

Sistema de Gestión para préstamo de recursos tecnológicos y espacios físicos para institución de educación superior.

Presentado por:

Hamilton López Villamizar

Universidad Nacional Abierta y a Distancia

Escuela de Ciencias Básicas, Tecnología e Ingeniería – ECBTI

Programa de Ingeniería de Sistemas

CEAD Tunja, Boyacá - 10/04/2025

## **Dedicatoria**

El desarrollo de este proyecto representa la culminación de un proceso o etapa importante en mi vida; Toda etapa cuenta con el acompañamiento constante de personas valiosas en nuestras vidas, que son claves para alcanzar las metas propuestas, como el llegar a ser un profesional integro. Por ello, dedico este logro, en primer lugar a Jehová Dios, quien me ha permitido tener capacidades para realizar lo alcanzado con este proyecto; También dedico este proyecto que culmine con tanto esfuerzo a mi prometida y futura esposa, por su apoyo incondicional, por creer en mi y en mis capacidades, esto me ha permitido mantener la motivación y fortaleza para continuar en busca de mis objetivos; Por ultimo también lo dedico a mis padres quienes siempre tuvieron fe en mi capacidades y me impulsaron siempre a ir mas halla.

## **Agradecimientos**

Deseo expresar mis más sinceros agradecimientos a todas las personas que me apoyaron durante este proceso, agradezco a la Universidad Nacional Abierta y a Distancia por brindarme la oportunidad de alcanzar mi sueño de ser profesional en el área de la ingeniería, esto gracias a la opción de poder trabajar de forma remota, permitiéndome así trabajar para solventar los gastos implicados en este proceso académico, además de ofrecerme la posibilidad de desarrollar este proyecto aplicado como opción de grado.

Agradezco a la ingeniera Yenny Stella Núñez Álvarez por su excelente acompañamiento en el desarrollo del proceso, sus aportes han sido fundamentales para mi crecimiento académico y profesional como futuro ingeniero.

También agradezco a la Universidad de Boyacá por permitirme ser parte de su equipo, así como por darme la oportunidad de aportar mis conocimientos en el apoyo de unos de sus muchos procesos, lo cual me ha permitido adquirir una valiosa experiencia en el campo profesional.

## Resumen

Diseño, desarrollo e implementación de un software de gestión de recursos tecnológicos y físicos para una institución de educación superior, con el objetivo de brindar acceso a estudiantes y funcionarios para solicitar dichos recursos en calidad de préstamo.

El sistema permite a los funcionarios con rol de administrador gestionar los recursos asignados a su propia dependencia, se diseña la interfaz gráfica, la arquitectura del sistema y la base de datos, así como el desarrollo completo del software, todo ajustado a los requerimientos del cliente o usuario final, cumpliendo con sus expectativas para su posterior despliegue en un entorno virtual.

***Palabras clave:*** Software, gestión, desarrollo web, programación, proyecto.

## **Abstract**

Design, development, and implementation of software for the management of technological and physical resources at a higher education institution, aimed at providing access to students and staff for requesting these resources on a loan basis.

The system allows staff members with administrative roles to manage the resources assigned to their respective departments. The graphical user interface, system architecture, and database are all designed, along with the complete software development, in accordance with the requirements of the client or end user, meeting their expectations for subsequent deployment in a virtual environment.

***Keywords:*** Software, management, web development, programming, project.

## Tabla de Contenido

Introducción.....	12
Planteamiento del problema .....	14
Justificación .....	16
Objetivos.....	19
Objetivo General.....	19
Objetivos Específicos .....	19
Marco de Referencia.....	20
Marco teórico .....	20
Marco conceptual .....	21
Marco Tecnológico.....	22
Marco Legal .....	24
Marco contextual.....	25
Metodología .....	27
Enfoque de Desarrollo .....	27
Herramientas y Tecnologías.....	33
Gestión de proyectos .....	35
Análisis de Requisitos.....	39
Perfiles de usuario y caracterización de actores .....	39
Requerimientos funcionales .....	40
Requerimientos no funcionales .....	45
Árbol de refinamiento funcional .....	47
Casos de uso .....	49
Diseño.....	59
Diseño Base de datos (MER) .....	59
Arquitectura del sistema.....	59
Diseño Interfaz grafica .....	61
Desarrollo .....	69
Desarrollo Back-End.....	69
Desarrollo Front-End.....	72
Base de datos.....	82

Pruebas .....	86
Despliegue del software .....	110
API REST – Base de datos .....	110
Front End .....	114
Referencias Bibliográficas .....	117

## Lista de tablas

<b>Tabla 1</b> Cronograma proyecto.....	36
<b>Tabla 2</b> Árbol de refinamiento funcional .....	47
<b>Tabla 3</b> Caso de Uso 1 .....	49
<b>Tabla 4</b> Case de Uso 2 .....	50
<b>Tabla 5</b> Caso de uso 3 .....	52
<b>Tabla 6</b> Case de Uso 4 .....	53
<b>Tabla 7</b> Caso de uso 5 .....	54
<b>Tabla 8</b> Caso de uso 6 .....	55
<b>Tabla 9</b> Caso de uso 7 .....	56
<b>Tabla 10</b> Case de uso 8 .....	57
<b>Tabla 11</b> Prueba acceso al modulo .....	86
<b>Tabla 12</b> Prueba acceso rol estudiante .....	88
<b>Tabla 13</b> Prueba autenticación con Google .....	90
<b>Tabla 14</b> Prueba registro de tipos de recurso.....	92
<b>Tabla 15</b> Prueba gestión estados reservas .....	94
<b>Tabla 16</b> Prueba registro de recursos .....	95
<b>Tabla 17</b> Prueba Registro reservas .....	99
<b>Tabla 18</b> Prueba gestión eventos.....	104
<b>Tabla 19</b> Prueba generación reportes .....	106

## Lista de Figuras

<b>Figura 1</b>	Fases, escala temporal y esfuerzo en un proyecto.....	35
<b>Figura 2</b>	Diagrama de caso de uso general.....	49
<b>Figura 3</b>	Caso de uso 1.....	50
<b>Figura 4</b>	<i>Caso de uso 2</i> .....	51
<b>Figura 5</b>	Caso de uso 3.....	53
<b>Figura 6</b>	Caso de uso 4.....	54
<b>Figura 7</b>	Caso de uso 5.....	55
<b>Figura 8</b>	Caso de uso 6.....	56
<b>Figura 9</b>	Caso de uso 7.....	57
<b>Figura 10</b>	Caso de uso 8.....	58
<b>Figura 11</b>	Modelo entidad relación (MER) .....	59
<b>Figura 12</b>	Arquitectura del sistema .....	61
<b>Figura 13</b>	Inicio de sesión .....	62
<b>Figura 14</b>	Gestión de reservas .....	63
<b>Figura 15</b>	Registro Reservas .....	63
<b>Figura 16</b>	Registro eventos .....	64
<b>Figura 17</b>	Panel Administrador .....	64
<b>Figura 18</b>	Gestión Recursos .....	65
<b>Figura 19</b>	Registro Recursos .....	65
<b>Figura 20</b>	Gestión Estados .....	66
<b>Figura 21</b>	Gestión tipos de Recursos.....	66
<b>Figura 22</b>	Gestión Novedades.....	67
<b>Figura 23</b>	Gestión de características .....	67
<b>Figura 24</b>	Gestión Reportes .....	68
<b>Figura 25</b>	Capa Conexión - API.....	69
<b>Figura 26</b>	Método conexión a base de datos .....	70
<b>Figura 27</b>	Capa servicios - API.....	70
<b>Figura 28</b>	Método consulta procedimientos .....	71
<b>Figura 29</b>	Capa API - API.....	71
<b>Figura 30</b>	End Point o controlador desde el que se conecta el Frontend .....	72
<b>Figura 31</b>	Estructura del proyecto .....	72
<b>Figura 32</b>	Módulos Front .....	73
<b>Figura 33</b>	Estructura de cada modulo.....	73
<b>Figura 34</b>	Rutas de modulo .....	74
<b>Figura 35</b>	Servicios .....	74
<b>Figura 36</b>	Interceptor .....	75
<b>Figura 37</b>	Inicio de sesión .....	76
<b>Figura 38</b>	Panel De Control .....	76
<b>Figura 39</b>	Gestión de recursos.....	77

<b>Figura 40</b>	Registro recursos parte 1 .....	77
<b>Figura 41</b>	Registro recurso parte 2 .....	78
<b>Figura 42</b>	Gestión de tipos de recurso .....	78
<b>Figura 43</b>	Gestión estado reservas.....	79
<b>Figura 44</b>	Gestión novedades.....	79
<b>Figura 45</b>	Gestión características recurso.....	80
<b>Figura 46</b>	Gestión reservas.....	80
<b>Figura 47</b>	Registrar reserva.....	81
<b>Figura 48</b>	Gestión recursos reserva .....	81
<b>Figura 49</b>	Gestión reportes.....	82
<b>Figura 50</b>	Registro de tabla.....	83
<b>Figura 51</b>	Tabla del esquema .....	83
<b>Figura 52</b>	Registro de “constraints” .....	83
<b>Figura 53</b>	Registro de secuencias .....	84
<b>Figura 54</b>	Listado de secuencias .....	84
<b>Figura 55</b>	Registro de paquete con procedimientos almacenados .....	85
<b>Figura 56</b>	Listado paquete con procedimientos almacenados .....	85
<b>Figura 57</b>	Evidencia prueba acceso funcionario.....	87
<b>Figura 58</b>	Evidencia prueba acceso funcionario administrativo .....	87
<b>Figura 59</b>	Evidencia prueba acceso rol funcionario .....	88
<b>Figura 60</b>	Evidencia acceso al modulo con el rol estudiante .....	89
<b>Figura 61</b>	Evidencia prueba acceso con rol estudiante II .....	90
<b>Figura 62</b>	Evidencia acceso mediante opción de Google .....	91
<b>Figura 63</b>	Evidencia prueba acceso opción Google II.....	92
<b>Figura 64</b>	Evidencia prueba acceso opción Google (administrativo sin registros) .....	92
<b>Figura 65</b>	Evidencia gestión de tipos de recursos.....	94
<b>Figura 66</b>	Evidencia prueba gestión estados de reserva. ....	95
<b>Figura 67</b>	Evidencia prueba registro recursos información Base .....	97
<b>Figura 68</b>	Evidencia prueba registro recurso – Registro características .....	97
<b>Figura 69</b>	Evidencia registro recursos – registro disponibilidad .....	98
<b>Figura 70</b>	Evidencia registro reservas respuesta .....	98
<b>Figura 71</b>	Evidencia registro recurso – Consulta registro creado .....	99
<b>Figura 72</b>	Evidencia prueba registro reserva rol funcionario o estudiante.....	100
<b>Figura 73</b>	Evidencia registro reserva rol funcionario o estudiante II.....	101
<b>Figura 74</b>	Evidencia registro reserva rol funcionario o estudiante III .....	101
<b>Figura 75</b>	Evidencia registro reserva rol funcionario o estudiante III .....	102
<b>Figura 76</b>	Evidencia prueba registro reserva rol Administrador I .....	102
<b>Figura 77</b>	Evidencia prueba registro reserva rol Administrador II .....	103
<b>Figura 78</b>	Evidencia prueba registro reserva rol Administrador III.....	103
<b>Figura 79</b>	Evidencia prueba registro reserva rol Administrador IV.....	104
<b>Figura 80</b>	Evidencia prueba gestión eventos I.....	105

<b>Figura 81</b>	Evidencia prueba gestión eventos II .....	106
<b>Figura 82</b>	Evidencia prueba generación de reportes I .....	107
<b>Figura 83</b>	Evidencia prueba generación de reportes II .....	107
<b>Figura 84</b>	Evidencia prueba generación de reportes III.....	108
<b>Figura 85</b>	Evidencia prueba generación de reportes IV.....	108
<b>Figura 86</b>	Compilación código fuente.....	110
<b>Figura 87</b>	Configuración servidor .....	111
<b>Figura 88</b>	Configuración FTP en visual .....	111
<b>Figura 89</b>	Ejecución publicación desde visual estudio .....	112
<b>Figura 90</b>	Publicación correcta API REST.....	112
<b>Figura 91</b>	Se configura base de datos en servidor .....	113
<b>Figura 92</b>	Se cargan los scripts de todas la base de datos, tables, procedimientos almacenados etc.....	113
<b>Figura 93</b>	Se configuran los certificados SSL para la API REST.....	114
<b>Figura 94</b>	Se crea carpeta de distribución desde el CLI de Angular.....	114
<b>Figura 95</b>	Se carga carpeta a Netlify .....	115
<b>Figura 96</b>	Aplicación publicada .....	115
<b>Figura 97</b>	Ingreso al módulo en versión publicada.....	116

## Introducción

Al realizar una detenida reflexión sobre la situación actual de las tecnologías digitales, especialmente aquellas que se desarrollan o ejecutan en internet, se destaca el impacto significativo de las mismas en todos los aspectos de la vida, desde la forma en que nos entretenemos hasta cómo se lleva a cabo la educación, se subraya la importancia de integrar tecnologías digitales para la gestión educativa, esto no solo abarca el ámbito académico de las instituciones educativas, sino también los aspectos administrativos, ya que prestar atención a esta área permite mejorar la eficiencia administrativa y, a su vez, el aprendizaje, la UNESCO, dentro de sus estrategias, ha establecido una enfocada en este aspecto: la Estrategia de la UNESCO sobre la Innovación Tecnológica en la Educación (2022-2025), estrategia que promueve el uso de sistemas de información y de gestión con el objetivo de modernizar la administración educativa, además, apoya la atención a estos aspectos al mencionar que la transformación digital en la educación superior es un factor crucial para fomentar la innovación, la colaboración y la competitividad en el ámbito académico (UNESCO, 2023).

Considerando estos antecedentes que resaltan la importancia de la modernización en las tecnologías de gestión educativa, se propone el diseño y desarrollo de un software destinado a mejorar la gestión de préstamos y reservas de recursos y espacios físicos en una institución de educación superior, este sistema permitirá a los usuarios de diversas dependencias conocer la disponibilidad de recursos como computadoras, video beams, cámaras digitales, micrófonos, portátiles, etc. así como de espacios físicos como laboratorios, salas y salones, la implementación de este software facilitará la planificación y utilización eficiente de estos elementos, optimizando su uso y disponibilidad.

Además, el proyecto contempla la inclusión de funcionalidades que permitan a los administradores del sistema generar informes detallados sobre el uso de los recursos, tanto a nivel de dependencias como en el contexto general de la institución, estos informes proporcionarán información valiosa para la toma de decisiones y la mejora continua de los procesos, contribuyendo a una gestión más efectiva y eficiente.

Para garantizar la eficiencia y modernización del sistema, se propone la adopción de nuevas tecnologías que faciliten una implementación ágil y eficiente, la exploración de metodologías de desarrollo ágil y la adaptación de tecnologías emergentes son consideradas esenciales para asegurar la flexibilidad y escalabilidad del software, permitiendo que este evolucione conforme a las necesidades y requerimientos de la institución.

A lo largo de este documento se expondrán las diferentes fases del desarrollo del proyecto, destacando la ejecución correcta del mismo y la aplicación de las habilidades y conocimientos adquiridos durante la carrera de Ingeniería de Sistemas, esto evidenciará el éxito de los objetivos planteados durante la etapa académica como estudiante, cada fase del desarrollo será detallada minuciosamente, resaltando las decisiones tomadas, los desafíos enfrentados y las soluciones implementadas, de este modo, se ofrecerá una visión completa del proceso de creación del software propuesto, demostrando la capacidad para aplicar los aprendizajes obtenidos en la práctica.

## **Planteamiento del problema**

En el contexto de la institución de educación superior para la cual se desarrolla el software, la gestión actual de préstamos y reservas de recursos se ha vuelto ineficiente y desarticulada debido al uso de un sistema existente que no ha evolucionado conforme a las actuales necesidades, sistemas como Excel y una base de datos en Microsoft Access son los medios utilizados para la gestión actual, pero debido a la alta demanda, se han vuelto muy ineficientes y lentos, además, la gestión se realiza de manera presencial, lo cual resulta poco práctico considerando las dimensiones de la institución, se necesita urgentemente que la gestión pueda realizarse de manera remota a través de Internet.

La problemática también radica en que el sistema actual, desarrollado hace algunos años, no fue concebido para manejar la magnitud actual de solicitudes, cada dependencia ha implementado soluciones de manera independiente, lo cual genera falta de uniformidad en la ejecución del proceso y dificulta la cohesión en la administración de los recursos, así como su auditoría para mejorar la calidad de los procesos.

Adicionalmente, otra de las problemáticas destacadas es que el sistema actual se encuentra basado en una tecnología que no resulta idónea para la gestión eficiente de esta información, la tecnología obsoleta no solo limita la capacidad de adaptación del sistema a las nuevas demandas, donde se pueden contar con decenas de solicitudes por semana y cientos por mes, sino que también aumenta la probabilidad de errores futuros, esta limitación tecnológica se vuelve especialmente crítica en el contexto de un crecimiento continuo en la ejecución del proceso de préstamos, reserva de equipos y espacios físicos.

Esta situación afecta negativamente la eficiencia para los usuarios, ya que la falta de coherencia en la ejecución del proceso conduce a una experiencia inconsistente, esto se ha reflejado en las encuestas de satisfacción del usuario, donde se expresan quejas respecto al sistema actual, que está desactualizado, para una institución que busca mejoras tecnológicas como parte de su plan estratégico de calidad, aspecto que resulta problemático y crítico.

## **Justificación**

La imperiosa necesidad de desarrollar un nuevo software para la gestión de préstamos y reservas de recursos y espacios físicos en una institución de educación superior surge como respuesta a las deficiencias evidentes en el sistema actual, donde este se encuentra desactualizado, agotadas las opciones básicas de gestión como Excel y Microsoft Access. La falta de cohesión y adaptabilidad, junto con la obsolescencia tecnológica, ha generado una serie de problemas que afectan tanto la eficiencia operativa como la experiencia de los usuarios.

La actual diversidad en la implementación del proceso por parte de cada dependencia dificulta la uniformidad y coordinación en la administración de recursos, esta falta de sinergia impacta directamente en la eficiencia del proceso y, por ende, en el flujo normal de trabajo dentro de la institución, además, la tecnología actualmente empleada no solo limita la capacidad de adaptación del sistema a las crecientes demandas, sino que también plantea riesgos de errores futuros que podrían afectar negativamente la calidad del servicio.

En este contexto, la propuesta de desarrollar un nuevo software no solo representa una solución a los problemas actuales, sino que también se posiciona como una oportunidad ideal para aplicar los conocimientos adquiridos durante el período académico en el área de ingeniería de sistemas, el proyecto no solo aborda los aspectos técnicos del desarrollo de software, sino que también involucra una serie de desafíos que van desde el análisis y planteamiento del problema, la recolección de requisitos, la propuesta de soluciones, la evaluación de objetivos, hasta la gestión de recursos.

Al realizar un análisis comparativo de lo que se busca lograr con un nuevo desarrollo, se evidencia además una mejora sustancial en eficiencia y usabilidad, desde los tiempos de

respuesta por cada solicitud hasta la generación de reportes. Estos son algunos de los factores que se busca mejorar.

Indicador	Sistema Actual (Access)	Nuevo Software Propuesto
Tiempo de respuesta por solicitud	8 minutos	20 segundos / 97% Mejora
Tasa de errores por mes	10 errores	3 errores / 70% Mejora
Número de tareas manuales por proceso	6	2 / 66% Mejora
Accesibilidad	Solo 1 pc en red local	Acceso web desde cualquier equipo
Eficiencia en uso de recursos tecnológicos	Baja: recursos ociosos, sin seguimiento en tiempo real	Alta: recursos optimizados con visualización en tiempo real

Además, también proporciona la oportunidad para que en la institución se adopten nuevas tecnologías de desarrollo y arquitecturas de software, el desarrollo de la propuesta con una API Web REST en el entorno de desarrollo .NET permite una arquitectura moderna y flexible para la gestión de datos, garantizando la interoperabilidad y la facilidad de integración con otros sistemas, esto facilita la obtención de datos desde la base de datos, la cual se gestiona con ORACLE, un sistema ampliamente utilizado en entornos empresariales que ofrece seguridad y rendimiento.

Por otro lado, la elección del framework ANGULAR para el desarrollo de aplicaciones web ofrece una experiencia de usuario fluida y altamente interactiva, desarrollado en TypeScript, un lenguaje moderno y eficiente, ANGULAR proporciona una estructura robusta y modular para la construcción de interfaces de usuario escalables y de alta calidad, esto contrasta significativamente con los proyectos monolíticos que se utilizan actualmente en la institución. Al adoptar nuevos esquemas, se promueve la renovación de los sistemas actuales, dando paso para

que más procesos dentro de la institución sean migrados a la propuesta a desarrollar, lo que justifica aún más la relevancia de esta nueva implementación.

Además, este proyecto se presenta como una opción de grado que permite consolidar los conocimientos teóricos y prácticos en un entorno de trabajo real, la ejecución de este proyecto como opción de grado implica un reto significativo al abarcar todas las facetas de un proyecto real, brindando la oportunidad de aplicar de manera integral y práctica los aprendizajes obtenidos a lo largo del período académico, asimismo, contribuirá a mi formación profesional al enfrentarme a los desafíos propios de un proyecto de envergadura y complejidad como este.

## **Objetivos**

### **Objetivo General**

Desarrollar un software web que permita optimizar la gestión de inventarios y recursos tecnológicos, así como el préstamo de espacios físicos en una institución de educación superior, mejorando la eficiencia en el uso de los recursos, facilitando su seguimiento y control.

### **Objetivos Específicos**

Recopilar y documentar requisitos, identificando el uso actual del proceso, así como las necesidades que actualmente no están siendo cubiertas, y documentando de manera clara los requisitos funcionales y no funcionales.

Diseñar la arquitectura e interfaz buscando satisfacer las necesidades ya identificadas, centrándose en la escalabilidad y la facilidad de mantenimiento futuras, así como en mejorar la experiencia del usuario final.

Desarrollar y codificar el software, utilizando las mejores prácticas de desarrollo y estándares de la industria, basándose en los requisitos establecidos por el cliente.

Ejecutar pruebas exhaustivas para validar la funcionalidad, integración y aceptación por parte del usuario, garantizando que el software cumpla con los requisitos y expectativas del cliente.

Desplegar el software de manera efectiva y mantener su funcionamiento, abordando posibles mejoras en los procesos y proporcionando una transición suave del entorno anterior al nuevo.

## **Marco de Referencia**

Actualmente, los procesos administrativos de las empresas y organizaciones han evolucionado al margen de la propia evolución de la tecnología, específicamente con el desarrollo de internet, dentro de esta tecnología se destacan las mejoras en los aplicativos web, que son sistemas de gestión de información que se ejecutan mediante el navegador web, presentando diferentes mejoras frente a aplicativos predecesores, como la facilidad de acceso para cualquier usuario de las mismas organizaciones, precisamente, en "Web-Based Human Resources", Alfred J. Walker discute en detalle cómo las aplicaciones web pueden mejorar significativamente los procesos administrativos. Destaca que las plataformas web permiten una mayor eficiencia y precisión en la gestión de datos, acceso en tiempo real a la información, y facilitan la toma de decisiones basadas en datos precisos y actualizados (Walker, 2003, p. 52). Es por eso que es relevante ejecutar un proyecto de esta escala en un proceso tan importante en una institución de educación superior, frente a esto, también podemos analizar los siguientes puntos.

### **Marco teórico**

Como respaldo académico y basados en las evidencias empíricas planteadas por diferentes metodologías, técnicas y tecnologías, contamos con bases sólidas para el desarrollo de un aplicativo web con las características propuestas:

- Gestión de proyectos: Además de las habilidades básicas en la gestión de proyectos, también es fundamental el uso de herramientas digitales para dicha gestión. Estas herramientas han revolucionado el proceso, convirtiendo la gestión en una disciplina estratégica para concretar ideas tangibles (J. Martínez, 2024).

- Teoría de las tecnologías web: Diferentes estudios respaldan el avance de las tecnologías web y su alto impacto en la competitividad empresarial, además de proporcionar una mayor eficiencia en las mismas (W3Techs, 2022).
- Diseño centrado en el usuario: Bernal Torres, M. . (2025). Destaca “Un diseño centrado en el usuario no solo busca hacer que la interacción sea fluida, sino que también mejora la satisfacción del cliente, lo que se traduce en un mayor compromiso y lealtad”. (p. 155).

La variedad de principios y estudios teóricos han demostrado cómo el diseño centrado en el usuario mejora la usabilidad del software y la adopción de nuevas tecnologías o productos tecnológicos (Norman, 2013).

### **Marco conceptual**

En el contexto del desarrollo, se establecen principios y directrices como guía en el desarrollo del proceso propuesto.

Metodologías ágiles y su relación con la gestión de proyectos de software: Estos conceptos destacan la importancia de un enfoque estructurado y flexible, apoyándose en metodologías como SCRUM y Kanban, estas metodologías no solo proporcionan una gestión eficiente e incremental, sino que también fomentan la adaptación a los cambios en los requisitos del proyecto a desarrollar, Medina Salgado, A. B. . (2024). Señala que “el uso de Scrum permite a los equipos gestionar de manera más efectiva sus recursos y tiempo, lo que conduce a un aumento significativo en la productividad” (p. 204).

Actualización tecnológica a tecnologías web: Este aspecto es de gran importancia para la adopción de procesos modernos y emergentes enfocados en la web, los procesos de actualización

son constantes, buscando no solo mejorar el rendimiento y la seguridad de los procesos administrativos, sino también mejorar la competitividad del usuario que implementa dicha actualización.

Diseño UI/UX: Proporciona un enfoque que actualmente es necesario para cualquier desarrollo, desde el inicio hasta el final de este, esto asegura no solo que un desarrollo sea funcional, sino también que proporcione un uso e interacción fáciles para el usuario final, todo esto gracias a los principios de diseño centrado en el usuario, aplicando el uso de la adaptabilidad a los dispositivos móviles algo fundamental en la actualidad, Bernal Torres, M. . (2025), menciona que “Las aplicaciones deben ofrecer una navegación intuitiva, tiempos de carga rápidos y diseños visualmente atractivos.”(p. 155).

### **Marco Tecnológico**

El éxito de un proyecto de base tecnológica está ligado a la elección correcta de tecnologías y herramientas de desarrollo que estén a la vanguardia, es fundamental usar herramientas actuales tal como lo menciona Gorton, I.(2020) “Las herramientas y tecnologías actuales permiten a los desarrolladores construir aplicaciones más eficientes y adaptativas, aprovechando las últimas innovaciones en el campo del desarrollo web”, esta correcta decisión permite definir la calidad, escalabilidad y sostenibilidad del proyecto, a continuación, se estableces estas bases para el desarrollo del actual proyecto.

### ***Lenguajes de programación***

La piedra angular del software desde sus inicios ha sido los lenguajes de programación. Estos nos permiten comunicarnos con un sistema de cómputo, que a su vez nos permite construir

y desarrollar software según las necesidades. En la actualidad, el número de lenguajes se ha diversificado. Para este caso, podemos mencionar dos de gran importancia para el desarrollo del software actual: C# y TypeScript. Ambos lenguajes están en alta demanda y son mantenidos por la empresa Microsoft. Precisamente, estos son dos de los factores por los cuales se seleccionan estos lenguajes para un desarrollo moderno: su alta demanda y, sobre todo, su alta versatilidad. Hablando específicamente de C Price (2021) destaca que "C# 10 ofrece un conjunto robusto de características que facilitan el desarrollo de aplicaciones modernas y mantenibles. Utilizar las últimas versiones de C# y .NET garantiza que los desarrolladores puedan aprovechar las mejoras en el rendimiento, seguridad y productividad" (p. 45). En cuanto a TypeScript, se puede destacar que es uno de los lenguajes más relevantes actualmente, debido a su flexibilidad para el desarrollo de software que se ejecuta a nivel de cliente o en navegadores, donde hay un creciente interés en realizar procesos directamente en los navegadores, dejando de lado las antiguas implementaciones que se ejecutaban a nivel de escritorio.

### ***Frameworks***

Ante el aumento de la complejidad del desarrollo de software y las necesidades de prontitud en el mismo, se proporcionan frameworks de desarrollo, estos frameworks ofrecen estructuras y herramientas ya estandarizadas, permitiendo así utilizar conceptos y prácticas para solucionar problemáticas particulares similares, al igual que el aumento en los lenguajes de programación, también hay una creciente cantidad de frameworks, sin embargo, siguiendo el criterio de uso de estándares actuales, podemos hablar de dos específicos para este caso: Angular y .NET, estas dos tecnologías son ampliamente reconocidas y utilizadas, algunas de las características que las hacen idóneas son su capacidad para permitir un desarrollo ágil con una

estructura clara, entornos robustos y eficientes, así como su alta escalabilidad, que es un factor clave para este proyecto, dado que es lo que el usuario solicita dentro de los requerimientos.

### ***Arquitectura de software***

La arquitectura de un software se refiere a la definición y organización del sistema. el diseño de la arquitectura de cualquier proyecto asegura la escalabilidad, mantenibilidad y eficiencia, tanto en el uso como en el desarrollo del mismo. Actualmente, existen muchos esquemas o arquitecturas en los cuales trabajar; en este caso, se destaca la arquitectura por capas, un esquema altamente utilizado durante muchos años. Esta arquitectura se enfoca en organizar el software mediante un conjunto de capas que interactúan entre sí de forma consecutiva. Este enfoque permite la modularización, la reutilización y facilita el mantenimiento futuro del software. Otra de las características importantes es que esta arquitectura puede adaptarse o configurarse con el fin de cumplir los requerimientos del cliente, y este punto de modificar la arquitectura mencionada es un estándar actual. Bass, Clements y Kazman (2021) argumentan que la arquitectura de software debe ser flexible para adaptarse a los cambios en los requisitos del negocio. Proponen "modificaciones de la arquitectura en capas, como la introducción de capas intermedias o la reorganización de las capas existentes, para mejorar la escalabilidad y la flexibilidad" (p. 102)

### **Marco Legal**

Para el desarrollo del presente documento se tienen en cuenta algunas normativas y legislaciones que garantizan el correcto desarrollo de este tipo de proyectos, además de la gestión de los datos que se manipularán durante y después del desarrollo.

Para la ejecución y elección del proyecto actual, se toma como base el marco legal, las políticas y los procedimientos establecidos por la UNAD para el desarrollo de opciones de trabajo de grado, todo esto se establece en el Acuerdo 006 del 28 de mayo de 2014, artículo 13, específicamente en el punto b, donde se establece el desarrollo de los proyectos de desarrollo tecnológico.

El desarrollo de un proyecto de base tecnológica, como un software, requiere el uso de herramientas de desarrollo, estas, a su vez, están reguladas por una serie de leyes y políticas que buscan proteger los derechos de los creadores, en el caso del proyecto a desarrollar, se utilizarán herramientas de software libre, que se distribuyen con una licencia que permite usar, modificar y distribuir el software sin restricciones. Estas licencias están basadas en la GPL, algunas de estas herramientas, se emplearán los lenguajes de programación C#, TypeScript y el entorno de desarrollo Visual Studio.

### **Marco contextual**

La digitalización y la gestión de procesos de forma eficiente en los últimos años han afectado todas las áreas de la sociedad, entre ellas los entornos educativos, volviéndose un tema crítico para las instituciones, en especial, se ha visto un cambio en el incremento de las tecnologías para mejorar los procesos administrativos, lo que afecta directamente la calidad de la educación, actualmente, una de esas instituciones de educación superior que busca mejorar sus procesos es la Universidad de Boyacá, una institución con una historia importante en la educación de la región, que desde hace muchos años ha sido un actor relevante en la educación. en busca de ser cada vez mejores, se enfoca en el desarrollo de software para sus procesos administrativos.

Dentro de sus muchos procesos está la administración de los recursos tecnológicos y espacios físicos de la universidad, un proceso administrativo a cargo de la división de tecnología, este proceso busca gestionar la disponibilidad de los recursos de forma ordenada y eficiente, para lograrlo, se han construido diferentes tecnologías, cada uno de ellos con tecnologías a la vanguardia para su momento de desarrollo. Sin embargo, frente a un mundo tecnológico que evoluciona y cambia rápidamente, la institución se enfrenta al reto de mantenerse a la vanguardia con tecnologías que apoyen sus procesos tecnológicos, aquí es donde el actual proyecto aplicado permite intervenir y apoyar estos procesos administrativos, buscando desarrollar soluciones modernas, eficientes y óptimas que permitan mantener sus procesos por mucho más tiempo, y continuar apoyando la misión y visión en la educación de la región.

## Metodología

### Enfoque de Desarrollo

En la actualidad, existen diversas metodologías de desarrollo de software, cada una con sus características específicas, como el modelo en cascada, el modelo en V y las metodologías ágiles, como lo es Scrum, la elección de la metodología es importante para asegurar el éxito del proyecto, ya que influye en la forma de gestionar los recursos, tiempos y cambios que surjan durante la ejecución. En este contexto, la flexibilidad y adaptabilidad que ofrecen las metodologías ágiles se destacan como la opción ideal, ya que permiten responder de manera rápida y eficiente a los cambios y necesidades que surjan, también se debe tomar en cuenta que es un proyecto desarrollado por una única persona, lo cual se adapta muy bien a la metodología ágil, "Los equipos que trabajan bajo metodologías ágiles tienden a mostrar un mayor nivel de motivación y productividad, ya que estas metodologías promueven la autonomía y la toma de decisiones compartida" (Fernández, 2020, p. 67). Por ello, es recomendable optar por una metodología ágil que permita una ejecución dinámica, ajustándose a las características de este proyecto.

Dado el análisis detallado de las distintas metodologías ágiles disponibles y considerando las particularidades del proyecto actual, se ha optado por la metodología Kanban, entre las diversas metodologías ágiles, Kanban destaca por sus múltiples ventajas, las cuales se alinean de manera precisa con las necesidades del proyecto, un punto importante a destacar es que Mena Inca (2023) señala que "la metodología Kanban facilitó la gestión del flujo de trabajo en el desarrollo de la aplicación web, permitiendo una visualización clara y continua de las tareas, lo que resultó en una mejora significativa en la eficiencia del equipo" (p. 45). A continuación, se

exponen las razones que fundamentan esta elección, así como las características clave que hacen de Kanban la opción más adecuada dentro del conjunto de metodologías ágiles.

Kanban es una metodología ágil que se centra en la visualización y gestión del flujo de trabajo para mejorar la eficiencia y productividad, se realiza mediante el uso de un tablero visual donde se representan las tareas y el estado de cada una, permitiendo una clara visualización del progreso, esta fomenta un flujo equilibrado del trabajo evitando sobrecargas, además de su fácil implementación la cual no requiere de una estructura compleja, lo que lo hace ideal para desarrolladores individuales, promoviendo también la mejora continua a través de la reflexión y análisis constante.

### **Flexibilidad y adaptabilidad**

Frente a un proyecto que se lleva a cabo por una sola persona, requiere una metodología flexible y adaptativa, kanban permite ajustar las prioridades y la carga de trabajo según sea necesario, sin la rigidez que podría imponer otras metodologías como Cascada o El modelo en V, asegurando el poder responder rápidamente a cambios y ajustes necesarios optimizando así el proceso de desarrollo.

### **Visualización del trabajo**

La visualización del flujo de trabajo es un fundamental, un tablero proporciona una visión precisa de las tareas pendientes, y su estado actual, esencial para gestionar y priorizar de manera eficiente, logrando así tener un control absoluto sobre el proceso del proyecto, identificando cuellos de botella y tomar decisiones informadas para mejorar la eficiencia.

### **Limitación del exceso de trabajo (WIP Limits)**

El uso de kanban permite establecer límites en la cantidad de trabajo en progreso, esto ayuda a evitar sobrecargas en las tareas, lo que resulta en un flujo constante de trabajo equilibrado, de este modo, se mantiene un ritmo sostenible, mejorando la calidad y la eficiencia.

### **Adaptación al tamaño del proyecto**

Para proyectos pequeños como el planteado, otras metodologías más pesadas pueden resultar ineficientes o innecesarias, a diferencia de estas Kanban ofrece una estructura ligera pero efectiva, como herramientas necesarias para gestionar eficientemente el trabajo sin aumentar la carga de trabajo en cada tarea.

### **Simplicidad**

Kanban es fácil de implementar, ya que no requiere roles claramente definidos ni estructuras complejas, es ideal para proyectos individuales, permitiendo concentrarse en tareas técnicas o enfocarse en una tarea a la vez, lo que ayuda a evitar distracciones y simplificar el proceso de desarrollo.

La selección de esta metodología de desarrollo se basa también en las evidencias del uso por muchas organizaciones, además de lo demostrado en los casos de éxito de las mismas organizaciones, Edge(2018) menciona que “la transición de Scrum a Kanban en la start-up resultó en una reducción de la sobrecarga administrativa y una mayor flexibilidad, permitiendo al equipo adaptarse rápidamente a cambios en el flujo de trabajo y prioridades” (p. 78). La capacidad de Kanban para ofrecer un flujo de trabajo continuo y gestión dinámica de prioridades lo convierte en una opción adecuada para este proyecto, su enfoque de visualización y la mejora

continua permiten tener una mayor eficiencia y satisfacción, lo que justifican la selección como metodología.

En el desarrollo de software, tenemos varias fases fundamentales para el éxito del proyecto, según Ramos Cardozzo(2016), el desarrollo de software se estructura en varias fases clase que van desde la recopilación y análisis de requisitos, el diseño o elaboración de una arquitectura que guiará el desarrollo, la implementación o codificación, y finalmente el mantenimiento, proporcionando la corrección de error y la adaptación del software, cada fase es crucial para asegurar el cumplimiento de expectativas del producto construido.

Al relacionar estas fases con la metodología propuesta kanban, se busca descomponer las fases del desarrollo en secciones más pequeñas, implementando tareas para estas secciones, aunque Kanban no tiene fase estrictamente definidas como otras metodologías, puede integrar o adaptarse en las siguientes características para el desarrollo:

**Tablero Kanban.** Este tablero propone los estados de las tareas o las secciones de las fases del desarrollo, estas fases son “Pendiente”, “En progreso” y “Completo”, dentro de cada columna del tablero con el estado de las tareas se disponen estas tareas con detalles a desarrollar.

**Tarjetas para las tareas.** dentro de estas tarjetas se establecen los detalles de las tareas a desarrollar, todas juntas componente una actividades propia de la fase de desarrollo, por ejemplo para la fase de análisis de requisitos, como tareas se puede establecer el análisis de los requisitos funcionales o el análisis de los casos de uso, una vez iniciada un tarea del tablero, se dispone en la columna de estado "En progreso", la tarea debe definir la fase del desarrollo a la que pertenece, lo que permite visualizar en qué punto del desarrollo se encuentra el proceso.

**Límites de trabajo.** Es vital implementar límites de WIP para asegurar que no se trabaje en demasiadas tareas a la vez, garantizando también el cumplimiento de las tareas de una fase antes de iniciar las tareas de otras fases, logrando así mantener un flujo constante e identificar los cuellos de botella.

Implementando estos conceptos o adaptación de la metodología, permitirá el cumplimiento de las diferentes fases del desarrollo, Análisis de requisitos, Diseño, Desarrollo e Implementación.

### *Fases de desarrollo*

Dentro de todo proyecto de software es vital tener un ciclo de vida claramente definido, este ciclo permite seguir una estructura garantizando el orden, calidad y éxito de todo proyecto, en su obra, Cervantes y Gómez (2020) describen que "las fases de desarrollo de software deben ser vistas no solo como una secuencia de pasos, sino como un ciclo iterativo en el que cada fase retroalimenta a las demás, mejorando constantemente el producto final" (p. 45), a continuación, se describen las fases a ejecutar en el actual proyecto, donde cada fase será monitoreada mediante la metodología Kanban.

- **Levantamiento y análisis de requisitos.** durante esta fase se busca identificar las necesidades del cliente, esto mediante la recolección de información propia de los procesos que busca apoyar el proyecto a desarrollar, dentro de los puntos a analizar son los requisitos funcionales y no funcionales entre otros muchos aspectos, precisamente se vale de herramientas útiles en esta fase para cumplir su meta, "El análisis de requisitos se realiza mediante encuestas, entrevistas y análisis de mercado para especificar claramente lo que se espera del software" (Informatec, 2023).

- **Diseño.** Durante esta fase da inicio con procesos propios de desarrollo, para este proyecto se plantea el diseño de la arquitectura de cómo se distribuirán las diferentes entidades y servicios que componen al software, la planificación de los componentes, el diseño de base de datos, y muy importante el cómo se construirá la interfaz gráfica cumpliendo con importante reglas de usabilidad, es un paso fundamental para definir como interactuara el software con el usuario final, "El diseño define cómo interactuarán las diferentes partes del software y se crea un modelo que guiará el proceso de codificación" (Software para Todo, 2023).
- **Desarrollo e implementación.** Los programadores codifican el código para la creación del software, usando como guía toda la información recopilada en fases anteriores, basando su diseño en lo previamente establecido, durante esta fase también es fundamental apearse a la metodología establecida, "durante la codificación, los equipos de desarrollo deben seguir principios ágiles, permitiendo que el código sea flexible y fácil de modificar, mientras se adapta a los cambios continuos de los requisitos" (Navarro et al., 2019).
- **Pruebas.** Exhaustivas pruebas buscando errores dentro del software y su funcionamiento, existen diferentes tipos de pruebas, unitarias y de integración, durante esta fase también se buscará el corregir todos los errores identificados, permitiendo así obtener un software de calidad, en el artículo de Robayo-Jácome (2023), se menciona: "Las pruebas continuas durante todo el ciclo de desarrollo permiten a las empresas asegurar la calidad del software, identificando y corrigiendo errores antes de que el producto llegue al usuario final" (Robayo-Jácome, 2023, p. 180).

### ***Recolección de requisitos***

En la búsqueda de identificar de forma correcta las necesidades del usuario final, se quiere la implementación correcta de técnicas de recolección de requisitos, proceso en el cual se identifica y gestionan las necesidades respecto al proyecto o producto a desarrollar, frente al actual proyecto, se implementará entrevistas, reuniones y también denominado reuniones de revisión de flujo de trabajo.

El objetivo de esta técnica es conocer el estado actual del proyecto utilizado, verificar el flujo de trabajo, en esto se incluye la recolección de los nuevos requisitos a implementar con la intención de mejorar las falencias, Sommerville (2011) menciona que "los requisitos del software no son simplemente una lista de funciones deseadas por los usuarios, sino una serie de restricciones, atributos de calidad y condiciones bajo las cuales el sistema debe operar" (p. 124), al ser Kanban una metodología considerada ágil, se realizarán estas reuniones en la fase inicial del desarrollo, pero también se podrá discutir el uso del proyecto durante el transcurso del desarrollo, discutiendo así el progreso de las tareas y ajustando las prioridades suministradas por el usuario final.

### **Herramientas y Tecnologías**

En el desarrollo de un proyecto de software es fundamental el considerar las herramientas para el desarrollo, en este punto es fundamental la selección de los lenguajes de programación, para este caso hablaremos de dos fundamentales, uno para el trabajo a nivel del servidor y otro a nivel del cliente en este caso hablamos de C# y TypeScript, dos lenguajes altamente utilizados en proyectos de software, ahora en la actualidad el utilizar un lenguaje de programación puro no es lo más recomendable pensando en mejorar la eficiencia del desarrollo, por lo que se ha dado gran

importancia a los frameworks de desarrollo, herramienta fundamentales en el actual paradigma de desarrollo, estos frameworks, son marcos de trabajo que proveen concepto y prácticas para el desarrollo.

De angular se puede destacar el uso de la lógica de negocio en el propio lado del cliente mostrando así muchos beneficios para el usuario como un uso más fácil del software permitiendo una mayor usabilidad y una mejor experiencia de usuario, Farinango Cañiza (2020) argumenta que "Angular ofrece una estructura robusta y flexible que permite el desarrollo eficiente de aplicaciones single-page, facilitando tanto la gestión de estados como la actualización dinámica del contenido" (p. 45).

Ahora al hablar de .net nos encontramos con muchos beneficios además de ser una tecnología que está a la vanguardia también se destaca lo importante de recibir soporte por una empresa como lo es Microsoft, y cuando hablamos de la combinación propuesta entre .net y angular, encontramos una selección ideas, donde .net está muy bien integrado para el desarrollo con este otro framework donde se complementan siendo .net el centro del desarrollo de la APIs, y Angular el desarrollo de las interfaces que interactuar con el usuario final, entre otros muchos beneficios de seleccionar. Net como tecnología lo encontramos por parte Rebé Martín (2023) que destaca ".NET proporciona una plataforma robusta y escalable que facilita el desarrollo de aplicaciones web complejas, ofreciendo herramientas y bibliotecas que optimizan el manejo de reglas y sanciones" (p. 30); también nos encontramos que, Net proporciona un entorno de desarrollo cómodo y gratuito como lo es Visual Studio un IDE desarrollo completo y muy integrado a esos frameworks antes analizados.

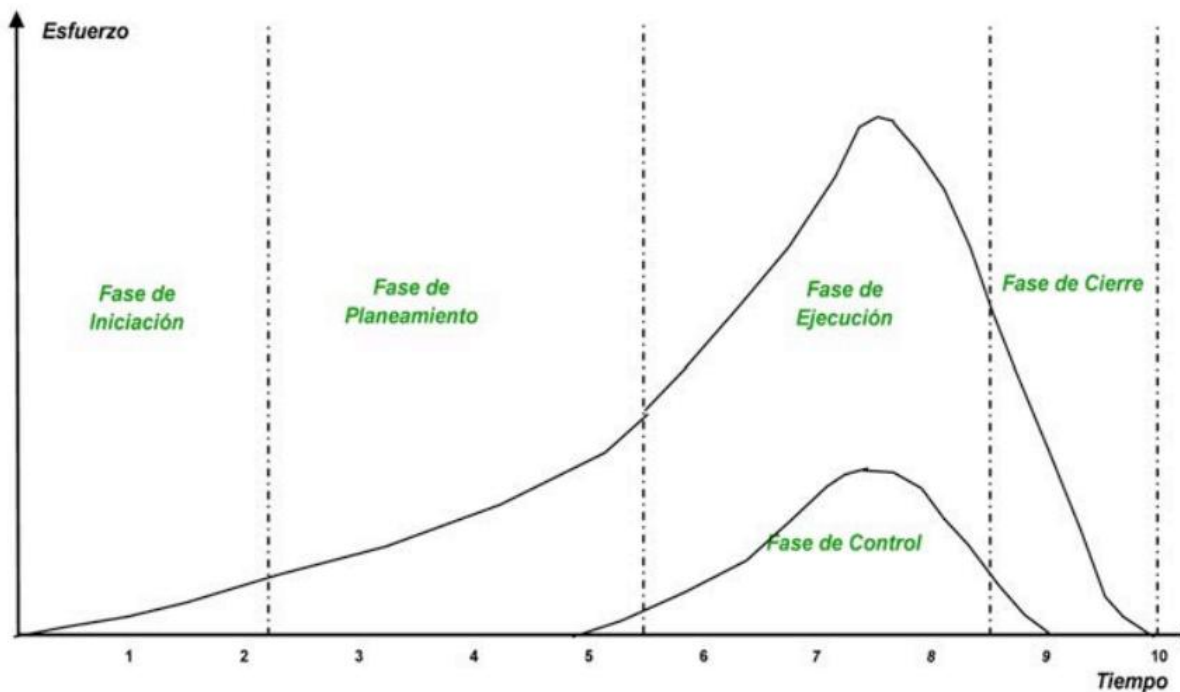
## Gestión de proyectos

Existen variedad de conceptos relacionados a la gestión de proyectos, según Pastor, R. A. T. (2009), se refiere al concepto como una herramienta de tipo gerencial, el cual se enfoca en desarrollar conjuntos de habilidades dentro de un grupo, con un propósito en común que es planificar, organizar, dirigir y controlar, los eventos asociados al proyecto. Es importante tener en cuenta los aspectos que afectan el desarrollo de un proyecto, como lo pueden ser el tiempo, costo, o la calidad requerida en el proyecto, frente a esto es importante tomarse el tiempo de analizar el objetivo del proyecto, los plazos fijados durante el desarrollo y los límites del presupuesto.

### Figura 1

*Fases, escala temporal y esfuerzo en un proyecto*

(Terrazas Pastor, 20009)



### *Planificación del proyecto*

Punto clave para la definición de los objetivos del proyecto y de los entregables que soportan dichos objetivos dentro de un tiempo determinado, en este punto es muy importante plantear un cronograma de actividades, este nos asegura que cada fase del desarrollo se complete dentro de los plazos y de forma exitosa, en este se desglosa las actividades y tareas con los tiempos a desarrollos ayudando a su en el monitoreo del progreso del proyecto, dando como resultado la siguiente propuesta para el actual proyecto.

**Tabla 1**

#### *Cronograma proyecto*

Fase	Actividad	Duración
Levantamiento y análisis de requisitos	• Reunión con cliente	4 Horas
	• Definir requerimientos	1 Semana
	• Documentación casos de Uso	3 Días
Diseño	• Diseño de Base de datos (MER)	1 Semana
	• Diseño de arquitectura	4 Días
	• Mockups Interfaz grafica	5 Días
	• Diseño de componentes	3 Días
Desarrollo	• Revision y aprobación de prototipos	1 Día
	• Desarrollo del Back End	2 Semanas
	• Desarrollo del Front End	3 Semanas
Pruebas	• Pruebas de integración.	9 Días
Implementación	• Ajustes finales	1 semana
	• Despliegue en producción	2 Días

Para la gestión de estas tareas dentro del cronograma creado es muy importante la implementación de herramientas automatizadas para su gestión según Pressman(2010), estas herramientas permiten a los gerentes de proyecto definir las tareas, establecer sus dependencias y asignar recursos humanos y tecnológicos de forma eficiente. Con el fin de alcanzar estos beneficios dentro del proyecto plasmado en este documento, se hace uso de una de estas herramientas muy conocidas actualmente en la gestión Notion.

Notion es una gran herramienta que ha tenido un auge importante en su uso en proyecto de desarrollo de software, esto por su gran variedad de herramientas integras en su entorno de trabajo, alguna de ellas son la visualización de un cronograma en diferentes vistas, progreso con gráficos, filtrar por la información exacta para poder seguir, automatizar flujos de trabajo etc. Mediante el uso de estas muchas herramientas se facilita la planificación, organización y seguimiento del proyecto, "Tus proyectos llegarán más rápido a la línea de meta y no tendrás que cambiar tanto de contexto" (Notion, n.d.), cabe mencionar también lo tiene una excelente integración con metodologías de desarrollo ágiles como lo son SCRUM, y la de más relevancia Kanban, metodología seleccionada para el desarrollo del actual proyecto.

### ***Control de versiones, gestión cambios, mecanismos de verificación y validación.***

Para todo proyecto construido por fases es fundamental contar con el control de lo construido o desarrollado en diferentes partes del tiempo del desarrollo, existen muchas razones por las cuales se debe tener estas herramientas:

- Seguimiento a las modificaciones a nivel de código que se realice en el proyecto, así como la información propia de quien es responsable de dicho desarrollo o

modificación, permitiendo así el auditar los el tipo de cambio y el momento en el que se desarrolló.

- Gestionar diferentes ramas en el desarrollo, logrando así el construir características nuevas sin afectar ramas o brazos principales, evaluando la viabilidad de esa nueva característica de forma segura.
- Seguridad y recuperación, es vital el almacenar el código que se va construyendo, proporcionando así seguridad a eventualidades con los equipos físicos sobre los que se está trabajando, contando con la capacidad de una rápida recuperación de versiones anteriores si se presenta dificultades en algún momento de agregar nuevas funcionalidades.

Existen otras muchas características que proporcionan estas herramientas para la gestión de cambios y control de versiones, además de existir una gran variedad precisamente de herramientas que cumplan con estas características. Para este proyecto se destacará el uso de una muy utilizada en el mundo del desarrollo y es Azure DevOps.

Azure DevOps es un producto de Microsoft aspecto que da seguridad en la calidad de la herramienta, teniendo un soporte continuo por parte de una empresa de renombre como lo es Microsoft, algunos de sus servicios de este, control de versiones, gestiona de requisitos, gestión de proyectos, etc. Algunas características importantes de su uso, su interfaz eficiente, "Azure DevOps proporciona una eficaz interfaz de panel que permite crear una vista personalizada de los datos y flujos de trabajo del proyecto" (Microsoft, n.d.); también encontramos su excelente sistema de control de código fuente, que a la vez es bastante robusto y confiable, "control de código fuente o de versiones son fundamentales para permitir a los desarrolladores colaborar sin problemas en los códigos base y mantener un historial completo de cambio" (Microsoft, n.d.).

## Análisis de Requisitos

### Perfiles de usuario y caracterización de actores

#### *Solicitante*

**Descripción.** Individuo o usuario que pertenece a la institución y accede al sistema con un acceso limitado a ciertas funciones, pero será el actor principal que realiza las solicitudes de préstamo de elementos.

**Necesidades.** Acceder de forma sencilla a la información sobre los elementos a solicitar préstamos, comprendiendo fácilmente si estos elementos están disponibles para préstamos; realizar el proceso de solicitud de forma rápida y de acceso rápido desde diferentes dispositivos.

#### *Administrador*

**Descripción.** Funcionario de la institución o usuario que es responsable de la gestión de todos los elementos del módulo de préstamos, así como tener el acceso a la gestión de todos los elementos y de las solicitudes.

**Necesidades.** Tener un control estricto de los datos de forma rápida y segura, desde la consulta de los elementos a prestar, ver la disponibilidad de los mismos, gestionar las solicitudes de manera eficiente y sin demoras, además de tener a la mano reportes sobre la gestión de todos los datos en el módulo de forma sencilla.

## **Requerimientos funcionales**

En el desarrollo actual, se han identificado requisitos que el sistema para satisfacer las expectativas del usuario final, en este caso, se definen las capacidades y servicios funcionales a ofrecer:

### ***Gestión de Usuario***

RF-001, validar las credenciales como usuario y contraseña del usuario que ingresa al sistema, consultando el rol del usuario dentro del módulo, además de validar los permisos del usuario para acceder a las diferentes funciones del sistema. *(No se realiza registro de usuario, edición y recuperación de contraseñas, al ser este módulo parte de un sistema integrado en el que otro modulo realiza dicho registro).*

RF-002, registrar o eliminar permisos a los diferentes usuarios del sistema, de forma que se evita o permite el solicitar elementos y espacio físicos en el sistema, además de cambiar los roles del usuario de solicitantes administrare o viceversa.

### ***Gestión elementos tecnológicos y espacios físicos***

RF-003, registro de los tipos de elementos o recursos, entre los cuales están los elementos físicos como por ejemplos los equipos de cómputo, o los propios espacios físico como salón y demás espacios.

RF-004, registrar o crear nuevos elementos, como equipos de cómputo, cámaras web, video beams, etc, este registro debe incluir todas las características del elemento, como la marca, el estado, la referencia o registrar si se requiere una aprobación para la reserva de los elementos.

RF-005, registrar o crear espacios físicos, como salones o instalaciones de la institución, este registro debe incluir todas las características del recinto como el número de sillas, el estado de la instalación o registrar si se requiere una aprobación para la reserva del espacio físico.

RF-006, modificar y eliminar elementos o espacios físicos registrados, se debe tener la posibilidad de modificar alguna de las características de estos, o eliminar del todo este recurso, asegurando que la modificación o eliminación no afecta otros procesos propios del proceso.

RF-007, consultar los elementos y espacios físicos, dependiendo del rol del usuario ya sea solicitante o administrado

RF-008, consultar disponibilidad de uno o más elementos, consulta de los detalles de los elementos registrados, tanto para los disponibles como para los no disponibles.

RF-009 Control de inventario y seguimiento de la ubicación de cada uno de los elementos.

### ***Sistema de Reservas***

RF-010, realizar una reserva o solicitud de reserva para los elementos o recursos que requieren una aprobación para la reserva, esto tanto desde el rol de administrador como integrantes de la institución con permiso para ello, relacionando uno o más elementos y solicitante a la reserva del recurso.

RF-011, modificar una reserva, cambiando los elementos relacionados a la reserva, o modificar al solicitante de la reserva, acciones permitidas tanto desde el rol del solicitante como desde un administrador.

RF-012, consultar una o más reservas realizadas tanto por el administrador del sistema como por los solicitantes de las reservas, obteniendo detalles de la reserva como los elementos el horario de la reserva, el estado de la reserva que pueden ser “Reservado”, “Prestado”, “Entregado”, “No asistió” o “Cancelada” e información sobre quién realiza la reserva del elemento.

RF-013, cancelar reservas de elementos, acción permitida tanto por los administradores como por los solicitantes de elementos, permitiendo así liberar disponibilidad de recursos, disponibilidad que se debe actualizar por el sistema para que se puedan aprovechar los elementos por otros usuarios.

RF-014, aprobar o denegar solicitudes de reserva, esto para los elementos que cuentan con esta característica de requerir una aprobación de reserva, acciones para los administradores los cuales tendrán la potestad de aprobar o denegar la solicitud de reserva.

RF-015, entrega y recepción de los elementos entregado en préstamo, acción realizada por el usuario administrador, donde entrega lo elemento o los recibe por el usuario solicitante, acción en la que se cambia los estados de las reservas, si ha sido entregado el elemento respetado de “Reservado” a “Prestado”, en el caso de que el solicitante entregue el elemento luego de ser prestado pasa de esta “Prestado” a “Entregado”, en este registro también queda diligenciada la información como el estado del elemento, tanto al ser presentado como entregado, quien lo recibe y entrega, fechas etc.

RF-016, Confirmación de reservas a tras de correo electrónico institucional.

RF-017, Notificaciones de recordatorios de reservas realizadas por los usuarios.

### ***Control de Préstamos***

RF-018, Control del tiempo del prestamos de los recursos, mediante la generación de notificación de recordativos para la devolución de los recursos pendiente o sin entregar.

RF-019, Registro de incidentes o daños a los recursos reportados durante el uso de los recursos prestados.

RF-020, Generación de reportes sobre el uso de los recursos y espacios por periodos académicos, dependencias, o programas académicos.

### ***Gestión horarios***

RF-021, registro de horarios, estos definirán el espacio de tiempo en el que un elemento estará disponible para el préstamo, registro que tendrá información sobre día de la semana, hora de inicio y fin del préstamo.

RF-022, consulta de horarios, tanto para uno como para más de un elemento, obteniendo datos de disponibilidad de los elementos en los espacios de tiempo del horario consultado.

### ***Reportes y estadísticas***

RF-023, consulta de reportes, acciones únicamente para administradores del sistema, deben ser reportes fáciles de acceder y consultar, se debe permitir tener diferentes reportes según las características de los que se desea consultar, por periodo de tiempo en el que se han realizado los préstamos, por tipos de elementos prestados, por usuario solicitante que realiza los préstamos, por las dependencias a la que pertenecen los solicitantes de los recursos, estado de los préstamos;

reportes que deben estar en diferentes formatos como PDF, Excel o consultar a nivel de interfaz gráfica en la web.

RF-024, consultar de estadísticas, acciones únicamente para usuario administradores, los cuales deben poder en forma de gráficas, las estadísticas de préstamos, estados de disponibilidad en diferentes espacios de tiempos.

### ***Integración con sistemas Externos***

RF-025, Integración al sistema integrado información institucional, juntos a sus demás modulo que proporcionan información sobre los usuarios, como lo son el directorio activo de la institución.

RF-026, Integración a las APIS de autenticación de servicio como Google, para dar acceso al usuario mediante sus correos institucionales.

### ***Seguridad y Privacidad***

RF-027, Cifrado de datos sensibles como los usuario y contraseñas e información de contacto.

RF-028, Uso de tecnologías de autenticación mediante Json Web Tokens, para la validación del acceso a los servidores de la institución.

RF-029 Cumplimento con normativas de protección de datos.

## **Requerimientos no funcionales**

### ***Escalabilidad***

El sistema debe permitir una escalabilidad horizontal lo que permite la adición de más módulo al servicio sin necesidad de rediseñar la arquitectura básica del sistema, esta característica es importante porque proporciona adaptación, mejorando la resiliencia del sistema para manejar aumentos en el tráfico de usuario o en la cantidad de datos procesados.

El sistema debe permitir manejar un gran volumen de usuario y transacción de forma continua, sin tener pérdidas en el rendimiento, mejorando a su vez la experiencia del usuario.

### ***Rendimiento***

El sistema debe ser capaz de responder solicitudes de uno o cientos de usuario un tiempo corto, evitando latencia entre la solicitud y la respuesta, buscando una respuesta de 2 segundo o menos entre cada operación considerada normal.

### ***Disponibilidad***

Los servicios del sistema deben estar como mínimo un el 99.9% del tiempo disponible para acceder a ellos, teniendo en cuentas planes de recuperación en el caso de sufrir percances en la conexión al sistema cuidando los datos de los usuarios.

### ***Modularidad***

Dentro de la arquitectura del sistema se deben contar con elementos modulares, lo que permite agregar, modificar o eliminar módulos sin afectar otros componentes, en este caso la

presentación de los datos debe permitir añadir nuevos módulos para aumentar las funcionalidades, sin la necesidad que las demás características se vean afectadas.

### ***Seguridad***

Todos los datos personales del usuario y del sistema, deben resguardados de forma segura, esto mediante el uso de protocolos de seguridad como https, para la encriptación de los mismo en la transmisión de los datos en las diferentes capas del sistema, proporcionando además controles de acceso mediante JWT, para regular o asegurar que los usuarios conectados sean únicamente los autorizados por el sistema y la institución.

Protección contra ataques cibernéticos comunes como la inyección de SQL, XSS, CSRF, etc. Esto mediante el uso de estándares de seguridad modernos.

### ***Usabilidad***

Los usuarios deben contar con una interfaz de usuario intuitiva y fácil de usar, con una curva de adaptación no mayor a 2 horas para los nuevos usuarios, además de tener la capacidad de usarse en dispositivos móviles, tabletas, y computadores de escritorio.

### ***Mantenibilidad***

Se debe contar con una documentación adecuada donde se incluya comentario detalladas sobre las funciones del sistema y manual para los roles dentro del sistema, se debe permitir el realizar actualización o parches con un impacto mínimo en la disponibilidad del servicio con el objetivo de mejorar las funciones del mismo.

## *Compatibilidad*

La interfaz y su funcionamiento debe estar disponible y ser compatible con todos los navegadores modernos, así como adaptarse a diferentes tamaños de pantalla sin perder la usabilidad de cada uno de los componentes.

## **Árbol de refinamiento funcional**

**Tabla 2**

*Árbol de refinamiento funcional*

Grupo funcional	Función
1. Gestión recursos tecnológicos y espacios físicos.	1.1. Registrar recurso
	1.2. Consultar recurso
	1.3. Eliminar recurso
	1.4. Modificar recurso
	1.5. Consultar disponibilidad recurso
	1.6. Control de inventario
2. Gestión usuarios	2.1. Validar usuario
	2.2. Consultar permisos módulo
	2.3. Registrar permisos usuarios
	2.4. Eliminar permisos usuario
3. Sistema de reservas	3.1. Registrar reservas
	3.2. Modificar reservas
	3.3. Eliminar o cancelar reservas
	3.4. Consultar reservas
	3.5. Registrar solicitudes de reserva
	3.6. Aprobar solicitudes reserva

- 
- 3.7. Denegar solicitudes de reserva
  - 3.8. Modificar estados de la reserva
  - 3.9. Notificación de reservas por email
  - 3.10. Recordatorios de reservas y entregas por email.
  - 3.11. Consultar Histórico de reservas.
  
  - 4.1. Control del tiempo de préstamo de recursos.
  - 4.2. Registro de incidentes y daños
  
  - 5.1. Consultar horarios de disponibilidad
  - 5.2. Registrar horarios
  - 5.3. Eliminar horarios
  - 5.4. Modificar horarios
  
  - 6.1. Consultar reporte estado reservas
  - 6.2. Consultar disponibilidad elementos
  - 6.3. Consultar estadísticas
  - 6.4. Consultas reservas usuario
  - 6.5. Consultar reservas por dependencia
  - 6.6. Consulta reservas por espacios de tiempo
  - 6.7. Estado y condición de los recursos.
  
  - 7.1. Autenticación directorio activo.
  - 7.2. Acceso a la información básica del usuario.
  
  - 8.1. Cifrado de información sensible.
  - 8.2. Cumplimiento de normativas.
-

## Casos de uso

A continuación, se describen los casos de uso de los grupos funcionales identificados, describiendo la manera en la que los usuario o actores interactúa con el sistema, estos permiten entender cómo se utilizara el software, las necesidades y expectativas de los usuarios.

### Figura 2

*Diagrama de caso de uso general*



*Fuente.* Autoría propia

#### 1. Actor: Estudiante / Docente:

##### *Case de Uso 1: Iniciar Sesión*

**Tabla 3**

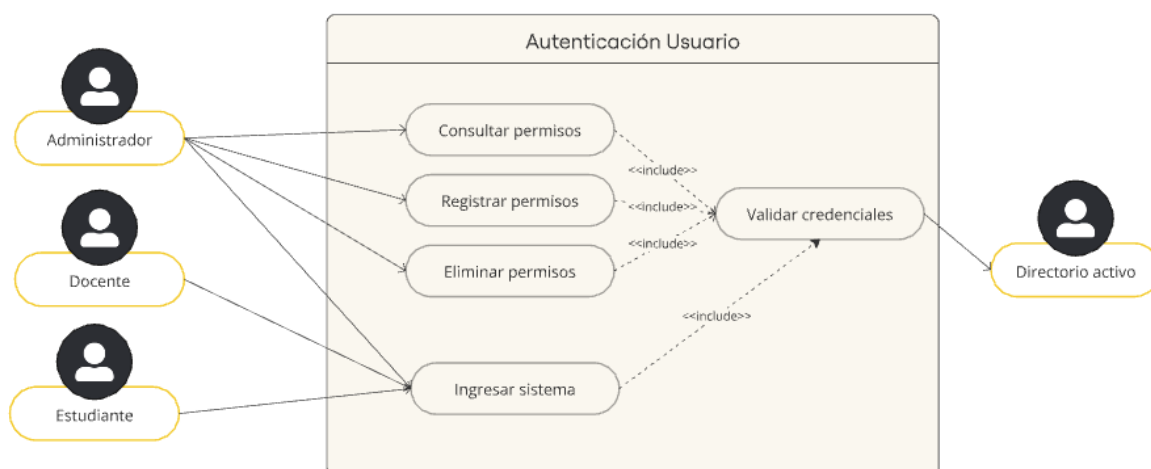
##### Caso de Uso 1

Caso de uso	Autenticación usuario
Descripción	El usuario accede al sistema mediante sus credenciales, validando la existencia del usuario y la información del mismo.

Actor principal	Usuario Estudiante / Docente
Flujo Principal	<ol style="list-style-type: none"> <li>1. El usuario accede a la página de inicio</li> <li>2. Diligencia sus credenciales de usuario y contraseña</li> <li>3. El sistema valida credenciales</li> <li>4. El sistema obtiene los datos del usuario</li> <li>5. Se carga la página principal con los datos del usuario</li> </ol>
Flujo Alternativo	<ul style="list-style-type: none"> <li>• Credenciales de usuario incorrectas se notifica al usuario, y se solicita intentar nuevamente</li> </ul>

**Figura 3**

*Caso de uso 1*



*Fuente.* Autoría propia

*Caso de uso 2: Gestión recursos tecnológicos y espacios físicos.*

**Tabla 4**

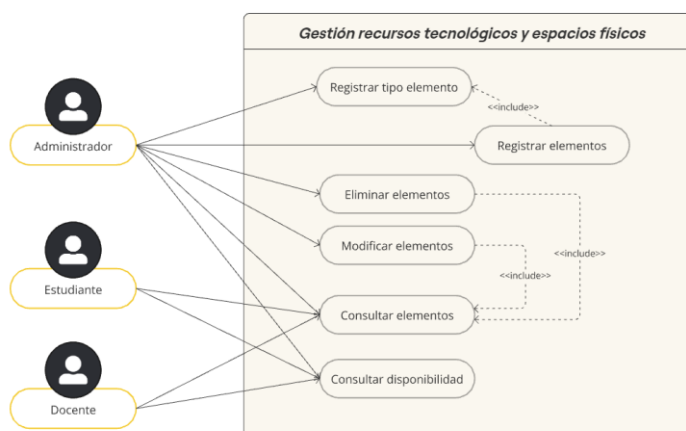
*Case de Uso 2*

Caso de uso	Gestión recursos tecnológicos y espacios físicos
-------------	--------------------------------------------------

Descripción	Proceso en el que los usuario con roles de estudiante o docente puede realizar la consulta las disponibilidad de los recursos, donde ya el rol de administrador realizar dicha gestión.
Actor principal	Usuario Estudiante / Docente
Flujo Principal	<ol style="list-style-type: none"> <li>1. Autenticarse en el módulo de préstamos.</li> <li>2. Seleccionar elemento o recurso.</li> <li>3. Consultar disponibilidad.</li> </ol>
Flujo Alternativo	<ul style="list-style-type: none"> <li>• No existen disponibilidad de recursos para el usuario estudiante o docente se notifica la no existencia del recuero o la disponibilidad.</li> <li>• Se anula los permisos para el rol que desea consultar los recursos, se notifica que se comunique con los administradores para solicitar permisos de consulta.</li> </ul>

#### Figura 4

##### Caso de uso 2



Fuente. Autoría propia

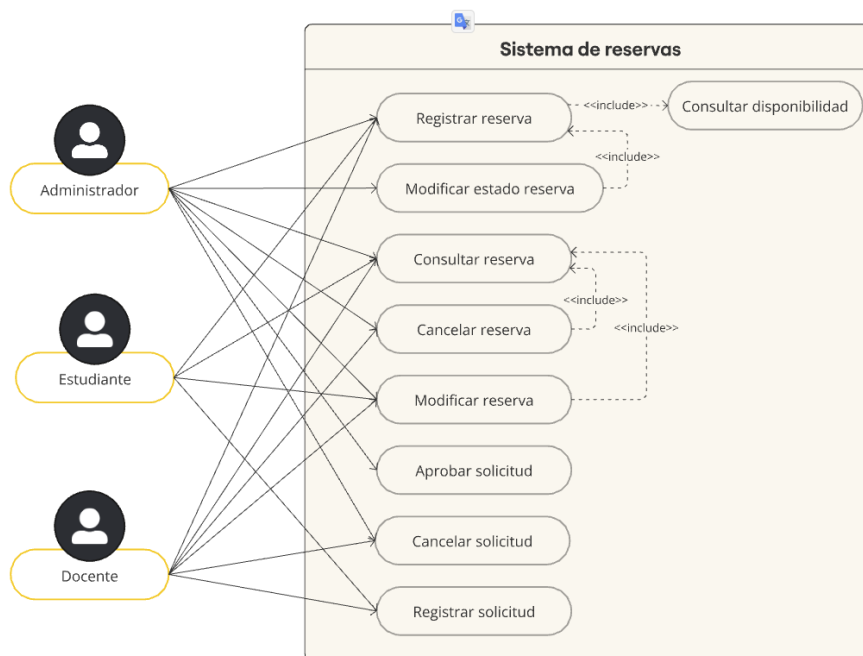
##### Caso de uso 3 Sistema de reservas:

**Tabla 5***Caso de uso 3*

Caso de uso	Sistema de reservas
Descripción	Escenario en el que los docentes y estudiante pueden realizar diferentes acciones como registrar, actualizar y eliminar reservas, es el escenario con más acciones de parte del usuario, también se puede realizar consulta de histórico de reservas para cada usuario, así como acciones propias del sistema relacionadas a las notificaciones por email.
Actor principal	Usuario Estudiante / Docente
Flujo Principal	<ol style="list-style-type: none"> <li>1. Autenticarse en el sistema.</li> <li>2. Consultar disponibilidad de recursos</li> <li>3. Seleccionar la disponibilidad y reservar recursos.</li> <li>4. Si se requiere se consulta reserva, actualiza o eliminar reserva.</li> <li>5. Se notifica vía correo las acciones del usuario. <ul style="list-style-type: none"> <li>• No existen disponibilidad de recursos para el usuario estudiante o docente se notifica la no existencia del recurso o la disponibilidad.</li> </ul> </li> </ol>
Flujo Alternativo	<ul style="list-style-type: none"> <li>• Se anula los permisos para el rol para la solicitud, edición o eliminar de reservas de los recursos, se notifica que se comunique con los administradores para solicitar permisos.</li> </ul>

**Figura 5**

Caso de uso 3

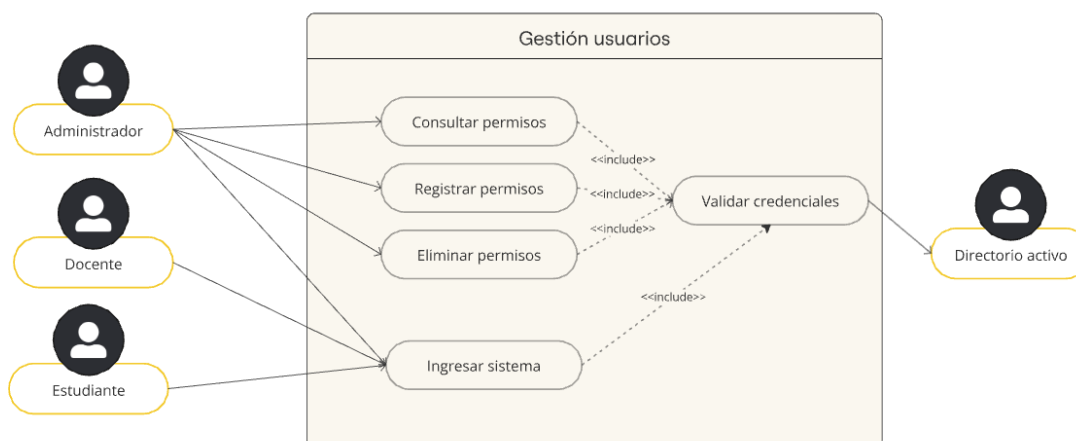
*Fuente.* Autoría propia**2. Actor: Administrador:****Caso de uso 4: Gestión Usuario****Tabla 6***Case de Uso 4*

Caso de uso	Gestión usuarios
Descripción	Como administrador del sistema es el proceso en el que se asigna o elimina permisos a los usuario del sistema con roles de docentes o estudiantes.
Actor principal	Usuario Administrador
Flujo Principal	<ol style="list-style-type: none"> <li>1. Autenticarse en el sistema</li> <li>2. Ir a la gestión usuarios</li> </ol>

- 
3. Consultar usuario del sistema
  4. Asignar o eliminar permisos a usuario
- 

**Figura 6**

Caso de uso 4

*Fuente. Autoría propia***Caso de uso 5: Gestión recursos tecnológicos y espacios físicos.****Tabla 7**

Caso de uso 5

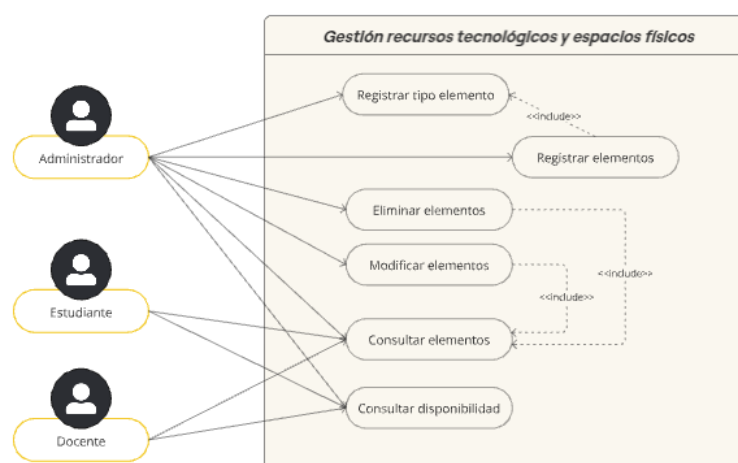
Caso de uso	Gestión recursos tecnológicos y espacios físicos
Descripción	Escenario en el que los administradores gestionan los recursos dentro del sistema, acciones de crear, editar y eliminar recursos, así como asignar disponibilidad, esto para que otros roles puedan consultar y reservar recursos.
Actor principal	Usuario Administrador
Flujo Principal	<ol style="list-style-type: none"> <li>1. Autenticarse en el sistema</li> <li>2. Ir a la gestión recursos</li> <li>3. Registrar, editar o eliminar recurso</li> <li>4. Ir a opciones gestión de disponibilidad</li> </ol>

---

---

 5. Registrar, editar o eliminar disponibilidad
 

---

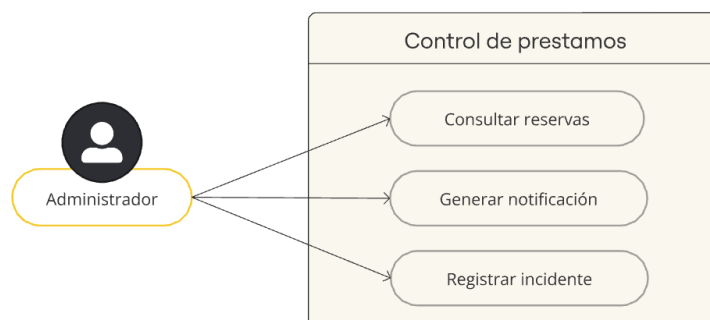
**Figura 7***Caso de uso 5**Fuente. Autoría propia***Caso de uso 6: Control prestamos****Tabla 8***Caso de uso 6*

Caso de uso	Control de prestamos
Descripción	Gestionar el tiempo de préstamos para los recursos, además de controlar incidencias y daños que sufren los recursos en el tiempo del préstamo.
Actor principal	Usuario Administrador
Flujo Principal	<ol style="list-style-type: none"> <li>1. Autenticar en el modulo</li> <li>2. Consultar reservas realizadas</li> <li>3. Seleccionar tiempo de préstamo o registrar incidencia o daño.</li> <li>4. Notificar vencimiento del préstamo al usuario que lo solicito.</li> </ol>

---

## Figura 8

### Caso de uso 6



Fuente. Autoría propia

### Caso de uso 7: Gestión Horarios

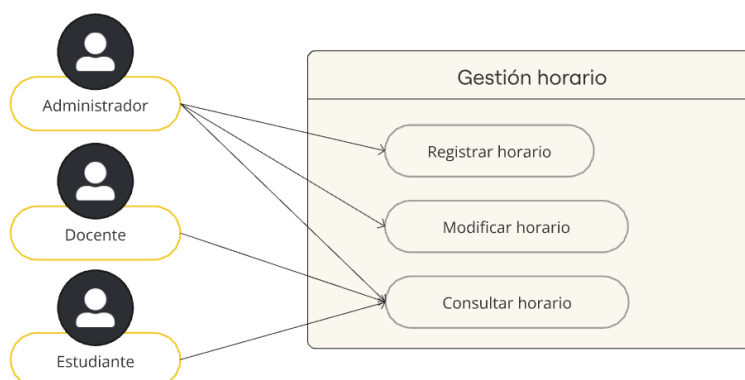
## Tabla 9

### Caso de uso 7

Caso de uso	Gestión Horarios
Descripción	Administrar los espacios de tiempo en el que van a estar disponibles los recursos que se van a destinar a préstamo, tarea a cargo de los administradores del sistema, donde los solicitantes solo podrán consultar los horarios en los que están disponibles los recursos.
Actor principal	Usuario Administrador
Flujo Principal	<ol style="list-style-type: none"> <li>1. Autenticar en el modulo</li> <li>2. Clic opción Horario”</li> <li>3. Registrar, Editar o eliminar horario relacionado a un elemento o recurso.</li> </ol>

## Figura 9

### Caso de uso 7



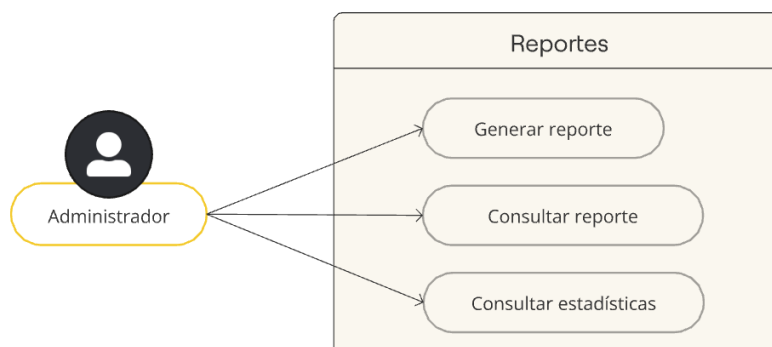
Fuente. Autoría propia

### Caso de uso 8: Reportes y estadísticas

## Tabla 10

### Case de uso 8

Caso de uso	Reportes y estadísticas
Descripción	Consultar información relevantes sobre el proceso, como el número de préstamos por usuario, o por dependencia, cantidad de elementos registrados, etc.
Actor principal	Usuario Administrador
Flujo Principal	<ol style="list-style-type: none"> <li>1. El administrador selecciona la opción “Generar reportes”</li> <li>2. El sistema da opciones para filtrar y parámetros para personalizar el reporte.</li> <li>3. Una vez seleccionado los parámetros se da clic en “Generar reporte”</li> <li>4. El sistema generara reporte en diferentes formatos según la selecciona del usuario (pdf, Excel, etc).</li> </ol>

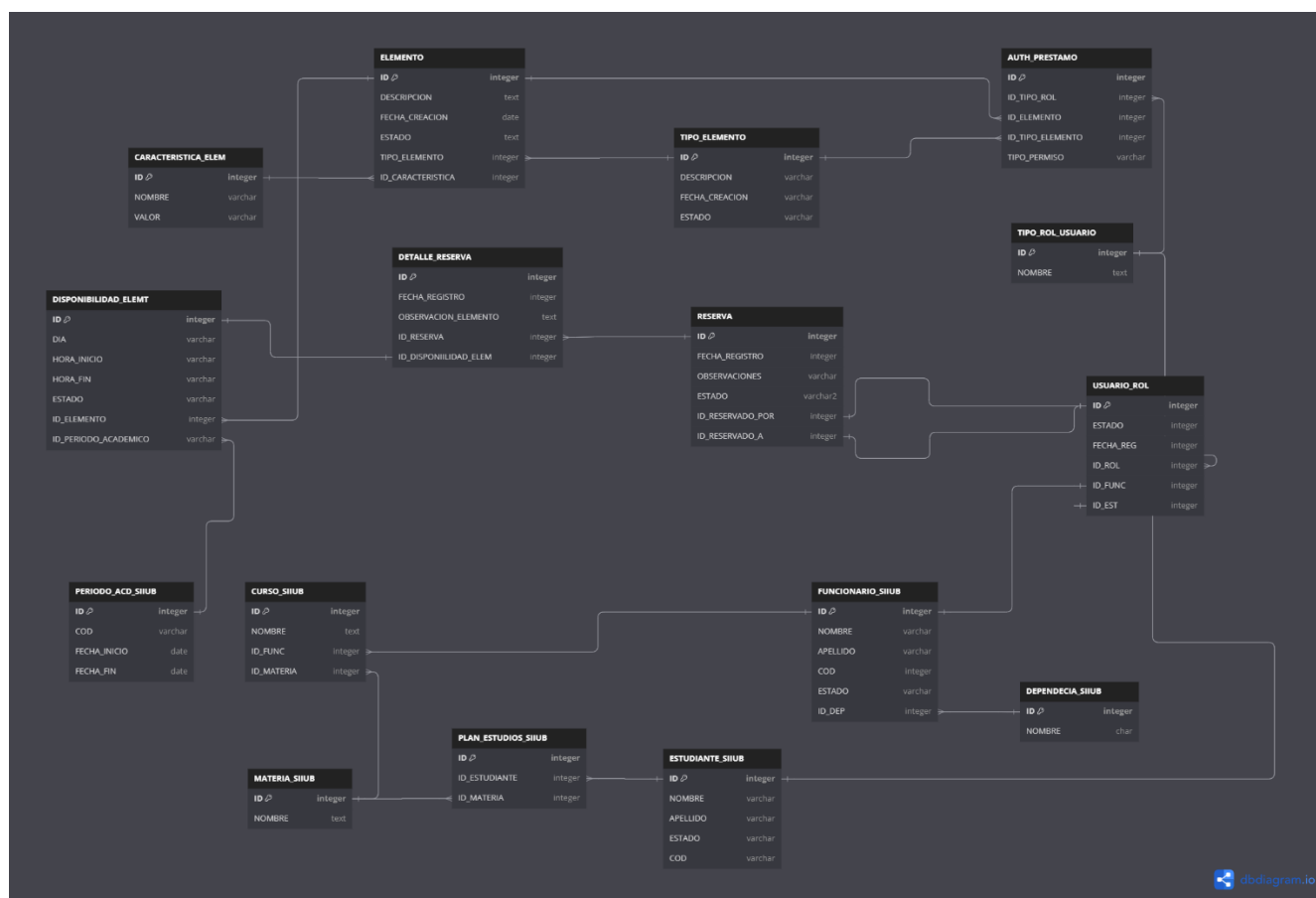
**Figura 10***Caso de uso 8**Fuente. Autoría propia*

## Diseño

### Diseño Base de datos (MER)

Figura 11

Modelo entidad relación (MER)



Fuente. Autoría propia

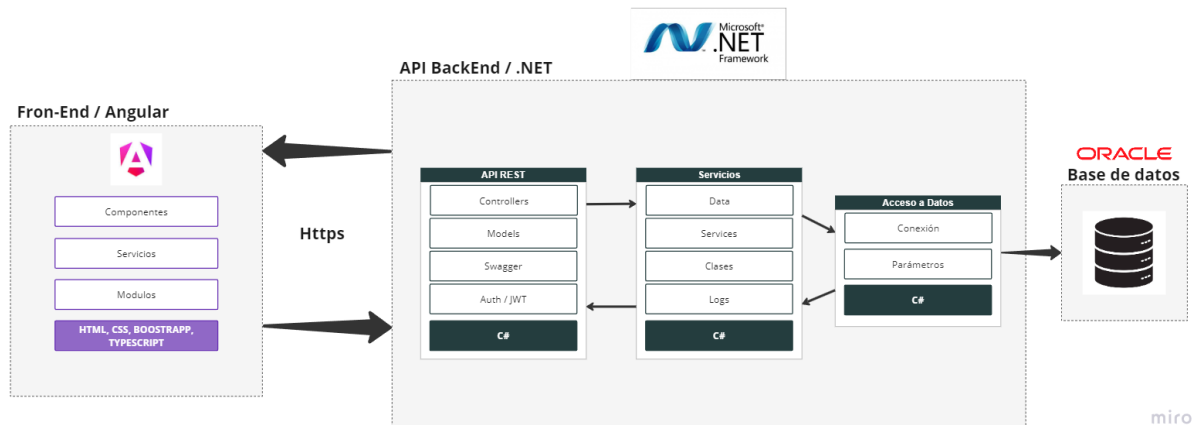
### Arquitectura del sistema

Con los avances en el desarrollo de software, cada día se enfrentan más retos, entre los cuales se destaca cómo organizar cada componente dentro de un sistema. Cada distribución o desarrollo presenta necesidades diferentes, por lo que, con el tiempo, han surgido diversas

arquitecturas de software con el propósito de ofrecer sistemas robustos y, sobre todo, escalables, que puedan satisfacer las necesidades de los clientes y usuarios.

Actualmente, existen muchos esquemas o arquitecturas en los cuales trabajar; en este caso, se destaca la arquitectura por capas, un esquema altamente utilizado durante muchos años. Esta arquitectura se enfoca en organizar el software mediante un conjunto de capas que interactúan entre sí de forma consecutiva. Este enfoque permite la popularización, la reutilización y facilita el mantenimiento futuro del software. Otra de las características importantes es que esta arquitectura puede adaptarse o configurarse con el fin de cumplir los requerimientos del cliente, y este punto de modificar la arquitectura mencionada es un estándar actual. Bass, Clements y Kazman (2021).

Es importante considerar aspectos de arquitecturas más modernas, como la arquitectura orientada a microservicios, que utiliza tecnologías como las API RESTful, las cuales son una característica clave en este esquema ampliamente utilizado en la actualidad. "El uso de las API RESTful permite e incluso fomenta una distribución más rápida de nuevas funciones y actualizaciones" (Alamilla Hernández, Pérez Romero, Sosa González, & Valentín Rodríguez, 2021, p. 5). Al tomar en cuenta estos aspectos de la arquitectura por capas y las ventajas de una API RESTful, se diseña una arquitectura de la siguiente manera:

**Figura 12***Arquitectura del sistema**Fuente. Autoría propia*

## Diseño Interfaz grafica


El diseño de la interfaz gráfica desempeña un papel fundamental, ya que influye directamente en la experiencia del usuario (UX) y en la facilidad de uso de la aplicación. Una interfaz intuitiva y eficiente permite a los usuarios interactuar con el sistema de manera efectiva, reduciendo la curva de aprendizaje y mejorando la productividad en la gestión de los recursos disponibles. Para lograr este objetivo, es esencial aplicar principios de UX y diseño de interfaz de usuario (UI), asegurando que la plataforma sea accesible, atractiva y funcional.

En la presente fase se expone el proceso de diseño de la interfaz del software de gestión de recursos, abordando los principios de UX/UI aplicados, la metodología utilizada y las ventajas

que ofrece Figma en la conceptualización y validación del diseño, además, se destacan los criterios empleados para garantizar una interfaz accesible e intuitiva.

## Figura 13

### *Inicio de sesión*



Logo Universidad

**Bienvenido al Modulo de Prestamo**  
Por favor Diligenciar sus credenciales

Funcionario

Usuario

Codigo

Contraseña

The image shows a login interface for a university library system. It features a red vertical bar on the left. The main content area is titled 'Logo Universidad' and 'Bienvenido al Modulo de Prestamo'. Below the title, there is a prompt 'Por favor Diligenciar sus credenciales' and a radio button selected for 'Funcionario'. There are three input fields for 'Usuario', 'Codigo', and 'Contraseña'. At the bottom, there are two buttons: 'Iniciar Sesion' (red) and 'Iniciar Google' (grey). To the right of the input fields is an illustration of a person interacting with a computer monitor, surrounded by gears, folders, and a plant.

*Fuente. Autoría propia*

Figura 14

## Gestión de reservas

SIIUB / Modulo Reservas

SIIUB / Home Hamilton / Docente

Bienvenido Hamilton Lopez

Eventos + Nueva Reserva +

**Reservas Registradas**

Reservado Por:	N° Elementos	Fecha Reserva	Estado	
Hamilton Lopez Villamizar	3	Lunes 13 de Octubre 2024	Reservado	Detalles Editar Eliminar
Hamilton Lopez Villamizar	1	Lunes 13 de Octubre 2024	Entregado	Detalles Editar Eliminar
Hamilton Lopez Villamizar	4	Lunes 13 de Octubre 2024	Reservado	Detalles Editar Eliminar
Hamilton Lopez Villamizar	2	Lunes 13 de Octubre 2024	En Reserva	Detalles Editar Eliminar
Hamilton Lopez Villamizar	3	Lunes 13 de Octubre 2024	Entregado	Detalles Editar Eliminar
Hamilton Lopez Villamizar	1	Lunes 13 de Octubre 2024	Reservado	Detalles Editar Eliminar
Hamilton Lopez Villamizar	3	Lunes 13 de Octubre 2024	En Reserva	Detalles Editar Eliminar

Fuente. Autoría propia

Figura 15

## Registro Reservas

SIIUB / Modulo Reservas

SIIUB / Home Hamilton / Docente

Bienvenido Hamilton Lopez

Consultar Reservas

**Registrar Reserva**

Dependencia: Reservado a:

Periodo Academico: Periodo Academico

Reservado a: Reservado por:

Funcionario: Evento: Estudiante: Estado:

Fecha Reserva: Hoy Lunes 10 de Octubre 2024

L 10 M 11 M 12 J 13 V 13 S 13

**Listado Elementos Reserva:** Novedades Agregar Recurso +

Elemento:	Fecha Inicio Reserva	Estado	
Computador Dell Inspiron 83400	Lunes 13 de Octubre 2024	Reservado	Detalles Editar Eliminar
Video Beam pson 3402	Lunes 13 de Octubre 2024	Reservado	Detalles Editar Eliminar

Fuente. Autoría propia

**Figura 16***Registro eventos*

SIIUB / Modulo Reservas

Hamilton / Docente

Consultar Reservas

Registrar Reserva

Periodo Academico:

Nombre:

Fecha Inicio:  Fecha Fin:

Funcionario Responsable:

Guardar

Dependencia:

Nombre	Fecha Inicio	Fecha Fin	Editar	Eliminar
Simpesin economica	15/10/2024	15/10/2024	Editar	Eliminar
Conservatorias Finanzas	20/10/2024	20/10/2024	Editar	Eliminar

Cerrar

Reservado Por:

Estado:

Agregar Recurso +

Editar Eliminar

Editar Eliminar

Editar Eliminar

*Fuente. Autoría propia***Figura 17***Panel Administrador*

SIIUB / Modulo Reservas

Hamilton / Administrador

Bienvenido Hamilton Lopez

Elementos Prestados: 20

Elementos Reservados: 30

Elementos Retrasados: 20

Gestion Elementos

Gestion Reservas

Gestion Reportes

Notificaciones

Proximo a Entregar  
Recurso: Computador Dell 54023  
Fecha inicio reserva: 20/10/2024 30:00 am

Retrasado  
Recurso: Computador Dell 54023  
Fecha inicio reserva: 20/10/2024 30:00 am

Proximo a Entregar  
Recurso: Computador Dell 54023  
Fecha inicio reserva: 20/10/2024 30:00 am

Proximo a Entregar  
Recurso: Computador Dell 54023  
Fecha inicio reserva: 20/10/2024 30:00 am

Proximo a Entregar  
Recurso: Computador Dell 54023  
Fecha inicio reserva: 20/10/2024 30:00 am

Proximo a Entregar  
Recurso: Computador Dell 54023  
Fecha inicio reserva: 20/10/2024 30:00 am

*Fuente. Autoría propia*

**Figura 18**  
*Gestión Recursos*

SIIUB / Home SIIUB / Modulo Reservas  
Hamilton / Administrador

**Recursos Registrados** Migración Nuevo Recurso +

Recursos Registrados - Division de tecnologia.

Recurso	Tipo Elemento	Fecha Registro	Estado
Computador DELL 6548	Computador	Lunes 13 de Octubre 2024	Inactivo
Computador DELL 6548	Computador	Lunes 13 de Octubre 2024	Activo
VideoBeam 5846	Video Beam	Lunes 13 de Octubre 2024	Inactivo
Cable HDMI	Cable	Lunes 13 de Octubre 2024	Activo
Computador DELL 5689	Computador	Lunes 13 de Octubre 2024	Activo
Cable HDMI	Cable	Lunes 13 de Octubre 2024	Inactivo
Computador DELL 5689	Computador	Lunes 13 de Octubre 2024	Activo

Fuente. Autoría propia

**Figura 19**  
*Registro Recursos*

SIIUB / Home SIIUB / Modulo Reservas  
Hamilton / Administrador

**Registro de Recurso** Gestion Estados Tipo Elemento Consultar Recursos

Registrar nuevo recurso - Division de Tecnologia

Tipo Elemento: Seleccionar Estado: Seleccionar Fecha Registro: Fecha Registro

Descripcion General

Disponibilidad: Novedades Caracteristicas

Disponible  Seleccionado

Periodo Academico: Subseccional

Dia Semana: Seleccionar Hora Inicio: Seleccionar Hora Fin: Seleccionar

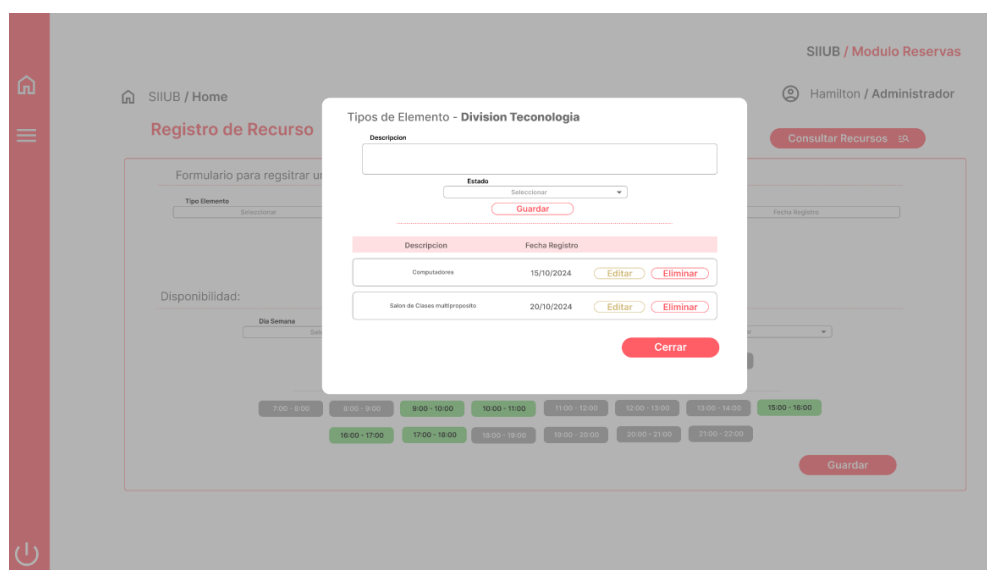
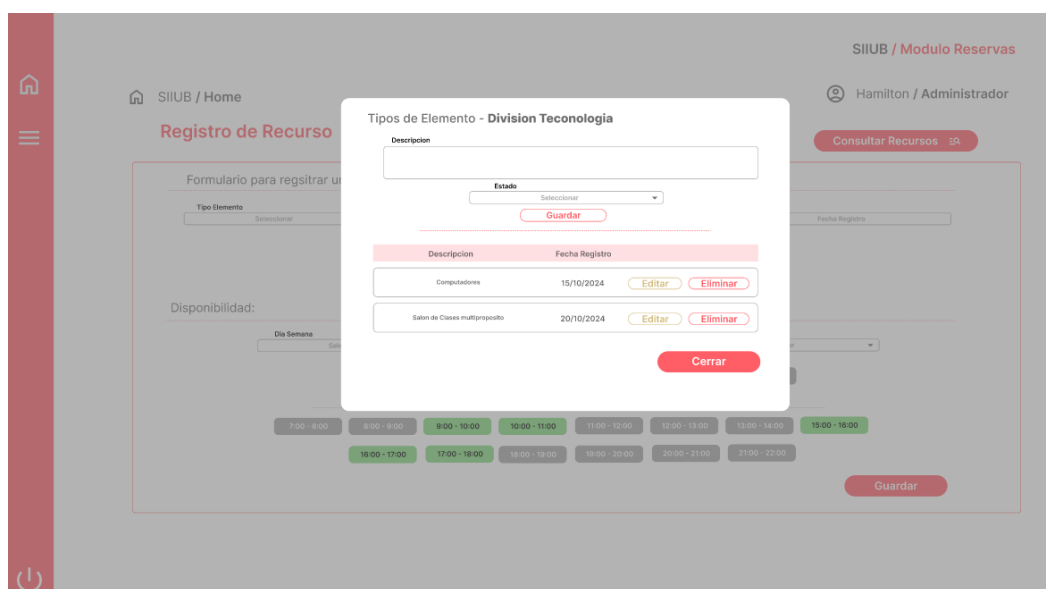
Lunes  Martes  Miércoles  Jueves  Viernes  Sabado

Horas Disponibilidad Lunes

7:00 - 7:30	7:30 - 8:00	8:00 - 8:30	8:30 - 9:00	9:00 - 9:30	9:30 - 10:00	10:00 - 10:30	10:30 - 11:00	11:00 - 11:30	11:30 - 12:00
12:00 - 12:30	12:30 - 13:00	13:00 - 13:30	13:30 - 14:00	14:00 - 14:30	14:30 - 15:00	15:00 - 15:30	15:30 - 16:00	16:00 - 16:30	16:30 - 17:00

Guardar

Fuente. Autoría propia

**Figura 20***Gestión Estados**Fuente. Autoría propia***Figura 21***Gestión tipos de Recursos**Fuente. Autoría propia*

**Figura 22***Gestión Novedades*

SIUB / Modulo Reservas

Hamilton / Administrador

Consultar Recursos

SIUB / Home

Registro de Recurso

Formulario para registrar un recurso

Tipo Elemento: Seleccionar

Fecha Registro:

Disponibilidad:

Di Semana:

Horas Disponibilidad Lunes

7:00 - 8:00 8:00 - 9:00 9:00 - 10:00 10:00 - 11:00 11:00 - 12:00 12:00 - 13:00 13:00 - 14:00 15:00 - 16:00 16:00 - 17:00 17:00 - 18:00 18:00 - 19:00 19:00 - 20:00 20:00 - 21:00 21:00 - 22:00

Guardar

**Novedad Recurso: Computador DELL 4589**

Descripción Novedad

Guardar

Descripción	Fecha Registro	Editar	Eliminar
Se entrega equipo con el cable de cargador en mal estado, el funcionario entrega en Hamilton mencionando.	15/10/2024	Editar	Eliminar
Se entrega equipo con el cable de cargador en mal estado, el funcionario entrega en Hamilton mencionando.	20/10/2024	Editar	Eliminar

Cerrar

*Fuente. Autoría propia***Figura 23***Gestión de características*

SIUB / Modulo Reservas

Hamilton / Administrador

Consultar Recursos

SIUB / Home

Registro de Recurso

Formulario para registrar un recurso

Tipo Elemento: Seleccionar

Fecha Registro:

Disponibilidad:

Di Semana:

Horas Disponibilidad Lunes

7:00 - 8:00 8:00 - 9:00 9:00 - 10:00 10:00 - 11:00 11:00 - 12:00 12:00 - 13:00 13:00 - 14:00 15:00 - 16:00 16:00 - 17:00 17:00 - 18:00 18:00 - 19:00 19:00 - 20:00 20:00 - 21:00 21:00 - 22:00

Guardar

**Características: Computador DELL 4589**

Nombre: Descripción / Valor

Estado: Seleccionar

Guardar

Nombre	Valor	Estado	Editar	Eliminar
Numero de Sillas	250	Activo	Editar	Eliminar
Tarjeta Grafica	Invidia 3050 ti	Activo	Editar	Eliminar
Cargador Batería	Propio del recurso	Inactivo	Editar	Eliminar

Cerrar

*Fuente. Autoría propia*

## Figura 24

### Gestión Reportes



Fuente. Autoría propia

## Desarrollo

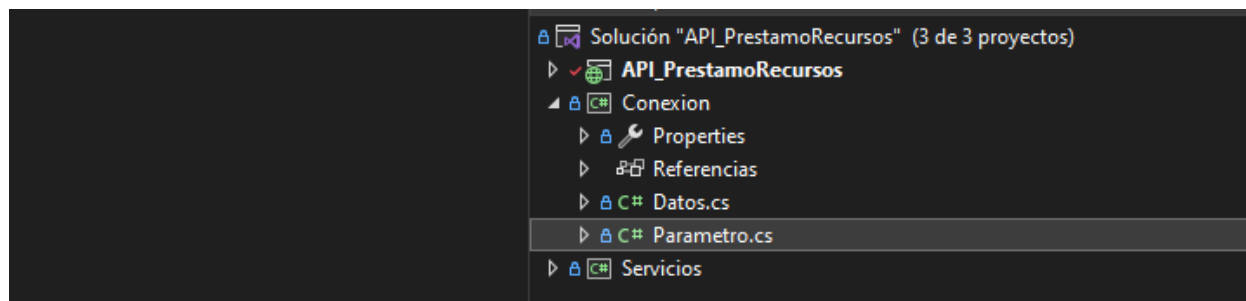
### Desarrollo Back-End

El desarrollo de software requiere una estructura bien definida que permita la integración eficiente entre sus componentes, en este contexto, el backend desempeña un papel fundamental al gestionar la lógica de negocio, el acceso a los datos y la comunicación con el frontend; Para este proyecto, se ha implementado una API construida sobre .NET framework, estructurada en tres capas: conexión, servicio y API, con el objetivo de optimizar la gestión de datos y mejorar la escalabilidad del sistema.

La primera capa, denominada de conexión, se encarga de establecer la comunicación con la base de datos, mediante el consumo de librerías propias de Oracle, esto permite mejorar el rendimiento y garantizar la seguridad en el acceso a la información.

### Figura 25

#### *Capa Conexión - API*



*Fuente.* Autoría propia

**Figura 26**

*Método conexión a base de datos*

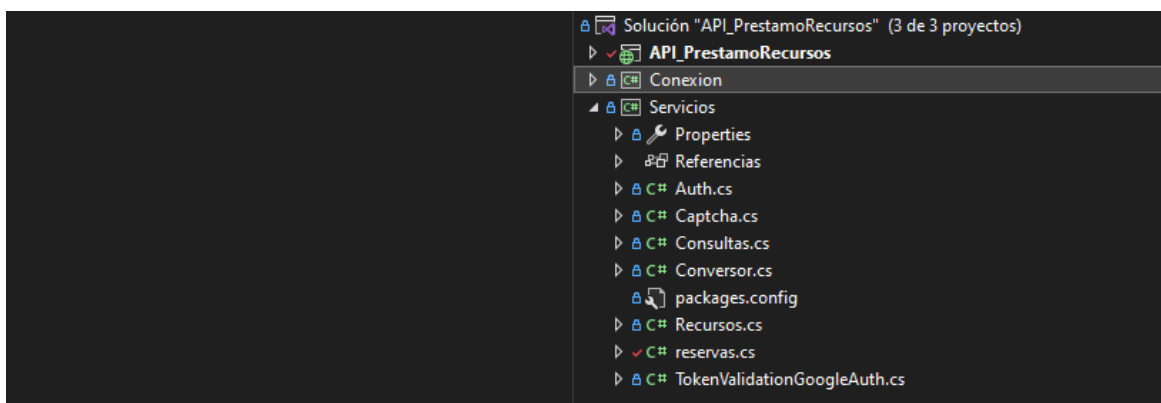
```
57  
58 #region ObtenerTabla  
59 0 referencias  
60 public System.Data.DataTable obtenerTabla(String nomStoredProcedure, List<Parametro> lsParametros)...  
108  
109 #endregion  
110  
111  
112  
113 #region obtenerCampo  
114 0 referencias  
115 public string obtenerCampo(String nomStoredProcedure, List<Parametro> lsParametros, string nomCampo)...  
174  
175 #endregion  
176
```

*Fuente. Autoría propia*

Posteriormente, la capa de servicios obtiene los datos recuperados desde la base de datos, utilizando procedimientos almacenados en Oracle, los analiza y transforma según los requerimientos del sistema, asegurando que la información procesada sea coherente y estructurada.

**Figura 27**

*Capa servicios - API*



*Fuente. Autoría propia*

**Figura 28***Método consulta procedimientos*

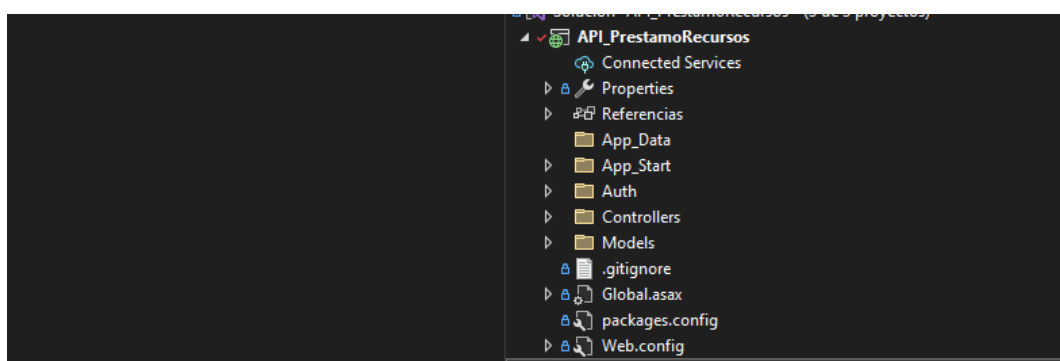
```

73  /// <summary>
74  /// Clase que permite consultar procedimiento almacenados y consultar basicas
75  /// </summary>
76  2 referencias
77  public class Consultas
78  {
79
80      Parametro parametroSP = new Parametro();
81      List<Parametro> listaParametro = new List<Parametro>();
82      Datos conexion = new Datos();
83
84
85
86
87  /// <summary>
88  /// Metodo que consume procedimiento almacenado para consultar las dependencias segun el usuario que consulta
89  /// </summary>
90  /// <param name="idUserio">id del usuario/</param>
91  /// <returns>Datatable con el listado de dependencias/</returns>
92  0 referencias
93  public DataTable ListadoDependencias(string idUsuario)
94  {
95      // Crear la lista de parámetros para enviar al procedimiento almacenado
96      List<Parametro> parametros = new List<Parametro>
97      {
98          // Parámetro de entrada (ID del usuario)
99          parametroSP.crearParametro("p_idUsuario", ParameterDirection.Input, 0, idUsuario, OracleDbType.Int32),
100
101          // Parámetro de salida para el cursor con los datos
102          parametroSP.crearParametro("p_cursor", ParameterDirection.Output, 0, null, OracleDbType.RefCursor)
103      };
104
105      // Llamar al método obtenerTabla con el nombre del procedimiento almacenado
106      return conexion.obtenerTabla("CONSULTAR_DEPENDENCIAS_USUARIO", parametros);
107  }
108

```

*Fuente.* Autoría propia

Finalmente, la capa API actúa como intermediaria entre el backend y el frontend, proporcionando los datos de manera eficiente y segura a través de endpoints definidos, estos transforman los datos a un formato JSON, y es enviado para ser presentados al usuario.

**Figura 29***Capa API - API*

*Fuente.* Autoría propia

**Figura 30**

*End Point o controlador desde el que se conecta el Frontend*

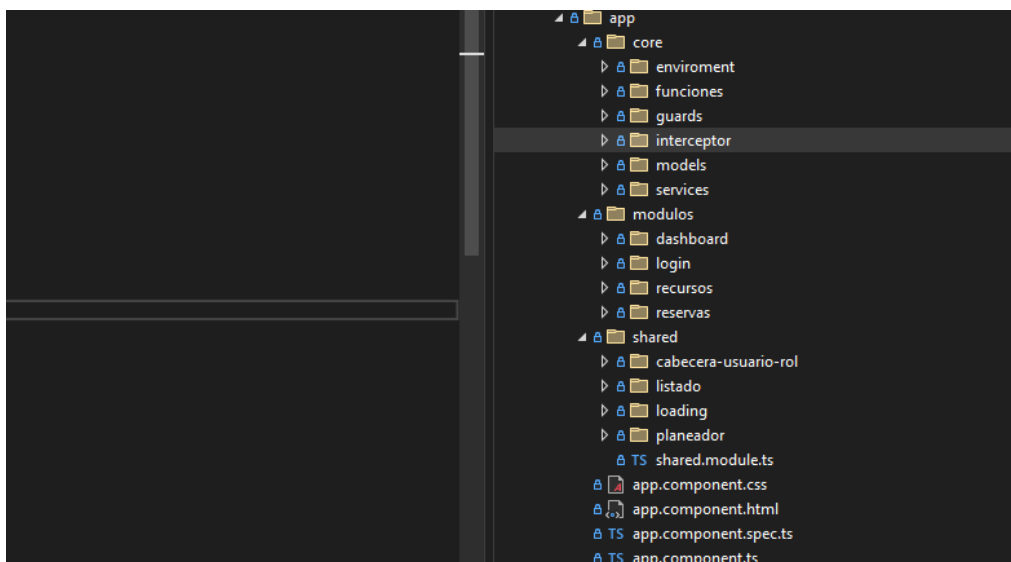
```
43 // <summary>
44 // Metodo que permite consultar las dependencias registradas
45 // </summary>
46 [HttpGet]
47 Dependencias
48 public async Task<IActionResult> consultarDependencias()
49 {
50     try
51     {
52         // se consultan las dependencias
53         DataTable Dependencias = _recursos.CrearDependencia();
54         if (Dependencias != null)
55         {
56             return Ok(Dependencias);
57         }
58         return BadRequest("Se ha presentado un error al consultar las dependencias registradas. Por favor, intente nuevamente más tarde.");
59     }
60     catch
61     {
62         return BadRequest("Se ha presentado un error al consultar las dependencias registradas. Por favor, intente nuevamente más tarde.");
63     }
64 }
65
```

*Fuente. Autoría propia*

## Desarrollo Front-End

**Figura 31**

*Estructura del proyecto*

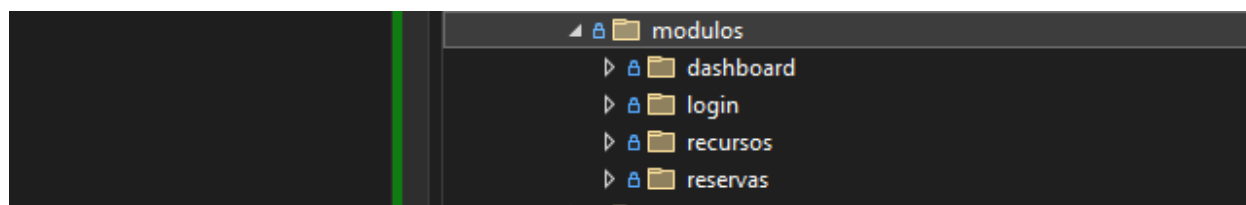


*Fuente. Autoría propia*

El desarrollo del frontend de este proyecto se ha llevado a cabo utilizando Angular, una de las tecnologías más destacadas en el desarrollo de aplicaciones web debido a su excelente desempeño, modularidad y facilidad de mantenimiento, la elección de Angular permite una estructura organizada del código, facilitando la creación de múltiples módulos para gestionar de manera eficiente las diferentes funcionalidades del sistema.

## Figura 32

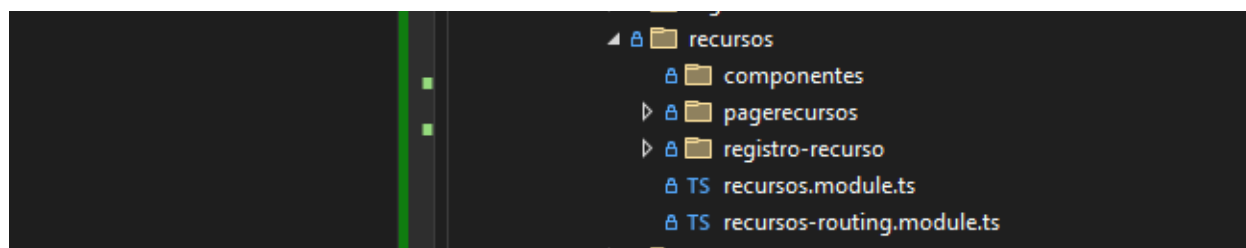
### *Módulos Front*



*Fuente.* Autoría propia

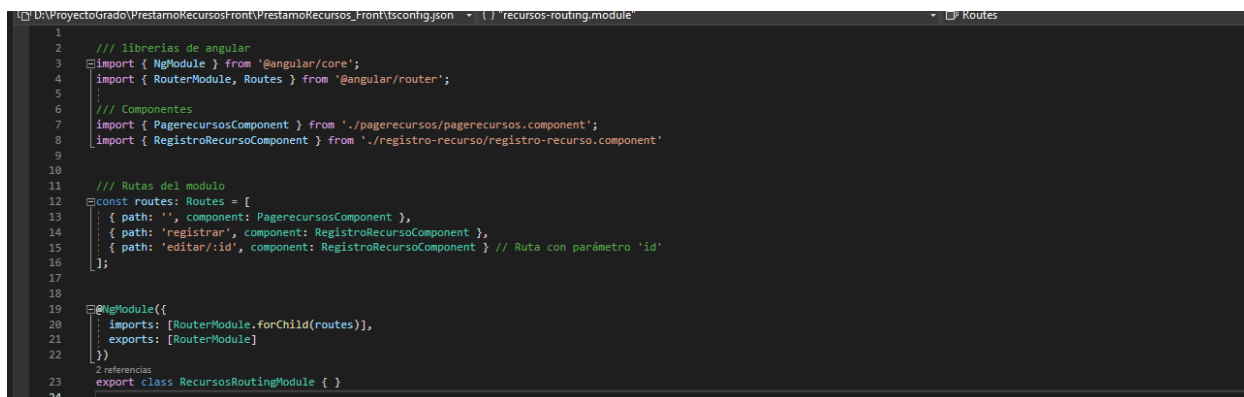
## Figura 33

### *Estructura de cada modulo*



*Fuente.* Autoría propia

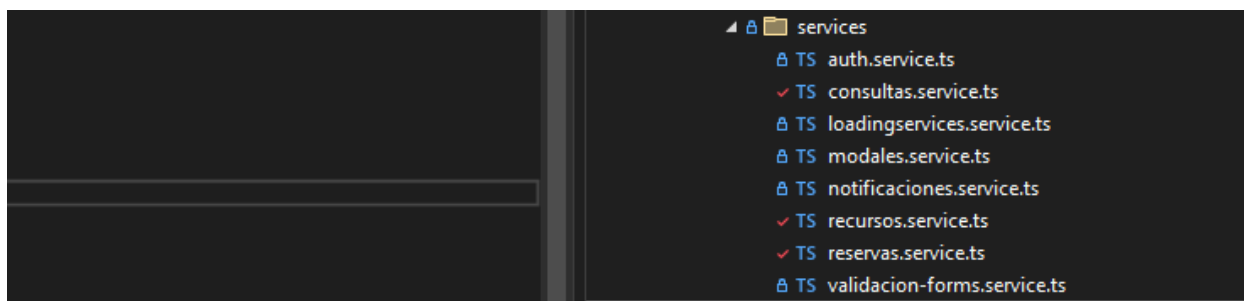
Cada módulo del frontend está diseñado para una gestión específica dentro del sistema, como la gestión de las rutas dentro de cada módulo, lo que favorece la escalabilidad y reutilización del código.

**Figura 34***Rutas de modulo*

```
1
2 // librerías de angular
3 import { NgModule } from '@angular/core';
4 import { RouterModule, Routes } from '@angular/router';
5
6 // Componentes
7 import { PagerecursosComponent } from '../pagerecursos/pagerecursos.component';
8 import { RegistroRecursoComponent } from '../registro-recurso/registro-recurso.component';
9
10 // Rutas del modulo
11
12 const routes: Routes = [
13   { path: '', component: PagerecursosComponent },
14   { path: 'registrar', component: RegistroRecursoComponent },
15   { path: 'editar/:id', component: RegistroRecursoComponent } // Ruta con parámetro 'id'
16 ];
17
18
19 @NgModule({
20   imports: [RouterModule.forChild(routes)],
21   exports: [RouterModule]
22 })
23 export class RecursosRoutingModule { }
24
```

*Fuente.* Autoría propia

Además, se han desarrollado servicios dedicados exclusivamente al consumo de la API, asegurando una comunicación eficiente y segura entre el frontend y el backend.

**Figura 35***Servicios*

*Fuente.* Autoría propia

Para optimizar la gestión de autenticación y autorización, se han implementado interceptores que permiten la inyección automática de tokens en cada solicitud, garantizando la seguridad en el acceso a los recursos protegidos del sistema, asimismo, se han desarrollado

interceptores adicionales para manejar mensajes de carga, proporcionando una mejor experiencia de usuario

### Figura 36

#### Interceptor

```
13
14 @Injectable()
15 export class LoadingInterceptor implements HttpInterceptor {
16
17
18   0 referencias
19   constructor(private loadingService: LoadingservicesService, private router: Router, private notificaciones: NotificacionesService) { }
20
21   intercept(req: HttpRequest<any>, next: HttpHandler): Observable<HttpEvent<any>> {
22
23     // Obtiene el token JWT desde el servicio de autenticación
24     const datos = sessionStorage.getItem('data');
25     const mantener = sessionStorage.getItem('Mantener');
26
27     if (datos && mantener)
28     {
29       const parsedData = JSON.parse(datos)
30       if (parsedData.Token) {
31         req = req.clone({
32           setHeaders: {
33             Authorization: `Bearer ${parsedData.Token}`,
34             'X-Custom-Bool': mantener // Agregar un nuevo parámetro booleano
35           },
36         });
37       }
38     }
39   }
```

*Fuente.* Autoría propia

En cuanto a la presentación y diseño del sistema, se ha utilizado Bootstrap como framework principal para garantizar una distribución responsiva y estilizada de los componentes. Gracias a Bootstrap, el sistema cuenta con un diseño adaptable a diferentes dispositivos, asegurando una interfaz intuitiva y accesible para los usuarios.

Figura 37

## Inicio de sesión

UB Universidad de Boyacá

Prestamo Recursos / SIUB

10 February 2025

**Módulo de préstamos de recursos**

Por favor, diligencie sus credenciales.

Seleccione primero si es un funcionario o no. Luego, diligencie el formulario o acceda mediante su cuenta institucional ingresando a través de Google. Los datos de acceso corresponden al usuario y la contraseña del correo electrónico institucional, sin especificar @uniboyaca.edu.co

Funcionario

Usuario

Código

Contraseña

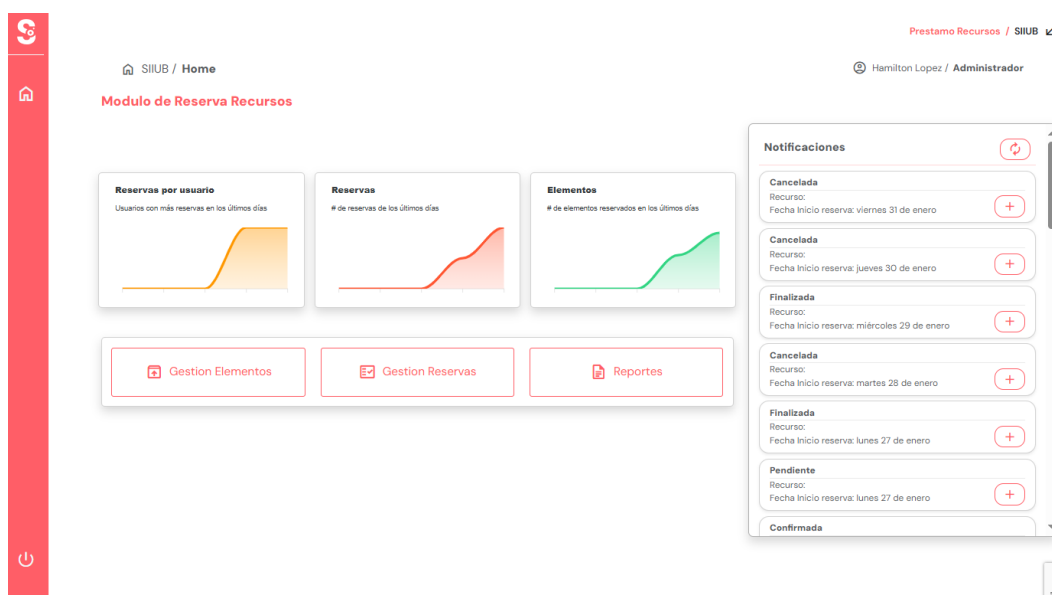
Mantener sesión iniciada

Iniciar sesión con Google

Fuente. Autoría propia

Figura 38

## Panel De Control



Fuente. Autoría propia

Figura 39

## Gestión de recursos

The screenshot displays the 'Recursos' management interface. On the left, a vertical red sidebar contains a home icon and a user profile icon. The main content area is titled 'Recursos' and includes a welcome message for 'Hamilton Lopez'. Below this is a table of 'Recursos registrados' with columns for Description, Registration Date, Status, and Resource Type. Each row includes 'Detalles', 'Editar', and 'Eliminar' buttons. The table lists six resources, including 'PRUEBA DOS NOVEDADES', 'Salon Edificio 10', 'CASOFA', 'ASDFACA', 'NOVEDADS', and 'COMPUTADOR'. To the right, a 'Detalles Reserva' panel shows the details for a reservation with description 'PRUEBA DOS NOVEDADES', registration date '27/01/2025', status 'A', and resource type 'Espacios Físicos'. It also includes 'Editar' and 'Eliminar' buttons and a section for 'Características del recurso' with fields for 'Marca' (DELL) and 'Estado' (A).

Fuente. Autoría propia

Figura 40

## Registro recursos parte 1

The screenshot shows the 'Registro de Recursos' form. The interface includes a vertical red sidebar on the left. The main area is titled 'Registro de Recursos' and features navigation buttons for 'Tipo elementos', 'Gestion Estados', and 'Consultar Recursos'. The form itself is titled 'Registrar Recurso - Division Tecnologia' and includes a '+ Nuevo Recurso' button. It contains several input fields: 'Periodo académico' (dropdown), 'Tipo Recurso' (dropdown), 'Estado' (dropdown set to 'ACTIVO'), and 'Fecha registro' (calendar icon). A 'Descripción' field is also present. Below the form, there are buttons for 'Características' and 'Novedades', and a 'Siguiente' button with a right-pointing arrow. A note at the bottom of the form states: 'A continuación, debe registrar las características del recurso que desea crear. Además, podrá añadir novedades relacionadas con el recurso, como posibles daños o detalles adicionales.'

Fuente. Autoría propia

Figura 41

## Registro recurso parte 2

The screenshot shows the 'Registro de Recursos' web application. The main form is titled 'Registrar Disponibilidad' and includes the following elements:

- Navigation:** A red sidebar on the left with a home icon and a power icon. The top right shows the user 'Hamilton Lopez / Administrador' and the page title 'Prestamo Recursos / SIUB'.
- Form Fields:**
  - 'Día de la semana': A dropdown menu with '--- Seleccionar ---'.
  - 'Hora Inicio': A dropdown menu with '--- Seleccionar ---'.
  - 'Hora Final': A dropdown menu with '--- Seleccionar ---'.
- Days of the Week:** A row of buttons for 'Lunes', 'Martes', 'Miércoles', 'Jueves', 'Viernes', and 'Sabado'. 'Lunes' is currently selected.
- Hours of Availability:** A grid of buttons for various time slots on 'Lunes':
 

06:00 - 6:30	06:30 - 7:00	07:00 - 7:30	07:30 - 8:00	08:00 - 8:30	08:30 - 9:00	09:00 - 9:30	09:30 - 10:00
10:00 - 10:30	10:30 - 11:00	11:00 - 11:30	11:30 - 12:00	12:00 - 12:30	12:30 - 13:00	13:00 - 13:30	13:30 - 14:00
14:00 - 14:30	14:30 - 15:00	15:00 - 15:30	15:30 - 16:00	16:00 - 16:30	16:30 - 17:00	17:00 - 17:30	17:30 - 18:00
18:00 - 18:30							
- Buttons:** 'Nuevo Recurso', 'Registrar Recurso', and 'Regresar'.

Fuente. Autoría propia

Figura 42

## Gestión de tipos de recurso

The screenshot shows the 'Registro de Recursos' web application with a modal window titled 'Tipos de recursos' open. The modal contains the following information:

- Header:** 'Tipos de recursos' with a close button (X).
- Description:** 'A continuación, podrá gestionar los tipos de recurso que se pueden registrar para la dependencia a la que pertenece, por ejemplo: equipos de cómputo, salones de clase, etc.'
- Form Fields:**
  - 'Descripción': A text input field.
  - 'Estado': A dropdown menu with 'Activo' selected.
- Buttons:** '+ Nuevo', 'Guardar', 'Cerrar'.
- Table:** A table listing existing resource types:
 

Descripción	Fecha Registro	Editar	Eliminar
Espacios Fisicos	24/01/2025	Editar	Eliminar
Equipos de computo	19/01/2025	Editar	Eliminar

Fuente. Autoría propia

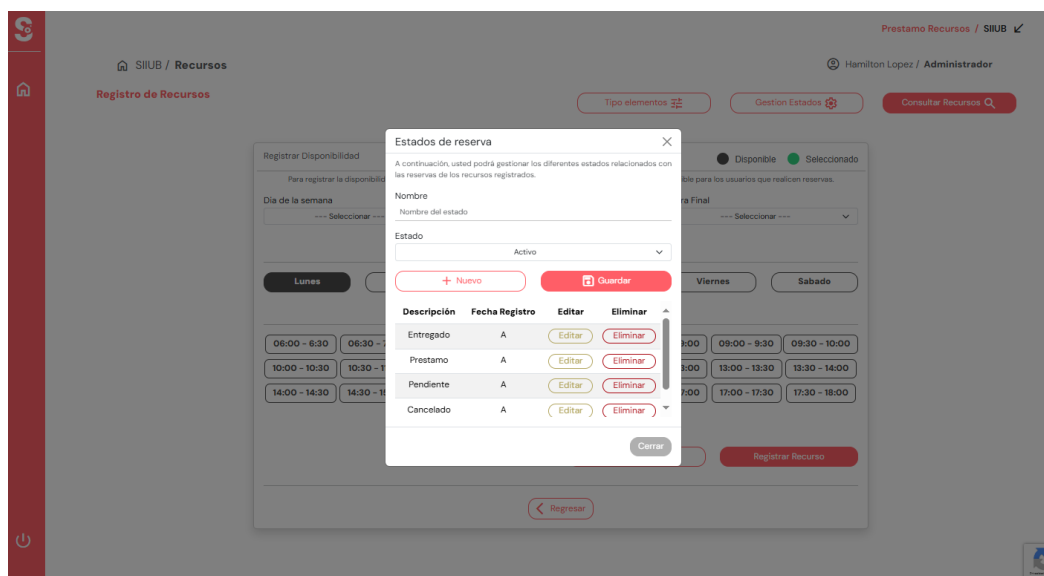
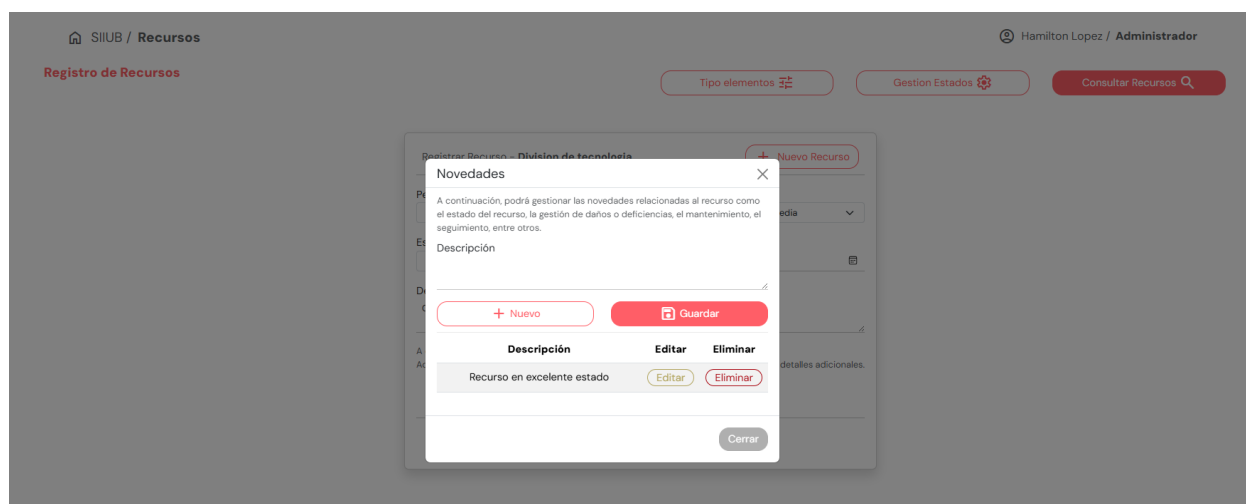
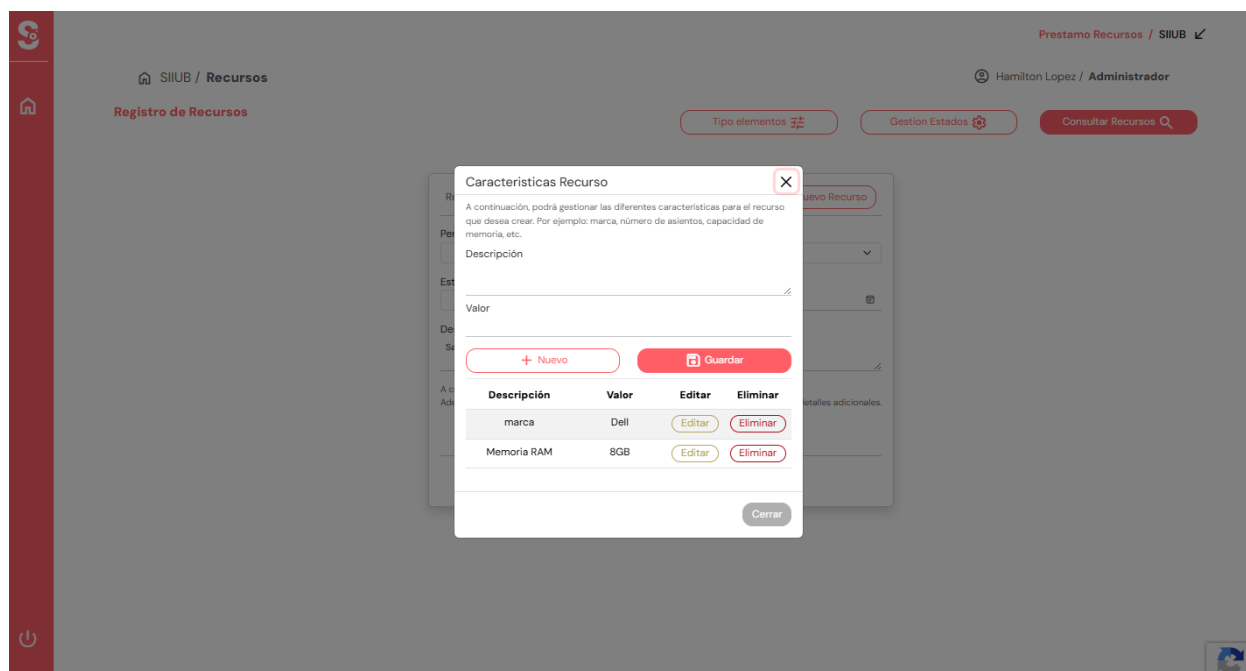
**Figura 43***Gestión estado reservas**Fuente. Autoría propia***Figura 44***Gestión novedades**Fuente. Autoría propia*

Figura 45

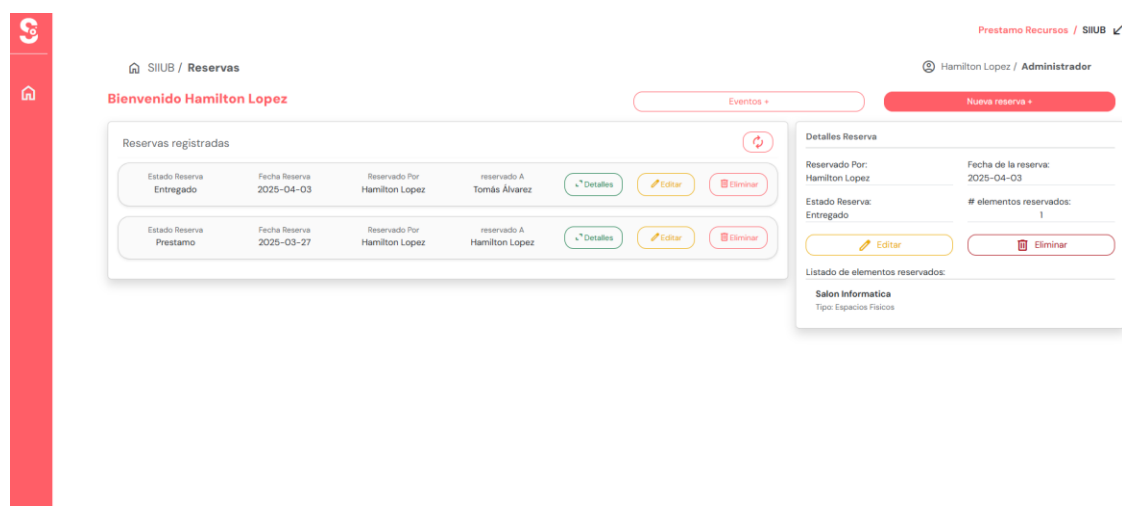
## Gestión características recurso



Fuente. Autoría propia

Figura 46

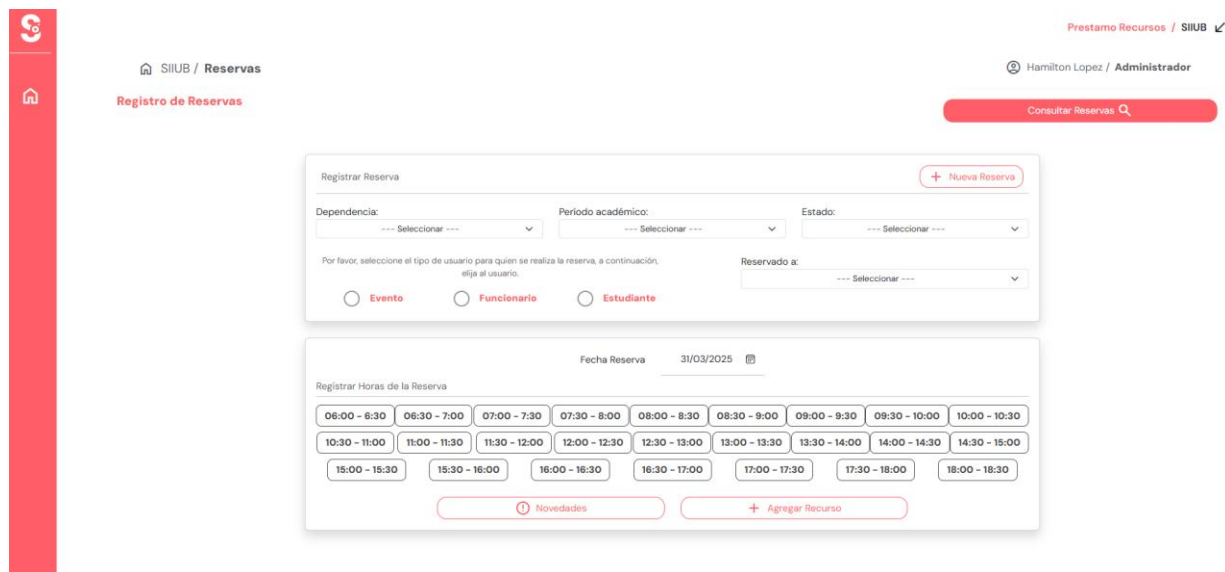
## Gestión reservas



Fuente. Autoría propia

Figura 47

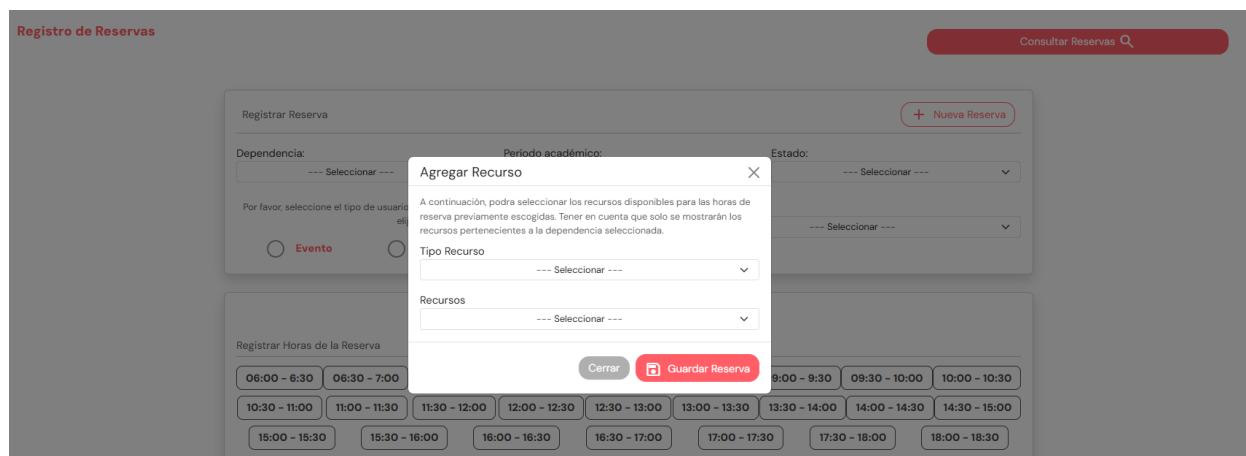
Registrar reserva



Fuente. Autoría propia

Figura 48

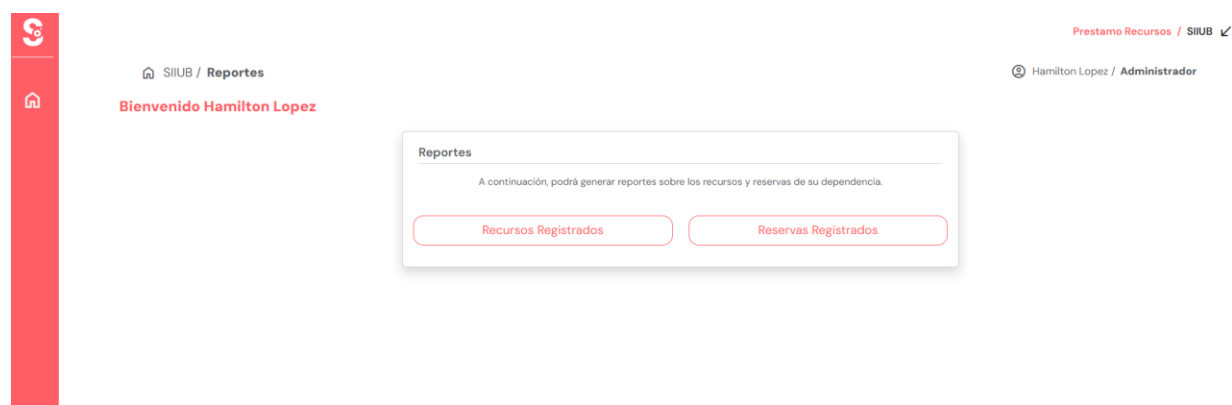
Gestión recursos reserva



Fuente. Autoría propia

## Figura 49

### Gestión reportes



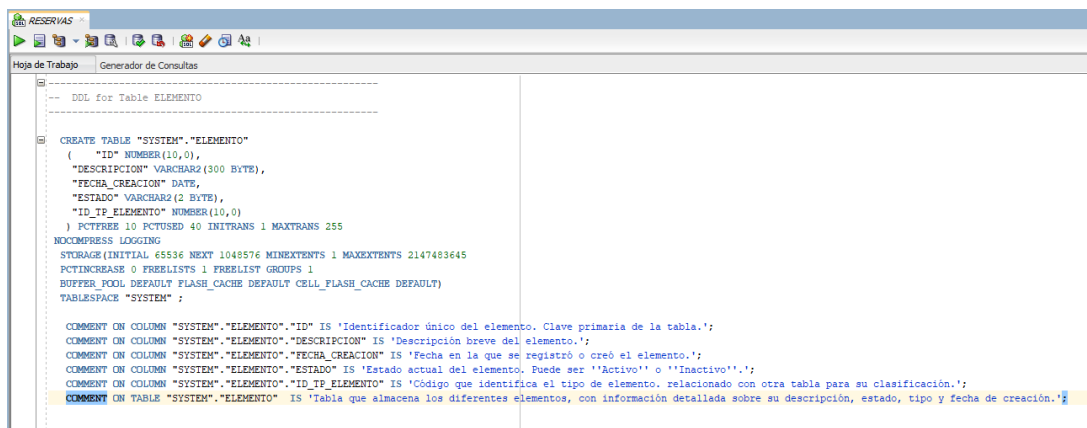
*Fuente.* Autoría propia

## Base de datos

Para el desarrollo de la base de datos, se siguen los requisitos del cliente, en este caso, utilizando el servicio de base de datos ORACLE. Primero, se crean las tablas del esquema, luego, se establecen las restricciones (constraints) para cada una de las tablas, con el fin de garantizar la integridad referencial, finalmente, se desarrollan los procedimientos almacenados que consume la API para la consulta de datos; Estos procedimientos son los encargados de una gran parte de la lógica del sistema.

Figura 50

## Registro de tabla



```

-- DDL for Table ELEMENTO
CREATE TABLE "SYSTEM"."ELEMENTO"
(
  "ID" NUMBER(10,0),
  "DESCRIPCION" VARCHAR2(300 BYTE),
  "FECHA_CREACION" DATE,
  "ESTADO" VARCHAR2(2 BYTE),
  "ID_TP_ELEMENTO" NUMBER(10,0)
) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255
NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "SYSTEM" ;

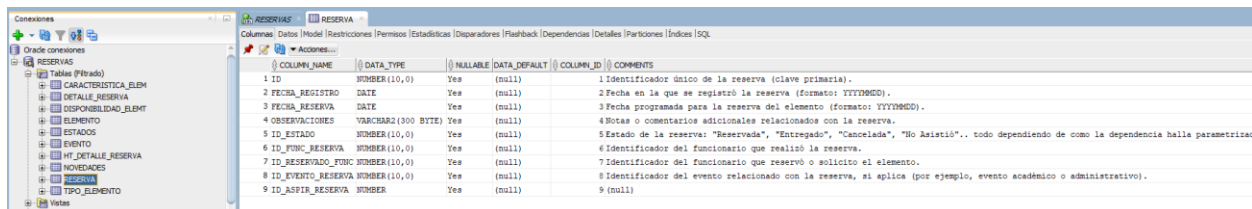
COMMENT ON COLUMN "SYSTEM"."ELEMENTO"."ID" IS 'Identificador único del elemento. Clave primaria de la tabla.';
COMMENT ON COLUMN "SYSTEM"."ELEMENTO"."DESCRIPCION" IS 'Descripción breve del elemento.';
COMMENT ON COLUMN "SYSTEM"."ELEMENTO"."FECHA_CREACION" IS 'Fecha en la que se registró o creó el elemento.';
COMMENT ON COLUMN "SYSTEM"."ELEMENTO"."ESTADO" IS 'Estado actual del elemento. Puede ser 'Activo' o 'Inactivo'.';
COMMENT ON COLUMN "SYSTEM"."ELEMENTO"."ID_TP_ELEMENTO" IS 'Código que identifica el tipo de elemento, relacionado con otra tabla para su clasificación.';
COMMENT ON TABLE "SYSTEM"."ELEMENTO" IS 'Tabla que almacena los diferentes elementos, con información detallada sobre su descripción, estado, tipo y fecha de creación.';

```

Fuente. Autoría propia

Figura 51

## Tabla del esquema

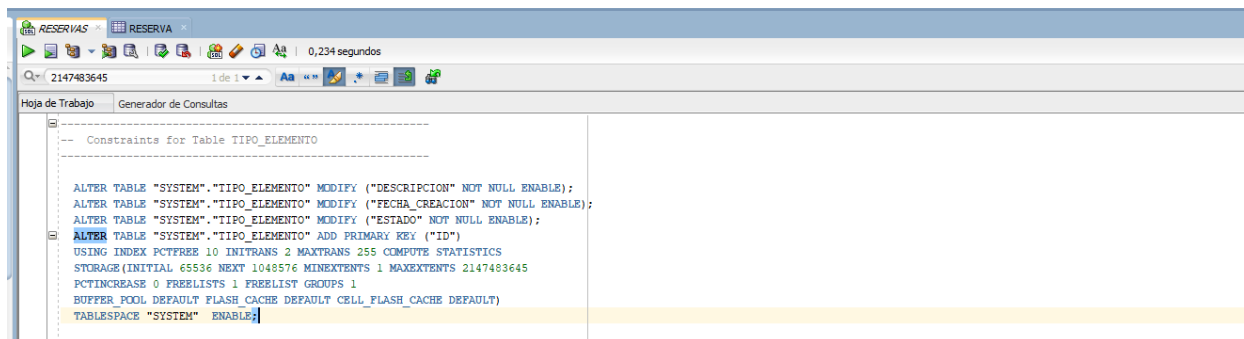


COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID	NUMBER(10,0)	Yes	(null)		1 Identificador único de la reserva (clave primaria).
2 FECHA_REGISTRO	DATE	Yes	(null)		2 Fecha en la que se registró la reserva (formato: YYYYMMDD).
3 FECHA_RESERVA	DATE	Yes	(null)		3 Fecha programada para la reserva del elemento (formato: YYYYMMDD).
4 OBSERVACIONES	VARCHAR2(300 BYTE)	Yes	(null)		4 Notas o comentarios adicionales relacionados con la reserva.
5 ID_ESTADO	NUMBER(10,0)	Yes	(null)		5 Estado de la reserva: 'Reservada', 'Entregado', 'Cancelada', 'No Asistió'.. todo dependiendo de como la dependencia halla parametrizada
6 ID_FUNC_RESERVA	NUMBER(10,0)	Yes	(null)		6 Identificador del funcionario que realizó la reserva.
7 ID_RESERVADO_FUNC	NUMBER(10,0)	Yes	(null)		7 Identificador del funcionario que reservó o solicitó el elemento.
8 ID_EVENTO_RESERVA	NUMBER(10,0)	Yes	(null)		8 Identificador del evento relacionado con la reserva, si aplica (por ejemplo, evento académico o administrativo).
9 ID_ASPIR_RESERVA	NUMBER	Yes	(null)		9 (null)

Fuente. Autoría propia

Figura 52

## Registro de "constraints"



```

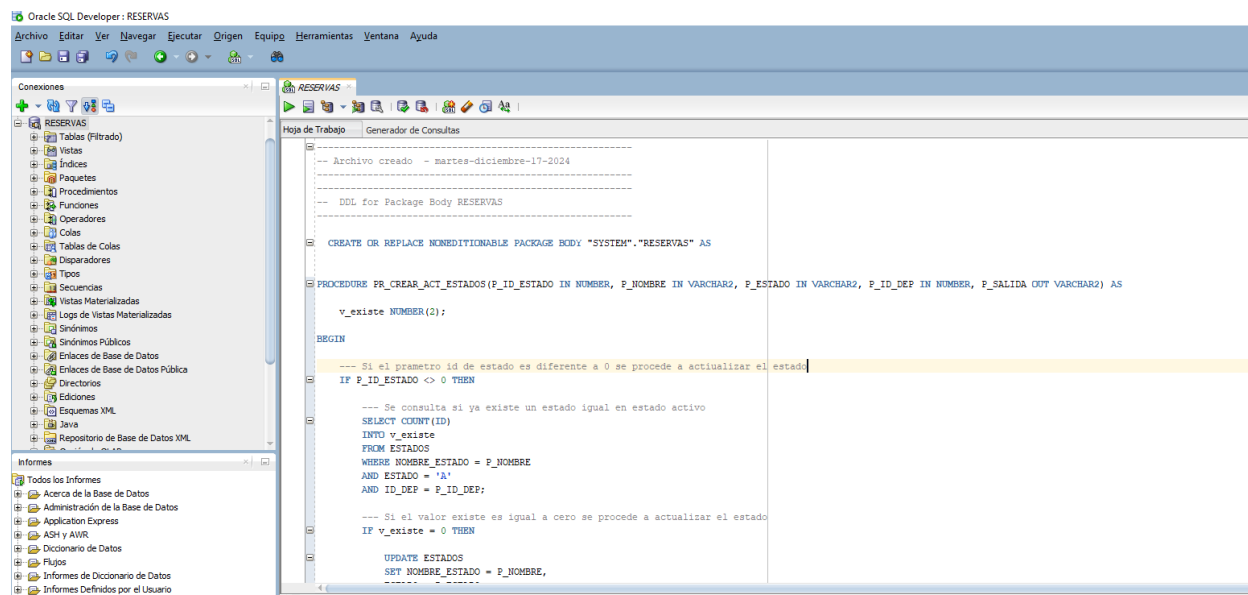
-- Constraints for Table TIPO_ELEMENTO
ALTER TABLE "SYSTEM"."TIPO_ELEMENTO" MODIFY ("DESCRIPCION" NOT NULL ENABLE);
ALTER TABLE "SYSTEM"."TIPO_ELEMENTO" MODIFY ("FECHA_CREACION" NOT NULL ENABLE);
ALTER TABLE "SYSTEM"."TIPO_ELEMENTO" MODIFY ("ESTADO" NOT NULL ENABLE);
ALTER TABLE "SYSTEM"."TIPO_ELEMENTO" ADD PRIMARY KEY ("ID")
USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "SYSTEM" ENABLE;

```



Figura 55

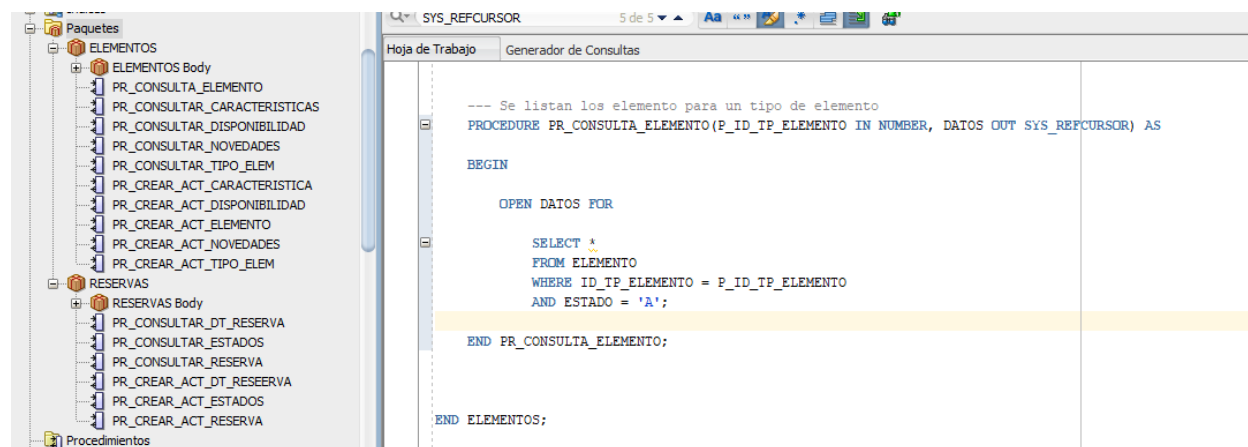
Registro de paquete con procedimientos almacenados



Fuente. Autoría propia

Figura 56

Listado paquete con procedimientos almacenados



Fuente. Autoría propia

## Pruebas

Para el desarrollo de este proyecto de software, se ha decidido utilizar pruebas de caja negra, ya que permiten evaluar el sistema desde la perspectiva del usuario final, verificando que cumpla con los requisitos y funcionalidades esperadas sin necesidad de conocer su código interno, este enfoque nos brinda una visión amplia sobre el comportamiento del software en diferentes escenarios, asegurando que los resultados sean consistentes y satisfactorios para el usuario, además de garantizar la calidad y usabilidad del producto final.

**Tabla 11**  
*Prueba acceso al modulo*

ID Prueba	PR-001
Descripción	Iniciar sesión con el rol de funcionario al sistema.
Precondiciones	El usuario debe estar registrado dentro del sistema con un rol específico para poder acceder.
Pasos Que Seguir	<ul style="list-style-type: none"> <li>• Dar check en la opción funcionario</li> <li>• Ingresa usuario y contraseña del usuario</li> <li>• Opcionalmente seleccionar si desea que la sesión se mantenga iniciada.</li> <li>• Dar clic en el botón iniciar sesión</li> </ul>
Resultado Esperado	Si el usuario esta registrado como usuario con el rol de funcionario se debe permitir acceder al módulo, si es un rol de administrado se redireccionará al panel de administración, si es un funcionario sin administración se cargar las reservas realizadas por el funcionario.
Resultado Obtenido	Se permite acceder al modulo para usuario registrados, además de cargar interfaz según si es administrador o funcionario.
Estado	Aprobado

**Figura 57***Evidencia prueba acceso funcionario*

UB Universidad de Boyacá

02 April 2025

**Módulo de préstamos de recursos**

⚡ **Por favor, diligencie sus credenciales.**

Seleccione primero si es un funcionario o no. Luego, diligencie el formulario o acceda mediante su cuenta institucional ingresando a través de Google. Los datos de acceso corresponden al usuario y la contraseña del correo electrónico institucional, sin especificar @uniboyaca.edu.co

✓ **Funcionario**


hamlopez ✓ @uniboyaca.edu.co

..... ✓

✓ Mantener sesión iniciada

⬇

🔗 Iniciar sesión con Google


*Fuente. Autoría propia***Figura 58***Evidencia prueba acceso funcionario administrativo*

SIUB / Home


Prestamo Recursos / SIUB

Hamilton Lopez / Administrador

**Modulo de Reserva Recursos**


**Elementos**

# de elementos reservados en los últimos días




**Reservas**

# de reservas de los últimos días



**Reservas por usuario**

Usuarios con más reservas en los últimos días



Gestion Elementos

Gestion Reservas

Reportes

**Reservas**

Entregado

Reservado Por: Hamilton Lopez  
Fecha reserva: jueves 03 de abril

Entregado

Reservado Por: Hamilton Lopez  
Fecha reserva: martes 01 de abril

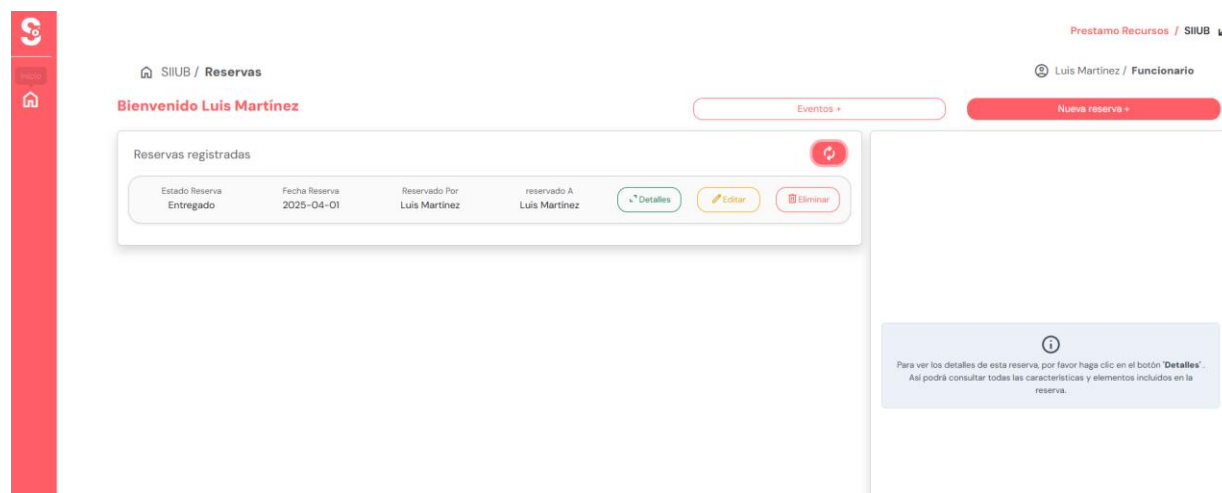
Entregado

Reservado Por: Luis Martinez  
Fecha reserva: martes 01 de abril

Prestamo

Reservado Por: Hamilton Lopez  
Fecha reserva: jueves 27 de marzo

*Fuente. Autoría propia*

**Figura 59***Evidencia prueba acceso rol funcionario**Fuente. Autoría propia***Tabla 12***Prueba acceso rol estudiante*

ID Prueba	PR-002
Descripción	Iniciar sesión con el rol de estudiante al sistema.
Precondiciones	El usuario debe estar registrado dentro del sistema con un rol específico en este caso estudiante para poder acceder.
Pasos Que Seguir	<ul style="list-style-type: none"> <li>• Ingresar usuario, código y contraseña del usuario</li> <li>• Opcionalmente seleccionar si desea que la sesión se mantenga iniciada.</li> <li>• Dar clic en el botón iniciar sesión</li> </ul>
Resultado Esperado	Si el usuario está registrado como usuario con el rol de estudiante se cargará el listado de reservas registradas por el estudiante.

Resultado Obtenido

Se permite acceder al estudiante al módulo para registrar nuevas reservas o consultar las reservas ya realizadas.

Estado

Aprobado

## Figura 60

### Evidencia acceso al modulo con el rol estudiante

UB Universidad de Boyacá

02 April 2025

**Módulo de préstamos de recursos**

Por favor, diligencie sus credenciales.

Seleccione primero si es un funcionario o no. Luego, diligencie el formulario o acceda mediante su cuenta institucional ingresando a través de Google. Los datos de acceso corresponden al usuario y la contraseña del correo electrónico institucional, sin especificar @uniboyaca.edu.co

Funcionario

Usuario Tom01 ✓ @uniboyaca.edu.co

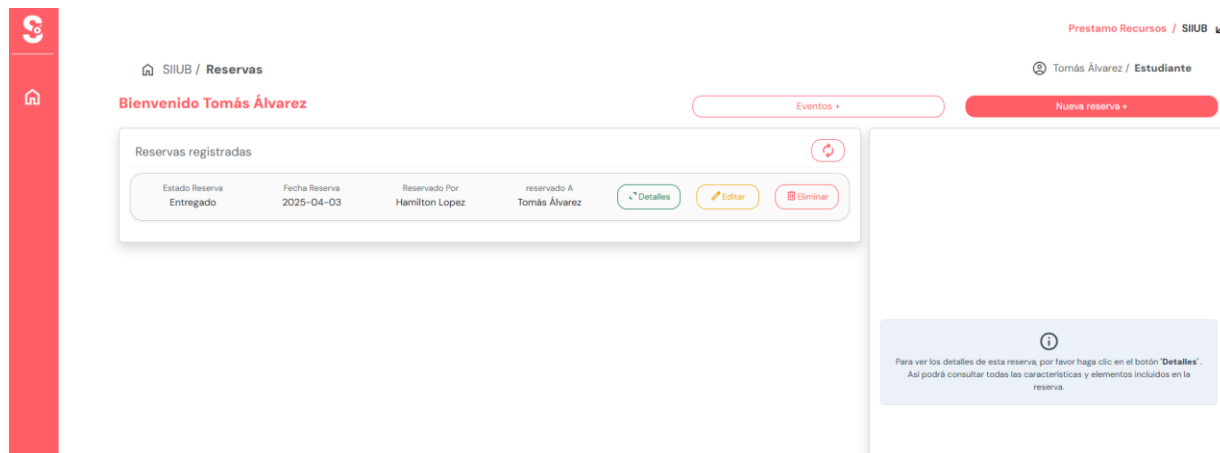
Código 100001 ✓

Contraseña ..... ✓

Mantener sesión iniciada

↓

Fuente. Autoría propia

**Figura 61***Evidencia prueba acceso con rol estudiante II**Fuente. Autoría propia***Tabla 13**  
*Prueba autenticación con Google*

ID Prueba	PR-003
Descripción	Iniciar sesión con la autenticación de Google.
Precondiciones	El usuario debe estar registrado dentro del sistema con un rol específico y debe contar con un correo de Gmail para poder acceder.
Pasos Que Seguir	<ul style="list-style-type: none"> <li>• Seleccionar la opción funcionario si es un funcionario si es estudiante no se selecciona la opción.</li> <li>• Dar clic en acceder como con cuenta Google.</li> <li>• Acceder con las credenciales de Google.</li> <li>• Si el rol es estudiante se debe cargar el Código del estudiante en el ventana modal, y dar clic en acceder.</li> </ul>

Resultado Esperado	Después de colocar el código en el caso de los estudiante y dar clic acceder se enviará al estudiante al listado de reservas realizadas por el usuario, en el caso de los funcionario al diligenciar las credenciales de Google se cargar el panel de gestión para los roles de administrador, para el caso de funcionarios se cargar el listado de reservas realizadas por el funcionario.
Resultado Obtenido	Se logra acceder al módulo de reservas desde la opción de autenticación de Google.
Estado	Aprobado

## Figura 62

### Evidencia acceso mediante opción de Google

**Módulo de préstamos de recursos**

Por favor, diligencie sus credenciales.

Selecciona primero si es un funcionario o no. Luego, diligencie el formulario o acceda mediante su cuenta institucional ingresando a través de Google. Los datos de acceso corresponden al usuario y la contraseña del correo electrónico institucional, sin especificar @uniboyaca.edu.co

Funcionario

Mantener sesión iniciada

Usuario

Código

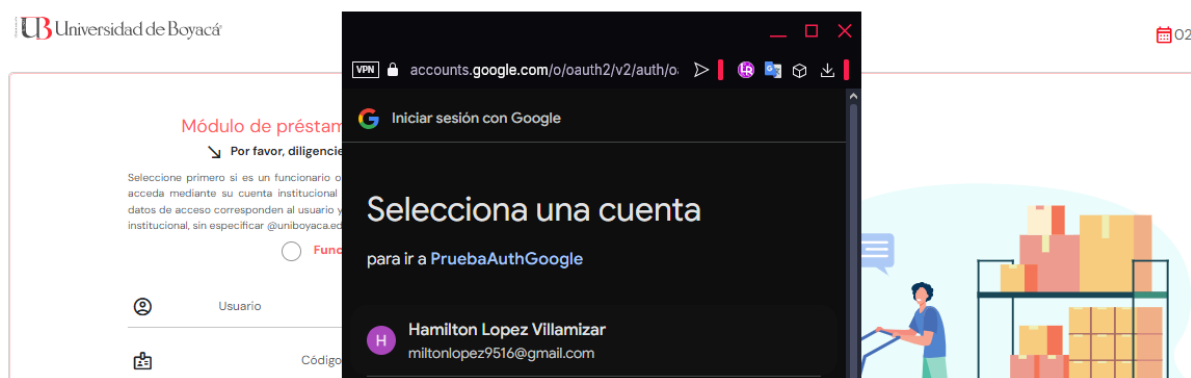
Contraseña

Inicia sesión como Hamilton miltonlopez9516@gmail.com

Fuente. Autoría propia

## Figura 63

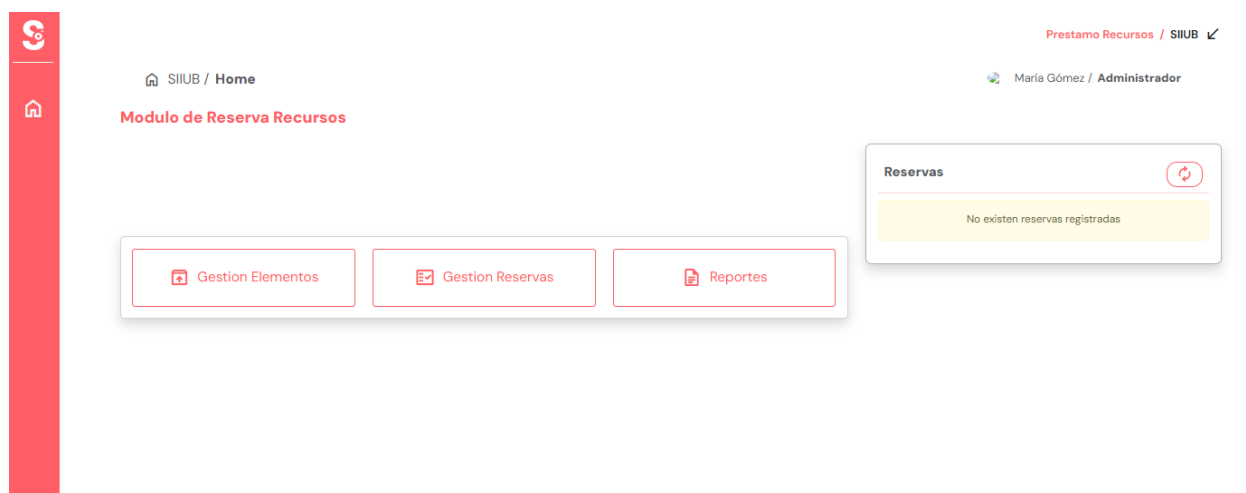
*Evidencia prueba acceso opción Google II*



*Fuente. Autoría propia*

## Figura 64

*Evidencia prueba acceso opción Google (administrativo sin registros)*



*Fuente. Autoría propia*

## Tabla 14

*Prueba registro de tipos de recurso*

ID Prueba	PR-004
Descripción	Gestionar de tipos de recursos

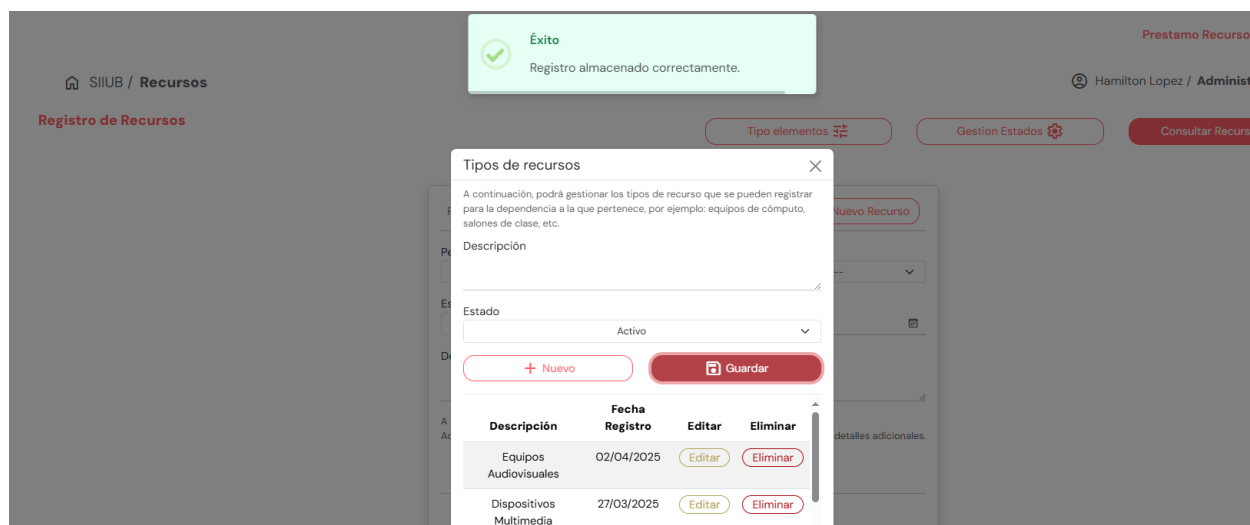
---

Precondiciones	El usuario debe tener rol como administrador de la dependencia a la que pertenece para poder registrar los tipos de recursos de la dependencia.
Pasos Que Seguir	<ul style="list-style-type: none"><li>• Dar clic en el botón Gestión Elementos.</li><li>• Clic botón Nuevo Recurso</li><li>• Clic botón Tipo Elemento</li><li>• Diligenciar descripción, estado del registro en este caso activo.</li><li>• Clic Guardar.</li><li>• Para editar un tipo de recurso dar clic en editar y modificar la descripción o el estado.</li><li>• Para eliminar un tipo de elemento en el botón eliminar.</li></ul>
Resultado Esperado	Cuando se crea un nuevo tipo de elemento y se da clic en guardar se debe actualizar el listado con un nuevo tipo de recurso; cuando se editar los valores en el listado deben cambiar por los registros modificados; una vez eliminado el registro debe desaparecer del listado.
Resultado Obtenido	Se permite el registrar tipos de elementos para las reservas.
Estado	Aprobado

---

## Figura 65

Evidencia gestión de tipos de recursos.



Fuente. Autoría propia

## Tabla 15

Prueba gestión estados reservas

ID Prueba	PR-005
Descripción	Gestionar de estados de reserva
Precondiciones	El usuario debe tener rol como administrador de la dependencia a la que pertenece para poder registrar los estados para las reservas de la dependencia.
Pasos Que Seguir	<ul style="list-style-type: none"> <li>• Dar clic en el botón Gestión Elementos.</li> <li>• Clic botón Nuevo Recurso</li> <li>• Clic botón gestión estados</li> <li>• Diligenciar nombre, estado del registro en este caso activo.</li> <li>• Clic Guardar.</li> <li>• Para editar un estado dar clic en editar y modificar la nombre o el estado.</li> </ul>

- Para eliminar un estado en el botón eliminar.

Resultado Esperado

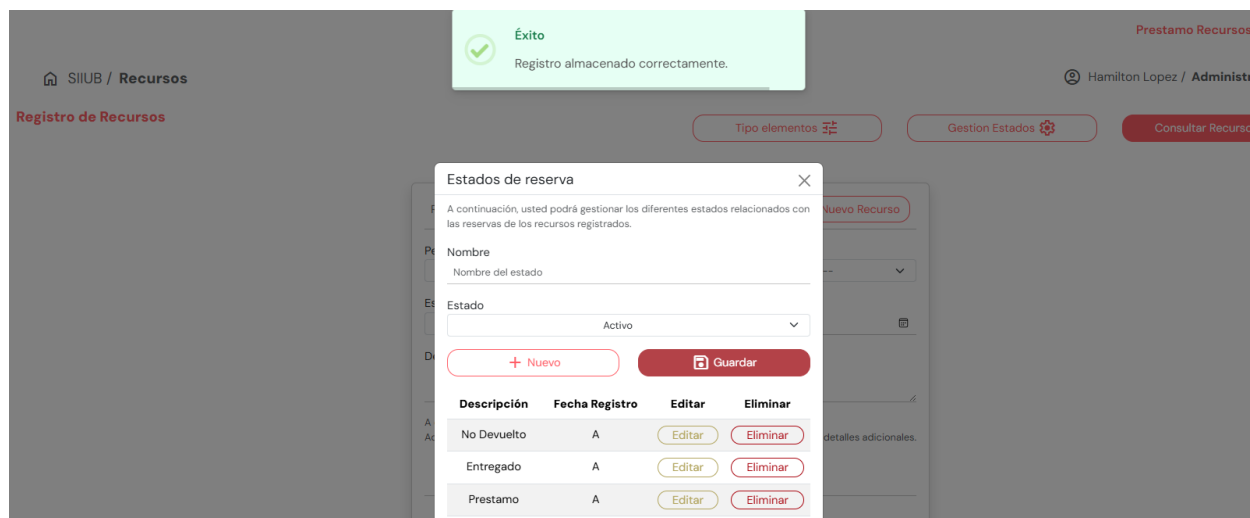
Cuando se crea un nuevo estado y se da clic en guardar se debe actualizar el listado de estado de reserva registrados; si se da clic en editar se cargan los datos del estado en el formulario una vez modificado se da clic en guardar, se actualizará el listado de estados registrados; en el caso de eliminar un registrado se eliminará del listado de estados.

Resultado Obtenido  
Estado

Se permite la gestión correcta de los estado de reserva.  
Aprobado

## Figura 66

*Evidencia prueba gestión estados de reserva.*



*Fuente. Autoría propia*

## Tabla 16

*Prueba registro de recursos*

ID Prueba	PR-006
Descripción	Registrar recursos o elementos con su disponibilidad.

---

Precondiciones	El usuario debe tener rol como administrador de la dependencia a la que pertenece para poder registrar nuevos recursos con su disponibilidad.
Pasos Que Seguir	<ul style="list-style-type: none"><li>• Dar clic en el botón Gestión Elementos.</li><li>• Clic botón Nuevo Recurso</li><li>• Diligenciar primer formulario con el periodo para el que se registra, el tipo de recurso, estado, fecha registro, descripción.</li><li>• Clic en el botón características, agregar las características del recurso.</li><li>• Opcionalmente se podrá registrar novedades al recurso como el posible estado de este.</li><li>• Se procede a registrar las disponibilidad, seleccionando el día de la semana, y las horas en las que estará disponible.</li><li>• Clic en guardar Registrar recurso.</li></ul>
Resultado Esperado	Se permite el registro de recursos para la dependencia los cuales estarán disponibles para reservar, tanto con los días como las horas para las que el recurso estará disponible, una vez registrado se debe listar el nuevo recurso en el listado de recursos.
Resultado Obtenido	Se permite el registro de recursos a reservar para la dependencia.
Estado	Aprobado

---

**Figura 67**

*Evidencia prueba registro recursos información Base*

Registrar Recurso - División de tecnología

Periodo académico: 202520 ✓

Tipo Recurso: Equipos Audiovisuales ✓

Estado: ACTIVO

Fecha registro: 02/04/2025

Descripción: Cámara Digital ✓

A continuación, debe registrar las características del recurso que desea crear. Además, podrá añadir novedades relacionadas con el recurso, como posibles daños o detalles adicionales.

Características Novedades

Siguiente >

*Fuente. Autoría propia*

**Figura 68**

*Evidencia prueba registro recurso – Registro características*

Éxito  
Registro realizado correctamente.

Características Recurso

A continuación, podrá gestionar las diferentes características para el recurso que desea crear. Por ejemplo: marca, número de asientos, capacidad de memoria, etc.

Descripción

Valor

+ Nuevo Guardar




Descripción	Valor	Editar	Eliminar
Marca	CANON	Editar	Eliminar

*Fuente. Autoría propia*

## Figura 69

### Evidencia registro recursos – registro disponibilidad

SIUB / Recursos Hamilton Lopez / Administrador

Registro de Recursos Tipo elementos  Gestion Estados  Consultar Recursos 

**Registrar Disponibilidad** ● Disponible ● Seleccionado

Para registrar la disponibilidad del recurso a crear, seleccione el día de la semana y la hora en que estará disponible para los usuarios que realicen reservas.

Día de la semana:  Hora Inicio:  Hora Final:

Días de la semana disponible:

**Lunes**

Horas disponibilidad día: **Lunes**




06:00 - 6:30	06:30 - 7:00	07:00 - 7:30	07:30 - 8:00	08:00 - 8:30	08:30 - 9:00	09:00 - 9:30	09:30 - 10:00	10:00 - 10:30
10:30 - 11:00	11:00 - 11:30	11:30 - 12:00	12:00 - 12:30	12:30 - 13:00	13:00 - 13:30	13:30 - 14:00	14:00 - 14:30	14:30 - 15:00
15:00 - 15:30	15:30 - 16:00	16:00 - 16:30	16:30 - 17:00	17:00 - 17:30	17:30 - 18:00	18:00 - 18:30		

Fuente. Autoría propia


## Figura 70

### Evidencia registro reservas respuesta

SIUB / Recursos Hamilton Lopez / Administrador

Registro de Recursos Tipo elementos  Gestion Estados  Consultar Recursos 

Registrar Recurso - División de tecnología + Nuevo Recurso



**Éxito**

El registro se ha realizado correctamente.

Fuente. Autoría propia

## Figura 71

### Evidencia registro recurso – Consulta registro creado

The screenshot displays the 'Recursos registrados' section of the SIUB system. It features a table with the following data:

Descripción	Fecha Registro	Estado	Tipo Recurso	Acciones
Camara Digital	2025-04-02	A	Equipos Audiovisuales	Detalles, Editar, Eliminar
Camara de Video	2025-03-29	A	Dispositivos Multimedia	Detalles, Editar, Eliminar
Simulacion de cirugia	2025-04-02	A	Espacios Fisicos	Detalles, Editar, Eliminar
Salon Informatica	2025-03-27	A	Espacios Fisicos	Detalles, Editar, Eliminar

The 'Detalles Reserva' panel on the right shows the following information for the 'Camara Digital' resource:

- Descripción: Camara Digital
- Fecha Registro: 2025-04-02
- Estado: A
- Tipo de recurso: Equipos Audiovisuales
- Características del recurso: Marca CANON, Estado: A

Fuente. Autoría propia

## Tabla 17

### Prueba Registro reservas

ID Prueba	PR-007
Descripción	Registro reservas
Precondiciones	El usuario debe estar registrado como estudiante o funcionario, deben existir recursos registrados para el periodo al que se esta registrando la reserva, los recursos a reservas deben tener disponibilidad dentro de las horas y días de la reserva.
Pasos Que Seguir	<ul style="list-style-type: none"> <li>Desde el rol de administrador, se debe dar clic en Gestionar Reserva.</li> <li>Dar clic en el botón Nueva Reserva.</li> <li>Diligenciar primera parte del formulario de reserva, dependencia, periodo académico, estado que tendrá la reserva, a quien va a ser registrado, en el rol de administrador, podrá registrar para evento, funcionarios o estudiantes, para los otros roles no se tendrá esta opción.</li> </ul>

- Seleccionar la fecha de reserva,  
seleccionar las horas de la reserva.

Resultado Esperado

Se debe permitir registrar en la reserva recursos que estand entro del día de la reserva seleccionado, y dentro de las horas seleccionado, de lo contrario no se cargar listado de recursos, una vez creado la reserva se debe listado actualizado el listado con los recursos creados.

Resultado Obtenido

Se permite el registro de reservas para el usuario que se accede, o para el funcionario, evento o estudiante que se halla seleccionado desde el rol de administrador.

Estado

Aprobado

## Figura 72

*Evidencia prueba registro reserva rol funcionario o estudiante.*

Registrar Reserva

Dependencia: Division de tecnologia | Periodo académico: 202520 | Estado: Prestamo

Fecha Reserva: 02/04/2025

Registrar Horas de la Reserva

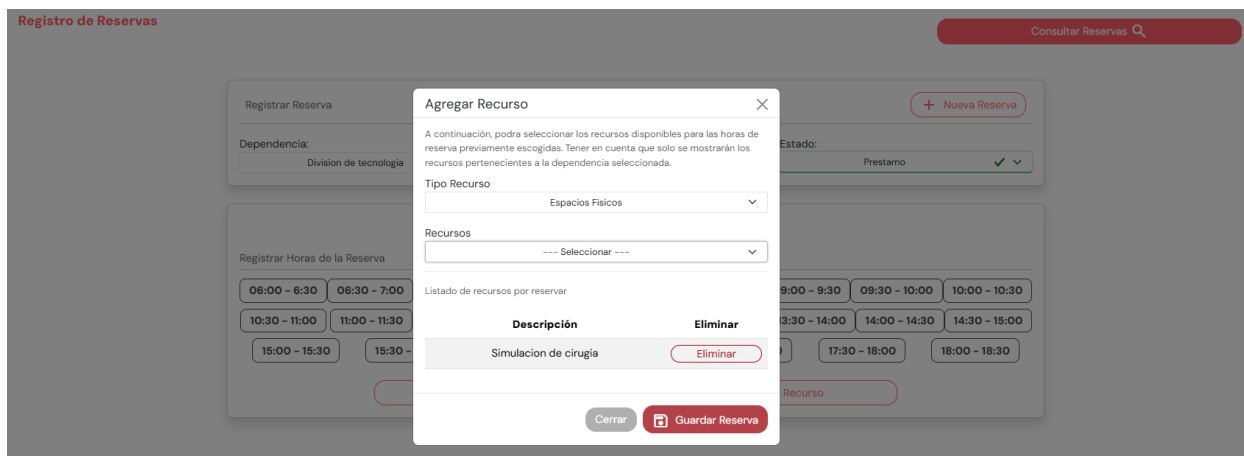
06:00 - 6:30	06:30 - 7:00	07:00 - 7:30	07:30 - 8:00	08:00 - 8:30	08:30 - 9:00	09:00 - 9:30	09:30 - 10:00	10:00 - 10:30
10:30 - 11:00	11:00 - 11:30	11:30 - 12:00	12:00 - 12:30	12:30 - 13:00	13:00 - 13:30	13:30 - 14:00	14:00 - 14:30	14:30 - 15:00
15:00 - 15:30	15:30 - 16:00	16:00 - 16:30	16:30 - 17:00	17:00 - 17:30	17:30 - 18:00	18:00 - 18:30		

+ Agregar Recurso

*Fuente. Autoría propia*

**Figura 73**

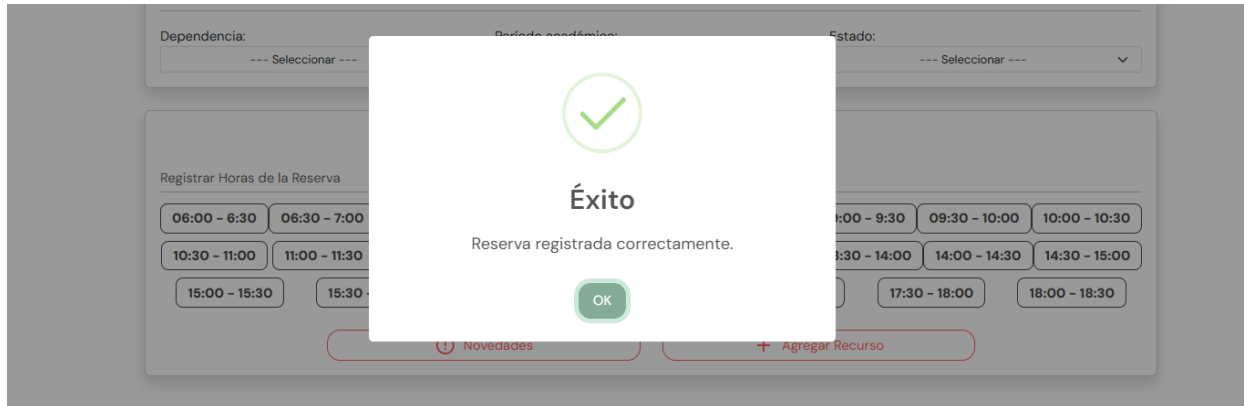
*Evidencia registro reserva rol funcionario o estudiante II*



*Fuente. Autoría propia*

**Figura 74**

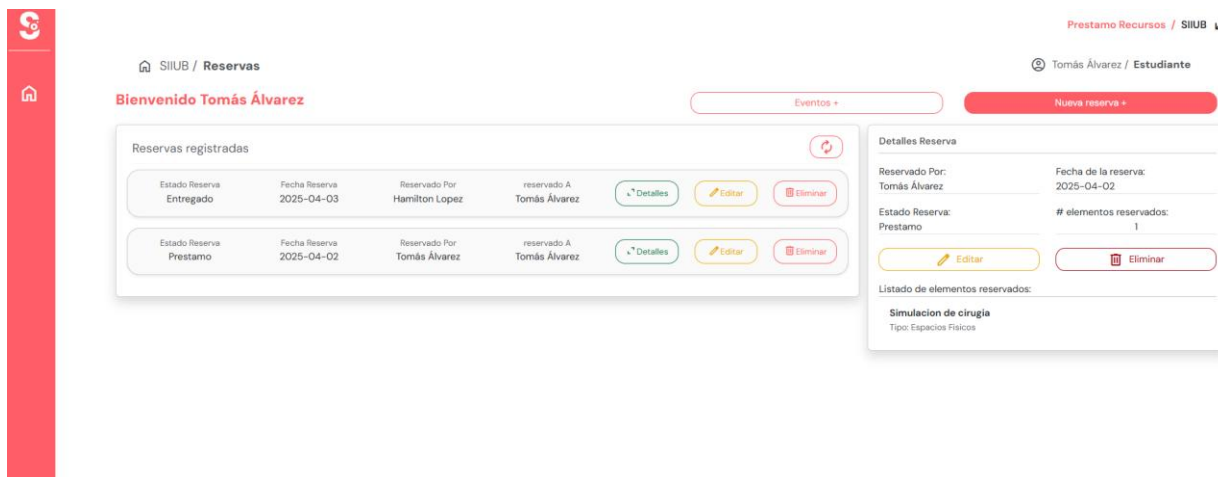
*Evidencia registro reserva rol funcionario o estudiante III*



*Fuente. Autoría propia*

**Figura 75**

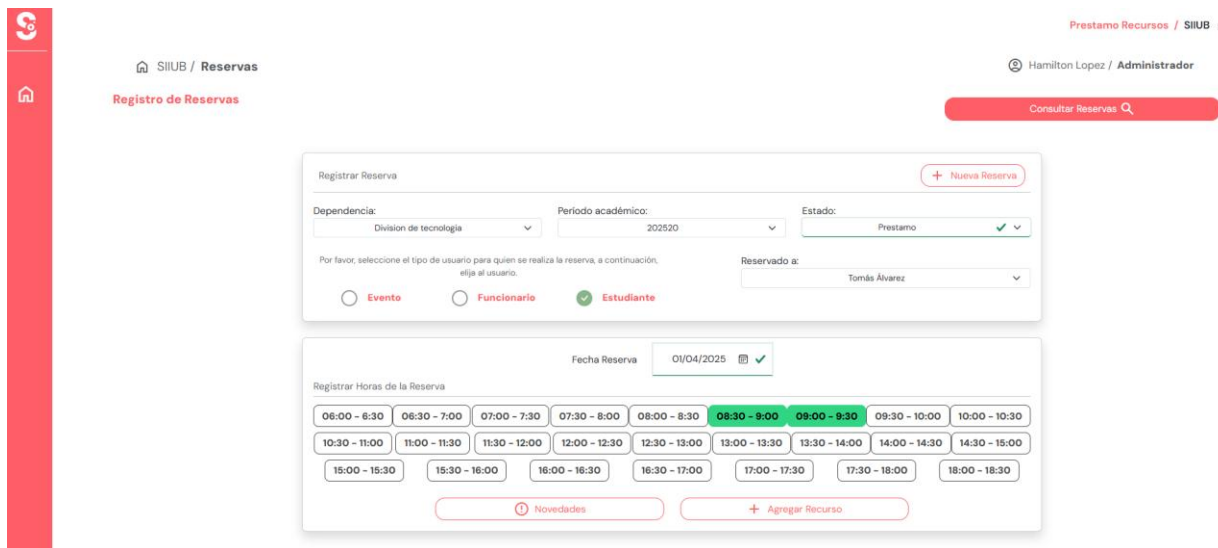
*Evidencia registro reserva rol funcionario o estudiante III*



Fuente. Autoría propia

**Figura 76**

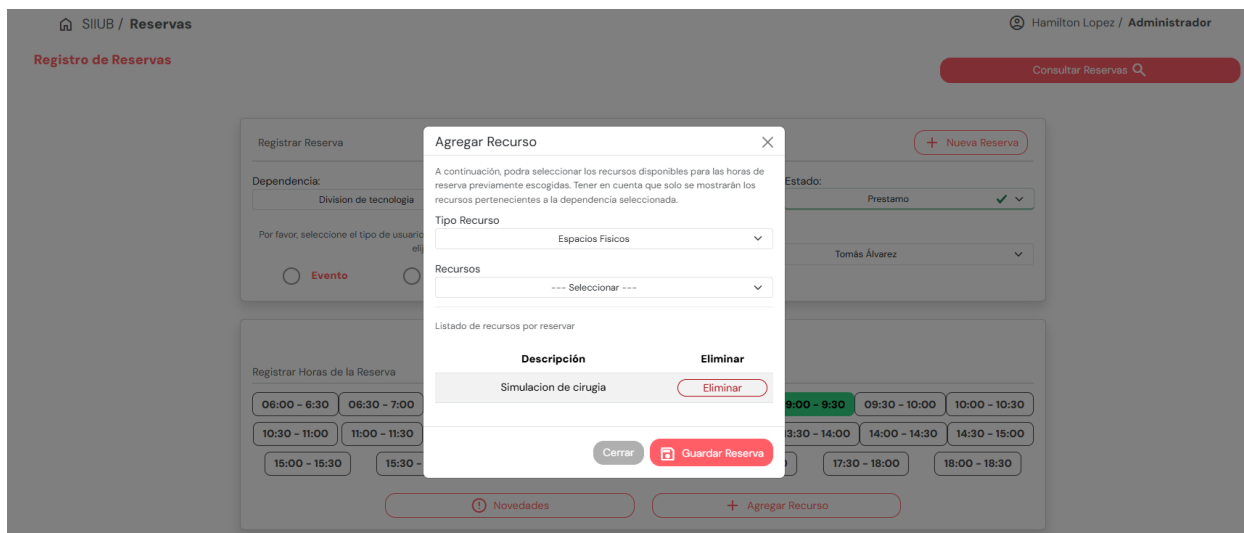
*Evidencia prueba registro reserva rol Administrador I*



Fuente. Autoría propia

**Figura 77**

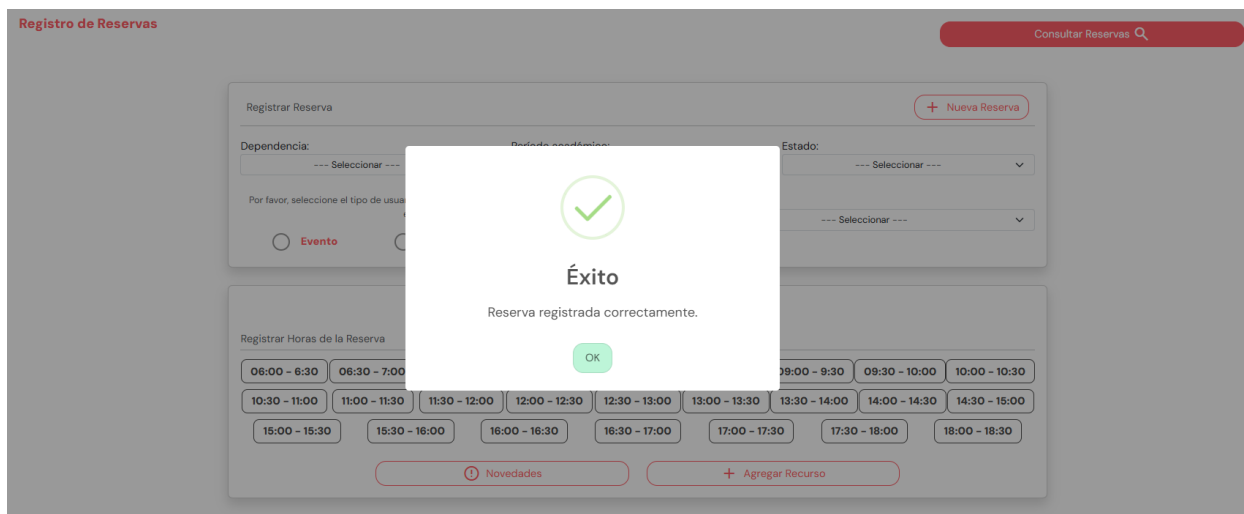
*Evidencia prueba registro reserva rol Administrador II*



*Fuente. Autoría propia*

**Figura 78**

*Evidencia prueba registro reserva rol Administrador III*



*Fuente. Autoría propia*

Figura 79

Evidencia prueba registro reserva rol Administrador IV

The screenshot displays the 'Reservas' section of the SIUB system. The main area contains a table of 'Reservas registradas' with the following data:

Estado Reserva	Fecha Reserva	Reservado Por	reservado A	Acciones
Entregado	2025-04-03	Hamilton Lopez	Tomás Álvarez	[*] Detalles, [Editar], [Eliminar]
Prestamo	2025-03-27	Hamilton Lopez	Hamilton Lopez	[*] Detalles, [Editar], [Eliminar]
Prestamo	2025-04-02	Tomás Álvarez	Tomás Álvarez	[*] Detalles, [Editar], [Eliminar]
Entregado	2025-04-01	Hamilton Lopez	Hamilton Lopez	[*] Detalles, [Editar], [Eliminar]
Entregado	2025-04-01	Luis Martínez	Luis Martínez	[*] Detalles, [Editar], [Eliminar]
Prestamo	2025-04-01	Hamilton Lopez	Tomás Álvarez	[*] Detalles, [Editar], [Eliminar]

The 'Detalles Reserva' panel on the right shows:

- Reservado Por: Hamilton Lopez
- Fecha de la reserva: 2025-04-01
- Estado Reserva: Prestamo
- # elementos reservados: 1
- Elemento: Simulacion de cirugia (Tipo: Espacios Fisicos)

Fuente. Autoría propia

**Tabla 18**  
Prueba gestión eventos

ID Prueba	PR-008
Descripción	Gestionar de eventos
Precondiciones	El usuario debe tener rol dentro del sistema, ya sea como funcionario o estudiante.
Pasos Que Seguir	<ul style="list-style-type: none"> <li>• Dar clic en el botón Gestión Reservas para el caso de los administradores.</li> <li>• Dar clic en el botón Eventos.</li> <li>• Diligenciar Fecha Inicio del Evento, Fecha Fin evento, funcionario responsable del evento, descripción o nombre del evento.</li> <li>• Clic botón Guardar.</li> </ul>

- Para editar el evento clic en botón editar, modificar fecha inicio o fin, nombre o funcionario encargado.
- Para eliminar evento clic botón eliminar, clic en el botón de confirmar.

Resultado Esperado

Se debe permitir al usuario el crear evento eliminar o modificar lo existentes de forma libre.

Resultado Obtenido

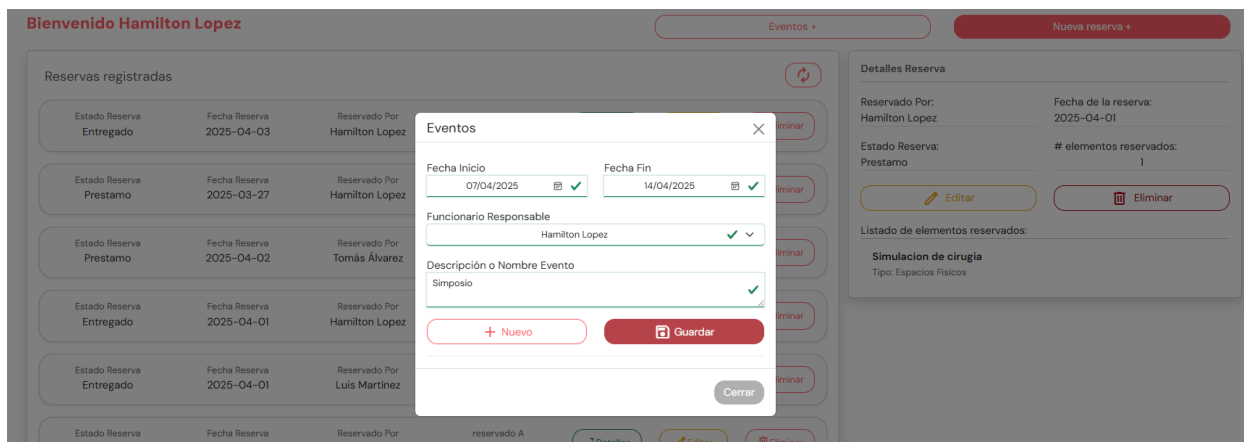
Se permite la gestión eventos por parte del usuario.

Estado

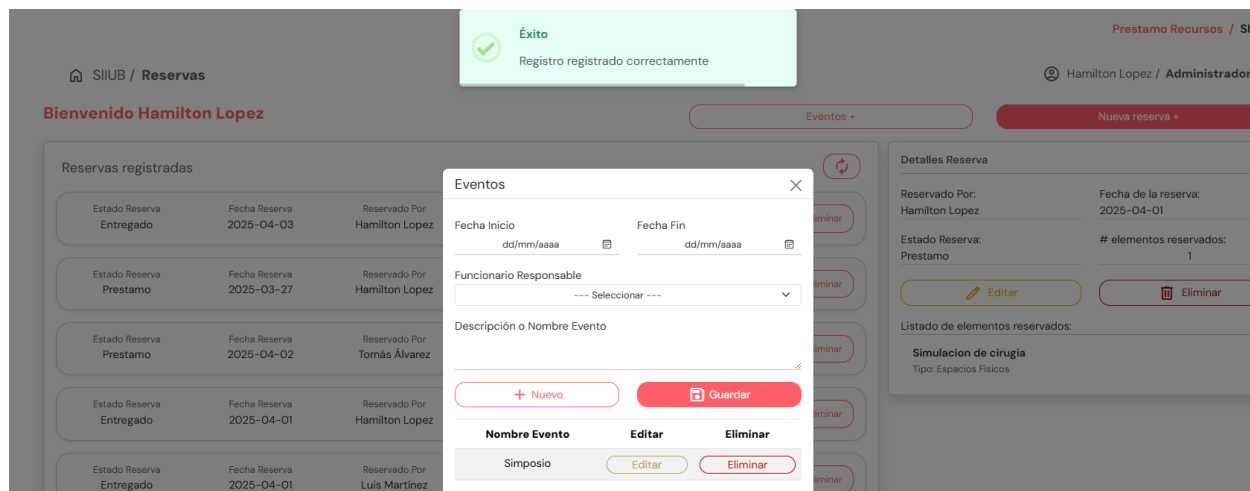
Aprobado

## Figura 80

### Evidencia prueba gestión eventos I



Fuente. Autoría propia

**Figura 81***Evidencia prueba gestión eventos II*

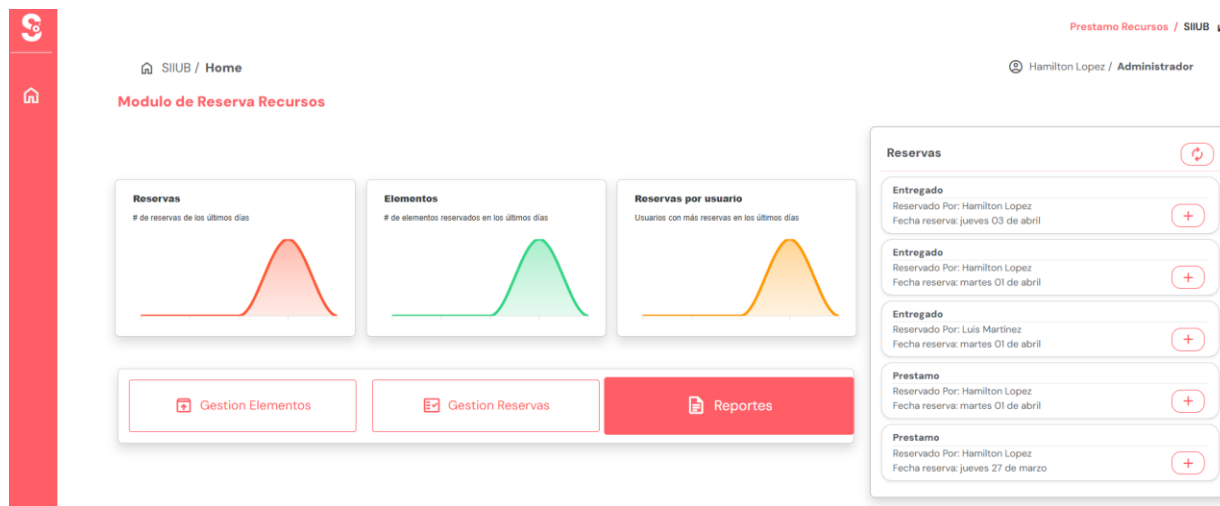
*Fuente. Autoría propia*

**Tabla 19***Prueba generación reportes*

ID Prueba	PR-009
Descripción	Generación de reportes
Precondiciones	El usuario debe tener rol dentro del sistema, con el rol de administrador para poder generar reportes, deben existir ya sea recursos o reservas registradas para poder crearlos.
Pasos Que Seguir	<ul style="list-style-type: none"> <li>• Dar clic en el botón Gestión Reportes.</li> <li>• Selecciona el tipo de reporte a generar dando clic en el botón.</li> </ul>
Resultado Esperado	Se debe descargar archivo de tipo Excel con los datos seleccionados a consultar.
Resultado Obtenido	Se descargar correctamente el archivo con los datos para poder ser procesados.
Estado	Aprobado

### Figura 82

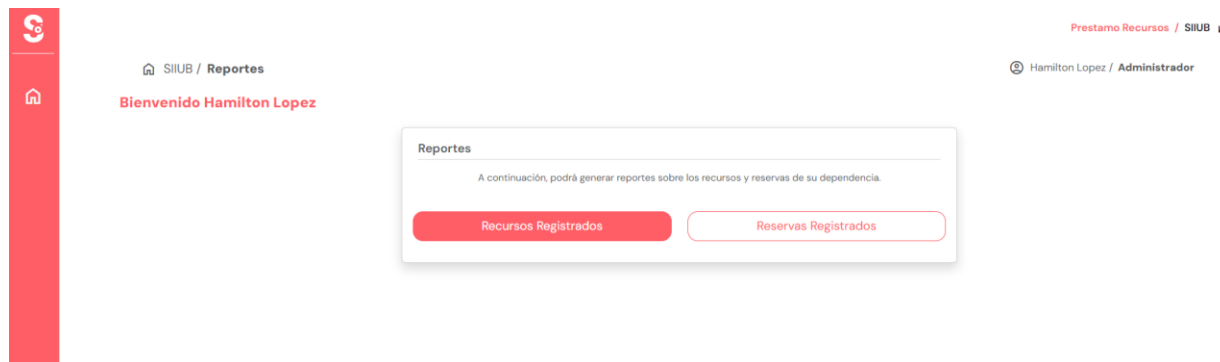
Evidencia prueba generación de reportes I



Fuente. Autoría propia

### Figura 83

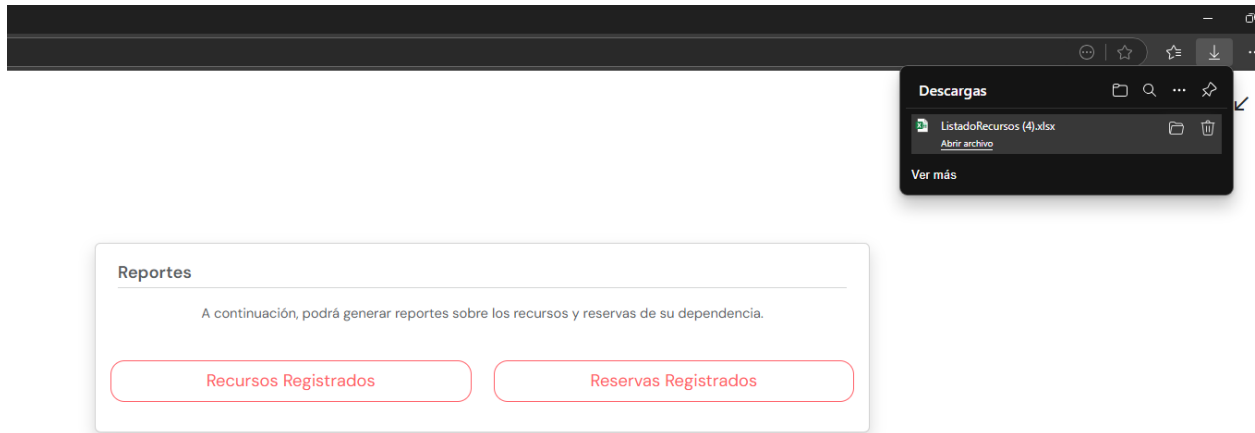
Evidencia prueba generación de reportes II



Fuente. Autoría propia

**Figura 84**

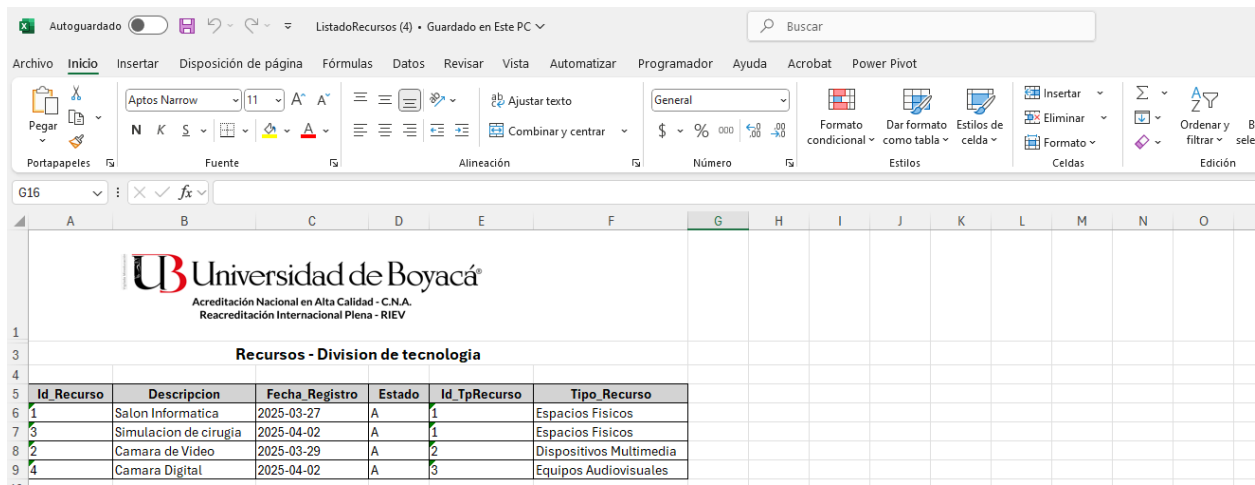
*Evidencia prueba generación de reportes III*



*Fuente. Autoría propia*

**Figura 85**

*Evidencia prueba generación de reportes IV*



*Fuente. Autoría propia*

Después de realizar las pruebas correspondientes y verificar su correcto funcionamiento a nivel técnico, se procede a llevar a cabo una validación cuantitativa del software, con el objetivo de determinar en qué medida se satisfacen los objetivos planteados al inicio del proyecto. Considerando su importancia en los procesos técnicos y operativos, se define una serie de métricas de calidad para evaluar el software de forma integral, proporcionando resultados objetivos.

Métrica	Valor Observado	Interpretación / Nivel de calidad
Tiempo promedio de respuesta	1.8 segundos	Excelente ( $\leq 2$ segundos se considera ideal)
Tasa de disponibilidad (uptime)	99.8%	Muy alta disponibilidad
Tasa de errores del sistema	0	Muy bajo índice de fallos
Ocupación de almacenamiento	650 MB	Alta optimización y baja duplicidad en uso
Uso de memoria RAM	40 MB	Eficiente, con margen para escalar
Número de actualizaciones correctivas	3	Baja cantidad, indica buena calidad inicial
Solicitudes de mejora recibidas	1	Participación, buena base para mejoras
Usabilidad	95/100	Excelente experiencia de uso
Tiempo de aprendizaje	3 horas	Curva de aprendizaje baja

## Despliegue del software

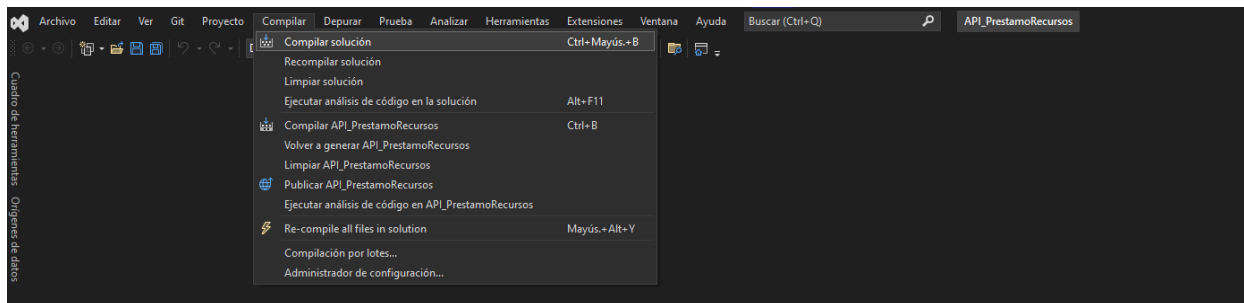
### API REST – Base de datos

Para el despliegue del software, se ha procurado llevar la implementación a un entorno similar al que el usuario utiliza en sus servidores. En este caso, se trata de un servicio basado en IIS (Internet Information Services) para el despliegue tanto de la API como de la base de datos. De este modo, se garantiza la correcta funcionalidad del sistema antes de su entrega al usuario final.

Una vez completado el desarrollo, se procede a compilar el código fuente de la API REST desde Visual Studio.

### Figura 86

#### *Compilación código fuente*



*Fuente. Autoría propia*

Se accede al servidor para obtener las credenciales, rutas y configuración del mismo, con el fin de conectarlo desde Visual Studio por FTP y realizar la publicación.

**Figura 87***Configuración servidor*

The screenshot shows the IIS Manager interface for the website 'gestionRecursos.somee.com'. The main configuration area includes:

- Default domain:** gestionRecursos.somee.com
- Title:** Aplicacion para la gestion de recursos
- Description:** Aplicacion para la gestion de recursos
- Subscription:** [Redacted]
- Service ID:** MPID4865051
- Invoices:** INVID1439871
- Hosting environment:** Windows Server 2022 (IIS 10.0, ASP, ASP.NET v2.0-4.8, ASP.NET Core)
- Server local path:** d:\DZHosts\LocalUser\jamilton25\www.gestionRecursos.somee.com

Buttons at the bottom of the main area: EDIT DESCRIPTION, STOP WEBSITE, ADD STORAGE.

**Storage:** Free: 16MB, Used: 2MB, Capacity: 143MB.

**URLs:** http://gestionRecursos.somee.com, http://www.gestionRecursos.somee.com. Button: MANAGE BINDINGS.

**FTP:** Addresses: ftp://gestionRecursos.somee.com/www.gestionRecursos.somee.com, ftp://155.254.246.42/www.gestionRecursos.somee.com. Username: [Redacted], Password: [Redacted]. Button: RESTORE ROOT FOLDER AND SECURITY SETTINGS.

**IIS Application pool:** ASP.NET version: .NET Framework 4.0 - 4.8, Managed Pipeline Mode: Integrated. Buttons: EDIT, RECYCLE POOL, IIS APPLICATIONS.

*Fuente.* Autoría propia

Se configura Visual Studio para conectar al servidor mediante FTP y se realiza la publicación desde la misma plataforma.

**Figura 88***Configuración FTP en visual*

The screenshot shows the Visual Studio interface for the project 'API PrestamoR...rsos: Publicar'. The 'Servidor FTP' configuration is displayed:

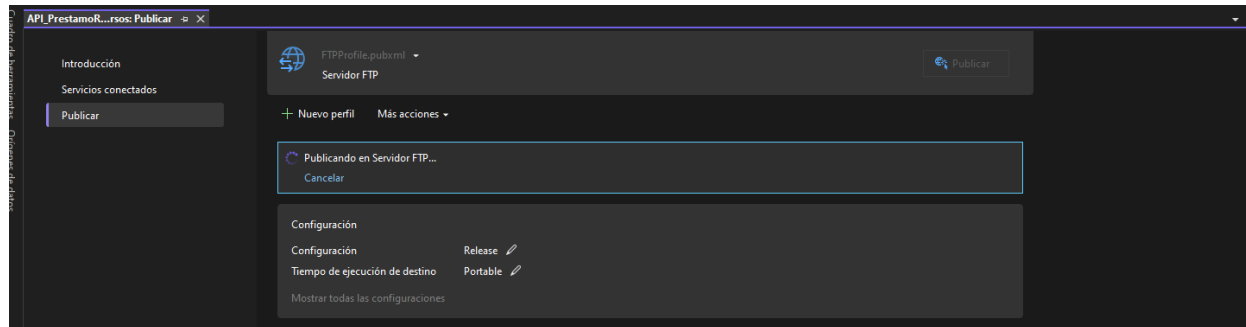
- FTPProfile.pubxml** - Servidor FTP. Button: Publicar.
- Configuración:** Configuración, Release, Tiempo de ejecución de destino, Portable. Button: Mostrar todas las configuraciones.
- Hospedaje:**
  - Servidor: ftp://155.254.246.42/www.gestionRecursos.somee.com
  - Ruta de acceso del sitio FTP: [Redacted]
  - Modo pasivo de FTP: true
  - Nombre de usuario: [Redacted]
  - Contraseña: [Redacted]
  - Sitio: http://www.gestionRecursos.somee.com

A green notification bar at the top states: 'La publicación se ha realizado correctamente el 28/03/2025 a las 9:42 p. m.' with an 'Abrir sitio' link.

*Fuente. Autoría propia*

## Figura 89

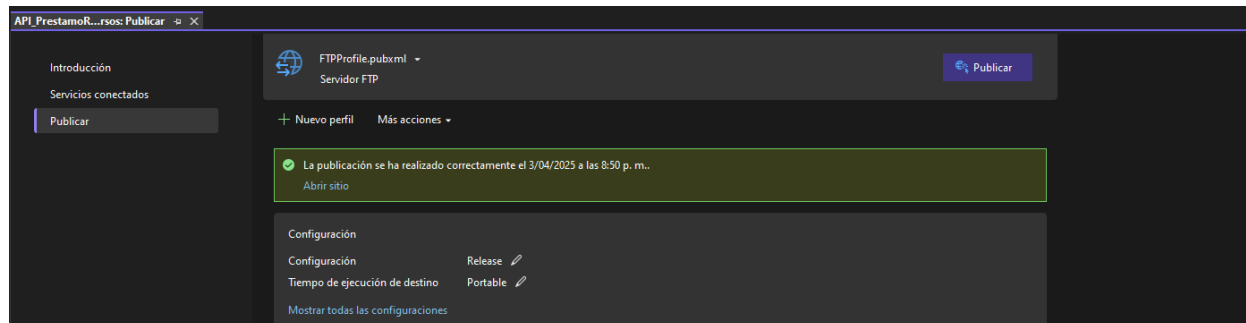
*Ejecución publicación desde visual estudio*



*Fuente. Autoría propia*

## Figura 90

*Publicación correcta API REST*




*Fuente. Autoría propia*

## Figura 91

*Se configura base de datos en servidor*

.../ Databases / dbReservas

Database details   Backup file manager   Run scripts   Restore   Backup   Attach   Get database files   Delete



**Database name:** dbReservas

**Full domain name:** dbReservas.mssql.somee.com

**SQL Version:** [blurred]

**Subscription:** Free hosting package (SPID1332770)

**Service ID:** MPID4865053

**Invoices:** INVID1439871

**Data file**   Capacity: 30MB

Used: 3MB   Free: 27MB

**Log file**   Capacity: 30MB

Used: 1MB   Free: 29MB

Connection details

SQL Server version: [blurred]

SQL Server address: [blurred]

Login name: [blurred]

Login password: [blurred]

Connection string: [blurred]

[CHANGE LOGIN PASSWORD](#)

Backup location FTP

**Address:** ftp://dbReservas.backup.somee.com/dbReservas\_MSSql\_Database\_Backup

**Username:** [blurred]

**Password:** \* Use the password from this control panel \*

[BACKUP FILE MANAGER](#)

*Fuente. Autoría propia*

## Figura 92

*Se cargan los scripts de todas la base de datos, tables, procedimientos almacenados etc.*

Enter your SQL query or T-SQL batch. It will be executed on database: dbReservas

```

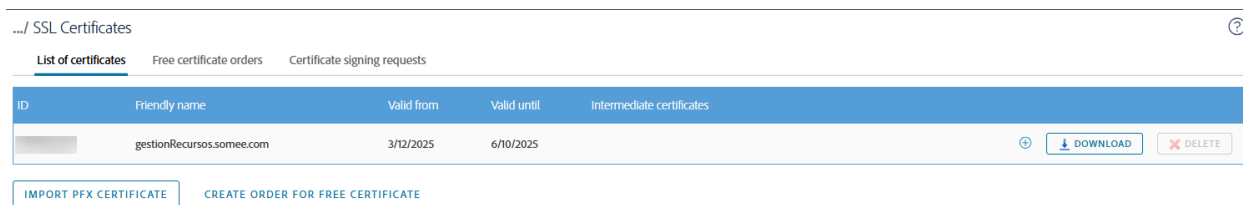
-----
-- DDL for Table DETALLE_RESERVA
-----
CREATE TABLE "SYSTEM"."DETALLE_RESERVA"
(
  "ID" NUMBER(10,0),
  "FECHA_REGISTRO" DATE,
  "OBSERVACION_ELEMENTO" VARCHAR2(300 BYTE),
  "ID_RESERVA" NUMBER(10,0),
  "ID_DISPONIBILIDAD" NUMBER(10,0),
  "ESTADO" VARCHAR2(2 BYTE)
)
  
```

[EXECUTE T-SQL](#)

*Fuente. Autoría propia*

## Figura 93

*Se configuran los certificados SSL para la API REST*



*Fuente. Autoría propia*

## Front End

Para el despliegue del front-end del software, se opta por un servicio para aplicaciones de una sola página (Single Page Application, SPA), que es la tecnología en la que fue desarrollado este software. Este se cargará en un servidor como Netlify. Para ello, primero se genera la carpeta de distribución suministrada por la interfaz de línea de comandos (CLI, por sus siglas en inglés) de Angular, y así podrá ser llevada al servidor para su despliegue.

## Figura 94

*Se crea carpeta de distribución desde el CLI de Angular.*

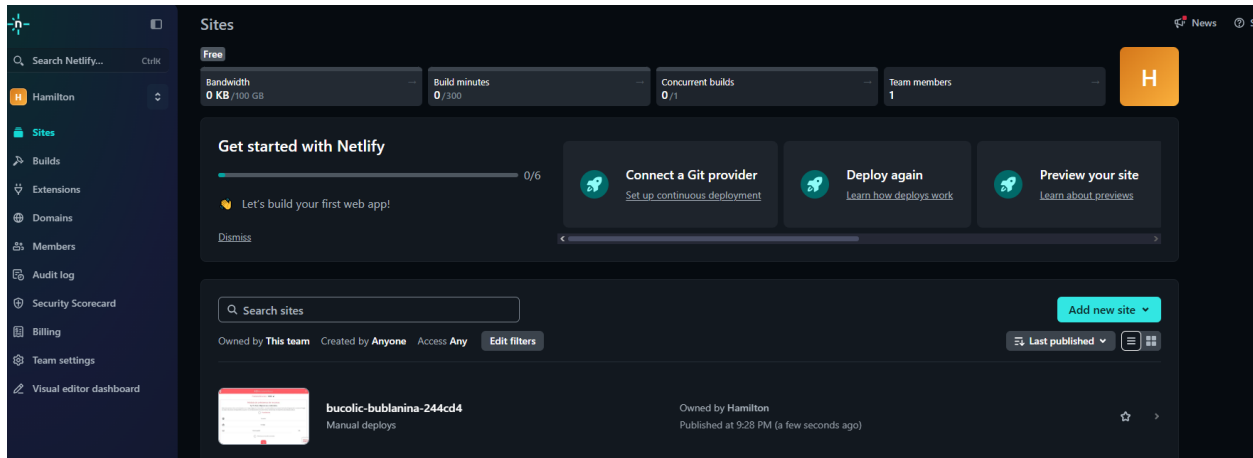
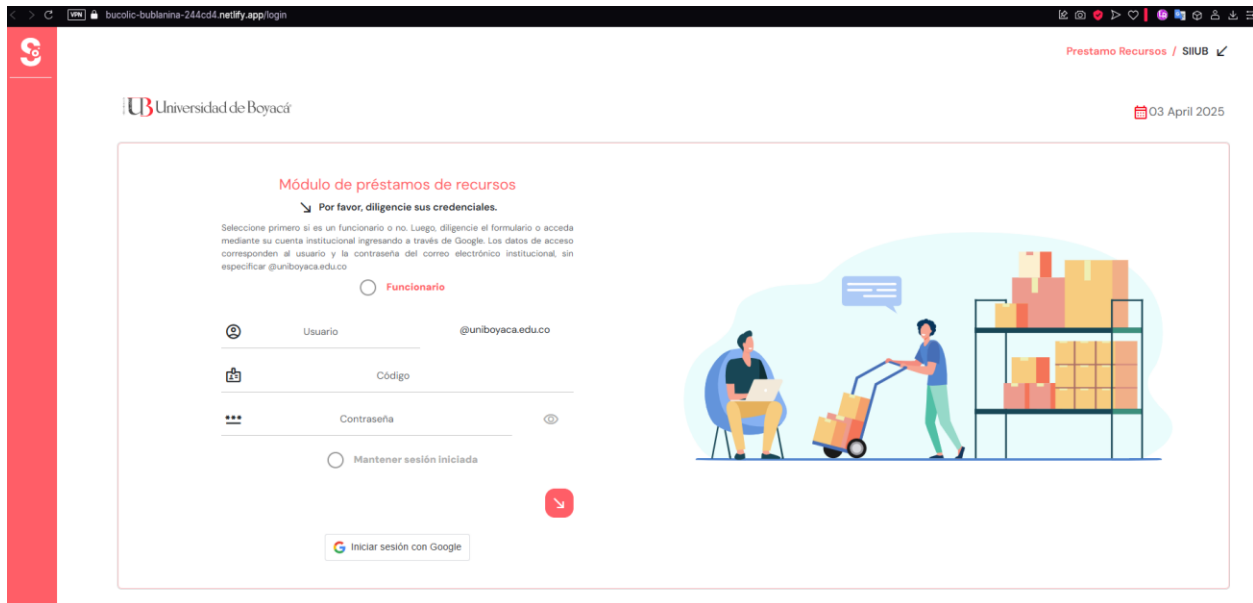
```

Símbolo del sistema para desarrolladores
+ PowerShell para desarrolladores
D:\ProyectoGrado\PrestamoRecursosFront\PrestamoRecursos_Front>ng build

Initial chunk files | Names | Raw size | Estimated transfer size
scripts-MT45546H.js | scripts | 647.35 kB | 147.88 kB
chunk-V6P3VD44.js | - | 275.34 kB | 73.53 kB
chunk-ZK0X3V2P.js | - | 73.62 kB | 18.10 kB
polyfills-FFH#D2TL.js | polyfills | 34.52 kB | 11.28 kB
main-M47LLKLP.js | main | 9.24 kB | 2.92 kB
chunk-EQVTEUXC.js | - | 1.17 kB | 519 bytes
chunk-TQKQYJ36.js | - | 634 bytes | 634 bytes
styles-E6544GJ6.css | styles | 546 bytes | 546 bytes

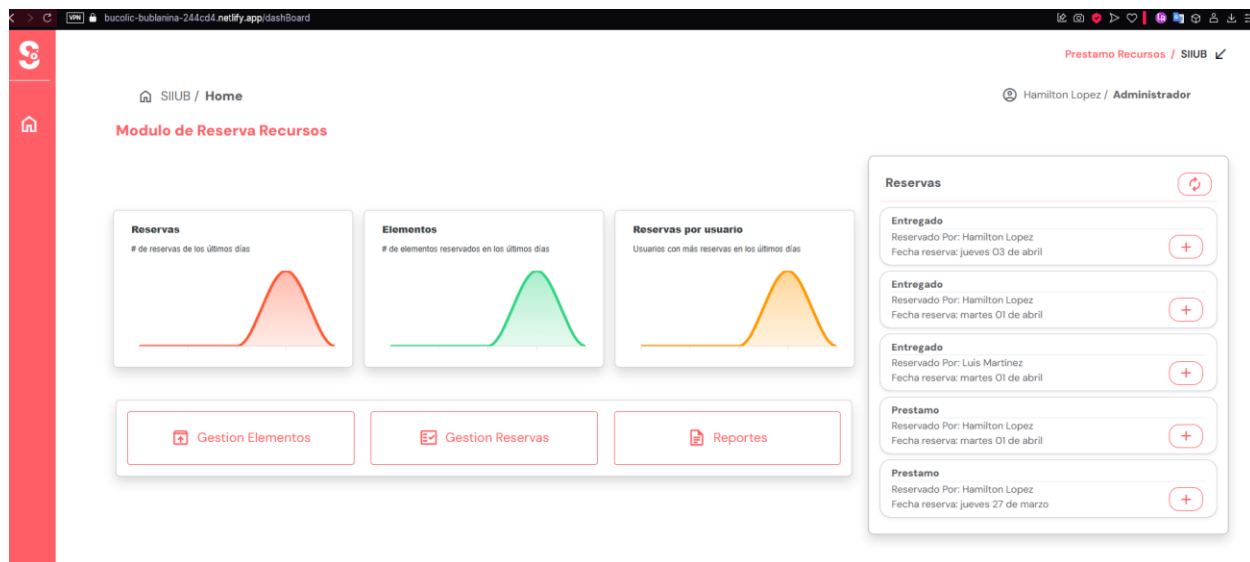
| Initial total | 1.04 MB | 255.41 kB
  
```

*Fuente. Autoría propia*

**Figura 95***Se carga carpeta a Netlify**Fuente. Autoría propia***Figura 96***Aplicación publicada**Fuente. Autoría propia*

## Figura 97

*Ingreso al módulo en versión publicada.*



*Fuente. Autoría propia*

## Referencias Bibliográficas

- Abuchar Porras, A. (2023). Metodologías ágiles para el desarrollo de software. Editorial Universidad Distrital Francisco José de Caldas.
- Álava Murillo, M. R. (2022). Estudio comparativo de tecnologías web de componentes, REACT. JS VS VUE. JS VS ANGULAR. JS para el proceso de desarrollo de aplicaciones web (Bachelor's thesis, Babahoyo: UTB-FAFI. 2022).
- Alamilla Hernández, L. M. ., Pérez Romero, V. A. ., Sosa González, S. A. ., & Valentín Rodríguez, J. A. . (2021). Arquitectura REST para el desarrollo de aplicaciones web empresariales. Revista Electrónica Sobre Tecnología, Educación Y Sociedad, 8(15).  
<https://www.ctes.org.mx/index.php/ctes/article/view/748>
- Barrón, E., & Morello, E. (2021). El problema de la gestión de la inclusión de las tecnologías de la información y la comunicación en la educación rural Peruana.
- Bass, L., Clements, P., Kazman, R. (2003). Software Architecture in Practice. Alemania: Addison-Wesley.
- Bass, L., Clements, P., & Kazman, R. (2021). Software architecture in practice (4th ed.). Addison-Wesley.
- Bernal Torres, M. . (2025). El Diseño Digital en la Mercadotecnia Contemporánea. Transregiones, (9), 149–174.  
<https://revistatransregiones.com/web/index.php/tr/article/view/131>
- Borja Saltos, G. N., & González Torres, J. E. (2024). Implementación de una plataforma E-

- Commerce para Softec WebStore aplicando la metodología Kanban (Bachelor's thesis, Universidad Estatal de Bolívar. Facultad de Ciencias Administrativas Gestión Empresarial e Informática. Carrera de Software).
- Edge, J. (2018). Kanban: La guía definitiva de la metodología Kanban para el desarrollo de software ágil. (n.p.): Amazon Digital Services LLC – Kdp.
- Farinango Caiza, E. P. (2020). Estudio del framework angular para desarrollar aplicaciones single-page. Desarrollo del sistema de información, seguimiento y control para la hacienda ganadera La Vega (Bachelor's thesis).
- Gelbmann, M., & Delamer, A. (2015). Usage of content management systems for websites.
- Gorton, I. (2020). Essential software architecture. Springer.
- Informatec. (2023). Las etapas del desarrollo de software: Una guía completa.<https://informatecdigital.com>.
- Jara, C. P. (2023-07-03). La automatización de procesos administrativos como estrategia de mejoramiento de atención al usuario en una Institución Educativa pública.  
<http://hdl.handle.net/10654/45304>
- Juan Martinez, (2024). Gestión de Proyectos Digitales: Curso Completo de Herramientas y Estrategias.<https://books.google.es/books?hl=es&lr=&id=eY02EQAAQBAJ&oi=fnd&pg=PP3&dq=gestión+de+proyectos&ots=1NVEBLYQHs&sig=0DSQMxb-1vaQdUyzsCMwhy3u13A#v=onepage&q=gestión%20de%20proyectos&f=false>
- Mena Inca, A. S. (2023). Desarrollo de una aplicación web para la gestión de la atención

psicológica de la Clínica Ocumedic utilizando la metodología Kanban.

Medina Salgado, A. B. . (2024). Metodologías Ágiles y su Contribución al Trabajo en Equipo:

Una Revisión Narrativa del Rol de Scrum en la Cohesión y Productividad

Organizacional. *Business Innova Sciences*, 5(3), 197-210.

<https://doi.org/10.5281/zenodo.13891388>

Microsoft. (n.d.). Servicios que ofrece Azure DevOps. Microsoft Learn.

<https://learn.microsoft.com/es-mx/azure/devops/user-guide/services?toc=%2Fazure%2Fdevops%2Fget-started%2Ftoc.json&view=azure-devops>

Norman, D. A. (2013). *The design of everyday things*. Basic Books.

Notion Labs Inc. (s.f.). Notion: Herramienta todo en uno para organizar tu trabajo y tu vida.

Notion. <https://www.notion.so/es-es>

Ojeda, J. C., & Fuentes, M. D. C. G. (2012). Taxonomía de los modelos y metodologías de desarrollo de software más utilizados. *Universidades*, (52), 37-47.

Pastor, R. A. T. (2009). Modelo conceptual para la gestión de proyectos. *Perspectivas*, (24), 165-188.

Pressman, R. S. (2014). *Ingeniería de software: Un enfoque práctico* (7.<sup>a</sup> ed.). McGraw-Hill.

Price, M. J. (2021). *C# 10 and .NET 6—Modern Cross-Platform Development: Build apps, websites, and services with ASP. NET Core 6, Blazor, and EF Core 6 using Visual Studio 2022 and Visual Studio Code*. Packt Publishing Ltd.

Ramos Cardozzo, D. (2016). *Desarrollo de Software: Requisitos, Estimaciones y Análisis*. 2 Edición. España: CreateSpace Independent Publishing Platform.

Rebé Martín, J. (2023). *Desarrollo de aplicación web utilizando ASP. NET para la gestión de reglas y sanciones en F1*.

Robayo-Jácome, D. J. (2023). *Calidad en el desarrollo de software desde una perspectiva bibliométrica en el periodo 2018 – 2022*. *INNOVA Research Journal*, 8(3), 173–188.  
<https://doi.org/10.33890/innova.v8.n3.2023.2299>

Ruiz de Eguino Fernández, A. (2020). *Aplicación del enfoque Lean Startup y metodologías ágiles para la gestión de proyectos en entornos innovadores*.

Sommerville, I. (2011). *Software Engineering*. Pearson Education.

Software para Todo. (2023). *Los 7 pasos del ciclo de vida del desarrollo de software: Todo lo que necesitas saber*. <https://softwarepara.es>.

Universidad Nacional Abierta y a Distancia. (2014). *Acuerdo 006 de 2014 [PDF]*. Consejo Académico.  
[https://sgeneral.unad.edu.co/images/documentos/consejoAcademico/acuerdos/2014/COA\\_C\\_ACUE\\_20140528\\_006.pdf](https://sgeneral.unad.edu.co/images/documentos/consejoAcademico/acuerdos/2014/COA_C_ACUE_20140528_006.pdf)

UNESCO. (2023). *Reflexiones sobre la importancia de la educación superior y el papel de las universidades en el contexto actual*. UNESCO.  
[https://unesdoc.unesco.org/ark:/48223/pf0000378847\\_spa](https://unesdoc.unesco.org/ark:/48223/pf0000378847_spa)

Walker, A. J. (2003). *Web-Based Human Resources*. McGraw-H