

Desarrollo de aplicación como banco digital de información médica de diagnósticos y medicamentos.

Presentado por:

Julio Cesar Fuentes Cortes

Asesor:

Luis Abdul Samacá Chavarriga Magíster en E-Learning y Redes Sociales

Universidad Nacional Abierta a Distancia UNAD

Escuela de ciencias básicas, Tecnología e Ingeniería-ECBTI

Ingeniera en Sistemas

Abril 2024

Nota de aceptación

Aprobado en cumplimiento de los requisitos exigidos por la Universidad Nacional Abierta y A Distancia (UNAD) para optar por el título de Ingeniero de Sistemas.

Director de la opción de grado Luis Abdul Samacá Chavarriaga

Jurado

Dedicatoria

En la culminación de mi carrera, agradezco principalmente a Dios y no habría sido posible sin el apoyo incondicional de mi madre que a pesar de las dificultades lo ha dado todo por verme como un profesional destacado, también a mi maravillosa esposa y a mis hermosos hijos que han sido mi apoyo, mi pilar más fuerte para alcanzar el éxito en la culminación de mi carrera, también agradezco a cada una de las personas que me han apoyado como tutores, compañeros tanto de la universidad como en el ambiente laboral.

Resumen

El presente proyecto aplicado tiene como objetivo principal el desarrollo de una aplicación que permita la sistematización del CIE en su versión 10, Clasificación internacional de enfermedades, 10.^a edición. Se trata de una clasificación de las enfermedades basados el que tiene como propósito permitir el registro sistemático, análisis, interpretación, y comparación de los datos de mortalidad y morbilidad recolectados en diferentes países o áreas, y en diferentes épocas.

Se utiliza para convertir los términos diagnósticos y otros problemas de salud, de palabras a códigos alfanuméricos que permiten su fácil almacenamiento y posterior recuperación para el análisis de la información. La finalidad última de esta aplicación es que permita consultar el CIE 10 de una manera rápida y sencilla que beneficie al personal de la salud.

Palabras Clave: CIE10, automatización, diagnósticos, factores, síntomas, exposición ocupaciones. Clasificación internacional de enfermedades, 10.^a edición

Abstract

The main objective of this applied project is the development of an application that allows the systematization of the ICD-10, International Classification of Diseases, 10th edition. It is a classification of diseases aimed at enabling the systematic recording, analysis, interpretation, and comparison of mortality and morbidity data collected in different countries or areas and at different times. It is used to convert diagnostic terms and other health problems from words to alphanumeric codes that allow for easy storage and subsequent retrieval for information analysis. The ultimate purpose of this application is to enable quick and easy access to ICD-10 for the benefit of healthcare personnel.

Keywords: ICD-10, automation, diagnostics, factors, symptoms, occupational exposure.

International Classification of Diseases, 10th edition.

Declaración de derechos de propiedad intelectual

Los autores de la presente propuesta manifiestan que conocen el contenido del Acuerdo 06 de 2008, Estatuto de Propiedad Intelectual de la UNAD, Artículo 39 referente a la cesión voluntaria y libre de los derechos de propiedad intelectual de los productos generados a partir del presente documento. Así mismo, conocen el contenido del Artículo 40 del mismo Acuerdo, relacionado con la autorización de uso del trabajo para fines de consulta y mención en los catálogos bibliográficos de la UNAD.

Tabla de contenido

Dedicatoria	3
Resumen	4
Abstract	5
Declaración de derechos de propiedad intelectual	6
Introducción	14
Planteamiento del problema	15
Formulación de pregunta.....	15
Justificación.....	16
Objetivos	17
Marco de referencia.....	18
Marco teórico	20
Evolución del software.....	20
Ingeniería del software	22
Fases de desarrollo,	23
Según McDonnell, S. (2004). Las fases de desarrollo son:.....	23
Modelos para el diseño de software	23
Metodologías para el desarrollo de software.....	24
Paradigmas en el desarrollo de software	25
Boceto del software	27
Inclusión de interfaces gráficas en el software.....	27
Marco Conceptual	28
Programación orientada a objetos.....	28
Bases de datos relacionales.....	29
Aplicativo web	29

Versión de la CIE	30
Diagnóstico	32
Procesos administrativos	32
Plataforma	32
Marco Contextual	33
Delimitaciones	34
Delimitación Cognitiva	34
Delimitación Temporal }	35
Delimitación Financiera	36
Metodología	37
Arquitectura De Software Modelo Vista Controlador	38
Módulos del MVC.....	38
Patrones de diseño	39
Tecnologías Web.....	40
Arquitectura.....	40
Segun Bass, Clements y Kazman (2012), son:.....	40
Seguridad informática	41
Variables.....	42
Universo	42
Población.....	42
Muestra.....	42
Diseño y desarrollo del software.....	43
Modelo de Desarrollo	43
Modelo en cascada	43
Modelo en prototipos	43

Modelo en espiral	44
Análisis	45
Identificación de actores.....	45
Identificación de roles y permisos.....	47
Identificación de requerimientos	48
Diseño.....	49
Modelo lógico relacional.....	49
Diccionario de datos	50
Modelo de clases	55
Modelo de flujo de datos	56
Diagramas de caso de uso	57
Diagrama de secuencia.....	59
Implementación.....	60
Módulo de inicio de sesión.....	61
Olvido contraseña.....	61
Menús aplicación.....	62
Noticias.....	62
Filtro avanzado	63
Botón Crear diagnóstico	64
Botón Editar	65
Botón eliminar.....	65
Mensaje eliminar	66
Botón inhabilitar.....	66
Mensaje inhabilitar.....	67
Sección Consultar.....	67

Sección Consultar.....	67
Nueva solicitud.....	68
Mis solicitudes.....	68
Mis Consultas.....	69
Principales diagnósticos	69
Configuración.....	70
Administración de noticias.....	70
Administración de usuarios.....	71
Creación y asignación perfil.....	71
Código fuente	72
Pruebas	73
Matriz de evaluación	75
Métricas de evaluación.....	75
Plan de evaluación.....	76
Pruebas funcionales.....	77
Pruebas no funcionales.....	78
Conclusiones	79
Recomendaciones.....	80
Referencias	81

Lista de tablas

Tabla 1. Cronograma de actividades	35
Tabla 2: Presupuesto inicial.	36
Tabla 3. Identificación de actores.....	45
Tabla 4. Roles y permisos para el usuario final.....	47
Tabla 6. Perfiles.....	50
Tabla 7. Usuarios.....	50
Tabla 8. Diagnostico.....	51
Tabla.9 Solicitudes Back	52
Tabla 10. Noticias.....	53
Tabla 11. Lee noticias	53
Tabla 12. Módulos.....	54
Tabla 13. Ingreso	54
Tabla 14. Pruebas funcionales.	73
Tabla 15. Matriz de evaluación del software.....	75
Tabla 16. Métricas	76
Tabla 17. Pruebas funcionales.....	77
Tabla 18. Pruebas no funcionales.	78

Lista de figuras

Figura 1. Módulos del MVC	38
Figura 2. Elementos de arquitectura de los patrones de diseño	39
Figura 3. Modelo cliente-servidor. Redes de computadoras	41
Figura 4. Modelo en cascada.	44
Figura 5. Modelo lógico	49
Figura 6. Modelo de clases.....	55
Figura 7. Diagrama de flujo- Consulta de diagnóstico.....	56
Figura 8. Diagrama de flujo- Nuevo diagnóstico.....	56
Figura 9. Diagrama de flujo- consulta otro especialista.....	56
Figura 10. Caso de uso: Autenticación.....	57
Figura 11. Caso de uso: Gestión de usuarios	57
Figura 12. Caso de uso: consultar diagnóstico.....	58
Figura 13. Caso de uso: Crear diagnóstico.....	58
Figura 14. Diagrama de secuencia	59
Figura 15. Inicio de sesión.....	61
Figura 16. Olvido contraseña	61
Figura 17. Menú de Contenido.....	62
Figura 18. Noticias interés plataforma	62
Figura 19. Ventana_emergente_noticias	62
Figura 20. Filtro avanzado home.....	63
Figura 21 Filtro avanzado home.....	63
Figura 22. Vista crear diagnóstico.....	64
Figura 23 Crear diagnostico	64

Figura 24. Edición de diagnostico.....	65
Figura 25. Eliminar diagnostico	65
Figura 26. Mensaje_eliminar.....	66
Figura 27. Inhabilitar diagnostico	66
Figura 28. Mensaje inhabilitar	67
Figura 29. Sección consultar	67
Figura 30. Nueva solicitud	68
Figura 31. Mis solicitudes	68
Figura 32. Mis consultas	69
Figura 33. Principales diagnósticos.....	69
Figura 34. Configuración noticias.....	70
Figura 35. Noticias	70
Figura 36. Usuarios	71
Figura 37. Crear y asigna	71
Figura 38. sql_consultas	72

Introducción

Las soluciones de software para la gestión de información ha sido una de las tendencias en diferentes áreas y ámbitos, especialmente a nivel corporativo, industrial y empresarial, sin embargo, esta misma digitalización ha permitido evidenciar la importancia del uso de software para gestionar y sistematizar diferentes formas de datos almacenados de forma física y digital, una tendencia en la actualidad. Los diferentes procesos de analizar la información de manera rápida y efectiva que permita su actualización y procesamiento, adicionalmente el uso de este tipo de soluciones informáticas ha ayudado en el proceso de aprendizaje en los usuarios de la disciplina de la medicina y ciencias de la salud, gracias a que brinda la oportunidad de realizar una revisión y actualización de cada una de las versiones del CIE, por medio del análisis y consolidación.

Con la elaboración de esta aplicación se pretende beneficiar al personal médico optimizando el uso del código CIE 10 que es el código Internacional de Enfermedades elaborado por la Organización Mundial de la Salud, tiene como propósito permitir el registro sistemático, análisis, interpretación, y comparación de los datos de mortalidad y morbilidad recolectados en diferentes países o áreas, en diferentes épocas. Se utiliza para convertir los términos diagnósticos y otros problemas de salud, de palabras a códigos alfanuméricos esto permiten su fácil almacenamiento y posterior recuperación para el análisis de la información, este documento es un archivo PDF, por tal motivo es tedioso para el médico consultarlo, se pretende con esta aplicación que el personal médico pueda agilizar sus procesos a la hora de diagnosticar una enfermedad en específico y pueda consultar los diagnósticos de otros especialistas para apoyar su propio diagnóstico.

Planteamiento del problema

En el transcurso del día son miles las consultas médicas en los centros hospitalarios sobre síntomas y enfermedades a tratar, el tiempo de consulta para cada paciente es muy corto por la alta cantidad de personas que lo requieren, así es como el médico debe consultar en material bibliográfico según la patología del paciente, es allí donde la aplicación crearía una disminución en el tiempo de consulta, tomando en cuenta la codificación internacional ya existente de enfermedades para la rápida evaluación y diagnóstico de la misma, así como también información sobre pacientes con la mismas características y posibles soluciones más rápidas y tratamientos más eficientes.

Formulación de pregunta

¿Cómo Desarrollar una aplicación que permita la sistematización del código internación de enfermedades (CIE)?

Justificación

Analizando y observando el desarrollo actual de la tecnología y la tendencia a desaparición medios físicos como los son los libros, revistas todo lo relacionado con el papel y analizando que los procesos cada vez se automatizan más y que todo lo que las personas realizan en forma manual ahora se hacen más automáticos, por esta razón tenemos que hablar del El Código Internacional de Enfermedades (CIE 10), elaborado por la Organización Mundial de la Salud, tiene como propósito permitir el registro sistemático, análisis, interpretación, y comparación de los datos de mortalidad y morbilidad. Se utiliza para convertir los términos diagnósticos y otros problemas de salud, de palabras a códigos alfanuméricos que permiten su fácil almacenamiento y posterior recuperación para el análisis de la información. Este está en un archivo PDF el cual dificulta al personal médico por eso se busca desarrollar una herramienta que permita de forma ágil y eficaz para así consultar y gestionar sus diagnósticos rápida y eficientemente. También se ha tomado de base la necesidad de los especialistas y se ha consultado a algunos de ellos como a Diana Marcela Díaz, profesional he especialista en medicina que en su concepto menciona “la aplicación me ayudará a tramitar diferentes diagnósticos que las personas contraen en sus labores cotidianas y que a veces son difícil de identificar pues son demasiado códigos y a veces suelen confundirse por la similitud en sus síntomas”

Objetivos

Objetivo General

Desarrollar una herramienta de software que permita realizar consultas de los diferentes diagnósticos y medicamentos basados en el CIE10 esto con el fin de ayudar a los especialistas a crear y gestionar diagnósticos propios y este histórico pueda beneficiar a otros auditores médicos.

Objetivos Específicos

Analizar la información del CIE 10, y con esto elaborar una aplicación de fácil consulta.

Diseñar una aplicación para el beneficio del personal médico especializado que permita consultar y gestionar basado en el CIE10

Realizar las pruebas correspondientes que permitan mejorar y optimizar la aplicación.

Elaborar y documentar la información correspondiente

Marco de referencia

Las soluciones de software se refieren a programas o aplicaciones informáticas diseñadas para abordar problemas específicos o satisfacer necesidades particulares. Estas soluciones pueden variar desde aplicaciones simples hasta sistemas complejos, y se utilizan en una amplia variedad de industrias y contextos. La satisfacción de las necesidades de los usuarios seguirá dependiendo del software, ya sea que le permita utilizar herramientas que faciliten el trabajo o tareas dadas o software que permita editar y gestionar todo aquello que se pueda incluir en la tecnología informática. cuando se desarrolla software, se hace para cumplir ciertos requerimientos específicos, por esto, podemos encontrar diferentes herramientas que, sumadas entre sí, pueden cumplir con el conjunto de necesidades de un usuario de manera integral, pero esto supone generalmente mayores costos de licenciamiento, soporte o administración, así sean herramientas privativas o libres.

Para garantizar la sistematización del CIE en su versión 10, se plantea la necesidad de diseñar, desarrollar e implementar un sistema modular integral de gestión de la información allí registrada, donde se define iniciar con la creación de una aplicación de gestión de la clasificación de las enfermedades con sus diagnósticos y medicamentos respetando el principio establecido desde el inicio de la primera publicación física y digital. Finalmente, y teniendo en cuenta la reflexión de Richards (2015) que nos orienta acerca de los diferentes patrones de arquitectura en el desarrollo de software, se define para el desarrollo del proyecto, la arquitectura o patrón de diseño MVC, debido a que, para la intencionalidad y las características de este desarrollo, es la que mejor se adapta, inicialmente como base para diseñar la separación de las diferentes capas que componen la aplicación; esto es, la capa de interfaz, la capa de datos y la capa lógica.

No sobra mencionar que la Clasificación Internacional de Enfermedades (CIE) es un sistema estandarizado utilizado a nivel mundial para clasificar enfermedades y otros problemas de salud con el objetivo de facilitar la recopilación y análisis de información estadística. La décima revisión de esta clasificación, conocida como CIE-10, ha tenido una historia y evolución significativas en el tiempo.

Marco teórico

Evolución del software

Según Brooks (1995), La evolución del software ha estado impulsada por avances tecnológicos, cambios en necesidades comerciales y la búsqueda continua de métodos más eficientes y efectivos para desarrollar aplicaciones, a su vez ha sido un proceso fascinante que ha abarcado varias décadas y experimentados cambios significativos en términos de tecnología, enfoques de desarrollo y paradigmas de diseño. A continuación, se proporciona una visión general de las principales etapas en la evolución del software:

Década de 1940: Programación en Lenguaje de Máquina: Los primeros programas se escribían directamente en lenguaje de máquina. El hardware y el software estaban altamente acoplados.

Década de 1950: Ensamblador y Lenguajes de Programación Introducción de lenguajes de ensamblador para simplificar la programación. Desarrollo de los primeros lenguajes de programación de alto nivel como Fortran y LISP.

Década de 1960: Sistemas Operativos la aparición de sistemas operativos que gestionaban recursos y facilitaban la ejecución de programas. Desarrollo de lenguajes como COBOL y ALGOL.

Década de 1970: Microprocesadores y Sistemas Distribuidos La aparición de microprocesadores permitió la proliferación de computadoras personales. Desarrollo de sistemas distribuidos.

Década de 1980: Revolución en el Software de Aplicación Expansión de la industria del software de aplicación. Surgimiento de interfaces gráficas de usuario (GUI) y sistemas operativos como MS-DOS y Windows.

Década de 1990: Internet y Desarrollo Orientado a Objetos La proliferación de Internet cambió la forma en que se desarrollan y entregan aplicaciones. El desarrollo orientado a objetos se volvió prominente con lenguajes como Java y C++.

Década de 2000: Desarrollo Ágil y Software como Servicio (SaaS) La metodología ágil ganó popularidad, enfocándose en la entrega incremental y la adaptabilidad. Auge de las aplicaciones basadas en la nube y el modelo de SaaS.

Década de 2010: DevOps y Auge de Tecnologías de Contenedores Integración y entrega continuas (CI/CD) se convirtieron en prácticas estándar. Tecnologías como Docker y Kubernetes facilitaron la implementación y gestión de aplicaciones.

Década de 2020: Enfoque en la Inteligencia Artificial y la Automatización Aumento en el uso de la inteligencia artificial y el aprendizaje automático en aplicaciones. Mayor énfasis en la automatización de procesos de desarrollo y operaciones (DevOps). Futuro: Computación Cuántica, Edge Computing y Nuevas Tecnologías Exploración de nuevas tecnologías como la computación cuántica y el cómputo en el borde (Edge Computing). Continuación de la evolución en respuesta a las demandas cambiantes del entorno tecnológico.

Ingeniería del software

Según Pressman (2014), La ingeniería de software es una disciplina formada por un conjunto de métodos, herramientas y técnicas que se utilizan en el desarrollo de programas informáticos, más conocidos como softwares. Para nosotros, este servicio trasciende a la programación, que es la base para crear una aplicación. La ingeniería de software engloba toda la gestión de un proyecto. Desde el análisis previo de la situación, el planteamiento del diseño hasta su implementación, pasando por las pruebas recurrentes para su correcto funcionamiento. Podríamos decir que la ingeniería del software es el continente donde se aloja el contenido, que sería el software en sí.

Etapas: Según Sommerville (2016),

Concepción. En esta primera fase se desarrolla el modelo de negocio. Es decir, conocemos las necesidades que debe de tener un software y empezamos a buscar las herramientas para cubrirlas.

Elaboración. Se detalla las características de la estructura del software.

Construcción. Tal y como su nombre indica en este paso empezamos a elaborar de forma tangible todo aquello que, de momento, solo hemos plasmado en forma de ideas.

Transición. Es el momento de la implementación y el desarrollo para los clientes o usuarios. Deben tener tiempo para familiarizarse con el nuevo software.

Fases de desarrollo,

Según McDonnell, S. (2004). Las fases de desarrollo son:

Análisis y requisitos: El cliente describe un problema que hay que resolver, lo que sirve de base para completar el resto de los requisitos del software. Los documentos de requisitos se generan a partir de los datos obtenidos del cliente y los usuarios (si es necesario) para garantizar que el equipo de desarrollo entiende lo que está creando.

Desarrollo: en la que los desarrolladores de software trabajan para hacer realidad su idea. se centran en la estructura de los datos e implementar las diferentes funcionalidades del a.

Pruebas y mantenimiento: Es la fase se centra en realizar un testeo a la aplicación para identificar fallas o errores que se presentan y realizar el correspondiente mantenimiento y dejar la aplicación 100% funcional.

Modelos para el diseño de software

Según Gamma et al. (1995), Los modelos de desarrollo de software, al fin y al cabo, consisten en la estructura y organización del proceso. Y es que, en resumidas cuentas, se necesita establecer unos flujos de trabajo claros para que un proyecto tenga el éxito esperado. Se aplican para todos los tipos de software.

Modelo lineal secuencial: Es el enfoque metodológico que ordena rigurosamente las etapas del proceso para el desarrollo de software, de tal forma que el inicio de cada etapa debe esperar a la finalización de la etapa anterior.

Modelo lineal secuencial: También conocido como desarrollo con prototiparían o modelo de desarrollo evolutivo, se inicia con la definición de los objetivos globales para el software, luego se identifican los requisitos conocidos y las áreas del esquema en donde es necesaria más definición. Este modelo se utiliza para dar al usuario una vista preliminar de parte del software. Este modelo es básicamente prueba y error ya que si al usuario no le gusta una parte del prototipo significa que la prueba fallo por lo cual se debe corregir el error que se tenga hasta que el usuario quede satisfecho.

Programación extrema: Como la programación extrema se centra en el desarrollo de software, suele ser implementada solamente por los equipos de ingeniería. Incluso los equipos de software suelen usarla únicamente para determinadas configuraciones.

Metodologías para el desarrollo de software

Modelos de desarrollo de software hay muchos, cada uno con sus propias ventajas y desventajas. Los más populares son los modelos ágiles, de los que hablaremos en las siguientes líneas.

Metodologías Ágiles: Según Cockburn (2002), Se enfocan en la iteración y el trabajo en equipo para el desarrollo de software, así como en la entrega de software de alta calidad en un plazo de tiempo más corto, entre otros principios ágiles. las principales metodologías ágiles son:

- Kanban.
- Scrum.
- Lean
- Programación extrema, también llamada XP.

Metodología Scrum. Schwab y Sutherland (2017), Es una metodología de desarrollo de software orientada a la entrega de productos de alta calidad a través de un proceso iterativo e incremental. En Scrum, un equipo de desarrollo se divide en equipos de trabajo autónomos que se encargan de desarrollar una parte del producto, cada equipo se reúne regularmente para revisar el progreso y planificar el trabajo futuro.

Metodología Cascada, Según Royce (1970), Es una forma de desarrollo de software en la que se siguen una serie de pasos estrictos y en orden. Esta metodología es muy detallada y deja poco margen de maniobra para el equipo de desarrollo. A menudo, se usa en proyectos grandes y complejos en los que es importante que todos los pasos se cumplan de forma estricta.

Metodología Prototipo Según Boehm (1988), Es una forma de desarrollar software en la que se crea un prototipo del software antes de comenzar el desarrollo completo. Esto permite que los desarrolladores obtengan una mejor comprensión de lo que el software debe hacer y cómo debe funcionar, lo que a su vez puede ayudar a reducir el tiempo y el costo del desarrollo.

Metodología Espiral. Según Boehm (1988), Se basa en un enfoque iterativo e incremental para el desarrollo de software. Se divide en cuatro fases: Inicio, Crecimiento, Madurez y Declive. Cada fase se divide en las siguientes subfases: Planificación, Análisis, Diseño, Implementación y Prueba. Sus principales características son.

Paradigmas en el desarrollo de software

Según Lethbridge y La gñere (2005), Para que una máquina o una computadora pueda realizar una tarea u operación, esta debe ser programada y cargada en su memoria, para lo que se debe definir un algoritmo apropiado, el que debe ser expresado en un lenguaje que la máquina pueda comprender, generalmente en un sistema binario.

Al enfoque del proceso de programación se le conoce como paradigma de programación, actualmente se reconocen principalmente los siguientes:

1. Paradigma imperativo, también llamado paradigma por procedimientos: Este es el más utilizado desde prácticamente los inicios de la programación y en la actualidad, está enfocado en las instrucciones y estrechamente ligado con la arquitectura de la máquina, por lo que puede ser implementado de manera eficiente, los principales lenguajes son C, Pascal y Basic.

2. Paradigma funcional: Se caracteriza por el uso de expresiones y funciones, en este paradigma un programa es un conjunto de funciones compuestas por funciones más simples, donde una función puede llamar a otra y el resultado de una función puede usarse como argumento de otra función, el lenguaje más reconocido de este paradigma es Lisp.

3. Paradigma declarativo, también conocido como paradigma de programación lógica: Basado en la idea de que un programa implementa relaciones antes que correspondencias, es decir que, las relaciones son más generales y fundamentales que las correspondencias, este paradigma de programación es de más alto nivel que la funcional y la imperativa, el lenguaje más reconocido es Prolog.

4. Paradigma orientado a objetos: Se basa en los conceptos de clases de objetos y objetos, donde un objeto puede ensamblarse con un conjunto de operaciones que se definen para estos o les pertenecen y una clase puede identificarse como un conjunto de objetos que comparten una serie de operaciones, este paradigma ha alcanzado gran popularidad a día de hoy y algunos de los lenguajes más reconocidos son Java y C++.

Boceto del software

Pressman, R. S. (2014). Saber cómo hacer un boceto de un sitio web o aplicación es vital para que el cliente sepa cuál será el resultado final, facilitar las idas y vueltas y prevenir inconvenientes. Si recién se comienza en el diseño web o se está pensando en renovar un sitio online, plasmar las ideas con anterioridad se pueden ahorrar horas de trabajo.

Inclusión de interfaces gráficas en el software

Pueden ser de distintos tipos, dependiendo del ámbito específico en que tengan lugar y de las estrategias que empleen para llevar a cabo su cometido, como en el caso de:

Interfaces de usuario: cuando sirven para comunicar al usuario de un sistema informático con este último. Por ejemplo: un teclado y un ratón constituyen una interfaz de usuario entre un PC y su usuario.

Interfaces de software: se refieren al funcionamiento específico de los programas informáticos y de la información virtual que “ocurre” o “tiene lugar” dentro del computador. Tal es el caso de las aplicaciones que empleamos a diario en nuestro trabajo con computadores.

Interfases de software-hardware: que se dedican a establecer un puente entre máquina y usuario, para “traducir” las instrucciones humanas al lenguaje del sistema y permitirle llevarlas a cabo exactamente, y al mismo tiempo “traducir” las respuestas del sistema del código binario a un lenguaje reconocible por el usuario.

Marco Conceptual

La producción de software es una ciencia compleja y utiliza una serie de criterios, estándares y fundamentos de la ingeniería de software para crear productos industriales y a su vez utiliza bases de la ingeniería como métodos, técnicas y herramientas para desarrollar productos innovadores regidos por metodologías y buenas prácticas, adicionalmente y para el desarrollo del proyecto actual, es necesario comprender ciertos conceptos claves que permiten identificar el público objetivo y sus necesidades.

Programación orientada a objetos

Según Booch (1994), Cuando se busca solucionar una problemática por medio de un software, una de las primeras decisiones que se deben tomar, es seleccionar tipo de modelo por medio del cual se va a codificar el aplicativo, en el caso de este proyecto, se define el uso del modelo de programación orientada a objetos, debido a que favorece el desarrollo e implementación de sistemas complejos, los sistemas desarrollados con este modelo de programación son construidos desde objetos autocontenidos que interactúan entre ellos, lo que genera una modularidad que favorece su diseño, implementación y mantenimiento.

Bases de datos relacionales

Según Date (2000), La tecnología moderna y más ampliamente utilizada para las bases de datos, es el modelo relacional, este modelo se encarga de tres aspectos de la información fundamentalmente; la estructura de datos, la manipulación de datos y la integridad de datos, para lo que es fundamental identificar los términos estructurales más importantes, como los atributos, la cardinalidad, el grado, el dominio, la clave primaria, y para operar los datos se usa el lenguaje estructurado SQL.

Aplicativo web

Smith, J. (2019). Las herramientas o el software que puede ser utilizado por los usuarios accediendo a un servidor web a través de internet o de una intranet por medio de un navegador, es decir, este tipo de software se codifica en uno o varios lenguajes que pueden ser interpretados por los navegadores web, los aplicativos web suponen una gran cantidad de ventajas, permite la centralización de los servicios y la información, puede ser ejecutado en múltiples plataformas, independientemente del sistema operativo, posee una gran facilidad para actualizar y mantener los aplicativos, en comparación con el software de escritorio y permite llegar a muchos más usuarios.

Versión de la CIE

Según Booch (1994), Las herramientas o el software que puede ser utilizado por los usuarios accediendo a un servidor web a través de internet o de una intranet por medio de un navegador, es decir, este tipo de software se codifica en uno o varios lenguajes que pueden ser interpretados por los navegadores web, los aplicativos web suponen una gran cantidad de ventajas, permite la centralización de los servicios y la información, puede ser ejecutado en múltiples plataformas, independientemente del sistema operativo, posee una gran facilidad para actualizar y mantener los aplicativos, en comparación con el software de escritorio y permite llegar a muchos más usuarios.

La primera versión de la CIE se desarrolló en el siglo XIX con el propósito de clasificar las causas de defunción. A lo largo del tiempo, se realizaron varias revisiones para abordar cambios en la comprensión médica y las necesidades estadísticas. CIE-9: La novena revisión, fue publicada en la década de 1970 y se convirtió en un estándar internacional.

Esta versión se centró en la codificación de enfermedades y causas de muerte. El Desarrollo de la CIE-10 genera la necesidad de una clasificación más detallada y actualizada que lleva al desarrollo de esta versión junto con la OMS en la década de los 80.

La versión final revisada de la CIE-10 fue publicada por la OMS en 1992. Su implementación comenzó en algunos países poco después de su lanzamiento.

La CIE-10 presenta una estructura jerárquica que permite clasificar las enfermedades desde niveles generales hasta niveles más específicos. Cada categoría tiene un código alfanumérico que indica su ubicación en la clasificación. La CIE-10 se convirtió en el estándar internacional para la codificación de enfermedades y problemas de salud. Se utiliza en ámbitos clínicos, de investigación y estadísticos en todo el mundo. Teniendo en cuenta la estructura, se diseña la aplicación que

permita su acceso rápido a partir de la búsqueda de la información médica y su complementación por parte del cuerpo médico y organismos de salud y académicos.

Algunos países han desarrollado adaptaciones específicas para satisfacer sus necesidades locales lo cual permite trazar un camino para versiones posteriores del software. Por ejemplo, en los Estados Unidos, se utiliza la CIE-10-CM (Clínica Modificación) para codificar enfermedades en el sistema de salud estadounidense.

En (2022), la OMS lanzó la Clasificación Internacional de Enfermedades, décima edición (CIE-11), como la versión más reciente. La CIE-11 presenta mejoras y actualizaciones con respecto a la CIE-10, reflejando avances en la comprensión médica y las necesidades actuales.

En resumen, la CIE-10 ha tenido un impacto significativo en la clasificación y codificación de enfermedades a nivel mundial desde su introducción en 1992. Su evolución a lo largo del tiempo refleja la necesidad constante de adaptarse a los avances médicos y las cambiantes demandas de la salud global.

Diagnóstico

Según López et al. (2020), es el Proceso en el que se identifica una enfermedad, afección o lesión por sus signos y síntomas. Para ayudar a hacer un diagnóstico, se pueden utilizar los antecedentes de salud o realizar un examen físico y pruebas, como análisis de sangre, pruebas con imágenes y biopsias.

Procesos administrativos

El proceso administrativo es el flujo continuo e interrelacionado de las actividades de planeación, organización, dirección y control, desarrolladas con el objetivo de aprovechar los recursos con los que cuenta la organización para hacerla efectiva

Plataforma

El proceso administrativo es el flujo continuo e interrelacionado de las actividades de planeación, organización, dirección y control, desarrolladas con el objetivo de aprovechar los recursos con los que cuenta la organización para hacerla efectiva

Marco Contextual

Torres-Pereira et al. (2020), las TIC es el elemento esencial de la transformación digital, además de innovador, sumamente necesario para evolucionar y adaptarse al mercado en prácticamente cualquier sector económico, especialmente en el sector de la salud, donde los paradigmas se han modificado y la manera de diagnosticar enfermedades y curarlas ha evolucionado a través de los últimos años, esto a su vez permite explotar diferentes ventajas que pueden beneficiar a los centros de formación de profesionales de la salud y los sistemas de atención primaria y especializada.

Hay pocas empresas orientadas al desarrollo de soluciones de software para identificar enfermedades y su diagnóstico a partir de sistemas estandarizados como el CIE. Parcialmente muchas de estas necesidades, se soportan en instrumentos físicos, sin embargo, reconocer la estructura y generar una herramienta que garantice su consulta y la posibilidad de aumentar información que los profesionales de la salud no son muy conocidas en la actualidad, tienden a manejar sistemas de consulta físico y carecen de la posibilidad de aumentar criterios de otros profesionales de la sal

Delimitaciones

La propuesta ha sido diseñada para ser implementada en los diferentes consultorios médicos ya que el CIE-10, existe pero forma digital en un archivo PDF que dificulta la consulta de los diagnóstico por su gran tamaño y complejidad y la idea es que la aplicación desarrollada necesite para su utilización tengan acceso a un dispositivo como lo es un computador, Tablet, smartphone y también deben contar con acceso a internet y pues lógicamente contar con los permisos correspondientes otorgados por el administrador de la aplicación.

Delimitación Cognitiva

La propuesta ha sido diseñada para ser implementada en los diferentes consultorios médicos ya que el CIE-10, existe pero forma digital en un archivo PDF que dificulta la consulta de los diagnóstico por su gran tamaño y complejidad y la idea es que la aplicación desarrollada necesite para su utilización tengan acceso a un dispositivo como lo es un computador, Tablet, smartphone y también deben contar con acceso a internet y pues lógicamente contar con los permisos correspondientes otorgados por el administrador de la aplicación.

Delimitación Temporal}

Para el desarrollo de la aplicación se estipulo un tiempo de 4 meses el cual fue suficiente para la implementación y desarrollo.

Tabla 1

Cronograma de actividades

Cronograma de actividades					
Actividad	Mes 1	Mes 2	Mes 3	Mes 4	Mes 5
análisis y recolección de información	■				
Desarrollo de bases de datos y estructura aplicación		■			
Diseño gráfico		■			
Desarrollo del Bachead			■		
Desarrollo Front end y pruebas			■		
Optimizar BD			■		
implementación seguridad hash			■		
Pruebas Funcionales				■	
Entrega y desarrollo de manuales				■	
Sustentación					■

Fuente: (Autor:)

Delimitación Financiera

Para el desarrollo del proyecto se define un presupuesto inicial.

Tabla 2
Presupuesto inicial.

Recurso	Descripción	Presupuesto
Equipo humano	1 desarrollador	\$2.500.000 (sueldo promedio de un desarrollador el cual asumiré)
Equipos y software	1 computador/ Visual conde XAMPP/	\$1.000.000 (se cuanta con todas las herramientas)
Viajes y salidas de campo	No cumple	
Materiales y suministros	No cumple	
Bibliografía	https://aprende-web.net/php/php1_4.php	
TOTAL: \$ 3.500.000		

Fuente: (Autor:) Nota: Referente a l presupuesto pactado para la implementación de la aplicación

Metodología

Según Jones y Matsumoto (2011), en la actualidad la metodología ágil scrum, permite abordar proyectos complejos desarrollados en entornos dinámicos y cambiantes de un modo flexible. Está basada en entregas parciales y regulares del producto final en base al valor que ofrecen a los clientes. Dicho en otras palabras: Scrum sirve para mejorar el trabajo colaborativo entre equipos.

Se trata de una metodología que ayuda a los equipos a aprender y organizarse en base a las experiencias a la vez que aborda problemas e invita a reflexionar sobre los éxitos y fracasos, todo ello bajo una serie de herramientas y recursos que permite a los equipos organizarse con mayor agilidad. Es por eso por lo que es de mucha utilidad actualmente para negocio esta permite mediante un ciclo o tiempo durante el que se desarrolla cada una de las tareas. Durante cada ciclo la cual tendrá un tiempo determinado para desarrollo estas actividades tendrán A modo de seguimiento del trabajo realizado durante la jornada específica, para así verificar el cumplimiento de los objetivos diarios o semanales esto con el fin de asegurar la correcta planificación de la siguiente jornada de trabajo o el siguiente paso en la ejecución del proyecto.

De acuerdo a lo anterior decidí utilizar esta metodología para implementación de este proyecto ya que es una metodología que se utiliza en la actualidad y es útil por su velocidad referente al entorno de negocio, y si miramos el CIE-10 está en constante cambio y esto es fundamental en la aplicación de esta aplicación, también es muy importante resaltar los beneficios por su adaptación a los cambios actuales y su gestión ágil por medio de fases que permiten solventar de una manera organizada los trabajos pendientes y puede estimar el tiempo en la ejecución de las tareas, también es útil para identificar errores o fallas al momento de la ejecución y poder solucionarlo de manera ágil y veloz

Arquitectura De Software Modelo Vista Controlador

MVC, Según Gamma et al. (1995), (Modelo-Vista-Controlador) es un patrón en el diseño de software comúnmente utilizado para implementar interfaces de usuario, datos y lógica de control. Enfatiza una separación entre la lógica de negocios y su visualización. Esta "separación de preocupaciones" proporciona una mejor división del trabajo y una mejora de mantenimiento. Algunos otros patrones de diseño se basan en MVC, como MVVM (Modelo-Vista-modelo de vista), MVP (Modelo-Vista-Presentador) y MVW (Modelo-Vista-Whatever)

Las tres partes del patrón de diseño

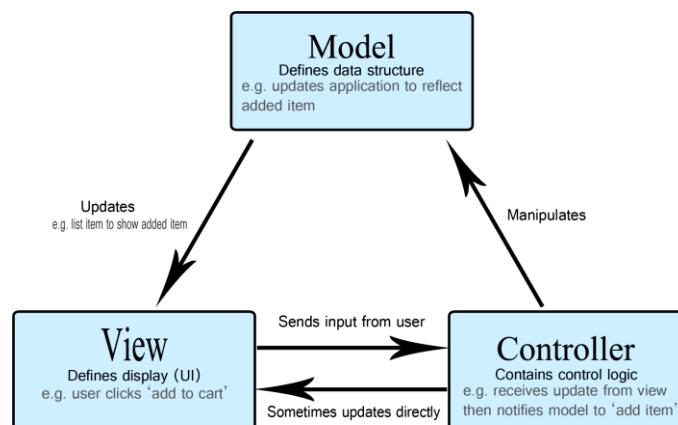
Modelo: Maneja datos y lógica de negocios.

Vista: Se encarga del diseño y presentación.

Controlador: Enruta comandos a los modelos y vistas.

Figura 1.

Módulos del MVC



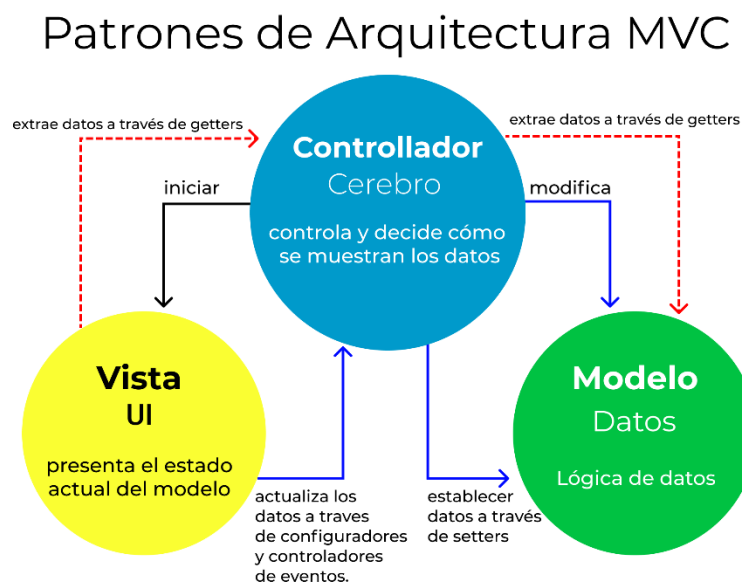
Fuente: (Mdn web docs.2017)

Patrones de diseño

Según Gamma et al. (1995), Los patrones de diseño son modelos muestra que sirven como guía para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos del diseño de interacción o interfaces, describen soluciones adecuadas a problemas que ocurren repetidamente, haciendo al diseño más flexible, elegante y principalmente reusable. Esta solución puede ser aplicada repetidamente para producir estructuras de diseño que lucen de forma similar al desarrollar aplicaciones diferentes. cada patrón se enfoca sobre un problema de diseño orientado a un objeto particular.

Figura 2

Elementos de arquitectura de los patrones de diseño



Fuente (Rafael Hernandez,2021)

Tecnologías Web

Según Sebasta (2014), Básicamente son aquellas que nos permiten crear las interfaces de usuarios y esto para poder establecer la comunicación un servidor que está básicamente basado en HTML, CSS y java Script. La tecnología web ha ido evolucionando en las últimas décadas y cabe mencionar que existen multitudes de aplicaciones, Frameworks, arquitecturas, librerías y sistemas de publicación claro está muchas diferentes versiones que con el tiempo reciben cambios.

Arquitectura

Segun Bass, Clements y Kazman (2012), son:

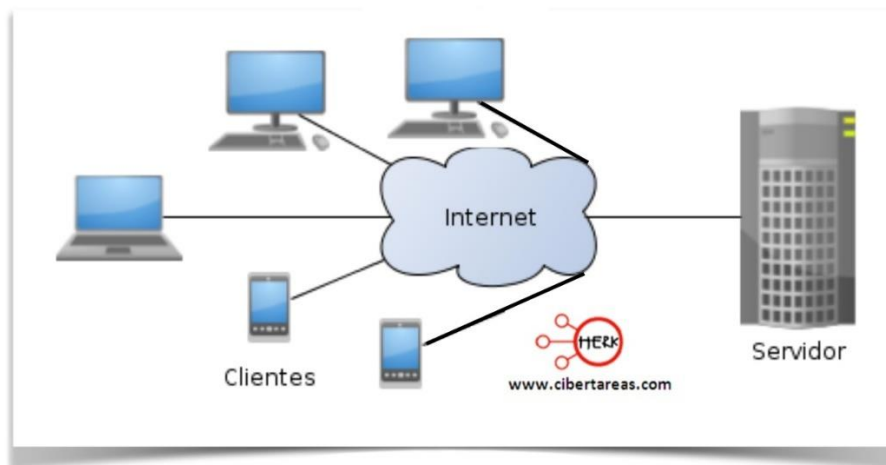
Navegador: Usa el usuario al momento de poder introducir una búsqueda o simplemente conocer la URL del sitio web.

Servidor: La parte más importante ya que es en aquel dónde se almacena de forma organizada la estructura de la información del sitio web esto para poder servir los contenidos en relación con las búsquedas que hacen las personas en el navegador.

HTML: Básicamente es un formato de documentos que van a componer la página web y está basada en etiquetas y al mismo tiempo sirve para estructurar la forma de poder mostrar el contenido de las páginas al usuario al momento de introducir una búsqueda o simplemente conocer la URL del sitio web.

Figura 3

Modelo cliente-servidor. Redes de computadoras



Fuente (Cibertareas.com,2014)

Seguridad informática

Según Stallings y Brown (2017), la seguridad informática también llamada ciberseguridad— se refiere a la protección de la información y, especialmente, al procesamiento que se hace de la misma, con el objetivo de evitar la manipulación de datos y procesos por personas no autorizadas. Su principal finalidad es que tanto personas como equipos tecnológicos y datos estén protegidos contra daños y amenazas hechas por terceros

Es por esto por lo que esta disciplina del área de la informática encargada de la protección de la privacidad de datos dentro de los sistemas informáticos se ha convertido en una parte indispensable para los negocios y la operación de las empresas.

Está claro que no existen sistemas 100% infalibles, por lo que las organizaciones que se comunican a través del mundo digital deben buscar los mecanismos oportunos para garantizar la

seguridad de sus datos, a través de alguno de los tipos de seguridad informática que existen y que deberían implementar en sus organismos.

Variables

Las variables que tenemos en este proyecto es el número de códigos diagnósticos que están plasmados en el CIE-10

Universo

Se toma como base 40 médicos especialistas, el cual indague sobre el uso de CIE-10 para la creación de los diagnósticos médicos

Población

Esta aplicación está orientada a toda la persona especialista en medicina específicamente los médicos

Muestra

De acuerdo con la investigación realizada, se tomó una muestra de 15 médicos especialistas que usan de modo continuo el Cie-10 y toman de base para crear y gestionar los diagnósticos médicos en sus pacientes

Diseño y desarrollo del software

Modelo de Desarrollo

Según Pressman, R. S. (2014). Los modelos de desarrollo de software ofrecen un marco de trabajo usado para controlar el proceso de desarrollo de sistemas de información, estos marcos de trabajo consisten en una filosofía de desarrollo de programas la cual debe de contar con las herramientas necesarias para la asistencia del proceso de desarrollo.

Modelo en cascada

Según Royce (1970), es el modelo en el cual se ordenan rigurosamente las etapas del desarrollo del software, de esto se obtiene que el inicio de una etapa de desarrollo deba de esperar el fin de la etapa anterior. De esto se obtiene que cualquier error detectado lleve al rediseño del área de código afectado, lo cual aumenta de costo el desarrollo del proyecto.

Modelo en prototipos

Según Boehm (1988), Pertenece a los modelos evolutivos, en el cual el prototipo debe de ser construido rápidamente y con la utilización escasa de recursos. El prototipo es construido para mostrárselo al cliente, obtener críticas y retroalimentación, con lo cual se obtendrán los requisitos específicos para la aplicación a partir de las metas graficas que son mostradas. Las etapas de este modelo son:

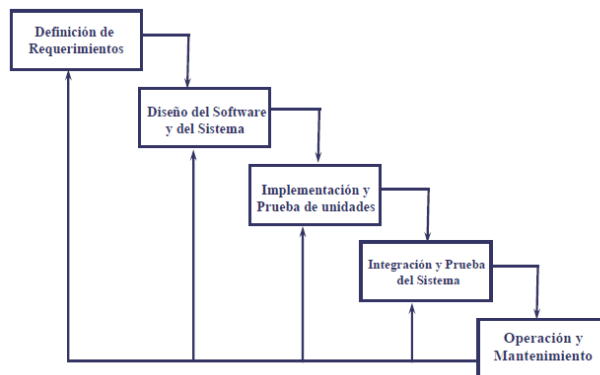
- Plan rápido
- Modelo
- Construcción del prototipo
- Entrega y retroalimentación
- Comunicación
- Entrega del desarrollo final

Modelo en espiral

Según Boehm, B. (1988), es el modelo en el cual las actividades se desarrollan en espiral, estas actividades se realizan conforme se van seleccionando de acuerdo con el análisis de riesgo. En cada iteración en este modelo, se deberán de tomar en cuenta los objetivos, las alternativas que se deberán de tomar de acuerdo con las características, estas son experiencia personal, requisitos a cumplir, las formas de gestión del sistema, entre otros. Este modelo tiene dos formas en las cuales se debe de planificar el proyecto, la forma angular, la cual indica únicamente el avance del software dentro del proyecto y la forma radial, la cual indica el aumento del costo dado que cada iteración conlleva más tiempo de desarrollo

Figura 4

Modelo en cascada.



Fuente: (Sites Google, Damián Gutiérrez,2011).

Análisis

Se escogió el modelo en cascada ya que primero se recopiló la información, se tuvo un análisis de la información recolectada referentes al CIE-10 luego se procedió a realizar un diseño y se después se hizo la implementación de la aplicación, finalmente se realizaron las pruebas correspondientes y el filtro A los diferentes diagnósticos y el último paso es solución a los errores identificados en las pruebas se realizó el correspondiente mantenimiento .

Identificación de actores

Se identifican los actores, aquellas personas que se requieren para desarrollar el proyecto desde su planeación hasta la puesta en marcha y comercialización, así mismo, se describen las funciones principales.

Tabla 3

Identificación Actores

	Levantamiento de información.
	Consolidación y análisis de información.
Director del proyecto	Realizar estudio de viabilidad.
	Planificación del proyecto.
	Definición de presupuesto.
	Definición de requisitos.
	Seguimiento y control en el desarrollo.
	Definición de estrategias

Diseñador gráfico	<p>Definición de paletas de colores corporativos.</p> <p>Creación de interfaz gráfica.</p> <p>Plantillas de diseño.</p>
Desarrollador	<p>Codificación del software.</p> <p>Creación e implementación de la base dedatos.</p> <p>documentación del software.</p> <p>Definir plan de pruebas y evaluación.</p> <p>Corrección de errores.</p> <p>despliegue del software.</p>
Administrador	<p>servicio post venta.</p> <p>Creación, modificación y eliminación de usuarios.</p> <p>Gestión de credenciales de acceso.</p> <p>Asignación de roles y permisos.</p> <p>Gestión del módulo.</p> <p>Mantenimiento base de datos.</p>
Usuario final	<p>Uso de las herramientas incorporadas en el software de acuerdo con el rol designado.</p> <p>Responsabilizarse del uso de las credencialesde acceso.</p>

Fuente: (Autor)

Identificación de roles y permiso

Se identifican los roles y se definen los permisos requeridos para cada tipo de usuario, de acuerdo con el rol que desempeña cada uno en la institución.

Tabla 4

Roles y permisos para el usuario final

Rol	Permisos y accesos
Administrador	Posee permisos absolutos en el software, puede modificar toda la información de la base de datos, crear, modificar, eliminar buscar y asignar permisos, roles y credenciales de acceso.
Auditor	Posee permisos limitados crear diagnósticos, buscar filtrar diagnósticos, ver los diagnósticos de todos los usuarios
Medico	Posee permisos limitados crear diagnósticos, buscar filtrar diagnósticos, ver los diagnósticos de los demás usuarios.
Usuario consulta	Posee permisos limitados, puede solo consultar información

Fuente: (Autor), **Nota:** Identifica los roles y permisos dentro de la aplicación

Identificación de requerimientos

Se identificaron los requerimientos funcionales y no funcionales con los que debe cumplir el software para garantizar la satisfacción de las necesidades propias del proceso.

Tabla 5.

Requerimientos funcionales y no funcionales

<i>REQ. FUNCIONALES</i>	<i>REQ. NO FUNCIONALES</i>
Cuenta con interfaz gráfica de usuario.	Seguro El sistema debe estar está protegido contra el acceso no autorizado.
Cuenta con un menú de opciones que permite al usuario navegar entre las herramientas según los permisos que posea	Fácil uso: El sistema es de fácil uso
Tiene herramientas de gestión y administración de credenciales.	Compatible: sistema puede ejecutarse en diferentes sistemas operativos y navegadores
Debe permitir filtrar los diagnósticos de los demás especialistas.	Fiabilidad: El sistema des confiable y cumplir con los requisitos del usuario.
Permite filtrar la información del cie 10 actualizado.	
Permite importar y actualizar el cie 10	

Fuente (Autor), Nota: Identifica los requerimiento funcionales y no funcionales en la aplicación

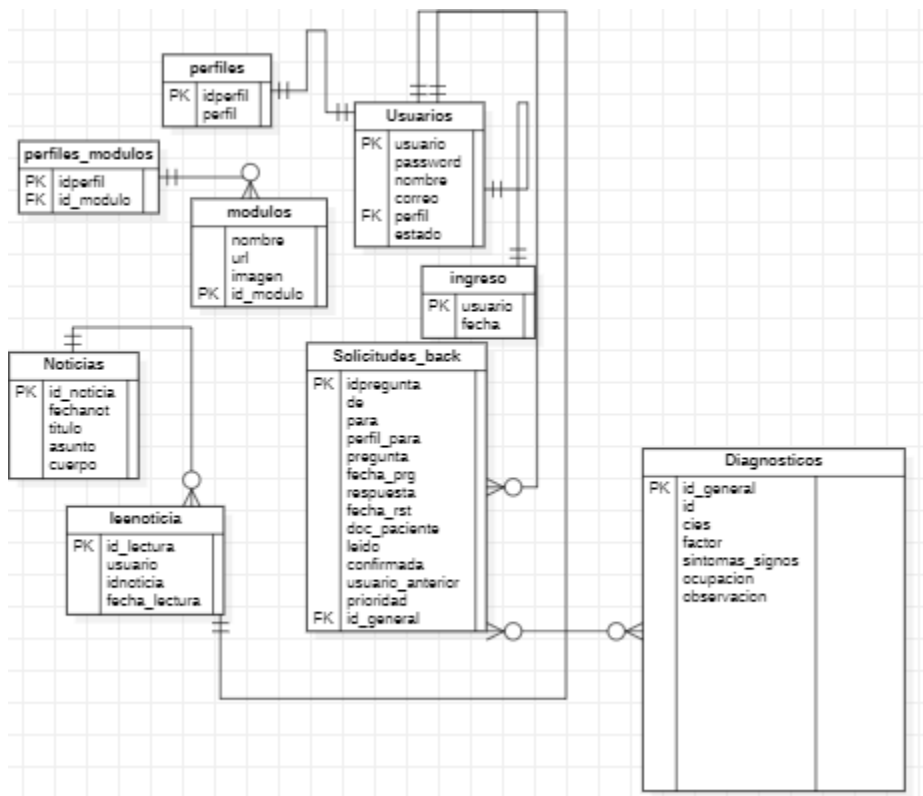
Diseño

Según Pressman, R. S. (2014). Teniendo en cuenta la fase anterior donde se implementó la codificación y funcionalidad se procede a crear un mapa de navegación y por ende un diseño de la aplicación teniendo en cuenta los patrones de diseño, colores corporativos imágenes fuentes e iconos correspondientes, luego se realiza la fusión con el código y se realizan las pruebas correspondientes.

Modelo lógico relacional

Figura 5

Modelo lógico



Fuente: (Autor, DB. 2022)

Diccionario de datos

Tabla 6

Perfiles

<i>campo</i>	<i>tamaño</i>	<i>tipo</i>	<i>Descripción</i>
<i>Idperfil</i>	20	<i>INT</i>	<i>Identificador tabla</i>
<i>Id_modulo</i>	20	<i>INT</i>	<i>Identificador tabla modulo</i>

Fuente :(Autor)

Tabla 7

Usuarios

<i>campo</i>	<i>tamaño</i>	<i>tipo</i>	<i>Descripción</i>
<i>usuario</i>	50	<i>varchar</i>	<i>Identificador tabla</i>
<i>password</i>	20	<i>varchar</i>	<i>clave</i>
<i>nombre</i>	100	<i>varchar</i>	<i>Nombre usuario</i>
<i>correo</i>	50	<i>varchar</i>	<i>Dirección electronica</i>
<i>perfil</i>	30	<i>varchar</i>	<i>Tipo de perfil</i>
<i>estado</i>	11	<i>int</i>	<i>Activo o inactivo</i>

Fuente :(Autor)

Tabla 8.

Diagnostico

<i>campo</i>	<i>tamaño</i>	<i>tipo</i>	<i>Descripción</i>
<i>Id_general</i>	20	<i>int</i>	<i>Identificador tabla</i>
<i>id</i>	20	<i>int</i>	<i>clave</i>
<i>cies</i>	500	<i>varchar</i>	<i>Nombre enfermedad</i>
<i>factor</i>	500	<i>varchar</i>	<i>Problemas de la enfermedad</i>
<i>Síntomas y signos</i>	30	<i>varchar</i>	<i>Probalidad enfermedad</i>
<i>ocupación</i>	11	<i>int</i>	<i>Aque se dedica</i>
<i>Observación</i>			<i>Observaciones del diagnostico</i>

Fuente :(Autor)

Tabla 9**Solicitudes Back**

<i>campo</i>	<i>tamaño</i>	<i>tipo</i>	<i>Descripción</i>
<i>Id_general</i>	20	<i>int</i>	<i>Identificador tabla</i>
<i>de</i>	50	<i>varchar</i>	<i>Identificador tabla</i>
<i>para</i>	20	<i>varchar</i>	<i>clave</i>
<i>Perfil_para</i>	100	<i>varchar</i>	<i>Nombre usuario</i>
<i>Pregunta</i>	50	<i>varchar</i>	<i>Dirección electronica</i>
<i>Fecha_prg</i>	30	<i>varchar</i>	<i>Tipo de perfil</i>
<i>respuesta</i>	11	<i>int</i>	<i>Activo o inactivo</i>
<i>Fecha_rpt</i>		<i>date</i>	<i>Fecha respuesta</i>
<i>Doc_paciente</i>	11	<i>int</i>	<i>Identificación paciente</i>
<i>leido</i>	11	<i>int</i>	<i>Lectura solicitud</i>
<i>confirmado</i>	11	<i>int</i>	<i>Si realizo gestion</i>
<i>Usuario_anterior</i>	50	<i>varchar</i>	<i>Usuario anterior que realizo proceso</i>
<i>prioridad</i>	11	11	<i>Si o no</i>

Fuente :(Autor)

Tabla 10

Noticias

campo	tamaño	tipo	Descripción
idnoticia	20	INT	Identificador tabla
Fecha_noticia	20	INT	fecha
titulo	50	varchar	Titulo noticia
asunto	50	varchar	Asunto noticia
cuerpo	500	varchar	Descripción de la noticia

Fuente :(Autor)

Tabla 11

Lee noticias

campo	tamaño	tipo	Descripción
Id_lectura	20	INT	Identificador tabla
usuario	20	INT	Identificador tabla modulo
idnoticias	11	int	Llave noticia
Fecha_lectura		date	Fecha lectura noticia

Fuente :(Autor)

Tabla 12

Módulos

campo	tamaño	tipo	Descripción
nombre	200	varchar	Nombre del modulo
url	80	varchar	Dirección modulo
imagen	200	varchar	Imagen o icono
Id_modulo	11	int	identificar

Fuente :(Autor)

Tabla 13

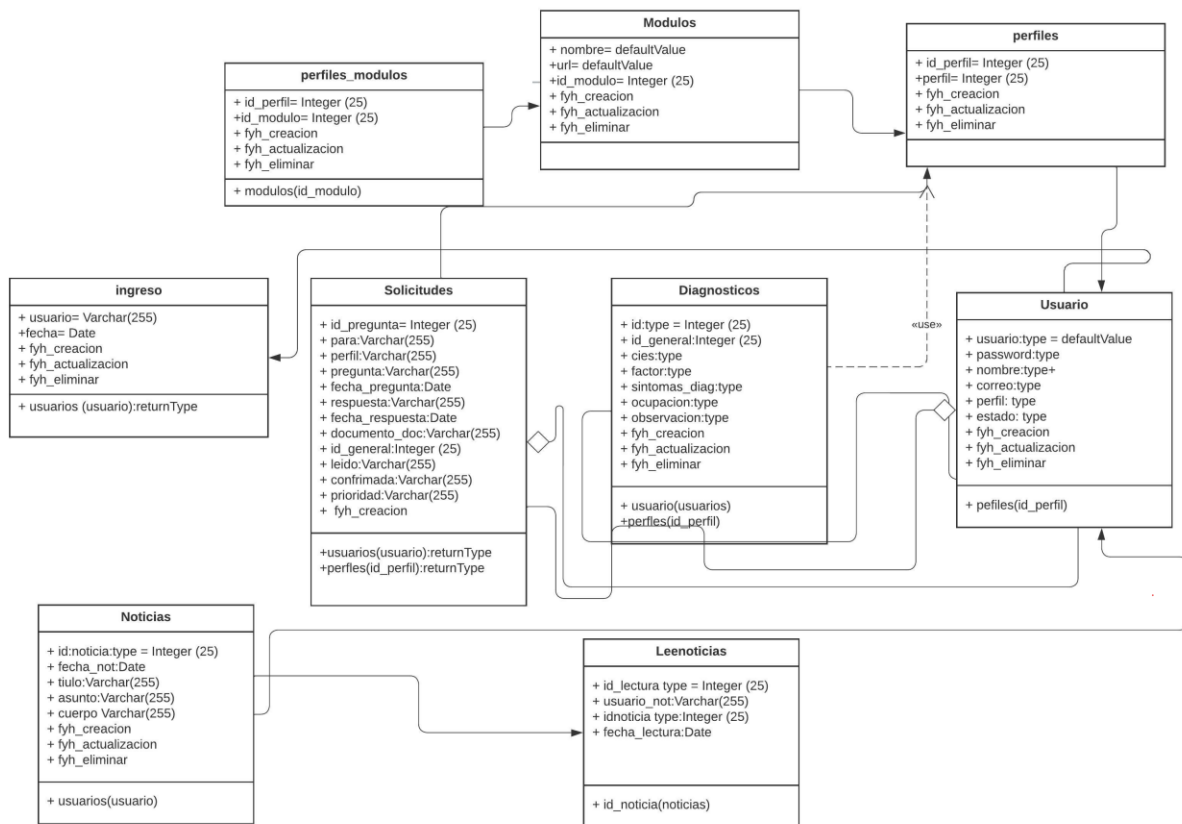
Ingreso

campo	tamaño	tipo	Descripción
usuario	100	varchar	Identificador tabla,nombre user
fecha		date	Fecha ingreso al sistema

Modelo de clases

Figura 6

Modelo de clases

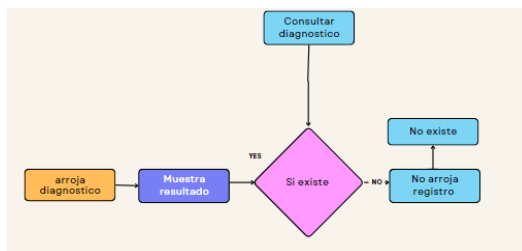


Fuente: (Autor)

Modelo de flujo de datos

Figura 7

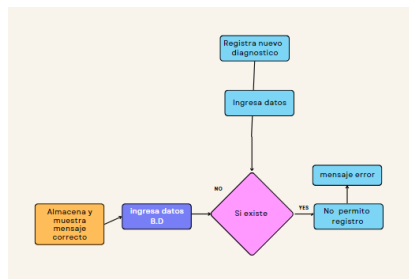
Diagrama de flujo- Consulta de diagnóstico



Fuente: (Autor)

Figura 8

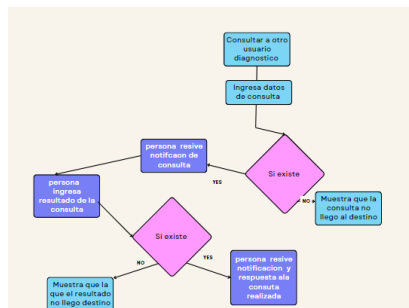
Diagrama de flujo- Nuevo diagnóstico



Fuente: (Autor)

Figura 9

Diagrama de flujo- consulta otro especialista

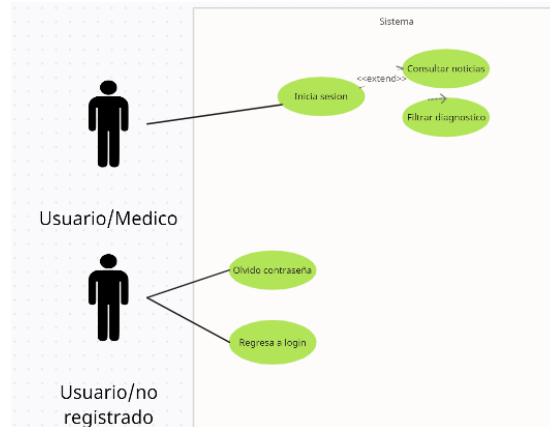


Fuente: (Autor)

Diagramas de caso de uso

Figura 10

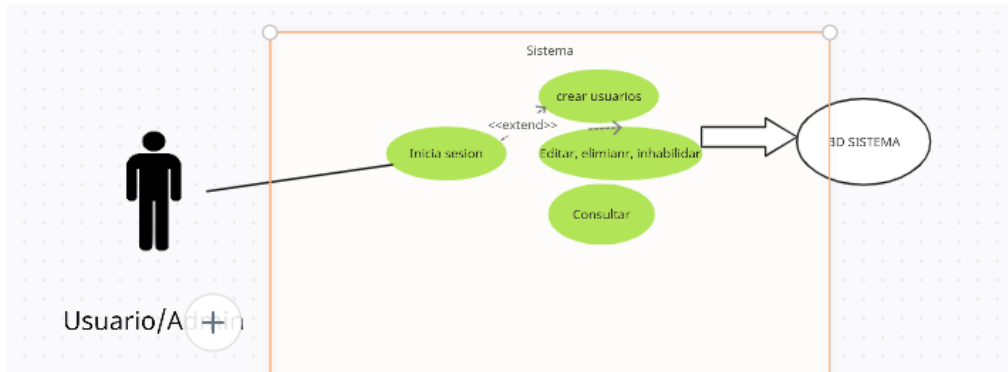
Caso de uso: Autenticación



Fuente: (Autor)

Figura 11

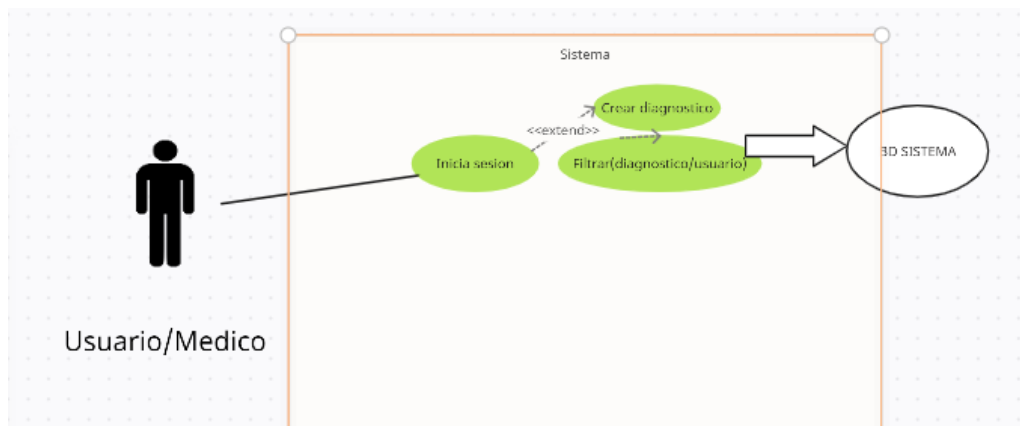
Caso de uso: Gestión de usuarios



Fuente: (Autor)

Figura 12

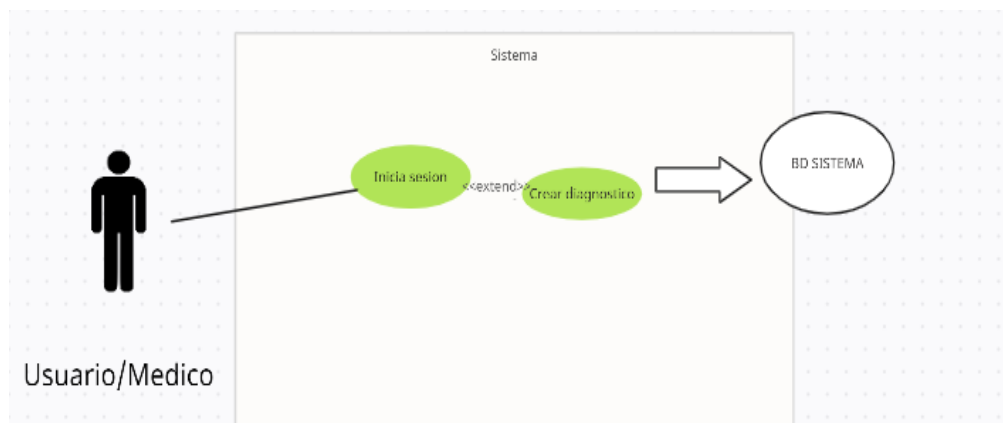
Caso de uso: consultar diagnóstico



Fuente: (Autor)

Figura 13

Caso de uso: Crear diagnóstico

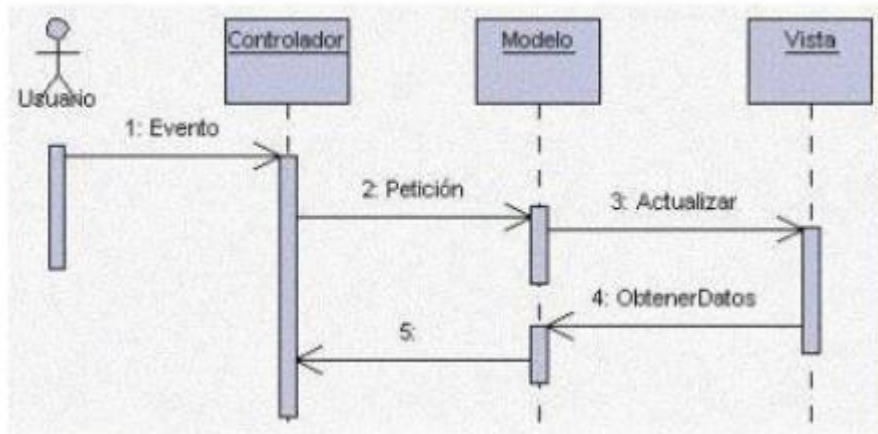


Fuente: (Autor)

Diagrama de secuencia

Figura 14

Diagrama de secuencia



Fuente: (Autor)

Implementación

En esta fase se realiza la codificación y el desarrollo del software, se toma la información obtenida en las etapas de análisis y diseño, y esta se traduce al lenguaje de programación definido, el proyecto se divide en diversos componentes que a su vez se desarrollan por separado y se integran paulatinamente, hasta tener un producto final que se procede a probar y evaluar, para garantizar que cumpla con los requerimientos acordados.

Para realizar la implementación, inicialmente se diseña un borrador o bosquejo de la interfaz y se define la interacción del usuario con ésta, para usar cada una de las herramientas definidas por el rol de cada usuario.

Módulo de inicio de sesión

donde se ingresan las credenciales de acceso y se validan con las credenciales que se encuentran definidas en la base de datos conectada al sistema e información

Figura 15

Inicio de sesión



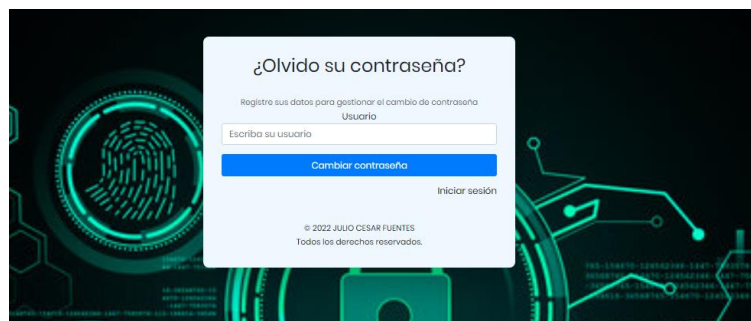
Fuente: (Autor)

Olvido contraseña

Aquí podremos colocar nuestro usuario y nos enviara un correo con un código para recuperar nuestra contraseña

Figura 16

Olvido contraseña



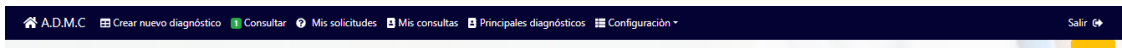
Fuente: (Autor)

Menús aplicación

Se muestran Los menús correspondientes que dispone la plataforma

Figura 17

Menú de Contenido



Fuente: (Autor)

Noticias

Se muestra el home inicial o pantalla principal donde dispone de un acceso a Noticias

Figura 18

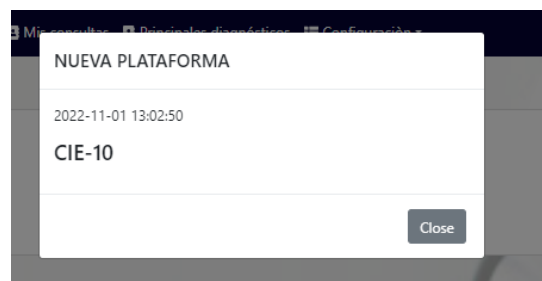
Noticias interés plataforma



Fuente: (Autor)

Figura 19

Ventana_emergente_noticias



Fuente: (Autor), **Nota.** Se muestra un panel de búsqueda avanzado con filtros

Filtro avanzado

Figura 20

Filtro avanzado home

id	Codigo Cie-10	Diagnostico	Titulo	Titulo - Nombre	Categoria
1	A00	Cólera	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
2	A00.0	Cólera debido a Vibrio cholerae 01, biotipo cholerae	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
3	A00.1	Cólera debido a Vibrio cholerae 01, biotipo El Tor	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
4	A00.9	Cólera, no especificado	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
5	A01	Fiebres tifoidea y paratifoidea	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
6	A01.0	Fiebre tifoidea	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
7	A01.00	Fiebre tifoidea, no especificada	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
8	A01.01	Meningitis tifoidea	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
9	A01.02	Fiebre tifoidea con afectación cardiaca	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
10	A01.03	Neumonía tifoidea	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES

Mostrando registros del 1 al 10 de un total de 99,867 registros

Anterior 1 2 3 4 5 ... 9987 Siguiente

Fuente: (Autor)

Crear diagnostico

Figura 21

Filtro avanzado home

id	Codigo Cie-10	Diagnostico	Titulo	Titulo - Nombre	Categoria
1	A00	Cólera	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
2	A00.0	Cólera debido a Vibrio cholerae 01, biotipo cholerae	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
3	A00.1	Cólera debido a Vibrio cholerae 01, biotipo El Tor	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
4	A00.9	Cólera, no especificado	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
5	A01	Fiebres tifoidea y paratifoidea	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
6	A01.0	Fiebre tifoidea	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
7	A01.00	Fiebre tifoidea, no especificada	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
8	A01.01	Meningitis tifoidea	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
9	A01.02	Fiebre tifoidea con afectación cardiaca	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES
10	A01.03	Neumonía tifoidea	CAPITULO 1	CIERTAS ENFERMEDADES INFECCIOSAS Y PARASITARIAS	ENFERMEDADES INFECCIOSAS INTESTINALES

Mostrando registros del 1 al 10 de un total de 99,867 registros

Anterior 1 2 3 4 5 ... 9987 Siguiente

Fuente: (Autor), Nota: Se muestra la pantalla donde se pueden modificar los datos de acceso del usuario.

Figura 22

Vista crear diagnóstico



Fuente: (Autor)

Botón Crear diagnóstico

Figura 23

Crear diagnóstico

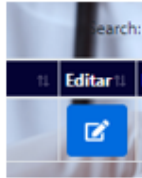
The image shows a form titled 'Crear Diagnóstico'. At the top, there is a blue button labeled 'Crear Diagnóstico'. Below the button, the form has the following fields: 'Nombre completo' (with a subtext 'Ingrese primero los apellidos seguidos de los no'), 'Documento de identidad' (with a subtext 'Ingrese su cédula'), 'Sexo' (a dropdown menu with 'Masculino' selected), 'Edad', 'Dirección', 'Teléfono', and 'Diagnóstico' (a large text area). At the bottom right, there are 'Cancelar' and 'Crear' buttons.

Fuente: (Autor)

Botón Editar

Figura 24

Edición de diagnostico

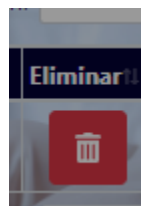
A screenshot of a web application window titled 'Actualizar diagnostico'. The form contains several input fields: 'Nombre' (julo fuentes), 'Documento de identidad' (110469), 'Edad' (80), 'Telefono' (0), 'Correo' (juliofuentes), 'Sexo' (Femenino), and 'Dirección' (calle6). There is also a large text area for 'Diagnostico' containing the word 'diagnostico'. At the bottom right, there are two buttons: 'Cerrar' and 'Guardar cambios'.

Fuente: (Autor)

Botón eliminar

Figura 25

Eliminar diagnostico

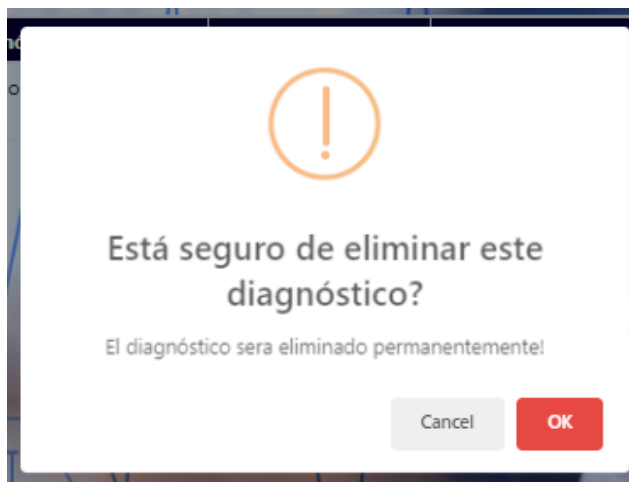


Fuente: (Autor)

Mensaje eliminar

Figura 26

Mensaje_eliminar



Fuente: (Autor)

Botón inhabilitar

Figura 27

Inhabilitar diagnostico

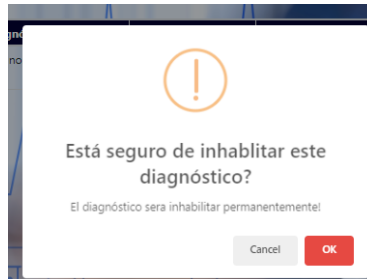


Fuente: (Autor)

Mensaje inhabilitar.

Figura 28

Mensaje inhabilitar



Fuente: (Autor)

Sección Consultar

Aquí podremos realizar preguntas a otros médicos o especialistas sobre alguna enfermedad o algún tema medico especifico.

Sección Consultar

aquí podremos realizar preguntas a otros médicos o especialistas sobre alguna enfermedad o algún tema medico especifico

Figura 29

Sección consultar



Fuente: (Autor)

Nueva solicitud

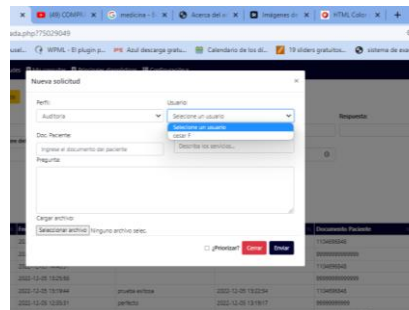
Muestra una pantalla emergente donde se pueden ingresar datos. Aquí podemos

Aquí podemos seleccionar el perfil, y el usuario a quien deseamos realizar

La pregunta en cuestión.

Figura 30

Nueva solicitud



Fuente: (Autor)

Sección Mis solicitudes

aquí podremos observar y revisar las preguntas que hemos realizado y estado en que se encuentran

Mis solicitudes

Figura 31

Mis solicitudes

Fuente: (Autor)

Mis Consultas

aquí podremos revisar las preguntas que nos han realizado a nosotros y estado en que se encuentran

Figura 32

Mis consultas

Showing 0 to 0 of 0 entries

De	Para	Pregunta	Fecha pregunta	Respuesta	Fecha respuesta	Documento del Paciente	Nombre del paciente	Archivo
No data available in table								

Showing 0 to 0 of 0 entries

Fuente: (Autor)

Principales diagnósticos

Aquí podremos ver y filtrar los principales diagnósticos médicos más comunes del especialista en salud y podemos buscar por cie-10, por la ocupación o el factor de riesgo para enfermedades comunes.

Figura 33

Principales diagnósticos

ID	Diagnóstico	CIE-10	Tipo	Estado	Operación	Operación
1	Asbestosis pulmonar	J61	Asbestosis	En fase progresiva	Asbestosis pulmonar	Operación de asbestosis pulmonar
2	Asbestosis pulmonar	J61	Asbestosis	En fase progresiva	Asbestosis pulmonar	Asbestosis
3	Asbestosis pulmonar	J61	Asbestosis	En fase progresiva	Asbestosis pulmonar	Operación de asbestosis pulmonar

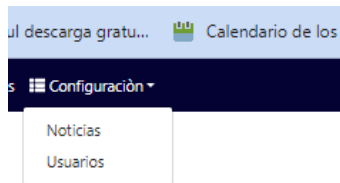
Fuente: (Autor)

Configuración

Aquí podremos administrar las noticias y usuarios y perfiles del sistema

Figura 34

Configuración noticias



Fuente: (Autor)

Administración de noticias

Aquí podemos crear, editar, eliminar y inhabilitar noticias

Figura 35

Noticias



Fuente: (Autor)

Administración de usuarios

Aquí podemos crear asignar perfil, editar, eliminar y inhabilitar usuarios.

Figura 36

Usuarios



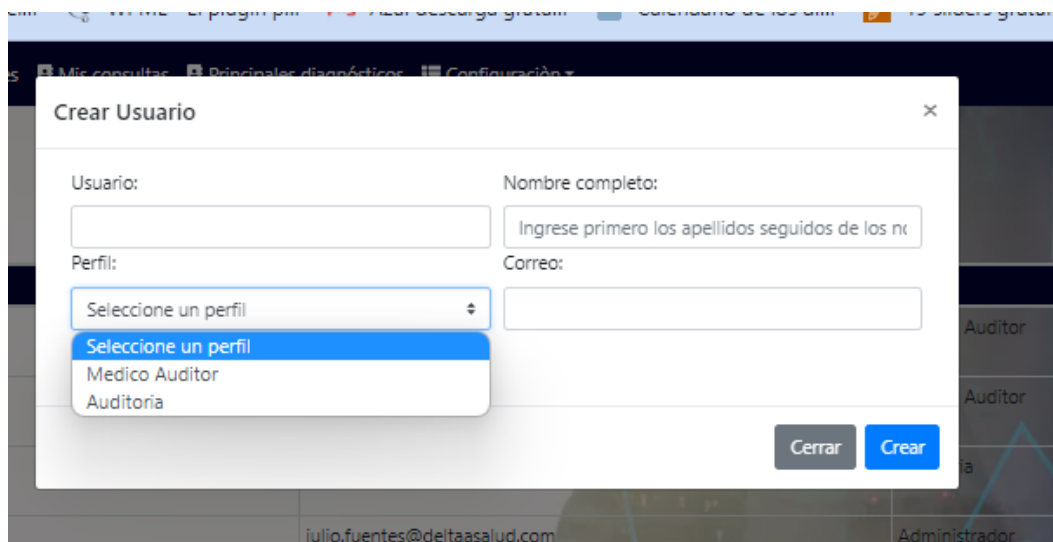
Usuario	Nombre y apellidos	Correo electrónico	Perfil	Editar	Eliminar	Inhabilitar
555	555	55555	Medico Auditor			
66666666	666666666666666666666666	66666	Medico Auditor			
cesar	cesar F	cesar@deltaasud.com	Auditoria			
cesar.f	JULIO CESAR FUENTES	julio.fuentes@deltaasud.com	Administrador			
lesy	lesy reyes martinez DE FUENTES	lesy.reyes@deltaasud.com	tecnologia			
prueba	andrea F	andrea@123	tecnologia			
quedouser	quedouser	quedouser@delta	Medico Auditor			

Fuente: (Autor)

Creación y asignación perfil.

Figura 37

Crear y asigna



Crear Usuario

Usuario:

Nombre completo:

Perfil: Seleccione un perfil

Correo:

Seleccione un perfil

Medico Auditor

Auditoria

Cerrar Crear

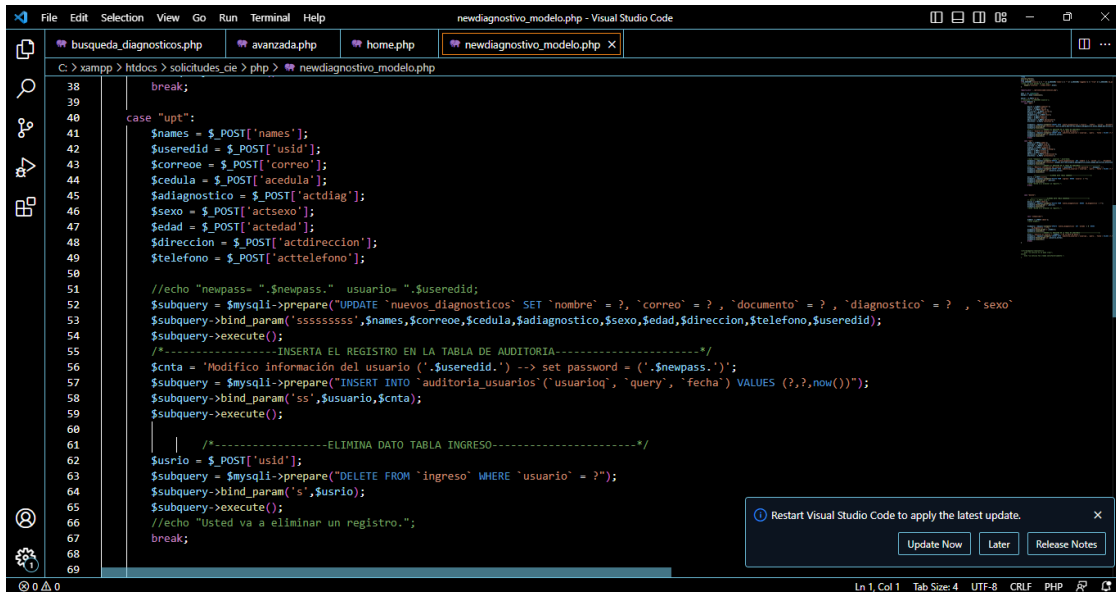
julio.fuentes@deltaasud.com Administrador

Fuente: (Autor)

Código fuente

Figura 38

sql_consultas



```
File Edit Selection View Go Run Terminal Help newdiagnostico_modelo.php - Visual Studio Code
busqueda_diagnosticos.php avanzada.php home.php newdiagnostico_modelo.php
C:\xampp\htdocs> solicitudes_cie > php > newdiagnostico_modelo.php
38 break;
39
40 case "upt":
41     $names = $_POST['names'];
42     $suseredid = $_POST['usid'];
43     $correeo = $_POST['correo'];
44     $cedula = $_POST['cedula'];
45     $adiagnostico = $_POST['actdiag'];
46     $sexo = $_POST['actsexo'];
47     $edad = $_POST['actedad'];
48     $direccion = $_POST['actdireccion'];
49     $telefono = $_POST['acttelefono'];
50
51     //echo "newpass= " . $newpass. " usuario= " . $suseredid;
52     $subquery = $mysqli->prepare("UPDATE `nuevos_diagnosticos` SET `nombre` = ?, `correo` = ?, `documento` = ?, `diagnostico` = ? , `sexo`
53     $subquery->bind_param('ssssssss', $names, $correeo, $cedula, $adiagnostico, $sexo, $edad, $direccion, $telefono, $suseredid);
54     $subquery->execute();
55     /*-----INSERTA EL REGISTRO EN LA TABLA DE AUDITORIA-----*/
56     $cnta = 'Modifico información del usuario (' . $suseredid. ') --> set password = (' . $newpass. ')';
57     $subquery = $mysqli->prepare("INSERT INTO `auditoria_usuarios` (`usuarioq`, `query`, `fecha`) VALUES (?, ?, now())");
58     $subquery->bind_param('ss', $susuario, $cnta);
59     $subquery->execute();
60
61     /*-----ELIMINA DATO TABLA INGRESO-----*/
62     $usrlo = $_POST['usid'];
63     $subquery = $mysqli->prepare("DELETE FROM `ingreso` WHERE `usuario` = ?");
64     $subquery->bind_param('s', $usrlo);
65     $subquery->execute();
66     //echo "Usted va a eliminar un registro.";
67     break;
68
69
```

Fuente: (Autor)

Pruebas

Según Myers (2011), es la etapa de software se integra con un entorno de pruebas con requerimientos de hardware similares a los que tendría en un ambiente de producción, se crean usuarios pruebas y se realizan las pruebas correspondientes de velocidad, agilidad la búsqueda, funcionalidad y estabilidad de la aplicación y después de esto se evalúa y se observa si la aplicación cumple con lo esperado y funciona de una manera óptima.

Tabla 14.

Pruebas funcionales.

Prueba	Resultado
Eficiencia y desempeño	Se realiza la prueba obteniendo por la gran cantidad de registros un poco de lentitud, pero debido a que los recursos están en un equipo básico al momento de ajustarlo a requerimientos mínimos se eliminara esa lentitud
Compatibilidad	Se realiza prueba tanto en equipos de escritorio con en dispositivos móviles o tables obteniendo, que la aplicación cuenta con adaptación requerida y se realizado pruebas en distintos navegadores donde la aplicación se ejecutó de manera normal.
Usabilidad	Se realizo un método de uso fácil y practico de usuario muy intuitivo donde, prima la accesibilidad fácil, la operatividad es muy sencilla y la protección frente a errores del usuario son fácil de identificar para el mismo.
Fiabilidad	Se realiza las pruebas correspondientes donde la única indisponibilidad que puede tener fallos sean la falta de conexión con el servidor para almacenar o consultar información, posee también una funcionalidad de un backus automático donde la información estará salvaguardado a diario

Seguridad

Se realizaron pruebas, la cual la aplicación respondió de manera efectiva donde se utilizó todo lo que se utiliza fue encriptación en contraseña y el método de recuperación es en un correo del usuario para evitar que ingresen a la aplicación de forma incorrecta, no tienen oportunidad de hacer inyección de código. También cada usuario tendrá opción de tener sus datos únicos si así lo desean por medio de perfiles personalizados.

Fuente: (Autor)

Matriz de evaluación

Se define una serie de elementos que permiten estructurar un plan de evaluación y ayudan a definir los lineamientos de la estrategia de mejoramiento continuo.

Tabla 15

Matriz de evaluación del software

Determinante	Descripción
Funcionalidad	Se evalúa de tal manera que cumpla con los requerimientos previstos y mencionados con anterioridad y se identifica que el software funcione muy versátil y ágil a la hora de realizar los diferentes procedimientos.
Confiabilidad	Se evalúa la capacidad y se realizan diferentes pruebas de sesiones de autenticación, encriptación y se trata de inyectar código para determinar que el software cumple con las medidas de seguridad adecuadas.
Usabilidad	Se evalúa la forma de navegabilidad y la facilidad que pueden tener los usuarios a la hora de usar la aplicación buscando la forma más sencilla y entendible de manejar los diferentes módulos.
Eficiencia	Se evalúa sobre todo la velocidad de búsqueda ya que cuenta con más de 1.000.000 de registros y se obtiene un resultado muy acorde a las necesidades.
Mantenibilidad	Se evaluar la forma de realizar los diferentes ajustes más para el usuario, que pueda realizar la modificación correspondiente de una manera sencilla y de ajustes de desarrollo se crea un manual correspondiente para realizar un mantenimiento sencillo.
Portabilidad	Se evalúa la capacidad que tiene la aplicación para adecuarlo a nuevas tecnologías adecuados a ambientes en la nube y con mejor infraestructura.

Fuente: (Autor)

Métricas de evaluación

Se define una serie de métricas que permiten evaluar las características fundamentales del software.

Tabla 16

Métricas: Identifica el nivel de medición de métrico de la aplicación

con la DB

Confiabilidad	Seguridad	Los protocolos y estándares de seguridad y autenticación son de alto nivel.
Confiabilidad	Disponibilidad	Alta tolerancia a fallos y facilidad de restauración.
Usabilidad	Intuitivo	Cuenta con una interfaz de usuario intuitiva, amigable y fácil de utilizar
Usabilidad	Aprendizaje	La curva de aprendizaje es baja.
Eficiencia	Recursos	Cada proceso del aplicativo debe hacer uso dinámico y optimizado de los recursos necesarios.
Eficiencia	Respuesta	Debe hacer uso solo de los recursos que necesita cada proceso, así los índices de eficiencia son altos.
Mantenibilidad	Preventivo	Permite la ejecución de rutinas de mantenimiento preventivo periódicas definidas.
Mantenibilidad	Correctivo	Facilita la corrección de incidencias o fallas.
Portabilidad	Homologación	Permite escalar y ser modificado para complementarse o reemplazar su código con nuevas instancias y tecnologías compatibles.
Portabilidad	Multiplataforma	Al tratarse de un aplicativo web, su ejecución debe garantizar en cualquier sistema operativo y navegador web.

Fuente: (Autor)

Plan de evaluación

Se define un plan de evaluación, en el cual se realizan diferentes pruebas con el fin de determinar y analizar las vulnerabilidades que se puedan presentar.

Pruebas funcionales

Tabla 17

Pruebas funcionales. Identifica las pruebas funcionales

<i>PRUEBAS FUNCIONALES</i>	<i>RESULTADOS</i>
<i>Prueba de seguridad y acceso, para impedir cualquier acceso sin el uso de las credenciales asignadas.</i>	<i>En cuanto se intenta acceder directamente a los módulos sin log, el sistema muestra una advertencia y redirige al usuario a la sección de autenticación.</i>
<i>Prueba de funcionamiento de todas las herramientas incorporadas en el software.</i>	<i>Se simula cada uno de los procesos dentro del sistema, encontrando funcionamiento óptimo. Se realizan filtros simultáneos y se llega a la conclusión de que no hay retardos muy amplios entre consultas, cada consulta tarda de 10 a 15 segundos.</i>
<i>Prueba de estabilidad y disponibilidad del software y la información.</i>	<i>El sistema genera un registro de errores, para consulta y análisis.</i>
<i>Prueba de reporte de errores, para identificar cuando se presente una incidencia.</i>	<i>Se intenta acceder a la base de datos y extraer información inyectando código y no se obtienen resultados.</i>
<i>Prueba de integridad y seguridad en los datos y la información de la base de datos a nivel institucional.</i>	
<i>Prueba de integración, para verificar que los componentes se encuentren integrados y en óptimo funcionamiento.</i>	<i>Se realiza prueba incorporando a un servidor en la nube y se ejecuta con éxito.</i>

Fuente: (Autor)

Pruebas no funcionales

Tabla 18

Pruebas no funcionales. Identifica las pruebas no funcionales

Pruebas no funcionales	Resultados
Prueba de carga, donde se evalúa el tiempo que tarda el software en cargarla interfaz, la base de datos y las herramientas en diferentes condiciones.	La carga de los recursos en servidor locales menor de 10 segundos ya que tarda un poco por la cantidad de registros en la base de datos
Prueba de estrés, donde se evalúa el funcionamiento del software en las condiciones de mayor exigencia.	Se realiza prueba de estrés, encontrando lentitud en la carga de los recursos ya que por la cantidad de diagnósticos almacenas y los pocos recursos locales tarda más o menos 20 segundos.
Prueba de escalabilidad, donde se evalúa la viabilidad de implementar y desplegar el software en escenarios de carga creciente de datos y trabajo.	Se realiza prueba en otro equipo con mayores capacidades y se reduce el tiempo de carga de los registros.
Prueba de portabilidad, donde se evalúa el comportamiento del software cuando se mueve a diferentes ambientes.	Se realiza el cambio a otros equipos sin obtener fallos en los ambientes ni fallasen los datos obtenidos de las bases de datos
Prueba de configuración, para verificar que todos los parámetros estén configurados correctamente	Se realizan pruebas con versiones superiores de servidore de base datos se verifica que tenía algunos errores se vuelven a exportar la base de datos y se verifica que funciona con normalidad.

Fuente: (Autor)

Conclusiones

Se implementó una aplicación que permite consultar diagnósticos médicos de manera eficiente.

Se desarrolló una aplicación que permite crear, modificar y consultar diagnósticos médicos para el beneficio del personal médico especializado.

Se realizaron las pruebas correspondientes para la optimización de la aplicación.

Se creó la correspondiente documentación (manuales)

Recomendaciones

Los diagnósticos se actualizan cada cierto tiempo se actualizan por esta razón se debe mantener esta información completamente actualizada.

Debemos realizar mantenimientos cada cierto tiempo para evitar retornos en las consultas, esto realiza optimizando la base de datos.

Mantener supervisado los servidores donde alojemos nuestra aplicación para evitar posibles Fallos por conexión.

Referencias

Planeación del proyecto. (2022).

https://drive.google.com/drive/folders/1_UF3B-eJQTEaW53a4KgDwe-nS3EYH6kl?usp=sharing o <https://youtu.be/rWL6s7SsuRo>

Mini salud. (2021). M. Interoperabilidad de Datos de la Historia Clínica en Colombia
Términos y siglas.

<https://www.minsalud.gov.co/ihc/Documentos%20compartidos/ABC-IHC.pdf>

Librería, con. (2017). Re qué es el proyecto aplicado dentro de la especialización en gestión de proyectos. <https://1library.co/document/yj74pwx5-proyecto-aplicado-dentro-especializacion-gesti%C3%B3n-proyectos.html/>

Claudia Ere (2018). Diseño Diferenciales

<https://www3.paho.org/relacsis/index.php/es/foros-relacsis/foro-becker-fci-oms/61-foros/consultas-becker/974-diferencias-con-cie-10-de-cie10-cm>

Maribel (2013). Diseño De Plantas Industriales

<https://es.slideshare.net/Mary2530/proyecto-aplicado-25659666/>.

Mes Lacre. (2006). Aproximaciones teóricas y algunos diagnósticos diferenciales aproximaciones teóricas y algunos diagnósticos diferenciales

https://www.sssalud.gob.ar/hospitales/archivos/cie_10_revi.pdf

Washington, D.C, E.U.A. (2003). Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud.

<https://ais.paho.org/classifications/chapters/pdf/volume2.pdf>

Monitor estratégico. (2005). Clasificación Internacional de Enfermedades (CIE),
<https://ais.paho.org/classifications/chapters/pdf/volume2.pdf>