales viajan los estudiantes.

Funcionario: el funcionario es el usuario que maneja la base de datos y el software.

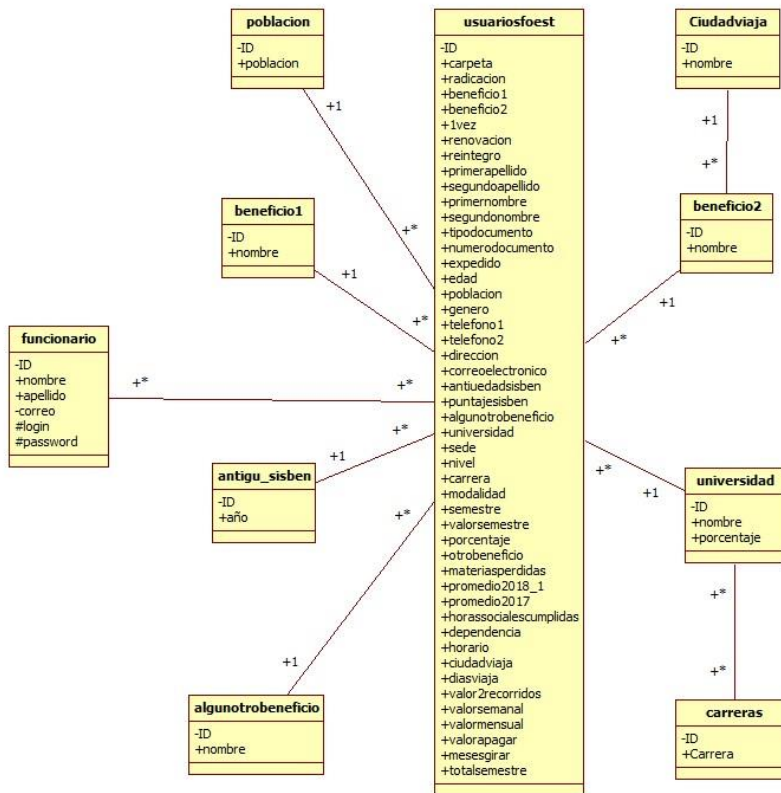
Poblacion: es el tipo de poblacion al cual pertenece el estudiante.

Universidad: son las instituciones las cuales tiene convenio con el FOEST, estas universidades deben estar registradas en el registro SNIES.

Usuariosfoest: son todos los estudiantes que están registrados en el fondo para la educación superior de Tocancipá

6.6.5 Identificación de Módulos

Modulo registro usuario



Capítulo 7 Modelo de Análisis

7.1. Arquitectura de Clases

La arquitectura utilizada en este proyecto es la arquitectura, Modelo, Vista y Control, en el cual el proyecto va a tener las siguientes dimensiones: Entidad (información), Borde (Interfaces) y Control (comportamiento).

7.1.1 Clases con estereotipos

La arquitectura para el modelo de análisis en este proyecto se basará en los siguientes tres estereotipos:

Estereotipo identidad: para los objetos que guardan información, estos objetos corresponden al dominio del problema

Estereotipo borde: para los objetos que implementan interfaces del sistema con el mundo externo, los cuales corresponden a todos los actores.

Estereotipo control: para los objetos que implementan el comportamiento y la lógica de los casos de uso.

7.1.2 Clases para casos de uso

Para cada caso de uso hay que identificar cada clase para este caso se identificaran las clases borde, entidad y control, de los casos de uso: validar funcionario, estudiantes foest, registrar funcionario, registrar estudiantes, registrar beneficio1, registrar beneficio2, registrar algún otro beneficio, registrar antigüedad Sisbén, registrar tipos de población, registrar universidades, registrar carreras, registrar ciudad y certificado estudiantes.

7.2. Identificación de Clases según Estereotipos

7.2.1 Borde

Validar funcionario: interactúan los actores usuario y base de datos foest, a través de las clases borde llamada InterfaceUsuario e InterfaceBasedeDatosRegistro, se utiliza una única pantalla: pantalla ingreso (P1) para la validación del usuario, cuya clase borde es PantallaIngreso.

Estudiantes foest: este caso de uso utiliza única mente la pantalla principal (p2), por lo cual solo incluye la clase borde PantallaPrincipal, y como también interactúa con el usuario a través de la clase borde InterfaceUsuario.

Registrar funcionario: interactúan los actores usuario y base de datos foest, a través de las clases borde llamada InterfaceUsuario e InterfaceBasedeDatosRegistro, también se deben incluir las clases bordes de las pantallas: pantalla crear funcionarios (P32) pantalla funcionarios registrados (P33), cuyas clases bordes son: PantallaRegistrarFuncionario y PantallaFuncionarioRegistrado.

Registrar estudiantes: interactúan los actores usuario y base de datos foest, a través de las clases borde llamada InterfaceUsuario e InterfaceBasedeDatosRegistro, también se deben incluir las clases bordes de las pantallas: pantalla registro estudiantes (P4) pantalla estudiantes registrados (P3), cuyas clases bordes son: PantallaRegistrarEstudiante y PantallaEstudianteRegistrado.

Registrar beneficio1: interactúan los actores usuario y base de datos foest, a través de las clases borde llamada InterfaceUsuario e InterfaceBasedeDatosRegistro, también se deben incluir las clases bordes de las pantallas: pantalla registro beneficio 1 (P7) pantalla beneficio 1 (P6), cuyas clases bordes son: PantallaRegistrarBeneficio1 y PantallaBeneficio1.

Registrar beneficio2: interactúan los actores usuario y base de datos foest, a través de las clases borde llamada InterfaceUsuario e InterfaceBasedeDatosRegistro, también se deben incluir las clases bordes de las pantallas: pantalla registro beneficio 2 (P10) pantalla beneficio 2 (P9), cuyas clases bordes son: PantallaRegistrarBeneficio2 y PantallaBeneficio2.

Registrar algún otro beneficio: interactúan los actores usuario y base de datos foest, a través de las clases borde llamada InterfaceUsuario e InterfaceBasedeDatosRegistro, también se deben incluir las

clases bordes de las pantallas: pantalla registro algún otro beneficio (P13) pantalla algún otro beneficio (P12), cuyas clases bordes son: PantallaRegistrarAlgúnOtroBeneficio y PantallaAlgúnOtroBeneficio.

Registrar antigüedad Sisbén: interactúan los actores usuario y base de datos foest, a través de las clases borde llamada InterfaceUsuario e InterfaceBasedeDatosRegistro, también se deben incluir las clases bordes de las pantallas: pantalla registro antigüedad Sisbén (P16) pantalla antigüedad Sisbén (P15), cuyas clases bordes son: PantallaRegistroAntigüedadSisben y PantallaAntigüedadSisben.

Registrar tipos de poblacion: interactúan los actores usuario y base de datos foest, a través de las clases borde llamada InterfaceUsuario e InterfaceBasedeDatosRegistro, también se deben incluir las clases bordes de las pantallas: pantalla registro tipos de poblacion (P19) pantalla tipos de poblacion (P18), cuyas clases bordes son: PantallaRegistroTiposPoblacion y PantallaTiposPoblacion.

Registrar universidades: interactúan los actores usuario y base de datos foest, a través de las clases borde llamada InterfaceUsuario e InterfaceBasedeDatosRegistro, también se deben incluir las clases bordes de las pantallas: pantalla registro universidades (P22) pantalla universidades (P21), cuyas clases bordes son: PantallaRegistroUniversidad y PantallaUniversidades.

Registrar carreras: interactúan los actores usuario y base de datos foest, a través de las clases borde llamada InterfaceUsuario e InterfaceBasedeDatosRegistro, también se deben incluir las clases bordes de las pantallas: pantalla registro carreras (P25) pantalla carreras (P24), cuyas clases bordes son: PantallaRegistroCarreras y PantallaCarreras.

Registrar ciudad: interactúan los actores usuario y base de datos foest, a través de las clases borde llamada InterfaceUsuario e InterfaceBasedeDatosRegistro, también se deben incluir las clases bordes de las pantallas: pantalla registro ciudad (P28) pantalla ciudad (P27), cuyas clases bordes son: PantallaRegistroCiudad y PantallaCiudad.

Certificado estudiante: interactúan los actores usuario y base de datos foest, a través de las clases borde llamada InterfaceUsuario e InterfaceBasedeDatosRegistro, también se deben incluir las clases bordes de las pantallas: pantallas estadísticas (P31) pantalla certificados (P30), cuyas clases bordes son: PantallaEstadísticas y PantallaCertificado.

Relación entre casos de uso, actores y clases borde.

Caso de uso	Actores	Clases borde
Validar funcionario	usuario y base de datos foest	InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaIngreso.
Estudiantes foest	usuario	PantallaPrincipal, InterfaceUsuario
Registrar funcionario	usuario y base de datos foest	InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistrarFuncionario, PantallaFuncionarioRegistrado.
Registrar estudiantes	usuario y base de datos foest	InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistrarEstudiante, PantallaEstudianteRegistrado.
Registrar beneficio1	usuario y base de datos foest	InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistrarBeneficio1, PantallaBeneficio1.
Registrar beneficio2	usuario y base de datos foest	InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistrarBeneficio2, PantallaBeneficio2.
Registrar algún otro beneficio	usuario y base de datos foest	InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistrarAlgúnOtroBeneficio,

		PantallaAlgunOtroBeneficio.
Registrar antigüedad Sisbén	usuario y base de datos foest	InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistroAntigüedadSisben, PantallaAntigüedadSisben.
Registrar tipos de poblacion	usuario y base de datos foest	InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistroTiposPoblacion, PantallaTiposPoblacion.
Registrar universidades	usuario y base de datos foest	InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistroUniversidad, PantallaUniversidades.
Registrar carreras	usuario y base de datos foest	InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistroCarreras, PantallaCarreras.
Registrar ciudad	usuario y base de datos foest	InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistroCiudad, PantallaCiudad.
Certificado estudiante	usuario y base de datos foest	InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaEstadísticas, PantallaCertificado.

7.2.2 Entidad

Validar funcionario: este caso de uso requiere validar la información guardada en el registro de usuario, cuya clase entidad es: RegistrarUsuarioFuncionario.

Estudiantes foest: este caso de uso administra las opciones del sistema, no requiere de ninguna clase entidad.

Registrar funcionario: este caso de uso requiere guardar información sobre el funcionario, cuya clase entidad es: RegistrarUsuarioFuncionario.

Registrar estudiantes: este caso de uso requiere guardar información sobre el estudiante, cuya clase entidad es: RegistrarUsuarioEstudiantes.

Registrar beneficio1: este caso de uso requiere guardar información sobre el beneficio 1, cuya clase entidad es: RegistrarBeneficio1.

Registrar beneficio2: este caso de uso requiere guardar información sobre el beneficio 2, cuya clase entidad es: RegistrarBeneficio2.

Registrar algún otro beneficio: este caso de uso requiere guardar información sobre otros beneficios, cuya clase entidad es: RegistrarAlgunOtroBeneficio.

Registrar antigüedad Sisbén: este caso de uso requiere guardar información sobre la antigüedad del Sisbén, cuya clase entidad es: RegistrarAntigüedadSisben.

Registrar tipos de población: este caso de uso requiere guardar información sobre el tipo de población, cuya clase entidad es: RegistrarTipoPoblacion.

Registrar universidades: este caso de uso requiere guardar información sobre las universidades, cuya clase entidad es: RegistrarUniversidades.

Registrar carreras: este caso de uso requiere guardar información sobre las carreras, cuya clase entidad es: RegistrarCarreras.

Registrar ciudad: este caso de uso requiere guardar información sobre las ciudades a donde viajan los estudiantes, cuya clase entidad es: RegistrarCiudad.

Certificado estudiante: este caso de uso requiere de la información relacionada con los estudiantes, cuyas clases entidades son: CertificadoEstudiante y Estadísticas.

Relación entre casos de uso, actores y clases entidad.

Caso de uso	Clases entidad
Validar funcionario	RegistrarUsuarioFuncionario.
Estudiantes foest	
Registrar funcionario	RegistrarUsuarioFuncionario.
Registrar estudiantes	RegistrarUsuarioEstudiantes.
Registrar beneficio1	RegistrarBeneficio1.
Registrar beneficio2	RegistrarBeneficio2.
Registrar algún otro beneficio	RegistrarAlgúnOtroBeneficio.
Registrar antigüedad Sisbén	RegistrarAntigüedadSisben.
Registrar tipos de poblacion	RegistrarTipoPoblacion.
Registrar universidades	RegistrarUniversidades
Registrar carreras	RegistrarCarreras
Registrar ciudad	RegistrarCiudad
Certificado estudiante	CertificadoEstudiante, Estadísticas.

7.2.3 Control

Validar funcionario: este caso de uso requiere de un controlador, para manejar la validación del usuario, cuya clase de control es: ManejadorFuncionario.

Estudiantes foest: este caso de uso requiere de un controlador para administrar las opciones del sistema, cuya clase de control es: ManejadorFoest.

Registrar funcionario: este caso de uso requiere un controlador para manejar la información sobre el funcionario, cuya clase de control es: ManejadorFuncionario.

Registrar estudiantes: este caso de uso requiere un controlador para manejar la información sobre los estudiantes, cuya clase de control es: ManejadorEstudiantes.

Registrar beneficio1: este caso de uso requiere un controlador para manejar la información sobre el beneficio 1, cuya clase de control es: ManejadorBeneficio1.

Registrar beneficio2: este caso de uso requiere un controlador para manejar la información sobre el beneficio 2, cuya clase de control es: ManejadorBeneficio2.

Registrar algún otro beneficio: este caso de uso requiere un controlador para manejar la información sobre otros beneficios, cuya clase de control es: ManejadorAlgunOtroBeneficio.

Registrar antigüedad Sisbén: este caso de uso requiere un controlador para manejar la información sobre la antigüedad del Sisbén, cuya clase de control es: ManejadorAntigüedadSisben.

Registrar tipos de poblacion: este caso de uso requiere un controlador para manejar la información sobre el tipo de poblacion, cuya clase de control es: ManejadorTipoPoblacion.

Registrar universidades: este caso de uso requiere un controlador para manejar la información sobre las universidades, cuya clase de control es: ManejadorUniversidades.

Registrar carreras: este caso de uso requiere un controlador para manejar la información sobre las carreras, cuya clase de control es: ManejadorCarreras.

Registrar ciudad: este caso de uso requiere un controlador para manejar la información sobre las ciudades a donde viajan los estudiantes, cuya clase de control es: ManejadorCiudad.

Certificado estudiante: este caso de administrar todo lo relacionado con los certificados y estadísticas, cuyas clases de control son: ManejadorCertificadoEstudiante y ManejadorEstadísticas.

Es importante incluir un controlador principal, el cual administrara los aspectos generales del sistema cuya clase de control es: ManejadorPrincipal.

Caso de uso	Clases control
Validar funcionario	ManejadorFuncionario.
Estudiantes foest	ManejadorFoest.
Registrar funcionario	ManejadorFuncionario.
Registrar estudiantes	ManejadorEstudiantes.
Registrar beneficio1	ManejadorBeneficio1.
Registrar beneficio2	ManejadorBeneficio2.
Registrar algún otro beneficio	ManejadorAlgunOtroBeneficio.
Registrar antigüedad Sisbén	ManejadorAntiguedadSisben.
Registrar tipos de poblacion	ManejadorTipoPoblacion.
Registrar universidades	ManejadorUniversidades.
Registrar carreras	ManejadorCarreras.
Registrar ciudad	ManejadorCiudad.
Certificado estudiante	ManejadorCertificadoEstudiante, ManejadorEstadísticas.
	ManejadorPrincipal.

7.3. Clases según Casos de Uso

7.3.1 Validar funcionario

El caso de uso validar funcionario maneja las siguientes clases:

Clases de control: ManejadorFuncionario.

Clases borde: InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaIngreso.

Clase entidad: RegistrarUsuarioFuncionario.

Y el controlador principal: ManejadorPrincipal.

7.3.2 Estudiantes foest

El caso de uso Estudiantes foest maneja las siguientes clases:

Clases de control: ManejadorFoest.

Clases borde: PantallaPrincipal, InterfaceUsuario

7.3.3 Registrar funcionario

El caso de uso registrar funcionario maneja las siguientes clases:

Clases de control: ManejadorFuncionario.

Clases borde: InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistrarFuncionario, PantallaFuncionarioRegistrado.

Clase entidad: RegistrarUsuarioFuncionario.

7.3.4 Registrar estudiantes

El caso de uso registrar estudiantes maneja las siguientes clases:

Clases de control: ManejadorEstudiantes.

Clases borde: InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistrarEstudiante, PantallaEstudianteRegistrado.

Clase entidad: RegistrarUsuarioEstudiantes.

7.3.5 Registrar beneficio1

El caso de uso registrar beneficio1 maneja las siguientes clases:

Clases de control: ManejadorBeneficio1.

Clases borde: InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistrarBeneficio1, PantallaBeneficio1.

Clase entidad: RegistrarBeneficio1.

7.3.6 Registrar beneficio2

El caso de uso registrar beneficio2 maneja las siguientes clases:

Clases de control: ManejadorBeneficio2.

Clases borde: InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistrarBeneficio2, PantallaBeneficio2.

Clase entidad: RegistrarBeneficio2.

7.3.7 Registrar algún otro beneficio

El caso de uso registrar algún otro beneficio maneja las siguientes clases:

Clases de control: ManejadorAlgunOtroBeneficio.

Clases borde: InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistrarAlgunOtroBeneficio, PantallaAlgunOtroBeneficio.

Clase entidad: RegistrarAlgunOtroBeneficio.

7.3.8 Registrar antigüedad Sisbén

El caso de uso registrar antigüedad Sisbén maneja las siguientes clases:

Clases de control: ManejadorAntigüedadSisben.

Clases borde: InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistroAntigüedadSisben, PantallaAntigüedadSisben.

Clase entidad: RegistrarAntigüedadSisben.

7.3.9 Registrar tipos de poblacion

El caso de uso registrar antigüedad Sisbén maneja las siguientes clases:

Clases de control: ManejadorTipoPoblacion.

Clases borde: InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistroTiposPoblacion, PantallaTiposPoblacion.

Clase entidad: RegistrarTipoPoblacion.

7.3.10 Registrar universidades

El caso de uso registrar universidades maneja las siguientes clases:

Clases de control: ManejadorUniversidades.

Clases borde: InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistroUniversidad, PantallaUniversidades.

Clase entidad: RegistrarUniversidades.

7.3.11 Registrar carreras

El caso de uso registrar carreras maneja las siguientes clases:

Clases de control: ManejadorCarreras.

Clases borde: InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistroCarreras, PantallaCarreras.

Clase entidad: RegistrarCarreras.

7.3.12 Registrar ciudad

El caso de uso registrar ciudad maneja las siguientes clases:

Clases de control: ManejadorCiudad.

Clases borde: InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaRegistroCiudad, PantallaCiudad.

Clase entidad: RegistrarCiudad.

7.3.13 Certificado estudiante

El caso de uso Certificado estudiante maneja las siguientes clases:

Clases de control: ManejadorCertificadoEstudiante, ManejadorEstadísticas.

Clases borde: InterfaceUsuario, InterfaceBasedeDatosRegistro, PantallaEstadísticas, PantallaCertificado.

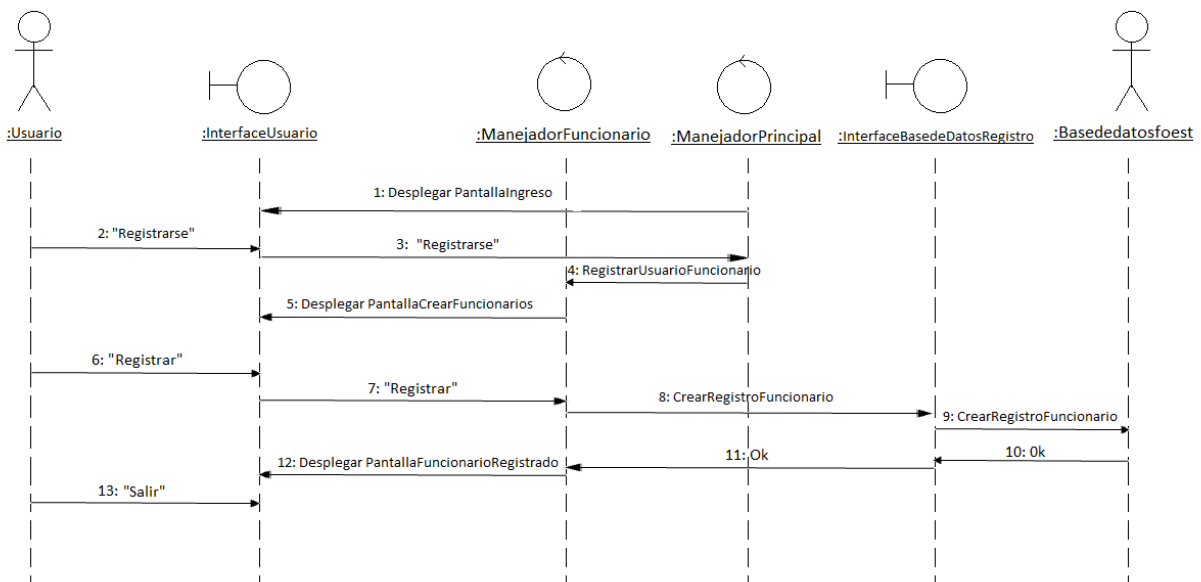
Clase entidad: CertificadoEstudiante, Estadísticas.

7.4. Diagramas de Secuencia

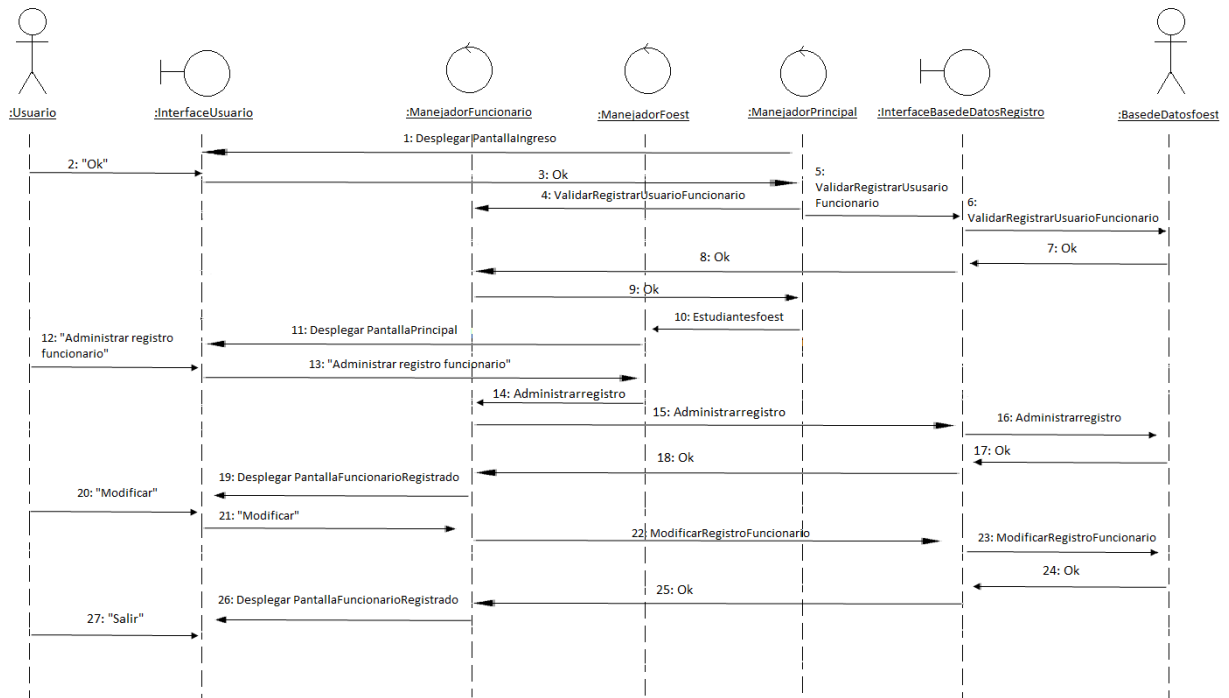
7.4.1 Registrar funcionario

Para este caso de uso, existen diversas secuencias que pueden ser instanciadas por un usuario, entre ellas encontramos:

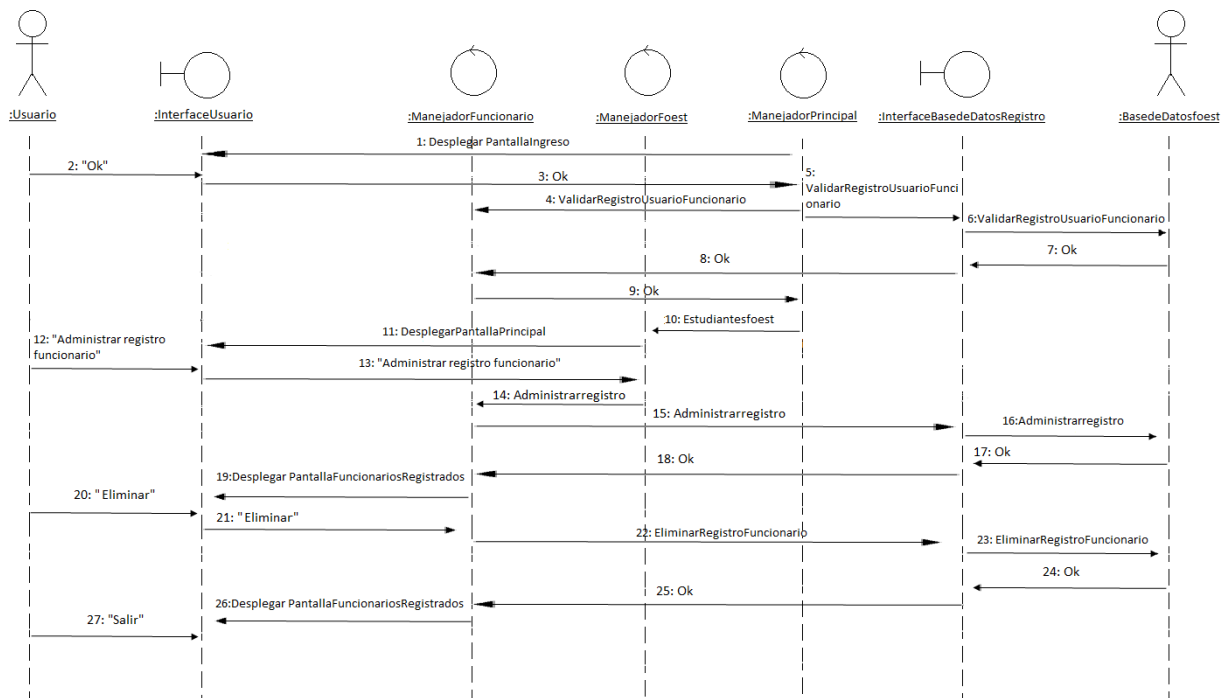
Registro funcionario:



Actualizar registro:



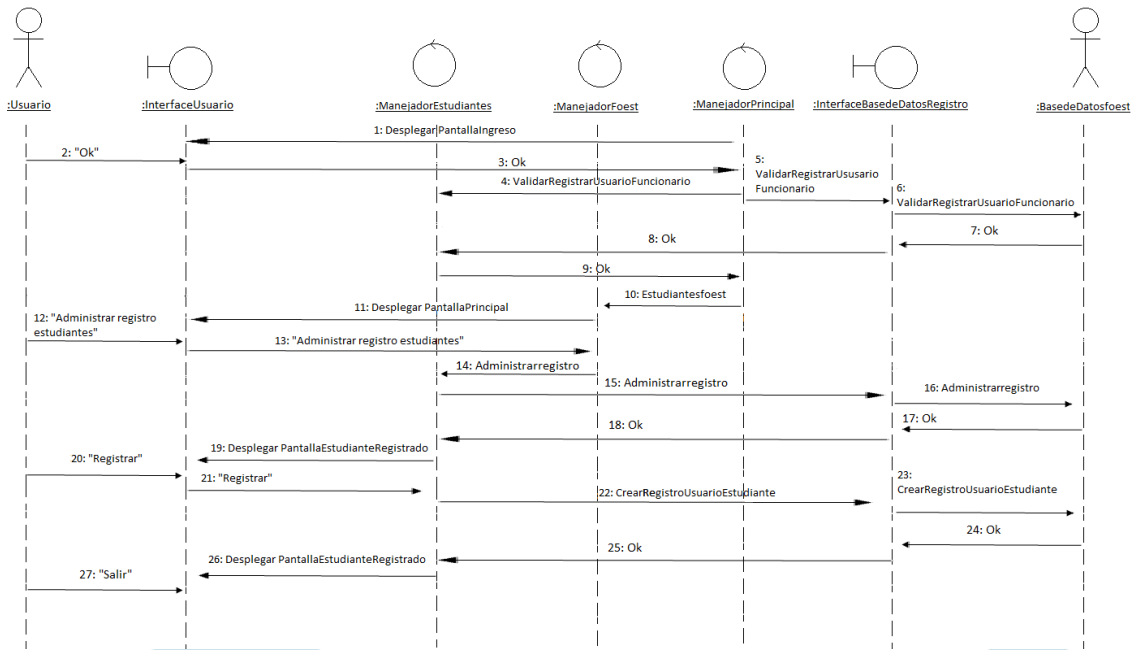
Eliminar registro:



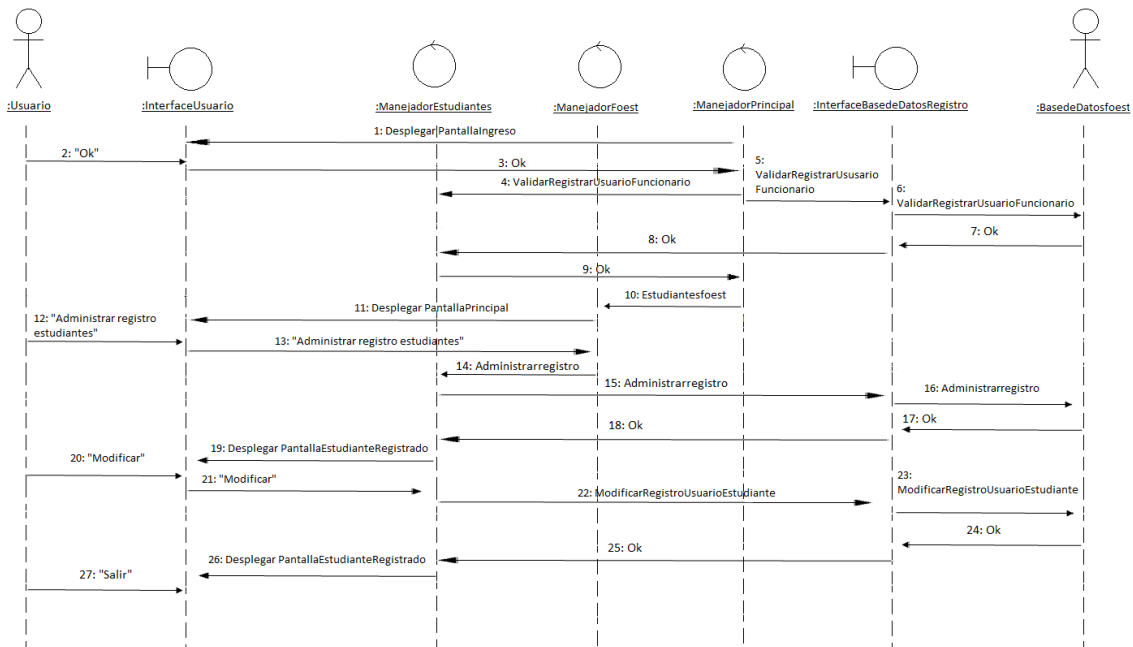
7.4.2 registrar estudiante

Para este caso de uso, existen diversas secuencias que pueden ser instanciadas por un usuario, entre ellas encontramos:

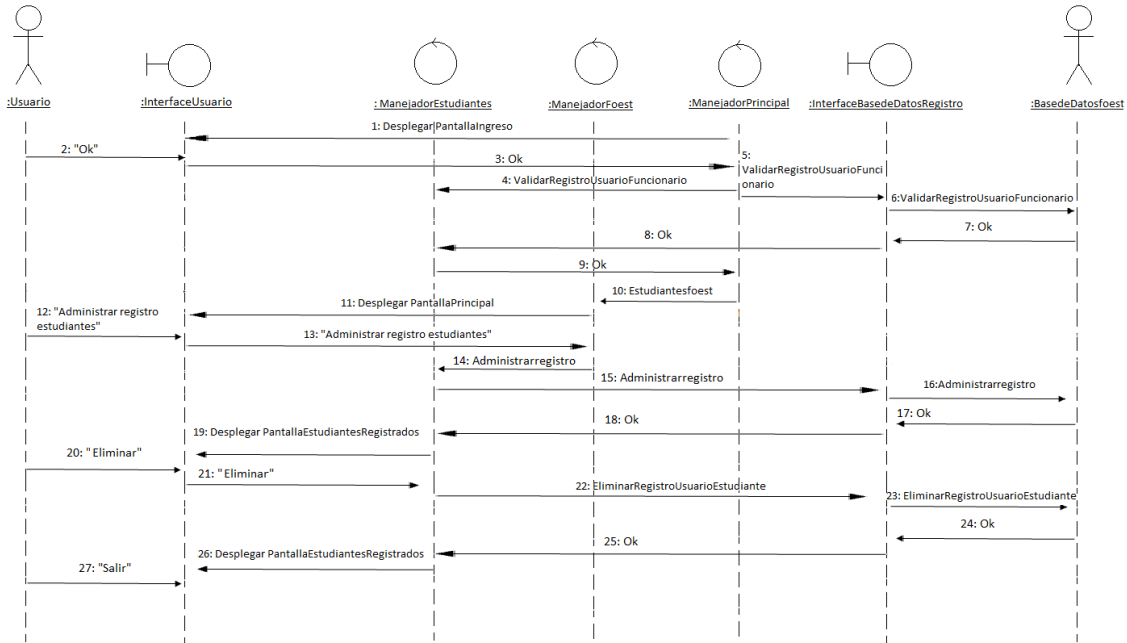
Registro estudiante:



Actualizar estudiante:



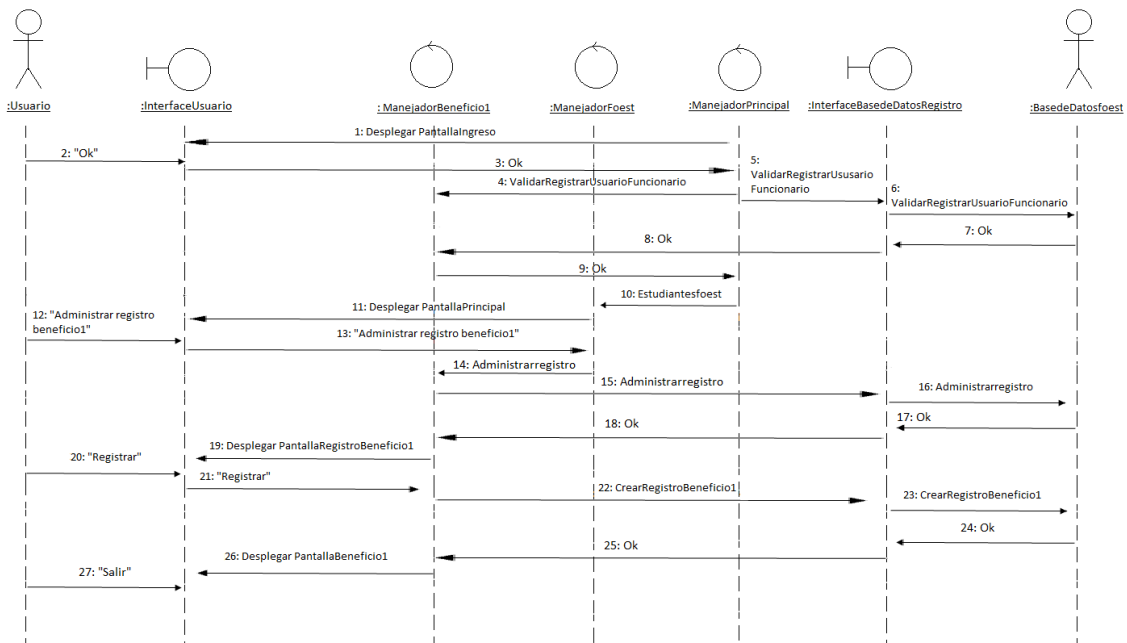
Eliminar estudiante



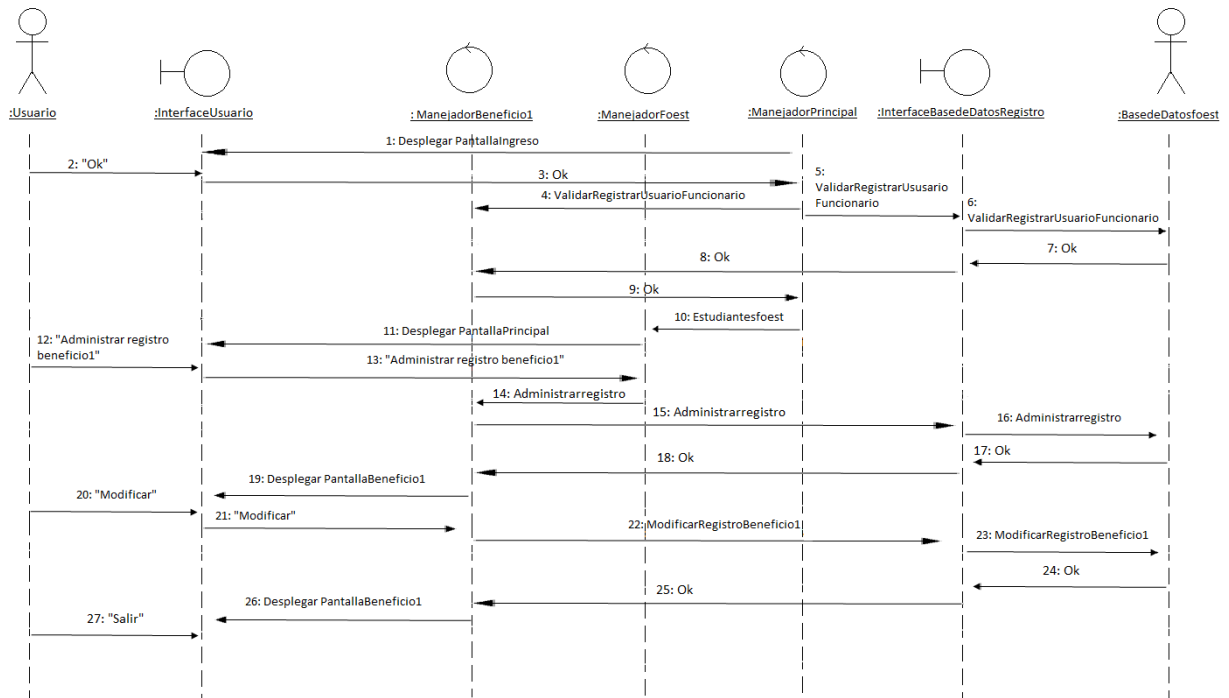
7.4.3 registrar Beneficio1

Para este caso de uso, existen diversas secuencias que pueden ser instanciadas por un usuario, entre ellas encontramos:

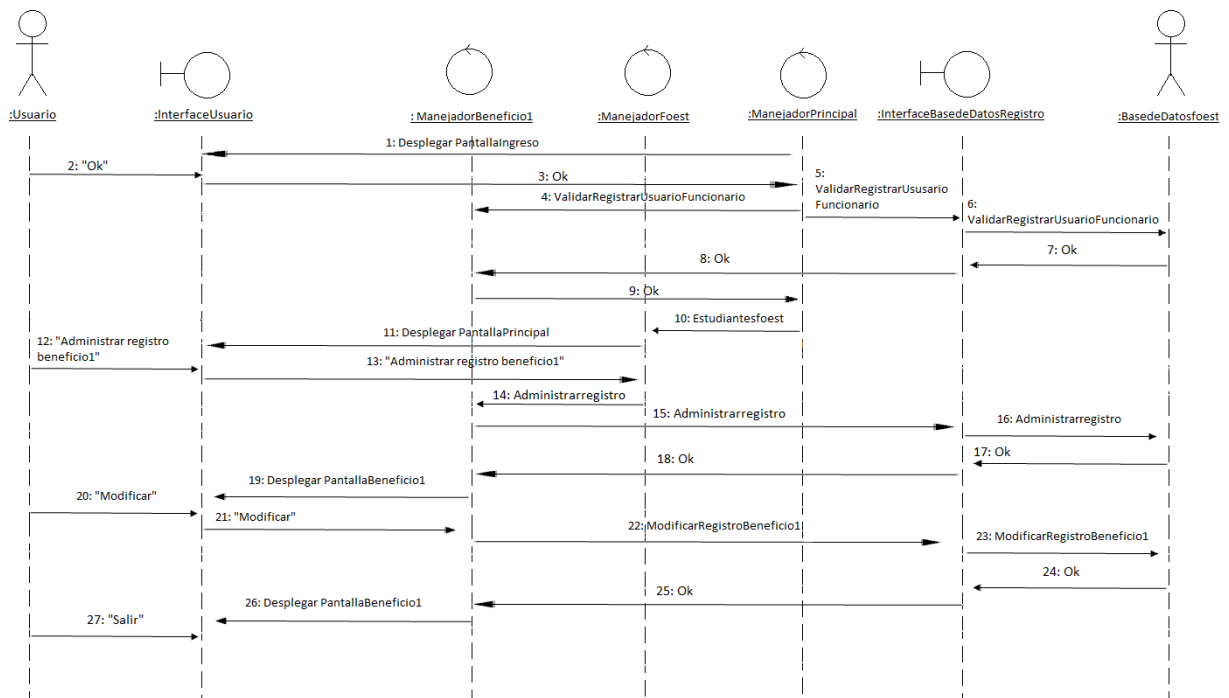
Registro Beneficio1



Actualizar beneficio1



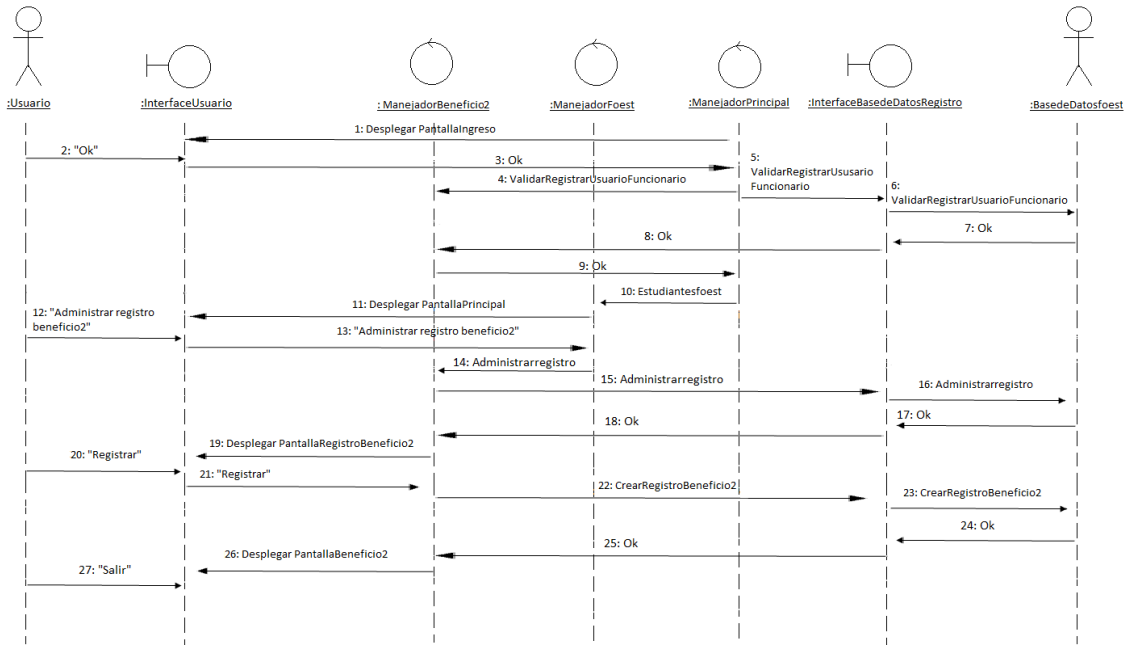
Eliminar beneficio1



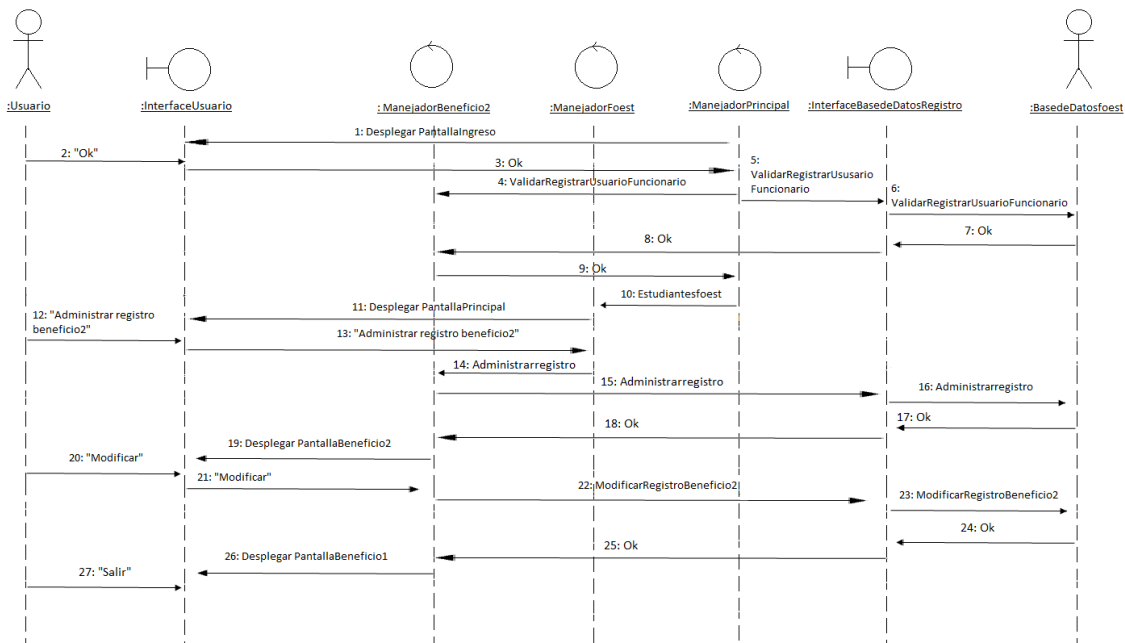
7.4.4 registrar Beneficio2

Para este caso de uso, existen diversas secuencias que pueden ser instanciadas por un usuario, entre ellas encontramos:

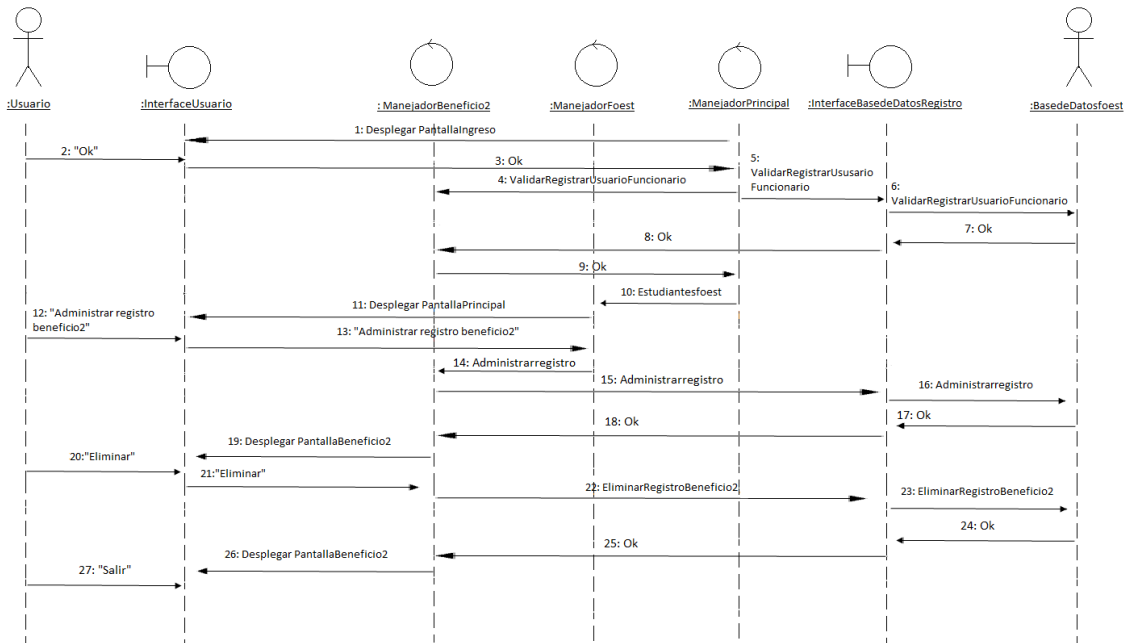
Registro Beneficio2



Actualizar Beneficio2



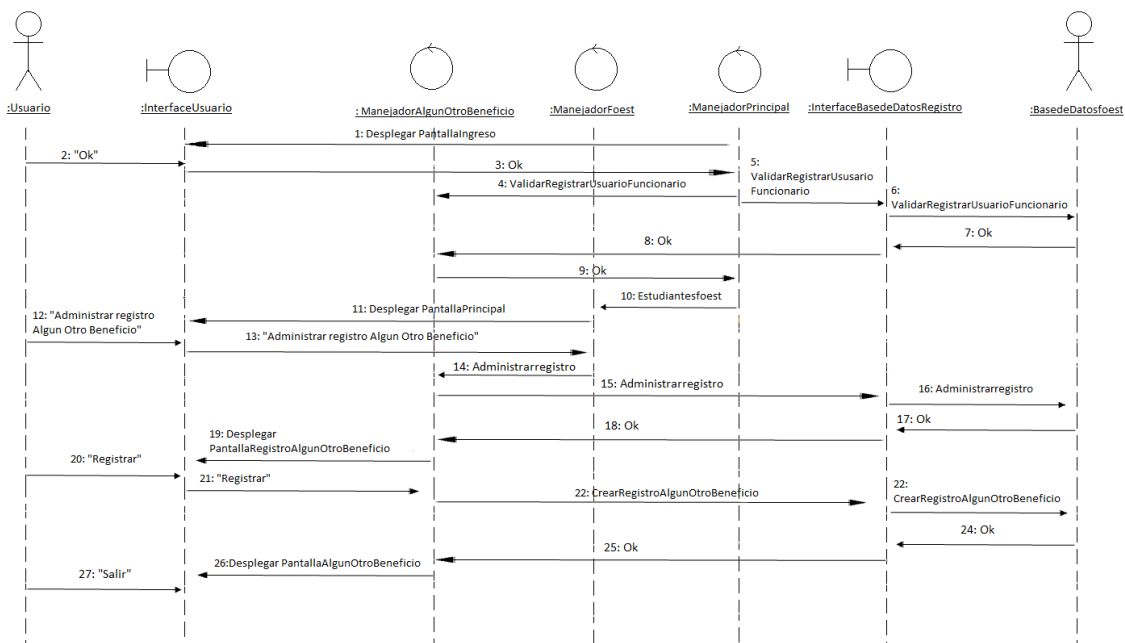
Eliminar beneficio2



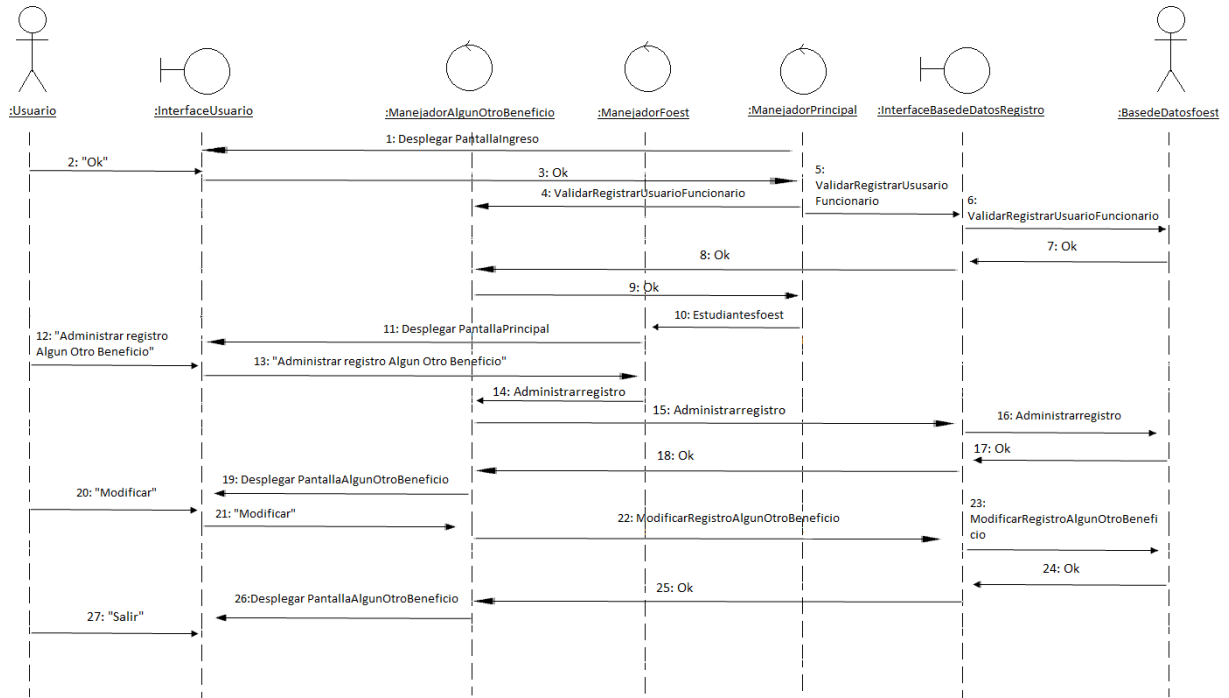
7.4.5 registrar Algunotrobeneficio

Para este caso de uso, existen diversas secuencias que pueden ser instanciadas por un usuario, entre ellas encontramos:

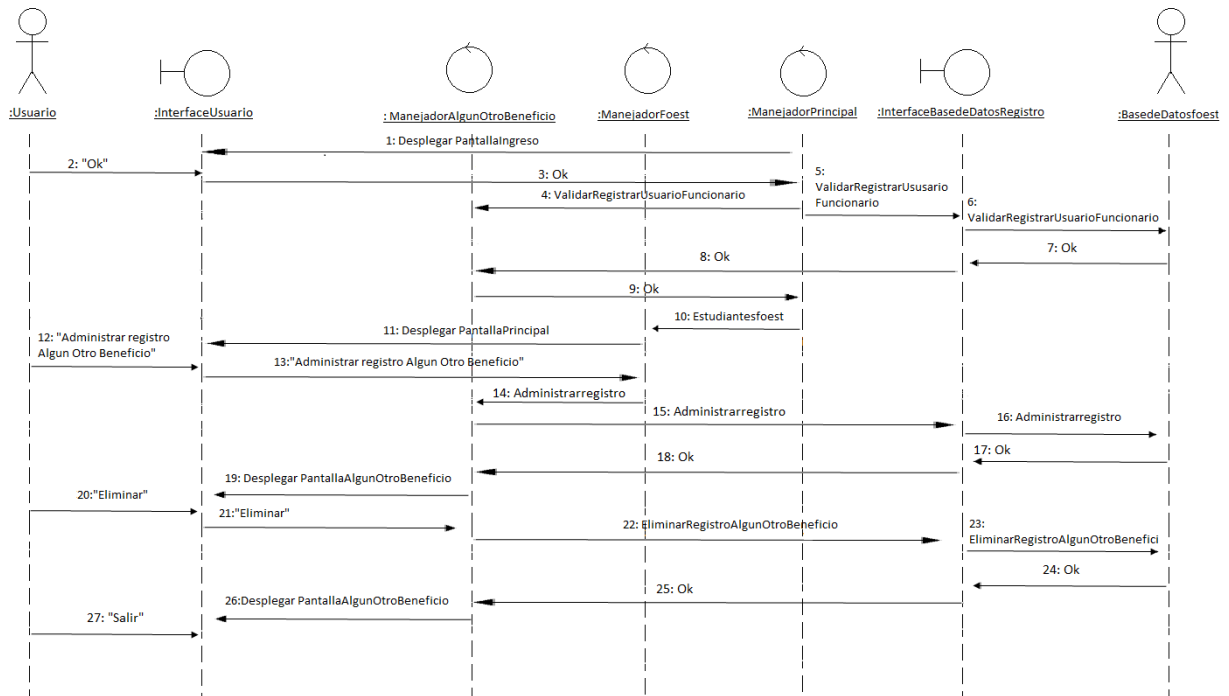
Registro Algunotrobeneficio



Actualizar Algunotrobeficio



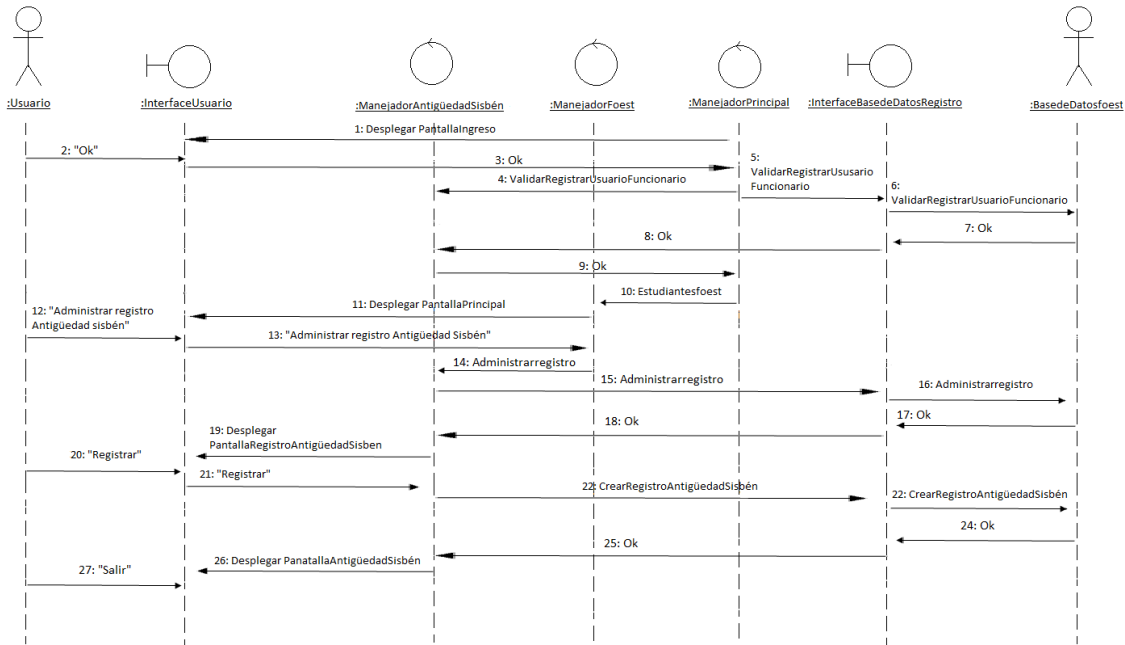
Eliminar Algunotrobeficio



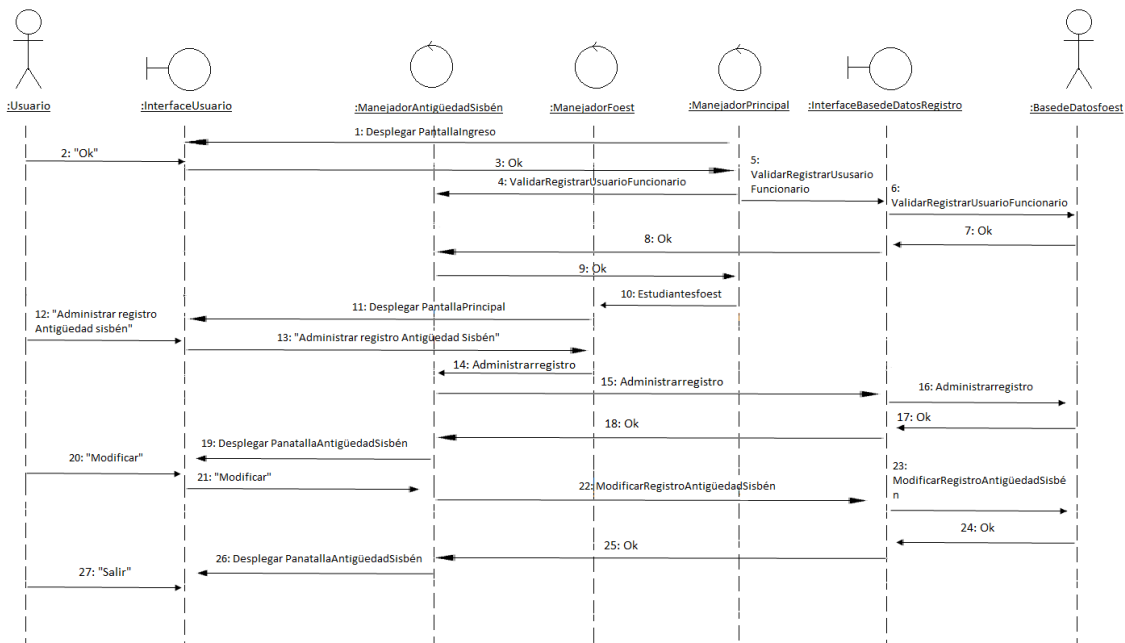
7.4.6 registrar AntigüedadSisbén

Para este caso de uso, existen diversas secuencias que pueden ser instanciadas por un usuario, entre ellas encontramos:

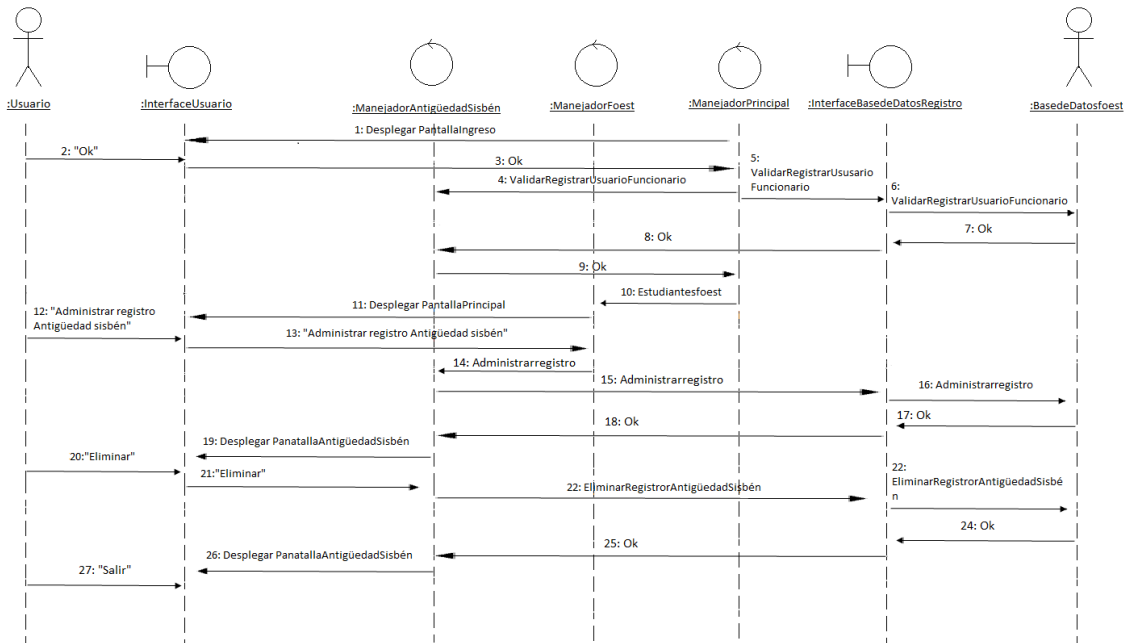
Registro AntigüedadSisbén



Actualizar AntigüedadSisbén



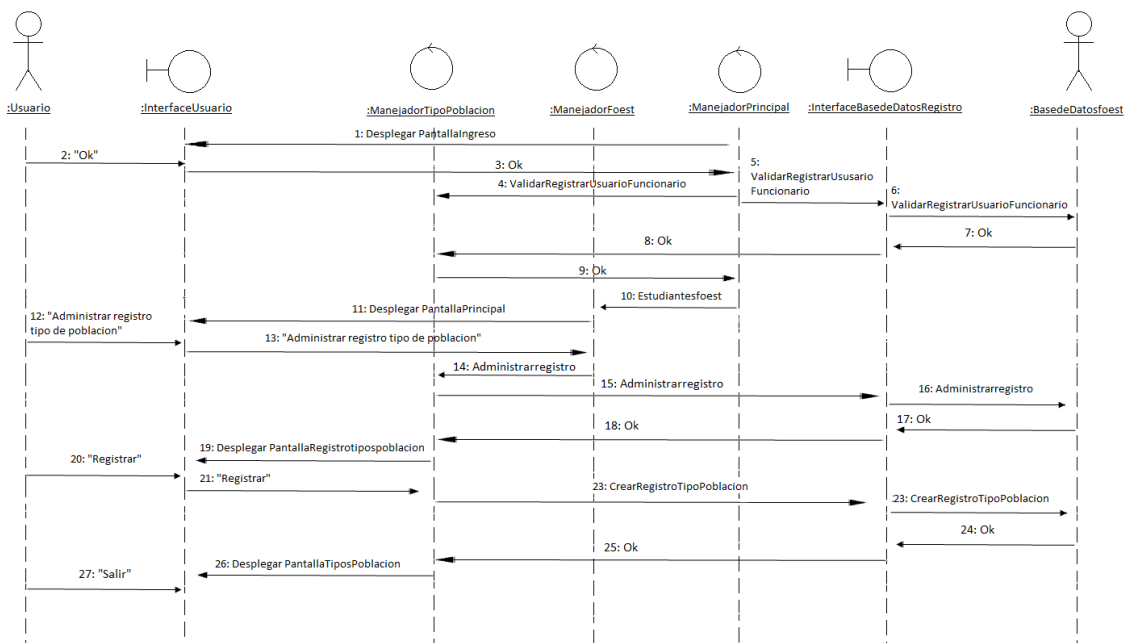
Eliminar antigüedadSisbén



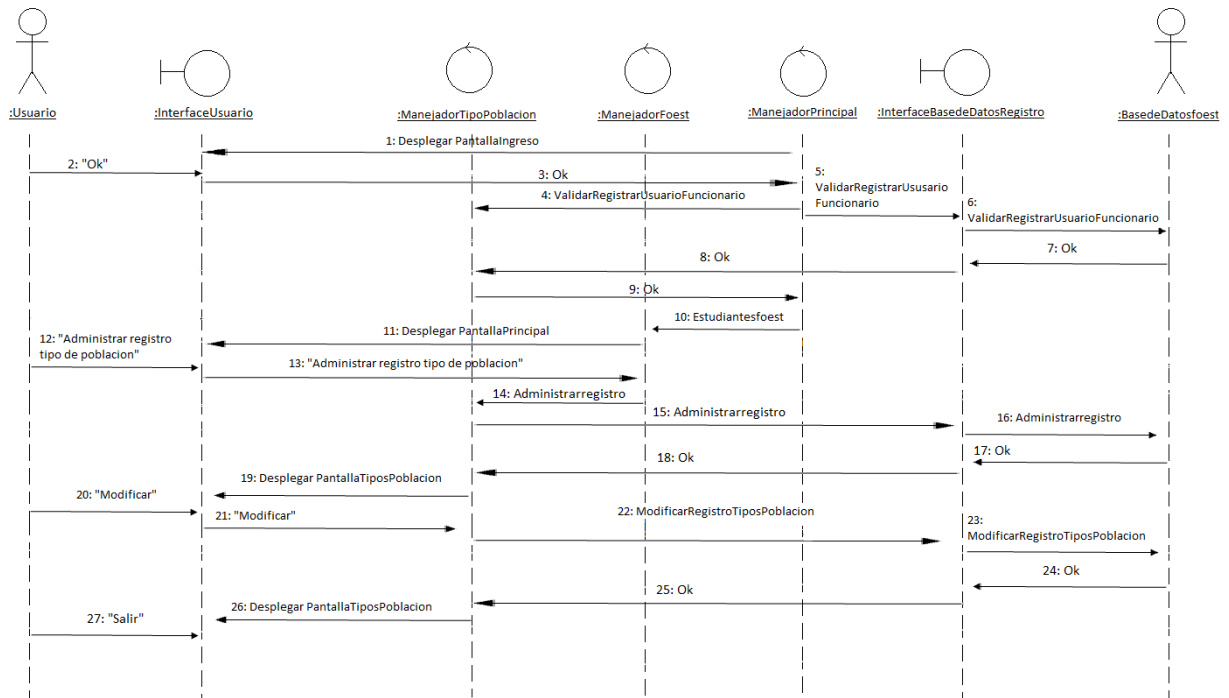
7.4.7 registrar tipos de poblacion

Para este caso de uso, existen diversas secuencias que pueden ser instanciadas por un usuario, entre ellas encontramos:

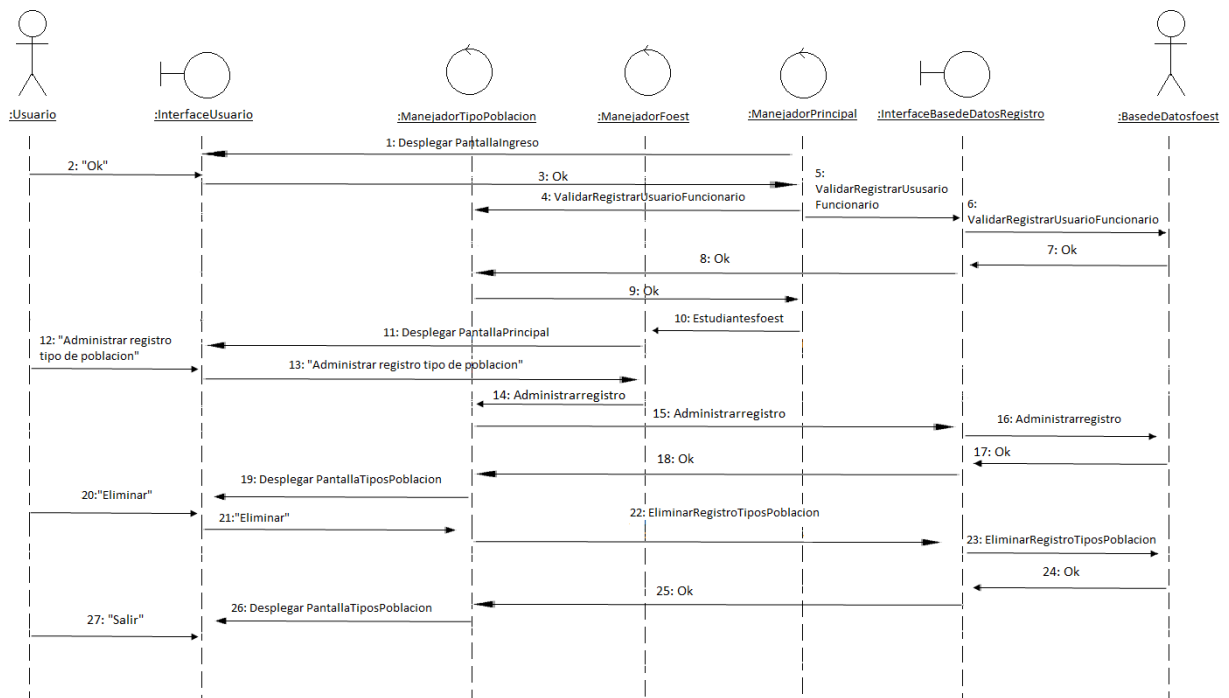
Registro tipos de poblacion



Actualización tipos de poblacion



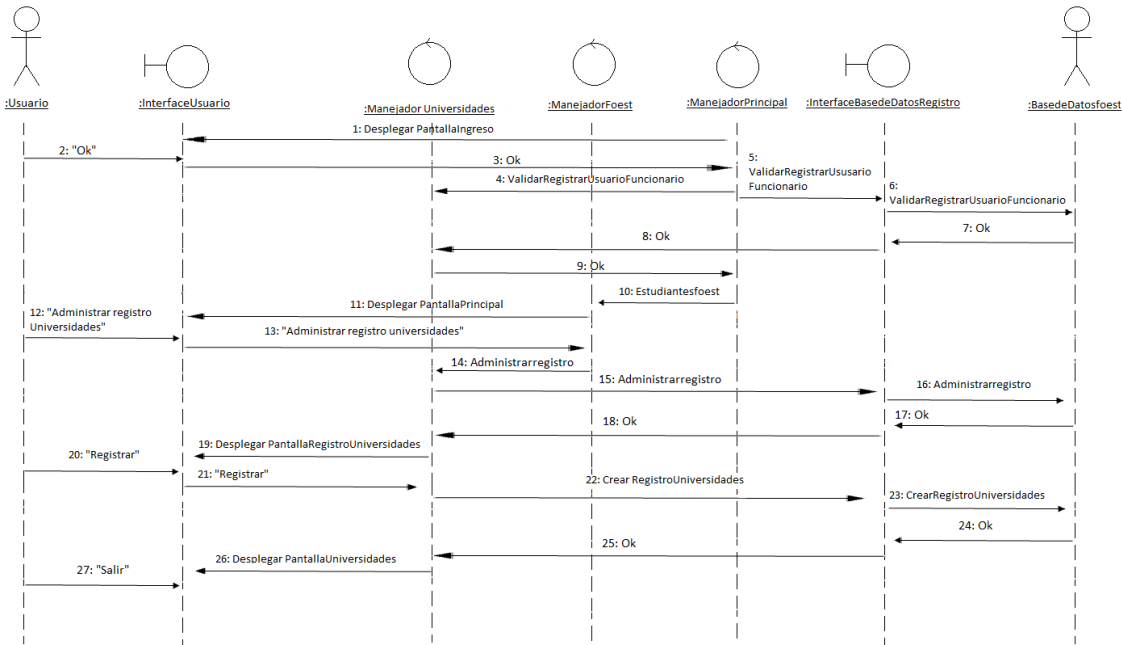
Eliminación tipos de poblacion



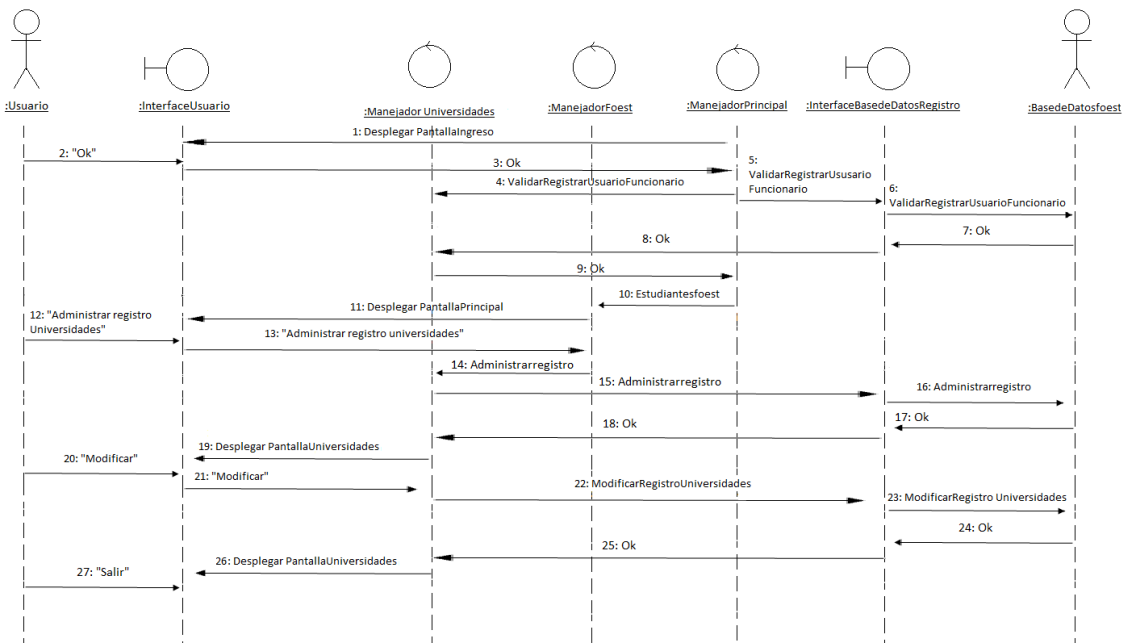
7.4.8 Registrar universidades

Para este caso de uso, existen diversas secuencias que pueden ser instanciadas por un usuario, entre ellas encontramos:

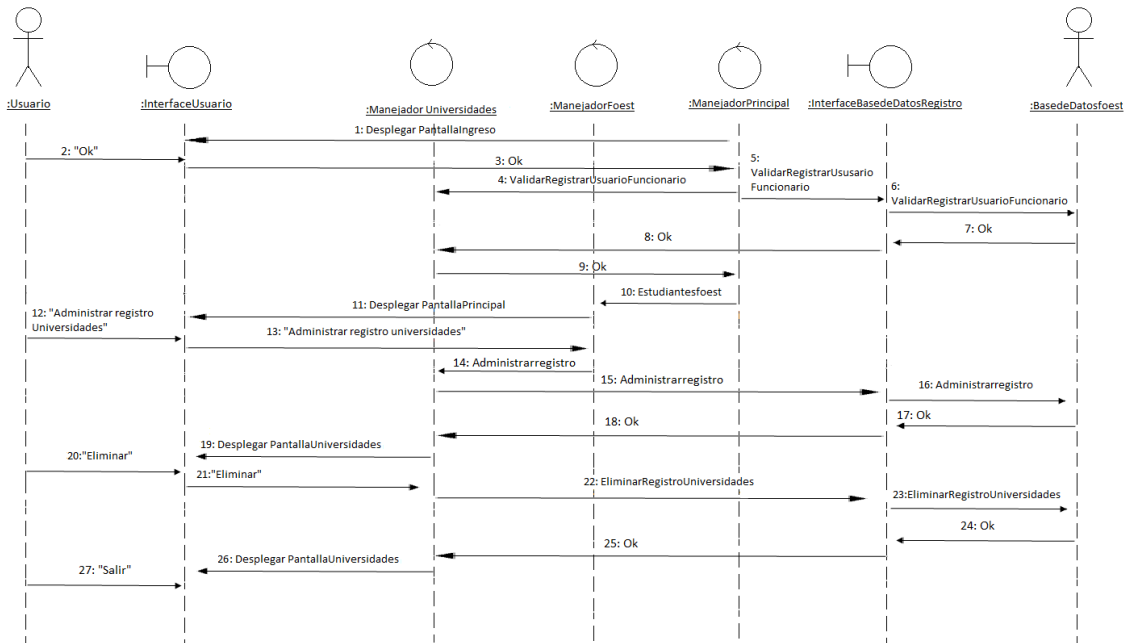
Registro universidades



Actualizar universidades



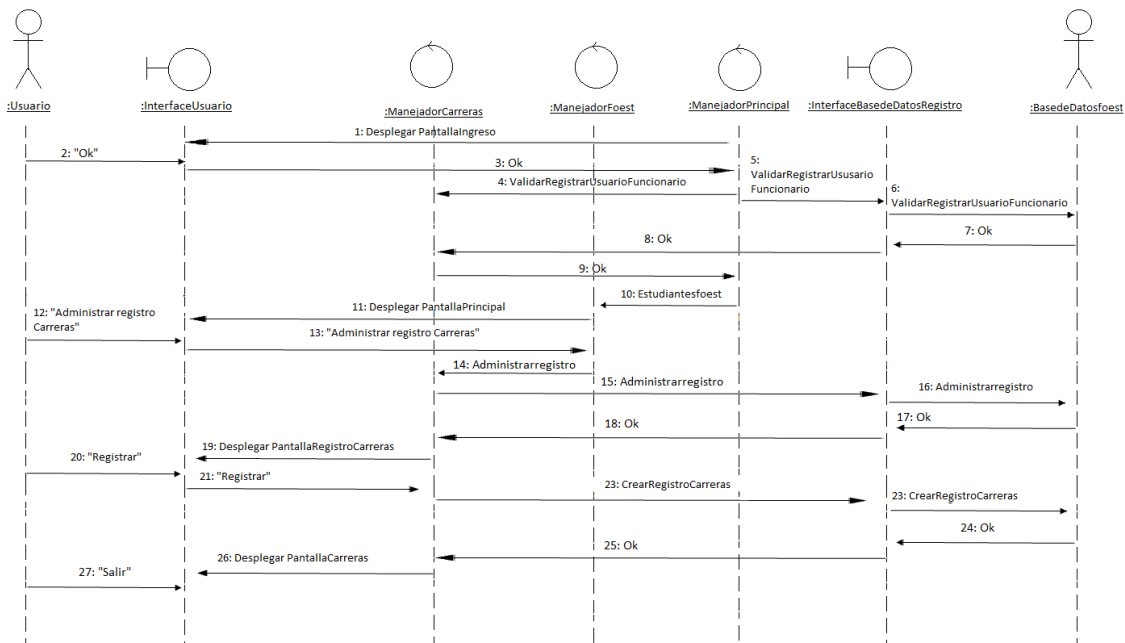
Eliminar universidades



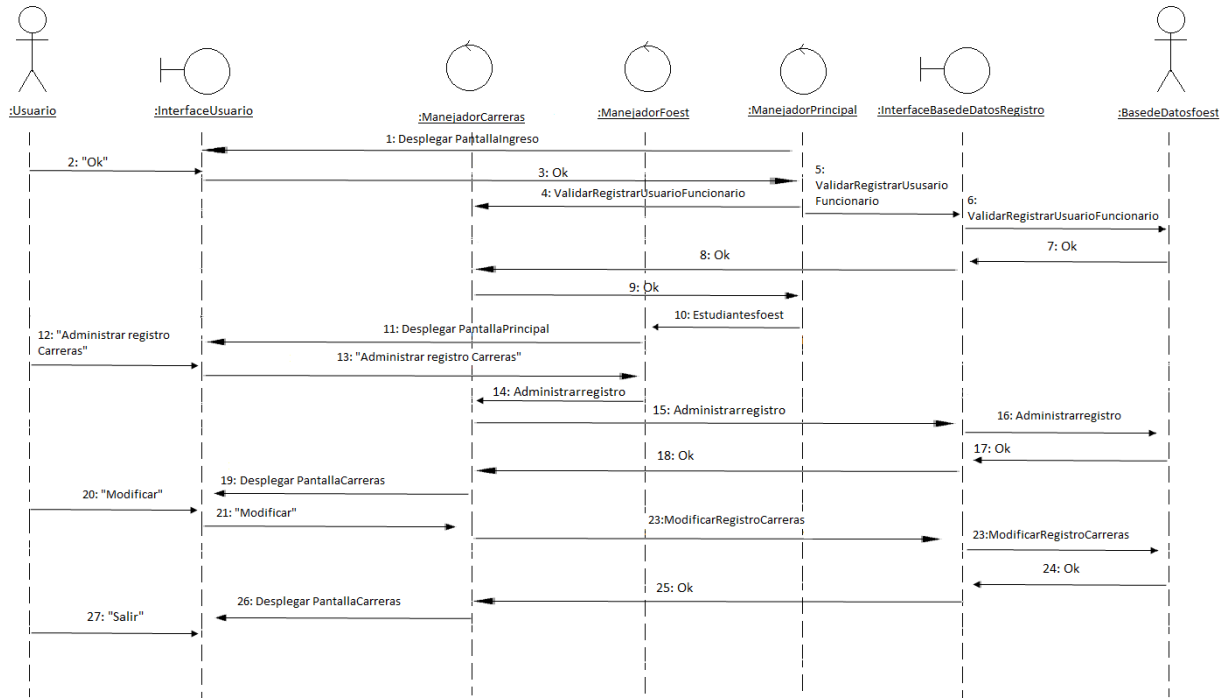
7.4.9 Registrar carreras

Para este caso de uso, existen diversas secuencias que pueden ser instanciadas por un usuario, entre ellas encontramos:

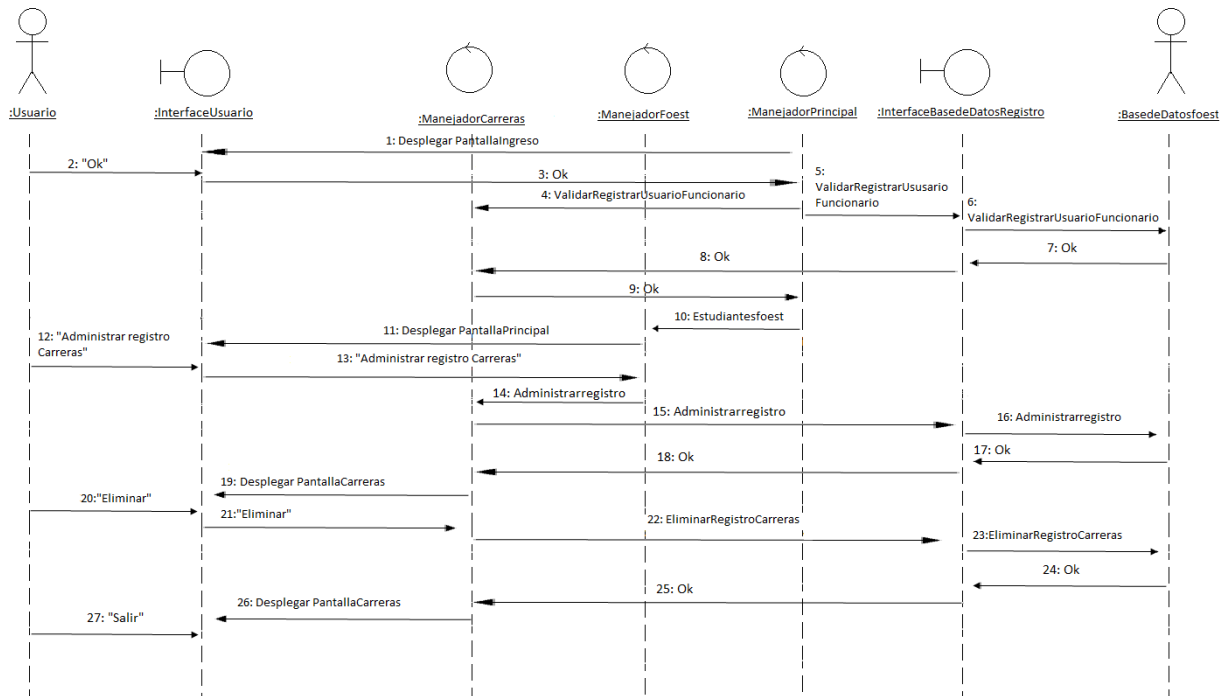
Registro carreras



Actualizar carreras



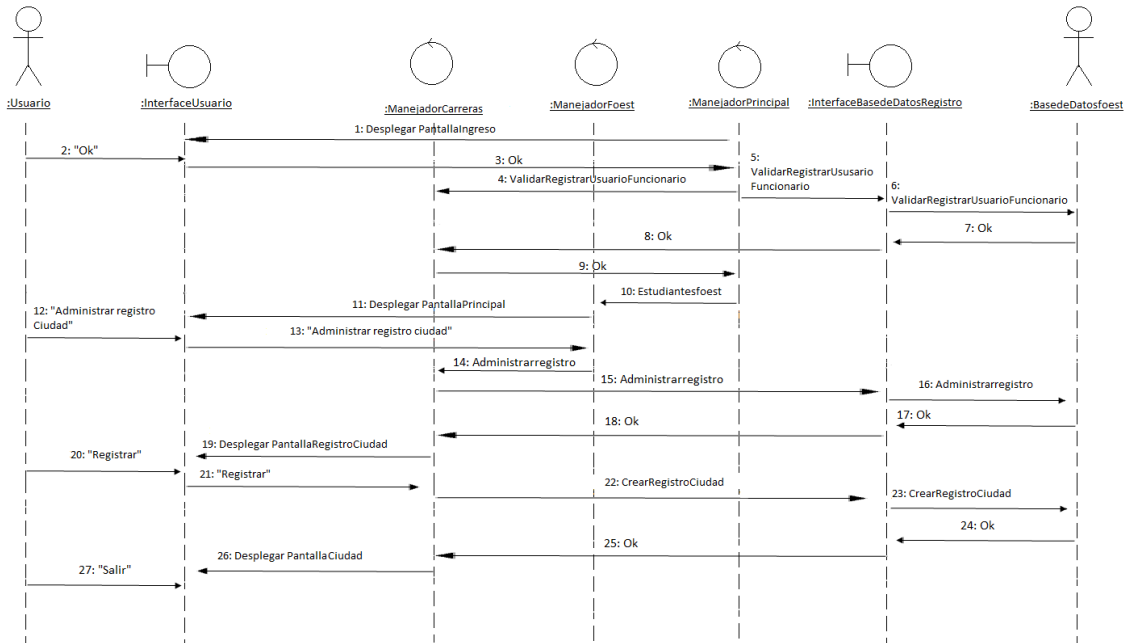
Eliminar carreras



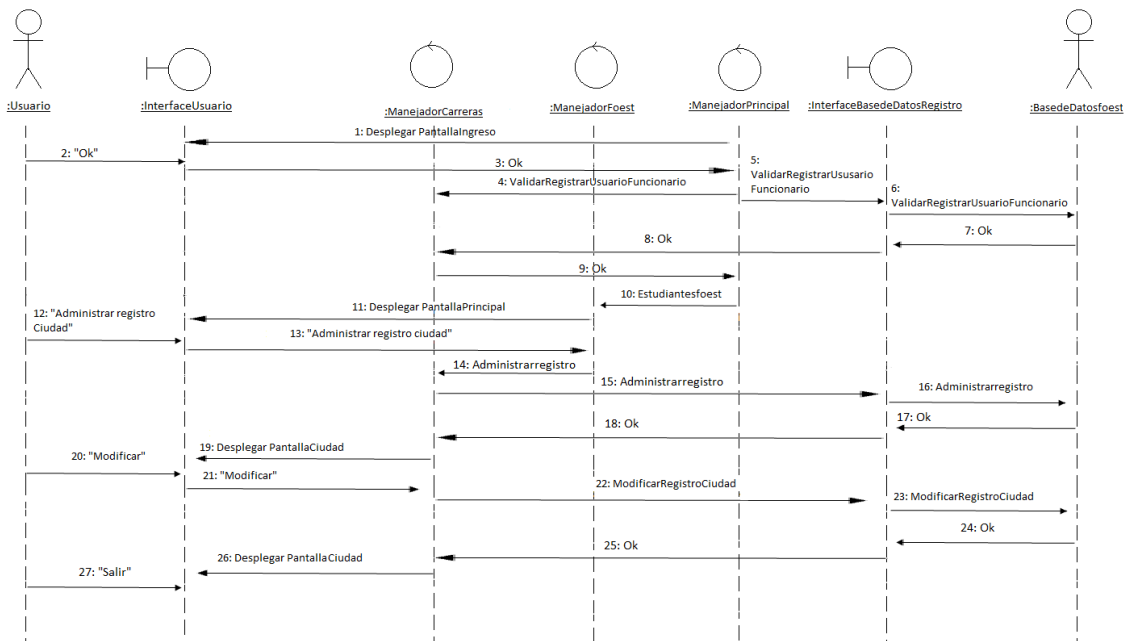
7.4.10 Registrar ciudad

Para este caso de uso, existen diversas secuencias que pueden ser instanciadas por un usuario, entre ellas encontramos:

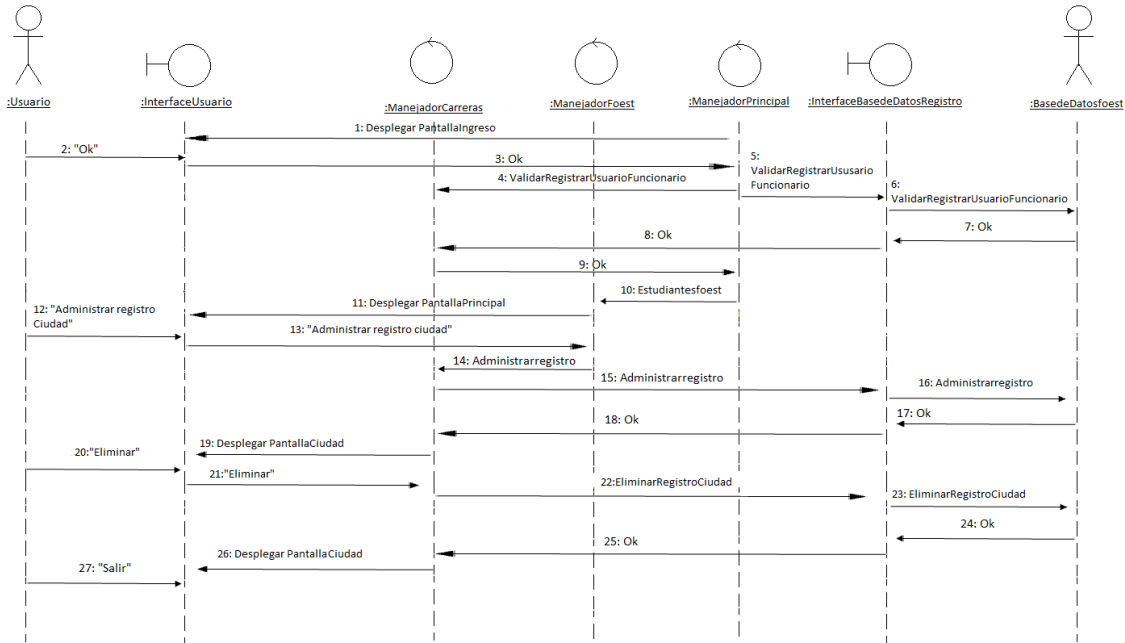
Registro ciudad



Actualizar ciudad



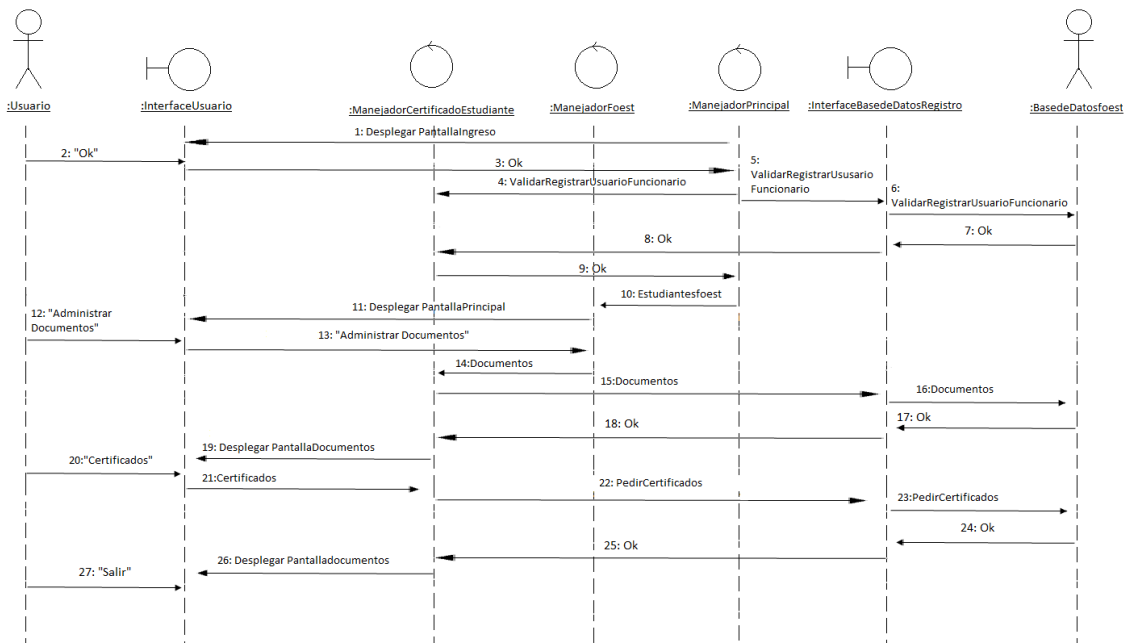
Eliminar ciudad



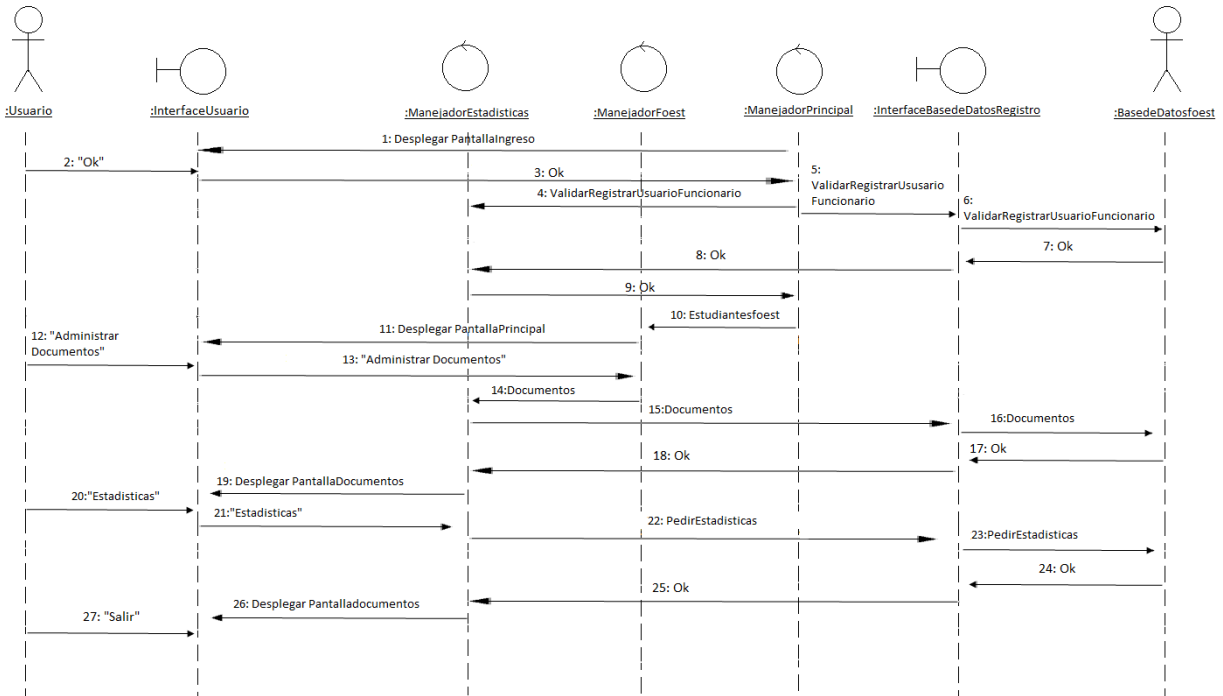
7.4.11 Certificado estudiante

Para este caso de uso, existen diversas secuencias que pueden ser instanciadas por un usuario, entre ellas encontramos:

Certificado estudiante



Estadísticas



7.5. Casos de Uso para el Sistema

7.5.1 Validar Funcionario

Caso de uso	Validar funcionario.
Actores	Usuario, base de datos foest.
Tipo	Inclusión.
Propósito	Valida a los usuarios funcionarios ya registrados para el uso del sistema.
Resumen	Este caso inicia cada vez que el usuario entra a su cuenta mediante un login y password, este lo verifica con el respectivo registro, para autorizar el ingreso al sistema.
Precondiciones	Se requiere que el <u>UsuarioFuncionario</u> se halla registrado antes.
Flujo principal	El <u>ManejadorPrincipal</u> solicita DesplegarPantallaIngreso a la <u>InterfaceUsuario</u> , la <u>InterfaceUsuario</u> despliega la <u>PantallaIngreso</u> (p1).

	<p>El usuario puede seleccionar las siguientes acciones: ingresar o salir.</p> <p>Si se selecciona “Ingresar”, se valida el registro del usuario mediante el uso del login y el password, insertados en la <u>PantallaIngreso</u>. La <u>PantallaIngreso</u> envía el evento “ingresar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “ingresar” al <u>ManejadorPrincipal</u>, el <u>ManejadorPrincipal</u> solicita ValidarRegistroUsuarioFuncionario a la <u>InterfaceBasedeDatosRegistro</u>. la <u>InterfaceBasedeDatosRegistro</u>, solicita ValidarRegistroUsuarioFuncionario a la <u>base de datos foest</u>, la <u>base de datos foest</u> valida al usuario y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorPrincipal</u>. Una vez validado el usuario (E1), el <u>ManejadorPrincipal</u> solicita Estudiantesfoest al <u>ManejadorFoest</u>. se continua con el caso de uso Estudiantes foest.</p> <p>Si se selecciona “salir”, a <u>PantallaIngreso</u> envía el evento “salir” a la <u>InterfaceUsuario</u>.</p>
Subflujos	Ninguno.
Excepciones	E1. No se realizó la validación el login y/o password incorrectos. Se solicita al usuario registrarse.

7.5.2 Estudiantes foest

Caso de uso	Estudiantes foest.
Actores	Usuario.
Tipo	Inclusión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	El usuario, tiene la capacidad de utilizar las opciones que ofrece el sistema
Precondiciones	Se requiere que el UsuarioFuncionario se halla validado correctamente antes.
Flujo principal	El <u>ManejadorPrincipal</u> solicita DesplegarPantallaPrincipal a la <u>InterfaceUsuario</u> , la <u>InterfaceUsuario</u> despliega la <u>PantallaPrincipal</u> (p2).

el usuario puede seleccionar las siguientes acciones: administrar registro funcionario, administrar registro estudiantes, administrar registro beneficio1, administrar registro beneficio2, administrar registro algún otro beneficio, administrar registro antigüedad Sisbén, administrar registro tipos de poblacion, administrar registro universidades, administrar registro carreras, administrar registro ciudad, administrar certificado estudiante y salir.

Si se selecciona “administrar registro usuario funcionario”, la PantallaPrincipal envía el evento “administrar registro” a la InterfaceUsuario, la InterfaceUsuario envía el evento “administrar registro” al ManejadorFoest, el ManejadorFoest solicita registrar usuario al ManejadorFuncionario, se continua con el caso de uso Registrar funcionario, subflujo administrar (s2).

Si se selecciona “administrar registro usuario estudiante”, la PantallaPrincipal envía el evento “administrar registro” a la InterfaceUsuario, la InterfaceUsuario envía el evento “administrar registro” al ManejadorFoest, el ManejadorFoest solicita registrar usuario al ManejadorEstudiantes, se continua con el caso de uso Registrar estudiantes, subflujo administrar (s2).

Si se selecciona “administrar registro Beneficio1”, la PantallaPrincipal envía el evento “administrar registro” a la InterfaceUsuario, la InterfaceUsuario envía el evento “administrar registro” al ManejadorFoest, el ManejadorFoest solicita registrar usuario al ManejadorBeneficio1, se continua con el caso de uso Registrar beneficio1, subflujo administrar (s2).

Si se selecciona “administrar registro Beneficio2”, la PantallaPrincipal envía el evento “administrar registro” a la InterfaceUsuario, la InterfaceUsuario envía el evento “administrar registro” al ManejadorFoest, el ManejadorFoest solicita registrar usuario al ManejadorBeneficio2, se continua con el caso de uso Registrar beneficio2, subflujo administrar (s2).

Si se selecciona “administrar registro algún otro Beneficio”, la PantallaPrincipal envía el evento “administrar registro” a la InterfaceUsuario, la InterfaceUsuario envía el evento “administrar registro” al ManejadorFoest, el ManejadorFoest solicita registrar usuario al ManejadorAlgunOtroBeneficio, se continua con el caso de uso Registrar algún otro beneficio, subflujo administrar (s2).

Si se selecciona “administrar registro antigüedad Sisbén”, la PantallaPrincipal envía el evento “administrar registro” a la InterfaceUsuario, la InterfaceUsuario envía el evento “administrar registro” al ManejadorFoest, el ManejadorFoest solicita registrar usuario al ManejadorAntigüedadSisben, se continua con el caso de uso Registrar antigüedad Sisbén, subflujo administrar (s2).

Si se selecciona “administrar registro tipos de poblacion”, la PantallaPrincipal envía el evento “administrar registro” a la InterfaceUsuario, la InterfaceUsuario envía el evento “administrar registro” al ManejadorFoest, el ManejadorFoest solicita registrar usuario al ManejadorTipoPoblacion, se continua con el caso de uso Registrar Tipos de poblacion, subflujo administrar (s2).

Si se selecciona “administrar registro universidades”, la PantallaPrincipal envía el evento “administrar registro” a la InterfaceUsuario, la InterfaceUsuario envía el

evento “administrar registro” al ManejadorFoest, el ManejadorFoest solicita registrar usuario al ManejadorUniversidades, se continua con el caso de uso Registrar universidades, subflujo administrar (s2).

Si se selecciona “administrar registro carreras”, la PantallaPrincipal envía el evento “administrar registro” a la InterfaceUsuario, la InterfaceUsuario envía el evento “administrar registro” al ManejadorFoest, el ManejadorFoest solicita registrar usuario al ManejadorCarreras, se continua con el caso de uso Registrar carreras, subflujo administrar (s2).

Si se selecciona “administrar registro ciudad”, la PantallaPrincipal envía el evento “administrar registro” a la InterfaceUsuario, la InterfaceUsuario envía el evento “administrar registro” al ManejadorFoest, el ManejadorFoest solicita registrar usuario al ManejadorCiudad, se continua con el caso de uso Registrar ciudad, subflujo administrar (s2).

Si se selecciona “administrar certificado estudiante”, la PantallaPrincipal envía el evento “administrar registro” a la InterfaceUsuario, la InterfaceUsuario envía el evento “administrar registro” al ManejadorFoest, el ManejadorFoest solicita registrar usuario al ManejadorDocumentos, se continua con el caso de uso certificados estudiantes, subflujo administrar (s2).

Subflujos	Ninguno.
Excepciones	Ninguno.

7.5.3 Registrar funcionario

Caso de uso	Registrar funcionario.
Actores	Usuario, base de datos foest.
Tipo	Básico.
Propósito	Permite registrar usuarios para que puedan hacer uso del sistema.
Resumen	El usuario inicia este caso de uso. Ofrece funcionalidad para crear, modificar y eliminar el registro de usuario con el software de gestión de documentos archivo para el fondo para la educación superior Tocancipá (FOEST).
Precondiciones	Todas las actividades que se realicen requieren iniciar el caso de uso validar usuario.
Flujo principal	Se presenta cuando se ejecuta el caso de uso validar usuario, dependiendo de la opción que realice el usuario
Subflujos	<p>El <u>ManejadorFuncionario</u> solicita <u>DesplegarPantallaFuncionariosRegistrados</u> a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> despliega la <u>PantallaFuncionarioRegistrado</u> (p33).</p> <p>El usuario puede seleccionar las siguientes acciones: registrar, modificar, eliminar y salir.</p> <p>Si se selecciona “registrar” (e1, e2, e3, e4), la <u>PantallaFuncionarioRegistrado</u> envía el evento “registrar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “registrar” al <u>ManejadorFuncionario</u>, el <u>ManejadorFuncionario</u> envía el evento “registrar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “registrar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda el usuario y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorFuncionario</u>, el <u>ManejadorFuncionario</u> despliega la pantalla <u>FuncionariosRegistrados</u>.</p>

	<p>Si se selecciona “modificar” (e1, e3, e4), la <u>PantallaFuncionarioRegistrado</u> envía el evento “modificar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “modificar” al <u>ManejadorFuncionario</u>, el <u>ManejadorFuncionario</u> envía el evento “modificar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “modificar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda el usuario y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorFuncionario</u>, el <u>ManejadorFuncionario</u> despliega la pantalla <u>FuncionariosRegistrados</u>.</p> <p>Si se selecciona “eliminar”, la <u>PantallaFuncionarioRegistrado</u> envía el evento “eliminar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “eliminar” al <u>ManejadorFuncionario</u>, el <u>ManejadorFuncionario</u> envía el evento “eliminar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “eliminar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> elimina el usuario y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorFuncionario</u>, el <u>ManejadorFuncionario</u> despliega la pantalla <u>FuncionariosRegistrados</u>.</p>
Excepciones	<p>E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.</p> <p>E2. Ya existe: ya existe un usuario con el mismo login, se solicita al usuario que cambie el login.</p> <p>E3. Login incorrecto: el login no es válido, se solicita al usuario que corrija el login.</p> <p>E4. Password incorrecto: el password no es válido, se solicita al usuario que corrija el password.</p>

7.5.4 Registrar estudiantes

Caso de uso	Registrar estudiantes.
Actores	Usuario, base de datos foest.
Tipo	Básico.
Propósito	Permite registrar los estudiantes que son beneficiarios del FOEST.
Resumen	El usuario inicia este caso de uso. Ofrece funcionalidad para crear, modificar y eliminar el registro de los estudiantes inscritos en el FOEST con el software de gestión de documentos archivo para el fondo para la educación superior Tocancipá (FOEST).
Precondiciones	Todas las actividades que se realicen requieren iniciar el caso de uso validar usuario.
Flujo principal	Se presenta cuando se ejecuta el caso de uso validar usuario, dependiendo de la opción que realice el usuario
Subflujos	<p>El <u>ManejadorEstudiantes</u> solicita <u>DesplegarPantallaEstudiantesRegistrados</u> a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> despliega la <u>PantallaEstudianteRegistrado</u> (p3).</p> <p>El usuario puede seleccionar las siguientes acciones: registrar, modificar, eliminar y salir.</p> <p>Si se selecciona “registrar” (e1, e2, e3), la <u>PantallaEstudianteRegistrado</u> envía el evento “registrar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “registrar” al <u>ManejadorEstudiantes</u>, el <u>ManejadorEstudiantes</u> envía el evento “registrar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “registrar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda el usuario y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorEstudiantes</u>, el <u>ManejadorEstudiantes</u> despliega la pantalla <u>EstudiantesRegistrados</u>.</p>

	<p>Si se selecciona “modificar” (e1, e2, e3), la <u>PantallaEstudianteRegistrado</u> envía el evento “modificar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “modificar” al <u>ManejadorEstudiantes</u>, el <u>ManejadorEstudiantes</u> envía el evento “modificar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “modificar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda el usuario y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorEstudiantes</u>, el <u>ManejadorEstudiantes</u> despliega la pantalla <u>EstudiantesRegistrados</u>.</p> <p>Si se selecciona “eliminar”, la <u>PantallaEstudianteRegistrado</u> envía el evento “eliminar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “eliminar” al <u>ManejadorEstudiantes</u>, el <u>ManejadorEstudiantes</u> envía el evento “eliminar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “eliminar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> elimina el usuario y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorEstudiantes</u>, el <u>ManejadorEstudiantes</u> despliega la pantalla <u>EstudiantesRegistrados</u>.</p>
Excepciones	<p>E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.</p> <p>E2. Login incorrecto: el login no es válido, se solicita al usuario que corrija el login.</p> <p>E3. Password incorrecto: el password no es válido, se solicita al usuario que corrija el password.</p>

7.5.5 registrar beneficio1

Caso de uso	Registrar beneficio 1.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	Permite al usuario registrar un beneficio nuevo en caso de que no exista el beneficio con el cual cuenta un estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.
Flujo principal	Se continua con el subflujo de beneficio 1 (s1) de lo contrario, si crea un nuevo beneficio, se continua con el subflujo nuevo beneficio (s2).
Subflujos	<p>El <u>ManejadorBeneficio1</u> solicita DesplegarPantallaBeneficio1 a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> despliega la <u>PantallaBeneficio1</u> (p6).</p> <p>El usuario puede seleccionar las siguientes acciones: registrar, modificar, eliminar y salir.</p> <p>Si se selecciona “registrar” (e1), la <u>PantallaBeneficio1</u> envía el evento “registrar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “registrar” al <u>ManejadorBeneficio1</u>, el <u>ManejadorBeneficio1</u> envía el evento “registrar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “registrar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda el beneficio y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorBeneficio1</u>, el <u>ManejadorBeneficio1</u> despliega la pantalla <u>Beneficio1</u>.</p> <p>Si se selecciona “modificar” (e1), la <u>PantallaBeneficio1</u> envía el evento “modificar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “modificar” al</p>

	<p><u>ManejadorBeneficio1</u>, el <u>ManejadorBeneficio1</u> envía el evento “modificar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “modificar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda el beneficio y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorBeneficio1</u>, el <u>ManejadorBeneficio1</u> despliega la pantalla <u>Beneficio1</u>.</p> <p>Si se selecciona “eliminar”, la <u>PantallaBeneficio1</u> envía el evento “eliminar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “eliminar” al <u>ManejadorBeneficio1</u>, el <u>ManejadorBeneficio1</u> envía el evento “eliminar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “eliminar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> elimina el beneficio y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorBeneficio1</u>, el <u>ManejadorBeneficio1</u> despliega la pantalla <u>Beneficio1</u>.</p>
Excepciones	E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.

7.5.6 Registro beneficio2

Caso de uso	Registrar beneficio 2.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	Permite al usuario registrar un beneficio nuevo en caso de que no exista el beneficio con el cual cuenta un estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.

Flujo principal	Se continua con el subflujo de beneficio 2 (s1) de lo contrario, si crea un nuevo beneficio, se continua con el subflujo nuevo beneficio (s2).
Subflujos	<p>El <u>ManejadorBeneficio2</u> solicita DesplegarPantallaBeneficio2 a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> despliega la <u>PantallaBeneficio2</u> (p6).</p> <p>El usuario puede seleccionar las siguientes acciones: registrar, modificar, eliminar y salir.</p> <p>Si se selecciona “registrar” (e1), la <u>PantallaBeneficio2</u> envía el evento “registrar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “registrar” al <u>ManejadorBeneficio2</u>, el <u>ManejadorBeneficio2</u> envía el evento “registrar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “registrar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda el beneficio y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorBeneficio2</u>, el <u>ManejadorBeneficio2</u> despliega la pantalla <u>Beneficio2</u>.</p> <p>Si se selecciona “modificar” (e1), la <u>PantallaBeneficio2</u> envía el evento “modificar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “modificar” al <u>ManejadorBeneficio2</u>, el <u>ManejadorBeneficio2</u> envía el evento “modificar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “modificar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda el beneficio y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorBeneficio2</u>, el <u>ManejadorBeneficio2</u> despliega la pantalla <u>Beneficio2</u>.</p>

	Si se selecciona “eliminar”, la <u>PantallaBeneficio2</u> envía el evento “eliminar” a la <u>InterfaceUsuario</u> , la <u>InterfaceUsuario</u> envía el evento “eliminar” al <u>ManejadorBeneficio2</u> , el <u>ManejadorBeneficio2</u> envía el evento “eliminar” a la <u>InterfaceBasedeDatosRegistro</u> , la <u>InterfaceBasedeDatosRegistro</u> envía el evento “eliminar” a <u>BasedeDatosfoest</u> , la <u>BasedeDatosfoest</u> elimina el beneficio y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u> , la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorBeneficio2</u> , el <u>ManejadorBeneficio2</u> despliega la pantalla <u>Beneficio2</u> .
Excepciones	E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.

7.5.7 Registrar algún otro beneficio

Caso de uso	Registrar algún otro beneficio.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	Permite al usuario registrar un beneficio nuevo en caso de que no exista el beneficio con el cual cuenta un estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.
Flujo principal	Se continua con el subflujo de algún otro beneficio (s1) de lo contrario, si crea un nuevo beneficio, se continua con el subflujo nuevo beneficio (s2).
Subflujos	El <u>ManejadorAlgunOtroBeneficio</u> solicita <u>DesplegarPantallaAlgunOtroBeneficio</u> a la <u>InterfaceUsuario</u> , la <u>InterfaceUsuario</u> despliega la <u>PantallaAlgunOtroBeneficio</u> (p12). El usuario puede seleccionar las siguientes acciones: registrar, modificar, eliminar y salir.

Si se selecciona “registrar” (e1), la PantallaAlgunOtroBeneficio envía el evento “registrar” a la InterfaceUsuario, la InterfaceUsuario envía el evento “registrar” al ManejadorAlgunOtroBeneficio, el ManejadorAlgunOtroBeneficio envía el evento “registrar” a la InterfaceBasedeDatosRegistro, la InterfaceBasedeDatosRegistro envía el evento “registrar” a BasedeDatosfoest, la BasedeDatosfoest guarda el beneficio y devuelve el Ok a la InterfaceBasedeDatosRegistro, la InterfaceBasedeDatosRegistro devuelve el Ok al ManejadorAlgunOtroBeneficio, el ManejadorAlgunOtroBeneficio despliega la pantalla Algunotrobeneficio.

Si se selecciona “modificar” (e1), la PantallaAlgunOtroBeneficio envía el evento “modificar” a la InterfaceUsuario, la InterfaceUsuario envía el evento “modificar” al ManejadorAlgunOtroBeneficio, el ManejadorAlgunOtroBeneficio envía el evento “modificar” a la InterfaceBasedeDatosRegistro, la InterfaceBasedeDatosRegistro envía el evento “modificar” a BasedeDatosfoest, la BasedeDatosfoest guarda el beneficio y devuelve el Ok a la InterfaceBasedeDatosRegistro, la InterfaceBasedeDatosRegistro devuelve el Ok al ManejadorAlgunOtroBeneficio, el ManejadorAlgunOtroBeneficio despliega la pantalla Algunotrobeneficio.

Si se selecciona “eliminar”, la PantallaAlgunOtroBeneficio envía el evento “eliminar” a la InterfaceUsuario, la InterfaceUsuario envía el evento “eliminar” al ManejadorAlgunOtroBeneficio, el ManejadorAlgunOtroBeneficio envía el evento “eliminar” a la InterfaceBasedeDatosRegistro, la InterfaceBasedeDatosRegistro envía el evento “eliminar” a BasedeDatosfoest, la BasedeDatosfoest elimina el beneficio y devuelve el Ok a la InterfaceBasedeDatosRegistro, la

	<u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorAlgunOtroBeneficio</u> , el <u>ManejadorAlgunOtroBeneficio</u> despliega la pantalla <u>Algunotrobeneficio</u> .
Excepciones	E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.

7.5.8 Registrar antigüedad Sisbén.

Caso de uso	Registrar antigüedad Sisbén.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	Permite al usuario registrar una antigüedad nueva en caso de que no exista la antigüedad con el cual cuenta un estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.
Flujo principal	Se continua con el subflujo de antigüedad Sisbén (s1) de lo contrario, si crea una nueva antigüedad, se continua con el subflujo nueva antigüedad (s2).
Subflujos	<p>El <u>ManejadorAntigüedadSisben</u> solicita <u>DesplegarPantallaAntigüedadSisbén</u> a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> despliega la <u>PantallaAntigüedadSisben</u> (p15). El usuario puede seleccionar las siguientes acciones: registrar, modificar, eliminar y salir.</p> <p>Si se selecciona “registrar” (e1), la <u>PantallaAntigüedadSisben</u> envía el evento “registrar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “registrar” al <u>ManejadorAntigüedadSisben</u>, el <u>ManejadorAntigüedadSisben</u> envía el evento “registrar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “registrar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda la</p>

	<p>antigüedad y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorAntigüedadSisben</u>, el <u>ManejadorAntigüedadSisben</u> despliega la pantalla <u>AntigüedadSisbén</u>.</p> <p>Si se selecciona “modificar” (e1), la <u>PantallaAntigüedadSisben</u> envía el evento “modificar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “modificar” al <u>ManejadorAntigüedadSisben</u>, el <u>ManejadorAntigüedadSisben</u> envía el evento “modificar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “modificar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda la antigüedad y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorAntigüedadSisben</u>, el <u>ManejadorAntigüedadSisben</u> despliega la pantalla <u>AntigüedadSisbén</u>.</p> <p>Si se selecciona “eliminar”, la <u>PantallaAntigüedadSisben</u> envía el evento “eliminar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “eliminar” al <u>ManejadorAntigüedadSisben</u>, el <u>ManejadorAntigüedadSisben</u> envía el evento “eliminar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “eliminar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> elimina la antigüedad y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorAntigüedadSisben</u>, el <u>ManejadorAntigüedadSisben</u> despliega la pantalla <u>AntigüedadSisbén</u>.</p>
Excepciones	E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.

7.5.9 Registrar Tipos de población.

Caso de uso	Registrar tipos de población.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	Permite al usuario registrar diferentes tipos de población en caso de que no exista el tipo de población al cual pertenece el estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.
Flujo principal	Se continua con el subflujo de tipo de población (s1) de lo contrario, si crea una nueva población, se continua con el subflujo nueva población (s2).
Subflujos	<p>El <u>ManejadorTipoPoblacion</u> solicita <u>DesplegarPantallaTipoPoblacion</u> a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> despliega la <u>PantallaTiposPoblacion</u>(p18).</p> <p>El usuario puede seleccionar las siguientes acciones: registrar, modificar, eliminar y salir.</p> <p>Si se selecciona “registrar” (e1), la <u>PantallaTiposPoblacion</u> envía el evento “registrar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “registrar” al <u>ManejadorTipoPoblacion</u>, el <u>ManejadorTipoPoblacion</u> envía el evento “registrar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “registrar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda la poblacion y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorTipoPoblacion</u>, el <u>ManejadorTipoPoblacion</u> despliega la pantalla <u>TipoPoblacion</u>.</p> <p>Si se selecciona “modificar” (e1), la <u>PantallaTiposPoblacion</u> envía el evento “modificar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “modificar” al</p>

	<p><u>ManejadorTipoPoblacion</u>, el <u>ManejadorTipoPoblacion</u> envía el evento “modificar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “modificar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda la poblacion y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorTipoPoblacion</u>, el <u>ManejadorTipoPoblacion</u> despliega la pantalla <u>TipoPoblacion</u>.</p> <p>Si se selecciona “eliminar”, la <u>PantallaTiposPoblacion</u> envía el evento “eliminar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “eliminar” al <u>ManejadorTipoPoblacion</u>, el <u>ManejadorTipoPoblacion</u> envía el evento “eliminar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “eliminar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> elimina la poblacion y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorTipoPoblacion</u>, el <u>ManejadorTipoPoblacion</u> despliega la pantalla <u>TipoPoblacion</u>.</p>
Excepciones	E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.

7.5.10 Registrar universidades.

Caso de uso	Registrar universidades.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	Permite al usuario registrar diferentes universidades en caso de que no esté registrada la universidad a la cual pertenece el estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.

Flujo principal	Se continua con el subflujo de universidades (s1) de lo contrario, si crea una nueva universidad, se continua con el subflujo nueva universidad (s2).
Subflujos	<p>El <u>ManejadorUniversidades</u> solicita DesplegarPantallaUniversidades a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> despliega la <u>PantallaUniversidades</u>(p21).</p> <p>El usuario puede seleccionar las siguientes acciones: registrar, modificar, eliminar y salir.</p> <p>Si se selecciona “registrar” (e1), la <u>PantallaUniversidades</u> envía el evento “registrar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “registrar” al <u>ManejadorUniversidades</u>, el <u>ManejadorUniversidades</u> envía el evento “registrar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “registrar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda la universidad y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorUniversidades</u>, el <u>ManejadorUniversidades</u> despliega la pantalla <u>Universidades</u>.</p> <p>Si se selecciona “modificar” (e1), la <u>PantallaUniversidades</u> envía el evento “modificar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “modificar” al <u>ManejadorUniversidades</u>, el <u>ManejadorUniversidades</u> envía el evento “modificar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “modificar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda la universidad y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorUniversidades</u>, el <u>ManejadorUniversidades</u> despliega la pantalla <u>Universidades</u>.</p>

	<p>Si se selecciona “eliminar”, la <u>PantallaUniversidades</u> envía el evento “eliminar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “eliminar” al <u>ManejadorUniversidades</u>, el <u>ManejadorUniversidades</u> envía el evento “eliminar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “eliminar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> elimina la universidad y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorUniversidades</u>, el <u>ManejadorUniversidades</u> despliega la pantalla <u>Universidades</u>.</p>
Excepciones	E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.

7.5.11 Registrar carreras.

Caso de uso	Registrar carreras.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	Permite al usuario registrar diferentes carreras en caso de que no esté registrada la carrera a la cual pertenece el estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.
Flujo principal	Se continua con el subflujo de carreras (s1) de lo contrario, si crea una nueva carrera, se continua con el subflujo nueva población (s2).
Subflujos	<p>El <u>ManejadorCarreras</u> solicita DesplegarPantallaCarreras a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> despliega la <u>PantallaCarreras</u>(p24).</p> <p>El usuario puede seleccionar las siguientes acciones: registrar, modificar, eliminar y salir.</p>

Si se selecciona “registrar” (e1), la PantallaCarreras envía el evento “registrar” a la InterfaceUsuario, la InterfaceUsuario envía el evento “registrar” al ManejadorCarreras, el ManejadorCarreras envía el evento “registrar” a la InterfaceBasedeDatosRegistro, la InterfaceBasedeDatosRegistro envía el evento “registrar” a BasedeDatosfoest, la BasedeDatosfoest guarda la carrera y devuelve el Ok a la InterfaceBasedeDatosRegistro, la InterfaceBasedeDatosRegistro devuelve el Ok al ManejadorCarreras, el ManejadorCarreras despliega la pantalla Carreras.

Si se selecciona “modificar” (e1), la PantallaCarreras envía el evento “modificar” a la InterfaceUsuario, la InterfaceUsuario envía el evento “modificar” al ManejadorCarreras, el ManejadorCarreras envía el evento “modificar” a la InterfaceBasedeDatosRegistro, la InterfaceBasedeDatosRegistro envía el evento “modificar” a BasedeDatosfoest, la BasedeDatosfoest guarda la carrera y devuelve el Ok a la InterfaceBasedeDatosRegistro, la InterfaceBasedeDatosRegistro devuelve el Ok al ManejadorCarreras, el ManejadorCarreras despliega la pantalla Carreras.

Si se selecciona “eliminar”, la PantallaCarreras envía el evento “eliminar” a la InterfaceUsuario, la InterfaceUsuario envía el evento “eliminar” al ManejadorCarreras, el ManejadorCarreras envía el evento “eliminar” a la InterfaceBasedeDatosRegistro, la InterfaceBasedeDatosRegistro envía el evento “eliminar” a BasedeDatosfoest, la BasedeDatosfoest elimina la carrera y devuelve el Ok a la InterfaceBasedeDatosRegistro, la InterfaceBasedeDatosRegistro devuelve el Ok al ManejadorCarreras, el ManejadorCarreras despliega la pantalla Carreras.

Excepciones	E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.
--------------------	--

7.5.12 Registrar Ciudad.

Caso de uso	Registrar ciudad.
Actores	Usuario, base de datos foest.
Tipo	Extensión.
Propósito	Ofrece las diversas opciones al usuario ya registrado para que use el sistema.
Resumen	Permite al usuario registrar diferentes ciudades en caso de que no esté registrada la ciudad a la cual viaja el estudiante.
Precondiciones	Se requiere que el usuario se halla registrado antes.
Flujo principal	Se continua con el subflujo de ciudades (s1) de lo contrario, si crea un nuevo registro para la ciudad, se continua con el subflujo nueva ciudad (s2).
Subflujos	<p>El <u>ManejadorCiudad</u> solicita <u>DesplegarPantallaCiudad</u> a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> despliega la <u>PantallaCiudad</u>(p27).</p> <p>El usuario puede seleccionar las siguientes acciones: registrar, modificar, eliminar y salir.</p> <p>Si se selecciona “registrar” (e1), la <u>PantallaCiudad</u> envía el evento “registrar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “registrar” al <u>ManejadorCiudad</u>, el <u>ManejadorCiudad</u> envía el evento “registrar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “registrar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda la ciudad y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u></p>

	<p>devuelve el Ok al <u>ManejadorCiudad</u>, el <u>ManejadorCiudad</u> despliega la pantalla <u>Ciudad</u>.</p> <p>Si se selecciona “modificar” (e1), la <u>PantallaCiudad</u> envía el evento “modificar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “modificar” al <u>ManejadorCiudad</u>, el <u>ManejadorCiudad</u> envía el evento “modificar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “modificar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> guarda la ciudad y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorCiudad</u>, el <u>ManejadorCiudad</u> despliega la pantalla <u>Ciudad</u>.</p> <p>Si se selecciona “eliminar”, la <u>PantallaCiudad</u> envía el evento “eliminar” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “eliminar” al <u>ManejadorCiudad</u>, el <u>ManejadorCiudad</u> envía el evento “eliminar” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “eliminar” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> elimina la ciudad y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorCiudad</u>, el <u>ManejadorCiudad</u> despliega la pantalla <u>Ciudad</u>.</p>
Excepciones	E1. Información incompleta: faltan espacios por completar, se solicita al usuario completar los espacios vacíos.

7.5.12 Certificado estudiante.

Caso de uso	Certificado Estudiante.
Actores	Usuario, base de datos foest
Tipo	Básico.
Propósito	Permite al usuario hacer estadísticas y certificados.
Resumen	El usuario inicia este caso de uso. Ofrece funcionalidad para crear, obtener, modificar y eliminar estadísticas y certificados.
Precondiciones	Se requiere que el usuario se halla validado correctamente.
Flujo principal	Se ejecuta el caso de uso validar usuario, dependiendo de la opción seleccionada por el usuario, se continuará con el respectivo subflujo.
Subflujos	<p>El <u>ManejadorDocumentos</u> solicita <u>DesplegarPantallaDocumentos</u> a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> despliega la <u>PantallaDocumentos</u>(p34).</p> <p>El usuario puede seleccionar las siguientes acciones: certificados, estadísticas y salir.</p> <p>Si se selecciona “certificados” (e1), la <u>PantallaDocumentos</u> envía el evento “pedir certificado” a la <u>InterfaceUsuario</u>, la <u>InterfaceUsuario</u> envía el evento “pedir certificado” al <u>ManejadorDocumentos</u>, el <u>ManejadorDocumentos</u> envía el evento “pedir certificado” a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> envía el evento “pedir certificado” a <u>BasedeDatosfoest</u>, la <u>BasedeDatosfoest</u> utiliza los datos del usuario y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u>, la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorDocumentos</u>, el <u>ManejadorDocumentos</u> despliega la pantalla <u>Certificados</u> (p31).</p>

	Si se selecciona “estadísticas” (e1), la <u>PantallaDocumentos</u> envía el evento “estadísticas” a la <u>InterfaceUsuario</u> , la <u>InterfaceUsuario</u> envía el evento “estadísticas” al <u>ManejadorDocumentos</u> , el <u>ManejadorDocumentos</u> envía el evento “estadísticas” a la <u>InterfaceBasedeDatosRegistro</u> , la <u>InterfaceBasedeDatosRegistro</u> envía el evento “estadísticas” a <u>BasedeDatosfoest</u> , la <u>BasedeDatosfoest</u> utiliza los datos de los usuarios y devuelve el Ok a la <u>InterfaceBasedeDatosRegistro</u> , la <u>InterfaceBasedeDatosRegistro</u> devuelve el Ok al <u>ManejadorDocumentos</u> , el <u>ManejadorDocumentos</u> despliega la pantalla <u>Estadísticas</u> (p30).
Excepciones	E1. El documento no pertenece a ningún estudiante inscrito o el espacio este vacío.

7.6. Diccionario de Clases

7.6.1 Interface Usuario

Está compuesto por la siguiente clase:

InterfaceUsuario: clase borde. Toda la interacción con el usuario se hace por medio del borde de usuario.

7.6.2 Ingreso

Está compuesto por las siguientes clases:

PantallaIngreso Clase Borde. Pantalla ingreso (p1)

ManejadorPrincipal Clase Control. El manejador principal es el encargado de desplegar la pantalla ingreso de interacción con el usuario.

7.6.3 Principal

Está compuesto por las siguientes clases:

PantallaPrincipal Clase Borde. Pantalla principal (p2)

ManejadorPrincipal Clase Control. El manejador principal es el encargado de desplegar la pantalla principal de interacción con el usuario, y luego delegar las diferentes funciones a los manejadores especializados apropiados.

7.6.4 Registro

Se divide en los siguientes módulos:

Usuario Funcionario:

Está compuesto por las siguientes clases:

PantallaRegistrarFuncionario Clase borde. Pantalla crear funcionario (P32).

PantallaFuncionarioRegistrado Clase borde. Pantalla funcionarios registrados (P33).

RegistrarUsuarioFuncionario Clase entidad. Para utilizar el sistema de registro de estudiantes, el usuario debe estar registrado en el sistema. El registro contiene information acerca del usuario que incluye nombre, apellido, teléfono, email, login y password.

ManejadorFuncionario Clase control. El manejador de registro de usuario funcionario se encarga de todo lo relacionado con el registro del usuario para poder utilizar el sistema.

Usuario Estudiante:

Está compuesto por las siguientes clases:

PantallaRegistrarEstudiante Clase borde. Pantalla registro estudiantes (P4).

PantallaEstudianteRegistrado Clase borde. Pantalla estudiantes registrados (P3).

RegistrarUsuarioEstudiantes Clase entidad. El registro contiene los siguientes campos son los siguientes: N° carpeta, fecha de radicación, beneficio 1, beneficio 2, 1° vez, renovación, reintegro, primer apellido, segundo apellido, primer nombre, segundo nombre, tipo de documento, número de documento Expedido en, edad, población especial, género, teléfono 1; teléfono 2, dirección, correo electrónico, antigüedad Sisbén en años, puntaje del Sisbén, recibe algún otro beneficio, universidad sede nivel carrera, modalidad, semestre, valor del semestre, porcentaje a aplicar monto aprobado, otro beneficio, materias perdidas promedio 2018_1 y 2017, horas sociales cumplidas, dependencia donde realizo horas sociales, Horario, ciudad a la que viaja, días que viaja, valor por 2 recorridos valor semanal, valor mensual bruto, valor a pagar por mes, meses a girar y total semestre; también los datos del representante legal si es un menor de edad.

ManejadorFuncionario Clase control. El manejador de registro de usuario estudiante se encarga de todo lo relacionado con el registro del usuario estudiantes.

Beneficio1:

Está compuesto por las siguientes clases:

PantallaRegistrarBeneficio1 Clase borde. Pantalla registro beneficio1 (P7).

PantallaBeneficio1 Clase borde. Pantalla beneficio1 (P6).

RegistrarBeneficio1 Clase entidad. El registro contiene el nombre del beneficio.

ManejadorBeneficio1 Clase control. El manejador de registro de beneficio1 se encarga de todo lo relacionado con el registro del beneficio.

Beneficio2:

Está compuesto por las siguientes clases:

PantallaRegistrarBeneficio2 Clase borde. Pantalla registro beneficio2 (P10).

PantallaBeneficio2 Clase borde. Pantalla beneficio2 (P9).

RegistrarBeneficio2 Clase entidad. El registro contiene el nombre del beneficio.

ManejadorBeneficio2 Clase control. El manejador de registro de beneficio2 se encarga de todo lo relacionado con el registro del beneficio.

Algún otro Beneficio:

Está compuesto por las siguientes clases:

PantallaRegistrarAlgúnOtroBeneficio Clase borde. Pantalla registro algún otro beneficio (P13).

PantallaAlgúnOtroBeneficio Clase borde. Pantalla algún otro beneficio (P12).

RegistrarAlgúnOtroBeneficio Clase entidad. El registro contiene el nombre del beneficio.

ManejadorAlgúnOtroBeneficio Clase control. El manejador de registro de algún otro beneficio se encarga de todo lo relacionado con el registro del beneficio.

Antigüedad Sisbén:

Está compuesto por las siguientes clases:

PantallaRegistroAntigüedadSisben Clase borde. Pantalla registro antigüedad Sisbén (P16).

PantallaAntigüedadSisben Clase borde. Pantalla antigüedad Sisbén (P15).

RegistrarAntigüedadSisben Clase entidad. El registro contiene el numero en años de antigüedad.

ManejadorAntigüedadSisben Clase control. El manejador de registro de antigüedad Sisbén se encarga de todo lo relacionado con el registro de la antigüedad.

Tipos de población:

Está compuesto por las siguientes clases:

PantallaRegistroTiposPoblacion Clase borde. Pantalla registro tipos de poblacion (P19).

PantallaTiposPoblacion Clase borde. Pantalla Tipos de poblacion (P18).

RegistrarTipoPoblacion Clase entidad. El registro contiene el nombre de la poblacion.

ManejadorTipoPoblacion Clase control. El manejador de registro de tipos de poblacion se encarga de todo lo relacionado con el registro de la poblacion.

Universidades:

Está compuesto por las siguientes clases:

PantallaRegistroUniversidad Clase borde. Pantalla registro universidades (P22).

PantallaUniversidades Clase borde. Pantalla universidades (P21).

RegistrarUniversidades Clase entidad. El registro contiene el nombre de la universidad y porcentaje.

ManejadorUniversidades Clase control. El manejador de registro de universidades se encarga de todo lo relacionado con el registro de la universidad.

Carreras:

Está compuesto por las siguientes clases:

PantallaRegistroCarreras Clase borde. Pantalla registro carreras (P25).

PantallaCarreras Clase borde. Pantalla carreras (P24).

RegistrarCarreras Clase entidad. El registro contiene el nombre de la carrera.

ManejadorCarreras Clase control. El manejador de registro de carreras se encarga de todo lo relacionado con el registro de la carrera.

Ciudad:

Está compuesto por las siguientes clases:

PantallaRegistroCiudad Clase borde. Pantalla registro ciudad (P28).

PantallaCiudad Clase borde. Pantalla ciudad (P27).

RegistrarCiudad Clase entidad. El registro contiene el nombre de la ciudad.

ManejadorCiudad Clase control. El manejador de registro de ciudad se encarga de todo lo relacionado con el registro de la ciudad.

Certificados estudiantes:

Está compuesto por las siguientes clases:

PantallaDocumentos Clase borde. Pantalla documentos (P34).

PantallaCertificado Clase borde. Pantalla certificados (P31).

PantallaEstadísticas Clase borde. Pantalla estadística (P30).

CertificadoEstudiante Clase entidad. Da el certificado pedido.

Estadísticas Clase entidad. Da las estadísticas de la base de datos.

ManejadorDocumentos Clase control. El manejador de documentos se encarga de todo lo relacionado con estadísticas y certificados.

Interfacebasededatos:

Está compuesto por la siguiente clase:

InterfaceBasedeDatosRegistro - Clase Borde. La información de cada usuario se almacena en la base de datos foest, el cual se accesa mediante la interface de la base de datos de registro. Esto permite validar a los distintos usuarios funcionarios, además de guardar información.

Capítulo 8 Modelo de Diseño

8.1. Diseño de Objetos

8.1.1 Tarjetas de clases

Estas son las tarjetas de clases del proyecto, solo se encuentran las características de nombre, descripción, modulo y estereotipo, las demás se encontrarán en los subíndices de más adelante.

Interface usuario:

Tarjeta para la clase interface usuario

Clase: InterfaceUsuario	
Descripción: toda la interacción con el usuario se realiza por medio de la interface de usuario	
Modulo: InterfaceUsuario	
Estereotipo: Borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

ManejadorPrincipal:

Tarjeta para la clase ManejadorPrincipal

Clase: ManejadorPrincipal	
Descripción: es el encargado de desplegar la pantalla principal de interacción con el usuario y después desplegar las diferentes funciones a los manejadores apropiados	
Modulo: Principal	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Pantalla ingreso:

Tarjeta para la clase pantalla ingreso

Clase: PantallaIngreso	
Descripción: pantalla ingreso (P1)	
Modulo: Principal	
Estereotipo: Borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Registro:

Este módulo se compone del módulo usuario funcionario, usuario estudiante, beneficio1, beneficio2, algún otro beneficio, antigüedad Sisbén, tipos de poblacion, universidades, carreras, ciudad, certificados estudiantes e InterfaceBasedeDatosRegistro.

Usuario funcionario:

Este módulo se compone de las clases ManejadorFuncionario, PantallaRegistrarFuncionario, PantallaFuncionarioRegistrado y RegistrarUsuarioFuncionario.

Tarjeta para la clase ManejadorFuncionario

Clase: ManejadorFuncionario	
Descripción: se encarga de todo lo relacionado con el registro del usuario funcionario	
Modulo: RegistrarUsuarioFuncionario	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaRegistrarFuncionario

Clase: PantallaRegistrarFuncionario	
Descripción: pantalla crear funcionarios (p32)	
Modulo: registro, usuario funcionario	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaFuncionarioRegistrado

Clase: PantallaFuncionarioRegistrado	
Descripción: pantalla funcionarios registrados (p33)	
Modulo: registro, usuario funcionario	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarUsuarioFuncionario

Clase: RegistrarUsuarioFuncionario	
Descripción: el usuario para que pueda utilizar el sistema, debe estar registrado, en el registro se debe ingresar la información del usuario: nombre, apellido, teléfono, email, login y password.	
Modulo: registro, usuario funcionario	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Usuarios estudiantes:

Este módulo se compone de las clases ManejadorEstudiantes, PantallaRegistrarEstudiante, PantallaEstudianteRegistrado y RegistrarUsuarioEstudiantes.

Tarjeta para la clase ManejadorEstudiantes

Clase: ManejadorEstudiantes	
Descripción: se encarga de todo lo relacionado con el registro del usuario estudiante	
Modulo: RegistrarUsuarioEstudiantes	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaRegistrarEstudiante

Clase: PantallaRegistrarEstudiante	
Descripción: pantalla registro estudiantes (p4)	
Modulo: registro, usuario estudiante	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaEstudianteRegistrado

Clase: PantallaEstudianteRegistrado	
Descripción: pantalla estudiantes registrados (p3)	
Modulo: registro, usuarios estudiantes	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	

Atributos:	

Tarjeta para la clase RegistrarUsuarioEstudiantes

Clase: RegistrarUsuarioEstudiantes	
Descripción: en el registro se debe ingresar la información del usuario: N° carpeta, fecha de radicación, beneficio 1, beneficio 2, 1° vez, renovación, reintegro, primer apellido, segundo apellido, primer nombre, segundo nombre, tipo de documento, numero de documento Expedido en, edad, población especial, genero, teléfono 1; teléfono 2, dirección, correo electrónico, antigüedad Sisbén en años, puntaje del Sisbén, recibe algún otro beneficio, universidad sede nivel carrera, modalidad, semestre, valor del semestre, porcentaje a aplicar monto aprobado, otro beneficio, materias perdidas promedio 2018_1 y 2017, horas sociales cumplidas, dependencia donde realizo horas sociales, Horario, ciudad a la que viaja, días que viaja, valor por 2 recorridos valor semanal, valor mensual bruto , valor a pagar por mes, meses a girar y total semestre; también los datos del representante legal si es un menor de edad.	
Modulo: registro, usuarios estudiantes	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Beneficio1:

Este módulo se compone de las clases ManejadorBeneficio1, PantallaRegistrarBeneficio1, PantallaBeneficio1 y RegistrarBeneficio1.

Tarjeta para la clase ManejadorBeneficio1

Clase: ManejadorBeneficio1	
Descripción: se encarga de todo lo relacionado con el registro del beneficio1	
Modulo: RegistrarBeneficio1	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

--	--

Tarjeta para la clase PantallaRegistrarBeneficio1

Clase: PantallaRegistrarBeneficio1	
Descripción: pantalla registro beneficio 1 (p7)	
Modulo: registro, beneficio1	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaBeneficio1

Clase: PantallaBeneficio1	
Descripción: pantalla beneficio1 (p6)	
Modulo: registro, beneficio1	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarBeneficio1

Clase: RegistrarBeneficio1	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, beneficio1	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Beneficio2:

Este módulo se compone de las clases ManejadorBeneficio2, PantallaRegistrarBeneficio2, PantallaBeneficio2 y RegistrarBeneficio2.

Tarjeta para la clase ManejadorBeneficio2

Clase: ManejadorBeneficio2	
Descripción: se encarga de todo lo relacionado con el registro del beneficio2	
Modulo: RegistrarBeneficio2	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaRegistrarBeneficio2

Clase: PantallaRegistrarBeneficio2	
Descripción: pantalla registro beneficio2 (p10)	
Modulo: registro, beneficio2	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaBeneficio2

Clase: PantallaBeneficio2	
Descripción: pantalla beneficio2 (p9)	
Modulo: registro, beneficio2	
Estereotipo: borde	
Propiedades:	

Superclases:
Subclases:
Atributos:

Tarjeta para la clase RegistrarBeneficio2

Clase: RegistrarBeneficio2
Descripción: en el registro se debe ingresar la información del beneficio: nombre.
Modulo: registro, beneficio2
Estereotipo: entidad
Propiedades:
Superclases:
Subclases:
Atributos:

Algún Otro Beneficio:

Este módulo se compone de las clases ManejadorAlgúnOtroBeneficio, PantallaRegistrarAlgúnOtroBeneficio, PantallaAlgúnOtroBeneficio, y RegistrarAlgúnOtroBeneficio.

Tarjeta para la clase ManejadorAlgúnOtroBeneficio

Clase: ManejadorAlgúnOtroBeneficio
Descripción: se encarga de todo lo relacionado con el registro del beneficio2
Modulo: RegistrarAlgúnOtroBeneficio
Estereotipo: Control
Propiedades:
Superclases:
Subclases:
Atributos:

Tarjeta para la clase PantallaRegistrarAlgunOtroBeneficio

Clase: PantallaRegistrarAlgunOtroBeneficio	
Descripción: pantalla registro algún otro beneficio (p13)	
Modulo: registro, Algunotrobeficio	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaAlgunOtroBeneficio

Clase: PantallaAlgunOtroBeneficio	
Descripción: pantalla algún otro beneficio (p12)	
Modulo: registro, Algunotrobeficio	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarAlgunOtroBeneficio

Clase: RegistrarAlgunOtroBeneficio	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, Algunotrobeficio	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Antigüedad Sisbén:

Este módulo se compone de las clases ManejadorAntigüedadSisben, PantallaRegistroAntigüedadSisben, PantallaAntigüedadSisben, y RegistrarAntigüedadSisben.

Tarjeta para la clase ManejadorAntigüedadSisben

Clase: ManejadorAntigüedadSisben	
Descripción: se encarga de todo lo relacionado con el registro de la antigüedad del Sisbén	
Modulo: RegistrarAntigüedadSisben	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaRegistroAntigüedadSisben

Clase: PantallaRegistroAntigüedadSisben	
Descripción: pantalla registro antigüedad Sisbén (p16)	
Modulo: registro, AntigüedadSisbén	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaAntigüedadSisben

Clase: PantallaAntigüedadSisben	
Descripción: pantalla antigüedad Sisbén (p15)	
Modulo: registro, AntigüedadSisbén	
Estereotipo: borde	
Propiedades:	

Superclases:
Subclases:
Atributos:

Tarjeta para la clase RegistrarAntigüedadSisben

Clase: RegistrarAntigüedadSisben
Descripción: en el registro se debe ingresar la información de la antigüedad Sisbén: cantidad de años.
Modulo: registro, AntigüedadSisbén
Estereotipo: entidad
Propiedades:
Superclases:
Subclases:
Atributos:

Tipos de poblacion:

Este módulo se compone de las clases ManejadorTipoPoblacion, PantallaRegistroTiposPoblacion, PantallaTiposPoblacion, y RegistrarTipoPoblacion.

Tarjeta para la clase ManejadorTipoPoblacion

Clase: ManejadorTipoPoblacion
Descripción: se encarga de todo lo relacionado con el registro del tipo de poblacion.
Modulo: RegistrarTipoPoblacion
Estereotipo: Control
Propiedades:
Superclases:
Subclases:
Atributos:

Tarjeta para la clase PantallaRegistroTiposPoblacion

Clase: PantallaRegistroTiposPoblacion	
Descripción: pantalla registro tipos de poblacion (p19)	
Modulo: registro, TipoPoblacion	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaTiposPoblacion

Clase: PantallaTiposPoblacion	
Descripción: pantalla tipos de poblacion (p18)	
Modulo: registro, TipoPoblacion	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarTipoPoblacion

Clase: RegistrarTipoPoblacion	
Descripción: en el registro se debe ingresar la información del tipo de poblacion: nombre.	
Modulo: registro, TipoPoblacion	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Universidades:

Este módulo se compone de las clases ManejadorUniversidades, PantallaRegistroUniversidad, PantallaUniversidades, y RegistrarUniversidades.

Tarjeta para la clase ManejadorUniversidades

Clase: ManejadorUniversidades	
Descripción: se encarga de todo lo relacionado con el registro de las universidades.	
Modulo: RegistrarUniversidades	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaRegistroUniversidad

Clase: PantallaRegistroUniversidad	
Descripción: pantalla registro universidades (p22)	
Modulo: registro, Universidad	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaUniversidades

Clase: PantallaUniversidades	
Descripción: pantalla universidades (p21)	
Modulo: registro, Universidad	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarUniversidades

Clase: RegistrarUniversidades	
Descripción: en el registro se debe ingresar la información de la universidad: nombre y porcentaje a aplicar.	
Modulo: registro, Universidad	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Carreras:

Este módulo se compone de las clases ManejadorCarreras, PantallaRegistroCarreras, PantallaCarreras, y RegistrarCarreras.

Tarjeta para la clase ManejadorCarreras

Clase: ManejadorCarreras	
Descripción: se encarga de todo lo relacionado con el registro de las carreras.	
Modulo: RegistrarCarreras	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaRegistroCarreras

Clase: PantallaRegistroCarreras	
Descripción: pantalla registro carreras (p25)	
Modulo: registro, Carreras	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaCarreras

Clase: PantallaCarreras	
Descripción: pantalla carreras (p24)	
Modulo: registro, Carreras	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarCarreras

Clase: RegistrarCarreras	
Descripción: en el registro se debe ingresar la información de la carrera: nombre.	
Modulo: registro, Carreras	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Ciudad:

Este módulo se compone de las clases ManejadorCiudad, PantallaRegistroCiudad, PantallaCiudad, y RegistrarCiudad.

Tarjeta para la clase ManejadorCiudad

Clase: ManejadorCiudad	
Descripción: se encarga de todo lo relacionado con el registro de la ciudad.	
Modulo: RegistrarCiudad	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaRegistroCiudad

Clase: PantallaRegistroCiudad	
Descripción: pantalla registro ciudad (p28)	
Modulo: registro, Ciudad	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaCiudad

Clase: PantallaCiudad	
Descripción: pantalla ciudad (p27)	
Modulo: registro, Ciudad	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarCiudad

Clase: RegistrarCiudad	
Descripción: en el registro se debe ingresar la información de la ciudad: nombre.	
Modulo: registro, Ciudad	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Certificación estudiantes:

:

Este módulo se compone de las clases ManejadorDocumentos, PantallaDocumentos, PantallaCertificado, PantallaEstadísticas, CertificadoEstudiante y Estadísticas.

Tarjeta para la clase ManejadorDocumentos

Clase: ManejadorDocumentos	
Descripción: se encarga de todo lo relacionado con informes como estadísticas y certificados.	
Modulo: CertificadoEstudiante y Estadísticas.	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaDocumentos

Clase: PantallaDocumentos	
Descripción: pantalla registro ciudad (p34)	
Modulo: Certificados y Estadísticas.	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaCertificado

Clase: PantallaCertificado	
Descripción: pantalla certificados (p31)	
Modulo: Certificados.	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaEstadísticas

Clase: PantallaEstadísticas	
Descripción: pantalla estadísticas (p30)	
Modulo: Estadísticas.	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase CertificadoEstudiante

Clase: CertificadoEstudiante	
Descripción: en el certificado se debe ingresar la información del estudiante: documento de identidad.	
Modulo: registro, Certificados	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase Estadísticas

Clase: Estadísticas	
Descripción: en estadísticas se toman los datos de los estudiantes registrado en la base de datos.	
Modulo: registro, Estadísticas	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Interface base de datos:

Tarjeta para la clase InterfaceBasedeDatosRegistro

Clase: InterfaceBasedeDatosRegistro	
Descripción: la información de los usuarios, beneficios y otros se almacena en la base de datos foest	
Modulo: Registro interfaceBD	
Estereotipo: interface	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

--	--

Estudiantes foest:

Tarjeta para la clase ManejadorFoest

Clase: ManejadorFoest	
Descripción: se encarga de enviar las peticiones particulares de opciones a los manejadores especializados para registro de las diferentes opciones.	
Modulo: estudiantes	
Estereotipo: control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Pantalla principal:

Tarjeta para la clase pantalla principal

Clase: PantallaPrincipal	
Descripción: pantalla principal (P2)	
Modulo: Principal	
Estereotipo: Borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

8.1.2 Responsabilidades

Interface usuario:

Tarjeta para la clase InterfaceUsuario con responsabilidades.

Clase: InterfaceUsuario	
Descripción: toda la interacción con el usuario se realiza por medio de la interface de usuario	
Modulo: InterfaceUsuario	
Estereotipo: Borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Despliega la PantallaIngreso	
Envía el evento "ingresar" al ManejadorPrincipal	
Envía el evento "OK" al ManejadorPrincipal	
Envía el evento "salir" al ManejadorPrincipal	
Despliega la PantallaPrincipal	
Envía el evento "administrar usuario funcionario" al ManejadorFoest	
Envía el evento "administrar usuario estudiante" al ManejadorFoest	
Envía el evento "administrar beneficio1" al ManejadorFoest	
Envía el evento "administrar beneficio2" al ManejadorFoest	
Envía el evento "administrar algún otro beneficio" al ManejadorFoest	
Envía el evento "administrar antigüedad Sisbén" al ManejadorFoest	
Envía el evento "administrar tipos de poblacion" al ManejadorFoest	
Envía el evento "administrar universidades" al ManejadorFoest	
Envía el evento "administrar carreras" al ManejadorFoest	
Envía el evento "administrar ciudad" al ManejadorFoest	
Envía el evento "administrar documentos" al ManejadorFoest	
Envía el evento "salir" al ManejadorFoest	
Despliega la PantallaFuncionarioRegistrado	
Envía el evento "registrar" al ManejadorFuncionario	
Envía el evento "modificar" al ManejadorFuncionario	
Envía el evento "eliminar" al ManejadorFuncionario	
Envía el evento "salir" al ManejadorFuncionario	

Despliega la PantallaEstudianteRegistrado	
Envía el evento “registrar” al ManejadorEstudiantes	
Envía el evento “modificar” al ManejadorEstudiantes	
Envía el evento “eliminar” al ManejadorEstudiantes	
Envía el evento “salir” al ManejadorEstudiantes	
Despliega la PantallaBeneficio1	
Envía el evento “registrar” al ManejadorBeneficio1	
Envía el evento “modificar” al ManejadorBeneficio1	
Envía el evento “eliminar” al ManejadorBeneficio1	
Envía el evento “salir” al ManejadorBeneficio1	
Despliega la PantallaBeneficio2	
Envía el evento “registrar” al ManejadorBeneficio2	
Envía el evento “modificar” al ManejadorBeneficio2	
Envía el evento “eliminar” al ManejadorBeneficio2	
Envía el evento “salir” al ManejadorBeneficio2	
Despliega la PantallaAlgunOtroBeneficio	
Envía el evento “registrar” al ManejadorAlgunOtroBeneficio	
Envía el evento “modificar” al ManejadorAlgunOtroBeneficio	
Envía el evento “eliminar” al ManejadorAlgunOtroBeneficio	
Envía el evento “salir” al ManejadorAlgunOtroBeneficio	
Despliega la PantallaAntiguedadSisben	
Envía el evento “registrar” al ManejadorAntiguedadSisben	
Envía el evento “modificar” al ManejadorAntiguedadSisben	
Envía el evento “eliminar” al ManejadorAntiguedadSisben	
Envía el evento “salir” al ManejadorAntiguedadSisben	
Despliega la PantallaTiposPoblacion	
Envía el evento “registrar” al ManejadorTipoPoblacion	
Envía el evento “modificar” al ManejadorTipoPoblacion	
Envía el evento “eliminar” al ManejadorTipoPoblacion	
Envía el evento “salir” al ManejadorTipoPoblacion	
Despliega la PantallaUniversidades	
Envía el evento “registrar” al ManejadorUniversidades	
Envía el evento “modificar” al ManejadorUniversidades	
Envía el evento “eliminar” al ManejadorUniversidades	
Envía el evento “salir” al ManejadorUniversidades	
Despliega la PantallaCarreras	
Envía el evento “registrar” al ManejadorCarreras	
Envía el evento “modificar” al ManejadorCarreras	
Envía el evento “eliminar” al ManejadorCarreras	
Envía el evento “salir” al ManejadorCarreras	
Despliega la PantallaCiudad	
Envía el evento “registrar” al ManejadorCiudad	

Envía el evento “modificar” al ManejadorCiudad	
Envía el evento “eliminar” al ManejadorCiudad	
Envía el evento “salir” al ManejadorCiudad	
Despliega la PantallaDocumentos	
Envía el evento “Pedir certificado” al ManejadorDocumentos	
Envía el evento “estadísticas” al ManejadorDocumentos	
Envía el evento “salir” al ManejadorDocumentos	
Despliega la PantallaCertificado	
Envía el evento “aceptar” al ManejadorCertificado	
Envía el evento “salir” al ManejadorCertificado	
Despliega la PantallaEstadísticas	
Envía el evento “aceptar” al ManejadorEstadísticas	
Envía el evento “salir” al ManejadorEstadísticas	

ManejadorPrincipal:

Tarjeta para la clase ManejadorPrincipal con responsabilidades.

Clase: ManejadorPrincipal	
Descripción: es el encargado de desplegar la pantalla principal de interacción con el usuario y después desplegar las diferentes funciones a los manejadores apropiados	
Modulo: Principal	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita desplegar pantalla a la InterfaceUsuario	
Solicita ValidarRegistroUsuarioFuncionario al Manejadorfuncionario	
Solicita estudiantes foest al ManejadorFoest	
Maneja el evento “salir”	
Sale del sistema	

Pantalla ingreso:

Tarjeta para la clase pantalla ingreso con responsabilidades.

Clase: PantallaIngreso	
Descripción: pantalla ingreso (P1)	
Modulo: Principal	
Estereotipo: Borde	

Propiedades:
Superclases:
Subclases:
Atributos:
Envía el evento "ingresar" a la InterfaceUsuario
Envía el evento "OK" a la InterfaceUsuario
Envía el evento "salir" a la InterfaceUsuario

Registro:

Este módulo se compone del módulo usuario funcionario, usuario estudiante, beneficio1, beneficio2, algún otro beneficio, antigüedad Sisbén, tipos de poblacion, universidades, carreras, ciudad, certificados estudiantes e InterfaceBasedeDatosRegistro.

Usuario funcionario:

Este módulo se compone de las clases ManejadorFuncionario, PantallaRegistrarFuncionario, PantallaFuncionarioRegistrado y RegistrarUsuarioFuncionario.

Tarjeta para la clase ManejadorFuncionario con responsabilidades.

Clase: ManejadorFuncionario
Descripción: se encarga de todo lo relacionado con el registro del usuario funcionario
Modulo: RegistrarUsuarioFuncionario
Estereotipo: Control
Propiedades:
Superclases:
Subclases:
Atributos:
CrearRegistroUsuarioFuncionario
Solicita ValidarRegistroUsuarioFuncionario a la InterfaceBasedeDatosRegistro
Solicita DesplegarPantallaRegistroFuncionario a la InterfaceUsuario
Maneja el evento "registrar"
Solicita CrearRegistroUsuarioFuncionario a la InterfaceBasedeDatosRegistro
Solicita funcionarioregistrado

Maneja el evento "Salir"	
Sale del sistema	
Solicita FuncionariosRegistrados a la InterfaceBasedeDatosRegistro	
Solicita DesplegarPantallaFuncionariosRegistrados a la InterfaceUsuario	
Maneja el evento "actualizar"	
Solicita ActualizarRegistroUsuarioFuncionario a la InterfaceBasedeDatosRegistro	
Maneja el evento "eliminar"	
Solicita EliminarRegistroUsuarioFuncionario a la InterfaceBasedeDatosRegistro	

Tarjeta para la clase PantallaRegistrarFuncionario con responsabilidades.

Clase: PantallaRegistrarFuncionario	
Descripción: pantalla crear funcionarios (p32)	
Modulo: registro, usuario funcionario	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
despliega	
Envía el evento "registrar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase PantallaFuncionarioRegistrado con responsabilidades.

Clase: PantallaFuncionarioRegistrado	
Descripción: pantalla funcionarios registrados (p33)	
Modulo: registro, usuario funcionario	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar" a la InterfaceUsuario	
Envía el evento "eliminar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase RegistrarUsuarioFuncionario con responsabilidades.

Clase: RegistrarUsuarioFuncionario	
Descripción: el usuario para que pueda utilizar el sistema, debe estar registrado, en el registro se debe ingresar la información del usuario: nombre, apellido, teléfono, email, login y password.	
Modulo: registro, usuario funcionario	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Usuarios estudiantes:

Este módulo se compone de las clases ManejadorEstudiantes, PantallaRegistrarEstudiante, PantallaEstudianteRegistrado y RegistrarUsuarioEstudiantes.

Tarjeta para la clase ManejadorEstudiantes con responsabilidades.

Clase: ManejadorEstudiantes	
Descripción: se encarga de todo lo relacionado con el registro del usuario estudiante	
Modulo: RegistrarUsuarioEstudiantes	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaRegistroEstudiantes a la InterfaceUsuario	
Maneja el evento "registrar"	
Solicita CrearRegistroUsuarioEstudiante a la InterfaceBasedeDatosRegistro	
Solicita EstudianteRegistrado	
Maneja el evento "Salir"	
Sale del sistema	

Solicita Estudiantes Registrados a la Interface Base de Datos Registro	
Solicita Desplegar Pantalla Estudiantes Registrados a la Interface Usuario	
Maneja el evento "actualizar"	
Solicita Actualizar Registro Usuario Estudiante a la Interface Base de Datos Registro	
Maneja el evento "eliminar"	
Solicita Eliminar Registro Usuario Estudiante a la Interface Base de Datos Registro	

Tarjeta para la clase Pantalla Registrar Estudiante con responsabilidades.

Clase: Pantalla Registrar Estudiante	
Descripción: pantalla registro estudiantes (p4)	
Modulo: registro, usuario estudiante	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "registrar" a la Interface Usuario	
Envía el evento "salir" a la Interface Usuario	

Tarjeta para la clase Pantalla Estudiante Registrado con responsabilidades.

Clase: Pantalla Estudiante Registrado	
Descripción: pantalla estudiantes registrados (p3)	
Modulo: registro, usuarios estudiantes	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar" a la Interface Usuario	
Envía el evento "eliminar" a la Interface Usuario	
Envía el evento "salir" a la Interface Usuario	

Tarjeta para la clase RegistrarUsuarioEstudiantes con responsabilidades.

Clase: RegistrarUsuarioEstudiantes	
Descripción: en el registro se debe ingresar la información del usuario: N° carpeta, fecha de radicación, beneficio 1, beneficio 2, 1° vez, renovación, reintegro, primer apellido, segundo apellido, primer nombre, segundo nombre, tipo de documento, numero de documento Expedido en, edad, población especial, genero, teléfono 1; teléfono 2, dirección, correo electrónico, antigüedad Sisbén en años, puntaje del Sisbén, recibe algún otro beneficio, universidad sede nivel carrera, modalidad, semestre, valor del semestre, porcentaje a aplicar monto aprobado, otro beneficio, materias perdidas promedio 2018_1 y 2017, horas sociales cumplidas, dependencia donde realizo horas sociales, Horario, ciudad a la que viaja, días que viaja, valor por 2 recorridos valor semanal, valor mensual bruto , valor a pagar por mes, meses a girar y total semestre; también los datos del representante legal si es un menor de edad.	
Modulo: registro, usuarios estudiantes	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Beneficio1:

Este módulo se compone de las clases ManejadorBeneficio1, PantallaRegistrarBeneficio1, PantallaBeneficio1 y RegistrarBeneficio1.

Tarjeta para la clase ManejadorBeneficio1 con responsabilidades.

Clase: ManejadorBeneficio1	
Descripción: se encarga de todo lo relacionado con el registro del beneficio1	
Modulo: RegistrarBeneficio1	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaBeneficio1 a la InterfaceUsuario	

Maneja el evento "registrar"	
Solicita CrearRegistroBeneficio1 a la InterfaceBasedeDatosRegistro	
Solicita Beneficio1	
Maneja el evento "Salir"	
Sale del sistema	
Solicita Beneficio1 a la InterfaceBasedeDatosRegistro	
Solicita DesplegarPantallaBeneficio1 a la InterfaceUsuario	
Maneja el evento "actualizar"	
Solicita ActualizarRegistroBeneficio1 a la InterfaceBasedeDatosRegistro	
Maneja el evento "eliminar"	
Solicita EliminarRegistroBeneficio1 a la InterfaceBasedeDatosRegistro	

Tarjeta para la clase PantallaRegistrarBeneficio1 con responsabilidades.

Clase: PantallaRegistrarBeneficio1	
Descripción: pantalla registro beneficio 1 (p7)	
Modulo: registro, beneficio1	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "registrar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase PantallaBeneficio1 con responsabilidades.

Clase: PantallaBeneficio1	
Descripción: pantalla beneficio1 (p6)	
Modulo: registro, beneficio1	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar" a la InterfaceUsuario	

Envía el evento “eliminar” a la InterfaceUsuario	
Envía el evento “salir” a la InterfaceUsuario	

Tarjeta para la clase RegistrarBeneficio1 con responsabilidades.

Clase: RegistrarBeneficio1	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, beneficio1	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Beneficio2:

Este módulo se compone de las clases ManejadorBeneficio2, PantallaRegistrarBeneficio2, PantallaBeneficio2 y RegistrarBeneficio2.

Tarjeta para la clase ManejadorBeneficio2 con responsabilidades.

Clase: ManejadorBeneficio2	
Descripción: se encarga de todo lo relacionado con el registro del beneficio2	
Modulo: RegistrarBeneficio2	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaBeneficio2 a la InterfaceUsuario	
Maneja el evento “registrar”	
Solicita CrearRegistroBeneficio2 a la InterfaceBasedeDatosRegistro	
Solicita Beneficio2	
Maneja el evento “Salir”	
Sale del sistema	
Solicita Beneficio2 a la InterfaceBasedeDatosRegistro	

Solicita DesplegarPantallaBeneficio2 a la InterfaceUsuario	
Maneja el evento "actualizar"	
Solicita ActualizarRegistroBeneficio2 a la InterfaceBasedeDatosRegistro	
Maneja el evento "eliminar"	
Solicita EliminarRegistroBeneficio2 a la InterfaceBasedeDatosRegistro	

Tarjeta para la clase PantallaRegistrarBeneficio2 con responsabilidades.

Clase: PantallaRegistrarBeneficio2	
Descripción: pantalla registro beneficio2 (p10)	
Modulo: registro, beneficio2	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "registrar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase PantallaBeneficio2 con responsabilidades.

Clase: PantallaBeneficio2	
Descripción: pantalla beneficio2 (p9)	
Modulo: registro, beneficio2	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar" a la InterfaceUsuario	
Envía el evento "eliminar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase RegistrarBeneficio2 con responsabilidades.

Clase: RegistrarBeneficio2	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, beneficio2	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Algún Otro Beneficio:

Este módulo se compone de las clases ManejadorAlgúnOtroBeneficio, PantallaRegistrarAlgúnOtroBeneficio, PantallaAlgúnOtroBeneficio, y RegistrarAlgúnOtroBeneficio.

Tarjeta para la clase ManejadorAlgúnOtroBeneficio con responsabilidades.

Clase: ManejadorAlgúnOtroBeneficio	
Descripción: se encarga de todo lo relacionado con el registro del beneficio2	
Modulo: RegistrarAlgúnOtroBeneficio	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaAlgúnOtroBeneficio a la InterfaceUsuario	
Maneja el evento "registrar"	
Solicita CrearRegistroAlgúnOtroBeneficio a la InterfaceBasedeDatosRegistro	
Solicita Algúnotrobeneficio	
Maneja el evento "Salir"	
Sale del sistema	
Solicita Algúnotrobeneficio a la InterfaceBasedeDatosRegistro	
Solicita DesplegarPantallaAlgúnOtroBeneficio a la InterfaceUsuario	

Maneja el evento "actualizar"	
Solicita ActualizarRegistroAlgunOtroBeneficio a la InterfaceBasedeDatosRegistro	
Maneja el evento "eliminar"	
Solicita EliminarRegistroAlgunOtroBeneficio a la InterfaceBasedeDatosRegistro	

Tarjeta para la clase PantallaRegistrarAlgunOtroBeneficio con responsabilidades.

Clase: PantallaRegistrarAlgunOtroBeneficio	
Descripción: pantalla registro algún otro beneficio (p13)	
Modulo: registro, Algunotrobeneficio	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "registrar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase PantallaAlgunOtroBeneficio con responsabilidades.

Clase: PantallaAlgunOtroBeneficio	
Descripción: pantalla algún otro beneficio (p12)	
Modulo: registro, Algunotrobeneficio	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar" a la InterfaceUsuario	
Envía el evento "eliminar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase RegistrarAlgunOtroBeneficio con responsabilidades.

Clase: RegistrarAlgunOtroBeneficio	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, Algunotrobeficio	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Antigüedad Sisbén:

Este módulo se compone de las clases ManejadorAntigüedadSisben, PantallaRegistroAntigüedadSisben, PantallaAntigüedadSisben, y RegistrarAntigüedadSisben.

Tarjeta para la clase ManejadorAntigüedadSisben con responsabilidades.

Clase: ManejadorAntigüedadSisben	
Descripción: se encarga de todo lo relacionado con el registro de la antigüedad del Sisbén	
Modulo: RegistrarAntigüedadSisben	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaAntiguedadsisben	
Maneja el evento "registrar"	
Solicita CrearRegistroAntiguedadsisben a la InterfaceBasedeDatosRegistro	
Solicita antigüedadSisbén	
Maneja el evento "Salir"	
Sale del sistema	
Solicita antigüedadSisbén a la InterfaceBasedeDatosRegistro	
Solicita DesplegarPantallaAntiguedadsisben a la InterfaceUsuario	
Maneja el evento "actualizar"	

Solicita ActualizarRegistroantigüedadSisbén a la InterfaceBasedeDatosRegistro	
Maneja el evento "eliminar"	
Solicita EliminarRegistroantigüedadSisbén a la InterfaceBasedeDatosRegistro	

Tarjeta para la clase PantallaRegistroAntigüedadSisben con responsabilidades.

Clase: PantallaRegistroAntigüedadSisben	
Descripción: pantalla registro antigüedad Sisbén (p16)	
Modulo: registro, AntigüedadSisbén	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "registrar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase PantallaAntigüedadSisben con responsabilidades.

Clase: PantallaAntigüedadSisben	
Descripción: pantalla antigüedad Sisbén (p15)	
Modulo: registro, AntigüedadSisbén	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar" a la InterfaceUsuario	
Envía el evento "eliminar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase RegistrarAntigüedadSisben con responsabilidades.

Clase: RegistrarAntigüedadSisben	
Descripción: en el registro se debe ingresar la información de la antigüedad Sisbén: cantidad de años.	
Modulo: registro, AntigüedadSisbén	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tipos de poblacion:

Este módulo se compone de las clases ManejadorTipoPoblacion, PantallaRegistroTiposPoblacion, PantallaTiposPoblacion, y RegistrarTipoPoblacion.

Tarjeta para la clase ManejadorTipoPoblacion con responsabilidades.

Clase: ManejadorTipoPoblacion	
Descripción: se encarga de todo lo relacionado con el registro del tipo de poblacion.	
Modulo: RegistrarTipoPoblacion	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaTipoPoblacion	
Maneja el evento "registrar"	
Solicita CrearRegistroTipoPoblacion a la InterfaceBasedeDatosRegistro	
Solicita TipoPoblacion	
Maneja el evento "Salir"	
Sale del sistema	
Solicita TipoPoblacion a la InterfaceBasedeDatosRegistro	
Solicita DesplegarPantallaTipoPoblacion a la InterfaceUsuario	
Maneja el evento "actualizar"	

Solicita ActualizarRegistroTipoPoblacion a la InterfaceBasedeDatosRegistro	
Maneja el evento "eliminar"	
Solicita EliminarRegistroTipoPoblacion a la InterfaceBasedeDatosRegistro	

Tarjeta para la clase PantallaRegistroTiposPoblacion con responsabilidades.

Clase: PantallaRegistroTiposPoblacion	
Descripción: pantalla registro tipos de poblacion (p19)	
Modulo: registro, TipoPoblacion	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "registrar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase PantallaTiposPoblacion con responsabilidades.

Clase: PantallaTiposPoblacion	
Descripción: pantalla tipos de poblacion (p18)	
Modulo: registro, TipoPoblacion	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar" a la InterfaceUsuario	
Envía el evento "eliminar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase RegistrarTipoPoblacion con responsabilidades.

Clase: RegistrarTipoPoblacion	
Descripción: en el registro se debe ingresar la información del tipo de poblacion: nombre.	
Modulo: registro, TipoPoblacion	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	

Atributos:	

Universidades:

Este módulo se compone de las clases ManejadorUniversidades, PantallaRegistroUniversidad, PantallaUniversidades, y RegistrarUniversidades.

Tarjeta para la clase ManejadorUniversidades con responsabilidades.

Clase: ManejadorUniversidades	
Descripción: se encarga de todo lo relacionado con el registro de las universidades.	
Modulo: RegistrarUniversidades	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaUniversidad	
Maneja el evento "registrar"	
Solicita CrearRegistroUniversidad a la InterfaceBasedeDatosRegistro	
Solicita Universidades	
Maneja el evento "Salir"	
Sale del sistema	
Solicita Universidades a la InterfaceBasedeDatosRegistro	
Solicita DesplegarPantallaUniversidad a la InterfaceUsuario	
Maneja el evento "actualizar"	
Solicita ActualizarRegistroUniversidad a la InterfaceBasedeDatosRegistro	
Maneja el evento "eliminar"	
Solicita EliminarRegistroUniversidad a la InterfaceBasedeDatosRegistro	

Tarjeta para la clase PantallaRegistroUniversidad con responsabilidades.

Clase: PantallaRegistroUniversidad	
Descripción: pantalla registro universidades (p22)	
Modulo: registro, Universidad	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "registrar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase PantallaUniversidades con responsabilidades.

Clase: PantallaUniversidades	
Descripción: pantalla universidades (p21)	
Modulo: registro, Universidad	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar" a la InterfaceUsuario	
Envía el evento "eliminar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase RegistrarUniversidades con responsabilidades.

Clase: RegistrarUniversidades	
Descripción: en el registro se debe ingresar la información de la universidad: nombre y porcentaje a aplicar.	
Modulo: registro, Universidad	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Carreras:

Este módulo se compone de las clases ManejadorCarreras, PantallaRegistroCarreras, PantallaCarreras, y RegistrarCarreras.

Tarjeta para la clase ManejadorCarreras con responsabilidades.

Clase: ManejadorCarreras	
Descripción: se encarga de todo lo relacionado con el registro de las carreras.	
Modulo: RegistrarCarreras	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaCarreras	
Maneja el evento "registrar"	
Solicita CrearRegistroCarreras a la InterfaceBasedeDatosRegistro	
Solicita Carreras	
Maneja el evento "Salir"	
Sale del sistema	
Solicita Carreras a la InterfaceBasedeDatosRegistro	
Solicita DesplegarPantallaCarreras a la InterfaceUsuario	
Maneja el evento "actualizar"	
Solicita ActualizarRegistroCarreras a la InterfaceBasedeDatosRegistro	
Maneja el evento "eliminar"	
Solicita EliminarRegistroCarreras a la InterfaceBasedeDatosRegistro	

Tarjeta para la clase PantallaRegistroCarreras con responsabilidades.

Clase: PantallaRegistroCarreras
Descripción: pantalla registro carreras (p25)
Modulo: registro, Carreras
Estereotipo: borde
Propiedades:
Superclases:

Subclases:	
Atributos:	
Envía el evento "registrar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase PantallaCarreras con responsabilidades.

Clase: PantallaCarreras	
Descripción: pantalla carreras (p24)	
Modulo: registro, Carreras	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar" a la InterfaceUsuario	
Envía el evento "eliminar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase RegistrarCarreras con responsabilidades.

Clase: RegistrarCarreras	
Descripción: en el registro se debe ingresar la información de la carrera: nombre.	
Modulo: registro, Carreras	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Ciudad:

Este módulo se compone de las clases ManejadorCiudad, PantallaRegistroCiudad, PantallaCiudad, y RegistrarCiudad.

Tarjeta para la clase ManejadorCiudad con responsabilidades.

Clase: ManejadorCiudad	
Descripción: se encarga de todo lo relacionado con el registro de la ciudad.	
Modulo: RegistrarCiudad	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaCiudad	
Maneja el evento "registrar"	
Solicita CrearRegistroCiudad a la InterfaceBasedeDatosRegistro	
Solicita Ciudad	
Maneja el evento "Salir"	
Sale del sistema	
Solicita Ciudad a la InterfaceBasedeDatosRegistro	
Solicita DesplegarPantallaCiudad a la InterfaceUsuario	
Maneja el evento "actualizar"	
Solicita ActualizarRegistroCiudad a la InterfaceBasedeDatosRegistro	
Maneja el evento "eliminar"	
Solicita EliminarRegistroCiudad a la InterfaceBasedeDatosRegistro	

Tarjeta para la clase PantallaRegistroCiudad con responsabilidades.

Clase: PantallaRegistroCiudad
Descripción: pantalla registro ciudad (p28)
Modulo: registro, Ciudad
Estereotipo: borde
Propiedades:
Superclases:

Subclases:	
Atributos:	
Envía el evento "registrar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase PantallaCiudad con responsabilidades.

Clase: PantallaCiudad	
Descripción: pantalla ciudad (p27)	
Modulo: registro, Ciudad	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar" a la InterfaceUsuario	
Envía el evento "eliminar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase RegistrarCiudad con responsabilidades.

Clase: RegistrarCiudad	
Descripción: en el registro se debe ingresar la información de la ciudad: nombre.	
Modulo: registro, Ciudad	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Certificación estudiantes:

Este módulo se compone de las clases ManejadorDocumentos, PantallaDocumentos, PantallaCertificado, PantallaEstadísticas, CertificadoEstudiante y Estadísticas.

Tarjeta para la clase ManejadorDocumentos con responsabilidades.

Clase: ManejadorDocumentos	
Descripción: se encarga de todo lo relacionado con informes como estadísticas y certificados.	
Modulo: CertificadoEstudiante y Estadísticas.	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaDocumentos	
Maneja el evento "Salir"	
Sale del sistema	
Maneja el evento "certificado"	
Solicita PedirCertificado a la InterfaceBasedeDatosRegistro	
Maneja el evento "estadísticas"	
Solicita Estadísticas a la InterfaceBasedeDatosRegistro	

Tarjeta para la clase PantallaDocumentos con responsabilidades.

Clase: PantallaDocumentos	
Descripción: pantalla registro ciudad (p34)	
Modulo: Certificados y Estadísticas.	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "certificado" a la InterfaceUsuario	
Envía el evento "estadísticas" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase PantallaCertificado con responsabilidades.

Clase: PantallaCertificado	
Descripción: pantalla certificados (p31)	
Modulo: Certificados.	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "aceptar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase PantallaEstadisticas con responsabilidades.

Clase: PantallaEstadisticas	
Descripción: pantalla estadisticas (p30)	
Modulo: Estadisticas.	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "aceptar" a la InterfaceUsuario	
Envía el evento "salir" a la InterfaceUsuario	

Tarjeta para la clase CertificadoEstudiante con responsabilidades.

Clase: CertificadoEstudiante	
Descripción: en el certificado se debe ingresar la información del estudiante: documento de identidad.	
Modulo: registro, Certificados	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase Estadísticas con responsabilidades.

Clase: Estadísticas	
Descripción: en estadísticas se toman los datos de los estudiantes registrado en la base de datos.	
Modulo: registro, Estadísticas	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Interface base de datos:

Tarjeta para la clase InterfaceBasedeDatosRegistro con responsabilidades.

Clase: InterfaceBasedeDatosRegistro	
Descripción: la información de los usuarios, beneficios y otros se almacena en la base de datos foest	
Modulo: Registro interfaceBD	
Estereotipo: interface	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita ValidarRegistroUsuarioFuncionario a la BasedeDatosfoest	
Solicita CrearRegistroUsuarioFuncionario a la BasedeDatosfoest	
Solicita Funcionarioregistrado a la BasedeDatosfoest	
Solicita ActualizarRegistroUsuarioFuncionario a la BasedeDatosfoest	
Solicita EliminarRegistroUsuarioFuncionario a la BasedeDatosfoest	
Solicita CrearRegistroUsuarioEstudiante a la BasedeDatosfoest	
Solicita EstudianteRegistrado a la BasedeDatosfoest	
Solicita ActualizarRegistroUsuarioEstudiante a la BasedeDatosfoest	

Solicita EliminarRegistroUsuarioEstudiante a la BasedeDatosfoest	
Solicita CrearRegistroBeneficio1 a la BasedeDatosfoest	
Solicita beneficio1 a la BasedeDatosfoest	
Solicita ActualizarRegistroBeneficio1 a la BasedeDatosfoest	
Solicita EliminarRegistrobeneficio1 a la BasedeDatosfoest	
Solicita CrearRegistroBeneficio2 a la BasedeDatosfoest	
Solicita beneficio2 a la BasedeDatosfoest	
Solicita ActualizarRegistroBeneficio2 a la BasedeDatosfoest	
Solicita EliminarRegistrobeneficio2 a la BasedeDatosfoest	
Solicita CrearRegistroAlgunOtroBeneficio a la BasedeDatosfoest	
Solicita Algunotrobeneficio a la BasedeDatosfoest	
Solicita ActualizarRegistroAlgunOtroBeneficio a la BasedeDatosfoest	
Solicita EliminarRegistroAlgunOtroBeneficio a la BasedeDatosfoest	
Solicita CrearRegistroAntiguedadsisben a la BasedeDatosfoest	
Solicita AntigüedadSisbén a la BasedeDatosfoest	
Solicita ActualizarRegistroantigüedadSisbén a la BasedeDatosfoest	
Solicita EliminarRegistroantigüedadSisbén a la BasedeDatosfoest	
Solicita CrearRegistroTipoPoblacion a la BasedeDatosfoest	
Solicita TipoPoblacion a la BasedeDatosfoest	
Solicita ActualizarRegistroTipoPoblacion a la BasedeDatosfoest	
Solicita EliminarRegistroTipoPoblacion a la BasedeDatosfoest	
Solicita CrearRegistroUniversidad a la BasedeDatosfoest	
Solicita universidades a la BasedeDatosfoest	
Solicita ActualizarRegistroUniversidad a la BasedeDatosfoest	
Solicita EliminarRegistroUniversidad a la BasedeDatosfoest	

Solicita CrearRegistroCarreras a la BasedeDatosfoest	
Solicita Carreras a la BasedeDatosfoest	
Solicita ActualizarRegistroCarreras a la BasedeDatosfoest	
Solicita EliminarRegistroCarreras a la BasedeDatosfoest	
Solicita CrearRegistroCiudad a la BasedeDatosfoest	
Solicita Ciudad a la BasedeDatosfoest	
Solicita ActualizarRegistroCiudad a la BasedeDatosfoest	
Solicita EliminarRegistroCiudad a la BasedeDatosfoest	
Solicita PedirCertificado a la BasedeDatosfoest	
Solicita estadísticas a la BasedeDatosfoest	

Estudiantes foest:

Tarjeta para la clase ManejadorFoest con responsabilidades.

Clase: ManejadorFoest	
Descripción: se encarga de enviar las peticiones particulares de opciones a los manejadores especializados para registro de las diferentes opciones.	
Modulo: estudiantes	
Estereotipo: control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaPrincipal a la InterfaceUsuario	
Maneja el evento "Administrar registro funcionario"	
Solicita registrar al ManejadorRegistroFuncionario	
Maneja el evento "Administrar registro estudiante"	
Solicita registrar al ManejadorRegistrarEstudiantes	
Maneja el evento "Administrar registro Beneficio1"	
Solicita registrar al ManejadorRegistroBeneficio1	
Maneja el evento "Administrar registro Beneficio2"	
Solicita registrar al ManejadorRegistroBeneficio2	

Maneja el evento "Administrar registro Algún otro Beneficio"	
Solicita registrar al ManejadorAlgúnOtroBeneficio	
Maneja el evento "Administrar registro Antigüedad Sisbén"	
Solicita registrar al ManejadorAntigüedadSisben	
Maneja el evento "Administrar registro Tipos de poblacion"	
Solicita registrar al ManejadorTipoPoblacion	
Maneja el evento "Administrar registro universidades"	
Solicita registrar al ManejadorUniversidades	
Maneja el evento "Administrar registro carrera"	
Solicita registrar al ManejadorCarreras	
Maneja el evento "Administrar registro ciudad"	
Solicita registrar al ManejadorCiudad	
Maneja el evento "Administrar documentos"	
Solicita registrar al ManejadorDocumentos	
Maneja el evento "salir"	
Sale del sistema	

Pantalla principal:

Tarjeta para la clase pantalla principal con responsabilidades.

Clase: PantallaPrincipal	
Descripción: pantalla principal (P2)	
Modulo: Principal	
Estereotipo: Borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "Administrar registro funcionario" a la InterfaceUsuario	
Envía el evento "Administrar registro estudiante" a la InterfaceUsuario	
Envía el evento "Administrar registro Beneficio1" a la InterfaceUsuario	
Envía el evento "Administrar registro Beneficio2" a la InterfaceUsuario	
Envía el evento "Administrar registro Algún otro Beneficio" a la InterfaceUsuario	

Envía el evento “Administrar registro Antigüedad Sisbén” a la InterfaceUsuario	
Envía el evento “Administrar registro Tipos de poblacion” a la InterfaceUsuario	
Envía el evento “Administrar registro universidades” a la InterfaceUsuario	
Envía el evento “Administrar registro carrera” a la InterfaceUsuario	
Envía el evento “Administrar registro ciudad” a la InterfaceUsuario	
Envía el evento “Administrar documentos” a la InterfaceUsuario	
Envía el evento “salir” a la InterfaceUsuario	

8.1.3 Colaboraciones

Interface usuario:

Tarjeta para la clase InterfaceUsuario con responsabilidades y colaboraciones.

Clase: InterfaceUsuario	
Descripción: toda la interacción con el usuario se realiza por medio de la interface de usuario	
Modulo: InterfaceUsuario	
Estereotipo: Borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Despliega	PantallaIngreso
Envía el evento “ingresar”	ManejadorPrincipal
Envía el evento “OK”	ManejadorPrincipal
Envía el evento “salir”	ManejadorPrincipal
Despliega	PantallaPrincipal
Envía el evento “administrar usuario funcionario”	ManejadorFoest
Envía el evento “administrar usuario estudiante”	ManejadorFoest
Envía el evento “administrar beneficio1”	ManejadorFoest
Envía el evento “administrar beneficio2”	ManejadorFoest

Envía el evento “administrar algún otro beneficio”	ManejadorFoest
Envía el evento “administrar antigüedad Sisbén”	ManejadorFoest
Envía el evento “administrar tipos de poblacion”	ManejadorFoest
Envía el evento “administrar universidades”	ManejadorFoest
Envía el evento “administrar carreras”	ManejadorFoest
Envía el evento “administrar ciudad”	ManejadorFoest
Envía el evento “administrar documentos”	ManejadorFoest
Envía el evento “salir”	ManejadorFoest
Despliega	PantallaFuncionarioRegistrado
Envía el evento “registrar”	ManejadorFuncionario
Envía el evento “modificar”	ManejadorFuncionario
Envía el evento “eliminar”	ManejadorFuncionario
Envía el evento “salir”	ManejadorFuncionario
Despliega	PantallaEstudianteRegistrado
Envía el evento “registrar”	ManejadorEstudiantes
Envía el evento “modificar”	ManejadorEstudiantes
Envía el evento “eliminar”	ManejadorEstudiantes
Envía el evento “salir”	ManejadorEstudiantes
Despliega	PantallaBeneficio1
Envía el evento “registrar”	ManejadorBeneficio1
Envía el evento “modificar”	ManejadorBeneficio1
Envía el evento “eliminar”	ManejadorBeneficio1
Envía el evento “salir”	ManejadorBeneficio1
Despliega	PantallaBeneficio2
Envía el evento “registrar”	ManejadorBeneficio2
Envía el evento “modificar”	ManejadorBeneficio2
Envía el evento “eliminar”	ManejadorBeneficio2
Envía el evento “salir”	ManejadorBeneficio2
Despliega	PantallaAlgunOtroBeneficio
Envía el evento “registrar”	ManejadorAlgunOtroBeneficio
Envía el evento “modificar”	ManejadorAlgunOtroBeneficio
Envía el evento “eliminar”	ManejadorAlgunOtroBeneficio
Envía el evento “salir”	ManejadorAlgunOtroBeneficio
Despliega	PantallaAntiguedadSisben
Envía el evento “registrar”	ManejadorAntiguedadSisben
Envía el evento “modificar”	ManejadorAntiguedadSisben
Envía el evento “eliminar”	ManejadorAntiguedadSisben
Envía el evento “salir”	ManejadorAntiguedadSisben
Despliega	PantallaTiposPoblacion
Envía el evento “registrar”	ManejadorTipoPoblacion
Envía el evento “modificar”	ManejadorTipoPoblacion
Envía el evento “eliminar”	ManejadorTipoPoblacion
Envía el evento “salir”	ManejadorTipoPoblacion

Despliega	PantallaUniversidades
Envía el evento “registrar”	ManejadorUniversidades
Envía el evento “modificar”	ManejadorUniversidades
Envía el evento “eliminar”	ManejadorUniversidades
Envía el evento “salir”	ManejadorUniversidades
Despliega	PantallaCarreras
Envía el evento “registrar”	ManejadorCarreras
Envía el evento “modificar”	ManejadorCarreras
Envía el evento “eliminar”	ManejadorCarreras
Envía el evento “salir”	ManejadorCarreras
Despliega	PantallaCiudad
Envía el evento “registrar”	ManejadorCiudad
Envía el evento “modificar”	ManejadorCiudad
Envía el evento “eliminar”	ManejadorCiudad
Envía el evento “salir”	ManejadorCiudad
Despliega	PantallaDocumentos
Envía el evento “Pedir certificado”	ManejadorDocumentos
Envía el evento “estadísticas”	ManejadorDocumentos
Envía el evento “salir”	ManejadorDocumentos
Despliega	PantallaCertificado
Envía el evento “aceptar”	ManejadorCertificado
Envía el evento “salir”	ManejadorCertificado
Despliega	PantallaEstadísticas
Envía el evento “aceptar”	ManejadorEstadísticas
Envía el evento “salir”	ManejadorEstadísticas

ManejadorPrincipal:

Tarjeta para la clase ManejadorPrincipal con responsabilidades y colaboraciones.

Clase: ManejadorPrincipal	
Descripción: es el encargado de desplegar la pantalla principal de interacción con el usuario y después desplegar las diferentes funciones a los manejadores apropiados	
Modulo: Principal	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita desplegar pantalla	InterfaceUsuario
Solicita ValidarRegistroUsuarioFuncionario	ManejadorFuncionario
Solicita estudiantes foest	ManejadorFoest

Maneja el evento "salir"	
Sale del sistema	

Pantalla ingreso:

Tarjeta para la clase pantalla ingreso con responsabilidades y colaboraciones.

Clase: PantallaIngreso	
Descripción: pantalla ingreso (P1)	
Modulo: Principal	
Estereotipo: Borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "ingresar"	InterfaceUsuario
Envía el evento "OK"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Registro:

Este módulo se compone del módulo usuario funcionario, usuario estudiante, beneficio1, beneficio2, algún otro beneficio, antigüedad Sisbén, tipos de poblacion, universidades, carreras, ciudad, certificados estudiantes e InterfaceBasedeDatosRegistro.

Usuario funcionario:

Este módulo se compone de las clases ManejadorFuncionario, PantallaRegistrarFuncionario, PantallaFuncionarioRegistrado y RegistrarUsuarioFuncionario.

Tarjeta para la clase ManejadorFuncionario con responsabilidades y colaboraciones.

Clase: ManejadorFuncionario	
Descripción: se encarga de todo lo relacionado con el registro del usuario funcionario	
Modulo: RegistrarUsuarioFuncionario	
Estereotipo: Control	

Propiedades:	
Superclases:	
Subclases:	
Atributos:	
CrearRegistroUsuarioFuncionario	
Solicita ValidarRegistroUsuarioFuncionario	InterfaceBasedeDatosRegistro
Solicita DesplegarPantallaRegistroFuncionario	InterfaceUsuario
Maneja el evento "registrar"	
Solicita CrearRegistroUsuarioFuncionario	InterfaceBasedeDatosRegistro
Solicita funcionarioregistrado	
Maneja el evento "Salir"	
Sale del sistema	
Solicita FuncionariosRegistrados	InterfaceBasedeDatosRegistro
Solicita DesplegarPantallaFuncionariosRegistrados	InterfaceUsuario
Maneja el evento "actualizar"	
Solicita ActualizarRegistroUsuarioFuncionario	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroUsuarioFuncionario	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaRegistrarFuncionario con responsabilidades y colaboraciones.

Clase: PantallaRegistrarFuncionario	
Descripción: pantalla crear funcionarios (p32)	
Modulo: registro, usuario funcionario	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
despliega	
Envía el evento "registrar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase PantallaFuncionarioRegistrado con responsabilidades y colaboraciones.

Clase: PantallaFuncionarioRegistrado	
Descripción: pantalla funcionarios registrados (p33)	
Modulo: registro, usuario funcionario	
Estereotipo: borde	
Propiedades:	

Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar"	InterfaceUsuario
Envía el evento "eliminar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase RegistrarUsuarioFuncionario con responsabilidades y colaboraciones.

Clase: RegistrarUsuarioFuncionario	
Descripción: el usuario para que pueda utilizar el sistema, debe estar registrado, en el registro se debe ingresar la información del usuario: nombre, apellido, teléfono, email, login y password.	
Modulo: registro, usuario funcionario	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Usuarios estudiantes:

Este módulo se compone de las clases ManejadorEstudiantes, PantallaRegistrarEstudiante, PantallaEstudianteRegistrado y RegistrarUsuarioEstudiantes.

Tarjeta para la clase ManejadorEstudiantes con responsabilidades y colaboraciones.

Clase: ManejadorEstudiantes	
Descripción: se encarga de todo lo relacionado con el registro del usuario estudiante	
Modulo: RegistrarUsuarioEstudiantes	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaRegistroEstudiantes	InterfaceUsuario
Maneja el evento "registrar"	
Solicita CrearRegistroUsuarioEstudiante	InterfaceBasedeDatosRegistro

Solicita EstudianteRegistrado	
Maneja el evento "Salir"	
Sale del sistema	
Solicita EstudiantesRegistrados	InterfaceBasedeDatosRegistro
Solicita DesplegarPantallaEstudiantesRegistrados	InterfaceUsuario
Maneja el evento "actualizar"	
Solicita ActualizarRegistroUsuarioEstudiante	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroUsuarioEstudiante	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaRegistrarEstudiante con responsabilidades y colaboraciones.

Clase: PantallaRegistrarEstudiante	
Descripción: pantalla registro estudiantes (p4)	
Modulo: registro, usuario estudiante	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "registrar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase PantallaEstudianteRegistrado con responsabilidades y colaboraciones.

Clase: PantallaEstudianteRegistrado	
Descripción: pantalla estudiantes registrados (p3)	
Modulo: registro, usuarios estudiantes	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar"	InterfaceUsuario
Envía el evento "eliminar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase RegistrarUsuarioEstudiantes con responsabilidades y colaboraciones.

Clase: RegistrarUsuarioEstudiantes

Descripción: en el registro se debe ingresar la información del usuario: N° carpeta, fecha de radicación, beneficio 1, beneficio 2, 1° vez, renovación, reintegro, primer apellido, segundo apellido, primer nombre, segundo nombre, tipo de documento, numero de documento Expedido en, edad, población especial, genero, teléfono 1; teléfono 2, dirección, correo electrónico, antigüedad Sisbén en años, puntaje del Sisbén, recibe algún otro beneficio, universidad sede nivel carrera, modalidad, semestre, valor del semestre, porcentaje a aplicar monto aprobado, otro beneficio, materias perdidas promedio 2018_1 y 2017, horas sociales cumplidas, dependencia donde realizo horas sociales, Horario, ciudad a la que viaja, días que viaja, valor por 2 recorridos valor semanal, valor mensual bruto , valor a pagar por mes, meses a girar y total semestre; también los datos del representante legal si es un menor de edad.

Modulo: registro, usuarios estudiantes

Estereotipo: entidad

Propiedades:

Superclases:

Subclases:

Atributos:

Beneficio1:

Este módulo se compone de las clases ManejadorBeneficio1, PantallaRegistrarBeneficio1, PantallaBeneficio1 y RegistrarBeneficio1.

Tarjeta para la clase ManejadorBeneficio1 con responsabilidades y colaboraciones.

Clase: ManejadorBeneficio1	
Descripción: se encarga de todo lo relacionado con el registro del beneficio1	
Modulo: RegistrarBeneficio1	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaBeneficio1	InterfaceUsuario
Maneja el evento "registrar"	
Solicita CrearRegistroBeneficio1	InterfaceBasedeDatosRegistro
Solicita Beneficio1	
Maneja el evento "Salir"	
Sale del sistema	
Solicita Beneficio1	InterfaceBasedeDatosRegistro

Solicita DesplegarPantallaBeneficio1	InterfaceUsuario
Maneja el evento "actualizar"	
Solicita ActualizarRegistroBeneficio1	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroBeneficio1	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaRegistrarBeneficio1 con responsabilidades y colaboraciones.

Clase: PantallaRegistrarBeneficio1	
Descripción: pantalla registro beneficio 1 (p7)	
Modulo: registro, beneficio1	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "registrar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase PantallaBeneficio1 con responsabilidades y colaboraciones.

Clase: PantallaBeneficio1	
Descripción: pantalla beneficio1 (p6)	
Modulo: registro, beneficio1	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar"	InterfaceUsuario
Envía el evento "eliminar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase RegistrarBeneficio1 con responsabilidades y colaboraciones.

Clase: RegistrarBeneficio1	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, beneficio1	
Estereotipo: entidad	
Propiedades:	

Superclases:
Subclases:
Atributos:

Beneficio2:

Este módulo se compone de las clases ManejadorBeneficio2, PantallaRegistrarBeneficio2, PantallaBeneficio2 y RegistrarBeneficio2.

Tarjeta para la clase ManejadorBeneficio2 con responsabilidades y colaboraciones.

Clase: ManejadorBeneficio2	
Descripción: se encarga de todo lo relacionado con el registro del beneficio2	
Modulo: RegistrarBeneficio2	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaBeneficio2	InterfaceUsuario
Maneja el evento "registrar"	
Solicita CrearRegistroBeneficio2	InterfaceBasedeDatosRegistro
Solicita Beneficio2	
Maneja el evento "Salir"	
Sale del sistema	
Solicita Beneficio2	InterfaceBasedeDatosRegistro
Solicita DesplegarPantallaBeneficio2	InterfaceUsuario
Maneja el evento "actualizar"	
Solicita ActualizarRegistroBeneficio2	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroBeneficio2	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaRegistrarBeneficio2 con responsabilidades y colaboraciones.

Clase: PantallaRegistrarBeneficio2
Descripción: pantalla registro beneficio2 (p10)
Modulo: registro, beneficio2
Estereotipo: borde

Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "registrar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase PantallaBeneficio2 con responsabilidades y colaboraciones.

Clase: PantallaBeneficio2	
Descripción: pantalla beneficio2 (p9)	
Modulo: registro, beneficio2	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar"	InterfaceUsuario
Envía el evento "eliminar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase RegistrarBeneficio2 con responsabilidades y colaboraciones.

Clase: RegistrarBeneficio2	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, beneficio2	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Algún Otro Beneficio:

Este módulo se compone de las clases `ManejadorAlgúnOtroBeneficio`, `PantallaRegistrarAlgúnOtroBeneficio`, `PantallaAlgúnOtroBeneficio`, y `RegistrarAlgúnOtroBeneficio`.

Tarjeta para la clase `ManejadorAlgúnOtroBeneficio` con responsabilidades y colaboraciones.

Clase: <code>ManejadorAlgúnOtroBeneficio</code>	
Descripción: se encarga de todo lo relacionado con el registro del beneficio2	
Modulo: <code>RegistrarAlgúnOtroBeneficio</code>	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita <code>DesplegarPantallaAlgúnOtroBeneficio</code>	<code>InterfaceUsuario</code>
Maneja el evento "registrar"	
Solicita <code>CrearRegistroAlgúnOtroBeneficio</code>	<code>InterfaceBasedeDatosRegistro</code>
Solicita <code>Algúnotrobeficio</code>	
Maneja el evento "Salir"	
Sale del sistema	
Solicita <code>Algúnotrobeficio</code>	<code>InterfaceBasedeDatosRegistro</code>
Solicita <code>DesplegarPantallaAlgúnOtroBeneficio</code>	<code>InterfaceUsuario</code>
Maneja el evento "actualizar"	
Solicita <code>ActualizarRegistroAlgúnOtroBeneficio</code>	<code>InterfaceBasedeDatosRegistro</code>
Maneja el evento "eliminar"	
Solicita <code>EliminarRegistroAlgúnOtroBeneficio</code>	<code>InterfaceBasedeDatosRegistro</code>

Tarjeta para la clase `PantallaRegistrarAlgúnOtroBeneficio` con responsabilidades y colaboraciones.

Clase: <code>PantallaRegistrarAlgúnOtroBeneficio</code>	
Descripción: pantalla registro algún otro beneficio (p13)	
Modulo: registro, <code>Algúnotrobeficio</code>	
Estereotipo: borde	
Propiedades:	
Superclases:	

Subclases:	
Atributos:	
Envía el evento "registrar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase PantallaAlgunOtroBeneficio con responsabilidades y colaboraciones.

Clase: PantallaAlgunOtroBeneficio	
Descripción: pantalla algún otro beneficio (p12)	
Modulo: registro, Algunotrobeficio	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar"	InterfaceUsuario
Envía el evento "eliminar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase RegistrarAlgunOtroBeneficio con responsabilidades y colaboraciones.

Clase: RegistrarAlgunOtroBeneficio	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, Algunotrobeficio	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Antigüedad Sisbén:

Este módulo se compone de las clases ManejadorAntigüedadSisben, PantallaRegistroAntigüedadSisben, PantallaAntigüedadSisben, y RegistrarAntigüedadSisben.

Tarjeta para la clase ManejadorAntigüedadSisben con responsabilidades y colaboraciones.

Clase: ManejadorAntigüedadSisben	
Descripción: se encarga de todo lo relacionado con el registro de la antigüedad del Sisbén	
Modulo: RegistrarAntigüedadSisben	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaAntiguedadsisben	
Maneja el evento "registrar"	
Solicita CrearRegistroAntiguedadsisben	InterfaceBasedeDatosRegistro
Solicita antigüedadSisbén	
Maneja el evento "Salir"	
Sale del sistema	
Solicita antigüedadSisbén	InterfaceBasedeDatosRegistro
Solicita DesplegarPantallaAntiguedadsisben	InterfaceUsuario
Maneja el evento "actualizar"	
Solicita ActualizarRegistroantigüedadSisbén	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroantigüedadSisbén	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaRegistroAntigüedadSisben con responsabilidades y colaboraciones.

Clase: PantallaRegistroAntigüedadSisben	
Descripción: pantalla registro antigüedad Sisbén (p16)	
Modulo: registro, AntigüedadSisbén	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "registrar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase PantallaAntigüedadSisben con responsabilidades y colaboraciones.

Clase: PantallaAntigüedadSisben	
Descripción: pantalla antigüedad Sisbén (p15)	
Modulo: registro, AntigüedadSisbén	
Estereotipo: borde	

Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar"	InterfaceUsuario
Envía el evento "eliminar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase RegistrarAntigüedadSisben con responsabilidades y colaboraciones.

Clase: RegistrarAntigüedadSisben	
Descripción: en el registro se debe ingresar la información de la antigüedad Sisbén: cantidad de años.	
Modulo: registro, AntigüedadSisbén	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tipos de poblacion:

Este módulo se compone de las clases ManejadorTipoPoblacion, PantallaRegistroTiposPoblacion, PantallaTiposPoblacion, y RegistrarTipoPoblacion.

Tarjeta para la clase ManejadorTipoPoblacion con responsabilidades y colaboraciones.

Clase: ManejadorTipoPoblacion	
Descripción: se encarga de todo lo relacionado con el registro del tipo de poblacion.	
Modulo: RegistrarTipoPoblacion	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaTipoPoblacion	
Maneja el evento "registrar"	
Solicita CrearRegistroTipoPoblacion	InterfaceBasedeDatosRegistro

Solicita TipoPoblacion	
Maneja el evento "Salir"	
Sale del sistema	
Solicita TipoPoblacion	InterfaceBasedeDatosRegistro
Solicita DesplegarPantallaTipoPoblacion	InterfaceUsuario
Maneja el evento "actualizar"	
Solicita ActualizarRegistroTipoPoblacion	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroTipoPoblacion	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaRegistroTiposPoblacion con responsabilidades y colaboraciones.

Clase: PantallaRegistroTiposPoblacion	
Descripción: pantalla registro tipos de poblacion (p19)	
Modulo: registro, TipoPoblacion	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "registrar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase PantallaTiposPoblacion con responsabilidades y colaboraciones.

Clase: PantallaTiposPoblacion	
Descripción: pantalla tipos de poblacion (p18)	
Modulo: registro, TipoPoblacion	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar"	InterfaceUsuario
Envía el evento "eliminar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase RegistrarTipoPoblacion con responsabilidades y colaboraciones.

Clase: RegistrarTipoPoblacion	
Descripción: en el registro se debe ingresar la información del tipo de poblacion: nombre.	

Modulo: registro, TipoPoblacion
Estereotipo: entidad
Propiedades:
Superclases:
Subclases:
Atributos:

Universidades:

Este módulo se compone de las clases ManejadorUniversidades, PantallaRegistroUniversidad, PantallaUniversidades, y RegistrarUniversidades.

Tarjeta para la clase ManejadorUniversidades con responsabilidades y colaboraciones.

Clase: ManejadorUniversidades	
Descripción: se encarga de todo lo relacionado con el registro de las universidades.	
Modulo: RegistrarUniversidades	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaUniversidad	
Maneja el evento "registrar"	
Solicita CrearRegistroUniversidad	InterfaceBasedeDatosRegistro
Solicita Universidades	
Maneja el evento "Salir"	
Sale del sistema	
Solicita Universidades	InterfaceBasedeDatosRegistro
Solicita DesplegarPantallaUniversidad	InterfaceUsuario
Maneja el evento "actualizar"	
Solicita ActualizarRegistroUniversidad	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroUniversidad	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaRegistroUniversidad con responsabilidades y colaboraciones.

Clase: PantallaRegistroUniversidad	
Descripción: pantalla registro universidades (p22)	
Modulo: registro, Universidad	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "registrar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase PantallaUniversidades con responsabilidades y colaboraciones.

Clase: PantallaUniversidades	
Descripción: pantalla universidades (p21)	
Modulo: registro, Universidad	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar"	InterfaceUsuario
Envía el evento "eliminar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase RegistrarUniversidades con responsabilidades y colaboraciones.

Clase: RegistrarUniversidades	
Descripción: en el registro se debe ingresar la información de la universidad: nombre y porcentaje a aplicar.	
Modulo: registro, Universidad	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Carreras:

Este módulo se compone de las clases ManejadorCarreras, PantallaRegistroCarreras, PantallaCarreras, y RegistrarCarreras.

Tarjeta para la clase ManejadorCarreras con responsabilidades y colaboraciones.

Clase: ManejadorCarreras	
Descripción: se encarga de todo lo relacionado con el registro de las carreras.	
Modulo: RegistrarCarreras	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaCarreras	
Maneja el evento "registrar"	
Solicita CrearRegistroCarreras	InterfaceBasedeDatosRegistro
Solicita Carreras	
Maneja el evento "Salir"	
Sale del sistema	
Solicita Carreras	InterfaceBasedeDatosRegistro
Solicita DesplegarPantallaCarreras	InterfaceUsuario
Maneja el evento "actualizar"	
Solicita ActualizarRegistroCarreras	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroCarreras	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaRegistroCarreras con responsabilidades y colaboraciones.

Clase: PantallaRegistroCarreras	
Descripción: pantalla registro carreras (p25)	
Modulo: registro, Carreras	
Estereotipo: borde	
Propiedades:	
Superclases:	

Subclases:	
Atributos:	
Envía el evento "registrar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase PantallaCarreras con responsabilidades y colaboraciones.

Clase: PantallaCarreras	
Descripción: pantalla carreras (p24)	
Modulo: registro, Carreras	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "actualizar"	InterfaceUsuario
Envía el evento "eliminar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase RegistrarCarreras con responsabilidades y colaboraciones.

Clase: RegistrarCarreras	
Descripción: en el registro se debe ingresar la información de la carrera: nombre.	
Modulo: registro, Carreras	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Ciudad:

Este módulo se compone de las clases ManejadorCiudad, PantallaRegistroCiudad, PantallaCiudad, y RegistrarCiudad.

Tarjeta para la clase ManejadorCiudad con responsabilidades y colaboraciones.

Clase: ManejadorCiudad	
Descripción: se encarga de todo lo relacionado con el registro de la ciudad.	
Modulo: RegistrarCiudad	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaCiudad	
Maneja el evento "registrar"	
Solicita CrearRegistroCiudad	InterfaceBasedeDatosRegistro
Solicita Ciudad	
Maneja el evento "Salir"	
Sale del sistema	
Solicita Ciudad	InterfaceBasedeDatosRegistro
Solicita DesplegarPantallaCiudad	InterfaceUsuario
Maneja el evento "actualizar"	
Solicita ActualizarRegistroCiudad	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroCiudad	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaRegistroCiudad con responsabilidades y colaboraciones.

Clase: PantallaRegistroCiudad	
Descripción: pantalla registro ciudad (p28)	
Modulo: registro, Ciudad	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "registrar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase PantallaCiudad con responsabilidades y colaboraciones.

Clase: PantallaCiudad	
Descripción: pantalla ciudad (p27)	
Modulo: registro, Ciudad	
Estereotipo: borde	
Propiedades:	
Superclases:	

Subclases:	
Atributos:	
Envía el evento "actualizar"	InterfaceUsuario
Envía el evento "eliminar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase RegistrarCiudad con responsabilidades y colaboraciones.

Clase: RegistrarCiudad	
Descripción: en el registro se debe ingresar la información de la ciudad: nombre.	
Modulo: registro, Ciudad	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Certificación estudiantes:

Este módulo se compone de las clases ManejadorDocumentos, PantallaDocumentos, PantallaCertificado, PantallaEstadísticas, CertificadoEstudiante y Estadísticas.

Tarjeta para la clase ManejadorDocumentos con responsabilidades y colaboraciones.

Clase: ManejadorDocumentos	
Descripción: se encarga de todo lo relacionado con informes como estadísticas y certificados.	
Modulo: CertificadoEstudiante y Estadísticas.	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaDocumentos	
Maneja el evento "Salir"	
Sale del sistema	
Maneja el evento "certificado"	
Solicita PedirCertificado	InterfaceBasedeDatosRegistro
Maneja el evento "estadísticas"	

Solicita Estadísticas	InterfaceBasedeDatosRegistro
-----------------------	------------------------------

Tarjeta para la clase PantallaDocumentos con responsabilidades y colaboraciones.

Clase: PantallaDocumentos	
Descripción: pantalla registro ciudad (p34)	
Modulo: Certificados y Estadísticas.	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "certificado"	InterfaceUsuario
Envía el evento "estadísticas"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase PantallaCertificado con responsabilidades y colaboraciones.

Clase: PantallaCertificado	
Descripción: pantalla certificados (p31)	
Modulo: Certificados.	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "aceptar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase PantallaEstadísticas con responsabilidades y colaboraciones.

Clase: PantallaEstadísticas	
Descripción: pantalla estadísticas (p30)	
Modulo: Estadísticas.	
Estereotipo: borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Envía el evento "aceptar"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

Tarjeta para la clase CertificadoEstudiante con responsabilidades y colaboraciones.

Clase: CertificadoEstudiante	
Descripción: en el certificado se debe ingresar la información del estudiante: documento de identidad.	
Modulo: registro, Certificados	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Tarjeta para la clase Estadisticas con responsabilidades y colaboraciones.

Clase: Estadisticas	
Descripción: en estadísticas se toman los datos de los estudiantes registrado en la base de datos.	
Modulo: registro, Estadísticas	
Estereotipo: entidad	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Interface base de datos:

Tarjeta para la clase InterfaceBasedeDatosRegistro con responsabilidades y colaboraciones.

Clase: InterfaceBasedeDatosRegistro	
Descripción: la información de los usuarios, beneficios y otros se almacena en la base de datos foest	
Modulo: Registro interfaceBD	
Estereotipo: interface	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	

Solicita ValidarRegistroUsuarioFuncionario	BasedeDatosfoest
Solicita CrearRegistroUsuarioFuncionario	BasedeDatosfoest
Solicita Funcionarioregistrado	BasedeDatosfoest
Solicita ActualizarRegistroUsuarioFuncionario	BasedeDatosfoest
Solicita EliminarRegistroUsuarioFuncionario	BasedeDatosfoest
Solicita CrearRegistroUsuarioEstudiante	BasedeDatosfoest
Solicita EstudianteRegistrado	BasedeDatosfoest
Solicita ActualizarRegistroUsuarioEstudiante	BasedeDatosfoest
Solicita EliminarRegistroUsuarioEstudiante	BasedeDatosfoest
Solicita CrearRegistroBeneficio1	BasedeDatosfoest
Solicita beneficio1	BasedeDatosfoest
Solicita ActualizarRegistroBeneficio1	BasedeDatosfoest
Solicita EliminarRegistrobeneficio1	BasedeDatosfoest
Solicita CrearRegistroBeneficio2	BasedeDatosfoest
Solicita beneficio2	BasedeDatosfoest
Solicita ActualizarRegistroBeneficio2	BasedeDatosfoest
Solicita EliminarRegistrobeneficio2	BasedeDatosfoest
Solicita CrearRegistroAlgunOtroBeneficio	BasedeDatosfoest
Solicita Algunotrobeneficio	BasedeDatosfoest
Solicita ActualizarRegistroAlgunOtroBeneficio	BasedeDatosfoest
Solicita EliminarRegistroAlgunOtroBeneficio	BasedeDatosfoest
Solicita CrearRegistroAntiguedadsisben	BasedeDatosfoest
Solicita AntigüedadSisbén	BasedeDatosfoest
Solicita ActualizarRegistroantigüedadSisbén	BasedeDatosfoest
Solicita EliminarRegistroantigüedadSisbén	BasedeDatosfoest
Solicita CrearRegistroTipoPoblacion	BasedeDatosfoest
Solicita TipoPoblacion	BasedeDatosfoest
Solicita ActualizarRegistroTipoPoblacion	BasedeDatosfoest
Solicita EliminarRegistroTipoPoblacion	BasedeDatosfoest
Solicita CrearRegistroUniversidad	BasedeDatosfoest
Solicita universidades	BasedeDatosfoest
Solicita ActualizarRegistroUniversidad	BasedeDatosfoest
Solicita EliminarRegistroUniversidad	BasedeDatosfoest
Solicita CrearRegistroCarreras	BasedeDatosfoest
Solicita Carreras	BasedeDatosfoest
Solicita ActualizarRegistroCarreras	BasedeDatosfoest
Solicita EliminarRegistroCarreras	BasedeDatosfoest
Solicita CrearRegistroCiudad	BasedeDatosfoest
Solicita Ciudad	BasedeDatosfoest
Solicita ActualizarRegistroCiudad	BasedeDatosfoest
Solicita EliminarRegistroCiudad	BasedeDatosfoest
Solicita PedirCertificado	BasedeDatosfoest

Solicita estadísticas	BasedeDatosfoest
-----------------------	------------------

Estudiantes foest:

Tarjeta para la clase ManejadorFoest con responsabilidades y colaboraciones.

Clase: ManejadorFoest	
Descripción: se encarga de enviar las peticiones particulares de opciones a los manejadores especializados para registro de las diferentes opciones.	
Modulo: estudiantes	
Estereotipo: control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Solicita DesplegarPantallaPrincipal	InterfaceUsuario
Maneja el evento "Administrar registro funcionario"	
Solicita registrar	ManejadorRegistroFuncionario
Maneja el evento "Administrar registro estudiante"	
Solicita registrar	ManejadorRegistrarEstudiantes
Maneja el evento "Administrar registro Beneficio1"	
Solicita registrar	ManejadorRegistroBeneficio1
Maneja el evento "Administrar registro Beneficio2"	
Solicita registrar	ManejadorRegistroBeneficio2
Maneja el evento "Administrar registro Algún otro Beneficio"	
Solicita registrar	ManejadorAlgunOtroBeneficio
Maneja el evento "Administrar registro Antigüedad Sisbén"	
Solicita registrar	ManejadorAntigüedadSisben
Maneja el evento "Administrar registro Tipos de poblacion"	
Solicita registrar	ManejadorTipoPoblacion
Maneja el evento "Administrar registro universidades"	
Solicita registrar	ManejadorUniversidades
Maneja el evento "Administrar registro carrera"	
Solicita registrar	ManejadorCarreras
Maneja el evento "Administrar registro ciudad"	
Solicita registrar	ManejadorCiudad

Maneja el evento "Administrar documentos"	
Solicita registrar	ManejadorDocumentos
Maneja el evento "salir"	
Sale del sistema	

Pantalla principal:

Tarjeta para la clase pantalla principal con responsabilidades y colaboraciones.

Clase: PantallaPrincipal	
Descripción: pantalla principal (P2)	
Modulo: Principal	
Estereotipo: Borde	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Envía el evento "Administrar registro funcionario"	InterfaceUsuario
Envía el evento "Administrar registro estudiante"	InterfaceUsuario
Envía el evento "Administrar registro Beneficio1"	InterfaceUsuario
Envía el evento "Administrar registro Beneficio2"	InterfaceUsuario
Envía el evento "Administrar registro Algún otro Beneficio"	InterfaceUsuario
Envía el evento "Administrar registro Antigüedad Sisbén"	InterfaceUsuario
Envía el evento "Administrar registro Tipos de poblacion"	InterfaceUsuario
Envía el evento "Administrar registro universidades"	InterfaceUsuario
Envía el evento "Administrar registro carrera"	InterfaceUsuario
Envía el evento "Administrar registro ciudad"	InterfaceUsuario
Envía el evento "Administrar documentos"	InterfaceUsuario
Envía el evento "salir"	InterfaceUsuario

8.1.4 Jerarquías

Interface usuario:

Tarjeta para la clase InterfaceUsuario con responsabilidades, colaboraciones y jerarquía.

Clase: InterfaceUsuario	
Descripción: toda la interacción con el usuario se realiza por medio de la interface de usuario	
Modulo: InterfaceUsuario	
Estereotipo: Borde	
Propiedades: Concreta	
Superclases:	
Subclases:	
Atributos:	
DesplegarPantalla	Pantalla; PantallaIngreso, PantallaPrincipal, PantallaFuncionarioRegistrado, PantallaEstudianteRegistrado, PantallaBeneficio1, PantallaBeneficio2, PantallaAlgunOtroBeneficio, PantallaAntiguedadSisben, PantallaTiposPoblacion, PantallaUniversidades, PantallaCarreras, PantallaCiudad, PantallaDocumentos, PantallaCertificado, PantallaEstadísticas
EnviarEvento	Manejador; ManejadorPrincipal, ManejadorFoest, ManejadorFuncionario, ManejadorEstudiantes, ManejadorBeneficio1, ManejadorBeneficio2, ManejadorAlgunOtroBeneficio, ManejadorAntiguedadSisben, ManejadorTipoPoblacion, ManejadorUniversidades, ManejadorCarreras, ManejadorCiudad, ManejadorDocumentos, ManejadorCertificado, ManejadorEstadísticas

Pantalla

Tarjeta para la clase pantalla con responsabilidades, colaboraciones y jerarquía

Clase: pantalla
Descripción: pantalla heredada por las demás clases del tipo de pantalla
Modulo: InterfaceUsuario
Estereotipo: Borde
Propiedades: Abstracta
Superclases:
Subclases: PantallaIngreso, PantallaPrincipal, PantallaRegistrarFuncionario, PantallaRegistrarEstudiante, PantallaRegistrarBeneficio1, PantallaRegistrarBeneficio2, PantallaRegistrarAlgunOtroBeneficio, PantallaRegistroAntiguedadSisben,

PantallaRegistroTiposPoblacion, PantallaRegistroUniversidad, PantallaRegistroCarreras, PantallaRegistroCiudad, PantallaDocumentos	
Atributos:	
DesplegarPantalla	
EnviarEvento	InterfaceUsuario

Manejador:

Tarjeta para la clase Manejador con responsabilidades, colaboraciones y jerarquía.

Clase: Manejador	
Descripción: superclase heredada por los manejadores del sistema	
Modulo: Principal	
Estereotipo: Control	
Propiedades: Abstracta	
Superclases:	
Subclases: ManejadorPrincipal, ManejadorFoest, ManejadorRegistroFuncionario, ManejadorRegistrarEstudiantes, ManejadorRegistroBeneficio1, ManejadorRegistroBeneficio2, ManejadorAlgunOtroBeneficio, ManejadorTipoPoblacion, ManejadorUniversidades, ManejadorCarreras, ManejadorCiudad, ManejadorDocumentos, ManejadorCertificado, ManejadorEstadísticas.	
Atributos:	
Maneja evento	
DesplegarPantalla	InterfaceUsuario
Manejar evento Estudiantesfoest	
Servicios	ManejadorFoest
Manejar evento salir	
Salir	

ManejadorPrincipal:

Tarjeta para la clase ManejadorPrincipal con responsabilidades, colaboraciones y jerarquía.

Clase: ManejadorPrincipal	
Descripción: es el encargado de desplegar la pantalla principal de interacción con el usuario y después desplegar las diferentes funciones a los manejadores apropiados	
Modulo: Principal	
Estereotipo: Control	
Propiedades: Concreta	
Superclases: manejador	
Subclases:	
Atributos:	

Manejar evento	
Maneja EventoValidar	
ValidarRegistroUsuarioFuncionario	ManejadorRegistroFuncionario

Pantalla ingreso:

Tarjeta para la clase pantalla ingreso con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaIngreso	
Descripción: pantalla ingreso (P1)	
Modulo: Principal	
Estereotipo: Borde	
Propiedades: concreta	
Superclases: pantalla	
Subclases:	
Atributos:	

Tarjeta para la clase datos con responsabilidades, colaboraciones y jerarquía

Clase: datos	
Descripción: superclase para todas las clases entidad	
Modulo: Dominio	
Estereotipo: Entidad	
Propiedades: Abstracta	
Superclases:	
Subclases: RegistrarUsuarioFuncionario, RegistrarUsuarioEstudiantes, RegistrarBeneficio1, RegistroBeneficio2, RegistrarAlgunOtroBeneficio, RegistrarAntigüedadSisben, RegistrarTipoPoblacion, RegistrarUniversidades, RegistrarCarreras, RegistrarCiudad, CertificadoEstudiante, Estadísticas	
Atributos:	

Registro:

Este módulo se compone del módulo usuario funcionario, usuario estudiante, beneficio1, beneficio2, algún otro beneficio, antigüedad Sisbén, tipos de poblacion, universidades, carreras, ciudad, certificados estudiantes e InterfaceBasedeDatosRegistro.

Usuario funcionario:

Este módulo se compone de las clases ManejadorFuncionario, PantallaFuncionario, PantallaRegistrarFuncionario, PantallaFuncionarioRegistrado y RegistrarUsuarioFuncionario.

Tarjeta para la clase ManejadorFuncionario con responsabilidades, colaboraciones y jerarquía.

Clase: ManejadorFuncionario	
Descripción: se encarga de todo lo relacionado con el registro del usuario funcionario	
Modulo: RegistrarUsuarioFuncionario	
Estereotipo: Control	
Propiedades: concreta	
Superclases: Manejador	
Subclases:	
Atributos:	
Manejar evento	
CrearRegistroUsuarioFuncionario	
Solicita ValidarRegistroUsuarioFuncionario	InterfaceBasedeDatosRegistro
Maneja el evento "registrar"	
Solicita CrearRegistroUsuarioFuncionario	InterfaceBasedeDatosRegistro
Solicita FuncionariosRegistrados	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	
Solicita ActualizarRegistroUsuarioFuncionario	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroUsuarioFuncionario	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaRegistrarFuncionario con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaFuncionario	
Descripción: superclase con diseño común para las pantallas de registro de usuario funcionario.	
Modulo: registro, usuario funcionario	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	
Subclases: pantallaregistrarfuncionario, PantallaFuncionarioRegistrado	
Atributos:	
despliega	

Tarjeta para la clase PantallaRegistrarFuncionario con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaRegistrarFuncionario	
Descripción: pantalla crear funcionarios (p32)	
Modulo: registro, usuario funcionario	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: PantallaFuncionario	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaFuncionarioRegistrado con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaFuncionarioRegistrado	
Descripción: pantalla funcionarios registrados (p33)	
Modulo: registro, usuario funcionario	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaFuncionario	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarUsuarioFuncionario con responsabilidades, colaboraciones y jerarquía.

Clase: RegistrarUsuarioFuncionario	
Descripción: el usuario para que pueda utilizar el sistema, debe estar registrado, en el registro se debe ingresar la información del usuario: nombre, apellido, teléfono, email, login y password.	
Modulo: registro, usuario funcionario	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos:	

Usuarios estudiantes:

Este módulo se compone de las clases ManejadorEstudiantes, PantallaEstudiantes, PantallaRegistrarEstudiante, PantallaEstudianteRegistrado y RegistrarUsuarioEstudiantes.

Tarjeta para la clase ManejadorEstudiantes con responsabilidades, colaboraciones y jerarquía.

Clase: ManejadorEstudiantes	
Descripción: se encarga de todo lo relacionado con el registro del usuario estudiante	
Modulo: RegistrarUsuarioEstudiantes	
Estereotipo: Control	
Propiedades: Concreta	
Superclases: Manejador	
Subclases:	
Atributos:	
Manejar evento	
Maneja el evento "registrar"	
Solicita CrearRegistroUsuarioEstudiante	InterfaceBasedeDatosRegistro
Solicita EstudianteRegistrado	InterfaceBasedeDatosRegistro
Solicita EstudiantesRegistrados	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	
Solicita ActualizarRegistroUsuarioEstudiante	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroUsuarioEstudiante	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaEstudiantes con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaEstudiantes	
Descripción: superclase con diseño común para las pantallas de registro de usuario estudiante	
Modulo: registro, usuario estudiante	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	
Subclases: PantallaRegistrarEstudiante, PantallaEstudianteRegistrado	
Atributos:	
despliega	

Tarjeta para la clase PantallaRegistrarEstudiante con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaRegistrarEstudiante	
Descripción: pantalla registro estudiantes (p4)	
Modulo: registro, usuario estudiante	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaEstudiantes	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaEstudianteRegistrado con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaEstudianteRegistrado	
Descripción: pantalla estudiantes registrados (p3)	
Modulo: registro, usuarios estudiantes	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaEstudiantes	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarUsuarioEstudiantes con responsabilidades, colaboraciones y jerarquía.

Clase: RegistrarUsuarioEstudiantes	
Descripción: en el registro se debe ingresar la información del usuario: N° carpeta, fecha de radicación, beneficio 1, beneficio 2, 1° vez, renovación, reintegro, primer apellido, segundo apellido, primer nombre, segundo nombre, tipo de documento, numero de documento Expedido en, edad, población especial, genero, teléfono 1; teléfono 2, dirección, correo electrónico, antigüedad Sisbén en años, puntaje del Sisbén, recibe algún otro beneficio, universidad sede nivel carrera, modalidad, semestre, valor del semestre, porcentaje a aplicar monto aprobado, otro beneficio, materias perdidas promedio 2018_1 y 2017, horas sociales cumplidas, dependencia donde realizo horas sociales, Horario, ciudad a la que viaja, días que viaja, valor por 2 recorridos valor semanal, valor mensual bruto , valor a pagar por mes, meses a girar y total semestre; también los datos del representante legal si es un menor de edad.	
Modulo: registro, usuarios estudiantes	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	

Subclases:	
Atributos:	

Beneficio1:

Este módulo se compone de las clases ManejadorBeneficio1, PantallaDiseñoBeneficio1, PantallaRegistrarBeneficio1, PantallaBeneficio1 y RegistrarBeneficio1.

Tarjeta para la clase ManejadorBeneficio1 con responsabilidades, colaboraciones y jerarquía.

Clase: ManejadorBeneficio1	
Descripción: se encarga de todo lo relacionado con el registro del beneficio1	
Modulo: RegistrarBeneficio1	
Estereotipo: Control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos:	
manejarevento	
Maneja el evento "registrar"	
Solicita CrearRegistroBeneficio1	InterfaceBasedeDatosRegistro
Solicita Beneficio1	
Solicita Beneficio1	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	
Solicita ActualizarRegistroBeneficio1	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroBeneficio1	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoBeneficio1 con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaDiseñoBeneficio1	
Descripción: Superclase con diseño común para las pantallas de registro de beneficio1	
Modulo: registro, beneficio1	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: Pantalla	

Subclases: PantallaRegistrarBeneficio1, PantallaBeneficio1	
Atributos:	
Despliega	

Tarjeta para la clase PantallaRegistrarBeneficio1 con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaRegistrarBeneficio1	
Descripción: pantalla registro beneficio 1 (p7)	
Modulo: registro, beneficio1	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoBeneficio1	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaBeneficio1 con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaBeneficio1	
Descripción: pantalla beneficio1 (p6)	
Modulo: registro, beneficio1	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoBeneficio1	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarBeneficio1 con responsabilidades y colaboraciones.

Clase: RegistrarBeneficio1	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, beneficio1	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	

Subclases:	
Atributos:	

Beneficio2:

Este módulo se compone de las clases ManejadorBeneficio2, PantallaDiseñoBeneficio2, PantallaRegistrarBeneficio2, PantallaBeneficio2 y RegistrarBeneficio2.

Tarjeta para la clase ManejadorBeneficio2 con responsabilidades, colaboraciones y jerarquía.

Clase: ManejadorBeneficio2	
Descripción: se encarga de todo lo relacionado con el registro del beneficio2	
Modulo: RegistrarBeneficio2	
Estereotipo: Control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos:	
manejarevento	
Maneja el evento "registrar"	
Solicita CrearRegistroBeneficio2	InterfaceBasedeDatosRegistro
Solicita Beneficio2	
Solicita Beneficio2	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	
Solicita ActualizarRegistroBeneficio2	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroBeneficio2	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoBeneficio2 con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaDiseñoBeneficio2	
Descripción: Superclase con diseño común para las pantallas de registro de beneficio2	
Modulo: registro, beneficio2	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: Pantalla	

Subclases: PantallaRegistrarBeneficio2, PantallaBeneficio2	
Atributos:	
Despliega	

Tarjeta para la clase PantallaRegistrarBeneficio2 con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaRegistrarBeneficio2	
Descripción: pantalla registro beneficio 2 (p7)	
Modulo: registro, beneficio2	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoBeneficio2	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaBeneficio2 con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaBeneficio2	
Descripción: pantalla beneficio2 (p6)	
Modulo: registro, beneficio2	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoBeneficio2	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarBeneficio2 con responsabilidades, colaboraciones y jerarquía.

Clase: RegistrarBeneficio2	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, beneficio2	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	

Subclases:	
Atributos:	

Algún Otro Beneficio:

Este módulo se compone de las clases `ManejadorAlgúnOtroBeneficio`, `PantallaDiseñoAlgúnOtroBeneficio`, `PantallaRegistrarAlgúnOtroBeneficio`, `PantallaAlgúnOtroBeneficio`, y `RegistrarAlgúnOtroBeneficio`.

Tarjeta para la clase `ManejadorAlgúnOtroBeneficio` con responsabilidades, colaboraciones y jerarquía.

Clase: <code>ManejadorAlgúnOtroBeneficio</code>	
Descripción: se encarga de todo lo relacionado con el registro del beneficio2	
Modulo: <code>RegistrarAlgúnOtroBeneficio</code>	
Estereotipo: Control	
Propiedades: Concreta	
Superclases: <code>Manejador</code>	
Subclases:	
Atributos:	
<code>manejarevento</code>	
Maneja el evento "registrar"	
Solicita <code>CrearRegistroAlgúnOtroBeneficio</code>	<code>InterfaceBasedeDatosRegistro</code>
Solicita <code>Algúnotrobeficio</code>	
Solicita <code>Algúnotrobeficio</code>	<code>InterfaceBasedeDatosRegistro</code>
Maneja el evento "actualizar"	
Solicita <code>ActualizarRegistroAlgúnOtroBeneficio</code>	<code>InterfaceBasedeDatosRegistro</code>
Maneja el evento "eliminar"	
Solicita <code>EliminarRegistroAlgúnOtroBeneficio</code>	<code>InterfaceBasedeDatosRegistro</code>

Tarjeta para la clase `PantallaDiseñoAlgúnOtroBeneficio` con responsabilidades, colaboraciones y jerarquía.

Clase: <code>PantallaDiseñoAlgúnOtroBeneficio</code>	
Descripción: pantalla registro algún otro beneficio (p13)	
Modulo: registro, <code>Algúnotrobeficio</code>	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	

Subclases: PantallaRegistrarAlgunOtroBeneficio, PantallaAlgunOtroBeneficio	
Atributos:	
despliega	

Tarjeta para la clase PantallaRegistrarAlgunOtroBeneficio con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaRegistrarAlgunOtroBeneficio	
Descripción: pantalla registro algún otro beneficio (p13)	
Modulo: registro, Algunotrobeficio	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoAlgunOtroBeneficio	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaAlgunOtroBeneficio con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaAlgunOtroBeneficio	
Descripción: pantalla algún otro beneficio (p12)	
Modulo: registro, Algunotrobeficio	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoAlgunOtroBeneficio	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarAlgunOtroBeneficio con responsabilidades, colaboraciones y jerarquía.

Clase: RegistrarAlgunOtroBeneficio	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, Algunotrobeficio	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	

Subclases:	
Atributos:	

Antigüedad Sisbén:

Este módulo se compone de las clases ManejadorAntigüedadSisben, PantallaDiseñoAntigüedadSisben, PantallaRegistroAntigüedadSisben, PantallaAntigüedadSisben, y RegistrarAntigüedadSisben.

Tarjeta para la clase ManejadorAntigüedadSisben con responsabilidades, colaboraciones y jerarquía.

Clase: ManejadorAntigüedadSisben	
Descripción: se encarga de todo lo relacionado con el registro de la antigüedad del Sisbén	
Modulo: RegistrarAntigüedadSisben	
Estereotipo: Control	
Propiedades: Concreta	
Superclases: Manejador	
Subclases:	
Atributos:	
manejarevento	
Maneja el evento "registrar"	
Solicita CrearRegistroAntigüedadSisben	InterfaceBasedeDatosRegistro
Solicita antigüedadSisbén	
Solicita antigüedadSisbén	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	
Solicita ActualizarRegistroantigüedadSisbén	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroantigüedadSisbén	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoAntigüedadSisben con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaDiseñoAntigüedadSisben	
Descripción: Superclase con diseño común para las pantallas de registro de antigüedad Sisbén	
Modulo: registro, AntigüedadSisbén	
Estereotipo: borde	
Propiedades: Abstracta	
Superclases: Pantalla	
Subclases: PantallaRegistroAntigüedadSisben, PantallaAntigüedadSisben	

Atributos:	
despliega	

Tarjeta para la clase PantallaRegistroAntigüedadSisben con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaRegistroAntigüedadSisben	
Descripción: pantalla registro antigüedad Sisbén (p16)	
Modulo: registro, AntigüedadSisbén	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoAntigüedadSisben	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaAntigüedadSisben con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaAntigüedadSisben	
Descripción: pantalla antigüedad Sisbén (p15)	
Modulo: registro, AntigüedadSisbén	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoAntigüedadSisben	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarAntigüedadSisben con responsabilidades, colaboraciones y jerarquía.

Clase: RegistrarAntigüedadSisben	
Descripción: en el registro se debe ingresar la información de la antigüedad Sisbén: cantidad de años.	
Modulo: registro, AntigüedadSisbén	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	

Atributos:	

Tipos de poblacion:

Este módulo se compone de las clases ManejadorTipoPoblacion, PantallaDiseñoTiposPoblacion, PantallaRegistroTiposPoblacion, PantallaTiposPoblacion, y RegistrarTipoPoblacion.

Tarjeta para la clase ManejadorTipoPoblacion con responsabilidades, colaboraciones y jerarquía.

Clase: ManejadorTipoPoblacion	
Descripción: se encarga de todo lo relacionado con el registro del tipo de poblacion.	
Modulo: RegistrarTipoPoblacion	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
manejarevento	
Maneja el evento "registrar"	
Solicita CrearRegistroTipoPoblacion	InterfaceBasedeDatosRegistro
Solicita TipoPoblacion	
Solicita TipoPoblacion	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	
Solicita ActualizarRegistroTipoPoblacion	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroTipoPoblacion	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoTiposPoblacion con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaDiseñoTiposPoblacion	
Descripción: Superclase con diseño común para las pantallas de registro de tipos de poblacion	
Modulo: registro, TipoPoblacion	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	
Subclases: PantallaRegistroTiposPoblacion, PantallaTiposPoblacion	
Atributos:	

despliega	

Tarjeta para la clase PantallaRegistroTiposPoblacion con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaRegistroTiposPoblacion	
Descripción: pantalla registro tipos de poblacion (p19)	
Modulo: registro, TipoPoblacion	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoTiposPoblacion	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaTiposPoblacion con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaTiposPoblacion	
Descripción: pantalla tipos de poblacion (p18)	
Modulo: registro, TipoPoblacion	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoTiposPoblacion	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarTipoPoblacion con responsabilidades, colaboraciones y jerarquía.

Clase: RegistrarTipoPoblacion	
Descripción: en el registro se debe ingresar la información del tipo de poblacion: nombre.	
Modulo: registro, TipoPoblacion	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos:	

Universidades:

Este módulo se compone de las clases ManejadorUniversidades, PantallaDiseñoUniversidad, PantallaRegistroUniversidad, PantallaUniversidades, y RegistrarUniversidades.

Tarjeta para la clase ManejadorUniversidades con responsabilidades, colaboraciones y jerarquía.

Clase: ManejadorUniversidades	
Descripción: se encarga de todo lo relacionado con el registro de las universidades.	
Modulo: RegistrarUniversidades	
Estereotipo: Control	
Propiedades: Concreta	
Superclases:	
Subclases:	
Atributos:	
manejarevento	
Maneja el evento "registrar"	
Solicita CrearRegistroUniversidad	InterfaceBasedeDatosRegistro
Solicita Universidades	
Solicita Universidades	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	
Solicita ActualizarRegistroUniversidad	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroUniversidad	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoUniversidad con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaDiseñoUniversidad	
Descripción: Superclase con diseño común para las pantallas de registro de universidades	
Modulo: registro, Universidad	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	
Subclases: PantallaRegistroUniversidad, PantallaUniversidades	
Atributos:	
despliega	

Tarjeta para la clase PantallaRegistroUniversidad con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaRegistroUniversidad	
Descripción: pantalla registro universidades (p22)	
Modulo: registro, Universidad	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoUniversidad	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaUniversidades con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaUniversidades	
Descripción: pantalla universidades (p21)	
Modulo: registro, Universidad	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoUniversidad	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarUniversidades con responsabilidades, colaboraciones y jerarquía.

Clase: RegistrarUniversidades	
Descripción: en el registro se debe ingresar la información de la universidad: nombre y porcentaje a aplicar.	
Modulo: registro, Universidad	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos:	

Carreras:

Este módulo se compone de las clases ManejadorCarreras, PantallaDiseñoCarreras, PantallaRegistroCarreras, PantallaCarreras, y RegistrarCarreras.

Tarjeta para la clase ManejadorCarreras con responsabilidades, colaboraciones y jerarquía.

Clase: ManejadorCarreras	
Descripción: se encarga de todo lo relacionado con el registro de las carreras.	
Modulo: RegistrarCarreras	
Estereotipo: Control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos:	
Maneja el evento "registrar"	
Solicita CrearRegistroCarreras	InterfaceBasedeDatosRegistro
Solicita Carreras	
Solicita Carreras	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	
Solicita ActualizarRegistroCarreras	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroCarreras	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoCarreras con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaDiseñoCarreras	
Descripción: superclase con diseño común para las pantallas de registro de carrera	
Modulo: registro, Carreras	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	
Subclases: PantallaRegistroCarreras, PantallaCarreras	
Atributos:	
despliega	

Tarjeta para la clase PantallaRegistroCarreras con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaRegistroCarreras	
Descripción: pantalla registro carreras (p25)	
Modulo: registro, Carreras	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoCarreras	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaCarreras con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaCarreras	
Descripción: pantalla carreras (p24)	
Modulo: registro, Carreras	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoCarreras	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarCarreras con responsabilidades, colaboraciones y jerarquía.

Clase: RegistrarCarreras	
Descripción: en el registro se debe ingresar la información de la carrera: nombre.	
Modulo: registro, Carreras	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos:	

Ciudad:

Este módulo se compone de las clases ManejadorCiudad, PantallaDiseñoCiudad, PantallaRegistroCiudad, PantallaCiudad, y RegistrarCiudad.

Tarjeta para la clase ManejadorCiudad con responsabilidades, colaboraciones y jerarquía.

Clase: ManejadorCiudad	
Descripción: se encarga de todo lo relacionado con el registro de las ciudades.	
Modulo: RegistrarCiudad	
Estereotipo: Control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos:	
Maneja el evento "registrar"	
Solicita CrearRegistroCiudad	InterfaceBasedeDatosRegistro
Solicita Ciudad	
Solicita Ciudad	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	
Solicita ActualizarRegistroCiudad	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroCiudad	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoCiudad con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaDiseñoCiudad	
Descripción: Superclase con diseño común para las pantallas de registro de ciudades	
Modulo: registro, Ciudad	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	
Subclases: PantallaRegistroCiudad, PantallaCiudad	
Atributos:	
despliega	

Tarjeta para la clase PantallaRegistroCiudad con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaRegistroCiudad	
Descripción: pantalla registro ciudad (p28)	
Modulo: registro, Ciudad	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoCiudad	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaCiudad con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaCiudad	
Descripción: pantalla ciudad (p27)	
Modulo: registro, Ciudad	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoCiudad	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarCiudad con responsabilidades, colaboraciones y jerarquía.

Clase: RegistrarCiudad	
Descripción: en el registro se debe ingresar la información de la ciudad: nombre.	
Modulo: registro, Ciudad	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos:	

Certificación estudiantes:

Este módulo se compone de las clases ManejadorDocumentos, PantallaDiseñoDocumentos, PantallaDocumentos, PantallaCertificado, PantallaEstadísticas, CertificadoEstudiante y Estadísticas.

Tarjeta para la clase ManejadorDocumentos con responsabilidades, colaboraciones y jerarquía.

Clase: ManejadorDocumentos	
Descripción: se encarga de todo lo relacionado con informes como estadísticas y certificados.	
Modulo: CertificadoEstudiante y Estadísticas.	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Maneja el evento "certificado"	
Solicita PedirCertificado	InterfaceBasedeDatosRegistro
Maneja el evento "estadísticas"	
Solicita Estadísticas	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoDocumentos con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaDiseñoDocumentos	
Descripción: superclase con diseño común para las pantallas de solicitud de documentos	
Modulo: Certificados y Estadísticas.	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	
Subclases: PantallaDocumentos, PantallaCertificado, PantallaEstadísticas	
Atributos:	
despliega	

Tarjeta para la clase PantallaDocumentos con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaDocumentos	
Descripción: pantalla registro ciudad (p34)	
Modulo: Certificados y Estadísticas.	
Estereotipo: borde	

Propiedades: concreta
Superclases: PantallaDiseñoDocumentos
Subclases:
Atributos:

Tarjeta para la clase PantallaCertificado con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaCertificado
Descripción: pantalla certificados (p31)
Modulo: Certificados.
Estereotipo: borde
Propiedades: concreta
Superclases: PantallaDiseñoDocumentos
Subclases:
Atributos:

Tarjeta para la clase PantallaEstadisticas con responsabilidades, colaboraciones y jerarquía.

Clase: PantallaEstadisticas
Descripción: pantalla estadisticas (p30)
Modulo: Estadisticas.
Estereotipo: borde
Propiedades: concreta
Superclases: PantallaDiseñoDocumentos
Subclases:
Atributos:

Tarjeta para la clase CertificadoEstudiante con responsabilidades, colaboraciones y jerarquía.

Clase: CertificadoEstudiante
Descripción: en el certificado se debe ingresar la información del estudiante: documento de identidad.
Modulo: registro, Certificados
Estereotipo: entidad
Propiedades: concreta
Superclases: datos

Subclases:	
Atributos:	

Tarjeta para la clase Estadísticas con responsabilidades, colaboraciones y jerarquía.

Clase: Estadísticas	
Descripción: en estadísticas se toman los datos de los estudiantes registrado en la base de datos.	
Modulo: registro, Estadísticas	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos:	

Interface base de datos:

Tarjeta para la clase InterfaceBasedeDatosRegistro con responsabilidades y colaboraciones.

Clase: InterfaceBasedeDatosRegistro	
Descripción: la información de los usuarios, beneficios y otros se almacena en la base de datos foest	
Modulo: Registro interfaceBD	
Estereotipo: interface	
Propiedades: concreta	
Superclases:	
Subclases:	
Atributos:	
Solicita ValidarRegistroUsuarioFuncionario	BasedeDatosfoest
Solicita CrearRegistroUsuarioFuncionario	BasedeDatosfoest
Solicita Funcionarioregistrado	BasedeDatosfoest
Solicita ActualizarRegistroUsuarioFuncionario	BasedeDatosfoest
Solicita EliminarRegistroUsuarioFuncionario	BasedeDatosfoest
Solicita CrearRegistroUsuarioEstudiante	BasedeDatosfoest
Solicita EstudianteRegistrado	BasedeDatosfoest
Solicita ActualizarRegistroUsuarioEstudiante	BasedeDatosfoest
Solicita EliminarRegistroUsuarioEstudiante	BasedeDatosfoest
Solicita CrearRegistroBeneficio1	BasedeDatosfoest
Solicita beneficio1	BasedeDatosfoest
Solicita ActualizarRegistroBeneficio1	BasedeDatosfoest

Solicita EliminarRegistrobeneficio1	BasedeDatosfoest
Solicita CrearRegistroBeneficio2	BasedeDatosfoest
Solicita beneficio2	BasedeDatosfoest
Solicita ActualizarRegistroBeneficio2	BasedeDatosfoest
Solicita EliminarRegistrobeneficio2	BasedeDatosfoest
Solicita CrearRegistroAlgunOtroBeneficio	BasedeDatosfoest
Solicita Algunotrobeneficio	BasedeDatosfoest
Solicita ActualizarRegistroAlgunOtroBeneficio	BasedeDatosfoest
Solicita EliminarRegistroAlgunOtroBeneficio	BasedeDatosfoest
Solicita CrearRegistroAntigüedadsisben	BasedeDatosfoest
Solicita AntigüedadSisbén	BasedeDatosfoest
Solicita ActualizarRegistroantigüedadSisbén	BasedeDatosfoest
Solicita EliminarRegistroantigüedadSisbén	BasedeDatosfoest
Solicita CrearRegistroTipoPoblacion	BasedeDatosfoest
Solicita TipoPoblacion	BasedeDatosfoest
Solicita ActualizarRegistroTipoPoblacion	BasedeDatosfoest
Solicita EliminarRegistroTipoPoblacion	BasedeDatosfoest
Solicita CrearRegistroUniversidad	BasedeDatosfoest
Solicita universidades	BasedeDatosfoest
Solicita ActualizarRegistroUniversidad	BasedeDatosfoest
Solicita EliminarRegistroUniversidad	BasedeDatosfoest
Solicita CrearRegistroCarreras	BasedeDatosfoest
Solicita Carreras	BasedeDatosfoest
Solicita ActualizarRegistroCarreras	BasedeDatosfoest
Solicita EliminarRegistroCarreras	BasedeDatosfoest
Solicita CrearRegistroCiudad	BasedeDatosfoest
Solicita Ciudad	BasedeDatosfoest
Solicita ActualizarRegistroCiudad	BasedeDatosfoest
Solicita EliminarRegistroCiudad	BasedeDatosfoest
Solicita PedirCertificado	BasedeDatosfoest
Solicita estadísticas	BasedeDatosfoest

Estudiantes foest:

Tarjeta para la clase ManejadorFoest con responsabilidades y colaboraciones.

Clase: ManejadorFoest	
Descripción: se encarga de enviar las peticiones particulares de opciones a los manejadores especializados para registro de las diferentes opciones.	
Modulo: estudiantes	
Estereotipo: control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos:	
Solicita DesplegarPantallaPrincipal	InterfaceUsuario
Maneja el evento "Administrar registro funcionario"	
Solicita registrar	ManejadorRegistroFuncionario
Maneja el evento "Administrar registro estudiante"	
Solicita registrar	ManejadorRegistrarEstudiantes
Maneja el evento "Administrar registro Beneficio1"	
Solicita registrar	ManejadorRegistroBeneficio1
Maneja el evento "Administrar registro Beneficio2"	
Solicita registrar	ManejadorRegistroBeneficio2
Maneja el evento "Administrar registro Algún otro Beneficio"	
Solicita registrar	ManejadorAlgúnOtroBeneficio
Maneja el evento "Administrar registro Antigüedad Sisbén"	
Solicita registrar	ManejadorAntigüedadSisben
Maneja el evento "Administrar registro Tipos de poblacion"	
Solicita registrar	ManejadorTipoPoblacion
Maneja el evento "Administrar registro universidades"	
Solicita registrar	ManejadorUniversidades
Maneja el evento "Administrar registro carrera"	
Solicita registrar	ManejadorCarreras
Maneja el evento "Administrar registro ciudad"	
Solicita registrar	ManejadorCiudad
Maneja el evento "Administrar documentos"	
Solicita registrar	ManejadorDocumentos
Maneja el evento "salir"	
Sale del sistema	

Pantalla principal:

Tarjeta para la clase PantallaPrincipal con responsabilidades y colaboraciones.

Clase: PantallaPrincipal	
Descripción: pantalla principal (P2)	
Modulo: Principal	
Estereotipo: Borde	
Propiedades: concreta	
Superclases: pantalla	
Subclases:	
Atributos:	

8.1.5 Contratos

Interface usuario:

Tarjeta para la clase InterfaceUsuario con responsabilidades, colaboraciones, jerarquía y contratos

Clase: InterfaceUsuario	
Descripción: toda la interacción con el usuario se realiza por medio de la interface de usuario	
Modulo: InterfaceUsuario	
Estereotipo: Borde	
Propiedades: Concreta	
Superclases:	
Subclases:	
Atributos:	
Contratos	
1. Desplegar pantalla	
DesplegarPantalla	Pantalla; PantallaIngreso, PantallaPrincipal, PantallaFuncionarioRegistrado, PantallaEstudianteRegistrado, PantallaBeneficio1, PantallaBeneficio2, PantallaAlgunOtroBeneficio, PantallaAntiguedadSisben, PantallaTiposPoblacion,

	PantallaUniversidades, PantallaCarreras, PantallaCiudad, PantallaDocumentos, PantallaCertificado, PantallaEstadisticas
2. Enviar evento	
EnviarEvento	Manejador; ManejadorPrincipal, ManejadorFoest, ManejadorFuncionario, ManejadorEstudiantes, ManejadorBeneficio1, ManejadorBeneficio2, ManejadorAlgunOtroBeneficio, ManejadorAntiguedadSisben, ManejadorTipoPoblacion, ManejadorUniversidades, ManejadorCarreras, ManejadorCiudad, ManejadorDocumentos, ManejadorCertificado, ManejadorEstadisticas

Pantalla

Tarjeta para la clase pantalla con responsabilidades, colaboraciones, jerarquía y contratos

Clase: pantalla	
Descripción: pantalla heredada por las demás clases del tipo de pantalla	
Modulo: InterfaceUsuario	
Estereotipo: Borde	
Propiedades: Abstracta	
Superclases:	
Subclases: PantallaIngreso, PantallaPrincipal, PantallaRegistrarFuncionario, PantallaRegistrarEstudiante, PantallaRegistrarBeneficio1, PantallaRegistrarBeneficio2, PantallaRegistrarAlgunOtroBeneficio, PantallaRegistroAntiguedadSisben, PantallaRegistroTiposPoblacion, PantallaRegistroUniversidad, PantallaRegistroCarreras, PantallaRegistroCiudad, PantallaDocumentos	
Atributos:	
Contratos	
1. Desplegar pantalla	
DesplegarPantalla	
Responsabilidades privadas	
EnviarEvento	InterfaceUsuario

Manejador:

Tarjeta para la clase Manejador con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: Manejador	
Descripción: superclase heredada por los manejadores del sistema	
Modulo: Principal	
Estereotipo: Control	

Propiedades: Abstracta	
Superclases:	
Subclases: ManejadorPrincipal, ManejadorFoest, ManejadorRegistroFuncionario, ManejadorRegistrarEstudiantes, ManejadorRegistroBeneficio1, ManejadorRegistroBeneficio2, ManejadorAlgunOtroBeneficio, ManejadorTipoPoblacion, ManejadorUniversidades, ManejadorCarreras, ManejadorCiudad, ManejadorDocumentos, ManejadorCertificado, ManejadorEstadísticas.	
Atributos:	
Contratos	
1. Manejar evento	
Maneja evento	
Responsabilidades privadas	
DesplegarPantalla	InterfaceUsuario (1)
Manejar evento Estudiantesfoest	ManejadorFoest (2)
Manejar evento salir	

ManejadorPrincipal:

Tarjeta para la clase ManejadorPrincipal con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: ManejadorPrincipal	
Descripción: es el encargado de desplegar la pantalla principal de interacción con el usuario y después desplegar las diferentes funciones a los manejadores apropiados	
Modulo: Principal	
Estereotipo: Control	
Propiedades: Concreta	
Superclases: manejador	
Subclases:	
Atributos:	
Contratos	
1. Manejar evento	
Manejar evento	
Responsabilidades privadas	
Maneja EventoValidar	ManejadorRegistroFuncionario (2)

Pantalla ingreso:

Tarjeta para la clase pantalla ingreso con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaIngreso	
Descripción: pantalla ingreso (P1)	
Modulo: Principal	
Estereotipo: Borde	
Propiedades: concreta	
Superclases: pantalla	
Subclases:	
Atributos:	

Tarjeta para la clase datos con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: datos	
Descripción: superclase para todas las clases entidad	
Modulo: Dominio	
Estereotipo: Entidad	
Propiedades: Abstracta	
Superclases:	
Subclases: RegistrarUsuarioFuncionario, RegistrarUsuarioEstudiantes, RegistrarBeneficio1, RegistroBeneficio2, RegistrarAlgunOtroBeneficio, RegistrarAntiguedadSisben, RegistrarTipoPoblacion, RegistrarUniversidades, RegistrarCarreras, RegistrarCiudad, CertificadoEstudiante, Estadísticas	
Atributos:	

Registro:

Este módulo se compone del módulo usuario funcionario, usuario estudiante, beneficio1, beneficio2, algún otro beneficio, antigüedad Sisbén, tipos de poblacion, universidades, carreras, ciudad, certificados estudiantes e InterfaceBasedeDatosRegistro.

Usuario funcionario:

Este módulo se compone de las clases ManejadorFuncionario, PantallaFuncionario, PantallaRegistrarFuncionario, PantallaFuncionarioRegistrado y RegistrarUsuarioFuncionario.

Tarjeta para la clase ManejadorFuncionario con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: ManejadorFuncionario	
Descripción: se encarga de todo lo relacionado con el registro del usuario funcionario	
Modulo: RegistrarUsuarioFuncionario	
Estereotipo: Control	
Propiedades: concreta	
Superclases: Manejador	
Subclases:	
Atributos:	
contratos	
1. Manejar evento	
Manejar evento	
2. Registrar usuario	
CrearRegistroUsuarioFuncionario	
Solicita ValidarRegistroUsuarioFuncionario	InterfaceBasedeDatosRegistro
Responsabilidades privadas	
Maneja el evento "registrar"	
Solicita CrearRegistroUsuarioFuncionario	InterfaceBasedeDatosRegistro
Solicita FuncionariosRegistrados	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	
Solicita ActualizarRegistroUsuarioFuncionario	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	
Solicita EliminarRegistroUsuarioFuncionario	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaRegistrarFuncionario con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaFuncionario	
Descripción: superclase con diseño común para las pantallas de registro de usuario funcionario.	
Modulo: registro, usuario funcionario	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	
Subclases: pantallaregistrarfuncionario, PantallaFuncionarioRegistrado	

Atributos:	
despliega	

Tarjeta para la clase PantallaRegistrarFuncionario con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaRegistrarFuncionario	
Descripción: pantalla crear funcionarios (p32)	
Modulo: registro, usuario funcionario	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: PantallaFuncionario	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaFuncionarioRegistrado con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaFuncionarioRegistrado	
Descripción: pantalla funcionarios registrados (p33)	
Modulo: registro, usuario funcionario	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaFuncionario	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarUsuarioFuncionario con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: RegistrarUsuarioFuncionario	
Descripción: el usuario para que pueda utilizar el sistema, debe estar registrado, en el registro se debe ingresar la información del usuario: nombre, apellido, teléfono, email, login y password.	
Modulo: registro, usuario funcionario	
Estereotipo: entidad	

Propiedades: concreta
Superclases: datos
Subclases:
Atributos:

Usuarios estudiantes:

Este módulo se compone de las clases ManejadorEstudiantes, PantallaEstudiantes, PantallaRegistrarEstudiante, PantallaEstudianteRegistrado y RegistrarUsuarioEstudiantes.

Tarjeta para la clase ManejadorEstudiantes con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: ManejadorEstudiantes
Descripción: se encarga de todo lo relacionado con el registro del usuario estudiante
Modulo: RegistrarUsuarioEstudiantes
Estereotipo: Control
Propiedades: Concreta
Superclases: Manejador
Subclases:
Atributos:
1. Manejar evento
Manejar evento
2. Registrar estudiante
Maneja el evento "registrar"
Solicita CrearRegistroUsuarioEstudiante
InterfaceBasedeDatosRegistro
Responsabilidades privadas
EstudiantesRegistrados
InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"
InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"
InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaEstudiantes con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaEstudiantes
Descripción: superclase con diseño común para las pantallas de registro de usuario estudiante
Modulo: registro, usuario estudiante
Estereotipo: borde

Propiedades: abstracta
Superclases: pantalla
Subclases: PantallaRegistrarEstudiante, PantallaEstudianteRegistrado
Atributos:
despliega

Tarjeta para la clase PantallaRegistrarEstudiante con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaRegistrarEstudiante
Descripción: pantalla registro estudiantes (p4)
Modulo: registro, usuario estudiante
Estereotipo: borde
Propiedades: concreta
Superclases: PantallaEstudiantes
Subclases:
Atributos:

Tarjeta para la clase PantallaEstudianteRegistrado con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaEstudianteRegistrado
Descripción: pantalla estudiantes registrados (p3)
Modulo: registro, usuarios estudiantes
Estereotipo: borde
Propiedades: concreta
Superclases: PantallaEstudiantes
Subclases:
Atributos:

Tarjeta para la clase RegistrarUsuarioEstudiantes con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: RegistrarUsuarioEstudiantes	
Descripción: en el registro se debe ingresar la información del usuario: N° carpeta, fecha de radicación, beneficio 1, beneficio 2, 1° vez, renovación, reintegro, primer apellido, segundo apellido, primer nombre, segundo nombre, tipo de documento, número de documento Expedido en, edad, población especial, género, teléfono 1; teléfono 2, dirección, correo electrónico, antigüedad Sisbén en años, puntaje del Sisbén, recibe algún otro beneficio, universidad sede nivel carrera, modalidad, semestre, valor del semestre, porcentaje a aplicar monto aprobado, otro beneficio, materias perdidas promedio 2018_1 y 2017, horas sociales cumplidas, dependencia donde realizo horas sociales, Horario, ciudad a la que viaja, días que viaja, valor por 2 recorridos valor semanal, valor mensual bruto, valor a pagar por mes, meses a girar y total semestre; también los datos del representante legal si es un menor de edad.	
Modulo: registro, usuarios estudiantes	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos:	

Beneficio1:

Este módulo se compone de las clases ManejadorBeneficio1, PantallaDiseñoBeneficio1, PantallaRegistrarBeneficio1, PantallaBeneficio1 y RegistrarBeneficio1.

Tarjeta para la clase ManejadorBeneficio1 con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: ManejadorBeneficio1	
Descripción: se encarga de todo lo relacionado con el registro del beneficio1	
Modulo: RegistrarBeneficio1	
Estereotipo: Control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos:	
Contrato	
1. Manejar eventos	
manejarevento	
2. Registrar beneficio	
Maneja el evento "registrar"	

Solicita CrearRegistroBeneficio1	InterfaceBasedeDatosRegistro
Responsabilidades privadas	
Solicita Beneficio1	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoBeneficio1 con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaDiseñoBeneficio1	
Descripción: Superclase con diseño común para las pantallas de registro de beneficio1	
Modulo: registro, beneficio1	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	
Subclases: PantallaRegistrarBeneficio1, PantallaBeneficio1	
Atributos:	
Despliega	

Tarjeta para la clase PantallaRegistrarBeneficio1 con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaRegistrarBeneficio1	
Descripción: pantalla registro beneficio 1 (p7)	
Modulo: registro, beneficio1	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoBeneficio1	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaBeneficio1 con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaBeneficio1	
Descripción: pantalla beneficio1 (p6)	
Modulo: registro, beneficio1	

Estereotipo: borde
Propiedades: concreta
Superclases: PantallaDiseñoBeneficio1
Subclases:
Atributos:

Tarjeta para la clase RegistrarBeneficio1 con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: RegistrarBeneficio1
Descripción: en el registro se debe ingresar la información del beneficio: nombre.
Modulo: registro, beneficio1
Estereotipo: entidad
Propiedades: concreta
Superclases: datos
Subclases:
Atributos:

Beneficio2:

Este módulo se compone de las clases ManejadorBeneficio2, PantallaDiseñoBeneficio2, PantallaRegistrarBeneficio2, PantallaBeneficio2 y RegistrarBeneficio2.

Tarjeta para la clase ManejadorBeneficio2 con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: ManejadorBeneficio2
Descripción: se encarga de todo lo relacionado con el registro del beneficio2
Modulo: RegistrarBeneficio2
Estereotipo: Control
Propiedades: concreta
Superclases: manejador
Subclases:
Atributos:
Contratos
1. Manejar eventos
manejarevento

2. Registrar beneficio 2	
Maneja el evento "registrar"	
Solicita CrearRegistroBeneficio2	InterfaceBasedeDatosRegistro
Responsabilidades privadas	
Solicita Beneficio2	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoBeneficio2 con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaDiseñoBeneficio2	
Descripción: Superclase con diseño común para las pantallas de registro de beneficio2	
Modulo: registro, beneficio2	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: PantallaRegistrarBeneficio2, PantallaBeneficio2	
Subclases:	
Atributos:	
Despliega	

Tarjeta para la clase PantallaRegistrarBeneficio2 con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaRegistrarBeneficio2	
Descripción: pantalla registro beneficio 2 (p7)	
Modulo: registro, beneficio2	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoBeneficio2	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaBeneficio2 con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaBeneficio2	
Descripción: pantalla beneficio2 (p6)	
Modulo: registro, beneficio2	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoBeneficio2	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarBeneficio2 con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: RegistrarBeneficio2	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, beneficio2	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos:	

Algún Otro Beneficio:

Este módulo se compone de las clases ManejadorAlgúnOtroBeneficio, PantallaDiseñoAlgúnOtroBeneficio, PantallaRegistrarAlgúnOtroBeneficio, PantallaAlgúnOtroBeneficio, y RegistrarAlgúnOtroBeneficio.

Tarjeta para la clase ManejadorAlgúnOtroBeneficio con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: ManejadorAlgúnOtroBeneficio	
Descripción: se encarga de todo lo relacionado con el registro del beneficio2	
Modulo: RegistrarAlgúnOtroBeneficio	
Estereotipo: Control	

Propiedades: Concreta	
Superclases: Manejador	
Subclases:	
Atributos:	
Contratos	
1. Manejar eventos	
manejarevento	
Responsabilidades privadas	
Maneja el evento "registrar"	InterfaceBasedeDatosRegistro
Solicita Algunotrobeneficio	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoAlgunOtroBeneficio con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaDiseñoAlgunOtroBeneficio
Descripción: pantalla registro algún otro beneficio (p13)
Modulo: registro, Algunotrobeneficio
Estereotipo: borde
Propiedades: abstracta
Superclases: pantalla
Subclases: PantallaRegistrarAlgunOtroBeneficio, PantallaAlgunOtroBeneficio
Atributos:
despliega

Tarjeta para la clase PantallaRegistrarAlgunOtroBeneficio con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaRegistrarAlgunOtroBeneficio
Descripción: pantalla registro algún otro beneficio (p13)
Modulo: registro, Algunotrobeneficio
Estereotipo: borde
Propiedades: concreta
Superclases: PantallaDiseñoAlgunOtroBeneficio
Subclases:
Atributos:

Tarjeta para la clase PantallaAlgunOtroBeneficio con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaAlgunOtroBeneficio	
Descripción: pantalla algún otro beneficio (p12)	
Modulo: registro, Algunotrobeficio	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoAlgunOtroBeneficio	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarAlgunOtroBeneficio con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: RegistrarAlgunOtroBeneficio	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, Algunotrobeficio	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos:	

Antigüedad Sisbén:

Este módulo se compone de las clases ManejadorAntigüedadSisben, PantallaDiseñoAntigüedadSisben, PantallaRegistroAntigüedadSisben, PantallaAntigüedadSisben, y RegistrarAntigüedadSisben.

Tarjeta para la clase ManejadorAntigüedadSisben con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: ManejadorAntigüedadSisben	
Descripción: se encarga de todo lo relacionado con el registro de la antigüedad del Sisbén	
Modulo: RegistrarAntigüedadSisben	
Estereotipo: Control	
Propiedades: Concreta	
Superclases: Manejador	
Subclases:	
Atributos:	
contratos	
1. Manejar evento	
manejarevento	
Responsabilidades privadas	
antigüedadSisbén	
Maneja el evento "registrar"	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoAntigüedadSisben con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaDiseñoAntigüedadSisben	
Descripción: Superclase con diseño común para las pantallas de registro de antigüedad Sisbén	
Modulo: registro, AntigüedadSisbén	
Estereotipo: borde	
Propiedades: Abstracta	
Superclases: Pantalla	
Subclases: PantallaRegistroAntigüedadSisben, PantallaAntigüedadSisben	
Atributos:	
despliega	

Tarjeta para la clase PantallaRegistroAntigüedadSisben con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaRegistroAntigüedadSisben	
Descripción: pantalla registro antigüedad Sisbén (p16)	
Modulo: registro, AntigüedadSisbén	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoAntigüedadSisben	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaAntigüedadSisben con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaAntigüedadSisben	
Descripción: pantalla antigüedad Sisbén (p15)	
Modulo: registro, AntigüedadSisbén	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoAntigüedadSisben	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarAntigüedadSisben con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: RegistrarAntigüedadSisben	
Descripción: en el registro se debe ingresar la información de la antigüedad Sisbén: cantidad de años.	
Modulo: registro, AntigüedadSisbén	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos:	

Tipos de poblacion:

Este módulo se compone de las clases ManejadorTipoPoblacion, PantallaDiseñoTiposPoblacion, PantallaRegistroTiposPoblacion, PantallaTiposPoblacion, y RegistrarTipoPoblacion.

Tarjeta para la clase ManejadorTipoPoblacion con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: ManejadorTipoPoblacion	
Descripción: se encarga de todo lo relacionado con el registro del tipo de poblacion.	
Modulo: RegistrarTipoPoblacion	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
1. Manejar evento	
manejarevento	
Responsabilidades privadas	
TipoPoblacion	
Maneja el evento "registrar"	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoTiposPoblacion con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaDiseñoTiposPoblacion	
Descripción: Superclase con diseño común para las pantallas de registro de tipos de poblacion	
Modulo: registro, TipoPoblacion	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	
Subclases: PantallaRegistroTiposPoblacion, PantallaTiposPoblacion	
Atributos:	
despliega	

Tarjeta para la clase PantallaRegistroTiposPoblacion con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaRegistroTiposPoblacion	
Descripción: pantalla registro tipos de poblacion (p19)	
Modulo: registro, TipoPoblacion	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoTiposPoblacion	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaTiposPoblacion con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaTiposPoblacion	
Descripción: pantalla tipos de poblacion (p18)	
Modulo: registro, TipoPoblacion	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoTiposPoblacion	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarTipoPoblacion con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: RegistrarTipoPoblacion	
Descripción: en el registro se debe ingresar la información del tipo de poblacion: nombre.	
Modulo: registro, TipoPoblacion	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos:	

Universidades:

Este módulo se compone de las clases ManejadorUniversidades, PantallaDiseñoUniversidad, PantallaRegistroUniversidad, PantallaUniversidades, y RegistrarUniversidades.

Tarjeta para la clase ManejadorUniversidades con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: ManejadorUniversidades	
Descripción: se encarga de todo lo relacionado con el registro de las universidades.	
Modulo: RegistrarUniversidades	
Estereotipo: Control	
Propiedades: Concreta	
Superclases:	
Subclases:	
Atributos:	
Contratos	
1. Manejar evento	
manejarevento	
Responsabilidades privadas	
Universidades	
Maneja el evento "registrar"	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoUniversidad con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaDiseñoUniversidad	
Descripción: Superclase con diseño común para las pantallas de registro de universidades	
Modulo: registro, Universidad	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	
Subclases: PantallaRegistroUniversidad, PantallaUniversidades	
Atributos:	
despliega	

Tarjeta para la clase PantallaRegistroUniversidad con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaRegistroUniversidad	
Descripción: pantalla registro universidades (p22)	
Modulo: registro, Universidad	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoUniversidad	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaUniversidades con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaUniversidades	
Descripción: pantalla universidades (p21)	
Modulo: registro, Universidad	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoUniversidad	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarUniversidades con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: RegistrarUniversidades	
Descripción: en el registro se debe ingresar la información de la universidad: nombre y porcentaje a aplicar.	
Modulo: registro, Universidad	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos:	

Carreras:

Este módulo se compone de las clases ManejadorCarreras, PantallaDiseñoCarreras, PantallaRegistroCarreras, PantallaCarreras, y RegistrarCarreras.

Tarjeta para la clase ManejadorCarreras con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: ManejadorCarreras	
Descripción: se encarga de todo lo relacionado con el registro de las carreras.	
Modulo: RegistrarCarreras	
Estereotipo: Control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos:	
Contratos	
1. Manejar eventos	
manejarevento	
Responsabilidades privadas	
Carreras	
Maneja el evento "registrar"	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoCarreras con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaDiseñoCarreras	
Descripción: superclase con diseño común para las pantallas de registro de carrera	
Modulo: registro, Carreras	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	
Subclases: PantallaRegistroCarreras, PantallaCarreras	
Atributos:	
despliega	

Tarjeta para la clase PantallaRegistroCarreras con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaRegistroCarreras	
Descripción: pantalla registro carreras (p25)	
Modulo: registro, Carreras	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoCarreras	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaCarreras con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaCarreras	
Descripción: pantalla carreras (p24)	
Modulo: registro, Carreras	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoCarreras	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarCarreras con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: RegistrarCarreras	
Descripción: en el registro se debe ingresar la información de la carrera: nombre.	
Modulo: registro, Carreras	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos:	

Ciudad:

Este módulo se compone de las clases ManejadorCiudad, PantallaDiseñoCiudad, PantallaRegistroCiudad, PantallaCiudad, y RegistrarCiudad.

Tarjeta para la clase ManejadorCiudad con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: ManejadorCiudad	
Descripción: se encarga de todo lo relacionado con el registro de las ciudades.	
Modulo: RegistrarCiudad	
Estereotipo: Control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos:	
Contratos	
1. Manejar eventos	
manejarevento	
Responsabilidades privadas	
Ciudad	
Maneja el evento "registrar"	InterfaceBasedeDatosRegistro
Maneja el evento "actualizar"	InterfaceBasedeDatosRegistro
Maneja el evento "eliminar"	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoCiudad con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaDiseñoCiudad	
Descripción: Superclase con diseño común para las pantallas de registro de ciudades	
Modulo: registro, Ciudad	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	
Subclases: PantallaRegistroCiudad, PantallaCiudad	
Atributos:	
despliega	

Tarjeta para la clase PantallaRegistroCiudad con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaRegistroCiudad	
Descripción: pantalla registro ciudad (p28)	
Modulo: registro, Ciudad	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoCiudad	
Subclases:	
Atributos:	

Tarjeta para la clase PantallaCiudad con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaCiudad	
Descripción: pantalla ciudad (p27)	
Modulo: registro, Ciudad	
Estereotipo: borde	
Propiedades: concreta	
Superclases: PantallaDiseñoCiudad	
Subclases:	
Atributos:	

Tarjeta para la clase RegistrarCiudad con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: RegistrarCiudad	
Descripción: en el registro se debe ingresar la información de la ciudad: nombre.	
Modulo: registro, Ciudad	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos:	

Certificación estudiantes:

Este módulo se compone de las clases ManejadorDocumentos, PantallaDiseñoDocumentos, PantallaDocumentos, PantallaCertificado, PantallaEstadísticas, CertificadoEstudiante y Estadísticas.

Tarjeta para la clase ManejadorDocumentos con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: ManejadorDocumentos	
Descripción: se encarga de todo lo relacionado con informes como estadísticas y certificados.	
Modulo: CertificadoEstudiante y Estadísticas.	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos:	
Contratos	
1. Manejar eventos	
Manejarevento	
Responsabilidades privadas	
Maneja el evento "certificado"	InterfaceBasedeDatosRegistro
Maneja el evento "estadísticas"	InterfaceBasedeDatosRegistro

Tarjeta para la clase PantallaDiseñoDocumentos con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaDiseñoDocumentos	
Descripción: superclase con diseño común para las pantallas de solicitud de documentos	
Modulo: Certificados y Estadísticas.	
Estereotipo: borde	
Propiedades: abstracta	
Superclases: pantalla	
Subclases: PantallaDocumentos, PantallaCertificado, PantallaEstadísticas	
Atributos:	
despliega	

Tarjeta para la clase PantallaDocumentos con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaDocumentos
Descripción: pantalla registro ciudad (p34)
Modulo: Certificados y Estadísticas.
Estereotipo: borde
Propiedades: concreta
Superclases: PantallaDiseñoDocumentos
Subclases:
Atributos:

Tarjeta para la clase PantallaCertificado con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaCertificado
Descripción: pantalla certificados (p31)
Modulo: Certificados.
Estereotipo: borde
Propiedades: concreta
Superclases: PantallaDiseñoDocumentos
Subclases:
Atributos:

Tarjeta para la clase PantallaEstadísticas con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: PantallaEstadísticas
Descripción: pantalla estadísticas (p30)
Modulo: Estadísticas.
Estereotipo: borde
Propiedades: concreta
Superclases: PantallaDiseñoDocumentos
Subclases:
Atributos:

Tarjeta para la clase CertificadoEstudiante con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: CertificadoEstudiante
Descripción: en el certificado se debe ingresar la información del estudiante: documento de identidad.

Modulo: registro, Certificados
Estereotipo: entidad
Propiedades: concreta
Superclases: datos
Subclases:
Atributos:

Tarjeta para la clase Estadísticas con responsabilidades, colaboraciones, jerarquía y contratos.

Clase: Estadísticas
Descripción: en estadísticas se toman los datos de los estudiantes registrado en la base de datos.
Modulo: registro, Estadísticas
Estereotipo: entidad
Propiedades: concreta
Superclases: datos
Subclases:
Atributos:

Interface base de datos:

Tarjeta para la clase InterfaceBasedeDatosRegistro con colaboraciones, jerarquía y contratos.

Clase: InterfaceBasedeDatosRegistro	
Descripción: la información de los usuarios, beneficios y otros se almacena en la base de datos foest	
Modulo: Registro interfaceBD	
Estereotipo: interface	
Propiedades: concreta	
Superclases:	
Subclases:	
Atributos:	
contratos	
1. Registrar usuario funcionario	
Solicita ValidarRegistroUsuarioFuncionario	BasedeDatosfoest
Solicita CrearRegistroUsuarioFuncionario	BasedeDatosfoest
Solicita Funcionarioregistrado	BasedeDatosfoest
Solicita ActualizarRegistroUsuarioFuncionario	BasedeDatosfoest

Solicita EliminarRegistroUsuarioFuncionario	BasedeDatosfoest
2. Registrar usuario estudiante	
Solicita CrearRegistroUsuarioEstudiante	BasedeDatosfoest
Solicita EstudianteRegistrado	BasedeDatosfoest
Solicita ActualizarRegistroUsuarioEstudiante	BasedeDatosfoest
Solicita EliminarRegistroUsuarioEstudiante	BasedeDatosfoest
3. Registrar beneficio 1	
Solicita CrearRegistroBeneficio1	BasedeDatosfoest
Solicita beneficio1	BasedeDatosfoest
Solicita ActualizarRegistroBeneficio1	BasedeDatosfoest
Solicita EliminarRegistrobeneficio1	BasedeDatosfoest
4. Registrar beneficio 2	
Solicita CrearRegistroBeneficio2	BasedeDatosfoest
Solicita beneficio2	BasedeDatosfoest
Solicita ActualizarRegistroBeneficio2	BasedeDatosfoest
Solicita EliminarRegistrobeneficio2	BasedeDatosfoest
5. Registrar algún otro beneficio	
Solicita CrearRegistroAlgunOtroBeneficio	BasedeDatosfoest
Solicita Algunotrobeneficio	BasedeDatosfoest
Solicita ActualizarRegistroAlgunOtroBeneficio	BasedeDatosfoest
Solicita EliminarRegistroAlgunOtroBeneficio	BasedeDatosfoest
6. Registrar antigüedad Sisbén	
Solicita CrearRegistroAntiguedadsisben	BasedeDatosfoest
Solicita AntigüedadSisbén	BasedeDatosfoest
Solicita ActualizarRegistroantigüedadSisbén	BasedeDatosfoest
Solicita EliminarRegistroantigüedadSisbén	BasedeDatosfoest
7. Registrar tipo de poblacion	
Solicita CrearRegistroTipoPoblacion	BasedeDatosfoest
Solicita TipoPoblacion	BasedeDatosfoest
Solicita ActualizarRegistroTipoPoblacion	BasedeDatosfoest
Solicita EliminarRegistroTipoPoblacion	BasedeDatosfoest
8. Registrar universidades	
Solicita CrearRegistroUniversidad	BasedeDatosfoest
Solicita universidades	BasedeDatosfoest
Solicita ActualizarRegistroUniversidad	BasedeDatosfoest
Solicita EliminarRegistroUniversidad	BasedeDatosfoest
9. Registrar carreras	
Solicita CrearRegistroCarreras	BasedeDatosfoest
Solicita Carreras	BasedeDatosfoest
Solicita ActualizarRegistroCarreras	BasedeDatosfoest
Solicita EliminarRegistroCarreras	BasedeDatosfoest
10. Registrar ciudad	

Solicita CrearRegistroCiudad	BasedeDatosfoest
Solicita Ciudad	BasedeDatosfoest
Solicita ActualizarRegistroCiudad	BasedeDatosfoest
Solicita EliminarRegistroCiudad	BasedeDatosfoest
11. Solicitar documentos	
Solicita PedirCertificado	BasedeDatosfoest
Solicita estadísticas	BasedeDatosfoest

Estudiantes foest:

Tarjeta para la clase ManejadorFoest con responsabilidades colaboraciones, jerarquía y contratos.

Clase: ManejadorFoest	
Descripción: se encarga de enviar las peticiones particulares de opciones a los manejadores especializados para registro de las diferentes opciones.	
Modulo: estudiantes	
Estereotipo: control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos:	
Contratos	
1. Manejar evento	
manejarevento	
2. Ofrecer servicios	
ofrecersevicios	
Maneja el evento "Administrar registro funcionario"	ManejadorRegistroFuncionario
Maneja el evento "Administrar registro estudiante"	ManejadorRegistrarEstudiantes
Maneja el evento "Administrar registro Beneficio1"	ManejadorRegistroBeneficio1
Maneja el evento "Administrar registro Beneficio2"	ManejadorRegistroBeneficio2
Maneja el evento "Administrar registro Algún otro Beneficio"	ManejadorAlgúnOtroBeneficio
Maneja el evento "Administrar registro Antigüedad Sisben"	ManejadorAntigüedadSisben
Maneja el evento "Administrar registro Tipos de poblacion"	ManejadorTipoPoblacion

Maneja el evento "Administrar registro universidades"	ManejadorUniversidades
Maneja el evento "Administrar registro carrera"	ManejadorCarreras
Maneja el evento "Administrar registro ciudad"	ManejadorCiudad
Maneja el evento "Administrar documentos"	ManejadorDocumentos

Pantalla principal:

Tarjeta para la clase PantallaPrincipal con responsabilidades colaboraciones, jerarquía y contratos.

Clase: PantallaPrincipal	
Descripción: pantalla principal (P2)	
Modulo: Principal	
Estereotipo: Borde	
Propiedades: concreta	
Superclases: pantalla	
Subclases:	
Atributos:	

8.1.6 Subsistemas

SubsistemaInterfaceUsuario

Tarjeta para el subsistema SubsistemaInterfaceUsuario

Subsistema: SubsistemaInterfaceUsuario	
Descripción: agrupa todos los objetos que participan en el manejo general de las interfaces de usuario	
Clases: InterfaceUsuario, PantallaIngreso, PantallaPrincipal, PantallaRegistrarFuncionario, PantallaRegistrarEstudiante, PantallaRegistrarBeneficio1, PantallaRegistrarBeneficio2, PantallaRegistrarAlgunOtroBeneficio, PantallaRegistroAntiguedadSisben, PantallaRegistroTiposPoblacion, PantallaRegistroUniversidad, PantallaRegistroCarreras, PantallaRegistroCiudad, PantallaDocumentos	
Contratos	Servidor
1. Desplegar pantalla	InterfaceUsuario

SubsistemaPrincipal

Tarjeta para el subsistema SubsistemaPrincipal

Subsistema: SubsistemaPrincipal	
Descripción: agrupa todos los objetos que participan en el manejo general del sistema	
Clases: ManejadorPrincipal	
Contratos	Servidor
1. Desplegar pantalla	ManejadorPrincipal

Subsistemaregistro

Tarjeta para el subsistema Subsistemaregistro

Subsistema: Subsistemaregistro	
Descripción: agrupa todos los objetos que participan en el manejo de registro de estudiantes	
Clases: ManejadorFuncionario, ManejadorEstudiantes, ManejadorBeneficio1, ManejadorBeneficio2, ManejadorAlgunOtroBeneficio, ManejadorAntiguedadSisben, ManejadorTipoPoblacion, ManejadorUniversidades, ManejadorCarreras, ManejadorCiudad, ManejadorDocumentos, ManejadorCertificado, ManejadorEstadísticas, RegistrarUsuarioFuncionario, RegistrarAntiguedadSisben, RegistrarUsuarioEstudiantes, RegistrarBeneficio1, RegistrarBeneficio2, RegistrarAlgunOtroBeneficio, RegistrarTipoPoblacion, RegistrarUniversidades, RegistrarCarreras, RegistrarCiudad, CertificadoEstudiante, Estadísticas	
InterfaceBasedeDatosRegistro	
Contratos	Servidor
1. Manejar evento	ManejadorEstudiantes, ManejadorBeneficio1, ManejadorBeneficio2, ManejadorAlgunOtroBeneficio, ManejadorAntiguedadSisben, ManejadorTipoPoblacion, ManejadorUniversidades, ManejadorCarreras, ManejadorCiudad, ManejadorDocumentos, ManejadorCertificado, ManejadorEstadísticas,
1. Registrar usuario estudiante	ManejadorEstudiantes

SubsistemaEstudianteFoest

Tarjeta para el subsistema SubsistemaEstudianteFoest

Subsistema: SubsistemaEstudianteFoest	
Descripción: agrupa todos los objetos que participan en el manejo del sistema de gestión de documentos Foest	
Clases: ManejadorPrincipal	
Contratos	Servidor
1. Manejar evento	ManejadorPrincipal

2. Ofrecer servicios estudiantes foest	ManejadorPrincipal
---	--------------------

Interface usuario:

Tarjeta para la clase InterfaceUsuario con responsabilidades, colaboraciones, jerarquía, contratos y subsistemas.

Clase: InterfaceUsuario	
Descripción: toda la interacción con el usuario se realiza por medio de la interface de usuario	
Modulo: InterfaceUsuario	
Estereotipo: Borde	
Propiedades: Concreta	
Superclases:	
Subclases:	
Atributos:	
Contratos	
1. Desplegar pantalla	
DesplegarPantalla	Pantalla; PantallaIngreso, PantallaPrincipal, PantallaFuncionarioRegistrado, PantallaEstudianteRegistrado, PantallaBeneficio1, PantallaBeneficio2, PantallaAlgunOtroBeneficio, PantallaAntiguedadSisben, PantallaTiposPoblacion, PantallaUniversidades, PantallaCarreras, PantallaCiudad, PantallaDocumentos, PantallaCertificado, PantallaEstadisticas
2. Enviar evento	
EnviarEvento	Manejador; SubsistemaPrincipal, Subsistemaregistro, SubsistemaEstudianteFoest

Pantalla

Tarjeta para la clase pantalla no hay modificación de clase

Principal:

Tarjeta para la clase Manejador con responsabilidades, colaboraciones, jerarquía, contratos y subsistemas.

Clase: Manejador	
Descripción: superclase heredada por los manejadores del sistema	
Modulo: Principal	
Estereotipo: Control	
Propiedades: Abstracta	
Superclases:	
Subclases: ManejadorPrincipal, ManejadorFoest, ManejadorRegistroFuncionario, ManejadorRegistrarEstudiantes, ManejadorRegistroBeneficio1, ManejadorRegistroBeneficio2, ManejadorAlgunOtroBeneficio, ManejadorTipoPoblacion, ManejadorUniversidades, ManejadorCarreras, ManejadorCiudad, ManejadorDocumentos, ManejadorCertificado, ManejadorEstadísticas.	
Atributos:	
Contratos	
1. Manejar evento	
Maneja evento	
Responsabilidades privadas	
DesplegarPantalla	SubsistemaInterfaceUsuario (1)
Manejar evento Estudiantesfoest	SubsistemaEstudianteFoest (2)
Manejar evento salir	

ManejadorPrincipal:

Tarjeta para la clase ManejadorPrincipal con responsabilidades, colaboraciones, jerarquía, contratos y subsistemas.

Clase: ManejadorPrincipal	
Descripción: es el encargado de desplegar la pantalla principal de interacción con el usuario y después desplegar las diferentes funciones a los manejadores apropiados	
Modulo: Principal	
Estereotipo: Control	
Propiedades: Concreta	
Superclases: manejador	
Subclases:	
Atributos:	
Contratos	
1. Manejar evento	
Manejar evento	

Responsabilidades privadas	
Maneja EventoValidar	Subsistemaregistro (2)

Pantalla ingreso:

No hay modificaciones de clases.

Registro:

Este módulo se compone del módulo usuario funcionario, usuario estudiante, beneficio1, beneficio2, algún otro beneficio, antigüedad Sisbén, tipos de poblacion, universidades, carreras, ciudad, certificados estudiantes e InterfaceBasedeDatosRegistro.

El módulo registro no hay modificación de clases

Estudiantes foest:

Tarjeta para la clase ManejadorFoest con responsabilidades colaboraciones, jerarquía y contratos.

Clase: ManejadorFoest	
Descripción: se encarga de enviar las peticiones particulares de opciones a los manejadores especializados para registro de las diferentes opciones.	
Modulo: estudiantes	
Estereotipo: control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos:	
Contratos	
1. Manejar evento	
manejarevento	

2. Ofrecer servicios	
ofrecerse servicios	
Responsabilidades privadas	
Maneja el evento "Administrar registro funcionario"	Subsistema registro
Maneja el evento "Administrar registro estudiante"	Subsistema registro
Maneja el evento "Administrar registro Beneficio1"	Subsistema registro
Maneja el evento "Administrar registro Beneficio2"	Subsistema registro
Maneja el evento "Administrar registro Algún otro Beneficio"	Subsistema registro
Maneja el evento "Administrar registro Antigüedad Sisbén"	Subsistema registro
Maneja el evento "Administrar registro Tipos de población"	Subsistema registro
Maneja el evento "Administrar registro universidades"	Subsistema registro
Maneja el evento "Administrar registro carrera"	Subsistema registro
Maneja el evento "Administrar registro ciudad"	Subsistema registro
Maneja el evento "Administrar documentos"	Subsistema registro

La clase PantallaPrincipal no hay modificación.

8.1.7 Atributos

Interface usuario:

Tarjeta para la clase InterfaceUsuario con responsabilidades, colaboraciones, jerarquía, contratos y atributos

Clase: InterfaceUsuario
Descripción: toda la interacción con el usuario se realiza por medio de la interface de usuario
Modulo: InterfaceUsuario
Estereotipo: Borde
Propiedades: Concreta
Superclases:
Subclases:

Atributos: manejador, pantalla	
Contratos	
2. Desplegar pantalla	
DesplegarPantalla (Pantalla)	Pantalla; PantallaIngreso, PantallaPrincipal, PantallaFuncionarioRegistrado, PantallaEstudianteRegistrado, PantallaBeneficio1, PantallaBeneficio2, PantallaAlgunOtroBeneficio, PantallaAntiguedadSisben, PantallaTiposPoblacion, PantallaUniversidades, PantallaCarreras, PantallaCiudad, PantallaDocumentos, PantallaCertificado, PantallaEstadisticas
2. Enviar evento	
EnviarEvento (Evento, Manejador)	Manejador; SubsistemaPrincipal, Subsistemaregistro, SubsistemaEstudianteFoest

Pantalla

Tarjeta para la clase pantalla con responsabilidades, colaboraciones, jerarquía, contratos y atributos

Clase: pantalla	
Descripción: pantalla heredada por las demás clases del tipo de pantalla	
Modulo: InterfaceUsuario	
Estereotipo: Borde	
Propiedades: Abstracta	
Superclases:	
Subclases: PantallaIngreso, PantallaPrincipal, PantallaRegistrarFuncionario, PantallaRegistrarEstudiante, PantallaRegistrarBeneficio1, PantallaRegistrarBeneficio2, PantallaRegistrarAlgunOtroBeneficio, PantallaRegistroAntiguedadSisben, PantallaRegistroTiposPoblacion, PantallaRegistroUniversidad, PantallaRegistroCarreras, PantallaRegistroCiudad, PantallaDocumentos	
Atributos: InterfaceUsuario, Manejador	
Contratos	
1. Desplegar pantalla	
DesplegarPantalla ()	
Responsabilidades privadas	
EnviarEvento (Evento)	InterfaceUsuario

Manejador:

Tarjeta para la clase Manejador con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: Manejador	
Descripción: superclase heredada por los manejadores del sistema	
Modulo: Principal	
Estereotipo: Control	
Propiedades: Abstracta	
Superclases:	
Subclases: ManejadorPrincipal, ManejadorFoest, ManejadorRegistroFuncionario, ManejadorRegistrarEstudiantes, ManejadorRegistroBeneficio1, ManejadorRegistroBeneficio2, ManejadorAlgunOtroBeneficio, ManejadorTipoPoblacion, ManejadorUniversidades, ManejadorCarreras, ManejadorCiudad, ManejadorDocumentos, ManejadorCertificado, ManejadorEstadísticas.	
Atributos: InterfaceUsuario, Pantalla, ManejadorFoest, manejador	
Contratos	
2. Manejar evento	
Manejarevento (evento)	
Responsabilidades privadas	
DesplegarPantalla ()	SubsistemaInterfaceUsuario (1)
ManejarEventoEstudiantesfoest ()	SubsistemaEstudianteFoest (2)
ManejarEventoSalir ()	

ManejadorPrincipal:

Tarjeta para la clase ManejadorPrincipal con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: ManejadorPrincipal
Descripción: es el encargado de desplegar la pantalla principal de interacción con el usuario y después desplegar las diferentes funciones a los manejadores apropiados
Modulo: Principal
Estereotipo: Control
Propiedades: Concreta
Superclases: manejador
Subclases:
Atributos: PantallaIngreso, ManejadorFoest, ManejadorRegistroFuncionario
Contratos
2. Manejar evento

Manejarevento (Evento)	
Responsabilidades privadas	
ManejaEventoValidar ()	Subsistemaregistro (2)

Datos

Tarjeta para la clase datos está limitada en relaciones, por lo cual no se agregan atributos

Registro:

Este módulo se compone del módulo usuario funcionario, usuario estudiante, beneficio1, beneficio2, algún otro beneficio, antigüedad Sisbén, tipos de poblacion, universidades, carreras, ciudad, certificados estudiantes e InterfaceBasedeDatosRegistro.

Usuario funcionario:

Este módulo se compone de las clases ManejadorFuncionario, PantallaFuncionario, PantallaRegistrarFuncionario, PantallaFuncionarioRegistrado y RegistrarUsuarioFuncionario.

Tarjeta para la clase ManejadorFuncionario con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: ManejadorFuncionario	
Descripción: se encarga de todo lo relacionado con el registro del usuario funcionario	
Modulo: RegistrarUsuarioFuncionario	
Estereotipo: Control	
Propiedades: concreta	
Superclases: Manejador	
Subclases:	
Atributos: PantallaRegistrarFuncionario, PantallaFuncionarioRegistrado, RegistrarUsuarioFuncionario, InterfaceBasedeDatosRegistro	
contratos	
1. Manejar evento	
Manejarevento (Evento)	

2. Registrar usuario	
CrearRegistroUsuarioFuncionario ()	
ValidarRegistroUsuarioFuncionario (String log, String pass)	InterfaceBasedeDatosRegistro
FuncionariosRegistrados ()	InterfaceBasedeDatosRegistro
Responsabilidades privadas	
FuncionariosRegistrados ()	
ManejaEventoRegistrar ()	InterfaceBasedeDatosRegistro
ManejaEventoActualizar ()	InterfaceBasedeDatosRegistro
ManejaEventoEliminar ()	InterfaceBasedeDatosRegistro

Tarjeta para la clase RegistrarUsuarioFuncionario con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: RegistrarUsuarioFuncionario	
Descripción: el usuario para que pueda utilizar el sistema, debe estar registrado, en el registro se debe ingresar la información del usuario: nombre, apellido, teléfono, email, login y password.	
Modulo: registro, usuario funcionario	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos: nombre, apellido, teléfono, email, login, password, rpassword	

Usuarios estudiantes:

Este módulo se compone de las clases ManejadorEstudiantes, PantallaEstudiantes, PantallaRegistrarEstudiante, PantallaEstudianteRegistrado y RegistrarUsuarioEstudiantes.

Tarjeta para la clase ManejadorEstudiantes con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: ManejadorEstudiantes	
Descripción: se encarga de todo lo relacionado con el registro del usuario estudiante	
Modulo: RegistrarUsuarioEstudiantes	
Estereotipo: Control	

Propiedades: Concreta	
Superclases: Manejador	
Subclases:	
Atributos: PantallaRegistrarEstudiante, PantallaEstudianteRegistrado, RegistrarUsuarioEstudiantes, InterfaceBasedeDatosRegistro	
1. Manejar evento	
Manejarevento (Evento)	
2. Registrar estudiante	
CrearRegistroUsuarioEstudiante ()	
EstudiantesRegistrados ()	InterfaceBasedeDatosRegistro
Responsabilidades privadas	
EstudiantesRegistrados ()	
ManejaEventoRegistrar ()	InterfaceBasedeDatosRegistro
ManejaEventoActualizar ()	InterfaceBasedeDatosRegistro
ManejaEventoEliminar ()	InterfaceBasedeDatosRegistro

Tarjeta para la clase RegistrarUsuarioEstudiantes con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: RegistrarUsuarioEstudiantes	
Descripción: en el registro se debe ingresar la información del usuario: N° carpeta, fecha de radicación, beneficio 1, beneficio 2, 1°vez, renovación, reintegro, primer apellido, segundo apellido, primer nombre, segundo nombre, tipo de documento, numero de documento Expedido en, edad, población especial, genero, teléfono 1; teléfono 2, dirección, correo electrónico, antigüedad Sisbén en años, puntaje del Sisbén, recibe algún otro beneficio, universidad sede nivel carrera, modalidad, semestre, valor del semestre, porcentaje a aplicar monto aprobado, otro beneficio, materias perdidas promedio 2018_1 y 2017, horas sociales cumplidas, dependencia donde realizo horas sociales, Horario, ciudad a la que viaja, días que viaja, valor por 2 recorridos valor semanal, valor mensual bruto , valor a pagar por mes, meses a girar y total semestre; también los datos del representante legal si es un menor de edad.	
Modulo: registro, usuarios estudiantes	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos: N° carpeta, fecha de radicación, beneficio 1, beneficio 2, 1°vez, renovación, reintegro, primer apellido, segundo apellido, primer nombre, segundo nombre, tipo de documento, numero de documento Expedido en, edad, población especial, genero, teléfono 1; teléfono 2, dirección, correo electrónico, antigüedad Sisbén en años, puntaje del Sisbén, recibe algún otro beneficio, universidad sede nivel carrera, modalidad, semestre, valor del semestre, porcentaje a aplicar monto aprobado, otro beneficio, materias perdidas promedio 2018_1 y 2017, horas sociales cumplidas, dependencia donde realizo horas sociales, Horario, ciudad a la que viaja, días que viaja, valor por 2 recorridos valor semanal, valor mensual bruto , valor a pagar por mes, meses a girar y total semestre; también los datos del representante legal si es un menor de edad.	

Beneficio1:

Este módulo se compone de las clases ManejadorBeneficio1, PantallaDiseñoBeneficio1, PantallaRegistrarBeneficio1, PantallaBeneficio1 y RegistrarBeneficio1.

Tarjeta para la clase ManejadorBeneficio1 con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: ManejadorBeneficio1	
Descripción: se encarga de todo lo relacionado con el registro del beneficio1	
Modulo: RegistrarBeneficio1	
Estereotipo: Control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos: PantallaRegistrarBeneficio1, PantallaBeneficio1, RegistrarBeneficio1, InterfaceBasedeDatosRegistro	
Contrato	
1. Manejar eventos	
Manejarevento (Evento)	
2. Registrar beneficio	
CrearRegistroBeneficio1 ()	
Beneficio1()	InterfaceBasedeDatosRegistro
Responsabilidades privadas	
Beneficio1()	
ManejaEventoRegistrar ()	InterfaceBasedeDatosRegistro
ManejaEventoActualizar ()	InterfaceBasedeDatosRegistro
ManejaEventoEliminar ()	InterfaceBasedeDatosRegistro

Tarjeta para la clase RegistrarBeneficio1 con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: RegistrarBeneficio1	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, beneficio1	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	

Subclases: ID, Nombre	
Atributos:	

Beneficio2:

Este módulo se compone de las clases ManejadorBeneficio2, PantallaDiseñoBeneficio2, PantallaRegistrarBeneficio2, PantallaBeneficio2 y RegistrarBeneficio2.

Tarjeta para la clase ManejadorBeneficio2 con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: ManejadorBeneficio2	
Descripción: se encarga de todo lo relacionado con el registro del beneficio2	
Modulo: RegistrarBeneficio2	
Estereotipo: Control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos: PantallaRegistrarBeneficio2, PantallaBeneficio2, RegistrarBeneficio2, InterfaceBasedeDatosRegistro	
Contratos	
1. Manejar eventos	
Manejarevento (Evento)	
2. Registrar beneficio 2	
CrearRegistroBeneficio2 ()	
Beneficio2()	InterfaceBasedeDatosRegistro
Responsabilidades privadas	
Beneficio2()	
ManejaEventoRegistrar ()	InterfaceBasedeDatosRegistro
ManejaEventoActualizar ()	InterfaceBasedeDatosRegistro
ManejaEventoEliminar ()	InterfaceBasedeDatosRegistro

Tarjeta para la clase RegistrarBeneficio2 con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: RegistrarBeneficio2	
Descripción: en el registro se debe ingresar la información del beneficio: nombre.	

Modulo: registro, beneficio2
Estereotipo: entidad
Propiedades: concreta
Superclases: datos
Subclases: ID, Nombre
Atributos:

Algún Otro Beneficio:

Este módulo se compone de las clases ManejadorAlgúnOtroBeneficio, PantallaDiseñoAlgúnOtroBeneficio, PantallaRegistrarAlgúnOtroBeneficio, PantallaAlgúnOtroBeneficio, y RegistrarAlgúnOtroBeneficio.

Tarjeta para la clase ManejadorAlgúnOtroBeneficio con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: ManejadorAlgúnOtroBeneficio
Descripción: se encarga de todo lo relacionado con el registro del beneficio2
Modulo: RegistrarAlgúnOtroBeneficio
Estereotipo: Control
Propiedades: Concreta
Superclases: Manejador
Subclases:
Atributos: PantallaRegistrarAlgúnOtroBeneficio, PantallaAlgúnOtroBeneficio, RegistrarAlgúnOtroBeneficio, InterfaceBasedeDatosRegistro
Contratos
1. Manejar eventos
Manejarevento (Evento)
Responsabilidades privadas
Algúnotrobeneficio ()
ManejaEventoRegistrar ()
ManejaEventoActualizar ()
ManejaEventoEliminar ()
InterfaceBasedeDatosRegistro
InterfaceBasedeDatosRegistro
InterfaceBasedeDatosRegistro

Tarjeta para la clase RegistrarAlgúnOtroBeneficio con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: RegistrarAlgúnOtroBeneficio

Descripción: en el registro se debe ingresar la información del beneficio: nombre.	
Modulo: registro, Algunotrobeneficio	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases: ID, nombre	
Atributos:	

Antigüedad Sisbén:

Este módulo se compone de las clases ManejadorAntigüedadSisben, PantallaDiseñoAntigüedadSisben, PantallaRegistroAntigüedadSisben, PantallaAntigüedadSisben, y RegistrarAntigüedadSisben.

Tarjeta para la clase ManejadorAntigüedadSisben con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: ManejadorAntigüedadSisben	
Descripción: se encarga de todo lo relacionado con el registro de la antigüedad del Sisbén	
Modulo: RegistrarAntigüedadSisben	
Estereotipo: Control	
Propiedades: Concreta	
Superclases: Manejador	
Subclases:	
Atributos: PantallaRegistroAntigüedadSisben, PantallaAntigüedadSisben, RegistrarAntigüedadSisben, InterfaceBasedeDatosRegistro.	
contratos	
1. Manejar evento	
Manejarevento (Evento)	
Responsabilidades privadas	
antigüedadSisbén ()	
ManejaEventoRegistrar ()	InterfaceBasedeDatosRegistro
ManejaEventoActualizar ()	InterfaceBasedeDatosRegistro
ManejaEventoEliminar ()	InterfaceBasedeDatosRegistro

Tarjeta para la clase RegistrarAntigüedadSisben con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: RegistrarAntigüedadSisben

Descripción: en el registro se debe ingresar la información de la antigüedad Sisbén: cantidad de años.	
Modulo: registro, AntigüedadSisbén	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos: ID, Años	

Tipos de poblacion:

Este módulo se compone de las clases ManejadorTipoPoblacion, PantallaDiseñoTiposPoblacion, PantallaRegistroTiposPoblacion, PantallaTiposPoblacion, y RegistrarTipoPoblacion.

Tarjeta para la clase ManejadorTipoPoblacion con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: ManejadorTipoPoblacion	
Descripción: se encarga de todo lo relacionado con el registro del tipo de poblacion.	
Modulo: RegistrarTipoPoblacion	
Estereotipo: Control	
Propiedades:	
Superclases:	
Subclases:	
Atributos: PantallaRegistroTiposPoblacion, PantallaTiposPoblacion, RegistrarTipoPoblacion, InterfaceBasedeDatosRegistro.	
contratos	
1. Manejar evento	
Manejarevento (Evento)	
Responsabilidades privadas	
TipoPoblacion ()	
ManejaEventoRegistrar ()	InterfaceBasedeDatosRegistro
ManejaEventoActualizar ()	InterfaceBasedeDatosRegistro
ManejaEventoEliminar ()	InterfaceBasedeDatosRegistro

Tarjeta para la clase RegistrarTipoPoblacion con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: RegistrarTipoPoblacion

Descripción: en el registro se debe ingresar la información del tipo de poblacion: nombre.	
Modulo: registro, TipoPoblacion	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos: ID, Nombre	

Universidades:

Este módulo se compone de las clases ManejadorUniversidades, PantallaDiseñoUniversidad, PantallaRegistroUniversidad, PantallaUniversidades, y RegistrarUniversidades.

Tarjeta para la clase ManejadorUniversidades con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: ManejadorUniversidades	
Descripción: se encarga de todo lo relacionado con el registro de las universidades.	
Modulo: RegistrarUniversidades	
Estereotipo: Control	
Propiedades: Concreta	
Superclases:	
Subclases:	
Atributos: PantallaRegistroUniversidad, PantallaUniversidades, RegistrarUniversidades, InterfaceBasedeDatosRegistro.	
Contratos	
1. Manejar evento	
Manejarevento (Evento)	
Responsabilidades privadas	
Universidades ()	
ManejaEventoRegistrar ()	InterfaceBasedeDatosRegistro
ManejaEventoActualizar ()	InterfaceBasedeDatosRegistro
ManejaEventoEliminar ()	InterfaceBasedeDatosRegistro

Tarjeta para la clase RegistrarUniversidades con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: RegistrarUniversidades

Descripción: en el registro se debe ingresar la información de la universidad: nombre y porcentaje a aplicar.	
Modulo: registro, Universidad	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos: ID, Nombre, Porcentajes	

Carreras:

Este módulo se compone de las clases ManejadorCarreras, PantallaDiseñoCarreras, PantallaRegistroCarreras, PantallaCarreras, y RegistrarCarreras.

Tarjeta para la clase ManejadorCarreras con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: ManejadorCarreras	
Descripción: se encarga de todo lo relacionado con el registro de las carreras.	
Modulo: RegistrarCarreras	
Estereotipo: Control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos: PantallaRegistroCarreras, PantallaCarreras, RegistrarCarreras, InterfaceBasedeDatosRegistro.	
Contratos	
1. Manejar eventos	
Manejarevento (Evento)	
Responsabilidades privadas	
Carreras ()	
ManejaEventoRegistrar ()	InterfaceBasedeDatosRegistro
ManejaEventoActualizar ()	InterfaceBasedeDatosRegistro
ManejaEventoEliminar ()	InterfaceBasedeDatosRegistro

Tarjeta para la clase RegistrarCarreras con responsabilidades, colaboraciones, jerarquía, contratos y atributos

Clase: RegistrarCarreras
Descripción: en el registro se debe ingresar la información de la carrera: nombre.
Modulo: registro, Carreras
Estereotipo: entidad
Propiedades: concreta
Superclases: datos
Subclases:
Atributos: ID, Nombre

Ciudad:

Este módulo se compone de las clases ManejadorCiudad, PantallaDiseñoCiudad, PantallaRegistroCiudad, PantallaCiudad, y RegistrarCiudad.

Tarjeta para la clase ManejadorCiudad con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: ManejadorCiudad
Descripción: se encarga de todo lo relacionado con el registro de las ciudades.
Modulo: RegistrarCiudad
Estereotipo: Control
Propiedades: concreta
Superclases: manejador
Subclases:
Atributos: PantallaRegistroCiudad, PantallaCiudad, RegistrarCiudad, InterfaceBasedeDatosRegistro
Contratos
1. Manejar eventos
Manejarevento (Evento)
Responsabilidades privadas
Ciudad ()
ManejaEventoRegistrar ()
ManejaEventoActualizar ()
ManejaEventoEliminar ()
InterfaceBasedeDatosRegistro
InterfaceBasedeDatosRegistro
InterfaceBasedeDatosRegistro

Tarjeta para la clase RegistrarCiudad con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: RegistrarCiudad
Descripción: en el registro se debe ingresar la información de la ciudad: nombre.
Modulo: registro, Ciudad
Estereotipo: entidad
Propiedades: concreta
Superclases: datos
Subclases:
Atributos: ID, Nombre

Certificación estudiantes:

Este módulo se compone de las clases ManejadorDocumentos, PantallaDiseñoDocumentos, PantallaDocumentos, PantallaCertificado, PantallaEstadísticas, CertificadoEstudiante y Estadísticas.

Tarjeta para la clase ManejadorDocumentos con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: ManejadorDocumentos
Descripción: se encarga de todo lo relacionado con informes como estadísticas y certificados.
Modulo: CertificadoEstudiante y Estadísticas.
Estereotipo: Control
Propiedades: Concreta
Superclases: manejador
Subclases:
Atributos: PantallaDocumentos, PantallaCertificado, PantallaEstadísticas, CertificadoEstudiante, Estadísticas, InterfaceBasedeDatosRegistro
Contratos
1. Manejar eventos
Manejarevento (Evento)
Responsabilidades privadas
Documentos ()
ManejaEventoCertificado ()
ManejaEventoEstadísticas ()
InterfaceBasedeDatosRegistro
InterfaceBasedeDatosRegistro

Tarjeta para la clase CertificadoEstudiante con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: CertificadoEstudiante	
Descripción: en el certificado se debe ingresar la información del estudiante: documento de identidad.	
Modulo: registro, Certificados	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos: Documento	

Tarjeta para la clase Estadísticas con responsabilidades, colaboraciones, jerarquía, contratos y atributos.

Clase: Estadísticas	
Descripción: en estadísticas se toman los datos de los estudiantes registrado en la base de datos.	
Modulo: registro, Estadísticas	
Estereotipo: entidad	
Propiedades: concreta	
Superclases: datos	
Subclases:	
Atributos: Datos	

Interface base de datos:

No requiere de atributos

Estudiantes foest:

Tarjeta para la clase ManejadorFoest con responsabilidades colaboraciones, jerarquía, contratos y atributos.

Clase: ManejadorFoest	
Descripción: se encarga de enviar las peticiones particulares de opciones a los manejadores especializados para registro de las diferentes opciones.	
Modulo: estudiantes	
Estereotipo: control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos:	
Contratos	
1. Manejar evento	
Manejarevento (Evento)	
2. Ofrecer servicios	
Ofrecersevicios ()	
Responsabilidades privadas	
ManejaEventoAdministrarRegistroFuncionario ()	Subsistemaregistro
ManejaEventoAdministrarRegistroEstudiante ()	Subsistemaregistro
ManejaEventoAdministrarRegistroBeneficio1 ()	Subsistemaregistro
ManejaEventoAdministrarRegistroBeneficio2 ()	Subsistemaregistro
ManejaEventoAdministrarRegistroAlgunOtroBeneficio ()	Subsistemaregistro
ManejaEventoAdministrarRegistroAntiguedadSisben ()	Subsistemaregistro
ManejaEventoAdministrarRegistroTiposPoblacion ()	Subsistemaregistro
ManejaEventoAdministrarRegistroUniversidades ()	Subsistemaregistro
ManejaEventoAdministrarRegistroCarrera ()	Subsistemaregistro
ManejaEventoAdministrarRegistroCiudad ()	Subsistemaregistro
ManejaEventoAdministrarDocumentos ()	Subsistemaregistro

Pantalla principal:

Tarjeta para la clase PantallaPrincipal no requiere de atributos

8.1.8 Algoritmos

Interface usuario:

Tarjeta para la clase InterfaceUsuario con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos.

Clase: InterfaceUsuario	
Descripción: toda la interacción con el usuario se realiza por medio de la interface de usuario	
Modulo: InterfaceUsuario	
Estereotipo: Borde	
Propiedades: Concreta	
Superclases:	
Subclases:	
Atributos: manejador, pantalla	
Contratos	
1. Desplegar pantalla	
DesplegarPantalla (Pantalla) Método encargado de desplegar las pantallas como parámetros. Se delega el despliegue particular a cada pantalla.	Pantalla; PantallaIngreso, PantallaPrincipal, PantallaFuncionarioRegistrado, PantallaEstudianteRegistrado, PantallaBeneficio1, PantallaBeneficio2, PantallaAlgunOtroBeneficio, PantallaAntiguedadSisben, PantallaTiposPoblacion, PantallaUniversidades, PantallaCarreras, PantallaCiudad, PantallaDocumentos, PantallaCertificado, PantallaEstadisticas
2. Enviar evento	
EnviarEvento (Evento, Manejador) Método encargado de recibir eventos del sistema de ventanas a través de las pantallas. Se envía el evento a los a manejadores	Manejador; SubsistemaPrincipal, Subsistemaregistro, SubsistemaEstudianteFoest

Pantalla

Tarjeta para la clase pantalla con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos

Clase: pantalla	
Descripción: pantalla heredada por las demás clases del tipo de pantalla	
Modulo: InterfaceUsuario	
Estereotipo: Borde	
Propiedades: Abstracta	
Superclases:	
Subclases: PantallaIngreso, PantallaPrincipal, PantallaRegistrarFuncionario, PantallaRegistrarEstudiante, PantallaRegistrarBeneficio1, PantallaRegistrarBeneficio2, PantallaRegistrarAlgúnOtroBeneficio, PantallaRegistroAntigüedadSisben, PantallaRegistroTiposPoblacion, PantallaRegistroUniversidad, PantallaRegistroCarreras, PantallaRegistroCiudad, PantallaDocumentos	
Atributos: InterfaceUsuario, Manejador	
Contratos	
1. Desplegar pantalla	
DesplegarPantalla ()	
Método encargado de desplegar la pantalla actual	
Responsabilidades privadas	
EnviarEvento (Evento)	InterfaceUsuario
Método encargado de recibir eventos del sistema de ventanas. Se envía el evento recibido a la InterfaceUsuario	

Manejador:

Tarjeta para la clase Manejador con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos.

Clase: Manejador
Descripción: superclase heredada por los manejadores del sistema
Modulo: Principal
Estereotipo: Control
Propiedades: Abstracta
Superclases:

Subclases: ManejadorPrincipal, ManejadorFoest, ManejadorRegistroFuncionario, ManejadorRegistrarEstudiantes, ManejadorRegistroBeneficio1, ManejadorRegistroBeneficio2, ManejadorAlgunOtroBeneficio, ManejadorTipoPoblacion, ManejadorUniversidades, ManejadorCarreras, ManejadorCiudad, ManejadorDocumentos, ManejadorCertificado, ManejadorEstadísticas.	
Atributos: InterfaceUsuario, Pantalla, ManejadorFoest, manejador	
Contratos	
1. Manejar evento	
Manejarevento (evento)	
Método encargado de recibir eventos del sistema de ventanas a través de la InterfaceUsuario	
Responsabilidades privadas	
DesplegarPantalla ()	SubsistemaInterfaceUsuario (1)
Método encargado de desplegar las pantallas administradas por los manejadores. Se solicita al SubsistemaInterfaceUsuario que las despliegue	
ManejarEventoEstudiantesfoest ()	SubsistemaEstudianteFoest (2)
Método encargado de solicitar al SubsistemaEstudianteFoest que ofrezca los servicios correspondientes	
ManejarEventoSalir ()	
Método encargado de salir del sistema	

ManejadorPrincipal:

Tarjeta para la clase ManejadorPrincipal con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos

Clase: ManejadorPrincipal
Descripción: es el encargado de desplegar la pantalla principal de interacción con el usuario y después desplegar las diferentes funciones a los manejadores apropiados
Modulo: Principal
Estereotipo: Control
Propiedades: Concreta
Superclases: manejador

Subclases:	
Atributos: PantallaIngreso, ManejadorFoest, ManejadorRegistroFuncionario	
Contratos	
1. Manejar evento	
Manejarevento (Evento)	
Método sobrescrito de la clase Manejador, encargado de recibir eventos del sistema de ventanas a través de la InterfaceUsuario	
Responsabilidades privadas	
ManejaEventoValidar ()	Subsistemaregistro (2)
Método encargado de solicitar al SubsistemaEstudianteFoest que de servicio al contrato de "Ofrecer Servicio"	

Datos

Tarjeta para la clase datos está limitada en relaciones, por lo cual no se agregan algoritmos

Registro:

Este módulo se compone del módulo usuario funcionario, usuario estudiante, beneficio1, beneficio2, algún otro beneficio, antigüedad Sisbén, tipos de poblacion, universidades, carreras, ciudad, certificados estudiantes e InterfaceBasedeDatosRegistro.

Usuario funcionario:

Este módulo se compone de las clases ManejadorFuncionario, PantallaFuncionario, PantallaRegistrarFuncionario, PantallaFuncionarioRegistrado y RegistrarUsuarioFuncionario.

Tarjeta para la clase ManejadorFuncionario con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos.

Clase: ManejadorFuncionario	
Descripción: se encarga de todo lo relacionado con el registro del usuario funcionario	
Modulo: RegistrarUsuarioFuncionario	
Estereotipo: Control	
Propiedades: concreta	
Superclases: Manejador	
Subclases:	
Atributos: PantallaRegistrarFuncionario, PantallaFuncionarioRegistrado, RegistrarUsuarioFuncionario, InterfaceBasedeDatosRegistro	
contratos	
1. Manejar evento	
Manejarevento (Evento)	
Método sobrescrito de la clase Manejador, encargado de recibir eventos del sistema de ventanas a través de la InterfaceUsuario	
2. Registrar usuario	
CrearRegistroUsuarioFuncionario ()	
Método encargado de desplegar una pantalla de creación de registro de funcionario a través del contrato de "Registrar Funcionario"	
ValidarRegistroUsuarioFuncionario (String log, String pass)	InterfaceBasedeDatosRegistro
Método encargado de solicitar a la InterfaceBaseDatosRegistro la validación de un usuario a través del contrato de "Registrar Funcionario "	
FuncionariosRegistrados ()	InterfaceBasedeDatosRegistro
Método encargado de solicitar a la InterfaceBaseDatosRegistro la obtención de un RegistrarUsuarioFuncionario a través del contrato de "Registrar Funcionario "	
Responsabilidades privadas	
FuncionariosRegistrados ()	
Método encargado de desplegar una pantalla de obtención de registro de usuario funcionario a través del contrato de "Registrar Funcionario "	
ManejaEventoRegistrar ()	InterfaceBasedeDatosRegistro

Método encargado de solicitar a la InterfaceBaseDatosRegistro la creación de un nuevo RegistrarUsuarioFuncionario a través del contrato de "Registrar Funcionario "	
ManejaEventoActualizar () Método encargado de solicitar a la InterfaceBaseDatosRegistro la actualización de un RegistrarUsuarioFuncionario a través del contrato de "Registrar Funcionario "	InterfaceBasedeDatosRegistro
ManejaEventoEliminar () Método encargado de solicitar a la InterfaceBaseDatosRegistro la eliminación de un RegistrarUsuarioFuncionario a través del contrato de "Registrar Funcionario "	InterfaceBasedeDatosRegistro

Usuarios estudiantes:

Este módulo se compone de las clases ManejadorEstudiantes, PantallaEstudiantes, PantallaRegistrarEstudiante, PantallaEstudianteRegistrado y RegistrarUsuarioEstudiantes.

Tarjeta para la clase ManejadorEstudiantes con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos.

Clase: ManejadorEstudiantes	
Descripción: se encarga de todo lo relacionado con el registro del usuario estudiante	
Modulo: RegistrarUsuarioEstudiantes	
Estereotipo: Control	
Propiedades: Concreta	
Superclases: Manejador	
Subclases:	
Atributos: PantallaRegistrarEstudiante, PantallaEstudianteRegistrado, RegistrarUsuarioEstudiantes, InterfaceBasedeDatosRegistro	
1. Manejar evento	
Manejarevento (Evento)	

Método sobrescrito de la clase Manejador, encargado de recibir eventos del sistema de ventanas a través de la InterfaceUsuario	
2. Registrar estudiante	
<p>CrearRegistroUsuarioEstudiante ()</p> <p>Método encargado de desplegar una pantalla de creación de registro de estudiantes a través del contrato de "Registrar Estudiantes"</p>	
<p>EstudiantesRegistrados ()</p> <p>Método encargado de solicitar a la InterfaceBaseDatosRegistro la obtención de los registros RegistrarUsuarioEstudiantes a través del contrato de "Registrar Estudiantes"</p>	InterfaceBasedeDatosRegistro
Responsabilidades privadas	
<p>EstudiantesRegistrados ()</p> <p>Método encargado de desplegar una pantalla de obtención de los registros de usuarios estudiantes a través del contrato de "Registrar Estudiantes"</p>	
<p>ManejaEventoRegistrar ()</p> <p>Método encargado de solicitar a la InterfaceBaseDatosRegistro la creación de un nuevo RegistrarUsuarioEstudiantes a través del contrato de "Registrar Estudiantes"</p>	InterfaceBasedeDatosRegistro
<p>ManejaEventoActualizar ()</p> <p>Método encargado de solicitar a la InterfaceBaseDatosRegistro la actualización de un RegistrarUsuarioEstudiantes a través del contrato de "Registrar Estudiantes"</p>	InterfaceBasedeDatosRegistro
<p>ManejaEventoEliminar ()</p> <p>Método encargado de solicitar a la InterfaceBaseDatosRegistro la eliminación de un RegistrarUsuarioFuncionario a través del contrato de "Registrar Estudiantes"</p>	InterfaceBasedeDatosRegistro

Beneficio1:

Este módulo se compone de las clases ManejadorBeneficio1, PantallaDiseñoBeneficio1, PantallaRegistrarBeneficio1, PantallaBeneficio1 y RegistrarBeneficio1.

Tarjeta para la clase ManejadorBeneficio1 con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos.

Clase: ManejadorBeneficio1	
Descripción: se encarga de todo lo relacionado con el registro del beneficio1	
Modulo: RegistrarBeneficio1	
Estereotipo: Control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos: PantallaRegistrarBeneficio1, PantallaBeneficio1, RegistrarBeneficio1, InterfaceBasedeDatosRegistro	
Contrato	
1. Manejar eventos	
Manejarevento (Evento)	
Método sobrescrito de la clase Manejador, encargado de recibir eventos del sistema de ventanas a través de la InterfaceUsuario	
2. Registrar beneficio	
CrearRegistroBeneficio1 ()	
Método encargado de desplegar una pantalla de creación de registro de beneficios a través del contrato de "Registrar beneficio 1"	
Beneficio1()	InterfaceBasedeDatosRegistro
Método encargado de solicitar a la InterfaceBaseDatosRegistro la obtención de los registros RegistrarBeneficio1 a través del contrato de "Registrar beneficio 1"	
Responsabilidades privadas	
Beneficio1()	

Método encargado de desplegar una pantalla de obtención de los registros de beneficio 1 a través del contrato de "Registrar beneficio 1"	
ManejaEventoRegistrar () Método encargado de solicitar a la InterfaceBaseDatosRegistro la creación de un nuevo RegistrarBeneficio1 a través del contrato de "Registrar beneficio 1"	InterfaceBasedeDatosRegistro
ManejaEventoActualizar () Método encargado de solicitar a la InterfaceBaseDatosRegistro la actualización de un RegistrarBeneficio1 a través del contrato de "Registrar beneficio 1"	InterfaceBasedeDatosRegistro
ManejaEventoEliminar () Método encargado de solicitar a la InterfaceBaseDatosRegistro la eliminación de un RegistrarBeneficio1 a través del contrato de "Registrar beneficio 1"	InterfaceBasedeDatosRegistro

Beneficio2:

Este módulo se compone de las clases ManejadorBeneficio2, PantallaDiseñoBeneficio2, PantallaRegistrarBeneficio2, PantallaBeneficio2 y RegistrarBeneficio2.

Tarjeta para la clase ManejadorBeneficio2 con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos.

Clase: ManejadorBeneficio2
Descripción: se encarga de todo lo relacionado con el registro del beneficio2
Modulo: RegistrarBeneficio2
Estereotipo: Control
Propiedades: concreta
Superclases: manejador
Subclases:
Atributos: PantallaRegistrarBeneficio2, PantallaBeneficio2, RegistrarBeneficio2, InterfaceBasedeDatosRegistro
Contratos
1. Manejar eventos

<p>Manejarevento (Evento)</p> <p>Método sobrescrito de la clase Manejador, encargado de recibir eventos del sistema de ventanas a través de la InterfaceUsuario</p>	
2. Registrar beneficio 2	
<p>CrearRegistroBeneficio2 ()</p> <p>Método encargado de desplegar una pantalla de creación de registro de beneficios a través del contrato de "Registrar beneficio 2"</p>	
<p>Beneficio2()</p> <p>Método encargado de solicitar a la InterfaceBaseDatosRegistro la obtención de los registros RegistrarBeneficio2 a través del contrato de "Registrar beneficio 2"</p>	InterfaceBasedeDatosRegistro
Responsabilidades privadas	
<p>Beneficio2()</p> <p>Método encargado de desplegar una pantalla de obtención de los registros de beneficio 2 a través del contrato de "Registrar beneficio 2"</p>	
<p>ManejaEventoRegistrar ()</p> <p>Método encargado de solicitar a la InterfaceBaseDatosRegistro la creación de un nuevo RegistrarBeneficio2 a través del contrato de "Registrar beneficio 2"</p>	InterfaceBasedeDatosRegistro
<p>ManejaEventoActualizar ()</p> <p>Método encargado de solicitar a la InterfaceBaseDatosRegistro la actualización de un RegistrarBeneficio2 a través del contrato de "Registrar beneficio 2"</p>	InterfaceBasedeDatosRegistro
<p>ManejaEventoEliminar ()</p> <p>Método encargado de solicitar a la InterfaceBaseDatosRegistro la eliminación de un RegistrarBeneficio2 a través del contrato de "Registrar beneficio 2"</p>	InterfaceBasedeDatosRegistro

Algún Otro Beneficio:

Este módulo se compone de las clases ManejadorAlgúnOtroBeneficio, PantallaDiseñoAlgúnOtroBeneficio, PantallaRegistrarAlgúnOtroBeneficio, PantallaAlgúnOtroBeneficio, y RegistrarAlgúnOtroBeneficio.

Tarjeta para la clase ManejadorAlgúnOtroBeneficio con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos.

Clase: ManejadorAlgúnOtroBeneficio	
Descripción: se encarga de todo lo relacionado con el registro del beneficio2	
Modulo: RegistrarAlgúnOtroBeneficio	
Estereotipo: Control	
Propiedades: Concreta	
Superclases: Manejador	
Subclases:	
Atributos: PantallaRegistrarAlgúnOtroBeneficio, PantallaAlgúnOtroBeneficio, RegistrarAlgúnOtroBeneficio, InterfaceBasedeDatosRegistro	
Contratos	
1. Manejar eventos	
Manejarevento (Evento)	
Método sobrescrito de la clase Manejador, encargado de recibir eventos del sistema de ventanas a través de la InterfaceUsuario	
Responsabilidades privadas	
Algúnotrobeficio ()	
Método encargado de desplegar una pantalla de obtención de los registros de algún otro beneficio a través del contrato de "Registrar algún otro beneficio"	
ManejaEventoRegistrar ()	InterfaceBasedeDatosRegistro
Método encargado de solicitar a la InterfaceBaseDatosRegistro la creación de un nuevo RegistrarAlgúnOtroBeneficio a través del contrato de "Registrar algún otro beneficio"	
ManejaEventoActualizar ()	InterfaceBasedeDatosRegistro

Método encargado de solicitar a la InterfaceBaseDatosRegistro la actualización de un RegistrarAlgunOtroBeneficio a través del contrato de "Registrar algún otro beneficio"	
ManejaEventoEliminar () Método encargado de solicitar a la InterfaceBaseDatosRegistro la eliminación de un RegistrarAlgunOtroBeneficio a través del contrato de "Registrar algún otro beneficio"	InterfaceBasedeDatosRegistro

Antigüedad Sisbén:

Este módulo se compone de las clases ManejadorAntigüedadSisben, PantallaDiseñoAntigüedadSisben, PantallaRegistroAntigüedadSisben, PantallaAntigüedadSisben, y RegistrarAntigüedadSisben.

Tarjeta para la clase ManejadorAntigüedadSisben con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos.

Clase: ManejadorAntigüedadSisben	
Descripción: se encarga de todo lo relacionado con el registro de la antigüedad del Sisbén	
Modulo: RegistrarAntigüedadSisben	
Estereotipo: Control	
Propiedades: Concreta	
Superclases: Manejador	
Subclases:	
Atributos: PantallaRegistroAntigüedadSisben, PantallaAntigüedadSisben, RegistrarAntigüedadSisben, InterfaceBasedeDatosRegistro.	
contratos	
1. Manejar evento	
Manejarevento (Evento)	
Método sobrescrito de la clase Manejador, encargado de recibir eventos del sistema de ventanas a través de la InterfaceUsuario	
Responsabilidades privadas	
antigüedadSisbén ()	

Método encargado de desplegar una pantalla de obtención de los registros de antigüedad Sisbén a través del contrato de "Registrar antigüedad Sisbén"	
ManejaEventoRegistrar () Método encargado de solicitar a la InterfaceBaseDatosRegistro la creación de un nuevo RegistrarAntigüedadSisben a través del contrato de "Registrar antigüedad Sisbén"	InterfaceBasedeDatosRegistro
ManejaEventoActualizar () Método encargado de solicitar a la InterfaceBaseDatosRegistro la actualización de un RegistrarAntigüedadSisben a través del contrato de "Registrar antigüedad Sisbén"	InterfaceBasedeDatosRegistro
ManejaEventoEliminar () Método encargado de solicitar a la InterfaceBaseDatosRegistro la eliminación de un RegistrarAntigüedadSisben a través del contrato de "Registrar antigüedad Sisbén"	InterfaceBasedeDatosRegistro

Tipos de poblacion:

Este módulo se compone de las clases ManejadorTipoPoblacion, PantallaDiseñoTiposPoblacion, PantallaRegistroTiposPoblacion, PantallaTiposPoblacion, y RegistrarTipoPoblacion.

Tarjeta para la clase ManejadorTipoPoblacion con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos.

Clase: ManejadorTipoPoblacion
Descripción: se encarga de todo lo relacionado con el registro del tipo de poblacion.
Modulo: RegistrarTipoPoblacion
Estereotipo: Control
Propiedades:
Superclases:
Subclases:

Atributos: PantallaRegistroTiposPoblacion, PantallaTiposPoblacion, RegistrarTipoPoblacion, InterfaceBasedeDatosRegistro.	
contratos	
1. Manejar evento	
Manejarevento (Evento)	
Método sobrescrito de la clase Manejador, encargado de recibir eventos del sistema de ventanas a través de la InterfaceUsuario	
Responsabilidades privadas	
TipoPoblacion ()	
Método encargado de desplegar una pantalla de obtención de los registros de tipo de poblacion a través del contrato de "Registrar tipo de poblacion"	
ManejaEventoRegistrar ()	InterfaceBasedeDatosRegistro
Método encargado de solicitar a la InterfaceBaseDatosRegistro la creación de un nuevo RegistrarTipoPoblacion a través del contrato de "Registrar tipo de poblacion"	
ManejaEventoActualizar ()	InterfaceBasedeDatosRegistro
Método encargado de solicitar a la InterfaceBaseDatosRegistro la actualización de un RegistrarTipoPoblacion a través del contrato de "Registrar tipo de poblacion"	
ManejaEventoEliminar ()	InterfaceBasedeDatosRegistro
Método encargado de solicitar a la InterfaceBaseDatosRegistro la eliminación de un RegistrarTipoPoblacion a través del contrato de "Registrar tipo de poblacion"	

Universidades:

Este módulo se compone de las clases ManejadorUniversidades, PantallaDiseñoUniversidad, PantallaRegistroUniversidad, PantallaUniversidades, y RegistrarUniversidades.

Tarjeta para la clase ManejadorUniversidades con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos.

Clase: ManejadorUniversidades	
Descripción: se encarga de todo lo relacionado con el registro de las universidades.	
Modulo: RegistrarUniversidades	
Estereotipo: Control	
Propiedades: Concreta	
Superclases:	
Subclases:	
Atributos: PantallaRegistroUniversidad, PantallaUniversidades, RegistrarUniversidades, InterfaceBasedeDatosRegistro.	
Contratos	
1. Manejar evento	
Manejarevento (Evento)	
Método sobrescrito de la clase Manejador, encargado de recibir eventos del sistema de ventanas a través de la InterfaceUsuario	
Responsabilidades privadas	
Universidades ()	
Método encargado de desplegar una pantalla de obtención de los registros de universidades a través del contrato de "Registrar universidades"	
ManejaEventoRegistrar ()	InterfaceBasedeDatosRegistro
Método encargado de solicitar a la InterfaceBaseDatosRegistro la creación de un nuevo RegistrarUniversidades a través del contrato de "Registrar universidades"	
ManejaEventoActualizar ()	InterfaceBasedeDatosRegistro

Método encargado de solicitar a la InterfaceBaseDatosRegistro la actualización de un RegistrarUniversidades a través del contrato de "Registrar universidades"	
ManejaEventoEliminar () Método encargado de solicitar a la InterfaceBaseDatosRegistro la eliminación de un RegistrarUniversidades a través del contrato de "Registrar universidades"	InterfaceBasedeDatosRegistro

Carreras:

Este módulo se compone de las clases ManejadorCarreras, PantallaDiseñoCarreras, PantallaRegistroCarreras, PantallaCarreras, y RegistrarCarreras.

Tarjeta para la clase ManejadorCarreras con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos.

Clase: ManejadorCarreras	
Descripción: se encarga de todo lo relacionado con el registro de las carreras.	
Modulo: RegistrarCarreras	
Estereotipo: Control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos: PantallaRegistroCarreras, PantallaCarreras, RegistrarCarreras, InterfaceBasedeDatosRegistro.	
Contratos	
1. Manejar eventos	
Manejarevento (Evento)	
Método sobrescrito de la clase Manejador, encargado de recibir eventos del sistema de ventanas a través de la InterfaceUsuario	
Responsabilidades privadas	
Carreras ()	

Método encargado de desplegar una pantalla de obtención de los registros de carreras a través del contrato de "Registrar carreras"	
ManejaEventoRegistrar () Método encargado de solicitar a la InterfaceBaseDatosRegistro la creación de un nuevo RegistrarCarreras a través del contrato de "Registrar carreras"	InterfaceBasedeDatosRegistro
ManejaEventoActualizar () Método encargado de solicitar a la InterfaceBaseDatosRegistro la actualización de un RegistrarCarreras a través del contrato de "Registrar carreras"	InterfaceBasedeDatosRegistro
ManejaEventoEliminar () Método encargado de solicitar a la InterfaceBaseDatosRegistro la eliminación de un RegistrarCarreras a través del contrato de "Registrar carreras"	InterfaceBasedeDatosRegistro

Ciudad:

Este módulo se compone de las clases ManejadorCiudad, PantallaDiseñoCiudad, PantallaRegistroCiudad, PantallaCiudad, y RegistrarCiudad.

Tarjeta para la clase ManejadorCiudad con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos.

Clase: ManejadorCiudad
Descripción: se encarga de todo lo relacionado con el registro de las ciudades.
Modulo: RegistrarCiudad
Estereotipo: Control
Propiedades: concreta
Superclases: manejador
Subclases:
Atributos: PantallaRegistroCiudad, PantallaCiudad, RegistrarCiudad, InterfaceBasedeDatosRegistro
Contratos
1. Manejar eventos

<p>Manejarevento (Evento)</p> <p>Método sobrescrito de la clase Manejador, encargado de recibir eventos del sistema de ventanas a través de la InterfaceUsuario</p>	
Responsabilidades privadas	
<p>Ciudad ()</p> <p>Método encargado de desplegar una pantalla de obtención de los registros de ciudad a través del contrato de "Registrar ciudad"</p>	
<p>ManejaEventoRegistrar ()</p> <p>Método encargado de solicitar a la InterfaceBaseDatosRegistro la creación de un nuevo RegistrarCiudad a través del contrato de "Registrar ciudad"</p>	InterfaceBasedeDatosRegistro
<p>ManejaEventoActualizar ()</p> <p>Método encargado de solicitar a la InterfaceBaseDatosRegistro la actualización de un RegistrarCiudad a través del contrato de "Registrar ciudad"</p>	InterfaceBasedeDatosRegistro
<p>ManejaEventoEliminar ()</p> <p>Método encargado de solicitar a la InterfaceBaseDatosRegistro la eliminación de un RegistrarCiudad a través del contrato de "Registrar ciudad"</p>	InterfaceBasedeDatosRegistro

Certificación estudiantes:

Este módulo se compone de las clases ManejadorDocumentos, PantallaDiseñoDocumentos, PantallaDocumentos, PantallaCertificado, PantallaEstadísticas, CertificadoEstudiante y Estadísticas.

Tarjeta para la clase ManejadorDocumentos con responsabilidades, colaboraciones, jerarquía, contratos, atributos y algoritmos.

Clase: ManejadorDocumentos	
Descripción: se encarga de todo lo relacionado con informes como estadísticas y certificados.	
Modulo: CertificadoEstudiante y Estadísticas.	
Estereotipo: Control	
Propiedades: Concreta	
Superclases: manejador	
Subclases:	
Atributos: PantallaDocumentos, PantallaCertificado, PantallaEstadísticas, CertificadoEstudiante, Estadísticas, InterfaceBasedeDatosRegistro	
Contratos	
1. Manejar eventos	
Manejarevento (Evento)	
Método sobrescrito de la clase Manejador, encargado de recibir eventos del sistema de ventanas a través de la InterfaceUsuario	
Responsabilidades privadas	
Documentos ()	
Método encargado de desplegar una pantalla de selección de los registros almacenados en la base de datos a través del contrato de "Documentos"	
ManejaEventoCertificado ()	InterfaceBasedeDatosRegistro
Método encargado de solicitar a la InterfaceBaseDatosRegistro la creación de un certificado a través del contrato de "Certificado Estudiante"	
ManejaEventoEstadísticas ()	InterfaceBasedeDatosRegistro
Método encargado de solicitar a la InterfaceBaseDatosRegistro la creación de estadísticas a través del contrato de "estadísticas"	

Interface base de datos:

Tarjeta para la clase InterfaceBasedeDatosRegistro con colaboraciones, contratos, atributos y algoritmos.

Clase: InterfaceBasedeDatosRegistro	
Descripción: la información de los usuarios, beneficios y otros se almacena en la base de datos foest	
Modulo: Registro interfaceBD	
Estereotipo: borde	
Propiedades: concreta	
Superclases:	
Subclases:	
Atributos:	
contratos	
1. Registrar usuario funcionario	
Solicita ValidarRegistroUsuarioFuncionario (RegistrarUsuarioFuncionario, String log, String pass) Método encargado de solicitar a la BasedeDatosfoest la validación de un usuario	BasedeDatosfoest
Solicita CrearRegistroUsuarioFuncionario (RegistrarUsuarioFuncionario) Método encargado de solicitar a la BasedeDatosfoest la creación de un nuevo RegistrarUsuarioFuncionario	BasedeDatosfoest
Solicita Funcionarioregistrado (RegistrarUsuarioFuncionario, String log) Método encargado de solicitar a la BasedeDatosfoest la obtención de un RegistrarUsuarioFuncionario	BasedeDatosfoest
Solicita ActualizarRegistroUsuarioFuncionario (RegistrarUsuarioFuncionario) Método encargado de solicitar a la BasedeDatosfoest la actualización de un RegistrarUsuarioFuncionario	BasedeDatosfoest
Solicita EliminarRegistroUsuarioFuncionario (RegistrarUsuarioFuncionario) Método encargado de solicitar a la BasedeDatosfoest la eliminación de un	BasedeDatosfoest

RegistrarUsuarioFuncionario	
2. Registrar usuario estudiante	
Solicita CrearRegistroUsuarioEstudiante (RegistrarUsuarioEstudiantes) Método encargado de solicitar a la BasedeDatosfoest la creación de un nuevo RegistrarUsuarioEstudiantes	BasedeDatosfoest
Solicita EstudianteRegistrado (RegistrarUsuarioEstudiantes) Método encargado de solicitar a la BasedeDatosfoest la obtención de los RegistrarUsuarioEstudiantes	BasedeDatosfoest
Solicita ActualizarRegistroUsuarioEstudiante (RegistrarUsuarioEstudiantes) Método encargado de solicitar a la BasedeDatosfoest la actualización de un RegistrarUsuarioEstudiantes	BasedeDatosfoest
Solicita EliminarRegistroUsuarioEstudiante (RegistrarUsuarioEstudiantes) Método encargado de solicitar a la BasedeDatosfoest la eliminación de un RegistrarUsuarioEstudiantes	BasedeDatosfoest
3. Registrar beneficio 1	
Solicita CrearRegistroBeneficio1 (RegistrarBeneficio1) Método encargado de solicitar a la BasedeDatosfoest la creación de un nuevo RegistrarBeneficio1	BasedeDatosfoest
Solicita beneficio1 (RegistrarBeneficio1) Método encargado de solicitar a la BasedeDatosfoest la obtención de los RegistrarBeneficio1	BasedeDatosfoest
Solicita ActualizarRegistroBeneficio1 (RegistrarBeneficio1) Método encargado de solicitar a la BasedeDatosfoest la actualización de un RegistrarBeneficio1	BasedeDatosfoest

<p>Solicita EliminarRegistrobeneficio1 (RegistrarBeneficio1)</p> <p>Método encargado de solicitar a la BasedeDatosfoest la eliminación de un RegistrarBeneficio1</p>	BasedeDatosfoest
4. Registrar beneficio 2	
<p>Solicita CrearRegistroBeneficio2 (RegistrarBeneficio2)</p> <p>Método encargado de solicitar a la BasedeDatosfoest la creación de un nuevo RegistrarBeneficio2</p>	BasedeDatosfoest
<p>Solicita beneficio2 (RegistrarBeneficio2)</p> <p>Método encargado de solicitar a la BasedeDatosfoest la obtención de los RegistrarBeneficio2</p>	BasedeDatosfoest
<p>Solicita ActualizarRegistroBeneficio2 (RegistrarBeneficio2)</p> <p>Método encargado de solicitar a la BasedeDatosfoest la actualización de un RegistrarBeneficio2</p>	BasedeDatosfoest
<p>Solicita EliminarRegistrobeneficio2 (RegistrarBeneficio2)</p> <p>Método encargado de solicitar a la BasedeDatosfoest la eliminación de un RegistrarBeneficio2</p>	BasedeDatosfoest
5. Registrar algún otro beneficio	
<p>Solicita CrearRegistroAlgunOtroBeneficio (RegistrarAlgunOtroBeneficio)</p> <p>Método encargado de solicitar a la BasedeDatosfoest la creación de un nuevo RegistrarAlgunOtroBeneficio</p>	BasedeDatosfoest
<p>Solicita Algunotrobeneficio (RegistrarAlgunOtroBeneficio)</p> <p>Método encargado de solicitar a la BasedeDatosfoest la obtención de los RegistrarAlgunOtroBeneficio</p>	BasedeDatosfoest
<p>Solicita ActualizarRegistroAlgunOtroBeneficio (RegistrarAlgunOtroBeneficio)</p>	BasedeDatosfoest

Método encargado de solicitar a la Base de Datos foest la actualización de un Registrar Algun Otro Beneficio	
Solicita Eliminar Registro Algun Otro Beneficio (Registrar Algun Otro Beneficio)	Base de Datos foest
Método encargado de solicitar a la Base de Datos foest la eliminación de un Registrar Algun Otro Beneficio	
6. Registrar antigüedad Sisbén	
Solicita Crear Registro Antigüedad adsisben (Registrar Antigüedad Sisben)	Base de Datos foest
Método encargado de solicitar a la Base de Datos foest la creación de un nuevo Registrar Antigüedad Sisben	
Solicita Antigüedad Sisbén (Registrar Antigüedad Sisben)	Base de Datos foest
v	
Solicita Actualizar Registro antigüedad Sisbén (Registrar Antigüedad Sisben)	Base de Datos foest
Método encargado de solicitar a la Base de Datos foest la actualización de un Registrar Antigüedad Sisben	
Solicita Eliminar Registro antigüedad Sisbén (Registrar Antigüedad Sisben)	Base de Datos foest
Método encargado de solicitar a la Base de Datos foest la eliminación de un Registrar Antigüedad Sisben	
7. Registrar tipo de población	
Solicita Crear Registro Tipo Poblacion (Registrar Tipo Poblacion)	Base de Datos foest
Método encargado de solicitar a la Base de Datos foest la creación de un nuevo Registrar Tipo Poblacion	
Solicita Tipo Poblacion (Registrar Tipo Poblacion)	Base de Datos foest
Método encargado de solicitar a la Base de Datos foest la obtención de los Registrar Tipo Poblacion	

<p>Solicita ActualizarRegistroTipoPoblacion (RegistrarTipoPoblacion)</p> <p>Método encargado de solicitar a la BasedeDatosfoest la actualización de un RegistrarTipoPoblacion</p>	BasedeDatosfoest
<p>Solicita EliminarRegistroTipoPoblacion (RegistrarTipoPoblacion)</p> <p>Método encargado de solicitar a la BasedeDatosfoest la eliminación de un RegistrarTipoPoblacion</p>	BasedeDatosfoest
8. Registrar universidades	
<p>Solicita CrearRegistroUniversidad (RegistrarUniversidades)</p> <p>Método encargado de solicitar a la BasedeDatosfoest la creación de un nuevo RegistrarUniversidades</p>	BasedeDatosfoest
<p>Solicita universidades (RegistrarUniversidades)</p> <p>Método encargado de solicitar a la BasedeDatosfoest la obtención de los RegistrarUniversidades</p>	BasedeDatosfoest
<p>Solicita ActualizarRegistroUniversidad (RegistrarUniversidades)</p> <p>Método encargado de solicitar a la BasedeDatosfoest la actualización de un RegistrarUniversidades</p>	BasedeDatosfoest
<p>Solicita EliminarRegistroUniversidad (RegistrarUniversidades)</p> <p>Método encargado de solicitar a la BasedeDatosfoest la eliminación de un RegistrarUniversidades</p>	BasedeDatosfoest
9. Registrar carreras	
<p>Solicita CrearRegistroCarreras (RegistrarCarreras)</p> <p>Método encargado de solicitar a la BasedeDatosfoest la creación de un nuevo RegistrarCarreras</p>	BasedeDatosfoest
<p>Solicita Carreras (RegistrarCarreras)</p>	BasedeDatosfoest

Método encargado de solicitar a la BasedeDatosfoest la obtención de los RegistrarCarreras	
Solicita ActualizarRegistroCarreras (RegistrarCarreras)	BasedeDatosfoest
Método encargado de solicitar a la BasedeDatosfoest la actualización de un RegistrarCarreras	
Solicita EliminarRegistroCarreras (RegistrarCarreras)	BasedeDatosfoest
Método encargado de solicitar a la BasedeDatosfoest la eliminación de un RegistrarCarreras	
10. Registrar ciudad	
Solicita CrearRegistroCiudad (RegistrarCiudad)	BasedeDatosfoest
Método encargado de solicitar a la BasedeDatosfoest la creación de un nuevo RegistrarCiudad	
Solicita Ciudad (RegistrarCiudad)	BasedeDatosfoest
Método encargado de solicitar a la BasedeDatosfoest la obtención de los RegistrarCiudad	
Solicita ActualizarRegistroCiudad (RegistrarCiudad)	BasedeDatosfoest
Método encargado de solicitar a la BasedeDatosfoest la actualización de un RegistrarCiudad	
Solicita EliminarRegistroCiudad (RegistrarCiudad)	BasedeDatosfoest
Método encargado de solicitar a la BasedeDatosfoest la eliminación de un RegistrarCiudad	
11. Solicitar documentos	
Solicita PedirCertificado (PedirCertificado)	BasedeDatosfoest
Método encargado de solicitar a la BasedeDatosfoest los datos de un RegistrarUsuarioEstudiantes	
Solicita estadísticas (Estadísticas)	BasedeDatosfoest

Método encargado de solicitar a la Base de Datos foest los datos de los RegistrarUsuarioEstudiantes

Estudiantes foest:

Tarjeta para la clase ManejadorFoest con responsabilidades colaboraciones, jerarquía, contratos, atributos y algoritmos.

Clase: ManejadorFoest	
Descripción: se encarga de enviar las peticiones particulares de opciones a los manejadores especializados para registro de las diferentes opciones.	
Modulo: estudiantes	
Estereotipo: control	
Propiedades: concreta	
Superclases: manejador	
Subclases:	
Atributos:	
Contratos	
1. Manejar evento	
Manejarevento (Evento)	
Método sobrescrito de la clase Manejador, encargado de recibir eventos del sistema de ventanas a través de la InterfaceUsuario	
2. Ofrecer servicios	
Ofrecersevicios ()	
Método encargado de hacer solicitudes de consultar, y administración de registros	
Responsabilidades privadas	
ManejaEventoAdministrarRegistroFuncionario ()	Subsistemaregistro
Método encargado de hacer la solicitud de administración de registros al Subsistemaregistro	
ManejaEventoAdministrarRegistroEstudiante ()	Subsistemaregistro
Método encargado de hacer la solicitud de administración de registros al Subsistemaregistro	

ManejaEventoAdministrarRegistroBeneficio1 () Método encargado de hacer la solicitud de administración de registros al Subsistemaregistro	Subsistemaregistro
ManejaEventoAdministrarRegistroBeneficio2 () Método encargado de hacer la solicitud de administración de registros al Subsistemaregistro	Subsistemaregistro
ManejaEventoAdministrarRegistroAlgunOtroBeneficio () Método encargado de hacer la solicitud de administración de registros al Subsistemaregistro	Subsistemaregistro
ManejaEventoAdministrarRegistroAntiguedadSisben () Método encargado de hacer la solicitud de administración de registros al Subsistemaregistro	Subsistemaregistro
ManejaEventoAdministrarRegistroTiposPoblacion () Método encargado de hacer la solicitud de administración de registros al Subsistemaregistro	Subsistemaregistro
ManejaEventoAdministrarRegistroUniversidades () Método encargado de hacer la solicitud de administración de registros al Subsistemaregistro	Subsistemaregistro
ManejaEventoAdministrarRegistroCarrera () Método encargado de hacer la solicitud de administración de registros al Subsistemaregistro	Subsistemaregistro
ManejaEventoAdministrarRegistroCiudad () Método encargado de hacer la solicitud de administración de registros al Subsistemaregistro	Subsistemaregistro
ManejaEventoAdministrarDocumentos () Método encargado de hacer la solicitud de administración de registros al Subsistemaregistro	Subsistemaregistro

Pantalla principal:

Tarjeta para la clase PantallaPrincipal no requiere de algoritmos

8.2. Diseño de sistema

8.2.1 interfaces graficas

Interface usuario:

Tarjeta para la clase InterfaceUsuario modificada según diseño grafico

Clase: InterfaceUsuario	
Descripción: toda la interacción con el usuario se realiza por medio de la interface de usuario	
Modulo: InterfaceUsuario	
Estereotipo: Borde	
Propiedades: Concreta	
Superclases:	
Subclases:	
Atributos: manejador, pantalla	
Contratos	
1. Desplegar pantalla	
DesplegarPantalla (Pantalla) Método encargado de desplegar las pantallas como parámetros. Se delega el despliegue particular a cada pantalla.	Pantalla; PantallaIngreso, PantallaPrincipal, PantallaFuncionarioRegistrado, PantallaEstudianteRegistrado, PantallaBeneficio1, PantallaBeneficio2, PantallaAlgunOtroBeneficio, PantallaAntiguedadSisben, PantallaTiposPoblacion, PantallaUniversidades, PantallaCarreras, PantallaCiudad, PantallaDocumentos, PantallaCertificado, PantallaEstadisticas
2. Enviar evento	
EnviarEvento (Evento) Método encargado de recibir eventos del sistema de ventanas a través de las pantallas. Se envía el evento a los a manejadores	Manejador; SubsistemaPrincipal, Subsistemaregistro, SubsistemaEstudianteFoest

Pantalla

Tarjeta para la Superclase pantalla modificada según diseño grafico

Clase: pantalla	
Descripción: pantalla heredada por las demás clases del tipo de pantalla	
Modulo: InterfaceUsuario	
Estereotipo: Borde	
Propiedades: Abstracta	
Superclases:	
Subclases: PantallaIngreso, PantallaPrincipal, PantallaRegistrarFuncionario, PantallaRegistrarEstudiante, PantallaRegistrarBeneficio1, PantallaRegistrarBeneficio2, PantallaRegistrarAlgunOtroBeneficio, PantallaRegistroAntiguedadSisben, PantallaRegistroTiposPoblacion, PantallaRegistroUniversidad, PantallaRegistroCarreras, PantallaRegistroCiudad, PantallaDocumentos	
Atributos: InterfaceUsuario, Manejador	
Contratos	
1. Desplegar pantalla	
DesplegarPantalla ()	
Método encargado de desplegar la pantalla actual	

8.2.2 Bases de datos

Diseño de la base de datos para la tabla correspondiente a la clase RegistrarUsuarioFuncionario

ID	NOMBRE	APELLIDO	TELEFONO	EMAIL	LOGIN	PASSWORD
1	Alejandro	Rozo	3118169125	rarv15@misena.edu.co	Alejandro	1A2B3C

Diseño de la base de datos para la tabla correspondiente a la clase RegistrarUsuarioEstudiantes

ITEM	NOCARPETA	FECHARADICACION	BENEFICIO1	BENEFICIO2	1VEZ	RN

REI	PAPELLIDO	SAPELLIDO	PNOMBRE	SNOMBRE	ID_US	NODOCUMENTO

EXPEDIDOEN	EDAD	POESPECIAL	GENERO	TEL1	TEL2	DIRECCION	CORREOE

ANTIGSISBEN	PUNTASISBEN	ROBROBENE	UNIVERSIDAD	SEDE	NIVEL	CARRERA

MODALIDAD	SEMSTRE	VALORSEMESTRE	PORCENTAJEA	MONTOAPRO	OTROBENE

MATERIASPER	PRO20181	PRO2017	HSCUMPLIDAS	DEPENDENCIA1	DEPENDENCIA2

DEPENDENCIA3	NOCUENTA	HORARIO	DIASVIAJADOS	CIUDADVIAJA	DIAS

VALORREOCR	VALORSEMANA	VALORMESBRUTO	VALORMESAPAGAR	MESES

TOTALSEMESTRE	PAPELLIDOR	SAPELLIDOR	PNOMBRER	SNOMBRER	NOCEDULAR

EXPEDIDAR	ESTADORAT	OBSEVACIONRAT

Diseño de la base de datos para la tabla correspondiente a la clase RegistrarBeneficio1

ID_BENEFICIO1	NOMBENEFICIO1
3	EXCELENCIA ACADEMICA

Diseño de la base de datos para la tabla correspondiente a la clase RegistrarBeneficio2

ID_BENEFICIO2	NOMBENEFICIO2
1	TRANSPORTE

Diseño de la base de datos para la tabla correspondiente a la clase RegistrarAlgunOtroBeneficio

IDALGUN	NOMBREALGUN
3	SER PILO PAGA

Diseño de la base de datos para la tabla correspondiente a la clase RegistrarAntiguedadSisben

ID_ASISBEN	NOMASISBEN
2	2-MES

Diseño de la base de datos para la tabla correspondiente a la clase RegistrarTipoPoblacion

ID_POBLACION	NOMPOBLACION
3	DISCAPACIDAD

Diseño de la base de datos para la tabla correspondiente a la clase RegistrarUniversidades

ID_UNIVERSIDAD	NOMUNIVERSIDAD	UNIPORCENTAJE
2	COLEGIO MAYOR DE NUESTRA SEÑORA DEL ROSARIO	30

Diseño de la base de datos para la tabla correspondiente a la clase RegistrarCarreras

ID_CARRERA	NOMCARRERA
8	ARQUITECTURA

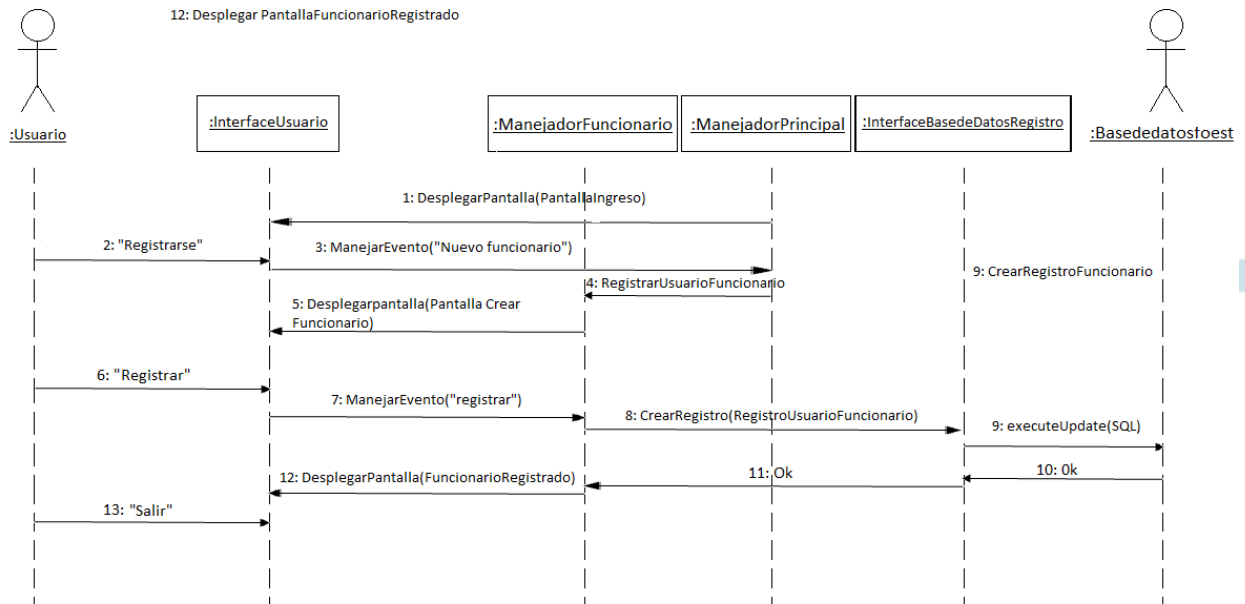
Diseño de la base de datos para la tabla correspondiente a la clase RegistrarCiudad

ID_CIUADAD	NOMCIUDAD
8	BOGOTA

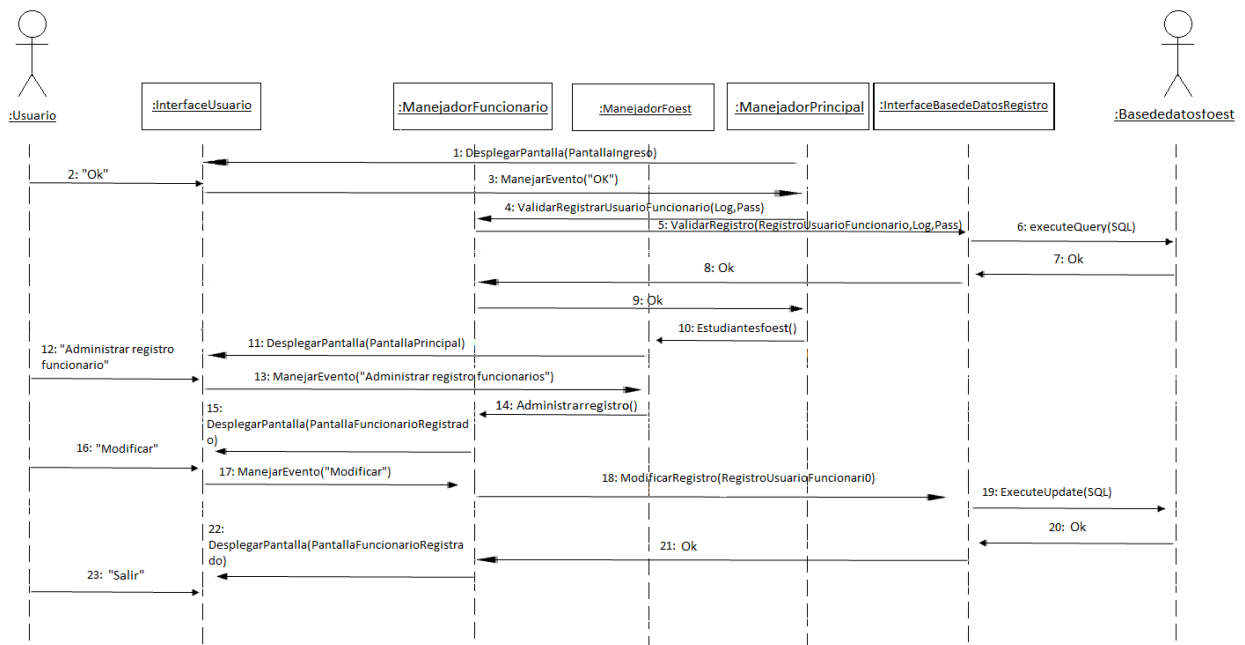


8.3. Diagrama de Secuencia del Diseño

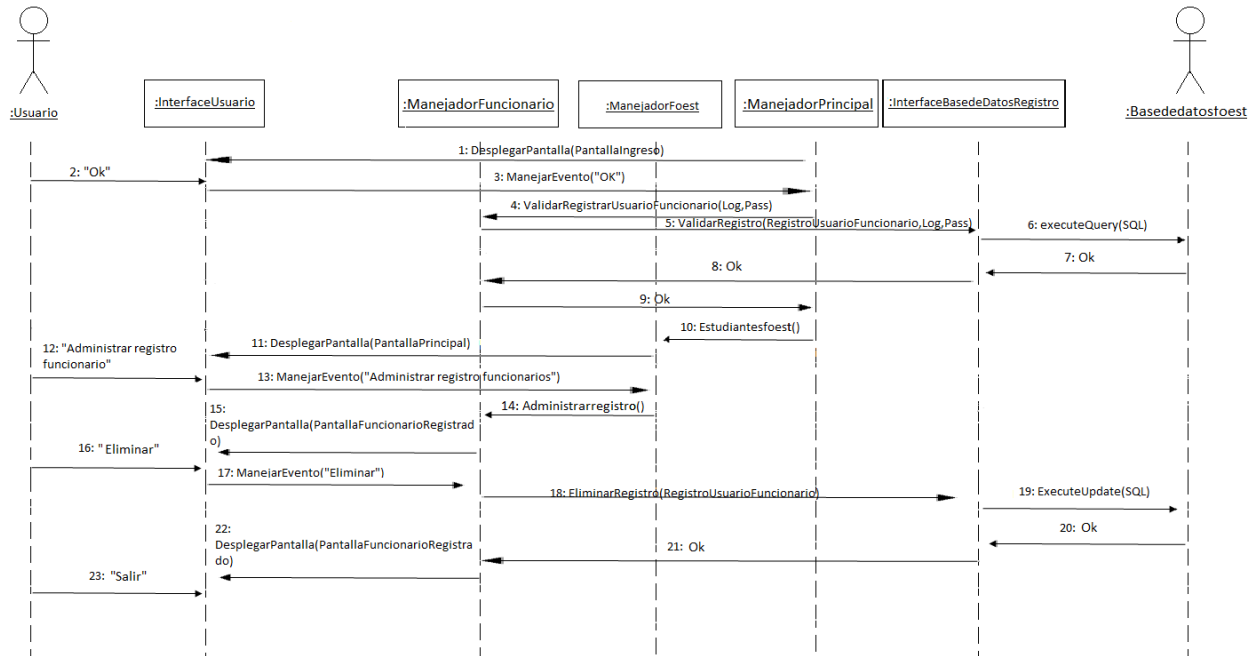
8.3.1 Registrar Usuario Funcionario: Crear Registro Usuario funcionario



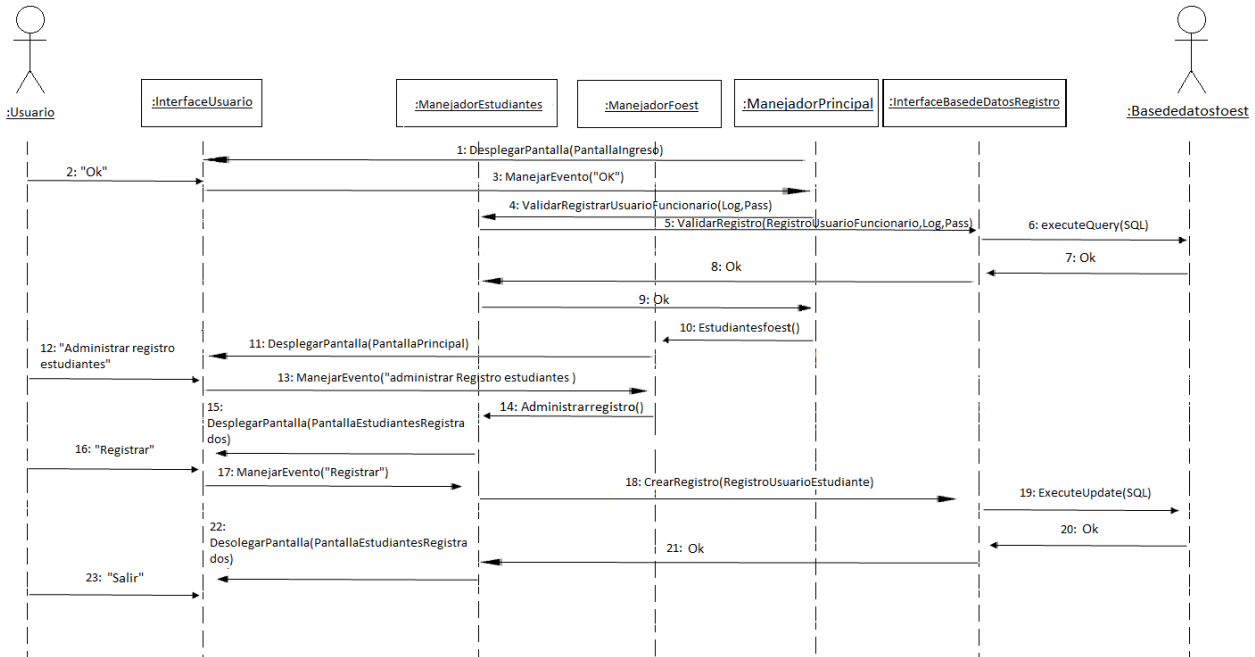
8.3.2 Registrar Usuario Funcionario: Actualizar Registro Usuario funcionario



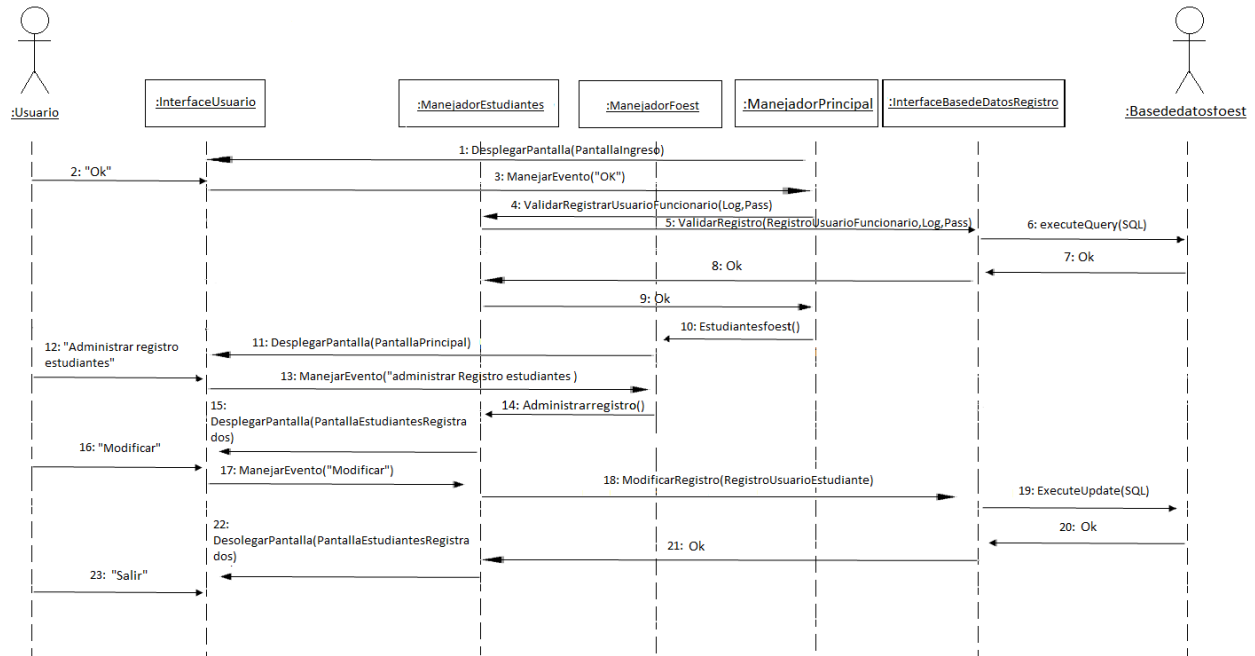
8.3.3 Registrar Usuario Funcionario: Eliminar Registro Usuario funcionario



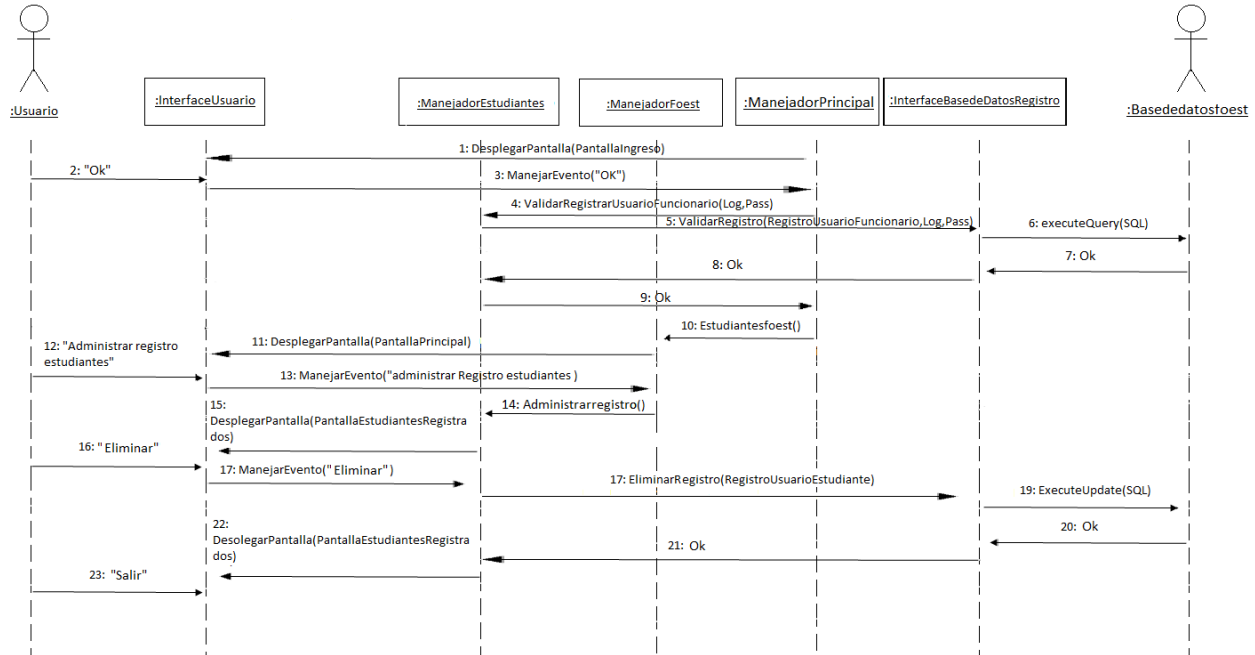
8.3.4 Registrar Usuario Estudiante: Crear Registro Usuario estudiante



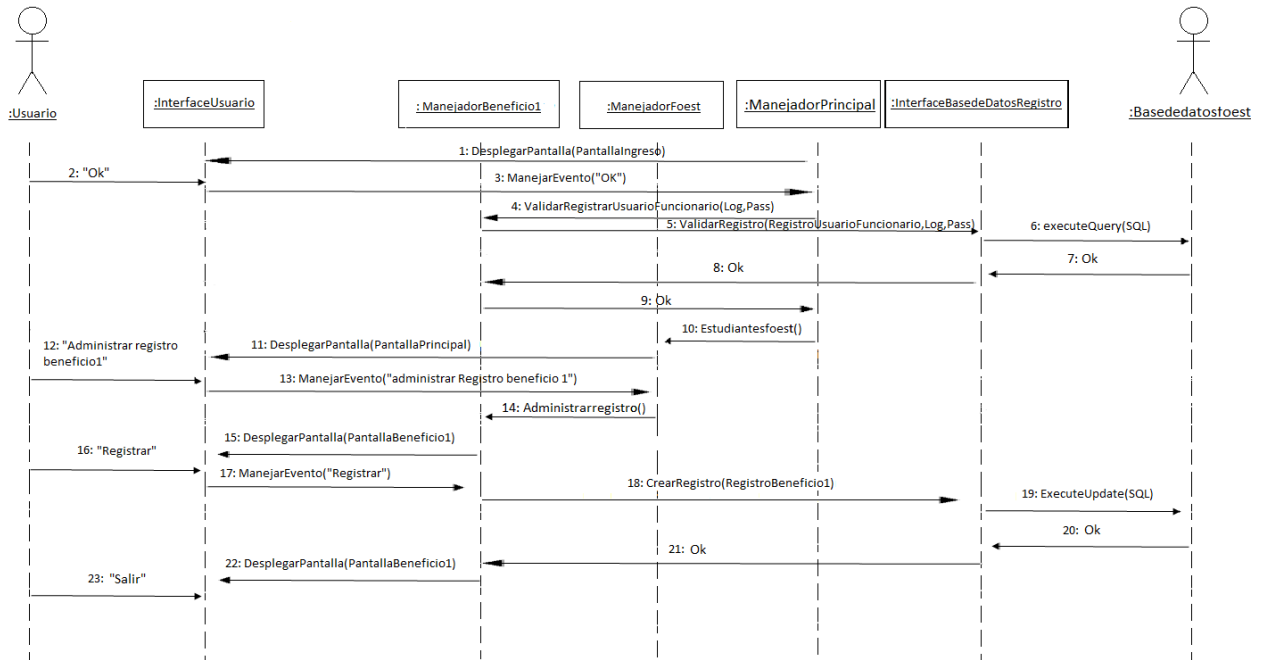
8.3.5 Registrar Usuario Estudiante: Actualizar Registro Usuario estudiante



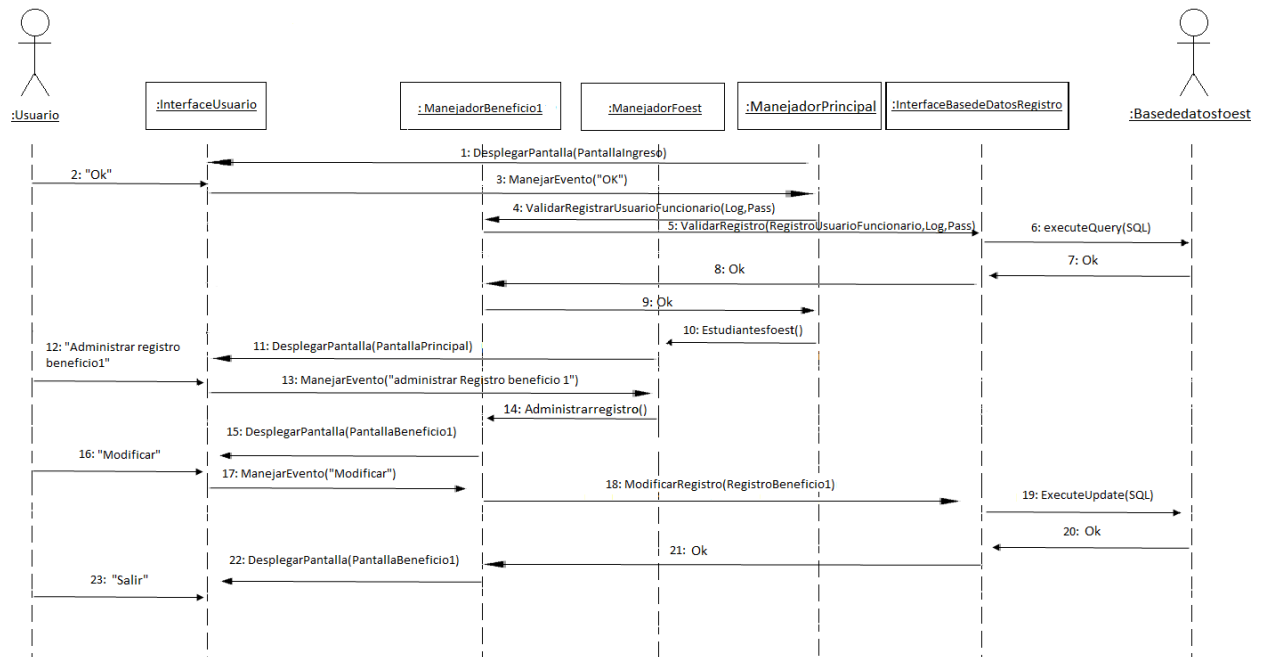
8.3.6 Registrar Usuario Estudiante: Eliminar Registro Usuario estudiante



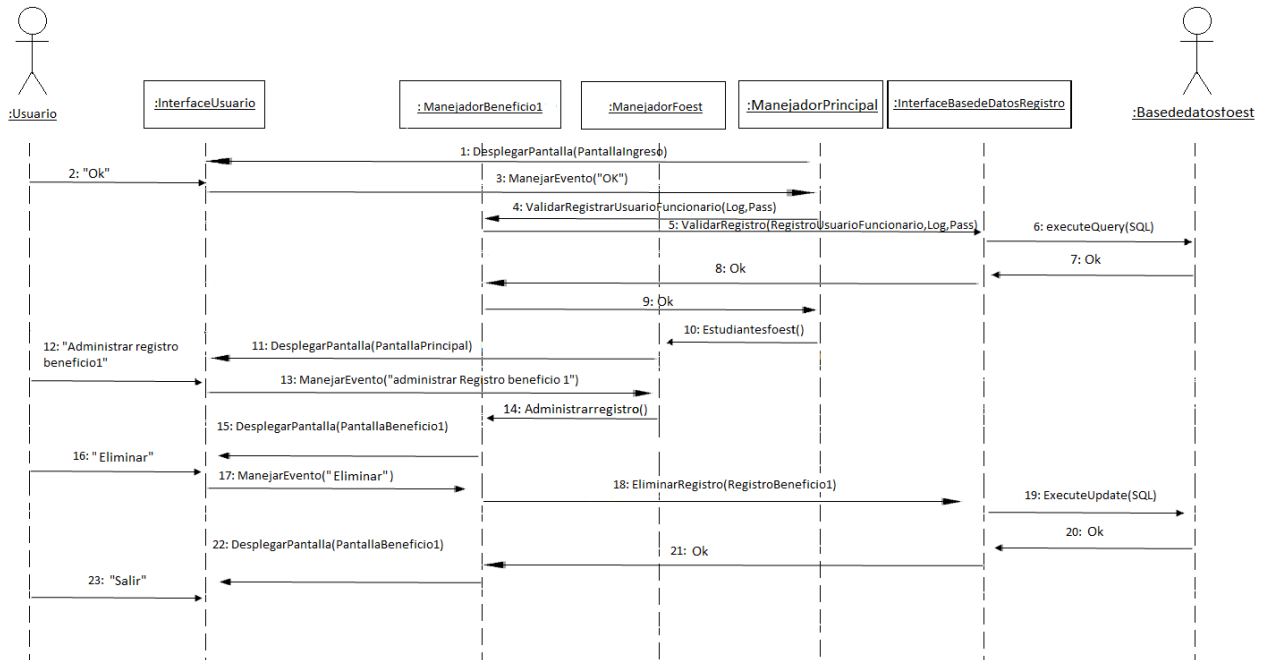
8.3.7 Registrar beneficio 1: Crear Registro beneficio 1



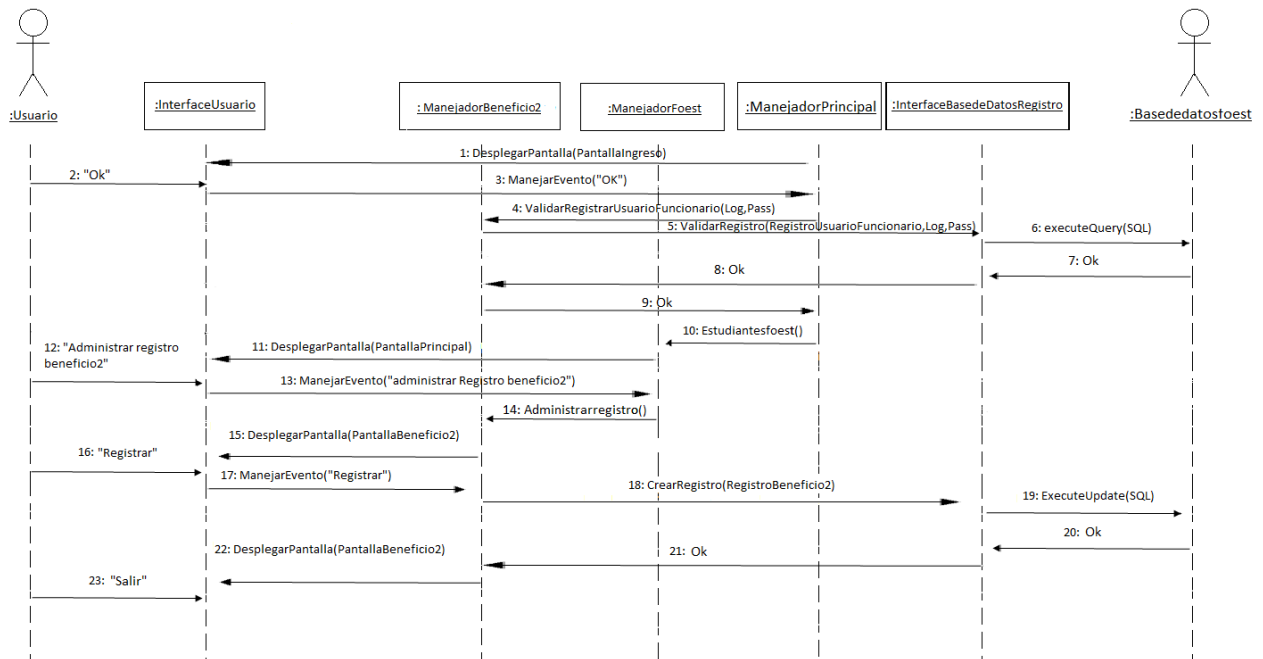
8.3.8 Registrar beneficio 1: Actualizar Registro beneficio 1



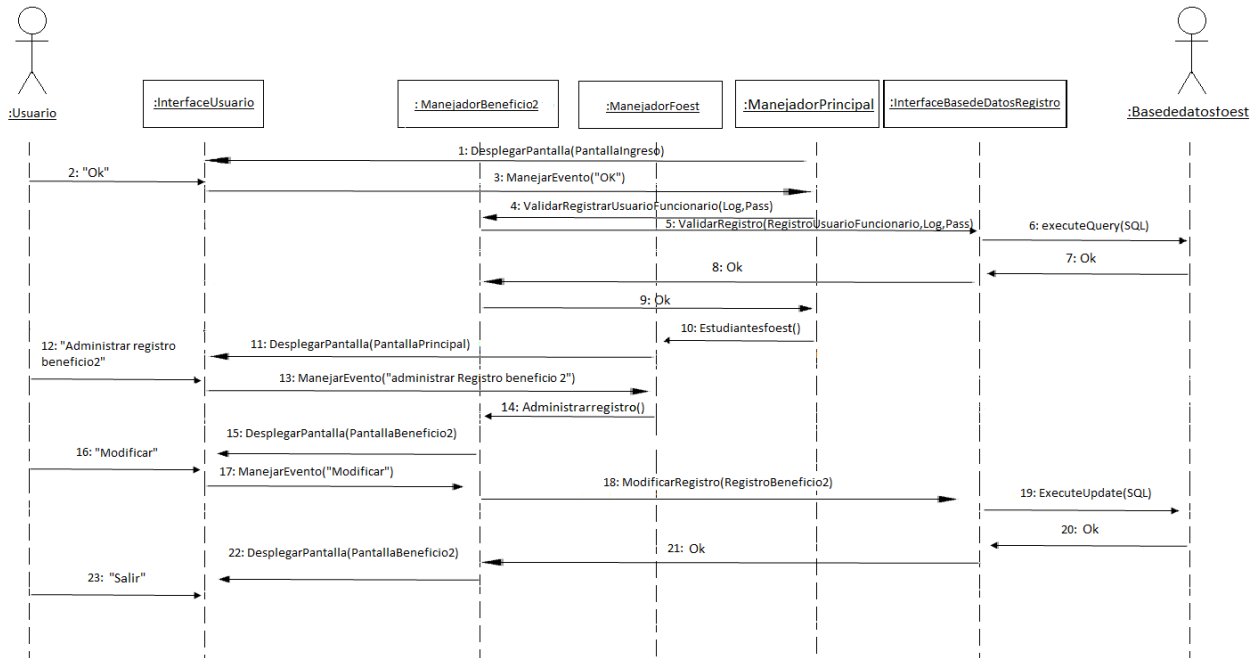
8.3.9 Registrar beneficio 1: Eliminar Registro beneficio 1



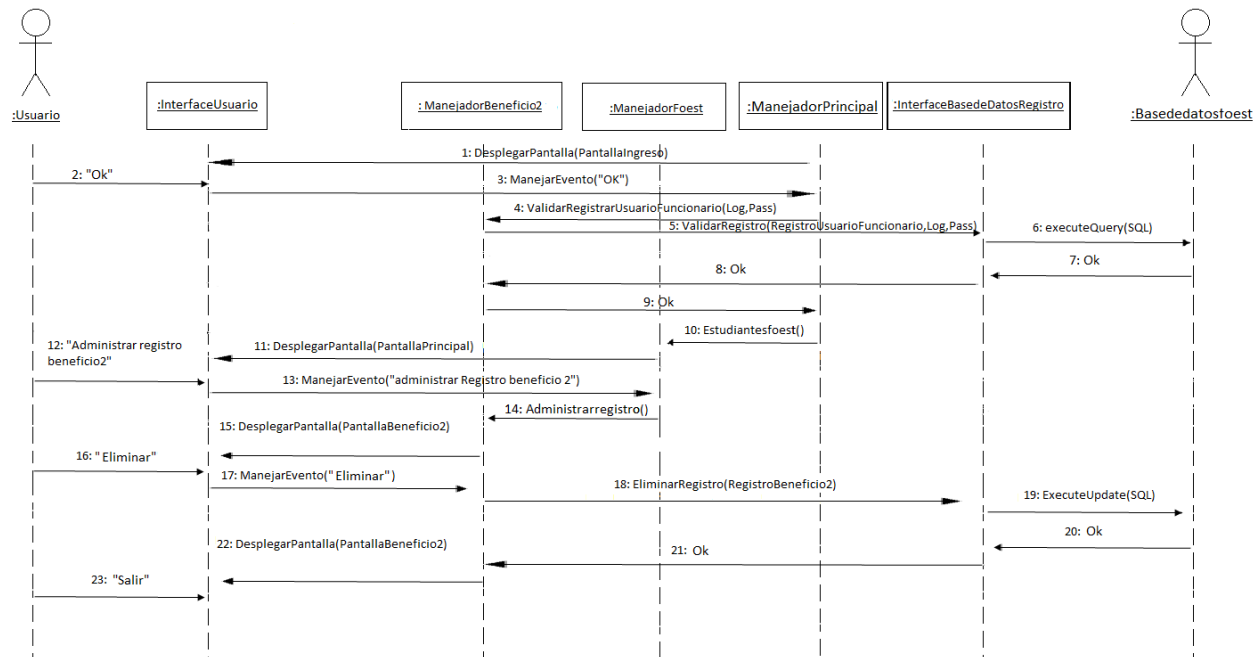
8.3.10 Registrar beneficio 2: Crear Registro beneficio 2



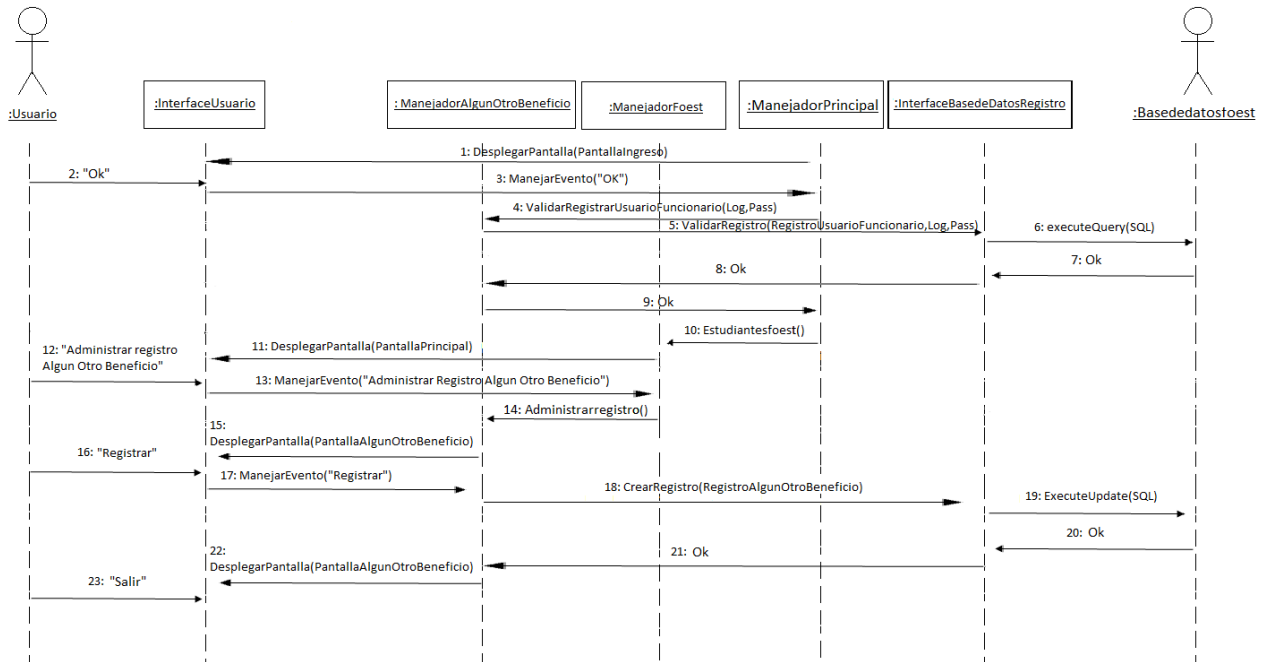
8.3.11 Registrar beneficio 2: Actualizar Registro beneficio 2



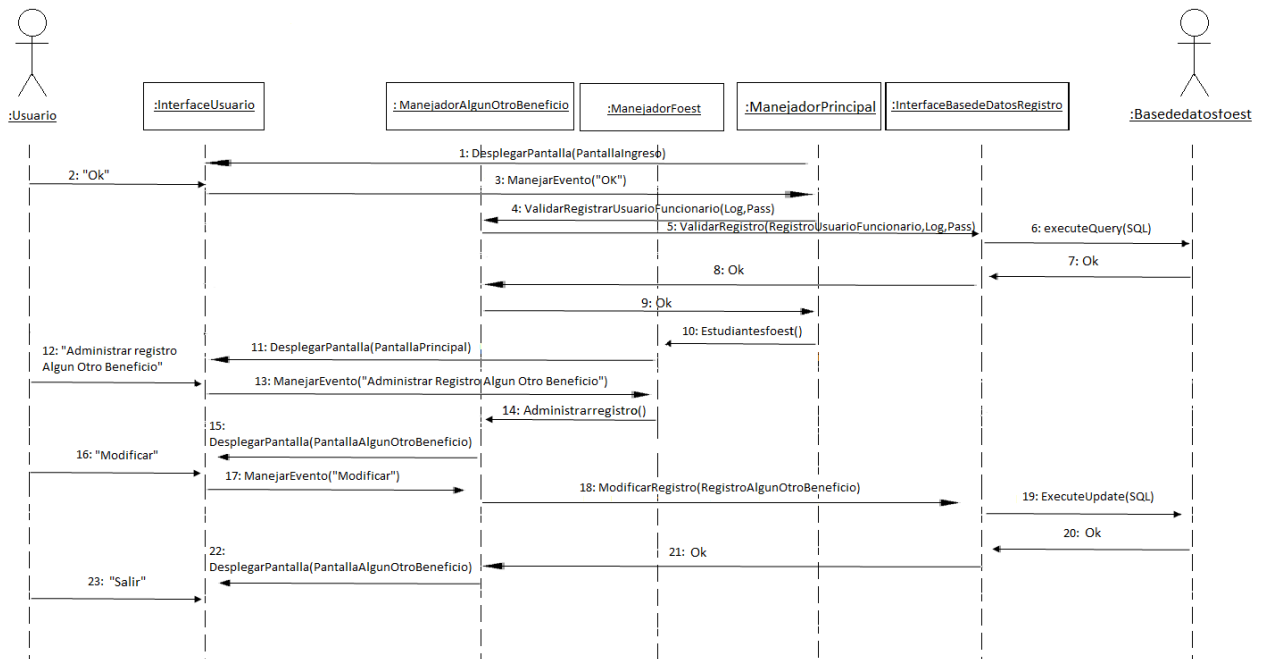
8.3.12 Registrar beneficio 2: Eliminar Registro beneficio 2



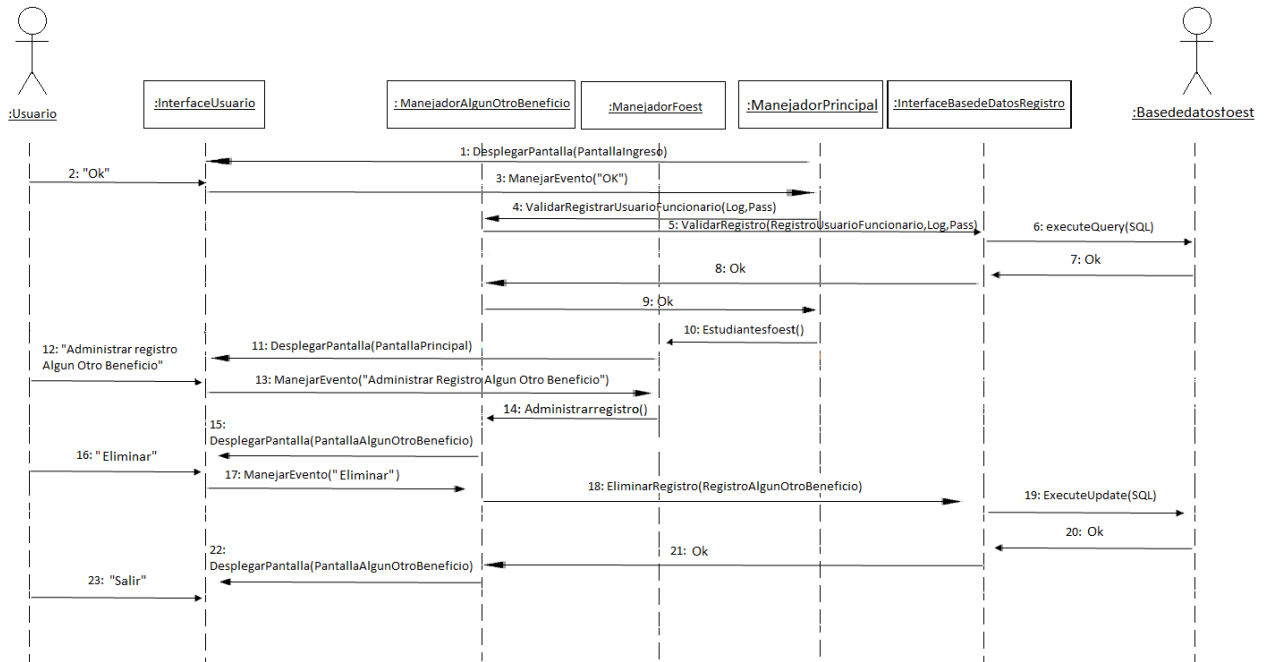
8.3.13 Registrar algún otro beneficio: Crear Registro algún otro beneficio



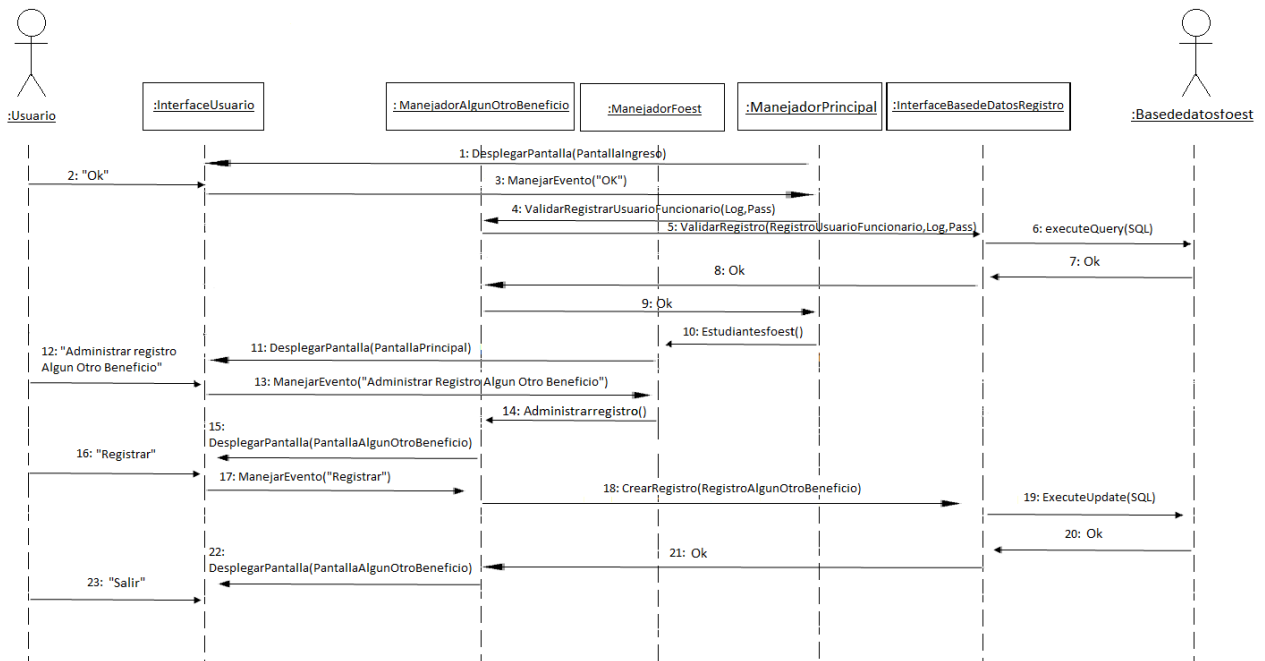
8.3.14 Registrar algún otro beneficio: Actualizar Registro algún otro beneficio



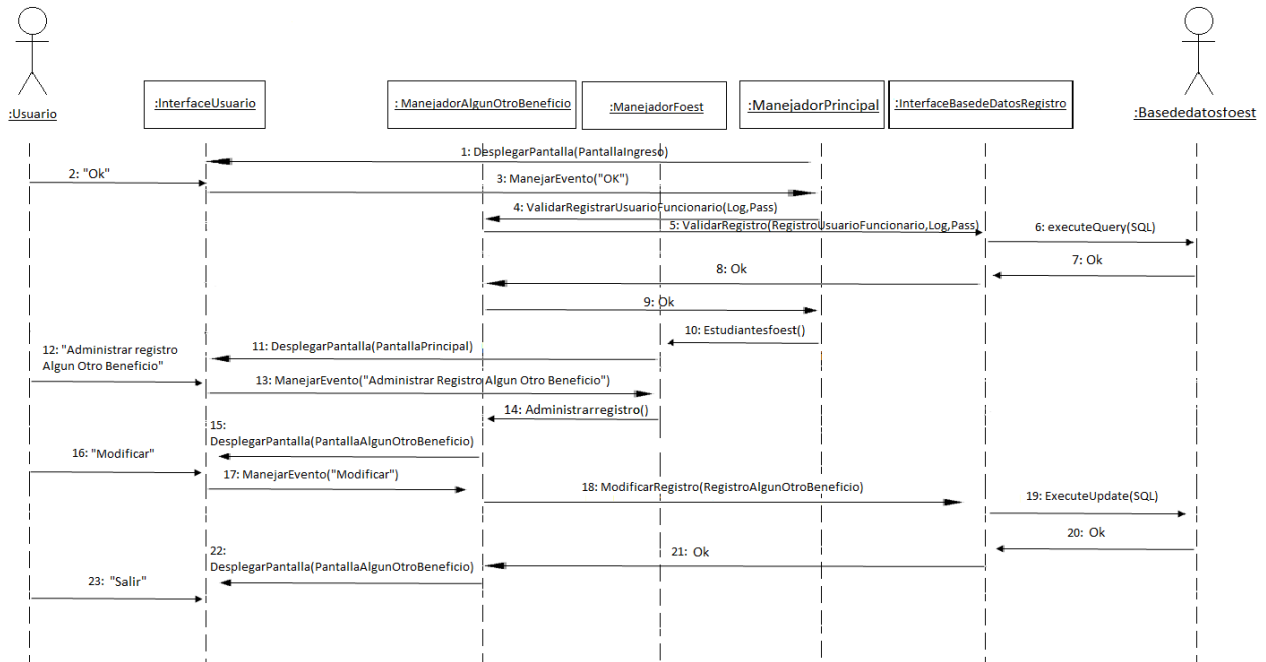
8.3.15 Registrar algún otro beneficio: Eliminar Registro algún otro beneficio



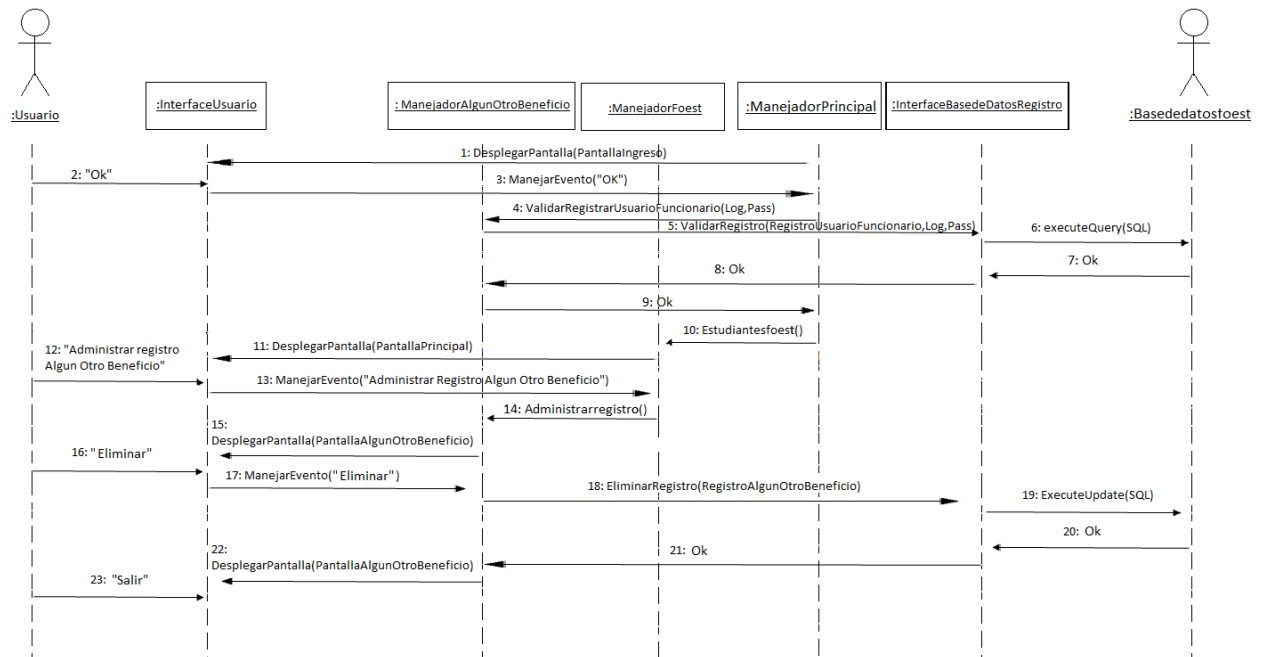
8.3.16 Registrar antigüedad Sisbén: Crear Registro antigüedad Sisbén



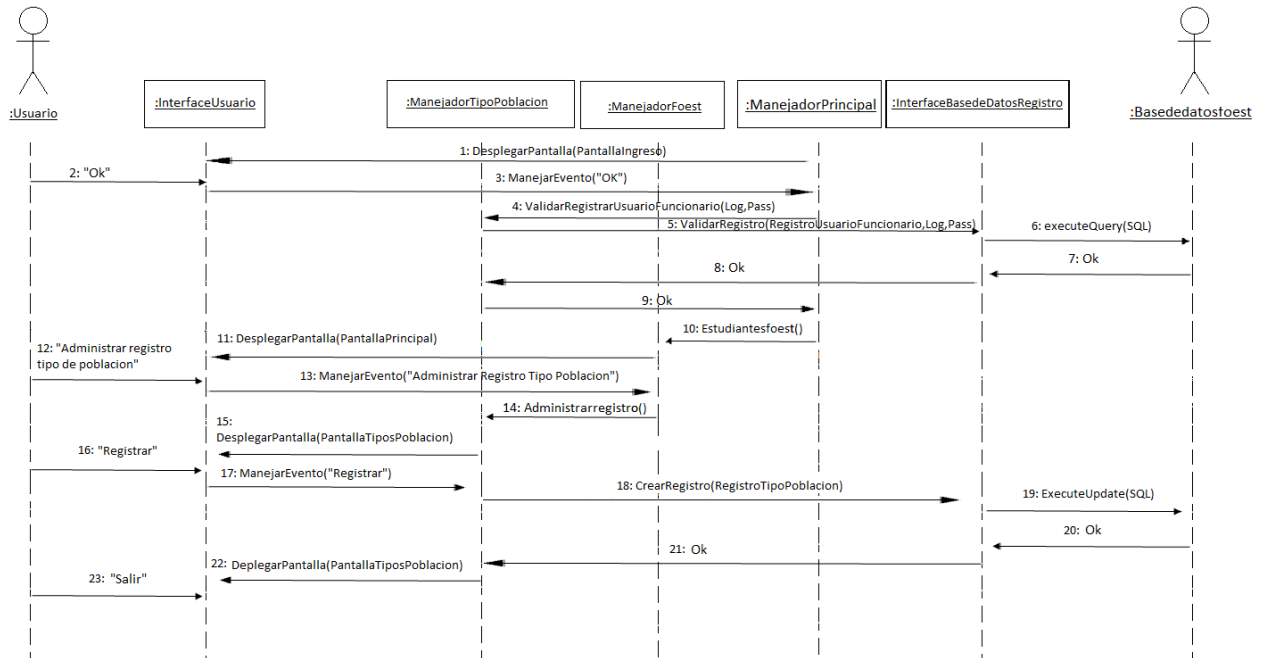
8.3.17 Registrar antigüedad Sisbén: Actualizar Registro antigüedad Sisbén



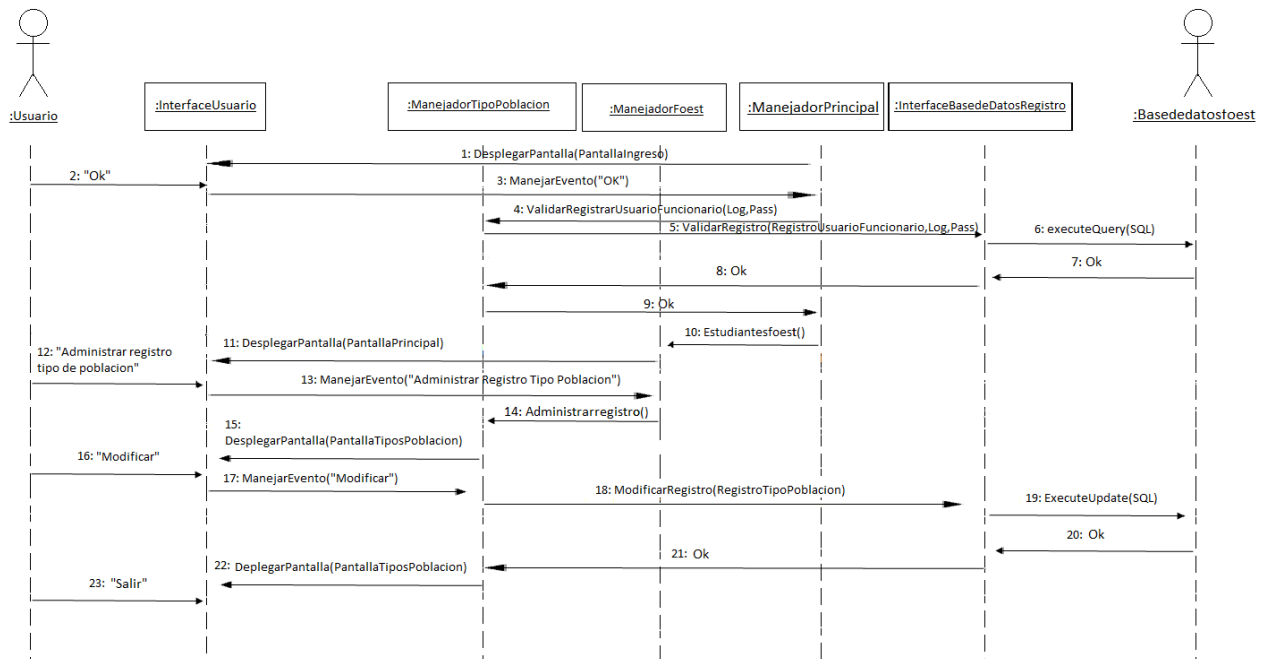
8.3.18 Registrar antigüedad Sisbén: Eliminar Registro antigüedad Sisbén



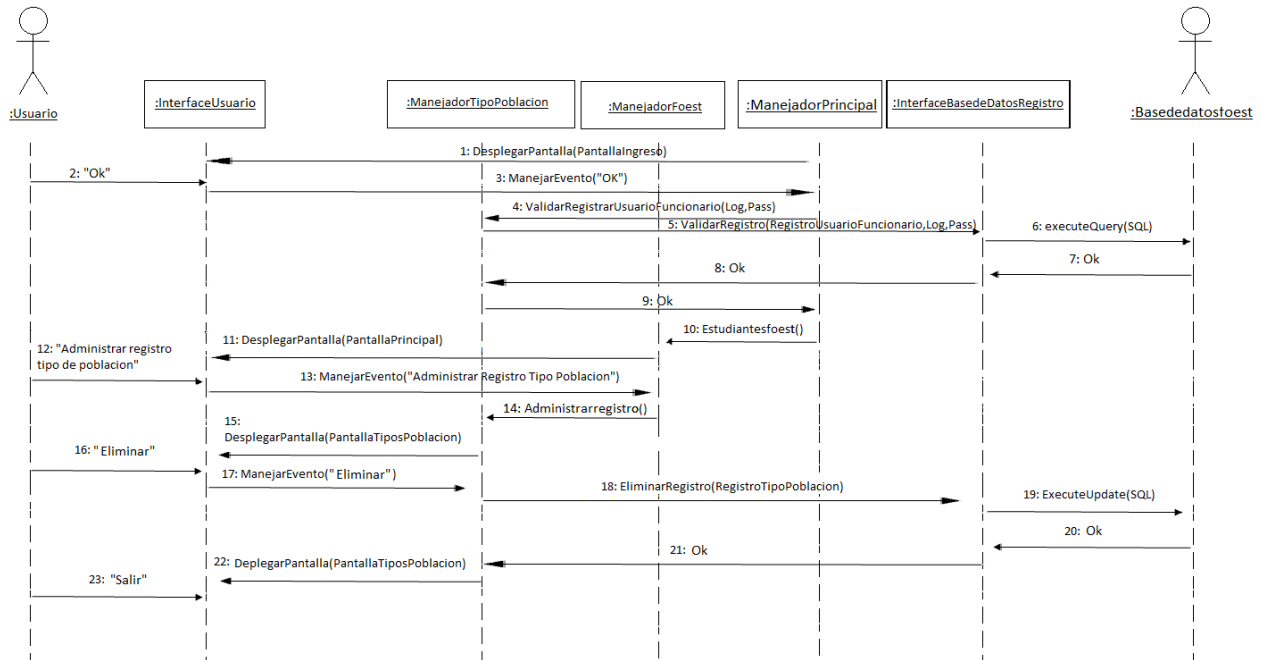
8.3.19 Registrar tipos de poblacion: Crear Registro tipos de poblacion



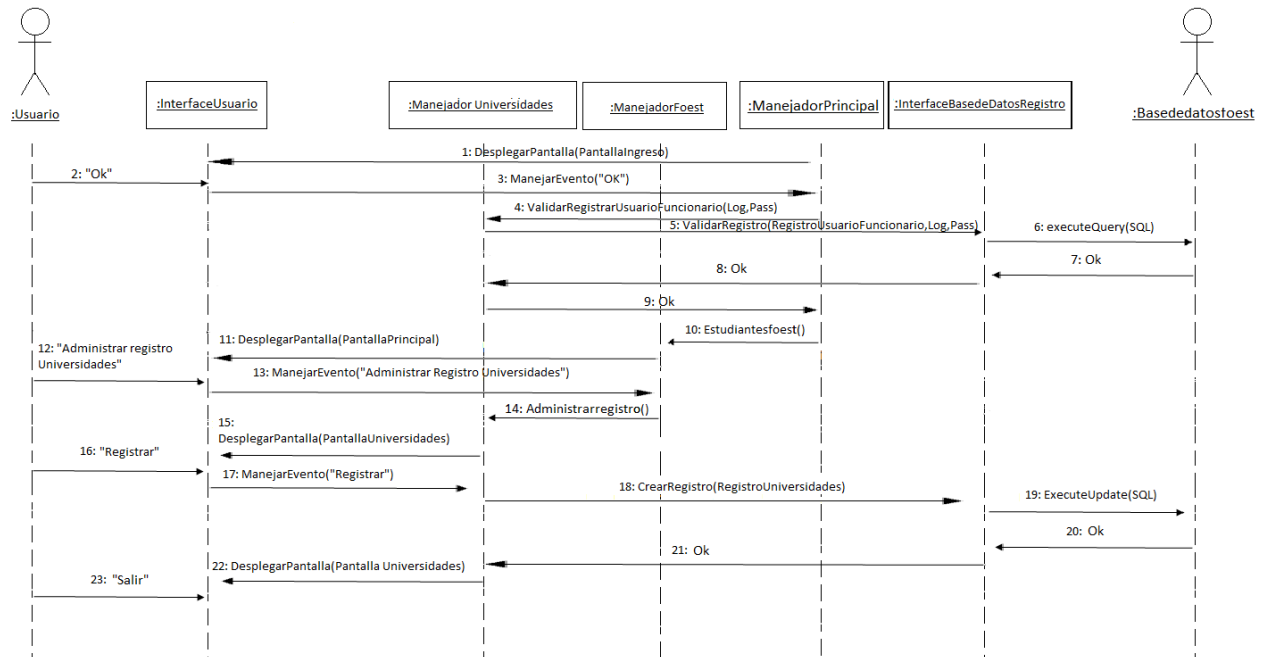
8.3.20 Registrar tipos de poblacion: Actualizar Registro tipos de poblacion



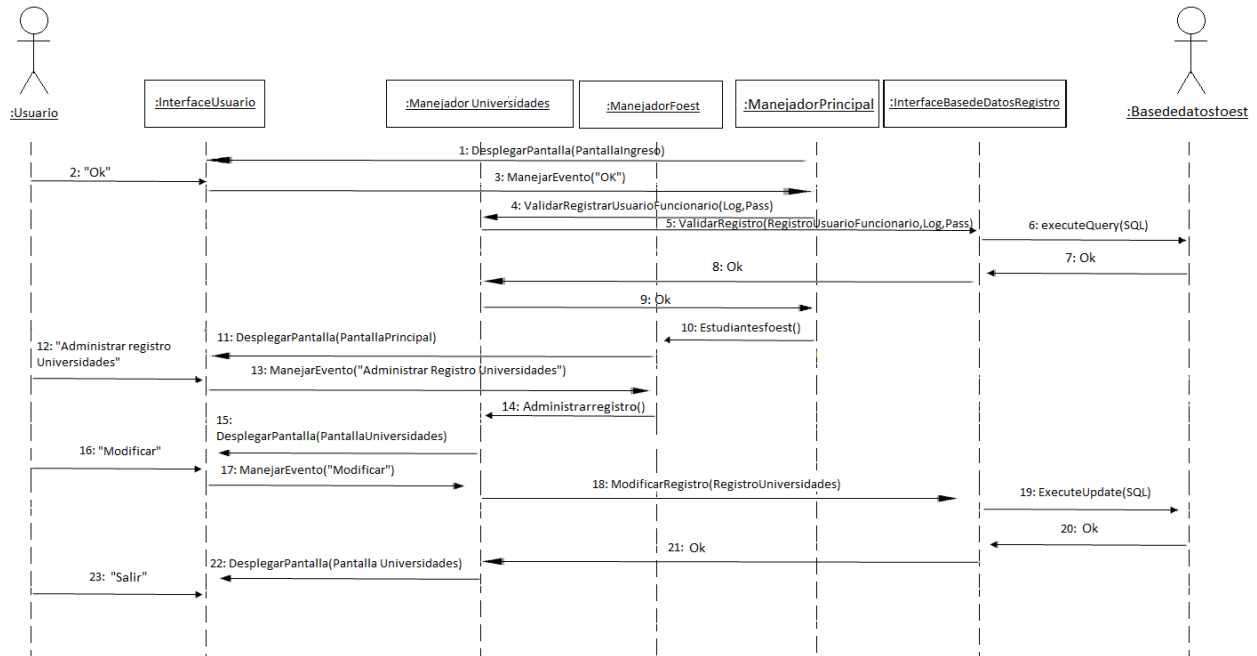
8.3.21 Registrar tipos de poblacion: Eliminar Registro tipos de poblacion



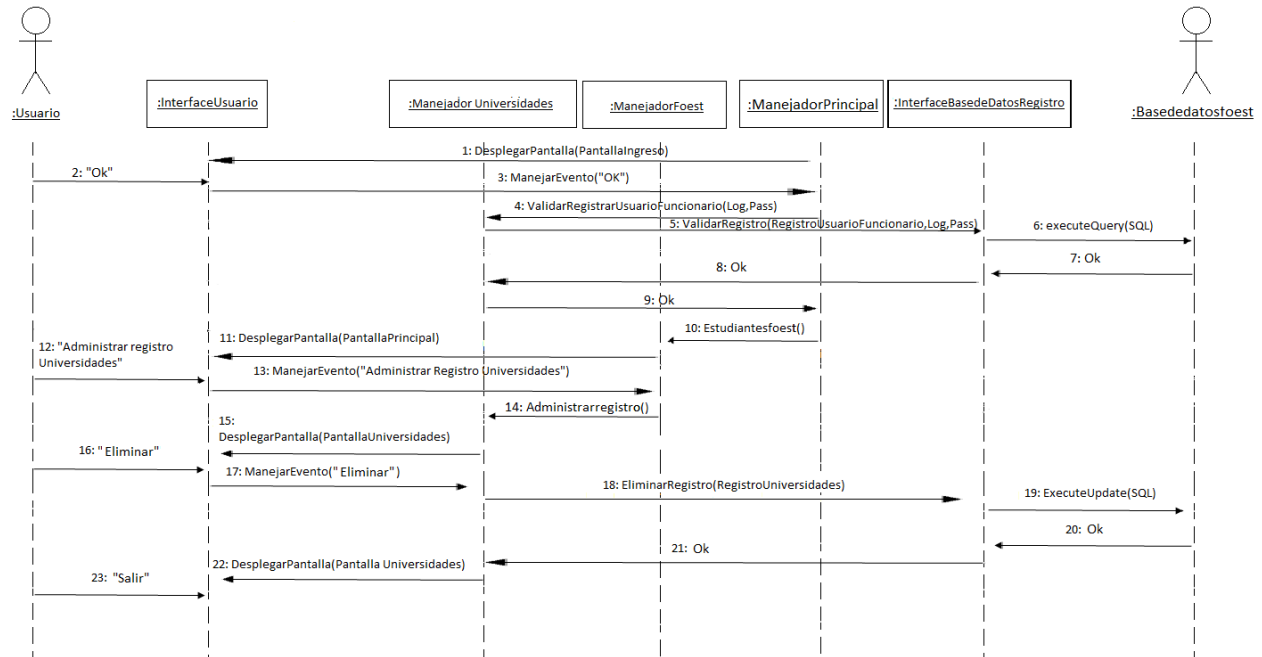
8.3.22 Registrar universidades: Crear Registro universidades



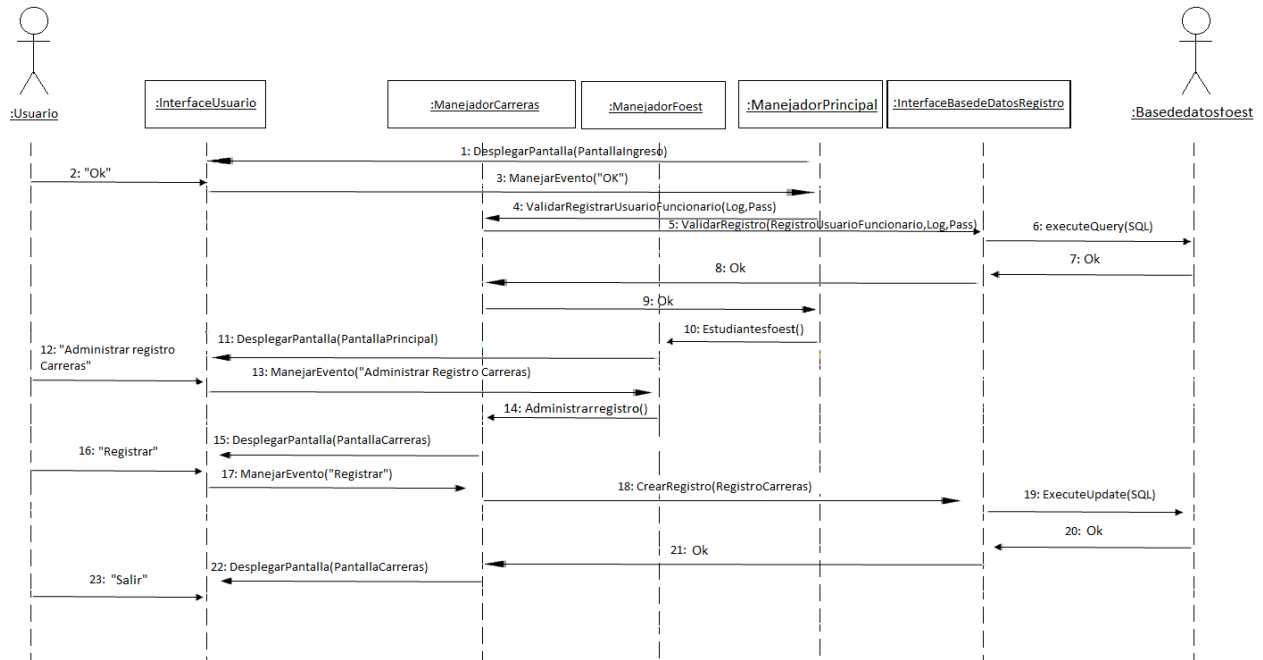
8.3.23 Registrar universidades: Actualizar Registro universidades



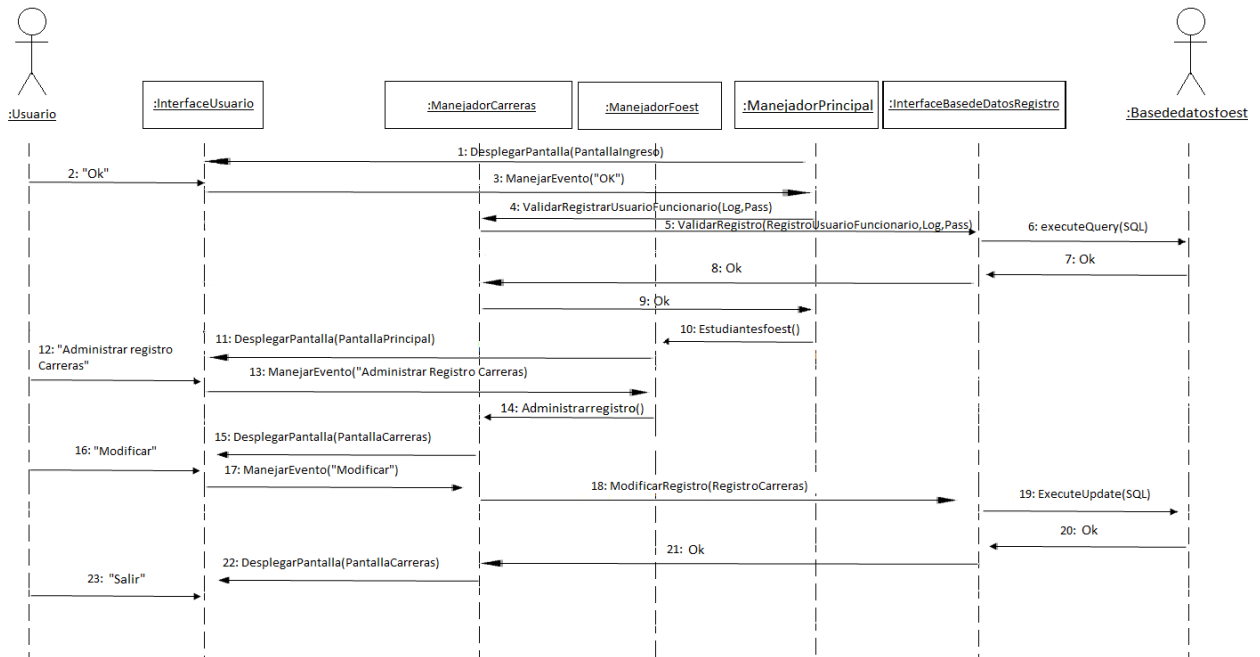
8.3.24 Registrar universidades: Eliminar Registro universidades



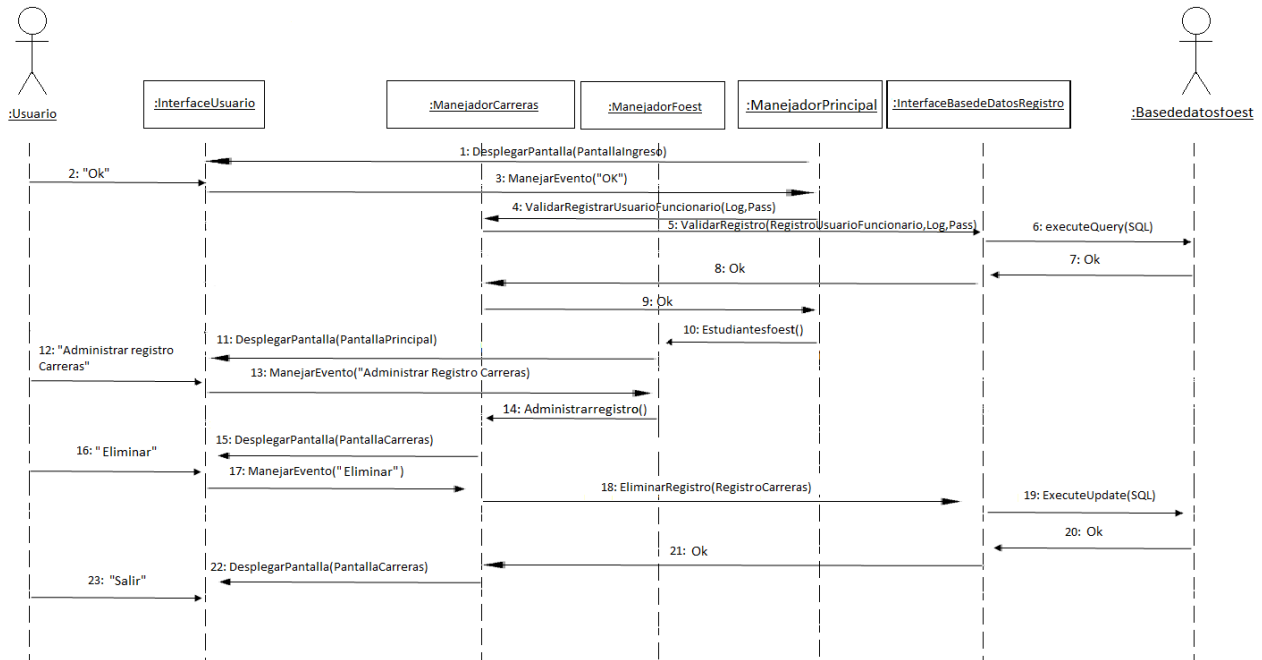
8.3.25 Registrar carreras: Crear Registro carreras



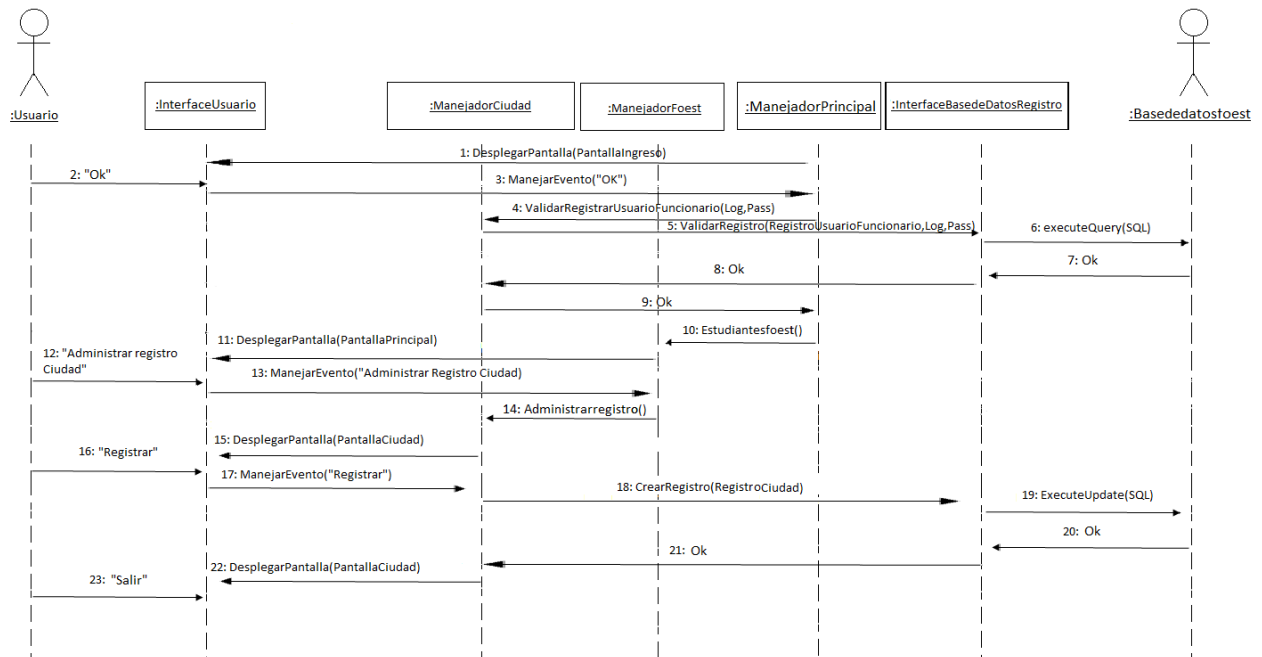
8.3.26 Registrar carreras: Actualizar Registro carreras



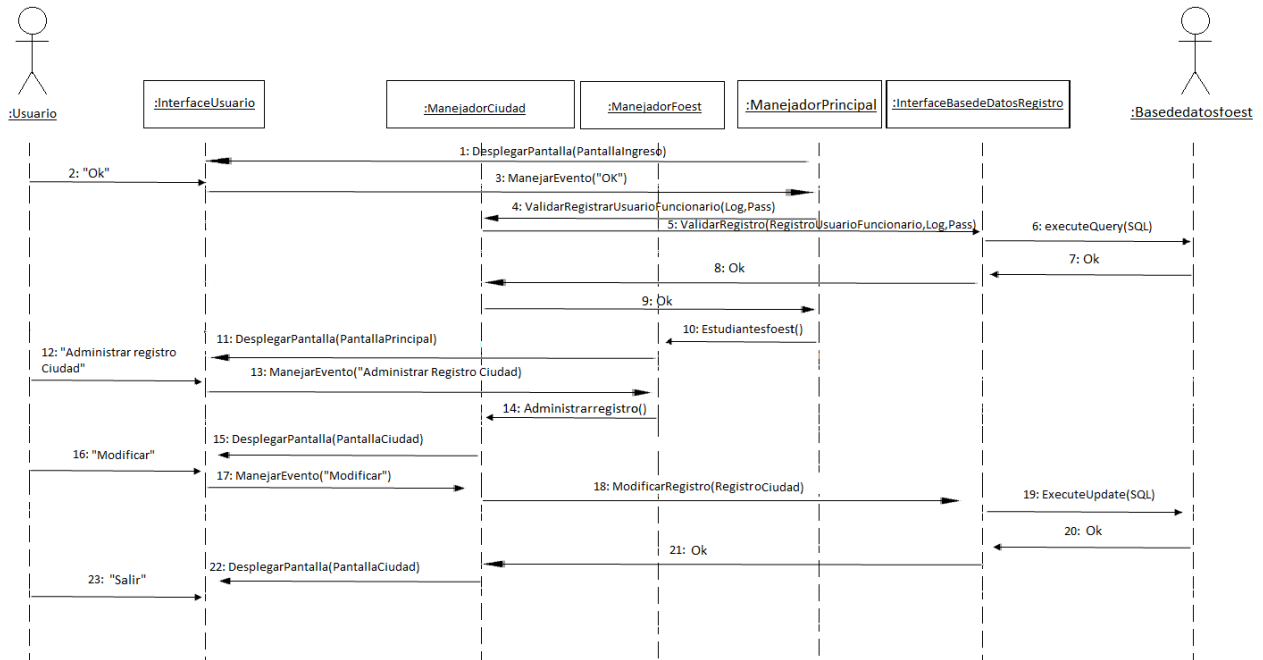
8.3.27 Registrar carreras: Eliminar Registro carreras



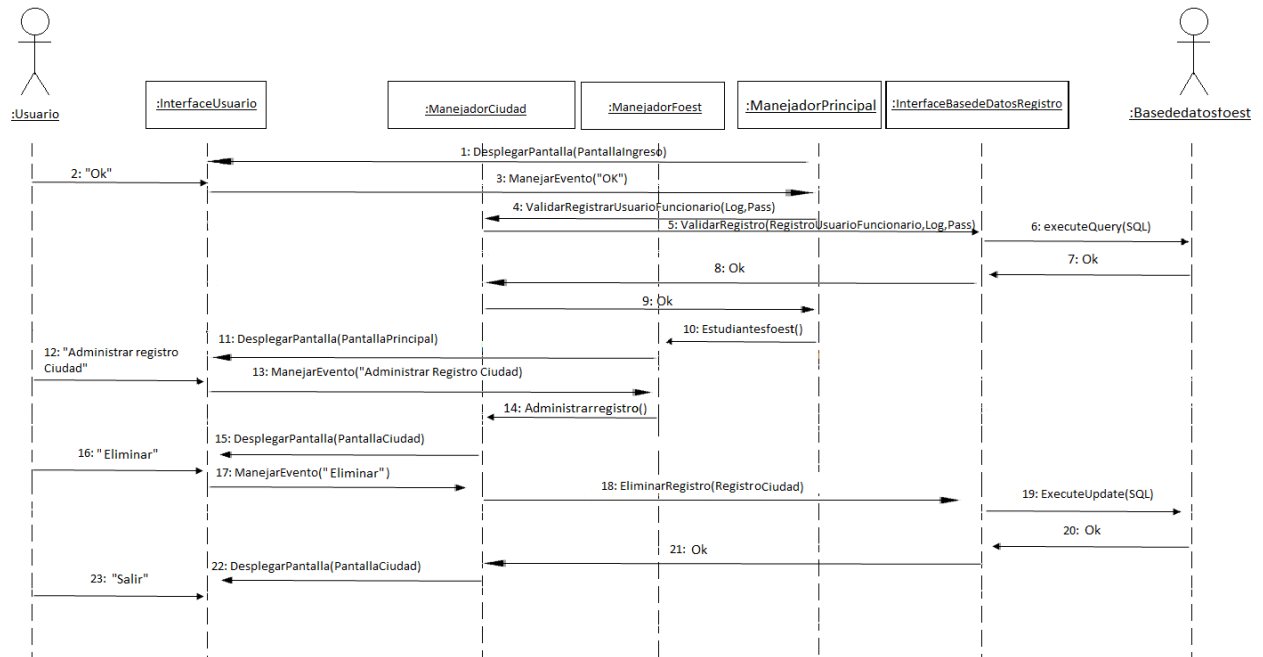
8.3.28 Registrar ciudad: Crear Registro ciudad



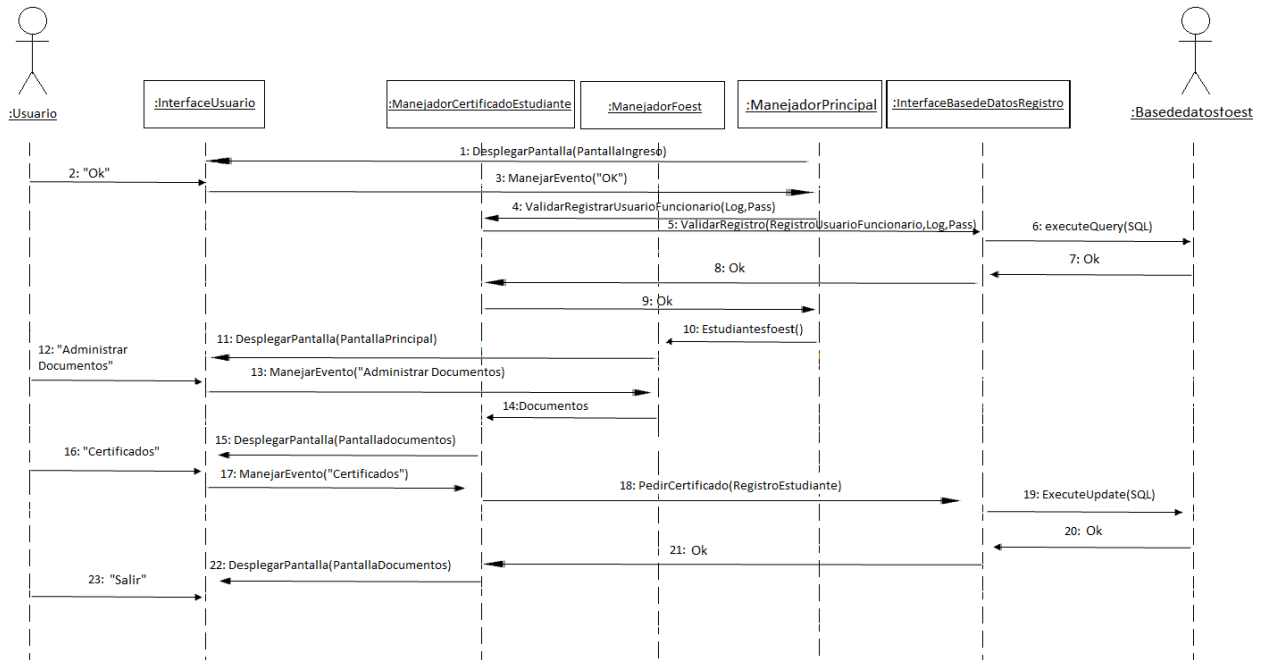
8.3.29 Registrar ciudad: Actualizar Registro ciudad



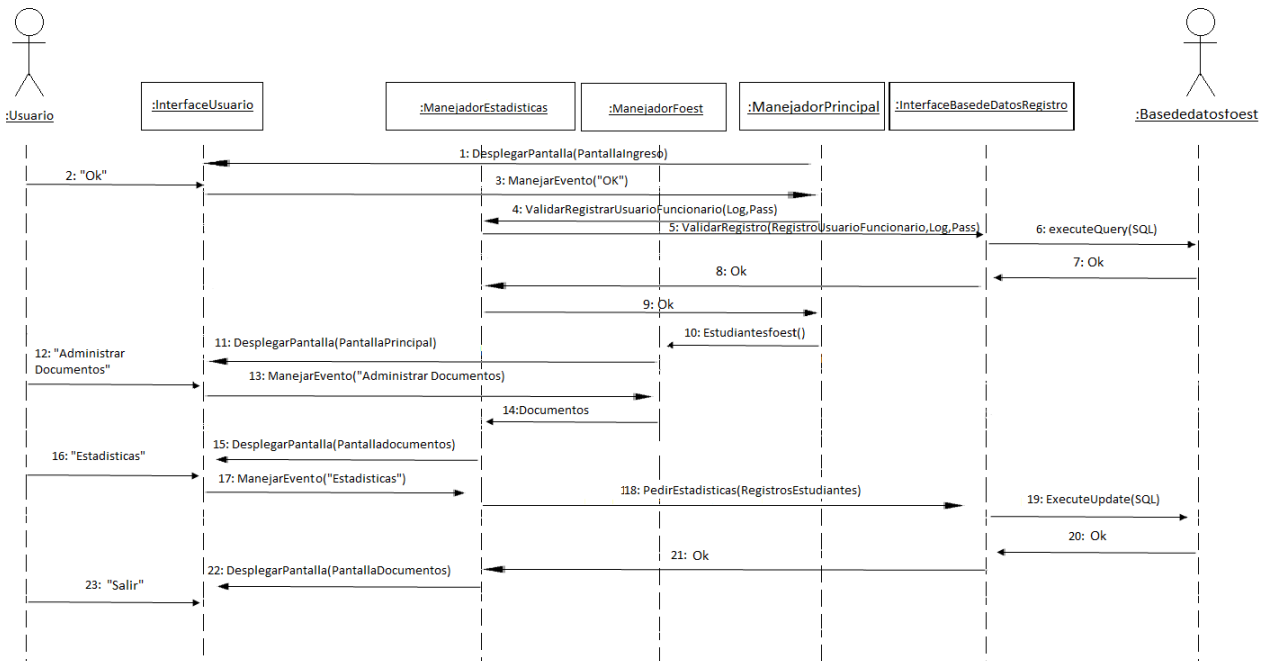
8.3.30 Registrar ciudad: Eliminar Registro ciudad



8.3.31 Certificado: pedir certificado



8.3.32 Estadísticas: Estadísticas



Capítulo 9: Modelo de Implementación Codificación

9.1. Programación en PHP

9.1.1 Login

```
<!doctype html>
<html>
<head>
//Nombre que tendrá en el título, este aparecerá en la pestaña del navegador
<title>Autenticación</title>
// Estilo de la página el cual es diferente a las demás
<style type="text/css">
    body
    {
        background: #efefef;
        font-family: 'Open Sans', sans-serif;
        color: #777;
    }
    .text
    {
        color: #4c56d8;
        font-family: Arial;
        font-size: small;
    }
    form
    {
        background: #ffffff;
        margin: 0 auto;
        width: 335px;
        padding: 1em;
        border: 1px solid #CCC;
        border-radius: 1em;
    }
    form div + div
    {
        margin-top: 1em;
    }
    label
    {
        display: inline-block;
        width: 180px;
        text-align: left;
    }
    input, select, textarea, option
```

```

        {
            font: 1em Arial;
            width: 210px;
            -moz-box-sizing: border-box;
            box-sizing: border-box;
            border: 1px solid #999;
        }
        input:focus, textarea:focus
        {
            border-color: #000;
        }
        .button
        {
            padding-left: 90px;
        }
        button
        {
            margin-left: .5em;
        }
    </style>
</head>
//todo lo almacenado en el body será lo que el usuario vera en pantalla
<body>
    <form>
        <spam class="text">
            <center>
//título de la página login
                <h1>Entrar al sistema</h1>
            </center>
        </spam>
    </form>
    <form action="valau.php" method="POST" style="background: #ffffff;">
        <spam class="text">
            <table border="0">
//nombre y contraseña para el ingreso del sistema
                <tr><td>Nombre de usuario:</td><td><input name="usuario" size="25" value=""/></td></tr>
                <tr><td>Contraseña:</td><td><input name="contrasena" size="25" type="password"/></td></tr>
                <tr><td/><td><input type="submit" value="Inicio de sesión"/></td></tr>
            </table>
        </spam>
    </form>
</body>
</html>

```

9.1.2 conexión

//conexión con mysql

```
<?php
    $con = new mysqli('127.0.0.1','root','foest1') or die( "No se conecto a la base de datos");
    $cons = "";
    $conn = new mysqli('127.0.0.1','root','foest1') or die( "No se conecto a la base de datos");
?>
```

9.1.3 validación usuario

```
<?php
// se llama al archivo de conexión
    include ("conexion.php");
//se llaman las variables de la
    $usuario = $_POST['usuario'];
    $contrasena = $_POST['contrasena'];
//se realiza una conexión con la tabla funcionario para verificar los datos
    $con = mysqli_query($con, "select * from funcionario where funcionario. USUARIO = '$usuario'");
    if($row = mysqli_fetch_array($con))
    {
        if($row['USUARIO'] == $usuario && $row['IDPASS'] == $contrasena)
        {
//conexión establecida
            session_start();
            $_SESSION["usuarios"] = $usuario;
            $_SESSION["valau"] = "SI";
            ?>
            <script type="text/javascript">
                alert("Bienvenido <?php echo $usuario; ?>");
                window.location.href='index.php';
            </script>
            <?php
        }
        else
        {
//conexión rechazada por espacios incompletos o incorrectos
            ?>
            <script type="text/javascript">
                alert("Datos incompletos o incorrectos");
                window.location.href='login.php';
            </script>
            <?php
```

```
        }
    }
    else
    {
        //conexión rechazada por usuario inexistente
        ?>
        <script type="text/javascript">
            alert("Usuario no registrado");
            window.location.href='login.php';
        </script>
        <?php
            exit();
        }
    ?>
```

9.1.4 archivo de bloqueo

//este archivo bloquea las diferentes pantallas para que ningún usuario que no esté validado no pueda acceder a ellas

```
<?php
// verifica la validacion del usuario
    session_start();
    if ($_SESSION["valau"] != "SI")
    {
        ?>
        //Error si no está conectado
        <script type="text/javascript">
            alert("Ingrese");
            window.location.href='login.php';
        </script>
        <?php
            exit();
        }
    ?>
```

9.1.5 hoja de estilos

//estilo de pantalla de las diferentes paginas o pantallas, este estilo abarca las tablas

@charset "utf-8";

//texto en general

.text

{

color: #4c56d8;

font-family: Arial;

font-size: small;

}

// inputs o text box y texarea

input, textarea

{

font: 1em sans-serif;

width: 15%;

-moz-box-sizing: border-box;

box-sizing: border-box;

border: 1px solid #999;

}

input:focus, textarea:focus

{

border-color: #000;

}

//botones

.button

{

padding-left: 4%;

}

button

{

margin-left: .5em;

}

//tablas

table

{

font-family: "Arial";

font-size: 12px;

margin: 15px;

border-collapse: collapse;

overflow-y: auto;

}

//Titulo de la tabla

th

{


```

font-size: 13px;
font-weight: normal;
padding: 8px;
background: #b9c9fe;
border-top: 4px solid #aabcfe;
border-bottom: 1px solid #fff;
color: #039;
}
//celdas de la tabla
td
{
padding: 8px;
background: #e8edff;
border-bottom: 1px solid #fff;
color: #669;
border-top: 1px solid transparent;
}
tr, td
{
overflow-y: auto;
overflow-x: hidden;
}
//cambio de colores entre fila y fila
tr:nth-child(even) td
{
background: #d0dafd;
}
tr:nth-child(odd) td
{
background: #e8edff;
}
tr:hover td
{
background: #d0dafd;
color: #339;
}

```

9.1.6 página principal

// Se incluye el archivo de bloqueo

```
<?php
    include ("blodese.php");
?>
```

```
<!doctype html>
```

```
<html>
```

```
<head>
```

```
    <meta charset="utf-8">
```

//título de la pagina

```
    <title> Base de datos</title>
```

//hoja de estilos

```
        <link rel="stylesheet" type="text/css" href="estilopri.css">
```

```
</head>
```

```
<body>
```

// Información que vera el usuario

```
        <form id="form0" style="width:98%;height:2.6%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid #CCC;"><div><center>
```

```
            <p>Sistema de información</p>
```

// busca el usuario que ingreso al sistema

```
        <div style="float: left;"><a href="opciones.php"><?php echo
$_SESSION["usuarios"]?></a></div>
```

// opción para salir del sistema

```
        <div style = "float: right;"><a href="salir.php">Salir</a></div>
```

```
        <p>&nbsp;</p>
```

```
        </center></div>
```

```
</form>
```

// funciones extra

```
        <form id="form2"method='POST' action='index.php' margin='0' style="float: left; width:18%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
```

```
        <p><a href="fuid.php">FUID</a></p>
```

```
        <p><a href="bene1.php">Beneficio 1 </a></p>
```

```
        <p><a href="bene2.php">Beneficio 2 </a></p>
```

```
        <p><a href="algunotro.php">Algún otro Beneficio</a></p>
```

```
        <p><a href="antsisben.php">Antigüedad Sisbén</a></p>
```

```
        <p><a href="poblacion.php">Tipos de población</a></p>
```

```
        <p><a href="universidad.php">Universidades</a></p>
```

```
        <p><a href="carrera.php">Carreras</a></p>
```

```
        <p><a href="ciudad.php">Ciudad</a></p>
```

```
</form>
```

// Función para crear nuevos periodos

```

<form for="form3" action='index.php' method='POST' margin='0' style="float: left;
width:80%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans
MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
  <center>
    <label>Nuevo periodo:</label>
    <p></p>
    <label>Año:</label>
    <input type="text" name="año" style="width: 100px" placeholder="año" required>
    <label>semestre:</label>
    <input type="text" name="semestre" style="width: 100px" placeholder="semestre"
required>
    <p></p>
    <input type="submit" name="crear" id="crear" value= "Nuevo"
formaction="nuevabase.php">
  </center>
</form>

```

// Función para listar los periodos

```

<form for="form4" action='index.php' method='POST' margin='0' style="float: left;
width:80%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans
MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  <?php
    include ("menuperiodo.php");
  ?>
</form>

</body>
</html>

```

9.1.7 Menú opciones

// es un menú común en las distintas páginas y pantallas donde se verán las distintas opciones auxiliares

```

<?php
  echo "<p><a href=index.php>Inicio</a></p>
<p><a href=fuid.php>FUID</a></p>
<p><a href=bene1.php>Beneficio 1 </a></p>
<p><a href=bene2.php>Beneficio 2 </a></p>
<p><a href=algunotro.php>Algún otro Beneficio</a></p>
<p><a href=antsisben.php>Antigüedad Sisbén</a></p>
<p><a href=poblacion.php>Tipos de población</a></p>
<p><a href=universidad.php>Universidades</a></p>

```

```
<p><a href=carrera.php>Carreras</a></p>
<p><a href=ciudad.php>Ciudad</a></p>";
?>
```

9.1.8 nueva base de datos

// En realidad crea una tabla en la base de datos

```
<?php
```

// verifica la validacion del usuario

```
include ("blodese.php");
```

// Incluye la conexión a la base de datos

```
include ("conexion.php");
```

// se toman los datos ingresados en la página o pantalla principal

```
if (isset($_POST["año"], $_POST["semestre"]) and $_POST["año"] != "" and $_POST["semestre"])
{
```

```
    $ano = $_POST["año"];
```

```
    $semestre = $_POST["semestre"];
```

```
    $estudiantes = "estudiantes";
```

```
    $tabla = $estudiantes.$ano.$semestre;
```

```
    $periodo = $ano."-".$semestre;
```

// se ingresa el comando para la creación de la tabla

```
    $sql = "CREATE TABLE IF NOT EXISTS $tabla (`ID_SUB` varchar(11)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `ITEM` int(11) NOT NULL,
`NOCARPETA` varchar(10) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`FECHARADICACION` varchar(11) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`BENEFICIO1` varchar(25) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`BENEFICIO2` varchar(25) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`1VEZ` text CHARACTER SET utf8 COLLATE utf8_spanish_ci, `RN` text CHARACTER SET utf8 COLLATE
utf8_spanish_ci, `REI` text CHARACTER SET utf8 COLLATE utf8_spanish_ci, `PAPELLIDO` varchar(100)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `SAPELLIDO` text CHARACTER SET
utf8 COLLATE utf8_spanish_ci, `PNOMBRE` text CHARACTER SET utf8 COLLATE utf8_spanish_ci NOT
NULL, `SNOMBRE` text CHARACTER SET utf8 COLLATE utf8_spanish_ci, `ID_US` varchar(5)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `NODOCUMENTO` int(11) DEFAULT
NULL, `EXPEDIDOEN` varchar(30) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`FECHANACIMIENTO` varchar(30) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`EDAD` int(3) DEFAULT NULL, `ESTADOCIVIL` varchar(30) CHARACTER SET utf8 COLLATE
utf8_spanish_ci DEFAULT NULL, `NEMBREEPS` varchar(100) CHARACTER SET utf8 COLLATE
utf8_spanish_ci DEFAULT NULL, `ESTADOEPS` varchar(30) CHARACTER SET utf8 COLLATE
utf8_spanish_ci DEFAULT NULL, `POESPECIAL` varchar(13) CHARACTER SET utf8 COLLATE
utf8_spanish_ci DEFAULT NULL, `GENERO` text CHARACTER SET utf8 COLLATE utf8_spanish_ci,
`TEL1` varchar(10) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `TEL2`
varchar(10) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `DIRECCION`
```

varchar(100) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `SECTOR` varchar(100)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `CORREOE1` varchar(100)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `CORREOE2` varchar(100)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `ANTIGSISBEN` varchar(8)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `PUNTASISBEN` varchar(6)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `TRBAJAACTUAL` varchar(10)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `SECTORLABO` varchar(10)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `NOMEMPRESA` varchar(100)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `CARGOLAB` varchar(100)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `VIVIENDA` varchar(10)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `TIEMPOVIVIENDO` varchar(10)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `ESTRATO` varchar(10)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `PERSONASCARGO` varchar(10)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `NUCLEOFAMILIAR` varchar(10)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `NOMBRECONTACTO` varchar(100)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `DOCUMENTOCONTACTO`
varchar(10) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `TELCONTACTO`
varchar(10) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`OCUPACIONCONTACTO` varchar(100) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT
NULL, `PARENTESCO` varchar(20) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`CORREOECONTACTO` varchar(100) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT
NULL, `NOMBRE REPRESENTANTE` varchar(30) CHARACTER SET utf8 COLLATE utf8_spanish_ci
DEFAULT NULL, `NOCEDULAR` varchar(11) CHARACTER SET utf8 COLLATE utf8_spanish_ci
DEFAULT NULL, `EXPEDIDAR` varchar(30) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT
NULL, `TELREPR` varchar(10) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`DIRECCIONREPR` varchar(30) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`PARENTESCOREP` varchar(100) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`INSTITUCIONEDUMEDIA` varchar(200) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT
NULL, `FECHAGRADO` varchar(100) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`PUNTAJEICFES` varchar(5) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`ANOPRESENTACION` varchar(100) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`PROMEDIOONCE` varchar(5) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`PUESTOONCE` varchar(4) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`ROTBENE` varchar(30) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`UNIVERSIDAD` varchar(200) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`SEDE` varchar(30) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `NIVEL`
varchar(20) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `CARRERA` varchar(200)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `MODALIDAD` varchar(16)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `SEMESTRE` varchar(5)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `HORARIO` varchar(10)
CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `SNIES` varchar(5) CHARACTER
SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `CONTINUACARRERA` varchar(30) CHARACTER
SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `VALORSEMESTRE` varchar(9) CHARACTER SET
utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `PORCENTAJEA` varchar(4) CHARACTER SET utf8
COLLATE utf8_spanish_ci DEFAULT NULL, `MONTOAPRO` varchar(11) CHARACTER SET utf8

```

COLLATE utf8_spanish_ci DEFAULT NULL, `OTROBENE` varchar(100) CHARACTER SET utf8
COLLATE utf8_spanish_ci DEFAULT NULL, `MATERIASPER` int(2) DEFAULT NULL,
`PROMEDIOANTERIOR` varchar(4) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL,
`PROMEDIOTRAANTERIOR` varchar(4) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT
NULL, `BENEFICIOSANTERIORES1` varchar(25) CHARACTER SET utf8 COLLATE utf8_spanish_ci
DEFAULT NULL, `BENEFICIOSANTERIORES2` varchar(25) CHARACTER SET utf8 COLLATE
utf8_spanish_ci DEFAULT NULL, `HSCUMPLIDAS` int(3) DEFAULT NULL, `DEPENDENCIA1`
varchar(100) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `DEPENDENCIA2`
varchar(100) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `DEPENDENCIA3`
varchar(100) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `NOCUENTA`
varchar(100) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `DIASVIAJADOS`
VARCHAR(5) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `CIUDADVIAJA`
varchar(30) CHARACTER SET utf8 COLLATE utf8_spanish_ci DEFAULT NULL, `DIAS` int(1) DEFAULT
NULL, `VALORRECOR` int(6) DEFAULT NULL, `VALORSEMANA` int(7) DEFAULT NULL,
`VALORMESBRUTO` int(7) DEFAULT NULL, `VALORMESPAGAR` int(6) DEFAULT NULL, `MESES`
int(1) DEFAULT NULL, `TOTALSEMESTRE` varchar(9) CHARACTER SET utf8 COLLATE utf8_spanish_ci
DEFAULT NULL, `RATIFICADO` varchar(1000) CHARACTER SET utf8 COLLATE utf8_spanish_ci
DEFAULT NULL, `APROBADO` varchar(100) CHARACTER SET utf8 COLLATE utf8_spanish_ci
DEFAULT NULL, `OBSERVACIONRAT` varchar(1000) CHARACTER SET utf8 COLLATE utf8_spanish_ci
DEFAULT NULL);

```

// si se crea la tabla el sistema crea nuevas pantallas para el nuevo periodo

```

if ($con->query($sql) === TRUE)
{
    include ("linknuevo.php");
    include ("tablanueva.php");
    include ("pantregistro.php");
    include ("documentomodificar.php");
    include ("modificarbeneфициario.php");
    include ("panestadistica.php");
    include ("manejadorregistro.php");
    include ("modificarregistro.php");
    include ("nuevobeneфициario.php");
    ?>
    <script type="text/javascript">
        alert("Ha sido creado exitosamente");
        window.location.href='index.php';
    </script>
    <?php
}
Else

```

// error en caso de que no se cree la tabla

```

{
    ?>

```

```

        }
    }
    else
    // error en caso de que quede un espacio libre en la pantalla principal en el registro de un nuevo
    periodo
    {
        ?>
        <script type="text/javascript">
            alert("Hubo un error <?php echo "$nuevo"?>");
            window.location.href='index.php';
        </script>
        <?php
    }
    ?>
    <script type="text/javascript">
        alert("Por favor, complete el formulario");
        window.location.href='index.php';
    </script>
    <?php
    }
    ?>

```

9.1.9 enlazador para nuevos periodos

// este se encarga de modificar el menú periodos, ingresando el nuevo periodo

```

<?php
    $archivo = "menuperiodo.php";
    if(file_exists ($archivo))
    {
        $mensaje = "<?php echo ' <p><a href=$periodo.php>Estudiantes $periodo</a></p>'?>";
    }
    else
    {
    }
    if ($archivo = fopen($archivo, "a"))
    {
        if (fwrite($archivo, "$mensaje"))
        {
        }
    }
    Else
    // error en caso de que no encuentre o pueda modificar el menú periodos
    {

```

```

?>
<script type="text/javascript">
    alert("Ha habido un problema menuperiodo.php");
    window.location.href='index.php';
</script>
<?php
}
fclose($archivo);
}
?>

```

9.1.10 nuevo periodo

// este se encarga de crear la pantalla del nuevo periodo tomando las variables ingresadas en la página principal en la creación de nuevo periodo

```

<<?php
    $base = "$periodo.php";
    if(file_exists ($base))
    {

    }
    else
    {
// se incluyen las conexiones a la base de datos y el archivo de bloqueo
        $mensaje = "<?php
            include (\\"blodese.php\\");
            include (\\"conexion.php\\");
            \\$sql = \\"select * from $tabla\\";
        ?>
        <!doctype html>
        <html>
            <head>
// se insertan los estilos
                <meta http-equiv=\\"Content-type\\" content=\\"text/html; charset=utf-8\\">
                <meta name=\\"viewport\\" content=\\"width=device-width,initial-scale=1, user-scalable=no\\">
                <title> Base de datos $periodo</title>
                <link rel=\\"stylesheet\\" type=\\"text/css\\" href=\\"estilopri.css\\">
                <link rel=\\"stylesheet\\" type=\\"text/css\\"
                    href=\\"https://cdn.datatables.net/1.10.19/css/jquery.dataTables.min.css\\">
                <link rel=\\"stylesheet\\" type=\\"text/css\\"
                    href=\\"https://cdn.datatables.net/buttons/1.5.6/css/buttons.dataTables.min.css\\">
                <style type=\\"text/css\\">
                    input

```



```

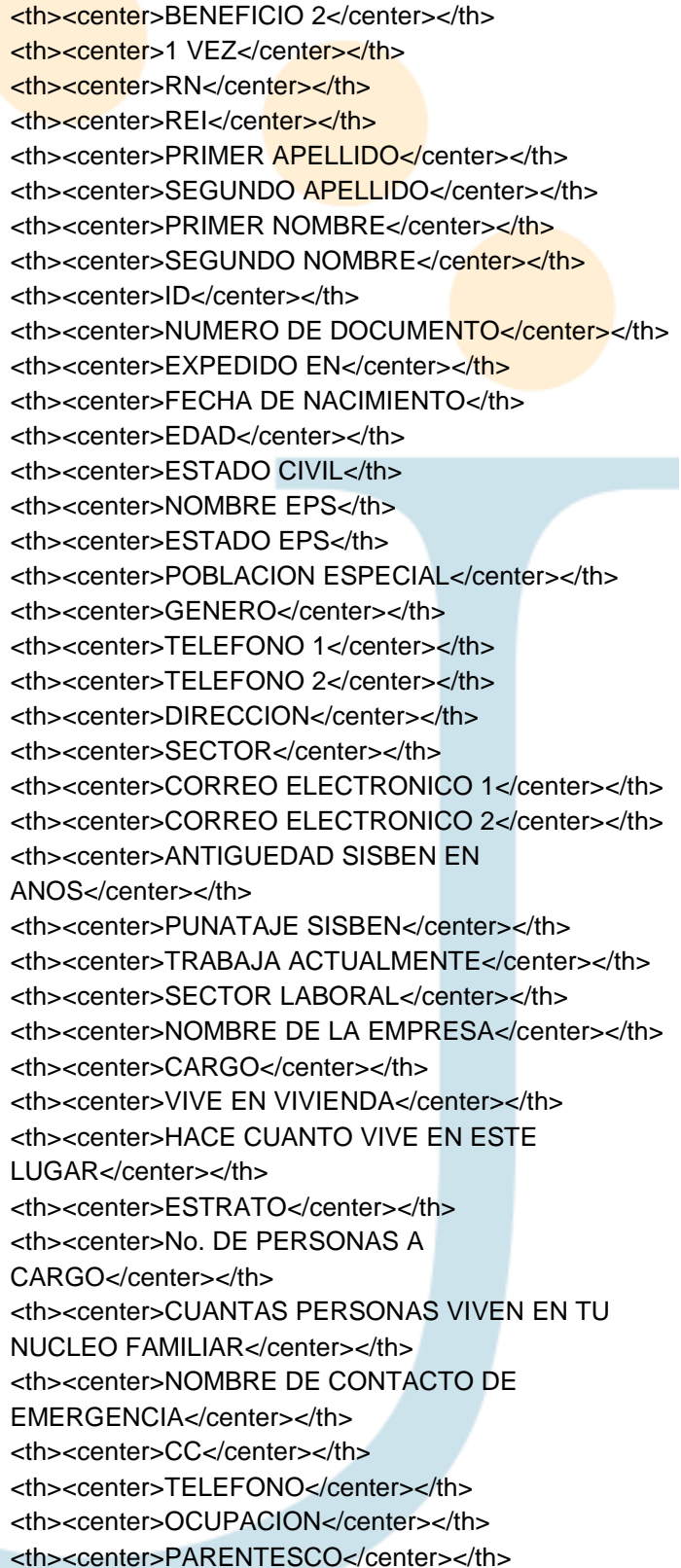
    {
      width: 50%;
    }
    table
    {
      height: 372px;
      width: 96%;
      text-align: left;
      display: block;
      overflow-x: auto;
    }
    th
    {
      position: static;
    }
  }
</style>
<script type="text/javascript" language="javascript" src="https://code.jquery.com/jquery-3.3.1.js"></script>
<script type="text/javascript" language="javascript" src="https://cdn.datatables.net/1.10.19/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" language="javascript" src="https://cdn.datatables.net/buttons/1.5.6/js/dataTables.buttons.min.js"></script>
<script type="text/javascript" language="javascript" src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
<script type="text/javascript" language="javascript" src="https://cdn.datatables.net/buttons/1.5.6/js/buttons.html5.min.js"></script>
<script type="text/javascript" class="init">
$(document).ready(function() {
  $('#example').DataTable( {
    dom: 'Bfrtip',
    buttons: [
      'excelHtml5'
    ]
  });
});
</script>
</head>
// esta parte contiene las variables necesarias, para variables específicas como nombre de la tabla, se toman las variables de la página principal segmento nuevo periodo, también incluye la vista del usuario
<body>
  <form id="form1" style="width:98.72%;height:auto;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid #CCC;"><div><center>

```

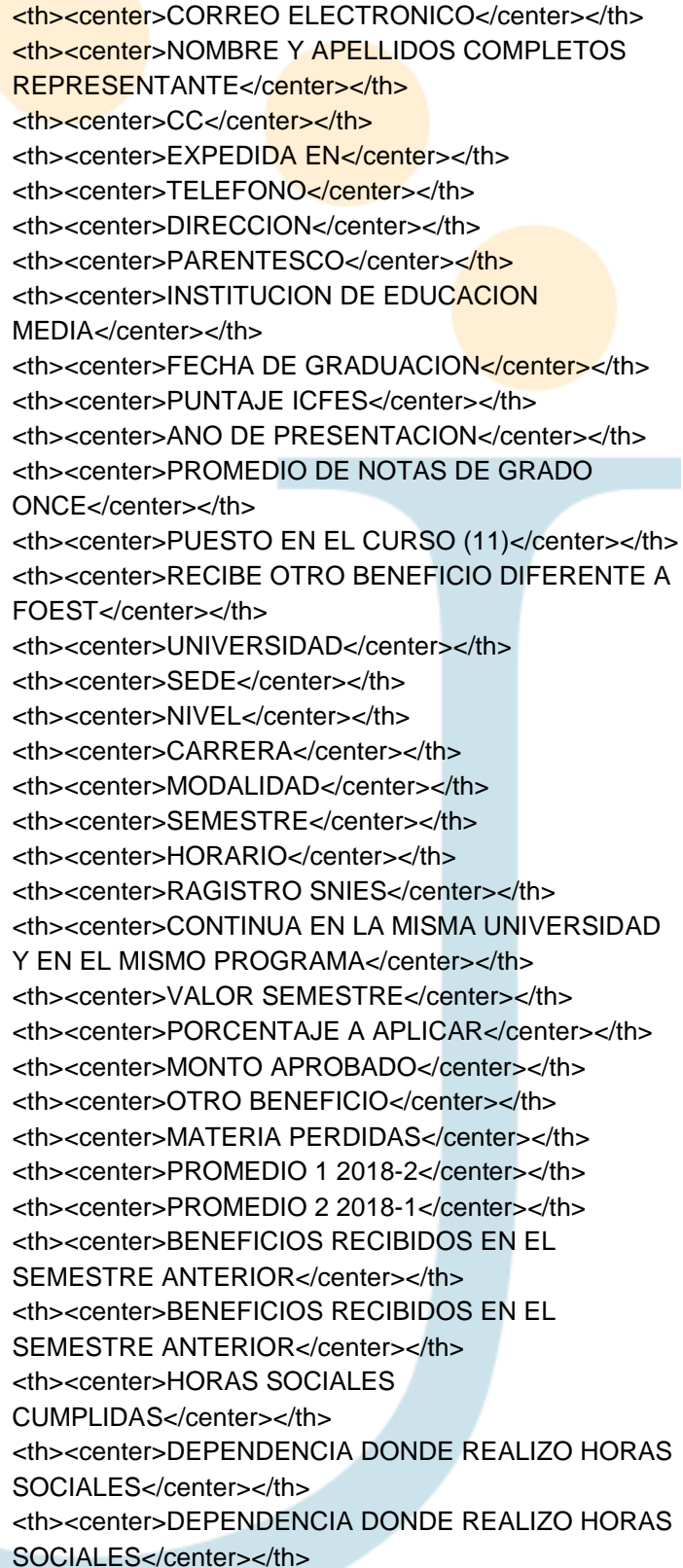
```

<p>Estudiantes $periodo</p>
  <div style="float: left;"><?php echo \$_SESSION["usuarios"]?></div>
  <div style = "float: right;"><a href="salir.php">Salir</a></div>
<p>&nbsp;</p>
</center></div>
</form>
<form id="form2" method="POST" action="index.php" margin="0" style="float:
left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans
MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  <?php
    include ("menuopciones.php");
  ?>
</form>
<form id="form3" method="POST" action="index.php" margin="0" style="float:
left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  <center>
    <p></p>
    <input name="Registrar" type="submit" id="enviar2"
formaction="nuevo$periodo.php" value="Nuevo registro" style="width:
150px">
    <input name="Modificar" type="submit" id="enviar3"
formaction="modificacion$periodo.php" value="Actualizar registro"
style="width: 180px">
    <input name="esta" type="submit" id="enviar4"
formaction="esta$periodo.php" value="Estadísticas" style="width: 230px">
    <p></p>
  </center>
</form>
<form id="form4" method="GET" action="$periodo.php" margin="0" style="float:
left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans
MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
</form>
<form id="form5" action="$periodo.php" method="POST" margin="0" style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans
MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
  <div class="fw-container">
    <table id="example" class="display" style="width:100%">
      <thead>
        <th><center>ID</center></th>
        <th><center>ITEM</center></th>
        <th><center>No. CARPETA</center></th>
        <th><center>FECHA DE RADICACION</center></th>
        <th><center>BENEFICIO 1</center></th>
      </thead>
    </table>
  </div>

```



<th><center>BENEFICIO 2</center></th>
<th><center>1 VEZ</center></th>
<th><center>RN</center></th>
<th><center>REI</center></th>
<th><center>PRIMER APELLIDO</center></th>
<th><center>SEGUNDO APELLIDO</center></th>
<th><center>PRIMER NOMBRE</center></th>
<th><center>SEGUNDO NOMBRE</center></th>
<th><center>ID</center></th>
<th><center>NUMERO DE DOCUMENTO</center></th>
<th><center>EXPEDIDO EN</center></th>
<th><center>FECHA DE NACIMIENTO</center></th>
<th><center>EDAD</center></th>
<th><center>ESTADO CIVIL</center></th>
<th><center>NOMBRE EPS</center></th>
<th><center>ESTADO EPS</center></th>
<th><center>POBLACION ESPECIAL</center></th>
<th><center>GENERO</center></th>
<th><center>TELEFONO 1</center></th>
<th><center>TELEFONO 2</center></th>
<th><center>DIRECCION</center></th>
<th><center>SECTOR</center></th>
<th><center>CORREO ELECTRONICO 1</center></th>
<th><center>CORREO ELECTRONICO 2</center></th>
<th><center>ANTIGUEDAD SISBEN EN ANOS</center></th>
<th><center>PUNATAJE SISBEN</center></th>
<th><center>TRABAJA ACTUALMENTE</center></th>
<th><center>SECTOR LABORAL</center></th>
<th><center>NOMBRE DE LA EMPRESA</center></th>
<th><center>CARGO</center></th>
<th><center>VIVE EN VIVIENDA</center></th>
<th><center>HACE CUANTO VIVE EN ESTE LUGAR</center></th>
<th><center>ESTRATO</center></th>
<th><center>No. DE PERSONAS A CARGO</center></th>
<th><center>CUANTAS PERSONAS VIVEN EN TU NUCLEO FAMILIAR</center></th>
<th><center>NOMBRE DE CONTACTO DE EMERGENCIA</center></th>
<th><center>CC</center></th>
<th><center>TELEFONO</center></th>
<th><center>OCUPACION</center></th>
<th><center>PARENTESCO</center></th>



<th><center>CORREO ELECTRONICO</center></th>
<th><center>NOMBRE Y APELLIDOS COMPLETOS
REPRESENTANTE</center></th>
<th><center>CC</center></th>
<th><center>EXPEDIDA EN</center></th>
<th><center>TELEFONO</center></th>
<th><center>DIRECCION</center></th>
<th><center>PARENTESCO</center></th>
<th><center>INSTITUCION DE EDUCACION
MEDIA</center></th>
<th><center>FECHA DE GRADUACION</center></th>
<th><center>PUNTAJE ICFES</center></th>
<th><center>ANO DE PRESENTACION</center></th>
<th><center>PROMEDIO DE NOTAS DE GRADO
ONCE</center></th>
<th><center>PUESTO EN EL CURSO (11)</center></th>
<th><center>RECIBE OTRO BENEFICIO DIFERENTE A
FOEST</center></th>
<th><center>UNIVERSIDAD</center></th>
<th><center>SEDE</center></th>
<th><center>NIVEL</center></th>
<th><center>CARRERA</center></th>
<th><center>MODALIDAD</center></th>
<th><center>SEMESTRE</center></th>
<th><center>HORARIO</center></th>
<th><center>RAGISTRO SNIES</center></th>
<th><center>CONTINUA EN LA MISMA UNIVERSIDAD
Y EN EL MISMO PROGRAMA</center></th>
<th><center>VALOR SEMESTRE</center></th>
<th><center>PORCENTAJE A APLICAR</center></th>
<th><center>MONTO APROBADO</center></th>
<th><center>OTRO BENEFICIO</center></th>
<th><center>MATERIA PERDIDAS</center></th>
<th><center>PROMEDIO 1 2018-2</center></th>
<th><center>PROMEDIO 2 2018-1</center></th>
<th><center>BENEFICIOS RECIBIDOS EN EL
SEMESTRE ANTERIOR</center></th>
<th><center>BENEFICIOS RECIBIDOS EN EL
SEMESTRE ANTERIOR</center></th>
<th><center>HORAS SOCIALES
CUMPLIDAS</center></th>
<th><center>DEPENDENCIA DONDE REALIZO HORAS
SOCIALES</center></th>
<th><center>DEPENDENCIA DONDE REALIZO HORAS
SOCIALES</center></th>

```

<th><center>DEPENDENCIA DONDE REALIZO HORAS
SOCIALES</center></th>
<th><center>No. CUENTA DAVIPLATA ES No. DE
TELEFONO</center></th>
<th><center>DIAS DE LA SEMANA QUE
VIAJA</center></th>
<th><center>CIUDAD A LA QUE VIAJA</center></th>
<th><center>TOTAL DIAS</center></th>
<th><center>VALOR POR 2
RECORRIDOS</center></th>
<th><center>VALOR SEMANAL</center></th>
<th><center>VALOR MENSUAL BRUTO</center></th>
<th><center>VALOR A PAGAR POR MES</center></th>
<th><center>MESES A GIRAR</center></th>
<th><center>TOTAL SEMESTRE</center></th>
<th><center>RATIFICADO</center></th>
<th><center>APROBADO</center></th>
<th><center>OBSERVACIONES</center></th>

```

```
</thead>
```

```
<tbody>
```

```
<?php
```

```
\$result = mysqli_query(\$con, \$sql);
```

```
if(!\$result )
```

```
{
```

```
    die('Ocurrio un error al obtener los valores
de la base de datos: ' . mysqli_error());
```

```
}
```

```
while(\$row = mysqli_fetch_array(\$result,
MYSQLI_ASSOC))
```

```
{
```

```
    echo
```

```
    \<tr><td><center>{\$row['ID_SUB']}</cen
ter></td>\</tr>
```

```
    \<td><center>{\$row['ITEM']}</center>
</td>\</td>
```


```
    \<td><center>{\$row['NOCARPETA']}</center>
</td>\</td>
```

```
    \<td><center>{\$row['FECHARADICACION']}</center>
</td>\</td>
```


```
    \<td><center>{\$row['BENEFICIO1']}</center>
</td>\</td>
```

```
    \<td><center>{\$row['BENEFICIO2']}</center>
</td>\</td>
```


```
    \<td><center>{\$row['1VEZ']}</center>
```



```
</td>\".
\"<td><center>{\$row[\"RN\"]}</center>
</td>\".
\"<td><center>{\$row[\"REI\"]}</center>
</td>\".
\"<td><center>{\$row[\"PAPELLIDO\"]}
</center></td>\".
\"<td><center>{\$row[\"SAPELLIDO\"]}
</center></td>\".
\"<td><center>{\$row[\"PNOMBRE\"]}
</center></td>\".
\"<td><center>{\$row[\"SNOMBRE\"]}
</center></td>\".
\"<td><center>{\$row[\"ID_US\"]}</center>
</td>\".
\"<td><center>{\$row[\"NODOCUMENTO
\"]}</center></td>\".
\"<td><center>{\$row[\"EXPEDIDOEN\"]}
</center></td>\".
\"<td><center>{\$row[\"
FECHANACIMIENTO\"]}</center></td>\".
\"<td><center>{\$row[\"EDAD\"]}
</center></td>\".
\"<td><center>{\$row[\"ESTADOCIVIL\"]}
</center></td>\".
\"<td><center>{\$row[\"NEMBREEPS\"]}
</center></td>\".
\"<td><center>{\$row[\"ESTADOEPS\"]}
</center></td>\".
\"<td><center>{\$row[\"POESPECIAL
\"]}</center></td>\".
\"<td><center>{\$row[\"GENERO\"]}
</center></td>\".
\"<td><center>{\$row[\"TEL1\"]}
</center></td>\".
\"<td><center>{\$row[\"TEL2\"]}
</center></td>\".
\"<td><center>{\$row[\"DIRECCION\"]}
</center></td>\".
\"<td><center>{\$row[\"SECTOR\"]}
</center></td>\".
\"<td><center>{\$row[\"CORREOE1\"]}
</center></td>\".
\"<td><center>{\$row[\"CORREOE2\"]}
</center></td>\".
```




```
\<td><center>{\$row[\"ANTIGSISBEN\"]}
</center></td>\".
\<td><center>{\$row[\"PUNTASISBEN
\"]}</center></td>\".
\<td><center>{\$row[\"TRBAJAACUAL
\"]}</center></td>\".
\<td><center>{\$row[\"SECTORLABO\"]}
</center></td>\".
\<td><center>{\$row[\"NOMEMPRESA
\"]}</center></td>\".
\<td><center>{\$row[\"CARGOLAB\"]}
</center></td>\".
\<td><center>{\$row[\"VIVIENDA\"]}
</center></td>\".
\<td><center>{\$row[
\"TIEMPOVIVIENDO\"]}</center></td>\".
\<td><center>{\$row[\"ESTRATO\"]}
</center></td>\".
\<td><center>{\$row[
\"PERSONASCARGO\"]}</center></td>\".
\<td><center>{\$row[
\"NUCLEOFAMILIAR\"]}</center></td>\".
\<td><center>{\$row[
\"NOMBRECONTACTO\"]}</center>
</td>\".
\<td><center>{\$row[
\"DOCUMENTOCONTACTO\"]}</center>
</td>\".
\<td><center>{\$row[\"TELCONTACTO
\"]}</center></td>\".
\<td><center>{\$row[
\"OCUPACIONCONTACTO\"]}</center>
</td>\".
\<td><center>{\$row[\"PARENTESCO\"]}
</center></td>\".
\<td><center>{\$row[
\"CORREOECONTACTO\"]}</center>
</td>\".
\<td><center>{\$row[
\"NOMBREREPRESENTANTE\"]}
</center></td>\".
\<td><center>{\$row[\"NOCEDULAR\"]}
</center></td>\".
\<td><center>{\$row[\"EXPEDIDAR\"]}
</center></td>\".
```



```

\"<td><center>{\$row[\"TELREPR\"]}
</center></td>\".
\"<td><center>{\$row[
\"DIRECCIONREPR\"]}</center></td>\".
\"<td><center>{\$row[
\"PARENTESCOREP\"]}</center></td>\".
\"<td><center>{\$row[
\"INSTITUCIONEDUMEDIA\"]}</center>
</td>\".
\"<td><center>{\$row[\"FECHAGRADO\"]}
</center></td>\".
\"<td><center>{\$row[\"PUNTAJEICFES
\"]}</center></td>\".
\"<td><center>{\$row[
\"ANOPRESENTACION\"]}</center>
</td>\".
\"<td><center>{\$row[
\"PROMEDIOONCE\"]}</center></td>\".
\"<td><center>{\$row[\"PUESTOONCE\"]}
</center></td>\".
\"<td><center>{\$row[\"ROTRIBENE\"]}
</center></td>\".
\"<td><center>{\$row[\"UNIVERSIDAD\"]}
</center></td>\".
\"<td><center>{\$row[\"SEDE\"]}</center>
</td>\".
\"<td><center>{\$row[\"NIVEL\"]}</center>
</td>\".
\"<td><center>{\$row[\"CARRERA\"]}
</center></td>\".
\"<td><center>{\$row[\"MODALIDAD\"]}
</center></td>\".
\"<td><center>{\$row[\"SEMESTRE\"]}
</center></td>\".
\"<td><center>{\$row[\"HORARIO\"]}
</center></td>\".
\"<td><center>{\$row[\"SNIES\"]}
</center></td>\".
\"<td><center>{\$row[
\"CONTINUACARRERA\"]}</center>
</td>\".
\"<td><center>{\$row[
\"VALORSEMESTRE\"]}</center></td>\".
\"<td><center>{\$row[\"PORCENTAJEA
\"]}</center></td>\".

```

```

\<td><center>{\$row["MONTOAPRO"]}
</center></td>\".
\<td><center>{\$row["OTROBENE"]}
</center></td>\".
\<td><center>{\$row["MATERIASPER"]}
</center></td>\".
\<td><center>{\$row[
\"PROMEDIOANTERIOR\"]}</center>
</td>\".
\<td><center>{\$row[
\"PROMEDIOTRAANTERIOR\"]}
</center></td>\".
\<td><center>{\$row[
\"BENEFICIOSANTERIORES1"]}
</center></td>\".
\<td><center>{\$row[
\"BENEFICIOSANTERIORES2"]}
</center></td>\".
\<td><center>{\$row["HSCUMPLIDAS
\"]}</center></td>\".
\<td><center>{\$row["DEPENDENCIA1
\"]}</center></td>\".
\<td><center>{\$row["DEPENDENCIA2
\"]}</center></td>\".
\<td><center>{\$row["DEPENDENCIA3
\"]}</center></td>\".
\<td><center>{\$row["NOCUENTA"]}
</center></td>\".
\<td><center>{\$row["DIASVIAJADOS
\"]}</center></td>\".
\<td><center>{\$row["CIUDADVIAJA
\"]}</center></td>\".
\<td><center>{\$row["DIAS\"]}</center>
</td>\".
\<td><center>{\$row["VALORRECOR
\"]}</center></td>\".
\<td><center>{\$row["VALORSEMANA
\"]}</center></td>\".
\<td><center>{\$row[
\"VALORMESBRUTO\"]}</center></td>\".
\<td><center>{\$row[
\"VALORMESPAGAR\"]}</center></td>\".
\<td><center>{\$row["MESES"]}
</center></td>\".
\<td><center>{\$row[

```

```

        \ "TOTALSEMESTRE\"])</center></td>\ ".
        \ "<td><center>{\$row["RATIFICADO\"])
        </center></td>\ ".
        \ "<td><center>{\$row["APROBADO\"])
        </center></td>\ ".
        \ "<td><center>{\$row[
        \ "OBSERVACIONRAT\"])</center></td>
        </tr>\ ";
    }
    mysqli_close(\$con);
}
?>
</tbody>
</table>
</div>
</form>
</body>
</html>";
}
if ($base = fopen($base, "a"))
{
    if (fwrite($base, "$mensaje"))
    {
    }
    Else
// error en caso de no dejarse escribir el archivo
    {
        ?>
        <script type="text/javascript">
            alert("Ha habido un problema tablanueva.php");
            window.location.href='index.php';
        </script>
        <?php
    }
    fclose($base);
}
?>

```

9.1.11 pantalla creación de registro por periodos

```
<?php
    $reg = "registro$periodo.php";
    if(file_exists ($reg))
    {
    }
    else
    {
        $mensa = "<?php
include(\"blodese.php\");
include (\"conexion.php\");
\$$consulta = registrobeneficiario(\$_GET['documento']);
function registrobeneficiario(\$NODOCUMENTO)
{
    include (\"conexion.php\");
    \$$benefi = mysqli_query(\$conn, \"SELECT * FROM `tabla` WHERE `tabla` .
    NODOCUMENTO = \"\$NODOCUMENTO\") or die (\"no conecto estudiantes20191\");;
    if (mysqli_num_rows(\$$benefi)>0)
    {
        ?>
        <script type=\"text/javascript\">
            alert(\"Usuario ya registrado en el periodo $periodo\");
            window.location.href='nuevo$periodo.php';
        </script>
        <?php
    }
    else
    {
        \$$usuario = mysqli_query(\$con , \"SELECT * FROM `fuid` WHERE `fuid` .
        NODOCUMENTO = \"\$NODOCUMENTO\") or die (\"No conecto fuid\");
        if (mysqli_num_rows(\$$usuario) > 0)
        {
            ?>
            <script type=\"text/javascript\">
                alert(\"Usuario ya registrado en el FUID\");
            </script>
            <?php
            \$$filas=mysqli_fetch_assoc(\$$usuario);
            return
                [
                    \$$filas['NODOCUMENTO'],
                    \$$filas['PRAPELLIDO'],
                    \$$filas['SEAPPELLIDO'],
```

```

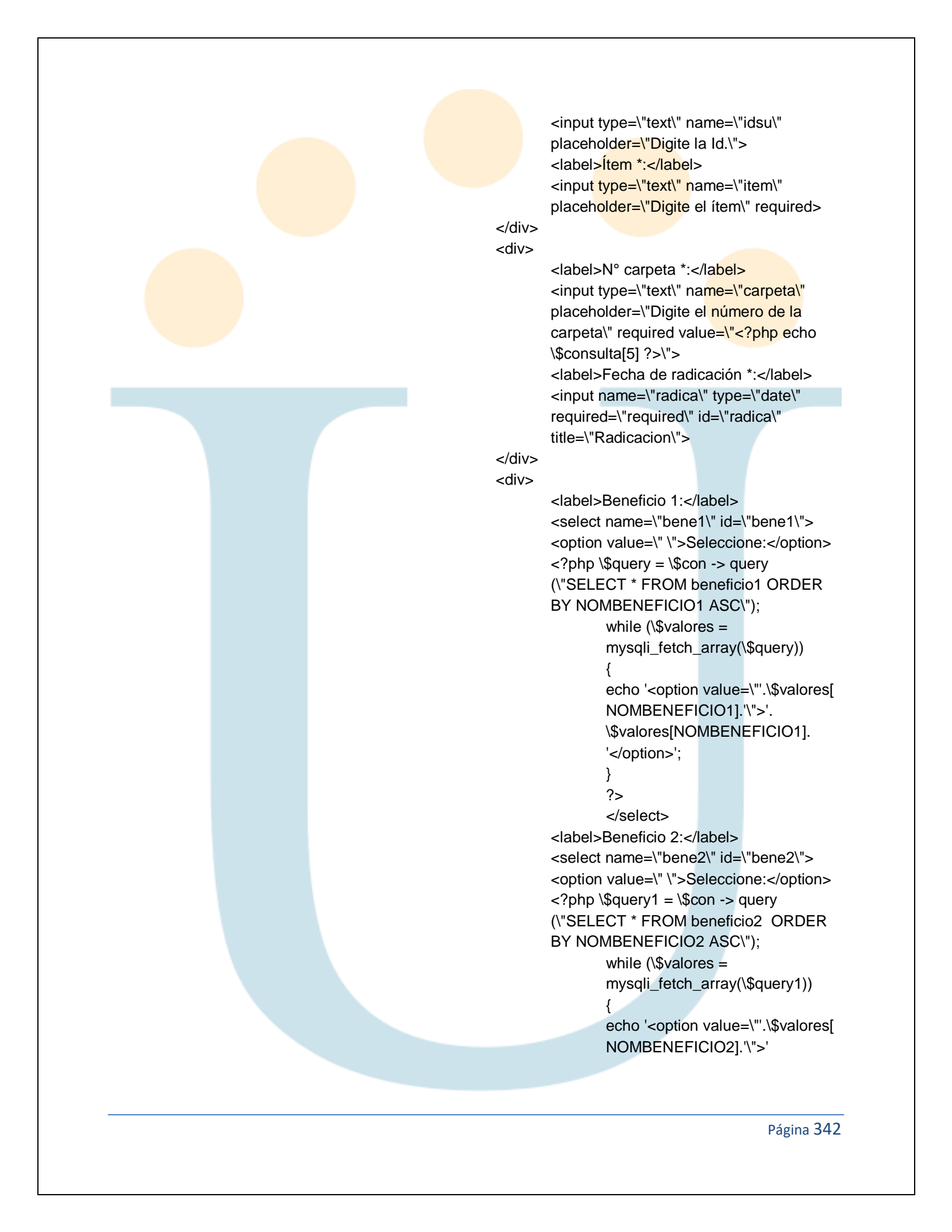
        ];
    }
    else
    {
        ?>
        <script type="text/javascript">
            alert("Usuario Nuevo por favor registrar en el FUID");
        </script>
        <?php
    }
}
?>
<!doctype html>
<html>
    <head>
        <title>Datos</title>
        <link rel="stylesheet" type="text/css" href="estilos.css">
        <script language="javascript">
            function multiplica(form)
            {
                var resultado;
                var resultado2;
                var resultado3;
                var montoaprovado;
                var dias=0;
                var valor=0;
                var mes=0;
                var mespago=0;
                var valsemestre=0;
                var porcentajes=0;
                var universidad=0;
                universidad = parseInt (form.univer.value);
                valsemestre = parseInt (form.vsemestre.value);
                porcentajes = parseInt (form.porcentaje.value);
                dias = parseInt (form.tdias.value);
                valor = parseInt (form.val2r.value);
                mes = parseInt (form.mesgi.value);
                mespago = parseInt (form.valp.value);
                montoaprovado = (valsemestre * porcentajes)/100;
                form.monto.value = montoaprovado;
                resultado = dias * valor;
            }
        </script>
    </head>
    <body>
        <div style="text-align: center;">
            <h2>Formulario de Datos</h2>
            <hr style="width: 50%; margin: auto;"/>
            <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;">
                <thead>
                    <tr>
                        <th style="width: 20%;>Nombre</th>
                        <th style="width: 20%;>Apellido</th>
                        <th style="width: 20%;>Carnet</th>
                        <th style="width: 20%;>Carnet</th>
                    </tr>
                </thead>
                <tbody>
                    <tr>
                        <td style="width: 20%;>
                            <input type="text" value="<?php echo <!--
                    </td>
                </tr>
            </tbody>
        </table>
    </body>
</html>

```

```

    form.vals.value=resultado;
    resultado2 = 4 * resultado;
    form.valm.value=resultado2;
    if (resultado2 < 156246)
        {
            form.valp.value = resultado2;
        }
    else
        {
            form.valp.value = 156246;
        }
    resultado3= mespago * mes;
    form.tsemestre.value=resultado3;
}
</script>
</head>
<body>
<form>
    <spam class="text">
    <div style = \"float: left;\"></div>
    <span style=\"float: left;\"><?php echo \\$_SESSION[\"usuarios\"];?></span>
    <div style=\"float: right;\" align=\"right\"><a href=\"salir.php\" align = \"right\">
Salir</a></div></spam>
    <spam class="text"></spam>
</form>
    <form>
    <spam class="text">
    <p><a href=\"nuevo$periodo.php\">Anterior</a></p>
    <H2><center>Ingrese los datos:</center></H2>
    Los campos marcados con asterisco (*) son obligatorios
    </spam>
</form>
<div></div>
    <form id="form1" action="creg$periodo.php" method="POST">
    <span class="text">
    <center>
    <table>
    <tr>
    <td><H3>
    <div>
    <label><center><h1>Datos
personales:</h1></center></label>
    </div>
    <div>
    <label>ID:</label>

```



```
<input type="text" name="idsu"
placeholder="Digite la Id.">
<label>Ítem *:</label>
<input type="text" name="item"
placeholder="Digite el ítem" required>
</div>
<div>
```

```
<label>Nº carpeta *:</label>
<input type="text" name="carpeta"
placeholder="Digite el número de la
carpeta" required value="<?php echo
\$consulta[5] ?>">
<label>Fecha de radicación *:</label>
<input name="radica" type="date"
required="required" id="radica"
title="Radicacion">
```

```
</div>
<div>
<label>Beneficio 1:</label>
<select name="bene1" id="bene1">
<option value="">Seleccione:</option>
<?php \$query = \$con -> query
("SELECT * FROM beneficio1 ORDER
BY NOMBENEFICIO1 ASC");
while (\$valores =
mysqli_fetch_array(\$query))
{
echo '<option value="".'.\$valores[
NOMBENEFICIO1].'\>'.
\$valores[NOMBENEFICIO1].
'\</option>';
}
?>
</select>
```

```
<label>Beneficio 2:</label>
<select name="bene2" id="bene2">
<option value="">Seleccione:</option>
<?php \$query1 = \$con -> query
("SELECT * FROM beneficio2 ORDER
BY NOMBENEFICIO2 ASC");
while (\$valores =
mysqli_fetch_array(\$query1))
{
echo '<option value="".'.\$valores[
NOMBENEFICIO2].'\>'
```

```

        .\${valores[NOMBENEFICIO2]}'
    </option>;
    }
    ?>
</select>

</div>
<div>
    <label>1° Vez:</label>
    <input type=\"text\" name=\"1vez\"
    placeholder=\" \">
    <label>Renovación:</label>
    <input type=\"text\" name=\"rn\"
    placeholder=\" \">

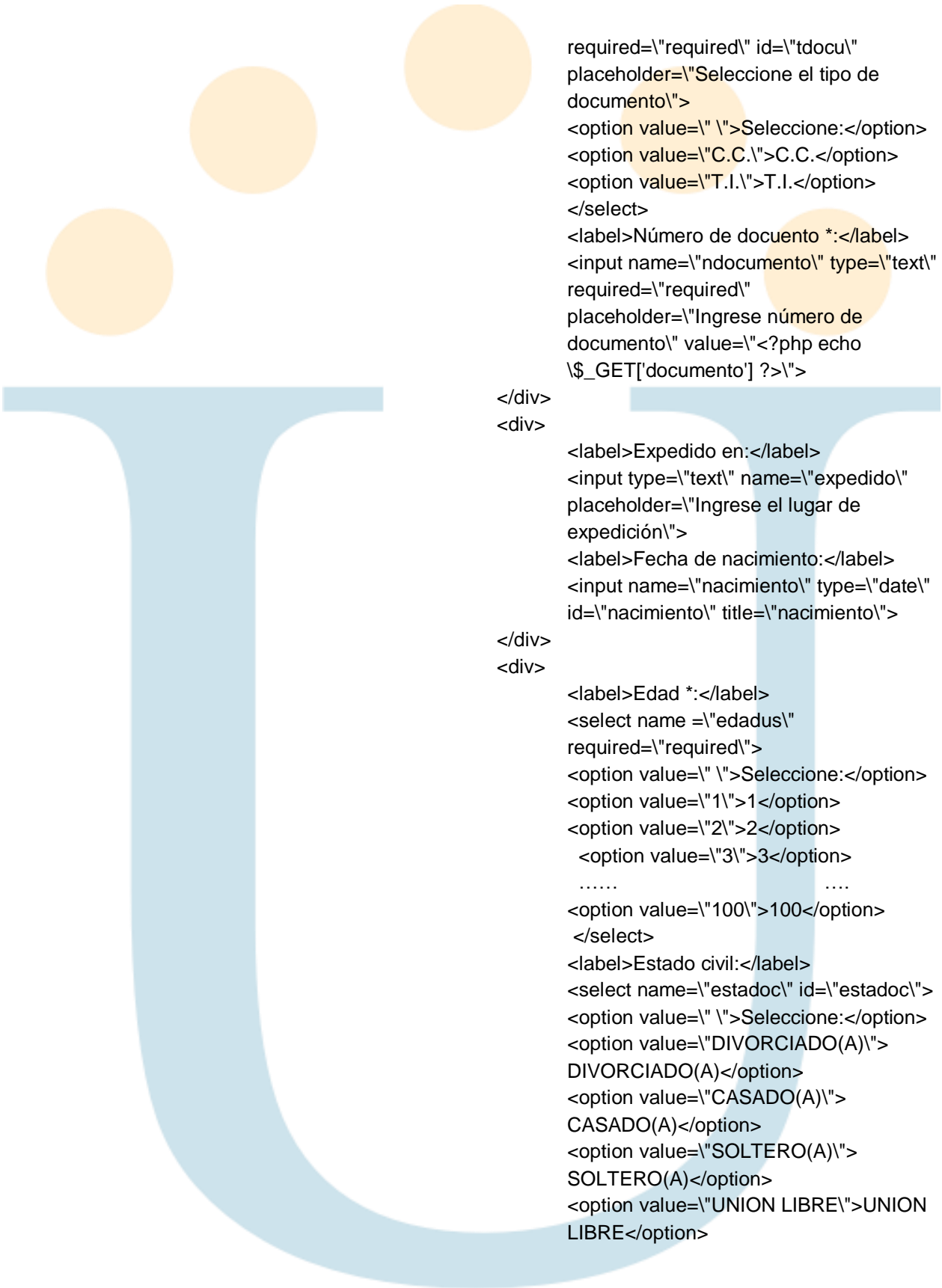
</div>
<div>
    <label>Reintegro:</label>
    <input type=\"text\" name=\"rei\"
    placeholder=\" \">

</div>
<div>
    <label>Primer apellido *:</label>
    <input name=\"1apellido\" type=\"text\"
    required=\"required\"
    placeholder=\"Ingrese primer apellido\"
    value=\"<?php echo \${consulta[1]} ?>\">
    <label>Segundo apellido *:</label>
    <input name=\"2apellido\" type=\"text\"
    required=\"required\"
    placeholder=\"Ingrese primer apellido\"
    value=\"<?php echo \${consulta[2]} ?>\">

</div>
<div>
    <label>Primer nombre *:</label>
    <input name=\"1nombre\" type=\"text\"
    required=\"required\"
    placeholder=\"Ingrese primer nombre\"
    value=\"<?php echo \${consulta[3]} ?>\">
    <label>Segundo nombre:</label>
    <input name=\"2nombre\" type=\"text\"
    placeholder=\"Ingrese segundo nombre\"
    value=\"<?php echo \${consulta[4]} ?>\">

</div>
<div>
    <label>Tipo de documento *:</label>
    <select name=\"tdocu\"

```



```
required=\required\ id=\tdocu\
placeholder=\Seleccione el tipo de
documento\>
<option value=\ \>Seleccione:</option>
<option value=\C.C.\>C.C.</option>
<option value=\T.I.\>T.I.</option>
</select>
<label>Número de documento *:</label>
<input name=\ndocumento\ type=\text\
required=\required\
placeholder=\Ingrese número de
documento\ value=\<?php echo
\$_GET['documento'] ?>\>
```

```
</div>
<div>
```

```
<label>Expedido en:</label>
<input type=\text\ name=\expedido\
placeholder=\Ingrese el lugar de
expedición\>
<label>Fecha de nacimiento:</label>
<input name=\nacimiento\ type=\date\
id=\nacimiento\ title=\nacimiento\>
```

```
</div>
<div>
```

```
<label>Edad *:</label>
<select name =\edadus\
required=\required\>
<option value=\ \>Seleccione:</option>
<option value=\1\>1</option>
<option value=\2\>2</option>
<option value=\3\>3</option>
.....
<option value=\100\>100</option>
</select>
<label>Estado civil:</label>
<select name=\estadoc\ id=\estadoc\>
<option value=\ \>Seleccione:</option>
<option value=\DIVORCIADO(A)\>
DIVORCIADO(A)</option>
<option value=\CASADO(A)\>
CASADO(A)</option>
<option value=\SOLTERO(A)\>
SOLTERO(A)</option>
<option value=\UNION LIBRE\>UNION
LIBRE</option>
```



```
<option value=\"VIUDO(A)\">
VIUDO(A)</option>
</select>
```

```
</div>
<div>
```

```
<label>Nombre de EPS:</label>
<input name=\"eps\" type=\"text\"
placeholder=\"Ingrese el nombre de la
EPS\">
```

```
<label>Estado EPS:</label>
<select name=\"estadoeps\"
id=\"estadoeps\">
<option value=\" \">Seleccione:</option>
<option value=\"BENEFICIARIO(A)\">
BENEFICIARIO(A)</option>
<option value=\"COTIZANTE\">
COTIZANTE</option>
<option value=\"SUBSIDIARIO\">
SUBSIDIARIO</option>
</select>
```

```
</div>
<div>
```

```
<label>Población especial:</label>
<select name=\"pespe\" id=\"pespe\">
<option value=\" \">Seleccione:</option>
<?php \$query2 = \$con -> query
(\"SELECT * FROM poblacion ORDER
BY NOMPOBLACION ASC\");
while (\$valores =
mysqli_fetch_array(\$query2))
{
echo '<option value=\"\".\$valores
[NOMPOBLACION].'\>.\$valores
[NOMPOBLACION].'\</option>';
}
?>
```

```
</select>
<label>Género *:</label>
<select name=\"gene\"
required=\"required\" id=\"gene\">
<option value=\" \">Seleccione:</option>
<option value=\"FEMENINO\">
FEMENINO</option>
<option value=\"MASCULINO\">
MASCULINO</option>
```

```

<option value="OTRO">OTRO</option>
<select>
</div>
<div>
<label>Teléfono 1 *:</label>
<input name="tel1" type="text"
required="required"
placeholder="Ingrese número
telefónico">
<label>Teléfono 2:</label>
<input type="text" name="tel2"
placeholder="Ingrese otro número
telefónico">
</div>
<div>
<label>Dirección:</label>
<input type="text" name="direccion"
placeholder="Ingrese dirección">
<label>Sector:</label>
<input type="text" name="sector"
placeholder="Ingrese el sector">
</div>
<div>
<label>Correo electrónico 1:</label>
<input type="text" name="correo"
placeholder="Ingrese correo">
<label>Correo electrónico 2:</label>
<input type="text" name="correo2"
placeholder="Ingrese otro correo">
</div>
<div>
<label>Antigüedad Sisbén en
años:</label>
<select name="asisben" id="asisben">
<option value="">Seleccione:</option>
<?php \$query3 = \$con -> query
("SELECT * FROM antigu_sisben");
while (\$valores =
mysqli_fetch_array(\$query3))
{
echo '<option value="' . \$valores
[NOMASISBEN].'\ ">'. \$valores
[NOMASISBEN].'\ ">';
}
?>

```

```
</select>
<label>Puntaje del Sisbén:</label>
<input type="text" name="psisben"
placeholder="Ingrese el puntaje">
</div>
<div>
```

```
<label>Trabaja actualmente:</label>
<select name="trabaja" id="trabaja">
<option value="" >Seleccione:</option>
<option value="SI">SI</option>
<option value="NO">NO</option>
</select>
```

```
<label>Sector laboral:</label>
<select name="sectorlab"
id="sectorlab">
<option value="" >Seleccione:</option>
<option value="NINGUNO">
NINGUNO</option>
<option value="PRIVADO">
PRIVADO</option>
<option value="PUBLICO">
PUBLICO</option>
</select>
```

```
</div>
<div>
```

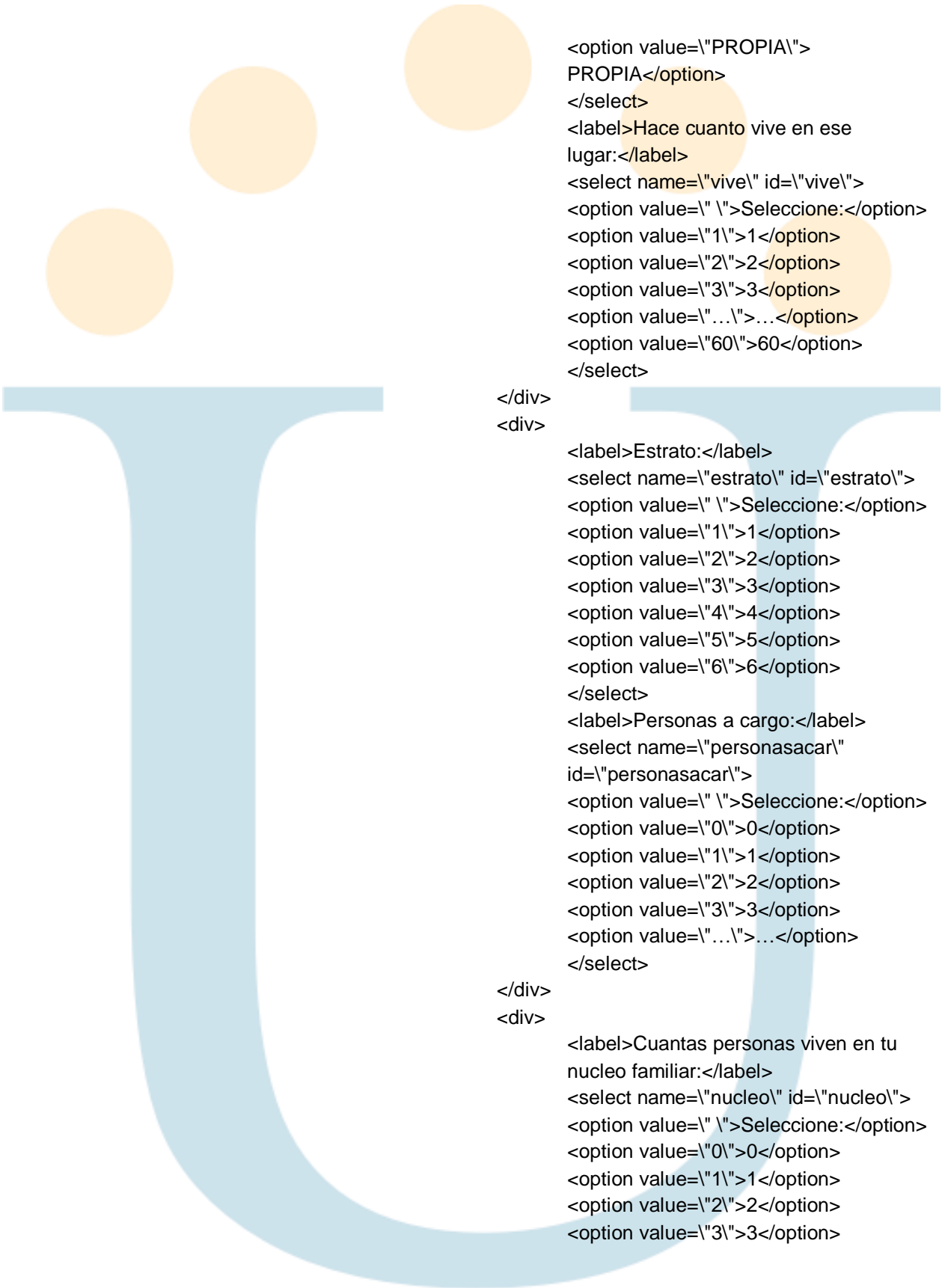
```
<label>Nombre de la empresa:</label>
<input type="text" name="empresa"
placeholder="Ingrese nombre de la
empresa">
<label>Cargo:</label>
<input type="text" name="cargo"
placeholder="Ingrese cargo">
```

```
</div>
<div>
```

```
<label><center><h1>Datos del
hogar:</h1></center></label>
```

```
</div>
<div>
```

```
<label>Vive en vivienda:</label>
<select name="vivienda" id="vivienda">
<option value="" >Seleccione:</option>
<option value="ARRENDADA">
ARRENDADA</option>
<option value="FAMILIAR">
FAMILIAR</option>
```



```
<option value=\"PROPIA\">
PROPIA</option>
</select>
<label>Hace cuanto vive en ese
lugar:</label>
<select name=\"vive\" id=\"vive\">
<option value=\" \">Seleccione:</option>
<option value=\"1\">1</option>
<option value=\"2\">2</option>
<option value=\"3\">3</option>
<option value=\"...\">...</option>
<option value=\"60\">60</option>
</select>
```

```
</div>
<div>
```

```
<label>Estrato:</label>
<select name=\"estrato\" id=\"estrato\">
<option value=\" \">Seleccione:</option>
<option value=\"1\">1</option>
<option value=\"2\">2</option>
<option value=\"3\">3</option>
<option value=\"4\">4</option>
<option value=\"5\">5</option>
<option value=\"6\">6</option>
</select>
<label>Personas a cargo:</label>
<select name=\"personasacar\"
id=\"personasacar\">
<option value=\" \">Seleccione:</option>
<option value=\"0\">0</option>
<option value=\"1\">1</option>
<option value=\"2\">2</option>
<option value=\"3\">3</option>
<option value=\"...\">...</option>
</select>
```

```
</div>
<div>
```

```
<label>Cuantas personas viven en tu
nucleo familiar:</label>
<select name=\"nucleo\" id=\"nucleo\">
<option value=\" \">Seleccione:</option>
<option value=\"0\">0</option>
<option value=\"1\">1</option>
<option value=\"2\">2</option>
<option value=\"3\">3</option>
```

```
<option value=\"...\">...</option>
<option value=\"20\">20</option>
</select>
```

```
</div>
<div>
```

```
<label><center><h1>Contacto de
emergencia:</h1></center></label>
```

```
</div>
<div>
```

```
<label>Nombre de contacto:</label>
<input type=\"text\" name=\"nombrecon\"
placeholder=\"Ingrese nombre del
contacto\">
```

```
<label>Cedula de Ciudadania:</label>
<input type=\"text\" name=\"cc\"
placeholder=\"Cedula del contacto\">
```

```
</div>
<div>
```

```
<label>Telefono:</label>
<input type=\"text\" name=\"telcon\"
placeholder=\"Telefono\">
<label>Ocupacion:</label>
<input type=\"text\" name=\"ocupacionc\"
placeholder=\"Ocupacion del contacto\">
```

```
</div>
<div>
```

```
<label>Parentesco:</label>
<input type=\"text\" name=\"parentesco\"
placeholder=\"Parentesco\">
<label>Correo electronico:</label>
<input type=\"text\" name=\"correocon\"
placeholder=\"Correo\">
```

```
</div>
<div>
```

```
<label><center><h1>Representante
Legal:</h1></center></label>
```

```
</div>
<div>
```

```
<label>Nombres y apellidos
completos:</label>
<input type=\"text\" name=\"nombrerep\"
placeholder=\"Ingrese nombre del
representante\">
<label>Cedula de Ciudadania:</label>
<input type=\"text\" name=\"ccrep\"
```

placeholder="Cedula del representante">

</div>

<div>

<label>Expedido en:</label>

<input type="text" name="expedidorep" placeholder="Ingrese el lugar de expedición">

<label>Telefono:</label>

<input type="text" name="Telrep" placeholder="Ingrese el número de telefono">

</div>

<div>

<label>Dirección:</label>

<input type="text" name="Dirrep" placeholder="Ingrese dirección de contacto">

<label>Parentesco:</label>

<input type="text" name="Paretescorep" placeholder="Parentesco">

</div>

<div>

<label><center><h1>Educacion media:</h1></center></label>

</div>

<div>

<label>Institución:</label>

<input type="text" name="institucion" placeholder="Nombre de la institucion">

<label>Fecha de graduacion:</label>

<input name="grado" type="date" id="grado" title="grado">

</div>

<div>

<label>Puntaje Icfes:</label>

<input type="text" name="icfes" placeholder="Puntaje">

<label>Año de Presentación</label>

<input type="text" name="presentaicfes" placeholder="Año de presentación">

</div>

<div>

```
<label>Promedio de notas de grado
Once:</label>
<input type="text" name="notas11"
placeholder="promedio">
<label>Puesto en el grado Once:</label>
<input type="text" name="puesto11"
placeholder="Puesto grado 11">
```

```
</div>
<div>
```

```
<label><center><h1>
</h1></center></label>
```

```
</div>
<div>
```

```
<label>Recibe algún otro beneficio:
</label>
<select name="robene" id="robene">
<option value="" >Seleccione:</option>
<?php \$query4 = \$con -> query
("SELECT * FROM algunotro ORDER BY
NOMBREALGUN ASC");
while (\$valores =
mysqli_fetch_array(\$query4))
{
echo '<option value="".'.\$valores
[NOMBREALGUN].'\>'.\$valores
[NOMBREALGUN].'\</option>';
}
?>
</select>
```

```
</div>
<div>
```

```
<label><center><h1>Información de la
universidad:</h1></center></label>
```

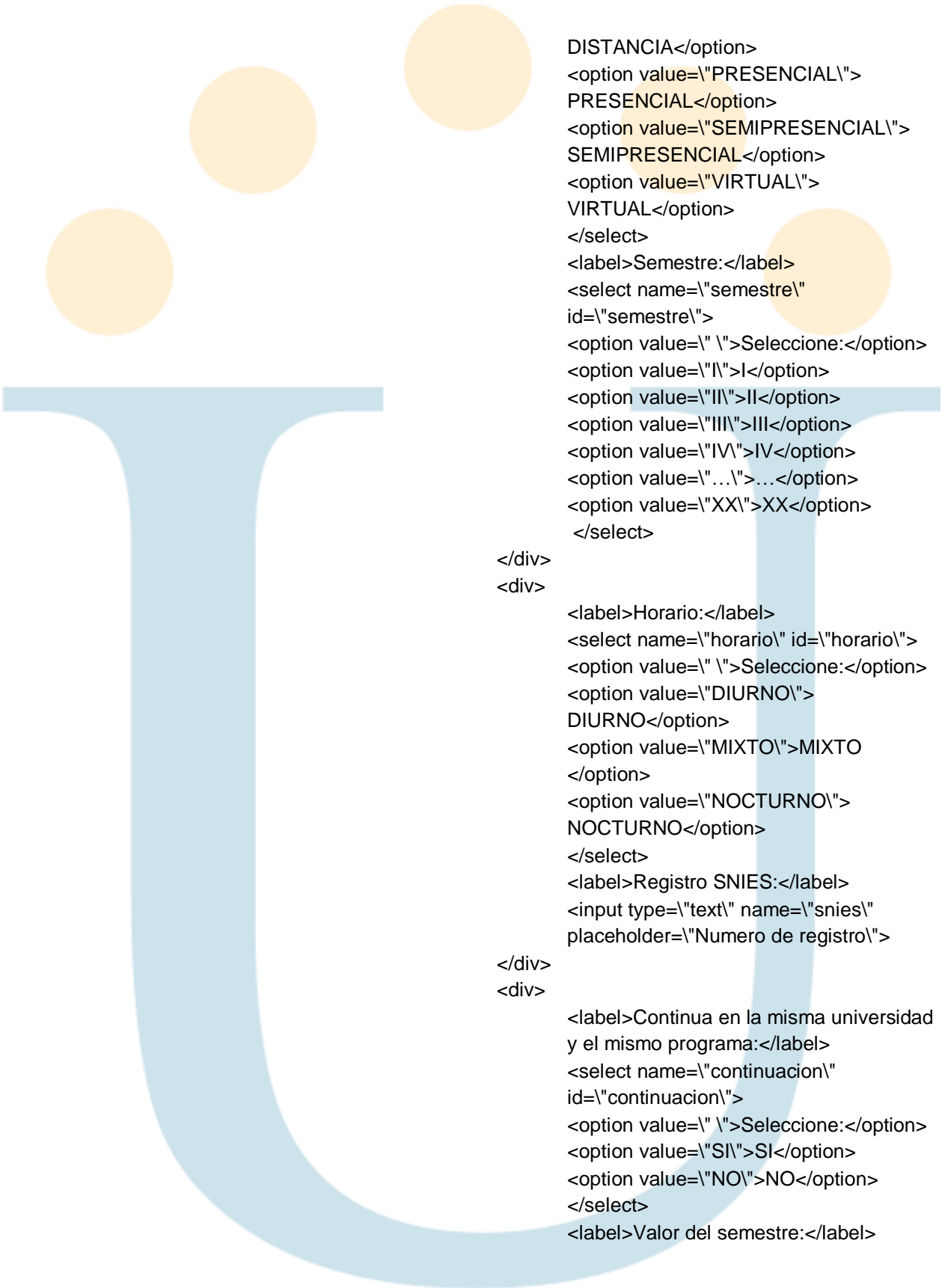
```
</div>
<div>
```

```
<label>Universidad:</label>
<select name="univer" id="univer">
<option value="" >Seleccione:</option>
<?php \$query5 = \$con -> query
("SELECT * FROM universidad ORDER
BY NOMUNIVERSIDAD ASC");
while (\$valores =
mysqli_fetch_array(\$query5))
{
echo '<option value="".'.\$valores
```

```

        [NOMUNIVERSIDAD].\">'.
        \$valores[NOMUNIVERSIDAD].
        '</option>';
    }
?>
</select>
<label>Sede:</label>
<input type=\"text\" name=\"sede\"
placeholder=\"Ingrese sede\">
</div>
<div>
<label>Nivel:</label>
<select name=\"niv\" id=\"niv\">
<option value=\" \">Seleccione:</option>
<option value=\"PREGRADO\">
PREGRADO</option>
<option value=\"POSTGRADO\">
POSTGRADO</option>
<option value=\"TECNICO\">
TECNICO</option>
<option value=\"TECNOLOGO\">
TECNOLOGO</option>
</select>
<label>Carrera:</label>
<select name=\"carrera\" id=\"carrera\">
<option value=\" \">Seleccione:</option>
<?php \$query6 = \$con -> query
(\"SELECT * FROM carreras ORDER BY
NOMCARRERA ASC\");
    while (\$valores =
    mysqli_fetch_array(\$query6))
    {
    echo '<option value=\"\".\$valores
[NOMCARRERA].\">'.\$valores
[NOMCARRERA].\"</option>';
    }
?>
</select>
</div>
<div>
<label>Modalidad:</label>
<select name=\"modalidad\"
id=\"modalidad\">
<option value=\" \">Seleccione:</option>
<option value=\"DISTANCIA\">

```

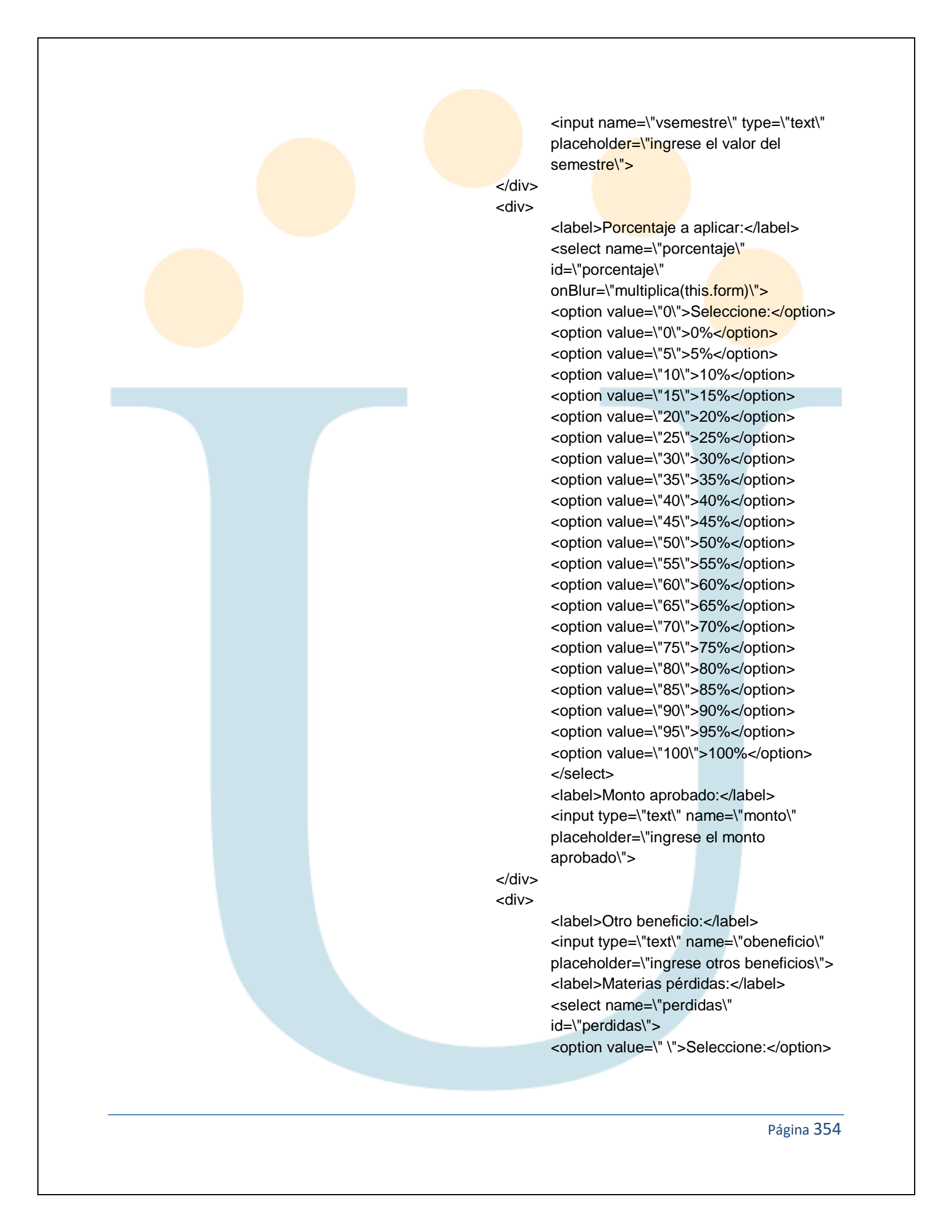
```
DISTANCIA</option>
<option value=\"PRESENCIAL\">
PRESENCIAL</option>
<option value=\"SEMIPRESENCIAL\">
SEMIPRESENCIAL</option>
<option value=\"VIRTUAL\">
VIRTUAL</option>
</select>
<label>Semestre:</label>
<select name=\"semestre\"
id=\"semestre\">
<option value=\" \">Seleccione:</option>
<option value=\"I\">I</option>
<option value=\"II\">II</option>
<option value=\"III\">III</option>
<option value=\"IV\">IV</option>
<option value=\"...\">...</option>
<option value=\"XX\">XX</option>
</select>
```

```
</div>
<div>
```

```
<label>Horario:</label>
<select name=\"horario\" id=\"horario\">
<option value=\" \">Seleccione:</option>
<option value=\"DIURNO\">
DIURNO</option>
<option value=\"MIXTO\">MIXTO
</option>
<option value=\"NOCTURNO\">
NOCTURNO</option>
</select>
<label>Registro SNIES:</label>
<input type=\"text\" name=\"snies\"
placeholder=\"Numero de registro\">
```

```
</div>
<div>
```

```
<label>Continúa en la misma universidad
y el mismo programa:</label>
<select name=\"continuacion\"
id=\"continuacion\">
<option value=\" \">Seleccione:</option>
<option value=\"SI\">SI</option>
<option value=\"NO\">NO</option>
</select>
<label>Valor del semestre:</label>
```



```
<input name=\"vsemestre\" type=\"text\"  
placeholder=\"ingrese el valor del  
semestre\">
```

```
</div>  
<div>
```

```
<label>Porcentaje a aplicar:</label>  
<select name=\"porcentaje\"  
id=\"porcentaje\"  
onBlur=\"multiplica(this.form)\">  
<option value=\"0\">Seleccione:</option>  
<option value=\"0\">0%</option>  
<option value=\"5\">5%</option>  
<option value=\"10\">10%</option>  
<option value=\"15\">15%</option>  
<option value=\"20\">20%</option>  
<option value=\"25\">25%</option>  
<option value=\"30\">30%</option>  
<option value=\"35\">35%</option>  
<option value=\"40\">40%</option>  
<option value=\"45\">45%</option>  
<option value=\"50\">50%</option>  
<option value=\"55\">55%</option>  
<option value=\"60\">60%</option>  
<option value=\"65\">65%</option>  
<option value=\"70\">70%</option>  
<option value=\"75\">75%</option>  
<option value=\"80\">80%</option>  
<option value=\"85\">85%</option>  
<option value=\"90\">90%</option>  
<option value=\"95\">95%</option>  
<option value=\"100\">100%</option>  
</select>  
<label>Monto aprobado:</label>  
<input type=\"text\" name=\"monto\"  
placeholder=\"ingrese el monto  
aprobado\">
```

```
</div>  
<div>
```

```
<label>Otro beneficio:</label>  
<input type=\"text\" name=\"obeneficio\"  
placeholder=\"ingrese otros beneficios\">  
<label>Materias pérdidas:</label>  
<select name=\"perdidas\"  
id=\"perdidas\">  
<option value=\" \">Seleccione:</option>
```

```

<option value="0">0</option>
<option value="1">1</option>
<option value="2">2</option>
<option value="3">3</option>
<option value="...">...</option>
<option value="10">10</option>
</select>

```

```
</div>
```

```
<div>
```

```

<label>Promedio 1:</label>
<input type="text" name="promedioant"
placeholder="Ingrese el promedio 1"
id="promedioant">
<label>Promedio 2:</label>
<input type="text" name="promediotra"
placeholder="Ingrese el promedio 2">

```

```
</div>
```

```
<div>
```

```

<label>Beneficios recibidos en el
semestre anterior:</label>
<select name="bene1ant"
id="bene1ant">
<option value="">Seleccione:</option>
<?php \$query = \$con -> query
("SELECT * FROM beneficio1 ORDER
BY NOMBENEFICIO1 ASC");
while (\$valores =
mysqli_fetch_array(\$query))
{
echo '<option value="' . \$valores
[NOMBENEFICIO1] . '">' . \$valores
[NOMBENEFICIO1] . '</option>';
}

```

```
?>
```

```
</select>
```

```

<label>Beneficios recibidos en el
semestre anterior:</label>

```

```
<select name="bene2ant"
```

```
id="bene2ant">
```

```
<option value="">Seleccione:</option>
```

```

<?php \$query = \$con -> query
("SELECT * FROM beneficio1 ORDER
BY NOMBENEFICIO1 ASC");

```

```

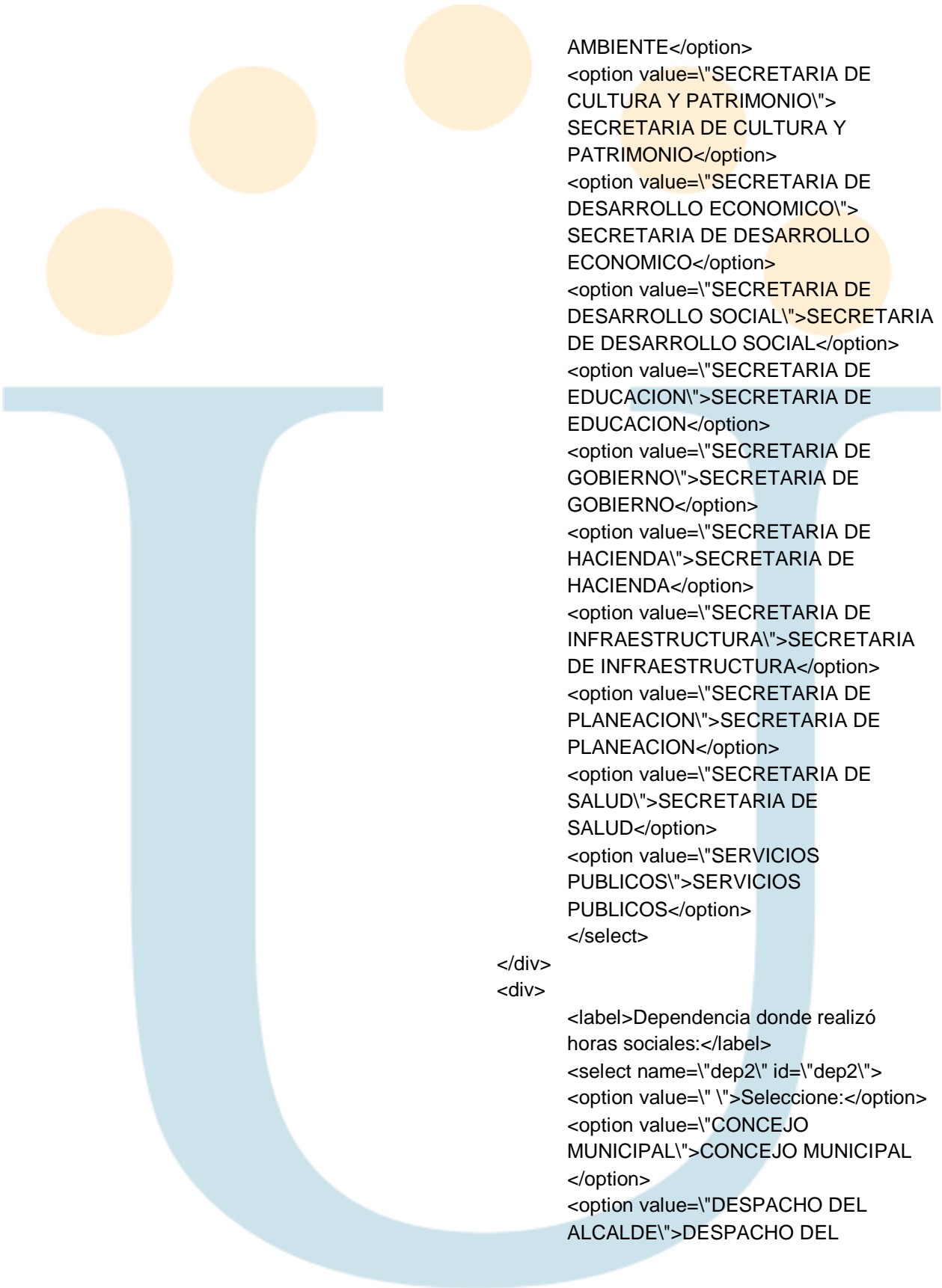
while (\$valores =
mysqli_fetch_array(\$query))

```

```

        {
        echo '<option value=\"\".\\$valores
[NOMBENEFICIO1].\">'.\\$valores
[NOMBENEFICIO1].'</option>';
        }
    ?>
</select>
</div>
<div>
<label>Horas sociales cumplidas:</label>
<select name=\"hscump\" id=\"hscump\">
<option value=\"\">Seleccione:</option>
<option value=\"1\">1</option>
<option value=\"2\">2</option>
<option value=\"3\">3</option>
<option value=\"...\">...</option>
<option value=\"100\">100</option>
</select>
<label>Dependencia donde realizó horas
sociales:</label>
<select name=\"dep1\" id=\"dep1\">
<option value=\"\">Seleccione:</option>
<option value=\"CONCEJO
MUNICIPAL\">CONCEJO MUNICIPAL
</option>
<option value=\"DESPACHO DEL
ALCALDE\">DESPACHO DEL
ALCALDE</option>
<option value=\"IMRD\">IMRD</option>
<option value=\"NINGUNO\">NINGUNO
</option>
<option value=\"OFICINA DE PRENSA\">
OFICINA DE PRENSA</option>
<option value=\"OFICINA JURIDICA Y
DE CONTRATACION\">OFICINA
JURIDICA Y DE CONTRATACION
</option>
<option value=\"PERSONERIA
MUNICIPAL\">PERSONERIA
MUNICIPAL</option>
<option value=\"SECRETARIA
ADMINISTRATIVA\">SECRETARIA
ADMINISTRATIVA</option>
<option value=\"SECRETARIA DE
AMBIENTE\">SECRETARIA DE

```

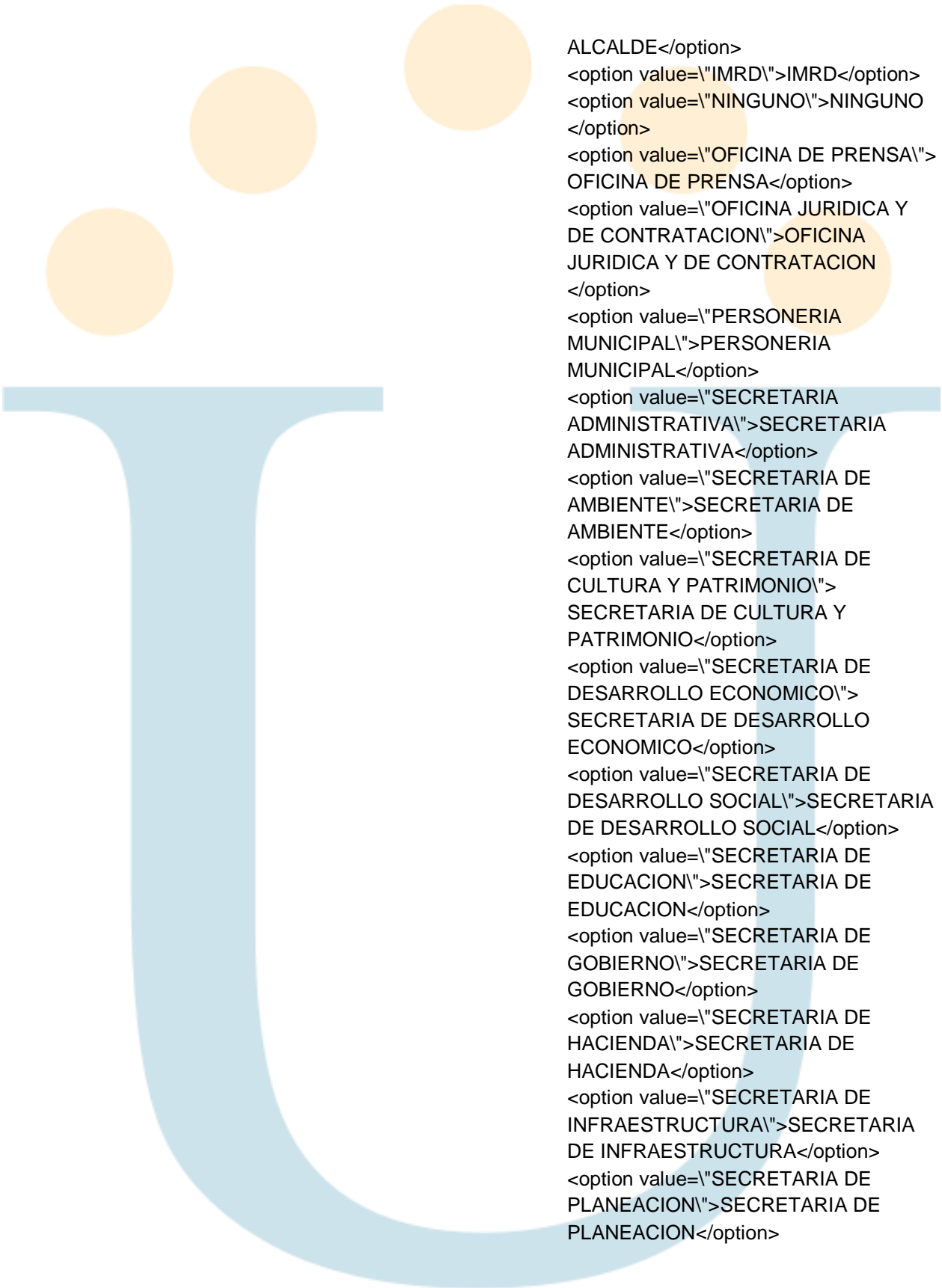


```
AMBIENTE</option>
<option value=\ "SECRETARIA DE
CULTURA Y PATRIMONIO\ ">
SECRETARIA DE CULTURA Y
PATRIMONIO</option>
<option value=\ "SECRETARIA DE
DESARROLLO ECONOMICO\ ">
SECRETARIA DE DESARROLLO
ECONOMICO</option>
<option value=\ "SECRETARIA DE
DESARROLLO SOCIAL\ ">SECRETARIA
DE DESARROLLO SOCIAL</option>
<option value=\ "SECRETARIA DE
EDUCACION\ ">SECRETARIA DE
EDUCACION</option>
<option value=\ "SECRETARIA DE
GOBIERNO\ ">SECRETARIA DE
GOBIERNO</option>
<option value=\ "SECRETARIA DE
HACIENDA\ ">SECRETARIA DE
HACIENDA</option>
<option value=\ "SECRETARIA DE
INFRAESTRUCTURA\ ">SECRETARIA
DE INFRAESTRUCTURA</option>
<option value=\ "SECRETARIA DE
PLANEACION\ ">SECRETARIA DE
PLANEACION</option>
<option value=\ "SECRETARIA DE
SALUD\ ">SECRETARIA DE
SALUD</option>
<option value=\ "SERVICIOS
PUBLICOS\ ">SERVICIOS
PUBLICOS</option>
</select>
```


```
</div>
```

```
<div>
```

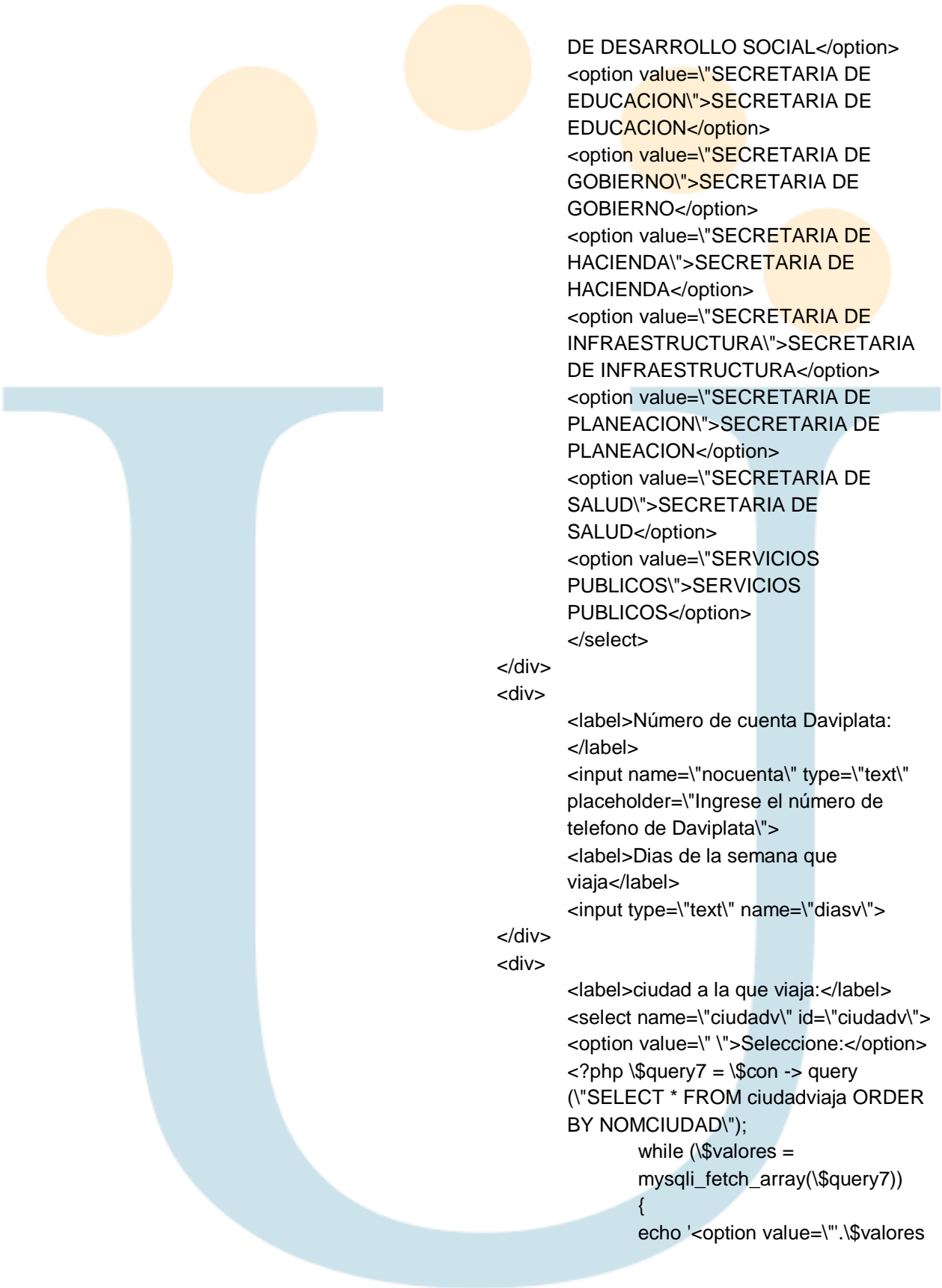
```
<label>Dependencia donde realizó
horas sociales:</label>
<select name=\ "dep2\ " id=\ "dep2\ ">
<option value=\ "\ ">Seleccione:</option>
<option value=\ "CONCEJO
MUNICIPAL\ ">CONCEJO MUNICIPAL
</option>
<option value=\ "DESPACHO DEL
ALCALDE\ ">DESPACHO DEL
```



ALCALDE</option>
<option value=\ "IMRD\ ">IMRD</option>
<option value=\ "NINGUNO\ ">NINGUNO
</option>
<option value=\ "OFICINA DE PRENSA\ ">
OFICINA DE PRENSA</option>
<option value=\ "OFICINA JURIDICA Y
DE CONTRATACION\ ">OFICINA
JURIDICA Y DE CONTRATACION
</option>
<option value=\ "PERSONERIA
MUNICIPAL\ ">PERSONERIA
MUNICIPAL</option>
<option value=\ "SECRETARIA
ADMINISTRATIVA\ ">SECRETARIA
ADMINISTRATIVA</option>
<option value=\ "SECRETARIA DE
AMBIENTE\ ">SECRETARIA DE
AMBIENTE</option>
<option value=\ "SECRETARIA DE
CULTURA Y PATRIMONIO\ ">
SECRETARIA DE CULTURA Y
PATRIMONIO</option>
<option value=\ "SECRETARIA DE
DESARROLLO ECONOMICO\ ">
SECRETARIA DE DESARROLLO
ECONOMICO</option>
<option value=\ "SECRETARIA DE
DESARROLLO SOCIAL\ ">SECRETARIA
DE DESARROLLO SOCIAL</option>
<option value=\ "SECRETARIA DE
EDUCACION\ ">SECRETARIA DE
EDUCACION</option>
<option value=\ "SECRETARIA DE
GOBIERNO\ ">SECRETARIA DE
GOBIERNO</option>
<option value=\ "SECRETARIA DE
HACIENDA\ ">SECRETARIA DE
HACIENDA</option>
<option value=\ "SECRETARIA DE
INFRAESTRUCTURA\ ">SECRETARIA
DE INFRAESTRUCTURA</option>
<option value=\ "SECRETARIA DE
PLANEACION\ ">SECRETARIA DE
PLANEACION</option>



```
<option value=\"SECRETARIA DE
SALUD\">SECRETARIA DE
SALUD</option>
<option value=\"SERVICIOS
PUBLICOS\">SERVICIOS
PUBLICOS</option>
</select>
<label>Dependencia donde realizó horas
sociales:</label>
<select name=\"dep3\" id=\"dep3\">
<option value=\" \">Seleccione:</option>
<option value=\"CONCEJO
MUNICIPAL\">CONCEJO MUNICIPAL
</option>
<option value=\"DESPACHO DEL
ALCALDE\">DESPACHO DEL
ALCALDE</option>
<option value=\"IMRD\">IMRD</option>
<option value=\"NINGUNO\">NINGUNO
</option>
<option value=\"OFICINA DE PRENSA\">
OFICINA DE PRENSA</option>
<option value=\"OFICINA JURIDICA Y
DE CONTRATACION\">OFICINA
JURIDICA Y DE CONTRATACION
</option>
<option value=\"PERSONERIA
MUNICIPAL\">PERSONERIA
MUNICIPAL</option>
<option value=\"SECRETARIA
ADMINISTRATIVA\">SECRETARIA
ADMINISTRATIVA</option>
<option value=\"SECRETARIA DE
AMBIENTE\">SECRETARIA DE
AMBIENTE</option>
<option value=\"SECRETARIA DE
CULTURA Y PATRIMONIO\">
SECRETARIA DE CULTURA Y
PATRIMONIO</option>
<option value=\"SECRETARIA DE
DESARROLLO ECONOMICO\">
SECRETARIA DE DESARROLLO
ECONOMICO</option>
<option value=\"SECRETARIA DE
DESARROLLO SOCIAL\">SECRETARIA
```



```

DE DESARROLLO SOCIAL</option>
<option value=\"SECRETARIA DE
EDUCACION\">SECRETARIA DE
EDUCACION</option>
<option value=\"SECRETARIA DE
GOBIERNO\">SECRETARIA DE
GOBIERNO</option>
<option value=\"SECRETARIA DE
HACIENDA\">SECRETARIA DE
HACIENDA</option>
<option value=\"SECRETARIA DE
INFRAESTRUCTURA\">SECRETARIA
DE INFRAESTRUCTURA</option>
<option value=\"SECRETARIA DE
PLANEACION\">SECRETARIA DE
PLANEACION</option>
<option value=\"SECRETARIA DE
SALUD\">SECRETARIA DE
SALUD</option>
<option value=\"SERVICIOS
PUBLICOS\">SERVICIOS
PUBLICOS</option>
</select>

```

```

</div>
<div>

```

```

<label>Número de cuenta Daviplata:
</label>
<input name=\"nocuenta\" type=\"text\"
placeholder=\"Ingrese el número de
telefono de Daviplata\">
<label>Dias de la semana que
viaja</label>
<input type=\"text\" name=\"dias\">

```

```

</div>
<div>

```

```

<label>ciudad a la que viaja:</label>
<select name=\"ciudad\" id=\"ciudad\">
<option value=\" \">Seleccione:</option>
<?php \$query7 = \$con -> query
(\"SELECT * FROM ciudadviaja ORDER
BY NOMCIUDAD\");
while (\$valores =
mysqli_fetch_array(\$query7))
{
echo '<option value=\"'.\$valores

```



```
[NOMCIUDAD].\">'.\$valores  
[NOMCIUDAD].\"</option>;  
}
```

```
?>  
</select>  
<label>Total Días :</label>  
<select name=\"tdias\" id=\"tdias\"  
style=\"text-align: right;\"  
onBlur=\"multiplica(this.form)\">  
<option value=\"0\">Seleccione:</option>  
<option value=\"0\">0</option>  
<option value=\"1\">1</option>  
<option value=\"2\">2</option>  
<option value=\"3\">3</option>  
<option value=\"4\">4</option>  
<option value=\"5\">5</option>  
<option value=\"6\">6</option>  
<option value=\"7\">7</option>  
</select>
```

```
</div>  
<div>
```

```
<label>Valor por 2 recorrido:</label>  
<input name=\"val2r\" type=\"text\"  
placeholder=\"Ingrese el valor del  
recorrido\" style=\"text-align: right;\">  
onBlur=\"multiplica(this.form)\">  
<label>Valor semanal:</label>  
<input type=\"text\" name=\"vals\"  
placeholder=\"Ingrese el valor semanal\"  
style=\"text-align: right;\">
```

```
</div>  
<div>
```

```
<label>Valor mensual bruto:</label>  
<input type=\"text\" name=\"valm\"  
placeholder=\"Ingrese el valor mensual  
bruto\" style=\"text-align: right;\">  
<label>Valor a pagar por mes:</label>  
<input type=\"text\" name=\"valp\"  
placeholder=\"Ingrese el valor a pagar por  
mes\" style=\"text-align: right;\">
```

```
</div>  
<div>
```

```
<label>Meses a girar:</label>  
<select name=\"mesgi\" id=\"mesgi\"  
style=\"text-align: right;\"
```

```

onBlur="multiplica(this.form)\")>
<option value="0">Seleccione:</option>
<option value="0">0</option>
<option value="1">1</option>
<option value="2">2</option>
<option value="3">3</option>
<option value="4">4</option>
<option value="5">5</option>
</select>
<label>Total semestre:</label>
<input type="text" name="tsemestre"
placeholder="Ingrese el total a pagar"
style="text-align: right;">

```

```

</div>
<div>
<label><center><h2>Ratificado:</h2>
</center></label>
</div>
<div>
<label>Ratificado:</label>
<textarea type="text" name="ratificado"
placeholder="Ingrese ratificado">
</textarea>
</div>
<div>
<label>Estado:</label>
<textarea type="text" name="estado"
placeholder="Ingrese el estado">
</textarea>
</div>
<div>
<label>observaciones:</label>
<textarea type="text"
name="observacion"
placeholder="Ingrese la
observación"></textarea>
</div>
</H3></td>

```

```

</tr>
</table>
</center>
</span>
<div class="button">
<input type="submit" value="Registrar">
</div>

```

```

        </form>
    </body>
</html>";
}
if ($reg = fopen($reg, "a"))
{
    if (fwrite($reg, "$mensa"))
    {
    }
    else
    {
        ?>
        <script type="text/javascript">
            alert("Ha habido un problema pantregistro.php");
            window.location.href='index.php';
        </script>
        <?php
    }
    fclose($reg);
}
?>

```

9.1.12 pantalla pedir documento para modificar (Cuando se crea un periodo nuevo)

```

<?php
    $modcedula = "modificacion$periodo.php";
    if(file_exists ($modcedula))
    {
    }
    else
    {
        $mens = "<?php
        include (\"blodese.php\");
    }
?>
<!doctype html>
<html>
<head>
    <title> Base de datos</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
</head>

```

```

<body>
  <form id="form1" style="width:98.72%;height:auto;background-color:white;color:blue;font-
  size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid
  #CCC;"><div><center>
  <p>Estudiantes $periodo</p>
  <div style="float: left;"><?php echo \$_SESSION["usuarios"]?></div>
  <div style = "float: right;"><a href="salir.php">Salir</a></div>
  <p>&nbsp;</p>
  </center></div>
  </form>
  <form id="form2"method='POST' action='index.php' margin='0' style="float: left;
  width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
  Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <?php
      include ("menuopciones.php");
    ?>
  </form>
  <form id="form3"method='POST' action='index.php' margin='0' style="float: left; width:82.62%;
  height:5.2%; background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-
  align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <div style="float: left;"><a href="$periodo.php">atras</a></div>
  </form>
  <form id="form4"method='GET' action='$periodo.php' margin='0' style="float: left; width:82.62%;
  height:5.4%; background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-
  align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <center>
      <p></p>
      <label>Número de documento:</label>
      <input type="text" name="documento" placeholder="Ingrese el
      documento" style="width: 240px" required>
      <div>
        <input name="Modificar" type="submit" id="enviar2"
        formaction="modificar$periodo.php" value="Actualizar" style="width:
        150px">
      </div>
      <p></p>
    </center>
  </form>
  <form for="form5" action='index.php' method='POST' margin='0' style="float: left; width:82.62%;
  height:195.4%; background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-
  align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
</form>
</body>
</html>;
}

```

```

if ($modcedula = fopen($modcedula, "a"))
{
    if (fwrite($modcedula, "$mens"))
    {
    }
    else
    {
        ?>
        <script type="text/javascript">
            alert("Ha habido un problema documentomodificar.php");
            window.location.href='index.php';
        </script>
        <?php
    }
    fclose($modcedula);
}
?>

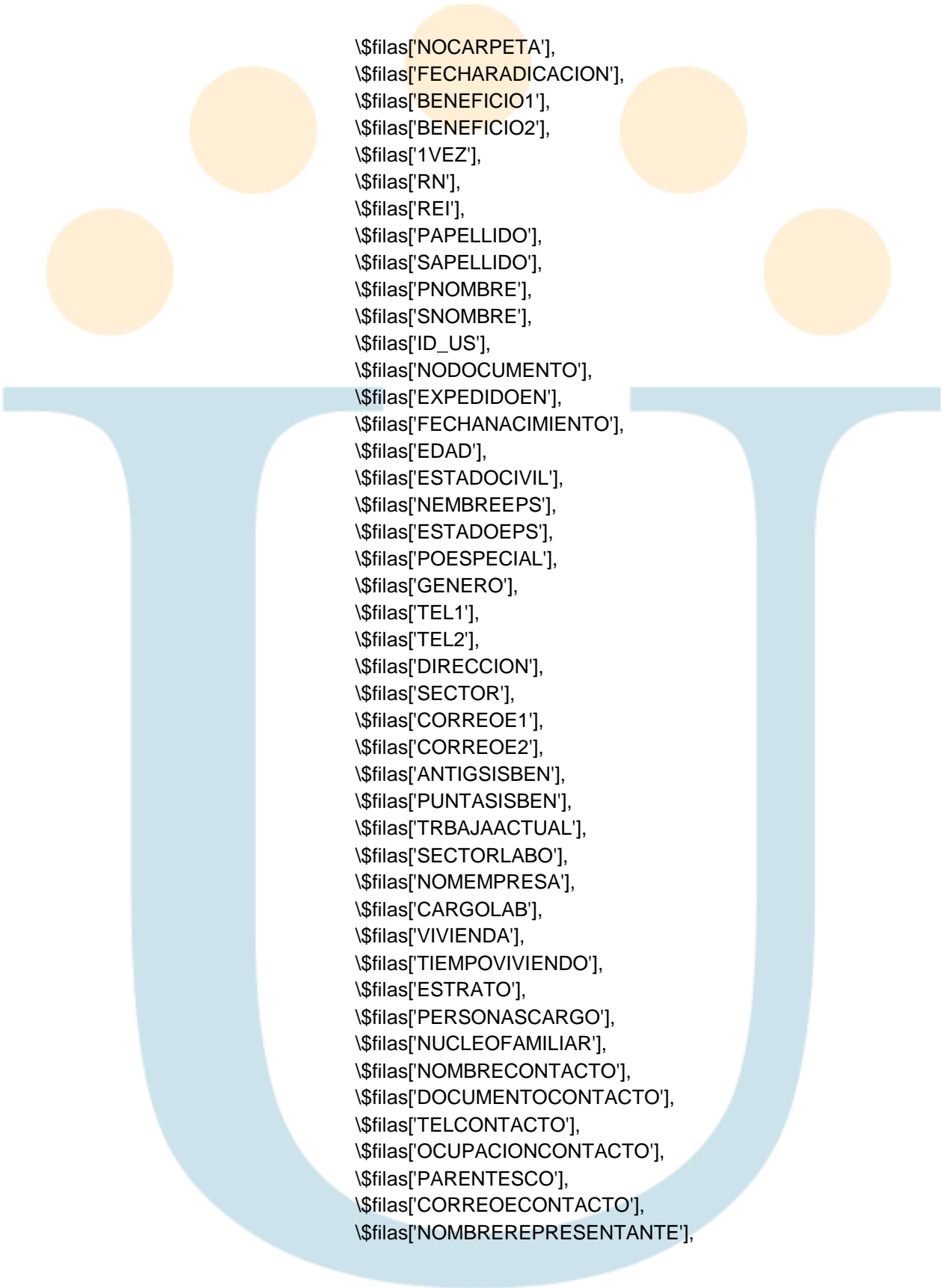
```

9.1.13 Pantalla para modificar datos de usuario (Cuando se crea un periodo nuevo)


```

<?php
    $bene = "modificar$periodo.php";
    if(file_exists ($bene))
    {
    }
    else
    {
        $men = "<?php
        include (\\"blodese.php\\");
        include (\\"conexion.php\\");
        \\$consulta = consultarregistro(\\$_GET['documento']);
        function consultarregistro(\\$NODOCUMENTO)
        {
            include (\\"conexion.php\\");
            \\$usuario = mysqli_query(\\$con , \\\"SELECT * FROM `\$tabla` WHERE `\$tabla` .
            NODOCUMENTO = \\$NODOCUMENTO\\");
            if (mysqli_num_rows(\\$usuario) > 0)
            {
                \\$filas=mysqli_fetch_assoc(\\$usuario);
                return
                [
                    \\$filas['ID_SUB'],
                    \\$filas['ITEM'],

```



\\$filas['NOCARPETA'],
\\$filas['FECHARADICACION'],
\\$filas['BENEFICIO1'],
\\$filas['BENEFICIO2'],
\\$filas['1VEZ'],
\\$filas['RN'],
\\$filas['RE1'],
\\$filas['PAPELLIDO'],
\\$filas['SAPELLIDO'],
\\$filas['PNOMBRE'],
\\$filas['SNOMBRE'],
\\$filas['ID_US'],
\\$filas['NODOCUMENTO'],
\\$filas['EXPEDIDOEN'],
\\$filas['FECHANACIMIENTO'],
\\$filas['EDAD'],
\\$filas['ESTADOCIVIL'],
\\$filas['NEMBREEPS'],
\\$filas['ESTADOEPS'],
\\$filas['POESPECIAL'],
\\$filas['GENERO'],
\\$filas['TEL1'],
\\$filas['TEL2'],
\\$filas['DIRECCION'],
\\$filas['SECTOR'],
\\$filas['CORREOE1'],
\\$filas['CORREOE2'],
\\$filas['ANTIGSISBEN'],
\\$filas['PUNTASISBEN'],
\\$filas['TRBAJAACTUAL'],
\\$filas['SECTORLABO'],
\\$filas['NOMEMPRESA'],
\\$filas['CARGOLAB'],
\\$filas['VIVIENDA'],
\\$filas['TIEMPOVIVIENDO'],
\\$filas['ESTRATO'],
\\$filas['PERSONASCARGO'],
\\$filas['NUCLEOFAMILIAR'],
\\$filas['NOMBRECONTACTO'],
\\$filas['DOCUMENTOCONTACTO'],
\\$filas['TELCONTACTO'],
\\$filas['OCUPACIONCONTACTO'],
\\$filas['PARENTESCO'],
\\$filas['CORREOECONTACTO'],
\\$filas['NOMBREREPRESENTANTE'],



\\$filas['NOCEDULAR'],
\\$filas['EXPEDIDAR'],
\\$filas['TELREPR'],
\\$filas['DIRECCIONREPR'],
\\$filas['PARENTESCOREP'],
\\$filas['INSTITUCIONEDUMEDIA'],
\\$filas['FECHAGRADO'],
\\$filas['PUNTAJEICFES'],
\\$filas['ANOPRESENTACION'],
\\$filas['PROMEDIOONCE'],
\\$filas['PUESTOONCE'],
\\$filas['ROTOBENE'],
\\$filas['UNIVERSIDAD'],
\\$filas['SEDE'],
\\$filas['NIVEL'],
\\$filas['CARRERA'],
\\$filas['MODALIDAD'],
\\$filas['SEMESTRE'],
\\$filas['HORARIO'],
\\$filas['SNIES'],
\\$filas['CONTINUACARRERA'],
\\$filas['VALORSEMESTRE'],
\\$filas['PORCENTAJEA'],
\\$filas['MONTOAPRO'],
\\$filas['OTROBENE'],
\\$filas['MATERIASPER'],
\\$filas['PROMEDIOANTERIOR'],
\\$filas['PROMEDIOTRAANTERIOR'],
\\$filas['BENEFICIOSANTERIORES1'],
\\$filas['BENEFICIOSANTERIORES2'],
\\$filas['HSCUMPLIDAS'],
\\$filas['DEPENDENCIA1'],
\\$filas['DEPENDENCIA2'],
\\$filas['DEPENDENCIA3'],
\\$filas['NOCUENTA'],
\\$filas['DIASVIAJADOS'],
\\$filas['CIUDADVIAJA'],
\\$filas['DIAS'],
\\$filas['VALORRECOR'],
\\$filas['VALORSEMANA'],
\\$filas['VALORMESBRUTO'],
\\$filas['VALORMESPAGAR'],
\\$filas['MESES'],
\\$filas['TOTALSEMESTRE'],
\\$filas['RATIFICADO'],

```

        \${filas['APROBADO']},
        \${filas['OBSERVACIONRAT']}
    ];
}
else
{
    ?>
    <script type="text/javascript">
        alert("\Usuario No existente");
        window.location.href='modificacion$periodo.php';
    </script>
    <?php
}
}
?>
<!doctype html>
<html>
    <head>
        <title>Datos</title>
        <link rel="stylesheet" type="text/css" href="estilos.css">
        <script language="javascript">
            function multiplica(form)
            {
                var resultado;
                var resultado2;
                var resultado3;
                var montoaprobado;
                var dias=0;
                var valor=0;
                var mes=0;
                var mespago=0;
                var valsemestre=0;
                var porcentajes=0;
                var universidad=0;
                universidad = parseInt (form.univer.value);
                valsemestre = parseInt (form.vsemestre.value);
                porcentajes = parseInt (form.porcentaje.value);
                dias = parseInt (form.tdias.value);
                valor = parseInt (form.val2r.value);
                mes = parseInt (form.mesgi.value);
                mespago = parseInt (form.valp.value);
                montoaprobado = (valsemestre * porcentajes)/100;
                form.monto.value = montoaprobado;
                resultado = dias * valor;
                form.vals.value=resultado;
            }
        </script>
    </head>
    <body>

```



```

resultado2 = 4 * resultado;
form.valm.value=resultado2;

if (resultado2 < 156246)
    {
        form.valp.value = resultado2;
    }
else
    {
        form.valp.value = 156246;
    }

```

```

resultado3= mespago * mes;
form.tsemestre.value=resultado3;
}

```

```

</script>
</head>
<body>
  <form>
    <spam class="text">
      <div style = "float: left;"></div>
      <span style="float: left;"><?php echo \$_SESSION["usuarios"];?></span>
      <div style="float: right;" align="right"><a href="salir.php" align = "right">
      Salir</a></div></spam>
    </form>
    <form>
      <spam class="text">
        <p><a href="modificacion$periodo.php">Anterior</a></p>
        <H2><center>Ingrese los datos:</center></H2>
        Los espacios marcados con asterisco (*) son obligatorios
      </spam>
    </form>
  <div></div>
  <form id="form1" action="modifica$periodo.php" method="POST">
    <span class="text">
      <center>
        <table>
          <tr>
            <td><H3>
              <div>
                <label><center><h1>Datos personales:
                </h1></center></label>
              </div>
            </td>
          </tr>
        </table>
      </center>
    </span>
  </form>

```

```

</div>
<label>ID:</label>
<input type="text" name="idsu"
placeholder="Digite la Id.\"
value="<?php
echo \$consulta[0]?>">
</div>
<label>Item:</label>
<input type="text" name="item"
placeholder="Digite el item\"
value="<?php echo \$consulta[1]?>">
</div>
<div>

```

```

<label>N° carpeta *:</label>
<input type="text" name="carpeta"
placeholder="Digite el numero de la
carpeta" value="<?php echo
\$consulta[2]?>">
<label>Fecha de radicacion *:</label>
<input name="radica" type="date"
required="required" id="radica"
title="Radicacion" value="<?php echo
\$consulta[3]?>">
</div>
<div>

```

```

<label>Beneficio 1:</label>
<select name="bene1" id="bene1" >
<option value="<?php echo
\$consulta[4]?>"><?php echo
\$consulta[4]?></option>
<option value="">Seleccione:</option>
<?php \$query = \$conn -> query
("SELECT * FROM beneficio1 ORDER
BY NOMBENEFICIO1 ASC");
while (\$valores =
mysqli_fetch_array(\$query))
{
echo '<option value="".'.\$valores
[NOMBENEFICIO1].'\>'.\$valores
[NOMBENEFICIO1].'\</option>';
}
?>
</select>
<label>Beneficio 2:</label>
<select name="bene2" id="bene2">

```

```

<option value="\ "><?php echo
\$consulta[5]?>\ "><?php echo
\$consulta[5]?></option>
<option value="\ ">Seleccione:</option>
<?php \$query1 = \$conn -> query
("\SELECT * FROM beneficio2 ORDER
BY NOMBENEFICIO2 ASC\");
    while (\$valores =
        mysqli_fetch_array(\$query1))
    {
        echo '<option value="\ ".\$valores
[NOMBENEFICIO2].'\ ">'.\$valores
[NOMBENEFICIO2].'\ "></option>';
    }
}

```

```

?>
</select>

```

```

</div>
<div>

```

```

<label>1° Vez:</label>
<input type="text" name="1vez"
placeholder=" " value="\ "><?php echo
\$consulta[6]?>\ ">
<label>Renovacion:</label>
<input type="text" name="rn"
placeholder=" " value="\ "><?php echo
\$consulta[7]?>\ ">

```

```

</div>
<div>

```

```

<label>Reintegro:</label>
<input type="text" name="rei"
placeholder=" " value="\ "><?php echo
\$consulta[8]?>\ ">

```

```

</div>
<div>

```

```

<label>Primer apellido *:</label>
<input name="1apellido" type="text"
required="required"
placeholder="Ingrese primer apellido"
value="\ "><?php echo \$consulta[9]?>\ ">
<label>Segundo apellido *:</label>
<input name="2apellido" type="text"
required="required"
placeholder="Ingrese primer apellido"
value="\ "><?php echo \$consulta[10]?>\ ">

```

```

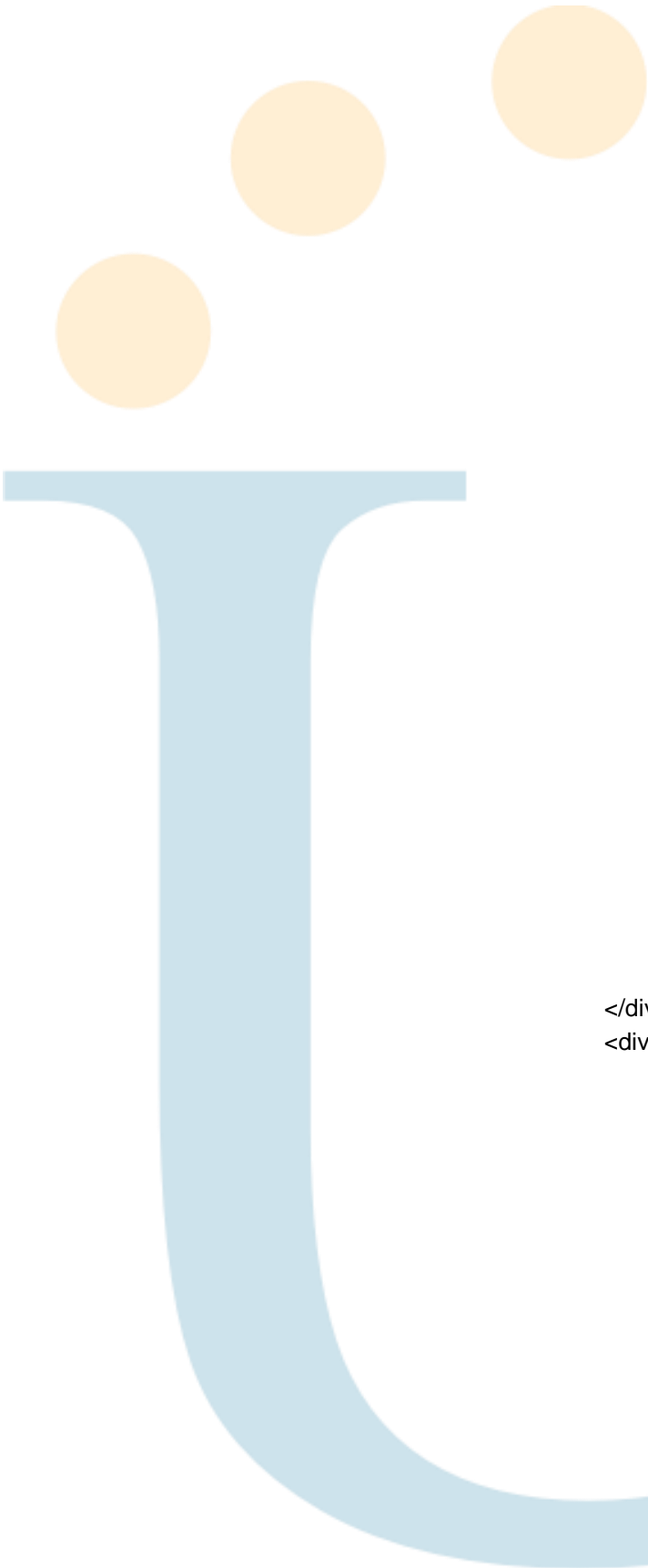
</div>

```

```

<div>
<label>Primer nombre *:</label>
<input name="\1nombre" type="text"
required="\required"
placeholder="\Ingrese primer nombre"
value="\<?php echo \$consulta[11]?>"\>
<label>Segundo nombre *:</label>
<input name="\2nombre" type="text"
required="\required"
placeholder="\Ingrese segundo nombre"
value="\<?php echo \$consulta[12]?>"\>
</div>
<div>
<label>Tipo de documento *:</label>
<select name="\tdocu"
required="\required" id="\tdocu"
placeholder="\Seleccione el tipo de
documento"\>
<option value="\<?php echo
\$consulta[13]?>"\><?php echo
\$consulta[13]?></option>
<option value="\ " ">Seleccione:</option>
<option value="\C.C.\ ">C.C.</option>
<option value="\T.I.\ ">T.I.</option>
</select>
<label>Numero de documento *:</label>
<input name="\ndocumento" type="text"
required="\required"
placeholder="\Ingrese numero de
documento" value="\<?php echo
\$consulta[14]?>"\>
</div>
<div>
<label>Expedido en:</label>
<input type="text" name="\expedido"
placeholder="\Ingrese el lugar de
expedicion" value ="\<?php echo
\$consulta[15]?>"\>
<label>Fecha de nacimiento:</label>
<input name="\nacimiento" type="date"
id="\nacimiento" title="\nacimiento"
value="\<?php echo \$consulta[16]?>"\>
</div>
<div>
<label>Edad *:</label>

```



```
<select name =\"edadus\"
required= \"required\">
<option value= \"<?php echo
\$consulta[17]?>\"><?php echo
\$consulta[17]?></option>
<option value= \" \">Seleccione:</option>
<option value= \"1\">1</option>
<option value= \"2\">2</option>
<option value= \"3\">3</option>
<option value= \"...\">...</option>
<option value= \"100\">100</option>
</select>
<label>Estado civil:</label>
<select name= \"estadoc\" id= \"estadoc\">
<option value= \"<?php echo
\$consulta[18]?>\"><?php echo
\$consulta[18]?></option>
<option value= \" \">Seleccione:</option>
<option value= \"DIVORCIADO(A)\">
DIVORCIADO(A)</option>
<option value= \"CASADO(A)\">
CASADO(A)</option>
<option value= \"SOLTERO(A)\">
SOLTERO(A)</option>
<option value= \"UNION LIBRE\">UNION
LIBRE</option>
<option value= \"VIUDO(A)\">VIUDO(A)
</option>
</select>
```

```
</div>
<div>
```

```
<label>Nombre de EPS:</label>
<input name= \"eps\" type= \"text\"
placeholder= \"Ingrese el nombre de la
EPS\" value= \"<?php echo
\$consulta[19]?>\">
<label>Estado EPS:</label>
<select name= \"estadoeps\"
id= \"estadoeps\">
<option value= \"<?php echo
\$consulta[20]?>\"><?php echo
\$consulta[20]?></option>
<option value= \" \">Seleccione:</option>
<option value= \"BENEFICIARIO(A)\">
BENEFICIARIO(A)</option>
```

```

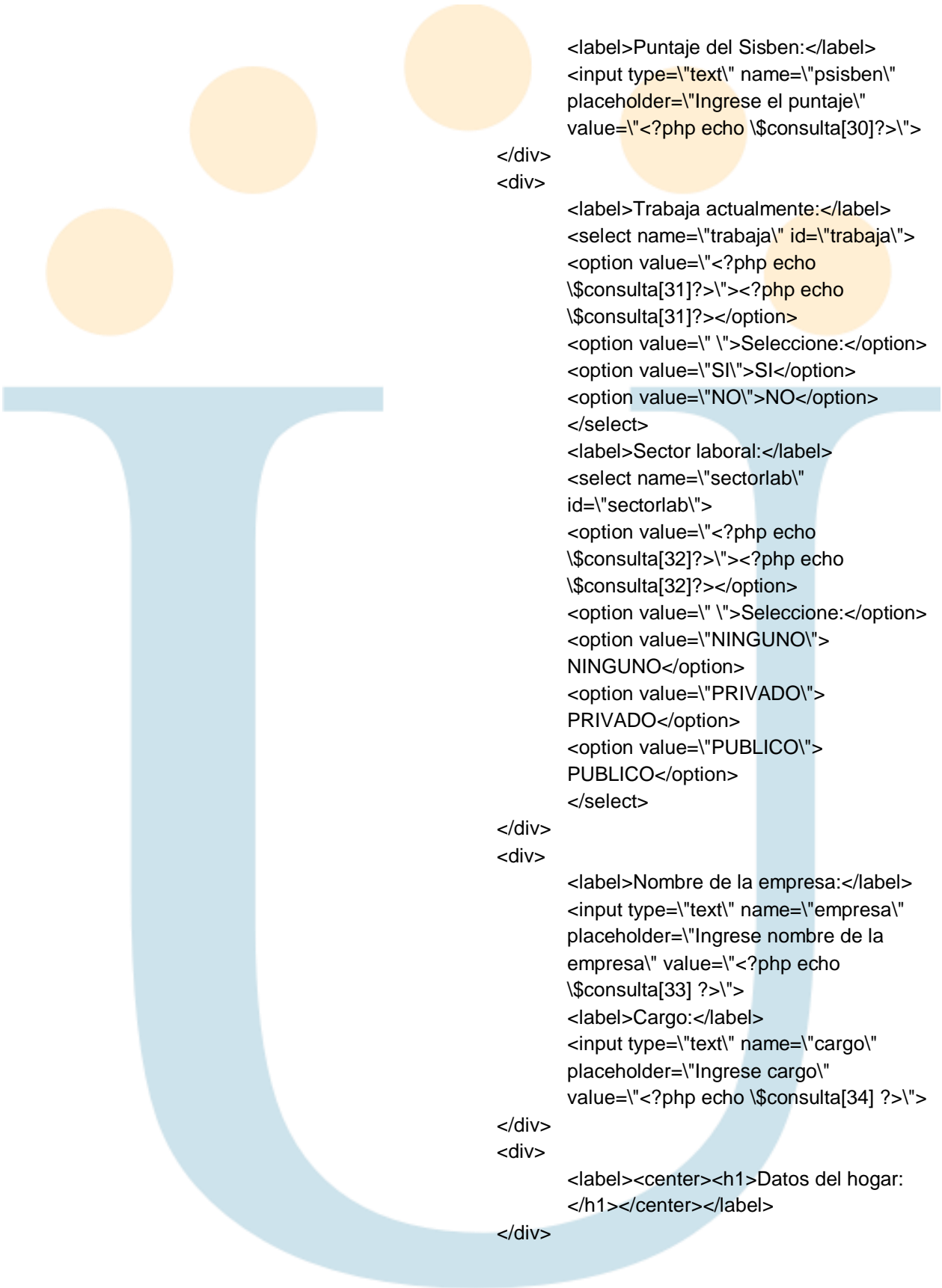
<option value="COTIZANTE">
COTIZANTE</option>
<option value="SUBSIDIARIO">
SUBSIDIARIO</option>
</select>
</div>
<div>
<label>Poblacion especial:</label>
<select name="pespe" id="pespe">
<option value=""><?php echo
\$consulta[21]?>\"><?php echo
\$consulta[21]?></option>
<option value="">Seleccione:</option>
<?php \$query2 = \$conn -> query
("SELECT * FROM poblacion ORDER
BY NOMPOBLACION ASC");
while (\$valores =
mysqli_fetch_array(\$query2))
{
echo '<option value="".'.\$valores
[NOMPOBLACION].'\>'.\$valores
[NOMPOBLACION].'\</option>';
}
?>
</select>
<label>Genero *:</label>
<select name="gene"
required="required" id="gene">
<option value=""><?php echo
\$consulta[22]?>\"><?php echo
\$consulta[22]?></option>
<option value="">Seleccione:</option>
<option value="FEMENINO">
FEMENINO</option>
<option value="MASCULINO">
MASCULINO</option>
<option value="OTRO">OTRO</option>
</select>
</div>
<div>
<label>Telefono 1 *:</label>
<input name="tel1" type="text"
required="required"
placeholder="Ingrese numero telefonico"
value=""><?php echo \$consulta[23]?>\">

```

```

<label>Telefono 2:</label>
<input type="text" name="tel2"
placeholder="Ingrese otro numero
telefonico" value="<?php echo
\$consulta[24]?>">
</div>
<div>
<label>Direccion:</label>
<input type="text" name="direccion"
placeholder="Ingrese direccion"
value="<?php echo \$consulta[25]?>">
<label>Sector:</label>
<input type="text" name="sector"
placeholder="Ingrese el sector"
value="<?php echo \$consulta[26] ?>">
</div>
<div>
<label>Correo electronico 1:</label>
<input type="text" name="correo"
placeholder="Ingrese correo"
value="<?php echo \$consulta[27]?>">
<label>Correo electronico 2:</label>
<input type="text" name="correo2"
placeholder="Ingrese otro correo"
value="<?php echo \$consulta[28] ?>">
</div>
<div>
<label>Antigüedad Sisben en años:
</label>
<select name="asisben" id="asisben">
<option value="<?php echo
\$consulta[29]?>"><?php echo
\$consulta[29]?></option>
<option value="">Seleccione:</option>
<?php \$query3 = \$conn -> query
("SELECT * FROM antigu_sisben");
while (\$valores =
mysqli_fetch_array(\$query3))
{
echo '<option value="".'.\$valores
[NOMASISBEN].'\>'.\$valores
[NOMASISBEN].'\</option>';
}
?>
</select>

```



```
<label>Puntaje del Sisben:</label>
<input type=\"text\" name=\"psisben\"
placeholder=\"Ingrese el puntaje\"
value=\"<?php echo \$consulta[30]?>\">
```

```
</div>
<div>
```

```
<label>Trabaja actualmente:</label>
<select name=\"trabaja\" id=\"trabaja\">
<option value=\"<?php echo
\$consulta[31]?>\"><?php echo
\$consulta[31]?></option>
<option value=\" \">Seleccione:</option>
<option value=\"SI\">SI</option>
<option value=\"NO\">NO</option>
</select>
```

```
<label>Sector laboral:</label>
<select name=\"sectorlab\"
id=\"sectorlab\">
<option value=\"<?php echo
\$consulta[32]?>\"><?php echo
\$consulta[32]?></option>
<option value=\" \">Seleccione:</option>
<option value=\"NINGUNO\">
NINGUNO</option>
<option value=\"PRIVADO\">
PRIVADO</option>
<option value=\"PUBLICO\">
PUBLICO</option>
</select>
```

```
</div>
<div>
```

```
<label>Nombre de la empresa:</label>
<input type=\"text\" name=\"empresa\"
placeholder=\"Ingrese nombre de la
empresa\" value=\"<?php echo
\$consulta[33] ?>\">
<label>Cargo:</label>
<input type=\"text\" name=\"cargo\"
placeholder=\"Ingrese cargo\"
value=\"<?php echo \$consulta[34] ?>\">
```

```
</div>
<div>
```

```
<label><center><h1>Datos del hogar:
</h1></center></label>
```

```
</div>
```


<div>

```
<label>Vive en vivienda:</label>
<select name="vivienda" id="vivienda">
<option value=""><?php echo
\$consulta[35]?>\<>?php echo
\$consulta[35]?></option>
<option value="" \>Seleccione:</option>
<option value="ARRENDADA">
ARRENDADA</option>
<option value="FAMILIAR">
FAMILIAR</option>
<option value="PROPIA">
PROPIA</option>
</select>
```

```
<label>Hace cuanto vive en ese lugar:
</label>
<select name="vive" id="vive">
<option value=""><?php echo
\$consulta[36]?>\<>?php echo
\$consulta[36]?></option>
<option value="" \>Seleccione:</option>
<option value="1">1</option>
<option value="2">2</option>
<option value="3">3</option>
<option value="...">...</option>
<option value="60">60</option>
</select>
```

</div>

<div>

```
<label>Estrato:</label>
<select name="estrato" id="estrato">
<option value=""><?php echo
\$consulta[37]?>\<>?php echo
\$consulta[37]?></option>
<option value="" \>Seleccione:</option>
<option value="1">1</option>
<option value="2">2</option>
<option value="3">3</option>
<option value="4">4</option>
<option value="5">5</option>
<option value="6">6</option>
</select>
```

```
<label>Personas a cargo:</label>
<select name="personasacar"
id="personasacar">
```

```

<option value="\<?php echo
\$consulta[38]?>\"><?php echo
\$consulta[38]?></option>
<option value="\ \">Seleccione:</option>
<option value=\"0\">0</option>
<option value=\"1\">1</option>
<option value=\"2\">2</option>
<option value=\"3\">3</option>
<option value=\"4\">4</option>
<option value=\"5\">5</option>
<option value=\"6\">6</option>
<option value=\"7\">7</option>
<option value=\"8\">8</option>
<option value=\"9\">9</option>
<option value=\"10\">10</option>
</select>

```

```

</div>
<div>

```

```

<label>Cuántas personas viven en tu
nucleo familiar:</label>
<select name=\"nucleo\" id=\"nucleo\">
<option value=\"<?php echo
\$consulta[39]?>\"><?php echo
\$consulta[39]?>
<option value="\ \">Seleccione:</option>
<option value=\"0\">0</option>
<option value=\"1\">1</option>
<option value=\"2\">2</option>
<option value=\"3\">3</option>
<option value=\"...\">...</option>
<option value=\"20\">20</option>
</select>

```

```

</div>
<div>

```

```

<label><center><h1>Contacto de
emergencia:</h1></center></label>

```

```


</div>
<div>

```

```

<label>Nombre de contacto:</label>
<input type=\"text\" name=\"nombrecon\"
placeholder=\"Ingrese nombre del
contacto\" value=\"<?php echo
\$consulta[40]?>\">
<label>Cedula de Ciudadania:</label>
<input type=\"text\" name=\"cc\"

```



```
placeholder="Cedula del contacto"
value="\<?php echo \$consulta[41]?>"
</div>
<div>
```

```
<label>Telefono:</label>
<input type="text" name="telcon"
placeholder="Telefono" value="\<?php
echo \$consulta[42]?>">
<label>Ocupacion:</label>
<input type="text" name="ocupacion"
placeholder="Ocupacion del contacto"
value="\<?php echo \$consulta[43]?>">
</div>
<div>
```

```
<label>Parentesco:</label>
<input type="text" name="parentesco"
placeholder="Parentesco"
value="\<?php echo \$consulta[44]?>">
<label>Correo electronico:</label>
<input type="text" name="correocon"
placeholder="Correo" value="\<?php
echo \$consulta[45]?>">
```

```
</div>
<div>
```

```
<label><center><h1>Representante
Legal:</h1></center></label>
```

```
</div>
<div>
```

```
<label>Nombres y apellidos completos:
</label>
<input type="text" name="nombrepr"
placeholder="Ingrese nombre del
representante" value="\<?php echo
\$consulta[46]?>">
<label>Cedula de Ciudadania:</label>
<input type="text" name="ccrep"
placeholder="Cedula del representante"
value="\<?php echo \$consulta[47]?>">
```

```
</div>
<div>
```

```
<label>Expedido en:</label>
<input type="text" name="expedidorep"
placeholder="Ingrese el lugar de
expedición" value="\<?php echo
```

```

    \${consulta[48]}?>\">
    <label>Teléfono:</label>
    <input type=\"text\" name=\"Telrep\"
    placeholder=\"Ingrese el número de
    teléfono\" value=\"<?php echo
    \${consulta[49]}?>\">
</div>
<div>
    <label>Dirección:</label>
    <input type=\"text\" name=\"Dirrep\"
    placeholder=\"Ingrese dirección de
    contacto\" value=\"<?php echo
    \${consulta[50]}?>\">
    <label>Parentesco:</label>
    <input type=\"text\"
    name=\"Parentesco\"
    placeholder=\"Parentesco\"
    value=\"<?php echo \${consulta[51]}?>\">
</div>
<label><center><h1>Educación media:
</h1></center></label>
</div>
<div>
    <label>Institución:</label>
    <input type=\"text\" name=\"institucion\"
    placeholder=\"Nombre de la institución\"
    value=\"<?php echo \${consulta[52]}?>\">
    <label>Fecha de graduación:</label>
    <input name=\"grado\" type=\"date\"
    id=\"grado\" title=\"grado\" value=\"<?php
    echo \${consulta[53]}?>\">
</div>
<div>
    <label>Puntaje Icfes:</label>
    <input type=\"text\" name=\"icfes\"
    placeholder=\"Puntaje\" value=\"<?php
    echo \${consulta[54]}?>\">
    <label>Año de Presentación</label>
    <input type=\"text\"
    name=\"presentaicfes\"
    placeholder=\"Año de presentación\"
    value=\"<?php echo \${consulta[55]}?>\">
</div>
<div>
    <label>Promedio de notas de grado

```

```

Once:</label>
<input type=\"text\" name=\"notas11\"
placeholder=\"promedio\" value=\"<?php
echo \$consulta[56]?>\">
<label>Puesto en el grado Once:</label>
<input type=\"text\" name=\"puesto11\"
placeholder=\"Puesto grado 11\"
value=\"<?php echo \$consulta[57]?>\">
</div>
<div>
<label><center><h1></h1></center>
</label>
</div>
<div>
<label>Recibe algun otro beneficio:
</label>
<select name=\"robene\" id=\"robene\">
<option value=\"<?php echo
\$consulta[58]?>\"><?php echo
\$consulta[58]?></option>
<option value=\" \">Seleccione:</option>
<?php \$query4 = \$conn -> query
(\"SELECT * FROM algunotro ORDER BY
NOMBREALGUN ASC\");
while (\$valores =
mysqli_fetch_array(\$query4))
{
echo '<option value=\"\".\$valores
[NOMBREALGUN].'\>'.\$valores
[NOMBREALGUN].'\</option>';
}
?>
</select>
</div>
<div>
<label><center><h1>Información de la
universidad:</h1></center></label>
</div>
<div>
<label>Universidad *:</label>
<select name=\"univer\"
required=\"required\" id=\"univer\">
<option value=\"<?php echo
\$consulta[59]?>\"><?php echo
\$consulta[59]?></option>

```

```

<option value="" \>Seleccione:</option>
<?php \$query5 = \$conn -> query
("SELECT * FROM universidad ORDER
BY NOMUNIVERSIDAD ASC");
    while (\$valores =
        mysqli_fetch_array(\$query5))
    {
        echo '<option value="" .\$valores
[NOMUNIVERSIDAD].">'.
        \$valores[NOMUNIVERSIDAD].'
        </option>';
    }
?>

```

```

</select>
<label>Sede:</label>
<input type="text" name="sede"
placeholder="Ingrese sede"
value=""<?php echo \$consulta[60]?>">
</div>
<div>
<label>Nivel *:</label>
<select name="niv" required="required"
id="niv">
<option value=""<?php echo
\$consulta[61]?>"><?php echo
\$consulta[61]?></option>
<option value="" \>Seleccione:</option>
<option value="PREGRADO">
PREGRADO</option>
<option value="POSTGRADO">
POSTGRADO</option>
<option value="TECNICO">
TECNICO</option>
<option value="TECNOLOGO">
TECNOLOGO</option>
</select>
<label>Carrera *:</label>
<select name="carrera"
required="required" id="carrera">
<option value=""<?php echo
\$consulta[62]?>"><?php echo
\$consulta[62]?></option>
<option value="" \>Seleccione:</option>
<?php \$query6 = \$conn -> query
("SELECT * FROM carreras ORDER BY

```

```

NOMCARRERA ASC\");
while (\$valores =
mysql_fetch_array(\$query6))
{
echo '<option value=\'.\$valores
[NOMCARRERA].\'>'.\$valores
[NOMCARRERA].\'</option>';
}
?>
</select>

```

```

</div>
<div>

```

```

<label>Modalidad *:</label>
<select name="modalidad"
required="required" id="modalidad">
<option value="\<?php echo
\$consulta[63]?>\"><?php echo
\$consulta[63]?></option>
<option value="\ " >Seleccione:</option>
<option value="DISTANCIA">
DISTANCIA</option>
<option value="PRESENCIAL">
PRESENCIAL</option>
<option value="SEMIPRESENCIAL">
SEMIPRESENCIAL</option>
<option value="VIRTUAL">
VIRTUAL</option>
</select>
<label>Semestre *:</label>
<select name="\semestre"
required="required" id="semestre">
<option value="\<?php echo
\$consulta[64]?>\"><?php echo
\$consulta[64]?></option>
<option value="\ " >Seleccione:</option>
<option value="I">I</option>
<option value="II">II</option>
<option value="III">III</option>
<option value="IV">IV</option>
<option value="...">...</option>
<option value="XX">XX</option>
</select>

```

```

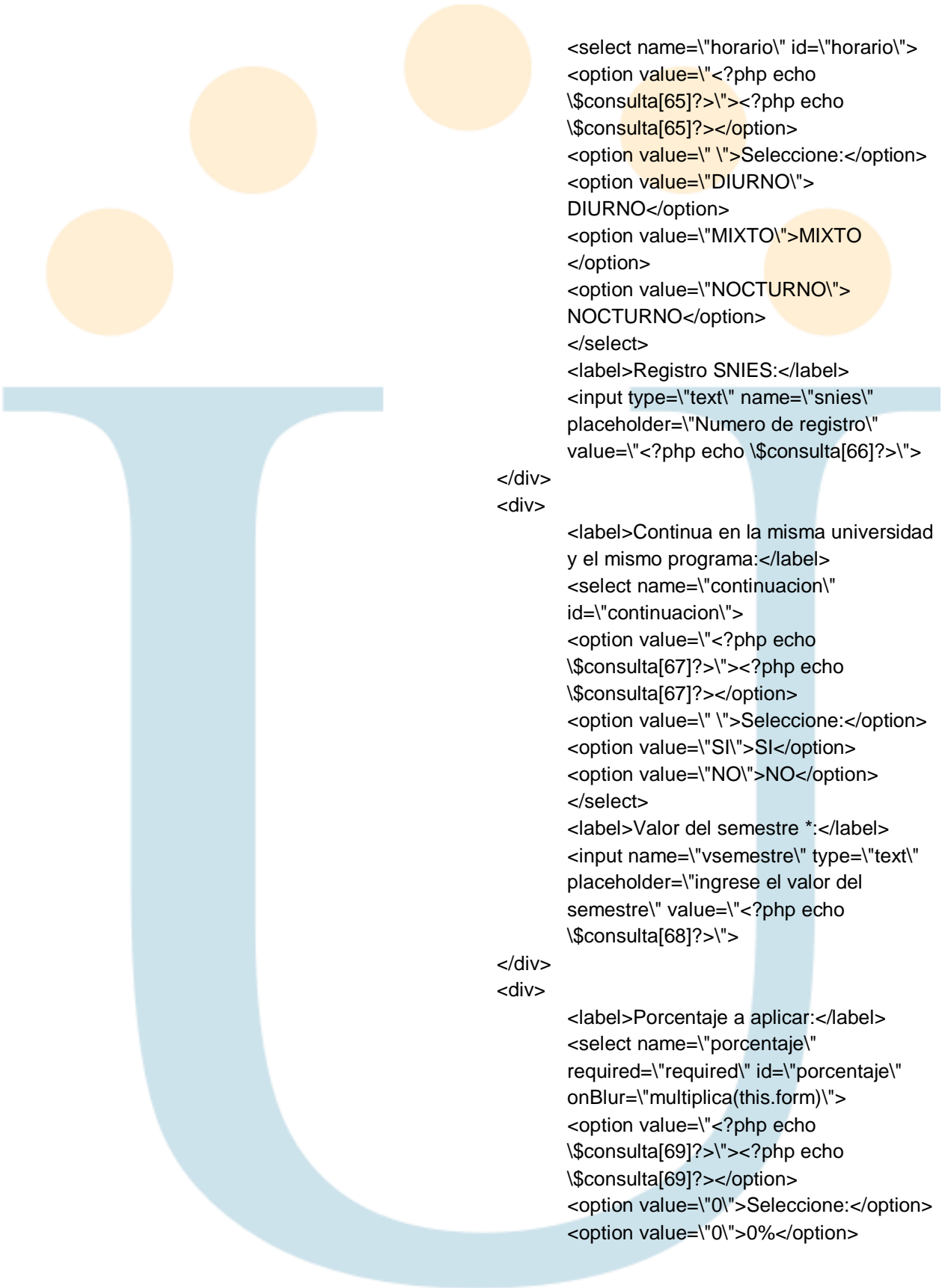
</div>
<div>

```

```

<label>Horario:</label>

```



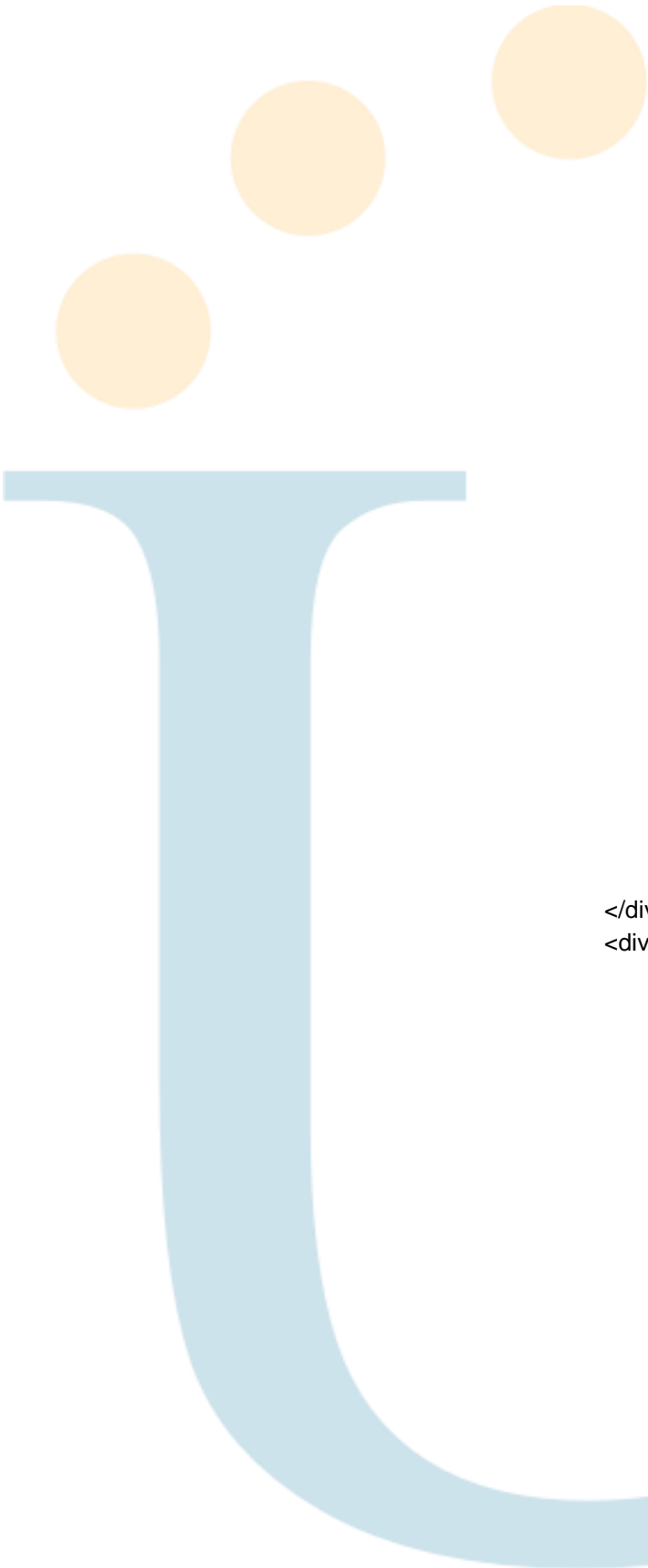
```
<select name="horario" id="horario">
<option value=""><?php echo
\$consulta[65]?>\"><?php echo
\$consulta[65]?></option>
<option value="">Seleccione:</option>
<option value="DIURNO">
DIURNO</option>
<option value="MIXTO">MIXTO
</option>
<option value="NOCTURNO">
NOCTURNO</option>
</select>
<label>Registro SNIES:</label>
<input type="text" name="snies"
placeholder="Numero de registro"
value=""><?php echo \$consulta[66]?>\">
```

```
</div>
<div>
```

```
<label>Continua en la misma universidad
y el mismo programa:</label>
<select name="continuacion"
id="continuacion">
<option value=""><?php echo
\$consulta[67]?>\"><?php echo
\$consulta[67]?></option>
<option value="">Seleccione:</option>
<option value="SI">SI</option>
<option value="NO">NO</option>
</select>
<label>Valor del semestre *:</label>
<input name="vsemestre" type="text"
placeholder="ingrese el valor del
semestre" value=""><?php echo
\$consulta[68]?>\">
```

```
</div>
<div>
```

```
<label>Porcentaje a aplicar:</label>
<select name="porcentaje"
required="required" id="porcentaje"
onBlur="multiplica(this.form)">
<option value=""><?php echo
\$consulta[69]?>\"><?php echo
\$consulta[69]?></option>
<option value="0">Seleccione:</option>
<option value="0">0%</option>
```

```
<option value=\ "5\ ">5%</option>
<option value=\ "10\ ">10%</option>
<option value=\ "15\ ">15%</option>
<option value=\ "20\ ">20%</option>
<option value=\ "25\ ">25%</option>
<option value=\ "30\ ">30%</option>
<option value=\ "35\ ">35%</option>
<option value=\ "40\ ">40%</option>
<option value=\ "45\ ">45%</option>
<option value=\ "50\ ">50%</option>
<option value=\ "55\ ">55%</option>
<option value=\ "60\ ">60%</option>
<option value=\ "65\ ">65%</option>
<option value=\ "70\ ">70%</option>
<option value=\ "75\ ">75%</option>
<option value=\ "80\ ">80%</option>
<option value=\ "85\ ">85%</option>
<option value=\ "90\ ">90%</option>
<option value=\ "95\ ">95%</option>
<option value=\ "100\ ">100%</option>
</select>
```

```
<label>Monto aprobado:</label>
<input type=\ "text\ " name=\ "monto\ "
placeholder=\ "ingrese el monto
aprobado\ " value=\ "<?php echo
\$consulta[70]?>\ ">
```

```
</div>
```

```
<div>
```

```
<label>Otro beneficio:</label>
<input type=\ "text\ " name=\ "obeneficio\ "
placeholder=\ "ingrese otros beneficios\ "
value=\ "<?php echo \$consulta[71]?>\ ">
```

```
<label>Materias perdidas:</label>
```

```
<select name=\ "perdidas\ "
id=\ "perdidas\ ">
```

```
<option value=\ "<?php echo
\$consulta[72]?>\ "><?php echo
\$consulta[72]?></option>
```

```
<option value=\ "\ ">Seleccione:</option>
```

```
<option value=\ "0\ ">0</option>
```

```
<option value=\ "1\ ">1</option>
```

```
<option value=\ "2\ ">2</option>
```

```
<option value=\ "3\ ">3</option>
```

```
<option value=\ "4\ ">4</option>
```

```
<option value=\ "5\ ">5</option>
```

```
<option value="6">6</option>
<option value="7">7</option>
<option value="8">8</option>
<option value="9">9</option>
<option value="10">10</option>
</select>
```

```
</div>
```

```
<div>
```

```
<label>Promedio 1 2018-2:</label>
<input type="text" name="promedioant"
placeholder="Ingrese el promedio de
semestre anterior" id="promedioant"
value="<?php echo \$consulta[73]?>">
<label>Promedio 2 2018-1:</label>
<input type="text" name="promediotra"
placeholder="Ingrese el promedio"
value="<?php echo \$consulta[74]?>">
```

```
</div>
```

```
<div>
```

```
<label>Beneficios recibidos en el
semestre anterior 1:</label>
<select name="bene1ant"
id="bene1ant">
<option value="<?php echo
\$consulta[75]?>"><?php echo
\$consulta[75]?></option>
<option value="">Seleccione:</option>
<?php \$query7 = \$conn -> query
("SELECT * FROM beneficio1 ORDER
BY NOMBENEFICIO1 ASC");
while (\$valores =
mysqli_fetch_array(\$query7))
{
echo '<option value="' . \$valores
[NOMBENEFICIO1] . '">' . \$valores
[NOMBENEFICIO1] . '</option>';
}
?>
```

```
</select>
```

```
<label>Beneficios recibidos en el
semestre anterior 2:</label>
<select name="bene2ant"
id="bene2ant">
<option value="<?php echo
\$consulta[76]?>"><?php echo
```

```

\${consulta[76]}?></option>
<option value="" \">Seleccione:</option>
<?php \${query7} = \${conn} -> query
(\\"SELECT * FROM beneficio1 ORDER
BY NOMBENEFICIO1 ASC\\");
while (\${valores} =
mysqli_fetch_array(\${query7}))
{
echo ' <option value="" .\${valores}
[NOMBENEFICIO1].\\">.\${valores}
[NOMBENEFICIO1].' </option>';
}
?>
</select>

```

```

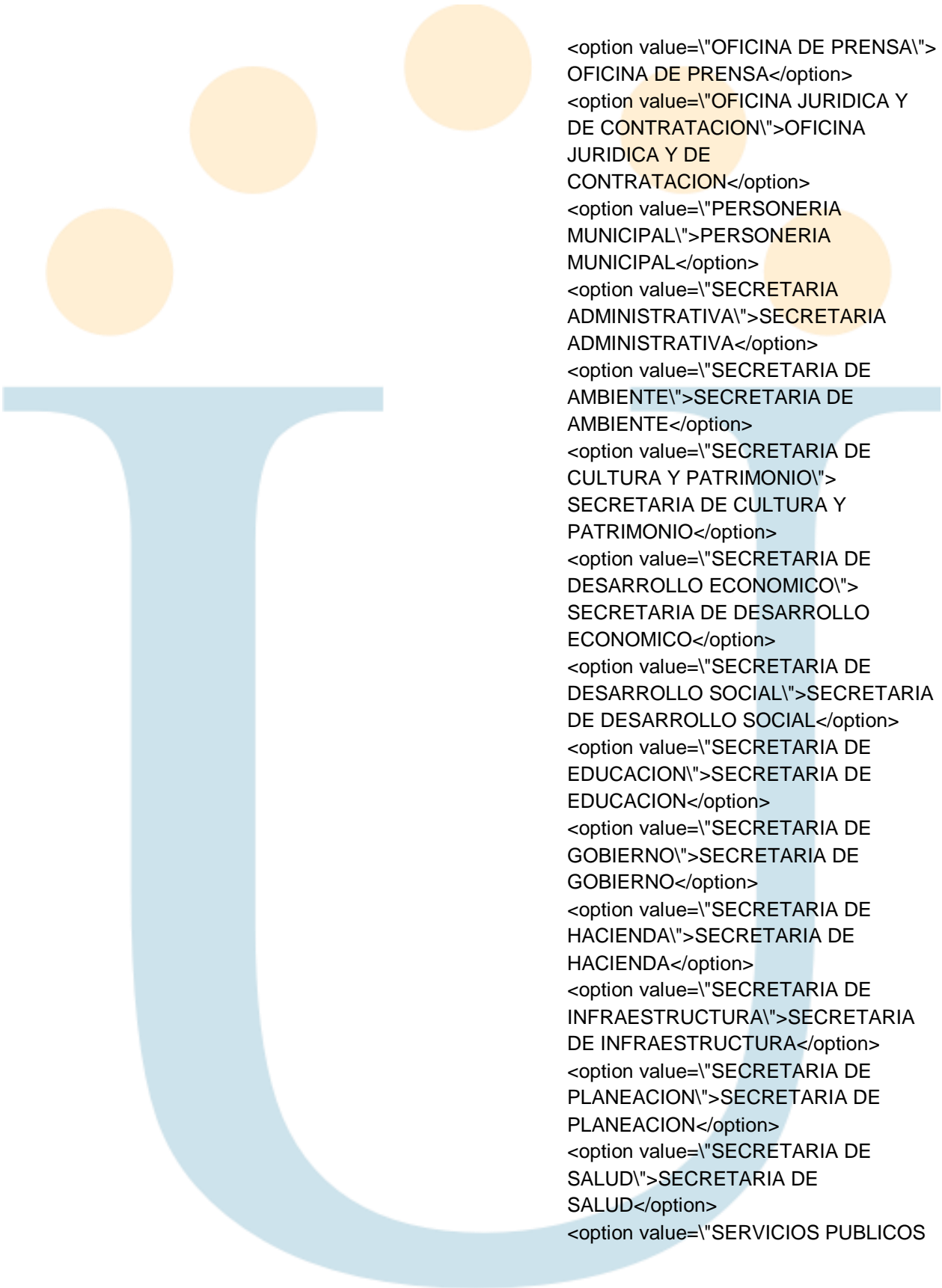
</div>
<div>

```

```

<label>Horas sociales cumplidas:</label>
<select name="hscump" id="hscump\\">
<option value="" <?php echo
\${consulta[77]}?>\\><?php echo
\${consulta[77]}?></option>
<option value="" \">Seleccione:</option>
<option value="1\\">1</option>
<option value="2\\">2</option>
<option value="3\\">3</option>
<option value="4\\">4</option>
<option value="...">...</option>
<option value="100\\">100</option>
</select>
<label>Dependencia donde realizo horas
sociales:</label>
<select name="dep1" id="dep1\\">
<option value="" <?php echo
\${consulta[78]}?>\\><?php echo
\${consulta[78]}?></option>
<option value="" \">Seleccione:</option>
<option value="CONCEJO
MUNICIPAL\\">CONCEJO
MUNICIPAL</option>
<option value="DESPACHO DEL
ALCALDE\\">DESPACHO DEL
ALCALDE</option>
<option value="IMRD\\">IMRD</option>
<option value="NINGUNO\\">
NINGUNO</option>

```



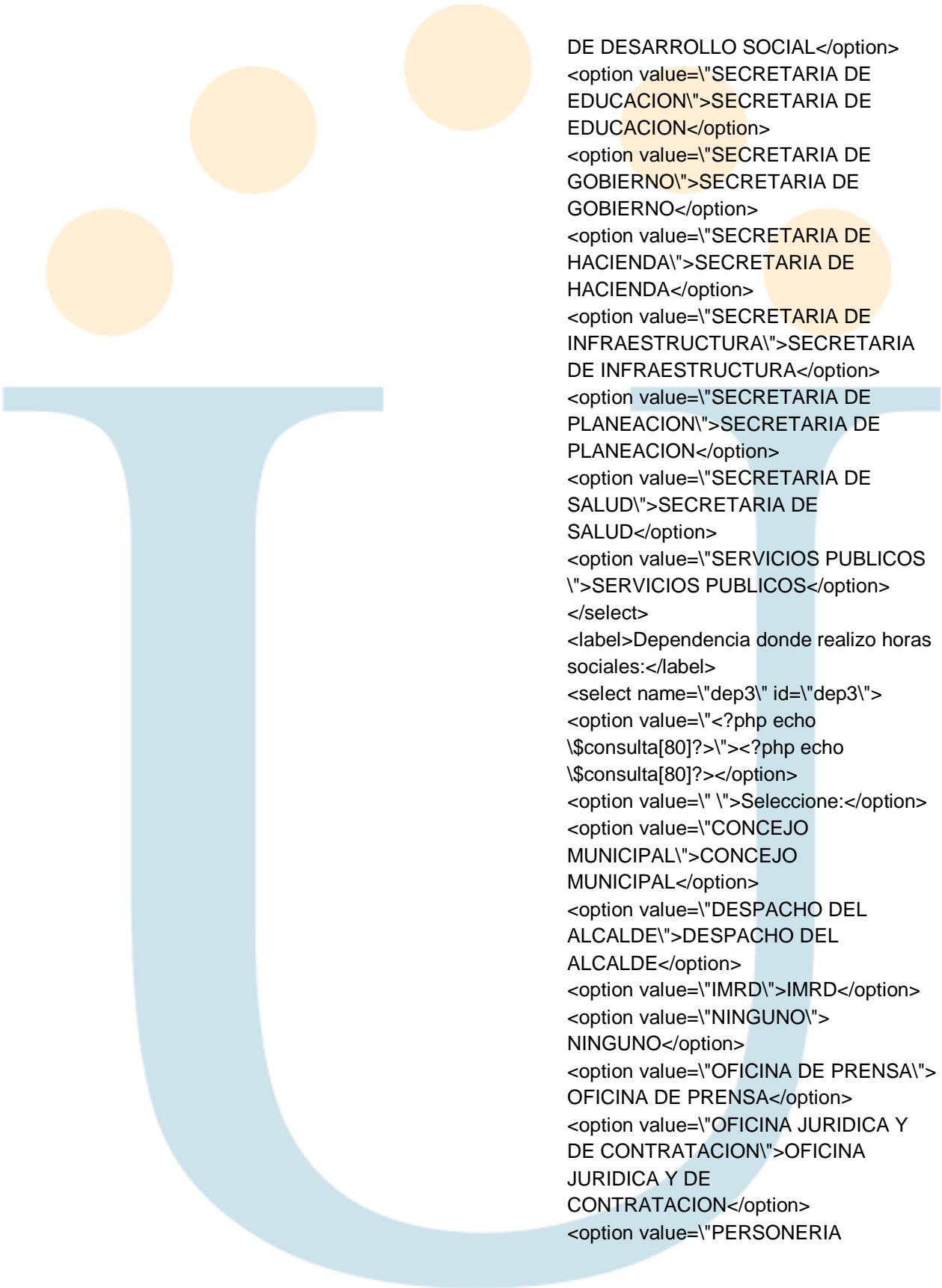
<option value="OFICINA DE PRENSA">
OFICINA DE PRENSA</option>
<option value="OFICINA JURIDICA Y
DE CONTRATACION">OFICINA
JURIDICA Y DE
CONTRATACION</option>
<option value="PERSONERIA
MUNICIPAL">PERSONERIA
MUNICIPAL</option>
<option value="SECRETARIA
ADMINISTRATIVA">SECRETARIA
ADMINISTRATIVA</option>
<option value="SECRETARIA DE
AMBIENTE">SECRETARIA DE
AMBIENTE</option>
<option value="SECRETARIA DE
CULTURA Y PATRIMONIO">
SECRETARIA DE CULTURA Y
PATRIMONIO</option>
<option value="SECRETARIA DE
DESARROLLO ECONOMICO">
SECRETARIA DE DESARROLLO
ECONOMICO</option>
<option value="SECRETARIA DE
DESARROLLO SOCIAL">SECRETARIA
DE DESARROLLO SOCIAL</option>
<option value="SECRETARIA DE
EDUCACION">SECRETARIA DE
EDUCACION</option>
<option value="SECRETARIA DE
GOBIERNO">SECRETARIA DE
GOBIERNO</option>
<option value="SECRETARIA DE
HACIENDA">SECRETARIA DE
HACIENDA</option>
<option value="SECRETARIA DE
INFRAESTRUCTURA">SECRETARIA
DE INFRAESTRUCTURA</option>
<option value="SECRETARIA DE
PLANEACION">SECRETARIA DE
PLANEACION</option>
<option value="SECRETARIA DE
SALUD">SECRETARIA DE
SALUD</option>
<option value="SERVICIOS PUBLICOS



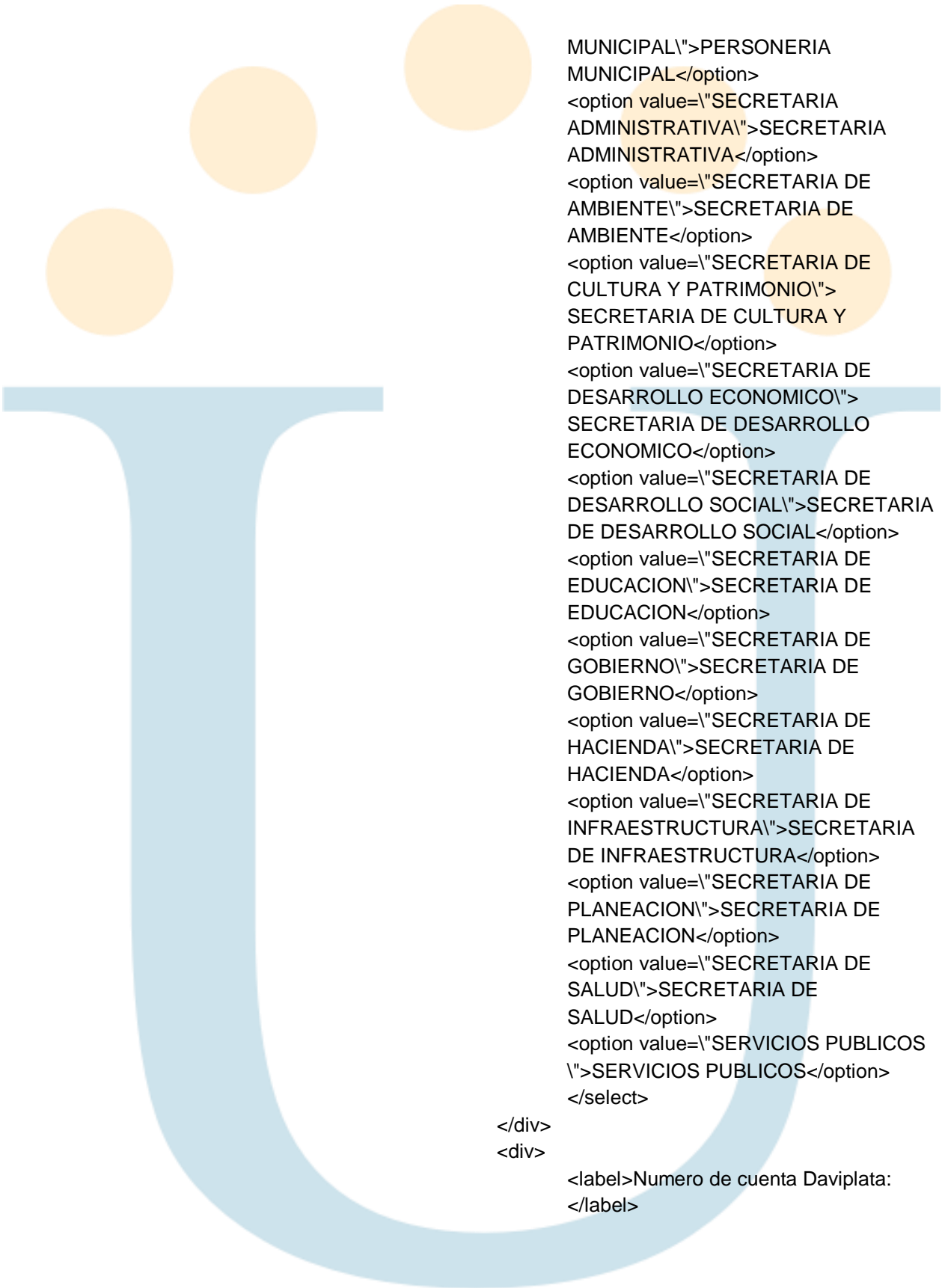
```
\">SERVICIOS PUBLICOS</option>
</select>
</div>
<div>
```

```
<label>Dependencia donde realizo horas
sociales:</label>
```

```
<select name=\"dep2\" id=\"dep2\">
<option value=\"<?php echo
\$consulta[79]?>\"><?php echo
\$consulta[79]?></option>
<option value=\" \">Seleccione:</option>
<option value=\"CONCEJO
MUNICIPAL\">CONCEJO
MUNICIPAL</option>
<option value=\"DESPACHO DEL
ALCALDE\">DESPACHO DEL
ALCALDE</option>
<option value=\"IMRD\">IMRD</option>
<option value=\"NINGUNO\">
NINGUNO</option>
<option value=\"OFICINA DE PRENSA\">
OFICINA DE PRENSA</option>
<option value=\"OFICINA JURIDICA Y
DE CONTRATACION\">OFICINA
JURIDICA Y DE
CONTRATACION</option>
<option value=\"PERSONERIA
MUNICIPAL\">PERSONERIA
MUNICIPAL</option>
<option value=\"SECRETARIA
ADMINISTRATIVA\">SECRETARIA
ADMINISTRATIVA</option>
<option value=\"SECRETARIA DE
AMBIENTE\">SECRETARIA DE
AMBIENTE</option>
<option value=\"SECRETARIA DE
CULTURA Y PATRIMONIO\">
SECRETARIA DE CULTURA Y
PATRIMONIO</option>
<option value=\"SECRETARIA DE
DESARROLLO ECONOMICO\">
SECRETARIA DE DESARROLLO
ECONOMICO</option>
<option value=\"SECRETARIA DE
DESARROLLO SOCIAL\">SECRETARIA
```



```
DE DESARROLLO SOCIAL</option>
<option value=\"SECRETARIA DE
EDUCACION\">SECRETARIA DE
EDUCACION</option>
<option value=\"SECRETARIA DE
GOBIERNO\">SECRETARIA DE
GOBIERNO</option>
<option value=\"SECRETARIA DE
HACIENDA\">SECRETARIA DE
HACIENDA</option>
<option value=\"SECRETARIA DE
INFRAESTRUCTURA\">SECRETARIA
DE INFRAESTRUCTURA</option>
<option value=\"SECRETARIA DE
PLANEACION\">SECRETARIA DE
PLANEACION</option>
<option value=\"SECRETARIA DE
SALUD\">SECRETARIA DE
SALUD</option>
<option value=\"SERVICIOS PUBLICOS
\">SERVICIOS PUBLICOS</option>
</select>
<label>Dependencia donde realizo horas
sociales:</label>
<select name=\"dep3\" id=\"dep3\">
<option value=\"<?php echo
\$consulta[80]?>\"><?php echo
\$consulta[80]?></option>
<option value=\" \">Seleccione:</option>
<option value=\"CONCEJO
MUNICIPAL\">CONCEJO
MUNICIPAL</option>
<option value=\"DESPACHO DEL
ALCALDE\">DESPACHO DEL
ALCALDE</option>
<option value=\"IMRD\">IMRD</option>
<option value=\"NINGUNO\">
NINGUNO</option>
<option value=\"OFICINA DE PRENSA\">
OFICINA DE PRENSA</option>
<option value=\"OFICINA JURIDICA Y
DE CONTRATACION\">OFICINA
JURIDICA Y DE
CONTRATACION</option>
<option value=\"PERSONERIA
```



```
MUNICIPAL">PERSONERIA
MUNICIPAL</option>
<option value="SECRETARIA
ADMINISTRATIVA">SECRETARIA
ADMINISTRATIVA</option>
<option value="SECRETARIA DE
AMBIENTE">SECRETARIA DE
AMBIENTE</option>
<option value="SECRETARIA DE
CULTURA Y PATRIMONIO">
SECRETARIA DE CULTURA Y
PATRIMONIO</option>
<option value="SECRETARIA DE
DESARROLLO ECONOMICO">
SECRETARIA DE DESARROLLO
ECONOMICO</option>
<option value="SECRETARIA DE
DESARROLLO SOCIAL">SECRETARIA
DE DESARROLLO SOCIAL</option>
<option value="SECRETARIA DE
EDUCACION">SECRETARIA DE
EDUCACION</option>
<option value="SECRETARIA DE
GOBIERNO">SECRETARIA DE
GOBIERNO</option>
<option value="SECRETARIA DE
HACIENDA">SECRETARIA DE
HACIENDA</option>
<option value="SECRETARIA DE
INFRAESTRUCTURA">SECRETARIA
DE INFRAESTRUCTURA</option>
<option value="SECRETARIA DE
PLANEACION">SECRETARIA DE
PLANEACION</option>
<option value="SECRETARIA DE
SALUD">SECRETARIA DE
SALUD</option>
<option value="SERVICIOS PUBLICOS
">SERVICIOS PUBLICOS</option>
</select>
```

```
</div>
<div>
```

```
<label>Numero de cuenta Daviplata:
</label>
```

```

<input name="\nocuenta\" type="\text\"
placeholder="\Ingrese el numero de
cuenta\" value="\<?php echo
\$consulta[81]?>\">
<label>Dias de la semana que viaja
</label>
<input type="\text\" name="\dias\"
value="\<?php echo \$consulta[82]?>\">
</div>
<div>
<label>ciudad a la que viaja *:</label>
<select name="ciudad\"
required="\required\" id="ciudad\">
<option value="\<?php echo
\$consulta[83]?>\"><?php echo
\$consulta[83]?></option>
<option value="\ \">Seleccione:</option>
<?php \$query7 = \$conn -> query
("SELECT * FROM ciudadviaja ORDER
BY NOMCIUDAD");
    while (\$valores =
        mysqli_fetch_array(\$query7))
    {
        echo '<option value="\'.$valores
[NOMCIUDAD].\">.\$valores
[NOMCIUDAD].\"></option>';
    }
?>
</select>
<label>Total Días :</label>
<select name="tdias\" id="tdias\"
style="\text-align: right;\"
onBlur="multiplica(this.form)\">
<option value="\<?php echo
\$consulta[84]?>\"><?php echo
\$consulta[84]?></option>
<option value="0\">Seleccione:</option>
<option value="0\">0</option>
<option value="1\">1</option>
<option value="2\">2</option>
<option value="3\">3</option>
<option value="4\">4</option>
<option value="5\">5</option>
<option value="6\">6</option>
<option value="7\">7</option>

```



```

</select>
</div>
<div>
<label>Valor por 2 recorridos *:</label>
<input name="\val2r\" type="\text\"
required="\required\"
placeholder="\Ingrese el valor del
recorrido\" style="\text-align: right;\"
value="\<?php echo \$consulta[85]?>\"
onBlur="\multiplica(this.form)\">
<label>Valor semanal:</label>
<input type="\text\" name="\vals\"
placeholder="\Ingrese el valor semanal\"
style="\text-align: right;\" value="\<?php
echo \$consulta[86]?>\">
</div>
<div>
<label>Valor mensual bruto:</label>
<input type="\text\" name="\valm\"
placeholder="\Ingrese el valor mensual
bruto\" style="\text-align: right;\"
value="\<?php echo \$consulta[87]?>\">
<label>Valor a pagar por mes:</label>
<input type="\text\" name="\valp\"
placeholder="\Ingrese el valor a pagar por
mes\" style="\text-align: right;\"
value="\<?php echo \$consulta[88]?>\">
</div>
<div>
<label>Meses a girar *:</label>
<select name="\mesgi\"
required="\required\" id="\mesgi\"
style="\text-align: right;\"
onBlur="\multiplica(this.form)\">
<option value="\<?php echo
\$consulta[89]?>\"><?php echo
\$consulta[89]?></option>
<option value="\0\">Seleccione:</option>
<option value="\0\">0</option>
<option value="\1\">1</option>
<option value="\2\">2</option>
<option value="\3\">3</option>
<option value="\4\">4</option>
<option value="\5\">5</option>
</select>

```

```

</div>
<div>
<label>Total semestre:</label>
<input type="text" name="tsemestre"
placeholder="Ingrese el total a pagar"
style="text-align: right;" value="<?php
echo \$consulta[90]?>">
</div>
</div>
<label><center><h2>Ratificado:</h2>
</center></label>
</div>
<div>
<label>Ratificado:</label>
<textarea type="text" name="ratificado"
placeholder="Ingrese ratificado"><?php
echo \$consulta[91]?></textarea>
</div>
<div>
<label>Estado:</label>
<textarea type="text" name="estado"
placeholder="Ingrese el estado"><?php
echo \$consulta[92]?></textarea>
</div>
<div>
<label>observaciones:</label>
<textarea type="text"
name="observacion"
placeholder="Ingrese la observación
\" ><?php echo \$consulta[93]?>
</textarea>
</div>
</H3></td>
</tr>
</table>
</center>
</span>
<div class="button">
<input type="submit" value="Modificar">
</div>
</form>
</body>
</html>";
}
if ($bene = fopen($bene, "a"))
{
    if (fwrite($bene, "$men"))

```

```

    {
    }
    else
    {
        ?>
        <script type="text/javascript">
            alert("Ha habido un problema modificarbeneficiario.php");
            window.location.href='index.php';
        </script>
        <?php
    }
    fclose($bene);
}
?>

```

9.1.14 pantalla estadísticas para nuevos periodos

```

<?php
    $esta = "esta$periodo.php";
    if(file_exists ($esta))
    {

    }
    else
    {
        $me = "<?php
include (\\"blodese.php\");
\\$conn = new mysqli('127.0.0.1', 'root', '', 'foest1') or die( \"No se conecto a la base de datos\");
\\$result = mysqli_query(\\$conn, \"select GENERO, count(*) as $tabla from $tabla group by
GENERO;\");
\\$resulta = mysqli_query(\\$conn, \"select BENEFICIO2, count(*) as $tabla from $tabla group by
BENEFICIO2;\");
\\$resultad = mysqli_query(\\$conn, \"select ID_US, count(*) as $tabla from $tabla group by
ID_US;\");
\\$resultado = mysqli_query(\\$conn, \"select POESPECIAL, count(*) as $tabla from $tabla group
by POESPECIAL;\");
\\$resultados = mysqli_query(\\$conn, \"SELECT CASE WHEN (EDAD BETWEEN 18 and 34)
THEN 'De 18 a 34' ELSE CASE WHEN (EDAD BETWEEN 0 and 17) THEN ' De 0 a 17' ELSE
CASE WHEN (EDAD BETWEEN 35 and 51) THEN 'De 35 a 51' ELSE CASE WHEN (EDAD
BETWEEN 52 and 68) THEN 'De 52 a 68' ELSE CASE WHEN (EDAD BETWEEN 69 and 85)
then 'De 69 a 85' ELSE CASE WHEN (EDAD >= 86) THEN 'De 86 o mas años' END END END
END END END rango, COUNT(*) total FROM $tabla GROUP BY rango\");
\\$resul = mysqli_query(\\$conn, \"select BENEFICIO1, count(*) as $tabla from $tabla group by
BENEFICIO1;\");

```

```

\ $resu = mysqli_query(\ $conn, \ "select ESTADOCIVIL, count(*) as $tabla from $tabla group by
ESTADOCIVIL;\");
\ $res = mysqli_query(\ $conn, \ "select ANTIGSISBEN, count(*) as $tabla from $tabla group by
ANTIGSISBEN;\");
\ $re = mysqli_query(\ $conn, \ "select TRBAJAACTUAL, count(*) as $tabla from $tabla group by
TRBAJAACTUAL;\");
\ $graficos = mysqli_query(\ $conn, \ "select VIVIENDA, count(*) as $tabla from $tabla group by
VIVIENDA;\");
\ $grafico = mysqli_query(\ $conn, \ "select ESTRATO, count(*) as $tabla from $tabla group by
ESTRATO;\");
\ $grafic = mysqli_query(\ $conn, \ "select ROTROBENE, count(*) as $tabla from $tabla group by
ROTROBENE;\");
\ $grafi = mysqli_query(\ $conn, \ "select UNIVERSIDAD, count(*) as $tabla from $tabla group by
UNIVERSIDAD;\");
\ $graf = mysqli_query(\ $conn, \ "select NIVEL, count(*) as $tabla from $tabla group by NIVEL;\");
\ $gra = mysqli_query(\ $conn, \ "select CARRERA, count(*) as $tabla from $tabla group by
CARRERA;\");
\ $gr = mysqli_query(\ $conn, \ "select MODALIDAD, count(*) as $tabla from $tabla group by
MODALIDAD;\");
\ $estadisticas = mysqli_query(\ $conn, \ "select SEMESTRE, count(*) as $tabla from $tabla group
by SEMESTRE;\");
\ $estadistica = mysqli_query(\ $conn, \ "select DEPENDENCIA1, count(*) as $tabla from $tabla
group by DEPENDENCIA1;\");
\ $estadistic = mysqli_query(\ $conn, \ "select DEPENDENCIA2, count(*) as $tabla from $tabla
group by DEPENDENCIA2;\");
\ $estadisti = mysqli_query(\ $conn, \ "select DEPENDENCIA3, count(*) as $tabla from $tabla group
by DEPENDENCIA3;\");
\ $estadist = mysqli_query(\ $conn, \ "select CIUDADVIAJA, count(*) as $tabla from $tabla group by
CIUDADVIAJA;\");

```

?>

```

<!doctype html>
<html lang="en">
<head>
  <title> Base de datos</title>
  <link rel="stylesheet" type="text/css" href="estilopri.css">
  <style type="text/css">
    table
    {
      height: auto;
      width: auto;
      text-align: left;
      display: block;
      overflow-x: auto;
    }
  th

```

```

    {
      position: static;
    }
  </style>
<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript">
  google.charts.load('current', {'packages':['corechart']});
  google.charts.setOnLoadCallback(drawChart);
  google.charts.setOnLoadCallback(drawChart1);
  google.charts.setOnLoadCallback(drawChart2);
  google.charts.setOnLoadCallback(drawChart3);
  google.charts.setOnLoadCallback(drawChart4);
  google.charts.setOnLoadCallback(drawChart5);
  google.charts.setOnLoadCallback(drawChart6);
  google.charts.setOnLoadCallback(drawChart7);
  google.charts.setOnLoadCallback(drawChart8);
  google.charts.setOnLoadCallback(drawChart9);
  google.charts.setOnLoadCallback(drawChart10);
  google.charts.setOnLoadCallback(drawChart11);
  google.charts.setOnLoadCallback(drawChart12);
  google.charts.setOnLoadCallback(drawChart13);
  google.charts.setOnLoadCallback(drawChart14);
  google.charts.setOnLoadCallback(drawChart15);
  google.charts.setOnLoadCallback(drawChart16);
  google.charts.setOnLoadCallback(drawChart17);
  google.charts.setOnLoadCallback(drawChart18);
  google.charts.setOnLoadCallback(drawChart19);
  google.charts.setOnLoadCallback(drawChart20);
  function drawChart()
  {
    var data = google.visualization.arrayToDataTable([
      ['Language', 'Rating'],
      <?php
      if(\$_result->num_rows > 0)
      {
        while(\$_row = mysqli_fetch_array(\$_result,
          MYSQLI_ASSOC))
        {
          echo "[".$row['GENERO']."",
            ".$row['$tabla']."\",";
        }
      }
      ?>
    ]);
    var options =

```

```

    {
        title: 'Genero',
        chartArea:
        {
            left: 0,
            top: 19,
            right: 0,
            width: '150%',
            height: '150%',
            legend: 'none',
        }
    }
};

var chart_div = document.getElementById('piechart');
var chart = new google.visualization.PieChart(chart_div);
google.visualization.events.addListener(chart,'ready', function ()
{chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+\"'\>'; });

chart.draw(data,options);
}
function drawChart1()
{
    var data = google.visualization.arrayToDataTable([
        ['Language', 'Rating'],
        <?php
        if(\$resulta->num_rows > 0)
        {
            while(\$row = mysqli_fetch_array(\$resulta,
            MYSQLI_ASSOC))
            {
                echo \"[\".\$row['BENEFICIO2'].\"],\".\",
                \".\$row['$tabla'].\"],\";
            }
        }
        ?>
    ]);
    var options =
    {
        title: 'Beneficio',
        is3D: false,
        chartArea:
        {
            left: 0,
            top: 19,
            right: 0,
            width: '150%',

```

```

        height: '150%',
    }
};
var chart_div = document.getElementById('piechart2');
var chart = new google.visualization.PieChart(chart_div);
google.visualization.events.addListener(chart,'ready', function ()
{chart_div.innerHTML = '
    ]);
    var options =
    {
        title: 'Documento',
        is3D: true,
        chartArea:
        {
            left: 0,
            top: 19,
            right: 0,
            width: '150%',
            height: '150%',
        }
    };
    var chart_div = document.getElementById('piechart3');
    var chart = new google.visualization.PieChart(chart_div);
    google.visualization.events.addListener(chart,'ready', function ()
    {chart_div.innerHTML = '
]);
var options =
{
    title: 'Poblacion',
    chartArea:
    {
        left: 0,
        top: 19,
        right: 0,
        width: '150%',
        height: '150%',
    }
};
var chart_div = document.getElementById('piechart4');
var chart = new google.visualization.PieChart(chart_div);
google.visualization.events.addListener(chart,'ready', function ()
{chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\">> });
chart.draw(data, options);
}
function drawChart4()
{
    var data = google.visualization.arrayToDataTable([
        ['Language', 'Rating'],
        <?php
        if(\$resultados->num_rows > 0)
        {
            while(\$row = mysqli_fetch_array(\$resultados,
            MYSQLI_ASSOC))
            {
                echo "\"\.$row['rango'].\", \"\.$row['total'].\",\";
            }
        }
    ]

```



```

    });
    var options =
    {
        title: 'Edad',
        chartArea:
        {
            left: 0,
            top: 19,
            right: 0,
            width: '150%',
            height: '150%',
        }
    };
    var chart_div = document.getElementById('piechart5');
    var chart = new google.visualization.PieChart(chart_div);
    google.visualization.events.addListener(chart,'ready', function ()
    {chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\>'; });
    chart.draw(data, options);
}
function drawChart5()
{
    var data = google.visualization.arrayToDataTable([
        ['Language', 'Rating'],
        <?php
        if(\$resul->num_rows > 0)
        {
            while(\$row = mysqli_fetch_array(\$resul,
            MYSQLI_ASSOC))
            {
                echo \"[\".\$row['BENEFICIO1'].\",
                \".\$row['$tabla'].\",\",
            }
        }
        <?php
    ]);
    var options =
    {
        title: 'Beneficio',
        is3D: false,
        chartArea:
        {
            left: 0,
            top: 19,
            right: 0,

```

```

        width: '150%',
        height: '150%',
    }
};

var chart_div = document.getElementById('piechart6');
var chart = new google.visualization.PieChart(chart_div);
google.visualization.events.addListener(chart,'ready', function ()
{chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\>'; });
chart.draw(data, options);
}
function drawChart6()
{
    var data = google.visualization.arrayToDataTable([
        ['Language', 'Rating'],
        <?php
        if(\$resu->num_rows > 0)
        {
            while(\$row = mysqli_fetch_array(\$resu,
            MYSQLI_ASSOC))
            {
                echo \"[\".\$row['ESTADOCIVIL'].\"\",
                \".\$row['$staba'].\"],\";
            }
        }
        ?>
    ]);
    var options =
    {
        title: 'Estado Civil',
        is3D: false,
        chartArea:
        {
            left: 0,
            top: 19,
            right: 0,
            width: '150%',
            height: '150%',
        }
    };
    var chart_div = document.getElementById('piechart7');
    var chart = new google.visualization.PieChart(chart_div);
    google.visualization.events.addListener(chart,'ready', function ()
    {chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\>'; });
    chart.draw(data, options);
}

```

```

function drawChart7()
{
    var data = google.visualization.arrayToDataTable([
        ['Language', 'Rating'],
        <?php
        if(\$res->num_rows > 0)
        {
            while(\$row = mysqli_fetch_array(\$res,
                MYSQLI_ASSOC))
            {
                echo \"[\".\$row['ANTIGSISBEN'].\"],\",
                    \".\$row['$tabla'].\"],\";
            }
        }
        >?
    ]);
    var options =
    {
        title: 'Estado Civil',
        is3D: false,
        chartArea:
        {
            left: 0,
            top: 19,
            right: 0,
            width: '150%',
            height: '150%',
        }
    };
    var chart_div = document.getElementById('piechart8');
    var chart = new google.visualization.PieChart(chart_div);
    google.visualization.events.addListener(chart,'ready', function ()
    {chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\">>'); });
    chart.draw(data, options);
}
function drawChart8()
{
    var data = google.visualization.arrayToDataTable([
        ['Language', 'Rating'],
        <?php
        if(\$re->num_rows > 0)
        {
            while(\$row = mysqli_fetch_array(\$re, MYSQLI_ASSOC))
            {
                echo \"[\".\$row['TRBAJAACTUAL'].\"],\",

```

```

        \".\$row['$tabla'].\",\";
    }
}
?>
]);
var options =
{
    title: 'Trabaja',
    is3D: false,
    chartArea:
    {
        left: 0,
        top: 19,
        right: 0,
        width: '150%',
        height: '150%',
    }
};
var chart_div = document.getElementById('piechart9');
var chart = new google.visualization.PieChart(chart_div);
google.visualization.events.addListener(chart,'ready', function ()
{chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\>'; });
chart.draw(data, options);
}
function drawChart9()
{
    var data = google.visualization.arrayToDataTable([
        ['Language', 'Rating'],
        <?php
        if(\$graficos->num_rows > 0)
        {
            while(\$row = mysqli_fetch_array(\$graficos,
            MYSQLI_ASSOC))
            {
                echo \"[\".\$row['VIVIENDA'].\",
                \".\$row['$tabla'].\",\";
            }
        }
        ?>
    ]);
    var options =
    {
        title: 'Vivienda',
        is3D: false,
        chartArea:

```

```

        {
            left: 0,
            top: 19,
            right: 0,
            width: '150%',
            height: '150%',
        }
    };
    var chart_div = document.getElementById('piechart10');
    var chart = new google.visualization.PieChart(chart_div);
    google.visualization.events.addListener(chart,'ready', function ()
    {chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\>'; });
    chart.draw(data, options);
}
function drawChart10()
{
    var data = google.visualization.arrayToDataTable([
        ['Language', 'Rating'],
        <?php
        if(\$grafico->num_rows > 0)
        {
            while(\$row = mysqli_fetch_array(\$grafico,
            MYSQLI_ASSOC))
            {
                echo \"[\".\$row['ESTRATO'].\",\",
                \".\$row['Stabla'].\",\",
            }
        }
        ?>
    ]);
    var options =
    {
        title: 'Estrato',
        is3D: false,
        chartArea:
        {
            left: 0,
            top: 19,
            right: 0,
            width: '150%',
            height: '150%',
        }
    };
    var chart_div = document.getElementById('piechart11');
    var chart = new google.visualization.PieChart(chart_div);

```

```

google.visualization.events.addListener(chart,'ready', function ()
{chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\>'; });
chart.draw(data, options);
}
function drawChart11()
{
    var data = google.visualization.arrayToDataTable([
        ['Language', 'Rating'],
        <?php
        if(\$grafic->num_rows > 0)
        {
            while(\$row = mysqli_fetch_array(\$grafic,
            MYSQLI_ASSOC))
            {
                echo \"[\".\$row['ROTROBENE'].\",
                \".\$row['$tabla'].\",\",
            }
        }
        ?>
    ]);
    var options =
    {
        title: 'Beneficios diferentes a FOEST',
        is3D: false,
        chartArea:
        {
            left: 0,
            top: 19,
            right: 0,
            width: '150%',
            height: '150%',
        }
    };
    var chart_div = document.getElementById('piechart12');
    var chart = new google.visualization.PieChart(chart_div);
    google.visualization.events.addListener(chart,'ready', function ()
    {chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\>'; });
    chart.draw(data, options);
}
function drawChart12()
{
    var data = google.visualization.arrayToDataTable([
        ['Language', 'Rating'],
        <?php
        if(\$grafi->num_rows > 0)

```

```

    {
        while(\$row = mysqli_fetch_array(\$grafi,
        MYSQLI_ASSOC))
        {
            echo \"[\".\$row['UNIVERSIDAD'].\"],\";
            echo \"[\".\$row['$staba'].\"],\";
        }
    }
    ?>
]);
var options =
{
    title: 'Universidad',
    is3D: false,
    chartArea:
    {
        left: 0,
        top: 19,
        right: 0,
        width: '150%',
        height: '150%',
    }
};
var chart_div = document.getElementById('piechart13');
var chart = new google.visualization.PieChart(chart_div);
google.visualization.events.addListener(chart,'ready', function ()
{chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+\">'; });
chart.draw(data, options);
}
function drawChart13()
{
    var data = google.visualization.arrayToDataTable([
    ['Language', 'Rating'],
    <?php
    if(\$graf->num_rows > 0)
    {
        while(\$row = mysqli_fetch_array(\$graf,
        MYSQLI_ASSOC))
        {
            echo \"[\".\$row['NIVEL'].\"],\".\$row['$staba'].\"],\";
        }
    }
    ?>
]);
var options =

```

```

    {
        title: 'Nivel',
        is3D: false,
        chartArea:
        {
            left: 0,
            top: 19,
            right: 0,
            width: '150%',
            height: '150%',
        }
    }
};

var chart_div = document.getElementById('piechart14');
var chart = new google.visualization.PieChart(chart_div);
google.visualization.events.addListener(chart,'ready', function ()
{chart_div.innerHTML = '
    ]);
    var options =
    {
        title: 'Carreras',
        is3D: false,
        chartArea:
        {
            left: 0,
            top: 19,
            right: 0,
            width: '150%',
            height: '150%',
        }
    }
};

```



```

    }
};
var chart_div = document.getElementById('piechart15');
var chart = new google.visualization.PieChart(chart_div);
google.visualization.events.addListener(chart,'ready', function ()
{chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\>'; });
chart.draw(data, options);
}
function drawChart15()
{
    var data = google.visualization.arrayToDataTable([
        ['Language', 'Rating'],
        <?php
        if(\$gr->num_rows > 0)
        {
            while(\$row = mysqli_fetch_array(\$gr, MYSQLI_ASSOC))
            {
                echo \"[\".\$row['MODALIDAD'].\",
                \".\$row['$tabla'].\",)\";
            }
        }
        ?>
    ]);
    var options =
    {
        title: 'Modalidad',
        is3D: false,
        chartArea:
        {
            left: 0,
            top: 19,
            right: 0,
            width: '150%',
            height: '150%',
        }
    };
    var chart_div = document.getElementById('piechart16');
    var chart = new google.visualization.PieChart(chart_div);
    google.visualization.events.addListener(chart,'ready', function ()
    {chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\>'; });
    chart.draw(data, options);
}
function drawChart16()
{
    var data = google.visualization.arrayToDataTable([

```

```

['Language', 'Rating'],
<?php
if(\Sestadisticas->num_rows > 0)
{
    while(\$row = mysqli_fetch_array(\Sestadisticas,
    MYSQLI_ASSOC))
    {
        echo "\"\.$row['SEMESTRE'].\",
        \"\.$row['$tabla'].\",\";
    }
}
?>
]);
var options =
{
    title: 'Modalidad',
    is3D: false,
    chartArea:
    {
        left: 0,
        top: 19,
        right: 0,
        width: '150%',
        height: '150%',
    }
};
var chart_div = document.getElementById('piechart17');
var chart = new google.visualization.PieChart(chart_div);
google.visualization.events.addListener(chart,'ready', function ()
{chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\>'; });
chart.draw(data, options);
}
function drawChart17()
{
    var data = google.visualization.arrayToDataTable([
    ['Language', 'Rating'],
    <?php
    if(\Sestadistica->num_rows > 0)
    {
        while(\$row = mysqli_fetch_array(\Sestadistica,
        MYSQLI_ASSOC))
        {
            echo "\"\.$row['DEPENDENCIA1'].\",
            \"\.$row['$tabla'].\",\";
        }
    }
}

```

```

    }
    ?>
});
var options =
{
    title: 'Dependencia 1',
    is3D: false,
    chartArea:
    {
        left: 0,
        top: 19,
        right: 0,
        width: '150%',
        height: '150%',
    }
};
var chart_div = document.getElementById('piechart18');
var chart = new google.visualization.PieChart(chart_div);
google.visualization.events.addListener(chart,'ready', function ()
{chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\>'; });
chart.draw(data, options);
}
function drawChart18()
{
    var data = google.visualization.arrayToDataTable([
        ['Language', 'Rating'],
        <?php
        if(\Sestadistic->num_rows > 0)
        {
            while(\$row = mysqli_fetch_array(\Sestadistic,
            MYSQLI_ASSOC))
            {
                echo \"[\".\$row['DEPENDENCIA2'].\",\",
                \".\$row['$tabla'].\",\",
            }
        }
        ?>
    ]);
    var options =
    {
        title: 'Dependencia 2',
        is3D: false,
        chartArea:
        {
            left: 0,

```

```

        top: 19,
        right: 0,
        width: '150%',
        height: '150%',
    }
};

var chart_div = document.getElementById('piechart19');
var chart = new google.visualization.PieChart(chart_div);
google.visualization.events.addListener(chart,'ready', function ()
{chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\">> });
chart.draw(data, options);
}
function drawChart19()
{
    var data = google.visualization.arrayToDataTable([
        ['Language', 'Rating'],
        <?php
        if(\$_stadisti->num_rows > 0)
        {
            while(\$_row = mysqli_fetch_array(\$_stadisti,
            MYSQLI_ASSOC))
            {
                echo \"[\".\$_row['DEPENDENCIA3'].\",
                \".\$_row['$tabla'].\",\",
            }
        }
        ?>
    ]);
    var options =
    {
        title: 'Dependencia 3',
        is3D: false,
        chartArea:
        {
            left: 0,
            top: 19,
            right: 0,
            width: '150%',
            height: '150%',
        }
    };

    var chart_div = document.getElementById('piechart20');
    var chart = new google.visualization.PieChart(chart_div);
    google.visualization.events.addListener(chart,'ready', function ()
    {chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\">> });

```

```

        chart.draw(data, options);
    }
    function drawChart20()
    {
        var data = google.visualization.arrayToDataTable([
            ['Language', 'Rating'],
            <?php
            if(\$_stadist->num_rows > 0)
            {
                while(\$_row = mysqli_fetch_array(\$_stadist,
                MYSQLI_ASSOC))
                {
                    echo "\"\$_row['CIUDADVIAJA'].\",
                    \"\$_row['$tabla'].\",\";
                }
            }
            ?>
        ]);
        var options =
        {
            title: 'Ciudad a la que viajan',
            is3D: false,
            chartArea:
            {
                left: 0,
                top: 19,
                right: 0,
                width: '150%',
                height: '150%',
            }
        };
        var chart_div = document.getElementById('piechart21');
        var chart = new google.visualization.PieChart(chart_div);
        google.visualization.events.addListener(chart,'ready', function ()
        {chart_div.innerHTML = '<img src=\"'+ chart.getImageURI()+'\">>'); });
        chart.draw(data, options);
    }
    function exportTableToExcel(tableID, filename = "")
    {
        var downloadLink;
        var dataType = 'application/vnd.ms-excel';
        var tableSelect = document.getElementById(tableID);
        var tableHTML = tableSelect.outerHTML.replace(/ /g, '%20');
        filename = filename?filename+'.xls':'usuarios.xls';
        downloadLink = document.createElement("a");

```

```

document.body.appendChild(downloadLink);
if(navigator.msSaveOrOpenBlob)
{
    var blob = new Blob(['utf-8', tableHTML],
        {
            type: dataType
        }
    );
    navigator.msSaveOrOpenBlob( blob, filename);
}
else
{
    downloadLink.href = 'data:' + dataType + ',' + tableHTML;
    downloadLink.download = filename;
    downloadLink.click();
}
}
</script>
</head>
<body>
<form id="form0" style="width:98.57%;height:2.6%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid #CCC;"><div><center>
<p>Sistema de información</p>
<div style="float: left;"><a href="opciones.php"><?php echo
$_SESSION["usuarios"]?></a></div>
<div style = "float: right;"><a href="salir.php">Salir</a></div>
<p>&nbsp;</p>
</center></div>
</form>
<form id="form2" method="POST" action="index.php" margin="0" style="float: left; width:18%; height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<?php
include ("menuopciones.php");
?>
</form>
<form id="form3" action="esta$periodo.php" margin="0" style="float: left;width: 80%; height:80%; background-color:white;color:blue;font-size: 14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid #CCC;">
<div style="float: left;"><a href="$periodo.php">atras</a></div><label for="textfield">
</label>
<label>Selecciona</label>
<select name="graficos">
<option value="0">Selecciona:</option>

```

```

        <option value=\"6\">Beneficio 1</option>
        <option value=\"2\">Beneficio 2</option>
        <option value=\"3\">Documento</option>
        <option value=\"5\">Edad</option>
        <option value=\"7\">Estado civil</option>
        <option value=\"4\">Poblacion especial</option>
        <option value=\"1\">Genero</option>
        <option value=\"8\">Antigüedad sisben</option>
        <option value=\"9\">Trabaja actualmente</option>
        <option value=\"10\">Tipo de vivienda</option>
        <option value=\"11\">Estrato</option>
        <option value=\"12\">Beneficios diferentes a FOEST</option>
        <option value=\"13\">Universidad</option>
        <option value=\"14\">Nivel</option>
        <option value=\"15\">Carrera</option>
        <option value=\"16\">Modalidad</option>
        <option value=\"17\">Semestre</option>
        <option value=\"18\">Dependencia horas sociales 1</option>
        <option value=\"19\">Dependencia horas sociales 2</option>
        <option value=\"20\">Dependencia horas sociales 3</option>
        <option value=\"21\">Ciudad a la que viajan</option>
    </select>
    <input type=\"submit\" name=\"Filtrar\" id=\"Filtrar\" value=\"Filtrar\">
</center>
<?php
    if (isset(\$_GET['graficos']))
    {
        switch(\$_GET['graficos'])
        {
            case 0:
                break;
            case 1:
                echo \"<center><h1>Genero</h1><center>\";
                echo \"<center><table id='tabla'><thead><div>
                <th><center>GENERO</center></th>
                <th><center>CANTIDAD</center></th>
                <th><center>PORCENTAJE</center></th>
                <th><center>TOTAL</center></th></div></thead>\";
                \$sql = \"select GENERO, count(*) as \$tabla from \$tabla group by
                GENERO;\";
                \$c = 0;
                \$a = 0;
                \$total = 0;
                \$result = mysqli_query(\$conn, \$sql);
                if(!\$result && !\$numero )

```

```

    {
        die('Ocurrió un error al obtener los valores de la base de datos: ' .
            mysqli_error());
    }
    while(\$row = mysqli_fetch_array(\$result, MYSQLI_ASSOC))
    {
        \$genero[\$c] = \$row["GENERO"];
        \$numero[\$c] = \$row["$tabla"];
        \$total = \$total + \$row["$tabla"];
        \$c++;
    }
    for (\$j=0; \$j<=\$c-1; \$j++)
    {
        \$a++;
        echo "<tr><td><center>\".\$genero[\$j].\"</center></td>\"";
        echo "<td><center>\".\$numero[\$j].\"</center></td>\"";
        echo "<td><center>\".
            .number_format(((\$numero[\$j]/\$total)*100), 1, ',', '').\"%</center>
            </td>\"";
        \$por[\$j]= round( (\$numero[\$j]/\$total)*100, 1);
        if (\$j==0)
        {
            echo "<td rowspan=\".\$c.\"><center>\".\$total.\"</center>
            </td></tr>\"";
        }
    }
    ?>
</table>
</center>
<p></p>
<?php
echo "<center><div id='piechart'></div></center>\"";
?>
<center>
<p></p>
<button onClick=\"exportTableToExcel('tabla', 'Genero Usuarios
    \$periodo')\"> Exportar a Excel</button>
<p></p>
</center>
<?php
break;
case 2:
echo "<center><h1>Beneficio 2</h1><center>\"";
echo "<center><table id='tabla'><thead><div>
<th><center>BENEFICIO</center></th>

```



```

<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>";
\${sqli} = "select BENEFICIO2, count(*) as $tabla from $tabla group by
BENEFICIO2;";
\${cc} = 0;
\${aa} = 0;
\${totale} = 0;
\${resulta} = mysqli_query(\${conn}, \${sqli});
if(!\${resulta} && !\${numeros} )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\${row} = mysqli_fetch_array(\${resulta}, MYSQLI_ASSOC))
{
    \${beneficio}[\${cc}] = \${row}["BENEFICIO2"];
    \${numeros}[\${cc}] = \${row}["$tabla"];
    \${totale} = \${totale} + \${row}["$tabla"];
    \${cc}++;
}
for (\${j}=0;\${j}<=\${cc}-1;\${j}++)
{
    \${aa}++;
    echo "<tr><td><center>\".\${beneficio}[\${j}].\"</center></td>\";
    echo "<td><center>\".\${numeros}[\${j}].\"</center></td>\";
    echo "<td><center>
    \".number_format((\${numeros}[\${j}]/\${totale})*100), 1, ',', ")\".%
    </center></td>\";
    \${porx}[\${j}] = round( (\${numeros}[\${j}]/\${totale})*100, 1);
    if (\${j}==0)
    {
        echo "<td rowspan= \".\${cc}.\"><center>\".\${totale}.\"
        </center></td></tr>\";
    }
}
?>
</table>
</center>
<p></p>
<?php
echo "<center><div id='piechart2'></div></center>\";
?>
</center>
<p></p>

```

```
<button onClick="exportTableToExcel('tabla', 'Beneficio 2
Usuarios $periodo')"> Exportar a Excel</button>
```

```
<p></p>
</center>
<?php
break;
```

case 3:

```
echo \"<center><h1>Documento</h1><center>\";
echo \"<center><table id='tabla'><thead><div>
<th><center>TIPO DE DOCUMENTO</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\";
\Sq = \"select ID_US, count(*) as $tabla from $tabla group by ID_US\";
\ $ccc = 0;
\ $aaa = 0;
\ $totales = 0;
\ $resultad = mysqli_query(\ $conn, \ $sq);
if(!\ $resultad && !\ $numer )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\ $row = mysqli_fetch_array(\ $resultad, MYSQLI_ASSOC))
{
    \ $documento[\ $ccc] = \ $row[\ "ID_US\""];
    \ $numer[\ $ccc] = \ $row[\ \"$tabla\""];
    \ $totales = \ $totales + \ $row[\ \"$tabla\""];
    \ $ccc++;
}
for (\ $j=0;\ $j<=\ $ccc-1;\ $j++)
{
    \ $aaa++;
    echo \"<tr><td><center>\".\ $documento[\ $j].\"</center></td>\";
    echo \"<td><center>\".\ $numer[\ $j].\"</center></td>\";
    echo \"<td><center>
    \".number_format(((\ $numer[\ $j]/\ $totales)*100), 1, ',', '').\"%
    </center></td>\";
    \ $porx[\ $j]= round( (\ $numer[\ $j]/\ $totales)*100, 1);
    if (\ $j==0)
    {
        echo \"<td rowspan='\ ".\ $ccc.\"><center>\".\ $totales.\"
        </center></td></tr>\";
    }
}
```

```

?>
</table>
</center>
<p></p>
<?php
echo \"<center><div id='piechart3'></div></center>\";
?>
<center>
<p></p>
<button onClick=\"exportTableToExcel('tabla', 'Documento
    Usuarios $periodo')\"> Exportar a Excel</button>
<p></p>
</center>
<?php
break;

```

case 4:

```

echo \"<center><h1>Poblacion especial</h1><center>\";
echo \"<center><table id='tabla'><thead><div>
<th><center>POBLACION</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\";
\\$s = \"select POESPECIAL, count(*) as $tabla from $tabla group by
POESPECIAL;\";
\\$cccc = 0;
\\$aaaa = 0;
\\$tota = 0;
\\$resultado = mysqli_query(\\$conn, \\$s);
if(!\\$resultado && !\\$nume )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\\$row = mysqli_fetch_array(\\$resultado, MYSQLI_ASSOC))
{
    \\$poblacion[\\$cccc] = \\$row[\"POESPECIAL\"];
    \\$nume[\\$cccc] = \\$row[\"$tabla\"];
    \\$tota = \\$tota + \\$row[\"$tabla\"];
    \\$cccc++;
}
for (\\$j=0;\\$j<=\\$cccc-1;\\$j++)
{
    \\$aaaa++;
    echo \"<tr><td><center>\".\\$poblacion[\\$j].\"</center></td>\";
    echo \"<td><center>\".\\$nume[\\$j].\"</center></td>\";
}

```

```

echo \"<td><center>\".number_format(((\\$nume[\\$j]/\\$tota)*100),
1, ',', ' ').\"%</center></td>\";
\\$xporx[\\$j]= round( (\\$nume[\\$j]/\\$tota)*100, 1);
if (\\$j==0)
{
    echo \"<td rowspan=\\\".\\$cccc.\\\"><center>\\\".\\$tota.\\\"
    </center></td></tr>\";
}
}
?>
</table>
</center>
<p></p>
<?php

```

```

echo \"<center><div id='piechart4'></div></center>\";
?>
<center>
<p></p>
    <button onClick=\"exportTableToExcel('tabla', 'Poblaciones
    Usuarios $periodo')\"> Exportar a Excel</button>
<p></p>
</center>
<?php
break;

```

case 5:

```

echo \"<center><h1>Edad</h1><center>\";
echo \"<center><table id='tabla'><thead><div>
<th><center>EDAD</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\";
\\$s = \"SELECT CASE WHEN (EDAD BETWEEN 18 and 34) THEN 'De 18
a 34' ELSE CASE WHEN (EDAD BETWEEN 0 and 17) THEN ' De 0 a 17'
ELSE CASE WHEN (EDAD BETWEEN 35 and 51) THEN 'De 35 a 51'
ELSE CASE WHEN (EDAD BETWEEN 52 and 68) THEN 'De 52 a 68'
ELSE CASE WHEN (EDAD BETWEEN 69 and 85) then 'De 69 a 85' ELSE
CASE WHEN (EDAD >= 86) THEN 'De 86 o mas años' END END END
END END END rango, COUNT(*) total FROM $tabla GROUP BY rango\";
\\$cccc = 0;
\\$aaaa = 0;
\\$tota = 0;
\\$resultados = mysqli_query(\\$conn, \\$s);
if(!\\$resultados && !\\$nume )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .

```

```

        mysqli_error());
    }
    while(\$row = mysqli_fetch_array(\$resultados, MYSQLI_ASSOC))
    {
        \$edad[\$cccc] = \$row["rango"];
        \$nume[\$cccc] = \$row["total"];
        \$tota = \$tota + \$row["total"];
        \$cccc++;
    }
    for (\$j=0; \$j<=\$cccc-1; \$j++)
    {
        \$aaaa++;
        echo "\<tr>\<td>\<center>\".\$edad[\$j].\" \</center>\</td>\</tr>\";
        echo "\<td>\<center>\".\$nume[\$j].\" \</center>\</td>\</tr>\";
        echo "\<td>\<center>\".number_format(((\$nume[\$j]/\$tota)*100),
        1, ',', ' ').\"%\" \</center>\</td>\</tr>\";
        \$xporx[\$j]= round( (\$nume[\$j]/\$tota)*100, 1);
        if (\$j==0)
        {
            echo "\<td rowspan=\".\$cccc.\">\<center>\".\$tota.\"
            \</center>\</td>\</tr>\";
        }
    }
    ?>
</table>
</center>
<p>\</p>
<?php
echo "\<center>\<div id='piechart5'>\</div>\</center>\";
?>
<center>
<p>\</p>
<button onClick="exportTableToExcel('tabla', 'Edades Usuarios
$periodo')\"> Exportar a Excel</button>
<p>\</p>
</center>
<?php
break;
case 6:
echo "\<center>\<h1>Beneficio 1</h1>\</center>\";
echo "\<center>\<table id='tabla'>\<thead>\<div>
<th>\<center>BENEFICIO</center>\</th>
<th>\<center>CANTIDAD</center>\</th>
<th>\<center>PORCENTAJE</center>\</th>
<th>\<center>TOTAL</center>\</th>\</div>\</thead>\";

```

```

\${sqli} = "select BENEFICIO1, count(*) as $tabla from $tabla group by
BENEFICIO1;";
\${cc} = 0;
\${aa} = 0;
\${totale} = 0;
\${resulta} = mysqli_query(\${conn}, \${sqli});
if(!\${resulta} && !\${numeros} )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\${row} = mysqli_fetch_array(\${resulta}, MYSQLI_ASSOC))
{
    \${beneficio}[\${cc}] = \${row}["BENEFICIO1"];
    \${numeros}[\${cc}] = \${row}["$tabla"];
    \${totale} = \${totale} + \${row}["$tabla"];
    \${cc}++;
}
for (\${j}=0;\${j}<=\${cc}-1;\${j}++)
{
    \${aa}++;
    echo "<tr><td><center>\".\${beneficio}[\${j}].\"</center></td>\";
    echo "<td><center>\".\${numeros}[\${j}].\"</center></td>\";
    echo "<td><center>
    \".number_format(((\${numeros}[\${j}]/\${totale})*100), 1, ',', '').\"%
    </center></td>\";
    \${porx}[\${j}] = round( (\${numeros}[\${j}]/\${totale})*100, 1);
    if (\${j}==0)
    {
        echo "<td rowspan=\".\${cc}.\"><center>\".\${totale}.\\"
        </center></td></tr>\";
    }
}
?>
</table>
</center>
<p></p>
<?php
echo "<center><div id='piechart6'></div></center>\";
?>
<center>
<p></p>
<button onClick="exportTableToExcel('$tabla', 'Beneficio 1
    Usuarios $periodo')\"> Exportar a Excel</button>
<p></p>

```

```

</center>
<?php
break;
case 7:
echo \"<center><h1>Estado civil</h1></center>\";
echo \"<center><table id='tabla'><thead><div>
<th><center>ESTADO</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\";
\\$sqli = \"select ESTADOCIVIL, count(*) as $tabla from $tabla group by
ESTADOCIVIL;\";
\\$cc = 0;
\\$aa = 0;
\\$totale = 0;
\\$resulta = mysqli_query(\\$conn, \\$sqli);
if(!\\$resulta && !\\$numeros )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\\$row = mysqli_fetch_array(\\$resulta, MYSQLI_ASSOC))
{
    \\$estado[\\$cc] = \\$row[\"ESTADOCIVIL\"];
    \\$numeros[\\$cc] = \\$row[\"$tabla\"];
    \\$totale = \\$totale + \\$row[\"$tabla\"];
    \\$cc++;
}
for (\\$j=0;\\$j<=\\$cc-1;\\$j++)
{
    \\$aa++;
    echo \"<tr><td><center>\".\\$estado[\\$j].\"</center></td>\";
    echo \"<td><center>\".\\$numeros[\\$j].\"</center></td>\";
    echo \"<td><center>
    \".number_format(((\\$numeros[\\$j]/\\$totale)*100), 1, ',', \"\").\"%
    </center></td>\";
    \\$porx[\\$j]= round( (\\$numeros[\\$j]/\\$totale)*100, 1);
    if (\\$j==0)
    {
        echo \"<td rowspan=\\\".\\$cc.\\\"><center>\".\\$totale.\"
        </center></td></tr>\";
    }
}
?>
</table>

```

```

</center>
<p></p>
<?php
echo \"<center><div id='piechart7'></div></center>\";
?>
<center>
<p></p>
        <button onClick=\"exportTableToExcel('tabla', 'Estado Civil
        Usuarios $periodo')\"> Exportar a Excel</button>
<p></p>
</center>
<?php
break;

```

case 8:

```

echo \"<center><h1>Antigüedad Sisben</h1></center>\";
echo \"<center><table id='tabla'><thead><div>
<th><center>ANTIGÜEDAD</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\";
\\$sqli = \"select ANTIGSISBEN, count(*) as $tabla from $tabla group by
ANTIGSISBEN;\";
\\$cc = 0;
\\$aa = 0;
\\$totale = 0;
\\$resulta = mysqli_query(\\$conn, \\$sqli);
if(!\\$resulta && !\\$numeros )
{
        die('Ocurrió un error al obtener los valores de la base de datos: ' .
        mysqli_error());
}
while(\\$row = mysqli_fetch_array(\\$resulta, MYSQLI_ASSOC))
{
        \\$antig[\\$cc] = \\$row[\"ANTIGSISBEN\"];
        \\$numeros[\\$cc] = \\$row[\"$tabla\"];
        \\$totale = \\$totale + \\$row[\"$tabla\"];
        \\$cc++;
}
for (\\$j=0;\\$j<=\\$cc-1;\\$j++)
{
        \\$aa++;
        echo \"<tr><td><center>\".\\$antig[\\$j].\"</center></td>\";
        echo \"<td><center>\".\\$numeros[\\$j].\"</center></td>\";
        echo \"<td><center>
        \".number_format(((\\$numeros[\\$j]/\\$totale)*100), 1, ',', \").\">%

```



```

</center></td>\";
\${porx[\$j]}= round( (\$numeros[\$j]/\$totale)*100, 1);
if (\$j==0)
{
    echo \"<td rowspan=\".\$cc.\"><center>\".\$totale.\"
    </center></td></tr>\";
}
}
?>
</table>
</center>
<p></p>
<?php
echo \"<center><div id='piechart8'></div></center>\";
?>
<center>
<p></p>
    <button onClick=\"exportTableToExcel('tabla', 'Antiguedad Sisben
    Usuarios $periodo')\"> Exportar a Excel</button>
<p></p>
</center>
<?php
break;
case 9:
echo \"<center><h1>Trabaja actualmente</h1><center>\";
echo \"<center><table id='tabla'><thead><div>
<th><center>TRABAJA</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\";
\${sqli} = \"select TRBAJAACTUAL, count(*) as $tabla from $tabla group by
TRBAJAACTUAL;\";
\${cc} = 0;
\${aa} = 0;
\${totale} = 0;
\${resulta} = mysqli_query(\${conn}, \${sqli});
if(!\${resulta} && !\${numeros} )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\${row} = mysqli_fetch_array(\${resulta}, MYSQLI_ASSOC))
{
    \${trabaja[\${cc}]} = \${row}[\\"TRBAJAACTUAL\"];
    \${numeros}[\${cc}] = \${row}[\\"$tabla\"];

```

```

        \${totale} = \${totale} + \${row}["$tabla"];
        \${cc}++;
    }
    for (\${j}=0;\${j}<=\${cc}-1;\${j}++)
    {
        \${aa}++;
        echo "\<tr>\<td>\<center>\".\${strabaja}[\${j}].\"</center>\</td>\</tr>\";
        echo "\<td>\<center>\".\${numeros}[\${j}].\"</center>\</td>\</tr>\";
        echo "\<td>\<center>
        \".number_format(((\${numeros}[\${j}]/\${totale})*100), 1, ',', ").\"%
        </center>\</td>\</tr>\";
        \${porx}[\${j}] = round( (\${numeros}[\${j}]/\${totale})*100, 1);
        if (\${j}==0)
        {
            echo "\<td rowspan=\".\${cc}.\">\<center>\".\${totale}.\"
            </center>\</td>\</tr>\";
        }
    }
    ?>
</table>
</center>
<p></p>
<?php
echo "\<center>\<div id='piechart9'>\</div>\</center>\";
?>
<center>
<p></p>
    <button onClick=\"exportTableToExcel('$tabla', 'Usuarios que
    trabajan $periodo')\"> Exportar a Excel</button>
<p></p>
</center>
<?php
break;
case 10:
echo "\<center>\<h1>Tipo de vivienda</h1>\</center>\";
echo "\<center>\<table id='tabla'>\<thead>\<div>
<th>\<center>VIVIENDA</center>\</th>
<th>\<center>CANTIDAD</center>\</th>
<th>\<center>PORCENTAJE</center>\</th>
<th>\<center>TOTAL</center>\</th>\</div>\</thead>\";
\${sqli} = \"select VIVIENDA, count(*) as $tabla from $tabla group by
VIVIENDA;\";
\${cc} = 0;
\${aa} = 0;
\${totale} = 0;

```

```

\ $resulta = mysqli_query(\ $conn, \ $sqli);
if(!\ $resulta && !\ $numeros )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\ $row = mysqli_fetch_array(\ $resulta, MYSQLI_ASSOC))
{
    \ $vivienda[\ $cc] = \ $row["VIVIENDA"];
    \ $numeros[\ $cc] = \ $row["$tabla"];
    \ $totale = \ $totale + \ $row["$tabla"];
    \ $cc++;
}
for (\ $j=0;\ $j<=\ $cc-1;\ $j++)
{
    \ $aa++;
    echo \ "<tr><td><center>\ ".$vivienda[\ $j].\ "</center></td>\ ";
    echo \ "<td><center>\ ".$numeros[\ $j].\ "</center></td>\ ";
    echo \ "<td><center>
    \ ".number_format(((\ $numeros[\ $j]/\ $totale)*100), 1, ',', ").\ "%
    </center></td>\ ";
    \ $porx[\ $j]= round( (\ $numeros[\ $j]/\ $totale)*100, 1);
    if (\ $j==0)
    {
        echo \ "<td rowspan=\ ".$cc.\ "><center>\ ".$totale.\
        </center></td></tr>\ ";
    }
}
?>
</table>
</center>
<p></p>
<?php
echo \ "<center><div id='piechart10'></div></center>\ ";
?>
<center>
<p></p>
<button onClick=\ "exportTableToExcel('tabla', 'Tipo de Vivienda
Usuarios $periodo')\ "> Exportar a Excel</button>
<p></p>
</center>
<?php
break;
case 11:
echo \ "<center><h1>Estrato</h1></center>\ ";

```

```

echo \"<center><table id='tabla'><thead><div>
<th><center>ESTRATO</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\";
\\$sqli = \"select ESTRATO, count(*) as $tabla from $tabla group by
ESTRATO;\";
\\$cc = 0;
\\$aa = 0;
\\$totale = 0;
\\$resulta = mysqli_query(\\$conn, \\$sqli);
if(!\\$resulta && !\\$numeros )
{
    die('Ocurrió un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\\$row = mysqli_fetch_array(\\$resulta, MYSQLI_ASSOC))
{
    \\$vivienda[\\$cc] = \\$row[\"ESTRATO\"];
    \\$numeros[\\$cc] = \\$row[\"$tabla\"];
    \\$totale = \\$totale + \\$row[\"$tabla\"];
    \\$cc++;
}
for (\\$j=0;\\$j<=\\$cc-1;\\$j++)
{
    \\$aa++;
    echo \"<tr><td><center>\\\".\\$vivienda[\\$j].\\\"</center></td>\\\";
    echo \"<td><center>\\\".\\$numeros[\\$j].\\\"</center></td>\\\";
    echo \"<td><center>
\\\".number_format(((\\$numeros[\\$j]/\\$totale)*100), 1, ',', '\\\".\\\"%
</center></td>\\\";
    \\$porx[\\$j]= round( (\\$numeros[\\$j]/\\$totale)*100, 1);
    if (\\$j==0)
    {
        echo \"<td rowspan=\\\".\\$cc.\\\"><center>\\\".\\$totale.\\\"
        </center></td></tr>\\\";
    }
}
?>
</table>
</center>
<p></p>
<?php
echo \"<center><div id='piechart11'></div></center>\\\";
?>

```

```

<center>
<p></p>
    <button onClick="exportTableToExcel('tabla', 'Estrato Usuarios
    $periodo')"> Exportar a Excel</button>
<p></p>
</center>
<?php
break;
case 12:
echo "\"<center><h1>Beneficios diferentes a FOEST</h1><center>\"";
echo "\"<center><table id='tabla'><thead><div>
<th><center>BENEFICIOS</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\"";
\\$sqli = \"select ROTROBENE, count(*) as $tabla from $tabla group by
ROTROBENE;\";
\\$cc = 0;
\\$aa = 0;
\\$totale = 0;
\\$resulta = mysqli_query(\\$conn, \\$sqli);
if(!\\$resulta && !\\$numeros )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\\$row = mysqli_fetch_array(\\$resulta, MYSQLI_ASSOC))
{
    \\$rotrobene[\\$cc] = \\$row[\"ROTROBENE\"];
    \\$numeros[\\$cc] = \\$row[\"$tabla\"];
    \\$totale = \\$totale + \\$row[\"$tabla\"];
    \\$cc++;
}
for (\\$j=0;\\$j<=\\$cc-1;\\$j++)
{
    \\$aa++;
    echo \"<tr><td><center>\\$.\\$rotrobene[\\$j].\\\"</center></td>\\\";
    echo \"<td><center>\\$.\\$numeros[\\$j].\\\"</center></td>\\\";
    echo \"<td><center>
\\$.number_format(((\\$numeros[\\$j]/\\$totale)*100), 1, ',', '\\').\\\"%
</center></td>\\\";
    \\$porx[\\$j]= round( (\\$numeros[\\$j]/\\$totale)*100, 1);
    if (\\$j==0)
    {
        echo \"<td rowspan=\\$.\\$cc.\\\"><center>\\$.\\$totale.\\\"

```

```

        </center></td></tr>\";
    }
}
?>
</table>
</center>
<p></p>
<?php
echo \"<center><div id='piechart12'></div></center>\";
?>
<center>
<p></p>
        <button onClick=\"exportTableToExcel('tabla', 'Beneficios
        Diferentes a FOEST Usuarios $periodo')\"> Exportar a
        Excel</button>
<p></p>
</center>
<?php
break;
case 13:
echo \"<center><h1>Universidades</h1></center>\";
echo \"<center><table id='tabla'><thead><div>
<th><center>UNIVERSIDAD</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\";
\$sqli = \"select UNIVERSIDAD, count(*) as $tabla from $tabla group by
UNIVERSIDAD;\";
\$cc = 0;
\$aa = 0;
\$totale = 0;
\$resulta = mysqli_query(\$conn, \$sqli);
if(!\$resulta && !\$numeros )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\$row = mysqli_fetch_array(\$resulta, MYSQLI_ASSOC))
{
    \$universidad[\$cc] = \$row[\"UNIVERSIDAD\"];
    \$numeros[\$cc] = \$row[\"$tabla\"];
    \$totale = \$totale + \$row[\"$tabla\"];
    \$cc++;
}
for (\$j=0; \$j<=\$cc-1; \$j++)

```

```

    {
        \${aa}++;
        echo \<tr><td><center>\$. \$universidad[\$j].\</center></td>\</tr>";
        echo \<td><center>\$. \$numeros[\$j].\</center></td>\</tr>";
        echo \<td><center>
            \$. number_format(((\$. \$numeros[\$j]/\$totale)*100), 1, ',', '').\</center></td>\</tr>";
        \${porx}[\$j]= round( (\$. \$numeros[\$j]/\$totale)*100, 1);
        if (\${j}==0)
        {
            echo \<td rowspan=\$. \$cc.\><center>\$. \$totale.\</center></td></tr>\</tr>";
        }
    }
}
?>
</table>
</center>
<p></p>
<?php
echo \<center><div id='piechart13'></div></center>\</tr>";
?>
<center>
<p></p>
<button onClick=\<exportTableToExcel('tabla', 'Universidades
    Usuarios \$periodo')\> Exportar a Excel</button>
<p></p>
</center>
<?php
break;
case 14:
echo \<center><h1>Nivel</h1></center>\</tr>";
echo \<center><table id='tabla'><thead><div>
<th><center>NIVEL</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\</tr>";
\${sql} = \<select NIVEL, count(*) as \$tabla from \$tabla group by NIVEL;\</tr>";
\${cc} = 0;
\${aa} = 0;
\${totale} = 0;
\${resulta} = mysqli_query(\${conn}, \${sql});
if(!\${resulta} && !\${numeros} )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
        mysqli_error());
}

```

```

    }
    while(\$row = mysqli_fetch_array(\$resulta, MYSQLI_ASSOC))
    {
        \$nivel[\$cc] = \$row["NIVEL"];
        \$numeros[\$cc] = \$row["$tabla"];
        \$totale = \$totale + \$row["$tabla"];
        \$cc++;
    }
    for (\$j=0;\$j<=\$cc-1;\$j++)
    {
        \$aa++;
        echo "<tr><td><center>\".\$nivel[\$j].\"</center></td>\";
        echo "<td><center>\".\$numeros[\$j].\"</center></td>\";
        echo "<td><center>
        \".number_format(((\$numeros[\$j]/\$totale)*100), 1, ',', '').\"%
        </center></td>\";
        \$porx[\$j]= round( (\$numeros[\$j]/\$totale)*100, 1);
        if (\$j==0)
        {
            echo "<td rowspan=\".\$cc.\"><center>\".\$totale.\"
            </center></td></tr>\";
        }
    }
    ?>
</table>
</center>
<p></p>
<?php
echo "<center><div id='piechart14'></div></center>\";
?>
<center>
<p></p>
    <button onClick="exportTableToExcel('tabla', 'Nivel de estudios
    Usuarios $periodo')\"> Exportar a Excel</button>
<p></p>
</center>
<?php
break;
case 15:
echo "<center><h1>Carrera</h1></center>\";
echo "<center><table id='tabla'><thead><div>
<th><center>CARRERA</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\";

```



```

\${sqli} = "select CARRERA, count(*) as $tabla from $tabla group by
CARRERA;";
\${cc} = 0;
\${aa} = 0;
\${totale} = 0;
\${resulta} = mysqli_query(\${conn}, \${sqli});
if(!\${resulta} && !\${numeros} )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\${row} = mysqli_fetch_array(\${resulta}, MYSQLI_ASSOC))
{
    \${scarrera}[\${cc}] = \${row}["CARRERA"];
    \${numeros}[\${cc}] = \${row}["$tabla"];
    \${totale} = \${totale} + \${row}["$tabla"];
    \${cc}++;
}
for (\${j}=0;\${j}<=\${cc}-1;\${j}++)
{
    \${aa}++;
    echo "<tr><td><center>\".\${scarrera}[\${j}].\"</center></td>\";
    echo "<td><center>\".\${numeros}[\${j}].\"</center></td>\";
    echo "<td><center>
    \".number_format(((\${numeros}[\${j}]/\${totale})*100), 1, ',', ')\".%
    </center></td>\";
    \${porx}[\${j}] = round( (\${numeros}[\${j}]/\${totale})*100, 1);
    if (\${j}==0)
    {
        echo "<td rowspan=\".\${cc}.\"><center>\".\${totale}.\"
        </center></td></tr>\";
    }
}
?>
</table>
</center>
<p></p>
<?php
    echo "<center><div id='piechart15'></div></center>\";
?>
<center>
<p></p>
    <button onClick=\"exportTableToExcel('$tabla', 'Carreras Usuarios
    $periodo')\"> Exportar a Excel</button>
<p></p>

```

```

</center>
<?php
break;
case 16:
echo \"<center><h1>Modalidad</h1><center>\";
echo \"<center><table id='tabla'><thead><div>
<th><center>MODALIDAD</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\";
\\$sqli = \"select MODALIDAD, count(*) as $tabla from $tabla group by
MODALIDAD;\";
\\$cc = 0;
\\$aa = 0;
\\$totale = 0;
\\$resulta = mysqli_query(\\$conn, \\$sqli);
if(!\\$resulta && !\\$numeros )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\\$row = mysqli_fetch_array(\\$resulta, MYSQLI_ASSOC))
{
    \\$modalidad[\\$cc] = \\$row[\"MODALIDAD\"];
    \\$numeros[\\$cc] = \\$row[\"$tabla\"];
    \\$totale = \\$totale + \\$row[\"$tabla\"];
    \\$cc++;
}
for (\\$j=0;\\$j<=\\$cc-1;\\$j++)
{
    \\$aa++;
    echo \"<tr><td><center>\".\\$modalidad[\\$j].\"</center></td>\";
    echo \"<td><center>\".\\$numeros[\\$j].\"</center></td>\";
    echo \"<td><center>
    \".number_format(((\\$numeros[\\$j]/\\$totale)*100), 1, ',', \"\").\"%
    </center></td>\";
    \\$porx[\\$j]= round( (\\$numeros[\\$j]/\\$totale)*100, 1);
    if (\\$j==0)
    {
        echo \"<td rowspan='\".\\$cc.\"><center>\".\\$totale.\"
        </center></td></tr>\";
    }
}
?>
</table>

```

```

</center>
<p></p>
<?php
echo \"<center><div id='piechart16'></div></center>\";
?>
<center>
<p></p>
<button onClick=\"exportTableToExcel('tabla', 'Modalidad de
estudios Usuarios $periodo')\"> Exportar a Excel</button>
<p></p>
</center>
<?php
break;

```

case 17:

```

echo \"<center><h1>Semestre</h1></center>\";
echo \"<center><table id='tabla'><thead><div>
<th><center>SEMESTRE</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\";
\\$sqli = \"select SEMESTRE, count(*) as $tabla from $tabla group by
SEMESTRE;\";
\\$cc = 0;
\\$aa = 0;
\\$totale = 0;
\\$resulta = mysqli_query(\\$conn, \\$sqli);
if(!\\$resulta && !\\$numeros )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\\$row = mysqli_fetch_array(\\$resulta, MYSQLI_ASSOC))
{
    \\$semestre[\\$cc] = \\$row[\"SEMESTRE\"];
    \\$numeros[\\$cc] = \\$row[\"$tabla\"];
    \\$totale = \\$totale + \\$row[\"$tabla\"];
    \\$cc++;
}
for (\\$j=0;\\$j<=\\$cc-1;\\$j++)
{
    \\$aa++;
    echo \"<tr><td><center>\".\\$semestre[\\$j].\"</center></td>\";
    echo \"<td><center>\".\\$numeros[\\$j].\"</center></td>\";
    echo \"<td><center>
    \".number_format(((\\$numeros[\\$j]/\\$totale)*100), 1, ',', \").\">%

```

```

</center></td>\";
\${porx[\$j]}= round( (\$numeros[\$j]/\$totale)*100, 1);
if (\$j==0)
{
    echo \"<td rowspan=\".\$cc.\"><center>\".\$totale.\"
    </center></td></tr>\";
}
}
?>
</table>
</center>
<p></p>
<?php

```

```

    echo \"<center><div id='piechart17'></div></center>\";
?>
<center>
<p></p>
    <button onClick=\"exportTableToExcel('tabla', 'semestre que
    cursan Usuarios $periodo')\"> Exportar a Excel</button>
<p></p>
</center>
<?php
break;
case 18:
    echo \"<center><h1>Dependencia horas sociales 1</h1></center>\";
    echo \"<center><table id='tabla'><thead><div>
    <th><center>DEPENDENCIA</center></th>
    <th><center>CANTIDAD</center></th>
    <th><center>PORCENTAJE</center></th>
    <th><center>TOTAL</center></th></div></thead>\";
    \$sqli = \"select DEPENDENCIA1, count(*) as $tabla from $tabla group by
    DEPENDENCIA1;\";
    \$cc = 0;
    \$aa = 0;
    \$totale = 0;
    \$resulta = mysqli_query(\$conn, \$sqli);
    if(!\$resulta && !\$numeros )
    {
        die('Ocurrio un error al obtener los valores de la base de datos: ' .
        mysqli_error());
    }
    while(\$row = mysqli_fetch_array(\$resulta, MYSQLI_ASSOC))
    {
        \$dep1[\$cc] = \$row[\"DEPENDENCIA1\"];
        \$numeros[\$cc] = \$row[\"$tabla\"];
    }
}

```

```

        \ $totale = \ $totale + \ $row["$tabla"];
        \ $cc++;
    }
    for (\ $j=0; \ $j<=\ $cc-1; \ $j++)
    {
        \ $aa++;
        echo \ "<tr><td><center>\ ".$dep1[\ $j].\ "</center></td>\ ";
        echo \ "<td><center>\ ".$numeros[\ $j].\ "</center></td>\ ";
        echo \ "<td><center>
        \ ".number_format(((\ $numeros[\ $j]/\ $totale)*100), 1, ',', "").\ "%
        </center></td>\ ";
        \ $porx[\ $j]= round( (\ $numeros[\ $j]/\ $totale)*100, 1);
        if (\ $j==0)
        {
            echo \ "<td rowspan=\ ".$cc.\ "><center>\ ".$totale.\
            </center></td></tr>\ ";
        }
    }
    ?>
</table>
</center>
<p></p>
<?php
echo \ "<center><div id='piechart18'></div></center>\ ";
?>
<center>
<p></p>
    <button onClick="exportTableToExcel('tabla', 'Dependencia 1
    Usuarios $periodo')\ "> Exportar a Excel</button>
<p></p>
</center>
<?php
break;

```

case 19:

```

echo \ "<center><h1>Dependencia horas sociales 2</h1></center>\ ";
echo \ "<center><table id='tabla'><thead><div>
<th><center>DEPENDENCIA</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\ ";
\ $sqli = \ "select DEPENDENCIA2, count(*) as $tabla from $tabla group by
DEPENDENCIA2;\ ";
\ $cc = 0;
\ $aa = 0;

```

```

\ $totale = 0;
\ $resulta = mysqli_query(\ $conn, \ $sqli);
if(!\ $resulta && !\ $numeros )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
mysqli_error());
}
while(\ $row = mysqli_fetch_array(\ $resulta, MYSQLI_ASSOC))
{
    \ $dep2[\ $cc] = \ $row["DEPENDENCIA2"];
    \ $numeros[\ $cc] = \ $row["$tabla"];
    \ $totale = \ $totale + \ $row["$tabla"];
    \ $cc++;
}
for (\ $j=0;\ $j<=\ $cc-1;\ $j++)
{
    \ $aa++;
    echo \ "<tr><td><center>\ ".$dep2[\ $j].\ "</center></td>\ ";
    echo \ "<td><center>\ ".$numeros[\ $j].\ "</center></td>\ ";
    echo \ "<td><center>
\ .number_format(((\ $numeros[\ $j]/\ $totale)*100), 1, ',', ").\ "%
</center></td>\ ";
    \ $porx[\ $j]= round( (\ $numeros[\ $j]/\ $totale)*100, 1);
    if (\ $j==0)
    {
        echo \ "<td rowspan=\ ".\ $cc.\ "><center>\ ".$totale.\ "
</center></td></tr>\ ";
    }
}
?>
</table>
</center>
<p></p>
<?php
echo \ "<center><div id='piechart19'></div></center>\ ";
?>
<center>
<p></p>
<button onClick=\ "exportTableToExcel('tabla', 'Dependencia 2
Usuarios $periodo')\ "> Exportar a Excel</button>
<p></p>
</center>
<?php
break;

```

case 20:

```

echo \"<center><h1>Dependencia horas sociales 3</h1><center>\";
echo \"<center><table id='tabla'><thead><div>
<th><center>DEPENDENCIA</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\";
\${sqli} = \"select DEPENDENCIA3, count(*) as $tabla from $tabla group by
DEPENDENCIA3;\";
\${cc} = 0;
\${aa} = 0;
\${totale} = 0;
\${resulta} = mysqli_query(\${conn}, \${sqli});
if(!\${resulta} && !\${numeros} )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\${row} = mysqli_fetch_array(\${resulta}, MYSQLI_ASSOC))
{
    \${dep3}[\${cc}] = \${row}["DEPENDENCIA3"];
    \${numeros}[\${cc}] = \${row}["$tabla"];
    \${totale} = \${totale} + \${row}["$tabla"];
    \${cc}++;
}
for (\${j}=0;\${j}<=\${cc}-1;\${j}++)
{
    \${aa}++;
    echo \"<tr><td><center>\".\${dep3}[\${j}].\"</center></td>\";
    echo \"<td><center>\".\${numeros}[\${j}].\"</center></td>\";
    echo \"<td><center>
    \".number_format(((\${numeros}[\${j}]/\${totale})*100), 1, ',', '').\"%
    </center></td>\";
    \${porx}[\${j}] = round( (\${numeros}[\${j}]/\${totale})*100, 1);
    if (\${j}==0)
    {
        echo \"<td rowspan='\".\${cc}.\"><center>\".\${totale}.\
        </center></td></tr>\";
    }
}
?>
</table>
</center>
<p></p>
<?php
echo \"<center><div id='piechart20'></div></center>\";

```

```

?>
<center>
<p></p>
<button onClick="exportTableToExcel('tabla', 'Dependencias 3
Usuarios $periodo')"> Exportar a Excel</button>
<p></p>
</center>
<?php
break;
case 21:
echo "\"<center><h1>Ciudad a la que viajan</h1><center>\"";
echo "\"<center><table id='tabla'><thead><div>
<th><center>CIUDAD</center></th>
<th><center>CANTIDAD</center></th>
<th><center>PORCENTAJE</center></th>
<th><center>TOTAL</center></th></div></thead>\"";
\\$sqli = \"select CIUDADVIAJA, count(*) as $tabla from $tabla group by
CIUDADVIAJA;\";
\\$cc = 0;
\\$aa = 0;
\\$totale = 0;
\\$resulta = mysqli_query(\\$conn, \\$sqli);
if(!\\$resulta && !\\$numeros )
{
    die('Ocurrio un error al obtener los valores de la base de datos: ' .
    mysqli_error());
}
while(\\$row = mysqli_fetch_array(\\$resulta, MYSQLI_ASSOC))
{
    \\$ciudad[\\$cc] = \\$row[\"CIUDADVIAJA\"];
    \\$numeros[\\$cc] = \\$row[\"$tabla\"];
    \\$totale = \\$totale + \\$row[\"$tabla\"];
    \\$cc++;
}
for (\\$j=0;\\$j<=\\$cc-1;\\$j++)
{
    \\$aa++;
    echo \"<tr><td><center>\".\\$ciudad[\\$j].\"</center></td>\";
    echo \"<td><center>\".\\$numeros[\\$j].\"</center></td>\";
    echo \"<td><center>
    \\$.number_format(((\\$numeros[\\$j]/\\$totale)*100), 1, ',', '').\"%
    </center></td>\";
    \\$porx[\\$j]= round( (\\$numeros[\\$j]/\\$totale)*100, 1);
    if (\\$j==0)
    {

```



```
        echo \"<td rowspan=\\\".\$cc.\\\"><center>\\\".\$totale.\\\"
        </center></td></tr>\\\";
    }
}
?>
</table>
</center>
<p></p>
<?php
echo \\\"<center><div id='piechart21'></div></center>\\\";
?>
<center>
<p></p>
<button onClick=\\\"exportTableToExcel('tabla', 'Ciudad a la que
viajan Usuarios $periodo')\\\"> Exportar a Excel</button>
<p></p>
</center>
<?php
break;
}
}
?>
</center>
</form>
</body>
</html>\\\";
}
if ($sesta = fopen($sesta, \"a\")
{
    if (fwrite($sesta, \"$me\"))
    {
    }
    else
    {
        ?>
        <script type=\\\"text/javascript\\\">
        alert(\"Ha habido un problema panestadistica.php\");
        window.location.href='index.php';
        </script>
        <?php
    }
    fclose($sesta);
}
?>
```

9.1.15 Archivo que toma las variables y las envía a la base de datos (para periodos nuevos)

```
<?php
    $crear = "creg$periodo.php";
    if(file_exists ($crear))
    {
    }
    else
    {
        $m = "<?php
        include (\"blodese.php\");
?>
<!doctype html>
<html>
<head>
<meta charset=\"utf-8\">
<title>Untitled Document</title>
</head>
<body>
    <?php
    if (isset(\$_POST[\"idsu\"]))
    {
        \$id = \$_POST[\"idsu\"];
        \$item = \$_POST[\"item\"];
        \$carpeta = \$_POST[\"carpeta\"];
        \$radica = \$_POST[\"radica\"];
        \$bene1 = \$_POST[\"bene1\"];
        \$bene2 = \$_POST[\"bene2\"];
        \$vez1 = \$_POST[\"1vez\"];
        \$rn = \$_POST[\"rn\"];
        \$rei = \$_POST[\"rei\"];
        \$apellido1 = \$_POST[\"1apellido\"];
        \$apellido2 = \$_POST[\"2apellido\"];
        \$nombre1 = \$_POST[\"1nombre\"];
        \$nombre2 = \$_POST[\"2nombre\"];
        \$tdocu = \$_POST[\"tdocu\"];
        \$ndocumento = \$_POST[\"ndocumento\"];
        \$expedido = \$_POST[\"expedido\"];
        \$nacimient = \$_POST[\"nacimient\"];
        \$edadus = \$_POST[\"edadus\"];
        \$estadoc = \$_POST[\"estadoc\"];
        \$eps = \$_POST[\"eps\"];
        \$estadoeps = \$_POST[\"estadoeps\"];
    }
}
```

\\$pespe = \\$_POST["pespe"];
\\$gene = \\$_POST["gene"];
\\$telefono1 = \\$_POST["tel1"];
\\$telefono2 = \\$_POST["tel2"];
\\$direccion = \\$_POST["direccion"];
\\$sector = \\$_POST["sector"];
\\$correoelec = \\$_POST["correo"];
\\$correoele2 = \\$_POST["correo2"];
\\$antiguedad = \\$_POST["asisben"];
\\$puntajesis = \\$_POST["psisben"];
\\$trabaja = \\$_POST["trabaja"];
\\$sectorlab = \\$_POST["sectorlab"];
\\$empresa = \\$_POST["empresa"];
\\$cargo = \\$_POST["cargo"];
\\$vivienda = \\$_POST["vivienda"];
\\$vive = \\$_POST["vive"];
\\$estrato = \\$_POST["estrato"];
\\$personasac = \\$_POST["personasacar"];
\\$nucleo = \\$_POST["nucleo"];
\\$nombrecon = \\$_POST["nombrecon"];
\\$cc = \\$_POST["cc"];
\\$telcon = \\$_POST["telcon"];
\\$ocupacionc = \\$_POST["ocupacionc"];
\\$parentesco = \\$_POST["parentesco"];
\\$correocon = \\$_POST["correocon"];
\\$nombrerep = \\$_POST["nombrerep"];
\\$ccrep = \\$_POST["ccrep"];
\\$expedidore = \\$_POST["expedidorep"];
\\$telrep = \\$_POST["Telrep"];
\\$dirrep = \\$_POST["Dirrep"];
\\$parenterep = \\$_POST["Paretescorep"];
\\$institucio = \\$_POST["institucion"];
\\$grado = \\$_POST["grado"];
\\$icfes = \\$_POST["icfes"];
\\$preseicfes = \\$_POST["presentaicfes"];
\\$notas11 = \\$_POST["notas11"];
\\$puesto11 = \\$_POST["puesto11"];
\\$alotrobene = \\$_POST["robene"];
\\$universida = \\$_POST["univer"];
\\$sede = \\$_POST["sede"];
\\$nivel = \\$_POST["niv"];
\\$carrera = \\$_POST["carrera"];
\\$modalidad = \\$_POST["modalidad"];
\\$semestre = \\$_POST["semestre"];
\\$horario = \\$_POST["horario"];

```
\$snies = \$_POST["snies"];
\$continuaci = \$_POST["continuacion"];
\$vasemestre = \$_POST["vsemestre"];
\$porcentaje = \$_POST["porcentaje"];
\$montoaprov = \$_POST["monto"];
\$otrobenefi = \$_POST["obeneficio"];
\$maperdidas = \$_POST["perdidas"];
\$promeditra = \$_POST["promediotra"];
\$promediant = \$_POST["promedioant"];
\$bene1ant = \$_POST["bene1ant"];
\$bene2ant = \$_POST["bene2ant"];
\$horascumpl = \$_POST["hscump"];
\$dependenc1 = \$_POST["dep1"];
\$dependenc2 = \$_POST["dep2"];
\$dependenc3 = \$_POST["dep3"];
\$nocuenta = \$_POST["nocuenta"];
\$diasviaja = \$_POST["dias"];
\$ciudadvia = \$_POST["ciudad"];
\$totaldias = \$_POST["tdias"];
\$valor2rec = \$_POST["val2r"];
\$valorseman = \$_POST["vals"];
\$valormes = \$_POST["valm"];
\$valormespa = \$_POST["valp"];
\$mesagirar = \$_POST["mesgi"];
\$totalsemes = \$_POST["tsemestre"];
\$ratificado = \$_POST["ratificado"];
\$estado = \$_POST["estado"];
\$observacion = \$_POST["observacion"];
\$porcentjeapli = \$_porcentaje.\ "%\ %";
```

```
include ("conexion.php");
```

```
\$con = mysqli_query(\$con, "INSERT INTO $tabla (ID_SUB, ITEM, NOCARPETA,
FECHARADICACION, BENEFICIO1, BENEFICIO2, 1VEZ, RN, REI, PAPELLIDO,
SAPELLIDO, PNOMBRE, SNOMBRE, ID_US, NODOCUMENTO, EXPEDIDOEN,
FECHANACIMIENTO, EDAD, ESTADOCIVIL, NEMBREEPS, ESTADOEPS,
POESPECIAL, GENERO, TEL1, TEL2, DIRECCION, SECTOR, CORREOE1,
CORREOE2, ANTIGSISBEN, PUNTASISBEN, TRBAJAACtual, SECTORLABO,
NOMEMPRESA, CARGOLAB, VIVIENDA, TIEMPOVIVIENDO, ESTRATO,
PERSONASCARGO, NUCLEOFAMILIAR, NOMBRECONTACTO,
DOCUMENTOCONTACTO, TELCONTACTO, OCUPACIONCONTACTO,
PARENTESCO, CORREOECONTACTO, NOMBREREPRESENTANTE, NOCEDULAR,
EXPEDIDAR, TELREPR, DIRECCIONREPR, PARENTESCOREP,
INSTITUCIONEDUMEDIA, FECHAGRADO, PUNTAJEICFES, ANOPRESENTACION,
PROMEDIOONCE, PUESTOONCE, ROTROBENE, UNIVERSIDAD, SEDE, NIVEL,
CARRERA, MODALIDAD, SEMESTRE, HORARIO, SNIES, CONTINUACARRERA,
```

VALORSEMESTRE, PORCENTAJEA, MONTOAPRO, OTROBENE, MATERIASPER,
 PROMEDIOANTERIOR, PROMEDIOTRAANTERIOR, BENEFICIOSANTERIORES1,
 BENEFICIOSANTERIORES2, HSCUMPLIDAS, DEPENDENCIA1, DEPENDENCIA2,
 DEPENDENCIA3, NOCUENTA, DIASVIAJADOS, CIUDADVIAJA, DIAS, VALORRECOR,
 VALORSEMANA, VALORMESBRUTO, VALORMESPAGAR, MESES,
 TOTALSEMESTRE, RATIFICADO, APROBADO, OBSERVACIONRAT) VALUES ('\\$id',
 '\\$item', '\\$carpeta', '\\$radica', '\\$bene1', '\\$bene2', '\\$vez1', '\\$rn', '\\$rei', '\\$apellido1',
 '\\$apellido2', '\\$nombre1', '\\$nombre2', '\\$tdocu', '\\$ndocumento', '\\$expedido',
 '\\$nacimient', '\\$edadus', '\\$estadoc', '\\$seps', '\\$estadoeps', '\\$spespe', '\\$gene',
 '\\$telefono1', '\\$telefono2', '\\$direccion', '\\$sector', '\\$correoelec', '\\$correoele2',
 '\\$antiguedad', '\\$puntajesis', '\\$trabaja', '\\$sectorlab', '\\$empresa', '\\$carg', '\\$vivienda',
 '\\$vive', '\\$estrato', '\\$personasac', '\\$nucleo', '\\$nombrecon', '\\$cc', '\\$telcon',
 '\\$ocupacionc', '\\$parentesco', '\\$correocon', '\\$nombrepr', '\\$ccrep', '\\$expedidore',
 '\\$telrep', '\\$dirrep', '\\$parenterep', '\\$institucio', '\\$grado', '\\$icfes', '\\$preseicfes',
 '\\$notas11', '\\$puesto11', '\\$alotrobene', '\\$universida', '\\$sede', '\\$nivel', '\\$carrera',
 '\\$modalidad', '\\$semestre', '\\$horario', '\\$snies', '\\$continuaci', '\\$vasemestre',
 '\\$porcentjeapli', '\\$montoaprov', '\\$otrobene', '\\$maperdidas', '\\$promediant',
 '\\$promeditra', '\\$bene1ant', '\\$bene2ant', '\\$horascumpl', '\\$dependenc1', '\\$dependenc2',
 '\\$dependenc3', '\\$nocuenta', '\\$diasviaja', '\\$ciudadvia', '\\$totaldias', '\\$valor2rec',
 '\\$valorseman', '\\$valormes', '\\$valormespa', '\\$mesagiar', '\\$totalsemes', '\\$ratificado',
 '\\$estado', '\\$observacion')\") or die ("no se conecto a la tabla")

```

?>
<script type="text/javascript">
    alert("Estudiante agregado exitosamente a la base de datos");
    window.location.href='$periodo.php';
</script>
<?php
}
else
{
    echo "error";
}
?>
</body>
</html>";
}
if ($crear = fopen($crear, "a"))
{
    if (fwrite($crear, "$m"))
    {
    }
    else
    {
    }
?>
  
```

```

        <script type="text/javascript">
            alert("Ha habido un problema manejadorregistro.php");
            window.location.href='index.php';
        </script>
    <?php
    }
    fclose($crear);
}
?>

```

9.1.16 Archivo que manipula un registro de un periodo, para actualizarlo

```

<?php
    $modifi = "modifica$periodo.php";
    if(file_exists ($modifi))
    {

    }
    else
    {
        $mensajes = "<?php
    include ("blodese.php");
    modificarestudiante(\$_POST["idsu"], \$_POST["item"], \$_POST["carpeta"],
    \$_POST["radical"], \$_POST["bene1"], \$_POST["bene2"], \$_POST["1vez"], \$_POST["rn"],
    \$_POST["rei"], \$_POST["1apellido"], \$_POST["2apellido"], \$_POST["1nombre"],
    \$_POST["2nombre"], \$_POST["tdocu"], \$_POST["ndocumento"], \$_POST["expedido"],
    \$_POST["nacimiento"], \$_POST["edadus"], \$_POST["estadoc"], \$_POST["eps"],
    \$_POST["estadoeps"], \$_POST["pespe"], \$_POST["gene"], \$_POST["tel1"],
    \$_POST["tel2"], \$_POST["direccion"], \$_POST["sector"], \$_POST["correo"],
    \$_POST["correo2"], \$_POST["asisben"], \$_POST["psisben"], \$_POST["trabaja"],
    \$_POST["sectorlab"], \$_POST["empresa"], \$_POST["cargo"], \$_POST["vivienda"],
    \$_POST["vive"], \$_POST["estrato"], \$_POST["personasacar"], \$_POST["nucleo"],
    \$_POST["nombrecon"], \$_POST["cc"], \$_POST["telcon"], \$_POST["ocupacion"],
    \$_POST["parentesco"], \$_POST["correocon"], \$_POST["nombrepr"], \$_POST["ccrep"],
    \$_POST["expedidorep"], \$_POST["Telrep"], \$_POST["Dirrep"], \$_POST["ParetSCOREP"],
    \$_POST["institucion"], \$_POST["grado"], \$_POST["icfes"], \$_POST["presentaicfes"],
    \$_POST["notas11"], \$_POST["puesto11"], \$_POST["robene"], \$_POST["univer"],
    \$_POST["sede"], \$_POST["niv"], \$_POST["carrera"], \$_POST["modalidad"],
    \$_POST["semestre"], \$_POST["horario"], \$_POST["snies"], \$_POST["continuacion"],
    \$_POST["vsemestre"], \$_POST["porcentaje"], \$_POST["monto"], \$_POST["obeneficio"],
    \$_POST["perdidas"], \$_POST["promedioant"], \$_POST["promediotra"],
    \$_POST["bene1ant"], \$_POST["bene2ant"], \$_POST["hscump"], \$_POST["dep1"],
    \$_POST["dep2"], \$_POST["dep3"], \$_POST["nocuenta"], \$_POST["dias"],

```

```

\$_POST["ciudadv"], \$_POST["tdias"], \$_POST["val2r"], \$_POST["vals"],
\$_POST["valm"], \$_POST["valp"], \$_POST["mesgi"], \$_POST["tsemestre"],
\$_POST["ratificado"], \$_POST["estado"], \$_POST["observacion"]);
function modificarestudiante(\$id, \$item, \$carpeta, \$radica, \$bene1, \$bene2, \$vez1, \$rn, \$rei,
\$apellido1, \$apellido2, \$nombre1, \$nombre2, \$tdocu, \$ndocumento, \$expedido, \$nacimiento,
\$edadus, \$estadoc, \$seps, \$estadoeps, \$pespe, \$gene, \$telefono1, \$telefono2, \$direccion,
\$sector, \$correoelec, \$correoele2, \$antiguedad, \$puntajesis, \$trabaja, \$sectorla, \$empresa,
\$cargo, \$vivienda, \$vive, \$estrato, \$personasac, \$nucleo, \$nombrecon, \$cc, \$stelcon,
\$ocupacionc, \$parentesco, \$correocon, \$nombrerep, \$ccrep, \$expedidore, \$telrep, \$dirrep,
\$parenterep, \$institucio, \$grado, \$icfes, \$preseicfes, \$notas11, \$puesto11, \$alotrobene,
\$universida, \$sede, \$nivel, \$carrera, \$modalidad, \$semestre, \$horario, \$snies, \$continuaci,
\$vasemestre, \$porcentaje, \$montoaprov, \$otrobene, \$maperdidas, \$promediat,
\$promeditra, \$bene1ant, \$bene2ant, \$horascumpl, \$dependenc1, \$dependenc2,
\$dependenc3, \$nocuenta, \$diasviaja, \$ciudadvia, \$totaldias, \$valor2rec, \$valorseman,
\$valormes, \$valormespa, \$mesagirar, \$totalsemes, \$ratificado, \$estado, \$observacion)

```

```
{
```

```

    \$porcentajeapli = \$porcentaje."%";
    include ("conexion.php");
    \$con = mysqli_query(\$con , "update `Stabla` set `ID_SUB` = \"\$id.\", `ITEM` =
    \"\$item.\", `NOCARPETA` = \"\$carpeta.\", `FECHARADICACION` =
    \"\$radica.\", `BENEFICIO1` = \"\$bene1.\", `BENEFICIO2` =
    \"\$bene2.\", `1VEZ` = \"\$vez1.\", `RN` = \"\$rn.\", `REI` = \"\$rei.\",
    `PAPELLIDO` = \"\$apellido1.\", `SAPELLIDO` = \"\$apellido2.\", `PNOMBRE`
    = \"\$nombre1.\", `SNOMBRE` = \"\$nombre2.\", `ID_US` = \"\$tdocu.\",
    `NODOCUMENTO` = \"\$ndocumento.\", `EXPEDIDOEN` = \"\$expedido.\",
    `FECHANACIMIENTO` = \"\$nacimiento.\", `EDAD` = \"\$edadus.\",
    `ESTADOCIVIL` = \"\$estadoc.\", `NEMBREEPS` = \"\$seps.\", `ESTADOEPS` =
    \"\$estadoeps.\", `POESPECIAL` = \"\$pespe.\", `GENERO` = \"\$gene.\",
    `TEL1` = \"\$telefono1.\", `TEL2` = \"\$telefono2.\", `DIRECCION` =
    \"\$direccion.\", `SECTOR` = \"\$sector.\", `CORREOE1` = \"\$correoelec.\",
    `CORREOE2` = \"\$correoele2.\", `ANTIGSISBEN` = \"\$antiguedad.\",
    `PUNTASISBEN` = \"\$puntajesis.\", `TRBAJAACtual` = \"\$trabaja.\",
    `SECTORLABO` = \"\$sectorla.\", `NOMEMPRESA` = \"\$empresa.\",
    `CARGOLAB` = \"\$cargo.\", `VIVIENDA` = \"\$vivienda.\", `TIEMPOVIVIENDO`
    = \"\$vive.\", `ESTRATO` = \"\$estrato.\", `PERSONASCARGO` =
    \"\$personasac.\", `NUCLEOFAMILIAR` = \"\$nucleo.\", `NOMBRECONTACTO`
    = \"\$nombrecon.\", `DOCUMENTOCONTACTO` = \"\$cc.\", `TELCONTACTO`
    = \"\$stelcon.\", `OCUPACIONCONTACTO` = \"\$ocupacionc.\",
    `PARENTESCO` = \"\$parentesco.\", `CORREOCONTACTO` =
    \"\$correocon.\", `NOMBREREPR` = \"\$nombrerep.\",
    `NOCEDULAR` = \"\$ccrep.\", `EXPEDIDAR` = \"\$expedidore.\", `TELREPR` =
    \"\$telrep.\", `DIRECCIONREPR` = \"\$dirrep.\", `PARENTESCOREP` =
    \"\$parenterep.\", `INSTITUCIONEDUMEDIA` = \"\$institucio.\",
    `FECHAGRADO` = \"\$grado.\", `PUNTAJEICFES` = \"\$icfes.\",
    `ANOPRESENTACION` = \"\$preseicfes.\", `PROMEDIOONCE` =

```

```

        `\".$notas11.\"`, `PUESTOONCE` = `\".$puesto11.\"`, `ROBROBENE` =
        `\".$alotrobene.\"`, `UNIVERSIDAD` = `\".$universida.\"`, `SEDE` = `\".$sede.\"`,
        `NIVEL` = `\".$nivel.\"`, `CARRERA` = `\".$carrera.\"`, `MODALIDAD` =
        `\".$modalidad.\"`, `SEMESTRE` = `\".$semestre.\"`, `HORARIO` = `\".$horario.\"`,
        `SNIES` = `\".$snies.\"`, `CONTINUACARRERA` = `\".$continuaci.\"`,
        `VALORSEMESTRE` = `\".$vasemestre.\"`, `PORCENTAJEA` =
        `\".$porcentajeapli.\"`, `MONTOAPRO` = `\".$montoaprov.\"`, `OTROBENE` =
        `\".$otrobene.\"`, `MATERIASPER` = `\".$maperdidas.\"`,
        `PROMEDIOANTERIOR` = `\".$promediant.\"`, `PROMEDIOTRAANTERIOR` =
        `\".$promeditra.\"`, `BENEFICIOSANTERIORES1` = `\".$bene1ant.\"`,
        `BENEFICIOSANTERIORES2` = `\".$bene2ant.\"`, `HSCUMPLIDAS` =
        `\".$horascumpl.\"`, `DEPENDENCIA1` = `\".$dependenc1.\"`, `DEPENDENCIA2`
        = `\".$dependenc2.\"`, `DEPENDENCIA3` = `\".$dependenc3.\"`, `NOCUENTA` =
        `\".$nocuenta.\"`, `DIASVIAJADOS` = `\".$diasviaja.\"`, `CIUDADVIAJA` =
        `\".$ciudadvia.\"`, `DIAS` = `\".$totaldias.\"`, `VALORRECOR` = `\".$valor2rec.\"`,
        `VALORSEMANA` = `\".$valorseman.\"`, `VALORMESBRUTO` = `\".$valormes.\"`,
        `VALORMESPAGAR` = `\".$valormespa.\"`, `MESES` = `\".$mesagitar.\"`,
        `TOTALSEMESTRE` = `\".$totalsemes.\"`, `RATIFICADO` =
        `\".$ratificado.\"`, `APROBADO` = `\".$estado.\"`, `OBSERVACIONRAT` =
        `\".$observacion.\" where NODOCUMENTO= `\".$ndocumento.\";`) or die (\"No se
        actualizo el registro\");
    }
}
?>
<script type=\"text/javascript\">
    alert(\"Registro del estudiante modificado exitosamente\");
    window.location.href='$periodo.php';
</script>;
}
if ($modifi = fopen($modifi, \"a\")
{
    if (fwrite($modifi, \"$mensajes\"))
    {
    }
    else
    {
        ?>
        <script type=\"text/javascript\">
            alert(\"Ha habido un problema modificarregistro.php\");
            window.location.href='index.php';
        </script>
        <?php
    }
    fclose($modifi);
}
}

```


?>

9.1.17 pantalla donde se pide el documento del beneficiario para la creación de un nuevo registro

```
<?php
    $cedu = "nuevo$periodo.php";
    if(file_exists ($cedu))
    {
    }
    else
    {
        $mensajess = "<?php
include (\\"blodese.php\\");
?>
<!doctype html>
<html>
<head>
    <title> Base de datos</title>
    <link rel=\\"stylesheet\" type=\\"text/css\" href=\\"estilopri.css\\\">
    <style type=\\"text/css\\\">
        table
        {
            height: 432px;
            width: 96%;
            text-align: left;
            display: block;
            overflow-x: auto;
        }
        th
        {
            position: static;
        }
    </style>
</head>

<body>
    <form id=\\"form1\" style=\\"width:98.72%; height:auto; background-color:white; color:blue;
font-size:14pt; font-family: Comic Sans MS; text-align: right; padding-right:1%; border: 1px solid
#CCC;\"><div><center>
    <p>Estudiantes $periodo</p>
    <div style=\\"float: left;\"><?php echo \$_SESSION[\"usuarios\"]?></div>
    <div style = \\"float: right;\"><a href=\\"salir.php\">Salir</a></div>
```

```

        <p>&nbsp;</p>
    </center></div>
</form>
<form id=\"form2\" method='POST' action='index.php' margin='0' style=\"float: left; width:16.1%;
height:95.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;\">
    <?php
        include (\"menuopciones.php\");
    ?>
</form>
<form id=\"form3\" method='POST' action='index.php' margin='0' style=\"float: left; width:82.62%;
height:5.2%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;\">
    <div style=\"float: left;\"><a href=\"\$periodo.php\">atras</a></div>
</form>
<form id=\"form4\" method='GET' action='\$periodo.php' margin='0' style=\"float: left; width:82.62%;
height:5.4%; background-color:white; color:blue;font-size:14pt; font-family: Comic Sans MS;
text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;\">
    <center>
        <p></p>
        <label>Número de documento:</label>
        <input type=\"text\" name=\"documento\" placeholder=\"Ingrese el
            documento\" style=\"width: 240px\" required>
        <div>
            <input name=\"registrar\" type=\"submit\" id=\"enviar2\"
                formaction=\"registro\$periodo.php\" value=\"Registrar\" style=\"width: 150px\">
        </div>
        <p></p>
    </center>
</form>
<form for=\"form5\" action='index.php' method='POST' margin='0' style=\"float: left; width:82.62%;
height:195.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;\">
</form>
</body>
</html>";
}
if ($cedu = fopen($cedu, "a"))
{
    if (fwrite($cedu, "$mensajess"))
    {
    }
    else
    {

```

```

?>
<script type="text/javascript">
    alert("Ha habido un nuevobeneficiario.php");
    window.location.href='index.php';
</script>
<?php
}
fclose($cedu);
}
?>

```

9.1.18 menú periodo

// es un menú donde aparecen los periodos registrados, este crece con la creación de los distintos periodos

```

<?php
    echo "<p><a href=2018-2.php>Estudiantes 2018-2</a></p>
    <p><a href=2019-1.php>Estudiantes 2019-1</a></p>";
?>

```

9.1.19 fuid

```

<?php
// verifica la validacion del usuario
    include ("blodese.php");
// Incluye la conexión a la base de datos
    include ("conexion.php");
// consulta la tabla fuid para obtener datos
    function lacon()
    {
        global $con, $cons;
        $sql ='select * from fuid';
        return $con->query($sql);
    }
?>
<!doctype html>
<html>
<head>
    <meta http-equiv="Content-type" content="text/html; charset=utf-8">
    <meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=no">
    <title> Base de datos FUID</title>

```

// se agrega la hoja de estilos

```
<link rel="stylesheet" type="text/css" href="estilopri.css">
<link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/1.10.19/css/jquery.dataTables.min.css">
<link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/buttons/1.5.6/css/buttons.dataTables.min.css">
```

// se agrega estilos únicos de la pagina

```
<style type="text/css">
  input
  {
    width: 200px;
  }
  table
  {
    height: 372px;
    width: 74%;
    text-align: left;
    display: block;
    overflow-x: auto;
  }
  th
  {
    position: static;
  }
</style>
<script type="text/javascript" language="javascript" src="https://code.jquery.com/jquery-3.3.1.js"></script>
<script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/1.10.19/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/dataTables.buttons.min.js"></script>
<script type="text/javascript" language="javascript"
src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
<script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/buttons.html5.min.js"></script>
<script type="text/javascript" class="init">
$(document).ready(function() {
  $('#example').DataTable( {
    dom: 'Bfrtip',
    buttons: [
      'excelHtml5'
    ]
  } );
});
</script>
```

```

</head>
// interfaz de usuario
<body>
  <form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-
size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid
#CCC;"><div><center>
  <p>Beneficiarios registrados</p>
// a el usuario que ingreso al sistema
  <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
// opción para salir del sistema
  <div style = "float: right;"><a href="salir.php">Salir</a></div>
  <p>&nbsp;</p>
</center></div>
</form>
//funciones extra
  <form id="form2"method='POST' action='index.php' margin='0' style="float:
left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans
MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  <?php
    include ("menuopciones.php");
  ?>
</form>
//registrar nuevo beneficiario
  <form id="form3"method='POST' action='index.php' margin='0' style="float:
left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans
MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  <center>
    <p></p>
    <input name="regfuid" type="submit" id="enviar2" formaction="regfuid.php"
value="Ingresar nuevo beneficiario" style="width:250px">
    <p></p>
  </center>
</form>
//muestra los registros de la tabla fuid
<form id="form5" action="fuid.php" method='POST' margin='0' style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans
MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
  <center>
  <div class="fw-container" style="width: 695px">
    <table id="example" class="display" style="width:100%">
      <thead>
        <tr>
          <th colspan="8"><center>Beneficiarios</center></th>
        </tr>
        <tr>

```

```
<th><center>Item</center></th>
<th><center>Numero de Documento</center></th>
<th><center>Primer Apellido</center></th>
<th><center>Segundo Apellido</center></th>
<th><center>Primer Nombre</center></th>
<th><center>Segundo Nombre</center></th>
<th><center>Numero de Carpeta</center></th>
<th><center>Modificar</center></th>
```

```
</tr>
```

```
</thead>
```

```
<tbody>
```

```
<?php
```

```
$cons = lacon();
```

```
while($row = $cons->fetch_assoc())
```

```
{
```

```
    echo "<tr>";
```

```
    echo "<td><center>"; echo $row['ITEM'];
```

```
    echo "</center></td>";
```

```
    echo "<td><center>"; echo
    $row['NDOCUMENTO']; echo
```

```
    "</center></td>";
```

```
    echo "<td><center>"; echo
```

```
    $row['PRAPELLIDO']; echo
```

```
    "</center></td>";
```

```
    echo "<td><center>"; echo
```

```
    $row['SEAPELLIDO']; echo
```

```
    "</center></td>";
```

```
    echo "<td><center>"; echo
```

```
    $row['PRIMNOMBRE']; echo
```

```
    "</center></td>";
```

```
    echo "<td><center>"; echo
```

```
    $row['SEGUNOMBRE']; echo
```

```
    "</center></td>";
```

```
    echo "<td><center>"; echo
```

```
    $row['NOCARPETA']; echo
```

```
    "</center></td>";
```

```
    echo "<td><center><a
```

```
    href='modfuid.php?ITEM=".$row['ITEM']."
```

```
    >";
```

```
    ?>
```

```
    <input type="submit"
```

```
    name="Modificar" id="Modificar"
```

```
    value="Modificar"
```

```
    formaction="modfuid.php?ITEM=
```

```
    <?php echo $row['ITEM'] ?>"
```

```

        class="btn btn-success"
        style="width: 70px">
        <?php
        echo "<a></center></td></tr>";
    }
?>
</tbody>
</table>
</div>
</center>
</form>
</body>
</html>

```

9.1.20 Registrar Fuid

// Se incluye el archivo de bloqueo

```

<?php
    include ("blodese.php");
?>
<!doctype html>
<html>
<head>
//titulo de la pagina
    <title> Base de datos</title>
//hoja de estilos
    <link rel="stylesheet" type="text/css" href="estilopri.css">
</head>
// interfaz de usuario
<body>
    <form id="form1" style="width:98.72%; height:2.6%; background-color:white; color:blue;
    font-size:14pt; font-family: Comic Sans MS; text-align: right; padding-right:1%; border: 1px solid
    #CCC;"><div><center>
        <p>Ingrese datos del beneficiario</p>
// a el usuario que ingreso al sistema
        <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
// opción para salir del sistema
        <div style = "float: right;"><a href="salir.php">Salir</a></div>
        <p>&nbsp;</p>
    </center></div>
    </form>
//funciones extra

```

```
<form id="form2"method='POST' action='index.php' margin='0' style="float: left; width:16.1%; height:95.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS; text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
```

```
<?php
    include ("menuopciones.php");
?>
```

```
</form>
```

//muestra los espacios para llenar con los datos del beneficiario

```
<form id="form3"method='POST' action='index.php' margin='0' style="float: left; width:82.62%; height:5.2%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS; text-align: left; padding-right:0.5%; border: 0.5px solid #CCC;">
```

```
<spam class="text">
```

Los espacios marcados con asterisco (*) son obligatorios

```
</spam>
```

```
</form>
```

```
<form id="form4"method='POST' action='cregfuid.php' margin='0' style="float: left; width:82.62%; height:5.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS; text-align: left; padding-right:0.5%; border: 0.5px solid #CCC;">
```

```
<span class="text">
```

```
<center>
```

```
<H3>
```

```
<div>
```

```
<label>Item y numero de carpeta*:</label>
```

```
<input type="text" name="item" placeholder="Digite el numero de item tambien sera el numero de carpeta" style="width:230px">
```

```
<p></p>
```

```
<label>Numero documento*:</label>
```

```
<input type="text" name="nocumento" placeholder="Digite el numero de documento" style="width:230px">
```

```
<p></p>
```

```
</div>
```

```
<div>
```

```
<label>Primer apellido*:</label>
```

```
<input type="text" name="papellido" placeholder="Digite el primer apellido" style="width:230px">
```

```
<p></p>
```

```
<label>Segundo apellido*:</label>
```

```
<input type="text" name="sapellido" placeholder="Digite el segundo apellido" style="width:230px">
```

```
<p></p>
```

```
</div>
```

```
<div>
```

```
<label>Primer nombre*:</label>
```

```
<input type="text" name="pnombre" placeholder="Digite el primer nombre" style="width:230px">
```



```

        <p></p>
        <label>Segundo apellido*:</label>
        <input type="text" name="snombre" placeholder="Digite el
        segundo nombre" style="width:230px">
    </div>
</H3>
</center>
</span>
<span class="button">
    <center>
        <input type="submit" value="Registrar" style="width:100px">
        <input type="submit" value="Volver" style="width: 100px" formaction="fuid.php">
    </center>
</span>
</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float: left; width:82.62%;
height:195.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
text-align: right; padding-right:0.5%; border: 0.5px solid #CCC;">
<p></p>
</form>
</body>
</html>

```

9.1.21 archivo que toma los datos de registro de la pantalla registrar fuid y los pasa a la base de datos

```

<?php
include ("blodese.php");
include ("conexion.php");
if (isset($_POST["item"], $_POST["nocumento"], $_POST["papellido"], $_POST["sapellido"],
$_POST["pnombre"], $_POST["snombre"]) and $_POST["item"] != "" and
$_POST["nocumento"] != "" and $_POST["papellido"] != "" and $_POST["sapellido"] != "" and
$_POST["pnombre"] != "" and $_POST["snombre"] != "")
{
    $item = $_POST["item"];
    $nocumento = $_POST["nocumento"];
    $papellido = $_POST["papellido"];
    $sapellido = $_POST["sapellido"];
    $pnombre = $_POST["pnombre"];
    $snombre = $_POST["snombre"];
    $con = mysqli_query($con , "insert into fuid (ITEM, NDOCUMENTO,
    PRAPELLIDO, SEPELLIDO, PRIMNOMBRE, SEGUNOMBRE, NOCARPETA)
    VALUES ('$item', '$nocumento', '$papellido', '$sapellido', '$pnombre',
    '$snombre', '$item)");
}

```

```

?>
<script type="text/javascript">
    alert("Beneficiario agregado exitosamente");
    window.location.href='fuid.php';
</script>
<?php
}
else
{
?>
<script type="text/javascript">
    alert("Por favor, complete el formulario");
    window.location.href='regfuid.php';
</script>
<?php
}
?>

```

9.1.22 pantalla modificar beneficiario fuid

```

<?php
include ("blodese.php");
$consulta = consultarfuid($_GET['ITEM']);
function consultarfuid($item)
{
    include ("conexion.php");
    $con = mysqli_query($con , "select * from fuid where fuid . ITEM = '$item'");
    $filas=mysqli_fetch_assoc($con);
    return
    [
        $filas['ITEM'],
        $filas['NDOCUMENTO'],
        $filas['PRAPELLIDO'],
        $filas['SEAPELLIDO'],
        $filas['PRIMNOMBRE'],
        $filas['SEGUNOMBRE']
    ];
}
?>
<!doctype html>
<html>
<head>
<title> Base de datos</title>

```

```

<link rel="stylesheet" type="text/css" href="estilopri.css">
<style type="text/css">
    table
    {
        height: 415px;
        width: 342px;
        text-align: left;
        display: block;
        overflow-x: auto;
    }
    th
    {
        position: static;
    }
</style>
</head>
<body>
<form id="form1" style="width:98.72%; height:2.6%; background-color:white; color:blue;
font-size:14pt; font-family: Comic Sans MS; text-align: right; padding-right:1%; border: 1px solid
#CCC;"><div><center>
    <p>Modifique beneficio</p>
    <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
    <div style = "float: right;"><a href="salir.php">Salir</a></div>
    <p>&nbsp;</p>
</center></div>
</form>
<form id="form2" method='POST' action='index.php' margin='0' style="float: left; width:16.1%;
height:95.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <?php
        include ("menuopciones.php");
    ?>
</form>
<form id="form3" method='POST' action='index.php' margin='0' style="float: left; width:82.62%;
height:5.2%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
text-align: left; padding-right:0.5%; border: 0.5px solid #CCC;">
    <spam class="text">
        Los espacios marcados con asterisco (*) son obligatorios
    </spam>
</form>
<form id="form4" method='POST' action='modfuid2.php' margin='0' style="float: left;
width:82.62%; height:5.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic
Sans MS; text-align: left; padding-right:0.5%; border: 0.5px solid #CCC;">
    <span class="text">

```

```

<center>
<H3>
<div>
<input type="hidden" name="ITEM" value="<?php echo
$_GET['ITEM']?>">
<label>Item y numero de carpeta*:</label>
<input type="text" name="ncarpeta"; value="<?php echo
$consulta[0] ?>" style="width:230px">
<p></p>
<label>Numero documento*:</label>
<input type="text" name="nodocumento" value="<?php echo
$consulta[1] ?>" style="width:230px">
<p></p>
</div>
<div>
<label>Primer apellido*:</label>
<input type="text" name="papellido" value="<?php echo
$consulta[2] ?>" style="width:230px">
<p></p>
<label>Segundo apellido*:</label>
<input type="text" name="sapellido" value="<?php echo
$consulta[3] ?>" style="width:230px">
<p></p>
</div>
<div>
<label>Primer nombre*:</label>
<input type="text" name="pnombre" value="<?php echo
$consulta[4] ?>" style="width:230px">
<p></p>
<label>Segundo apellido*:</label>
<input type="text" name="snombre" value="<?php echo
$consulta[5] ?>" style="width:230px">
</div>
</H3>
</center>
</span>
<span class="button">
<center>
<input type="submit" value="Modificar" style="width:100px">
<input type="submit" value="Volver" style="width: 100px"
formaction="fuid.php">
</center>
</span>
</form>

```

```

<form for="form5" action='index.php' method='POST' margin='0' style="float: left; width:82.62%;
height:195.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
text-align: right; padding-right:0.5%; border: 0.5px solid #CCC;">
    <p></p>
</form>
</body>
</html>

```

9.1.23 archivo que lleva los datos de la pantalla modificar a la base de datos

```

<?php
include ("blodese.php");
modificarbeneicio1($_POST['ITEM'], $_POST['ncarpeta'], $_POST['nodocumento'],
$_POST['papellido'], $_POST['sapellido'], $_POST['pnombre'], $_POST['snombre']);
function modificarbeneicio1
($item,$ncarpeta,$nodocumento,$papellido,$sapellido,$pnombre,$snombre)
{
    include ("conexion.php");
    $conn = mysqli_query($con , "UPDATE `fuid` SET `ITEM` = '$ncarpeta',
`NDOCUMENTO` = '$nodocumento', `PRAPELLIDO` = '$papellido', `SEPELLIDO` =
'$sapellido', `PRIMNOMBRE` = '$pnombre', `SEGUNOMBRE` = '$snombre',
`NOCARPETA` = '$ncarpeta' WHERE `fuid`.`ITEM` = $item;");
}
?>
<script type="text/javascript">
    alert("Beneficiario modificado exitosamente");
    window.location.href='fuid.php';
</script>

```

9.1.24 pantalla beneficios 1 registrados

```

<?php
include ("blodese.php");
include ("conexion.php");
function lacon()
{
    global $con, $cons;
    $sql ='select * from beneficio1 ORDER BY NOMBENEFICIO1 ASC';
    return $con->query($sql);
}
?>

```

```

<!doctype html>
<html>
<head>
  <meta http-equiv="Content-type" content="text/html; charset=utf-8">
  <meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=no">
  <title> Beneficios registrados</title>
  <link rel="stylesheet" type="text/css" href="estilopri.css">
  <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/1.10.19/css/jquery.dataTables.min.css">
  <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/buttons/1.5.6/css/buttons.dataTables.min.css">
  <style type="text/css">
    input
    {
      width: 200px;
    }
    table
    {
      width: 74%;
      text-align: left;
      display: block;
      overflow-x: auto;
    }
    th
    {
      position: static;
    }
  </style>
  <script type="text/javascript" language="javascript" src="https://code.jquery.com/jquery-3.3.1.js"></script>
  <script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/1.10.19/js/jquery.dataTables.min.js"></script>
  <script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/dataTables.buttons.min.js"></script>
  <script type="text/javascript" language="javascript"
src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
  <script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/buttons.html5.min.js"></script>
  <script type="text/javascript" class="init">
$(document).ready(function() {
  $('#example').DataTable( {
    dom: 'Bfrtip',
    buttons: [
      ]
    }
  );
});
  </script>

```

```

    });
  });
</script>
</head>
<body>
  <form id="form1" style="width:98.72%; height:2.6%; background-color:white; color:blue;
  font-size:14pt; font-family: Comic Sans MS; text-align: right; padding-right:1%; border: 1px solid
  #CCC;"><div><center>
    <p>Beneficios registrados</p>
    <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
    <div style = "float: right;"><a href="salir.php">Salir</a></div>
    <p>&nbsp;</p>
  </center></div>
</form>
  <form id="form2"method='POST' action='index.php' margin='0' style="float: left; width:16.1%;
  height:95.4%; background-color:white ;color:blue; font-size:14pt; font-family: Comic Sans MS;
  text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <?php
      include ("menuopciones.php");
    ?>
  </form>
  <form id="form3"method='POST' action='index.php' margin='0' style="float: left; width:82.62%;
  height:5.2%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
  text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <center>
      <p></p>
      <input name="regbene1" type="submit" id="enviar2" formaction="regbene1.php"
      value="Ingresar nuevo beneficio" style="width:230px">
      <p></p>
    </center>
  </form>
  <form id="form4" method='GET' margin='0' style="float: left; width:82.62%; height:5.4%;
  background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS; text-align: left;
  padding-right:0.5%;border: 0.5px solid #CCC;">
  </form>
  <form id="form5" action="fuid.php" method='POST' margin='0' style="float: left; width:82.62%;
  height:195.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
  text-align: right; padding-right:0.5%; border: 0.5px solid #CCC;">
    <center>
      <div class="fw-container" style="width: 348px">
        <table id="example" class="display" style="width:100%">
          <thead>
            <tr>
              <th colspan="3"><center>Beneficios 1</center></th>

```

```

</tr>
<tr>
<th><center>Nombre</center></th>
<th><center>Modificar</center></th>
<th><center>Eliminar</center></th>
</tr>
</thead>
<tbody>
<?php
$cons = lacon();
while($row = $cons->fetch_assoc())
{
    echo "<tr>";
    echo "<td>"; echo $row['NOMBENEFICIO1'];
    echo "</td>";
    echo "<td><center><a
href='modbene1.php?ID_BENEFICIO1=
".$row['ID_BENEFICIO1'].>";
?>
<input type='submit' name='Modificar'
id='Modificar' value='Modificar'
formaction='modbene1.php?ID_BENEFICIO1=
<?php echo $row['ID_BENEFICIO1'] ?>'
class='btn btn-success' style='width: 70px'>
<?php
echo '</a></center></td>";
    echo "<td><center>
<a href='elimbene1.php?NOMBENEFICIO1=' .
$row['NOMBENEFICIO1'].>";
?>
<input type='submit' name='Eliminar'
id='Eliminar' value='Eliminar'
formaction='elimbene1.php?NOMBENEFICIO1=
<?php echo $row['NOMBENEFICIO1'] ?>'
class='btn btn-success' style='width: 70px'>
<?php
echo "</a></center></td></tr>";
}
?>
</tbody>
</table>
</div>
</center>
</form>
</body>

```



```
</html>
```

9.1.25 pantalla registro beneficio1

```
<?php
```

```
include ("blodese.php");
```

```
?>
```

```
<!doctype html>
```

```
<html>
```

```
<head>
```

```
<title> Base de datos</title>
```

```
<link rel="stylesheet" type="text/css" href="estilopri.css">
```

```
<style type="text/css">
```

```
table
```

```
{
```

```
height: 415px;
```

```
width: 342px;
```

```
text-align: left;
```

```
display: block;
```

```
overflow-x: auto;
```

```
}
```

```
th
```

```
{
```

```
position: static;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<form id="form1" style="width:98.72%; height:2.6%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS; text-align: right; padding-right:1%; border: 1px solid #CCC;"><div><center>
```

```
<p>Ingrese el beneficio</p>
```

```
<div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
```

```
<div style = "float: right;"><a href="salir.php">Salir</a></div>
```

```
<p>&nbsp;</p>
```

```
</center></div>
```

```
</form>
```

```
<form id="form2" method='POST' action='index.php' margin='0' style="float: left; width:16.1%; height:95.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS; text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
```

```
<?php
```

```
include ("menuopciones.php");
```

```

?>
</form>
<form id="form3" method='POST' action='index.php' margin='0' style="float: left; width:82.62%;
height:5.2%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  <spam class="text">
    Los espacios marcados con asterisco (*) son obligatorios
  </spam>
</form>
<form id="form4" method='POST' action='cregbene1.php' margin='0' style="float: left;
width:82.62%; height:5.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  <span class="text">
    <center>
      <H3>
        <div>
          <label>Nombre Beneficio *:</label>
          <input type="text" name="nomben1" placeholder="Digite el
nombre" style="width:230px">
        </div>
      </H3>
    </center>
  </span>
  <span class="button">
    <center>
      <input type="submit" value="Registrar" style="width:100px">
      <input type="submit" value="Volver" style="width: 100px"
formaction="bene1.php">
    </center>
  </span>
</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float: left; width:82.62%;
height:195.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
  <p></p>
</form>
</body>
</html>

```

9.1.26 archivo que lleva los datos de registro beneficio 1 a la base de datos

```
<?php
include ("blodese.php");
include ("conexion.php");
if (isset($_POST["nomben1"]) and $_POST["nomben1"] != "")
{
    $nombre = $_POST["nomben1"];
    $con = mysqli_query($con , "insert into beneficio1
(NOMBENEFICIO1)VALUES('$nombre')");
    ?>
        <script type="text/javascript">
            alert("Beneficio agregado exitosamente");
            window.location.href='bene1.php';
        </script>
    <?php
    }
else
    {
        ?>
            <script type="text/javascript">
                alert("Por favor, complete el formulario");
                window.location.href='regbene1.php';
            </script>
        <?php
    }
?>
```

9.1.27 pantalla modificar beneficio 1

```
<?php
include ("blodese.php");
$consulta = consultarbeneficio1($_GET['ID_BENEFICIO1']);
function consultarbeneficio1($ID_BENEFICIO1)
{
    include ("conexion.php");
    $con = mysqli_query($con , "select * from beneficio1 where beneficio1 . ID_BENEFICIO1
= '$ID_BENEFICIO1'");
    $filas=mysqli_fetch_assoc($con);
    return
    [
        $filas['NOMBENEFICIO1']
    ];
}
```

```

    }
?>
<!doctype html>
<html>
<head>
    <title> Base de datos</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <style type="text/css">
        table
        {
            height: 415px;
            width: 342px;
            text-align: left;
            display: block;
            overflow-x: auto;
        }
        th
        {
            position: static;
        }
    </style>
</head>

<body>
    <form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;
font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid
#CCC;"><div><center>
        <p>Modifique beneficio</p>
        <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
        <div style = "float: right;"><a href="salir.php">Salir</a></div>
        <p>&nbsp;</p>
    </center></div>
</form>
    <form id="form2" method='POST' action='index.php' margin='0' style="float: left; width:16.1%;
height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;
text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
        <?php
            include ("menuopciones.php");
        ?>
    </form>
    <form id="form3" method='POST' action='index.php' margin='0' style="float: left;width:82.62%;
height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;
text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
        <spam class="text">
            Los espacios marcados con asterisco (*) son obligatorios

```

```

        </spam>
    </form>
    <form id="form4"method='POST' action='modbene12.php' margin='0' style="float: left;
width:82.62%; height:5.4%;background-color:white;color:blue;font-size:14pt;
font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <span class="text">
        <center>
            <H3>
                <div>
                    <input type="hidden" name="ID_BENEFICIO1" value="<?php
echo $_GET['ID_BENEFICIO1']?>">
                    <label>Nombre Beneficio *:</label>
                    <input type="text" name="NOMBENEFICIO1"; value="<?php echo
$consulta[0] ?>"style="width:230px">
                </div>
            </H3>
        </center>
    </span>
    <span class="button">
        <center>
            <input type="submit" value="Modificar" style="width:100px">
            <input type="submit" value="Volver" style="width: 100px"
formaction="bene1.php">
        </center>
    </span>
</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float: left; width:82.62%;
height:195.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
    <p></p>
</form>
</body>
</html>

```

9.1.28 archivo que lleva los datos de modificar beneficio 1 a la base de datos

```

<?php
include ("blodese.php");
modificarbene1($_POST['NOMBENEFICIO1'], $_POST['ID_BENEFICIO1']);
function modificarbene1($NOMBENEFICIO1,$ID_BENEFICIO1)
{
    include ("conexion.php");
    $conn = mysqli_query($con , "update beneficio1 set

```

```

        NOMBENEFICIO1=".$NOMBENEFICIO1." where
        ID_BENEFICIO1=".$ID_BENEFICIO1."");
    }
?>
<script type="text/javascript">
    alert("Beneficio modificado exitosamente");
    window.location.href='bene1.php';
</script>

```

9.1.29 archivo que borra datos de beneficio 1 de la base de datos

```

<?php
    include ("blodese.php");
    Eliminarbeneficio1($_GET['NOMBENEFICIO1']);
    function Eliminarbeneficio1($NOMBENEFICIO1)
    {
        include ("conexion.php");
        $con = mysqli_query($con , "delete from beneficio1 where beneficio1 .
        NOMBENEFICIO1 = '$NOMBENEFICIO1'");
    }
?>
<script type="text/javascript">
    alert("Beneficio eliminado exitosamente");
    window.location.href='bene1.php';
</script>

```

9.1.30 pantalla beneficios 2 registrados

```

<?php
    include ("blodese.php");
    include ("conexion.php");
    function lacon()
    {
        global $con, $cons;
        $sql ='select * from beneficio2 ORDER BY NOMBENEFICIO2 ASC';
        return $con->query($sql);
    }
?>
<!doctype html>
<html>
<head>

```

```

<meta http-equiv="Content-type" content="text/html; charset=utf-8">
<meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=no">
<title> Beneficios registrados</title>
  <link rel="stylesheet" type="text/css" href="estilopri.css">
  <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/1.10.19/css/jquery.dataTables.min.css">
  <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/buttons/1.5.6/css/buttons.dataTables.min.css">
  <style type="text/css">
    input
    {
      width: 200px;
    }
  table
  {
    width: 74%;
    text-align: left;
    display: block;
    overflow-x: auto;
  }
  th
  {
    position: static;
  }
</style>
<script type="text/javascript" language="javascript" src="https://code.jquery.com/jquery-
3.3.1.js"></script>
<script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/1.10.19/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/dataTables.buttons.min.js"></script>
<script type="text/javascript" language="javascript"
src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
<script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/buttons.html5.min.js"></script>
<script type="text/javascript" class="init">
$(document).ready(function() {
  $('#example').DataTable( {
    dom: 'Bfrtip',
    buttons: [
      ]
  } );
});
</script>

```

```

</head>
<body>
  <form id="form1" style="width:98.72%; height:2.6%; background-color:white; color:blue;
  font-size:14pt; font-family: Comic Sans MS; text-align: right; padding-right:1%; border: 1px solid
  #CCC;"><div><center>
    <p>Beneficios registrados</p>
    <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
    <div style = "float: right;"><a href="salir.php">Salir</a></div>
    <p>&nbsp;</p>
  </center></div>
</form>
  <form id="form2"method='POST' action='index.php' margin='0' style="float: left; width:16.1%;
  height:95.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
  text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <?php
      include ("menuopciones.php");
    ?>
  </form>
  <form id="form3"method='POST' action='index.php' margin='0' style="float: left; width:82.62%;
  height:5.2%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
  text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <center>
      <p></p>
      <input name="regbene2" type="submit" id="enviar2" formaction="regbene2.php"
      value="Ingresar nuevo beneficio" style="width:230px">
      <p></p>
    </center>
  </form>
  <form id="form4" method='GET' margin='0' style="float: left; width:82.62%; height:5.4%;
  background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
  text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  </form>
  <form id="form5" action="fuid.php" method='POST' margin='0' style="float: left; width:82.62%;
  height:195.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
  text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
    <center>
      <div class="fw-container" style="width: 327px">
        <table id="example" class="display" style="width:100%">
          <thead>
            <tr>
              <th colspan="3"><center>Beneficios
              2</center></th>
            </tr>
          </thead>
        </table>
      </div>
    </center>
  </form>

```



```
<th><center>Nombre</center></th>
<th><center>Modificar</center></th>
<th><center>Eliminar</center></th>
</tr>
</thead>
<tbody>
<?php
    $cons = lacon();
    while($row = $cons->fetch_assoc())
    {
        echo "<tr>";
        echo "<td>"; echo
        $row['NOMBENEFICIO2']; echo "</td>";
        echo "<td><center><a
        href='modbene2.php?ID_BENEFICIO2=
        ".$row['ID_BENEFICIO2']."'>
        <button type='button'
        class='btn btn-success'>Modificar
        </button></a></center></td>";
        echo "<td><center><a
        href='elimbene2.php?NOMBENEFICIO2=
        ".$row['NOMBENEFICIO2']."'>
        ?>
        <button formaction=
        \"elimbene2.php?NOMBENEFICIO2=
        <?php echo $row['NOMBENEFICIO2'] ?>\"
        class=\"btn btn-danger btn-sm\"
        onclick=\"return confirmarRegistro()\">
        <span class=\"fa fa-trash\">
        </span>Eliminar</button>
        <?php
        echo "</a></center></td></tr>";
    }
?>
</tbody>
</table>
</div>
</center>
</form>
</body>
</html>
```

9.1.31 pantalla registro beneficio 2

```
<?php
    include ("blodese.php");
?>
<!doctype html>
<html>
<head>
    <title> Base de datos</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <style type="text/css">
        table
        {
            height: 415px;
            width: 342px;
            text-align: left;
            display: block;
            overflow-x: auto;
        }
        th
        {
            position: static;
        }
    </style>
</head>
<body>
    <form id="form1" style="width:98.72%; height:2.6%; background-color:white; color:blue;
    font-size:14pt; font-family: Comic Sans MS; text-align: right; padding-right:1%; border: 1px solid
    #CCC;"><div><center>
        <p>Ingrese el beneficio</p>
        <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
        <div style = "float: right;"><a href="salir.php">Salir</a></div>
        <p>&nbsp;</p>
    </center></div>
</form>
    <form id="form2" method='POST' action='index.php' margin='0' style="float: left; width:16.1%;
    height:95.4%; background-color:white; color:blue; font-size:14pt; font-family: Comic Sans MS;
    text-align: left; padding-right:0.5%; border: 0.5px solid #CCC;">
        <?php
            include ("menuopciones.php");
        ?>
    </form>
    <form id="form3" method='POST' action='index.php' margin='0' style="float: left; width:82.62%;
```

```

height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <spam class="text">
        Los espacios marcados con asterisco (*) son obligatorios
    </spam>
</form>
<form id="form4" method='POST' action='cregbene2.php' margin='0' style="float: left; width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <span class="text">
        <center>
            <H3>
                <div>
                    <label>Nombre Beneficio *:</label>
                    <input type="text" name="nomben2" placeholder="Digite el nombre" style="width:230px">
                </div>
            </H3>
        </center>
    </span>
    <span class="button">
        <center>
            <input type="submit" value="Registrar" style="width:100px">
            <input type="submit" value="Volver" style="width: 100px"
            formaction="bene2.php">
        </center>
    </span>
</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float: left; width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
    <p></p>
</form>
</body>
</html>

```

9.1.32 archivo que lleva los datos de registro beneficio 2 a la base de datos

```

<?php
include ("blodese.php");
include ("conexion.php");
if (isset($_POST["nomben2"]) and $_POST["nomben2"] != "")
{

```

```

$nombre = $_POST["nomben2"];
$con = mysqli_query($con , "insert into beneficio2(NOMBENEFICIO2)
VALUES('$nombre')");
?>
<script type="text/javascript">
    alert("Beneficio agregado exitosamente");
    window.location.href='bene2.php';
</script>
<?php
}
else
{
    ?>
    <script type="text/javascript">
        alert("Por favor, complete el formulario");
        window.location.href='regbene2.php';
    </script>
    <?php
}
?>

```

9.1.33 pantalla modificar beneficio 2

```

<?php
include ("blodese.php");
$con = mysqli_query($con , "select * from beneficio2 where beneficio2 . ID_BENEFICIO2
= '$ID_BENEFICIO2'");
function consultarbeneficio2($ID_BENEFICIO2)
{
    include ("conexion.php");
    $con = mysqli_query($con , "select * from beneficio2 where beneficio2 . ID_BENEFICIO2
= '$ID_BENEFICIO2'");
    $filas=mysqli_fetch_assoc($con);
    return
    [
        $filas['NOMBENEFICIO2']
    ];
}
?>
<!doctype html>
<html>
<head>
<title> Base de datos</title>
<link rel="stylesheet" type="text/css" href="estilopri.css">

```

```

<style type="text/css">
  table
  {
    height: 415px;
    width: 342px;
    text-align: left;
    display: block;
    overflow-x: auto;
  }
  th
  {
    position: static;
  }
</style>
</head>
<body>
  <form id="form1" style="width:98.72%; height:2.6%; background-color:white; color:blue; font-
  size:14pt; font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid
  #CCC;"><div><center>
    <p>Modifique beneficio</p>
    <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
    <div style = "float: right;"><a href="salir.php">Salir</a></div>
    <p>&nbsp;</p>
  </center></div>
</form>
<form id="form2"method='POST' action='index.php' margin='0' style="float: left; width:16.1%;
height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-
align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  <?php
    include ("menuopciones.php");
  ?>
</form>
<form id="form3"method='POST' action='index.php' margin='0' style="float: left; width:82.62%;
height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-
align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  <spam class="text">
    Los espacios marcados con asterisco (*) son obligatorios
  </spam>
</form>
<form id="form4"method='POST' action='modbene22.php' margin='0' style="float: left;
width:82.62%; height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  <span class="text">
    <center>

```

```

<H3>
<div>
<input type="hidden" name="ID_BENEFICIO2" value="<?php
echo $_GET['ID_BENEFICIO2']?>">
<label>Nombre Beneficio *:</label>
<input type="text" name="NOMBENEFICIO2"; value="<?php echo
$consulta[0] ?>" style="width:230px">
</div>
</H3>
</center>
</span>
<span class="button">
<center>
<input type="submit" value="Modificar" style="width:100px">
<input type="submit" value="Volver" style="width: 100px"
formaction="bene2.php">
</center>
</span>
</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float: left; width:82.62%;
height:195.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-
align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
<p></p>
</form>
</body>
</html>

```

9.1.34 archivo que lleva los datos de modificar beneficio 2 a la base de datos

```

<?php
include ("blodese.php");
modificarbene2($_POST['NOMBENEFICIO2'], $_POST['ID_BENEFICIO2']);
function modificarbene2($NOMBENEFICIO2,$ID_BENEFICIO2)
{
include ("conexion.php");
$con = mysqli_query($con , "update beneficio2 set NOMBENEFICIO2=
'".$NOMBENEFICIO2."' where ID_BENEFICIO2='".$ID_BENEFICIO2."'");
}
?>
<script type="text/javascript">
alert("Beneficio modificado exitosamente");
window.location.href='bene2.php';
</script>

```

9.1.35 archivo que borra datos de beneficio 2 de la base de datos

```
<?php
    include ("blodese.php");
?>
<?php
    Eliminarbeneficio2($_GET['NOMBENEFICIO2']);
    function Eliminarbeneficio2($NOMBENEFICIO2)
    {
        include ("conexion.php");
        $con = mysqli_query($con , "delete from beneficio2 where beneficio2 . NOMBENEFICIO2
        = '$NOMBENEFICIO2'");
    }
?>
<script type="text/javascript">
    alert("Beneficio eliminado exitosamente");
    window.location.href='bene2.php';
</script>
```

9.1.36 pantalla otros beneficios registrados

```
<?php
    include ("blodese.php");
    include ("conexion.php");
    function lacon()
    {
        global $con, $cons;
        $sql ='select * from algunotro ORDER BY NOMBREALGUN ASC';
        return $con->query($sql);
    }
?>
<!doctype html>
<html>
<head>
    <meta http-equiv="Content-type" content="text/html; charset=utf-8">
    <meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=no">
    <title> Beneficios registrados</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <link rel="stylesheet" type="text/css"
    href="https://cdn.datatables.net/1.10.19/css/jquery.dataTables.min.css">
    <link rel="stylesheet" type="text/css"
    href="https://cdn.datatables.net/buttons/1.5.6/css/buttons.dataTables.min.css">
```

```

<style type="text/css">
  input
  {
    width: 200px;
  }
  table
  {
    width: 74%;
    text-align: left;
    display: block;
    overflow-x: auto;
  }
  th
  {
    position: static;
  }
</style>
<script type="text/javascript" language="javascript" src="https://code.jquery.com/jquery-3.3.1.js"></script>
<script type="text/javascript" language="javascript" src="https://cdn.datatables.net/1.10.19/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" language="javascript" src="https://cdn.datatables.net/buttons/1.5.6/js/dataTables.buttons.min.js"></script>
<script type="text/javascript" language="javascript" src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
<script type="text/javascript" language="javascript" src="https://cdn.datatables.net/buttons/1.5.6/js/buttons.html5.min.js"></script>
<script type="text/javascript" class="init">
$(document).ready(function() {
    $('#example').DataTable( {
        dom: 'Bfrtip',
        buttons: [

        ]
    } );
});
</script>
</head>
<body>
<form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid #CCC;"><div><center>
<p>Beneficios registrados</p>
<div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
<div style = "float: right;"><a href="salir.php">Salir</a></div>

```



```

        <p>&nbsp;</p>
    </center></div>
</form>
<form id="form2"method="POST" action='index.php' margin='0' style="float:
left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <?php
        include ("menuopciones.php");
    ?>
</form>
<form id="form3"method="POST" action='index.php' margin='0' style="float:
left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <center>
        <p></p>
        <input name="regbene2" type="submit" id="enviar2"
        formaction="regalgunotro.php" value="Ingresar nuevo beneficio"
        style="width:230px">
        <p></p>
    </center>
</form>
<form id="form4"method="GET" margin='0' style="float: left;width:82.62%;height:5.4%;background-
color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-
right:0.5%;border: 0.5px solid #CCC;">

</form>
<form id="form5" action="fuid.php" method='POST' margin='0' style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family:
Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
    <center>
        <div class="fw-container" style="width: 363px">
            <table id="example" class="display" style="width:100%">
                <thead>
                    <tr>
                        <th colspan="3"><center>Beneficios 2</center></th>
                    </tr>
                    <tr>
                        <th><center>Nombre</center></th>
                        <th><center>Modificar</center></th>
                        <th><center>Eliminar</center></th>
                    </tr>
                </thead>
                <tbody>
                    <?php
                        $cons = lacon();

```

```

while($row = $cons->fetch_assoc())
{
    echo "<tr>";
    echo "<td>"; echo $row['NOMBREALGUN']; echo "</td>";
    echo "<td><center><a
href='modalgunotro.php?IDALGUN="
.$row['IDALGUN']."'> <button type='button' class='btn btn-
success'>Modificar</button></a></center></td>";
    echo "<td><center><a
href='elimalgunotro.php?NOMBREALGUN="
.$row['NOMBREALGUN']."'>";
    ?>
    <button
formaction="elimalgunotro.php?NOMBREALGUN=
<?php echo $row['NOMBREALGUN'] ?>" class="btn btn-
danger btn-sm" onclick="return
confirmarRegistro()"><span class="fa fa-
trash"></span>Eliminar</button>
<?php
        echo "</a></center></td></tr>";
    }
    ?>
</tbody>
</table>
</div>
</center>
</form>
</body>
</html>

```

9.1.37 pantalla registro otros beneficiarios

```

<?php
    include ("blodese.php");
?>
<!doctype html>
<html>
<head>
    <title> Base de datos</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <style type="text/css">
        table
        {

```

```

        height: 415px;
        width: 342px;
        text-align: left;
        display: block;
        overflow-x: auto;
    }
    th
    {
        position: static;
    }
</style>
</head>

<body>
    <form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-
    size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid
    #CCC;"><div><center>
        <p>Ingrese el beneficio</p>
        <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
        <div style = "float: right;"><a href="salir.php">Salir</a></div>
        <p>&nbsp;</p>
    </center></div>
    </form>
    <form id="form2"method='POST' action='index.php' margin='0' style="float:
    left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
    Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
        <?php
            include ("menuopciones.php");
        ?>
    </form>
    <form id="form3"method='POST' action='index.php' margin='0' style="float:
    left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic
    Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
        <spam class="text">
            Los espacios marcados con asterisco (*) son obligatorios
        </spam>
    </form>
    <form id="form4"method='POST' action='cregalgunotro.php' margin='0' style="float:
    left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
    Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
        <span class="text">
            <center>
                <H3>
                    <div>
                        <label>Beneficio *:</label>

```

```

        <input type="text" name="nombene2" placeholder="Digite
        el nombre" style="width:230px">
    </div>
</H3>
</center>
</span>
<span class="button">
    <center>
        <input type="submit" value="Registrar" style="width:100px">
        <input type="submit" value="Volver" style="width: 100px"
        formaction="algunotro.php">
    </center>
</span>
</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family:
Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
    <p></p>
</form>
</body>
</html>

```

9.1.38 archivo que lleva los datos de registro otros beneficios a la base de datos

```

<?php
include ("blodese.php");
include ("conexion.php");
if (isset($_POST["nombene2"]) and $_POST["nombene2"] != "")
{
    $nombre = $_POST["nombene2"];
    $con = mysqli_query($con , "insert into algunotro
(NOMBREALGUN)VALUES('$nombre')");
    ?>
    <script type="text/javascript">
        alert("Beneficio agregado exitosamente");
        window.location.href='algunotro.php';
    </script>
    <?php
}
else
{
    ?>
    <script type="text/javascript">

```

```

        alert("Por favor, complete el formulario");
        window.location.href='regalgunotro.php';
    </script>
    <?php
}
?>

```

9.1.39 pantalla modificar otros beneficios

```

<?php
include ("blodese.php");
$consulta = consultarotrobeneficio($_GET['IDALGUN']);
function consultarotrobeneficio($IDALGUN)
{
    include ("conexion.php");
    $con = mysqli_query($con , "select * from algunotro where algunotro . IDALGUN =
'$IDALGUN'");
    $filas=mysqli_fetch_assoc($con);
    return
    [
        $filas['NOMBREALGUN']
    ];
}
?>
<!doctype html>
<html>
<head>
<title> Base de datos</title>
<link rel="stylesheet" type="text/css" href="estilopri.css">
<style type="text/css">
    table
    {
        height: 415px;
        width: 342px;
        text-align: left;
        display: block;
        overflow-x: auto;
    }
    th
    {
        position: static;
    }
</style>

```

```
</head>
```

```
<body>
```

```
<form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid #CCC;"><div><center>
```

```
<p>Modifique beneficio</p>
```

```
<div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
```

```
<div style = "float: right;"><a href="salir.php">Salir</a></div>
```

```
<p>&nbsp;</p>
```

```
</center></div>
```

```
</form>
```

```
<form id="form2"method='POST' action='index.php' margin='0' style="float:left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
```

```
<?php
```

```
include ("menuopciones.php");
```

```
?>
```

```
</form>
```

```
<form id="form3"method='POST' action='index.php' margin='0' style="float:left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
```

```
<spam class="text">
```

```
Los espacios marcados con asterisco (*) son obligatorios
```

```
</spam>
```

```
</form>
```

```
<form id="form4"method='POST' action='modalgunotro2.php' margin='0' style="float:left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
```

```
<span class="text">
```

```
<center>
```

```
<H3>
```

```
<div>
```

```
<input type="hidden" name="IDALGUN" value="<?php echo $_GET['IDALGUN']?>">
```

```
<label>Nombre Beneficio *:</label>
```

```
<input type="text" name="NOMBREALGUN"; value="<?php echo $consulta[0] ?>"style="width:230px">
```

```
</div>
```

```
</H3>
```

```
</center>
```

```
</span>
```

```
<span class="button">
```

```
<center>
```

```
<input type="submit" value="Modificar" style="width:100px">
```

```

        <input type="submit" value="Volver" style="width: 100px"
        formaction="algunotro.php">
    </center>
</span>
</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family:
Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
    <p></p>
</form>
</body>
</html>

```

9.1.40 archivo que lleva los datos de modificar otros beneficios a la base de datos

```

<?php
    include ("blodese.php");
    modificarotrobeneficio($_POST['NOMBREALGUN'], $_POST['IDALGUN']);
    function modificarotrobeneficio($NOMBREALGUN,$IDALGUN)
    {
        include ("conexion.php");
        $con = mysqli_query($con , "update algunotro set NOMBREALGUN=
        ".$NOMBREALGUN." where IDALGUN=".$IDALGUN."");
    }
?>
<script type="text/javascript">
    alert("Beneficio modificado exitosamente");
    window.location.href='algunotro.php';
</script>

```

9.1.41 archivo que borra datos de otros beneficios de la base de datos

```

<?php
    include ("blodese.php");
    Eliminarotrobeneficio($_GET['NOMBREALGUN']);
    function Eliminarotrobeneficio($NOMBREALGUN)
    {
        include ("conexion.php");
        $con = mysqli_query($con , "delete from algunotro where algunotro .
        NOMBREALGUN = '$NOMBREALGUN'");
    }

```

```
?>
<script type="text/javascript">
    alert("Beneficio eliminado exitosamente");
    window.location.href='algunotro.php';
</script>
```

9.1.42 pantalla antigüedad sisben

```
<?php
    include ("blodese.php");
    include ("conexion.php");
?>
<!doctype html>
<html>
<head>
    <meta http-equiv="Content-type" content="text/html; charset=utf-8">
    <meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=no">
    <title> Antigüedad Sisben</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/1.10.19/css/jquery.dataTables.min.css">
    <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/buttons/1.5.6/css/buttons.dataTables.min.css">
    <style type="text/css">
        input
        {
            width: 200px;
        }
        table
        {
            width: 74%;
            text-align: left;
            display: block;
            overflow-x: auto;
        }
        th
        {
            position: static;
        }
    </style>
    <script type="text/javascript" language="javascript" src="https://code.jquery.com/jquery-3.3.1.js"></script>
    <script type="text/javascript" language="javascript"
```



```

src="https://cdn.datatables.net/1.10.19/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/dataTables.buttons.min.js"></script>
<script type="text/javascript" language="javascript"
src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
<script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/buttons.html5.min.js"></script>
<script type="text/javascript" class="init">
$(document).ready(function() {
    $('#example').DataTable( {
        dom: 'Bfrtip',
        buttons: [
            ]
    } );
} );
</script>
</head>
<body>
<form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid #CCC;"><div><center>
<p>Antigüedad Sisben</p>
<div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
<div style = "float: right;"><a href="salir.php">Salir</a></div>
<p>&nbsp;</p>
</center></div>
</form>
<form id="form2"method='POST' action='index.php' margin='0' style="float:left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<?php
include ("menuopciones.php");
?>
</form>
<form id="form3"method='POST' action='index.php' margin='0' style="float:left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<center>
<p></p>
<input name="antsisben" type="submit" id="enviar2"
formaction="regantsisben.php" value="Ingresar antigüedad" style="width:230px">
<p></p>
</center>
</form>

```

```
<form id="form4" method='GET' margin='0' style="float: left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
```

```
</form>
```

```
<form id="form5" action="fuid.php" method='POST' margin='0' style="float: left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
```

```
<center>
```

```
<div class="fw-container" style="width: 363px">
```

```
<table id="example" class="display" style="width:100%">
```

```
<thead>
```

```
<tr>
```

```
<th colspan="4"><center>Antigüedad del  
sisben</center></th>
```

```
</tr>
```

```
<tr>
```

```
<th>Item</th>
```

```
<th>Tiempo</th>
```

```
<th>Modificar</th>
```

```
<th>Eliminar</th>
```

```
</tr>
```

```
</thead>
```

```
<tbody>
```

```
<?php
```

```
function lacon()
```

```
{
```

```
global $con, $cons;
```

```
$sql ='select * from antigu_sisben ORDER BY  
ID_ASISBEN';
```

```
return $con->query($sql);
```

```
}
```

```
$cons = lacon();
```

```
while($row = $cons->fetch_assoc())
```

```
{
```

```
echo "<tr>";
```

```
echo "<td>"; echo $row['ID_ASISBEN']; echo  
"</td>";
```

```
echo "<td>"; echo $row['NOMASISBEN']; echo  
"</td>";
```

```
echo "<td><center><a  
href='modantsisben.php?ID_ASISBEN="
```

```
".$row['ID_ASISBEN']."'> <button type='button'  
class='btn btn-success'>Modificar
```

```
</button></a></center></td>";
```

```

        }
        ?>
    </tbody>
</table>
</div>
</center>
</form>
</body>
</html>

```

```

echo "<td><center><a
href='elimantsisben.php?NOMASISBEN=
'.'. $row['NOMASISBEN'].''><button type='button'
class='btn btn-danger'>Eliminar</button></a>
</center></td>";
echo "</tr>";

```

9.1.43 pantalla registro antigüedad sisben

```

<?php
    include ("blodese.php");
?>
<!doctype html>
<html>
<head>
    <title> Base de datos</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <style type="text/css">
        table
        {
            height: 415px;
            width: 342px;
            text-align: left;
            display: block;
            overflow-x: auto;
        }
        th
        {
            position: static;
        }
    </style>
</head>
<body>

```

```

<form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-
size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid
#CCC;"><div><center>
<p>Ingrese el año</p>
<div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
<div style = "float: right;"><a href="salir.php">Salir</a></div>
<p>&nbsp;</p>
</center></div>

```

```

</form>

```

```

<form id="form2" method='POST' action='index.php' margin='0' style="float:
left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<?php

```

```

include ("menuopciones.php");

```

```

?>

```

```

</form>

```

```

<form id="form3" method='POST' action='index.php' margin='0' style="float:
left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">

```

```

<spam class="text">

```

```

Los espacios marcados con asterisco (*) son obligatorios

```

```

<div>Ejemplo 3-AÑOS</div>

```

```

</spam>

```

```

</form>

```

```

<form id="form4" method='POST' action='cregantsisben.php' margin='0' style="float:
left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">

```

```

<span class="text">

```

```

<center>

```

```

<H3>

```

```

<div>

```

```

<label>Año *:</label>

```

```

<input type="text" name="nombene2" placeholder="Digite
el año" style="width:230px">

```

```

</div>

```

```

</H3>

```

```

</center>

```

```

</span>

```

```

<span class="button">

```

```

<center>

```

```

<input type="submit" value="Registrar" style="width:100px">

```

```

<input type="submit" value="Volver" style="width: 100px"

```

```

formaction="antsisben.php">

```

```

</center>

```

```

</span>

```

```

</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family:
Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
    <p></p>
</form>
</body>
</html>

```

9.1.44 archivo que lleva los datos de registro antigüedad sisben a la base de datos

```

<?php
include ("blodese.php");
include ("conexion.php");
if (isset($_POST["nombene2"]) and $_POST["nombene2"] != "")
{
    $nombre = $_POST["nombene2"];
    $con = mysqli_query($con , "insert into antigu_sisben
(NOMASISBEN)VALUES('$nombre')");
    ?>
    <script type="text/javascript">
        alert("Dato agregado exitosamente");
        window.location.href='antsisben.php';
    </script>
    <?php
}
else
{
    ?>
    <script type="text/javascript">
        alert("Por favor, complete el formulario");
        window.location.href='regantsisben.php';
    </script>
    <?php
}
?>

```

9.1.45 pantalla modificar antigüedad sisben

```
<?php
include ("blodese.php");
$consulta = consultarantsisben($_GET['ID_ASISBEN']);
function consultarantsisben($ID_ASISBEN)
{
    include ("conexion.php");
    $con = mysqli_query($con , "select * from antigu_sisben where antigu_sisben .
    ID_ASISBEN = '$ID_ASISBEN'");
    $filas=mysqli_fetch_assoc($con);
    return
    [
        $filas['NOMASISBEN']
    ];
}
?>
<!doctype html>
<html>
<head>
<title> Base de datos</title>
<link rel="stylesheet" type="text/css" href="estilopri.css">
<style type="text/css">
    table
    {
        height: 415px;
        width: 342px;
        text-align: left;
        display: block;
        overflow-x: auto;
    }
    th
    {
        position: static;
    }
</style>
</head>
<body>
<form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-
size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid
#CCC;"><div><center>
<p>Modifique beneficio</p>
<div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
<div style = "float: right;"><a href="salir.php">Salir</a></div>
```

```

        <p>&nbsp;</p>
    </center></div>
</form>
<form id="form2"method="POST" action='index.php' margin='0' style="float:
left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <?php
        include ("menuopciones.php");
    ?>
</form>
<form id="form3"method="POST" action='index.php' margin='0' style="float:
left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <spam class="text">
        Los espacios marcados con asterisco (*) son obligatorios
    </spam>
</form>
<form id="form4"method="POST" action='modantsisben2.php' margin='0' style="float:
left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <span class="text">
        <center>
            <H3>
                <div>
                    <input type="hidden" name="ID_ASISBEN" value="<?php echo
$_GET['ID_ASISBEN']?>">
                    <label>Año *:</label>
                    <input type="text" name="NOMASISBEN"; value="<?php echo
$consulta[0] ?>"style="width:230px">
                </div>
            </H3>
        </center>
    </span>
    <span class="button">
        <center>
            <input type="submit" value="Modificar" style="width:100px">
            <input type="submit" value="Volver" style="width: 100px"
formaction="antsisben.php">
        </center>
    </span>
</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family:
Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
    <p></p>

```

```
</form>
</body>
</html>
```

9.1.46 archivo que lleva los datos de modificar antigüedad sisben a la base de datos

```
<?php
include ("blodese.php");
modificarbeneicio1($_POST['NOMASISBEN'], $_POST['ID_ASISBEN']);
function modificarbeneicio1($NOMASISBEN,$ID_ASISBEN)
{
    include ("conexion.php");
    $con = mysqli_query($con , "update antigu_sisben set NOMASISBEN='
    ".$NOMASISBEN."' where ID_ASISBEN='".$ID_ASISBEN."'");
}
?>
<script type="text/javascript">
    alert("Dato modificado exitosamente");
    window.location.href='antsisben.php';
</script>
```

9.1.47 archivo que borra datos de antigüedad sisben de la base de datos

```
<?php
include ("blodese.php");
Eliminarantsisben($_GET['NOMASISBEN']);
function Eliminarantsisben($NOMASISBEN)
{
    include ("conexion.php");
    $con = mysqli_query($con , "delete from antigu_sisben where antigu_sisben .
    NOMASISBEN = '$NOMASISBEN'");
}
?>
<script type="text/javascript">
    alert("Dato eliminado exitosamente");
    window.location.href='antsisben.php';
</script>
```


9.1.48 pantalla tipos de poblacion

```
<?php
    include ("blodese.php");
    include ("conexion.php");
?>
<!doctype html>
<html>
<head>
    <meta http-equiv="Content-type" content="text/html; charset=utf-8">
    <meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=no">
    <title>Poblacion registrada</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/1.10.19/css/jquery.dataTables.min.css">
    <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/buttons/1.5.6/css/buttons.dataTables.min.css">
    <style type="text/css">
        input
        {
            width: 200px;
        }
        table
        {
            width: 74%;
            text-align: left;
            display: block;
            overflow-x: auto;
        }
        th
        {
            position: static;
        }
    </style>
    <script type="text/javascript" language="javascript" src="https://code.jquery.com/jquery-3.3.1.js"></script>
    <script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/1.10.19/js/jquery.dataTables.min.js"></script>
    <script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/dataTables.buttons.min.js"></script>
    <script type="text/javascript" language="javascript"
src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
    <script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/buttons.html5.min.js"></script>
    <script type="text/javascript" class="init">
```

```

$(document).ready(function() {
    $('#example').DataTable( {
        dom: 'Bfrtip',
        buttons: [
            ]
    });
});
</script>
</head>
<body>
<form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid #CCC;"><div><center>
<p>Poblacion registrada</p>
<div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
<div style = "float: right;"><a href="salir.php">Salir</a></div>
<p>&nbsp;</p>
</center></div>
</form>
<form id="form2"method='POST' action='index.php' margin='0' style="float: left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<?php
include ("menuopciones.php");
?>
</form>
<form id="form3"method='POST' action='index.php' margin='0' style="float: left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<center>
<p></p>
<input name="registro" type="submit" id="enviar2" formaction="regpoblacion.php" value="Ingresar nuevo tipo de poblacion" style="width:300px">
<p></p>
</center>
</form>
<form id="form4"method='GET' margin='0' style="float: left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
</form>
<form id="form5" action="fuid.php" method='POST' margin='0' style="float: left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">

```

```

<center>
  <div class="fw-container" style="width: 316px">
    <table id="example" class="display" style="width:100%">
      <thead>
        <tr>
          <th colspan="3"><center>Tipos de poblacion
especial</center></th>
        </tr>
        <tr>
          <th>Nombre</th>
          <th>Modificar</th>
          <th>Eliminar</th>
        </tr>
      </thead>
      <tbody>
        <?php
function lacon()
{
    global $con, $cons;
    $sql ='select * from poblacion ORDER BY
NOMPOBLACION ASC';
    return $con->query($sql);
}
$cons = lacon();
while($row = $cons->fetch_assoc())
{
    echo "<tr>";
    echo "<td>"; echo
$row['NOMPOBLACION']; echo "</td>";
    echo "<td><center><a
href='modpoblacion.php?ID_POBLACION=
".$row['ID_POBLACION']."'> <button
type='button' class='btn btn-
success'>Modificar</button></a>
</center></td>";
    echo "<td><center><a
href='elim poblacion.php?NOMPOBLACIO
N=".$row['NOMPOBLACION']."'>";
    ?>
    <button
formaction="elim poblacion.php?NOMPOB
LACION=?>?>" class="btn
btn-danger btn-sm" onclick="return

```

```

confirmarRegistro())<span class="fa fa-
trash"></span>Eliminar</button>
<?php
echo "</a></center></td></tr>";
}
?>
</tbody>
</table>
</div>
</center>
</form>
</body>
</html>

```

9.1.49 pantalla registro tipos de poblacion

```

<?php
include ("blodese.php");
?>
<!doctype html>
<html>
<head>
<title> Base de datos</title>
<link rel="stylesheet" type="text/css" href="estilopri.css">
<style type="text/css">
table
{
height: 415px;
width: 342px;
text-align: left;
display: block;
overflow-x: auto;
}
th
{
position: static;
}
</style>
</head>
<body>
<form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-
size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid

```

```

#CCC;"><div><center>
    <p>Ingrese la poblacion especial:</p>
    <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
    <div style = "float: right;"><a href="salir.php">Salir</a></div>
    <p>&nbsp;</p>
</center></div>
</form>
<form id="form2"method='POST' action='index.php' margin='0' style="float:
left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <?php
        include ("menuopciones.php");
    ?>
</form>
<form id="form3"method='POST' action='index.php' margin='0' style="float:
left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <spam class="text">
        Los espacios marcados con asterisco (*) son obligatorios
    </spam>
</form>
<form id="form4"method='POST' action='cregpoblacion.php' margin='0' style="float:
left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <span class="text">
        <center>
            <H3>
                <div>
                    <label>Poblacion *:</label>
                    <input type="text" name="nomben1" placeholder="Digite el
nombre"style="width:230px">
                </div>
            </H3>
        </center>
    </span>
    <span class="button">
        <center>
            <input type="submit" value="Registrar" style="width:100px">
            <input type="submit" value="Volver" style="width: 100px"
formaction="poblacion.php">
        </center>
    </span>
</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family:

```

```

        Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
        <p></p>
    </form>
</body>
</html>

```

9.1.50 archivo que lleva los datos de registro tipos de poblacion a la base de datos

```

<?php
    include ("blodese.php");
    include ("conexion.php");
    if (isset($_POST["nomben1"]) and $_POST["nomben1"] != "")
    {
        $nombre = $_POST["nomben1"];
        $con = mysqli_query($con , "insert into poblacion
        (NOMPOBLACION)VALUES('$nombre')");
        ?>
        <script type="text/javascript">
            alert("Tipo de poblcion agregada exitosamente");
            window.location.href='poblacion.php';
        </script>
        <?php
    else
    {
        ?>
        <script type="text/javascript">
            alert("Por favor, complete el formulario");
            window.location.href='regpoblacion.php';
        </script>
        <?php
    }
?>

```

9.1.51 pantalla modificar tipos de poblacion

```

<?php
    include ("blodese.php");
    $consulta = consultarpoblacion($_GET['ID_POBLACION']);
    function consultarpoblacion($ID_POBLACION)
    {

```

```

include ("conexion.php");
$con = mysqli_query($con , "select * from poblacion where poblacion . ID_POBLACION =
'$ID_POBLACION'");
$filas=mysqli_fetch_assoc($con);
return
    [
        $filas['NOMPOBLACION']
    ];
}
?>
<!doctype html>
<html>
<head>
    <title> Base de datos</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <style type="text/css">
        table
        {
            height: 415px;
            width: 342px;
            text-align: left;
            display: block;
            overflow-x: auto;
        }
        th
        {
            position: static;
        }
    </style>
</head>
<body>
    <form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-
size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid
#CCC;"><div><center>
        <p>Modifique la poblacion</p>
        <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
        <div style = "float: right;"><a href="salir.php">Salir</a></div>
        <p>&nbsp;</p>
    </center></div>
</form>
    <form id="form2" method='POST' action='index.php' margin='0' style="float:
left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
        <?php

```

```

        include ("menuopciones.php");
    ?>
</form>
<form id="form3"method='POST' action='index.php' margin='0' style="float:
left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <spam class="text">
        Los espacios marcados con asterisco (*) son obligatorios
    </spam>
</form>
<form id="form4"method='POST' action='modpoblacion2.php' margin='0' style="float:
left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <span class="text">
        <center>
            <H3>
                <div>
                    <input type="hidden" name="ID_POBLACION" value="<?php echo
                    $_GET['ID_POBLACION']?>">
                    <label>Nombre Poblacion *:</label>
                    <input type="text" name="NOMPOBLACION"; value="<?php echo
                    $consulta[0] ?>"style="width:230px">
                </div>
            </H3>
        </center>
    </span>
    <span class="button">
        <center>
            <input type="submit" value="Modificar" style="width:100px">
            <input type="submit" value="Volver" style="width: 100px"
            formaction="poblacion.php">
        </center>
    </span>
</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family:
Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
    <p></p>
</form>
</body>
</html>

```


9.1.52 archivo que lleva los datos de modificar tipos de poblacion a la base de datos

```
<?php
    include ("blodese.php");
    modificarpoblacion($_POST['NOMPOBLACION'], $_POST['ID_POBLACION']);
    function modificarpoblacion($NOMPOBLACION,$ID_POBLACION)
    {
        include ("conexion.php");
        $con = mysqli_query($con , "update poblacion set
        NOMPOBLACION=".$NOMPOBLACION." where ID_POBLACION='
        ".$ID_POBLACION."'");
    }
?>
<script type="text/javascript">
    alert("Poblacion modificada exitosamente");
    window.location.href='poblacion.php';
</script>
```

9.1.53 archivo que borra datos de tipos de poblacion de la base de datos

```
<?php
    include ("blodese.php");
?>
<?php
    Eliminarpoblacion($_GET['NOMPOBLACION']);
    function Eliminarpoblacion($NOMPOBLACION)
    {
        include ("conexion.php");
        $con = mysqli_query($con , "delete from poblacion where poblacion . NOMPOBLACION =
        '$NOMPOBLACION'");
    }
?>
<script type="text/javascript">
    alert("Tipo de poblacion eliminado exitosamente");
    window.location.href='poblacion.php';
</script>
```

9.1.54 pantalla universidades registradas

```
<?php
    include ("blodese.php");
    include ("conexion.php");
?>
<!doctype html>
<html>
<head>
    <meta http-equiv="Content-type" content="text/html; charset=utf-8">
    <meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=no">
    <title>Universidades</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/1.10.19/css/jquery.dataTables.min.css">
    <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/buttons/1.5.6/css/buttons.dataTables.min.css">
    <style type="text/css">
        input
        {
            width: 200px;
        }
        table
        {
            width: 74%;
            height: 450px;
            text-align: left;
            display: block;
            overflow-x: auto;
        }
        th
        {
            position: static;
        }
    </style>
    <script type="text/javascript" language="javascript" src="https://code.jquery.com/jquery-3.3.1.js"></script>
    <script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/1.10.19/js/jquery.dataTables.min.js"></script>
    <script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/dataTables.buttons.min.js"></script>
    <script type="text/javascript" language="javascript"
src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
    <script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/buttons.html5.min.js"></script>
```

```

<script type="text/javascript" class="init">
$(document).ready(function() {
    $('#example').DataTable( {
        dom: 'Bfrtip',
        buttons: [
            ]
    });
});
</script>
</head>
<body>
<form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid #CCC;"><div><center>
<p>Universidades</p>
<div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
<div style = "float: right;"><a href="salir.php">Salir</a></div>
<p>&nbsp;</p>
</center></div>
</form>
<form id="form2" method='POST' action='index.php' margin='0' style="float: left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<?php
include ("menuopciones.php");
?>
</form>
<form id="form3" method='POST' action='index.php' margin='0' style="float: left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<center>
<p></p>
<input name="registro" type="submit" id="enviar2"
formaction="reguniversidad.php" value="Registrar nueva universidad"
style="width:300px">
<p></p>
</center>
</form>
<form id="form4" method='GET' margin='0' style="float: left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
</form>
<form id="form5" action="fuid.php" method='POST' margin='0' style="float:

```

left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">

```
<center>
  <div class="fw-container" style="width: 660px">
    <table id="example" class="display" style="width:100%">
      <thead>
        <tr>
          <th colspan="4"><center>Universidades</center></th>
        </tr>
        <tr>
          <th>Universidad</th>
          <th>Porcentaje universidad</th>
          <th>Modificar</th>
          <th>Eliminar</th>
        </tr>
      </thead>
      <tbody>
        <?php
          function lacon()
          {
            global $con, $cons;
            $sql ='select * from universidad ORDER BY
            NOMUNIVERSIDAD ASC';
            return $con->query($sql);
          }
          $cons = lacon();
          while($row = $cons->fetch_assoc())
          {
            echo "<tr>";
            echo "<td>"; echo $row['NOMUNIVERSIDAD'];
            echo "</td>";
            echo "<td><center>"; echo
            $row['UNIPORCENTAJE']; echo "</center></td>";
            echo "<td><center><a
            href='modiuniversidad.php?ID_UNIVERSIDAD=
            ".$row['ID_UNIVERSIDAD']."'> <button
            type='button' class='btn btn-success'>Modificar
            </button></a></center></td>";
            echo "<td><center><a
            href='elimuniversidad.php?NOMUNIVERSIDAD=
            ".$row['NOMUNIVERSIDAD']."'>";
            ?>
            <button
            formaction="elimuniversidad.php?NOMUNIVERSI
```

```

        ?>
        </tbody>
    </table>
</div>
</center>
</form>
</body>
</html>

```

```

DAD=<?php echo $row['NOMUNIVERSIDAD']
?>" class="btn btn-danger btn-sm" onclick="return
confirmarRegistro()"><span class="fa fa-
trash"></span>Eliminar</button>
<?php
echo "</a></center></td></tr>";

```

9.1.55 pantalla registro universidades

```

<?php
    include ("blodese.php");
?>
<!doctype html>
<html>
<head>
    <title> Base de datos</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <style type="text/css">
        table
        {
            height: 415px;
            width: 342px;
            text-align: left;
            display: block;
            overflow-x: auto;
        }
        th
        {
            position: static;
        }
    </style>
</head>
<body>

```

```

<form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-
size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid
#CCC;"><div><center>
<p>Ingrese la universidad:</p>
<div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
<div style = "float: right;"><a href="salir.php">Salir</a></div>
<p>&nbsp;</p>
</center></div>
</form>

```

```

<form id="form2" method='POST' action='index.php' margin='0' style="float:
left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<?php

```

```

include ("menuopciones.php");
?>
</form>

```

```

<form id="form3" method='POST' action='index.php' margin='0' style="float:
left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<spam class="text">
Los espacios marcados con asterisco (*) son obligatorios
</spam>
</form>

```

```

<form id="form4" method='POST' action='creguniversidad.php' margin='0' style="float:
left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<span class="text">
<center>
<H3>
<div><label>Nombre *:</label>
<input type="text" name="nomuni" placeholder="Digite el nombre."
style="width:500px"></div>
<p></p>
<div><label>porcentaje sin signo *:</label>
<input type="text" name="porcentaje" placeholder="Digite el
porcentaje" style="width:230px"></div>
</H3>
</center>
</span>
<span class="button">
<center>
<input type="submit" value="Registrar" style="width:100px">
<input type="submit" value="Volver" style="width: 100px"
formaction="universidad.php">
</center>

```

```

        </span>
    </form>
    <form for="form5" action='index.php' method='POST' margin='0' style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family:
Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
        <p></p>
    </form>
</body>
</html>

```

9.1.56 archivo que lleva los datos de registro universidades a la base de datos

```

<?php
include ("blodese.php");
include ("conexion.php");
if (isset($_POST["nomuni"], $_POST["porcentajeu"]) and $_POST["nomuni"] != "" and
$_POST["porcentajeu"] != "")
{
    $nombre = $_POST["nomuni"];
    $porcentaje = $_POST["porcentajeu"];
    $con = mysqli_query($con , "insert into universidad
(NOMUNIVERSIDAD,UNIPORCENTAJE)VALUES('$nombre','$porcentaje')");
    ?>
    <script type="text/javascript">
        alert("Universidad agregada exitosamente");
        window.location.href='universidad.php';
    </script>
    <?php
else
{
    ?>
    <script type="text/javascript">
        alert("Por favor, complete el formulario");
        window.location.href='reguniversidad.php';
    </script>
    <?php
}
?>

```

9.1.57 pantalla modificar universidades

```
<?php
include ("blodese.php");
$consulta = consultaruniversidad($_GET['ID_UNIVERSIDAD']);
function consultaruniversidad($ID_UNIVERSIDAD)
{
include ("conexion.php");
$con = mysqli_query($con , "select * from universidad where universidad .
ID_UNIVERSIDAD = '$ID_UNIVERSIDAD'");
$filas=mysqli_fetch_assoc($con);
return
[
$filas['NOMUNIVERSIDAD'],
$filas['UNIPORCENTAJE']
];
}
?>
<!doctype html>
<html>
<head>
<title> Base de datos</title>
<link rel="stylesheet" type="text/css" href="estilopri.css">
<style type="text/css">
table
{
height: 415px;
width: 342px;
text-align: left;
display: block;
overflow-x: auto;
}
th
{
position: static;
}
</style>
</head>
<body>
<form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-
size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid
#CCC;"><div><center>
<p>Modifique la Universidad</p>
<div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
```



```

        <div style = "float: right;"><a href="salir.php">Salir</a></div>
        <p>&nbsp;</p>
    </center></div>
</form>
<form id="form2"method='POST' action='index.php' margin='0' style="float:
left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <?php
        include ("menuopciones.php");
    ?>
</form>
<form id="form3"method='POST' action='index.php' margin='0' style="float:
left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <spam class="text">
        Los espacios marcados con asterisco (*) son obligatorios
    </spam>
</form>
<form id="form4"method='POST' action='modiuniversidad2.php' margin='0' style="float:
left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <span class="text">
        <center>
            <H3>
                <div>
                    <input type="hidden" name="ID_UNIVERSIDAD" value="<?php
echo $_GET['ID_UNIVERSIDAD']?>">
                    <label>Nombre *:</label>
                    <input type="text" name="NOMUNIVERSIDAD";
                    value="<?php echo $consulta[0] ?>"style="width:500px">
                    <p></p>
                    <label>porcentaje sin signo *:</label>
                    <input type="text" name="UNIPORCENTAJE";
                    value="<?php echo $consulta[1] ?>"style="width:230px">
                </div>
            </H3>
        </center>
    </span>
    <span class="button">
        <center>
            <input type="submit" value="Modificar" style="width:100px">
            <input type="submit" value="Volver" style="width: 100px"
            formaction="universidad.php">
        </center>
    </span>

```

```

</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family:
Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
    <p></p>
</form>
</body>
</html>

```

9.1.58 archivo que lleva los datos de modificar universidades a la base de datos

```

<?php
    include ("blodese.php");
    modificaruniversidad($_POST['NOMUNIVERSIDAD'], $_POST['UNIPORCENTAJE'],
    $_POST['ID_UNIVERSIDAD']);
    function modificaruniversidad($NOMUNIVERSIDAD,$UNIPORCENTAJE,$ID_UNIVERSIDAD)
    {
        include ("conexion.php");
        $con = mysqli_query($con , "update universidad set NOMUNIVERSIDAD =
        ".$NOMUNIVERSIDAD.",UNIPORCENTAJE=".$UNIPORCENTAJE." where
        ID_UNIVERSIDAD=".$ID_UNIVERSIDAD."");
    }
?>
<script type="text/javascript">
    alert("Universidad modificada exitosamente");
    window.location.href='universidad.php';
</script>

```

9.1.59 archivo que borra datos de universidades de la base de datos

```

<?php
    include ("blodese.php");
?>
<?php
    Eliminaruniversidad($_GET['NOMUNIVERSIDAD']);
    function Eliminaruniversidad($NOMUNIVERSIDAD)
    {
        include ("conexion.php");
        $con = mysqli_query($con , "delete from universidad where universidad .
        NOMUNIVERSIDAD = '$NOMUNIVERSIDAD'");
    }

```

```
?>
<script type="text/javascript">
    alert("Universidad eliminada exitosamente");
    window.location.href='universidad.php';
</script>
```

9.1.60 pantalla carreras registradas

```
<?php
    include ("blodese.php");
    include ("conexion.php");
?>
<!doctype html>
<html>
<head>
    <meta http-equiv="Content-type" content="text/html; charset=utf-8">
    <meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=no">
    <title>Carreras</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/1.10.19/css/jquery.dataTables.min.css">
    <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/buttons/1.5.6/css/buttons.dataTables.min.css">
    <style type="text/css">
        input
        {
            width: 200px;
        }
        table
        {
            width: 74%;
            text-align: left;
            display: block;
            overflow-x: auto;
        }
        th
        {
            position: static;
        }
    </style>
    <script type="text/javascript" language="javascript" src="https://code.jquery.com/jquery-3.3.1.js"></script>
    <script type="text/javascript" language="javascript"
```

```

src="https://cdn.datatables.net/1.10.19/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/dataTables.buttons.min.js"></script>
<script type="text/javascript" language="javascript"
src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
<script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/buttons.html5.min.js"></script>
<script type="text/javascript" class="init">
$(document).ready(function() {
    $('#example').DataTable( {
        dom: 'Bfrtip',
        buttons: [
            ]
    } );
} );
</script>
</head>
<body>
<form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid #CCC;"><div><center>
<p>Carreras</p>
<div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
<div style = "float: right;"><a href="salir.php">Salir</a></div>
<p>&nbsp;</p>
</center></div>
</form>
<form id="form2"method='POST' action='index.php' margin='0' style="float:left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<?php
include ("menuopciones.php");
?>
</form>
<form id="form3"method='POST' action='index.php' margin='0' style="float:left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<center>
<p></p>
<input name="registrar" type="submit" id="enviar2" formaction="regcarrera.php" value="Registrar nueva carrera" style="width:300px">
<p></p>
</center>
</form>

```

```
<form id="form4" method='GET' margin='0' style="float: left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
```

```
</form>
```

```
<form id="form5" action="fuid.php" method='POST' margin='0' style="float: left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
```

```
<center>
```

```
<div class="fw-container" style="width: 660px">
```

```
<table id="example" class="display" style="width:100%">
```

```
<thead>
```

```
<tr>
```

```
<th colspan="3"><center>Carreras</center></th>
```

```
</tr>
```

```
<tr>
```

```
<th>Nombre</th>
```

```
<th>Modificar</th>
```

```
<th>Eliminar</th>
```

```
</tr>
```

```
</thead>
```

```
<tbody>
```

```
<?php
```

```
function lacon()
```

```
{
```

```
    global $con, $cons;
```

```
    $sql ='select * from carreras ORDER BY  
    NOMCARRERA ASC';
```

```
    return $con->query($sql);
```

```
}
```

```
$cons = lacon();
```

```
while($row = $cons->fetch_assoc())
```

```
{
```

```
    echo "<tr>";
```

```
    echo "<td>"; echo $row['NOMCARRERA']; echo  
    "</td>";
```

```
    echo "<td> <a
```

```
    href='modcarrera.php?ID_CARRERA=
```

```
    ".$row['ID_CARRERA']."'> <button type='button'  
    class='btn btn-success'>Modificar</button> </a>  
    </td>";
```

```
    echo "<td><center><a
```

```
    href='elimcarrera.php?NOMCARRERA=".$row['NOMCARRERA']."'>";
```

```
?>
```

```

        <button
        formaction="elimcarrera.php?NOMCARRERA=<?
        php echo $row['NOMCARRERA'] ?>" class="btn
        btn-danger btn-sm" onclick="return
        confirmarRegistro()"><span class="fa fa-
        trash"></span>Eliminar</button>
        <?php
        echo "</a></center></td></tr>";
    }
}
?>
</tbody>
</table>
</div>
</center>
</form>
</body>
</html>

```

9.1.61 pantalla registro carreras

```

<?php
    include ("blodese.php");
?>
<!doctype html>
<html>
<head>
    <title> Base de datos</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <style type="text/css">
        table
        {
            height: 415px;
            width: 342px;
            text-align: left;
            display: block;
            overflow-x: auto;
        }
        th
        {
            position: static;
        }
    </style>
</head>

```

```

<body>
  <form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid #CCC;"><div><center>
    <p>Ingrese la carrera:</p>
    <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
    <div style = "float: right;"><a href="salir.php">Salir</a></div>
    <p>&nbsp;</p>
  </center></div>
</form>
  <form id="form2"method='POST' action='index.php' margin='0' style="float:left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <?php
      include ("menuopciones.php");
    ?>
  </form>
  <form id="form3"method='POST' action='index.php' margin='0' style="float:left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <spam class="text">
      Los espacios marcados con asterisco (*) son obligatorios
    </spam>
  </form>
  <form id="form4"method='POST' action='cregcarrera.php' margin='0' style="float:left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <span class="text">
      <center>
        <H3>
          <div><label>Nombre *:</label>
          <input type="text" name="nomuni" placeholder="Digite el nombre." style="width:500px"></div>
        </H3>
      </center>
    </span>
    <span class="button">
      <center>
        <input type="submit" value="Registrar" style="width:100px">
        <input type="submit" value="Volver" style="width: 100px" formaction="carrera.php">
      </center>
    </span>
  </form>

```

```

<form for="form5" action='index.php' method='POST' margin='0' style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family:
Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
    <p></p>
</form>
</body>
</html>

```

9.1.62 archivo que lleva los datos de registro carreras a la base de datos

```

<?php
include ("blodese.php");
include ("conexion.php");
if (isset($_POST["nomuni"]) and $_POST["nomuni"] != "")
{
    $nombre = $_POST["nomuni"];
    $con = mysqli_query($con , "insert into carreras
(NOMCARRERA)VALUES('$nombre')");
    ?>
    <script type="text/javascript">
        alert("Carrera agregada exitosamente");
        window.location.href='carrera.php';
    </script>
    <?php
}
else
{
    ?>
    <script type="text/javascript">
        alert("Por favor, complete el formulario");
        window.location.href='regcarrera.php';
    </script>
    <?php
}
?>

```


9.1.63 pantalla modificar carreras

```
<?php
    include ("blodese.php");
    $consulta = consultarcarrera($_GET['ID_CARRERA']);
    function consultarcarrera($ID_CARRERA)
    {
        include ("conexion.php");
        $con = mysqli_query($con , "select * from carreras where carreras . ID_CARRERA =
        '$ID_CARRERA'");
        $filas=mysqli_fetch_assoc($con);
        return
        [
            $filas['NOMCARRERA']
        ];
    }
?>
<!doctype html>
<html>
<head>
    <title> Base de datos</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <style type="text/css">
        table
        {
            height: 415px;
            width: 342px;
            text-align: left;
            display: block;
            overflow-x: auto;
        }
        th
        {
            position: static;
        }
    </style>
</head>
<body>
    <form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-
    size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid
    #CCC;"><div><center>
        <p>Modifique la carrera</p>
        <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
        <div style = "float: right;"><a href="salir.php">Salir</a></div>
```

```

        <p>&nbsp;</p>
    </center></div>
</form>
<form id="form2"method="POST" action='index.php' margin='0' style="float:
left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <?php
        include ("menuopciones.php");
    ?>
</form>
<form id="form3"method="POST" action='index.php' margin='0' style="float:
left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <spam class="text">
        Los espacios marcados con asterisco (*) son obligatorios
    </spam>
</form>
<form id="form4"method="POST" action='modcarrera2.php' margin='0' style="float:
left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <span class="text">
        <center>
            <H3>
                <div>
                    <input type="hidden" name="ID_CARRERA" value="<?php echo
$_GET['ID_CARRERA']?>">
                    <label>Nombre *:</label>
                    <input type="text" name="NOMCARRERA"; value="<?php echo
$consulta[0] ?>"style="width:500px">
                </div>
            </H3>
        </center>
    </span>
    <span class="button">
        <center>
            <input type="submit" value="Modificar" style="width:100px">
            <input type="submit" value="Volver" style="width: 100px"
formaction="carrera.php">
        </center>
    </span>
</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family:
Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
    <p></p>

```

```
</form>
</body>
</html>
```

9.1.64 archivo que lleva los datos de modificar carreras a la base de datos

```
<?php
include ("blodese.php");
modificarbeneicio1($_POST['NOMCARRERA'], $_POST['ID_CARRERA']);
function modificarbeneicio1($NOMCARRERA,$ID_CARRERA)
{
    include ("conexion.php");
    $con = mysqli_query($con , "update carreras set NOMCARRERA=".$NOMCARRERA."
    where ID_CARRERA=".$ID_CARRERA."");
}
?>
<script type="text/javascript">
    alert("Carrera modificada exitosamente");
    window.location.href='carrera.php';
</script>
```

9.1.65 archivo que borra datos de carreras de la base de datos

```
<?php
include ("blodese.php");
Eliminarcarrera($_GET['NOMCARRERA']);
function Eliminarcarrera($NOMCARRERA)
{
    include ("conexion.php");
    $con = mysqli_query($con , "delete from carreras where carreras . NOMCARRERA =
    '$NOMCARRERA'");
}
?>
<script type="text/javascript">
    alert("Carrera eliminada exitosamente");
    window.location.href='carrera.php';
</script>
```

9.1.66 pantalla ciudad registradas

```
<?php
    include ("blodese.php");
    include ("conexion.php");
?>
<!doctype html>
<html>
<head>
    <meta http-equiv="Content-type" content="text/html; charset=utf-8">
    <meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=no">
    <title>Ciudades a las que viajan</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/1.10.19/css/jquery.dataTables.min.css">
    <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/buttons/1.5.6/css/buttons.dataTables.min.css">
    <style type="text/css">
        input
        {
            width: 200px;
        }
        table
        {
            width: 74%;
            text-align: left;
            display: block;
            overflow-x: auto;
        }
        th
        {
            position: static;
        }
    </style>
    <script type="text/javascript" language="javascript" src="https://code.jquery.com/jquery-3.3.1.js"></script>
    <script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/1.10.19/js/jquery.dataTables.min.js"></script>
    <script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/dataTables.buttons.min.js"></script>
    <script type="text/javascript" language="javascript"
src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
    <script type="text/javascript" language="javascript"
src="https://cdn.datatables.net/buttons/1.5.6/js/buttons.html5.min.js"></script>
    <script type="text/javascript" class="init">
```

```

$(document).ready(function() {
    $('#example').DataTable( {
        dom: 'Bfrtip',
        buttons: [
            ]
    });
});
</script>
</head>
<body>
<form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid #CCC;"><div><center>
<p>Ciudades a las que viajan</p>
<div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
<div style = "float: right;"><a href="salir.php">Salir</a></div>
<p>&nbsp;</p>
</center></div>
</form>
<form id="form2"method='POST' action='index.php' margin='0' style="float:left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<?php
include ("menuopciones.php");
?>
</form>
<form id="form3"method='POST' action='index.php' margin='0' style="float:left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
<center>
<p></p>
<input name="registrar" type="submit" id="enviar2" formaction="regciudad.php" value="Registrar nueva ciudad" style="width:300px">
<p></p>
</center>
</form>
<form id="form4"method='GET' margin='0' style="float:left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
</form>
<form id="form5" action="fuid.php" method='POST' margin='0' style="float:left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">

```

```

<center>
  <div class="fw-container" style="width: 298px">
    <table id="example" class="display" style="width:100%">
      <thead>
        <tr>
          <th colspan="3"><center>Ciudad</center></th>
        </tr>
        <tr>
          <th>Nombre</th>
          <th>Modificar</th>
          <th>Eliminar</th>
        </tr>
      </thead>
      <tbody>
        <?php
          function lacon()
          {
            global $con, $cons;
            $sql ='select * from ciudadviaja ORDER BY
              NOMCIUDAD ASC';
            return $con->query($sql);
          }
          $cons = lacon();
          while($row = $cons->fetch_assoc())
          {
            echo "<tr>";
            echo "<td>"; echo $row['NOMCIUDAD']; echo
              "</td>";
            echo "<td><center><a
              href='modciudad.php?ID_CIUDAD=". $row['ID_CI
              UDAD']. "'> <button type='button' class='btn btn-
              success'>Modificar</button></a></center></td>";
            echo "<td><center><a
              href='elimciudad.php?NOMCIUDAD=". $row['NOM
              CIUDAD']. "'>";
              ?>
              <button
                formation="elimciudad.php?NOMCIUDAD=?ph
                p echo $row['NOMCIUDAD'] ?>" class="btn btn-
                danger btn-sm" onclick="return
                confirmarRegistro()"><span class="fa fa-
                trash"></span>Eliminar</button>
              <?php
                echo "</a></center></td></tr>";
          }
        </tbody>
      </table>
    </div>
  </center>

```

```

?>
</tbody>
</table>
</div>
</center>
</form>
</body>
</html>

```

9.1.67 pantalla registro ciudad

```

<?php
    include ("blodese.php");
?>
<!doctype html>
<html>
<head>
    <title> Base de datos</title>
    <link rel="stylesheet" type="text/css" href="estilopri.css">
    <style type="text/css">
        table
        {
            height: 415px;
            width: 342px;
            text-align: left;
            display: block;
            overflow-x: auto;
        }
        th
        {
            position: static;
        }
    </style>
</head>

<body>
    <form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid #CCC;"><div><center>
        <p>Ingrese la ciudad:</p>
        <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
        <div style = "float: right;"><a href="salir.php">Salir</a></div>
        <p>&nbsp;</p>

```

```

</center></div>
</form>
<form id="form2"method='POST' action='index.php' margin='0' style="float:
left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  <?php
    include ("menuopciones.php");
  ?>
</form>
<form id="form3"method='POST' action='index.php' margin='0' style="float:
left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  <spam class="text">
    Los espacios marcados con asterisco (*) son obligatorios
  </spam>
</form>
<form id="form4"method='POST' action='cregciudad.php' margin='0' style="float:
left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  <span class="text">
    <center>
      <H3>
        <div><label>Nombre *:</label>
        <input type="text" name="nomuni" placeholder="Digite el nombre."
        style="width:500px"></div>
      </H3>
    </center>
  </span>
  <span class="button">
    <center>
      <input type="submit" value="Registrar" style="width:100px">
      <input type="submit" value="Volver" style="width: 100px"
      formaction="ciudad.php">
    </center>
  </span>
</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family:
Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
<p></p>
</form>
</body>
</html>

```


9.1.68 archivo que lleva los datos de registro ciudad a la base de datos

```
<?php
    include ("blodese.php");
    include ("conexion.php");
    if (isset($_POST["nomuni"]) and $_POST["nomuni"] != "")
    {
        $nombre = $_POST["nomuni"];
        $con = mysqli_query($con , "insert into ciudadviaja(NOMCIUDAD)VALUES('$nombre')");
        ?>
        <script type="text/javascript">
            alert("Ciudad agregada exitosamente");
            window.location.href='ciudad.php';
        </script>
    }
    else
    {
        ?>
        <script type="text/javascript">
            alert("Por favor, complete el formulario");
            window.location.href='regciudad.php';
        </script>
    }
<?php
?>
```

9.1.69 pantalla modificar ciudad

```
<?php
    include ("blodese.php");
    $consulta = consultarciudad($_GET['ID_CIUADAD']);
    function consultarciudad($ID_CIUADAD)
    {
        include ("conexion.php");
        $con = mysqli_query($con , "select * from ciudadviaja where ciudadviaja . ID_CIUADAD =
'$ID_CIUADAD'");
        $filas=mysqli_fetch_assoc($con);
        return
        [
            $filas['NOMCIUDAD']
        ];
    }
}
```

```

?>
<!doctype html>
<html>
<head>
  <title> Base de datos</title>
  <link rel="stylesheet" type="text/css" href="estilopri.css">
  <style type="text/css">
    table
    {
      height: 415px;
      width: 342px;
      text-align: left;
      display: block;
      overflow-x: auto;
    }
  th
  {
    position: static;
  }
</style>
</head>

<body>
  <form id="form1" style="width:98.72%;height:2.6%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:right;padding-right:1%;border: 1px solid #CCC;"><div><center>
    <p>Modifique la ciudad</p>
    <div style="float: left;"><?php echo $_SESSION["usuarios"]?></div>
    <div style = "float: right;"><a href="salir.php">Salir</a></div>
    <p>&nbsp;</p>
  </center></div>
</form>
  <form id="form2"method='POST' action='index.php' margin='0' style="float: left;width:16.1%;height:95.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <?php
      include ("menuopciones.php");
    ?>
  </form>
  <form id="form3"method='POST' action='index.php' margin='0' style="float: left;width:82.62%;height:5.2%;background-color:white;color:blue;font-size:14pt;font-family: Comic Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
    <spam class="text">
      Los espacios marcados con asterisco (*) son obligatorios
    </spam>

```

```

</form>
<form id="form4" method='POST' action='modciudad2.php' margin='0' style="float:
left;width:82.62%;height:5.4%;background-color:white;color:blue;font-size:14pt;font-family: Comic
Sans MS;text-align:left;padding-right:0.5%;border: 0.5px solid #CCC;">
  <span class="text">
    <center>
      <H3>
        <div>
          <input type="hidden" name="ID_CIUADAD" value="<?php echo
$_GET['ID_CIUADAD']?>">
          <label>Nombre *:</label>
          <input type="text" name="NOMCIUDAD"; value="<?php echo
$con consulta[0] ?>" style="width:500px">
        </div>
      </H3>
    </center>
  </span>
  <span class="button">
    <center>
      <input type="submit" value="Modificar" style="width:100px">
      <input type="submit" value="Volver" style="width: 100px"
formaction="ciudad.php">
    </center>
  </span>
</form>
<form for="form5" action='index.php' method='POST' margin='0' style="float:
left;width:82.62%;height:195.4%;background-color:white;color:blue;font-size:14pt;font-family:
Comic Sans MS;text-align:right;padding-right:0.5%;border: 0.5px solid #CCC;">
  <p></p>
</form>
</body>
</html>

```

9.1.70 archivo que lleva los datos de modificar ciudad a la base de datos

```

<?php
include ("blodese.php");
modificarciudad($_POST['NOMCIUDAD'], $_POST['ID_CIUADAD']);
function modificarciudad($NOMCIUDAD,$ID_CIUADAD)
{
  include ("conexion.php");
  $con = mysqli_query($con , "update ciudadviaja set NOMCIUDAD=" . $NOMCIUDAD . "
where ID_CIUADAD=" . $ID_CIUADAD . "");
}

```

```
}
?>
<script type="text/javascript">
    alert("Ciudad modificada exitosamente");
    window.location.href='ciudad.php';
</script>
```

9.1.71 archivo que borra datos de ciudad de la base de datos

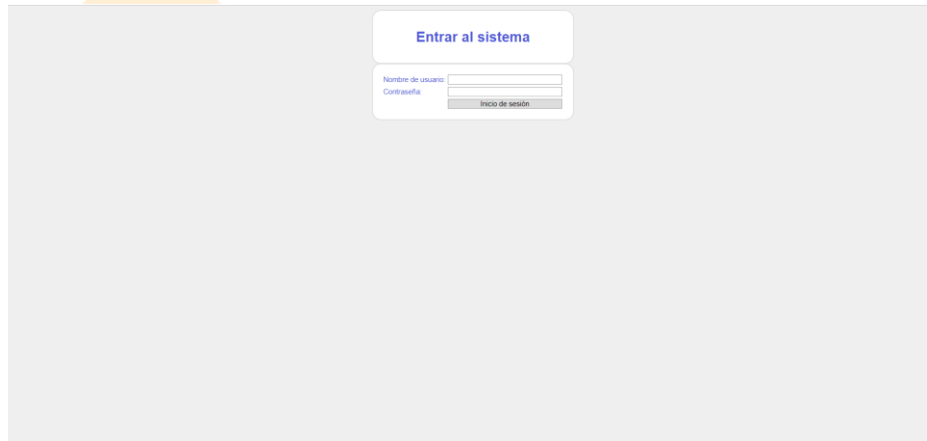
```
<?php
    include ("blodese.php");
    Eliminarciudad($_GET['NOMCIUDAD']);
    function Eliminarciudad($NOMCIUDAD)
    {
        include ("conexion.php");
        $con = mysqli_query($con , "delete from ciudadviaja where ciudadviaja . NOMCIUDAD =
        '$NOMCIUDAD'");
    }
?>
<script type="text/javascript">
    alert("Ciudad eliminada exitosamente");
    window.location.href='ciudad.php';
</script>
```

Capítulo 10: Modelo de Pruebas

10.1. Pruebas de caja negra

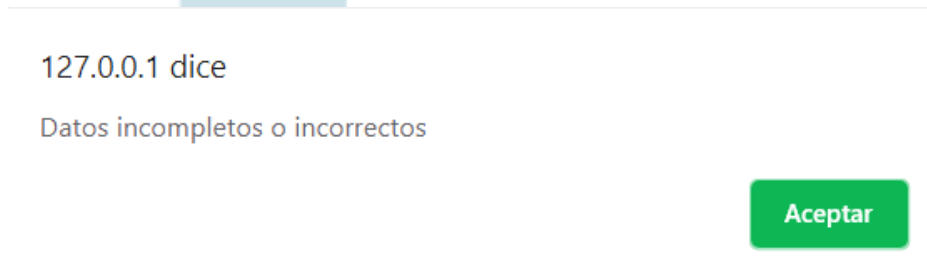
10.1.1 Prueba de validación

Se ingreso al sistema y aparece la ventana de validación o login



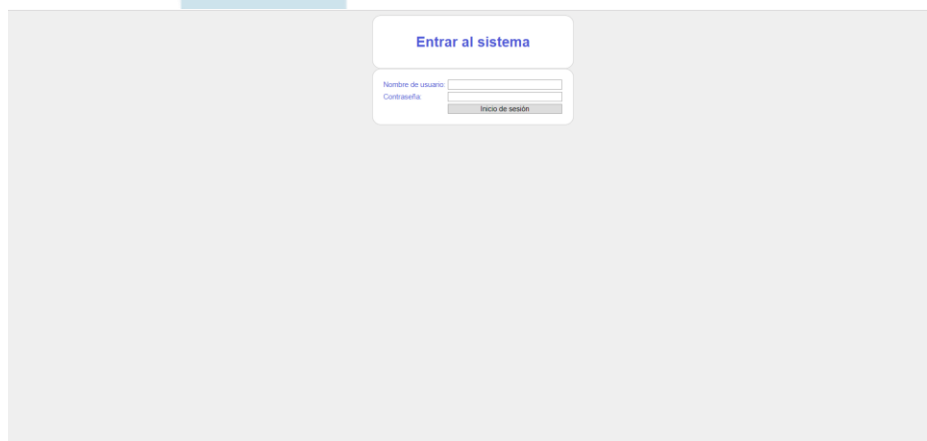
The screenshot shows a login form with the title "Entrar al sistema". It contains two input fields: "Nombre de usuario" and "Contraseña". Below the "Contraseña" field is a "Inicio de sesión" button. The form is set against a light gray background.

se utilizaron el usuario: foest1 y la contraseña: se deja en blanco, y aparece una notificación



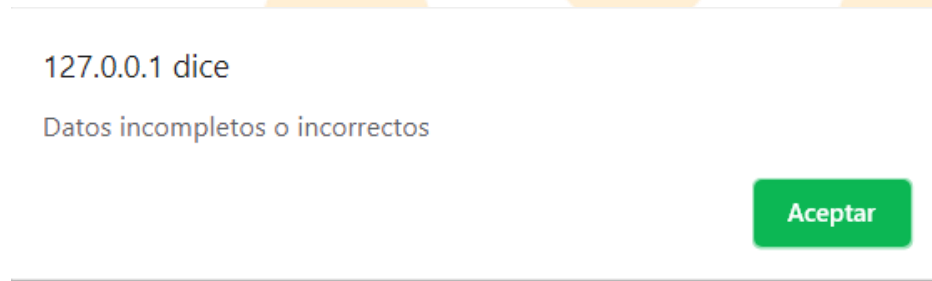
The screenshot shows a notification message with the text "127.0.0.1 dice" and "Datos incompletos o incorrectos". A green "Aceptar" button is located at the bottom right of the notification area.

se da clic en aceptar y nos lleva a la página de validación

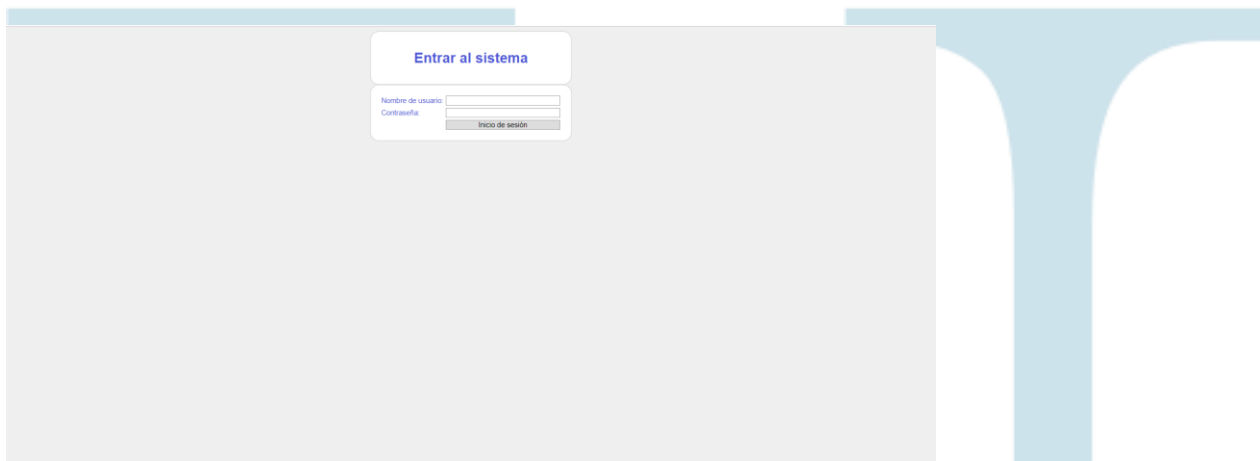


This screenshot is identical to the one above, showing the login form titled "Entrar al sistema" with the "Nombre de usuario" and "Contraseña" fields and the "Inicio de sesión" button.

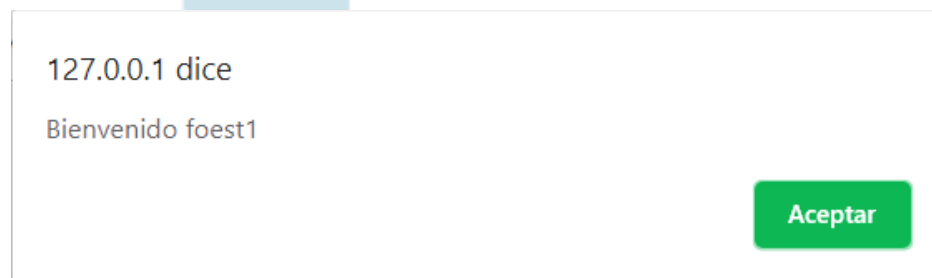
se utilizaron el usuario: foest1 y la contraseña: incorrecta, y aparece la misma notificación



se da clic en aceptar y nos lleva a la página de validación



se utilizaron el usuario: foest1 y la contraseña: correcta, y aparece el mensaje de bienvenida



Se da clic en aceptar y nos lleva a la página principal

The screenshot shows a web application interface titled "Sistema de información". On the left is a sidebar with the following links: [foest1](#), [FUI/D](#), [Beneficio 1](#), [Beneficio 2](#), [Algun otro Beneficio](#), [Antigüedad Sisbén](#), [Tipos de población](#), [Universidades](#), [Correnas](#), and [Ciudad](#). The main content area features a "Nuevo periodo:" section with input fields for "Año:" and "Semestre:" (labeled "semestre"), and a "Nuevo" button. Below this, there is a table with three rows of data: "Estudiantes 2018-2", "Estudiantes 2019-1", and "Estudiantes 2019-2". A "Salir" link is located in the top right corner.

10.1.2 Prueba de creación de funcionario

En la página principal le damos clic sobre el nombre de usuario

This screenshot is identical to the one above, showing the "Sistema de información" page with the same sidebar, navigation links, and main content area containing the "Nuevo periodo:" form and the table of student records.

El cual nos lleva a una pantalla donde veremos los usuarios registrados

The screenshot shows a web application interface titled "Usuarios". At the top, there are two buttons: "Registrar nuevo usuario" and "Modificar mi contraseña". Below these buttons is a table with the following structure:

Usuarios	
Nombre	Eliminar
foest1	Eliminar
foest2	Eliminar
foest3	Eliminar
foest4	Eliminar

The sidebar on the left contains the same navigation links as in the previous screenshots. A "Salir" link is located in the top right corner.

Le damos clic en registrar nuevo usuario y nos lleva a la página de registro de funcionario

foest1 Salir

Los espacios marcados con asterisco (*) son obligatorios

Nombre *

Clave *

Confirmar clave *

- Inicio
- FUI/D
- Beneficio 1
- Beneficio 2
- Algún otro Beneficio
- Antigüedad Sisbén
- Tipos de población
- Universidades
- Carreras
- Ciudad

Llenamos los espacios con nombre, contraseña y confirmación de contraseña, y luego clic en registrar

foest1 Salir

Los espacios marcados con asterisco (*) son obligatorios

Nombre *

Clave *

Confirmar clave *

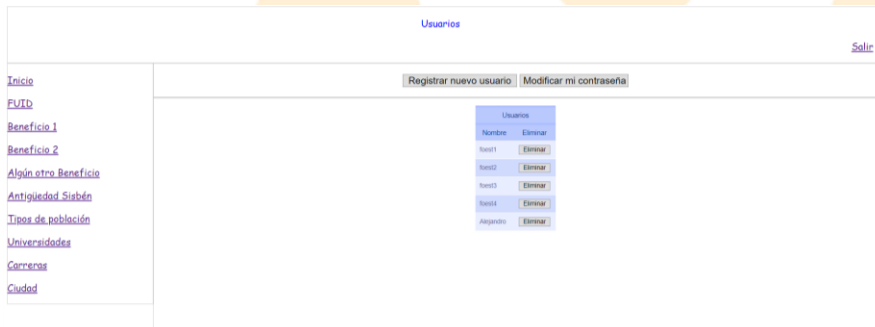
- Inicio
- FUI/D
- Beneficio 1
- Beneficio 2
- Algún otro Beneficio
- Antigüedad Sisbén
- Tipos de población
- Universidades
- Carreras
- Ciudad

Aparece un mensaje que notifica la creación del usuario

127.0.0.1 dice

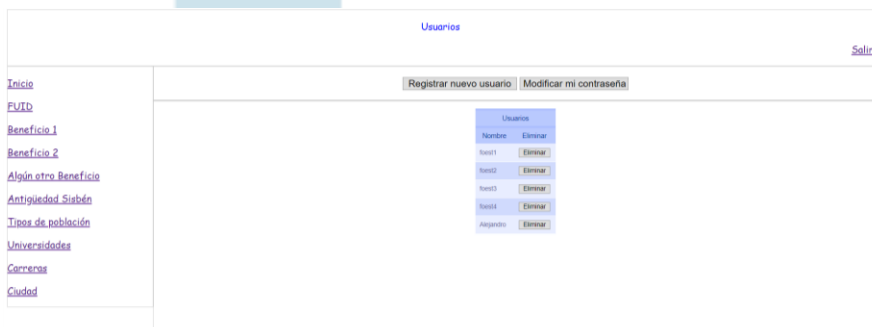
usuario agregado exitosamente

Damos aceptar y en la página de funcionarios nos aparece el nuevo usuario, este usuario nos servirá para ingresar al sistema



10.1.3 Prueba de cambio de contraseña de funcionario

En la página de usuarios registrados damos clic en el botón Modificar mi contraseña



Se abre la página para modificar contraseña y la llenamos con la contraseña actual del usuario, la nueva contraseña, y la confirmación de la nueva contraseña



The screenshot shows a web interface titled "Usuarios" with a "Salir" link in the top right. On the left is a navigation menu with links: Inicio, FUI/D, Beneficio 1, Beneficio 2, Algún otro Beneficio, Antigüedad Siabén, Tipos de población, Universidades, Carreras, and Ciudad. The main content area has a heading "Los espacios marcados con asterisco (*) son obligatorios" and three input fields: "Contraseña *" (with asterisks), "Nueva contraseña *", and "Confirmar nueva contraseña *". Below the fields are "Modificar" and "Volver" buttons.

Error en caso de contraseña actual incorrecta

127.0.0.1 dice

Contraseña no diligenciada o incorrecta

Aceptar

Error en caso de que la contraseña nueva y confirmar contraseña no coincidan

127.0.0.1 dice

Los valores no coinciden

Aceptar

Quando la contraseña es incorrecta o la nueva contraseña no coincide con confirmar contraseña si le damos en aceptar no devuelve a la página de cambio de contraseña

Usuarios [Salir](#)

Los espacios marcados con asterisco (*) son obligatorios

Inicio

FUI/D

Beneficio 1

Beneficio 2

Algún otro Beneficio

Antigüedad Sisbén

Tipos de población

Universidades

Carreras

Ciudad

Contraseña *

Nueva contraseña *

Confirmar nueva contraseña *

Modificar Volver

Quando la contraseña es correcta

127.0.0.1 dice

Usuario actualizado

Aceptar

Si le damos en aceptar nos manda a la página de usuarios registrados

Usuarios [Salir](#)

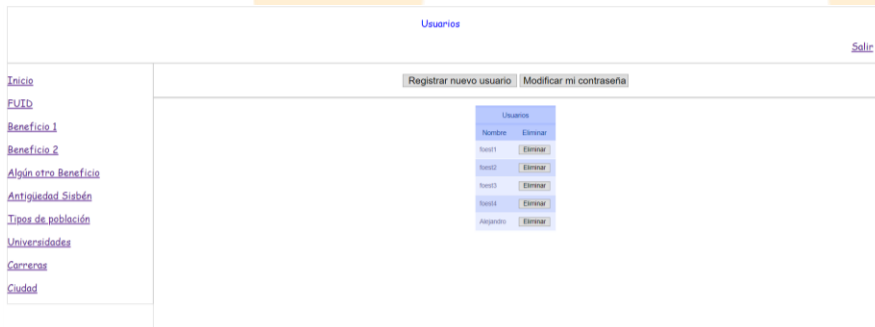
Registrar nuevo usuario Modificar mi contraseña

Usuarios

Nombre	Eliminar
Usuario1	Eliminar
Usuario2	Eliminar
Usuario3	Eliminar
Usuario4	Eliminar
Administrador	Eliminar

10.1.4 Prueba de eliminación de funcionario

En la página de usuarios al frente del usuario que deseamos eliminar hay un botón de eliminar



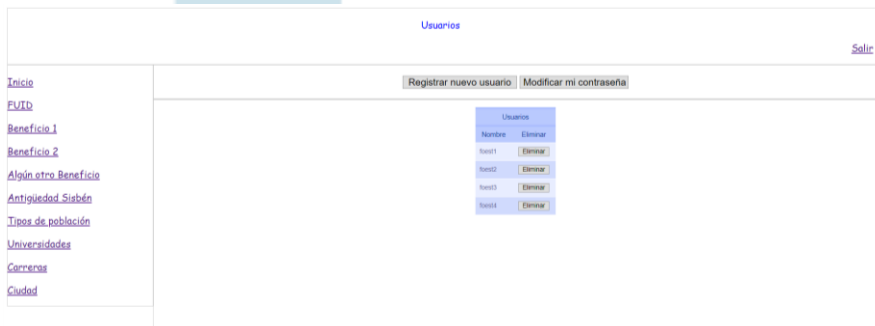
Si pulsamos el botón de eliminar sale el siguiente mensaje

127.0.0.1 dice

Usuario eliminado exitosamente

Aceptar

Si pulsamos en aceptar desaparece el usuario eliminado



10.1.5 Prueba de creación de un nuevo periodo

En la página principal hay una sección que se llama nuevo periodo, allí ingresamos el año y el periodo y le damos clic en nuevo

Sistema de información

foest1 Salir

Nuevo periodo:

Año: semestre:

- [FUTD](#)
- [Beneficio 1](#)
- [Beneficio 2](#)
- [Algún otro Beneficio](#)
- [Antigüedad Sisbén](#)
- [Tipos de población](#)
- [Universidades](#)
- [Comercios](#)
- [Ciudad](#)

[Estudiantes 2018-2](#)

[Estudiantes 2019-1](#)

[Estudiantes 2019-2](#)

Sistema de información

foest1 Salir

Nuevo periodo:

Año: 2020 semestre: 1

- [FUTD](#)
- [Beneficio 1](#)
- [Beneficio 2](#)
- [Algún otro Beneficio](#)
- [Antigüedad Sisbén](#)
- [Tipos de población](#)
- [Universidades](#)
- [Comercios](#)
- [Ciudad](#)

[Estudiantes 2018-2](#)

[Estudiantes 2019-1](#)

[Estudiantes 2019-2](#)

Aparece un mensaje diciendo que se creó el periodo

127.0.0.1 dice

Ha sido creado exitosamente

En la página principal, aparecerá el periodo creado

Sistema de información

foest1 [Salir](#)

Nuevo periodo:

Año: semestre:

[Inicio](#)

[FUTD](#)

[Beneficio 1](#)

[Beneficio 2](#)

[Algún otro Beneficio](#)

[Antigüedad Sisbén](#)

[Tipos de población](#)

[Universidades](#)

[Carmenas](#)

[Ciudad](#)

[Estudiantes 2018-2](#)

[Estudiantes 2019-1](#)

[Estudiantes 2019-2](#)

[Estudiantes 2020-1](#)

10.1.6 Prueba de registro de un nuevo beneficiario del periodo

Se ingresa al periodo donde se desea registrar el usuario

Estudiantes 2019_1

foest1 [Salir](#)

Search:

ID	ITEM	№ CARPETA	FECHA DE RADICACION	BENEFICIO 1	BENEFICIO 2	1 VEZ	RH	RIB	PRIMER APELLIDO	SEGUNDO APELLIDO	PRIMER NOMBRE	SEGUNDO NOMBRE	ID	NUMERO DE DOCUMENTO	EXPEDIDO EN
2		1	2019-03-28						PIÑEDA	MARTINEZ	CARLO	ANDRES		107709543	

[Inicio](#)

[FUTD](#)

[Beneficio 1](#)

[Beneficio 2](#)

[Algún otro Beneficio](#)

[Antigüedad Sisbén](#)

[Tipos de población](#)

[Universidades](#)

[Carmenas](#)

[Ciudad](#)

Se da clic en nuevo registro y aparece una ventana donde se pide el documento del beneficiario a registrar

Estudiantes 2019_1

foest1 [Salir](#)

[Inicio](#)

[FUTD](#)

[Beneficio 1](#)

[Beneficio 2](#)

[Algún otro Beneficio](#)

[Antigüedad Sisbén](#)

[Tipos de población](#)

[Universidades](#)

[Carmenas](#)

[Ciudad](#)

[otras](#)

Número de documento: Ingrese el documento

Si el usuario existe aparece el siguiente mensaje

127.0.0.1 dice

Usuario ya registrado en el periodo 2019-1

[Aceptar](#)

Si damos clic en aceptar este nos devuelve a la página que pide el documento del beneficiario a registrar

foest1 [Salir](#)

[Inicio](#) [atras](#)

[FUID](#) [Número de documento: Ingrese el documento](#) [Registrar](#)

[Beneficio 1](#)

[Beneficio 2](#)

[Algún otro Beneficio](#)

[Antigüedad Sisbén](#)

[Tipos de población](#)

[Universidades](#)

[Comunas](#)

[Ciudad](#)

Si el usuario no existe en el fuid arroja el siguiente mensaje

127.0.0.1 dice

Usuario Nuevo por favor registrar en el FUID

[Aceptar](#)

Si damos clic en aceptar nos lleva a la página de registro, el cual contara con el espacio de documento lleno, este formulario sacara el monto aprobado de manera automática, para ello pedirá el valor del semestre y el porcentaje a aplicar, también de manera automática sacara el valor aprobado para el subsidio de transporte, en este se pedirá la cantidad de días y el valor del recorrido, estos llenaran de manera automática la cuenta por semana y por mes, después pedirá los meses, los cuales se ingresaran y se multiplicaran por el valor de mes que se halló.

Inicio Salir

Ingrese los datos:

Los campos marcados con asterisco (*) son obligatorios

Datos personales:

ID:	<input type="text" value="Digite la Id."/>	Item *:	<input type="text" value="Digite el item"/>
N° carpeta *:	<input type="text" value="Digite el número de la carpeta"/>	Fecha de radicación *:	<input type="text" value="dd/mm/aaaa"/>
Beneficio 1:	<input type="text" value="Seleccione:"/>	Beneficio 2:	<input type="text" value="Seleccione:"/>
1° Vez:	<input type="text"/>	Renovación:	<input type="text"/>
Reintegro:	<input type="text"/>		
Primer apellido *:	<input type="text" value="Ingrese primer apellido"/>	Segundo apellido *:	<input type="text" value="Ingrese primer apellido"/>
Primer nombre *:	<input type="text" value="Ingrese primer nombre"/>	Segundo nombre:	<input type="text" value="Ingrese segundo nombre"/>
Tipo de documento *:	<input type="text" value="Seleccione:"/>	Número de documento *:	<input type="text" value="1077092046"/>
Expedido en:	<input type="text" value="Ingrese el lugar de expedición"/>	Fecha de nacimiento:	<input type="text" value="dd/mm/aaaa"/>

Si el usuario esta registrado en el fuid aparecerá el siguiente mensaje

127.0.0.1 dice

Usuario ya registrado en el FUID

Aceptar

Si damos clic en aceptar, tomara los datos registrados del fuid y los mostrara en la pantalla de registro, este formulario sacara el monto aprobado de manera automática, para ello pedirá el valor del semestre y el porcentaje a aplicar, también de manera automática sacara el valor aprobado para el subsidio de transporte, en este se pedirá la cantidad de días y el valor del recorrido, estos llenaran de manera automática la cuenta por semana y por mes, después pedirá los meses, los cuales se ingresaran y se multiplicaran por el valor de mes que se halló.

Inicio Salir

Ingrese los datos:

Los campos marcados con asterisco (*) son obligatorios

Datos personales:

ID:	<input type="text" value="Digite la Id."/>	Item *:	<input type="text" value="Digite el item"/>
N° carpeta *:	<input type="text" value="7"/>	Fecha de radicación *:	<input type="text" value="dd/mm/aaaa"/>
Beneficio 1:	<input type="text" value="Seleccione:"/>	Beneficio 2:	<input type="text" value="Seleccione:"/>
1° Vez:	<input type="text"/>	Renovación:	<input type="text"/>
Reintegro:	<input type="text"/>		
Primer apellido *:	<input type="text" value="OJEDA"/>	Segundo apellido *:	<input type="text" value="ACEVEDO"/>
Primer nombre *:	<input type="text" value="MARLON"/>	Segundo nombre:	<input type="text" value="ANDRES"/>
Tipo de documento *:	<input type="text" value="Seleccione:"/>	Número de documento *:	<input type="text" value="1077088311"/>
Expedido en:	<input type="text" value="Ingrese el lugar de expedición"/>	Fecha de nacimiento:	<input type="text" value="dd/mm/aaaa"/>

En la parte inferior del documento encontramos el botón registrar el cual nos mandara a la página del periodo y allí se puede ver el usuario registrado

ID	ITEM	No. CARPETA	FECHA DE RADICACION	BENEFICIO 1	BENEFICIO 2	1 VEZ	RN	REI	PRIMER APELLIDO	SEGUNDO APELLIDO	PRIMER NOMBRE	SEGUNDO NOMBRE	ID	NUMERO DE DOCUMENTO	EXPEDIDO EN
2	1		2019-03-28						PIÑEDA	MARTINEZ	CARLO	ANDRES		1077090643	
01	1	3000	2019-03-28	LINEA 4	TRANSPORTE	X	X	X	ROZO	VEGA	RAFAEL	ALEJANDRO	C.C.	107334851	GACHANCIP

10.1.7 Prueba de una exportación de un periodo

Para la exportación de un periodo en la pantalla del mismo aparece un botón llamado Excel

ID	ITEM	No. CARPETA	FECHA DE RADICACION	BENEFICIO 1	BENEFICIO 2	1 VEZ	RN	REI	PRIMER APELLIDO	SEGUNDO APELLIDO	PRIMER NOMBRE	SEGUNDO NOMBRE	ID	NUMERO DE DOCUMENTO	EXPEDIDO EN
2	1		2019-03-28						PIÑEDA	MARTINEZ	CARLO	ANDRES		1077090643	
01	1	3000	2019-03-28	LINEA 4	TRANSPORTE	X	X	X	ROZO	VEGA	RAFAEL	ALEJANDRO	C.C.	107334851	GACHANCIP

Si le damos clic este descargar un archivo de Excel con los datos de la tabla

ID	ITEM	No. CARPETA	FECHA DE RADICACION	BENEFICIO 1	BENEFICIO 2	1 VEZ	RN	REI	PRIMER APELLIDO	SEGUNDO APELLIDO	PRIMER NOMBRE	SEGUNDO NOMBRE	ID	NUMERO DE DOCUMENTO	EXPEDIDO EN
2	1		2019-03-28						PIÑEDA	MARTINEZ	CARLO	ANDRES		1077090643	
01	1	3000	2019-03-28	LINEA 4	TRANSPORTE	X	X	X	ROZO	VEGA	RAFAEL	ALEJANDRO	C.C.	107334851	GACHANCIP

Ejecutamos el Excel para verificar los datos descargados

Base de datos 2019-1

ID	ITEM	No. CARPETA	FECHA DE RADICACION	BENEFICIO 1	BENEFICIO 2	1 VEZ	RN	REI	PRIMER APELLIDO	SEGUNDO APELLIDO	PRIMER NOMBRE	SEGUNDO NOMBRE	ID	NUMERO DE DOCUMENTO
2	1	3000	2019-03-28	LINEA 4	TRANSPORTE	X	X	X	ROZO	VEGA	RAFAEL	ALEJANDRO	C.C.	107353

10.1.8 Prueba de modificación de un beneficiario

En el periodo seleccionamos la opción actualizar registro

Estudiantes 2019_1

foest1 [Salir](#)

[Inicio](#) [FUI/D](#) [Beneficio 1](#) [Beneficio 2](#) [Algun otro Beneficio](#) [Antigüedad Sisben](#) [Tipos de población](#) [Universidades](#) [Comercios](#) [Ciudad](#)

[Nuevo registro](#) [Actualizar registro](#) [Estadísticas](#)

Excel Search: _____

ID	ITEM	No. CARPETA	FECHA DE RADICACION	BENEFICIO 1	BENEFICIO 2	1 VEZ	RN	REI	PRIMER APELLIDO	SEGUNDO APELLIDO	PRIMER NOMBRE	SEGUNDO NOMBRE	ID	NUMERO DE DOCUMENTO	EXPEDICION EN
2	1	3000	2019-03-28	LINEA 4	TRANSPORTE	X	X	X	ROZO	VEGA	RAFAEL	ALEJANDRO	C.C.	107353	GACHANCOP

Nos llevará a una pantalla donde pedirá el documento del beneficiario

The screenshot shows a web application interface with a title bar 'Estudiantes 2019_1' and a 'Salir' button. On the left is a navigation menu with links: Inicio, FUIID, Beneficio 1, Beneficio 2, Algún otro Beneficio, Antigüedad Siabén, Tipos de población, Universidades, Carreras, and Ciudad. The main content area has a sub-header 'Inicio' and a form with the label 'Número de documento: Ingrese el documento' and an 'Actualizar' button.

Si el usuario no existe en el período arrojará el siguiente mensaje

The screenshot shows a dialog box with the text '127.0.0.1 dice' and 'Usuario No existente'. A green 'Aceptar' button is located at the bottom right of the dialog.

Si damos clic en aceptar, nos llevará a una pantalla donde pedirá el documento del beneficiario

This screenshot is identical to the one above, showing the 'Estudiantes 2019_1' interface with the document number input field and the 'Actualizar' button.

Si el usuario existe nos llevara a una ventana donde se vera la información del beneficiario

Datos personales:

ID:	01	Item:	1
N° carpeta *:	3000	Fecha de radicación *:	28/03/2019
Beneficio 1:	LINEA 4	Beneficio 2:	TRANSPORTE
1° Vez:	X	Renovación:	X
Reintegro:	X		
Primer apellido *:	ROZO	Segundo apellido *:	VEGA
Primer nombre *:	RAFAEL	Segundo nombre *:	ALEJANDRO
Tipo de documento *:	C.C.	Numero de documento *:	1073534051
Expedido en:	GACHANCIPA	Fecha de nacimiento:	29/01/1995
Edad *:	25	Estado civil:	SOLTERO(A)
Nombre de EPS:	SURA	Estado EPS:	BENEFICIARIO(A)
Población especial:	NO	Genero *:	M
Telefono 1 *:	3118169125	Telefono 2:	3008826202

Se cambian los datos necesarios y en la parte inferior damos clic en modificar

127.0.0.1 dice

Registro del estudiante modificado exitosamente

Aceptar

Damos clic en aceptar y nos lleva a la página del periodo en la cual se ven los cambios realizados

Estudiantes 2019_1

ID *	ITEM	Nº CARPETA	FECHA DE RADICACIÓN	BENEFICIO 1	BENEFICIO 2	1° VEZ	RN	REI	PRIMER APELLIDO	SEGUNDO APELLIDO	PRIMER NOMBRE	SEGUNDO NOMBRE	ID	NUMERO DE DOCUMENTO	EXPEDIDO EN
2	1		2019-03-28						FRIEDA	MARTINEZ	CARLO	ANDRES		107709043	
01	1	3000	2019-03-28	LINEA 4	TRANSPORTE	X			ROZO	VEGA	RAFAEL	ALEJANDRO	C.C.	1073534051	GACHANCIPA

10.1.9 Prueba de estadísticas

En la página del periodo damos clic en el botón estadísticas



The screenshot shows the 'Estudiantes 2019_1' application interface. At the top, there are buttons for 'Nuevo registro', 'Actualizar registro', and 'Estadísticas'. Below these is a search bar and an 'Excel' button. A table displays student records with columns for ID, ITEM, No. CARPETA, FECHA DE RADICACION, BENEFICIO 1, BENEFICIO 2, 1 VÉZ, RN, REI, PRIMER APELLIDO, SEGUNDO APELLIDO, PRIMER NOMBRE, SEGUNDO NOMBRE, ID, NUMERO DE DOCUMENTO, and EXPEDICION EN. The table contains two rows of data.

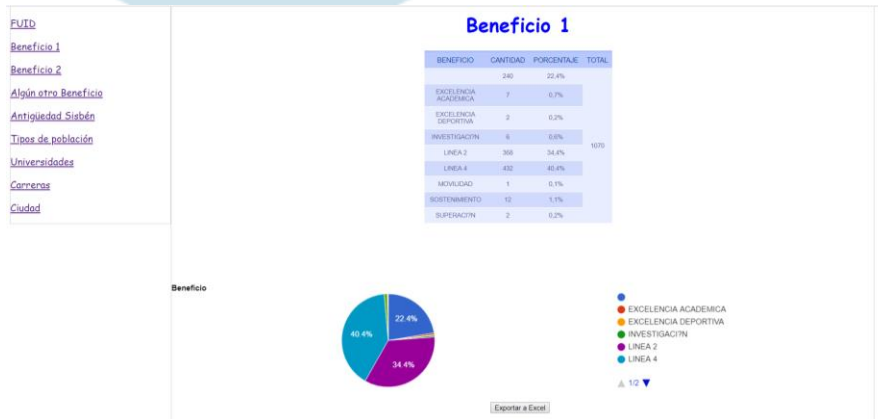
ID	ITEM	No. CARPETA	FECHA DE RADICACION	BENEFICIO 1	BENEFICIO 2	1 VÉZ	RN	REI	PRIMER APELLIDO	SEGUNDO APELLIDO	PRIMER NOMBRE	SEGUNDO NOMBRE	ID	NUMERO DE DOCUMENTO	EXPEDICION EN
2	1		2019-03-28						PIREDA	MARTINEZ	CARLO	ANDRES		107706943	
01	1	3000	2019-03-28	LINEA 4	TRANSPORTE	X			RODO	VEGA	RAFAEL	ALJANDRO	C.C.	107304001	GACHANCOP

Aparece una opción de seleccionar en la cual podemos seleccionar Beneficio 1, Beneficio 2, Documento, Edad, Estado civil, Poblacion especial, Genero, Antigüedad Sisbén, Trabaja actualmente, Tipo de vivienda, Estrato, Beneficios diferentes a FOEST, Universidad, Nivel, Carrera, Modalidad, Semestre, Dependencia horas sociales 1, Dependencia horas sociales 2, Dependencia horas sociales 3, Ciudad a la que viajan; después de seleccionado se presiona el botón filtrar.



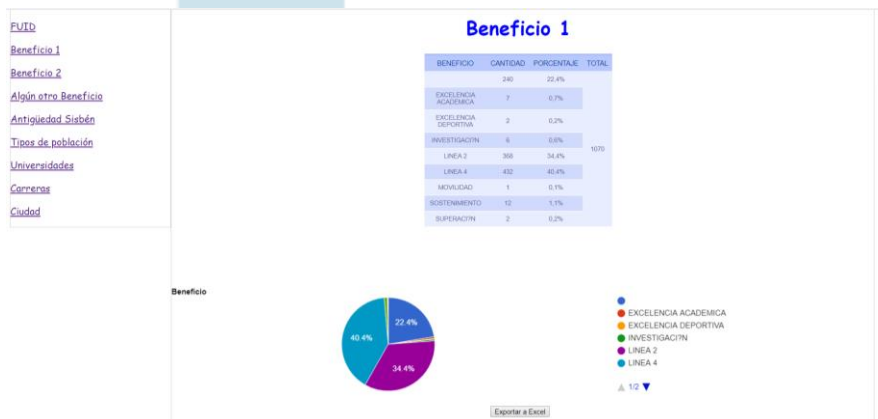
The screenshot shows the 'Sistema de información' application interface. At the top, there are buttons for 'Inicio', 'FUI/D', 'Beneficio 1', 'Beneficio 2', 'Algun otro Beneficio', 'Antigüedad Sisbén', 'Tipos de población', 'Universidades', 'Comunas', and 'Ciudad'. Below these is a dropdown menu labeled 'Selecciono' with 'Selecciona' selected, and a 'Filtrar' button.

Después nos mostrara la opción que seleccionamos, la cual mostrara un cuadro y un grafico



10.1.10 Prueba de exportación de estadísticas

En la opción seleccionada de estadísticas en la parte inferior hay un botón llamado exportar a Excel



Una vez presionado el botón descargara un archivo en Excel con información de la tabla, el grafico se exporta dando clic sobre él y presionando la opción de guardar imagen o copiar imagen

The screenshot shows a web application with a sidebar on the left containing navigation links: [Beneficio 1](#), [Beneficio 2](#), [Algún otro Beneficio](#), [Antigüedad Sisbén](#), [Tipos de población](#), [Universidades](#), [Comunas](#), and [Ciudad](#). The main content area displays a table with the following data:

BENEFICIO	CANTIDAD	PORCENTAJE	TOTAL
	240	22.4%	
EXCELENCIA ACADÉMICA	7	0.7%	
EXCELENCIA DEPORTIVA	2	0.2%	
INVESTIGACIÓN	6	0.6%	
LINEA 2	368	34.4%	1070
LINEA 4	432	40.4%	
MOVILIDAD	1	0.1%	
SOSTENIMIENTO	12	1.1%	
SUPERACTIV	2	0.2%	

Below the table is a pie chart representing the data. A context menu is open over the pie chart with the following options: 'Abrir imagen en una pestaña nueva', 'Guardar imagen como...', 'Copiar imagen', 'Copiar dirección de la imagen', 'Bloquear elemento', 'Inspeccionar', and 'Exportar a Excel'. The 'Exportar a Excel' option is highlighted.

Se abre el archivo de Excel

The screenshot shows an Excel spreadsheet with the following data:

BENEFICIO	CANTIDAD	PORCENTAJE	TOTAL
	240	22.4%	
EXCELENCIA ACADÉMICA	7	0.7%	
EXCELENCIA DEPORTIVA	2	0.2%	
INVESTIGACIÓN	6	0.6%	
LINEA 2	368	34.4%	1070
LINEA 4	432	40.4%	
MOVILIDAD	1	0.1%	
SOSTENIMIENTO	12	1.1%	
SUPERACTIV	2	0.2%	

10.1.11 Prueba de creación de beneficiario en fuid

En la página principal, o en las en las distintas paginas o pantallas aparece una opción llamada FUID

Sistema de información

foest1 [Salir](#)

Nuevo periodo:

Año: semestre:

[Inicio](#)
[FUID](#)
[Beneficio 1](#)
[Beneficio 2](#)
[Algún otro Beneficio](#)
[Antigüedad Sisbén](#)
[Tipos de población](#)
[Universidades](#)
[Comunas](#)
[Ciudad](#)

[Estudiantes 2018-2](#)
[Estudiantes 2019-1](#)
[Estudiantes 2019-2](#)
[Estudiantes 2020-1](#)

Le damos clic y nos dirigirá a una pantalla donde se verán los beneficiarios registrados

Beneficiarios registrados

foest1 [Salir](#)

Item #	Numero de Documento	Primer Apellido	Segundo Apellido	Primer Nombre	Segundo Nombre	Numero de Carpeta	Modifs	
1	107709983	PINEDA	MARTINEZ	CARLO	ANDRES	1	Modifs	
2	35421981	BALLEA	REYES	MARIA	MERCEDES	2	Modifs	
3	4269782	GARCIA	MATEUS	JOSE	ALFONSO	3	Modifs	
4	107709936	ESPITA	PRADA	JULIE	CATHERINE	4	Modifs	
5	107709880	CADIZ	STAN	TORRES	LADY	KATHERINE	5	Modifs
6	107709854	LAMIES	CUEVAS	INGRID	KATHERINE	6	Modifs	
7	107709831	CUEVA	ACEVEDO	MARILYN	ANDRES	7	Modifs	

Showing 1 to 10 of 2,277 entries

Previous 1 2 3 4 5 ... 228 Next

Seleccionamos la opción de ingrese nuevo beneficiario la cual nos llevara a un formulario el cual se llena con datos del beneficiario.

Ingreso datos del beneficiario

foest1 [Salir](#)

Los espacios marcados con asterisco (*) son obligatorios

Item y numero de carpeta* [Digite el numero de item tambien]

Numero documento* [Digite el numero de documento]

Primer apellido* [Digite el primer apellido]

Segundo apellido* [Digite el segundo apellido]

Primer nombre* [Digite el primer nombre]

Segundo apellido* [Digite el segundo nombre]

Después de lleno damos clic en modificar y nos aparece el siguiente mensaje

127.0.0.1 dice

Beneficiario agregado exitosamente

Aceptar

Le damos clic en aceptar, este nos llevara a la pantalla fuid, en la cual se puede revisar el número de registros debajo de la tabla o buscando el número de documento en el cuadro.

Beneficiarios registrados

[Salir](#)

Ingresar nuevo beneficiarioSearch:

Excel

Beneficiarios						
Item #	Numero de Documento	Primer Apellido	Segundo Apellido	Primer Nombre	Segundo Nombre	Numero de Caratula
2271	107709173	MEYES	ORTIZ	CARLOS	ALBERTO	2271
2272	107709469	MENDOZA	VASQUEZ	DEGO	ARMANDO	2272
2273	100300367	CABRERA	MORENO	MARIA	CARLA	2273
2274	100300338	SALAZAR	HERNANDEZ	SEBASTIAN		2274
2275	100300743	BERNAL	HURTADO	BRYAN		2275
2276	100300338	MARTINEZ	ROCHA	JUAN	DEGO	2276
2277	100300743	BOLIVAR	VALERO	JANEY	DIYANA	2277
2278	2002275	VEGA	RODRIGUEZ	ELDA	RODRIGUEZ	2278

Showing 2,271 to 2,278 of 2,278 entries

Previous 1 ... 224 225 226 227 228 Next

10.1.12 Prueba de exportación de fuid

Para exportar los datos del FUID damos clic en el botón llamado Excel

Beneficiarios registrados

[Salir](#)

Ingresar nuevo beneficiarioSearch:

Excel

Beneficiarios						
Item #	Numero de Documento	Primer Apellido	Segundo Apellido	Primer Nombre	Segundo Nombre	Numero de Caratula
2271	107709173	MEYES	ORTIZ	CARLOS	ALBERTO	2271
2272	107709469	MENDOZA	VASQUEZ	DEGO	ARMANDO	2272
2273	100300367	CABRERA	MORENO	MARIA	CARLA	2273
2274	100300338	SALAZAR	HERNANDEZ	SEBASTIAN		2274
2275	100300743	BERNAL	HURTADO	BRYAN		2275
2276	100300338	MARTINEZ	ROCHA	JUAN	DEGO	2276
2277	100300743	BOLIVAR	VALERO	JANEY	DIYANA	2277
2278	2002275	VEGA	RODRIGUEZ	ELDA	RODRIGUEZ	2278

Showing 2,271 to 2,278 of 2,278 entries

Previous 1 ... 224 225 226 227 228 Next

El cual descargara un archivo

foest1 Salir

Inicio

FUID

Beneficio 1

Beneficio 2

Algún otro Beneficio

Antigüedad Sisben

Tipos de población

Universidades

Comunas

Ciudad

Ingresar nuevo beneficiario

Excel Search:

Beneficiarios						
Item #	Numero de Documento	Primer Apellido	Segundo Apellido	Primer Nombre	Segundo Nombre	Numero de Carpeta
2271	107709173	NEYES	ORTIZ	CARLOS	ALBERTO	2271
2272	107709409	MENCKZA	VASQUEZ	DIEGO	ARIBANCO	2272
2273	100300387	CABRERA	MORENO	MARIA	CARLA	2273
2274	100300308	SALAZAR	HERNANDEZ	SEBASTIAN		2274
2275	100300193	BERNAL	HURTADO	BRAYAN		2275
2276	100300308	MARTINEZ	ROCHA	JUAN	DIEGO	2276
2277	100300143	BOLIVAR	VALEDO	JANEY	DAIANA	2277
2278	20022726	VEGA	RODRIGUEZ	ELDA	RODRIGUEZ	2278

Showing 2,271 to 2,278 of 2,278 entries

Previous 1 ... 224 225 226 227 228 Next

Base de datos FUID.xlsx Mostrar todo

Se abre el archivo

Autoguardado Base de datos FUID.xlsx - Vista protegida - Excel

Archivo Inicio Insertar Disposición de página Fórmulas Datos Revisar Vista Programador Complementos Ayuda Prueba de carga Nitro Pro Equipo ¿Qué desea hacer? Compartir Comentarios

VISTA PROTEGIDA. Tenga cuidado: los archivos de Internet pueden contener virus. Si no tiene que editarlo, es mejor que siga en Vista protegida. Habilitar edición

Base de datos FUID

Item	Numero de Documento	Primer Apellido	Segundo Apellido	Primer Nombre	Segundo Nombre	Numero de Carpeta	Modificar
1	107709041	PINEDA	MARTINEZ	CAMILO	ANDRES	1	
2	20421661	BALLEEN	REYES	MARIA	MERCEDES	2	
3	4265782	GAMBOA	MATEUS	JOSE	ALFONSO	3	
4	107709036	ESPITIA	PRADA	JULIE	CATHERINE	4	
5	107708880	CASTAÑEDA	TORRES	LADY	KATHERINE	5	
6	107708854	LINARES	CUEVAS	INGRID	KATHERINE	6	
7	107708831	OHEDA	ACEVEDO	MARILYN	ANDRES	7	
8	200200204	HERSLE	GUTIERREZ	KEVIN		8	
9	107709241	NAVAS	PEÑA	KAREN	LICETH	9	
10	107708985	TIEMPOS	MORENO	NESTOR	DARNEY	10	
11	107708952	MAJAVER	BERNAL	HENRY		11	
12	107709564	RACHEN	GONZALEZ	JUAN	CAMILO	12	
13	1020803676	JOJICA	RAMIREZ	CINDY	ESTEFANIA	13	
14	107708983	CUBIDES	SANCHEZ	JHONATAN	ALEXANDER	14	
15	107708931	RIEY	CABREDO	MARIA	ALEJANDRA	15	
16	9343902	MARTINEZ	CARRERA	JHON	EDINSON	16	
17	107317628	RODRIGUEZ	GUTIERREZ	DIEGO	ALEJANDRO	17	
18	107708027	GARZON	SALAMANCA	JENNY	PAGLA	18	
19	107708939	RIATO	CASTELBLANCO	LAURA	MILENA	19	
20	107708874	CAMPOS	NIETO	MIGUEL	ANGEL	20	
21	107332836	GARZON	HERERA	JORGE	MANUEL	21	
22	1077088175	GARZON	SALAMANCA	DIEGO	FERNANDO	22	
23	1077091404	TINACA	MONTAÑO	JEISSON	DUVAN	23	
24	1101116531	RUJE	VERA	LIDY	MARIELLA	24	
25	1077090366	VELANDIA	MORA	ERIKA	LORENA	25	
26	1075877931	GARCIA	BELTRAN	STEFANY	ALEJANDRA	26	
27	1077090994	CETINA	GIL	ADRIANA	MARIA	27	
28	21022162	GONZALES	MALDONADO	LIZ	ANANDA	28	
29	1020817754	ALBORNOZ	MAMBUSCAY	MAHARAYIT	JHARDANNY	29	

Sheet1

10.1.13 Prueba de modificación de beneficiario fuid

En la pantalla FUID al frente del beneficiario que se quiera modificar hay un botón el cual se llama modificar

foest1 Salir

Inicio Ingresar nuevo beneficiario

FUID

Beneficio 1

Beneficio 2

Algún otro Beneficio

Antigüedad Sisbén

Tipos de población

Universidades

Comunas

Ciudad

Beneficiarios

Item #	Numero de Documento	Primer Apellido	Segundo Apellido	Primer Nombre	Segundo Nombre	Numero de Carpeta	Modificar
2271	107709173	REYES	ORTIZ	CARLOS	ALBERTO	2271	Modificar
2272	107709469	MENEZGA	VARGUEZ	DIEGO	ARMANDO	2272	Modificar
2273	100006807	CABRERA	MORENO	MARIA	CAMILA	2273	Modificar
2274	100006808	SALAZAR	HERNANDEZ	SEBASTIAN		2274	Modificar
2275	100006743	BERNAL	HURTADO	OSIVAN		2275	Modificar
2276	100006808	MARTINEZ	BOCHA	JUAN	DIEGO	2276	Modificar
2277	100006743	BOLIVAR	VALERO	JANEY	DIYANA	2277	Modificar
2278	20922735	VEGA	RODRIGUEZ	ELDA	RODRIGUEZ	2278	Modificar

Showing 2,271 to 2,278 of 2,278 entries

Previous 1 ... 224 225 226 227 **228** Next

Esta opción nos lleva a una pantalla donde se vera la información del beneficiario a modificar

foest1 Salir

Inicio

FUID

Beneficio 1

Beneficio 2

Algún otro Beneficio

Antigüedad Sisbén

Tipos de población

Universidades

Comunas

Ciudad

Modifique beneficio

Los espacios marcados con asterisco (*) son obligatorios

Item y numero de carpeta: 2278

Numero documento*: 20922735

Primer apellido*: VEGA

Segundo apellido*: RODRIGUEZ

Primer nombre*: ELDA

Segundo apellido*: RODRIGUEZ

Se cambian los datos necesarios y se da clic en modificar y aparece el siguiente mensaje

127.0.0.1 dice

Beneficiario modificado exitosamente

Damos clic en aceptar y nos lleva a la pantalla FUID donde se verá el cambio realizado

The screenshot shows a web interface titled "Beneficiarios registrados". On the left is a navigation menu with links: Inicio, FUID, Beneficio 1, Beneficio 2, Algún otro Beneficio, Antigüedad Sisbén, Tipos de población, Universidades, Carreras, and Ciudad. The main content area has a "Salir" link at the top right and a "Ingresar nuevo beneficiario" button. Below this is a search bar and a table of beneficiaries.

Item #	Numero de Documento	Primer Apellido	Segundo Apellido	Primer Nombre	Segundo Nombre	Numero de Carreta	Modificar
2271	107709173	REYES	ORTIZ	CARLOS	ALBERTO	2271	Modificar
2272	107709469	MENEZES	VAREZ	DIEGO	ARMANDO	2272	Modificar
2273	100300387	CABRERA	MORENO	MARIA	CARLA	2273	Modificar
2274	100300309	SALAZAR	HERNANDEZ	SEBASTIAN		2274	Modificar
2275	100300743	BERNAL	HURTADO	BRYAN		2275	Modificar
2276	100300308	MARTINEZ	ROCHA	JUAN	DIEGO	2276	Modificar
2277	100300743	BOLIVAR	VALEDO	JANEY	DIANA	2277	Modificar
2278	20022776	VEGA	RODRIGUEZ	ELDA	RODRIGUEZ	2278	Modificar

Showing 2,271 to 2,278 of 2,278 entries
Previous 1 ... 224 225 226 227 228 Next

10.1.14 Prueba de creación de beneficio 1

En la página principal, o en las en las distintas paginas o pantallas aparece una opción llamada Beneficio 1

The screenshot shows a web interface titled "Sistema de información". On the left is a navigation menu with links: Inicio, FUID, Beneficio 1, Beneficio 2, Algún otro Beneficio, Antigüedad Sisbén, Tipos de población, Universidades, Carreras, and Ciudad. The main content area has a "Salir" link at the top right and a "Nuevo periodo:" section with a form for "Año" and "Semestre". Below this is a "Nuevo" button and a list of benefit periods: "Estudiantes 2018-2", "Estudiantes 2019-1", "Estudiantes 2019-2", and "Estudiantes 2020-1".

Le damos clic y nos dirigirá a una pantalla donde se verán los beneficiarios registrados

The screenshot shows a web interface titled "Beneficiarios registrados". On the left is a navigation menu with links: Inicio, FUID, Beneficio 1, Beneficio 2, Algún otro Beneficio, Antigüedad Sisbén, Tipos de población, Universidades, Carreras, and Ciudad. The main content area has a "Salir" link at the top right and a "Ingresar nuevo beneficiario" button. Below this is a search bar and a list of beneficiaries for "Beneficio 1".

Nombre	Modificar	Eliminar
BEDA GESTIONARIA	Modificar	Eliminar
EXCELENCIA ACADÉMICA	Modificar	Eliminar
EXCELENCIA CULTURAL	Modificar	Eliminar
EXCELENCIA DEPORTIVA	Modificar	Eliminar
INVESTIGACION	Modificar	Eliminar
LINEA 3	Modificar	Eliminar
LINEA 4	Modificar	Eliminar
MELIOR OPES	Modificar	Eliminar
MOVILIDAD	Modificar	Eliminar
NINGUNO	Modificar	Eliminar

Showing 1 to 10 of 12 entries
Previous 1 2 Next

Seleccionamos la opción de ingreso nuevo beneficio la cual nos llevara a un formulario el cual se llena con el nombre del beneficio

foest1 Salir

Ingreso el beneficio

Inicio

FUI/D

Beneficio 1

Beneficio 2

Algún otro Beneficio

Antigüedad Sisbén

Tipos de población

Universidades

Comenzos

Ciudad

Los espacios marcados con asterisco (*) son obligatorios

Nombre Beneficio *

Registrar Volver

Después de lleno damos clic en registrar y nos aparece el siguiente mensaje

127.0.0.1 dice

Beneficio agregado exitosamente

Aceptar

Le damos clic en aceptar, este nos llevara a la pantalla Beneficios 1, en la cual se puede revisar el número de registros debajo de la tabla o buscando el nombre en el cuadro.

foest1 Salir

Beneficios registrados

Ingresar nuevo beneficio

Search:

Nombre	Modificar	Eliminar
INGRESO	Modificar	Eliminar
SOSTENIMIENTO	Modificar	Eliminar
SUPERACION	Modificar	Eliminar

Showing 11 to 13 of 13 entries

Previous 1 2 Next

10.1.15 Prueba de modificación beneficio 1

En la pantalla beneficio 1 al frente del beneficio que se quiera modificar hay un botón el cual se llama modificar

The screenshot shows a web interface titled "Beneficios registrados". On the left is a navigation menu with links: Inicio, FUI/D, Beneficio 1, Beneficio 2, Algún otro Beneficio, Antigüedad Sisbén, Tipos de población, Universidades, Carreras, and Ciudad. The main content area has a "Salir" link in the top right and an "Ingresar nuevo beneficio" button. Below that is a search bar and a table of benefits. The table has columns for "Nombre", "Modificar", and "Eliminar". Two rows are visible: "NINGUNO" and "SOSTENIMIENTO", both with "Modificar" and "Eliminar" buttons. Below the table, it says "Showing 11 to 13 of 13 entries" and has "Previous", "1", "2", and "Next" navigation links.

Esta opción nos lleva a una pantalla donde se vera la información del beneficio a modificar

The screenshot shows a web interface titled "Modifique beneficio". On the left is the same navigation menu as in the previous screenshot. The main content area has a "Salir" link in the top right. Below it is a note: "Los espacios marcados con asterisco (*) son obligatorios". There is a text input field labeled "Nombre Beneficio *" with the value "mejor estudiante". Below the input field are two buttons: "Modificar" and "Volver".

Se cambian los datos necesarios y se da clic en modificar y aparece el siguiente mensaje

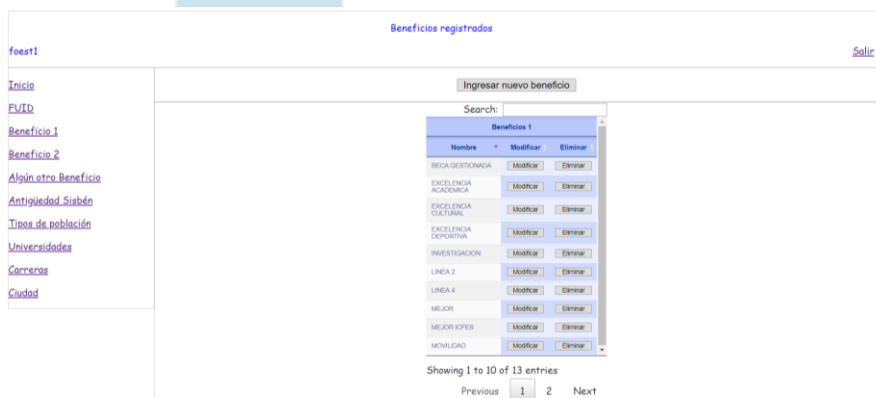
The screenshot shows a dialog box with a white background and a grey border. It contains the text "127.0.0.1 dice" and "Beneficio modificado exitosamente". At the bottom right of the dialog box is a green button with the text "Aceptar".

Damos clic en aceptar y nos lleva a la pantalla Beneficio 1 donde se verá el cambio realizado



10.1.16 Prueba de eliminación beneficio 1

Para la eliminación de beneficio 1 al frente del nombre se encuentra el botón eliminar



Se presiona el botón sale el siguiente mensaje

127.0.0.1 dice

Beneficio eliminado exitosamente

Aceptar

Le damos clic en aceptar y nos envía a la pantalla beneficio 1 donde ya no aparecerá el beneficio eliminado

The screenshot shows a web interface titled "Beneficios registrados". On the left is a navigation menu with links: Inicio, FUI/D, Beneficio 1, Beneficio 2, Algún otro Beneficio, Antigüedad Sisbén, Tipos de población, Universidades, Carreros, and Ciudad. The main content area has a header "foest1" and a "Salir" link. Below the header is a section "Ingresar nuevo beneficio" with a search bar. A dropdown menu titled "Beneficios 1" is open, displaying a table of benefit types:

Nombre	Modificar	Eliminar
BECA GESTIONADA	Modificar	Eliminar
EXCELENCIA ACADEMICA	Modificar	Eliminar
EXCELENCIA CULTURAL	Modificar	Eliminar
EXCELENCIA DEPORTIVA	Modificar	Eliminar
INVESTIGACION	Modificar	Eliminar
LINEA 2	Modificar	Eliminar
LINEA 4	Modificar	Eliminar
MEJOR IC/PES	Modificar	Eliminar
MOVILIDAD	Modificar	Eliminar
NINGUNO	Modificar	Eliminar

Below the table, it says "Showing 1 to 10 of 12 entries" and includes "Previous", "1", "2", and "Next" navigation buttons.

10.1.17 Prueba de creación de beneficio 2

En la página principal, o en las en las distintas paginas o pantallas aparece una opción llamada Beneficio 2

The screenshot shows a web interface titled "Sistema de información". On the left is a navigation menu with links: foest1, FUI/D, Beneficio 1, Beneficio 2, Algún otro Beneficio, Antigüedad Sisbén, Tipos de población, Universidades, Carreros, and Ciudad. The main content area has a header "foest1" and a "Salir" link. Below the header is a section "Nuevo periodo:" with a form containing "Año:" (with a dropdown), "semestre:" (with a dropdown), and a "Nuevo" button. Below the form is a list of student records:

- Estudiantes 2018-2
- Estudiantes 2019-1
- Estudiantes 2019-2
- Estudiantes 2020-1

Le damos clic y nos dirigirá a una pantalla donde se verán los beneficios registrados

Nombre	Modificar	Eliminar
MOVILIDAD	Modificar	Eliminar
INGRESO	Modificar	Eliminar
SOSTENIMIENTO	Modificar	Eliminar
TRANSPORTE	Modificar	Eliminar

Seleccionamos la opción de ingreso nuevo beneficio la cual nos llevara a un formulario el cual se llena con el nombre del beneficio

Nombre Beneficio * Digite el nombre

Registrar Volver

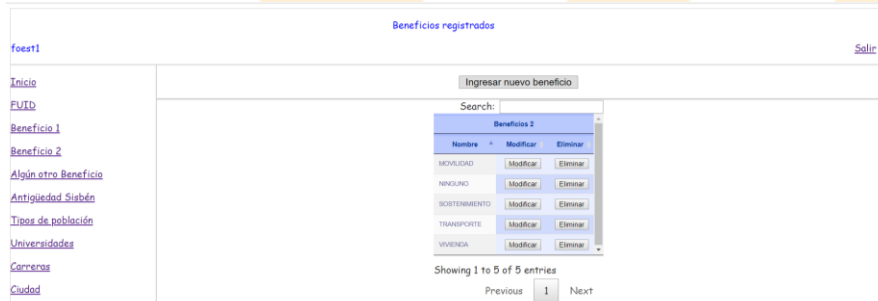
Después de lleno damos clic en registrar y nos aparece el siguiente mensaje

127.0.0.1 dice

Beneficio agregado exitosamente

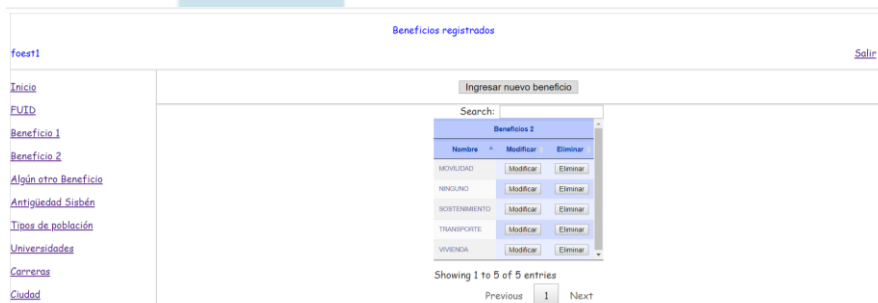
Aceptar

Le damos clic en aceptar, este nos llevara a la pantalla Beneficios 1, en la cual se puede revisar el número de registros debajo de la tabla o buscando el nombre en el cuadro.



10.1.18 Prueba de modificación beneficio 2

En la pantalla beneficio 2 al frente del beneficio que se quiera modificar hay un botón el cual se llama modificar



Esta opción nos lleva a una pantalla donde se vera la información del beneficio a modificar

foest1 [Salir](#)

Los espacios marcados con asterisco (*) son obligatorios

Nombre Beneficio * VIVIENDA

[Modificar](#) [Volver](#)

[Inicio](#)
[FUI/D](#)
[Beneficio 1](#)
[Beneficio 2](#)
[Algún otro Beneficio](#)
[Antigüedad Sisbén](#)
[Tipos de población](#)
[Universidades](#)
[Carreras](#)
[Ciudad](#)

Se cambian los datos necesarios y se da clic en modificar y aparece el siguiente mensaje

127.0.0.1 dice

Beneficio modificado exitosamente

[Aceptar](#)

Damos clic en aceptar y nos lleva a la pantalla Beneficio 2 donde se verá el cambio realizado

foest1 [Salir](#)

Ingresar nuevo beneficio

Search:

Nombre	Modificar	Eliminar
MOVILIDAD	Modificar	Eliminar
NINGUNO	Modificar	Eliminar
SOSTENIMIENTO	Modificar	Eliminar
TRANSPORTE	Modificar	Eliminar
VIVE	Modificar	Eliminar

Showing 1 to 5 of 5 entries

[Previous](#) 1 [Next](#)

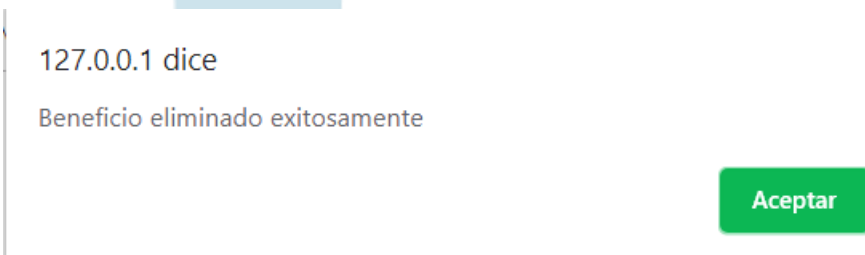
[Inicio](#)
[FUI/D](#)
[Beneficio 1](#)
[Beneficio 2](#)
[Algún otro Beneficio](#)
[Antigüedad Sisbén](#)
[Tipos de población](#)
[Universidades](#)
[Carreras](#)
[Ciudad](#)

10.1.19 Prueba de eliminación beneficio 2

Para la eliminación de beneficio 2 al frente del nombre se encuentra el botón eliminar



Se presiona el botón sale el siguiente mensaje



Le damos clic en aceptar y nos envía a la pantalla beneficio 2 donde ya no aparecerá el beneficio eliminado



10.1.20 Prueba de creación de algún otro beneficio

En la página principal, o en las en las distintas paginas o pantallas aparece una opción llamada algún otro beneficio

Sistema de información

foest1 [Salir](#)

Nuevo periodo:

Año: semestre:

[Inicio](#)

[FUTD](#)

[Beneficio 1](#)

[Beneficio 2](#)

[Algún otro Beneficio](#)

[Antigüedad Sisbén](#)

[Tipos de población](#)

[Universidades](#)

[Comunas](#)

[Ciudad](#)

[Estudiantes 2018-2](#)

[Estudiantes 2019-1](#)

[Estudiantes 2019-2](#)

[Estudiantes 2020-1](#)

Le damos clic y nos dirigirá a una pantalla donde se verán los beneficios registrados

Beneficios registrados

foest1 [Salir](#)

Search:

Nombre	Modificar	Eliminar
4X1	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
BECA SER SABANA	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
BECA UNIVERSITARIA	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
CREDITO DE VICTIMAS	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
FESOLUN	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
ICETEX	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
NINGUNO	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
OTRO	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
SER PLETOGIA	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>

Showing 1 to 9 of 9 entries

Previous Next

Seleccionamos la opción de ingrese nuevo beneficio la cual nos llevara a un formulario el cual se llena con el nombre del beneficio

Ingrese el beneficio

foest1 [Salir](#)

Inicio

FUTD

Beneficio 1

Beneficio 2

Algun otro Beneficio

Antigüedad Sisbén

Tipos de población

Universidades

Comenzó

Ciudad

Los espacios marcados con asterisco (*) son obligatorios

Nombre Beneficio *

Después de lleno damos clic en registrar y nos aparece el siguiente mensaje

127.0.0.1 dice

Beneficio agregado exitosamente

Aceptar

Le damos clic en aceptar, este nos llevara a la pantalla algún otro beneficio, en la cual se puede revisar el número de registros debajo de la tabla o buscando el nombre en el cuadro.

Beneficios registrados

foest1 [Salir](#)

Ingresar nuevo beneficio

Search:

Beneficios 2		
Nombre	Modificar	Eliminar
AKI	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
BECA BIR SAMANA	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
BECA UNIVERSITARIA	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
CREDITO DE VICTIMAS	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
FESOLUN	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
HABITANTE DE TUCUMAN	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
ICETEX	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
NINGUNO	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
OTRO	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
SER PLEO PAGA	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>

Showing 1 to 10 of 10 entries

Previous Next

10.1.21 Prueba de modificación algún otro beneficio

En la pantalla algún otro beneficio al frente del beneficio que se quiera modificar hay un botón el cual se llama modificar

The screenshot shows the 'Beneficios registrados' page. On the left is a navigation menu with links: Inicio, FUI/D, Beneficio 1, Beneficio 2, Algún otro Beneficio, Antigüedad Siabén, Tipos de población, Universidades, Carreras, and Ciudad. The main content area has a header 'Beneficios registrados' and a 'Salir' link. Below the header is a button 'Ingresar nuevo beneficio'. A search box contains 'Beneficio 2'. Below the search box is a table with 10 rows of benefits, each with 'Modificar' and 'Eliminar' buttons. The table data is as follows:

Nombre	Modificar	Eliminar
AKI	Modificar	Eliminar
BECA BER SABAÑA	Modificar	Eliminar
BECA UNIVERSARIA	Modificar	Eliminar
CREDITO DE VICTIMAS	Modificar	Eliminar
FERCUN	Modificar	Eliminar
HABITANTE DE TOCANCIPA	Modificar	Eliminar
KIETEX	Modificar	Eliminar
NINGUNO	Modificar	Eliminar
OTRO	Modificar	Eliminar
SER PLEO PAGA	Modificar	Eliminar

At the bottom of the table, it says 'Showing 1 to 10 of 10 entries' with 'Previous' and 'Next' navigation buttons.

Esta opción nos lleva a una pantalla donde se vera la información del beneficio a modificar

The screenshot shows the 'Modifique beneficio' page. On the left is the same navigation menu as in the previous screenshot. The main content area has a header 'Modifique beneficio' and a 'Salir' link. Below the header is a text input field with the value 'HABITANTE DE TOCANCIPA'. Below the input field are 'Modificar' and 'Volver' buttons. A note above the input field says 'Los espacios marcados con asterisco (*) son obligatorios'.

Se cambian los datos necesarios y se da clic en modificar y aparece el siguiente mensaje

The screenshot shows a success message dialog box with the text: '127.0.0.1 dice' and 'Beneficio modificado exitosamente'. There is a green 'Aceptar' button at the bottom right.

Damos clic en aceptar y nos lleva a la pantalla Beneficio 2 donde se verá el cambio realizado



10.1.22 Prueba de eliminación algún otro beneficio

Para la eliminación de algún otro beneficio al frente del nombre se encuentra el botón eliminar



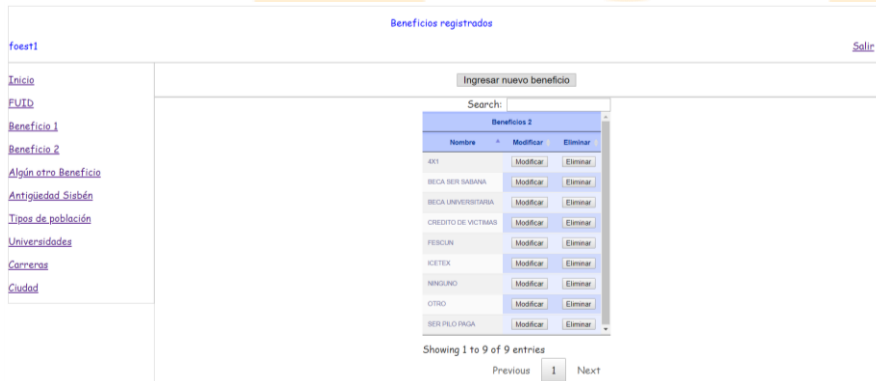
Se presiona el botón sale el siguiente mensaje

127.0.0.1 dice

Beneficio eliminado exitosamente

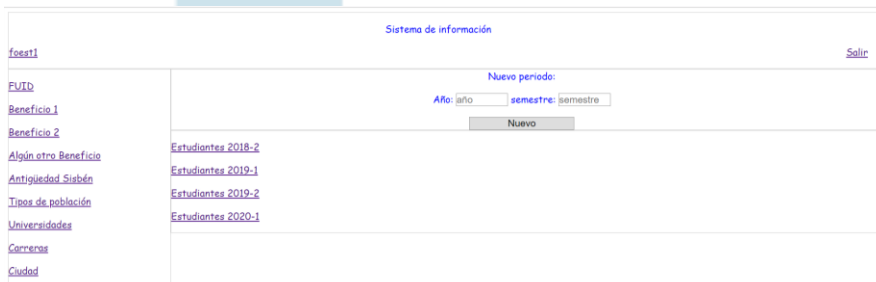
Aceptar

Le damos clic en aceptar y nos envía a la pantalla algún otro beneficio donde ya no aparecerá el beneficio eliminado



10.1.23 Prueba de creación antigüedad Sisbén

En la página principal, o en las en las distintas paginas o pantallas aparece una opción llamada antigüedad Sisbén



Le damos clic y nos dirigirá a una pantalla donde se verán la antigüedad registrada

Antigüedad Sisben

foest1 [Salir](#)

Inicio

[FUI/D](#)

[Beneficio 1](#)

[Beneficio 2](#)

[Algun otro Beneficio](#)

[Antigüedad Sisben](#)

[Tipos de población](#)

[Universidades](#)

[Carreras](#)

[Ciudad](#)

Ingresar antigüedad

Search:

Item #	Tiempo	Modificar	Eliminar
1	1 MESES	Modificar	Eliminar
2	2 MESES	Modificar	Eliminar
3	3 MESES	Modificar	Eliminar
4	4 MESES	Modificar	Eliminar
5	5 MESES	Modificar	Eliminar
6	6 MESES	Modificar	Eliminar
7	7 MESES	Modificar	Eliminar
8	8 MESES	Modificar	Eliminar
9	9 MESES	Modificar	Eliminar
10	10 MESES	Modificar	Eliminar

Showing 1 to 10 of 29 entries

Previous 1 2 3 Next

Seleccionamos la opción de ingrese antigüedad la cual nos llevara a un formulario el cual se llena con el año

Ingrese el año

foest1 [Salir](#)

Inicio

[FUI/D](#)

[Beneficio 1](#)

[Beneficio 2](#)

[Algun otro Beneficio](#)

[Antigüedad Sisben](#)

[Tipos de población](#)

[Universidades](#)

[Carreras](#)

[Ciudad](#)

Los espacios marcados con asterisco (*) son obligatorios
(Ejemplo: 3-AÑOS)

Año *:

Después de lleno damos clic en registrar y nos aparece el siguiente mensaje

127.0.0.1 dice

Dato agregado exitosamente

Le damos clic en aceptar, este nos llevara a la pantalla antigüedad Sisben, en la cual se puede revisar el número de registros debajo de la tabla o buscando el nombre en el cuadro.

Antigüedad Sisben

foest1 [Salir](#)

[Inicio](#)
[FUTD](#)
[Beneficio 1](#)
[Beneficio 2](#)
[Algún otro Beneficio](#)
[Antigüedad Sisben](#)
[Tipos de población](#)
[Universidades](#)
[Comunas](#)
[Ciudad](#)

Ingresar antigüedad

Search:

Item #	Tiempo	Modificar	Eliminar
1	1 MESES	Modificar	Eliminar
2	2 MESES	Modificar	Eliminar
3	3 MESES	Modificar	Eliminar
4	4 MESES	Modificar	Eliminar
5	5 MESES	Modificar	Eliminar
6	6 MESES	Modificar	Eliminar
7	7 MESES	Modificar	Eliminar
8	8 MESES	Modificar	Eliminar
9	9 MESES	Modificar	Eliminar
10	10 MESES	Modificar	Eliminar

Showing 1 to 10 of 30 entries

Previous 1 2 3 Next

10.1.24 Prueba de modificación antigüedad Sisben

En la pantalla antigüedad Sisben al frente de la antigüedad que se quiera modificar por error en la escritura del nombre hay un botón el cual se llama modificar

Antigüedad Sisben

foest1 [Salir](#)

[Inicio](#)
[FUTD](#)
[Beneficio 1](#)
[Beneficio 2](#)
[Algún otro Beneficio](#)
[Antigüedad Sisben](#)
[Tipos de población](#)
[Universidades](#)
[Comunas](#)
[Ciudad](#)

Ingresar antigüedad

Search:

Item #	Tiempo	Modificar	Eliminar
1	1 MESES	Modificar	Eliminar
2	2 MESES	Modificar	Eliminar
3	3 MESES	Modificar	Eliminar
4	4 MESES	Modificar	Eliminar
5	5 MESES	Modificar	Eliminar
6	6 MESES	Modificar	Eliminar
7	7 MESES	Modificar	Eliminar
8	8 MESES	Modificar	Eliminar
9	9 MESES	Modificar	Eliminar
10	10 MESES	Modificar	Eliminar

Showing 1 to 10 of 30 entries

Previous 1 2 3 Next

Esta opción nos lleva a una pantalla donde se vera la información de la antigüedad a modificar

Modifique beneficio

foest1 [Salir](#)

[Inicio](#)
[FUTD](#)
[Beneficio 1](#)
[Beneficio 2](#)
[Algún otro Beneficio](#)
[Antigüedad Sisben](#)
[Tipos de población](#)
[Universidades](#)
[Comunas](#)
[Ciudad](#)

Los espacios marcados con asterisco (*) son obligatorios

Año: 18 AÑOS

[Modificar](#) [Volver](#)

Se cambian los datos necesarios y se da clic en modificar y aparece el siguiente mensaje

127.0.0.1 dice

Dato modificado exitosamente

Aceptar

Damos clic en aceptar y nos lleva a la pantalla antigüedad Sisbén donde se verá el cambio realizado

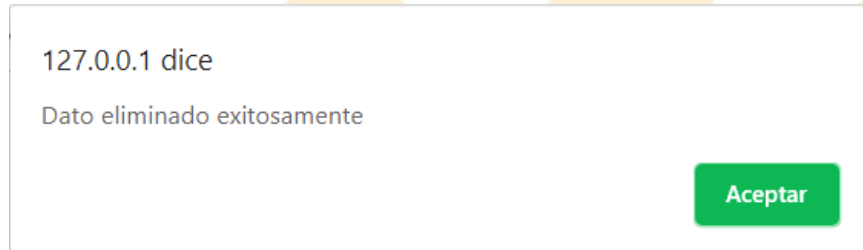
The screenshot shows the 'Antigüedad Sisbén' application interface. On the left, there is a navigation menu with links: Inicio, FUI/D, Beneficio 1, Beneficio 2, Algún otro Beneficio, Antigüedad Sisbén, Tipos de población, Universidades, Carreras, and Ciudad. The main content area has a title 'Antigüedad Sisbén' and a 'Salir' button. Below the title is a search bar and a table of items. The table has columns for 'Item #', 'Tiempo', 'Modificar', and 'Eliminar'. The table contains 10 rows of data, with the first row being 21, 9.ARG.01, and the last row being 30, 18.A.10.01. Below the table, it says 'Showing 21 to 30 of 30 entries' and has navigation buttons for 'Previous', '1', '2', '3', and 'Next'.

10.1.25 Prueba de eliminación antigüedad Sisbén

Para la eliminación de antigüedad sisben al frente del nombre se encuentra el botón eliminar

This screenshot is identical to the previous one, showing the 'Antigüedad Sisbén' application interface. The 'Eliminar' button for the first item (21, 9.ARG.01) is highlighted in blue, indicating it is the focus of the test.

Se presiona el botón sale el siguiente mensaje

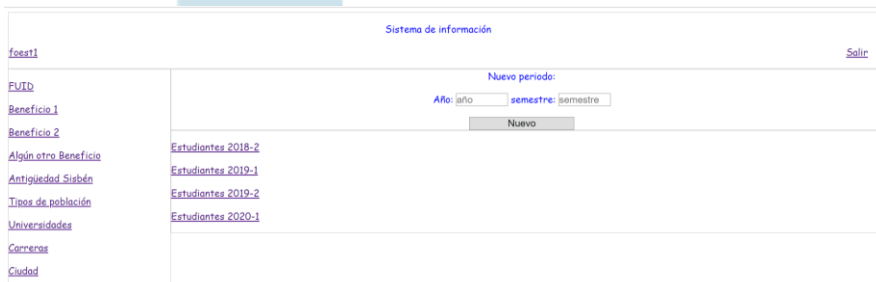


Le damos clic en aceptar y nos envía a la pantalla antigüedad beneficio donde ya no aparecerá la antigüedad eliminado



10.1.26 Prueba de creación población especial

En la página principal, o en las en las distintas paginas o pantallas aparece una opción llamada tipos de población



Le damos clic y nos dirigirá a una pantalla donde se verán la poblacion especial registrada

The screenshot shows a web interface titled "Poblacion registrada". On the left is a navigation menu with links: Inicio, FUI/D, Beneficio 1, Beneficio 2, Algún otro Beneficio, Antigüedad Siabén, Tipos de población, Universidades, Comercios, and Ciudad. The main content area has a button "Ingresar nuevo tipo de poblacion" and a search bar. Below the search bar is a table of special population types:

Nombre	Modificar	Eliminar
AFRID	Modificar	Eliminar
DISCAPACIDAD	Modificar	Eliminar
INDIGENA	Modificar	Eliminar
LITIBQ	Modificar	Eliminar
NO	Modificar	Eliminar
VICTIMA	Modificar	Eliminar

At the bottom, it says "Showing 1 to 6 of 6 entries" with "Previous" and "Next" navigation buttons.

Seleccionamos la opción de ingrese nuevo tipo de poblacion la cual nos llevara a un formulario el cual se llena con el nombre

The screenshot shows a form titled "Ingrese la poblacion especial:". It has a navigation menu on the left. The main form area contains a text input field labeled "Poblacion *" with a placeholder "Escriba el nombre". Below the input field are two buttons: "Registrar" and "Volver". A note at the top of the form says "Los espacios marcados con asterisco (*) son obligatorios".

Después de lleno damos clic en registrar y nos aparece el siguiente mensaje

127.0.0.1 dice

Tipo de poblacion agregada exitosamente

Aceptar

Le damos clic en aceptar, este nos llevara a la pantalla tipos de poblacion, en la cual se puede revisar el número de registros debajo de la tabla o buscando el nombre en el cuadro.



10.1.27 Prueba de modificación poblacion especial

En la pantalla tipos de poblacion al frente de la poblacion que se quiera modificar por error en la escritura del nombre hay un botón el cual se llama modificar



Esta opción nos lleva a una pantalla donde se vera la información de la poblacion a modificar

Modifique la poblacion

foest1 [Salir](#)

Inicio

FLUTD

Beneficio 1

Beneficio 2

Algun otro Beneficio

Antigüedad Sisbén

Tipos de población

Universidades

Comercio

Ciudad

Los espacios marcados con asterisco (*) son obligatorios

Nombre Poblacion *

Se cambian los datos necesarios y se da clic en modificar y aparece el siguiente mensaje

127.0.0.1 dice

Poblacion modificada exitosamente

Aceptar

Damos clic en aceptar y nos lleva a la pantalla tipos de poblacion donde se verá el cambio realizado

Poblacion registrada

foest1 [Salir](#)

Inicio

FLUTD

Beneficio 1

Beneficio 2

Algun otro Beneficio

Antigüedad Sisbén

Tipos de población

Universidades

Comercio

Ciudad

Ingresar nuevo tipo de poblacion

Search:

Tipos de poblacion especial		
Nombre	Modificar	Eliminar
AFRO	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
DISCAPACIDAD	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
GITANO O ROM	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
INDIGENA	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
LITRIG	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
NO	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>
VICTIMA	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>

Showing 1 to 7 of 7 entries

Previous Next

10.1.28 Prueba de eliminación poblacion especial

Para la eliminación de poblacion al frente del nombre se encuentra el botón eliminar

The screenshot shows a web application interface titled "Poblacion registrada". On the left is a navigation menu with links: Inicio, FUI/D, Beneficio 1, Beneficio 2, Algún otro Beneficio, Antigüedad Sisbén, Tipos de población, Universidades, Comunas, and Ciudad. The main content area has a search bar with the text "Ingresar nuevo tipo de poblacion" and a "Search:" field. Below the search bar is a table with 7 rows, each representing a type of special population. Each row has a "Nombre" column and "Modificar" and "Eliminar" buttons. The entries are: AFRO, DISCAPACIDAD, GITANO O ROM, INDIENA, LITIBO, NO, and VICTIMA. At the bottom of the table, it says "Showing 1 to 7 of 7 entries" and has "Previous", "1", and "Next" navigation buttons.

Se presiona el botón sale el siguiente mensaje

127.0.0.1 dice

Tipo de poblacion eliminado exitosamente

Aceptar

Le damos clic en aceptar y nos envía a la pantalla tipos de poblacion donde ya no aparecerá la poblacion eliminada

This screenshot is similar to the first one, showing the "Poblacion registrada" interface. However, the search results now show only 6 entries, indicating that the previously selected entry has been successfully removed. The "Showing 1 to 6 of 6 entries" text at the bottom of the table confirms this.

10.1.29 Prueba de creación de registro de universidad

En la página principal, o en las en las distintas paginas o pantallas aparece una opción llamada universidades

The screenshot shows a web interface titled 'Sistema de información'. On the left is a vertical menu with links: foest1, FUI/D, Beneficio 1, Beneficio 2, Algún otro Beneficio, Antigüedad Sisbén, Tipos de población, Universidades, Comenos, and Ciudad. The main content area has a 'Nuevo periodo:' section with 'Año:' and 'Semestre:' dropdowns and a 'Nuevo' button. Below this are links for 'Estudiantes 2018-2', 'Estudiantes 2019-1', 'Estudiantes 2019-2', and 'Estudiantes 2020-1'. A 'Salir' link is in the top right corner.

Le damos clic y nos dirigirá a una pantalla donde se verán las universidades registrada

The screenshot shows a web interface titled 'Universidades'. It features a 'Registrar nueva universidad' button at the top. Below it is a search bar and a table of registered universities. The table has columns for 'Universidad', 'Porcentaje universidad', 'Modificar', and 'Eliminar'. The table lists several universities with a 30% percentage. At the bottom, it says 'Showing 1 to 10 of 101 entries' and has pagination controls for 'Previous', '1', '2', '3', '4', '5', '11', and 'Next'.

Universidad	Porcentaje universidad	Modificar	Eliminar
COLEGIO MAYOR DE NUESTRA SEÑORA DEL ROSARIO	30	Modificar	Eliminar
CORPORACION DE EDUCACION TECNOLOGICA COLUSION AREDES GROUP	30	Modificar	Eliminar
CORPORACION ESCUELA DE ARTES Y LETRAS	30	Modificar	Eliminar
CORPORACION TECNOLOGICA INDUSTRIAL COLUMBIANA TECNICO	30	Modificar	Eliminar
CORPORACION UNIFICADA NACIONAL DE EDUCACION SUPERIOR - CUN	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA CENDA	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA DE ASTURIAS	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA DEL HULA-CORRIOLA	30	Modificar	Eliminar

Seleccionamos la opción de registrar nueva universidad la cual nos llevara a un formulario el cual se llena con el nombre y el porcentaje.

Ingrese la universidad:

foest1 Salir

Inicio

EVTD

Beneficio 1

Beneficio 2

Algún otro Beneficio

Antigüedad Sisbén

TiPos de población

Universidades

Comercio

Ciudad

Los espacios marcados con asterisco (*) son obligatorios

Nombre *

porcentaje sin signo *

Registrar Volver

Después de lleno damos clic en registrar y nos aparece el siguiente mensaje

127.0.0.1 dice

Universidad agregada exitosamente

Aceptar

Le damos clic en aceptar, este nos llevara a la pantalla universidad, en la cual se puede revisar el número de registros debajo de la tabla o buscando el nombre en el cuadro.

Universidades

foest1 Salir

Registrar nueva universidad

Search:

Universidad	Porcentaje universidad	Modificar	Eliminar
COLEGIO MAYOR DE NUESTRA SEÑORA DEL ROSARIO	30	Modificar	Eliminar
CORPORACION DE EDUCACION TECNOLOGICA COLSUBSIDIO AIRBUS GROUP	30	Modificar	Eliminar
CORPORACION ESCUELA DE ARTES Y LETRAS	30	Modificar	Eliminar
CORPORACION TECNOLOGICA INDUSTRIAL COLOMBIANA TECNICO	30	Modificar	Eliminar
CORPORACION IMPAGADA NACIONAL DE EDUCACION SUPERIOR - OJUN	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA CENIDA	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA DE ASTURIAS	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA DEL HUILA-CORHUILA	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA BESIAMERICANA	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA BESIAMERICANA CONTADURIA TOCANCIPA	30	Modificar	Eliminar

Showing 1 to 10 of 102 entries

Previous 1 2 3 4 5 11 Next

10.1.30 Prueba de modificación de registro de universidad

En la pantalla universidades al frente de la universidad que se quiera modificar por error en la escritura del nombre o porcentaje hay un botón el cual se llama modificar

Universidad	Porcentaje universidad	Modificar	Eliminar
COLEGIO MAYOR DE NUESTRA SEÑORA DEL ROSARIO	30	Modificar	Eliminar
CORPORACION DE EDUCACION TECNOLOGICA COLSUBSIDIO-ARREBIS GROUP	30	Modificar	Eliminar
CORPORACION ESCUELAS DE ARTES Y LETRAS	30	Modificar	Eliminar
CORPORACION TECNOLOGICA INDUSTRIAL COLOMBIANA TECNICO	30	Modificar	Eliminar
CORPORACION UNIFICADA NACIONAL DE EDUCACION SUPERIOR - CUN	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA CENEA	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA DE ARTURIAS	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA DEL HUILA-CORHUILA	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA BESOGAMERICANA	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA BESOGAMERICANA CORRIQUERA YOUNGWIN	30	Modificar	Eliminar

Esta opción nos lleva a una pantalla donde se vera la información de la universidad a modificar

Los espacios marcados con asterisco (*) son obligatorios

Nombre * HARVARD

porcentaje sin signo * 5%

Modificar Volver

Se cambian los datos necesarios y se da clic en modificar y aparece el siguiente mensaje

127.0.0.1 dice

Universidad modificada exitosamente

Aceptar

Damos clic en aceptar y nos lleva a la pantalla universidades donde se verá el cambio realizado

The screenshot shows a web application interface for managing universities. On the left is a navigation menu with links like 'Inicio', 'FUI/D', 'Beneficio 1', 'Beneficio 2', 'Algún otro Beneficio', 'Antigüedad Sisbén', 'Tipos de población', 'Universidades', 'Comenzar', and 'Ciudad'. The main content area has a header 'Universidades' and a 'Salir' button. Below the header is a 'Registrar nueva universidad' button and a search bar. A table lists various universities with columns for 'Universidad', 'Porcentaje universidad', 'Modificar', and 'Eliminar'. The table shows 10 entries, with the first one highlighted. At the bottom, it says 'Showing 41 to 50 of 102 entries' and includes pagination controls.

Universidad	Porcentaje universidad	Modificar	Eliminar
FUNDACION UNIVERSITARIA LOS LIBERTADORES	30	Modificar	Eliminar
FUNDACION UNIVERSITARIA MONSERRATE	30	Modificar	Eliminar
FUNDACION UNIVERSITARIA PANAMERICANA	30	Modificar	Eliminar
FUNDACION UNIVERSITARIA UNIHORIZONTE	30	Modificar	Eliminar
FUNDACION UNIVERSITARIA NEWELL	30	Modificar	Eliminar
HAWAII	10	Modificar	Eliminar
INSTITUCION UNIVERSITARIA COLEGIOS DE COLOMBIA - UNICC SIN CONTENIDO	25	Modificar	Eliminar
INSTITUCION UNIVERSITARIA COLEGIOS DE COLOMBIA - UNICC SIN CONTENIDO	30	Modificar	Eliminar
INSTITUCION UNIVERSITARIA COLOMBIA AMERICANA - UNICA	30	Modificar	Eliminar
INSTITUCION UNIVERSITARIA DE COLOMBIA	30	Modificar	Eliminar

10.1.31 Prueba de eliminación de registro de universidad

Para la eliminación de una universidad al frente del nombre se encuentra el botón eliminar

This screenshot is identical to the previous one, showing the 'Universidades' management interface. The table lists various universities, and the 'Eliminar' button is visible for each entry. The interface includes a navigation menu, a search bar, and pagination controls.

Se presiona el botón sale el siguiente mensaje

127.0.0.1 dice

Universidad eliminada exitosamente

Aceptar

Le damos clic en aceptar y nos envía a la pantalla universidades donde ya no aparecerá la universidad eliminada

Universidad	Porcentaje universidad	Modificar	Eliminar
COLEGIO MAYOR DE NUESTRA SEÑORA DEL ROSARIO	30	Modificar	Eliminar
CORPORACION DE EDUCACION TECNOLÓGICA COLUBERDO-ARREUS GROUP	30	Modificar	Eliminar
CORPORACION ESCUELA DE ARTES Y LETRAS	30	Modificar	Eliminar
CORPORACION TECNOLÓGICA INDUSTRIAL COLOMBIANA TECNICO	30	Modificar	Eliminar
CORPORACION UNIFICADA NACIONAL DE EDUCACION SUPERIOR - CUN	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA CENDA	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA DE ARTURIAS	30	Modificar	Eliminar
CORPORACION UNIVERSITARIA DEL HULA-CORRALA	30	Modificar	Eliminar
UNIVERSIDAD NACIONAL INGENIERIA AERONAUTICA	30	Modificar	Eliminar

10.1.32 Prueba de creación de registro de carreras

En la página principal, o en las en las distintas paginas o pantallas aparece una opción llamada carreras

Nuevo periodo:

Año: año semestre semestre

Nuevo

[Estudiantes 2018-2](#)

[Estudiantes 2019-1](#)

[Estudiantes 2019-2](#)

[Estudiantes 2020-1](#)

Le damos clic y nos dirigirá a una pantalla donde se verán las carreras registradas registrada

The screenshot shows a web interface titled 'Carreras'. On the left is a navigation menu with links: Inicio, FUI/D, Beneficio 1, Beneficio 2, Algún otro Beneficio, Antigüedad Sisbén, Tipos de población, Universidades, Carreras, and Ciudad. The main content area has a 'Registrar nueva carrera' button at the top. Below it is a search bar and a table of careers. The table has columns for 'Nombre', 'Modificar', and 'Eliminar'. The careers listed are: ADMINISTRACION AMBIENTAL, ADMINISTRACION DE SEGURIDAD Y SALUD OCUPACIONAL, ADMINISTRACION DE EMPRESAS, ADMINISTRACION HOTELERA Y TURISTICA, ADMINISTRACION PUBLICA, ADMINISTRACION SALUD OCUPACIONAL, ARQUITECTURA, ARTES ESCENICAS, ARTES MUSICALES, and AUXILIAR DE ENFERMERIA. At the bottom, it says 'Showing 1 to 10 of 127 entries' with pagination controls.

Seleccionamos la opción de registrar nueva carrera la cual nos llevara a un formulario el cual se llena con el nombre.

The screenshot shows a form titled 'Ingrese la carrera:'. It has a navigation menu on the left similar to the previous screenshot. The main form area has a text input field labeled 'Nombre * [Digite el nombre]'. Below the input field are two buttons: 'Registrar' and 'Volver'. A note above the input field says 'Los espacios marcados con asterisco (*) son obligatorios'.

Después de lleno damos clic en registrar y nos aparece el siguiente mensaje

The screenshot shows a success message. It says '127.0.0.1 dice' followed by 'Carrera agregada exitosamente'. At the bottom right of the message area is a green button labeled 'Aceptar'.

Le damos clic en aceptar, este nos llevara a la pantalla universidad, en la cual se puede revisar el número de registros debajo de la tabla o buscando el nombre en el cuadro.

foest1 Salir

Registrar nueva carrera

Search:

Nombre	Modificar	Eliminar
FISICA	[Modificar]	[Eliminar]
FISIOTERAPIA	[Modificar]	[Eliminar]
GASTRONOMIA	[Modificar]	[Eliminar]
GASTRONOMIA SOMMELIER	[Modificar]	[Eliminar]
GASTRONOMIA Y COCINA	[Modificar]	[Eliminar]
GERENCIA EN SEGURIDAD Y SALUD EN EL TRABAJO	[Modificar]	[Eliminar]
GERENCIA FINANCIERA INTERNACIONAL	[Modificar]	[Eliminar]
GESTION BANCARIA	[Modificar]	[Eliminar]
GESTION DE LA CALIDAD, MEDIO AMBIENTE Y SEGURIDAD	[Modificar]	[Eliminar]
GESTION INTEGRADA DE LA CALIDAD	[Modificar]	[Eliminar]

Showing 41 to 50 of 128 entries

Previous 1 ... 4 **5** 6 ... 13 Next

10.1.33 Prueba de modificación de registro de carreras

En la pantalla carreras al frente del nombre de la que se quiera modificar por error en la escritura del nombre hay un botón el cual se llama modificar

foest1 Salir

Registrar nueva carrera

Search:

Nombre	Modificar	Eliminar
FISICA	[Modificar]	[Eliminar]
FISIOTERAPIA	[Modificar]	[Eliminar]
GASTRONOMIA	[Modificar]	[Eliminar]
GASTRONOMIA SOMMELIER	[Modificar]	[Eliminar]
GASTRONOMIA Y COCINA	[Modificar]	[Eliminar]
GERENCIA EN SEGURIDAD Y SALUD EN EL TRABAJO	[Modificar]	[Eliminar]
GERENCIA FINANCIERA INTERNACIONAL	[Modificar]	[Eliminar]
GESTION BANCARIA	[Modificar]	[Eliminar]
GESTION DE LA CALIDAD, MEDIO AMBIENTE Y SEGURIDAD	[Modificar]	[Eliminar]
GESTION INTEGRADA DE LA CALIDAD	[Modificar]	[Eliminar]

Showing 41 to 50 of 128 entries

Previous 1 ... 4 **5** 6 ... 13 Next

Esta opción nos lleva a una pantalla donde se vera la información de la carrera a modificar

foest1 Salir

Modifique la carrera

Los espacios marcados con asterisco (*) son obligatorios

Nombre * FISICA

[Modificar] [Volver]

Se cambian los datos necesarios y se da clic en modificar y aparece el siguiente mensaje

127.0.0.1 dice

Carrera modificada exitosamente

Aceptar

Damos clic en aceptar y nos lleva a la pantalla carreras donde se verá el cambio realizado

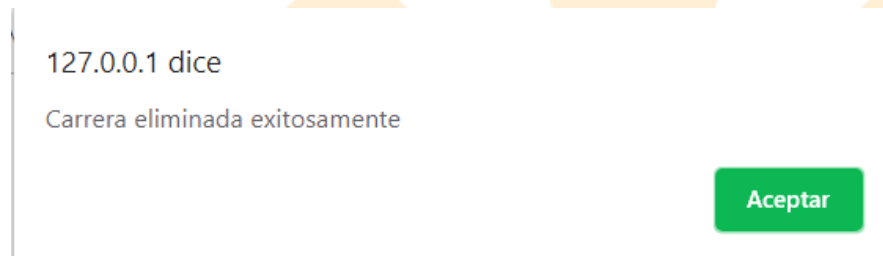
Nombre	Modificar	Eliminar
DERECHO	Modificar	Eliminar
DISEÑO DE MODAS	Modificar	Eliminar
DISEÑO GRAFICO	Modificar	Eliminar
DISEÑO PARA LA COMUNICACION GRAFICA	Modificar	Eliminar
DOCTORADO EN EDUCACION	Modificar	Eliminar
DOCTORADO EN FISICA	Modificar	Eliminar
ECONOMIA	Modificar	Eliminar
ELECTROMECANICA	Modificar	Eliminar
ELECTRONICA	Modificar	Eliminar
ENFERMERIA	Modificar	Eliminar

10.1.34 Prueba de eliminación de registro de carreras

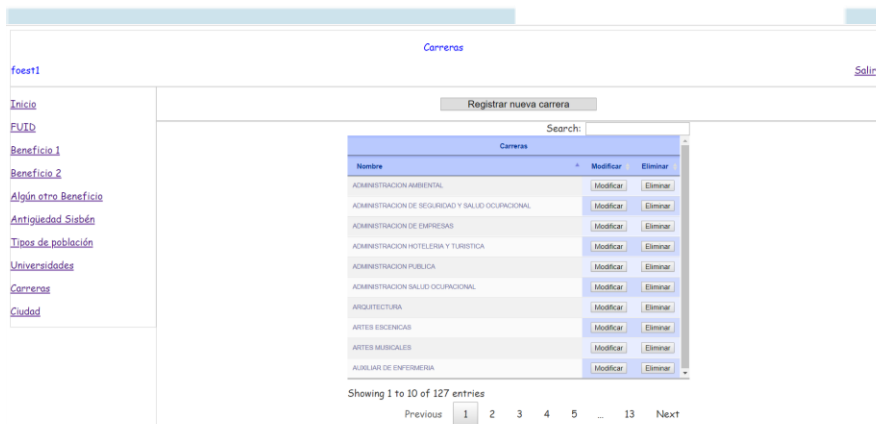
Para la eliminación de una carrera al frente del nombre se encuentra el botón eliminar

Universidad	Porcentaje universidad	Modificar	Eliminar
FUNDACION UNIVERSITARIA LOS LIBERTADORES	30	Modificar	Eliminar
FUNDACION UNIVERSITARIA MONSERRATE	30	Modificar	Eliminar
FUNDACION UNIVERSITARIA PANAMERICANA	30	Modificar	Eliminar
FUNDACION UNIVERSITARIA UNIHORIZONTE	30	Modificar	Eliminar
FUNDACION UNIVERSITARIA INEVEL	30	Modificar	Eliminar
HAWANE	30	Modificar	Eliminar
INSTITUCION UNIVERSITARIA COLEGIO DE COLOMBIA - UNICC COPIAÑO	25	Modificar	Eliminar
INSTITUCION UNIVERSITARIA COLEGIO DE COLOMBIA - UNICC BIV CONVENIO	30	Modificar	Eliminar
INSTITUCION UNIVERSITARIA COLOMBO AMERICANA - UNICA	30	Modificar	Eliminar
INSTITUCION UNIVERSITARIA DE COLOMBIA	30	Modificar	Eliminar

Se presiona el botón sale el siguiente mensaje

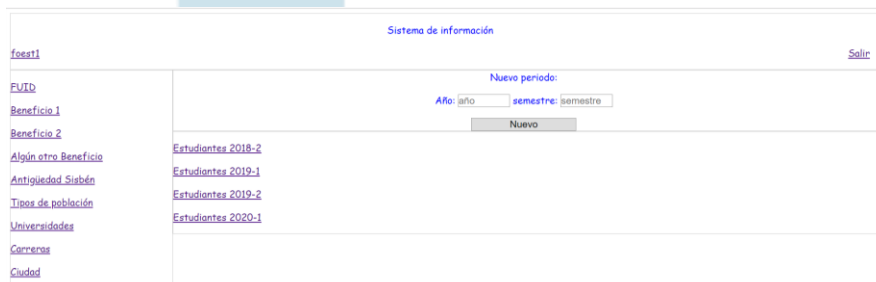


Le damos clic en aceptar y nos envía a la pantalla universidades donde ya no aparecerá la universidad eliminada



10.1.35 Prueba de creación de registro de ciudad

En la página principal, o en las en las distintas paginas o pantallas aparece una opción llamada ciudad



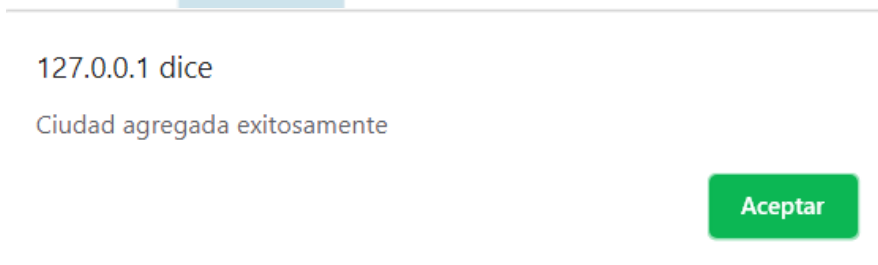
Le damos clic y nos dirigirá a una pantalla donde se verán las ciudades registradas registrada



Seleccionamos la opción de registrar nueva ciudad la cual nos llevara a un formulario el cual se llena con el nombre.



Después de lleno damos clic en registrar y nos aparece el siguiente mensaje



Le damos clic en aceptar, este nos llevara a la pantalla ciudad, en la cual se puede revisar el número de registros debajo de la tabla o buscando el nombre en el cuadro.



10.1.36 Prueba de modificación de registro de ciudad

En la pantalla ciudad al frente del nombre de la que se quiera modificar por error en la escritura hay un botón el cual se llama modificar



Esta opción nos lleva a una pantalla donde se vera la información de la ciudad a modificar

foest1 [Salir](#)

Modifique la ciudad

Los espacios marcados con asterisco (*) son obligatorios

Nombre * SAGHANCIPA

[Modificar](#) [Volver](#)

[Inicio](#)
[FUI/D](#)
[Beneficio 1](#)
[Beneficio 2](#)
[Algún otro Beneficio](#)
[Antigüedad Siabén](#)
[Tipos de población](#)
[Universidades](#)
[Camerón](#)
[Ciudad](#)

Se cambian los datos necesarios y se da clic en modificar y aparece el siguiente mensaje

127.0.0.1 dice

Ciudad modificada exitosamente

[Aceptar](#)

Damos clic en aceptar y nos lleva a la pantalla carreras donde se verá el cambio realizado

foest1 [Salir](#)

Ciudades a las que viajan

Registrar nueva ciudad

Search:

Nombre	Modificar	Eliminar
BOGOTÁ	Modificar	Eliminar
CAJICÁ	Modificar	Eliminar
CHÍA	Modificar	Eliminar
FAGATÁ	Modificar	Eliminar
GACHÁ	Modificar	Eliminar
LA CARA	Modificar	Eliminar
MORQUENA	Modificar	Eliminar
BERGUELE	Modificar	Eliminar
SOPO	Modificar	Eliminar
TOGANCIPA	Modificar	Eliminar

Showing 1 to 10 of 12 entries

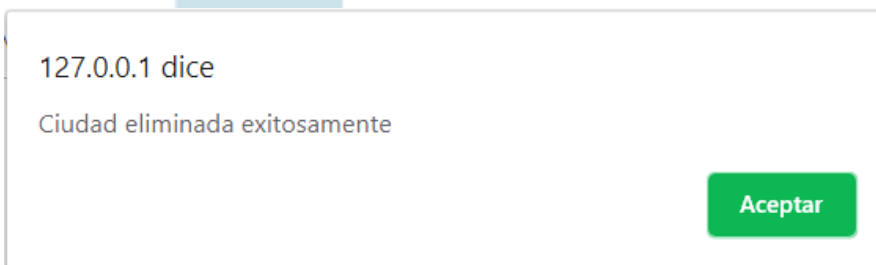
Previous [1](#) [2](#) Next

10.1.37 Prueba de eliminación de registro de ciudad

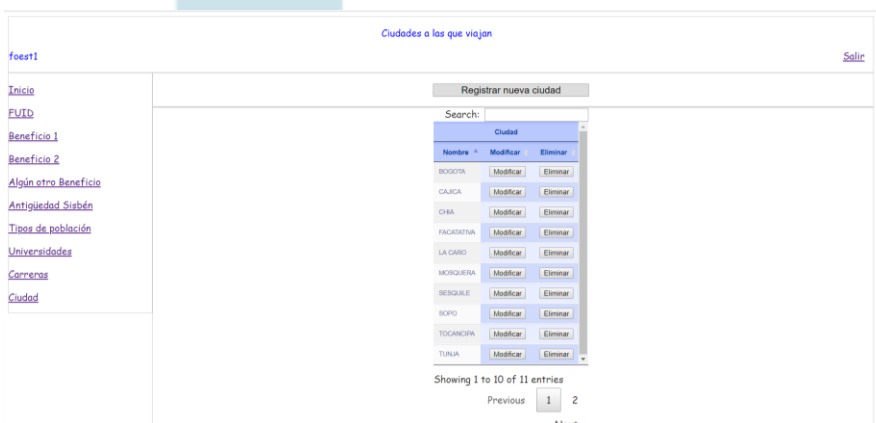
Para la eliminación de una ciudad al frente del nombre se encuentra el botón eliminar



Se presiona el botón sale el siguiente mensaje



Le damos clic en aceptar y nos envía a la pantalla universidades donde ya no aparecerá la universidad eliminada



Capítulo 11 Conclusiones

- La ingeniería de software es muy importante para planificar proyectos, ya que demuestra lo que se piensa desarrollar paso a paso.
- Las aplicaciones en PHP son aplicaciones multiplataforma, o sea que no hay necesidad de cambiar el código o el lenguaje si se ejecuta en diferentes sistemas operativos, tanto d dispositivos móviles y PCs.
- Programar en PHP requiere de usar navegadores web para la visualización de los programas creados.
- La programación orientada a objetos nos enseña una manera de programar creando objetos y dividiendo el sistema o aplicación en subsistemas los cuales se encargan de ejecutar una parte del código.
- Los programas creados con cualquier lenguaje de programación, tiene el fin de agilizar procesos y resguardar información

Capítulo 12 Referencias Bibliográficas

- Weitzenfeld, Alfredo. Ingeniería de Software Orientada a Objetos con UML, Java e internet. Mexico City: Cengage Learning, 2005. Gale Virtual Reference Library. "Desarrollo de Software Orientado a Objetos." Ingeniería de Software Orientada a Objetos con UML, Java e Internet. Alfredo Weitzenfeld. Mexico City: Cengage Learning, 2005. [193]. Gale Virtual Reference Library. Web. 8 July 2014.
- www.unad.edu.co curso programación de sitios web
- Jorge Eduardo Salazar Zúñiga & Orlando Puentes Andrade. Programación de sitios web – PHP (100). Colombia.
- Yhon Jerson Robles Puentes (mayo 2015). Fase 4 –manejo de base de datos