



Instituto Superior Tecnológico
AMAZÓNICO

**“DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA
WEB PARA EL SEGUIMIENTO Y CONTROL DE LOS
PROYECTOS INFORMÁTICOS DE LA EMPRESA MY
WEB SERVICE”**

CARRERA:

TECNOLOGÍA SUPERIOR EN DESARROLLO DE SOFTWARE

LÍNEA DE INVESTIGACIÓN:

ADAPTACIÓN TECNOLÓGICA E INNOVACIÓN

SUBLÍNEA DE INVESTIGACIÓN:

SISTEMAS DE INFORMACIÓN

Tesis previa a la obtención del título
de Tecnóloga Superior en Desarrollo
de Software

AUTOR:

ORDOÑEZ POMA LADY MARIBEL

DIRECTOR:

CHAMBA SACA DIEGO JONATHAN

FECHA:

09 de diciembre de 2022

YANTZAZA – ZAMORA CHINCHIPE – ECUADOR

2022

CERTIFICACIÓN

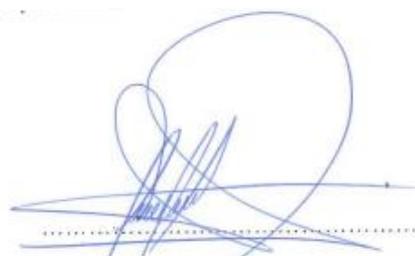
Yantzaza, 25 de octubre de 2022

YO, DIEGO JONATHAN CHAMBA SACA, DIRECTOR DE LA TESIS A PETICIÓN VERBAL DE LA PARTE INTERESADA

CERTIFICO:

Que una vez revisado el proyecto de tesis denominado “**DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA WEB PARA EL SEGUIMIENTO Y CONTROL DE LOS PROYECTOS INFORMÁTICOS DE LA EMPRESA MY WEB SERVICE**”, elaborado por la Srta. **Ordoñez Poma Lady Maribel**, con cédula de ciudadanía N° 1950102564, luego de las rectificaciones realizadas al mismo, se encuentra **APROBADO**, para que sea presentado al respectivo tribunal de grado.

Es todo cuanto puedo certificar en honor a la verdad, autorizando a que se continúe con el proceso correspondiente.



Ing. Diego Jonathan Chamba Saca

DIRECTOR

MIEMBROS DEL TRIBUNAL



Ing. Nelly E. Cueva R

**MIEMBRO DEL
TRIBUNAL DE GRADO**



Ing. Diego V. Guamán J

**MIEMBRO DEL
TRIBUNAL DE GRADO**



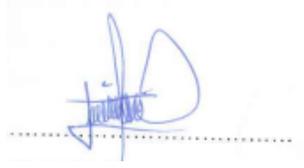
Lcdo. Luis A. Jiménez A

**MIEMBRO DEL
TRIBUNAL DE GRADO**

AUTORIA

Yo, Lady Maribel Ordoñez Poma, declaro ser autora del presente proyecto de tesis y eximimos al Instituto Superior Tecnológico Amazónico y a sus representantes jurídicos de posibles reclamos o acciones legales del contenido de la misma.

Además, se acepta y se autoriza al Instituto Superior Tecnológico Amazónico, la publicación del proyecto de titulación en las plataformas institucionales que tenga.

Firma: 

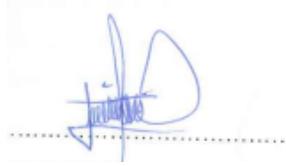
Cédula: 1950102564

Fecha: 21 de octubre de 2022

CESIÓN DE DERECHOS

Autorizo al Instituto Superior Tecnológico Amazónico, para que haga de este proyecto de investigación, un documento disponible para su lectura, consulta y proceso de investigación.

Cedo los derechos en líneas patrimoniales de mi proyecto de investigación, con fines de difusión pública, además apruebo la reproducción de este Proyecto de Tesis, dentro de las regulaciones del Instituto, siempre y cuando esta reproducción no suponga ganancia económica potencial; y se realice respetando mis derechos de autor.

A handwritten signature in blue ink, consisting of stylized, overlapping loops and lines, positioned above a horizontal dashed line.

Lady Maribel Ordoñez Poma

AUTORA

C.I. 1950102564

DEDICATORIA

Primeramente, a Dios por permitirme llegar a concluir con mis objetivos y metas que me he propuesto en mi vida.

A mis padres por ser las personas que me han apoyado y acompañado durante todo mi trayecto académico y de vida, depositando su entera confianza en cada paso que doy y sin dudar ni un solo momento en mi capacidad y perseverancia.

A mis compañeros por apoyarnos en los momentos más difíciles motivándonos a llegar al final del camino sabiendo que llegaremos a ser unos excelentes profesionales.

A nuestros docentes, gracias por todo el conocimiento impartido, por la paciencia, el tiempo, la comprensión y el apoyo que transmitieron durante el desarrollo de nuestra formación académica y profesional.

Es por ello que gracias a todos ustedes he podido convertirme en la persona de bien que soy hoy.

AGRADECIMIENTO

Dejo constancia de mi sincero agradecimiento al Instituto Superior Tecnológico Amazónico, al Área Administrativa, y a todos los Docentes de la Carrera de Tecnología Superior en Desarrollo de Software, quienes con dedicación y capacidad académica nos impartieron valiosos conocimientos y sus sabias experiencias brindándonos su apoyo en todos los momentos de nuestra formación profesional.

De manera especial al Ing. Diego Jonathan Chamba Saca, director del proyecto de tesis, por su colaboración, paciencia y apoyo desinteresado para la culminación del presente proyecto, siempre dejándome presente que puedo lograr cualquier meta que me proponga.

Así mismo a l propietario de la empresa My Web Service, quien me brindó la oportunidad de desarrollar el presente proyecto de Tesis, el mismo que deja una valiosa herramienta administrativa trazado para los años venideros de la administración.

LA AUTORA

CONTENIDO

CERTIFICACIÓN.....	II
MIEMBROS DEL TRIBUNAL.....	III
AUTORIA	IV
CESIÓN DE DERECHOS	V
DEDICATORIA	VI
AGRADECIMIENTO	VII
CONTENIDO	VIII
1. TITULO	1
2. RESUMEN	2
3. INTRODUCCIÓN	4
4. REVISIÓN DE LITERATURA.....	5
4.1. Administración	5
4.5. Sistema de gestión administrativa	6
4.6. Java	6
4.7. Spring Boot.....	7
4.8. Thymeleaf.....	7
4.9. Servidor de aplicaciones Tomcat.....	8
4.10. Modelo Vista Controlador	8
4.11. Visual Studio Code.....	9
4.12. StartUML.....	9
5. MATERIALES Y MÉTODOS	11
5.7. Materiales	11
5.2. Métodos de investigación	13
5.3. Técnicas de recolección de datos	13
5.4. Metodología:	14
6. RESULTADOS	15
6.1. Análisis de requisitos.....	15
6.2. Revision del análisis y diseño preliminar.....	25
6.3. Diseño	40
6.4. Implementación	43
7. CONCLUSIONES	76
8. RECOMENDACIONES	77
9. BIBLIOGRAFÍA.....	78

10.	ANEXOS.....	80
10.1.	Anexo 1: Certificación de traducción.....	80
10.2.	Anexo 2: Entrevista.....	81
10.3.	Anexo 3: Ficha de observación.....	84
10.4.	Anexo 4: Resultados del test de usabilidad	86
10.5.	Anexo 5: Proyecto de Tesis.....	90

Índice de figuras

Figura 1	Arquitectura API Rest	11
Figura 2	Modelo del dominio de la empresa My Web Service	18
Figura 3	Prototipo Pagina de Bienvenida Administrador	19
Figura 4	Prototipo Pagina de Usuarios activos	19
Figura 5	Prototipo Pagina de Proyectos	20
Figura 6	Prototipo Pagina de Proyectos Step 1: Datos del proyecto	20
Figura 7	Prototipo Pagina de Proyectos Step2: Administrador y desarrolladores	21
Figura 8	Prototipo Pagina de Administrar actividades.....	21
Figura 9	Prototipo Pagina de Gestionar avances.....	22
Figura 10	Prototipo Pagina de Inicio de sesión.....	22
Figura 11	Prototipo Pagina de Recuperar contraseña	23
Figura 12	Diagrama de casos de uso Administrador	24
Figura 13	Diagrama de casos de uso Administrador del proyecto y desarrollador	24
Figura 14	Diagrama de robustez: Inicio de sesión	35
Figura 15	Diagrama de robustez: Recuperar contraseña.....	36
Figura 16	Diagrama de robustez: Administrar proyecto	37
Figura 17	Diagrama de robustez: Administrar actividades	37
Figura 18	Diagrama de robustez: Gestionar avances	38
Figura 19	Diagrama de robustez: Gestionar usuarios - crear.....	38
Figura 20	Diagrama de robustez: Gestionar usuarios - dar de baja	39
Figura 21	Diagrama de robustez: Gestionar usuarios - dar de alta	39
Figura 22	Diagrama de robustez: Actualizar datos	40
Figura 23	Diagrama de secuencia: Administrar proyecto.....	41
Figura 24	Diagrama de secuencia: Administrar actividades - crear - proceso REST	41
Figura 25	Diagrama de secuencia: Administrar actividades - crear	42
Figura 26	Diagrama de secuencia: Gestionar avances - crear - proceso REST	42
Figura 27	Diagrama de secuencia: Gestionar avances - crear.....	43
Figura 28	Diagrama de secuencia: Crear usuario.....	43
Figura 29	Diagrama de componentes del sistema.....	44
Figura 30	Interface Usuario Repositorio	45
Figura 31	Interface Usuario repositorio.....	45
Figura 32	Interface Usuario ServiceImplements	46
Figura 33	Clase controladora.....	47
Figura 34	Carga de archivo a AWS-S3.....	48
Figura 35	Motor de plantillas Thymeleaf	49
Figura 36	Dashboard del sistema: Pagina de Bienvenida.....	49
Figura 37	Dashboard del sistema: Pagina de Usuarios	50
Figura 38	Dashboard del sistema: Pagina de proyectos	50
Figura 39	Dashboard del sistema: Pagina de nuevo proyecto Step 1	51
Figura 40	Dashboard del sistema: Pagina de nuevo proyecto Step 2	51
Figura 41	Dashboard del sistema: Pagina de actividades.....	52
Figura 42	Dashboard del sistema: Pagina de avances	52
Figura 43	Resultados de pruebas de carga JMeter	75

Índice de tablas

Tabla 1	Recursos humanos utilizados para el proyecto	11
Tabla 2	Recursos Hardware y software	12
Tabla 3	Recursos de oficina	12
Tabla 4	Presupuesto total del proyecto.....	13
Tabla 5	Requisitos funcionales	17
Tabla 6	Requisitos no funcionales	17
Tabla 7	Especificación de caso de uso: Iniciar sesión	26
Tabla 8	Especificación de caso de uso: Administrar Proyecto	27
Tabla 9	Especificación de caso de uso: Gestionar usuarios	31
Tabla 10	Caso prueba – Inicio de sesión.....	53
Tabla 11	Caso prueba – Gestión de usuarios.....	57
Tabla 12	Caso prueba – Administrar proyecto.....	64
Tabla 13	Formato de evaluación test SUML	74

1. TITULO

“Desarrollo e implementación de un sistema web para el seguimiento y control de los proyectos informáticos de la empresa My Web Service”

2. RESUMEN

El presente proyecto muestra el proceso de creación de un sistema web para gestionar el proceso de administración de los proyectos de software que desarrolla la empresa My Web Service, la arquitectura que se ha empleado ha sido Api Rest que permite el desarrollo de servicios de aplicaciones con ventajas como son: visibilidad, fiabilidad, escalabilidad, independencia de la plataforma o lenguajes, trabaja con el protocolo cliente/servidor; también se trabajó con la arquitectura cliente-servidor; el sistema funcionará en cualquier computador o dispositivo móvil.

El sistema web permite al usuario realizar los procesos acordes al rol; Administrador. – gestión de usuarios, gestión de proyectos, gestión de actividades; Administrador de Proyecto. – gestión de actividades, gestión de avances; Desarrollador. – gestión de avances. Con el sistema se logra que los procesos se realicen de manera eficaz y eficiente, controlando de manera fluida el flujo de la información que se genera en cada uno de los procesos. El sistema ha sido desarrollado tomando como base fundamental las necesidades del usuario y teniendo en cuenta el perfil que corresponde para mostrar los procesos que le corresponden.

Utilizando la metodología Iconix se logró identificar la problemática a través de un análisis previo con la ayuda de los métodos deductivo, inductivo y analítico; y de técnicas de recolección de información como la entrevista y ficha de observación. Aplicando la primera fase de análisis de requisitos se estableció las funcionalidades del sistema, modelo del dominio, prototipo y finalmente los casos de uso. En la fase de análisis y diseño preliminar se realizó las especificaciones de casos de uso y diagramas de robustez. Dentro de la fase de diseño se elaboró diagramas de secuencia, en la fase de implementación se realiza diagrama de componentes, código del sistema con lenguaje Java y JavaScript. Además de realizar pruebas al sistema para garantizar su calidad y eficiencia. Y finalmente se muestra conclusiones y recomendaciones.

Palabras Clave: Sistema web, Api Rest, Java, Thymeleaf, Iconix

ABSTRACT

This project shows the process of creating a web system to manage the administration process of software projects developed by the company My Web Service, the architecture that has been used has been Api Rest that allows the development of application services with advantages such as: visibility, reliability, scalability, independence of the platform or languages, works with the client/server protocol; we also worked with the client-server architecture; The system will work on any computer or mobile device.

The web system allows the user to carry out the processes according to the role; Administrator. – user management, project management, activity management; Project Manager. – activity management, progress management; developer. – progress management. With the system it is achieved that the processes are carried out effectively and efficiently, fluidly controlling the flow of information that is generated in each of the processes. The system has been developed based on the fundamental needs of the user and taking into account the corresponding profile to show the processes that correspond to it.

Using the Iconix methodology, the problem will be identified through a previous analysis with the help of deductive, inductive and analytical methods; and information gathering techniques such as the interview and observation sheet. Applying the first phase of requirements analysis, the functionalities of the system, domain model, prototype and finally the costs of use are confirmed. In the analysis and preliminary design phase, the specifications of use cases and robustness diagrams were made. Within the design phase, sequence diagrams were elaborated, in the implementation phase, component diagram, system code with Java language and JavaScript were made. In addition to testing the system to ensure its quality and efficiency. And finally conclusions and recommendations are shown.

Keywords: Website system Api Rest, Java, Thymeleaf, Iconix

El certificado de traducción de este documento se lo puede revisar en el Anexo 1

3. INTRODUCCIÓN

En el presente proyecto de investigación se propone el desarrollo de un sistema de administración de proyectos para la empresa My Web Service, el objetivo principal es mejorar y organizar el proceso de seguimiento y control de los proyectos de desarrollo de software.

Por este motivo se ha creído conveniente la implementación de un sistema web que ayude a la sistematización de los procesos, disminuyendo así el tiempo que el personal emplea en cada actividad.

A continuación, se da una breve descripción de cada una de las secciones por las cuales está estructurado el proyecto:

En la revisión literaria se enmarcan temas de relevancia para la construcción del sistema.

En la sección de materiales y métodos, se hace referencia a recursos humanos, hardware y software; métodos utilizados: deductivo, inductivo y analítico; técnicas de recolección de datos como: observación y entrevista; además de la metodología Iconix para estructurar el proyecto en fases e ir cumpliendo diferentes hitos.

Los resultados de la tesis contienen el desarrollo de la propuesta en donde se evalúa los objetivos y como se los efectuó, para lo cual se emplea las fases de la metodología Iconix.

Las conclusiones planteadas son parte de la síntesis y valoración de los diferentes objetivos planteados, mientras que las recomendaciones son sugerencias planteadas para en un futuro mejorar este proyecto, además de consejos sobre los aspectos que han favorecido en el desarrollo del mismo.

La bibliografía se ha utilizado las normas APA y en los anexos se incluyen las diferentes fotografías, entrevista y fichas de observación, que son evidencia de lo que se ha realizado en el proyecto.

4. REVISIÓN DE LITERATURA

Para conocer los términos utilizados en el proyecto se muestran algunas definiciones.

4.1. Administración

Proviene **del latín ad–ministrare**, que significa “**estar bajo el mando del otro, prestar un servicio**”. Es el conjunto de funciones cuya finalidad es administrar. Es considerada la técnica que busca obtener resultados de máxima eficiencia, por medio de la coordinación de las personas, cosas y sistemas que forman una organización o entidad, también tiene como objetivo el estudio de las técnicas utilizadas en la planificación, en la integración, control de los recursos y la unificación de una organización con la finalidad de obtener de ello diversos beneficios, ya sea de índole social o económico, este último va a depender de los fines que dicha organización tenga por objetivo (Y., 2021).

4.2. Gestión

El término gestión es utilizado para referirse al conjunto de acciones, o **diligencias que permiten la realización de cualquier actividad o deseo**. Dicho de otra manera, una gestión se refiere a todos aquellos trámites que se realizan con la finalidad de resolver una situación o materializar un proyecto. En el entorno empresarial o comercial, la gestión es asociada con la administración de un negocio (R, 2021).

4.3. Gestión administrativa

La gestión administrativa como elemento fundamental dentro de la administración, y aspecto aglutinador y sistémico, han sido el soporte y sostén del desarrollo empresarial, económico, social y tecnológico del planeta en los últimos siglos y en especial las últimas décadas. La gestión administrativa es un proceso que contempla cuatro funciones fundamentales: planificación, organización, dirección y control.

- **Planificación:** es la primera función administrativa, consiste en definir las metas, trazar los objetivos, establecer los recursos y las actividades que se desarrollarán en un periodo de tiempo determinado.
- **Organización:** consiste en armar una estructura para distribuir los recursos materiales, financieros, humanos, tecnológicos y de información, para desarrollar su trabajo, y poder alcanzar los objetivos planificados.

- Dirección: Incluye la ejecución de las estrategias trazadas, orientando los esfuerzos hacia los objetivos trazados, a través del liderazgo, la motivación y la comunicación.
- Control: Consiste en verificar que las tareas previstas estén alineadas con las estrategias planificadas, con el fin de corregir cualquier problema y evaluar los resultados, para un mejor proceso de toma de decisiones (González Rodríguez, Viteri Intriago, Izquierdo Morán, & Verdezoto Cordova, 2021).

4.5. Sistema de gestión administrativa

Un sistema de gestión es una herramienta que permite controlar, planificar, organizar y automatizar las tareas administrativas de una organización. Un sistema de gestión analiza los rendimientos y los riesgos de una empresa, con el fin de otorgar un ambiente laboral más eficiente y sostenible.

Algunas empresas o PyMEs cuentan con actividades que no están automatizadas, que con frecuencia se soportan en sistemas departamentales y casi siempre en hojas Excel desarrolladas individualmente por los usuarios implicados en cada una de las funciones. Un software de gestión unifica la operación de todas las áreas del negocio para alinearlas con los objetivos de la empresa. Es importante que cada área tenga claro y definido tanto su objetivo como sus metas. Esto da la garantía de direccionar tu negocio más fácil y eficazmente (ERP, 2021).

4.6. Java

Java es un lenguaje de programación y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems. Hay muchas aplicaciones y sitios web que no funcionarán a menos que tenga Java instalado y cada día se crean más. Java es rápido, seguro y fiable. Desde portátiles hasta centros de datos, desde consolas para juegos hasta súper computadoras, desde teléfonos móviles hasta Internet, Java está en todas partes. Para la construcción de este sistema informático se utilizará la versión 8 de Java, ahora conocido como OpenJDK 1.8. (Java.com, s.f.).

4.7. Spring Boot

Spring Boot, para este proyecto se utilizó la versión 2.4.2 de Spring Boot, Spring facilita la creación de aplicaciones independientes basadas en el lenguaje de programación Java y en el framework Spring, que tienen un alto grado de producción que puede "simplemente ejecutar".

Las características más importantes que se puedan lograr con este framework, se detallan a continuación:

- Cree aplicaciones Spring independientes
- Incruste Tomcat, Jetty o Undertow directamente (no es necesario implementar archivos WAR)
- Proporcione dependencias 'de inicio' obstinadas para simplificar su configuración de compilación.
- Configure automáticamente las bibliotecas de Spring y de terceros siempre que sea posible.
- Proporcione funciones listas para producción, como métricas, verificaciones de estado y configuración externa.
- Absolutamente sin generación de código y sin requisitos de configuración XML. (Spring.io, s.f.)

4.8. Thymeleaf

Thymeleaf es un moderno motor de plantillas Java del lado del servidor para entornos web e independientes.

El objetivo principal de Thymeleaf es traer elegantes plantillas naturales a su flujo de trabajo de desarrollo: HTML que se puede mostrar correctamente en los navegadores y también funciona como prototipos estáticos, lo que permite una colaboración más sólida en los equipos de desarrollo.

Con módulos para Spring Framework, una gran cantidad de integraciones con sus herramientas favoritas y la capacidad de conectar su propia funcionalidad, Thymeleaf es ideal para el desarrollo web JVM HTML5 moderno, aunque puede hacer mucho más. (Thymeleaf org, 2022)

4.9. Servidor de aplicaciones Tomcat

El framework Spring Boot tiene embebido y configurado el servidor de aplicaciones Tomcat, listo para desplegar a producción cualquier sistema construido en el lenguaje de programación Java.

Tomcat, también llamado Apache Tomcat o Jakarta Tomcat es un contenedor open source de servlets para la implementación de Java Servlet, JavaServer Pages (JSP), Java Expression Language y Java WebSocket. Las especificaciones de estos son desarrolladas bajo el Java Community Process.

Existe una confusión de conceptos entre Apache Tomcat y Apache, esto se debe a que coloquialmente al servidor Apache HTTP se le conoce como Apache. Tanto Tomcat como Apache HTTP son proyectos de Apache Software Foundation, cada uno implementado en un lenguaje diferente, Apache Tomcat en Java y Apache HTTP en C y XML. Aunque la diferencia fundamental es que el objetivo de Tomcat es servir específicamente aplicaciones Java, mientras que Apache es un servidor HTTP de propósito general. (Tomcat, 2022)

4.10. Modelo Vista Controlador

Modelo Vista Controlador (MVC) es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

Se trata de un modelo muy maduro y que ha demostrado su validez a lo largo de los años en todo tipo de aplicaciones, y sobre multitud de lenguajes y plataformas de desarrollo.

- El Modelo que contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia.
- La Vista, o interfaz de usuario, que compone la información que se envía al cliente y los mecanismos interacción con éste.
- El Controlador, que actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno. (Modelo vista controlador (MVC), s.f.).

4.11. Visual Studio Code

Visual Studio Code es un editor de código fuente que permite trabajar con diversos lenguajes de programación, admite gestionar tus propios atajos de teclado y refactorizar el código. Es gratuito, de código abierto y nos proporciona una utilidad para descargar y gestionar extensiones con las que podemos personalizar y potenciar esta herramienta (Code, s.f.).

Para el proyecto se instala varios plugin, dentro de estos angular snippets, autodocblocker, format HTML in PHP, JavaScript and TypeScript Nightly, Material Icon Theme y Terminal.

4.12. StartUML

Con respecto a la herramienta para realizar los diferentes diagramas de cada fase de la metodología se utilizó StartUML (StarUML, s.f.), aquí se trabajó con:

- Diagrama de casos de uso: muestra el comportamiento que se espera de un software ante la acción del actor.
- Diagrama de clase (modelo del dominio): se muestran todas las entidades, atributos, relaciones y restricciones que se analizaron dentro del problema encontrado.
- Diagrama de robustez: son para verificar que la especificación de casos de uso es correcta y está completa.
- Diagrama de secuencia: describe cómo y en qué orden un grupo de objetos funcionan en conjunto.
- Diagrama de componentes: muestra como un sistema está dividido en componentes y a la vez como están las dependencias de los componentes.

4.13. PostgreSQL

PostgreSQL es un potente sistema de base de datos relacional de objetos de código abierto, abierto con más de 30 años de desarrollo activo que le ha ganado una sólida reputación por su fiabilidad, solidez de funciones y rendimiento. (PostgreSQL, s.f.)

4.14. SQL

SQL es un lenguaje estándar de programación para bases de datos, este lenguaje se utiliza para acceder y manipular las bases de datos ya sean en MySQL, Oracle, DB2, SQL Server, Access. También se dice que es un lenguaje estructurado y un estándar ANSI para el acceso y la manipulación de cualquier base de datos, se compone de sentencias SQL, cada una con una funcionalidad diferente, por ejemplo:

- Creación de la base de datos (CREATE DATABASE)
- Creación de la tabla (CREATE TABLE) (SQL, 2012)

4.15. Api Rest

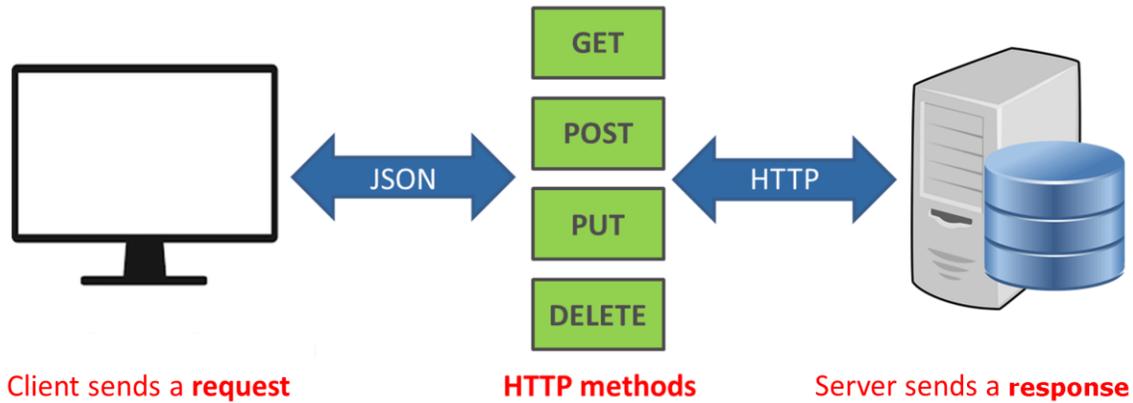
Es una forma de describir la forma en que los programas o los sitios webs intercambian datos. El formato de intercambio de datos normalmente es JSON o XML.

¿Para qué utilizar un API?

- Ofrecer datos a aplicaciones que se ejecutan en un móvil
- Ofrecer datos a otros desarrolladores con un formato más o menos estándar.
- Ofrecer datos a nuestra propia web/aplicación
- Consumir datos de otras aplicaciones o sitios Web (Arquitectura de una API REST
· Desarrollo de aplicaciones web, s.f.)

Figura 1

Arquitectura API Rest



5. MATERIALES Y MÉTODOS

Para el desarrollo de este proyecto de titulación se utilizó con conjunto de materiales de oficina, software y equipos, los cuales ayudaron en la construcción e implantación del sistema; también se aplicó un conjunto de métodos de investigación como el deductivo, inductivo y analítico, permitiendo comprender a fondo el objeto de estudio; como apoyo también se empleó las técnicas de recolección de datos como observación y entrevista. Para la gestión del proyecto se utilizó la metodología Iconix, donde en cada fase se logró cumplir con los objetivos planteados de forma que se dio solución al problema de manera eficiente. A continuación, se detalla cada uno de estos:

5.7. Materiales

Dentro de los materiales utilizados en el proyecto de investigación se tiene los siguientes:

Dentro del recurso humano se tiene la total participación del tesista y como asesoramiento al director de tesis, quien apoyo en la ejecución correcta del proyecto.

Tabla 1

Recursos humanos utilizados para el proyecto

Nombre	Cargo	Horas	Valor U.	Valor T
Lady Ordoñez	Egresado	400	\$5.00	\$2,000.00
Ing. Diego Chamba	Director de Tesis	15	\$7.00	\$105.00

TOTAL \$2,105.00

En la siguiente tabla se muestra el equipamiento necesario para construir el sistema.

Tabla 2
Recursos Hardware y software

Descripción	Cantidad	Valor U.	Valor T.
Hardware			
Laptop	1	\$1000	\$1000
Impresora	1	\$50	\$50
Flash memory 8GB	1	\$10	\$10
Software			
Windows 10 Home	1	\$169	\$169
Paquete Office	1	\$119.99	\$119.99
Visual Studio Code	1	\$0	\$0
StartUML	1	\$0	\$0
TOTAL			\$1348.99

A continuación, se presenta una tabla de imprevistos que puede haber al desarrollar la tesis.

Tabla 3
Recursos de oficina

Descripción	Cantidad	Valor U.	Valor T.
Tinta de impresora	1	\$20	\$20
Resma de papel bond	1	\$10	\$10
Transporte	1	\$10	\$10
Internet	20 horas	\$0.50	\$10
TOTAL			\$50

A continuación, se detalla el costo total de la construcción del proyecto, así como de ciertos imprevistos que puedan ocurrir.

Tabla 4
Presupuesto total del proyecto

Descripción	Valor
Recursos Humanos	\$2,105.00
Materiales y Herramientas	\$1,348.99
Imprevistos	\$50.00
TOTAL	\$3,503.00

5.2. Métodos de investigación

Dentro de los métodos de investigación se aplicó los siguientes:

5.2.1. Deductivo

Método empleado en el proyecto partiendo desde el objetivo general, se va concluyendo con cada una de las funcionalidades del sistema, que resolverán los problemas actuales presentes en la microempresa y que finalizará con una implementación y capacitación.

5.2.2. Inductivo

Este método se lo aplica en el proyecto, al momento que se determinan cada uno de los inconvenientes que tienen los usuarios que trabajan en la microempresa, con el propósito de llegar a la construcción del sistema basado en las necesidades reales de estos usuarios y que al final se cumpla con sus expectativas.

5.2.3. Analítico

Este método se lo aplica al momento de interpretar y analizar detalladamente los requisitos del usuario y los diferentes diagramas que comprendieron el análisis de tal manera que, al guiarse en estos se puede estructurar el diseño funcional del sistema. Además, se lo emplea al analizar cada una de las pruebas realizadas al software, con la finalidad de entregar un producto de calidad.

5.3. Técnicas de recolección de datos

5.3.1. Observación:

Consiste principalmente en ver los problemas que presenta la microempresa y sus usuarios, además servirá para la recolección de información en forma preliminar.

5.3.2. Entrevista:

Se aplican para recolectar la información de forma más precisa y clara directamente de las personas involucradas en el uso del sistema, de tal manera que a la hora de hacer los requerimientos sean en base a las necesidades existentes.

5.4. Metodología:

Para el desarrollo del proyecto se aplicó la metodología Iconix, la cual constituye una guía para planificar y organizar el proceso de desarrollo. Esta metodología se guía a través de los casos de uso y sigue un ciclo de vida iterativo e incremental, es decir, que permite dividir el proyecto en fases, cada fase está dividida en iteraciones ofreciendo como resultado un incremento del producto que mejorará las funcionalidades en el desarrollo del sistema.

Las fases que tiene Iconix son:

5.4.1. Análisis de requisitos

Se realizó el levantamiento de información, permitiendo definir los requisitos del sistema, además se logró realizar el modelo del dominio de los objetos de la vida real cuyo comportamiento o datos deben ser almacenados en el sistema.

A partir de este modelado, se realizó un pequeño prototipo basado en el storyboard de la interfaz gráfica obtenida previamente, el cual se mostrará al cliente y se refinará en sucesivas reuniones. Normalmente este prototipo suele converger en dos o tres iteraciones.

Una vez definido el prototipo y los requisitos del sistema por parte del equipo de desarrollo y en acuerdo con el cliente, se procedió a realizar los diagramas de casos de uso.

5.4.2. Análisis y diseño preliminar

A partir de los casos de uso se realizó las fichas o especificaciones de casos de uso, en las cuales se pone el nombre (hace referencia al caso de uso), una descripción, precondiciones que debe cumplir para que el caso de uso pueda iniciar sin dificultades, postcondiciones que debe cumplir al finalizar de forma correcta, flujos normales todo lo que ocurre si es correcto, flujo alternativo en caso de ocurrir algún problema.

Después se realizó los diagramas de robustez los que permitieron añadir nuevas relaciones en el diagrama de clases e implicar el análisis narrativo de los casos de uso.

5.4.3. Diseño

Se realizó los diagramas de secuencia, los cuales derivan de las fichas de casos de uso quienes implican los casos de uso y por ende los requisitos.

5.4.4. Implementación

En esta fase se realizó el código, para eso se tomó como base fundamental las fases de análisis de requisitos, análisis preliminar y diseño. Para la codificación se lo hizo con lenguaje de programación Java empleando el framework Spring Boot. Además, se aplicó pruebas de funcionalidad para garantizar la calidad y eficiencia.

6. RESULTADOS

En base a la metodología Iconix, el proyecto de titulación se ha estructurado en 4 fases, y en cada una de estas se realizó un conjunto de resultados que permitieron implementar una solución óptima. A continuación, se detalla cada una de estas:

6.1. Análisis de requisitos

La primera fase consistió en conocer y analizar el territorio, el campo de estudio, identificar como está estructurada la empresa, los problemas que tienen, las actividades que se ejecutan, además de obtener los requisitos con los que el sistema funcionará.

Esta fase comprende: descripción de empresa, problemática de la situación actual, requisitos obtenidos mediante una ficha de observación y entrevista, modelo del dominio, prototipo y diagrama de casos de uso. A continuación, se detalla cada uno de estos:

6.1.1. Descripción de la empresa

My Web Service creada en el año de 2020, se dedica a la solución de problemas mediante la construcción o desarrollo de sistemas informáticos personalizados o a la medida de las necesidades del usuario. Se encuentra ubicada en la ciudad de Yantzaza provincia de Zamora Chinchipe.

6.1.2. Problemática de la situación actual

Al ser una empresa dedicada a la automatización de procesos mediante el desarrollo de software, debe llevar el control del flujo de trabajo de los procesos que se realizan, tales como: Gestión de Proyectos, Gestión de Actividades y avances; al no contar con un sistema que optimice la ejecución de estos procesos, provoca que no se dé un seguimiento y control oportuno a las actividades encargadas.

6.1.3. Requisitos

Al aplicar la técnica de recolección de información (Entrevista y ficha de observación), ayudaron a determinar las funcionalidades del sistema.

6.1.3.1. Entrevista:

Con la entrevista aplicada al propietario se obtuvieron los siguientes datos: las actividades que se realizan: gestión de los proyectos de software, administración de los usuarios, gestión de las actividades y monitoreo de los avances. Los procesos o pasos que se realizan para realizar cada una de estas actividades, cantidad de personas que trabajan en la empresa y las actividades que tienen como responsabilidad.

También, se preguntó al propietario si estaría de acuerdo en el desarrollo e implantación de un sistema que les ayude a optimizar los procesos que se realizan en la empresa, además si está dispuesto a brindar toda la información, documentación y apoyo necesario para el desarrollo e implantación del sistema, como respuesta menciono que está en total acuerdo. La entrevista aplicada se la encuentra en el Anexo 2.

6.1.3.2. Ficha de observación:

Al aplicar la ficha de observación en la microempresa se constató que no se dispone de una herramienta informática que permite realizar un adecuado seguimiento y control de las actividades que realiza cada desarrollador de acuerdo al proyecto que se le ha asignado. (Ver anexo 3)

6.1.3.3. Requerimientos funcionales y no funcionales:

En base a la información obtenida se estableció los requisitos funcionales del sistema, los cuales definen las funcionalidades del sistema. También se determinó los requisitos no funcionales, los cuales determinan las restricciones que tiene el sistema.

Tabla 5
Requisitos funcionales

Código	Requerimiento
RF-01	El sistema controlará el acceso y permitirá que ingresen solo el personal autorizado con su correo y contraseña.
RF-02	El sistema permitirá al usuario recuperar su contraseña de ingreso al sistema.
RF-03	El sistema permitirá registrar, actualizar, activar o inactivar a los usuarios que manipularan el sistema.
RF-04	El sistema enviará un correo al usuario cuando se haga la creación de su cuenta en el sistema.
RF-05	El sistema permite al administrador gestionar los usuarios
RF-06	El Sistema permite al administrador gestionar la información de los proyectos y asignar responsables
RF-07	El Sistema permite al administrador del Proyecto, gestionar las actividades y avances realizados
RF-08	El Sistema permite al Desarrollador, gestionar los avances de las actividades

Tabla 6
Requisitos no funcionales

Código	Requerimiento
RNF-01	El sistema es adaptativo para dispositivos móviles
RNF-02	El sistema utiliza base de datos PostgreSQL version 14
RNF-03	El Sistema utiliza el lenguaje de programación Java version 11
RNF-04	El sistema utiliza el motor de plantillas Thymeleaf

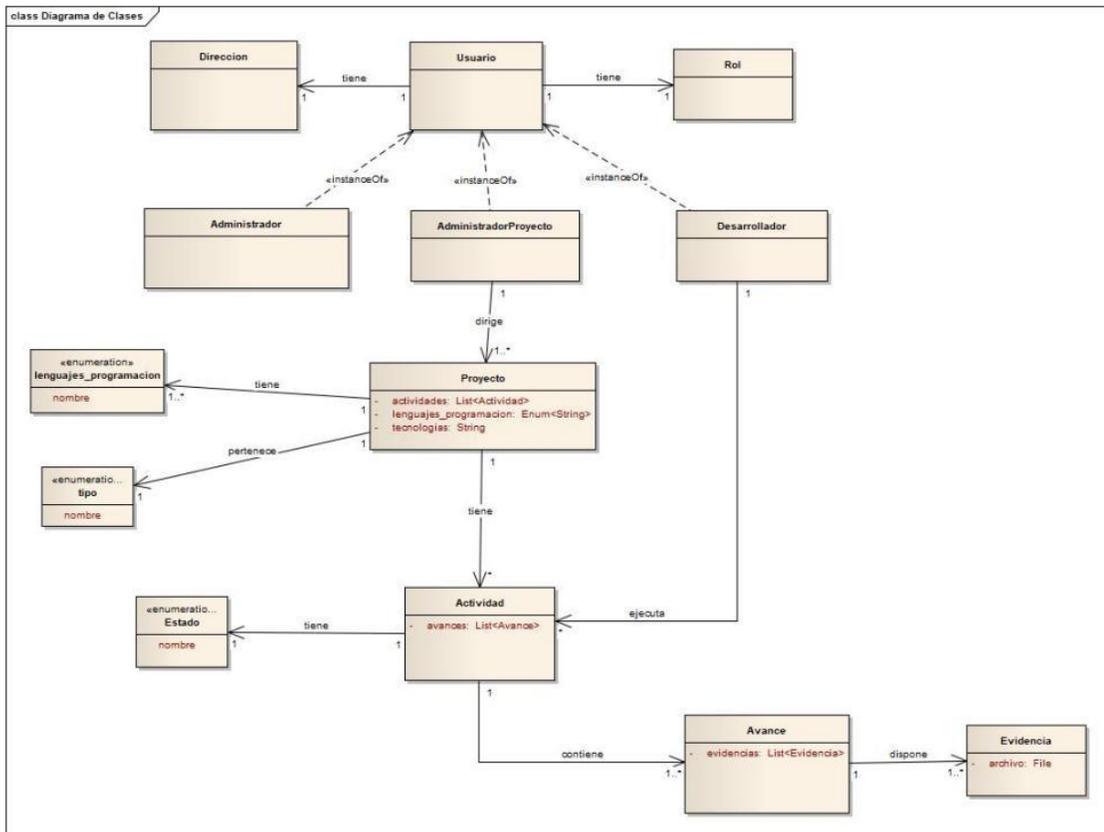
- RNF-05 El sistema se despliega en el servidor de aplicaciones Tomcat version 10
- RNF-06 El sistema exige la autenticación de usuarios
- RNF-07 El sistema brinda confiabilidad, rendimiento y seguridad de los datos.

6.1.4. Modelo del dominio

El modelo del dominio permitió determinar de forma sencilla los objetos que serán almacenados y manipulados en el sistema. El modelo del dominio es un diagrama de clases bastante simplificado, sin embargo, se logra evidenciar todas las entidades que están relacionadas para que funcione el sistema.

Figura 2

Modelo del dominio de la empresa My Web Service



6.1.5. Prototipo

Con la finalidad de tener una vista de la interfaz de usuario, se realizó el prototipo basado en el storyboard que fue mostrado al cliente y se logró acordar el diseño de la interfaz gráfica.

Figura 3

Prototipo Pagina de Bienvenida Administrador

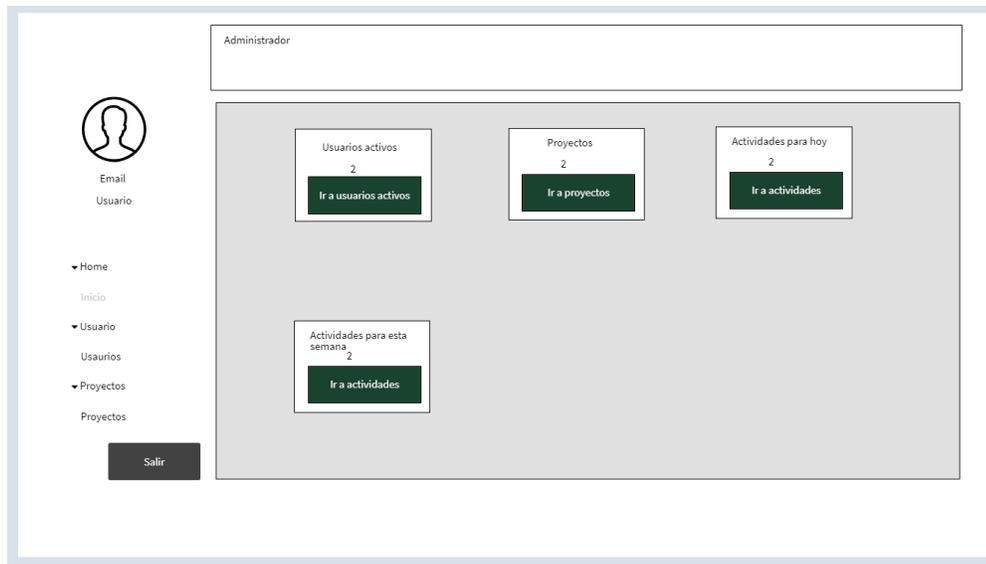


Figura 4

Prototipo Pagina de Usuarios activos

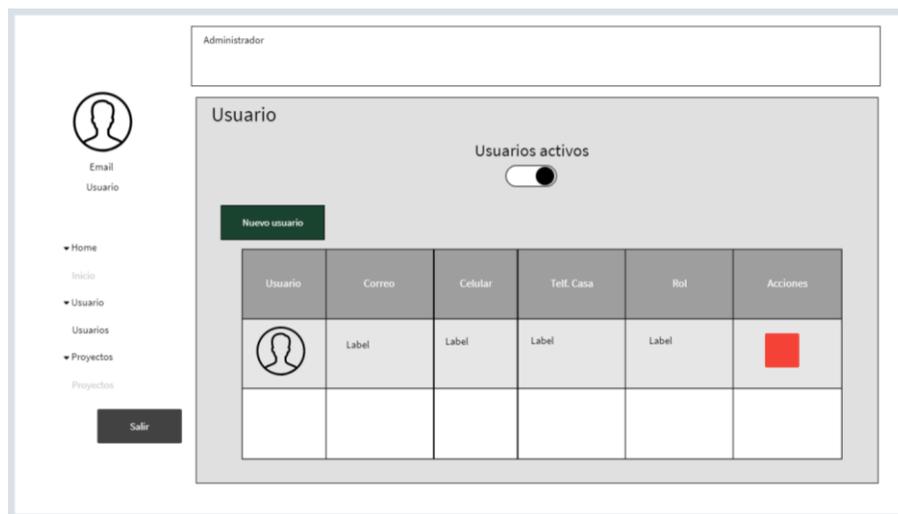


Figura 5

Prototipo Pagina de Proyectos

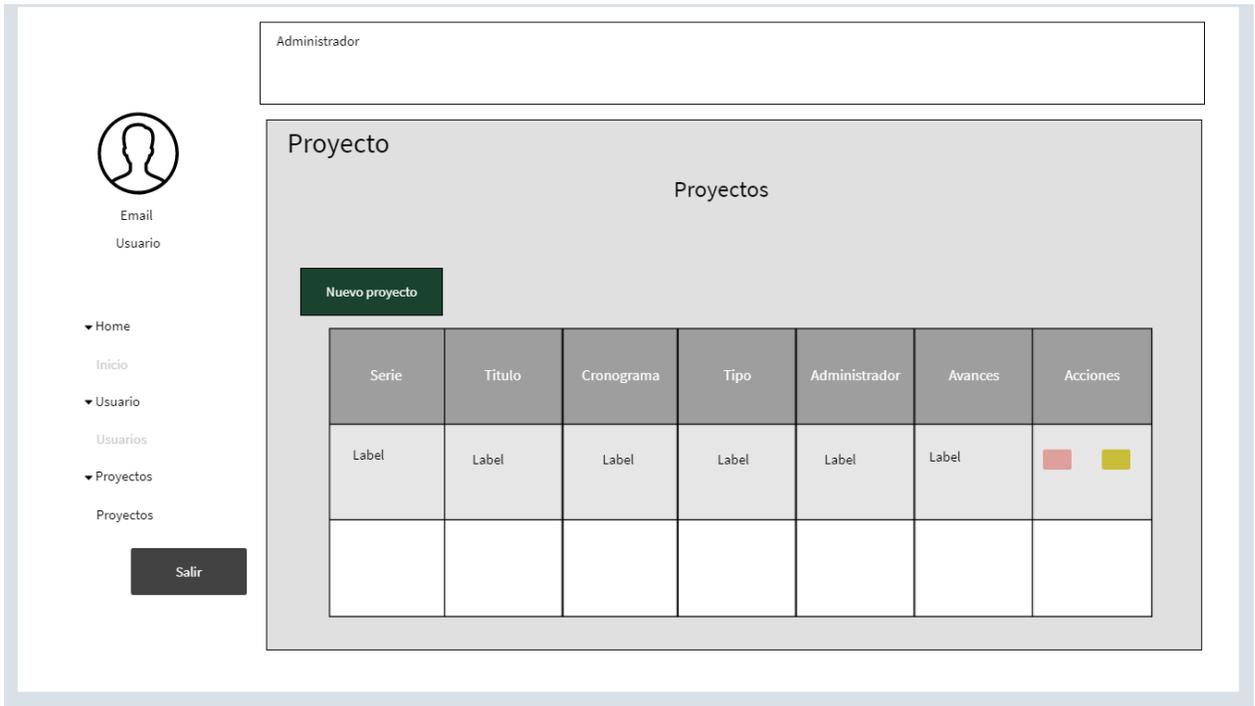


Figura 6

Prototipo Pagina de Proyectos Step 1: Datos del proyecto

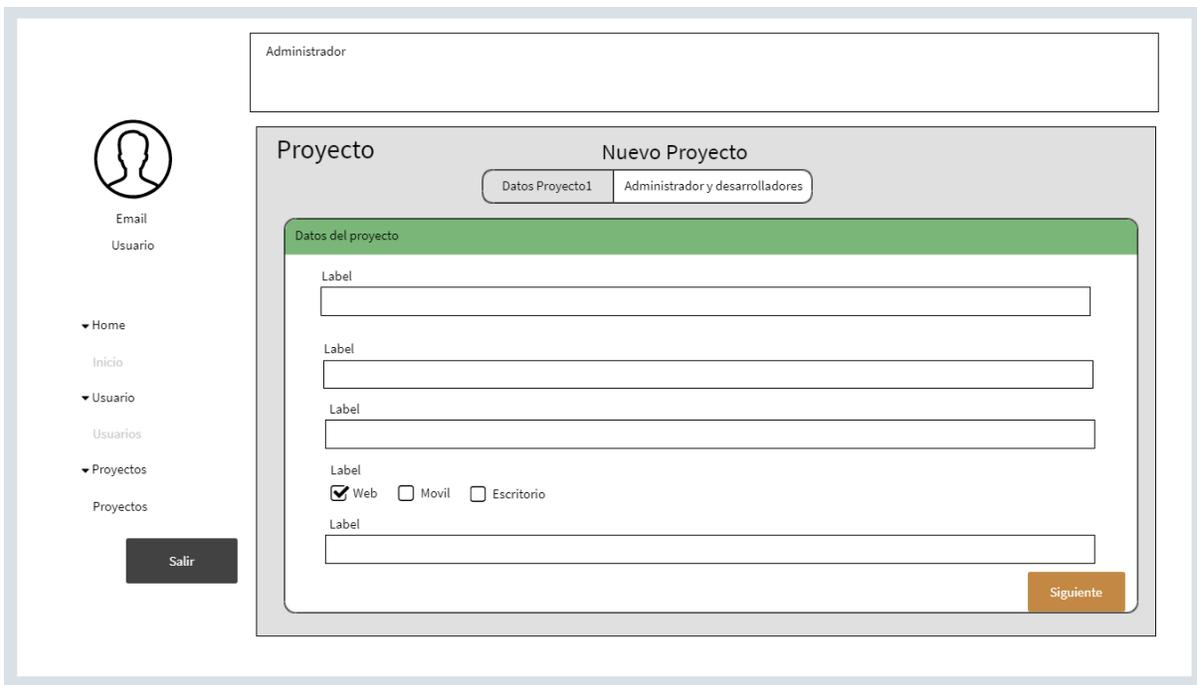


Figura 7

Prototipo Pagina de Proyectos Step2: Administrador y desarrolladores

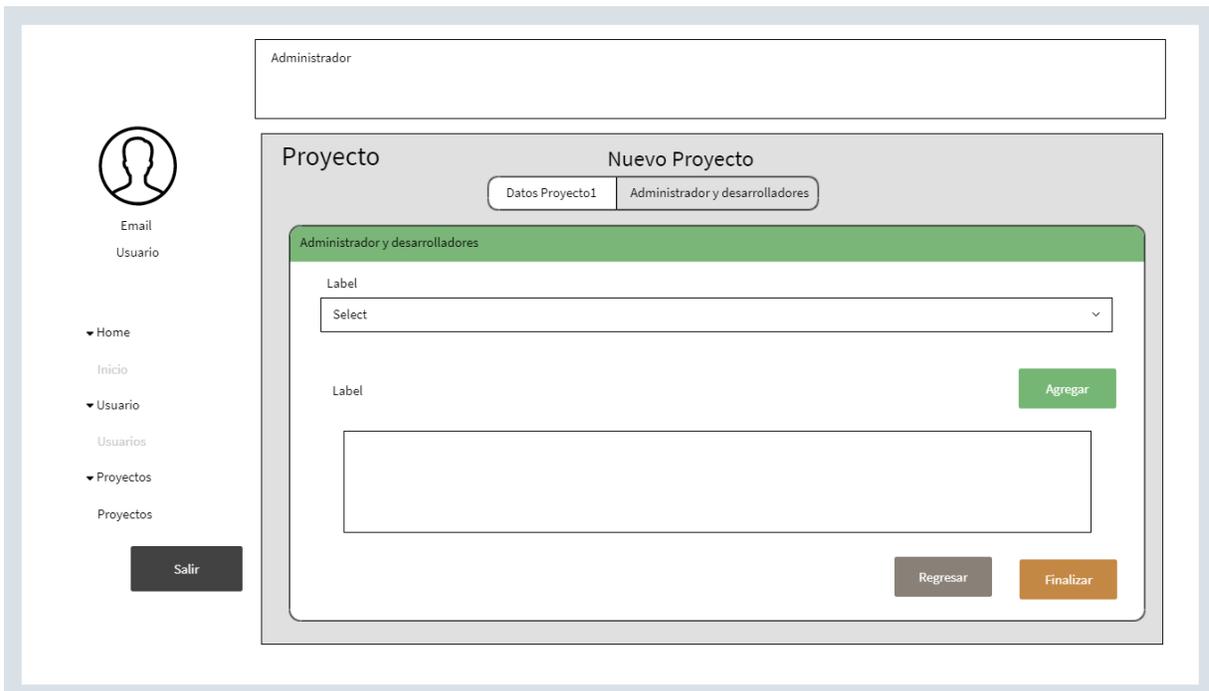


Figura 8

Prototipo Pagina de Administrar actividades

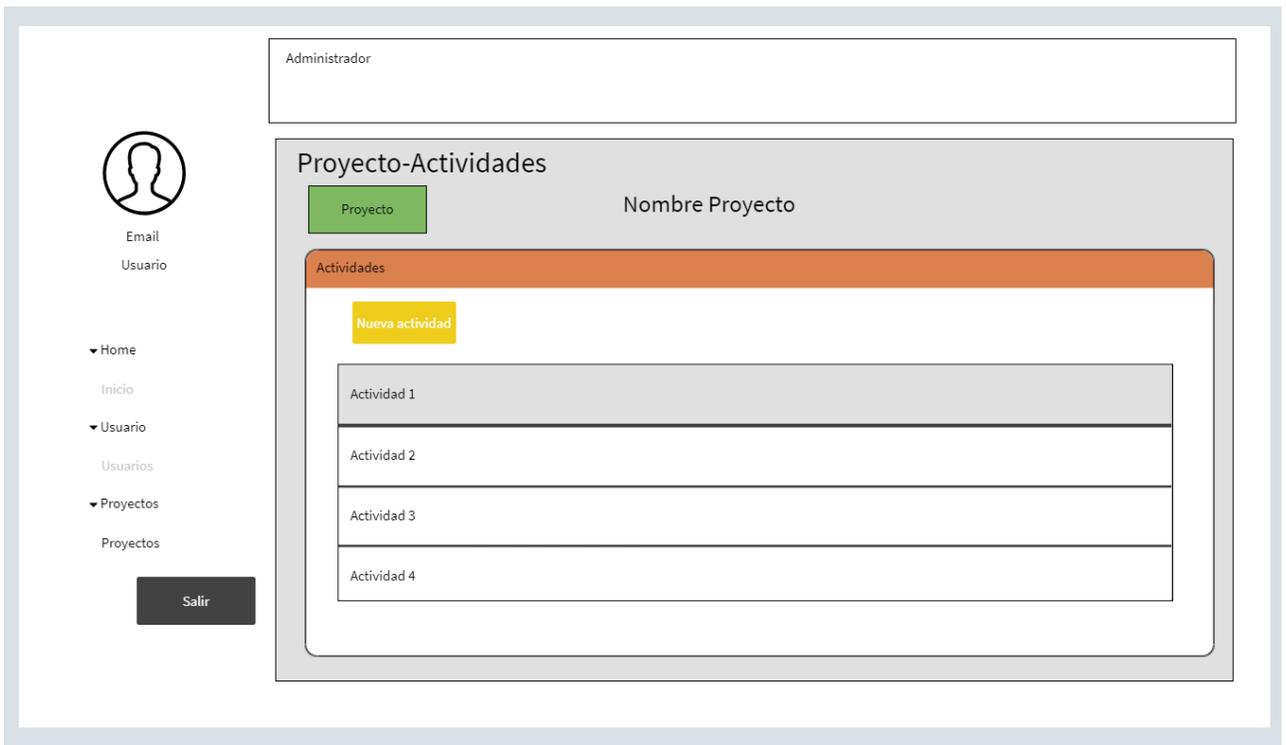


Figura 9

Prototipo Pagina de Gestionar avances

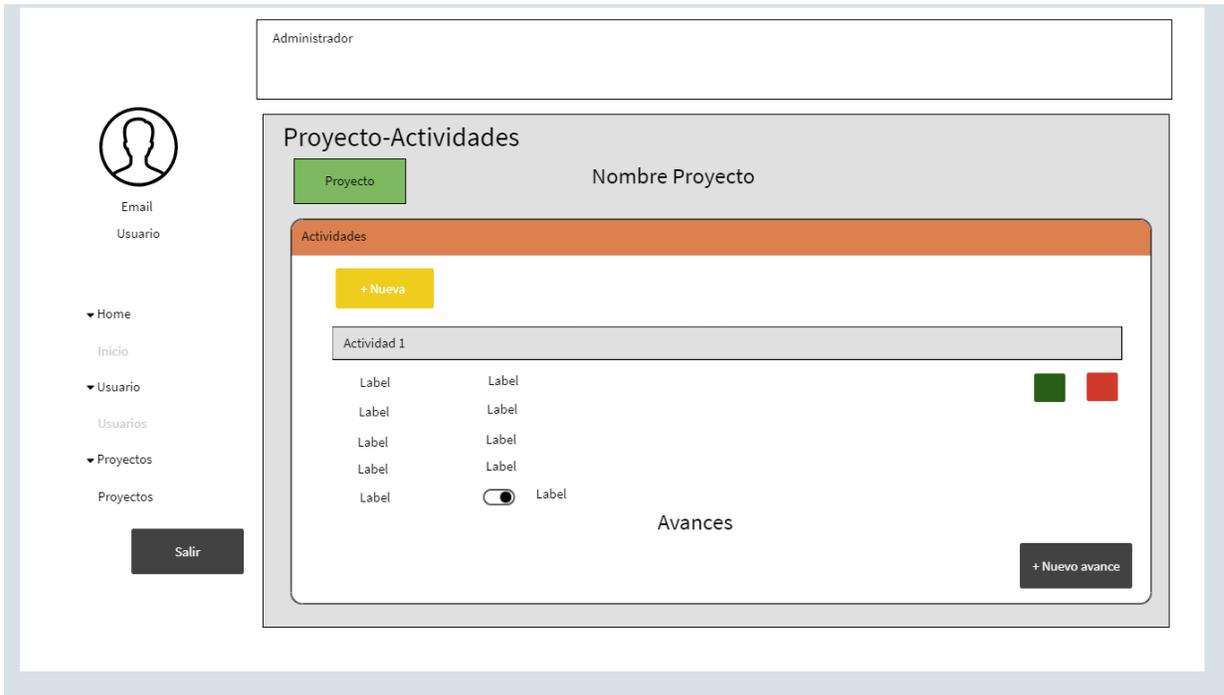


Figura 10

Prototipo Página de Inicio de sesión

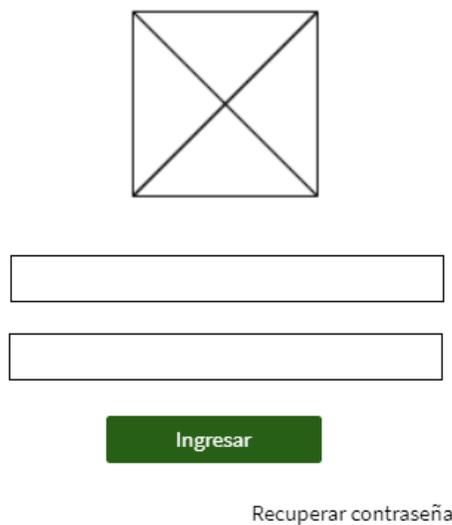
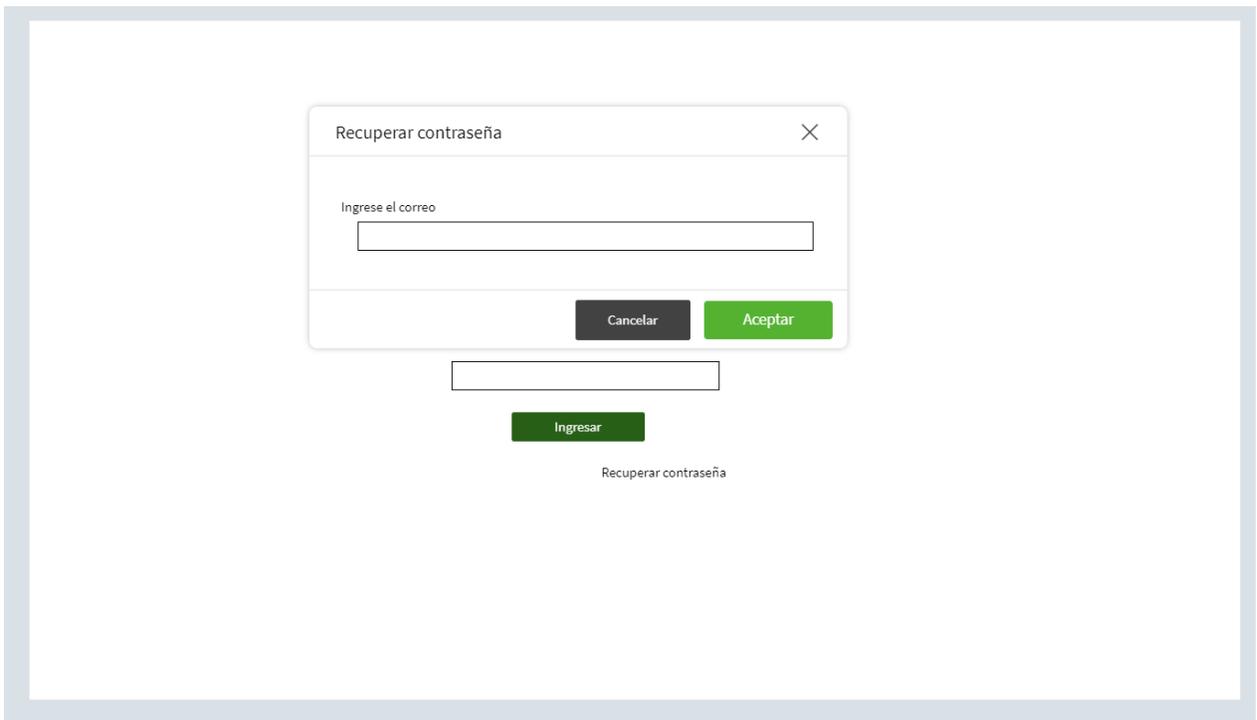


Figura 11

Prototipo Pagina de Recuperar contraseña



6.1.6. Diagrama de casos de uso

La elaboración de los casos de uso permitió capturar las funcionalidades del sistema tal como se muestra al usuario.

Figura 12

Diagrama de casos de uso Administrador

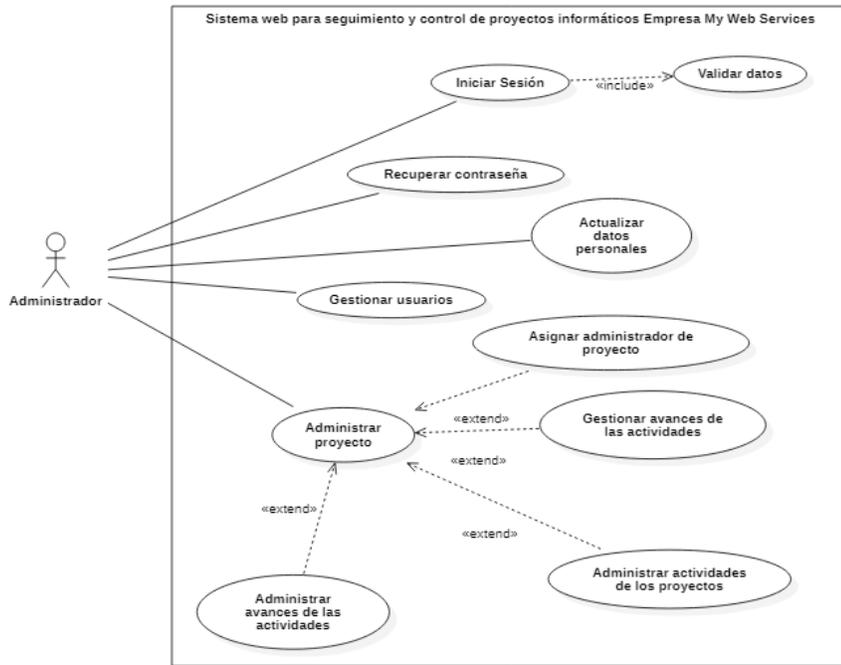
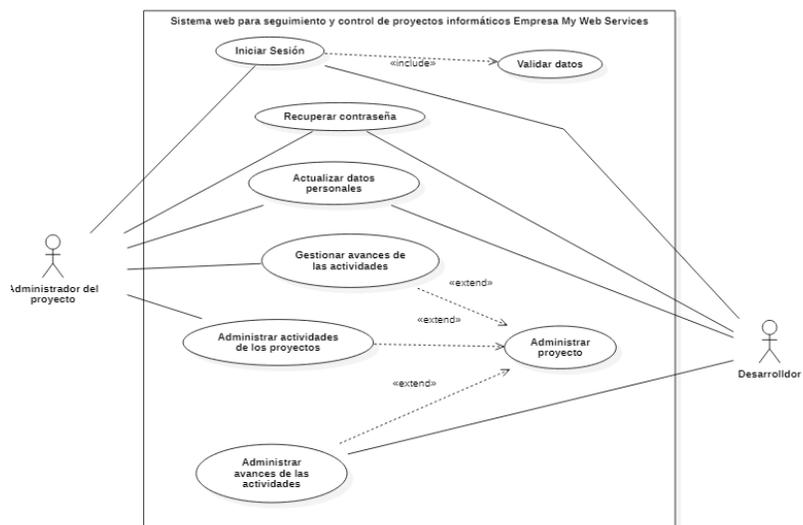


Figura 13

Diagrama de casos de uso Administrador del proyecto y desarrollador



6.2. Revisión del análisis y diseño preliminar

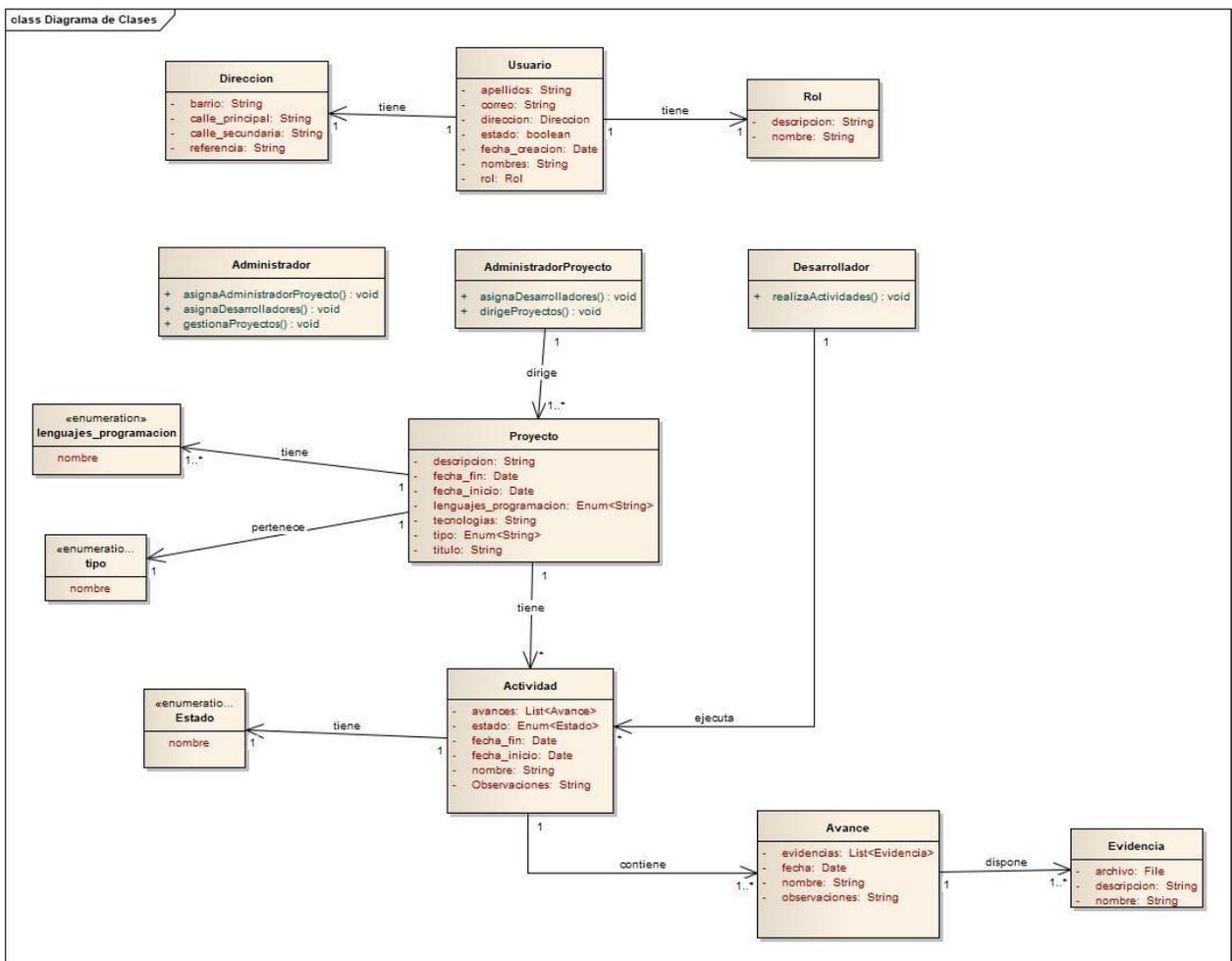
Una vez realizado el análisis de requisitos se procede con el análisis y diseño preliminar, para identificar detalladamente la arquitectura funcional del software y la lógica de negocio

6.2.1. Diagrama de clases

Con el diagrama de clases se puede evidenciar de forma detallada la descripción de las clases que conforman el modelo del sistema informático

Figura 14

Diagrama de Clases



6.2.2. Especificaciones de casos de uso

El desarrollo de la ficha de casos de uso, se lo hace con la finalidad de tener el caso de uso con una descripción textual de todas las maneras que los actores previstos podrían trabajar con el software. En estas se describen los pasos que el actor sigue para realizar determinada tarea.

Tabla 7
Especificación de caso de uso: Iniciar sesión

Caso de uso	Iniciar sesión		
Código	CU01	RF-01	Inicio de sesión
Actores	Usuarios (Administrador, Administrador proyecto)		
Descripción	El presente caso de uso les permitirá ingresar al sistema, para ello deberá ingresar usuario y contraseñas correctas.		
Precondición	El usuario debe tener una cuenta en el sistema		
Postcondición	Las acciones que corresponden al tipo de perfil o rol del usuario		
Secuencia Normal (Curso normal)			
Acción			
	Paso	Actor	Sistema
	1	El usuario ingresara la siguiente url en el navegador de su preferencia	
	2		El sistema muestra la ventana principal del sistema
	3	El usuario seleccionara la opción Acceder , ubicada en la parte superior derecha de la ventana presentada.	

	4		El sistema muestra un formulario con: dos campos de texto (Nombre de usuario, Contraseña), y un botón de Acceder.
	5	El usuario ingresa sus datos en los campos Nombre de usuario* y Contraseña* , luego da clic en el botón Acceder	
	6		EL sistema presenta la ventana del usuario acorde al rol/perfil que este tenga.
Excepciones (Curso alterno)			
	Acción		
	Paso	Actor	Sistema
<i>Datos erróneos o inexistentes</i>	5	El usuario ingresa sus datos en los campos Nombre de usuario* y Contraseña* , luego da clic en el botón Acceder	
	6		El sistema muestra un mensaje de Datos erróneos. Por favor, inténtelo otra vez.

Tabla 8
Especificación de caso de uso: Administrar Proyecto

Caso de uso	Administrar proyecto		
Código	CU05	RF-04, RF-05, RF-07-RF-08, RF-09, RF-10	Administrar proyecto

Actores	Usuario(Administrador, Administrador del proyecto, Desarrollador)		
Descripción	El presente caso de uso les permitirá al administrador gestionar el proceso de proyectos y realizar las actividades de: Crear proyecto, Editar proyecto, y más adelante gestionar el proyecto (Administrar actividades, gestionar avances).		
Precondición	El actor debe tener acceso a trabajar con los procesos de administrar el proyecto.		
Postcondición	El usuario podrá, asignar administrador de proyecto y desarrolladores, administrar actividades y gestionar avances de proyectos.		
Secuencia Normal (Curso normal)			
Escenario 1: Crear proyecto			
	Acción		
	Paso	Actor	Sistema
	1	El usuario selecciona la opción Proyectos	
	2		El sistema muestra las opciones: Nuevo proyecto , Listado de proyectos
	3	El usuario elige la opción Nuevo proyecto	
	4		El sistema muestra una pantalla con 2 steps.
	5	El usuario ingresa datos en el step 1: Titulo, Fecha Inicio, fecha fin, descripción, selecciona el tipo, selecciona lenguajes de programación, selecciona tecnologías y selecciona Siguiente.	

	6		El sistema muestra el step 2
	7	El usuario seleccionar administrador y agrega desarrolladores, presiona Finalizar	
	8		El sistema muestra la información ingresada, actualiza la lista de proyectos y presenta mensaje: Proyecto registrado con éxito.

Escenario 2: Administrar actividades

	Acción		
	Paso	Actor	Sistema
Administrador-Administrador del proyecto	1	El usuario selecciona la opción Proyectos	
	2		El sistema muestra las opciones: Nuevo proyecto , Listado de proyectos
	3	El usuario elige de la lista de proyectos la opción Actividades el proyecto correspondiente.	
	4		El sistema muestra una pantalla con: Nueva actividad, Lista de actividades.
	5	El usuario selecciona el botón Nueva actividad.	
	6		El sistema carga un formulario con la información para llenar la actividad.

	7	El usuario ingresa y seleccionar los datos necesario y agrega la actividad.	
	8		El sistema valida la información y envía un mensaje: Actividad creada con éxito.

Escenario 3: Gestionar avances

	Acción		
Administrador- Administrador del proyecto	Paso	Actor	Sistema
	1	El usuario selecciona la opción Proyectos	
	2		El sistema muestra las opciones: Nuevo proyecto , Listado de proyectos
	3	El usuario elige de la lista de proyectos la opción Actividades el proyecto correspondiente.	
	4		El sistema muestra una pantalla con: Nueva actividad, Lista de actividades.
	5	El usuario selecciona la actividad correspondiente.	

	6		El sistema carga un formulario con la información de la actividad: Cronograma, responsables, descripción, estado, opción de finalizar actividad, botones de edición, eliminación, sección avances, botón nuevo avance.
	7	El usuario ingresa y seleccionar los datos necesario y agrega el avance.	
	8		El sistema valida la información y envía un mensaje: Avance creado con éxito.
Excepciones (Curso alterno)			
	Acción		
	Paso	Actor	Sistema
Crear proyecto: No ha ingresado datos obligatorios.	5	El usuario selecciona el Proyecto, e ingresa los datos, luego da clic en el botón Finalizar .	
	6		El sistema muestra un mensaje: Llenar datos obligatorios: Título, Fecha inicio, fecha fin, Tipo, Lenguaje de programación, tecnologías, descripción, Administrador y desarrolladores.

Tabla 9
Especificación de caso de uso: Gestionar usuarios

Caso de uso	Gestionar usuarios		
Código	CU02	RF-06	Gestión de usuario
Actores	Usuarios (Administrador, Administrador del proyecto, desarrollador)		
Descripción	El presente caso de uso le permitirá al administrador crear los usuarios (administrado, administrador del proyecto y desarrollador), editar, mostrar, dar de alta y dar de baja a usuarios.		
Precondición	El usuario debe tener una cuenta en el sistema para iniciar sesión.		
Postcondición	El usuario tipo administrador podrá tener el control de sistema, el usuario de tipo administrador y desarrollador solo podrán actualizar su información y foto personal.		
Secuencia Normal (Curso normal)			
Escenario 1: Crear usuario			
	Acción		
	Paso	Actor	Sistema
	1	El administrador selecciona la opción Usuarios del menú.	
	2		El sistema muestra una pantalla con: un botón Nuevo usuario, Usuarios activos.
	3	El administrador elige la opción Nuevo usuario	

	4		El sistema muestra un formulario con los datos del nuevo usuario: usuario, correo, celular, rol , y un botón de Crear usuario.
	5	El administrador selecciona el rol e ingresa los datos, luego da clic en el botón Crear usuario.	
	6		El sistema almacena la información ingresada, actualiza la lista de usuarios y presenta mensaje: Usuario creado con éxito.

Escenario 2: Editar usuario

Acción		
Paso	Actor	Sistema
1	El usuario en su perfil selecciona la opción editar.	
2		El sistema carga una pantalla con los datos de: nombre, celular, rol, teléfono, correo.
5	El administrador ingresa los datos respectivos.	

	8		El sistema actualiza la información editada, actualiza al usuario y presenta un mensaje: Usuario actualizado con éxito.
Escenario 4: Dar de baja usuarios			
	Acción		
	Paso	Actor	Sistema
	1	El administrador selecciona la opción Usuarios del menú.	
	2		El sistema muestra: Nuevo usuario, listado de usuarios activos.
	3	El administrador elige el usuario y seleccionar dar de baja.	
	4		El sistema carga una pantalla con
	5	El administrador elige la opción eliminar	
	6		El sistema muestra una pantalla de confirmación
	7	El administrador elige la opción Si	
	8		El sistema da de baja al usuario, y presenta un mensaje: Usuario dado de baja con éxito.

Excepciones (Curso alterno)			
	Acción		
	Paso	Actor	Sistema
No ha ingresado datos obligatorios.	7	El administrador selecciona Nuevo usuario, no ingresa todos los datos; presiona Crear usuario.	
	8		El sistema despliega un mensaje indicando al administrador que debe llenar los campos obligatorios.
Eliminar usuario	7	El administrador elige la opción No	
	8		El sistema retorna a la lista de usuarios

6.2.3. Diagramas de robustez

Esta técnica se utiliza para analizar los pasos de un caso de uso y validar su lógica de tal forma que se asegure que es bastante robusto.

Figura 15

Diagrama de robustez: Inicio de sesión

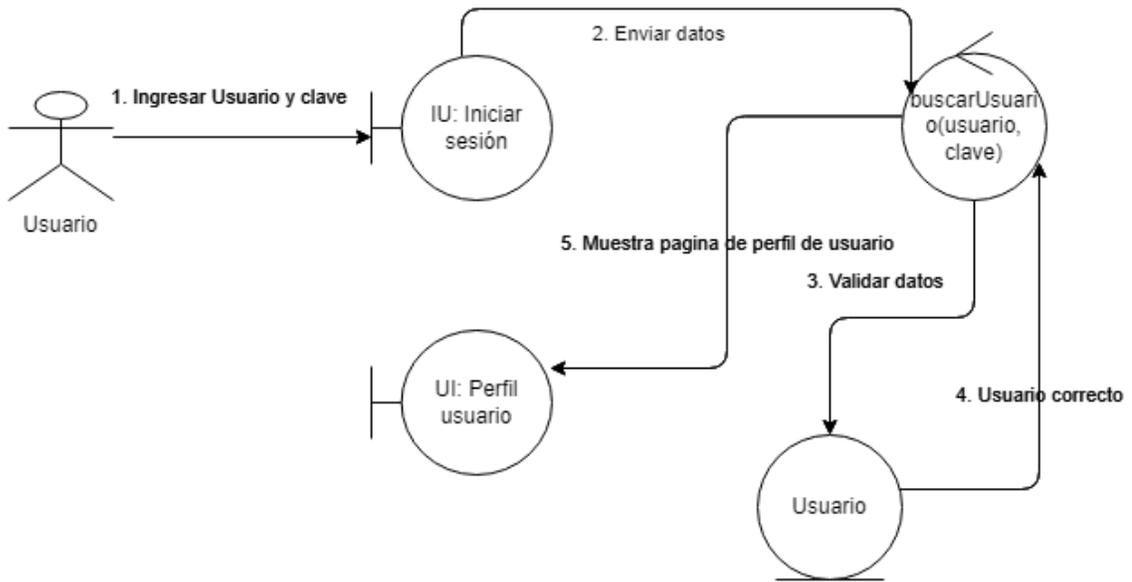


Figura 16
Diagrama de robustez: Recuperar contraseña

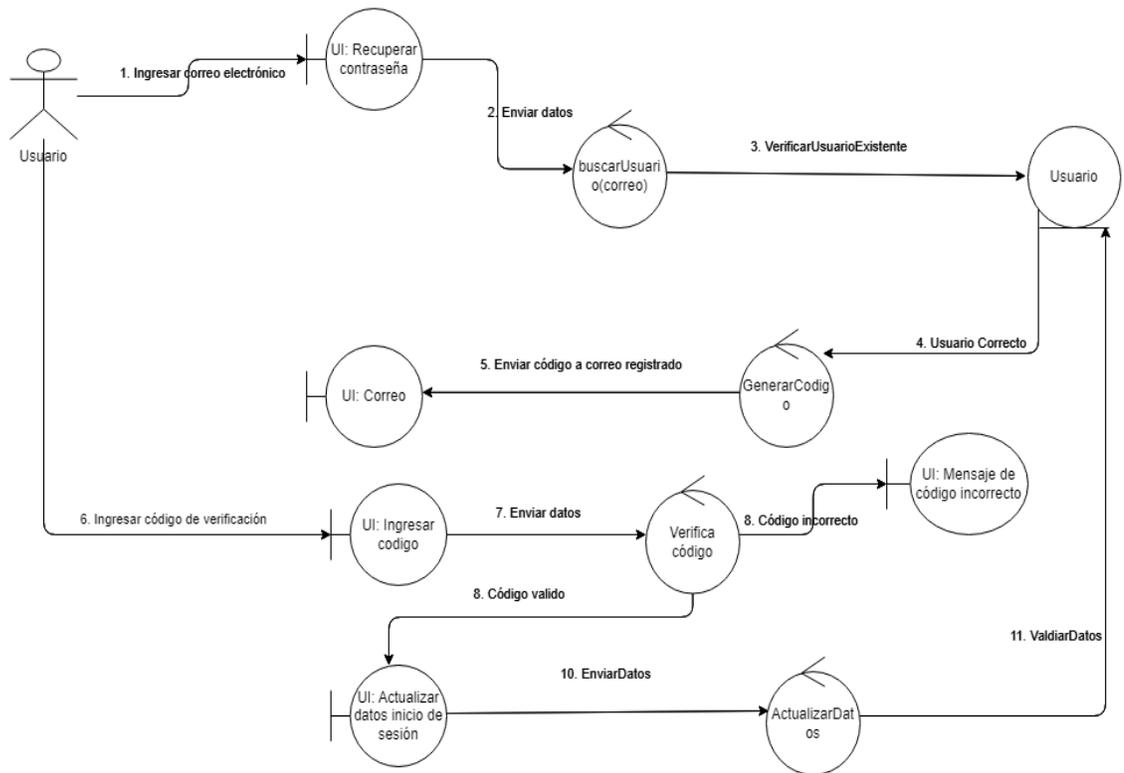


Figura 17

Diagrama de robustez: Administrar proyecto

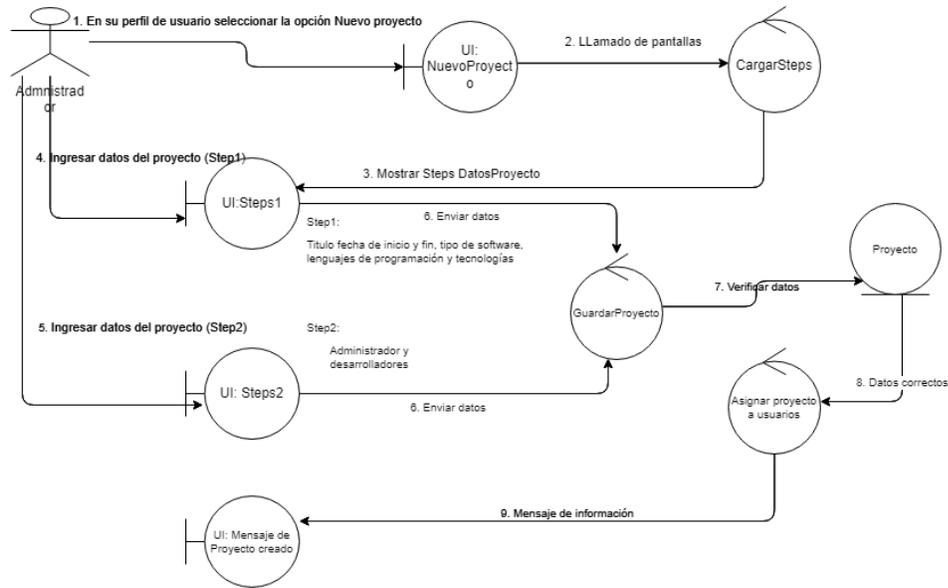


Figura 18

Diagrama de robustez: Administrar actividades

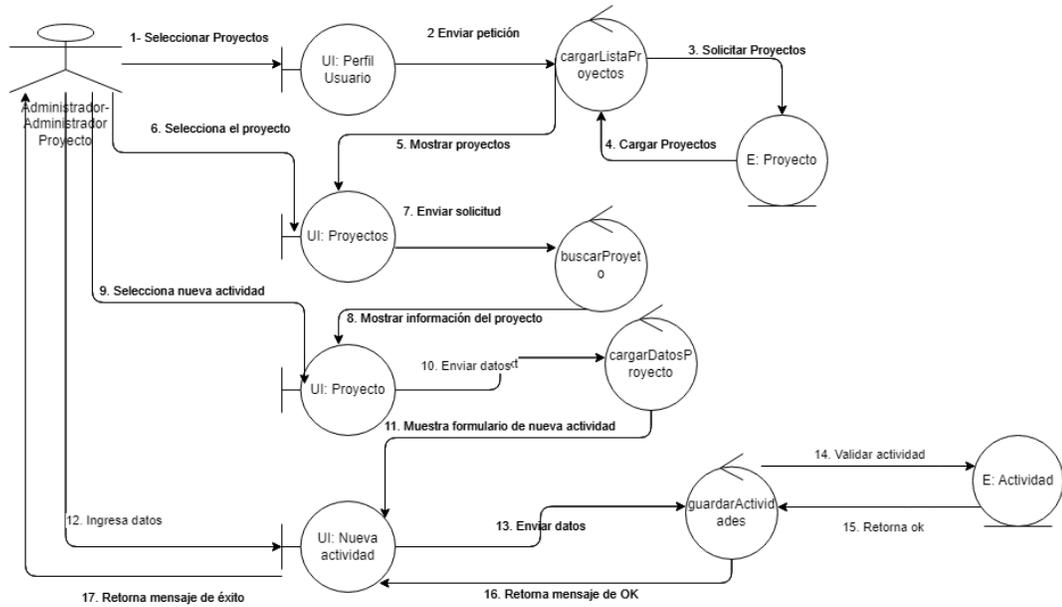


Figura 19

Diagrama de robustez: Gestionar avances

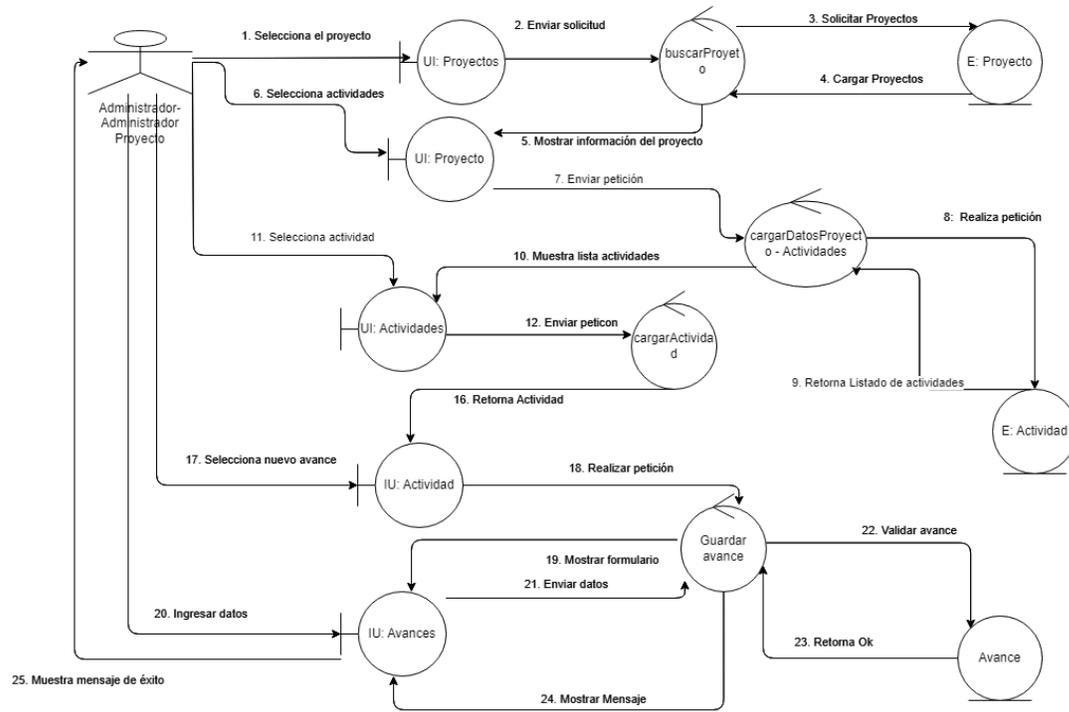


Figura 20

Diagrama de robustez: Gestionar usuarios - crear

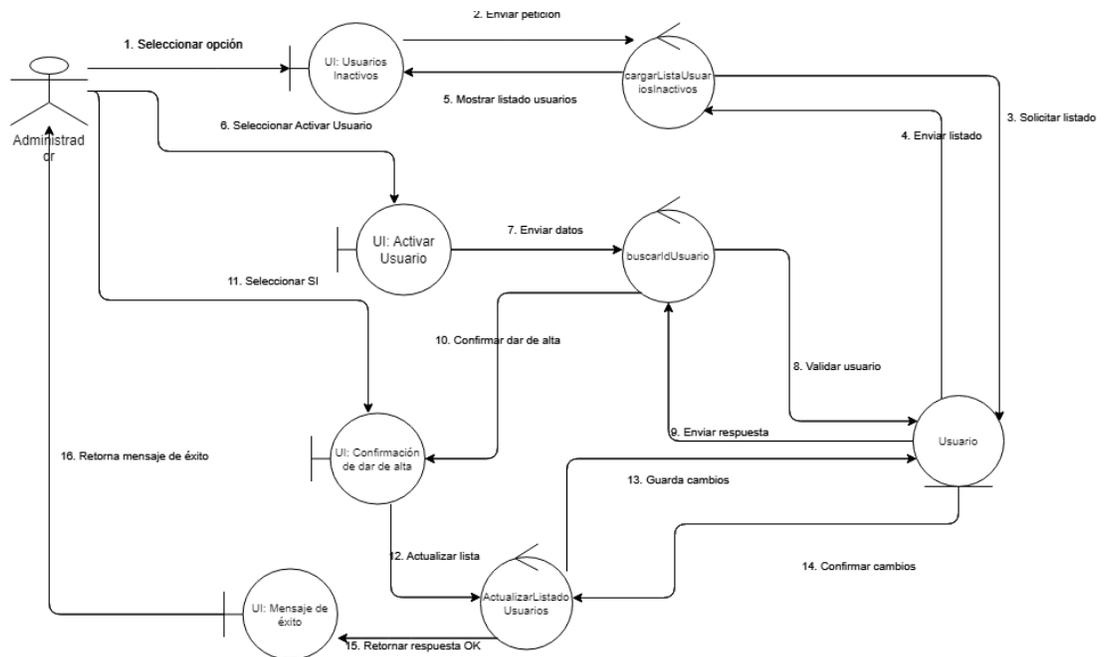


Figura 21

Diagrama de robustez: Gestionar usuarios - dar de baja

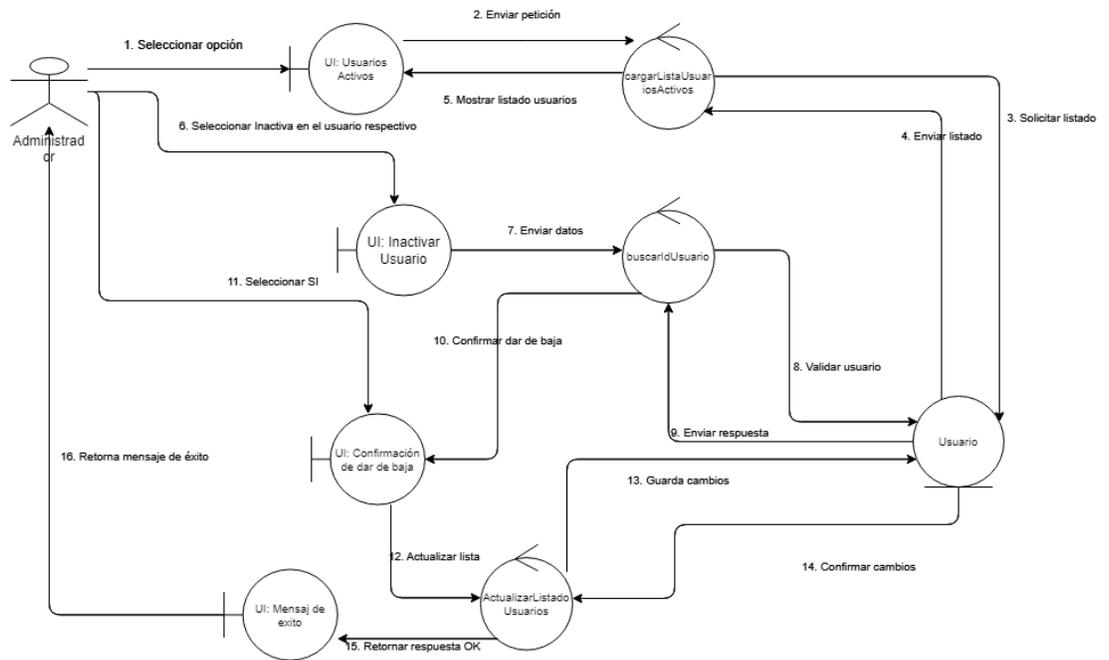


Figura 22

Diagrama de robustez: Gestionar usuarios - dar de alta

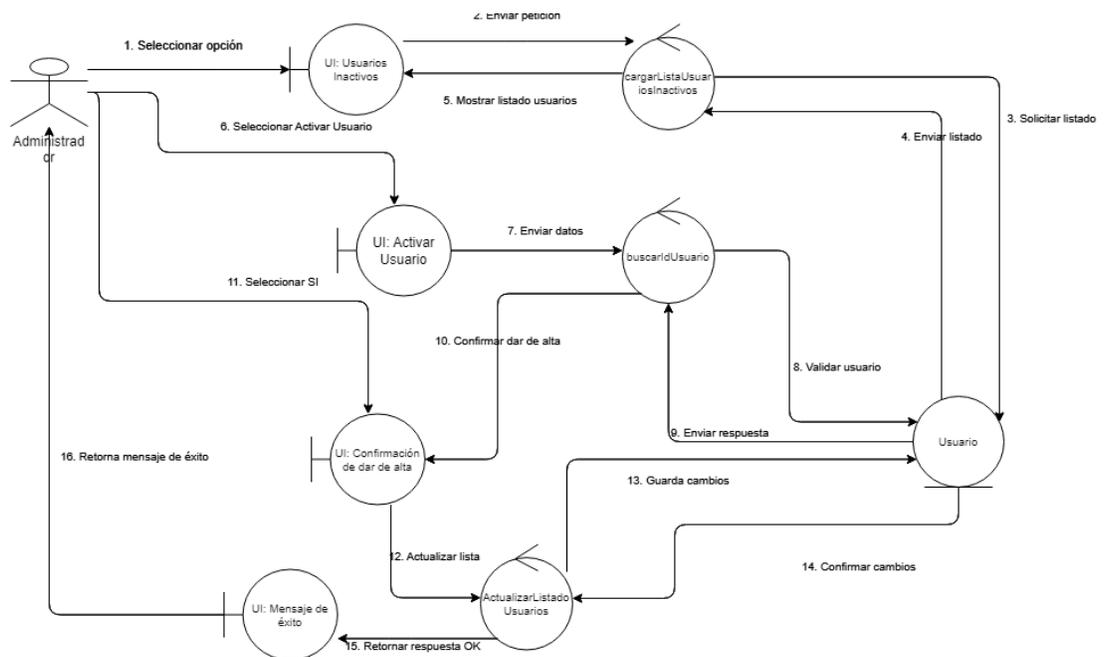
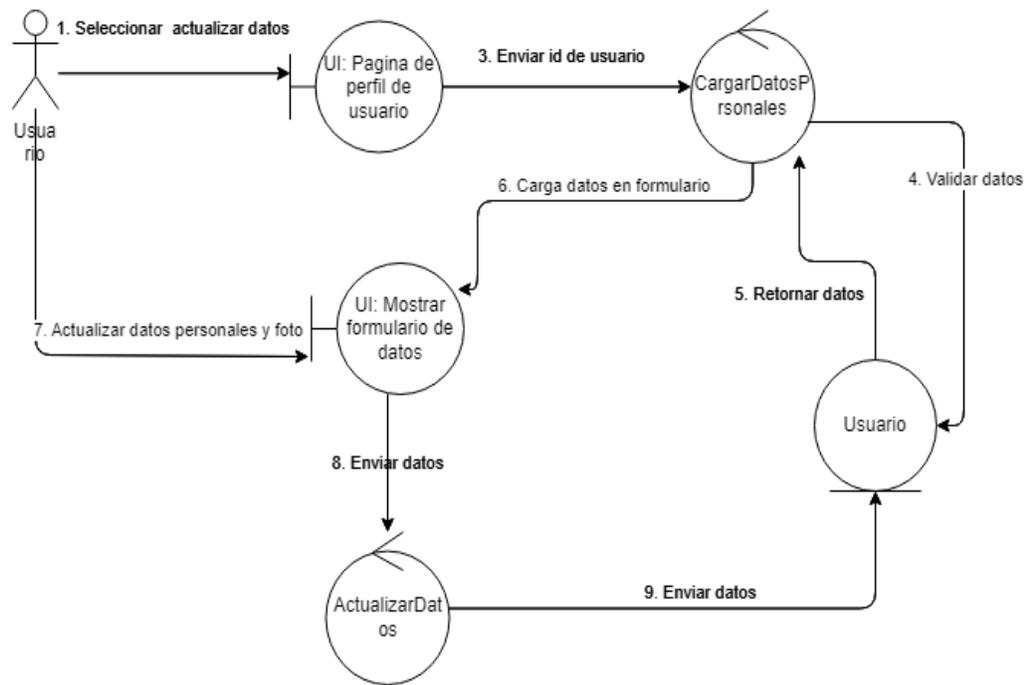


Figura 23

Diagrama de robustez: Actualizar datos



6.3. Diseño

En esta fase se procede a trabajar con los diagramas de secuencia los cuales modelan la interacción entre objetos y sistema. Cabe indicar que estos diagramas derivan directamente de las fichas de caso de uso que están relacionadas con los casos de uso que se relacionan con los requisitos.

Figura 24

Diagrama de secuencia: Administrar proyecto

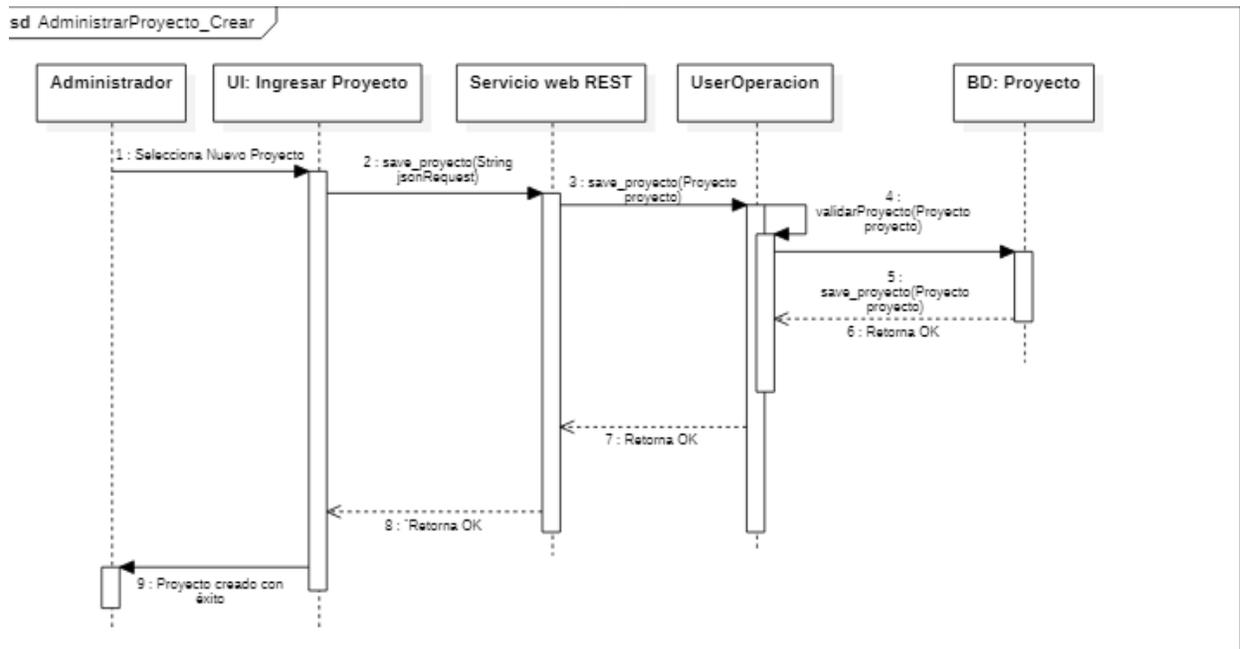


Figura 25

Diagrama de secuencia: Administrar actividades - crear - proceso REST

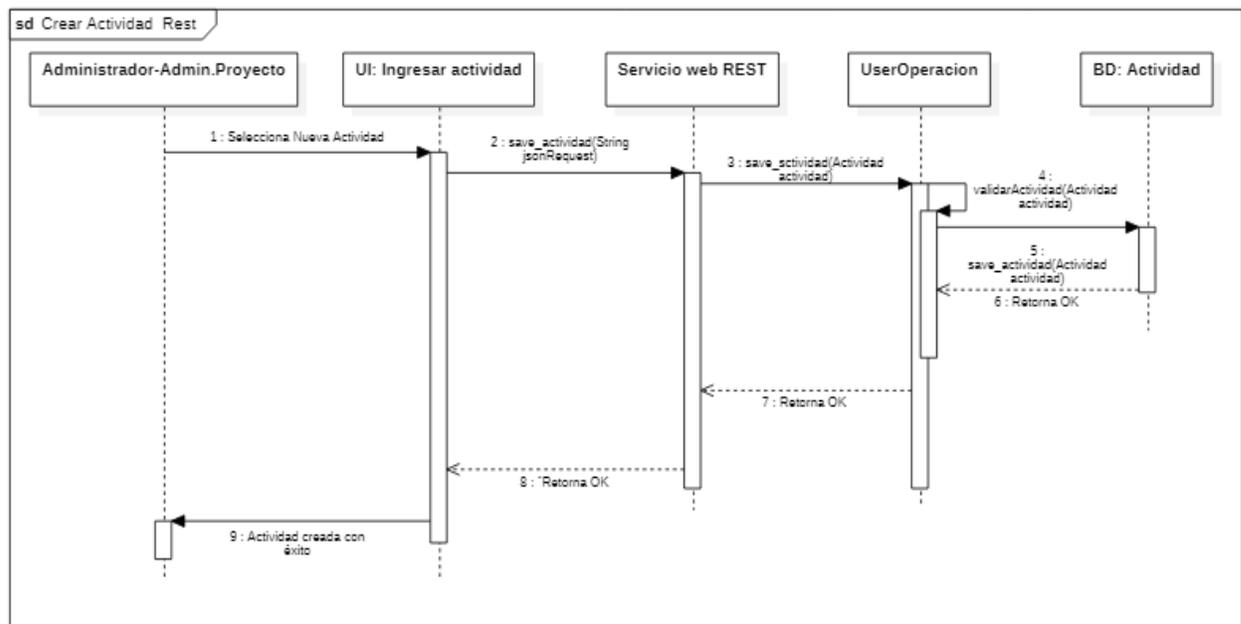


Figura 26

Diagrama de secuencia: Administrar actividades - crear

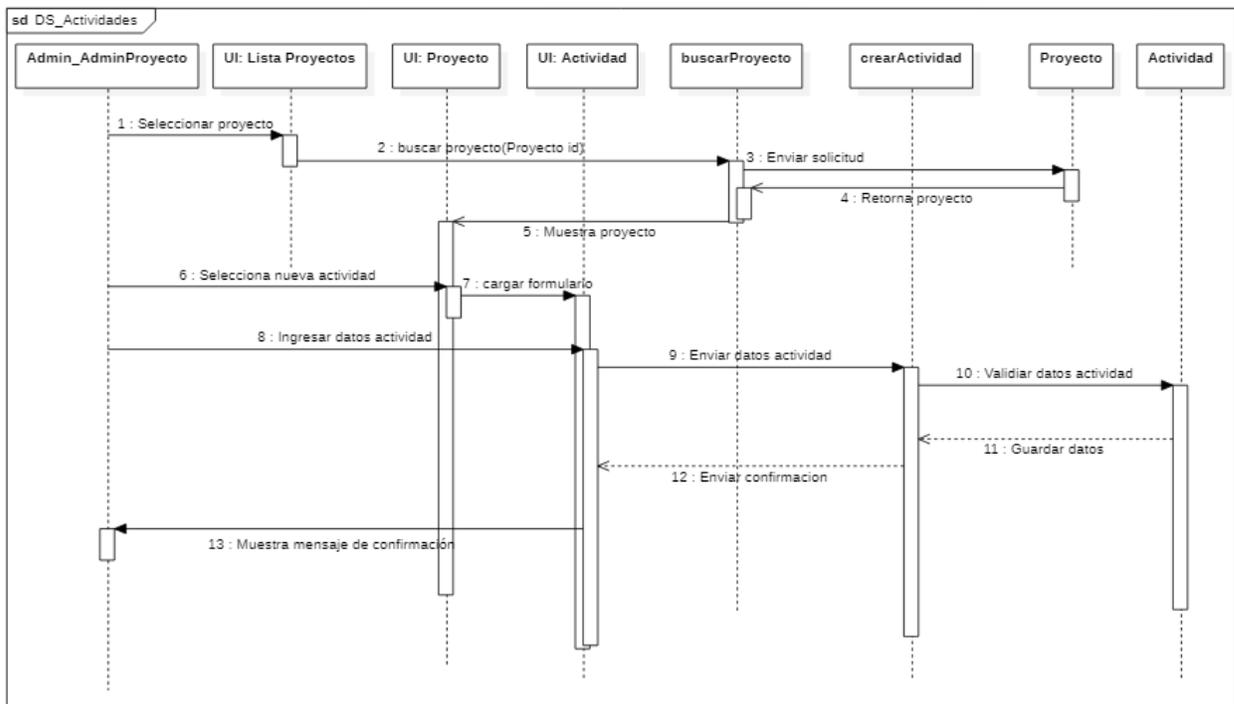


Figura 27

Diagrama de secuencia: Gestionar avances - crear - proceso REST

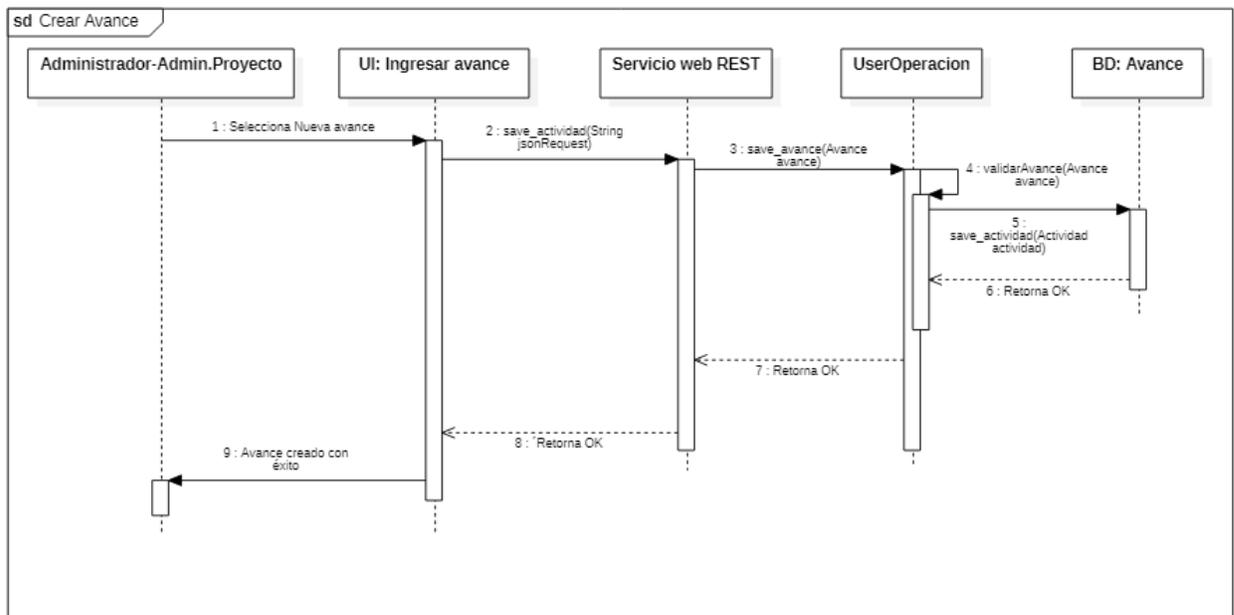


Figura 28

Diagrama de secuencia: Gestionar avances - crear

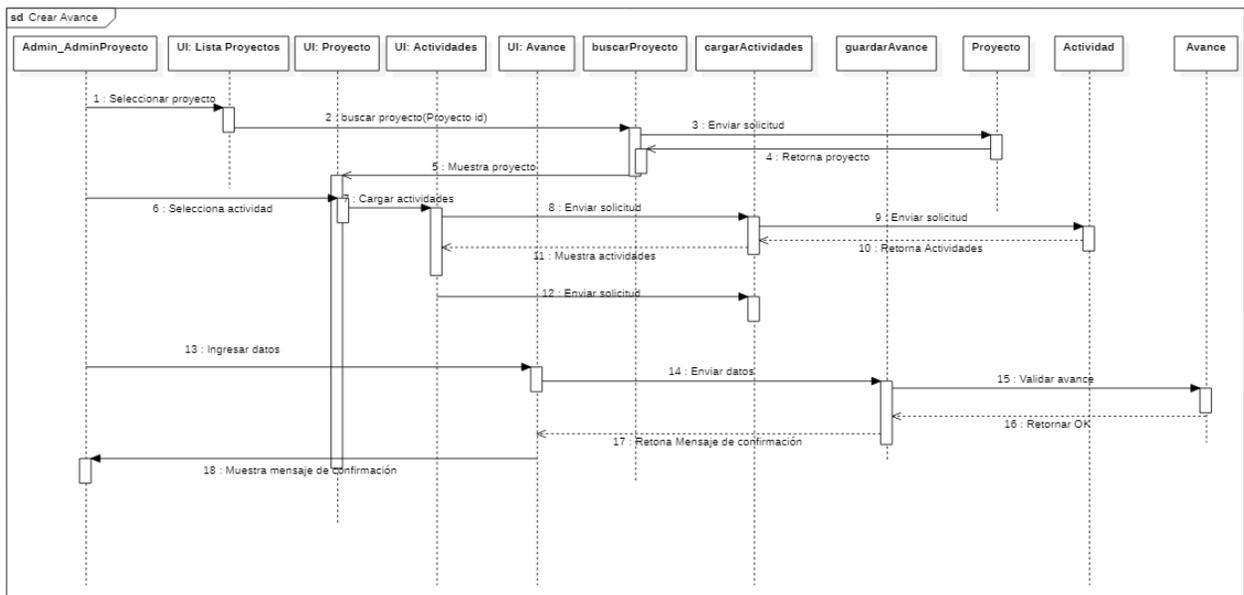
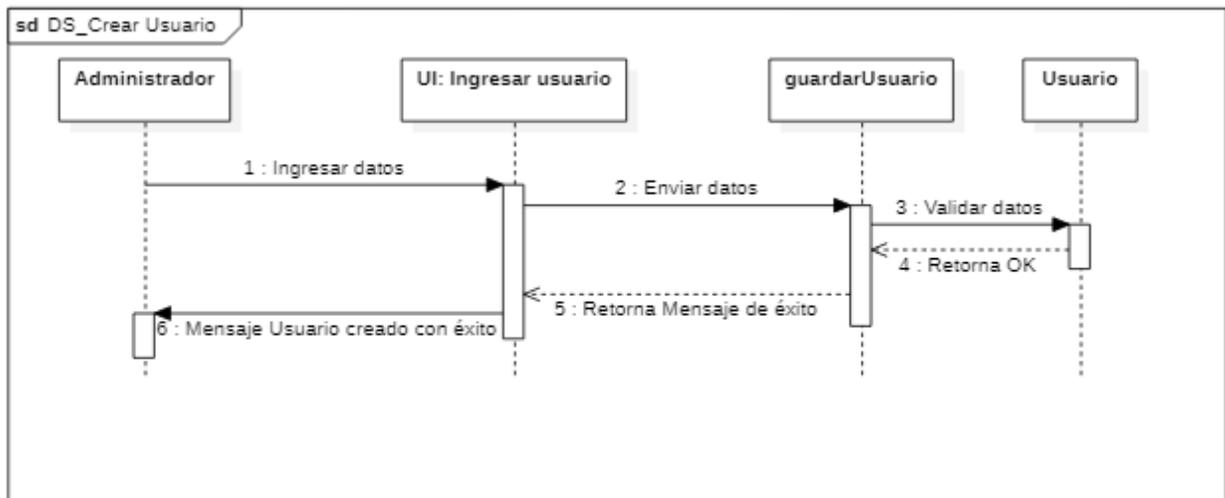


Figura 29

Diagrama de secuencia: Crear usuario

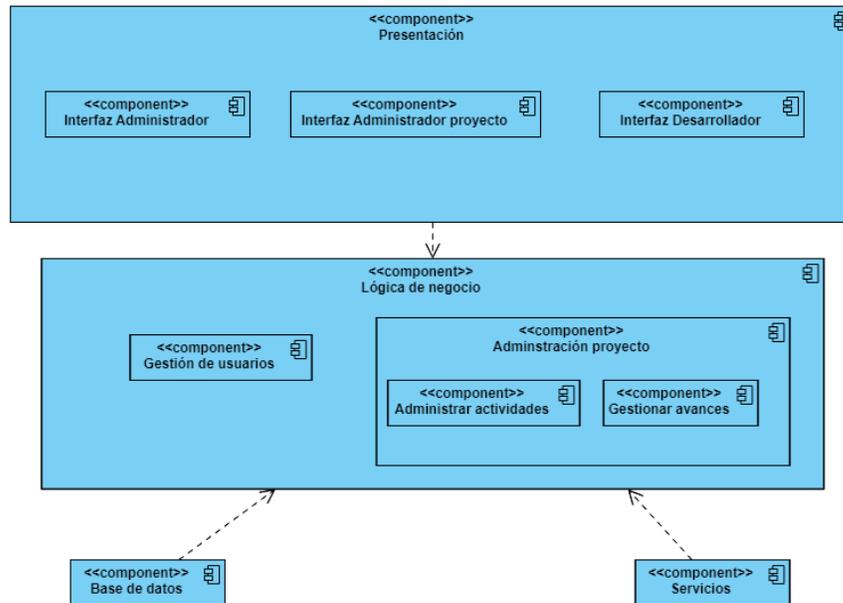


6.4. Implementación

En el siguiente diagrama de componentes se representa la forma estática del sistema mostrando de forma general las vistas de usuario, la lógica de negocio, la base de datos y servicios del sistema.

Figura 30

Diagrama de componentes del sistema.



6.4.1. Escritura de código

Spring boot dispone de un módulo para base de datos denominado Spring Data JPA, el cual permite trabajar con diferentes base de datos a través de la configuración de un archivo denominado application.properties, para que JPA haga la persistencia de los datos, las clases deben ser anotadas con @Entity.

Figura 31
Interface Usuario Repositorio

```
@Entity
@Table(name = "proyectos")
public class Proyecto implements Serializable {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    private String titulo;
    private String serie;
    @Temporal(TemporalType.DATE)
    @DateTimeFormat(pattern = "yyyy-MM-dd")
    private Date fecha_inicio;
    @Temporal(TemporalType.DATE)
    @DateTimeFormat(pattern = "yyyy-MM-dd")
    @NotNull(message = "Fecha de Finalización: Campo Obligatorio")
    private Date fecha_fin;
    private Date fecha_creacion;
    private boolean activado;
    private String observacion;
    private String descripcion;
    @NotEmpty(message = "Tecnologías: Campo Obligatorio")
    private String tecnologias;
    @NotEmpty(message = "Lenguajes de Programación: Campo Obligatorio")
    private String lenguajes_programacion;
}
```

Las interfaces que van a actuar como repositorios, extienden de la interfaz genérica propia de Spring Boot JPA llamada CrudRepository, en donde se especifica la clase con su respectivo id, esta interfaz genérica no solo provee los métodos necesarios para realizar el CRUD, sino que también; permite crear métodos de consulta personalizados a través de queries.

Figura 32
Interface Usuario repositorio

```
public interface IUserRepository extends CrudRepository<Usuario, Long> {

    @Query("FROM Usuario us WHERE us.enabled = true")
    public List<Usuario> findAllActive();
    @Query("FROM Usuario us WHERE us.enabled = false")
    public List<Usuario> findAllInactive();
    @Query("FROM Usuario us")
    public List<Usuario> findProjectManagersAndDevelopers();
    public Usuario findByUsername(String username);
    public Usuario findByCorreo(String correo);
    @Query("FROM Role r")
    public List<Role> findRoles();

}
```

En spring boot las clases que van a actuar como servicios, deben ser anotadas con la anotación `@Service`, de tal manera que en el futuro puedan ser inyectadas en las clases controladoras

Figura 33
Interface Usuario ServiceImplements

```
@Service
public class UsuarioServiceImpl implements IUserarioService {

    @Autowired
    private IUserarioRepository repository;

    @Override
    @Transactional
    public Usuario guardar(Usuario persona) {
        return this.repository.save(persona);
    }

    @Override
    @Transactional(readOnly = true)
    public Usuario buscarPorId(Long id) {
        return this.repository.findById(id).orElse(null);
    }
}
```

En spring boot las clases que van a actuar como controladoras, deben ser anotadas con la anotación `@Controller`, y una anotación adicional denominada `SessionAttributes` en donde se indica el objeto con el que se va a trabajar en los formularios de las vistas, para enviar información a las vistas se utiliza objetos de la clase `Model` o en su defecto se envía a través de colecciones `Map`.

Figura 34

Clase controladora

```
@Controller
@SessionAttributes("usuario")
public class UsuarioController {

    @Autowired
    private UsuarioServiceImpl service;
    @Autowired
    private BCryptPasswordEncoder passwordEncoder;
    @Autowired
    private JavaMailSender javaMailSender;
    @Autowired
    private AWSS3Service awss3Service;

    // *****----- NAVEGACION ----- *****
    @RequestMapping(value = "/usuario/list-users", method = RequestMethod.GET)
    public String ir_listar(Map<String, Object> model, Authentication authentication)
    {
        List<Usuario> usuarios = this.service.listarActivos();
        if (authentication != null) {
            System.out.println("Usuario Autenticado: " + authentication);
        }
        Usuario usRemove = new Usuario();
        for (Usuario us : usuarios) {
            if (us.getCorreo().equals(authentication.getName())) {
                usRemove = us;
            }
        }
        usuarios.remove(usRemove);

        model.put("roles", this.service.listarRoles());
        model.put("titulo", "Usuarios Activos");
        model.put("usuarios", usuarios);
        model.put("bandera", true);
        model.put("usuario", new Usuario());
        return "usuario/list-users";
    }
}
```

Para el almacenamiento de archivos se empleó AWS con su módulo S3, en Spring Boot se tuvo que hacer las configuraciones necesarias a través de un RestController para establecer comunicación con el repositorio de Amazon que permita realizar la carga y descarga de los archivos, AWS de forma gratuita otorga 5GB de almacenamiento.

Figura 35

Carga de archivo a AWS-S3

```
@RestController
@RequestMapping("/s3")
public class UploadFileController {

    @Autowired
    private AWSS3Service awss3Service;

    @PostMapping(value = "/upload")
    public ResponseEntity<String> uploadFile(@RequestParam(value = "file") MultipartFile file) {
        awss3Service.uploadFile(file);
        String response = "El archivo " + file.getOriginalFilename() + " fue cargado correctamente a
S3";
        return new ResponseEntity<String>(response, HttpStatus.OK);
    }
    @GetMapping(value = "/list")
    public ResponseEntity<List<String>> listFiles() {
        return new ResponseEntity<List<String>>(awss3Service.getObjectsFromS3(), HttpStatus.OK);
    }

    @GetMapping(value = "/download")
    public ResponseEntity<Resource> download(@RequestParam("key") String key) {
        InputStreamResource resource = new InputStreamResource(awss3Service.downloadFile(key));
        return ResponseEntity.ok().header(HttpHeaders.CONTENT_DISPOSITION, "attachment; filename=\"" +
key + "\"")
        .body(resource);
    }
}
```

Para el frontEnd se empleó el motor de plantillas paraSpring Boot denominado Thymeleaf, el cual permite procesar la información que proviene desde los controladores a través de su propio lenguaje de expresiones, además; brinda la capacidad de aplicar condicionales y bucles en los documentos html.

Figura 36

Motor de plantillas Thymeleaf

```
<!DOCTYPE html>
<html lang="es" xmlns:th="http://www.thymeleaf.org">
<head th:replace="layout/layout::head">
</head>
<body>
<!-- Loader starts-->
<div class="loader-wrapper">
  <div class="theme-loader"></div>
</div>
<div class="page-wrapper compact-wrapper modern-sidebar" id="pageWrapper">
  <div th:replace="layout/layout::page-main-header"></div>
  <div class="page-body-wrapper modern-sidebar">
    <header th:replace="layout/layout::header"> </header>
    <!-- Page Sidebar Ends-->
    <div class="page-body">
      <!-- Container-fluid starts-->
      <div class="container-fluid">
        <div class="page-header">
          <div class="row">
            <div class="col-lg-12">
              <h3>Usuario</h3>
              <h2 class="text-center mt-4" th:text="${titulo}"></h2>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
</body>
</html>
```

Figura 37

Dashboard del sistema: Pagina de Bienvenida

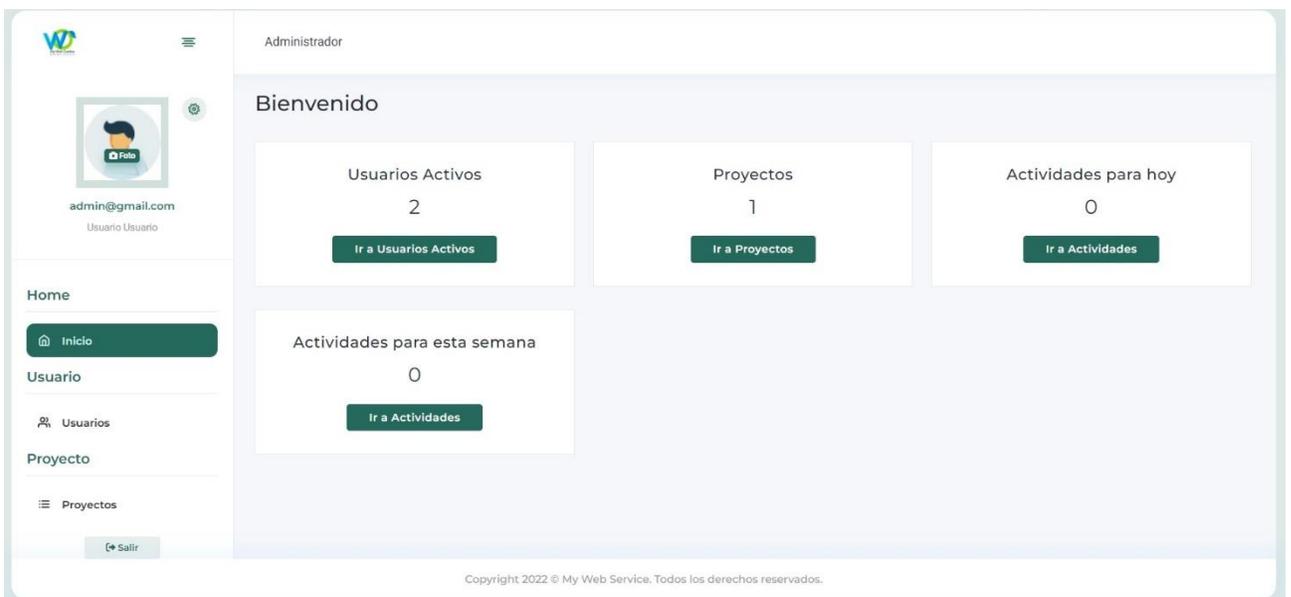


Figura 38

Dashboard del sistema: Pagina de Usuarios

Administrador

Usuario

Usuarios Activos

[Nuevo Usuario](#)

Usuario	Correo	Celular	Telef. casa	Rol	Acciones
Julio Martinez	julio@gmail.com	3333333333	44343234322	Administrador de Proyecto	-
Luis Hurtado	luis@gmail.com	4567897645	2222222222	Desarrollador	-

Copyright 2022 © My Web Service. Todos los derechos reservados.

Figura 39

Dashboard del sistema: Pagina de proyectos

Administrador

Proyecto

Proyectos

[Nuevo Proyecto](#)

Serie	Titulo	Cronograma	Tipo	Administrador	Avance	Acciones
91KT	Proyecto doña petrona	Inicio 2022-11-13 Fin 2022-11-20	Web	Julio Martinez	0.0%	Editar Actividades

Copyright 2022 © My Web Service. Todos los derechos reservados.

Figura 40

Dashboard del sistema: Pagina de nuevo proyecto Step 1

1 - Datos del Proyecto 2 - Administrador y Desarrolladores

1 - DATOS DEL PROYECTO

Título*

Fecha de Inicio*

dd/mm/aaaa

Fecha de Finalización*

dd/mm/aaaa

Tipo*

Web Móvil Escritorio

Lenguajes de Programación*

--- Seleccione Lenguajes ---

Tecnologías*

--- Seleccione Tecnologías ---

Descripción

Siguiete ->

Figura 41

Dashboard del sistema: Pagina de nuevo proyecto Step 2

Administrador

1 - Datos del Proyecto 2 - Administrador y Desarrolladores

2 - ADMINISTRADOR Y DESARROLLADORES

Administrador del Proyecto*

--- Seleccione el Administrador ---

Desarrolladores

+ Agregar

No ha seleccionado un Desarrollador

Regresar Finalizar

Figura 42

Dashboard del sistema: Pagina de actividades

The screenshot shows the 'Proyectos - Actividades' interface for 'Proyecto Doña Petrona'. At the top, there is a header 'Proyectos - Actividades' and a sub-header 'Proyecto Doña Petrona'. Below this is a brown bar labeled 'Actividades'. A yellow button '+ Nueva Actividad' is positioned on the left. The main content area contains two activity entries: 'Actividad 1: Entrevista' and 'Actividad 2: Diagramas de casos de Uso', each with a blue expandable arrow on the right.

Figura 43

Dashboard del sistema: Pagina de avances

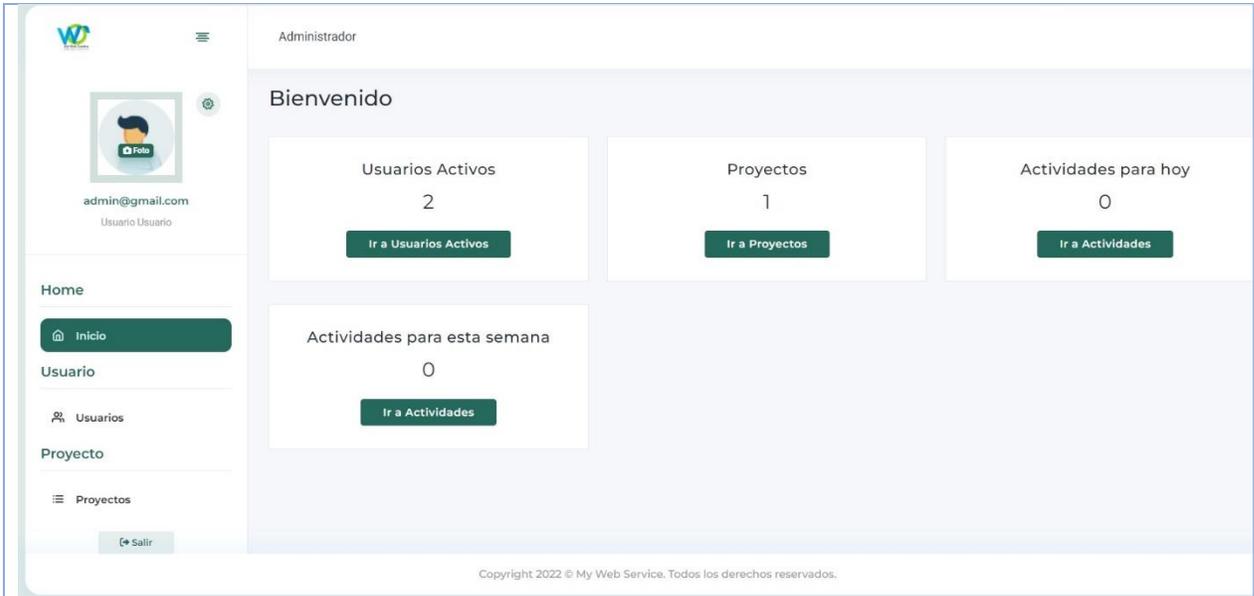
The screenshot displays the 'Avances' section for 'Actividad 1: Entrevista'. It features a brown header 'Actividades' and a yellow '+ Nueva Actividad' button. The activity details are shown in a light blue box with a collapse arrow. The details include: 'Cronograma Actividad' with start date '2022-11-19' and end date '2022-11-24'; 'Responsable' as 'Julio Martinez'; 'Descripción' as 'asdas asda asdsa asd'; and 'Estado' as 'En Ejecución'. There are two action buttons: a green one with a share icon and a red one with a stop icon. A toggle switch for 'Finalizar Actividad' is set to 'Marcar como Actividad Completada'. Below the details is a section titled 'Avances' with a '+ Nuevo Avance' button. A light blue message box at the bottom states 'La actividad no tiene avances registrados'.

6.4.2. Pruebas del sistema y de aceptación basadas en casos de uso

Tabla 10

Caso prueba – Inicio de sesión

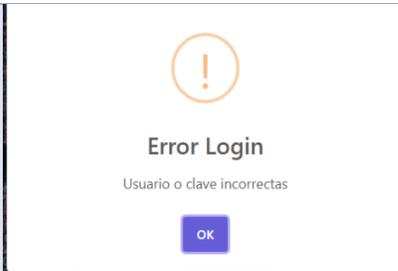
Caso de prueba: RF01 Inicio de sesión		
Procedimiento de Prueba		
Escenario 1: Inicio de sesión		
Pasos	Resultado esperado del sistema	Cumple
1.- Ingresar al sistema con la url 2.- Ingrese el correo y contraseña 3.- Clic en el botón Ingresar 4.- Se valida la información con los datos registrados en la base de datos 5.-Presenta un mensaje Inicio de sesión 6.-Se muestra la ventana principal acorde a su rol de usuario.	Ingresar el usuario del sistema	Si
Resultado Obtenido:		
		



Escenario 2: Datos incorrectos

Pasos	Resultado esperado del sistema	Cumple
1.- Ingresar al sistema con la url 2.- Ingrese el correo y contraseña 3.- Clic en el botón Ingresar 4.- Se valida la información con los datos registrados en la base de datos 5.-Presenta un mensaje Error de login 6.-Se selecciona la opción OK 7. Se regresa a la ventana de inicio de sesión	Validación de datos	Si

Resultado Obtenido:



Escenario 3: Recuperar contraseña

Pasos	Resultado esperado del sistema	Cumple
<ol style="list-style-type: none"> 1.- Ingresar al sistema con la url 2.- Ingrese el correo y contraseña 3.- Clic en el botón Ingresar 4.- Se valida la información con los datos registrados en la base de datos 5.-Presenta un mensaje Error de login 6.-Se selecciona la opción OK 7. Se regresa a la ventana de inicio de sesión 8. Seleccione la opción Recuperar contraseña 9. Ingrese correo electrónico y presione el botón Enviar correo 10. Revise su correo electrónico y anote el código enviado 11. Ingrese el código enviado a su correo 12. Presione el botón Aceptar 13. Ingrese nueva la contraseña y verifique la contraseña 14. Presione el botón aceptar 	<p style="text-align: center;">Recuperar contraseña</p>	<p style="text-align: center;">Si</p>

15. Se muestra un mensaje de notificación indicando que se ha cambiado la contraseña.

Resultado Obtenido:



A screenshot of a web application interface showing a modal dialog box titled "Restablecer Contraseña" (Reset Password). The dialog contains two text input fields. The first is labeled "Contraseña:" and contains the text "admin1234". The second is labeled "Repetir Contraseña:" and also contains "admin1234". Below the second input field, there is a green confirmation message: "Las contraseñas coinciden" (The passwords match). At the bottom right of the dialog are two buttons: "Cerrar" (Close) and "Aceptar" (Accept).

Tabla 11
 Caso prueba – Gestión de usuarios

Caso de prueba: RF02 Gestionar Usuarios		
Procedimiento de Prueba		
Escenario 1: Nuevo usuario		
Pasos	Resultado esperado del sistema	Cumple
1.- Ingresar al sistema con el rol administrador 2.- Seleccionar del menú: Usuarios 3.- Clic en la opción Nuevo usuario 4.- Llenar el formulario y seleccione el rol 5.- Clic en el botón Crear usuario 6.- Se valida la información y se almacena en la base de datos 6.-Presenta un mensaje de Usuario Guardado con éxito. 7.-Se limpian las cajas de texto del formulario y se queda en la ventana actual. 8. Revise su correo electrónico para conocer su cuenta de usuario: correo y contraseña.	Ingresar el usuario del sistema	Si
Resultado Obtenido:		

Usuario

Nuevo Usuario

DATOS DE USUARIO

Usuario*

Correo*

Celular*

Teléfono de casa

Rol*

Crear Usuario

Nuevo Usuario

Procesando!!



Por favor espere



Registro

Usuario **Sara Petronila** Registrado con éxito

OK

Escenario 2: Editar usuario

Pasos	Resultado esperado del sistema	Cumple
1.- Ingresar a la página de gestión mediante su cuenta de usuario. 2.- En su perfil encontrara el botón de editar.	Modificar la información del usuario	Si

<p>3.- Cambiar datos</p> <p>4.- Clic en el botón Actualizar</p> <p>5.- Se valida la información y se almacena en la base de datos</p> <p>6.-Presenta un mensaje de Datos actualizados con éxito.</p> <p>7.-Regresa al entorno personal</p>		
--	--	--

Resultado Obtenido:

Usuario

Nuevo Usuario

DATOS DE USUARIO

Usuario*

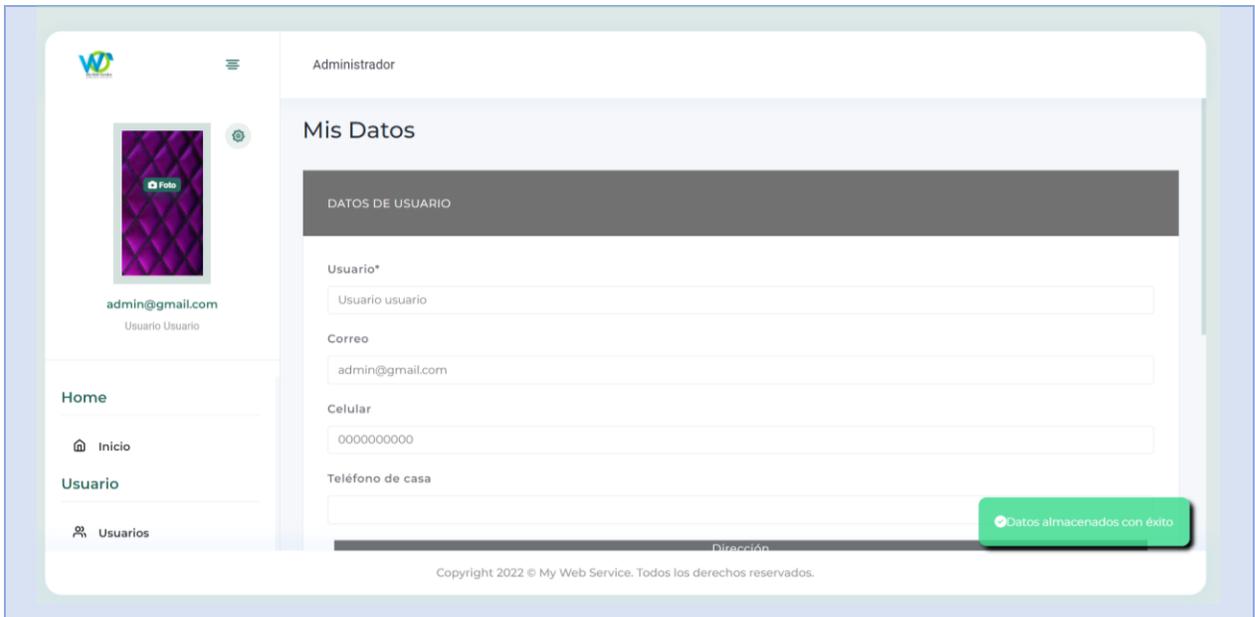
Correo*

Celular*

Teléfono de casa

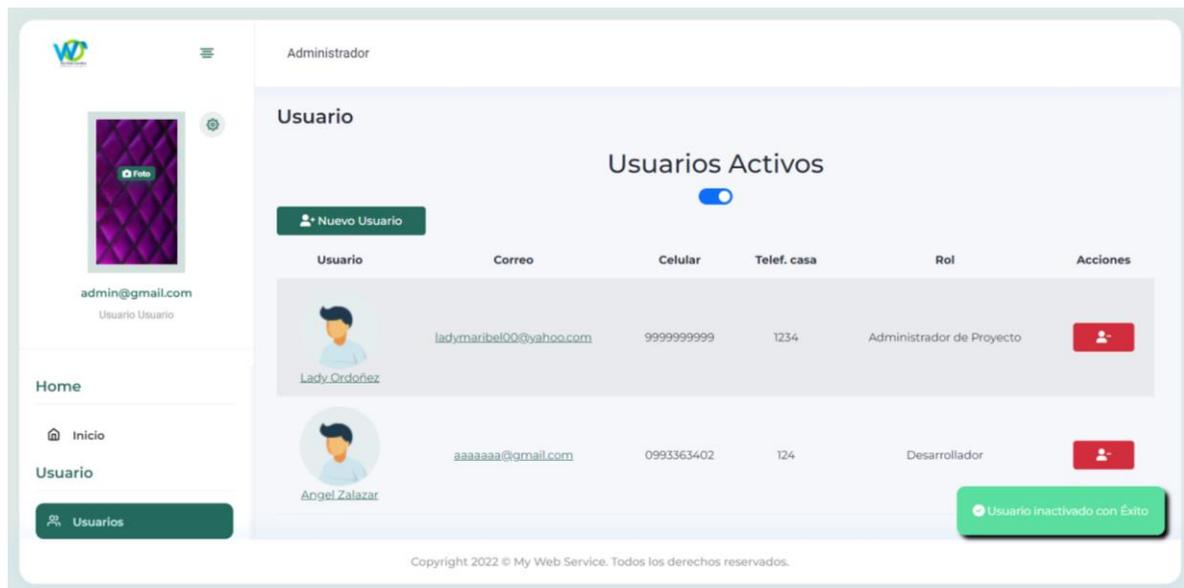
Rol*

Crear Usuario



Escenario 3: Dar de baja usuario

Pasos	Resultado esperado del sistema	Cumple
<p>1.- Ingresar al sistema mediante su cuenta de usuario.</p> <p>2.- Seleccionar del menú: Usuarios</p> <p>3.- Clic en el botón Inactivar de un registro en la columna Acciones de la tabla.</p> <p>4.- Del modal de confirmación dar clic en el botón Si</p> <p>5.- Se da de baja el usuario y se guardan los cambios en la base de</p> <p>6.-Presenta un mensaje de Usuario nombreUsuario inactivado con éxito.</p> <p>7.-Regresa a la lista de los usuarios activos.</p>	<p>Dar de baja un usuario</p>	<p>Si</p>
<p>Resultado Obtenido:</p>		



Escenario 4: Dar de alta usuario

Pasos	Resultado esperado del sistema	Cumple
1.- Ingresar al sistema con su cuenta de usuario. 2.- Seleccionar del menú: Usuarios 3.- Clic en la opción switch-checkbox para cambiar de usuarios activos a usuarios Inactivos 4.- Clic en el botón Activar de un registro en la columna Acciones de la tabla Inactivos.	Dar de alta un usuario	Si

5.- Del modal de confirmación dar clic en el botón Si

6.- Se da de alta el usuario y se guardan los cambios en la base de datos.

7.-Presenta un mensaje de Usuario activado con éxito.

8.-Regresa a la lista de usuarios inactivos

Resultado Obtenido:

Administrador

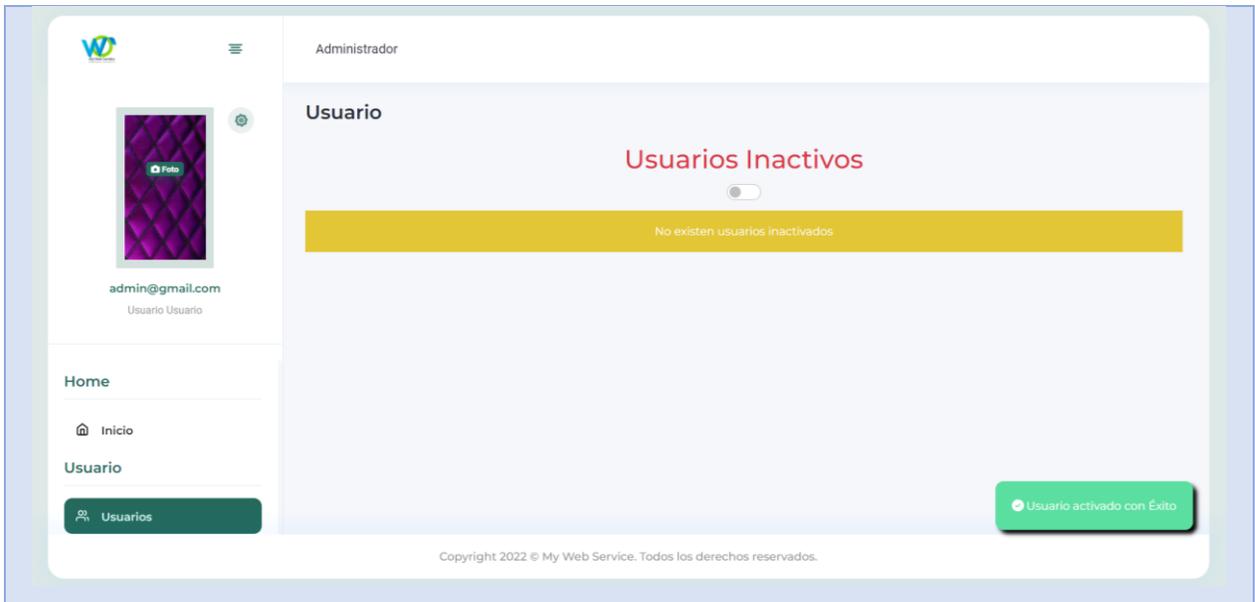
Usuario

Usuarios Inactivos

Apellidos y Nombres	Correo	Celular	Acciones
Luis Hurtado	luis@gmail.com	4567897645	

Activación de Usuario

¿Está seguro de **ACTIVAR** nuevamente al usuario **Maribel Ordoñez**?



Escenario 4: Datos obligatorios no ingresados

Pasos	Resultado esperado del sistema	Cumple
<ol style="list-style-type: none"> 1.- Ingresar al sistema con su cuenta de usuario 2.- Seleccionar del menú: Usuarios 3.- Clic en la opción Listado 4.- Clic en el botón Editar de un registro en la columna Acciones de la tabla. 5.- Cambiar datos 6.- Clic en el botón Actualizar 7.- <i>Se valida la información (mensajes de campos obligatorios)</i> 8.- <i>Llenar campos obligatorios</i> 9.- Regresa al punto 6 10.- Se almacena en la base de datos 11.-Presenta un mensaje de Datos actualizados con éxito. 	Campos requeridos	Si

12.-Regresa a la lista de usuarios.

Resultado Obtenido:

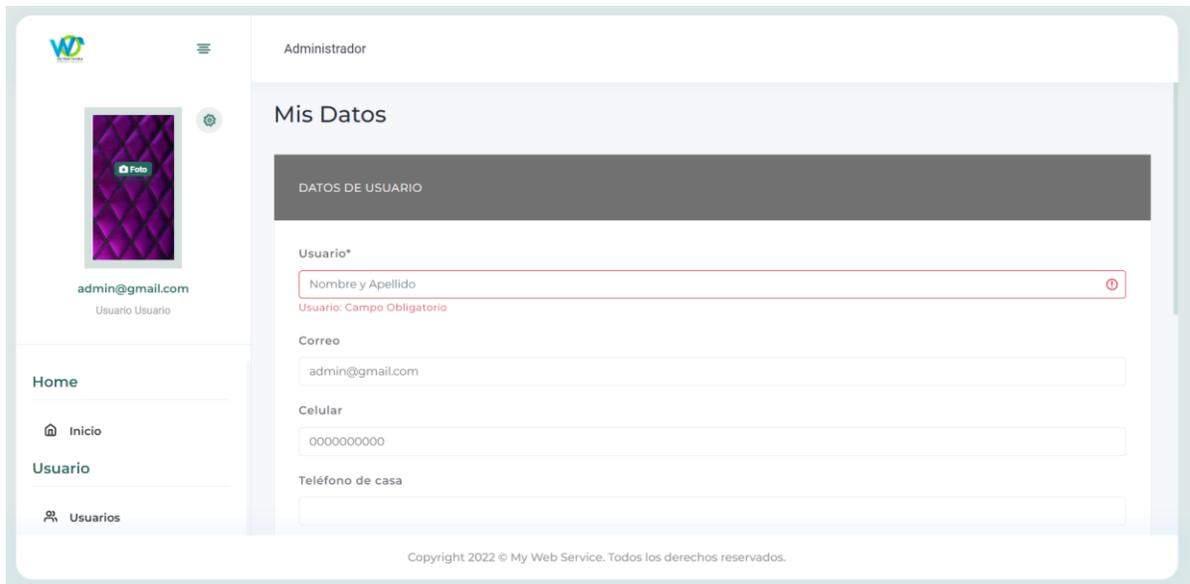


Tabla 12
Caso prueba – Administrar proyecto

Caso de prueba: RF04 Administrar proyecto (RF 05,06, 08,09,10)

Procedimiento de Prueba

Escenario 1: Nuevo proyecto (RF-04-, RF-05)

Pasos	Resultado esperado del sistema	Cumple
1.- Ingresar al sistema con los datos de cuenta registrados. 2.- Seleccionar del menú: Proyectos 3.- Clic en la opción Nuevo proyecto 4.- Llenar el formulario: 4.1.- Step 1: Titulo, fecha de inicio y fecha de fin, tipo, lenguaje de	Proyecto creado	Si

programación, tecnologías y descripción. Selecciona siguiente

4.2.- Step 2: selecciona administrador de proyecto y desarrolladores.

5. Clic en el botón Finalizar

6.- Se valida la información y se almacena en la base de datos

7.-Presenta un mensaje de Unidad de material registrado con éxito.

8.-Regresa a la pantalla de los proyectos.

Resultado Obtenido:

The screenshot shows a web application interface for project registration. At the top, there is a navigation bar with two tabs: "1 - Datos del Proyecto" (active) and "2 - Administrador y Desarrolladores". Below the navigation bar, the form is titled "1 - DATOS DEL PROYECTO". The form contains the following fields and controls:

- Título***: A text input field.
- Fecha de Inicio***: A date input field with a placeholder "dd/mm/aaaa" and a calendar icon.
- Fecha de Finalización***: A date input field with a placeholder "dd/mm/aaaa" and a calendar icon.
- Tipo***: Radio buttons for "Web", "Móvil", and "Escritorio".
- Lenguajes de Programación***: A dropdown menu with the placeholder "--- Seleccione Lenguajes ---" and a search icon.
- Tecnologías***: A dropdown menu with the placeholder "--- Seleccione Tecnologías ---" and a search icon.
- Descripción**: A text area for entering project details.

At the bottom right of the form, there is a "Siguinte →" button.

Administrador

Crear Proyecto

1 - Datos del Proyecto 2 - Administrador y Desarrolladores

2 - ADMINISTRADOR Y DESARROLLADORES

Administrador del Proyecto*

Lady Ordoñez

Desarrolladores

Nombres y Apellidos	Eliminar
Lady Ordoñez	

+ Agregar

← Regresar Finalizar

Proyecto

Proyectos

Nuevo Proyecto

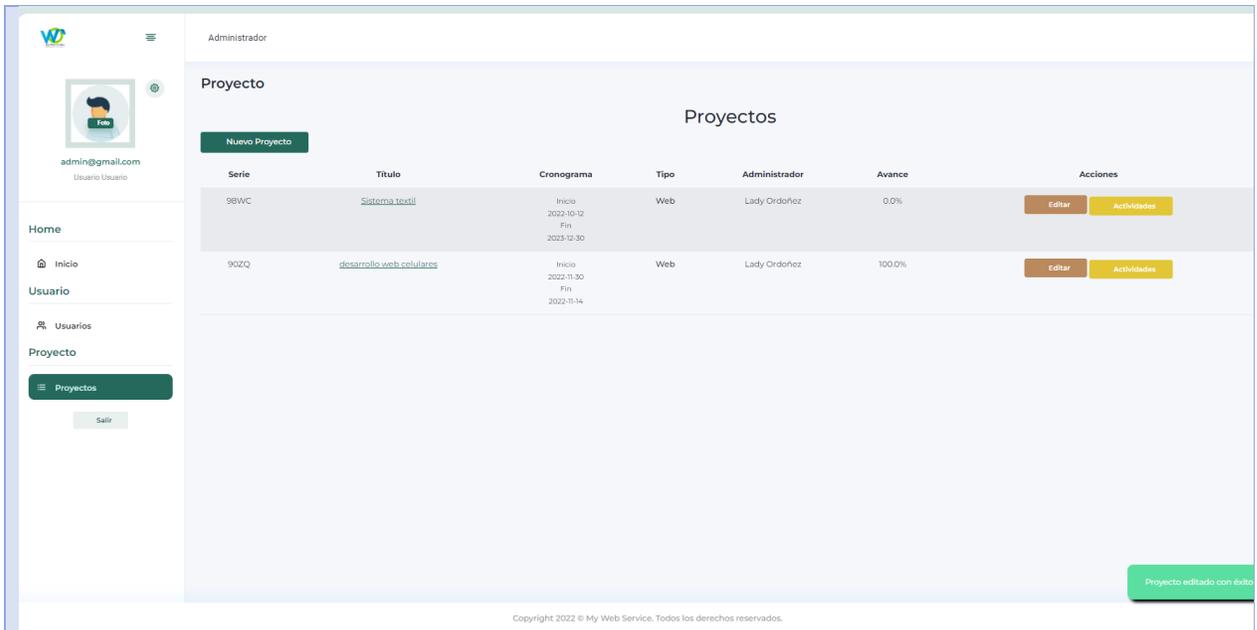
Serie	Título	Cronograma	Tipo	Administrador	Avance	Acciones
90ZQ	desarrollo web celulares	Inicio 2022-11-30 Fin 2022-11-14	Web	Lady Ordoñez	100.0%	
98WC	Sistema textil	Inicio 2022-10-12 Fin 2023-12-30	Web	Lady Ordoñez	0.0%	

Proyecto creado con éxito

Copyright 2022 © My Web Service. Todos los derechos reservados.

Escenario 2: Editar proyecto (RF-04, RF-05)

Pasos	Resultado esperado del sistema	Cumple
1.- Ingresar a la página de gestión mediante su cuenta de usuario. 2.- Seleccionar del menú: Proyectos 3.- Clic en el icono Editar de un registro en la columna Acciones de la tabla Listado de proyectos 4.- Cambiar datos 5.- Clic en el botón editar proyecto	Editar proyecto	Si



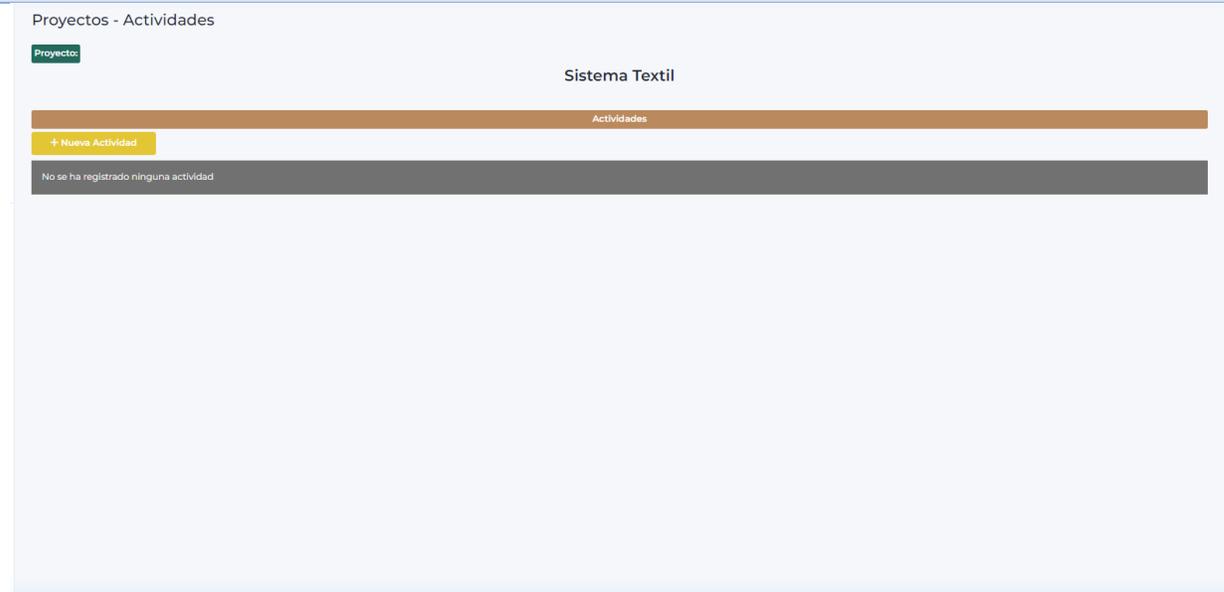
Escenario 3: Administrar actividades (RF-07, RF-09) (crear)

Pasos	Resultado esperado del sistema	Cumple
<p>1.- Ingresar al sistema con los datos de cuenta registrados.</p> <p>2.- Seleccionar del menú: Proyectos</p> <p>3. Seleccionar el icono Actividades de la columna de Acciones de la tabla de proyectos.</p> <p>5. El sistema muestra la ventana con espacio para crear las actividades del proyecto.</p> <p>6. Seleccione el botón Nueva actividad.</p>	<p>Actividad creada con éxito.</p>	<p>Si</p>

7. Se muestra un formulario con los datos requeridos para la actividad del proyecto.

8. El sistema muestra un mensaje de actividad creada con éxito.

Resultado Obtenido:



The screenshot shows a modal form titled 'Actividad' with a close button (X). The form contains the following fields:

- Nombre***: Input field with the value 'Casos de uso'.
- Inicio***: Date picker field with the value '09/11/2022'.
- Fin***: Date picker field with the value '10/12/2022'.
- Responsable***: Dropdown menu with the selected value 'Lady Ordoñez'.
- Descripción**: Text area with the value 'De cada requerimiento'.

At the bottom of the form are two buttons: 'Cancelar' (orange) and 'Guardar' (green).

Proyectos - Actividades

Proyecto: **Sistema Textil**

Actividades

+ Nueva Actividad

Actividad 1: Casos de uso

Actividad creada con éxito

Copyright 2022 © My Web Service. Todos los derechos reservados.

Escenario 4: Gestionar avance - crear

Pasos	Resultado esperado del sistema	Cumple
1.- Ingresar al sistema con los datos de cuenta registrados. 2.- Seleccionar del menú: Proyectos 3. Seleccionar el icono Actividades de la columna de Acciones de la tabla de proyectos. 4. Seleccionar la actividad en la que se va a crear el avance. 5. Presiona el botón Nuevo avance.	Avance creado	Si

6. Ingresa los datos requeridos para el avance y presiona Guardar nuevo avance

7. EL sistema muestra un mensaje avance creado con éxito y regresa a la pantalla de actividades de ese proyecto.

Resultado Obtenido:

The screenshot displays a web interface for project management. At the top, it shows 'Proyectos - Actividades' with a dropdown menu for 'Proyecto:' set to 'Sistema Textil'. Below this is a section for 'Actividades' with a '+ Nueva Actividad' button. A specific activity is selected: 'Actividad 1: Casos de uso'. The activity details include a 'Cronograma Actividad' with start and end dates, a 'Responsable' (Lady Ordoñez), a 'Descripción' (De cada requerimiento), and an 'Estado' (En Ejecución). There is a toggle for 'Finalizar Actividad' set to 'Marcar como Actividad Completada'. Below the activity details is a section for 'Avances' with a '+ Nuevo Avance' button. A message at the bottom states 'La actividad no tiene avances registrados'.

Administrador

FORMULARIO PARA AVANCE DE ACTIVIDAD

Información de la Actividad

Actividad: Casos de uso

Cronograma: Inicio: 2022-11-09, Fin: 2022-12-10

Descripción: De cada requerimiento

Proyecto: [Ver información del proyecto](#)

FORMULARIO PARA AVANCE DE ACTIVIDAD

Título del avance*

Evidencia de avance
 No se eligió ningún archivo

Descripción

[Guardar nuevo avance](#)

Home

Inicio

Usuario

Usuarios

Proyecto

Proyectos

[Salir](#)

Proyectos - Actividades

Proyecto:

Sistema Textil

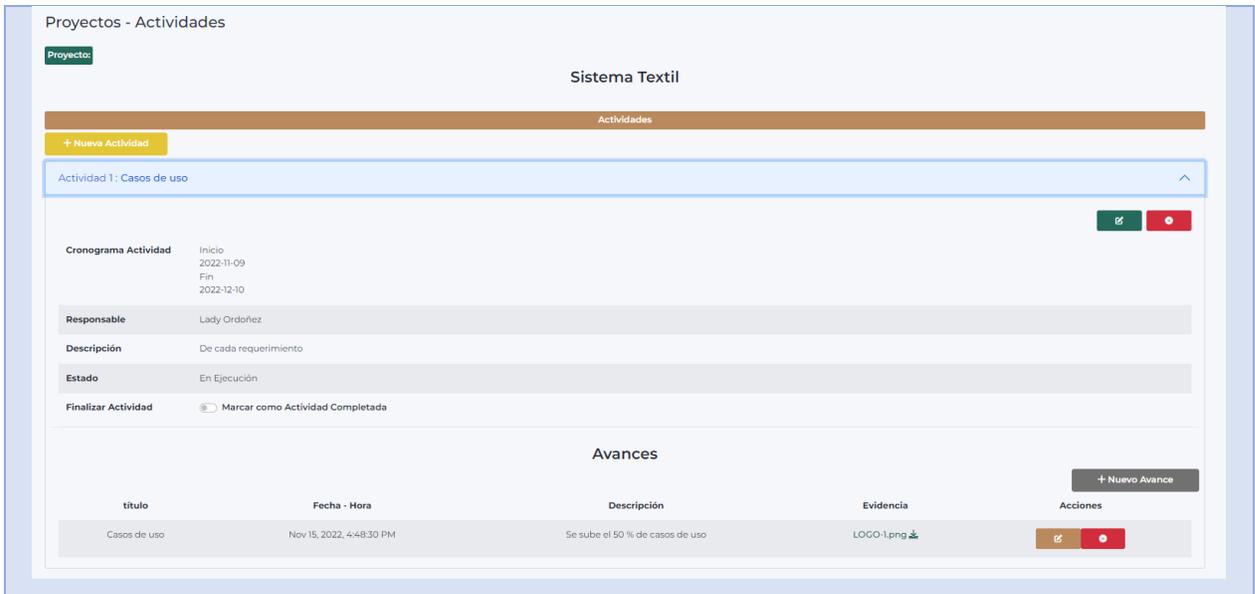
Actividades

[+ Nueva Actividad](#)

Actividad 1: Casos de uso

[Avance guardado con éxito!](#)

Copyright 2022 © My Web Service. Todos los derechos reservados.



6.4.3. Evaluación del sistema

Para aplicar el proceso de evaluación del sistema web se ha puesto en marcha el uso del sistema dentro de la empresa, el desarrollo consiste en que los diferentes usuarios manipulen el sistema, dependiendo del rol que tenga el usuario se mostrara los módulos con las actividades, tales como: llenar formularios, visualizar datos, modificar o actualizar datos, cargar imágenes, entre otros.

Los sujetos experimentales fueron la propietaria de la empresa, los empleados y algunos clientes. Para medir eficientemente la usabilidad del sistema se ha empleado un test de usabilidad, en donde cada usuario debe ir contestando un conjunto de preguntas que tienen opciones de respuesta determinado por la escala de likert. El objetivo de aplicar este tipo de evaluación, es determinar el grado de satisfacción del usuario al operar el software y obtener conclusiones para poder mejorar los servicios y usabilidad del sistema. Para la ejecución de esta prueba se ha seleccionado el estándar SUMI (The Software Usability Measurement Inventory).

Las preguntas que conformaron el test, se las puede describe en la siguiente tabla:

Tabla 13
Formato de evaluación test SUMI

PREGUNTA	DE ACUERDO	INDECISO	DESACUERDO
El sistema responde lentamente a los datos de entrada			
Las advertencias emitidas son útiles			
El software se detiene o cuelga inesperadamente			
Es muy difícil manipular el software			
La interfaz es amigable			
La información se presenta bien estructurada y ordenada			
Disfruto del tiempo que se dedicó a usar este software			
Toma demasiado tiempo aprender las funciones del software			

PREGUNTA	DE ACUERDO	INDECISO	DESACUERDO
Trabajar con este software es muy satisfactorio			
No hay suficiente información en la pantalla cuando se necesita			
Le gustaría usar este software todos los días			
Usar este software es frustrante			
La velocidad de este software es lo suficientemente rápido			
La distribución de los menús parece bastante lógica			
Es fácil moverse de un proceso a otro			
El software muestra mensajes informativos después de ejecutar una actividad			
El software es intuitivo			
Los colores y tamaños de las fuentes son atractivos			
Es fácil hacer que el software haga exactamente lo que quiero			

Es fácil ver a simple vista las opciones que ofrece el software

El resumen de los resultados obtenidos al aplicar esta evaluación, se los puede evidenciar en el Anexo 4.

6.4.4. Pruebas de carga JMeter

Para las pruebas de carga se utilizó la herramienta de JMeter, se creó un total de 200 hilos, y como resultado nos arrojó la figura 27, en la que muestra que:

Numero de muestras: 200

Desviación: 43

Rendimiento: 869,691/minuto

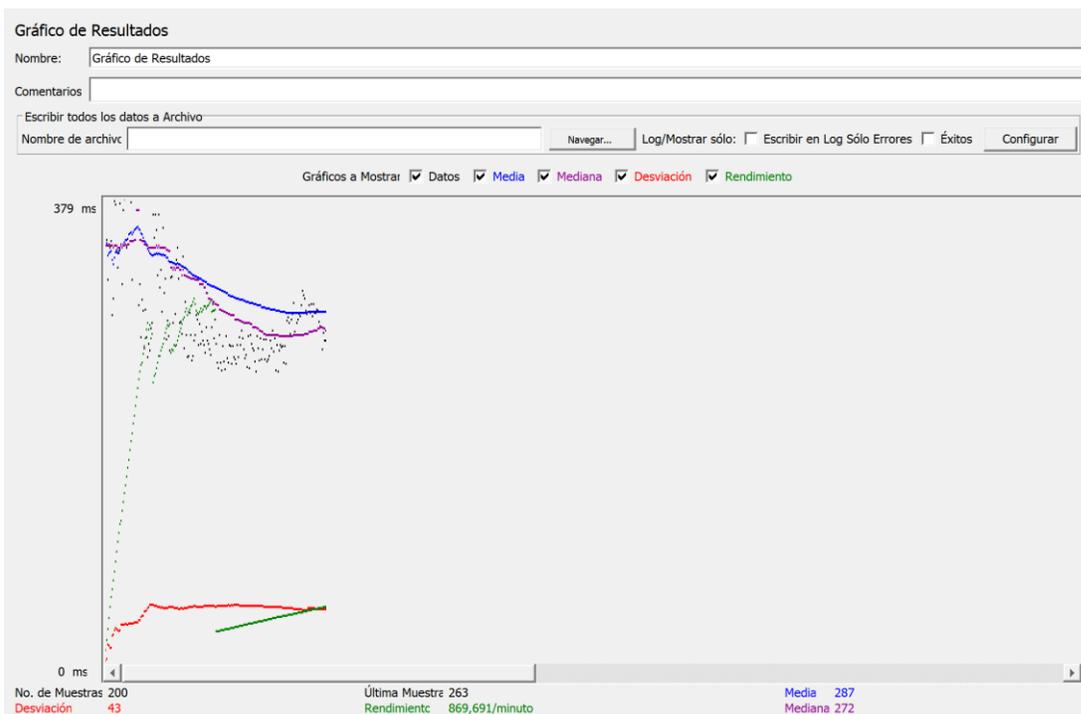
Media: 287

Mediana: 272

Como conclusión de estos datos se dice que en rendimiento es mucho mayor a la desviación por lo tanto el sistema si puede manejar varias solicitudes.

Figura 44

Resultados de pruebas de carga JMeter



7. CONCLUSIONES

Luego de haber finalizado el proyecto se presenta las siguientes conclusiones:

- La utilización de la metodología Iconix para el desarrollo del proyecto, permitió definir y determinar correctamente las necesidades de los usuarios, dado que la interacción continua con los usuarios produjo que las fases se desarrollen de forma interactiva logrando establecer alcances puntuales y funcionales que permiten contar con un enfoque disciplinario de las tareas y responsabilidades dentro del equipo de trabajo.
- Con la ayuda de las diferentes técnicas de recolección aplicadas a los actores se logró obtener los requerimientos del sistema para su construcción y lograr cumplir con las necesidades de los usuarios, dentro de las cuales se tiene: gestión de usuarios, gestión de proyectos, gestión de actividades, gestión de avances.
- Con la ayuda de herramientas de desarrollo como el Framework Spring Boot se trabajó el backend y para el frontend el motor de plantillas Thymeleaf, estas herramientas permitieron codificar un software eficiente y eficaz para la empresa, además de proveer un sistema bastante amigable con el usuario.
- Con las pruebas de funcionalidad y carga realizadas al sistema se pudo corroborar el correcto desempeño de los diferentes requisitos del sistema, así como el tiempo de respuesta de cada uno de estos a peticiones del usuario.

8. RECOMENDACIONES

Una vez culminado el proyecto se presenta las siguientes recomendaciones:

- Quienes están inmersos en el desarrollo de sistemas de software, se recomienda definir las actividades o tareas dentro de cada fase, esto permitirá que el proyecto sea construido correctamente y evitar que durante el proceso se produzcan fuertes cambios que afecten a la funcionalidad y por tal motivo aumente los costos de producción.
- En el proceso de desarrollo de software es de vital importancia integrar patrones orientados a pruebas (TDD), que permitan garantizar la correcta funcionalidad de cada uno de los métodos implementados, y de esta manera asegurar la correcta funcionalidad del software.
- Se recomienda tomar como guía este trabajo para investigaciones relacionadas, con administración de proyectos TI y gestión de proyectos de software, sugiriendo que se aproveche el Api Rest construido, para aplicaciones web progresivas o aplicaciones móviles que complementen la funcionalidad de este sistema informático.

9. BIBLIOGRAFÍA

- Arquitectura de una API REST · Desarrollo de aplicaciones web.* (s.f.). Obtenido de Juanda.gitbooks.io: <https://juanda.gitbooks.io/webapps/content/api/arquitectura-api-rest.html>
- Code, V. (s.f.). *Visual Studio Code - Code Editing.* Obtenido de <https://code.visualstudio.com/>
- ERP, E. (2021, 07 19). *Evaluando ERP.* Obtenido de Sistema de gestión ¿Qué es? ¿Cuántos tipos hay?: <https://www.evaluandoerp.com/software-erp/sistema-de-gestion/#:%7E:text=Un%20sistema%20de%20gesti%C3%B3n%20es,laboral%20m%C3%A1s%20eficiente%20y%20sostenible.>
- González Rodríguez, S. S., Viteri Intriago, D. A., Izquierdo Morán, A. M., & Verdezoto Cordova, G. O. (2021). Modelo de gestión administrativa para el desarrollo empresarial del Hotel Barros en la ciudad de Quevedo. En S. S. González Rodríguez, D. A. Viteri Intriago, A. M. Izquierdo Morán, & G. O. Verdezoto Cordova. *Revista Universidad y Sociedad*,.
- Java.com.* (s.f.). Obtenido de https://www.java.com/es/download/help/whatis_java.html
- Kruchten, P. (2006). *Planos Arquitectónicos: El Modelo de “4+1” Vistas de la Arquitectura del Software.* Obtenido de IEEE Software : http://cic.puj.edu.co/wiki/lib/exe/fetch.php?media=materias:modelo4_1.pdf
- Muller. (2020, Agosto 19). *Gestiopolis.* Obtenido de <https://www.gestiopolis.com/que-es-inventario-tipos-utilidad-contabilizacion-y-valuacion/>
- Odoo S.A. (2021). Obtenido de Odoo Fabricación - Industria 4.0 Cloud MRP: https://www.odoo.com/es_ES/app/manufacturing
- Peralta, J. (s.f.). *Diferencia entre Software a la Medida y Software Estándar | GBS Global BestTech Systems.* Obtenido de Gbts.com.mx.: <http://www.gbts.com.mx/diferencia-entre-software-a-la-medida-y-software-estandar/>

PostgreSQL. (s.f.). Obtenido de <https://www.postgresql.org/>

R. (2021, Febrero 7). *Concepto de - Definición de*. Obtenido de Gestión:
<https://conceptodefinicion.de/gestion/>

Spring.io. (s.f.). Obtenido de Spring Boot: <https://spring.io/projects/spring-boot>

SQL. (2012). Obtenido de <http://sql.11sql.com/sql-intro.htm>

StartUML - EcuRed . (s.f.). Obtenido de Ecured.cu: <https://www.ecured.cu/StartUML>

StarUML. (s.f.). Obtenido de <https://staruml.io/>

Y., A. (2021, Junio 29). *Administración*. Obtenido de
<https://conceptodefinicion.de/administracion/>

10.ANEXOS

10.1. Anexo 1: Certificación de traducción

Zumbi, 26 de noviembre del 2022

Jessica Astrid Quichimbo Lalangui
Licenciada en Ciencias de la Educación Mención Idioma Inglés

CERTIFICO:

Que se ha realizado la traducción de español a inglés del resumen derivado de la tesis de titulada **“Desarrollo e Implementación de un sistema web para el seguimiento y control de los proyectos informáticos de la empresa My Web Service”** De autoría de la Srta. **Lady Maribel Ordoñez Poma**, portadora de la cedula de identidad número **1950102564**, egresada de la Carrera Desarrollo de Software del Instituto Superior Tecnológico Amazónico, misma que se encuentra bajo la tutoría del Ing. Diego Chamba Saca, previo a la obtención del título de Tecnóloga Superior en Desarrollo de Software.

Es todo cuanto puedo certificar en honor a la verdad, facultando al interesado hacer uso del presente en lo que considere conveniente.



.....
Jessica Astrid Quichimbo Lalangui
1105216285
Licenciada en Ciencias de la Educación Mención Idioma Inglés

10.2. Anexo 2: Entrevista



ENTREVISTA DIRIGIDA A LA EMPRESA MY WEB SERVICE

I. DATOS

a. ENTREVISTADO:

Nombres:	Juan Pablo	Apellidos:	Morocho
Teléfono/celular:	0969407691	Correo electrónico:	

b. ENTREVISTADOR:

Nombres:	Lady Maribel	Apellidos:	Ordoñez Poma
Teléfono/celular:		Correo electrónico:	

II. OBJETIVO:

Con la presente entrevista se pretende conocer y obtener la información necesaria para el desarrollo del sistema web, cabe indicar que el desarrollo de la entrevista esta con preguntas sobre como se realizan los procesos actualmente y también como se los puede realizar con la implementación del sistema.

III. DESARROLLO:

ESTUDIO PRELIMINAR	
¿Cuenta con un sistema informático que ayude a la optimización de los procesos en la empresa?	
Si ()	No (X)
1. En caso de responder SI en la pregunta anterior, ¿El sistema actual optimiza todos los procesos de su empresa?	
.....	
2. En caso de responder NO, ¿Estaria dispuesto a implantar un sistema informático en su empresa, y de ser así apoyaria con toda la información y documentación necesaria para su desarrollo?	
Si (X)	No ()
¿Por qué?	
Se requiere de un sistema informático que permita dar seguimiento a todos los proyectos de desarrollo de software que lleva a cabo la empresa, de esta manera optimizar la asignación de actividades a los desarrolladores.	

Si el entrevistado está dispuesto en la implantación de un sistema, se continua con la entrevista.

PROCESO ADMINISTRATIVO:

2.1. ¿Qué actividades realiza y Cómo las realiza?

Actividad 1: Registro de proyectos

Descripción:

Para el registro de la información del proyecto, se utiliza hojas de cálculo en Excel, en donde se almacena los datos del usuario o empresa que se beneficiará del sistema desarrollado por nosotros.

Actividad 2: Asignación de Actividades

Descripción:

Se delega mediante oficio a los programadores que trabajarán en el desarrollo del sistema informático, indicando quién será el responsable de dar seguimiento al proyecto y el plazo de desarrollo que se ha acordado con el usuario final.

Actividad 3: Revisión de Avances

Descripción:

El administrador del proyecto es el encargado de evidenciar el cumplimiento de los avances de cada una de las actividades que desarrollan los programadores, para lo cual los desarrolladores entregan un documento de acuerdo a un formato establecido por nuestra empresa, en donde describen la actividad realizada con su respectivo medio de verificación. Yo también reviso de vez en cuando estos avances.

Actividad 4: Entrega del producto

Descripción:

Una vez que se ha culminado con el desarrollo del sistema informático o servicio requerido por el interesado, se procede al despliegue en servidores contratados por nosotros mismos, para después entregar las credenciales de acceso al propietario y un breve manual de usabilidad del software. Brindamos una garantía de 3 meses para realizar mantenimiento al sistema o corregir cualquier bug o error que se pueda presentar.

2.2. ¿A manera de resumen, cómo es el proceso de desarrollo de software en su empresa?

Primeramente, se hace un estudio preliminar sobre el sistema requerido por el interesado o empresa interesada, si es factible de desarrollar se procede a tomar la información general y a armar un breve bosquejo de los requerimientos que necesita la parte interesada. Luego se procede a asignar los responsables del proyecto, para lo cual, se especifica a los trabajadores que actuarán como desarrolladores y el que actuará como Administrador del proyecto. El administrador del proyecto asignará actividades a cada uno de los desarrolladores, teniendo como punto de partida las fases del ciclo de vida de

software y los procesos que se deben cumplir en cada una de estas fases, con sus respectivos medios de verificación. Por lo general trabajamos con la metodología Iconix y con XP, entonces en base a la estructura de la metodología se asignan las tareas. Por lo general se desarrolla en el backEnd con los lenguajes de programación Java, Php y Python, y; en el frontEnd se utiliza los framework Angular o motores de plantilla de acuerdo al lenguaje seleccionado.

Una vez culminado el sistema informático y validado por la parte interesada se procede a desplegar el software en los servidores que tenemos contratados, a excepción que la parte interesada tenga sus propios servidores y que cumplan con las especificaciones del sistema.

Yantzaza, 28 de agosto de 2022



Entrevistador



Entrevistado

b. Proceso de Desarrollo de software

i. Actividades

ii. Avances de las actividades – medios de verificación

Código	Proceso	Descripción	Puntos claves	Actor/Responsable
PDS-01	Asignación de actividades	Asignar actividades a los desarrolladores indicando los plazos y una descripción de lo que deben realizar de acuerdo a la metodología de Desarrollo de software seleccionada para el Proyecto	Descripción de el proceso que se debe desarrollar, tiempos de entrega	Administrador del Proyecto
PDS-02	Revisión de actividades	Se revisa los medios de verificación que los programadores presentan, se utiliza un formato	Formato para la presentación de avances	Administrador del Proyecto
PDS-03	Cumplimiento de actividades	Las actividades asignadas pueden tener uno o varios avances dependiendo la complejidad y tamaño de la actividad	Uno o varios avances por actividad	Desarrolladores

Observaciones:

La empresa no dispone de un sistema informático que permita optimizar el servicio de Desarrollo de software que ofrecen.



Lady Ordoñez
Investigadora

10.4. Anexo 4: Resultados del test de usabilidad

En la siguiente tabla se puede observar los resultados que se obtuvieron del test de usabilidad aplicado al software, el total de participantes que realizaron el test fueron 3. Para cada pregunta está representado el número de frecuencias que obtuvieron.

Tabla 14

Resultados obtenidos – evaluación al software.

PREGUNTA	DE ACUERDO	INDECISO	DESACUERDO
El sistema responde lentamente a los datos de entrada	0	0	3
Las advertencias emitidas son útiles	3	0	0
El software se detiene o cuelga inesperadamente	0	0	3
Es muy difícil manipular el software	0	1	2
La interfaz es amigable	2	0	0
La información se presenta bien estructurada y ordenada	3	0	0
Disfruto del tiempo que se dedicó a usar este software	3	0	0
Toma demasiado tiempo aprender las funciones del software	2	1	0

Trabajar con este software es muy satisfactorio	3	0	0
No hay suficiente información en la pantalla cuando se necesita	0	0	3

PREGUNTA	DE ACUERDO	INDECISO	DESACUERDO
Le gustaría usar este software todos los días	3	0	0
Usar este software es frustrante	0	0	3
La velocidad de este software es lo suficientemente rápido	3	0	0
La distribución de los menús parece bastante lógica	3	0	0
Es fácil moverse de un proceso a otro	3	0	0
El software muestra mensajes informativos después de ejecutar una actividad	3	0	0
El software es intuitivo	2	1	0
Los colores y tamaños de las fuentes son atractivos	3	0	0

Es fácil hacer que el software haga exactamente lo que quiero	3	0	0
Es fácil ver a simple vista las opciones que ofrece el software	3	0	0

Resultados del test

Del test aplicado se seleccionan cinco preguntas relevantes para su realizar su gráfico.

- **Pregunta:** La información se presenta bien estructurada y ordenada

Se obtuvieron los siguientes resultados

Figura 45

Resultado Pregunta: La información se presenta bien estructurada y ordenada



- **Pregunta:** Es muy difícil manipular el software.

Se obtuvieron los siguientes resultados

Figura 46

Resultado Pregunta: Es muy difícil manipular el software.



- **Pregunta:** Es fácil moverse de un proceso a otro.

Se obtuvieron los siguientes resultados



- **Pregunta:** La velocidad de este software es lo suficientemente rápido.

Se obtuvieron los siguientes resultados.

Figura 47

Resultado Pregunta: La velocidad de este software es lo suficientemente rápido.



- **Pregunta:** Es fácil hacer que el software haga exactamente lo que quiero

Se obtuvieron los siguientes resultados

Figura 48

Resultado Pregunta: Es fácil hacer que el software haga exactamente lo que quiero



10.5. Anexo 5: Proyecto de Tesis



“DESARROLLO E IMPLEMENTACION DE UN SISTEMA WEB DE SEGUIMIENTO Y CONTROL DE LOS PROYECTOS INFORMÁTICOS DE LA EMPRESA MY WEB SERVICE”

LINEA DE INVESTIGACIÓN:

ADAPTACIÓN TECNOLÓGICA E
INNOVACIÓN

CARRERA:

TECNOLOGIA SUPERIOR EN DESARROLLO DE SOFTWARE

AUTOR:

Ordoñez Poma Lady Maribel

TUTOR:

Ing. Chamba Saca Diego Jonathan

FECHA:

15 de julio de 2022

YANZATZA-ZAMORA CHINCHIPE-ECUADOR

2022

**1. TEMA: DESARROLLO E IMPLEMENTACION DE UN SISTEMA
WEB DE SEGUIMIENTO Y CONTROL DE LOS PROYECTOS
INFORMÁTICOS DE LA EMPRESA MY WEB SERVICE**

2. PROBLEMA

La revolución tecnológica es un tema muy complejo, pero como paradigma es algo elocuente en la sociedad, hoy hablar de sistemas y de los avances de la tecnología del software y hardware. En países como Estados Unidos, Japón, China las evoluciones de las herramientas tecnológicas son en su mayoría para los procesos automáticos que generan tanto empresas públicas como privadas.

La adopción es cada vez mayor de la automatización industrial, las fábricas inteligentes, la Internet de las cosas y la impresión 3D están cambiando el proceso productivo, para “la llegada de la Inteligencia Artificial (IA) será inminente en todos los sectores del planeta”, los usos de las nuevas tecnologías para el desarrollo son perturbadoras para las economías en desarrollo, ya sea que sean indiferente a las nuevas tecnologías.

My Web Service, es una empresa que provee soluciones tecnológicas y de programación de acceso a Internet a clientes personales y corporativos, pudiendo acceder desde su casa, negocio o empresa a toda la variedad mundial de información disponible. Es una empresa que nos permite conectarnos con el mundo de la información. Y que su servicio es muy importante.

Los procesos manuales de la empresa My Web Service provocan pérdida de tiempo, ya que se realizan de forma manual, provocando inseguridad, duplicidad y pérdida de información adquirida.

3. JUSTIFICACION

Mediante el presente proyecto de investigación permitirá poner en práctica los conocimientos adquiridos, el cual nos servirá para realizar el proyecto de tesis siguiendo las fases de una metodología y así poder realizar una aplicación web de calidad, que cumpla con todas las necesidades de la empresa My Web Service.

La realización de dicho proyecto implicará el consumo de ciertos recursos económicos como gastos en dominios y servidores, impresión de documentos y otras cosas, estos gastos serán totalmente a responsabilidad del autor del proyecto.

Dicho sistema web lo realizare mediante el uso de herramienta de “Enterprise Architect”, para los procesos de análisis, recolección de información y demás herramientas que ayuden en el desarrollo de la aplicación, ya que dichas herramientas son completas y fácil de usar. Además, utilizare herramientas de lenguaje de programación como Java, Spring Boot, thymeleaf, la base de datos será posgretSql y un hosting.

Mediante este sistema web se pretende controlar todos los proyectos informáticos realizados dentro de dicha empresa, logrando obtener una aplicación eficiente, con la finalidad de lograr que nuestra aplicación web realice los debidos procesos con eficiencia y rapidez.

Durante el desarrollo de la aplicación no utilizaremos medios ecológicos, como papel, tinta, etc., materiales innecesarios que contaminan el medio ambiente y todo esto se llevaría a cabo de manera web.

4. REVISION DE LITERATURA

4.1. Contexto y estado del arte

4.1.1. Contexto

RESEÑA HISTÓRICA DE LA EMPRESA MY WEB SERVICE

Figura 1. Elaboración propia



La empresa se encuentra ubicada en las calles Av. Iván Riófrío entre primero de Mayo y Armando Arias junto al Gad Municipal de Yantzaza.

My Web Services es una empresa con un equipo profesional con una amplia experiencia en el sector de las TICs y desarrollo de páginas web. Cuentan con todo el conocimiento requerido para ser un soporte y ayudar a la comunidad con la implementación de soluciones informáticas y tecnológicas.

Tienen siempre presente las herramientas actuales en el desarrollo de programas, para brindar y asesorar a todos sus clientes diferentes alternativas

para la implementación de su proyecto; garantizando un producto final de excelente calidad, rendimiento y seguridad.

Misión

Ofrecer soluciones y confianza con el cumplimiento en tareas realizadas, enfoque empresarial, rendimiento y eficacia en el desarrollo de proyectos.

Visión

Ser un referente en Latinoamérica y el caribe en la generación de negocios y soluciones tecnológicas disruptivas.

4.1.2. Estado del arte

Mediante el desarrollo e implementación de un sistema web de seguimiento y control de los proyectos informáticos realizados por la empresa My Web Service, busca optimizar los procesos manuales y reducir el consumo de materiales de oficina. Al implementar este sistema estaríamos generando un impacto amigable con el medio ambiente, además el sistema contará con un usuario y contraseña para el personal que labora en la empresa y con esto buscamos automatizar los procesos manuales mediante el cual se podrá realizar los registros de una manera mucha más rápido y eficiente y evitando perdida de información.

El sistema fue desarrollado por la Escuela Superior Politécnica de Chimborazo por parte del autor José Maldonado en el año 2016 denominado “Desarrollo e implementación de un sistema para el seguimiento y evaluación de las practicas pre-profesionales para la facultad de Ingeniería Escuela Civil de la Puce” permitirá llevar un registro adecuado de las actividades realizadas por los estudiantes, ya que en la actualidad la Facultad de Ingeniería Escuela Civil no cuenta con un sistema de administración, que proporcione una información eficiente a las autoridades y estudiantes de manera ágil y oportuna.

Conclusión: A diferencia de la escuela politécnica, el sistema a desarrollar se enfocará precisamente en el control y seguimiento de proyectos proporcionando así a la empresa una herramienta mucho más útil que permita reemplazar los procesos manuales con los que actualmente trabaja a un sistema eficiente.

4.2. DESARROLLO ESPECÍFICO DE LA CONTRIBUCIÓN

Las herramientas tecnológicas que se muestran en este apartado, son las que se emplearán como base para la construcción del sistema informático de la empresa My Web Service, el motivo por el cual fueron seleccionadas, se debe primeramente a sus licencias gratuitas de libre acceso, su excelente documentación y amplia comunidad que permiten solventar cualquier inconveniente que se presente.

4.2.1. Comparación de Lenguajes de Programación

En la siguiente tabla se hace una comparación de los principales lenguajes de programación más utilizados para programación web:

Tabla 1.

Cuadro comparativo de lenguajes de programación

LENGUAJES DE PROGRAMACION		
Lenguajes de programacion con mayor relevancia		
Registros: 3	Nro. De Tabla: 1	
Java	Python	Php
<p>Definición: Es un lenguaje robusto, permite programar aplicativos orientados a la web, móvil y escritorio.</p> <p>Ventajas: Es multiplataforma, dinámico, gran abanico de librerías y comunidad muy grande</p> <p>Desventajas:</p>	<p>Definición: Es un lenguaje de programación de alto nivel que se utiliza para desarrollar aplicaciones de todo tipo.</p> <p>Ventajas: Es fácil y sencillo de aprender, está desarrollado bajo una licencia de código abierto,</p>	<p>Definición: Es un lenguaje de programación destinado a desarrollar aplicaciones web, favoreciendo la conexión entre los servidores y la interfaz de usuario</p> <p>Ventajas: Fácil de aprender y usar, versátil, de código abierto, no requiere complejos</p>

Disminuye el rendimiento de ejecución de los programas, requiere experiencia en programación, es de sintaxis completa.	permite una diversidad de desarrollos de una manera fácil, ágil y rápida. Desventajas: Algunas librerías que trae por defecto no son del gusto de amplio de la comunidad, y optan a usar librerías de terceros.	entornos para su desarrollo. Desventajas: Mala calidad de manejo de errores, seguridad baja.
--	--	---

Nota. Elaboración Propia

Para la programación del sistema informático para la empresa My Web Service, se utilizará el lenguaje de programación Java, debido a que es un lenguaje robusto, con una alta seguridad, apoyado por una gran comunidad y documentación que permite solventar cualquier problema que se presente en el desarrollo del software, a demás; una característica fundamental es que al emplear el framework Spring Boot ,se mejora la lógica de negocio y lógica de aplicación del sistema, permitiendo construir un producto final escalable y mantenible.

4.2.2. Java

Java es un lenguaje de programación y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems. Hay muchas aplicaciones y sitios web que no funcionarán a menos que tenga Java instalado y cada día se crean más. Java es rápido, seguro y fiable. Desde portátiles hasta centros de datos, desde consolas para juegos hasta súper computadoras, desde teléfonos móviles hasta Internet, Java está en todas partes. Para la construcción de este sistema informático se utilizará en le BACKEND la versión 11 de java. (Java.com, s.f.).

4.2.3. Spring Boot

Spring Boot facilita la creación de aplicaciones independientes basadas en el lenguaje de programación Java y en el framework Spring, que tienen un alto grado de producción que puede "simplemente ejecutar".

Las características más importantes que se puedan lograr con este framework, se detallan a continuación:

- Cree aplicaciones Spring independientes
- Incruste Tomcat, Jetty o Undertow directamente (no es necesario implementar archivos WAR)
- Proporcione dependencias 'de inicio' obstinadas para simplificar su configuración de compilación.
- Configure automáticamente las bibliotecas de Spring y de terceros siempre que sea posible.
- Proporcione funciones listas para producción, como métricas, verificaciones de estado y configuración externa.
- Absolutamente sin generación de código y sin requisitos de configuración XML. (Spring.io, s.f.)

Para este proyecto se empleará la versión 2.7.1 de Spring Boot.

4.2.4. Visual Studio Code

Visual Studio Code es un editor de código fuente que permite trabajar con diversos lenguajes de programación, admite gestionar tus propios atajos de teclado y refactorizar el código. Es gratuito, de código abierto y nos proporciona una utilidad para descargar y gestionar extensiones con las que podemos personalizar y potenciar esta herramienta.

Las extensiones de Visual Studio Code nos otorgan infinidad de opciones, como colorear tabulaciones, etiquetas o recomendaciones de autocompletado. También hay extensiones que nos ayudan con el lenguaje de programación que vayamos a usar, como por ejemplo para Python, C / C++, JavaScript, etc. (Code, s.f.)

4.2.5. StartUML

Dentro de las herramientas utilizadas en el proceso de ingeniería de sistemas informáticos se encuentra StarUML (StarUML, s.f.) , la cual se utiliza

para modelar el sistema basado en los estándares UML 2.0 y MDA (Modelo de Arquitectura Dirigida). Esta aplicación en un principio era un producto comercial, pero hace un tiempo pasó a ser un proyecto de licencia abierta GNU/GPL.

El software heredó todas las características de la versión comercial y poco a poco las ha ido mejorando; entre las mismas se encuentran:

- Soporte completo al diseño UML mediante:
- Diagrama de casos de uso
- Diagrama de clase
- Diagrama de secuencia
- Diagrama de colaboración.
- Diagrama de estados
- Diagrama de actividad.
- Diagrama de componentes
- Diagrama de despliegue.
- Diagrama de composición estructural (UML 2.0)
- Define elementos propios para los diagramas, que no necesariamente pertenecen al estándar de UML.
- Presenta la capacidad de generar código a partir de los diagramas y viceversa, actualmente funciona para los lenguajes C++, C# y Java.
- Genera documentación en formatos Word, Excel y PowerPoint sobre los diagramas.
- Ofrece plantillas de proyectos.
- Permite la posibilidad de crear plugins para el programa. (StartUML - EcuRed , s.f.).

4.2.6. Comparación de Bases de Datos

Tabla 2.

Cuadro comparativo de bases de datos

BASES DE DATOS

Gestores de Bases de Datos con mayor relevancia.

Registros: 3

Nro. De Tabla: 2

MySql	PostgreSql	MongoDB
<p>Características:</p> <ul style="list-style-type: none"> -Permite escoger múltiples motores de almacenamiento para cada tabla. -Agrupación de transacciones, pudiendo reunir las de forma múltiple desde varias conexiones con el fin de incrementar el número de transacciones por segundo. -Conectividad segura. -Ejecución de transacciones y uso de claves foráneas. -Presenta un amplio subconjunto del lenguaje SQL. -Replicación -Disponible en casi todas las plataformas o sistemas. 	<p>Características:</p> <ul style="list-style-type: none"> - Alta concurrencia. - Soporte para múltiples tipos de datos de manera nativa. - Soporte a triggers. - Trabajo con vistas. - Objeto/relacional. - Soporte para bases de datos distribuidas. - Soporte para gran cantidad de lenguajes. <p>PostgreSQL es capaz de trabajar con funciones internas, que se ejecutan en el servidor, escritas en diversos lenguajes como C, C++, Java, PHP, Python o Ruby. Además, ofrece interfaces para ODBC y JDBC, así como interfaces de programación para infinidad de lenguajes de programación.</p>	<p>Características:</p> <ul style="list-style-type: none"> - Potente sintaxis de consultas: lo que permite hacer consultas, desde las más sencillas hasta las más complejas, obteniendo todo tipo de información. - Indexación. - Se pueden crear índice, gestionarlos y rehacerlos, y tienen un gran impacto en el rendimiento de las consultas. - Transacciones - Base de datos distribuida. - Capacidad de aumentar los recursos relacionados con la memoria o la CPU del servidor

Nota. Elaboración Propia

Para el sistema informático se empleará la base de datos PostgreSQL, porque es capaz de funcionar de manera estable en el servidor, es consistente y tolerante a fallos, es compatible con el modelo relacional y principalmente tiene la capacidad de responder de manera confiable cuando se está trabajando con miles de registros de información.

4.2.7. PostgreSQL

PostgreSQL es un potente sistema de base de datos relacional de objetos de código abierto, abierto con más de 30 años de desarrollo activo que le ha ganado una sólida reputación por su fiabilidad, solidez de funciones y rendimiento. (PostgreSQL, s.f.).

Las principales prestaciones que ofrece PostgreSQL, por lo que es bastante popular y de uso comercial son:

- **Alta concurrencia.** Es capaz de atender a muchos clientes al mismo tiempo y entregar la misma información de sus tablas, sin bloqueos.
- **Soporte para múltiples tipos de datos de manera nativa.** Ofrece los tipos de datos habituales en los sistemas gestores, pero además muchos otros que no están disponibles en otros competidores, como direcciones IP, direcciones MAC, Arrays, números decimales con precisión configurable, figuras geométricas, etc.
- **Soporte a triggers.** Permite definir eventos y generar acciones cuando estos se disparan.
- **Trabajo con vistas.** Esto quiere decir que pueden consultar los datos de manera diferente al modo en el que se almacenan.
- **Objeto-relacional.** Otra de sus principales características, que permite trabajar con sus datos como si fueran objetos y ofrece mecanismos de la orientación a objetos, como herencia de tablas.
- **Soporte para bases de datos distribuidas.** Donde el trabajo con transacciones asegura que estas tendrán éxito cuando han podido realizarse en todos los sistemas involucrados.
- **Soporte para gran cantidad de lenguajes.** PostgreSQL es capaz de trabajar con funciones internas, que se ejecutan en el servidor, escritas en diversos lenguajes como C, C++, Java, PHP, Python o Ruby. Además, ofrece interfaces para ODBC y JDBC, así como interfaces de programación para infinidad de lenguajes de programación. (Arsys, 2018).

4.2.8. SQL

SQL es un lenguaje estándar de programación para bases de datos, este lenguaje se utiliza para acceder y manipular las bases de datos ya sean en MYSQL, Oracle, DB2, SQL Server, Access. También se dice que es un lenguaje estructurado y un estándar ANSI para el acceso y la manipulación de cualquier

base de datos, se compone de sentencias SQL, cada una con una funcionalidad diferente, por ejemplo:

- Creación de la base de datos (CREATE DATABASE)
- Creación de la tabla (CREATE TABLE) (SQL, 2012)

4.2.9. Comparación de Hosting – VPS

Tabla 3.

Cuadro comparativo de Hosting

HOSTING			
Hosting de mayor relevancia.			
Registros: 4	Nro. De Tabla: 3		
Características	Hostinger	BanaHosting	Sered
Precio	\$45 - mensual	\$60 - mensual	\$50 - mensual
Capacidad	200GB	120GB	120GB
Tráfico/mes	ilimitado	ilimitado	8 TB
CPU	8	4	4
RAM	12GB	8GB	8GB
Panel de control	varios	Cpanel	Cpanel

Nota. Elaboración Propia

El servidor virtual personal que se va a emplear para implantar el sistema informático es Hostinger, por su seguridad y servicios variados que ofrece. El pago o contrato de uso que se hará con este Hosting será por 6 meses, el cuál será pagado por el gerente de la empresa.

4.2.10. Comparación de servidores de Aplicaciones

Tabla 4.

Cuadro comparativo de servidores de Aplicaciones

SERVIDORES DE APLICACIONES			
Servidores de Aplicaicones con mayor relevancia.			
Registros: 7	Nro. De Tabla: 4		

Apache	Nginx	Apache Tomcat
<p>Ventajas:</p> <ul style="list-style-type: none"> - Software estable y confiable. - Código abierto y gratuito, incluso para uso comercial. - Parches de seguridad actualizados regularmente. - Compatible con Unix y Windows. - Enorme comunidad y soporte fácilmente disponible <p>Desventajas:</p> <ul style="list-style-type: none"> - Ciertos problemas de rendimiento en sitios web que tienen demasiado tráfico - Puede generar vulnerabilidades de seguridad. 	<p>Ventajas:</p> <ul style="list-style-type: none"> - Servidor de archivos estáticos y dinámicos - Sirve como proxy inverso + servidor de cache - Soporte de autenticación HTTP - Soporta índices y auto indexado - Acepta compresión Gzip y Brotli - Soporte para HTTPS con certificados SSL <p>Desventajas:</p> <ul style="list-style-type: none"> - No se integra con PHP de forma nativa. Es necesario usar FastCGI para poder integrarlo. - No es bueno para la creación de módulos: 	<p>Ventajas:</p> <ul style="list-style-type: none"> - Servidor de aplicaciones open source. - Fácil integración con Apache HTTP Server y con IIS. - Muy estable en sistemas UNIX. - Buena documentación online. - Compatible con Java - No requiere mucha memoria para arrancar. <p>Desventajas:</p> <ul style="list-style-type: none"> - Requiere que tanto los ambientes locales de desarrollo como los diferentes ambientes de producción estén sincronizados con la última versión personalizada de Tomcat

Nota. Elaboración Propia

El servidor de aplicaciones que se empleará para desplegar el sistema informático será Apache Tomcat, por su robustez, seguridad y estabilidad, además porque el framework Spring Boot lo integra preconfigurado y listo para trabajar en ambientes de producción optimizando de mejor manera las prestaciones de hardware que disponga el hosting o vps.

4.2.11. Servidor de aplicaciones Tomcat

El framework Spring Boot tiene embebido y configurado el servidor de aplicaciones Tomcat, listo para desplegar a producción cualquier sistema construido en el lenguaje de programación Java.

Tomcat, también llamado Apache Tomcat o Jakarta Tomcat es un contenedor open source de servlets para la implementación de Java Servlet, JavaServer

Pages (JSP), Java Expression Language y Java WebSocket. Las especificaciones de estos son desarrolladas bajo el Java Community Process.

Existe una confusión de conceptos entre Apache Tomcat y Apache, esto se debe a que coloquialmente al servidor Apache HTTP se le conoce como Apache. Tanto Tomcat como Apache HTTP son proyectos de Apache Software Foundation, cada uno implementado en un lenguaje diferente, Apache Tomcat en Java y Apache HTTP en C y XML. Aunque la diferencia fundamental es que el objetivo de Tomcat es servir específicamente aplicaciones Java, mientras que Apache es un servidor HTTP de propósito general. (Tomcat, 2022)

4.2.12. Modelo Vista Controlador

Modelo Vista Controlador (MVC) es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

Se trata de un modelo muy maduro y que ha demostrado su validez a lo largo de los años en todo tipo de aplicaciones, y sobre multitud de lenguajes y plataformas de desarrollo.

- El Modelo que contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia.
- La Vista, o interfaz de usuario, que compone la información que se envía al cliente y los mecanismos interacción con éste.
- El Controlador, que actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno. (Modelo vista controlador (MVC), s.f.).

4.2.13. Comparación de Tecnologías para el FrontEnd

Tabla 5.
Cuadro comparativo FrontEnd

FRONTEND Tecnologías con mayor relevancia

Registros: 3	Nro. De Tabla: 5	
Angular	React	Thymeleaf
<p>Características:</p> <ul style="list-style-type: none"> -Extiende HTML añadiendo funcionalidad y permitiendo crear plantillas dinámicas poderosas. - Permite usar TypeScript pero no es obligatorio. - Código reutilizable - Es opensource. 	<p>Características:</p> <ul style="list-style-type: none"> - Composición de componentes. - Desarrollo Declarativo Vs Imperativo. - Flujo de datos unidireccional. - Performance gracias al DOM Virtual. - Isomorfismo. - Componentes con y sin estado. 	<p>Características:</p> <ul style="list-style-type: none"> - Motor de plantillas Java para XML, XHTML y HTML5. - Trabaja en entornos web y no web (desconectados). - Basado en conjunto de características modulares llamadas Dialectos. - Muchas plantillas listas para ser utilizadas. - Soporte completo a la internacionalización.

Nota. Elaboración Propia

Como en el sistema informático para el lado BackEnd se va a emplear el lenguaje de programación Java con su framework Spring Boot, se ha optado para la parte BackEnd emplear el motor de plantillas Thymeleaf, ya que permite una integración completa con Java, además; permite realizar tareas que se conocen como natural templating, es decir; añade atributos y etiquetas, sobre todo HTML, permitiendo que nuestras plantillas se puedan renderizar en local, y esa misma plantilla después utilizarla también para que sea procesada dentro del motor de plantillas. Por lo cual las tareas de diseño y programación se pueden llevar conjuntamente.

5. OBJETIVOS

5.2. OBJETIVO GENERAL

Desarrollar e implantar un sistema web para el seguimiento y control de los proyectos informáticos de la empresa My Web Service del cantón Yantzaza.

5.3. OBJETIVOS ESPECÍFICOS

- Analizar la situación actual de la empresa My Web Service para obtener los requerimientos del sistema
- Establecer el diseño de la arquitectura funcional del sistema informático y realizar la codificación empleando el lenguaje de programación Java.
- Verificar la funcionalidad del sistema mediante la aplicación de pruebas con la finalidad de corroborar que el sistema realiza las necesidades planteadas por el usuario.
- Implantar el sistema informático en la empresa My Web Service

6. HIPÓTESIS

6.2. HIPÓTESIS GENERAL

La propuesta de desarrollar un sistema informático personalizado para la empresa My Web Service, mejora considerablemente el proceso de administración de eventos sociales, venta y alquiler de productos para fiestas y control del inventario.

6.3. HIPÓTESIS ESPECÍFICAS

- El levantamiento de información para la fase de análisis permite identificar los requerimientos funcionales y no funcionales del software, logrando construir un producto personalizado que solucione los problemas reales de la empresa, y; que aporte a su vez a la fase de diseño.

- En la fase de codificación los frameworks y arquitecturas utilizadas, inciden positivamente en el rendimiento del sistema y permiten brindar una interfaz gráfica amigable, para un uso fácil del sistema por parte de los usuarios.
- La realización de las pruebas de funcionalidad y rendimiento aplicadas al sistema, permiten evidenciar si el producto a entregar cumple con un nivel de calidad y seguridad óptimo.
- En la última fase de instalación e implantación del sistema, los recursos informáticos asignados son los óptimos para el correcto desempeño del producto y la capacitación efectuada a los usuarios satisface la mayoría de sus inquietudes.

7. METODOLOGÍA

Para el desarrollo del presente proyecto se utilizará la metodología de desarrollo ICONIX y algunos métodos de investigación como el deductivo, inductivo y analítico, además se aplicarán técnicas para el levantamiento de información como la observación y entrevista. A continuación, se detalla cada uno de estos:

- **Metodología de desarrollo Iconix:** Esta metodología es la base en la que se desarrollará el proyecto partiendo por las siguientes fases (Metodología Iconix, 2014) :
 - **Análisis de requisitos:** En esta fase se realiza un levantamiento de información, de la cual se obtendrán los requisitos que al principio deberían ser parte del sistema y los diferentes stakeholders que intervendrán.
 - **Análisis y Diseño Preliminar:** En esta fase se realiza la vinculación de los casos de uso a los objetos, mediante la elaboración del Análisis de Robustez, que permite capturar el Qué hacer y luego ayuda a decidir Cómo hacerlo.
 - **Diseño detallado:** En esta fase se procede a elaborar un diseño detallado de los casos de uso, por tal motivo, el texto de los casos de uso debe estar completo, correcto y detallado. Además, se debería contar con casi todos los conceptos del modelo de dominio.

- **Implementación y Pruebas:** Después de realizar un buen análisis y diseño, se procede a generar el código, para este proyecto se va a emplear como base el lenguaje de programación Java. Se emplean diferentes tipos de pruebas, para garantizar que el producto sea de calidad, eficiente y seguro.

- **Métodos de Investigación:** A continuación, se presentan los métodos que intervienen en el desarrollo del proyecto
 - **Deductivo:** Este método se lo emplea en el proyecto cuando en base al objetivo general, se va concluyendo con cada una de las funcionalidades del sistema, que resolverán los problemas actuales presentes en la empresa y que finalizará con una implementación y capacitación.
 - **Inductivo:** Este método se lo aplica en el proyecto, al momento que se determinan cada uno de los inconvenientes que tienen los usuarios que trabajan en la empresa, con el propósito de llegar a la construcción del sistema basado en las necesidades reales de estos usuarios y que al final se cumpla con sus expectativas.
 - **Analítico:** Este método se lo aplica al momento de interpretar y analizar detalladamente los requisitos del usuario y los diferentes diagramas que comprendieron el análisis de tal manera que, al guiarse en estos se puede estructurar el diseño funcional del sistema. Además, se lo emplea al analizar cada una de las pruebas realizadas al software, con la finalidad de entregar un producto de calidad.

- **Técnicas**
 - **Observación:** Consiste principalmente en ver los problemas que presenta la empresa y sus usuarios, además servirá para la recolección de información en forma preliminar.
 - **Entrevista:** Se aplica Anexo para recolectar la información de forma más precisa y clara directamente de las personas involucradas en el uso del

sistema, de tal manera que a la hora de hacer los requerimientos sean en base a las necesidades existentes.

- **Tablas Comparativas:** Permiten identificar las características, ventajas y desventajas en este caso de las tecnologías que se van a emplear en el desarrollo del sistema informático, y de esta forma determinar las que tengan las mejores prestaciones y servicios.

9. RECURSOS Y PRESUPUESTO

El siguiente detalle de los recursos se formula en base al cronograma de trabajo, en él se especifican los requerimientos humanos, económicos, materiales, técnicos y tecnológicos necesarios para la elaboración del proyecto.

El tiempo requerido para la elaboración del proyecto es de 3 meses y medio, suponiendo 20 días hábiles de media al mes, y trabajando a razón de 5 horas diarias, se obtiene un total de 350 horas.

Tabla 6.

Recursos humanos utilizados para el proyecto

PRESUPUESTO Talento Humano				
Registros: 2		Nro. De Tabla: 6		
Nombre	Cargo	Horas	Valor U.	Valor T
Lady Ordoñez	Egresado	350	\$10.00	\$3,500.00
Ing. Diego Chamba	Tutor de Tesis	32	\$10.00	\$320.00
TOTAL				\$3,820.00

Nota. Elaboración Propia.

En la siguiente tabla se muestra el equipamiento necesario para construir el sistema.

Tabla 7.

Recursos Hardware y software

PRESUPUESTO Recursos Hardware y Software.				
Registros: 3		Nro. De Tabla: 7		
Descripción	Cantidad	Valor U.	Valor T.	
Hardware				
Laptop	1	\$1000	\$1000	
Impresora	1	\$50	\$50	
Flash memory 8GB	1	\$10	\$10	

Software			
Windows 10 Home	1	\$169	\$169
Paquete Office	1	\$119.99	\$119.99
Visual Studio Code	1	\$0	\$0
StartUML	1	\$0	\$0
Java	1	\$0	\$0
PostgreSql	1	\$0	\$0
VPS – Hostinger / 6 meses	1	\$72.00	\$72.00
TOTAL			\$1420.00

Nota. Elaboración Propia.

A continuación, se presenta una tabla de imprevistos que puede haber al desarrollar la tesis.

Tabla 8.

Recursos de oficina

**PRESUPUESTO
Recurso de Oficina.**

Registros: 6			Nro. De Tabla: 8
Descripción	Cantidad	Valor U.	Valor T.
Tinta de impresora	1	\$20	\$20
Resma de papel bond	1	\$10	\$10
Transporte	1	\$10	\$10
Internet	20 horas	\$0.50	\$10
TOTAL			\$50

Nota. Elaboración Propia.

A continuación, se detalla el costo total de la construcción del proyecto, así como de ciertos imprevistos que puedan ocurrir.

Tabla 9.

Presupuesto total del proyecto

PRESUPUESTO
Costo total de Proyecto.

Registros: 3

Nro. De Tabla: 9

Descripción	Valor
Recursos Humanos	\$3,820.00
Hardware y Software	1,420.00
Oficina	\$50.0
Imprevistos	\$20.0
TOTAL	\$5,310.00

Nota. Elaboración Propia.

10. BIBLIOGRAFÍA

Metodología Iconix. (16 de 2 de 2014). Obtenido de <http://metodologiaiconix.blogspot.com/>

Code, V. (s.f.). *Visual Studio Code - Code Editing.* Obtenido de <https://code.visualstudio.com/>

Odoo S.A. (2021). *Odoo ERP a tu servicio.* Obtenido de Odoo Fabricación - Industria 4.0 Cloud MRP: https://www.odoo.com/es_ES/app/manufacturing

Tomcat. (2022). *Características del servidor de aplicaciones Tomcat.* Obtenido de <https://tomcat.apache.org/>

Reyes, C. y Téllez, K. (2016). *Diseño de una aplicación móvil para eventos.* Obtenido de: <https://repository.javeriana.edu.co/bitstream/handle/10554/19945/ReyesJimenezCamiloErnesto2016.pdf?sequence=3>

Coronado, P. y Murrieta, E. (2004). *E-fiesta, software para planificación y administración de eventos sociales.* Obtenido de: <https://www.dspace.espol.edu.ec/bitstream/123456789/419/1/737.pdf>

Eventia. (2015). *Software para administración de eventos.* Obtenido de corporación Eventia: <https://www.eventtia.com/es/software-organizacion-eventos>

Java.com. (s.f.). *Características de Java.* Obtenido de https://www.java.com/es/download/help/whatis_java.html

Spring.io. (s.f.). *Framework Spring Boot.* Obtenido de Spring Boot: <https://spring.io/projects/spring-boot>

Peralta, J. (s.f.). *Diferencia entre Software a la Medida y Software Estándar* | GBS Global BestTech Systems. Obtenido de Gbts.com.mx.: <http://www.gbts.com.mx/diferencia-entre-software-a-la-medida-y-software-estandar/>

PostgreSQL. (s.f.). *Base de datos PostgreSQL*. Obtenido de <https://www.postgresql.org/>

Arsys. (2018). *¿Qué es PostgreSQL y por qué llevarlo a Cloud?*. Obtenido de: <https://www.arsys.es/blog/postgresql-servidores#:~:text=PostgreSQL%20es%20un%20sistema%20de,consistente%20y%20tolerante%20a%20fallos>.

Kruchten, P. (2006). *Planos Arquitectónicos: El Modelo de "4+1" Vistas de la Arquitectura del Software*. Obtenido de IEEE Software : http://cic.puj.edu.co/wiki/lib/exe/fetch.php?media=materias:modelo4_1.pdf

Modelo vista controlador (MVC). (s.f.). Obtenido de Servicio de Informática ASP.NET MVC 3 Framework.: [https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html#:~:text=Modelo%20Vista%20Controlador%20\(MVC\)%20es,control%20en%20tres%20componentes%20distintos.&text=La%20Vista%20o%20interfaz%20de,los%20mecanismos%20interacci%C3](https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html#:~:text=Modelo%20Vista%20Controlador%20(MVC)%20es,control%20en%20tres%20componentes%20distintos.&text=La%20Vista%20o%20interfaz%20de,los%20mecanismos%20interacci%C3)

SQL. (2012). *Lenguaje de consulta Sql*. Obtenido de <http://sql.11sql.com/sql-intro.htm>

StartUML - EcuRed . (s.f.). Obtenido de EcuRed.cu: <https://www.ecured.cu/StartUML>

StarUML. (s.f.). *Características y Modelado*. Obtenido de <https://staruml.io/>