

UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

“SISTEMA DE INFORMACIÓN WEB PARA LA GESTIÓN Y CONTROL DE DOCUMENTACIÓN PERSONAL”

CASO: Unidad de Registros
Gobierno Autónomo Municipal de El Alto

Para Optar al Título de Licenciatura en Ingeniería de Sistemas
MENCIÓN: INFORMÁTICA Y COMUNICACIONES

Postulante: Juan Daniel Mamani Quispe

Tutor Metodológico: M. Sc. Lic. Ing. Fanny Helen Perez Mamani

Tutor Especialista: Lic. Cristian Mercado Quispe

Tutor Revisor: Ing. Marisol Arguedas Balladares

EL ALTO - BOLIVIA

2024

**DECLARACIÓN JURADA DE
AUTENTICIDAD Y RESPONSABILIDAD**

Yo, Juan Daniel Mamani Quispe **estudiante con C.I. 13606050 LP** mediante la presente **declaro** de manera pública que la propuesta del **TRABAJO DE GRADO** titulada “**SISTEMA DE INFORMACIÓN WEB PARA LA GESTIÓN Y CONTROL DE DOCUMENTACIÓN PERSONAL**” es original, siendo resultado de mi trabajo personal y no constituye una copia o replica de trabajos similares elaborados,

Autorizo la publicación del resumen de mi propuesta en internet y me comprometo a responder a todos los cuestionamientos que se desprenden de su lectura.

Asimismo, me hago responsable ante la universidad o terceros, de cualquiera irregularidad o daño que pudiera ocasionar, por el incumplimiento de lo declarado.

De identificarse falsificación, plagio, fraude, o que el **TRABAJO DE GRADO** haya sido publicado anteriormente; asumo las consecuencias y sanciones que de mi acción se deriven, responsabilizándome por todas las cargas legales que se deriven de ello sometiéndome a las normas establecidas y vigentes de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

El Alto, junio del 2024.

.....
Juan Daniel Mamani Quispe
C.I. 13606050 LP
juandanielquispeperedo@gmail.com

Dedicatoria

El presente proyecto está dedicado a mis padres, quienes, con su amor, apoyo y sacrificio, han hecho posible que hoy logre concluir mi proyecto de grado. Gracias por ser mi inspiración y ejemplo de perseverancia, dedicación y compromiso. Sin su guía, sabiduría y orientación, no habría sido posible alcanzar esta meta. Agradezco infinitamente su confianza y paciencia, así como todo lo que han hecho por mi formación académica y personal. Les dedico este logro con todo mi corazón, y espero poder retribuirles, de alguna manera, todo lo que me han dado. Los amo y respeto profundamente.

Agradecimiento

Agradezco sinceramente a todas las personas que me apoyaron en la realización de este proyecto de grado. En primer lugar, quisiera expresar mi gratitud a mis tutores por su guía constante, sus conocimientos y su paciencia.

También quiero agradecer a mis amigos y familiares por su constante apoyo emocional y por creer en mí durante todo este proceso.

Además, me gustaría agradecer al Gobierno Autónomo Municipal de El Alto por brindarme la oportunidad de llevar a cabo este proyecto y por proporcionarme los recursos necesarios para su realización. Asimismo, quiero agradecer al Ing. Diego Goitia y al Tec. Wilmer Zambrana, por su valiosa contribución y asesoramiento durante mi investigación.

Finalmente, quiero expresar mi profunda gratitud a todos aquellos que participaron en este proyecto al dedicar su tiempo, proporcionar su opinión y compartir sus experiencias. Este proyecto no hubiera sido posible sin su ayuda y apoyo.

¡Gracias a todos por su contribución a este proyecto de grado!

ÍNDICE

1. Capítulo 1: Marco Preliminar	1
1.1 <i>Introducción</i>	1
1.2 <i>Antecedentes</i>	2
1.2.1 Antecedentes Institucionales	2
1.2.2 Misión	3
1.2.3 Visión	3
1.2.4 Objetivo	3
1.2.5 Organigrama.....	4
1.3 <i>Antecedente Afines al Trabajo de Grado</i>	5
1.3.1 Antecedentes Internacionales.....	5
1.3.2 Antecedentes Nacionales	5
1.3.3 Antecedentes Locales	6
1.4 <i>Planteamiento del Problema</i>	6
1.4.1 Problema Principal.....	7
1.4.2 Problemas Secundarios.....	8
1.5 <i>Formulación del Problema</i>	8
1.6 <i>Objetivos</i>	9
1.6.1 Objetivo General.....	9
1.6.2 Objetivos Específicos	9
1.7 <i>Justificación</i>	10
1.7.1 Justificación Técnica.....	10
1.7.2 Justificación Económica	10
1.7.3 Justificación Social	11

1.8	<i>Metodología</i>	11
1.8.1	Metodología de Desarrollo.....	12
1.8.2	Estimación del Costo.....	12
1.8.3	Métricas de Calidad.....	13
1.8.4	Seguridad.....	13
1.8.5	Pruebas de Software.....	14
1.8.6	Pruebas de Estrés.....	15
1.8.7	Métodos para la recolección de información.....	15
1.9	<i>Herramientas</i>	15
1.9.1	Hardware.....	16
1.9.2	Arquitectura de Software por Capas.....	16
1.9.3	Tecnologías Frontend o del Lado del Cliente.....	17
1.9.4	Tecnologías Backend o del Lado del Servidor.....	18
1.10	<i>Límites y Alcances</i>	19
1.10.1	Límites.....	19
1.10.2	Alcances.....	20
1.11	<i>Aportes</i>	21
2.	Capítulo 2: Marco Teórico	22
2.1	<i>Introducción</i>	22
2.2	<i>Conceptos Fundamentales</i>	22
2.2.1	Sistema.....	22
2.2.2	Dato.....	23
2.2.3	Información.....	24
2.3	<i>Sistema de Información Web</i>	25

2.3.1	Elementos de los Sistemas de Información:	26
2.3.2	Ciclo de Desarrollo	26
2.3.2.1	Etapas de Desarrollo.:	27
2.4	<i>Metodología de Desarrollo de Software</i>	29
2.4.1	Metodología de Desarrollo de Software OpenUP	29
2.4.1.1	Principios del OpenUP.	30
2.4.1.2	Ciclo de Vida y Procesos de OpenUP.	31
2.4.1.3	Ventajas.	33
2.4.1.4	Desventajas.....	34
2.5	<i>Arquitectura de Software</i>	34
2.5.1	Arquitectura por Capas.....	36
2.5.1.1	Capa de Presentación.....	37
2.5.1.2	Capa de Reglas de Negocio (Empresarial).....	37
2.5.1.3	Capa de Datos	37
2.5.2	Características de la Arquitectura por Capas.....	38
2.5.3	Principios de la Arquitectura por Capas.....	39
2.5.4	Ventajas de la Arquitectura por Capas.....	39
2.5.5	Desventajas de la Arquitectura por Capas.....	40
2.5.6	Conclusiones de la Arquitectura	41
2.6	<i>Herramientas de Desarrollo Utilizadas</i>	41
2.6.1	Mongo DB	42
2.6.2	Express.js.....	43
2.6.3	React.....	44
2.6.4	Node.js	45
2.6.5	Axios	46

2.6.6	Redux Toolkit.....	47
2.6.7	Tailwind CSS	48
2.6.8	Bibliotecas y Librerías Utilizadas en el Proyecto.....	49
2.7	<i>Método de Estimación de Costos (COCOMO II)</i>	50
2.7.1	Características de modelo COCOMO II	51
2.7.2	La Importancia de COCOMO II en un Mercado Diverso	51
2.7.3	Modelos de Estimación COCOMO II	52
2.7.3.1	Modelo Post-Arquitectura.	53
2.8	<i>Centralización, Gestión y Control de Documentación Digital</i>	60
2.8.1	Importancia de la Gestión Documental	60
2.8.2	Beneficios de la Centralización de Documentos	61
2.9	<i>Seguridad y Pruebas de Software</i>	63
2.9.1	Seguridad y Validación de Datos en Aplicaciones Web	63
2.9.2	Prevención de Ataques.....	64
2.9.3	Políticas de Seguridad en Aplicaciones	65
2.9.4	Ejemplos de Problemas Comunes en Seguridad de Aplicaciones	66
2.9.5	Integración de la Seguridad en el Ciclo de Desarrollo.....	68
2.9.6	Pruebas de Software	70
2.9.6.1	Pruebas de Caja Blanca.....	70
2.9.6.2	Pruebas de Caja Negra.....	70
2.9.6.3	Pruebas de Estrés (Stress Testing)	71
2.10	<i>Métricas de Calidad: Modelo FURPS</i>	72
2.10.1	Requerimientos Funcionales (F)	72
2.10.2	Requerimientos no Funcionales (URPS)	72
2.11	<i>Métodos para la recolección de información</i>	73

2.11.1	Encuestas.....	73
2.11.2	Entrevista	74
2.11.3	Observaciones.....	74
2.11.4	Revisión de registros	75
2.12	<i>Beneficios del Sistema</i>	75
2.13	<i>Conclusiones del Marco Teórico</i>	76
3.	Capítulo 3: Marco Aplicativo	78
3.1	<i>Introducción</i>	78
3.2	<i>Aplicación de la metodología OpenUp</i>	79
3.2.1	Primera Iteración	79
3.2.1.1	Fase de Inicio.....	79
3.2.1.1.1	Comprensión de las Necesidades del Proyecto.....	79
3.2.1.1.2	Definición de Objetivos.....	81
3.2.1.2	Fase de Elaboración.....	83
3.2.1.2.1	Diseño y Detallado de Requerimientos.. ..	83
3.2.1.2.2	Diagrama Navegacional:.....	97
3.2.1.2.3	Diseño de Interfaz.....	99
3.2.1.3	Fase de Construcción.....	111
3.2.1.3.1	Modelado de Datos para MongoDB:	111
3.2.1.3.2	Desarrollo del Frontend.....	115
3.2.1.3.3	Seguridad en el Frontend.....	117
3.2.1.3.4	Desarrollo del Backend.....	119
3.2.1.3.5	Seguridad en el Backend.....	121
3.2.1.3.6	Interfaz de Usuario en Funcionamiento.....	122
3.2.1.4	Fase de Transición.....	127

3.2.1.4.1 Realización de Pruebas Exhaustivas.....	127
3.2.1.4.2 Implementación en Servidores.....	128
4. Capítulo 4: Calidad, Costo, Seguridad y Pruebas	129
4.1 <i>Introducción.....</i>	129
4.2 <i>Medición y Evaluación de la Calidad del Software mediante el Modelo FURPS ...</i>	130
4.2.2 Recopilación de Datos.....	134
4.2.3 Resultados Obtenidos	135
4.2.3.1 Análisis del Factor Funcionalidad.....	135
4.2.3.2 Análisis del factor Usabilidad.....	135
4.2.3.3 Análisis del factor Fiabilidad.....	136
4.2.3.4 Análisis del factor Rendimiento.....	136
4.2.3.5 Análisis del factor Soporte.....	137
4.2.4 Análisis global.....	137
4.3 <i>Estimación de Costos y Esfuerzos del Software mediante el Modelo COCOMO II</i>	138
4.4 <i>Seguridad Informática y Protección de Datos</i>	143
4.4.1 Planteamiento de las Políticas de Seguridad.....	143
4.4.1.1 Mantenimiento del Software.. ..	143
4.4.1.2 Manejo de la Base de Datos.....	144
4.4.1.3 Manejo de Archivos.....	144
4.4.1.4 Planteamiento de Políticas para Usuarios.....	145
4.4.1.5 Administradores del Sistema.....	145
4.4.1.6 Integración de Nuevos Módulos.....	146
4.4.2 Características de Seguridad Implementadas.....	147
4.4.2.1 Estructura de Archivos.....	147
4.4.2.2 Autenticación y Autorización.....	147

4.4.2.3 Control de Acceso	147
4.4.2.4 Unicidad de Nombres de Usuario.....	148
4.4.2.5 Validación Rigurosa.....	148
4.4.3 Protocolo ante Incidentes de Seguridad	148
4.4.3.1 Detección y Respuesta.....	148
4.4.3.2 Notificación de Incidentes.....	149
4.4.3.3 Revisiones Post-Incidentes.	149
4.4.4 Recomendaciones de Seguridad.....	149
4.5 Pruebas.....	151
4.5.1 Pruebas de Caja Blanca	152
4.5.2 Pruebas de Caja Negra	156
4.5.3 Pruebas de Estrés	164
5. Capítulo 5: Conclusiones y Recomendaciones.....	167
5.1 Introducción.....	167
5.2 Conclusiones	167
5.3 Recomendaciones.....	168

ÍNDICE DE FIGURAS

Figura 1 Organigrama de la Unidad de Registros	4
Figura 2 Proceso de Desarrollo de Sistemas de Información	28
Figura 3 Ciclo de Vida de OpenUP	32
Figura 4 Patrón de Arquitectura por capas	38
Figura 5 Diversidad del Mercado Actual y Futuro del desarrollo de software	52
Figura 6 Diagrama de Caso de Uso: Autenticación del Usuario	84
Figura 7 Diagrama de Caso de Uso: Registrar Información Personal	85
Figura 8 Diagrama de Caso de Uso: Registrar Información Complementaria	85
Figura 9 Diagrama de Caso de Uso: Registrar Información Académica	86
Figura 10 Diagrama de Caso de Uso: Registrar los Estudios Realizados	86
Figura 11 Diagrama de Caso de Uso: Registrar Seguro de Salud y Declaraciones Juradas	87
Figura 12 Diagrama de Caso de Uso: Registrar Información de Experiencia Laboral	87
Figura 13 Diagrama de Caso de Uso: Registrar Cumplimiento de Documentación	88
Figura 14 Diagrama de Caso de Uso: Registrar Información de Dependientes	88
Figura 15 Diagrama de Caso de Uso: Registrar Información de Discapacidad	89
Figura 16 Diagrama de Caso de Uso: Impresión de Documentación	89
Figura 17 Diagrama de Caso de Uso: Revisión de Información	90
Figura 18 Diagrama de Navegación: Vistas del Revisor	97
Figura 19 Diagrama de Navegación: Vistas del Usuario	98
Figura 20 Diseño de Interfaz: Login	99
Figura 21 Diseño de Interfaz: Registro de Usuarios	100
Figura 22 Diseño de Interfaz: Mensaje de Bienvenida	100
Figura 23 Diseño de Interfaz: Módulo Registro de Información Personal	101
Figura 24 Diseño de Interfaz: Módulo Registro de Información Complementaria	102
Figura 25 Diseño de Interfaz: Módulo Registro de Información Académica	103

Figura 26 Diseño de Interfaz: Módulo Registro de Información Estudios Realizados	104
Figura 27 Diseño de Interfaz: Módulo Registro de Información del Seguro de Salud y Declaraciones Juradas	105
Figura 28 Diseño de Interfaz: Módulo Registro de Información de la Experiencia Laboral .	106
Figura 29 Diseño de Interfaz: Módulo Registro de Tenencia de Documentación.....	107
Figura 30 Diseño de Interfaz: Módulo Registro de Familiares Dependientes	108
Figura 31 Diseño de Interfaz: Módulo Registro de Discapacidad	109
Figura 32 Diseño de Interfaz: Módulo Revisión de Información.....	110
Figura 33 Diseño de Interfaz: Barra de Navegación (Navbar)	110
Figura 34 Modelo de Datos: Documento Información Personal.....	111
Figura 35 Modelo de Datos: Documento Información Complementaria.....	112
Figura 36 Modelo de Datos: Documento Información Académica	112
Figura 37 Modelo de Datos: Documento de los Estudios Cursados	112
Figura 38 Modelo de Datos: Documento Seguro de Salud y Declaraciones Juradas	113
Figura 39 Modelo de Datos: Documento de Tenencia de Documentación	113
Figura 40 Modelo de Datos: Documento de Experiencia Laboral.....	113
Figura 41 Modelo de Datos: Documento Información de Dependientes.....	114
Figura 42 Modelo de Datos: Documento Información de Discapacidad.....	114
Figura 43 Modelo de Datos: Documento de Usuario	114
Figura 44 Estructura del Frontend.....	115
Figura 45 Expresiones Regulares Implementadas en el sistema.	117
Figura 46 Validación de Extensión y Tamaño de Archivos	118
Figura 47 Integración de Formik y Expresiones Regulares	118
Figura 48 Estructura del Backend	119
Figura 49 Interfaz de Usuario: Login	122
Figura 50 Interfaz de Usuario: Registro de Usuarios	123

Figura 51 Interfaz de Usuario: Visualización de Datos Personales.....	123
Figura 52 Interfaz de Usuario: Crear y Actualizar los Datos Personales.....	124
Figura 53 Interfaz de Usuario: Visualización de Datos Complementarios.....	124
Figura 54 Interfaz de Usuario: Crear y Actualizar los Datos Complementarios.....	125
Figura 55 Interfaz de Usuario: Modal Editar Título Profesional.....	125
Figura 56 Interfaz de Usuario: Impresión de Documentación	126
Figura 57 Interfaz de Usuario: Revisión de Información	126
Figura 58 Interfaz de Usuario: Alertas.....	126
Figura 59 Peticiones desde Postman para Evaluar la Respuesta del Backend	127
Figura 60 Implementación en servidores utilizando la herramienta WinScp	128
Figura 61 Formulario para la Estimación de Esfuerzo, Costo y Tiempo de Desarrollo Utilizando COCOMO II.....	142
Figura 62 Módulo Inicial de Acceso y Registro Personal.....	165

ÍNDICE DE TABLAS

Tabla 1	Factores de Escala	55
Tabla 2	Valores del Factor de Escala.....	56
Tabla 3	Valores de los Factores Producto.....	57
Tabla 4	Valores de los Factores Producto.....	58
Tabla 5	Valores de los Factores de Personal.....	59
Tabla 6	Valores de los Factores del Proyecto.	60
Tabla 7	Descripción de los Casos de Uso de Visualización de de Información.....	91
Tabla 8	Descripción de los Casos de Uso de Crear Información.....	92
Tabla 9	Descripción de los Casos de Uso de Actualizar Información.....	93
Tabla 10	Descripción de los Casos de Uso de Verificación de la Información.....	94
Tabla 11	Descripción de los Casos de Uso de los CRUDs de Información Adicional.....	95
Tabla 12	Preguntas para Evaluar los Factores FURPS	131
Tabla 13	Factores FURPS que Aplican Escala Likert	134
Tabla 14	Valoración de la Funcionalidad del Sistema	135
Tabla 15	Valoración de la Usabilidad del Sistema.....	135
Tabla 16	Valoración de la Fiabilidad del Sistema.....	136
Tabla 17	Valoración del Rendimiento del Sistema	137
Tabla 18	Valoración del Soporte de Sistema	137
Tabla 19	Valoración Considerando todos los aspectos FURPS.....	138
Tabla 20	Mantenimiento de Software.....	143
Tabla 21	Manejo de la Base de Datos	144
Tabla 22	Manejo de Archivos.....	144
Tabla 23	Usuarios del Sistema	145
Tabla 24	Administradores del Sistema.....	146

Tabla 25	Desarrollo de Módulos Nuevos	146
Tabla 26	Recomendaciones de seguridad	149
Tabla 27	Casos de Prueba de Caja Blanca.....	152
Tabla 28	Casos de Prueba de Caja Negra.....	157
Tabla 29	Resultados de Pruebas de Estrés	165

ÍNDICE DE ECUACIONES

Ecuacion 1 Esfuerzo Estimado usando los 17 factores multiplicadores.	53
Ecuacion 2 Esfuerzo Nominal.....	53
Ecuacion 3 Factor Exponencial de Escala (B)	54

RESUMEN

El proyecto de grado se enfoca en desarrollar un sistema web que ofrece una solución eficaz y eficiente para gestionar y controlar la documentación personal. Facilita la revisión, certificación y emisión de reportes, además de proporcionar un seguimiento detallado de la información del personal en tiempo real.

Este sistema brinda a los funcionarios la capacidad de imprimir cartas y certificaciones kardex necesarias para diversos trámites institucionales. No solo optimiza la gestión documental, sino que también agiliza los procesos administrativos, mejorando la eficiencia y el servicio ofrecido tanto a los funcionarios como a la institución en su conjunto.

El diseño del sistema web se ha orientado para ser intuitivo y fácil de usar por el personal del Gobierno Autónomo Municipal de El Alto. Ofrece una variedad de funcionalidades destinadas a mejorar la productividad y eficiencia del personal de la Unidad de Registros.

El proyecto de grado también incluye la implementación de medidas de seguridad para proteger la información del personal y garantizar la privacidad de los datos. La aplicación web se desarrolla utilizando tecnologías modernas de programación y diseño, asegurando su compatibilidad con diversos dispositivos y navegadores.

Se han empleado herramientas de código abierto como Node.js con Express en el backend y MongoDB para la base de datos. En el frontend, se ha seleccionado React junto con Tailwind CSS para el desarrollo de la interfaz de usuario. Además, se ha integrado la biblioteca Formik para la gestión de formularios. Estas elecciones se basan en su fiabilidad, rendimiento y amplio respaldo comunitario, garantizando un desarrollo eficiente y una implementación exitosa del sistema.

Palabras clave: Documentación, Certificación, Gestión, MongoDB.

ABSTRACT

The degree project focuses on developing a web system that offers an effective and efficient solution to manage and control personal documentation. It facilitates the review, certification and issuance of reports, in addition to providing detailed monitoring of personnel information in real time.

This system provides officials with the ability to print letters and kardex certifications necessary for various institutional procedures. It not only optimizes document management, but also streamlines administrative processes, improving efficiency and the service offered to both officials and the institution as a whole.

The design of the web system has been oriented to be intuitive and easy to use by the staff of the Autonomous Municipal Government of El Alto. It offers a variety of functionalities aimed at improving the productivity and efficiency of the Records Unit staff.

The degree project also includes the implementation of security measures to protect personnel information and guarantee data privacy. The web application is developed using modern programming and design technologies, ensuring its compatibility with various devices and browsers.

Open source tools such as Node.js with Express in the backend and MongoDB for the database have been used. On the frontend, React along with Tailwind CSS has been selected for UI development. In addition, the Formik library has been integrated for forms management. These choices are based on their reliability, performance and broad community support, ensuring efficient development and successful implementation of the system.

Keywords: *Documentation, Certification, Management, MongoDB.*



CAPITULO I

MARCO PRELIMINAR



**INGENIERÍA
DE SISTEMAS**

UNIVERSIDAD PÚBLICA DE EL ALTO

1. Capítulo 1: Marco Preliminar

1.1 Introducción

En los últimos años, los sistemas de información en las organizaciones públicas y privadas son una necesidad, buscando herramientas cada vez más rápidas para el procesamiento de datos, debido al gran volumen de información que se maneja. Estos sistemas ayudan a la centralización, gestión y control de la información, lo que posteriormente facilita la toma de decisiones. Todo esto con el fin de conseguir resultados que vayan en beneficio de la institución y alcanzarlos objetivos trazados como organización.

En la Unidad de Registro, dependiente de dirección de Talento Humano perteneciente al Gobierno Autónomo municipal de El Alto, se maneja una gran cantidad de información. Esta información se origina cuando una persona quiere acceder a un puesto de trabajo en la institución, teniendo que llenar formularios, adjuntar documentos personales, académicos y documentos de experiencia laboral. Actualmente, esta información se almacena en carpetas, lo que genera desorganización, mala gestión, falta de control y retrasos en la verificación de estos documentos.

El proyecto “Sistema de Información Web para la Gestión y Control de Documentación Personal” es una aplicación web que centraliza la información mediante el uso de una base de datos, interfaces de usuario con formularios que el funcionario, previa contratación deberá llenar. Esto ayudará a la Unidad de Registros a controlar y hacer seguimiento de la información ingresada en el sistema, teniendo un acceso a la información de una manera oportuna y precisa.

Se utiliza OpenUP, que es una metodología de desarrollo de software ágil y ligera que se enfoca en la colaboración en equipo, la entrega iterativa e incremental y la retroalimentación continua del equipo. Seguiremos el patrón de arquitectura por capas, un enfoque de diseño de sistemas o aplicaciones que divide un sistema en componentes o capas distintas, donde cada

capa tiene una función específica y se comunica con las capas adyacentes de una manera bien definida.

Las herramientas que se utilizan para el desarrollo e implementación del sistema serán: JavaScript como lenguaje de programación tanto para el backend y frontend, se utilizó React que es una biblioteca de JavaScript para crear interfaces interactivas, para dar estilo a las interfaces de usuario se utiliza Tailwind CSS que es un Framework de CSS, NodeJS para ejecutar código JavaScript del lado del servidor, MongoDB como base de datos NoSQL y Visual Studio Code como editor de código fuente. Finalmente, se utilizó el método COCOMO II para la estimación de costo del software.

1.2 Antecedentes

El Alto, ciudad boliviana de larga data, emergió tras demandas por infraestructura en los años 40. Organizaciones vecinales como FSUTCEA y FEJUVE lideraron la búsqueda de autonomía. La Ley 728, promulgada en 1985, estableció su autonomía. En el siglo XXI, El Alto es un referente nacional, marcado por movimientos sociales y políticos.

1.2.1 Antecedentes Institucionales

“El Alto, una ciudad de importancia en Bolivia, tiene una rica historia que se remonta a más de un milenio en paralelo con La Paz. Su creación como ciudad estuvo precedida por demandas de servicios de infraestructura básica en la década de 1940, cuando se formaron las primeras juntas de vecinos en la llanura de El Alto. Estas organizaciones vecinales, como la Federación Sindical Única de Trabajadores Campesinos de El Alto (FSUTCEA) y la Federación de Juntas Vecinales de El Alto (FEJUVE), jugaron un papel fundamental en la búsqueda de autonomía municipal.

En 1957, se creó el Consejo Central de Vecinos de El Alto de La Paz, que más tarde se convirtió en la Federación de Juntas Vecinales de El Alto, con el propósito de garantizar servicios básicos para la población y la creación de la Cuarta Sección de la provincia Murillo.

La autonomía de El Alto se logró con la Ley 728 promulgada en 1985, que estableció a El Alto como la Cuarta Sección municipal de la provincia Murillo con su capital. En la década de 1980, se crearon el escudo, la bandera y el himno de El Alto, y en 1988, se elevó oficialmente a la población de El Alto al rango de ciudad.

A lo largo del siglo XX, El Alto fue testigo de importantes eventos, desde la llegada de servicios de transporte hasta movimientos sociales y políticos significativos. La población de la ciudad experimentó un crecimiento constante y, para el siglo XXI, El Alto se ha convertido en un referente nacional en Bolivia, marcado por su historia de movilizaciones y luchas por los derechos de sus habitantes” (Fernández Rojas, 2023).

1.2.2 Misión

“El Alto ciudad emergente con alma y espíritu de unidad, con planificación del bienestar social, participativo, desconcentrado, incluyente que garantiza derechos, seguridad del espacio público, con acceso a la salud, educación y protagonista del desarrollo nacional” (Gobierno Autónomo Municipal de El Alto, 2023a).

1.2.3 Visión

“Gobierno Autónomo Municipal de El Alto, con identidad, ciudad ajayu con derecho a la salud, educación y sin violencia, con empleo, modernidad, industrias y emprendimientos. para el desarrollo sustentable y planificado, articulador de la región metropolitana” (Gobierno Autónomo Municipal de El Alto, 2023c).

1.2.4 Objetivo

“La gestión de recursos humanos es clave para el éxito de cualquier organización, y el Gobierno Autónomo Municipal de El Alto no es la excepción. Uno de los objetivos prioritarios de la institución es el registro, integración y actualización de información sobre el personal de ítem y eventual. Es por eso que la Unidad de Registros tiene como objetivo: “Registrar, integrar

y actualizar información referida al personal de Ítem y Eventual del Órgano Ejecutivo del Gobierno Autónomo Municipal de El Alto” (Gobierno Autónomo Municipal de El Alto, 2023b).

1.2.5 Organigrama

Para entender mejor la dinámica y la forma en que opera la Unidad de Registro, se presenta a continuación su organigrama, el cual detalla la posición en la que se encuentra y su responsabilidad en la institución.

Tal y como se observa en la Figura 1, se indica la posición de la Unidad de Registros en la estructura organizativa del Gobierno Autónomo Municipal de El Alto.

Figura 1

Organigrama de la Unidad de Registros



1.3 Antecedente Afines al Trabajo de Grado

A continuación, se indicarán algunos antecedentes locales, nacionales e internacionales que han sido de gran utilidad como guía. Estos antecedentes proporcionarán una visión amplia y valiosa para abordar los temas centrales del trabajo de grado.

1.3.1 Antecedentes Internacionales

- “Desarrollo de un Sistema Web de Control Documental Aplicado a una Empresa Automotriz en el cual se desarrolló una aplicación web capaz de gestionar y administrar la carga de documentación fiscal de proveedores (facturas: PDF, XML), para el control y seguimiento de los mismos, utilizando software libre, así como también la metodología de programación orientada a objetos en conjunto con el modelo de desarrollo MVC (Modelo – Vista - Controlador)” (Hernández Aguilar, 2018, pág. 3).
- “Desarrollo de un Sistema de Evaluación de Postulaciones al Magíster en Tecnologías de la Información. En el mencionado proyecto el objetivo general fue desarrollar un sistema Web que automatizara parte del flujo de trabajo asociado al procesamiento de postulaciones al Magíster en Tecnologías de la Información; programa que es impartido por la Escuela de Postgrado de la FCFM (Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile) a través del Departamento de Ciencias de la Computación” (Salas Vega, 2021, pág. 3).

1.3.2 Antecedentes Nacionales

- “Sistema Web para la Administración y Control de Registros Diarios para el Estudio Jurídico Villegas y Asociados. Este proyecto tuvo el objetivo de desarrollar un sistema web que contribuya con la administración y control de registros diarios en el estudio jurídico Villegas & Asociados, para que brinde información oportuna y actualizada tanto para abogados como para clientes” (Ruiz Oblitas, 2022, pág. 6).

- “Sistema Web de Gestión de Clientes, Registro de Pedidos y Envíos. El objetivo del proyecto fue desarrollar un sistema web para la gestión de clientes, registro de pedidos y envíos para la Florería ‘Dalia’” (Rivera Cruz, 2021, pág. 7).

1.3.3 Antecedentes Locales

- “Desarrollo de un Sistema Web de Control de Registro de Archivos Notariales. El proyecto tenía como finalidad implementar un sistema orientado a la web que permita controlar y almacenar los registros de archivos y hacer un seguimiento de la información almacenada en la notaría número quince de la ciudad de El Alto” (Ali Ticona, 2020, pág. 4).
- “Sistema de Información para el Registro y Seguimiento Académico. La finalidad del proyecto era desarrollar un sistema de información para el registro y seguimiento académico de estudiantes, para automatizar los procesos de inscripción de estudiantes, registro de docentes, avances académicos y calificaciones en la Academia de Formación Profesional “BELA”, que coadyuve en la mejor administración y dirección de la academia” (Calamani Salas, 2020, pág. 6).
- “Sistema de Información Web de Control Administrativo. El proyecto tenía como objetivo el desarrollo de un sistema de información web de control administrativo que genere la información del gimnasio, permitiendo organizar y reducir el tiempo en el control de manera confiable y eficaz para su mejor rendimiento en la administración” (Yahuasi Apaza, 2020, pág. 5).

1.4 Planteamiento del Problema

Las instituciones públicas hoy en día enfrentan la obligación de contar con sistemas de información eficientes, debido al manejo de una gran cantidad de datos y procesos. La falta de sistemas de gestión adecuados puede llevar a la desorganización, pérdida de información y dificultades en el acceso a datos críticos. Esto, a su vez, entorpece los procesos

administrativos, dificulta el seguimiento y control de la información necesaria para una administración efectiva y transparente. Además, la ausencia de un manejo adecuado de la información puede resultar en el incumplimiento de normativas legales, afectando la rendición de cuentas y la confianza en la gestión pública.

1.4.1 Problema Principal

En el Gobierno Autónomo Municipal de El Alto, la Unidad de Registros es responsable de recibir los documentos personales de los funcionarios, distribuidos en varias instancias, conformadas por unidades, direcciones, secretarías y más. Actualmente, esta unidad enfrenta varios problemas debido a la acumulación de una gran cantidad de documentación solicitada previa a la contratación, junto con sus copias de respaldo. Esto genera desorganización y dificulta el acceso a la información. Además, esta situación provoca la pérdida de documentos y dificulta el seguimiento y control de la información de los funcionarios. Para más detalles, consulte el Anexo B.

Según el sitio web oficial del Gobierno Autónomo Municipal de El Alto, sección de transparencia, apartado talento humano, la cantidad de funcionarios alcanza un total de 1538 personas, distribuidas en 238 instancias que incluyen despachos, secretarías, direcciones y otras unidades administrativas (Gobierno Autónomo Municipal de El Alto, 2024).

La desorganización de documentos personales, como títulos profesionales y libretas militares en formato impreso, generando duplicidad de información innecesaria debido a que se entrega fotocopias constantes de la misma información. Esto no solo dificulta la gestión interna y la elaboración de informes al complicar el acceso a la información de respaldo necesaria para decisiones efectivas, sino que también obstaculiza cualquier proceso de verificación de la documentación presentada, especialmente cuando se precisa rapidez y eficiencia.

1.4.2 Problemas Secundarios

- La acumulación excesiva y el entrepapelado de documentos, junto con la falta de un sistema de organización eficiente, y generando duplicidad de información de manera innecesaria.
- La revisión y el llenado manual de una gran cantidad de documentos incrementan la probabilidad de cometer errores en el registro y verificación de información, lo cual puede tener graves repercusiones en la precisión de los datos de los funcionarios y en la toma de decisiones basadas en la información proporcionada.
- La demora en el acceso a la información, debido a la carencia de un sistema de archivo eficiente, ocasiona demoras significativas en la obtención de datos cruciales, lo que afecta en la eficiencia de los procesos de contratación.
- La falta de una base de datos digital con copias de seguridad adecuadas pone en riesgo la información de perderse permanentemente en caso de extravío o daño de los archivos físicos, complicando la capacidad de verificar y autenticar la información presentada por los funcionarios.
- La desorganización documental y la falta de un sistema que guarde la digitalización de archivos personales, obstaculizan el acceso a la información crucial, esto se agrava si la información física se pierde, no se cuenta con un respaldo electrónico que permita recuperar esos datos.

1.5 Formulación del Problema

¿Cómo se puede centralizar y gestionar eficientemente la extensa y variada documentación personal de los funcionarios recibida en la Unidad de Registros, con el fin de optimizar el proceso de revisión y asegurar un acceso oportuno, preciso y confiable a la información mediante la digitalización?

1.6 Objetivos

1.6.1 Objetivo General

Establecer un Sistema de Información Web para centralizar, gestionar y controlar la documentación de los funcionarios, con el fin de optimizar el proceso de seguimiento previo a la contratación, garantizando la precisión y confiabilidad de la información y facilitando su acceso para mejorar la productividad institucional.

1.6.2 Objetivos Específicos

- Desarrollar un sistema eficiente de organización documental para funcionarios, reduciendo la acumulación y el entrapapelado mediante la digitalización y categorización.
- Implementar procesos automatizados de revisión y llenado de información para minimizar errores en el registro y verificación de datos, mejorando la precisión y evitando repercusiones graves en la toma de decisiones, además, generar una certificación de ficha Kardex que resuma la información registrada en el sistema.
- Incorporar un sistema de archivo digital eficiente para almacenar y recuperar documentos importantes, como archivos PDF e imágenes, agilizando los procesos de contratación.
- Integrar una base de datos NoSQL para gestionar eficientemente la documentación de respaldo, garantizando su integridad, disponibilidad a largo plazo y accesibilidad, con el fin de asegurar registros precisos y confiables, facilitando futuras verificaciones.
- Desplegar una plataforma web personalizada que facilite el registro y la visualización intuitiva de información relevante sobre funcionarios, promoviendo una gestión eficiente y accesible de los datos.

1.7 Justificación

1.7.1 Justificación Técnica

El Gobierno Autónomo Municipal de El Alto se encuentra equipado con una unidad de sistemas que dispone de servidores, equipos de cómputo y acceso a la red de internet.

La Unidad de Registros, como parte integral de esta institución, cuenta con la capacidad de solicitar alojamiento web en los servidores de la entidad. Este alojamiento web permitirá el almacenamiento de los archivos necesarios para la aplicación web propuesta en este proyecto. Además, se verificó que la Unidad de Registros dispone de computadoras conectadas a una red local y acceso a internet. Esto posibilita que los funcionarios puedan acceder al sistema a través de un navegador web y llevar a cabo la verificación de los documentos ingresados al sistema de manera eficiente y oportuna.

1.7.2 Justificación Económica

Actualmente debido a la falta de un sistema de información web que permita centralizar la información, se recurre a respaldos mediante fotocopias, haciendo uso de materiales de escritorio, archivadores donde alojan los archivos personales, formularios en formato Word para el llenado de datos personales, lo que resulta en la acumulación de muchas hojas y estantes donde almacenar toda esta información, generando gastos económicos en el transcurso del tiempo.

El proyecto propuesto se enfoca en el uso de herramientas de software libre para evitar costos asociados a licencias de software pagado. Esta elección no solo busca ahorrar recursos financieros, sino también promover la accesibilidad y la colaboración en el desarrollo tecnológico. El software libre fomenta una comunidad activa de usuarios y desarrolladores, garantizando flexibilidad y soporte técnico gratuito.

La ausencia de un sistema centralizado para la gestión de la documentación del personal, genera duplicidad de información de manera continua en formato de fotocopias.

Además, la falta de un sistema eficiente aumenta el riesgo de pérdida de documentos, lo que podría resultar en la necesidad de dedicar recursos adicionales para recuperar o reemplazar la información extraviada.

1.7.3 Justificación Social

El sistema permitirá obtener información requerida de forma rápida y segura. Facilitará a los trabajadores de la Unidad de Registros la verificación de la documentación de una manera más eficiente, permitiéndoles interactuar con la documentación de una forma amigable y rápida.

Las personas que quieran trabajar en el Gobierno Autónomo Municipal de El Alto podrán acceder desde un navegador web para registrar sus datos personales, académicos, laborales y familiares tal como se les solicita previa contratación, asimismo podrán hacer seguimiento de la información que ingresaron en el sistema. Este sistema permitirá la centralización de la información, realizando un solo registro por funcionario y de esta manera evitando la duplicidad de la información y generando ahorros en gastos innecesarios.

En los dos casos puntuales se pretende agilizar el proceso, evitar demoras para ambas partes y evitar filas, así como también prevenir una mala atención que se pudiese ocasionar al momento de la revisión de la misma generando colas, pérdida de tiempo y solucionar inconvenientes como ser él extravió de documentación presentada o la excesiva acumulación de documentos, lo cual conlleva a una mala organización.

1.8 Metodología

Una metodología es un conjunto de prácticas, técnicas, procedimientos y reglas que establecen un marco de trabajo que define cómo se planificarán, ejecutarán, controlarán y finalizarán las actividades del proyecto. Además, proporciona un enfoque estructurado y sistemático para abordar los desafíos y riesgos inherentes al proyecto, facilitando así la

colaboración eficiente entre los miembros del equipo y asegurando la calidad y culminación del proyecto.

1.8.1 Metodología de Desarrollo

Se abarcará el proyecto con OpenUP que es una metodología de desarrollo de software ágil y ligera que se enfoca en la colaboración en equipo, la entrega iterativa e incremental y la retroalimentación continua del equipo. OpenUP es una variante del Proceso Unificado de Rational (RUP), que se enfoca en proporcionar una base sólida para el desarrollo iterativo y la mejora continua del proceso de desarrollo de software.

"Es un modelo de desarrollo de software, es parte del Framework de modelo de proceso de Eclipse (Eclipse Process Framework), desarrollado por la fundación Eclipse. Mantiene las características esenciales de Rational Unified Process (RUP), las cuales son: Desarrollo incremental, uso de casos de uso y escenarios, manejo de riesgos y diseño basado en la arquitectura" (Gimson, 2015).

Las fases de OpenUP son las siguientes:

- Inicio.
- Elaboración.
- Construcción.
- Transición.

1.8.2 Estimación del Costo

Para la estimación de costo se utilizará el modelo COCOMO II (Constructive Cost Model II) es un modelo de estimación de costos de software, que se utiliza para estimar los recursos necesarios para desarrollar un proyecto de software.

"El modelo COCOMO II es más completo en comparación a COCOMO I. COCOMO I fue diseñado para software desarrollado usando lenguajes y construcciones de procedimiento, mientras que en los escenarios actuales la mayoría de los lenguajes y software se desarrollan

usando paradigmas orientados a objetos para los cuales COCOMO II es más apropiado" (SOOluciona, s. f.).

1.8.3 Métricas de Calidad

Para el presente proyecto se utilizará la métrica de calidad FURPS, es un enfoque de evaluación de la calidad de software que se centra en cinco características clave:

- Funcionalidad.
- Usabilidad.
- Confiabilidad.
- Rendimiento.
- Soporte.

“El modelo FURPS ha sido utilizado para el diseño y validación de interfaces para los usuarios finales, evaluando su funcionalidad, usabilidad, confiabilidad, desempeño y soporte, para tener como salida final un producto que cumpla las reglas de negocio” (modelosred, s. f.).

1.8.4 Seguridad

“La seguridad de las aplicaciones web es la práctica de proteger los sitios web, las aplicaciones y las API contra los ataques. Es una disciplina amplia, pero sus objetivos en última instancia son mantener el buen funcionamiento de las aplicaciones web y proteger a las empresas del vandalismo cibernético, el robo de datos, la competencia poco ética y otras consecuencias negativas” (CLOUDFLARE, 2023).

La seguridad es un aspecto crítico en cualquier proyecto de software. En este proyecto, se implementarán medidas de seguridad a nivel de software para garantizar la protección de datos y sistemas. Se establecerán reglas de seguridad, incluyendo políticas de creación de datos, para controlar el acceso y asegurar la integridad de la información.

Ya que el activo más importante que tiene la organización es la propia información, por lo que tienen que existir técnicas que las mantengan seguras, mucho más allá de la seguridad física que se puede establecer gracias a los equipos con los que cuenta la organización para almacenar dicha información.

1.8.5 Pruebas de Software

Caja Blanca. “La prueba de caja blanca se conoce a veces como «prueba de caja de cristal», y se refiere a un proceso de prueba en el que el probador tiene acceso completo a toda la información que hay detrás del software. Esto incluye el acceso al código fuente y a los documentos de diseño y las instrucciones para el cliente del paquete” (ZAPTEST, 2023b).

Estas pruebas se enfocan principalmente en la estructura interna del código, omitiendo la entrada o salida. El objetivo principal es probar la lógica del programa principalmente el algoritmo que se diseña. Estas pruebas de caja blanca son importantes, porque con ellas se obtiene la reducción y disminución en un gran porcentaje de los errores que puedan presentar el software y mejorando la calidad y confiabilidad en la codificación.

Caja Negra. También suelen ser llamadas funcionales y basadas en especificaciones. Estas pruebas pretenden examinar el programa en busca de que cuente con las funcionalidades que debe tener y como lleva a cabo las mismas, analizando siempre los resultados que devuelve y probando todas las entradas en sus valores válidos e inválidos.

“Las pruebas de caja negra consisten en probar un sistema o programa informático sin tener conocimiento previo de su funcionamiento interno. Esto no sólo se refiere a no conocer el código fuente en sí, sino que implica no haber visto ninguna de las documentaciones de diseño que rodean al software. Los probadores se limitan a dar entrada y recibir salida como lo haría un usuario final” (ZAPTEST, 2023b).

1.8.6 Pruebas de Estrés

“Las pruebas de estrés de software también conocidas como pruebas de resistencia son un recurso utilizado durante la etapa de Testing, con el objetivo de probar los límites de un sistema y prever escenarios de riesgo ante cargas extremas” (Testing iT, 2022).

Para este tipo de pruebas se necesita de herramientas, porque consiste en simular las peticiones realizadas por usuarios como puede llegar a ocurrir en eventos de promociones o múltiples solicitudes y ver como el sistema reacciona ante estos casos.

1.8.7 Métodos para la recolección de información

En la fase de desarrollo de un sistema de información, la recopilación de datos es esencial para entender en profundidad los requisitos y necesidades de los usuarios. Para lograrlo, se emplearán diversos métodos de recolección de información. Estos métodos desempeñarán un papel fundamental al obtener datos valiosos que orientarán el diseño y funcionamiento del sistema. A continuación, se detallan los métodos que serán utilizados con el propósito de alcanzar una comprensión completa y precisa de los procesos y requerimientos.

- **Encuestas:** Usadas para obtener información de los funcionarios.
- **Entrevistas:** Constantes para identificar necesidades y requisitos.
- **Observaciones:** Para entender cómo se gestionan los documentos.
- **Revisión de registros:** Examina documentos existentes.

1.9 Herramientas

Para el desarrollo del sistema de información web propuesto, se empleará una laptop en cuanto al hardware y los recursos a utilizar en cuanto a software serán: JavaScript como lenguaje de programación, React como framework para el frontend que es una biblioteca de JavaScript, NodeJS para interpretar código JavaScript del lado del servidor, Express que es un Framework de aplicaciones web de código abierto y gratuito para Node.js y MongoDB que

es un sistema de gestión de bases de datos NoSQL (No Relacional) de código abierto y gratuito.

1.9.1 Hardware

Para el desarrollo de la aplicación web y alojar las herramientas de software se utilizará una laptop, el equipo de cómputo encargado de alojar las herramientas de software:

Laptop: Es una computadora portátil de peso y tamaño ligero, su tamaño es aproximado al de un portafolio. La laptop con la que se desarrollará el siguiente proyecto tiene las siguientes características.

- Procesador: Procesador Intel® Core™ i5-10300H (caché de 8 M, hasta 4,50 GHz).
- Memoria RAM: 8 GB.
- Tipo de memoria RAM: DDR4-SDRAM.
- Almacenamiento: 500 GB SSD.
- Tamaño de pantalla: 39,6 cm (15.6 pulgadas).
- Resolución de pantalla: 1920 x 1080 Píxeles.
- Adaptador de gráficos independientes: NVIDIA® GeForce® GTX 1650 Ti.
- Memoria de la tarjeta gráfica: 4 GB.
- Sistema operativo: S.O. Windows 10 Home.

1.9.2 Arquitectura de Software por Capas

La arquitectura por capas se enfoca en el diseño de un software que se organiza en distintos niveles y diferentes funcionalidades, donde cada capa es responsable de un conjunto de tareas específicas y se comunica con niveles adyacentes que estén bien definidas.

"La arquitectura en capas es un modelo de diseño de software, cuya base es la separación de las diferentes funcionalidades del sistema en capas o niveles, donde cada capa se encarga de un conjunto de tareas específicas y se comunica con los niveles adyacentes mediante interfaces bien definidas" (HubSpot, 2021).

1.9.3 *Tecnologías Frontend o del Lado del Cliente*

En la clasificación de "Tecnologías Frontend" se engloban aquellas herramientas y lenguajes de programación que se enfocan en la creación de las interfaces de usuario y en el establecimiento de la comunicación con el servidor en el contexto del desarrollo web. Estas tecnologías desempeñan un papel fundamental en la forma en que los usuarios interactúan con las aplicaciones y los sitios web. Algunas de las tecnologías que emplearemos en este ámbito incluyen:

- **JavaScript:** "JavaScript es un lenguaje de programación que los desarrolladores utilizan para hacer páginas web interactivas. Desde actualizar fuentes de redes sociales a mostrar animaciones y mapas interactivos, las funciones de JavaScript pueden mejorar la experiencia del usuario de un sitio web. Como lenguaje de scripting del lado del servidor, se trata de una de las principales tecnologías de la World Wide Web. Por ejemplo, al navegar por Internet, en cualquier momento en el que vea un carrusel de imágenes, un menú desplegable "click-to-show" (clic para mostrar), o cambien de manera dinámica los elementos de color en una página web, estará viendo los efectos de JavaScript" (AWS, 2023).
- **React:** Es una popular biblioteca de JavaScript utilizada para construir interfaces de usuario interactivas y dinámicas en aplicaciones web. Se destaca por su eficiencia y facilidad de uso, permitiendo a los desarrolladores crear componentes reutilizables que responden de manera rápida a las interacciones del usuario, lo que mejora la experiencia de navegación en línea.

"ReactJS es una de las librerías más populares de JavaScript para el desarrollo de aplicaciones móviles y web. Creada por Facebook, React contiene una colección de fragmentos de código JavaScript reutilizables utilizados para crear interfaces de usuario (UI) llamadas componentes" (HOSTINGER TUTORIALES, 2023a).

- **Tailwind CSS:** Es un Framework de diseño CSS utilitario que permite a los desarrolladores crear interfaces de usuario personalizadas de manera rápida y eficiente. Es altamente personalizable y ofrece un conjunto de clases CSS predefinidas que se pueden utilizar para aplicar estilos y diseños de manera coherente en toda la página web.

"Tailwind, en pocas palabras, es un framework CSS que da prioridad a la utilidad sobre el propio estilo, pero además a diferencia de otros frameworks CSS como Bootstrap o Bulma, Tailwind no provee una serie de componentes predefinidos. En su lugar, este framework opera en un nivel inferior y te proporciona un conjunto de clases de ayuda para estructura y estilado, de forma que, usando dichas clases, puedas crear rápidamente diseños personalizados con facilidad" (OpenWebinars, 2023b).

1.9.4 Tecnologías Backend o del Lado del Servidor

Las Tecnologías Backend son aquellas que se utilizan para desarrollar el lado del servidor de una aplicación web. Estas tecnologías se encargan de procesar y almacenar los datos, gestionar la lógica de negocio y manejar las solicitudes y respuestas del servidor. Para tal propósito haremos uso de las siguientes tecnologías.

- **Node.js:** Es un entorno de tiempo de ejecución de JavaScript que permite a los desarrolladores usar JavaScript tanto en el lado del cliente (navegador web) como en el lado del servidor. Antes de la aparición de Node.js, el uso de JavaScript se limitaba al cliente y los desarrolladores tenían que usar otros lenguajes en el lado del servidor, lo que aumentaba la complejidad de las aplicaciones.

"Node.js fue creado por los desarrolladores originales de JavaScript. Lo transformaron de algo que solo podía ejecutarse en el navegador en algo que se podría ejecutar en los ordenadores como si de aplicaciones independientes se tratara. Gracias a Node.js se puede ir un paso más allá en la programación con JavaScript no solo creando sitios web interactivos,

sino teniendo la capacidad de hacer cosas que otros lenguajes de secuencia de comandos como Python pueden crear" (OpenWebinars, 2023a).

- **Express:** Express es un popular framework para Node.js que se utiliza para crear aplicaciones web del lado del servidor. Permite crear servidores web rápidos y escalables utilizando el lenguaje de programación JavaScript, que también se utiliza en el lado del cliente.

"Expressjs es un framework rápido, minimalista y flexible de Node.js. Permite crear APIs y aplicaciones web fácilmente, provee un conjunto de características como manejo de rutas (direccionamiento), archivos estáticos, uso de motor de plantillas, integración con bases de datos, manejo de errores, middlewares entre otras" (Medium, 2023).

1.10 Límites y Alcances

1.10.1 Límites

El sistema de información web que tendrá las siguientes limitaciones:

- El sistema web estará disponible para funcionarios del Gobierno Autónomo Municipal de El Alto y para la población en general que desee trabajar en la institución, siempre y cuando hayan completado previamente su registro en el sistema.
- Solo almacenará información personal de los funcionarios que están postulando a un contrato en la institución o los que ya cuenten con un contrato.
- La gestión de la información solo la podrán realizar funcionarios que conforman la Unidad de Registros.
- Los documentos de respaldo que se subirán a la plataforma serán únicamente en formato PDF.
- Los permisos para editar formularios ya llenados solo serán otorgados por funcionarios de la Unidad de Registros.

- Los encargados de evaluar la información ingresada en el sistema serán los funcionarios de la Unidad de Registros.

1.10.2 Alcances

El sistema de información web que se pretende desarrollar estará basado en los requerimientos obtenidos inicialmente. Los módulos que se pretenden cubrir son:

- a) Módulo de autenticación y control de las credenciales de acceso (usuario/contraseña).
- b) Módulo de revisión:
 - Visualización de la información de un funcionario en particular.
 - Habilitación para la edición de formularios.
 - Formulario de revisión de la documentación personal.
 - Búsqueda de información del personal.
- c) Módulo de registro.
 - Formulario de datos personales.
 - Formulario de datos laborales.
 - Formulario de licencia de conducir y registro de movilidades.
 - Formulario de datos laborales.
 - Formulario de registros académicos:
 - Formulario de experiencias laborales.
 - Formulario familiar y dependientes.
 - Subida de documentos previo contrato.
 - Formulario de discapacidad (Titular / Tutor).
- d) Módulo de certificación.
 - Certificación de los datos y la documentación ingresada.
- e) Módulo de generación de reportes.

- Reporte de la documentación ingresada de un funcionario mediante búsqueda.
- Reporte de las revisiones que se realizar a la documentación de un funcionario en particular.

1.11 Aportes

Este sistema representa una contribución significativa para la gestión de la documentación presentada por los funcionarios, eliminando la pérdida de archivos personales y centralizando todos los documentos en una base de datos.

Se logró proporcionar un acceso más rápido a la información, facilitando a los funcionarios que necesitan realizar descargos con la documentación presentada. La implementación de este proyecto mejora notablemente las condiciones de trabajo en la Unidad de Registros, permitiendo una recepción más eficiente de los documentos.

Al ser una aplicación web, los funcionarios pueden acceder desde cualquier dispositivo, utilizando sus credenciales personales para ingresar a su información y verificar los datos que ingresaron en el sistema. También se reduce considerablemente el tiempo necesario para dar seguimiento a los documentos personales presentados y para controlar la información que se ingresa en el sistema.



CAPITULO II

MARCO TEÓRICO



**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO

2. Capítulo 2: Marco Teórico

2.1 Introducción

Este capítulo tiene como propósito establecer las bases teóricas necesarias para la comprensión y desarrollo del proyecto. Se describirán los conceptos fundamentales y tecnologías que servirán como base para la implementación del sistema. Si bien no se pretende ofrecer una exposición exhaustiva de todas las metodologías, técnicas y herramientas que se utilizarán en el proyecto, se proporcionará una sólida base para facilitar su comprensión.

La construcción de un proyecto exitoso requiere una comprensión sólida de los conceptos y las tecnologías relevantes. En este contexto, se explorarán temas relacionados con el sistema web, sistemas de información, metodologías, conceptos de software, registros personales, información personal entre otros, los cuales son esenciales para la interpretación de la problemática, elaboración del sistema y su posterior implementación.

El marco teórico, por lo tanto, constituye un pilar fundamental que guiará el desarrollo del proyecto, proporcionando una base sólida para el entendimiento y la aplicación de los conceptos, técnicas y herramientas que se ponen en práctica a lo largo de la elaboración del presente proyecto.

2.2 Conceptos Fundamentales

2.2.1 Sistema

“Un sistema es un conjunto ordenado de componentes relacionados entre sí, ya se trate de elementos materiales o conceptuales, dotado de una estructura, una composición y un entorno particular. Se trata de un término que aplica a diversas áreas del saber, como la física, la biología y la informática o computación” (concepto, 2023b).

En el contexto de este proyecto de grado, se concibe un sistema como una estructura organizada que integra componentes interconectados para alcanzar un objetivo específico de manera eficiente. Este enfoque sistemático permite coordinar y gestionar recursos y procesos de manera conjunta, optimizando así la ejecución de tareas y la toma de decisiones para alcanzar los objetivos.

“Un sistema es un conjunto de elementos relacionados entre sí que funciona como un todo. Si bien cada uno de los elementos de un sistema puede funcionar de manera independiente, siempre formará parte de una estructura mayor. Del mismo modo, un sistema puede ser, a su vez, un componente de otro sistema.

La palabra sistema procede del latín *systema*, y este del griego σύστημα (*systema*), identificado en español como ‘unión de cosas de manera organizada’. De esta palabra se derivan otras como *antisistema* o *ecosistema*.” (Significados, 2023).

2.2.2 Dato

"Un dato es la representación de una variable que puede ser cuantitativa o cualitativa que indica un valor que se le asigna a las cosas y se representa a través de una secuencia de símbolos, números o letras. Los datos describen hechos empíricos. Para examinarlos deben ser organizados o tabulados, ya que un dato por sí mismo no puede demostrar demasiado, sino que se debe evaluar el conjunto para examinar los resultados" (concepto, 2023a).

"Un dato es un conjunto discreto, de factores objetivos sobre un hecho real. Dentro de un contexto empresarial, el concepto de dato es definido como un registro de transacciones. Un dato no dice nada sobre el porqué de las cosas, y por sí mismo tiene poca o ninguna relevancia o propósito. Las organizaciones actuales normalmente almacenan datos mediante el uso de tecnologías. Desde un punto de vista cuantitativo, las empresas evalúan la gestión de los datos en términos de coste, velocidad y capacidad.

Todas las organizaciones necesitan datos y algunos sectores son totalmente dependientes de ellos: bancos, compañías de seguros, agencias gubernamentales y la seguridad social son ejemplos obvios. En este tipo de organizaciones la buena gestión de los datos es esencial para su funcionamiento, ya que operan con millones de transacciones diarias. Pero en general, para la mayoría de las empresas tener muchos datos no siempre es bueno. Las organizaciones almacenan datos sin sentido. Realmente esta actitud no tiene sentido por dos razones. La primera es que demasiados datos hacen más complicado identificar aquellos que son relevantes. Segundo, y todavía más importante, es que los datos no tienen significado en sí mismos. Los datos describen únicamente una parte de lo que pasa en la realidad y no proporcionan juicios de valor o interpretaciones, y por lo tanto no son orientativos para la acción. La toma de decisiones se basará en datos, pero estos nunca dirán que se debe hacer. Los datos no dicen nada acerca de lo que es importante o no. A pesar de todo, los datos son importantes para las organizaciones, ya que son la base para la creación de información." (Carrión, s. f., pág. 1).

2.2.3 Información

La información es conocimiento o datos que se han organizado de manera que sean comprensibles y útiles. Es una herramienta que nos ayuda a entender el mundo, tomar decisiones y comunicarnos con los demás. La información puede ser escrita, hablada, visual o en cualquier otra forma, y se utiliza en una amplia variedad de campos, desde la ciencia y la tecnología hasta la educación y la comunicación. En resumen, la información es lo que nos permite aprender, crecer y conectarnos con el mundo que nos rodea.

“La información es capaz de cambiar la forma en que el receptor percibe algo, es capaz de impactar sobre sus juicios de valor y comportamientos. Tiene que informar; son datos que marcan la diferencia. La palabra ‘informar’ significa originalmente “dar forma a” y la información es capaz de formar a las personas.

A diferencia de los datos, la información tiene significado (relevancia y propósito). No sólo puede formar potencialmente al que la recibe, sino que está organizada para algún propósito. Los datos se convierten en información cuando su creador les añade significado.

Las computadoras nos pueden ayudar a añadir valor y transformar datos en información, pero es muy difícil que nos puedan ayudar a analizar el contexto de dicha información. Un problema muy común es confundir la información (o el conocimiento) con la tecnología que la soporta. Muchas veces se comenta que tener un teléfono no garantiza mantener conversaciones brillantes. En definitiva, que actualmente tengamos acceso a más tecnologías de la información no implica que hayamos mejorado nuestro nivel de información” (Carrión, s. f., pág. 1-2).

2.3 Sistema de Información Web

“Diferentes autores reconocen la web como un sistema de información que va más allá de un simple servidor informático, ya que interpretan la web como un depósito documental que funciona a modo de memoria documental, con tareas que superan la mera publicación para convertirse en un sistema de organización de la información. El concepto repositorio de información marca una forma de almacenamiento que va más allá del concepto base. Partiendo de esta concepción de la web como sistema de información y asumiendo a las administraciones públicas como entes difusores de información, se estudia la web como una entidad de información que permite acercar al ciudadano parte de los contenidos que las administraciones ofrecen tradicionalmente por otros medios” (Muñoz Cañavate & Chain Navarro, 1999).

"Un sistema de información (SI) es un conjunto de elementos interrelacionados con el propósito de prestar atención a las demandas de información de una organización, para elevar el nivel de conocimientos que permitan un mejor apoyo a la toma de decisiones y desarrollo de acciones" (Peña Ayala, 2006).

2.3.1 Elementos de los Sistemas de Información:

Según el mismo libro de Peña Ayala (2006, pág), los componentes más importantes de un sistema de información son:

- **Financieros.** Es el aspecto económico que permite la adquisición, contratación y mantenimiento de los demás recursos que integran un sistema de información.
- **Administrativos.** Es la estructura orgánica de objetivos, lineamientos, funciones, procedimientos, departamentalización, dirección y control de las actividades; que sustenta la creación y uso de los sistemas.
- **Humanos.** Está compuesto por dos grupos:
 - El técnico, que posee los conocimientos especializados en el desarrollo de sistemas, siendo estos los: Administradores, Líderes de Proyecto, Analistas, Programadores, Operadores y Capturistas.
 - El usuario, representado por las personas interesadas en el manejo de información vía cómputo, como apoyo al mejor desempeño de sus actividades, siendo estos los: Funcionarios, Contadores, Ingenieros, Empleados, Público, etc.
- **Materiales.** Son aquellos elementos físicos que soportan el funcionamiento de un sistema de información, por ejemplo: local de trabajo, instalaciones eléctricas y de aire acondicionado, medios de comunicación, mobiliario, maquinaria, papelería, etc.
- **Tecnológicos.** Es el conjunto de conocimientos, experiencias, metodologías y técnicas; que orientan la creación, operación y mantenimiento de un sistema.

2.3.2 Ciclo de Desarrollo

“La construcción de un sistema de información implica la conjugación de esfuerzos, conocimientos, experiencias, recursos y tiempo muy valiosos. Para garantizar el éxito del proyecto y emplear al máximo los elementos disponibles, es necesario contar con un adecuado

rumbo de acción. Esto se logra mediante la utilización de una metodología que establezca las etapas con objetivos, actividades y técnicas necesarias en la creación de un sistema.

La ejecución de las etapas sigue un orden cronológico, en el cual los resultados producidos por una etapa alimentan las funciones de la siguiente. Además, se aplican los conceptos del proceso administrativo para regular las acciones de cada etapa y, por lo tanto, de todo el proyecto. Por lo tanto, es esencial planificar y controlar la realización de las actividades” (Peña Ayala, 2006).

2.3.2.1 Etapas de Desarrollo. El papel que desempeñan las etapas que conforman el ciclo de desarrollo de los sistemas de información se describe de la siguiente manera:

- **Análisis:** Define los requerimientos de información y la forma más adecuada de atenderlos.
- **Diseño:** Describe la estructura, funciones e interrelaciones de los componentes del sistema.
- **Programación:** Elabora los elementos del sistema, como programas, archivos, reportes, etc.
- **Implantación:** Prueba e instala el sistema construido.
- **Operación:** Hace uso del sistema.
- **Mantenimiento:** Modifica el sistema en función de los nuevos requerimientos que se van presentando.

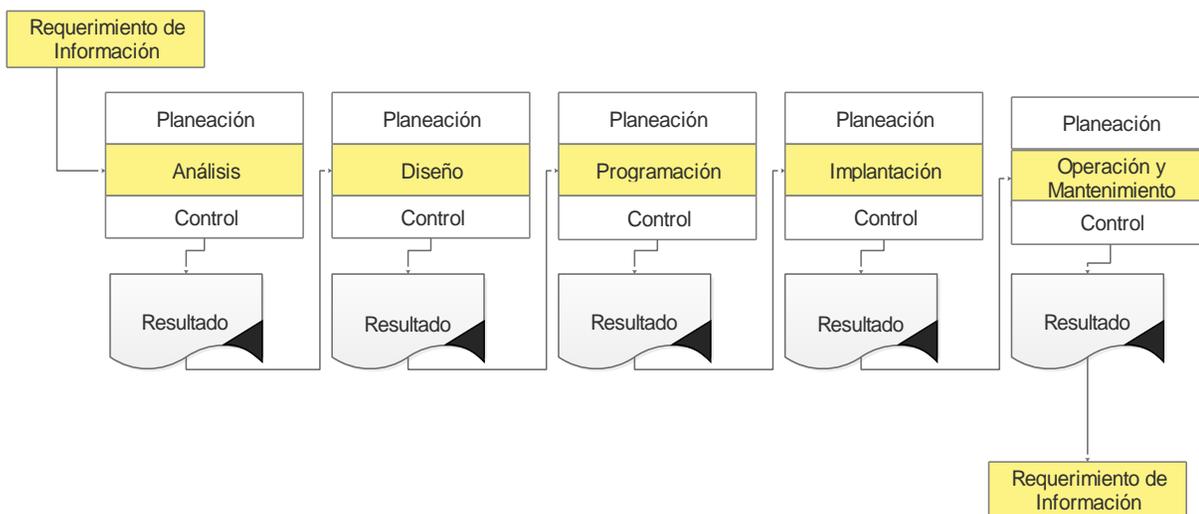
En cuanto a la participación del proceso administrativo:

- **Planeación:** Se establecen los objetivos, estrategias, programas de acción y determinación de recursos.
- **Control:** Se evalúan los resultados conforme a las metas esperadas y se realizan las correcciones necesarias, influenciando la ejecución de la siguiente etapa.
- **Importancia de los Sistemas de Información:**

- **Toma de Decisiones:** Los sistemas de información proporcionan datos y análisis que respaldan la toma de decisiones efectivas en empresas, gobierno y otras organizaciones, lo que permite abordar problemas, identificar oportunidades y tomar medidas informadas.
- **Eficiencia Operativa:** Automatizan tareas y procesos, mejorando la eficiencia y reduciendo costos en las operaciones de las organizaciones. Esto aumenta la productividad y optimiza los recursos.
- **Acceso a la Información:** Los sistemas de información permiten un acceso rápido y oportuno a la información relevante, lo que es esencial para la toma de decisiones, la resolución de problemas y la colaboración efectiva. Esto mejora la eficacia y la capacidad de respuesta de una organización.
- **Comunicación y Colaboración:** Facilitan la comunicación y la colaboración entre individuos y equipos a través de herramientas como el correo electrónico, la mensajería instantánea y las redes sociales empresariales. Esto promueve la cooperación y el intercambio de conocimientos en el entorno laboral.

Figura 2

Proceso de Desarrollo de Sistemas de Información



Nota: Adaptado de Sistemas de Información (p. 12), A. Peña, 2006, Revillagigedo 83.

2.4 Metodología de Desarrollo de Software

Una metodología de desarrollo de software es un enfoque sistemático y estructurado para planificar, diseñar, implementar y mantener proyectos de software. Estas metodologías establecen un conjunto de procesos, prácticas y directrices que ayudan a los equipos de desarrollo a gestionar eficazmente el ciclo de vida del software. En este contexto, exploraremos en detalle la metodología de desarrollo de software OpenUP (Proceso Unificado Ágil) y su relevancia en el contexto de nuestro proyecto.

2.4.1 Metodología de Desarrollo de Software OpenUP

“OpenUP es un método y un proceso de desarrollo de software propuesto por un conjunto de empresas de tecnología, quienes lo donaron en el año 2007 a la Fundación Eclipse. La fundación lo ha publicado bajo una licencia libre y lo mantiene como método de ejemplo dentro del proyecto Eclipse Process Framework” (Wikipedia, 2023).

“OpenUP es un Proceso Unificado que aplica enfoques iterativos e incrementales dentro de un ciclo de vida estructurado, utiliza una filosofía ágil que se enfoca en la naturaleza de colaboración en el desarrollo de software. Es una herramienta agnóstica que puede extenderse para hacer frente a una amplia variedad de proyectos. Está basado en casos de uso, la gestión del riesgo, y una arquitectura centrada a impulsar el desarrollo” (Ríos Salgado et al., 2013).

2.4.1.1 Principios del OpenUP. A continuación, presentaré los principios de OpenUP, una metodología de desarrollo de software. Estos principios, como se han documentado en Wikipedia, son fundamentales para guiar el proceso de desarrollo. A continuación, se detallan los principios clave del OpenUP:

- **Colaborar para sincronizar intereses y compartir conocimiento:** Este principio promueve prácticas que impulsan un ambiente de equipo saludable, facilitan la colaboración y desarrollan un conocimiento compartido del proyecto.
- **Equilibrar las prioridades para maximizar el beneficio obtenido por los interesados en el proyecto:** Este principio promueve prácticas que permiten a los participantes de los proyectos desarrollar una solución que maximice los beneficios obtenidos por los participantes y que cumple con los requisitos y restricciones del proyecto.
- **Centrarse en la arquitectura de forma temprana para minimizar el riesgo y organizar el desarrollo:** Priorizar arquitectura temprana reduce riesgos y estructura desarrollo eficientemente.
- **Desarrollo evolutivo para llevar a cabo retroalimentación y una mejora continua:** Este principio promueve prácticas que permiten a los equipos de desarrollo obtener retroalimentación temprana y continua de los participantes del proyecto, permitiendo demostrarles incrementos progresivos en la funcionalidad.

2.4.1.2 Ciclo de Vida y Procesos de OpenUP. El desarrollo de software se rige por un conjunto de procesos y etapas que guían la creación de soluciones tecnológicas. En este contexto, OpenUP (Proceso Unificado Abierto) emerge como una metodología que aboga por un enfoque iterativo y colaborativo en la construcción de proyectos de software. Es fundamental comprender la estructura general del ciclo de vida y cómo se integra con los principios ágiles. Explorar el ciclo de vida y procesos de OpenUP, detallando cómo esta metodología se convierte en una guía efectiva para el desarrollo de software ágil y exitoso.

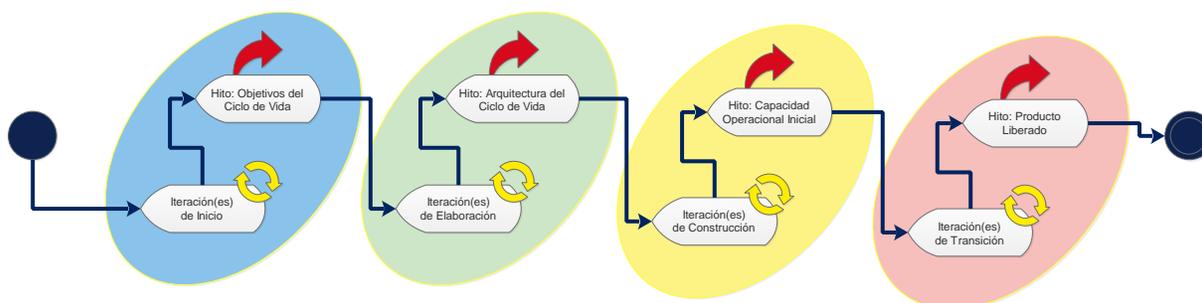
Según (Ríos Salgado et al., 2013), proporcionan una definición del ciclo de vida de OpenUP que engloba los siguientes conceptos clave:

El ciclo de vida de un proyecto, según la metodología OpenUP, permite que los integrantes del equipo de desarrollo aporten con micro-incrementos, que pueden ser el resultado del trabajo de unas pocas horas o unos pocos días. El progreso se puede visualizar diariamente, ya que la aplicación va evolucionando en función de estos micro-incrementos.

El objetivo de OpenUP es ayudar al equipo de desarrollo, a lo largo de todo el ciclo de vida de las iteraciones, para que sea capaz de añadir valor de negocio a los clientes, de una forma predecible, con la entrega de un software operativo y funcional al final de cada iteración. El ciclo de vida del proyecto provee a los clientes de: una visión del proyecto, transparencia y los medios para que controlen la financiación, el riesgo, el ámbito, el valor de retorno esperado, y otros aspectos relevantes.

Todo proyecto en OpenUP consta de cuatro fases: inicio, elaboración, construcción y transición. Cada una de estas fases se divide a su vez en iteraciones. (párr. 1-3)

En la figura 3 se muestran estas fases y su relación. A continuación, se presenta una descripción detallada de cada fase del ciclo de vida de OpenUP:

Figura 3*Ciclo de Vida de OpenUP*

Nota: Adaptado de Aplicación de la Metodología OpenUp (p. 3), S. Salgado, C. Hinojosa & R. Delgado, 2013, Documento del proyecto de graduación, Escuela Politécnica del Ejército.

- **Fase de inicio:** En esta fase, las necesidades de cada participante del proyecto son tomadas en cuenta y plasmadas en objetivos del proyecto. Se definen para el proyecto: el ámbito, los límites, el criterio de aceptación, los casos de uso críticos, una estimación inicial del coste y un boceto de la planificación.
- **Fase de elaboración:** En esta fase se realizan tareas de análisis del dominio y definición de la arquitectura del sistema. Se debe elaborar un plan de proyecto, estableciendo unos requisitos y una arquitectura estables. Por otro lado, el proceso de desarrollo, las herramientas, la infraestructura a utilizar y el entorno de desarrollo también se especifican en detalle en esta fase. Al final de la fase se debe tener una definición clara y precisa de los casos de uso, los actores, la arquitectura del sistema y un prototipo ejecutable de la misma.
- **Fase de construcción:** Todos los componentes y funcionalidades del sistema que falten por implementar son realizados, probados e integrados en esta fase. Los resultados obtenidos en forma de incrementos ejecutables deben ser desarrollados de la forma más rápida posible sin dejar de lado la calidad de lo desarrollado.

- **Fase de transición:** Esta fase corresponde a la introducción del producto en la comunidad de usuarios, cuando el producto está lo suficientemente maduro. La fase de la transición consta de las subfases de pruebas de versiones beta, pilotaje y capacitación de los usuarios finales y de los encargados del mantenimiento del sistema. En función de la respuesta obtenida por los usuarios puede ser necesario realizar cambios en las entregas finales o implementar alguna funcionalidad más.

2.4.1.3 Ventajas. La metodología OpenUP ofrece varias ventajas que hacen que sea una opción atractiva para abordar proyectos de desarrollo de software. A continuación, se destacará algunas de las principales ventajas de OpenUP:

- **Enfoque Colaborativo:** Fomenta una fuerte colaboración entre los miembros del equipo de desarrollo, impulsando la comunicación y el intercambio de conocimientos para construir un ambiente de equipo saludable y un conocimiento compartido sobre el proyecto.
- **Flexibilidad y Adaptabilidad:** Se ajusta a proyectos de diversos tamaños y naturalezas gracias a su enfoque modular y capacidad de adaptación a las cambiantes necesidades, lo que la hace apropiada para proyectos en constante evolución.
- **Minimización de Riesgos:** Pone un fuerte énfasis en la arquitectura temprana, reduciendo riesgos al abordar desafíos técnicos y organizativos desde el principio, disminuyendo la probabilidad de problemas importantes en etapas posteriores.
- **Retroalimentación Continua:** Promueve el desarrollo evolutivo y la obtención de retroalimentación constante de los interesados, permitiendo mejoras continuas y ajustes del proyecto según necesidades reales.
- **Entrega de Valor Constante:** Busca entregar software operativo y funcional al final de cada iteración, permitiendo a clientes y partes interesadas obtener beneficios tangibles de manera predecible a lo largo del proyecto.

- **Eficiencia y Productividad:** Se enfoca en la eficiencia y productividad al fomentar la colaboración, la reutilización de componentes y la eliminación de trabajo innecesario.

En pocas palabras, OpenUP representa un enfoque armonioso y cooperativo para el desarrollo de software, que es altamente adaptable a las variaciones de un proyecto y se enfoca principalmente en proporcionar valor de manera constante.

2.4.1.4 Desventajas

- **Requiere Disciplina del Equipo:** Al igual que en otras metodologías ágiles, OpenUP demanda que el equipo sea comprometido y siga las prácticas y ceremonias establecidas, lo que implicaría un desafío en equipos menos organizados.
- **La Experiencia del Equipo es Necesaria:** Puede no ser la elección más adecuada para equipos nuevos o sin experiencia en desarrollo ágil, ya que OpenUP requiere un nivel sólido de conocimientos y habilidades en desarrollo de software.
- **Necesidad de Documentación:** A pesar de su enfoque en la simplicidad, OpenUP aún implica la creación de documentación, lo cual podría considerarse una desventaja si se busca un enfoque extremadamente minimalista.
- **Mantener la Comunicación Constante es Fundamental:** La comunicación efectiva es esencial en OpenUP. Si el equipo no puede mantener una comunicación continua, la metodología podría no ser tan efectiva.
- **No Adecuada para Todos los Tamaños de Proyectos:** OpenUP brilla especialmente en proyectos de tamaño mediano, por lo que podría no ser la elección más apropiada para proyectos extremadamente pequeños o enormes.

2.5 Arquitectura de Software

La arquitectura de software se refiere a la estructura fundamental y organización de un sistema de software. Es donde se define cómo se construirá y funcionará una aplicación o sistema informático. En otras palabras, es el diseño principal que establece cómo los diferentes

componentes y módulos del software interactuarán entre sí, cómo se gestionará la información y cómo se cumplirán los requisitos del proyecto.

Esta arquitectura proporciona una visión global del sistema y sirve como base para tomar decisiones clave a lo largo del proceso de desarrollo de software. En esencia, la arquitectura de software es el cimiento sobre el cual se construye todo el sistema, asegurando que funcione de manera eficiente, sea mantenible y cumpla con sus objetivos.

"Una arquitectura es el conjunto de decisiones significativas sobre la organización de un sistema de software, la selección de los elementos estructurales y sus interfaces por los cuales está compuesto el sistema, junto con su comportamiento como especificado en las colaboraciones entre esos elementos, la composición de estos elementos estructurales y de comportamiento en subsistemas progresivamente más grandes, y el estilo arquitectónico que guía esta organización, estos elementos y sus interfaces, sus colaboraciones, y su composición" (Kruchten, 1999).

Una de las estructuras arquitectónicas más utilizadas y apreciadas en el mundo del desarrollo de software es la arquitectura por capas. Este enfoque proporciona una forma organizada y modular de diseñar sistemas de software, permitiendo una clara separación de responsabilidades y una mayor flexibilidad en la gestión y el mantenimiento del código. En un mundo donde la adaptabilidad y la escalabilidad son esenciales, comprender cómo funciona la arquitectura por capas es fundamental.

En este contexto, el proyecto se centra en la implementación de una arquitectura por capas específica para abordar las necesidades de centralización, gestión y control de la documentación de los funcionarios de la alcaldía. A lo largo de este trabajo, se explorarán los conceptos clave de esta arquitectura y cómo se aplican en el proyecto, considerando las particularidades y los objetivos ya definidos.

Es importante tener en cuenta que la arquitectura por capas no es una solución universal, pero es una herramienta valiosa en la caja de herramientas del desarrollador de software. En última instancia, el éxito de su implementación dependerá de la comprensión precisa de los requisitos del proyecto y de la capacidad de adaptación a las necesidades específicas de su institución.

2.5.1 *Arquitectura por Capas*

La arquitectura por capas se asemeja a la construcción de una casa, pero en lugar de edificarla de una sola vez, se divide en diferentes niveles. Cada nivel tiene una función particular: el primero es como los cimientos, el segundo sería donde residen las personas, el tercero puede considerarse como el ático, y así sucesivamente. Cada nivel se comunica con el nivel adyacente, pero no necesita conocer lo que sucede en los otros niveles. Esto simplifica la construcción, modificación o reparación de la estructura sin afectar el conjunto. En el ámbito del software, es similar a fragmentar una aplicación en componentes con tareas específicas, lo que facilita su desarrollo y mantenimiento.

“La arquitectura en capas es una de las más utilizadas, no solo por su simplicidad, sino porque también es utilizada por defecto cuando no estamos seguros que arquitectura debemos de utilizar para la aplicación.

La arquitectura en capas consta en dividir la aplicación en capas, con la intención de que cada capa tenga un rol muy definido, como podría ser, una capa de presentación (UI), una capa de reglas de negocio (servicios) y una capa de acceso a datos (DAO), sin embargo, este estilo arquitectónico no define cuantas capas debe de tener la aplicación, sino más bien, se centra en la separación de la aplicación en capas (Aplica el principio Separación de preocupaciones (SoC)).

En la práctica, la mayoría de las veces este estilo arquitectónico es implementado en 4 capas: presentación, negocio, persistencia y base de datos, sin embargo, es habitual ver que

la capa de negocio y persistencia se combinan en una sola capa, sobre todo cuando la lógica de persistencia está incrustada dentro de la capa de negocio” (Blancarte, s. f.).

Según Arévalo Lizardo (2010), las tres capas fundamentales de un patrón de arquitectura por capas, que son componentes esenciales en la organización y estructura de un sistema o aplicación:

2.5.1.1 Capa de Presentación

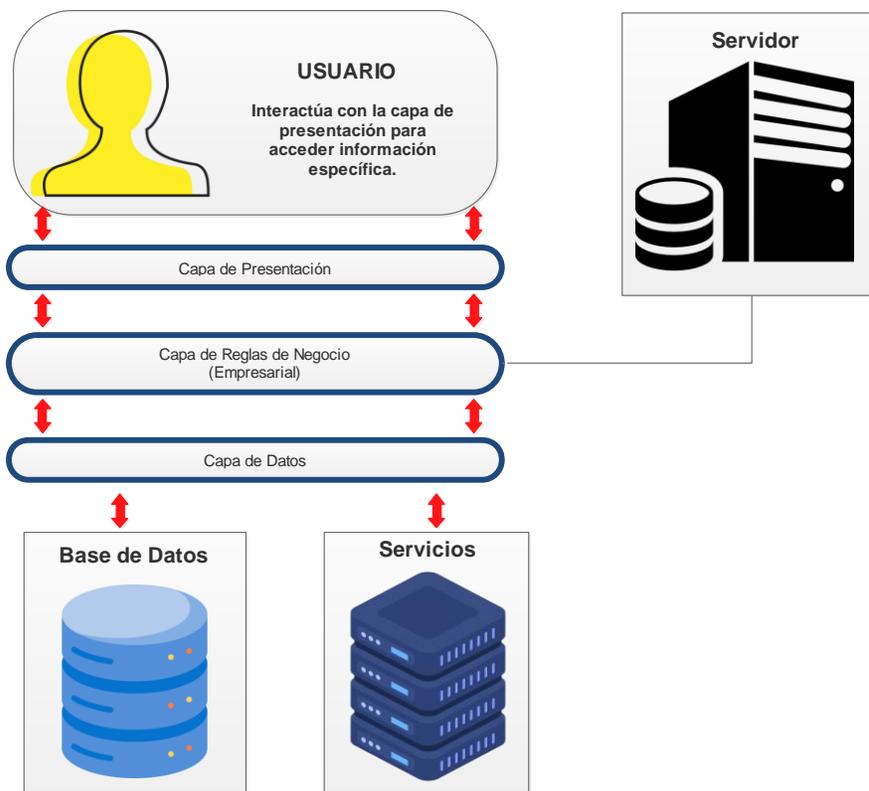
Referente a la interacción entre el usuario y el software. Puede ser tan simple como un menú basado en líneas de comando o tan complejo como una aplicación basada en formas. Su principal responsabilidad es mostrar información al usuario, interpretar los comandos de este y realizar algunas validaciones simples de los datos ingresados.

2.5.1.2 Capa de Reglas de Negocio (Empresarial)

La Capa de Negocio, también llamada Lógica de Dominio, implementa la funcionalidad principal de la aplicación. Realiza cálculos basados en datos de usuarios y validaciones, controla la interacción con la capa de acceso a datos y servicios externos. Puede diseñarse para ser utilizado directamente por componentes de presentación o encapsulado como un servicio, accesible a través de una interfaz que gestiona la comunicación con los clientes o invoca otros flujos o componentes de negocio.

2.5.1.3 Capa de Datos

Esta capa en una arquitectura de capas proporciona la infraestructura necesaria para gestionar de manera eficiente y segura la persistencia de datos, facilitando así la manipulación y gestión de la información dentro de la aplicación.

Figura 4*Patrón de Arquitectura por capas***2.5.2 Características de la Arquitectura por Capas**

Según Peláez (s. f.), el estilo de arquitectura basado en capas se identifica por las siguientes características:

- Describe la descomposición de servicios de forma que la mayoría de la interacción ocurre solamente entre capas vecinas.
- Las capas de una aplicación pueden residir en la misma máquina física (misma capa) o puede estar distribuido sobre diferentes computadores (n-capas).
- Los componentes de cada capa se comunican con otros componentes en otras capas a través de interfaces muy bien definidas.
- Este modelo ha sido descrito como una “pirámide invertida de re-uso” donde cada capa agrega responsabilidad y abstracción a la capa directamente sobre ella.

2.5.3 Principios de la Arquitectura por Capas

También (Peláez, s. f.) al diseñar para utilizar este estilo de arquitectura, se aplican principios comunes que son esenciales en el proceso de diseño y estos incluyen:

- **Abstracción.** La arquitectura basada en capas abstrae la vista del modelo como un todo mientras que provee suficiente detalle para entender las relaciones entre capas.
- **Funcionalidad claramente definida.** El diseño claramente define la separación entre la funcionalidad de cada capa. Capas superiores como la capa de presentación envía comandos a las capas inferiores como la capa de negocios y la capa de datos y los datos fluyen hacia y desde las capas en cualquier sentido.
- **Alta cohesión.** Cada capa contiene funcionalidad directamente relacionadas con la tarea de dicha capa.
- **Reutilizable.** Las capas inferiores no tienen ninguna dependencia con las capas superiores, permitiéndoles ser reutilizables en otros escenarios.
- **Desacople.** La comunicación entre las capas está basada en la abstracción lo que provee un desacople entre las capas.

2.5.4 Ventajas de la Arquitectura por Capas

- **Separación de Responsabilidades.** Significa que cada capa de nuestro sistema tiene un trabajo específico, como si cada una tuviera su tarea. Esto ayuda a mantener las cosas organizadas.
- **Fácil de Desarrollar.** Este tipo de arquitectura es sencillo de usar y mucha gente la utiliza, por lo que es fácil de aprender y aplicar en proyectos.
- **Fácil de Probar.** Como nuestra aplicación está dividida en capas, podemos probar cada una por separado, como si revisáramos cada parte de un rompecabezas una por una.

- **Fácil de Mantener.** Debido a que cada capa hace un trabajo específico, si algo sale mal, podemos identificar fácilmente dónde está el problema y corregirlo. También es simple hacer cambios porque sabemos exactamente qué hace cada parte.
- **Mayor Seguridad.** Gracias a la separación de las capas, podemos aislar nuestros servidores en diferentes grupos, como si los pusiéramos en habitaciones separadas, lo que hace que sea más difícil para las personas malintencionadas intentar dañar el sistema.

Al igual que Platzi (s. f.), destaca varios beneficios importantes de esta arquitectura basada en capas, una variedad de ventajas clave, tales como:

- Abstracción, encapsulamiento, capas de funcionalidad muy bien definidas, alta cohesión, reusabilidad (las capas inferiores de la pirámide no tienen dependencias de las superiores por lo que es fácil reutilizarlas) y débil acoplamiento (comunicación basada en abstracciones)
- La separación, reduce el riesgo y el impacto de los cambios tecnológicos. Por ejemplo, al cambiar el mecanismo de persistencia de la aplicación, las capas que lo consumen no tienen por qué realizar cambios. Además, aumenta el rendimiento (a partir de cierta carga de trabajo) al distribuir las capas sobre múltiples capas físicas. También aumenta la escalabilidad y la tolerancia a fallos.
- Este escenario es propicio para implementar una buena política de pruebas, permitiendo conmutar entre diferentes implementaciones de los interfaces (mock, servicios reales...).

2.5.5 Desventajas de la Arquitectura por Capas

- **Complejidad y Sobrecarga.** La introducción de capas adicionales puede aumentar la complejidad de la aplicación. Cada capa agrega una sobrecarga de comunicación y gestión, lo que puede dificultar la comprensión y el mantenimiento de la aplicación.

- **Posible Degradación del Rendimiento.** La comunicación entre capas, especialmente a través de redes o internet, puede llevar a una degradación del rendimiento debido a la latencia y el procesamiento adicional necesario para pasar datos entre capas.
- **Dificultad en la Depuración y el Seguimiento de Problemas.** Cuando ocurre un problema en la aplicación, identificar la capa específica en la que se originó el problema puede ser un desafío. La falta de aislamiento entre capas puede hacer que la depuración y el seguimiento de problemas sean complicados.

2.5.6 Conclusiones de la Arquitectura

En conclusión, la arquitectura por capas se erige como un enfoque sólido y versátil en el diseño de sistemas de software. Su capacidad para separar responsabilidades, su facilidad de implementación y su bajo costo de mantenimiento la convierten en una opción atractiva para una amplia gama de aplicaciones. Sin embargo, es esencial ser consciente de sus limitaciones, como posibles restricciones en el rendimiento y la escalabilidad, y considerar cuidadosamente su idoneidad en contextos de alto procesamiento de datos o de crecimiento acelerado.

En última instancia, la arquitectura por capas se presenta como una herramienta valiosa en el arsenal de un arquitecto de software, ofreciendo un equilibrio entre simplicidad y funcionalidad que puede ser clave para el éxito en numerosos proyectos de desarrollo de software.

2.6 Herramientas de Desarrollo Utilizadas

En este apartado, se explorará las herramientas de desarrollo clave que desempeñaron un papel fundamental en la implementación exitosa de este proyecto. El enfoque principal del desarrollo se basó en el stack MERN, que integra MongoDB, Express.js, React y Node.js para crear una aplicación web eficiente y dinámica. Cada una de estas tecnologías desempeñó un papel específico en la construcción de la aplicación, desde la gestión de datos hasta la

creación de la interfaz de usuario. A continuación, se profundizará en cada una de estas herramientas, destacando su contribución al proyecto y su importancia en el contexto de la arquitectura de software utilizada.

2.6.1 *Mongo DB*

En este proyecto, se optó por MongoDB como la base de datos principal. MongoDB es una base de datos NoSQL que se destaca por su flexibilidad y escalabilidad. A diferencia de las bases de datos tradicionales basadas en tablas, MongoDB almacena los datos en documentos similares a archivos, lo que lo hace perfecto para gestionar información diversa y en constante evolución. Su capacidad para manejar grandes volúmenes de datos y su eficiente integración con aplicaciones Node.js lo convierten en una elección sólida para este sistema. Además, su modelo de documentos JSON permite una representación natural de los datos y facilita la adaptación a medida que los requisitos del proyecto evolucionan.

“MongoDB (del inglés humongous, "enorme") es un sistema de base de datos NoSQL orientado a documentos de código abierto y escrito en C++, que en lugar de guardar los datos en tablas lo hace en estructuras de datos BSON (similar a JSON) con un esquema dinámico. Al ser un proyecto de código abierto, sus binarios están disponibles para los sistemas operativos Windows, GNU/Linux, OS X y Solaris y es usado en múltiples proyectos o implementaciones en empresas como MTV Network, Craigslist, BCI o Foursquare” (OpenWebinars, 2019).

- **Uso en el Proyecto:** MongoDB fue fundamental para gestionar eficientemente los datos personales de los funcionarios y las revisiones realizadas por los evaluadores. Proporcionó flexibilidad y escalabilidad, adaptándose a las necesidades cambiantes del proyecto. Registró información de usuarios, citas y manuales de registro, siendo esencial para el uso adecuado del sistema.

- **Ventajas:** MongoDB ofreció flexibilidad, escalabilidad y una integración perfecta con Node.js. Su modelo de documentos JSON facilitó la representación y manipulación de datos. Contribuyó a la adaptabilidad del proyecto y la eficiencia en todas las etapas de desarrollo.
- **Desafíos:** Se enfrentaron desafíos en la transición de procesos de importación/exportación de datos hacia MongoDB, requiriendo una revisión profunda de flujos de datos. Sin embargo, se superaron con éxito gracias a la flexibilidad de MongoDB.
- **Impacto:** MongoDB permitió una gestión ágil de datos, una interfaz de usuario dinámica y escalabilidad para manejar un crecimiento continuo de datos. Mejoró la calidad y eficiencia del sistema, influyendo de manera positiva en el proyecto en su conjunto.

2.6.2 *Express.js*

Express.js es un marco de desarrollo web para Node.js, una plataforma de JavaScript en el lado del servidor. Express simplifica la creación de aplicaciones web y API al proporcionar un conjunto de herramientas y funcionalidades predefinidas que permiten a los desarrolladores construir sitios web y aplicaciones de manera más eficiente. Es ampliamente utilizado en el desarrollo web para crear servidores web y gestionar rutas, solicitudes HTTP y respuestas de manera sencilla.

“Express.js, a veces también llamado «Express», es un framework de backend Node.js minimalista, rápido y similar a Sinatra, que proporciona características y herramientas robustas para desarrollar aplicaciones de backend escalables. Le ofrece el sistema de enrutamiento y características simplificadas para ampliar el framework con componentes y partes más potentes en función de los casos de uso de su aplicación” (Kinsta, 2022).

- **Uso en el Proyecto:** Express.js desempeñó un papel fundamental en la definición y manejo de las rutas de la aplicación. Se utilizaron rutas para gestionar la interacción entre el cliente y el servidor, permitiendo la creación, lectura, actualización y eliminación de datos de manera eficiente. Además, Express.js facilitó la implementación de middleware para procesar solicitudes y respuestas.
- **Ventajas:** Las ventajas clave de Express.js incluyen su rapidez en el desarrollo y su escalabilidad. Gracias a su estructura modular, se logró una organización efectiva del código, lo que simplificó el mantenimiento y la adición de nuevas características.
- **Desafíos:** A pesar de sus numerosas ventajas, algunos desafíos surgieron al utilizar Express.js, como la necesidad de gestionar la seguridad y la autenticación de manera adecuada. Se requiere un enfoque cuidadoso en la definición de rutas y controladores para evitar problemas de seguridad.
- **Impacto:** El uso de Express.js tuvo un impacto significativo en el desarrollo y los resultados del proyecto. Permitted la creación de un backend robusto que manejó de manera eficiente las solicitudes del cliente, lo que llevó a una experiencia de usuario más fluida. Además, Express.js contribuyó a la rapidez en la implementación de nuevas características, lo que resultó en una mayor versatilidad de la aplicación.

2.6.3 *React*

React, una biblioteca de JavaScript ampliamente utilizada, desempeñó un papel central en la construcción de la interfaz de usuario. React se destacó por su capacidad para crear interfaces de usuario interactivas, eficientes y de alta calidad, y se convirtió en la elección principal para la parte frontal de la aplicación.

“React le ayuda a crear interfaces de usuario interactivas de forma sencilla. Diseña vistas simples para cada estado en su aplicación, y React se encargará de actualizar y

renderizar de manera eficiente los componentes correctos cuando los datos cambien” (React, 2023).

- **Uso en el Proyecto:** Se utilizó para desarrollar una interfaz de usuario altamente interactiva. Gracias a su arquitectura basada en componentes, se construyeron elementos reutilizables que permitieron una mayor eficiencia en el desarrollo y proporcionaron una experiencia de usuario fluida y atractiva.
- **Ventajas:** React se destacó por su capacidad para crear componentes reutilizables, lo que redujo significativamente la duplicación de código y aceleró el desarrollo de nuevas características. Su enfoque en la actualización eficiente del DOM (Modelo de Objetos del Documento) permitió una experiencia de usuario receptiva sin recargar la página por completo.
- **Desafíos:** Aunque React ofrece numerosas ventajas, también presenta desafíos, como la curva de aprendizaje para quienes no están familiarizados con su sintaxis y conceptos. Además, se requiere una cuidadosa gestión del estado de la aplicación para evitar problemas de rendimiento y errores.
- **Impacto:** La elección de React tuvo un impacto significativo en el proyecto, ya que permitió la creación de una interfaz de usuario atractiva y altamente funcional. Los componentes reutilizables simplificaron el mantenimiento y la expansión de la aplicación, lo que se tradujo en una mayor eficiencia en el desarrollo y una experiencia de usuario mejorada.

2.6.4 Node.js

En esta iniciativa, Node.js fue esencial como el motor de ejecución en el servidor. Node.js, una plataforma de código abierto, que se destaca por su habilidad para ejecutar código JavaScript en el servidor, lo que lo convierte en una herramienta clave para impulsar la lógica y las operaciones en el lado del servidor en aplicaciones web y móviles.

“Node.js es un entorno de ejecución de un solo hilo y multiplataforma basado en el motor V8 de JavaScript de Google Chrome. Es un software de código abierto para construir aplicaciones de red escalables y en tiempo real” (HOSTINGER TUTORIALES, 2023b).

- **Uso en el Proyecto:** Node.js se utilizó para gestionar la lógica del servidor en el proyecto. Permitió la implementación de rutas, controladores y la manipulación de solicitudes HTTP de manera efectiva.
- **Ventajas:** Entre las ventajas que Node.js aportó al proyecto, se destaca su capacidad para manejar solicitudes simultáneas de manera eficiente. Esto fué especialmente relevante para garantizar una respuesta rápida en una aplicación con múltiples usuarios.
- **Desafíos:** Un desafío que enfrenté con Node.js fue la curva de aprendizaje, ya que no estaba familiarizado con la programación en el lado del servidor utilizando JavaScript. Sin embargo, lo superé mediante la búsqueda de información y recursos adecuados.
- **Impacto:** El uso de Node.js en el proyecto tuvo un impacto significativo en el rendimiento y la capacidad de respuesta de la aplicación, lo que contribuyó a una experiencia del usuario mejorada.

2.6.5 *Axios*

Axios, es una biblioteca JavaScript, que desempeñó un papel vital en el proyecto como cliente HTTP. Axios se ha convertido en una elección común para la realización de solicitudes HTTP en aplicaciones web basadas en JavaScript. Esta biblioteca se caracteriza por su simplicidad y facilidad de uso, lo que la hace ideal para interactuar con servicios web.

- **Uso en el Proyecto:** Se utilizó en el proyecto para facilitar la comunicación entre el cliente y el servidor. Esto involucró la realización de solicitudes GET, POST, DELETE y PUT para la obtención y envío de datos entre la aplicación y el servidor.

- **Ventajas:** Las ventajas de Axios se hicieron evidentes en la sencillez de su implementación y la capacidad de gestionar solicitudes asíncronas, lo que garantizó una experiencia del usuario fluida.
- **Desafíos:** Uno de los desafíos potenciales al usar Axios radicó en el manejo de errores de red o solicitudes fallidas. Sin embargo, con un manejo adecuado de excepciones, se superó este obstáculo de manera eficiente.
- **Impacto:** Desempeñó un papel fundamental en la comunicación entre el cliente y el servidor, lo que contribuyó en gran medida a la capacidad de respuesta de la aplicación y la capacidad de obtener y enviar datos de manera eficiente.

2.6.6 *Redux Toolkit*

Redux Toolkit, una librería popular de JavaScript, desempeñó un papel esencial en el proyecto al facilitar la gestión del estado de la aplicación. Esta librería se ha convertido en una elección común para el manejo de estados en aplicaciones web basadas en React. Su simplicidad y eficacia, respaldada por la comunidad de desarrollo, la hacen ideal para implementar un almacenamiento de datos predecible.

“Implementar Redux Toolkit en una aplicación de React es una forma eficiente y simplificada de gestionar el estado global de su aplicación. Redux Toolkit proporciona utilidades y patrones recomendados para trabajar con Redux, lo que reduce la cantidad de código boilerplate necesario” (Medium, s. f.).

- **Uso en el Proyecto:** Redux Toolkit se empleó en el proyecto para centralizar y gestionar el estado de la aplicación, permitiendo un acceso consistente a los datos en toda la interfaz de usuario. Se configuraron "slices" para diferentes partes de la aplicación, lo que facilitó la organización y la gestión de los datos.
- **Ventajas:** Las ventajas de Redux Toolkit se hicieron evidentes en la estructura ordenada que proporcionó para el estado de la aplicación. Esto condujo a un código

más limpio y mantenible, y permitió una administración eficiente del flujo de datos en la aplicación.

- **Desafíos:** Uno de los desafíos posibles al utilizar Redux Toolkit fue la curva de aprendizaje, debido a una cantidad considerable de estados que maneja el sistema. Sin embargo, la inversión en la configuración inicial demostró ser beneficiosa a lo largo del proyecto.
- **Impacto:** Redux Toolkit tuvo un impacto significativo en la organización y la administración del estado de la aplicación. Contribuyó a una experiencia del usuario más veloz y permitió una mayor flexibilidad en la gestión de datos.

2.6.7 Tailwind CSS

Tailwind CSS es una ampliamente reconocida herramienta de diseño de código abierto que desempeñó un papel fundamental en el proyecto al acelerar la creación de la interfaz de usuario. A diferencia de los marcos convencionales que proporcionan componentes predefinidos, Tailwind CSS se apoya en un enfoque de diseño utilitario, lo que habilita una personalización minuciosa de la interfaz.

- **Uso en el Proyecto:** En el proyecto, se empleó Tailwind CSS para diseñar y dar estilo de forma eficaz a los componentes de React. Se utilizaron clases de estilo en línea en los elementos HTML para establecer la apariencia y la disposición de los componentes de la aplicación.
- **Ventajas:** Las ventajas de Tailwind CSS se hicieron evidentes en su enfoque orientado a la utilidad, lo que permitió el desarrollo rápido de componentes personalizados sin necesidad de escribir CSS personalizado. Esto condujo a una interfaz de usuario consistente y fácil de mantener.

- **Desafíos:** Uno de los desafíos al utilizar Tailwind CSS fue la adaptación a este enfoque de diseño utilitario, que puede ser diferente a las prácticas tradicionales. Sin embargo, con la documentación y recursos disponibles, el equipo superó este obstáculo.
- **Impacto:** Tailwind CSS tuvo un impacto significativo en la velocidad y la coherencia del desarrollo de la interfaz de usuario. Facilitó la creación de una interfaz atractiva y funcional de manera eficiente.

2.6.8 Bibliotecas y Librerías Utilizadas en el Proyecto

En este proyecto, se hizo un uso de diversas bibliotecas y librerías para optimizar el desarrollo y la funcionalidad del sistema. A continuación, se presenta una breve descripción de algunas de las principales bibliotecas y librerías empleadas:

- **Vite:** Una herramienta de desarrollo web ultrarrápida, sirvió como base para la construcción de la interfaz de usuario. Su velocidad de compilación y recarga en tiempo real facilitaron un flujo de trabajo eficiente y un desarrollo ágil.
- **Date-fns:** La librería Date-fns se utilizó para abordar operaciones relacionadas con fechas y horas en el proyecto. Con su amplia gama de funciones, facilitó la gestión precisa de datos temporales y contribuyó a la lógica de programación del sistema.
- **Bcrypt:** Fue esencial para la gestión de la seguridad y la encriptación de contraseñas en el proyecto. Esta librería garantizó un almacenamiento seguro de las credenciales de los usuarios, mitigando posibles amenazas de seguridad.
- **JWT (JSON Web Tokens):** La implementación de JSON Web Tokens fue una parte fundamental de la autenticación y la seguridad en el sistema. La librería JWT permitió la creación, validación y gestión de tokens de manera eficiente, asegurando la autenticación segura de los usuarios.
- **Formik:** Es una librería de JavaScript que se utiliza para gestionar y simplificar la creación de formularios en aplicaciones web. Proporciona herramientas para manejar

la validación, el envío de datos y la interacción con formularios de manera eficiente y efectiva. Formik es ampliamente utilizado en proyectos de desarrollo web para mejorar la experiencia de los usuarios al interactuar con formularios, como los de inicio de sesión, registro y recopilación de datos.

- **jsPDF:** Es una biblioteca JavaScript ampliamente reconocida que permite la generación de documentos PDF en aplicaciones web. Con esta herramienta, los desarrolladores pueden crear documentos dinámicos como informes, facturas, formularios y más, directamente en el navegador del usuario final. jsPDF es una solución eficaz y versátil para aplicaciones que requieren la creación y descarga de documentos PDF de manera sencilla y personalizada.

Estas son solo algunas de las bibliotecas y librerías utilizadas en el proyecto, cada una desempeñando un papel vital en aspectos específicos del desarrollo.

2.7 Método de Estimación de Costos (COCOMO II)

El Modelo Constructivo de Costos II, comúnmente abreviado como COCOMO II, es un enfoque ampliamente utilizado en la industria del software para estimar los costos, el esfuerzo y el tiempo necesarios para llevar a cabo proyectos de desarrollo de software. Este modelo se basa en una serie de fórmulas que tienen en cuenta diversas variables, como el tamaño del proyecto, la complejidad, la experiencia del equipo y otros factores. COCOMO II es una herramienta esencial para la gestión de proyectos de software, ya que proporciona estimaciones sólidas que ayudan a planificar y controlar los recursos y el cronograma del proyecto de manera efectiva. Su capacidad para adaptarse a diferentes tipos de proyectos y su enfoque en la precisión lo convierten en una herramienta invaluable para los profesionales de la industria del software.

“Modelo COCOMO II, modelo de estimación que se encuentra en la jerarquía de modelos de estimación de software con el nombre de COCOMO, por Constructive Cost Model

(Modelo Constructivo de Coste). El modelo COCOMO original se ha convertido en uno de los modelos de estimación de coste del software más utilizados y estudiados en la industria” (EcuRed, s. f.).

2.7.1 Características de modelo COCOMO II

Las siguientes consideraciones son esenciales para comprender el enfoque del modelo COCOMO II en la estimación de costos de desarrollo de software. Estas características son fundamentales para la aplicación efectiva de COCOMO II en proyectos de desarrollo de software. A continuación, se detallan las características clave del modelo que guían la estimación de costos y esfuerzo en este contexto.

- Enfocarse en factores de costo, como el tamaño del código, la escala y los multiplicadores de esfuerzo.
- Suponer que los requisitos permanecerán estables tras la planificación inicial, con margen para ajustes inevitables.
- Estimar costos desde la revisión de requisitos hasta la aprobación de los módulos y el proyecto en su conjunto.
- Considerar solo actividades directamente relacionadas con el proyecto, excluyendo tareas realizadas por personal no directamente involucrado.
- Emplear la unidad "mes-persona" en lugar de una moneda específica para calcular costos, permitiendo conversiones basadas en promedios salariales en diferentes etapas del proyecto.

2.7.2 La Importancia de COCOMO II en un Mercado Diverso

La complejidad y diversidad del mercado actual y futuro en el ámbito del desarrollo de software plantea un desafío de considerable magnitud. Los profesionales se encuentran frente a una amplia gama de factores que abarcan múltiples plataformas, lenguajes de programación y enfoques metodológicos. En este contexto, COCOMO II se posiciona como una herramienta

esencial para la gestión de proyectos, ofreciendo estimaciones precisas y adaptables que capacitan a los expertos para navegar exitosamente en este entorno en constante evolución.

Según Gómez Adriana et al. (s. f.), se señala que estos retos emergen como una reacción a la heterogeneidad presente y futura en el campo del desarrollo de software. Este panorama diverso puede ser visualizado mediante el esquema propuesto en su libro.

Figura 5

Diversidad del Mercado Actual y Futuro del desarrollo de software



Nota: Adaptado de Modelo de Estimación COCOMO (p. 26), S. Gómez, s. f...

2.7.3 Modelos de Estimación COCOMO II

COCOMO II está compuesto por tres modelos denominados: Composición de Aplicación, Diseño Temprano y Post-Arquitectura.

- **Composición de Aplicación:** Se utiliza en la etapa de prototipado de proyectos de desarrollo de software para diseñar y evaluar prototipos.
- **Diseño Temprano:** El modelo Diseño Temprano se emplea en las primeras etapas de desarrollo para evaluar opciones de hardware y software con información limitada, utilizando Puntos Función y pocos factores de costo.
- **Post-Arquitectura:** Se utiliza después de establecer la estructura del software y al construirlo, así como para las actualizaciones y correcciones posteriores.

Este proyecto se enfocará en el modelo Post-Arquitectura de COCOMO II, el cual desempeña un papel esencial en el desarrollo de software una vez que se ha definido la arquitectura del sistema. Se explorará cómo aplicar este modelo de manera efectiva en el

contexto del proyecto, considerando aspectos clave como la estimación del tamaño del software y la influencia de los factores de costo.

2.7.3.1 Modelo Post-Arquitectura. Gómez Adriana et al. (s. f.) señalan que este modelo es el más detallado en términos de estimación y se aplica específicamente cuando la arquitectura del proyecto se encuentra completamente definida.

“El esfuerzo nominal se ajusta usando 17 factores multiplicadores de esfuerzo. El mayor número de multiplicadores permite analizar con más exactitud el conocimiento disponible en las últimas etapas de desarrollo, ajustando el modelo de tal forma que refleje fielmente el producto de software bajo desarrollo” (Gómez Adriana et al., s. f.).

Ecuacion 1

Esfuerzo Estimado usando los 17 factores multiplicadores.

$$PM_{estimado} = PM_{nominal} \times \prod_{i=1}^{17} EM_i$$

Para lo cual se necesita:

Ecuacion 2

Esfuerzo Nominal

$$PM_{nominal} = A \times (KSLOC)^B$$

Donde:

- **KSLOC:** Es el tamaño del software a desarrollar expresado en miles de líneas de código fuente.
- **A:** Es una constante que captura los efectos lineales sobre el esfuerzo de acuerdo a la variación del tamaño, (A=2.94).
- **B:** Es el factor exponencial de escala, toma en cuenta las características relacionadas con la economía de escala producidas cuando un proyecto de software incrementa su tamaño.

Factor Exponencial de Escala (B): El Factor Exponencial de Escala (B) en COCOMO II post-arquitectura es un parámetro que se utiliza en la estimación de esfuerzo y tiempo de desarrollo de proyectos de software, reflejando la relación entre el tamaño del proyecto y la complejidad, donde un valor más alto de B indica un proyecto más complejo.

Ecuación 3

Factor Exponencial de Escala (B)

$$B = 1.01 + 0.01 \times \sum_{j=1}^5 W_j$$

Donde:

- $B < 1.0$, indica economía de escala
- $B = 1.0$, equilibrio en proyectos pequeños
- $B > 1.0$, muestra costos crecientes debido a la complejidad de integración.

El cálculo de B se basa en factores exponenciales que influyen en la productividad y varían desde Muy Bajo hasta Extra Alto, cada uno con un peso específico. A continuación, una breve descripción de los mismos.

- **Precedencia (Precedence- PREC):** Este factor refleja la experiencia previa y el conocimiento adquirido en proyectos similares.
- **Desarrollo de personal (Development Flexibility - FLEX):** Este factor considera la disponibilidad de personal experimentado y la flexibilidad de programación.
- **Requisitos del sistema (Architecture/Risk Resolution - RESL):** Este factor refleja la claridad y la estabilidad de los requisitos del sistema.
- **Calidad del proceso de software (Team Cohesion - TEAM):** Este factor se relaciona con la calidad del proceso de desarrollo de software y la cohesión del equipo.
- **Experiencia del usuario final (Customer/End-User Involvement - PMAT):** Este factor refleja la participación del usuario final en el proceso de desarrollo.

Estos factores se utilizan para ajustar la estimación inicial del esfuerzo y el costo de un proyecto, teniendo en cuenta las circunstancias específicas de ese proyecto. Cada uno de los factores de escala tiene un rango de valores que afecta la estimación final. Al considerar estos factores, COCOMO II busca proporcionar estimaciones más precisas y ajustadas a las condiciones particulares de cada proyecto de desarrollo de software.

A continuación, se presentan dos ilustraciones que muestran la escala de valores que pueden tomar los factores de escala y el valor asociado a cada uno de estos factores. Estas representaciones visuales permiten comprender mejor cómo se distribuyen y clasifican los distintos factores según su magnitud o impacto, facilitando así la interpretación y análisis de los datos relacionados con dichos factores.

Tabla 1

Factores de Escala

Factor de Escala Wj	Muy Bajo	Bajo	Normal	Alto	Muy Alto	Extra Alto
Precedencia PREC	Completamente sin precedentes	Ampliamente sin precedentes	Algún precedente	Generalmente Familiar	Ampliamente Familiar	Completamente Familiar
Flexibilidad en el desarrollo FLEX	Rigurosa	Relajación Ocasional	Alguna Relajación	Conformidad en general	Alguna Conformidad	Metas generales
Arquitectura/ Resolución de riesgo RESL	Poca (20%)	Alguna (40%)	Siempre (60%)	Generalmente 75%)	Principalment e (90%)	Completo (100%)
Cohesión de equipo TEAM	Interacciones difíciles	Interacciones con alguna dificultad	Interacciones básicamente cooperativas	Ampliamente Cooperativas	Altamente Cooperativas	Interacciones Sin Fisuras
Madurez del proceso PMAT	No se aplica	Casi nunca (<10%)	Ocasionalmente (10-40%)	La mitad de las veces (40-60%)	A menudo (60-90%)	Casi Siempre (90%)

Nota: Valoraciones de los Factores de Escala. Adaptado de COCOMO: Un Modelo de Estimación de Proyectos de Software (p. 43), por A. Gómez, M. del C. López, S. Migani, & A. Otazú, s. f..

Tabla 2

Valores del Factor de Escala.

	Parametros de Factores de Escala					
	VLO	LO	NOM	HI	VHI	XHI
PREC	6,20	4,96	3,72	2,48	1,24	0,00
FLEX	5,06	4,05	3,04	2,03	1,01	0,00
RESL	7,07	5,65	4,24	2,83	1,41	0,00
TEAM	5,48	4,38	3,39	2,19	1,1	0,00
PNAT	7,8	6,24	4,68	3,12	1,56	0,00

Nota: Ponderación de los FE. Adaptado de COCOMO: Un Modelo de Estimación de Proyectos de Software (p. 43), por A. Gómez, M. del C. López, S. Migani, & A. Otazú, s. f..

Exploración de los 17 Factores Multiplicadores de Esfuerzo

Factores de Producto

- **RELY (Confiabilidad):** Evalúa la confiabilidad necesaria, desde "Muy Bajo" hasta "Muy Alto" según el impacto de las fallas.
- **DATA (Tamaño Base de Datos):** Relaciona el esfuerzo con el tamaño de la base de datos.
- **CPLX (Complejidad):** Evalúa la complejidad de las operaciones en el producto.
- **RUSE (Reusabilidad):** Mide el esfuerzo adicional para crear componentes reutilizables.
- **DOCU (Documentación):** Evalúa la adecuación de la documentación, desde "Muy Bajo" hasta "Muy Alto".

A continuación, se presenta un gráfico que ilustra los valores de los cinco Factores de Producto: RELY (Confiabilidad Requerida), DATA (Tamaño de la Base de Datos), CPLX (Complejidad del Producto), RUSE (Requerimientos de Reusabilidad) y DOCU (Documentación del Proyecto).

Tabla 3

Valores de los Factores Producto.

	Parámetros de Factores de Producto					
	Muy Bajo	Bajo	Normal	Alto	Muy Alto	Extra Alto
RELY	0,82	0,92	1,00	1,10	1,26	0,00
DATA	0,00	0,90	1,00	1,14	1,28	0,00
docu	0,81	0,91	1,00	1,11	1,23	0,00
cplx	0,73	0,87	1,00	1,17	1,34	1,74
ruse	0,00	0,95	1,00	1,07	1,15	1,24

Nota: Adaptado de COCOMO: Un Modelo de Estimación de Proyectos de Software (p. 50), por A. Gómez, M. del C. López, S. Migani, & A. Otazú, s. f.

Factores de la Plataforma

- **PVOL (Volatilidad de la Plataforma):** Evalúa la frecuencia de cambios en la plataforma subyacente, incluyendo hardware y software base.
- **STOR (Restricción de Almacenamiento):** Mide el grado de restricción de almacenamiento principal en porcentaje.
- **TIME (Restricción de Tiempo de Ejecución):** Representa la restricción de tiempo de ejecución en porcentaje, desde Nominal hasta Extra Alto.

A continuación, se presenta un gráfico que ilustra los valores de estos Factores de la Plataforma para comprender su influencia en la estimación de costos.

Tabla 4*Valores de los Factores Producto.*

	PARÁMETROS de Factores de Plataforma					
	Muy Bajo	Bajo	Normal	Alto	Muy Alto	Extra Alto
TIME	0,00	0,00	1,00	1,11	1,29	1,29
STOR	0,00	0,00	1,00	1,05	1,17	1,46
PVOL	0,00	0,87	1,00	1,15	1,3	0,00

Nota: Adaptado de COCOMO: Un Modelo de Estimación de Proyectos de Software (p. 51), por A. Gómez, M. del C. López, S. Migani, & A. Otazú, s. f.

Factores del Personal

- **ACAP (Capacidad del Analista):** Evalúa la habilidad del analista en diseño, análisis, comunicación y cooperación, sin considerar la experiencia.
- **PCAP (Capacidad del Programador):** Destaca la importancia de la capacidad del programador en el uso de herramientas y habilidades para trabajar en equipo.
- **PCON (Continuidad del Personal):** Mide la permanencia anual del personal en un proyecto de software, desde 3% (Muy Alto) hasta 48% (Muy Bajo).
- **AEXP (Experiencia en la Aplicación):** Evalúa la experiencia del equipo en aplicaciones similares, desde Muy Bajo (menos de 2 meses) hasta Muy Alto (6 o más años).
- **PEXP (Experiencia en la Plataforma):** Reconoce la influencia de la experiencia en nuevas plataformas, desde Muy Bajo (menos de 2 meses) hasta Muy Alto (6 o más años).
- **LTEX (Experiencia en Lenguaje y Herramientas):** Mide la experiencia del equipo en el uso de lenguajes y herramientas de programación, incluyendo análisis, diseño y administración de configuración.

A continuación, presentamos un resumen de los valores asociados a estos Factores del Personal que influyen en la estimación de costos, desde la capacidad de los analistas hasta la experiencia en lenguajes y herramientas.

Tabla 5

Valores de los Factores de Personal.

PARÁMETROS de Factores del Personal						
	Muy Bajo	Bajo	Normal	Alto	Muy Alto	Extra Alto
ACAP	1,42	1,19	1,00	0,85	0,71	0,00
AEXP	1,22	1,10	1,00	0,88	0,81	0,00
PCAP	1,34	1,15	1,00	0,88	0,76	0,00
PEXP	1,19	1,09	1,00	0,91	0,85	0,00
LTEX	1,20	1,09	1,00	0,91	0,84	0,00
PCON	1,29	1,12	1,00	0,9	0,81	0,00

Nota: Adaptado de COCOMO: Un Modelo de Estimación de Proyectos de Software (p. 52), por A. Gómez, M. del C. López, S. Migani, & A. Otazú, s. f.

Factores del Proyecto

- **TOOL (Uso de Herramientas de Software):** Evalúa el uso de herramientas de software de Muy Bajo a Muy Alto.
- **SITE (Desarrollo Multisitio):** Refleja las condiciones de trabajo en un entorno multisitio, considerando ubicación y comunicación.
- **SCED (Cronograma Requerido):** Mide las restricciones de tiempo, representando un porcentaje de extensión o aceleración de plazos en relación al valor nominal, lo que afecta el esfuerzo requerido en distintas etapas del desarrollo.

A continuación, se presenta un resumen de los valores asociados a estos Factores del Proyecto que impactan en la estimación de costos. Desde el uso de herramientas de software hasta la gestión de plazos, estos factores son fundamentales en la planificación y gestión de proyectos de desarrollo de software.

Tabla 6

Valores de los Factores del Proyecto.

PARÁMETROS de Factores del Proyecto						
	Muy Bajo	Bajo	Normal	Alto	Muy Alto	Extra Alto
TOOL	1,17	1,09	1,00	0,9	0,78	0,00
SCED	1,43	1,14	1,00	1,00	1,00	0,00
SITE	1,22	1,09	1,00	0,93	0,86	0,8

Nota: Adaptado de COCOMO: Un Modelo de Estimación de Proyectos de Software (p. 52), por A. Gómez, M. del C. López, S. Migani, & A. Otazú, s. f.

2.8 Centralización, Gestión y Control de Documentación Digital

La centralización, gestión y control de documentación digital se refiere a la práctica de unificar y administrar de manera eficiente todos los documentos electrónicos de una organización. Esto implica asegurar un acceso controlado, un seguimiento detallado y una gestión eficaz de los archivos digitales, lo que es esencial para optimizar los flujos de trabajo y garantizar la integridad de los datos.

2.8.1 Importancia de la Gestión Documental

La gestión documental es esencial para cualquier empresa en la era digital. Un sistema de gestión documental digital simplifica procesos, mejora la eficiencia y evita la pérdida de datos. Las empresas, independientemente de su tamaño, manejan grandes volúmenes de documentos en papel y formato digital, y esta gestión eficiente de documentos agiliza el flujo de trabajo y mejora la productividad.

La transición hacia una gestión documental digital representa una decisión estratégica inteligente para las organizaciones, eliminando procesos manuales y flujos de trabajo engorrosos. La capacidad de buscar y acceder instantáneamente a la información en entornos digitales no solo ahorra tiempo y recursos, sino que también brinda una ventaja competitiva crucial al permitir decisiones informadas y oportunas, impulsando la eficiencia interna y

mejorando la capacidad de adaptación en un entorno empresarial caracterizado por la rapidez y la demanda constante de información.

“Para muchas empresas, el uso de un sistema de gestión documental digitales (DMS) supone una notable simplificación de sus procesos de trabajo cotidianos y una valiosa ayuda en el manejo de documentos importantes. Las empresas de todos los sectores procesan enormes volúmenes de documentos en formato digital, pero también en papel. De hecho, los trabajadores consumen una media de unas 10.000 hojas de papel al año. En una empresa mediana con 100 o incluso 500 empleados, son cantidades inimaginables de documentos que se mueven dentro de la empresa” (DocuWare, s. f.).

2.8.2 Beneficios de la Centralización de Documentos

La centralización de documentos es una estrategia clave en la gestión de información, ya que permite a las organizaciones, ya sean públicas o privadas, reunir y administrar sus documentos de manera más eficiente. Esta práctica proporciona numerosos beneficios que repercuten directamente en la optimización de procesos y en la toma de decisiones más fundamentadas. En el contexto de las instituciones públicas, la centralización de documentos es esencial para promover la transparencia, la eficiencia y el cumplimiento de regulaciones.

En el siguiente apartado, se profundizará en la identificación de los beneficios de la centralización de documentos, tal como se han identificado durante la elaboración de este proyecto. Se analizará detenidamente cómo esta estrategia influye en la gestión de documentos en el ámbito público, resaltando sus ventajas clave y su impacto en la eficiencia, transparencia y cumplimiento de regulaciones en el contexto gubernamental.

- **Acceso Rápido y Fácil:** La centralización de documentos permite un acceso rápido y sencillo a la información clave. Los usuarios pueden encontrar documentos en cuestión de segundos en lugar de perder tiempo buscando en archivos físicos.

- **Ahorro de Espacio:** Al almacenar documentos de forma digital, se elimina la necesidad de espacio físico para archivar papeles, lo que ahorra costos y reduce la necesidad de espacio de almacenamiento.
- **Reducción de Errores:** La automatización de procesos reduce los errores humanos al buscar, archivar o compartir documentos, lo que aumenta la precisión y la calidad del trabajo.
- **Seguridad Mejorada:** Los sistemas de gestión documental proporcionan opciones para controlar y restringir el acceso a documentos confidenciales, garantizando la seguridad de la información.
- **Colaboración Efectiva:** Varios usuarios pueden trabajar en un mismo documento de forma simultánea, facilitando la colaboración en tiempo real.
- **Control de Versiones:** Los sistemas de gestión documental mantienen un historial de versiones de documentos, lo que facilita el seguimiento de cambios y garantiza que siempre se utilice la versión más actualizada.
- **Cumplimiento Normativo:** Facilita el cumplimiento de regulaciones y estándares al garantizar que los documentos estén organizados y disponibles para auditorías y revisión.
- **Respaldo y Recuperación:** Se pueden realizar copias de seguridad regulares de documentos, lo que asegura que la información crítica esté protegida y disponible en caso de pérdida de datos.
- **Eficiencia en Procesos:** Los flujos de trabajo automatizados agilizan la aprobación y distribución de documentos, lo que mejora la eficiencia operativa.
- **Mejora la Productividad:** Los empleados pueden dedicar menos tiempo a buscar documentos y más tiempo a tareas productivas.

- **Reducción de Costos:** La eliminación del papel y la optimización de procesos reducen los costos asociados a la impresión y el almacenamiento físico.
- **Toma de Decisiones Informada:** La disponibilidad de información actualizada y relevante facilita la toma de decisiones informadas y estratégicas.
- **Integración con Sistemas:** La integración con otros sistemas permite la automatización de procesos y la sincronización de datos, lo que mejora la eficiencia.
- **Sostenibilidad:** La reducción del uso de papel y la optimización de recursos contribuyen a prácticas empresariales más sostenibles y respetuosas con el medio ambiente.

2.9 Seguridad y Pruebas de Software

En el presente proyecto se aborda dos pilares críticos del desarrollo de aplicaciones: garantizar la robustez y la integridad del software, y protegerlo contra posibles amenazas cibernéticas. En esta sección, se explora en profundidad cómo las pruebas de software son fundamentales para detectar y solucionar vulnerabilidades, y cómo la seguridad informática se integra en el ciclo de desarrollo. Se abordan prácticas recomendadas para lograr aplicaciones seguras y de calidad en aplicaciones web.

Este apartado servirá como un referente sólido para comprender la importancia de la seguridad y las pruebas de software en el desarrollo de proyectos, subrayando su relevancia en un mundo digital en constante evolución y enfocado en la satisfacción de las necesidades de los usuarios y la protección de datos e información crítica.

2.9.1 Seguridad y Validación de Datos en Aplicaciones Web

Validación de Datos de Entrada: La validación de datos de entrada en aplicaciones web es un componente esencial para garantizar la integridad y seguridad de los datos que los usuarios ingresan a través de formularios y campos de texto. En el contexto del proyecto, se implementó un sólido sistema de validación haciendo uso de la biblioteca Formik en el lado del

cliente (front-end). Formik es una herramienta eficaz que simplifica la validación de formularios en aplicaciones web, permitiendo establecer reglas y criterios específicos para cada campo de entrada.

En el front-end, se aprovechó las capacidades de Formik para definir expresiones regulares y reglas personalizadas que los usuarios deben cumplir al ingresar datos. Esto incluye validar direcciones de correo electrónico, contraseñas seguras, fechas válidas y otros criterios relevantes para mi aplicación. Estas validaciones en el front-end permiten ofrecer una experiencia de usuario mejorada al proporcionar retroalimentación inmediata sobre los datos ingresados.

Por otro lado, en el back-end, el sistema de validación se realiza con expresiones regulares para garantizar que los datos ingresados cumplan con los estándares de seguridad y calidad. Las expresiones regulares son especialmente útiles para verificar datos sensibles, evitando que se introduzcan caracteres maliciosos o inválidos que puedan comprometer la seguridad de la aplicación.

En resumen, la validación de datos de entrada en el proyecto se basa en la combinación de Formik, expresiones regulares y reglas personalizadas, tanto en el front-end como en el back-end. Esto asegura que los datos proporcionados por los usuarios sean precisos, seguros y cumplan con los estándares de calidad necesarios para el correcto funcionamiento de la aplicación. Estas prácticas de validación se alinean con las mejores prácticas de desarrollo de aplicaciones web y contribuyen a la creación de un sistema más robusto y seguro.

2.9.2 *Prevención de Ataques*

En el contexto del proyecto, se priorizó la seguridad de la aplicación. Para lograrlo, se implementó un sólido sistema de validación de datos de entrada a través de expresiones regulares. Este enfoque permite que los datos ingresados en los campos cumplan con

patrones especificados. Cualquier intento de introducir información no válida o potencialmente peligrosa se bloqueó de manera automática, lo que minimizó significativamente el riesgo de ataques de inyección de datos o código malicioso.

Además, se estableció una estructura de archivos que define rutas de manera clara y controlada. Esto es esencial para prevenir amenazas relacionadas con la manipulación de rutas que podrían dar lugar a intentos de acceso a recursos no autorizados. La creación de archivos específicos para establecer rutas contribuyó a mantener la integridad del sistema y a reducir las posibles vulnerabilidades.

En términos de autenticación y autorización de usuarios, se adoptó el uso de JSON Web Tokens (JWT). Esta elección brindó una capa adicional de seguridad al validar la identidad de los usuarios y gestionar sus sesiones de manera segura. Solo los usuarios autenticados y autorizados pueden acceder a áreas específicas de la aplicación, lo que previene ataques de suplantación de identidad y garantiza la seguridad de las funciones de autenticación.

Estas implementaciones son esenciales para fortalecer la seguridad del sistema y protegerlo contra amenazas cibernéticas. Al adoptar estas prácticas de prevención, se contribuye a que la aplicación cumpla con los estándares de seguridad recomendados en el desarrollo de software y ofrezca a los usuarios un entorno digital más seguro y confiable.

2.9.3 Políticas de Seguridad en Aplicaciones

En el desarrollo de la aplicación, se han establecido políticas de seguridad para garantizar la integridad, confidencialidad y disponibilidad de los datos, así como para proteger la aplicación contra posibles amenazas. Estas políticas abordan varios aspectos fundamentales:

Se han definido cuatro niveles de usuario dentro de la aplicación: revisor, usuario, coordinador y administrador. Cada nivel tiene sus propios permisos y roles específicos. Esto

se traduce en un control de acceso y acciones muy preciso. Los usuarios solo pueden realizar acciones para las cuales están autorizados, lo que garantiza la seguridad de los datos y la funcionalidad de la aplicación.

Para asegurar que cada usuario esté claramente identificado, se ha implementado una política que requiere que los nombres de usuario sean únicos en todo el sistema. Esto evita la creación de múltiples cuentas con el mismo nombre de usuario, lo que podría generar confusiones y problemas de seguridad.

La aplicación aplica validaciones rigurosas en todos los formularios y campos. Estas validaciones incluyen la verificación de usuarios válidas, contraseñas seguras y otros requisitos específicos. Estas medidas no solo garantizan que los datos sean coherentes y seguros, sino que también protegen contra posibles ataques.

Se realizan auditorías de seguridad regulares para evaluar posibles vulnerabilidades en la aplicación. Esto implica revisar la efectividad de las políticas de seguridad existentes y tomar medidas para abordar cualquier debilidad detectada. Estas auditorías son esenciales para garantizar que la aplicación siga siendo segura con el tiempo.

Estas políticas de seguridad desempeñan un papel crucial en la protección de la aplicación y la información que maneja. Garantizan que los niveles de usuario se gestionen adecuadamente, que los datos sean coherentes y seguros, y que se realicen auditorías periódicas para mantener un alto nivel de protección. Estas medidas son fundamentales para mantener la integridad y la confiabilidad de la aplicación en un entorno digital en constante evolución.

2.9.4 Ejemplos de Problemas Comunes en Seguridad de Aplicaciones

En el contexto de seguridad de aplicaciones, es crucial comprender y abordar los problemas comunes que pueden surgir durante el desarrollo y la implementación de una

aplicación. Aquí se presentan ejemplos de problemas comunes en seguridad de aplicaciones que es importante tener en cuenta:

- **Inyección de SQL:** Este problema ocurre cuando los atacantes insertan comandos SQL maliciosos en campos de entrada de una aplicación, lo que puede darles acceso no autorizado a la base de datos o permitirles manipular datos.
- **Cross-Site Scripting (XSS):** El XSS ocurre cuando se permite que los atacantes inyecten scripts maliciosos en el contenido web que se entrega a otros usuarios. Esto puede comprometer la seguridad de los datos y la privacidad de los usuarios.
- **Falta de autenticación y control de acceso:** Si una aplicación no gestiona adecuadamente la autenticación y el control de acceso, los atacantes pueden acceder a funciones o datos protegidos sin autorización.
- **Exposición de datos sensibles:** La exposición de datos sensibles ocurre cuando la aplicación almacena información confidencial, como contraseñas o datos de tarjetas de crédito, de manera insegura, lo que la hace accesible a personas no autorizadas.
- **Falta de actualizaciones de seguridad:** No mantener la aplicación y sus componentes actualizados con parches y actualizaciones de seguridad puede dejarla vulnerable a amenazas conocidas.
- **Falta de cifrado:** La falta de cifrado de datos durante la transferencia o el almacenamiento puede permitir que los atacantes intercepten y accedan a información confidencial.
- **Errores de configuración:** Configuraciones incorrectas o inseguras en servidores, bases de datos o en la propia aplicación pueden dejarla vulnerable a ataques.
- **Gestión inadecuada de sesiones:** No gestionar adecuadamente las sesiones de usuario puede permitir que los atacantes secuestren sesiones válidas y accedan a cuentas de usuario.

- **Divulgación de información detallada del error:** Revelar información detallada de errores en mensajes de error puede dar pistas a los atacantes sobre cómo explotar vulnerabilidades.

Es esencial abordar estos problemas comunes de seguridad durante el proceso de desarrollo y realizar pruebas rigurosas para identificar y mitigar las vulnerabilidades. Esto garantizará que la aplicación sea segura y protegerá tanto los datos de los usuarios como la integridad de la aplicación en sí.

2.9.5 Integración de la Seguridad en el Ciclo de Desarrollo

La integración de la seguridad en el ciclo de desarrollo es un enfoque esencial para garantizar que la seguridad esté presente en cada etapa del proceso de creación de una aplicación. Aquí se describen algunos aspectos clave de cómo se puede lograr esta integración:

- **Concientización en seguridad:** Desde el inicio del ciclo de desarrollo, es importante ser consciente de los aspectos de seguridad. Todos los profesionales involucrados deben comprender las amenazas y vulnerabilidades comunes, así como las mejores prácticas de seguridad.
- **Evaluación de riesgos:** Antes de comenzar a escribir código, se debe realizar una evaluación de riesgos. Esto implica identificar posibles amenazas, evaluar su impacto y probabilidad, y priorizar la mitigación de riesgos en función de los recursos disponibles.
- **Diseño seguro:** Durante la fase de diseño, se debe considerar la seguridad, esto incluye definir claramente los requisitos de seguridad, identificar y proteger activos críticos y definir los controles de seguridad necesarios.
- **Pruebas continuas:** A lo largo del ciclo de desarrollo, se deben realizar pruebas continuas de seguridad. Esto incluye, escaneo de vulnerabilidades y pruebas de

seguridad automatizadas. Estas pruebas ayudan a identificar y abordar las vulnerabilidades de manera efectiva.

- **Revisión de código seguro:** Se deben llevar a cabo revisiones de código enfocadas en la seguridad. Esto implica analizar el código en busca de vulnerabilidades conocidas y aplicar mejores prácticas de codificación segura.
- **Educación continua:** La educación y la capacitación en seguridad deben ser un componente continuo del ciclo de desarrollo. A medida que evolucionan las amenazas y las mejores prácticas, tanto para usuarios y desarrolladores.

La integración de la seguridad en el ciclo de desarrollo es un proceso continuo y multidisciplinario que requiere un enfoque proactivo y colaborativo. Al adoptar este enfoque, las instituciones pueden reducir significativamente el riesgo de vulnerabilidades de seguridad en sus aplicaciones y proteger la información crítica.

2.9.6 Pruebas de Software

2.9.6.1 Pruebas de Caja Blanca. Las pruebas de caja blanca, también llamadas pruebas estructurales o de caja clara, son un método de prueba de software en el que se evalúa el comportamiento interno de un programa o sistema. En este enfoque, se examina minuciosamente el código fuente y la lógica interna de la aplicación para detectar posibles errores y deficiencias. El principal objetivo al utilizar las pruebas de caja blanca es asegurarse de que el código esté correcto y eficiente, y de que la aplicación cumpla con las especificaciones y los requisitos que he establecido en el proyecto. Lo que hace especial a este enfoque es que brinda una visión completa de la estructura interna del software, lo que permite realizar una evaluación detallada de su funcionamiento y detectar cualquier posible vulnerabilidad que deba abordarse en el proyecto.

“Las pruebas de caja blanca en ingeniería de software pueden consistir en probar el código y el diseño interno del software para verificar el flujo de entrada-salida y comprobar el diseño, la usabilidad y la seguridad del software” (ZAPTEST, 2023a).

“El objetivo de las pruebas de caja blanca es realizar pruebas que cubran la estructura interna de un sistema, por estructura interna se refiere a código, arquitectura y flujos de trabajo. En las pruebas de caja blanca, el código es visible para los testers, por lo que también se denominan pruebas de caja transparente o pruebas de caja abierta” (Diario de QA, 2022).

2.9.6.2 Pruebas de Caja Negra. Las pruebas de caja negra son como si estuviera mirando una "caja negra" misteriosa. En lugar de tener que bucear en el código fuente del software, simplemente se concentra en lo que el software hace desde el exterior, es decir, en las entradas que se le proporcionan y las salidas que brinda. Esto permite verificar que el software cumple con las reglas y requisitos sin necesidad de entender su estructura interna.

“Las pruebas de caja negra consisten en probar un sistema o programa informático sin tener conocimiento previo de su funcionamiento interno. Esto no sólo se refiere a no conocer

el código fuente en sí, sino que implica no haber visto ninguna de las documentaciones de diseño que rodean al software. Los probadores se limitan a dar entrada y recibir salida como lo haría un usuario final. Aunque se trata de una simple definición de prueba de caja negra, establece el sistema general” (ZAPTEST, 2023b).

“Las técnicas de prueba son mecanismos que los testers utilizan con el objetivo de identificar los casos de prueba con mayor probabilidad de encontrar defectos y obtener la mayor cobertura posible en cuanto a las pruebas de algún sistema. Cuando hablamos de técnicas de caja negra, esto significa que al poner en práctica estas técnicas, solo nos concentraremos en los datos de entrada y los resultados obtenidos en la prueba, sin ver la estructura interna de la funcionalidad” (Diario de QA, 2022).

2.9.6.3 Pruebas de Estrés (Stress Testing)

La prueba de estrés se refiere a un proceso de evaluación que somete a una aplicación, sistema o componente a condiciones extremas, como cargas de trabajo intensas o recursos limitados, con el fin de medir su capacidad para funcionar bajo presión y su resistencia ante situaciones adversas. El objetivo principal de estas pruebas es identificar posibles debilidades, cuellos de botella y fallas, permitiendo a los evaluadores o desarrolladores optimizar el rendimiento y la estabilidad del sistema. Estas pruebas son esenciales para garantizar que la aplicación o sistema pueda operar de manera confiable y efectiva, incluso en circunstancias críticas, como picos de demanda o eventos inesperados.

“Las pruebas de estrés del software son un componente esencial del proceso de pruebas, y están diseñadas para identificar vulnerabilidades, puntos débiles y fallos potenciales que pueden producirse cuando un sistema se somete a una carga intensa o a condiciones adversas. Al simular un alto tráfico de usuarios, escasez de recursos y entradas de datos extremas, las pruebas de estrés pueden revelar información valiosa sobre el rendimiento de una aplicación” (ZAPTEST, 2023c).

2.10 Métricas de Calidad: Modelo FURPS

En el contexto de este proyecto, se hará uso de un enfoque de evaluación de la calidad de software conocido como el modelo FURPS. Este modelo se ha convertido en un marco ampliamente aceptado para evaluar y medir la calidad del software, centrándose en cinco características fundamentales que influyen en la satisfacción del usuario y el rendimiento del software.

Según (Solano & Torres, 2013), FURPS fue establecido tomando como la base el modelo de McCall, Hewlett-Packard lo desarrolla en 1987. Los factores de calidad que lo componen son los siguientes: funcionalidad (Functionality), usabilidad (Usability), confiabilidad (Reliability), desempeño (Performance) y capacidad de soporte (Supportability). Formando un acrónimo de donde proviene su nombre. Se divide en dos grupos:

2.10.1 *Requerimientos Funcionales (F)*

Debe ser capaz de realizarse sin tomar en cuenta restricciones físicas, y se definen a través de las entradas y salidas esperadas.

- **Funcionalidad:** Esta dimensión se centra en la capacidad del software para cumplir con los requisitos funcionales específicos, asegurando que realice de manera efectiva las tareas y funciones para las que fue diseñado, lo que es esencial para su utilidad y efectividad.

2.10.2 *Requerimientos no Funcionales (URPS)*

Describen atributos del sistema o del ambiente.

- **Usabilidad:** Se refiere a la facilidad de interacción de los usuarios con el software, evaluando la amigabilidad de la interfaz, la navegación intuitiva y la eficiencia en las tareas. Una alta usabilidad mejora la satisfacción y la experiencia del usuario.

- **Confiabilidad:** Se refiere a la capacidad del software de funcionar de manera consistente y sin fallos, garantizando su disponibilidad y operación correcta en todo momento al minimizar errores y bloqueos.
- **Rendimiento:** Se enfoca en la velocidad y eficiencia del software, evaluando su capacidad para responder rápidamente a las solicitudes del usuario, manejar la carga de trabajo y utilizar eficazmente los recursos del sistema.
- **Soporte:** La dimensión de soporte abarca servicios y recursos para el mantenimiento y mejora del software, como documentación, soporte técnico, actualizaciones y personalización.

La utilización del modelo FURPS en este proyecto se fundamenta en su capacidad para proporcionar una evaluación integral de la calidad del software, considerando múltiples dimensiones que impactan directamente en la satisfacción del usuario y el éxito del proyecto. Esta elección de enfoque fortalece la base teórica y metodológica del proyecto, asegurando una evaluación rigurosa y precisa de la calidad del software desarrollado.

2.11 Métodos para la recolección de información

2.11.1 Encuestas

“Una encuesta es una serie de preguntas dirigidas a los participantes en la investigación. Las encuestas pueden ser administradas en persona, por correo, teléfono o electrónicamente (como correo electrónico o en Internet). También pueden administrarse a un individuo o a un grupo. Las encuestas son utilizadas para tener información sobre muchas personas y pueden incluir elección múltiple/forzada o preguntas abiertas (como información demográfica, salud, conocimiento, opiniones, creencias, actitudes o habilidades)” (OriHhs, s. f.).

Se realizaron encuestas a los funcionarios de la Unidad de Registros para analizar el flujo de la información, el procedimiento de recepción de la documentación, y los datos que los

funcionarios tienen que llenar previo a la contratación. Estos datos sirvieron de base para diseñar las interfaces del sistema web de información.

2.11.2 Entrevista

“Una entrevista es una interacción que involucra al investigador y a un(os) participante(s) en que las preguntas se formulan en persona, por teléfono o incluso de manera electrónica (correo electrónico o Internet). Durante una entrevista, se hacen preguntas para obtener información detallada sobre el participante acerca del tema en estudio. Las preguntas pueden ser similares a las formuladas en una encuesta” (OriHhs, s. f.).

Ya que es un sistema que actualmente no hay antecedentes previos en la institución, se tienen que realizar constantes entrevistas al usuario que manipule el mismo, identificando la necesidad, requerimientos, etc. Para el desarrollo de un sistema que mitigue la problemática actual de la institución.

2.11.3 Observaciones

“Las observaciones son registros tomados que no requieren participación. Estos registros se hacen mientras los participantes están involucrados en conductas rutinarias y se utilizan como un indicador de lo que los participantes de hecho hacen, en lugar de apoyarse completamente en los relatos que los participantes hacen de su propia conducta. Un ejemplo sería un investigador observando los planes educativos usados en un aula por un maestro de escuela pública” (OriHhs, s. f.).

Se realizaron observaciones de como los funcionarios previa contratación adjuntan sus documentos personales. Una vez presentada la documentación, ese analizó el flujo que sigue, el lugar donde es almacenada y si los documentos tienen distintivos o códigos. Todo esto para diseñar un sistema funcional que cumpla con los requerimientos establecidos.

2.11.4 Revisión de registros

“La revisión de registros tiene lugar cuando un investigador examina y extrae información de documentos que contienen datos sobre el participante. Los registros revisados en una investigación pueden ser públicos o privados. Ejemplo de ello es un investigador recolectando información acerca de un padecimiento a partir de los historiales médicos de los pacientes” (OriHhs, s. f.).

Es necesario la revisión de la documentación o plantillas de los formularios, ya que indicarán como se llenan los formularios, el orden cronológico del mismo, además de ayudar en la estructuración de la base de datos mediante los datos registrados, e indicar los documentos que necesitan respaldo, para solicitar la misma información en el sistema web.

2.12 Beneficios del Sistema

El sistema que se presenta en este proyecto de grado brinda una serie de beneficios significativos que mejoran la gestión de la documentación y las condiciones de trabajo de los funcionarios. A continuación, se detallan algunos de los beneficios clave que se derivan de la implementación exitosa de este sistema:

- **Mayor Eficiencia en la Gestión Documental:** Este sistema desempeña un papel fundamental al evitar la pérdida de archivos personales y al centralizar la documentación en una base de datos accesible. Los funcionarios pueden contar con una herramienta eficaz para el seguimiento y la gestión de sus documentos, lo que simplifica sus tareas diarias.
- **Acceso Rápido a la Información:** La aplicación web proporciona un acceso rápido a la información almacenada en el sistema. Esto es especialmente beneficioso para los funcionarios que necesitan acceder a su documentación para realizar descargos o consultas relacionadas con la documentación presentada.

- **Mejora de las Condiciones de Trabajo:** La implementación de este proyecto ha tenido un impacto positivo en las condiciones de trabajo de los funcionarios de la Unidad de Registros. La herramienta agiliza el proceso de recepción y seguimiento de los documentos, lo que se traduce en un entorno laboral más eficiente y satisfactorio.
- **Acceso Remoto y Personalizado:** Dado que se trata de una aplicación web, los funcionarios pueden acceder al sistema desde cualquier equipo con conexión a Internet. Cada funcionario tiene credenciales de acceso que le permiten ingresar a su información personal y verificar los datos registrados en el sistema. Esto facilita la gestión de su documentación desde ubicaciones diversas.
- **Reducción de Tiempos de Seguimiento:** El sistema ha demostrado ser eficaz en la reducción de los tiempos necesarios para realizar seguimientos de documentos personales presentados. La información se almacena y actualiza de manera centralizada, lo que simplifica la monitorización de los procesos y la revisión de la documentación.

En resumen, la implementación de este sistema proporciona una plataforma valiosa para la gestión de documentos y contribuye significativamente a mejorar la eficiencia y la comodidad de los funcionarios. Además, ofrece un acceso rápido y personalizado a la información y simplifica la labor de seguimiento, lo que impacta positivamente en la Unidad de Registros y en el funcionamiento general de la institución. Estos beneficios son una clara evidencia del valor añadido de este sistema en el contexto de la gestión documental.

2.13 Conclusiones del Marco Teórico

El análisis del Marco Teórico me permitió comprender que el desarrollo de proyectos de software es un desafío multifacético que abarca diversas habilidades, modelos y metodologías. A través de esta revisión, he podido apreciar la complejidad inherente en el proceso y la necesidad de un enfoque integral que implica la cuidadosa selección de

estrategias en cada etapa del ciclo de desarrollo. Sin embargo, es importante destacar que la implementación práctica de estos enfoques puede ser un desafío en sí mismo.

En la actualidad, en la era digital, es evidente que los proyectos de software desempeñan un papel vital en la transformación de las organizaciones y la prestación de servicios. La importancia de la calidad y eficacia en el desarrollo de software se destaca, y reconozco su relevancia en la ejecución exitosa del presente proyecto de grado.



CAPITULO III

MARCO APLICATIVO



**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO

3. Capítulo 3: Marco Aplicativo

3.1 Introducción

Este capítulo tiene como objetivo presentar la planificación, análisis y diseño del "Sistema de Información Web para la Gestión y Control de Documentación Personal" en el contexto de la Unidad de Registro, que opera bajo la Dirección de Talento Humano en el Gobierno Autónomo Municipal de El Alto. Este proyecto se ha concebido como una respuesta a los desafíos que enfrenta la Unidad en la administración de documentos y registros diarios.

En el desarrollo de este sistema, se ha optado por una combinación de enfoques metodológicos para garantizar un enfoque integral y eficiente. Se ha elegido la metodología ágil OpenUP, que se centra en la colaboración en equipo, la entrega iterativa e incremental, y la retroalimentación continua del equipo. A esto se agrega el patrón de arquitectura por capas, que divide el sistema en componentes con funciones específicas y una comunicación bien definida.

Para la gestión de la interfaz de usuario y la experiencia del usuario, se utilizó React, una biblioteca de JavaScript que permite crear interfaces interactivas. Además, se aplicó Tailwind CSS para dar estilo a las interfaces de usuario, logrando así un diseño atractivo y funcional.

El lado del servidor se apoya en NodeJS, que ejecuta código JavaScript de manera eficiente. La base de datos se basa en MongoDB, una base de datos NoSQL que ofrece escalabilidad y flexibilidad. La elección de estas tecnologías se ha basado en la necesidad de un sistema ágil y receptivo a las demandas cambiantes. Finalmente, se utilizó el método COCOMO II para la estimación de costos del software.

Este proyecto se presenta como una solución innovadora para la Unidad de Registro, permitiéndoles superar los desafíos actuales relacionados con la organización y gestión de la documentación personal. Al adoptar un enfoque moderno y tecnológico, se espera que el

sistema proporcione un control efectivo, un acceso oportuno y una gestión más eficiente de la información, contribuyendo así a los objetivos organizativos del Gobierno Autónomo Municipal de El Alto.

La implementación de la metodología OpenUP se basa en la necesidad de adoptar un enfoque estructurado y adaptable para el desarrollo del proyecto. Esperando que el uso de OpenUP proporcione la estructura y la flexibilidad necesarias para lograr una gestión efectiva y alcanzar los objetivos establecidos.

3.2 Aplicación de la metodología OpenUp

3.2.1 Primera Iteración

3.2.1.1 Fase de Inicio. Esta fase en la primera iteración de la metodología OpenUP es el punto de partida. Aquí, se comprenden las necesidades del proyecto, se definen objetivos concretos y se planifica el cronograma. Estos pasos iniciales establecen una base sólida para el éxito continuo del proyecto.

3.2.1.1.1 Comprensión de las Necesidades del Proyecto. En esta fase crucial, se llevaron a cabo una serie de actividades con el objetivo de comprender en profundidad las necesidades del proyecto y las expectativas de los usuarios. Las actividades clave incluyeron:

- **Reuniones con los Encargados del Sistema y Usuarios:** Se realizaron reuniones tanto con los encargados de gestionar el sistema como con los usuarios finales. Estas reuniones proporcionaron una oportunidad para obtener información valiosa sobre las necesidades y requisitos específicos del proyecto.
- **Entrevistas Personales y Cuestionarios Verbales:** Se llevaron a cabo entrevistas personales con los interesados clave y se utilizaron cuestionarios verbales para recopilar información detallada sobre lo que se esperaba de la aplicación.

- **Apuntes y Registro de Puntos Sobresalientes:** Durante estas interacciones, se tomaron apuntes detallados y se registraron los aspectos más destacados de las discusiones. Esto ayudó a capturar y sintetizar los requerimientos más importantes.
- **Visita a la Unidad de Archivos:** Se realizó una visita a la Unidad de Archivos donde se almacena la documentación relevante. Esta visita permitió obtener información adicional y acceder a documentos necesarios para el proyecto.

Como resultado de estas actividades, se obtuvo una comprensión sólida de lo que se requería para el proyecto. Los usuarios expresaron su necesidad de una aplicación web que consta de 9 formularios de registro personal, que incluyen:

- **Registros Personales:** Este formulario recopila información personal clave, como nombre completo, lugar de nacimiento, tipo de documento, y permite la carga de una foto y un archivo PDF de respaldo.
- **Registros Complementarios:** Aquí se recopilan datos adicionales, incluyendo dirección de domicilio, información de contacto, servicio militar, licencia de conducir y referencias familiares.
- **Registros Académicos:** Este formulario permite a los usuarios registrar su nivel académico, idioma nativo y detalles de cursos de funcionario público obligatorios.
- **Estudios Cursados:** En este formulario, los usuarios ingresan información detallada sobre los títulos obtenidos, idiomas cursados y certificaciones de cursos completados.
- **Datos de Seguro y Declaraciones Juradas:** Este formulario incluye información sobre seguros de salud y declaraciones realizadas en contraloría.
- **Documentos Institucionales:** Aquí se registran documentos institucionales, como declaraciones juradas de domicilio, formularios AVC, y otras documentaciones similares.

- **Experiencia Laboral:** Los usuarios detallan su historial laboral, especificando si trabajaron en la institución, en lugares externos u otros.
- **Dependientes:** En este formulario, se registra información sobre los hijos o familiares dependientes de los usuarios.
- **Datos de Discapacidad:** Los usuarios pueden indicar si tienen familiares o si ellos mismos se encuentran en estado de discapacidad.

Además de estos formularios, se identificaron requerimientos adicionales:

- **Módulo de Revisor:** Se requiere un módulo especializado que permitirá a los revisores verificar y validar la información registrada por los usuarios. Este módulo será esencial para garantizar la precisión y calidad de los datos.
- **Funcionalidad de Login y Registro:** Deberá implementarse una funcionalidad de inicio de sesión y registro para que los usuarios, revisores puedan acceder a sus perfiles y realizar los registros necesarios.
- **Generación de Certificaciones:** Se debe desarrollar una funcionalidad que permita a los usuarios generar certificaciones personalizadas basadas en la información que han registrado. Esto será fundamental para proporcionar a los usuarios documentación oficial de su información.

Estos requerimientos adicionales se suman a la comprensión sólida de las necesidades del proyecto y proporcionan una base completa y detallada para el diseño y desarrollo subsiguientes de la aplicación web. La fase de comprensión de las necesidades del proyecto ha sido fundamental para establecer una visión clara y completa del proyecto.

3.2.1.1.2 Definición de Objetivos. A continuación, en base a "Comprensión de las Necesidades del Proyecto", se presentan los objetivos clave que se buscan alcanzar para desarrollar un sistema prometedor. Estos objetivos servirán como la brújula para guiar el

proyecto y garantizar que esté alineado con las necesidades de los usuarios y los requisitos de la institución.

- **Automatización de Registro:** El objetivo fundamental es automatizar y simplificar el proceso de registro de información personal, eliminando la necesidad de registros en papel o procesos manuales.
- **Precisión y Veracidad de Datos:** Se busca asegurar que la información registrada sea precisa y veraz, minimizando errores y duplicación de datos.
- **Seguridad de la Información:** El objetivo es garantizar la seguridad de la información personal registrada, protegiéndola contra accesos no autorizados o filtraciones.
- **Facilitar la Administración:** Se pretende simplificar la administración de registros y la gestión de datos para el personal encargado.
- **Accesibilidad y Usabilidad:** El objetivo es crear una aplicación web accesible y fácil de usar para los usuarios finales, independientemente de sus habilidades técnicas.
- **Eficiencia y Ahorro de Tiempo:** Se busca reducir el tiempo y los recursos necesarios para el registro y la gestión de información personal.
- **Cumplimiento Legal:** El objetivo es asegurar que la aplicación cumple con todas las regulaciones y leyes relacionadas con la privacidad y la gestión de datos personales.
- **Documentación Oficial:** Se busca permitir a los usuarios generar documentación oficial y certificaciones basadas en la información registrada.
- **Colaboración y Verificación:** El objetivo es facilitar la colaboración entre usuarios y revisores para garantizar la precisión de los registros.
- **Rendimiento y Escalabilidad:** Se pretende garantizar que la aplicación web funcione de manera eficiente, incluso a medida que la cantidad de usuarios y registros aumenta con el tiempo.

Estos objetivos proporcionan una dirección clara para el proyecto y son la base sobre la cual se desarrollarán las siguientes fases del ciclo de vida del proyecto.

Este enfoque en los objetivos asegura que el proyecto se alinee con las necesidades de los usuarios y las metas de la Unidad de Registros, lo que es esencial para el éxito a largo plazo del proyecto.

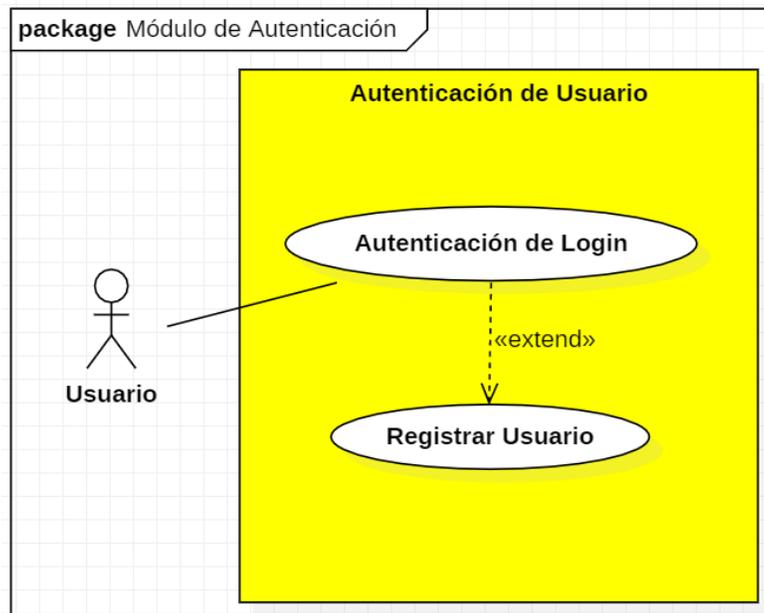
3.2.1.2 Fase de Elaboración. Antes de pasar al proceso de codificación, es esencial sentar las bases adecuadas para el proyecto. Esto implica comprender a fondo los requisitos, crear una estructura de navegación clara y plasmar visualmente los conceptos con Balsamiq Mockup para su validación. Esta fase es fundamental para asegurar un camino exitoso en la construcción y posterior implementación del proyecto.

3.2.1.2.1 Diseño y Detallado de Requerimientos. Es la parte del desarrollo de software donde se toman los requisitos previamente identificados y se definen de manera más detallada. Se elaboran diagramas para especificar cómo se implementarán los requisitos. Durante esta etapa, se profundiza en los requisitos funcionales y no funcionales, se crean diagramas para visualizar la interacción con el sistema y se definen detalles técnicos. El objetivo es transformar los requisitos iniciales en especificaciones detalladas que guiarán la implementación del sistema.

- **Requerimientos Funcionales:** Se refieren a las especificaciones detalladas que describen las funciones, características y comportamientos que nuestro sistema debe cumplir para satisfacer las necesidades de nuestros usuarios. Estos requisitos funcionales son vitales para definir las capacidades y el comportamiento del sistema. A continuación, se detallarán los diagramas de casos de uso que visualizarán de manera efectiva cómo se implementarán y cumplirán estos requisitos en el proyecto. Posteriormente, se detallarán las tablas de descripción de cada caso de uso para una comprensión más completa y detallada de la funcionalidad del sistema.

Figura 6

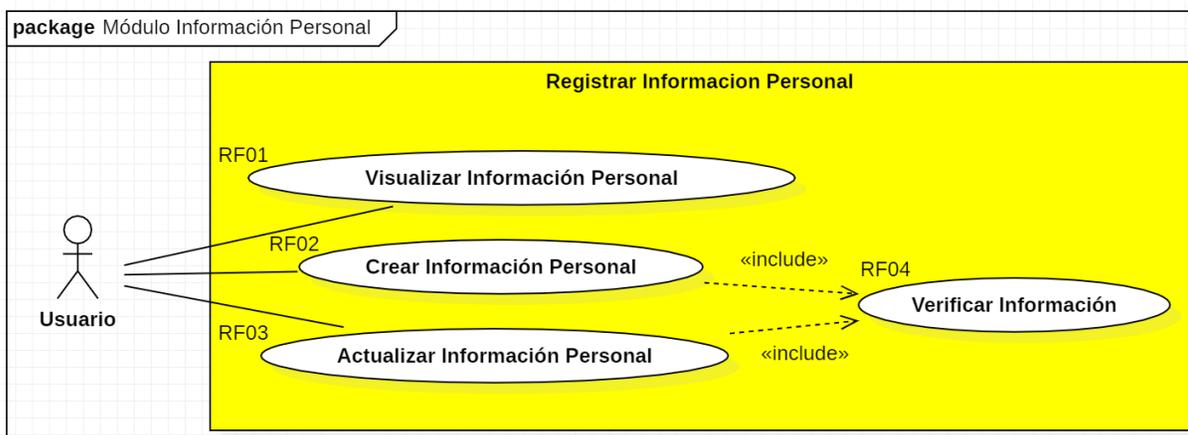
Diagrama de Caso de Uso: Autenticación del Usuario



Nota: En el diagrama se representa el módulo de autenticación y registro del usuario.

Figura 7

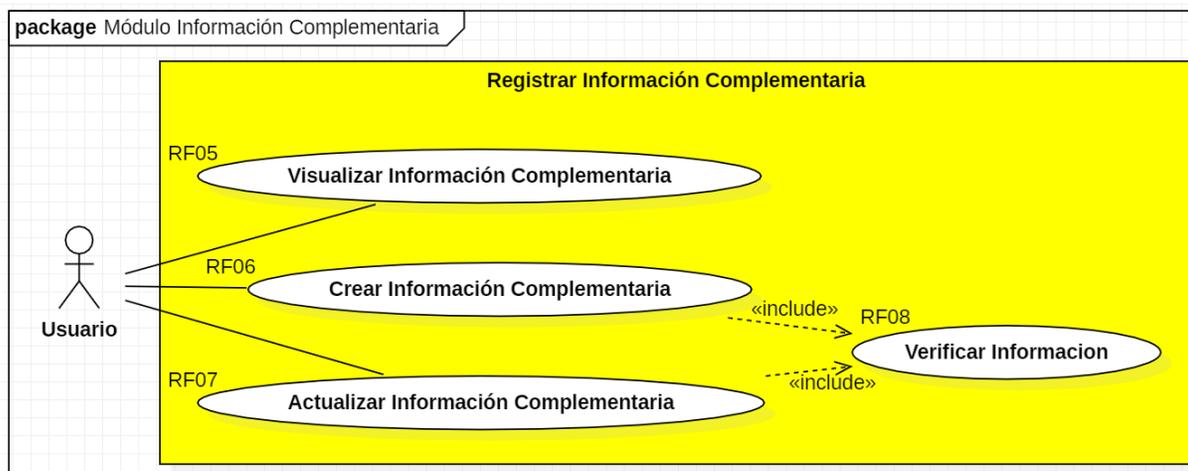
Diagrama de Caso de Uso: Registrar Información Personal



Nota: En el diagrama se representa el proceso de Visualizar, Crear y Actualizar la Información Personal del Usuario.

Figura 8

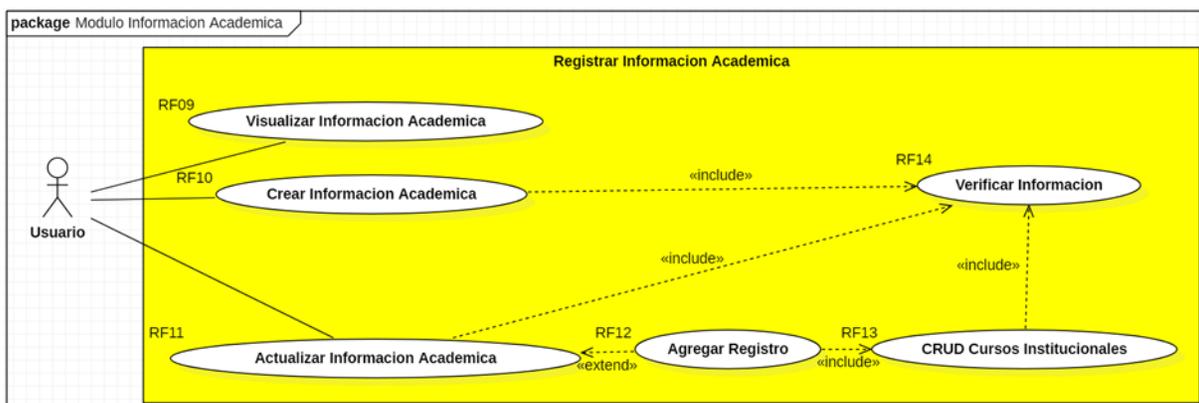
Diagrama de Caso de Uso: Registrar Información Complementaria



Nota: En el diagrama se representa el proceso de Visualizar, Crear y Actualizar la Información Complementaria del Usuario.

Figura 9

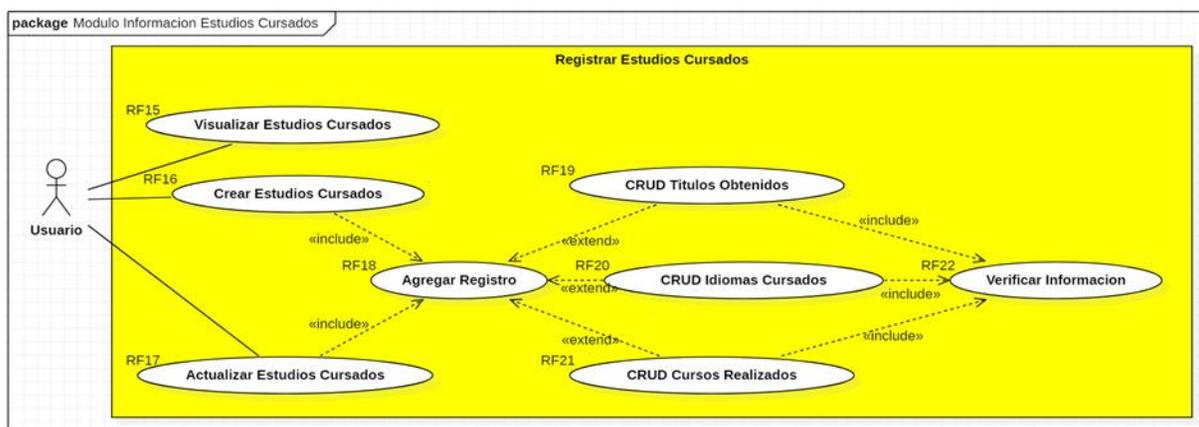
Diagrama de Caso de Uso: Registrar Información Académica



Nota: En el diagrama se representa el proceso de Visualizar, Crear y Actualizar la Información Académica, junto con un CRUD de los Cursos Institucionales realizados por el usuario.

Figura 10

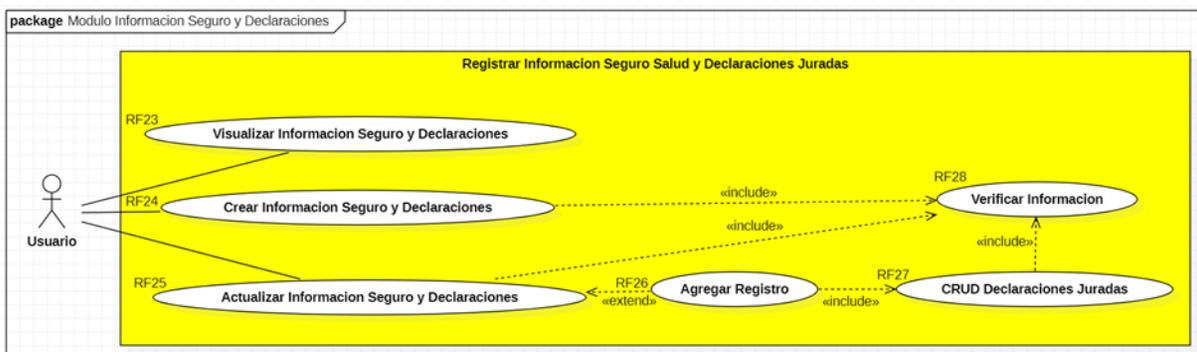
Diagrama de Caso de Uso: Registrar los Estudios Realizados



Nota: El diagrama representa el proceso de Visualización, Creación y Actualización de la Información de Estudios Realizados. Este proceso solo se inicia si existe un registro previamente agregado, el cual puede pertenecer a uno de los tres CRUDs disponibles: Títulos Obtenidos, Idiomas Cursados y Cursos Realizados.

Figura 11

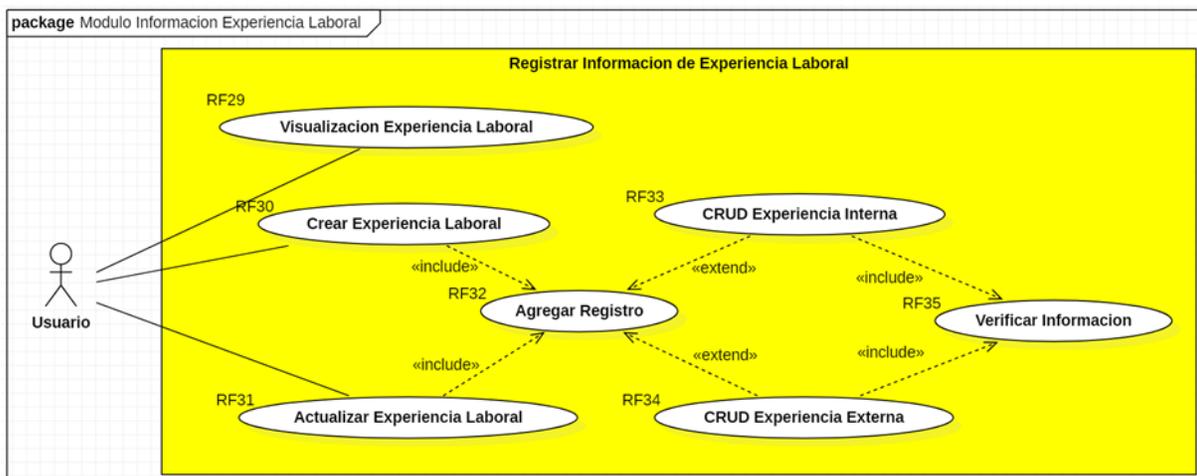
Diagrama de Caso de Uso: Registrar Seguro de Salud y Declaraciones Juradas



Nota: El diagrama ilustra el proceso de Visualización, Creación y Actualización de la Información de Seguro de Salud y Declaraciones Juradas. Además, se puede agregar un registro en el CRUD Declaración Jurada en caso de ser necesario.

Figura 12

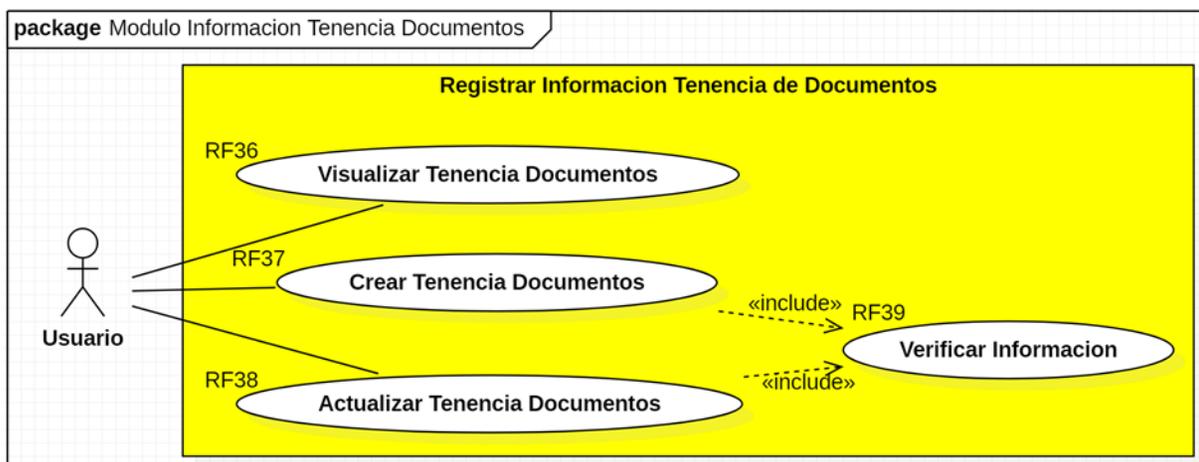
Diagrama de Caso de Uso: Registrar Información de Experiencia Laboral



Nota: El diagrama representa el proceso de Visualización, Creación y Actualización de la Información de Experiencia Laboral. Este proceso solo se inicia si existe un registro previamente agregado, el cual puede pertenecer a uno de los dos CRUDs disponibles: Experiencia Interna, Experiencia Externa.

Figura 13

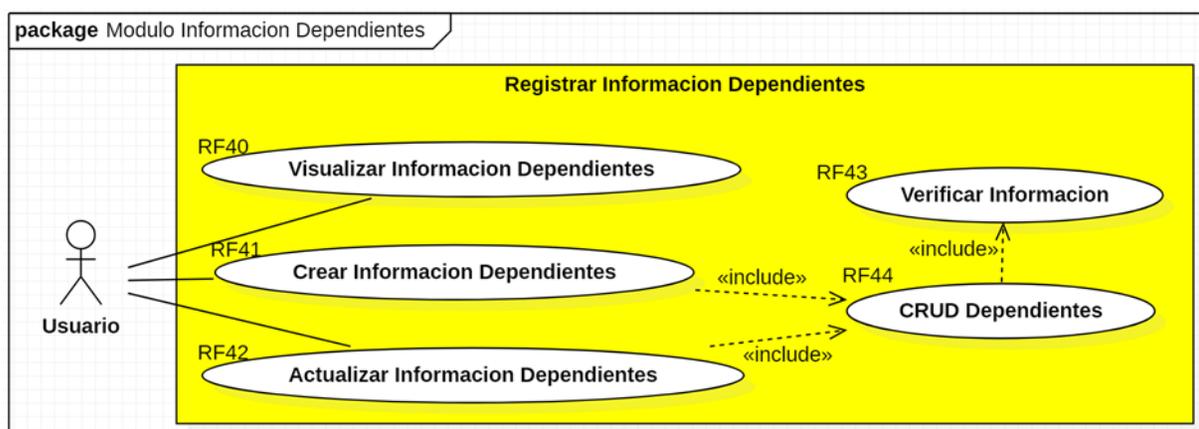
Diagrama de Caso de Uso: Registrar Cumplimiento de Documentación



Nota: En el diagrama se representa el proceso de Visualizar, Crear y Actualizar la Información de Cumplimiento de Documentación.

Figura 14

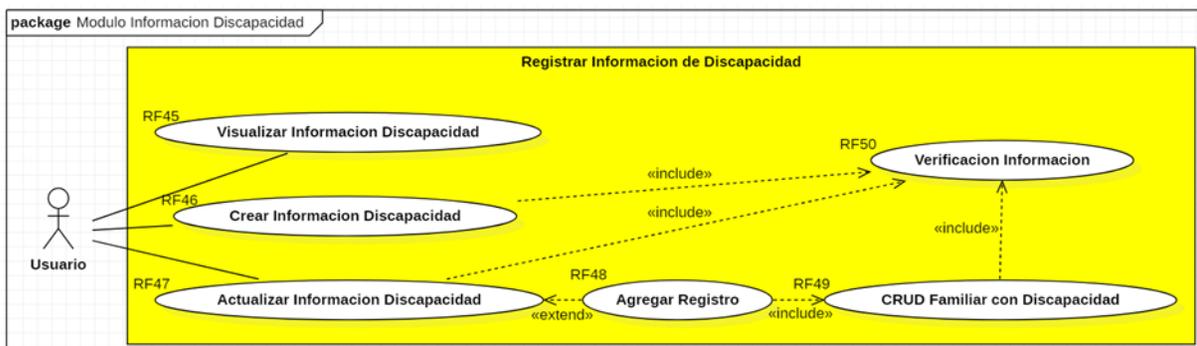
Diagrama de Caso de Uso: Registrar Información de Dependientes



Nota: El diagrama representa el proceso de Visualización, Creación y Actualización de la Información de los dependientes del usuario. Este proceso solo se inicia si existe un registro previamente agregado, el cual puede pertenecer al CRUD Dependientes.

Figura 15

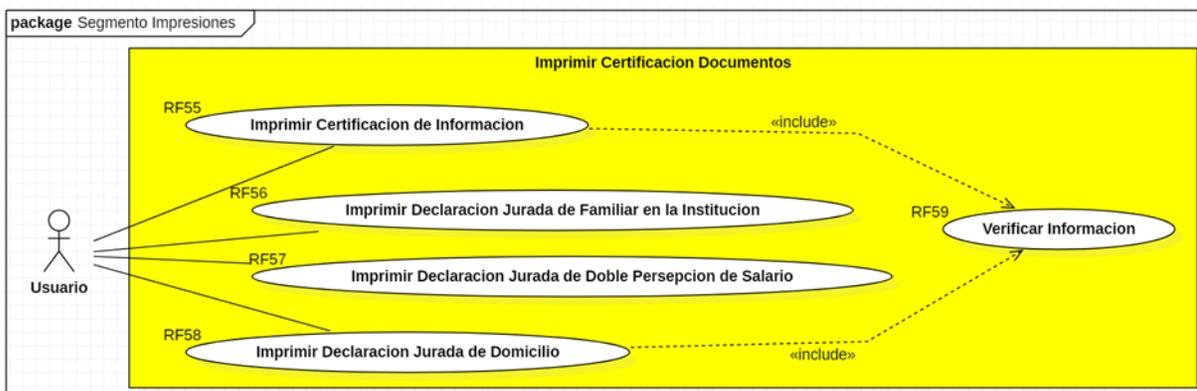
Diagrama de Caso de Uso: Registrar Información de Discapacidad



Nota: El diagrama ilustra el proceso de Visualización, Creación y Actualización de la Información de Discapacidad. Además, se puede agregar un registro en el CRUD Familiar con Discapacidad en caso de ser necesario.

Figura 16

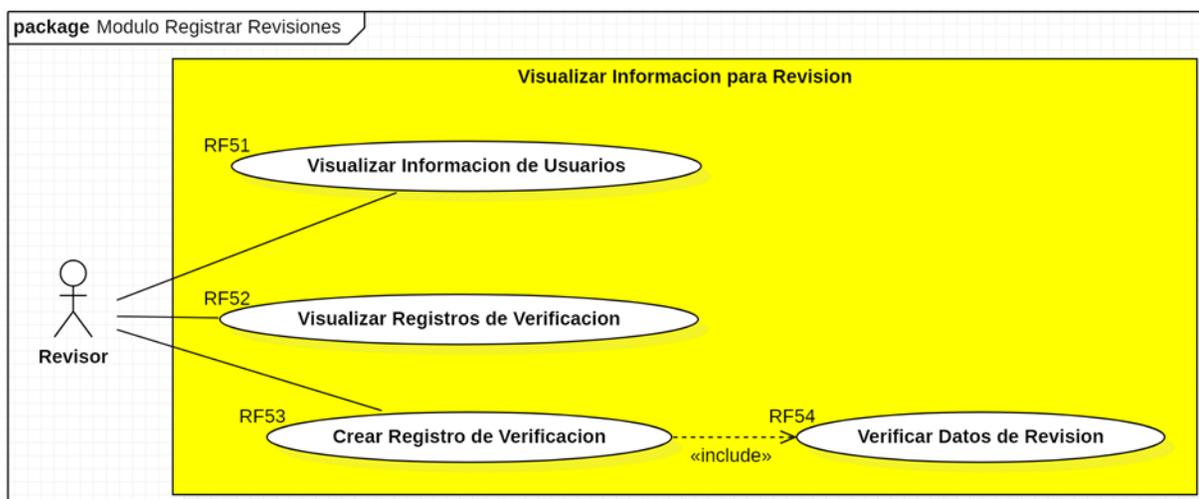
Diagrama de Caso de Uso: Impresión de Documentación



Nota: El diagrama ilustra el proceso de documentación que el sistema debe emitir a solicitud del usuario.

Figura 17

Diagrama de Caso de Uso: Revisión de Información



Nota: El diagrama ilustra el proceso en el cual el sistema puede visualizar la información del usuario, visualizar registros de verificaciones anteriores y crear nuevos registros de verificación con los parámetros proporcionados por el funcionario revisor.

Tabla 7

Descripción de los Casos de Uso de Visualización de de Información.

Nombre del Caso de Uso:	Visualizar Informacion	
Requerimientos:	RF01, RF05, RF09, RF15, RF23, RF29, RF36, RF40, RF45, RF51	
Actores:	Usuario - Revisores	
Pre-Condicion:	El Usuario tiene que estar Autenticado	
Flujo Basico:	Usuario - Revisores	Sistema
Visualizar U1	> Seleccionar la sección Formularios	
U2	> Seleccionar el Formulario	
U3		> Cargar información del Usuario Logueado
U4		> Despliegue de la tarjeta de Información
Visualizar R1	> Seleccionar la sección Revisión	
R2	> Ingresar el CI del usuario a revisar	
R3		> Cargar información del usuario ingresado para la búsqueda
R4		> Depliegue del formulario toggle de informacion
R5	> Presionar el button toggle para ver la Informacion de interes	
R6		> Sistema muestra el card de información
Exepcion:	Usuario - Revisores	Sistema
EXC: U	Usuario Logueado sin informacion.	Sin carga de Datos
EXC: R	Revisor ingreso datos Erroneos	Sin información de la búsqueda ingresada

Post-Condicion: Se verifica la informacion durante la visualizacion de la misma.

Nota: Se ilustra los actores y los pasos involucrados en el proceso de visualización de datos, facilitando así su análisis e interpretación.

Tabla 8

Descripción de los Casos de Uso de Crear Información.

Nombre del Caso de Uso:	Crear Información	
Requerimientos:	RF02, RF06, RF10, RF16, RF24, RF30, RF37, RF41, RF46, RF53, RF61.	
Actores:	Usuario	
Pre-Condición:	El Usuario tiene que estar Autenticado	
Flujo Básico:	Usuario	Sistema
CREAR U1	> Seleccionar la sección Formularios.	
U2	> Seleccionar el formulario de registro.	
U3	> Verificar que la Información personal se encuentre registrada.	
U4	> Completar el formulario de registro y presionar registrar.	
U5		> Proceso de creación del registro.
U6		> Respuesta al usuario de creación exitosa y envió de datos para la visualización.
U7	> Alerta de registro exitosa o duplicidad de datos.	
U8		> Carga de la información en la tarjeta de visualización del Frontend.

Excepción:	Usuario	Sistema
EXC: U1	Registro Creado Previamente.	Envío de Alerta de Error. Devolución de errores
EXC: U2	Datos ingresados erróneos.	controlados por campos inválidos.
Post-Condición:	Verificar Información Registrada	

Nota: Esta figura detalla los procedimientos y actores esenciales involucrados en la creación y manejo de información.

Tabla 9

Descripción de los Casos de Uso de Actualizar Información.

Nombre del Caso de Uso:	Actualizar Información	
Requerimientos:	RF03, RF07, RF11, RF17, RF25, RF31, RF38, RF42, RF47.	
Actores:	Usuario	
Pre-Condición:	El Usuario tiene que estar Autenticado y Registro previamente creado.	
Flujo Básico:	Usuario	Sistema
ACTUALIZAR U1	> Seleccionar la sección Formularios.	
U2	> Seleccionar el formulario de registro.	
U3	> Seleccionar la opción de Edición de Registro.	
U4	> Modificar campos y enviar solicitud.	
U5	> Proceso de actualización del registro.	
U6	> Respuesta al usuario de actualización exitosa y envío de datos para la visualización.	

U7	> Alerta de actualización de registro.	
U8		> Carga de la información en la tarjeta de visualización del Frontend.
Excepción:	Usuario	Sistema
EXC: U1	Registro no autorizado para modificar.	Envío de Alerta de Error.
EXC: U2	Actualizaciones no permitidas.	Devolución de errores controlados por campos inválidos.
Post-Condición:	Verificar Información Registrada	

Nota: Esta figura describe los procesos y actores clave necesarios para la actualización de información.

Tabla 10

Descripción de los Casos de Uso de Verificación de la Información.

Nombre del Caso de Uso:	Verificar Información	
Requerimientos:	RF04, RF08, RF14, RF22, RF28, RF35, RF39, RF43, RF50, RF54, RF59.	
Actores:	Usuario - Revisores	
Pre-Condición:	El Usuario tiene que estar Autenticado.	
Flujo Básico:	Usuario	Sistema
VERIFICAR U1	> Ingreso de Datos.	
U2	> Validación de datos en Tiempo Real.	
U3	> Leer indicadores que muestran Error/Éxito.	
U4	> Validación al enviar.	
U5		> Verificación de los campos, tanto en Fronted o Backend.

U6		> Manejo de posibles errores.
U7	> Alerta de posibles errores en el envío de datos.	
Excepción:	Usuario	Sistema
EXC: U1	Ingreso de datos inválidos.	Datos validos no contemplados.
EXC: U2	Condiciones no comprendidas.	Mensajes de Alertas erróneos.
Post-Condición:	Comprender las alertas emergentes.	

Nota: Esta figura describe el proceso de la verificación de la Información.

Tabla 11

Descripción de los Casos de Uso de los CRUDs de Información Adicional.

Nombre del Caso de Uso:	CRUD de Información Adicional	
Requerimientos:	RF13, RF19, RF20, RF21, RF27, RF33, RF34, RF44, RF49	
Actores:	Usuario	
Pre-Condición:	El Usuario tiene que estar Autenticado.	
Flujo Básico:	Usuario	Sistema
CREATE C1	> Ingreso de Datos solicitados.	
C2		Validar los datos del formulario.
C3	> Leer alertas de campos inválidos.	
U4	> Enviar solicitud de registro.	
U5		> Registrar datos del formulario seleccionado.
READ R1	> Ingresar en la sección Formulario. > Seleccionar la información a visualizar.	> Establecer la tarjeta de visualización de Información.

UPDATE U1	> Seleccionar la sección Formulario.	
U2	> Seleccionar el formulario a editar.	
U3	> Ingresar Cambios.	
U4		> Validar cambios.
U5		> Registrar cambios.
DELETE D1	> Seleccionar la sección Formulario.	
D2	> Seleccionar el registro a eliminar.	
D3	> Eliminar registro.	
D4		Validar cambios realizados.
D5		Registrar cambios.
Excepción:	Usuario	Sistema
EXC: U1		Registros no se pueden eliminar.
Post-Condición:	Se realizan los cambios y luego se confirman los mismos.	

Nota: Esta figura describe el proceso de la Agregación de Información Adicional que es un CRUD lógico en todos los casos.

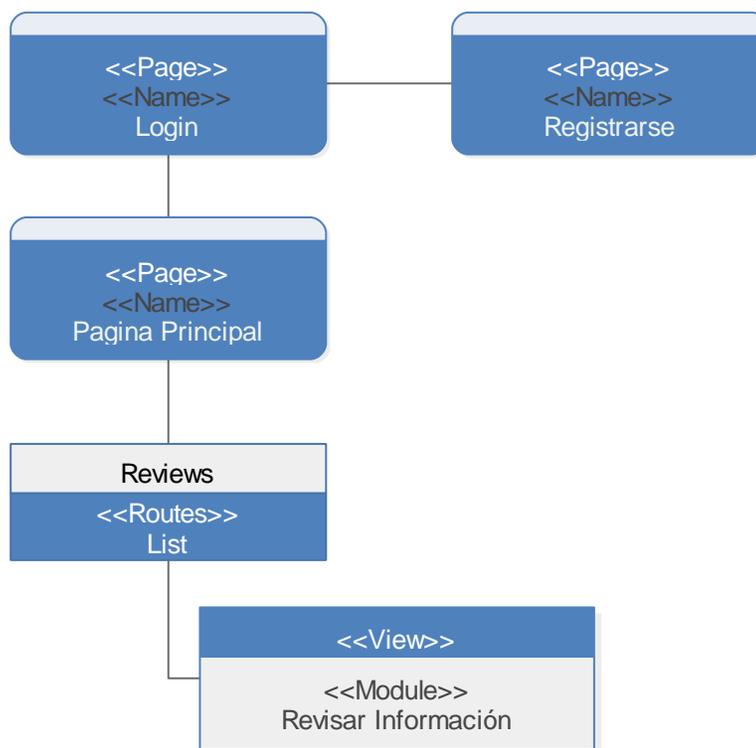
3.2.1.2.2 Diagrama Navegacional:

En el desarrollo de software, es crucial comprender el concepto de un diagrama navegacional. En pocas palabras, un diagrama navegacional es una representación visual que muestra la manera en que los usuarios pueden moverse a través de la interfaz de un sistema interactivo.

Diagrama Navegacional del Revisor. Es una representación gráfica que se realizó para mostrar cómo los revisores o evaluadores pueden navegar por la interfaz del sistema. Este diagrama ayuda a esquematizar y planificar la experiencia del revisor, destacando la estructura y secuencia de las páginas o secciones relevantes desde la perspectiva de los revisores de información.

Figura 18

Diagrama de Navegación: Vistas del Revisor

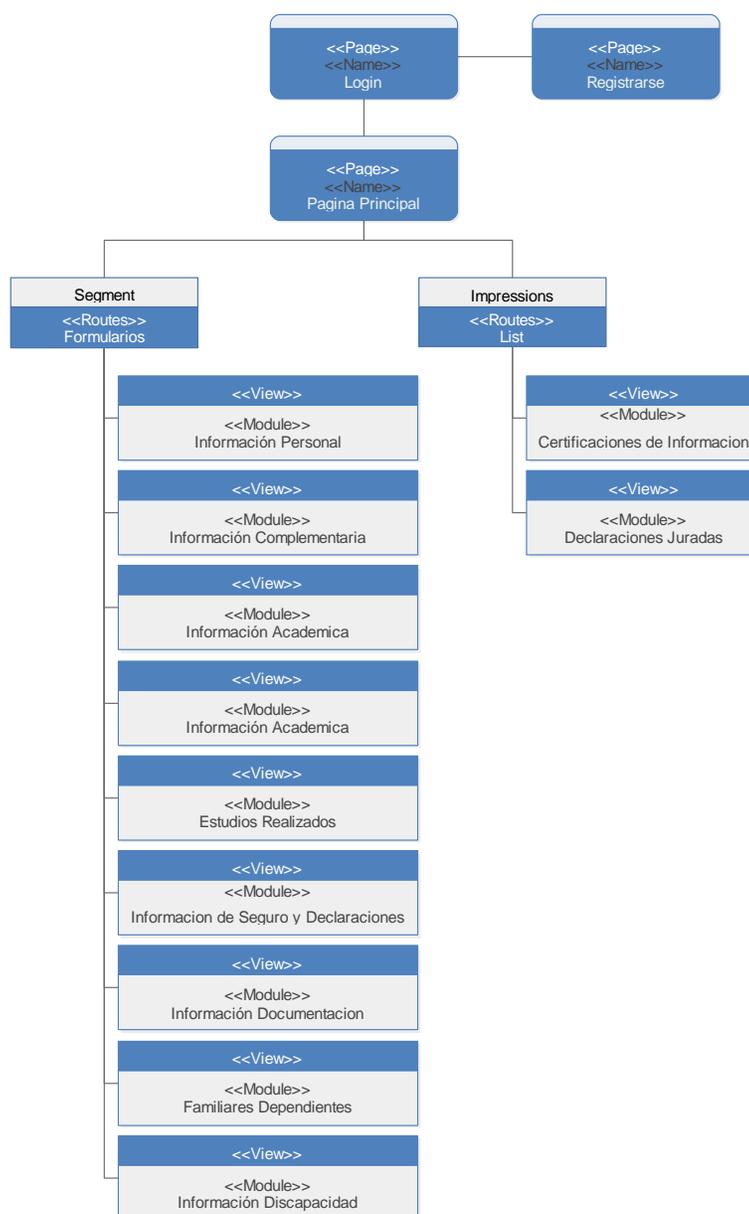


Nota: Diagrama de las Rutas y Vistas que puede acceder el Revisor de la institución.

Diagrama Navegacional del Usuario. Se refiere a una representación gráfica para ilustrar cómo los usuarios pueden navegar a través de la aplicación web. Este diagrama ayuda a visualizar y planificar la experiencia del usuario, mostrando la estructura y la secuencia de las páginas y secciones desde la perspectiva del usuario.

Figura 19

Diagrama de Navegación: Vistas del Usuario



Nota: Diagrama de las Rutas y Vistas que puede acceder el Usuario.

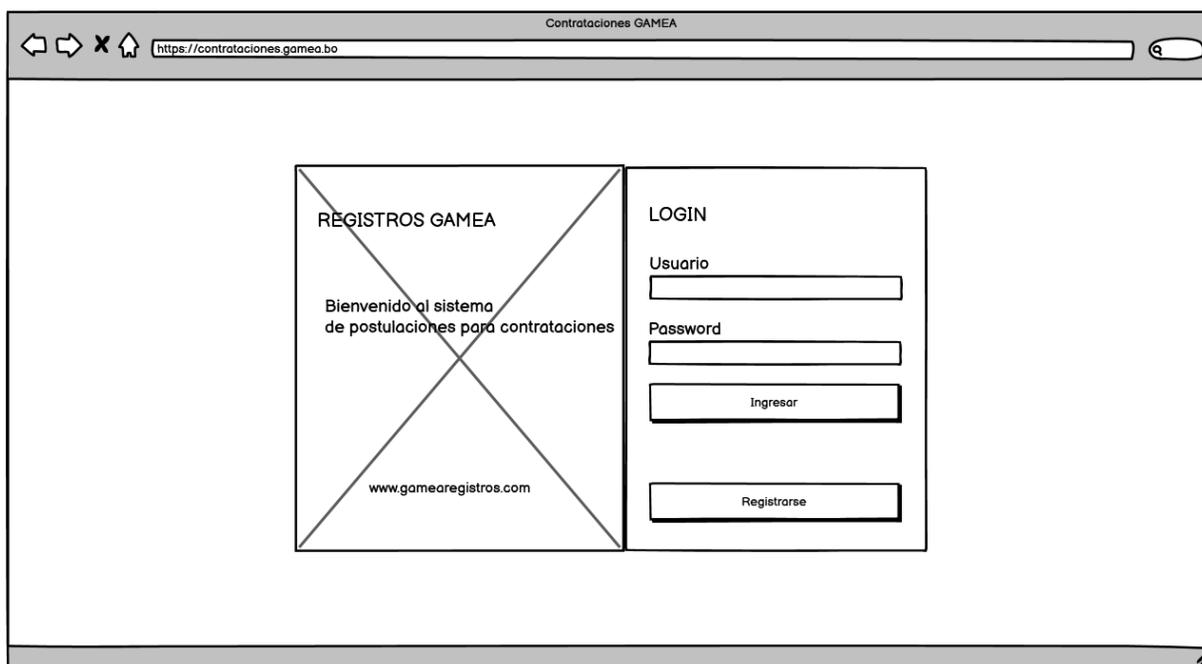
3.2.1.2.3 Diseño de Interfaz

Esta etapa es esencial para garantizar una experiencia de usuario efectiva y atractiva. Involucra la disposición de elementos visuales, como botones, menús, gráficos y texto, con el objetivo de facilitar la navegación, la comprensión y la interacción del usuario. El diseño de interfaz tiene un impacto significativo en la usabilidad y la satisfacción del usuario, y es crucial para el éxito general de diseño de interfaz del usuario en general.

Diseños de Interfaz de Acceso y Registro del Usuario. El acceso eficiente y seguro de los usuarios existentes, junto con la incorporación sencilla de nuevos usuarios, son elementos esenciales de una experiencia positiva. Los diseños de estas interfaces se centran en la combinación de atractivo visual y facilidad de uso, lo que garantiza que todos los usuarios puedan acceder al sistema.

Figura 20

Diseño de Interfaz: Login



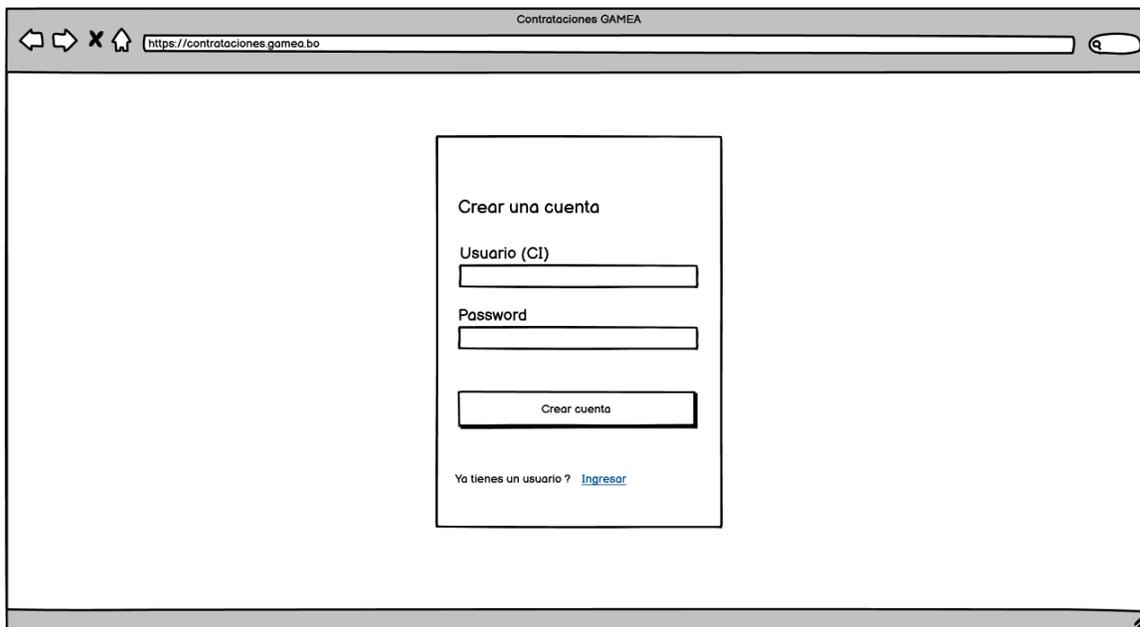
The image shows a web browser window with the address bar displaying 'https://contrataciones.gamea.bo'. The page content is divided into two main sections:

- REGISTROS GAMEA:** A box containing the text 'Bienvenido al sistema de postulaciones para contrataciones' and the website URL 'www.gamearegistros.com'. This section is crossed out with a large 'X'.
- LOGIN:** A form with the following elements:
 - Label: 'Usuario' followed by an input field.
 - Label: 'Password' followed by an input field.
 - Button: 'Ingresar' (Login).
 - Button: 'Registrarse' (Register).

Nota: Interfaz de Acceso de Usuarios al sistema web.

Figura 21

Diseño de Interfaz: Registro de Usuarios



The screenshot shows a web browser window with the address bar containing "https://contrataciones.gamea.bo". The page title is "Contrataciones GAMEA". The main content is a registration form titled "Crear una cuenta". The form includes two input fields: "Usuario (CI)" and "Password". Below these fields is a "Crear cuenta" button. At the bottom of the form, there is a link: "Ya tienes un usuario ? [Ingresar](#)".

Nota: Interfaz de Registro de Usuarios nuevos que no tengan acceso al sistema.

Figura 22

Diseño de Interfaz: Mensaje de Bienvenida



The screenshot shows a web browser window with the address bar containing "https://contrataciones.gamea.bo". The page title is "Contrataciones GAMEA". The main content is a welcome message for new users. The message is enclosed in a rectangular box with a large 'X' drawn across it. The text inside the box reads: "POSTULACIONES GAMEA", "Al iniciar recuerde que una vez guardado el formulario no podrá editar los datos del mismo. Conteste los 5 formularios a continuación y una vez completado imprima el mismo y presente a la unidad de Registros de la Unidad de Registros.", and "Comenzar ->". Below the box, the URL "www.gamearegistros.com" is displayed.

Nota: Interfaz de bienvenida a usuarios recientemente registrados.

Diseños de Interfaz de Registro de Información. Se diseñaron módulos que permitirán registrar la información de los usuarios a través de distintos formularios segmentados. A continuación, se describen los formularios que serán visualizados para recopilar dicha información.

Figura 23

Diseño de Interfaz: Módulo Registro de Información Personal

Contrataciones GAMEA

https://contrataciones.gamea.bo

GOBIERNO AUTONOMO MUNICIPAL DE EL ALTO

REGISTRO FICHA DE PERSONAL

El presente cuestionario debera ser llenado considerando el respaldo documentado respectivo de cada uno de los datos. La Informacion registrada sera considerada como final para los fines consiguientes.

1

I. DATOS PERSONALES I

Nombres Primer Apellido Segundo Apellido Estado Civil

Apellido de Casada Tipo de documento Numero Documento Expedido Expedido/Extranjero Pais

Fecha de Nacimiento Lugar de Nacimiento Provincias Departamento Nacionalidad Pais

El PDF que ingrese tiene que contener los siguientes documentos:

- Carnet anverso - reverso
- Certificado de Nacimiento
- Certificado de Matrimonio en caso de casado(a).
-

PDF - Agregar respaldo

- Tamaño 4x4
- Fondo Rojo
- Vestimenta Formal

Nota: En esta interfaz, los usuarios podrán registrar sus datos personales y adjuntar una fotografía junto con la documentación de respaldo necesaria. Esto permitirá una recopilación completa y segura de la información personal.

Figura 24

Diseño de Interfaz: Módulo Registro de Información Complementaria

Contrataciones GAMEA

https://contrataciones.gamea.bo



GOBIERNO AUTONOMO MUNICIPAL DE EL ALTO

REGISTRO FICHA DE PERSONAL

El presente cuestionario debera ser llenado considerando el respaldo documentado respectivo de cada uno de los datos. La Informacion registrada sera considerada como final para los fines consiguientes.

2

II. DATOS COMPLEMENTARIOS

Direccion Domicilio

Calle o Avenida # de Domicilio Zona/urbanizacion Distrito Municipio

Edificio/ (Opcional) Email (Personal) Email (Trabajo) Telefono Celular

Servicio Militar (Si Corresponde)

Sexo Grupo Sanguineo Serie-Nro Libreta Militar Matricula Año Tipo de Servicio

M F N Servicio Militar

Licencia de Conducir (Opcional)

Licencia de conducir Nro LICENCIA DE CONDUCIR Categoria

SI NO T

Datos del familiar en caso de emergencia

Nombre y Apellido Parentesco Telefono celular del familiar

Hermana

El PDF que ingrese tiene que contener los siguientes documentos:

- Croquis de Domicilio.(Agregar referencia)
- Libreta de Servicio Militar (Si Corresponde).
- Licencia de Conducir (Si Corresponde).

PDF - Agregar respaldo

Nota: En esta interfaz, los usuarios podrán registrar datos complementarios, como su dirección de domicilio, información sobre el servicio militar, licencia de conducir y los datos de un familiar de contacto en caso de emergencia, además de adjuntar la documentación de respaldo necesaria.

Figura 25

Diseño de Interfaz: Módulo Registro de Información Académica

Contrataciones GAMEA
https://contrataciones.gamea.bo



GOBIERNO AUTONOMO MUNICIPAL DE EL ALTO

REGISTRO FICHA DE PERSONAL

El presente cuestionario debera ser llenado considerando el respaldo documentado respectivo de cada uno de los datos. La Informacion registrada sera considerada como final para los fines consiguientes.

5

VI. NIVEL ACADEMICO MAXIMO ALCANZADO

Nivel de Formacion
 Bachiller Tecnico Medio Tecnico Superior Licenciatura Some text Maestria

Certificado de competencia Otro:

VII. Conocimiento en Idioma Nativo (OBLIGATORIO 1)

IDIOMA
 Lee Escribe Habla

Cursos de Capacitacion Realizados en la Institucion (OBLIGATORIO 4)

Debe ingresar mínimamente las cuatro capacitaciones que la institución solicita previa contratación.

Nombre de la Capacitacion	De ...	Fecha de Inicio	Fecha de Conclusion	
<input type="text"/>	Opcion 1	<input type="text" value="/ /"/>	<input type="text" value="/ /"/>	<input type="button" value="AGREGAR"/>

El PDF que ingrese tiene que contener los siguientes documentos:

- Certificado del Titulo máximo alcanzado
- Certificado del Idioma Nativo
- Certificado de la capacitación 1
- Certificado de la capacitación 2
- Certificado de la capacitación 3
- Certificado de la capacitación 4

PDF - Agregar respaldo

Nota: En esta interfaz, los usuarios podrán ingresar información académica esencial, incluyendo su nivel de educación, idioma nativo y cursos institucionales, además de adjuntar un archivo PDF de respaldo. Esto facilitará la recopilación precisa de los datos académicos requeridos.

Figura 26

Diseño de Interfaz: Módulo Registro de Información Estudios Realizados

Contrataciones GAMEA

https://contrataciones.gamea.bo



GOBIERNO AUTONOMO MUNICIPAL DE EL ALTO

REGISTRO FICHA DE PERSONAL

El presente cuestionario deberá ser llenado considerando el respaldo documentado respectivo de cada uno de los datos. La Información registrada será considerada como final para los fines consiguientes.

5

VI. ESTUDIOS CURSADOS / EN PROCESO (OPCIONAL)

Título Obtenido Carrera Institución / Universi

Lugar Ciudad Pais

Diploma Academico Revalidado SI NO SI NO Institución que revalida Numero de título

Título Profesional Revalidado SI NO SI NO Institución que revalida Numero de título

AGREGAR

Conocimientos en Idiomas (OPCIONAL)

IDIOMA Lee Escribe Habla

AGREGAR

Cursos de Capacitación Realizados (OPCIONAL)

Debe ingresar las capacitaciones realizadas en las 3 ultimas gestiones.

Nombre de la Capacitación Institución Carga HORARIA Fecha de Inicio Fecha de Conclusion

AGREGAR

El PDF que ingrese tiene que contener los siguientes documentos:

- Curriculum Vitae Documentado incluido los certificados que registro previamente.
- Certificado del Idioma registrado previamente (Original).

PDF - Agregar respaldo

AGregar

INDICACIONES:

- Todos los certificados adjunto al PDF deben ser originales

GUARDAR

SIGUIENTE FORMULARIO

Nota: En esta interfaz, los usuarios podrán proporcionar detalles sobre su historial educativo, incluyendo carreras completadas, idiomas estudiados y cursos realizados, adjuntando un archivo PDF de respaldo.

Figura 27

Diseño de Interfaz: Módulo Registro de Información del Seguro de Salud y Declaraciones

Juradas

Contrataciones GAMEA

https://contrataciones.gamea.bo

GOBIERNO AUTONOMO MUNICIPAL DE EL ALTO

REGISTRO FICHA DE PERSONAL

El presente cuestionario debera ser llenado considerando el respaldo documentado respectivo de cada uno de los datos. La Informacion registrada sera considerada como final para los fines consiguientes.

3

II. DATOS SEGURO Y DECLARACION

Llenar si corresponde

Personal Nuevo/Antiguo:

Nombre de Seguro Medico:

Seguro social:

Indicar:

Matricula del asegurado:

numero del empleador:

Tipo de Seguro: Publico Privado

anterior afp:

NUA / CUA:

CAS: SI NO años:

Numero de Cuenta Bancaria en BANCO UNION:

Declaracion jurada de bienes y rentas

Numero de Declaracion:

Antes:

Durante:

Despues:

El PDF que ingrese tiene que contener los siguientes documentos:

- Curriculum Vitae Documentado incluido los certificados que registro previamente.
- Certificado del Idioma registrado previamente (Original).

PDF - Agregar respaldo:

INDICACIONES:

- Todos los certificados adjunto al PDF deben ser originales

Nota: A través de esta interfaz, los usuarios podrán ingresar información relativa a su seguro de salud y realizar declaraciones juradas. Asimismo, tendrán la posibilidad de adjuntar un archivo PDF de respaldo para respaldar estos datos.

Figura 28

Diseño de Interfaz: Módulo Registro de Información de la Experiencia Laboral

Contrataciones GAMEA

https://contrataciones.gamea.bo

GOBIERNO AUTONOMO MUNICIPAL DE EL ALTO

REGISTRO FICHA DE PERSONAL

El presente cuestionario debera ser llenado considerando el respaldo documentado respectivo de cada uno de los datos. La Informacion registrada sera considerada como final para los fines consiguientes.

6

VI. EXPERIENCIA LABORAL

VI. Experiencia Laboral General (OPCIONAL)

Institucion Tipo de Institucion Privada Publica Otro: Cargo Desempeñado

Indique si fue: Pais Fecha de Inicio Fecha de Conclusion

Referente al cargo que postula Otro

VI. Experiencia Laboral Dentro del GAMEA (OPCIONAL)

Institucion Tipo de Institucion Privada Publica Otro: Cargo Desempeñado

Pais Fecha de Inicio Fecha de Conclusion

AGREGAR

AGREGAR

El PDF que ingrese tiene que contener los siguientes documentos:

- Curriculum Vitae Documentado incluido los certificados que registro previamente.
- Certificado del Idioma registrado previamente (Original).

PDF - Agregar respaldo

INDICACIONES:

- Todos los certificados adjunto al PDF deben ser originales

SIGUIENTE FORMULARIO

Nota: Mediante esta interfaz, los usuarios podrán registrar su experiencia laboral tanto dentro de la institución o en otras instituciones. También se les permitirá adjuntar un archivo PDF para respaldar esta información.

Figura 29

Diseño de Interfaz: Módulo Registro de Tenencia de Documentación

Contrataciones GAMEA

https://contrataciones.gamea.bo

GOBIERNO AUTONOMO MUNICIPAL DE EL ALTO

REGISTRO FICHA DE PERSONAL

El presente cuestionario debera ser llenado considerando el respaldo documentado respectivo de cada uno de los datos. La Informacion registrada sera considerada como final para los fines consiguientes.

8

AGREGAR LOS SIGUIENTES FORMULARIOS

Indique los formularios con los que cuenta

- Formulario DTH Nro 03, Declaracion jurada - Incompatibilidad de doble percepcion en el sector publico.
- Formulario DTH Nro 07, Declaracion jurada de Parentesco
- Formulario DTH Nro 08, Declaracion jurada de Domicilio
- Formulario (AVC - 04)
- Formulario (AVC - 07)
- Formulario (AVC - 06)
- Formulario (AVC - 09)

Agregue los formularios en el orden establecido en la lista.

Nota: A través de esta interfaz, los usuarios podrán indicar la documentación relevante con la que cuenten, como declaración jurada de doble percepción de salario, información de parentesco, declaraciones juradas de domicilio y formularios AVC para el seguro de salud. También tendrán la opción de adjuntar archivos PDF para respaldar esta información.

Figura 30

Diseño de Interfaz: Módulo Registro de Familiares Dependientes

Contrataciones GAMEA

https://contrataciones.gamea.bo

GOBIERNO AUTONOMO MUNICIPAL DE EL ALTO

REGISTRO FICHA DE PERSONAL

El presente cuestionario debera ser llenado considerando el respaldo documentado respectivo de cada uno de los datos. La Informacion registrada sera considerada como final para los fines consiguientes.

7

VIII. DEPENDIENTES

Nombre y Apellidos Sexo M F N Parentesco Fecha de Nacimiento / /

Tipo de Documento # documento de Identificacion Expedido

Nro de documento relacion de dependencia Tipo de Documento de respaldo relacion de dependencia

Cer. Nacimiento Cer. Matrimonio Libreta Familiar Otro:

Genere un pdf con su cedula de identidad
 Con sus datos personales ...

PDF - Agregar respaldo

Nota: En esta interfaz, los usuarios podrán especificar a sus familiares dependientes. Además, tendrán la posibilidad de adjuntar archivos PDF para respaldar esta información.

Figura 31

Diseño de Interfaz: Módulo Registro de Discapacidad

Contrataciones GAMEA

https://contrataciones.gamea.bo

GOBIERNO AUTONOMO MUNICIPAL DE EL ALTO

REGISTRO FICHA DE PERSONAL

El presente cuestionario debera ser llenado considerando el respaldo documentado respectivo de cada uno de los datos. La Informacion registrada sera considerada como final para los fines consiguientes.

9

DISCAPACIDAD

TITULAR TUTOR LEGAL padre o madre conyugue

FORMULARIO DE DISCAPACIDAD (TITULAR)

Item/ Contrato Eventu Numero de Carnet IBC - PDF - Carnet de Discapacidad Agregar **GUARDAR**

FORMULARIO DE DISCAPACIDAD (TUTOR)

Parentesco con el Discapacitado # Carnet de Discapacidad # de cedula de Identidad

El PDF debe contener los siguientes documentos.

- CARNET DE DISCAPACIDAD OTORGADO POR LA IBC - CODEPDIS
- CEDULA DE IDENTIDAD DE LA PERSONA CON DISCAPACIDAD (EN CASO QUE EL FUNCIONARIO SEA (PADRE/ MADRE/TUTOR/CONYUGUE) DE LA PERSONA CON DISCAPACIDAD
- DECLARATORIA DE TUTORIA EMITIDA POR EL JUEZ COMPETENTE (EN CASO DE QUE EL FUNCIONARIO SEA PADRE/MADRE/TUTOR/CONYUGUE)
- FORMULARIO DTH N°2 DECLARACION JURADA DE PERSONAS CON DISCAPACIDAD, CONYUGUES, PADRES, MADRES Y/O TUTORES DE HIJOS CON DISCAPACIDAD

PDF - Agregar respaldo Agregar **AGREGAR** **Guardar**

IMPRIMIR FORMULARIO

Nota: En esta interfaz, los usuarios podrán ingresar información sobre discapacidades y también indicar si sus familiares tienen alguna discapacidad. Además, tendrán la opción de adjuntar archivos PDF para respaldar esta información.

Figura 32

Diseño de Interfaz: Módulo Revisión de Información

Contrataciones GAMEA

https://contrataciones.gamea.bo

Admin

Documentacion Revisión

Revisión de documentación

Admin_user_1

Revisión de Formulario

FORMULARIOS	EDICION	VISUALIZAR
Formulario-01: Datos personales	Habilitar edicion	ver
Formulario-02: Datos personales	Habilitar edicion	ver
Formulario-03: Datos	Habilitar edicion	ver
Formulario-04: Datos	Habilitar edicion	ver
Formulario-05: Datos	Habilitar edicion	ver
Formulario-06: Datos	Habilitar edicion	ver
Formulario-07: Datos	Habilitar edicion	ver
Formulario-08: Datos	Habilitar edicion	ver

II. DATOS PERSONALES II

Se Grupo Sa Estado Civil Serie-Nro Lib Matricula Año Tipo de Servi

Dato de familiar en caso de emergencia

Nombre y Apellido Parentesco Telefono Celular

Dirección Domicilio

Calle o Avenida Zona Nombre Edificio - Lugar Telefono de Domicilio

Casilla Email (Personal) Email (Trabajo) Telefono Celular

Archivos respaldatorios

[2387982_Formulario-01.pdf](#)

Formulario-01: Datos Personales
Observacion:
 Cumple No Cumple

Formulario-02: Datos Personales
Observacion:
 Cumple No Cumple

Formulario-03: Datos Personales
Observacion:
 Cumple No Cumple

Formulario-04: Datos Personales
Observacion:
 Cumple No Cumple

Formulario-05: Datos Personales
Observacion:
 Cumple No Cumple

Formulario-06: Datos Personales
Observacion:
 Cumple No Cumple

Formulario-07: Datos Personales
Observacion:
 Cumple No Cumple

Formulario-08: Datos Personales
Observacion:
 Cumple No Cumple

Formulario-09: Datos Personales
Observacion:
 Cumple No Cumple

CI del Funcionario

Nota: A través de esta interfaz, los revisores podrán examinar la información proporcionada por los usuarios con el fin de validarla.

Figura 33

Diseño de Interfaz: Barra de Navegación (Navbar)

Contrataciones GAMEA

https://contrataciones.gamea.bo

User

Formularios

Certificaciones

salir

Revisión de documentación

Admin_user_1

Logo of the Government of El Alto

Nota: Esta interfaz presenta la barra de navegación (navbar).

3.2.1.3 Fase de Construcción

Durante esta etapa fundamental, se adentra en la escritura del código fuente, dando vida a las funcionalidades planificadas y siguiendo el diseño y arquitectura que previamente fue establecido. Haciendo un énfasis especial en las pruebas y validaciones, asegurando que el software cumpla con los requisitos y funcione correctamente. En esta fase de construcción del proyecto de grado, es el momento en el que la idea conceptual se convierte en un producto real a través del desarrollo del código fuente.

3.2.1.3.1 Modelado de Datos para MongoDB: Definición de la Estructura de Documentos. En el contexto del desarrollo de una aplicación, es fundamental establecer un modelo de datos sólido y bien estructurado que se utilizará en la base de datos. Este modelo de datos define cómo serán organizados, almacenados y recuperados los datos en MongoDB, y desempeña un papel crucial en la eficacia y la coherencia de nuestro sistema. A través de esta fase de modelado, se allana el camino para una gestión de datos eficiente y un rendimiento óptimo en la aplicación.

Figura 34

Modelo de Datos: Documento Información Personal

```

/* MODELO DE LA INFORMACION PERSONAL
datosPersonales = {
  registroPersonal: { nombres, primerApellido, segundoApellido, estadoCivil, apellidoCasada },
  documentoPersonal: { tipoDocumento, numeroDocumento, lugarExpedicion, complementoCI, extranjero },
  informacionNacimiento: { fechaNacimiento, localidadNacimiento, provinciaNacimiento,
                           departamentoNacimiento, nacionalidad, paisNacimiento },
  documentosRespaldo: {
    documentosPDF: [{ nombrePDF, referenciaPDF, fechaRegistro, estadoPDF }],
    documentosImagen: [{ nombreImagen, referenciaImagen, fechaRegistro, estadoImagen } ]
  }
},
informacionUsuario = { identificacionUsuario },
estadoRegistro,
fechaRegistro,
revisiones = [{revisorName, fechaRevision, estadoRevision,}]

```

Nota: Este documento detalla la estructura de datos personales almacenada en MongoDB.

Figura 35

Modelo de Datos: Documento Información Complementaria

```

/** MODELO DE LA INFORMACION COMPLEMENTARIA
datosComplementarios = {
  direcciónDomicilio: { zona, calleAvenida, númeroDomicilio, distrito, municipio, númeroEdificio, correoElectrónicoPersonal,
    correoElectrónicoTrabajo, teléfonoCelular },
  servicioMilitar: { sexo, grupoSanguíneo, númeroLibretaMilitar, matrículaLibreta, gestiónLibreta, servicioLibreta },
  licenciaConducir: { tieneLicencia, númeroLicencia, categoríaLicencia },
  familiarReferencia: { nombreCompleto, parentesco, teléfonoCelularFamiliar },
  documentosRespaldo: { documentosPDF: [ { nombrePDF, referenciaPDF, fechaRegistroPDF, estadoPDF } ] }
},
informaciónUsuario = { identificaciónUsuario },
estadoRegistro,
fechaRegistro,
revisiones= [ { nombreRevisor, fechaRevisión, estadoRevisión } ]

```

Nota: Este documento describe la estructura de la información complementaria almacenada en MongoDB.

Figura 36

Modelo de Datos: Documento Información Académica

```

/** MODELO DE LA INFORMACION ACADEMICA
estudiosRequeridos = {
  estudioAlcanzado: { nivelAlcanzado, nombreCertificado, otroCertificado },
  certificaciones: { contrato, idiomaNativo: { idioma, habla, escribe, lee},
    cursosInstitucionales: [{ uid, nombreCurso, institucion, fechaInicio, fechaFin } ]
  },
  docRespaldo: { docPdf: [{ pdfRespaldo, pdfReferencia, fechaRegistro, estado } ] }
}
informaciónUsuario = { identificaciónUsuario },
estadoRegistro,
fechaRegistro,
revisiones= [ { nombreRevisor, fechaRevisión, estadoRevisión } ]

```

Nota: Este documento describe la estructura de la información académica almacenada en MongoDB.

Figura 37

Modelo de Datos: Documento de los Estudios Cursados

```

/** MODELO DE LA INFORMACION DE ESTUDIOS REALIZADOS
datosEstudio = {
  estudiosRealizados: [ { identificador, titulo, carrera, institucionEducativa, ciudad, pais,
    logrosAcademicos: { logroObtenido, logroRevalidado, institucionLogroRev, numDiploma, },
    titulosProfesionales: { tituloObtenido, tituloRevalidado, institucionTituloRev, numTitulo, }, }, ],
  idiomasEstudiados: [ { identificador, idioma, institucion, habla, escritura, lectura, fechaInicioEstudio, fechaFinEstudio, }, ],
  capacitacionesRealizadas: [ { identificador, nombreCurso, institucion, horasCursadas, fechaInicioCurso, fechaFinCurso, }, ],
  documentosRespaldo: { documentosPDF: [ { pdfRespaldo, pdfReferencia, fechaRegistroPDF, estadoDocumentoPDF, }, ], },
};
informaciónUsuario = { identificaciónUsuario },
estadoRegistro,
fechaRegistro,
revisiones= [ { nombreRevisor, fechaRevisión, estadoRevisión } ]

```

Nota: Este documento describe la estructura de la información de los estudios realizados que es almacenada en MongoDB.

Figura 38

Modelo de Datos: Documento Seguro de Salud y Declaraciones Juradas

```

/** MODELO DE INFORMACION DE SEGURO Y DECLARACIONES JURADAS
datosSeguro = {
  seguroFuncionario: { tipoContratoSeguro,
    seguroSalud: { nombreSeguroSalud, seguroSocialSalud, titularSeguroSalud, matriculaSeguroSalud, numEmpleadorSeguroSalud,
                  tipoSeguroSalud, },
    afpPensiones: { anteriorAfpPensiones, nuaCuaAfpPensiones, }, cas: { casContrato, canTiempoCas, },
    cuentaBancaria: { numCuentaBancaria, }, },
  declaracionJurada: [ { uidDeclaracionJurada, numDeclaracionJurada, antesDeclaracionJurada, duranteDeclaracionJurada,
                        despuesDeclaracionJurada, }, ],
  docRespaldo: { docPdf: [ { pdfRespaldo, pdfReferencia, fechaRegistroPdf, estadoPdf, }, ], },
},
informaciónUsuario = { identificaciónUsuario },
estadoRegistro,
fechaRegistro,
revisiones= [ { nombreRevisor, fechaRevisión, estadoRevisión } ]

```

Nota: Este documento describe la estructura de la información del seguro de salud y las declaraciones juradas almacenada en MongoDB.

Figura 39

Modelo de Datos: Documento de Tenencia de Documentación

```

/** MODELO DE INFORMACION DE DOCUMENTACION
datoDocumentacion = {
  declaracionesJuradas: { percepcionDeclaracionJurada, parentescoDeclaracionJurada, domicilioDeclaracionJurada, },
  formulariosAVC: { formAVC4FormulariosAVC, formAVC7FormulariosAVC, formAVC6FormulariosAVC, formAVC9FormulariosAVC, },
  docRespaldo: { docPdf: [ { pdfRespaldoDocPdf, pdfReferenciaDocPdf, fechaRegistroPdfDocPdf, estadoPdfDocPdf, }, ], },
informaciónUsuario = { identificaciónUsuario },
estadoRegistro,
fechaRegistro,
revisiones= [ { nombreRevisor, fechaRevisión, estadoRevisión } ]

```

Nota: Este documento describe la estructura de la tenencia de documentación almacenada en MongoDB.

Figura 40

Modelo de Datos: Documento de Experiencia Laboral

```

/** MODELO DE INFORMACION DE EXPERIENCIA LABORAL
datosLaborales = {
  experienciaLaboral: [ { uidExperiencia, institucionExperiencia, tipoInstitucionExperiencia: { tipoInstitucion, descripcionOtro },
                        cargoExperiencia, referenteCargoExperiencia, paisExperiencia, fechaInicioExperiencia, fechaFinExperiencia } ],
  experienciaGubernamental: [ { uidExperienciaGubernamental, instanciaExperienciaGubernamental, cargoExperienciaGubernamental,
                                fechaInicioExperienciaGubernamental, fechaFinExperienciaGubernamental } ],
  docRespaldo: { docPdf: [ { pdfRespaldoDocPdf, pdfReferenciaDocPdf, fechaRegistroPdfDocPdf, estadoPdfDocPdf } ] }
},
informaciónUsuario = { identificaciónUsuario },
estadoRegistro,
fechaRegistro,
revisiones= [ { nombreRevisor, fechaRevisión, estadoRevisión } ]

```

Nota: Este documento describe la estructura de la información de la experiencia laboral almacenada en MongoDB.

Figura 41

Modelo de Datos: Documento Información de Dependientes

```

/** MODELO DE INFORMACION DE DEPENDENCIA
datosDependencia = {
  dependientes: [ { uidDependiente, nombreCompletoDependiente, sexoDependiente, parentescoDependiente,
                  fechaNacDependiente, tipDocumentoDependiente, numDocumentoDependiente, expDocumentoDependiente,
                  relacionDependiente: { tipDocumentoRelacion, desDocumentoRelacion, numDocumentoRelacion } } ],
  docRespaldo: { docPdf: [ { pdfRespaldoDocPdf, pdfReferenciaDocPdf, fechaRegistroPdfDocPdf, estadoPdfDocPdf } ] }
},
informaciónUsuario = { identificaciónUsuario },
estadoRegistro,
fechaRegistro,
revisiones= [ { nombreRevisor, fechaRevisión, estadoRevisión } ]

```

Nota: Este documento describe la estructura de la información de familiares dependientes almacenada en MongoDB.

Figura 42

Modelo de Datos: Documento Información de Discapacidad

```

/** MODELO DE INFORMACION DE DISCAPACIDAD
datosDiscapacidad = {
  tipoDeContrato, discapacitadoTitular: { numeroDeCarnetIBC },
  discapacitadoTutor: [ { uidTutor, parentescoTutor, numeroDeCarnetIBC, numeroDeCarnetCI } ],
  docRespaldo: { docPdf: [ { pdfRespaldoDocPdf, pdfReferenciaDocPdf, fechaRegistroPdfDocPdf, estadoPdfDocPdf } ] }
},
informaciónUsuario = { identificaciónUsuario },
estadoRegistro,
fechaRegistro,
revisiones= [ { nombreRevisor, fechaRevisión, estadoRevisión } ]

```

Nota: Este documento describe la estructura de la información de discapacidad almacenada en MongoDB.

Figura 43

Modelo de Datos: Documento de Usuario

```

/** MODELO DE USUARIO
datosUsuario = { nombre: 'Juan Daniel', imagen:'imagen.jpg' },
credencialesUsuario = { cedulaUsuario: '1234567', contraseña: '*****', rol: 'usuario' },
estado= 'activo',
fechaCreacion= '2023-11-05'

```

Nota: Este documento describe la estructura de la información del usuario del sistema almacenada en MongoDB.

3.2.1.3.2 Desarrollo del Frontend

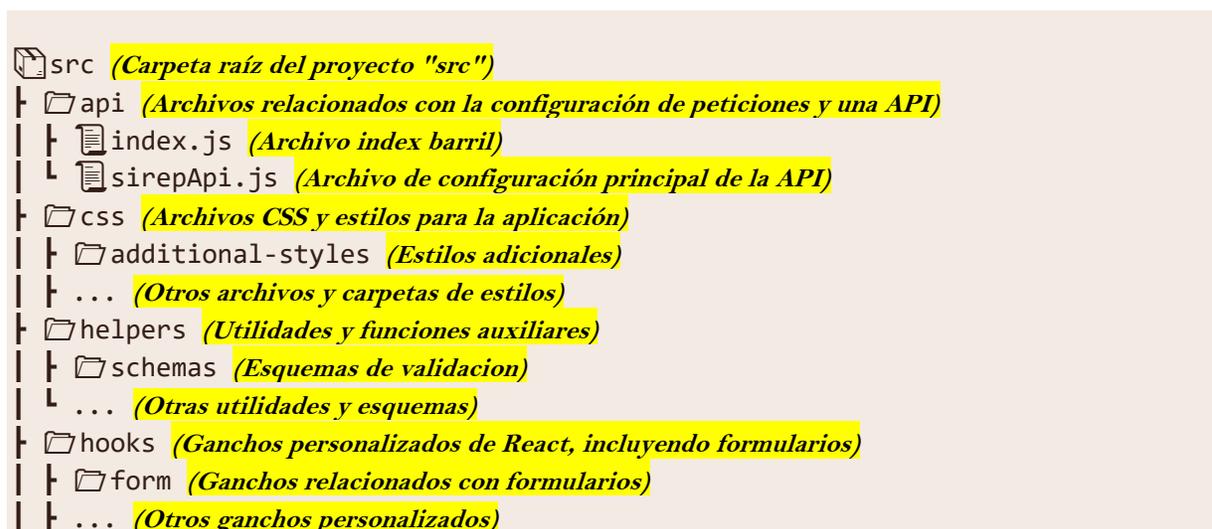
En esta sección, se enfocará en el desarrollo del frontend. Aquí es donde se transformarán los diseños en código, lo que permitirá crear una interfaz de usuario atractiva y funcional para la aplicación.

Desarrollo del código fuente: Tras el arduo trabajo de desarrollo del código fuente en el frontend, se presenta una vista parcial de la estructura de archivos que conforma la esencia del proyecto. Esta descripción se ofrece con el propósito de proporcionar una visión general sin adentrarse en la totalidad del código, ya que la abundancia de componentes hace que sea un desafío plasmarlos en su totalidad. Cada archivo y directorio mencionado aquí juega un papel fundamental en la construcción de una aplicación web robusta y versátil.

El siguiente código pertenece a un proyecto realizado en React, caracterizado por su arquitectura de componentes bien definidos y una gestión de estados segmentados. Cada componente cumple una función específica en la aplicación, y la segmentación de estados garantiza una administración eficiente de los datos. A continuación, se muestra en detalle esta estructura que resalta la solidez de este enfoque en el desarrollo web con React.

Figura 44

Estructura del Frontend



- └─ images *(Imágenes y recursos gráficos)*
 - └─ user-avatar-32.png *(Ejemplo de imagen de avatar de usuario)*
- └─ jsModelPDF *(Archivos relacionados con modelos de PDF en JavaScript)*
 - └─ DeclarationDomicile.jsx *(Modelo de declaración de domicilio)*
 - └─ ... *(Otros archivos relacionados con PDF)*
- └─ pages *(Componentes y vistas de páginas específicas)*
 - └─ certification *(Páginas relacionadas con certificaciones)*
 - └─ form *(Páginas de formularios)*
 - └─ Form1.jsx *(Página de formulario 1)*
 - └─ ... *(Otras páginas de formularios)*
 - └─ information *(Páginas de información)*
 - └─ InformationStudies.jsx *(Página de estudios de información)*
 - └─ ... *(Otras páginas de información)*
 - └─ modals *(Páginas de ventanas modales)*
 - └─ ModalLanguages.jsx *(Página de idiomas modal)*
 - └─ ... *(Otras páginas de ventanas modales)*
 - └─ verification *(Páginas de verificación)*
 - └─ ... *(Otras páginas específicas)*
- └─ partials *(Componentes reutilizables, acciones y elementos de interfaz de usuario)*
 - └─ actions *(Acciones)*
 - └─ addRecord *(Añadir registros)*
 - └─ alerts *(Alertas)*
 - └─ auth *(Autenticación)*
 - └─ cards *(Tarjetas)*
 - └─ CardData.jsx *(Componente de datos de tarjeta)*
 - └─ expandedComponent *(Componentes de expansión)*
 - └─ ExpandedLanguages.jsx *(Componente de idiomas Expansión)*
 - └─ ... *(Otros componentes expandidos)*
 - └─ form *(Formularios)*
 - └─ Input.jsx *(Componente de entrada)*
 - └─ InputAuth.jsx *(Componente de entrada de autenticación)*
 - └─ ... *(Otros componentes de formulario)*
 - └─ icons *(Iconos)*
 - └─ Tiger.jsx *(Icono de tigre)*
 - └─ ... *(Otros iconos)*
- └─ router *(Archivos de enrutamiento y rutas de la aplicación)*
 - └─ AppRouter.jsx *(Enrutador principal de la aplicación)*
 - └─ AuthRouter.jsx *(Enrutador de autenticación)*
 - └─ FormRouter.jsx *(Enrutador de formularios)*
 - └─ index.js *(Archivo índice barril)*
- └─ store *(Gestión del estado de la aplicación, con Redux)*
 - └─ form *(Estado relacionado con formularios)*
 - └─ academicSlice.js *(Estado para datos académicos)*
 - └─ ... *(Otros archivos de gestión de estado)*
- └─ App.jsx *(Archivo principal de la aplicación)*

main.jsx	(Archivo principal de entrada para el código de la aplicación)
sirp.svg	(Archivo SVG que contiene el ícono de la aplicación)

Nota: Esta es una vista parcial de los directorios con los que cuenta el frontend, dando una descripción y la finalidad de cada carpeta y/o archivo.

3.2.1.3.3 Seguridad en el Frontend

Implementación de Expresiones regulares: Durante el desarrollo del proyecto, hemos integrado expresiones regulares como parte fundamental de nuestra estrategia de validación de datos. Estas herramientas nos han permitido implementar controles robustos para asegurar la integridad y precisión de la información ingresada por los usuarios

Figura 45

Expresiones Regulares Implementadas en el sistema.

```

src > helpers > JS regExpressions.js > ...
1
2  ⚠️ TODO: EXPRESIONES REGULARES PARA VALIDAR CAMPOS
3
4  /** PASSWORD
5  /**? min 5 characters, 1 upper case letter, 1 lower case letter, 1 numeric digit.
6  export const passwordRules = /^(?=\S*\d)(?=\S*[a-z])(?=\S*[A-Z])\S{5,}$/;
7
8  /** EMAIL
9  /**? son para validar estandares de correo electronico
10 export const emaildRules = /^[a-zA-Z0-9,_%+-]+@[a-zA-Z0-9,-]+\.[a-zA-Z]{2,}$/;
11
12 /** NOMBRES EN MAYUSCULAS
13 /**? nos permite validar nombres en mayusculas
14 // export const stringdRules = /^[A-ZÁÉÍÓÚÑ]+\s?*[A-ZÁÉÍÓÚÑ]+$/;
15 export const stringdRules = /\s*[A-ZÁÉÍÓÚÑ]+\s*$/;
16

```

Validación de la extensión y tamaño del archivo: Se valido tanto la extensión como el tamaño de los archivos ingresados. Esta validación asegura que solo se acepten tipos de archivos específicos, relevantes y seguros para el sistema, evitando potenciales amenazas de seguridad. Además, la verificación del tamaño garantiza que los archivos sean adecuados para un procesamiento eficiente, optimizando así el rendimiento del sistema desarrollado. Este proceso meticuloso es esencial para mantener la integridad de los datos y asegurar la eficiencia operativa del proyecto.

Figura 46

Validación de Extensión y Tamaño de Archivos

```

155   f1_filePdf: yup
156     .mixed()
157     .test("valid_type", "Archivo de respaldo valido con extenciones ['pdf']",
158       value => isValidFileTypeEdit(value && value.name.toLowerCase(), "document"))
159     .test("valid_size", "Exede el maximo permitido, que es: (100KB)",
160       value => isValidFileSize(value && value.size, "document")),
161   });
162
163   function isValidFileType(fileName, fileType) {
164     return fileName && validFileExtensions[fileType].indexOf(fileName.split('.').pop()) > -1;
165   }
166
167   function isValidFileSize(size, typeFile) {
168     if(Boolean(!size)) { return true; }
169     if (typeFile === "image") { return size <= MAX_FILE_SIZE_IMG; }
170     if (typeFile === "document") { return size <= MAX_FILE_SIZE_PDF; }
171     return false;
172   }

```

Figura 47

Integración de Formik y Expresiones Regulares

```

50
51   export const externalWorkingSchema = yup.object().shape({
52     f6_ext_1: yup
53       .string().required('El nombre de la institucion es obligatorio.')
54       .matches(descriptionRules, 'Verifique el nombre de la institucion.')
55       .max(50, 'El nombre de la institucion no puede tener mas de 50 caracteres.'),
56     f6_ext_2: yup
57       .string()
58       .matches(descriptionRules, 'Verifique tipo de institucion seleccionado.')
59       .max(15, 'El tipo de institucion no puede tener mas de 15 caracteres.'),
60     f6_ext_3: yup
61       .string()
62       .matches(descriptionRules, 'Verifique el tipo de institucion(OTRO).')
63       .max(20, 'El tipo de institucion no puede tener mas de 20 caracteres.')
64       .when('f6_ext_2', (f6_ext_2, field) =>
65         f6_ext_2[0] === 'OTRO'
66         ?field.required('Indique el otro tipo de institucion que selecciono.')
67         .matches(descriptionRules, 'El campo solo debe contener letras.')
68         :field
69       ),
70     f6_ext_4: yup
71       .string().required('El cargo que ocupo es obligatorio.')
72       .matches(descriptionRules, 'Verifique el tipo de cargo que ocupo.')
73       .max(60, 'El cargo desempeñado no puede tener mas de 60 caracteres.'),
74     f6_ext_5: yup
75       .string().required('Indique si es referente al cargo que postula.')
76       .matches(descriptionRules, 'Verifique la informacion referente al cargo que desempeño.')
77       .max(4, 'La referencia al cargo no puede exeder los 4 caracteres.'),
78     f6_ext_6: yup
79       .string().required('El pais donde ejercicio laboralmente es obligatorio.')
80       .matches(descriptionRules, 'Verifique el lugar donde deseño su trabajo.')
81       .max(25, 'El lugar no puede tener más de 50 caracteres.'),
82     f6_ext_7: yup
83       .date().required('La fecha de inicio es obligatorio.'),
84     f6_ext_8: yup
85       .date().required('La fecha de conclusion es obligatorio.'),
86   });
87

```

Nota: La validación utilizando Formik y expresiones regulares se implementará de manera consistente en todos los formularios para asegurar la precisión y la integridad de los datos ingresados.

3.2.1.3.4 Desarrollo del Backend

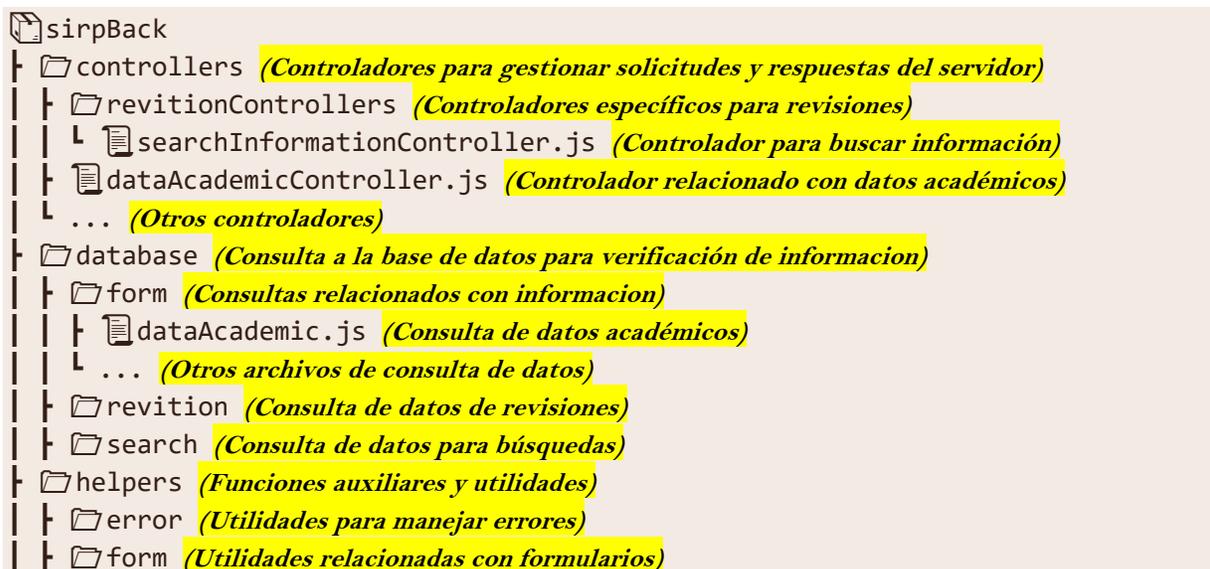
En esta sección, se enfocará en el desarrollo del backend. Aquí es donde se llevará a cabo la implementación de la lógica de negocio y la funcionalidad principal de la aplicación.

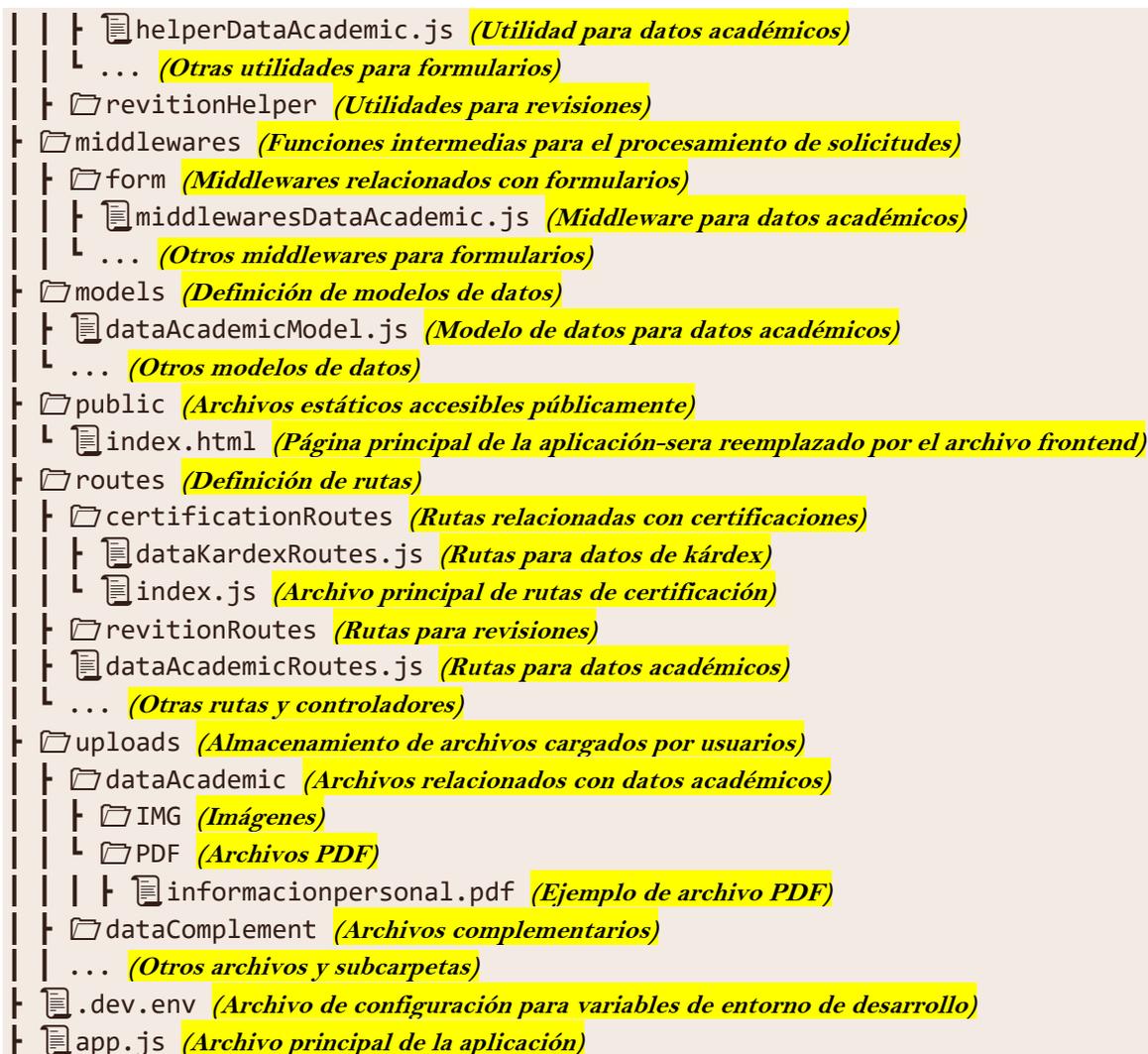
Desarrollo del código fuente: Después del proceso de desarrollo del código fuente en el backend, se presenta una vista parcial a la estructura de archivos que conforma la columna vertebral del proyecto. Esta descripción tiene como objetivo proporcionar una visión general sin adentrarse en la totalidad del código, ya que la riqueza de componentes hace que sea un desafío representarlos en su totalidad. Cada archivo y directorio mencionado aquí desempeña un papel crucial en la construcción de un backend robusto y versátil

El siguiente código forma parte de un proyecto en Node.js, caracterizado por su arquitectura de controladores, modelos, rutas y middleware bien definida. Cada componente tiene una función específica en la aplicación, y la estructura modular garantiza una administración eficiente de las solicitudes y respuestas. A continuación, se detalla esta estructura que resalta la robustez de este enfoque en el desarrollo del backend en Node.js.

Figura 48

Estructura del Backend





Nota: Esta es una vista parcial de los directorios con los que cuenta el backend, dando una descripción y la finalidad de cada carpeta y/o archivo.

Uno de los aspectos más notables de este proyecto es la aplicación de una arquitectura por capas cuidadosamente diseñada. Esta arquitectura ha sido la columna vertebral del desarrollo de la aplicación, permitiendo una organización sistemática y eficiente de todos los componentes del sistema. Cada capa se ha diseñado de manera independiente y se encarga de una función específica, lo que ha simplificado la gestión, la escalabilidad y el mantenimiento de la aplicación. Gracias a esta arquitectura, se ha logrado una separación clara de responsabilidades, lo que facilita el trabajo y la adaptación a futuros cambios. La

implementación de esta arquitectura por capas refleja el compromiso de este proyecto con las mejores prácticas de desarrollo de software, garantizando un producto final sólido y versátil.

3.2.1.3.5 Seguridad en el Backend

Por motivos de seguridad, los detalles específicos del proceso de validación en el backend no se mostrarán. Sin embargo, a continuación se describe el enfoque utilizado, que incluye la validación de datos ingresados, las opciones select consideradas, la validación de archivos, la verificación del token de acceso y el segmentación de rutas. Este enfoque integral asegura la protección de datos y la integridad del sistema ante posibles vulnerabilidades y accesos no autorizados.

- **Validación de Datos Ingresados.** Proceso de verificación y aseguramiento de que los datos introducidos por usuarios o sistemas externos cumplan con los criterios especificados, como formatos adecuados, longitudes correctas y caracteres permitidos. Esto ayuda a evitar datos incorrectos o maliciosos que puedan comprometer la integridad del sistema.
- **Opciones Select que se Contemplan.** Implementación de controles selectivos en formularios, donde los usuarios eligen entre opciones predefinidas. Se verifica que solo se seleccionen y envíen opciones válidas para garantizar que el sistema procese la información de manera correcta y coherente.
- **Validación de archivos:** Proceso de verificación de archivos cargados para asegurar que cumplen con criterios específicos, como tipos de archivo permitidos y tamaños aceptables. Esto ayuda a mitigar riesgos asociados con archivos maliciosos o corruptos que podrían comprometer la seguridad o el funcionamiento del sistema.
- **Validación del Token de Acceso.** Verificación de la autenticidad y validez de los tokens de acceso utilizados para la autenticación y autorización de usuarios. Se

asegura que los tokens sean válidos y no hayan sido manipulados, protegiendo así las áreas protegidas del sistema contra accesos no autorizados.

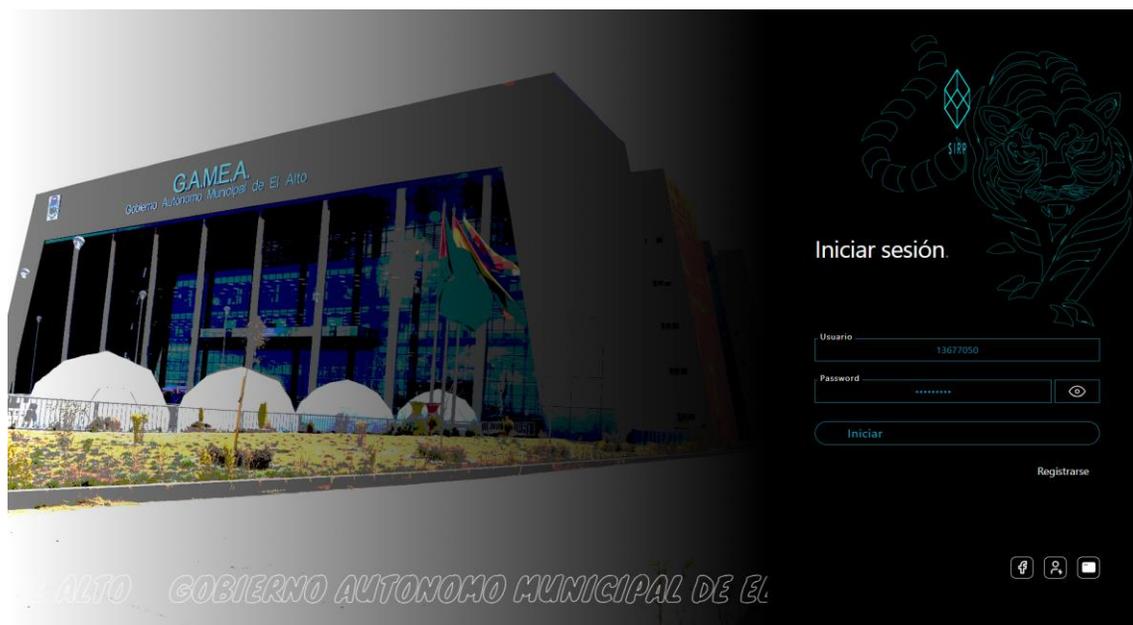
- **Segmentación de rutas.** Implementación de medidas para restringir el acceso a rutas y recursos específicos del sistema basadas en el contexto y los permisos del usuario autenticado. Esto garantiza que solo los usuarios autorizados puedan acceder a funciones y datos sensibles, fortaleciendo la seguridad general del sistema.

3.2.1.3.6 Interfaz de Usuario en Funcionamiento

En esta sección se presentará exclusivamente la apariencia de algunas interfaces gráficas seleccionadas. La decisión de limitar la visualización responde a la intención de ofrecer una representación más concisa y enfocada del proyecto, evitando una saturación de información. En particular, se mostrará la apariencia de interfaces relacionadas con datos personales, datos complementarios y algunos modales clave. Este enfoque permitirá resaltar de manera efectiva los aspectos visuales más significativos de la aplicación.

Figura 49

Interfaz de Usuario: Login



Nota: Este es la página de acceso, inicio de sesión de usuarios.

Figura 50

Interfaz de Usuario: Registro de Usuarios

Nota: Este es la página de registro de usuarios.

Figura 51

Interfaz de Usuario: Visualización de Datos Personales

I. DATOS PERSONALES		
NOMBRE COMPLETO	DOCUMENTO	NUMERO DE DOCUMENTO
JUAN DANIEL MAMANI QUISPE	CI	13677050
FECHA DE NACIMIENTO	PROVINCIA DE NAC	DEPARTAMENTO DE NAC
10 de febrero de 2000	MURILLO	LA PAZ

Documentos PDF		
#	PDF	Fecha
1	boleta_inscripcion73_07_2023JUAN DANIEL MAMANI QUISPE.pdf	12/10/2023
2	2731-Texto del artículo-30687-1-10-20220816.pdf	15/10/2023

Imágenes		
#	IMG	Fecha
1	imgPerfil.jpg	12/10/2023

Fecha de Registro: 12 de octubre de 2023

Nota: Esta interfaz es donde se visualiza los datos personales registrados.

Figura 52

Interfaz de Usuario: Crear y Actualizar los Datos Personales

I. DATOS PERSONALES

Nombres: JUAN DANIEL | Primer Apellido: MAMANI | Segundo Apellido: QUISPE | Estado Civil: SOLTERO/A
 Apellido de Casado: | Documento: CI | Numero de Documento: 13677050 | Complemento: |
 Expedido: LA PAZ | Expedido/Estranjero: | Pais de Nacimiento: BOLIVIA | Nacionalidad: BOLIVIANA
 Fecha de Nacimiento: 10/02/2000 | Localidad: | Provincia: MURILLO | Departamento: LA PAZ

Foto de Identidad: [Foto] | Seleccionar archivo: Ninguno archivo selec.

#	PDF	Fecha
1	Boleta_Inscripcion19_07_2023JUAN DANIEL MAMANI QUISPE.pdf	12/10/2023
2	Z713-Texto del artículo-10587-1-10-20220816.pdf	15/10/2023

Imágenes

#	IMG	Fecha
1	imgPerfil.jpg	12/10/2023

El PDF que ingrese debe contener los siguientes archivos
 • Carnet de identidad Digitalizado. **(Obligatorio)**
 • Certificado de Nacimiento. **(Obligatorio)**
 • Certificado de Matrimonio. **(Si corresponde)**

PDF Datos Personales: Seleccionar archivo | Ninguno archivo selec.

Actualizar Registro (Botón Azul)
 Cancelar Edicion (Botón Rojo)

Nota: Esta interfaz es donde se introducen los datos y suben los archivos de respaldo de la información personal.

Figura 53

Interfaz de Usuario: Visualización de Datos Complementarios

II. DATOS COMPLEMENTARIOS

Dirección de Domicilio
 ZONA/URBANIZACION: FRANZA TAMAYO | CALLE: ISMAEL SOTO MAYOR | NUMERO DE DOMICILIO: 1474 | DISTRITO: 14 | MUNICIPIO: EL ALTO | EMAIL PERSONAL: juandanielquispeperedo@gmail.com | NUMERO DE CELULAR: 74033698

Servicio Militar
 SEXO: MASCULINO | GRUPO SANGUINEO: ORH+ | NUMERO DE LIBRETA: 1234 | NUMERO DE MATRICULA: 13495393-SD | AÑO: 2016 | TIPO DE SERVICIO: SERVICIO PREMILITAR

Licencia de Conducir
 LICENCIA DE CONDUCIR: SI | NUMERO DE LICENCIA: 11313145 | CATEGORIA: A

Familiar de Referencia
 NOMBRE DE FAMILIAR: JORGE DANIEL MQ | PARENTESCO: HE | NUMERO DE CELULAR: 74069369

#	PDF	Fecha
1	645-Texto del artículo-825-1-10-20180524.pdf	17/10/2023

Fecha de Registro: 17 de octubre de 2023

Editar Registros (Botón Rojo)

Nota: Esta interfaz es donde se visualiza los datos complementarios registrados.

Figura 54

Interfaz de Usuario: Crear y Actualizar los Datos Complementarios

II. DATOS COMPLEMENTARIOS

Datos de Referencia

Zona/Urbанизación: FRANZA TAMAYO | Calle o Avenida: ISMAEL SOTO MAYOR | Numero de Domicilio: 1474 | Distrito: 14 | Municipio: EL ALTO

Edificio (Opcional): | Email Personal: juandanielquispeperedo@gmail.com | Email Trabajo (Opcional): | Numero Celular: 74033698

Servicio Militar (Si Corresponde)

Sexo: M F O | Grupo Sangre: ORH+ | Nro. Libreta Militar: 1234 | Matricula: 13495393-SD | Año: 2016 | Tipo de Servicio Militar: SERVICIO PREMILITAR

Licencia de Conducir (Opcional)

Licencia de Conducir: SI NO | Numero de Licencia: 11313145 | Categoria: CATEGORIA A

Familiar de Referencia

Nombre y Apellido: JORGE DANIEL MQ | Parentesco: HERMANO/A | Celular: 74069369

Documentos PDF

#	PDF	Fecha
1	645_Texto del articulo-825.1.30-20180524.pdf	17/10/2023

El PDF que ingrese debe contener los siguientes archivos

- Croquis de Domicilio. (Obligatorio)
- Libreta de Servicio Militar. (Obligatorio en Varones)
- Licencia de Conducir. (Opcional)
- Cedula de identidad digitalizada del Familiar. (Obligatorio)

PDF Datos Complementarios

Seleccionar archivo | Ninguno archivo selec.

Actualizar Registro

Cancelar Edicion

Nota: Esta interfaz es donde se introducen los datos y suben los archivos de respaldo de los datos complementarios.

Figura 55

Interfaz de Usuario: Modal Editar Título Profesional

IV. ESTUDIOS CURSADOS / EN PROCESO (OPCIONAL)

Titulos Obtenidos

Titulo Obtenido: LICENCIATURA EN INFORMATICA Y TELECOMUNICACION | Carrera: INGENIERIA DE SISTEMAS

Institucion / Universidad: UNIVERSIDAD PUBLICA DE EL ALTO

Lugar / Ciudad: LA PAZ | Pais: BOLIVIA

Diploma Academico: SI NO | Revalidado: SI NO

Institucion que lo revalida: UNIVERSIDAD PUBLICA DE EL ALTO | Numero de Diploma: 45287

Titulo Profesional: SI NO | Revalidado: SI NO

Institucion que lo revalida: UNIVERSIDAD PUBLICA DE EL ALTO | Numero de Titulo: 64846

Conocimientos

Idioma: INGLES TECNICO I | Institucion: UPEA

Fecha de Inicio: dd/mm/aaaa

Cancelar | Guardar

Nota: Esta interfaz es para editar los datos de un título profesional registrado.

Figura 56

Interfaz de Usuario: Impresión de Documentación



Nota: Esta interfaz es para impresiones de documentación.

Figura 57

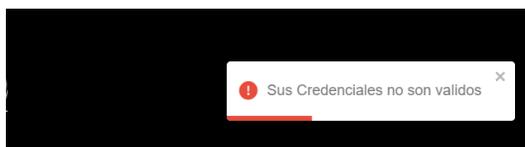
Interfaz de Usuario: Revisión de Información

#	Nombre del Formulario	Visualizar	Habilitar
1	Datos Personales.	<input type="checkbox"/>	Habilitar Edición
2	Datos Complementarios.	<input checked="" type="checkbox"/>	Habilitar Edición
3	Datos Académicos.	<input type="checkbox"/>	Habilitar Edición
4	Estudios Cursados.	<input type="checkbox"/>	Habilitar Edición
5	Datos de Seguro Bienes y Rentas.	<input type="checkbox"/>	Habilitar Edición
6	Experiencia Laboral.	<input type="checkbox"/>	Habilitar Edición
7	Documentos Solicitados.	<input type="checkbox"/>	Habilitar Edición
8	Familiares Dependientes.	<input type="checkbox"/>	Habilitar Edición
9	Datos de Discapacidad.	<input type="checkbox"/>	Habilitar Edición

Nota: Esta interfaz es para revisión de la información registrada de los usuarios.

Figura 58

Interfaz de Usuario: Alertas



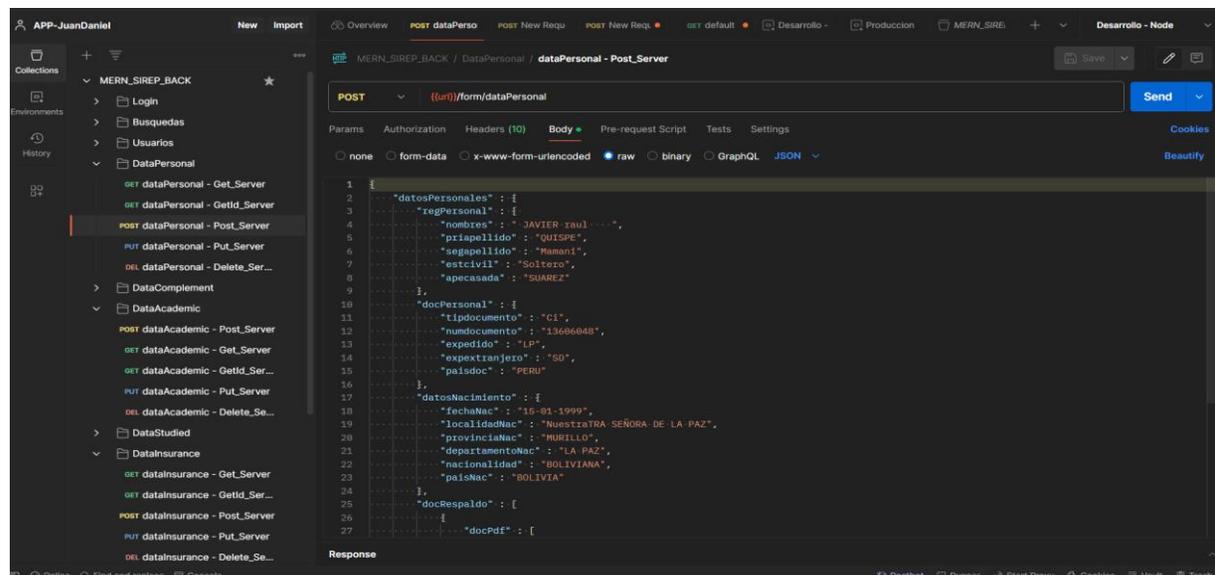
Nota: Es el formato en que se presentan las alertas.

3.2.1.4 Fase de Transición. La Fase de Transición en la metodología OpenUP representa el momento crucial en el que el software desarrollado se prepara para su entrega y despliegue. Durante esta etapa, se lleva a cabo la verificación y validación finales del software, se realizan pruebas exhaustivas y se garantiza que el producto esté listo para su implementación en el entorno de producción. La Fase de Transición es esencial para garantizar que el software se entregue de manera exitosa y que cumpla con los requisitos y expectativas del cliente.

3.2.1.4.1 Realización de Pruebas Exhaustivas. Las pruebas de software son esenciales para garantizar la calidad, rendimiento y las respuestas que se esperan durante el desarrollo, es por eso que se utilizó Postman para mandar peticiones al back y ver las respuestas iniciales sin el frontend, y posteriormente se realizar pruebas en conjunto.

Figura 59

Peticiones desde Postman para Evaluar la Respuesta del Backend

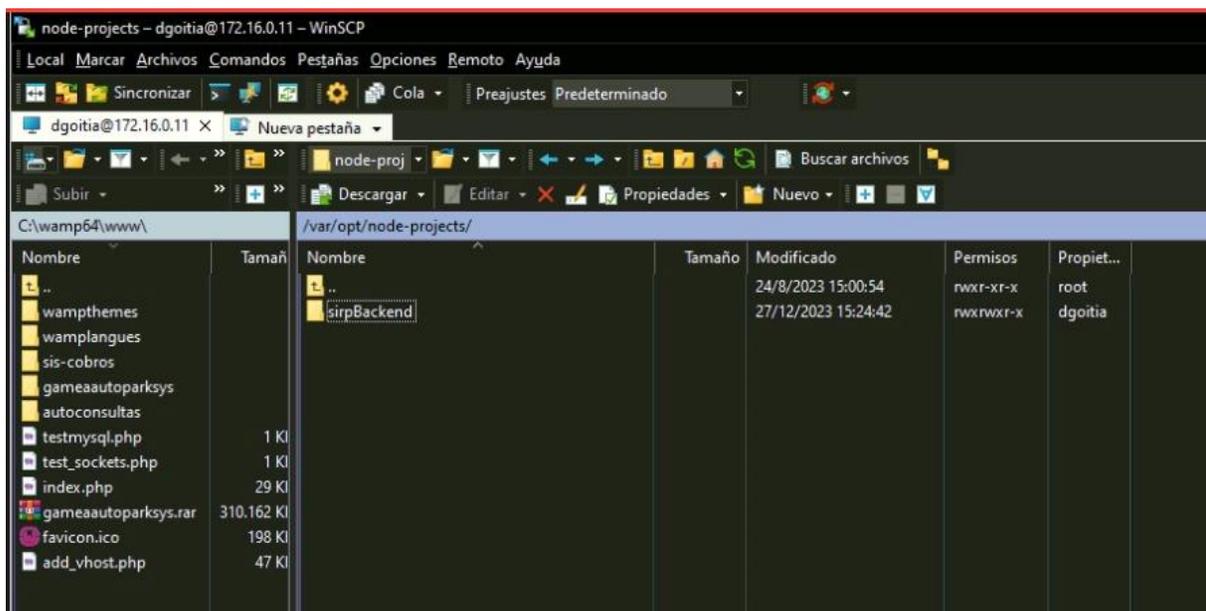


Nota: Se evalúa las peticiones de tipo GET(Obtener), POST(Registrar), PUT(Actualizar) y analiza el resultado de cada una de ellas.

3.2.1.4.2 Implementación en Servidores. Para la implementación del proyecto en el servidor, se utilizó WinSCP, un cliente SFTP gráfico para Windows que emplea SSH. Su interfaz intuitiva permitió una transferencia segura y eficiente de los archivos necesarios desde la máquina local al servidor remoto. Se estableció una conexión protegida, y mediante el uso de la funcionalidad de arrastrar y soltar de WinSCP, se subieron los archivos de configuración, scripts y aplicaciones web al directorio correspondiente del servidor. Posteriormente, se verificaron los permisos y propiedades de los archivos para asegurar su correcta ejecución. Este proceso fue crucial para garantizar que la aplicación se desplegara correctamente y estuviera operativa para su acceso en línea.

Figura 60

Implementación en servidores utilizando la herramienta WinScp



CAPITULO IV
CALIDAD, COSTOS,
SEGURIDAD Y
PRUEBAS



INGENIERÍA
DE SISTEMAS

UNIVERSIDAD PÚBLICA DE EL ALTO

4. Capítulo 4: Calidad, Costo, Seguridad y Pruebas

4.1 Introducción

En este capítulo se realiza una evaluación integral del software desarrollado, abordando aspectos críticos como la calidad, el costo y la seguridad. La evaluación de la calidad se basa en el modelo FURPS (Funcionalidad, Usabilidad, Confiabilidad, Rendimiento y Soporte), utilizando cuestionarios dirigidos al usuario final para evaluar cómo el software cumple con sus expectativas y necesidades específicas.

En cuanto a la estimación de costos, se aplicó el modelo COCOMO II (Constructive Cost Model) para calcular el presupuesto necesario para el desarrollo del producto. Este modelo proporciona una estimación detallada basada en características como el tamaño del proyecto, la complejidad y el entorno de desarrollo.

En el ámbito de la seguridad, se plantearon políticas de seguridad para garantizar la protección de datos y la integridad del sistema, además se describen las características implementadas durante el desarrollo del software, tales como controles de acceso, cifrado de datos y auditorías regulares. Finalmente, se presentan recomendaciones clave para mejorar la seguridad del sistema, incluyendo prácticas de seguridad adicionales y medidas preventivas para fortalecer la protección del software frente a amenazas externas e internas.

También se realizaron diversas pruebas, tales como pruebas de caja blanca para evaluar la estructura interna del software, pruebas de caja negra para validar la funcionalidad desde la perspectiva del usuario, y pruebas de estrés para evaluar la capacidad y estabilidad del sistema bajo condiciones de carga máxima.

Este capítulo proporciona un análisis que evalúa el desempeño del software desde diferentes perspectivas, también establece bases sólidas para asegurar su calidad, gestionar costos efectivamente y garantizar la seguridad integral del sistema desarrollado.

4.2 Medición y Evaluación de la Calidad del Software mediante el Modelo FURPS

Se llevó a cabo un análisis de la calidad del software, centrándose en los aspectos fundamentales de Funcionalidad, Usabilidad, Fiabilidad, Rendimiento y Soporte (FURPS). Este enfoque estructurado permitirá evaluar de manera integral cada aspecto del software en desarrollo. La Funcionalidad se evaluará en términos de cumplimiento de los requisitos y la capacidad para realizar las tareas previstas. La Usabilidad se analizará para garantizar una experiencia de usuario intuitiva y satisfactoria. La Fiabilidad se examinará para asegurar que el software funcione de manera consistente y sin fallos. El Rendimiento se medirá para verificar que el software pueda manejar la carga esperada de manera eficiente. Finalmente, el Soporte se evaluará para asegurar que el software sea fácil de mantener y actualizar en el futuro. Este análisis detallado permitirá identificar áreas de mejora y garantizar que el software cumpla con los objetivos previstos.

Se utilizaron cuestionarios específicos para evaluar los factores FURPS, que abarcan Funcionalidad, Usabilidad, Fiabilidad, Rendimiento y Soporte. Estos cuestionarios constan de una serie de preguntas diseñadas para obtener una comprensión detallada de la percepción y experiencia de los usuarios con respecto al sistema desarrollado. Cada pregunta está cuidadosamente elaborada para abordar aspectos clave de cada factor, como la capacidad funcional, la facilidad de uso, la fiabilidad de los datos, el rendimiento del sistema y el soporte del código. Las respuestas proporcionadas por los usuarios en estos cuestionarios se utilizarán como insumo crucial para identificar áreas de mejora y perfeccionamiento en el software, garantizando así que cumpla con los estándares de calidad deseados y satisfaga las necesidades de los usuarios de manera efectiva y eficiente.

A continuación, se detallan todas las preguntas realizadas a diversos tipos de usuarios del sistema, incluyendo administradores, revisores y usuarios generales. Estas preguntas están diseñadas para evaluar de manera integral la experiencia y percepción que cada grupo

tiene sobre el sistema que utilizan. Al recopilar las opiniones de estos diferentes roles, se pretende obtener una visión completa y detallada de cómo el sistema cumple con las necesidades y expectativas de sus usuarios, identificar áreas de mejora y reconocer los aspectos positivos que deben mantenerse y potenciarse.

Tabla 12

Preguntas para Evaluar los Factores FURPS

PREGUNTAS DE EVALUACIÓN	
Funcionalidad	
CÓDIGO	Pregunta
FU01	¿Qué tan satisfecho estás con la capacidad del sistema para organizar y clasificar tus registros personales?
FU02	¿Cómo calificarías la facilidad de uso del sistema para ingresar y editar tus registros personales?
FU03	¿Consideras que el sistema cumple adecuadamente con el despliegue en diferentes dispositivos?
FU04	¿Qué tan confiable encuentras la seguridad y la privacidad de tus datos personales dentro del sistema?
FU05	¿Cómo evaluarías la capacidad del sistema para generar Certificaciones o Cartas basadas en tus registros personales?
FU06	¿Qué tan satisfactorio encuentras el soporte técnico proporcionado por el sistema?
FU07	¿Cuál es tu nivel de satisfacción con la velocidad de respuesta del sistema?
FU08	¿Qué tan efectivas consideras las funciones de búsqueda y filtrado del sistema?
FU09	¿Qué tan intuitiva te parece la interfaz de usuario del sistema?
FR01	¿Consideras que la información ingresada en el sistema es precisa y confiable en relación con los datos originales que debes revisar?
FR02	¿Encuentras que la información registrada en el sistema es fácil de comprender y verificar durante el proceso de revisión?
FR03	¿Qué tan útil encuentras la función de búsqueda del sistema para localizar información específica dentro de tus registros?

FA01	¿Cómo evaluarías la efectividad del sistema en la gestión y administración de los registros personales de los usuarios?
FA02	¿Qué tan intuitivo encuentras el proceso de búsqueda y recuperación de información dentro del sistema?
FA03	¿Cuál es tu opinión sobre la organización y la claridad de la interfaz del sistema para acceder al control de roles?

Usabilidad

CÓDIGO **Pregunta**

UU01	¿Encuentras que el proceso de ingreso de tus datos personales en el sistema es intuitivo y fácil de seguir?
UU02	¿Qué tan claro te resulta el diseño de las pantallas de registro de datos personales en el sistema?
UU03	¿Te resulta sencillo realizar modificaciones o actualizaciones en tus registros personales una vez que están ingresados en el sistema?
UU04	¿Qué tan rápido puedes encontrar la información específica que necesitas dentro del sistema?
UR01	¿Consideras que la interfaz del sistema facilita la identificación de errores o inconsistencias en los datos registrados?
UR02	¿Qué tan fácil es para ti acceder a las funciones de ayuda o soporte cuando tienes dificultades para utilizar el sistema?
UR03	¿Cuál es tu experiencia al utilizar las herramientas de edición y corrección de datos dentro del sistema?
UA01	¿Encuentras que el panel de administración del sistema proporciona herramientas claras y eficientes para realizar tareas de mantenimiento y gestión?
UA02	¿Qué tan intuitivo es el proceso para configurar permisos y roles de usuario en el sistema?
UA02	¿Qué tan efectivo encuentras el sistema para agregar y distribuir comunicados importantes utilizando las herramientas disponibles en el panel de administración?

Fiabilidad

CÓDIGO **Pregunta**

CU01	¿Con qué frecuencia has experimentado problemas de fiabilidad, como caídas del sistema o pérdida de datos, al utilizar el sistema?
------	------------------------------------------------------------------------------------------------------------------------------------

CU02	¿Cómo evaluarías la estabilidad del sistema en términos de su capacidad para mantenerse en línea y accesible cuando lo necesitas?
CU03	¿Cuál es tu nivel de confianza en la precisión y consistencia de los datos registrados en el sistema?
CR01	¿Qué tan seguro te sientes con respecto a la integridad de los datos que revisas en el sistema?
CR02	¿Has notado alguna vez inconsistencias en la forma en que se manejan los datos dentro del sistema durante el proceso de revisión?
CA01	¿Has experimentado alguna vez fallos del sistema que hayan afectado la disponibilidad de los datos para los usuarios?
CA02	¿Cuál es tu nivel de confianza en la capacidad del sistema para proteger los datos confidenciales de los usuarios?
Rendimiento	
CÓDIGO	Pregunta
DU01	¿Cómo calificarías la velocidad de carga y respuesta del sistema al interactuar con él?
DR01	¿Has experimentado retrasos significativos al realizar operaciones o tareas dentro del sistema?
DR02	¿Has experimentado algún retraso significativo al revisar o procesar los datos dentro del sistema?
DA01	¿Has tenido problemas con la estabilidad del sistema mientras realizas tareas de revisión, como bloqueos o caídas inesperadas?
DA02	¿Has tenido problemas con la estabilidad del sistema mientras realizas tareas administrativas críticas, como configuraciones o actualizaciones?
Soporte	
CÓDIGO	Pregunta
MA01	¿Se sigue un estilo de escritura coherente y estándar para la codificación del sistema, como CamelCase o PascalCase, para garantizar la consistencia y legibilidad del código?
MA02	¿El despliegue del sistema se realiza de manera eficiente en diferentes plataformas, asegurando su disponibilidad y accesibilidad en diversos entornos operativos?
MA03	¿Es fácil ajustar la configuración de la conexión a la base de datos según las necesidades?

4.2.2 Recopilación de Datos

Se realizaron encuestas dirigidas a tres grupos distintos: 10 usuarios, 3 revisores y 3 administradores del sistema, con el objetivo de obtener una perspectiva completa sobre la experiencia y percepción del sistema. Para evaluar cinco aspectos clave, se implementó una escala Likert con valoraciones del 1 al 5, asignando peso a las siguientes opciones: Muy satisfactorio, Satisfactorio, Regular, Insatisfactorio y Muy insatisfactorio. Cada grupo participante recibió un cuestionario específicamente diseñado que se detalla en la sección de anexos para capturar sus opiniones y evaluaciones de manera precisa.

Cabe mencionar que los criterios son el total de las preguntas realizadas a los usuarios finales del sistema.

Tabla 13

Factores FURPS que Aplican Escala Likert

FACTORES FURPS QUE APLICAN ESCALA LIKERT					
Factores	No	Actores		Criterios	Total
			Tipo		
FUNCIONALIDAD	10		Usuarios	9	90
	3		Revisores	3	9
	3		Administradores	3	9
USABILIDAD	10		Usuarios	4	40
	3		Revisores	3	9
	3		Administradores	3	9
FIABILIDAD	10		Usuarios	3	30
	3		Revisores	2	6
	3		Administradores	2	6
RENDIMIENTO	10		Usuarios	1	10
	3		Revisores	2	6
	3		Administradores	2	6
SOPORTE	2		Responsable	3	6
Total				40	236

Nota: En esta tabla se especifica los criterios que se consideran en cada cuestionario con respecto a los factores FURPS.

4.2.3 Resultados Obtenidos

4.2.3.1 Análisis del Factor Funcionalidad. Tras examinar cómo los usuarios perciben la funcionalidad del sistema, se evidencia un alto nivel de satisfacción. Es notable que el 85% de las respuestas otorgaron la calificación más alta, lo que refleja un fuerte respaldo hacia su desempeño. Por otro lado, solo un reducido 2% de las respuestas manifestaron una percepción menos favorable. Estos resultados revelan una tendencia mayoritaria hacia la satisfacción con el sistema, como se detalla a continuación.

Tabla 14

Valoración de la Funcionalidad del Sistema

VALORACIÓN DE LA FUNCIONALIDAD DEL SISTEMA		
	Frecuencia	Porcentaje
MUY SATISFACTORIO	92	85%
SATISFACTORIO	8	7%
REGULAR	4	4%
INSATISFACTORIO	2	2%
MUY INSATISFACTORIO	2	2%
Total	108	100%

4.2.3.2 Análisis del factor Usabilidad. El análisis de usabilidad revela que el 71% de las respuestas se ubicaron en la categoría de puntaje más alto en la escala, lo que indica un amplio consenso entre los usuarios sobre la eficacia del sistema SIRP. Sin embargo, es importante destacar que existe un porcentaje alto de usuarios que se mantienen neutrales en su evaluación. Este grupo representa el 16% de los encuestados, como se muestra en la siguiente tabla.

Tabla 15

Valoración de la Usabilidad del Sistema

VALORACIÓN DE LA USABILIDAD DEL SISTEMA		
	Frecuencia	Porcentaje
MUY SATISFACTORIO	41	71%
SATISFACTORIO	4	7%

REGULAR	9	16%
INSATISFACTORIO	2	3%
MUY INSATISFACTORIO	2	3%
Total	58	100%

Estos resultados destacan una marcada tendencia hacia una experiencia positiva de uso del sistema, con una mayoría significativa de usuarios expresando altos niveles de satisfacción.

4.2.3.3 Análisis del factor Fiabilidad. Al analizar las respuestas de los usuarios, destaca que, por tercera vez consecutiva, la categoría más alta sigue siendo la más común, representando un 76% del total. Es importante resaltar que cerca del 10% de esta valoración indica insatisfacción con la fiabilidad de la información. La evaluación de la fiabilidad del sistema se distribuye de la siguiente manera.

Tabla 16

Valoración de la Fiabilidad del Sistema

VALORACIÓN DE LA FIABILIDAD DEL SISTEMA		
	Frecuencia	Porcentaje
MUY SATISFACTORIO	32	76%
SATISFACTORIO	3	7%
REGULAR	3	7%
INSATISFACTORIO	2	5%
MUY INSATISFACTORIO	2	5%
Total	42	100%

4.2.3.4 Análisis del factor Rendimiento. La valoración del rendimiento del sistema, basada en una muestra de 22 preguntas, refleja que la mayoría de los usuarios percibe el sistema de manera positiva: el 91% lo califica como satisfactorio o muy satisfactorio, mientras que solo un 9% lo considera regular. No se registraron opiniones negativas.

Tabla 17*Valoración del Rendimiento del Sistema*

VALORACIÓN DEL RENDIMIENTO DEL SISTEMA		
	Frecuencia	Porcentaje
MUY SATISFACTORIO	17	77%
SATISFACTORIO	3	14%
REGULAR	2	9%
INSATISFACTORIO	0	0%
MUY INSATISFACTORIO	0	0%
Total	22	100%

4.2.3.5 Análisis del factor Soporte. El soporte del sistema SIRP es su punto más débil, ya que solo el 50% de las respuestas lo califican como satisfactorio. Además, hay una división igual entre respuestas regulares y satisfactorias, lo que indica que hay áreas que se podrían mejorar en el soporte del sistema.

Tabla 18*Valoración del Soporte de Sistema*

VALORACIÓN DEL SOPORTE DEL SISTEMA		
	Frecuencia	Porcentaje
MUY SATISFACTORIO	0	0%
SATISFACTORIO	3	50%
REGULAR	3	50%
INSATISFACTORIO	0	0%
MUY INSATISFACTORIO	0	0%
Total	6	100%

4.2.4 Análisis global

Al resumir los factores evaluados a través de la escala de valoración, se observa una evaluación mayoritariamente positiva, con un total del 86% de respuestas favorables. Sin embargo, existe un margen para mejora, ya que un 9% de los encuestados califican el sistema como regular y un 6% muestran insatisfacción.

Tabla 19

Valoración Considerando todos los aspectos FURPS.

VALORACIÓN GENERAL DEL SISTEMA						
ESCALA	Positivo		Neutral	Negativo		Total
	5	4	3	2	1	
FUNCIONALIDAD	92	8	4	2	2	108
USABILIDAD	41	4	9	2	2	58
FIABILIDAD	32	3	3	2	2	42
RENDIMIENTO	17	3	2	0	0	22
SOPORTE	0	3	3	0	0	6
FRECUENCIA	182	21	21	6	6	236
PORCENTAJE	77%	9%	9%	3%	3%	100%
	86%		9%	5%		

4.3 Estimación de Costos y Esfuerzos del Software mediante el Modelo COCOMO II

En la fase de definición del marco teórico, se tomó la decisión estratégica de emplear COCOMO II como la metodología principal para la estimación de costos. Esta metodología proporciona una estructura robusta y confiable para evaluar los recursos financieros requeridos.

Al emplear COCOMO II, se podrá calcular los costos tomando en cuenta la complejidad y una serie de factores diversos relacionados con nuestro proyecto. Este procedimiento se desarrollará a lo largo de 15 pasos específicos. Los resultados de este proceso se mostrarán en la Figura 54, un formulario organizado con columnas, filas y celdas que registrarán los resultados obtenidos. Dicho formulario estará disponible al finalizar esta secuencia de pasos.

Paso 1: Se identifican los módulos que conforman el sistema y se les asigna un número y un nombre, ingresándolos en las columnas 1 y 2.

Paso 2: Para determinar el tamaño de cada módulo, es necesario calcular cuántas líneas de código fuente tendrá cada uno aproximadamente. Este número, expresado como SLOC (Líneas de Código Fuente), dará una idea del alcance y la complejidad de cada parte del sistema y estos valores los ingresamos en la columna 3 de la.

Paso 3: Una tarea clave en el proyecto es determinar el tamaño del sistema en términos de SLOC (Líneas de Código Fuente). Esto implica sumar todos los módulos que previamente se identificaron y sumar las líneas de código fuente. Una vez que se haya calculado esta cifra, será anotada en la celda 28. Este paso proporcionará una medida cuantitativa del alcance y la complejidad del proyecto, lo que será fundamental para una planificación y gestión efectivas del desarrollo.

Paso 4: Ahora se procede a calcular el Factor Exponencial de Escala (B), que es crucial para el modelo COCOMO II. Este factor se determina considerando cinco factores diferentes, que son PREC (Precisión), FLEX (Flexibilidad), RESL (Resolución), TEAM (Equipo) y MAT (Madurez). Para el proyecto, se asumirá que estos factores están en un nivel nominal, lo que significa que no hay circunstancias excepcionales que influyan significativamente en ellos. Calcular este factor ayudará a ajustar las estimaciones de esfuerzo y costos de manera más precisa, lo que es esencial para una planificación efectiva del proyecto.

$$B = 1.01 + 0.01 \times (3.72 + 3.04 + 4.24 + 3.29 + 4.68) = 1.1997 = 1.20$$

Paso 5: Ahora se procede a calcular dos métricas importantes para el proyecto. En primer lugar, se determina el Esfuerzo Nominal requerido para desarrollar el sistema, que se representa como $PM_{Nominal}$, y se registra en la celda 29. Esta medida dará una estimación del esfuerzo total necesario para completar el proyecto.

En segundo lugar, se calcula la Productividad del Proyecto y se registra en la celda 30. Esta métrica ayudará a entender la eficiencia con la que se está utilizando los recursos para avanzar

en el desarrollo del sistema. Ambas medidas son fundamentales para monitorear y gestionar el progreso del proyecto de manera efectiva

$$PM_{Nominal} = (KSLOC/PM)_{Nominal} = 2,94 * (27.163)^{1.20} = 154.56$$

$$Productividad_{Nominal} = (KSLOC/PM_{Nominal}) = \frac{27163}{154.56} = 175.74$$

Paso 6: En la columna 22 de la tabla de registro, se procede a calcular y anotar el Esfuerzo Nominal por Módulo $PM_{Nominal, Módulo}$. Este valor se obtiene dividiendo el tamaño del módulo (registrado en la columna 3) por la Productividad del Proyecto (especificada en la celda 30). Este paso es crucial para establecer una medida cuantitativa del esfuerzo requerido para cada módulo, lo que facilitará una planificación y gestión más efectivas del desarrollo del proyecto.

Paso 7: Se evaluarán los factores de costo de cada módulo, asignando niveles de "Muy Bajo" a "Extra Alto" según su importancia. Los valores obtenidos se usarán como multiplicadores de esfuerzo para completar el análisis. Esto permitirá una estimación de costo más precisa, considerando aspectos como fiabilidad (RELY), tamaño de la base de datos (DATA), complejidad del producto (CPLX), reutilización de software (RUSE), documentación (DOCU), restricciones de tiempo (TIME), restricciones de almacenamiento (STOR), volatilidad de la plataforma (PVOL), habilidad del analista (ACAP), habilidad del programador (PCAP), continuidad del personal (PCON), experiencia en aplicaciones (AEXP), experiencia en plataforma (PEXP), experiencia en lenguajes y herramientas (LTEX), uso de herramientas (TOOL), capacidad del sitio (SITE) y presiones de calendario (SCED).

Los multiplicadores de esfuerzo reflejarán la influencia de cada factor en la carga de trabajo y los recursos necesarios. Este proceso asegura una estimación más precisa del costo y es fundamental para asignar recursos adecuados a cada módulo, facilitando así una planificación efectiva del proyecto todos estos valores se registrarán en las columnas del 4 al 20.

Paso 8: Para obtener el Factor de Ajuste del Esfuerzo (EAF) para cada módulo, se multiplicarán los multiplicadores de esfuerzo de la columna 4 a la 20 para cada fila. Los

resultados de estas multiplicaciones se ingresarán en la columna 21 de la tabla. Este proceso permitirá calcular el impacto combinado de todos los factores de costo en el esfuerzo requerido para el desarrollo de cada módulo, proporcionando así una valoración más precisa.

Paso 9: Calcular el Esfuerzo Estimado por Módulo $PM_{Estimado,Módulo}$ en la columna 23. Esto se realiza multiplicando el valor de $PM_{Nominal,Módulo}$ en la columna 22 por el Factor de Ajuste (EAF_M) correspondiente en la columna 21.

Paso 10: Sumar los valores calculados en el paso anterior para determinar el Esfuerzo Estimado Total del Sistema ($PM_{Estimado}$). Registrar este valor en la celda 31 de la tabla.

Paso 11: Calcular el Tiempo de Desarrollo Estimado del proyecto (TDEV) y registrar este valor en la celda 34.

$$TDEV = [3.0 \times 22^{0.33+0.2x(1.2-1.01)}] \times \frac{100\%}{100} = 9.56$$

Paso 12: Registrar en la columna 24 el Costo por Mes-Persona de cada módulo, expresado en miles de dólares. Luego, multiplicar estos costos por los $PM_{Estimado,Módulo}$ correspondientes de la columna 23 para obtener el Costo Estimado de cada módulo, y anotar estos valores en la columna 25.

Paso 13: Sumar los valores obtenidos en el paso anterior para calcular el costo total del sistema y registrar este valor en la celda 32.

Paso 14: Para cada módulo, calcular y registrar en la columna 26 el Costo por Instrucción en dólares estadounidenses (\$US), que se obtiene dividiendo el Costo de Desarrollo (columna 25) entre el Tamaño del Módulo (columna 3) de la tabla.

Paso 15: Para cada módulo, calcular y registrar en la columna 27 la Productividad, que se obtiene dividiendo el Tamaño del Módulo (columna 3) entre el Esfuerzo Estimado por módulo $PM_{Nominal,Módulo}$, (columna 23).

Figura 61

Formulario para la Estimación de Esfuerzo, Costo y Tiempo de Desarrollo Utilizando COCOMO II.

Número Modulo	Nombre Modulo	SLOC	Producto					Plataforma				Personal					Proyecto				EAF	PM Nominal Mes-Pers	PM Estimado Mes-Pers	Costo Mes-Pers Dolares	Costo	Costo x Instrucc Dolares	Productividad SLOC/Mes-Pers	
			RELY	DATA	CPLX	RUSE	DOCU	TIME	STOR	PVOL	ACAP	PCAP	PCON	AEXP	PEXP	LTEX	TOOL	SITE	SCED									
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27		
1	Módulo Datos Personales	2003,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	1,00	1,00	0,73	1,00	1,00	0,64	2,40	1,54	300	460,80	0,23	1304,04		
2	Módulo Datos Complementarios	1502,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	0,87	1,00	0,73	1,00	1,00	0,55	2,75	1,52	300	455,84	0,30	988,50		
3	Módulo Datos Academicos	1756,00	1,00	1,00	0,85	1,00	1,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	1,00	1,00	0,73	1,00	1,00	0,54	2,56	1,38	300	414,59	0,24	1270,64		
4	Módulo Estudios Cursados	1200,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	1,00	1,00	0,73	1,00	1,00	0,64	2,34	1,49	300	445,84	0,37	807,46		
5	Módulo Seguro, Bienes y Rentas	1900,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	0,87	1,00	1,00	0,87	1,00	1,00	1,00	0,73	1,00	1,00	0,55	2,10	1,16	300	348,10	0,18	1637,47		
6	Módulo Experiencia Laboral	1800,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	1,00	1,00	0,73	1,00	1,00	0,64	2,30	1,46	300	438,22	0,24	1232,26		
7	Módulo Documentos Institucionales	1562,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	1,00	1,00	0,73	1,00	1,00	0,55	2,30	1,27	300	381,25	0,24	1229,11		
8	Módulo Familiares Dependientes	1150,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	1,00	0,87	0,73	1,00	1,00	0,55	2,12	1,17	300	351,41	0,31	981,75		
9	Módulo Datos de Discapacidad	1925,00	1,00	1,00	1,00	1,00	0,87	1,00	0,87	0,87	1,00	1,00	1,00	1,00	1,00	1,00	0,73	1,00	1,00	0,48	2,30	1,11	300	331,69	0,17	1741,09		
10	Módulo Usuarios	1400,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	1,00	1,00	0,73	1,00	1,00	0,64	2,63	1,67	300	501,09	0,36	838,17		
11	Módulo Roles	1300,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	1,00	1,00	0,73	1,00	1,00	0,64	2,32	1,47	300	442,03	0,34	882,29		
12	Módulo Comunicados	1336,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	1,00	1,00	0,73	1,00	1,00	0,64	2,63	1,67	300	501,09	0,38	799,85		
13	Módulo Revision de Informacion	1650,00	1,00	1,00	0,87	1,00	0,87	1,00	1,00	0,87	1,00	1,00	1,00	1,00	0,87	1,00	0,73	1,00	1,00	0,42	2,35	0,98	300	294,84	0,18	1678,87		
14	Módulo Certificaciones	1452,00	0,54	1,00	1,00	1,00	1,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	1,00	1,00	0,73	1,00	1,00	0,34	2,76	0,95	300	283,97	0,20	1533,99		
15	Módulo Documentos Firmados	1366,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	1,00	1,00	0,73	1,00	1,00	0,64	2,20	1,40	300	419,17	0,31	977,66		
16	Módulo Comunicados	2240,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	1,00	1,00	0,73	1,00	1,00	0,64	2,73	1,73	300	520,15	0,23	1291,94		
17	Módulo Informacion	1621,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	0,87	1,00	1,00	1,00	1,00	1,00	1,00	0,73	1,00	1,00	0,64	2,23	1,42	300	424,88	0,26	1144,55		
	28	27163,00	Total SLOC										Esfuerzo Estamado PM est										31	23,38	32	7014,97	33	1161,65
	29	154,56	Esfuerzo PM(nominal)										Tiempo de Desarrollo										34	9,56		Costo Total Estimado		Productividad Estimada
	30	175,74	Productividad {KSLOC/PM}nominal																									
	35	1,20	Factor Exponencial de Escala (B)																									
	36	300,00	Costo Mes-Persona \$																									

4.4 Seguridad Informática y Protección de Datos

A continuación, se plantean las siguientes medidas de seguridad para asegurar el correcto funcionamiento del sistema y proteger los datos de manera integral, con el objetivo primordial de garantizar la robustez, integridad y seguridad del software desarrollado frente a las amenazas cibernéticas actuales. Las medidas propuestas están alineadas con las mejores prácticas del sector de desarrollo de software, buscando no solo asegurar la eficiencia operativa de la aplicación, sino también establecer un entorno digital seguro y adaptable en un contexto de continua evolución tecnológica.

4.4.1 Planteamiento de las Políticas de Seguridad

4.4.1.1 Mantenimiento del Software. El mantenimiento del software implica gestionar y mejorar un sistema después de su implementación inicial, incluyendo la corrección de errores, actualizaciones de seguridad, optimización del rendimiento y nuevas funcionalidades. A continuación, se plantean políticas de seguridad para proteger el software.

Tabla 20

Mantenimiento de Software

Política de Seguridad	Descripción
Implementación de Parches y Actualizaciones	Establecer procedimientos para aplicar regularmente parches de seguridad y actualizaciones.
Pruebas Previa a la Implementación de Cambios	Realizar pruebas exhaustivas antes de implementar cambios en el software en producción.
Control de Versiones	Utilizar sistemas de control de versiones para mantener un registro de cambios y revertir si es necesario.
Gestión de Licencias	Garantizar que todas las licencias de software estén actualizadas y sean legales como ser el SSL.
Monitoreo de Vulnerabilidades	Implementar herramientas para monitorear y remediar vulnerabilidades conocidas en el software.

4.4.1.2 Manejo de la Base de Datos. El manejo de la base de datos implica administrar eficientemente la información almacenada para garantizar su integridad y disponibilidad. A continuación, se plantean las siguientes políticas para optimizar su gestión y asegurar un funcionamiento óptimo.

Tabla 21

Manejo de la Base de Datos

Política de Seguridad	Descripción
Backup y Recuperación de Datos	Establecer procedimientos regulares de backup y pruebas de restauración de datos.
Acceso y Privilegios	Limitar el acceso a la base de datos según roles en la institución.
Encriptación de Datos Sensibles	Encriptar datos sensibles almacenados en la base de datos para proteger la confidencialidad.
Auditorías y Monitoreo	Realizar auditorías periódicas de la base de datos y monitorear el acceso no autorizado.

4.4.1.3 Manejo de Archivos. El manejo y cuidado de archivos es crucial ya que almacena respaldos de información presentada en formato físico. A continuación, se presentan una serie de políticas destinadas a mejorar estos procesos.

Tabla 22

Manejo de Archivos

Política de Seguridad	Descripción
Almacenamiento Seguro	Almacenar archivos en ubicaciones seguras con acceso controlado y auditoría de acceso.
Encriptación de Archivos Sensibles	Encriptar archivos que contengan datos sensibles para proteger la confidencialidad.
Control de Acceso y Versiones	Mantener control de versiones de archivos y limitar el acceso basado en roles y necesidades.
Escaneo de Archivos en Busca de Malware	Implementar escaneos periódicos de archivos para detectar y eliminar malware.

Respaldos Regulares	Realizar copias de seguridad programadas de los archivos para garantizar la disponibilidad y recuperación en caso de pérdida de datos.
Integridad y Autenticidad	Implementar mecanismos para garantizar la integridad y autenticidad de los archivos almacenados, como firmas digitales o hashes criptográficos.

4.4.1.4 Planteamiento de Políticas para Usuarios. El establecimiento de políticas claras para los usuarios es fundamental para asegurar un uso seguro y eficiente de los sistemas informáticos dentro de la institución. A continuación, se muestran algunas políticas a nivel usuario que buscan mantener la seguridad y confidencialidad de los datos.

Tabla 23

Usuarios del Sistema

Política de Seguridad	Descripción
Política de Contraseñas Seguras	Requerir contraseñas robustas y cambios periódicos.
Capacitación en Seguridad	Ofrecer formación regular sobre buenas prácticas de seguridad y conciencia sobre amenazas cibernéticas.
Acceso Basado en Roles	Asignar permisos de acceso según roles definidos y el principio de menor privilegio.
Monitoreo de Actividades de Usuario	Registrar y monitorear las actividades de usuario para detectar comportamientos sospechosos.
Política de Uso	Establecer reglas claras sobre el uso de la información que genera el sistema y responsabilidad sobre ella.

4.4.1.5 Administradores del Sistema. Los administradores del sistema deben gestionar el acceso de manera restringida, desarrollar nuevos módulos en entornos controlados, y vigilar de forma continua las vulnerabilidades en los despliegues futuros. A continuación, se plantean políticas a nivel administrador.

Tabla 24*Administradores del Sistema*

Política de Seguridad	Descripción
Acceso Controlado	Limitar el acceso administrativo a personas autorizadas.
Registro y Auditoría de Actividades Administrativas	Registrar y auditar todas las acciones administrativas realizadas en el sistema.
Segregación de Entornos de Desarrollo, Pruebas y Producción	Mantener ambientes separados para desarrollo, pruebas y producción para evitar riesgos.
Gestión de Vulnerabilidades	Implementar herramientas y procedimientos para identificar y mitigar vulnerabilidades en el sistema.
Plan de Respuesta a Incidentes	Desarrollar y probar un plan de respuesta a incidentes para actuar eficazmente ante emergencias de seguridad.

4.4.1.6 Integración de Nuevos Módulos. En la fase de escalado del sistema, es crucial implementar políticas para prevenir caídas y garantizar la estabilidad. A continuación, se presentan algunas políticas de seguridad clave para esta etapa.

Tabla 25*Desarrollo de Módulos Nuevos*

Política de Seguridad	Descripción
Revisión de Código y Pruebas	Realizar revisión de código y pruebas exhaustivas antes de integrar nuevos módulos al sistema.
Evaluación de Seguridad de Terceros	Verificar la seguridad de componentes de terceros integrados al sistema antes de su implementación.
Pruebas de Penetración	Realizar pruebas de penetración regulares para identificar y corregir vulnerabilidades.
Política de Desarrollo Seguro	Implementar prácticas de desarrollo seguro.
Auditorías de Seguridad en Nuevos Módulos	Realizar auditorías de seguridad periódicas en nuevos módulos integrados al sistema.

4.4.2 Características de Seguridad Implementadas

4.4.2.1 Estructura de Archivos. Para asegurar la integridad y seguridad de los archivos del sistema, se implementan las siguientes medidas:

- Se adopta una estructura de archivos clara y controlada. Los archivos se organizan en directorios jerárquicos, facilitando su localización y manejo.
- Se implementaron visualización de información por usuario logueado y rol de acceso al recurso digitalizado.
- Solo el personal autorizado tiene acceso a directorios y archivos críticos. Esto se logra con la implementación de un servidor dedicado al sistema SIRP.
- Implementación de validación de extensiones de archivos mediante funciones específicas integradas en una línea de verificación de datos ingresados, asegurando la integridad y seguridad de la información.
- Se implementó validación por volumen de información para evitar la carga de archivos con gran cantidad de datos, garantizando un funcionamiento eficiente del sistema.

4.4.2.2 Autenticación y Autorización. Para validar la identidad de los usuarios y gestionar las sesiones de forma segura, se utiliza los “JSON Web Tokens (JWT)”, los cuales permiten acceso solo a usuarios autenticados y autorizados.

- Utilizar sesiones HTTPS y cookies seguras para la transmisión de tokens JWT.
- Política de Contraseñas Fuertes: Establecer requisitos de complejidad para las contraseñas, incluyendo longitud mínima, combinación de caracteres especiales, números y letras mayúsculas y minúsculas.

4.4.2.3 Control de Acceso. Se establecieron cuatro niveles de usuario de la aplicación: Revisor, Usuario, Coordinador y Administrador cada uno con permisos y roles específicos para un control de acceso preciso.

- Revisor: Permisos de lectura y verificación de datos.

- Usuario: Permisos para registrar y actualizar su propia información.
- Coordinador: Permisos para gestionar información dentro de su área de responsabilidad.
- Administrador: Permisos completos para gestionar el sistema, usuarios y configuración.

4.4.2.4 Unicidad de Nombres de Usuario. Para prevenir problemas de seguridad se establece que los nombres de usuario registrados deben ser únicos para cada caso por lo cual será su número de carnet, esto evitará la duplicidad de registros y permitirá un control más preciso al momento de gestionar quien accede al sistema o revocar este permiso de ser necesario.

4.4.2.5 Validación Rigurosa. Todos los campos de los formularios de registro y actualización cuentan con validaciones exhaustivas dependiendo del tipo de dato que contendrán, para mantener la coherencia y seguridad de la información que será almacenada en la base de datos.

4.4.3 Protocolo ante Incidentes de Seguridad

4.4.3.1 Detección y Respuesta. Para la “detección temprana” de incidentes de seguridad, se debe monitorear constantemente el sistema y utilizar herramientas automatizadas que alerten. Una vez detectado un incidente, se debe responder con rapidez, lo que permite mitigar los efectos del incidente de manera eficiente y minimizar cualquier posible daño.

4.4.3.2 Notificación de Incidentes. En caso de un incidente de seguridad, se notifica de inmediato a los responsables del área de desarrollo de software y a las partes interesadas. Cada acción tomada en respuesta al incidente debe documentarse, desde la detección inicial hasta la resolución final. Esta documentación será esencial para entender la naturaleza del incidente, las respuestas dadas y para servir como un registro para futuras auditorías y mejoras de seguridad.

4.4.3.3 Revisiones Post-Incidentes. Una vez un incidente de seguridad ha sido resuelto, se revisa exhaustivamente el mismo. Analizando las causas raíz, evaluando cómo se manejó el incidente y determinando qué se puede mejorar. Esta revisión ayudará a fortalecer las políticas y procedimientos de seguridad, y a desarrollar planes de acción para prevenir incidentes similares en el futuro.

4.4.4 Recomendaciones de Seguridad

A continuación, se presentan algunas recomendaciones clave para asegurar la integridad y seguridad del sistema, abordando aspectos críticos como la validación de datos ingresados, la selección de opciones, la gestión de archivos y la verificación de tokens de acceso, entre otros.

Tabla 26

Recomendaciones de seguridad

ASPECTO DE SEGURIDAD	RECOMENDACIÓN	FRECUENCIA
ACTUALIZACIÓN DE CONTRASEÑAS	Los usuarios deben actualizar sus contraseñas regularmente para prevenir accesos no autorizados y mantener la seguridad. Implementar políticas que requieran contraseñas fuertes y únicas.	Cada 3 meses.

GENERACIÓN DE BACKUPS	Realizar copias de seguridad de todos los datos críticos para asegurar la recuperación en caso de pérdida de datos. Utilizar almacenamiento externo y cifrado para mayor seguridad.	Diario y semanal (con almacenamiento externo)
REVISIÓN DE PERMISOS DE ACCESO	Revisar y actualizar los permisos de acceso a sistemas y datos para garantizar que solo personal autorizado tenga acceso.	Trimestral
CAPACITACIÓN EN SEGURIDAD	Capacitar al personal en prácticas de seguridad de la información para mantenerlos informados sobre las mejores prácticas y amenazas actuales. Ofrecer cursos y simulacros de seguridad.	Anual
AUDITORÍAS DE SEGURIDAD	Realizar auditorías internas y externas para evaluar el cumplimiento de políticas y detectar posibles vulnerabilidades. Documentar hallazgos y planes de mitigación.	Anual
ACTUALIZACIÓN DE SOFTWARE Y SISTEMAS	Actualizar regularmente el software y sistemas operativos para aplicar parches de seguridad y corregir vulnerabilidades conocidas. Automatizar las actualizaciones cuando sea posible.	Mensual
GESTIÓN DE INCIDENTES DE SEGURIDAD	Implementar un plan de respuesta a incidentes y realizar simulacros para preparar al personal ante posibles amenazas. Establecer un equipo de respuesta a incidentes. Establecer un equipo de respuesta a incidentes.	Semestral
MONITOREO DE ACTIVIDADES	Monitorear y registrar las actividades en los sistemas de información para detectar y responder a actividades sospechosas.	Continuo

PROTECCIÓN CONTRA MALWARE	Utilizar y actualizar herramientas antivirus y antimalware para proteger los sistemas contra software malicioso. Realizar análisis regulares y actualizar las definiciones de virus.	Continuo
ORDENAMIENTO DE DOCUMENTOS	Clasificar y almacenar documentos según su nivel de sensibilidad para asegurar la confidencialidad y facilitar el acceso controlado. Implementar políticas de retención y destrucción de documentos.	Mensual

4.5 Pruebas

Inicialmente, se realizaron pruebas de caja blanca para verificar la lógica interna del código y asegurar que cada ruta de ejecución y condición fuese evaluada correctamente. Estas pruebas se enfocaron en la estructura y el flujo del código, garantizando que las funciones individuales y los módulos interactúen como se esperaba.

Posteriormente, se llevaron a cabo pruebas de caja negra. Este enfoque se centró en evaluar la funcionalidad del sistema desde una perspectiva externa, sin considerar el funcionamiento interno del código. Se probaron diversos escenarios y casos de uso para validar que el sistema cumpliera con los requisitos funcionales especificados, asegurando que las entradas proporcionadas producían las salidas esperadas.

Para culminar, se realizaron pruebas de estrés, destinadas a evaluar la robustez y la estabilidad del sistema bajo condiciones extremas. Estas pruebas implicaron someter el sistema a una alta carga de peticiones para determinar su capacidad de manejo y su punto de ruptura. Se midió el rendimiento y la capacidad de respuesta del sistema, asegurando que pudiera soportar el volumen de solicitudes previsto sin degradación significativa del rendimiento.

Las pruebas mencionadas serán descritas con más detalle en las siguientes tablas.

4.5.1 Pruebas de Caja Blanca

Las pruebas de caja blanca son esenciales para asegurar que el software funcione según lo diseñado internamente, garantizando su calidad y fiabilidad antes de su despliegue en producción es por eso que se evaluó las siguientes pruebas.

Tabla 27

Casos de Prueba de Caja Blanca.

CASOS	RESULTADOS ESPERADOS	APROBADO
ACCESO AL SISTEMA	Para ingresar al sistema debe validarse las Credenciales de acceso, siendo estas el nombre de usuario y la contraseña. Si alguna de estas es incorrecta entonces se le notifica al usuario el error presentado y se mantiene la ventana de Login hasta que se ingrese los datos correctos.	Si
REGISTRO Y ACTUALIZACIÓN DE USUARIOS	Al momento de registrar o actualizar algún usuario, se debe verificar que todos los datos requeridos en el formulario de carácter obligatorio sean llenados, además de validar correctamente los tipos de datos ingresados en cada campo (textos, números, etc.) de no ser así se debe notificar al usuario para que los complete.	Si
VISTA PRINCIPAL DE LOS USUARIOS	Una vez se tienen las condiciones correctas para el envío de los datos del formulario se confirma el registro o actualización mediante la interacción del usuario con un mensaje de confirmación. Al culminar un registro o una actualización, y después de los correspondientes mensajes de confirmación del cambio realizado, la vista	

<p>REGISTRO Y ACTUALIZACIÓN DE INFORMACIÓN PERSONAL</p>	<p>principal donde se visualizan los datos registrados en una tabla ordenada, debe actualizarse automáticamente.</p> <p>Esta tabla debe permitir realizar búsquedas, filtrar la información y ordenar la información correspondiente a los usuarios.</p> <p>Al momento de registrar los datos personales de cada postulante, se debe verificar que todos los datos requeridos en el formulario de carácter obligatorio sean llenados, además de validar correctamente los tipos de datos ingresados en cada campo (textos, números, fotos, o documentos pdf si correspondiera) de no ser así se debe notificar para que se completen.</p>	Si
<p>REGISTRO Y ACTUALIZACIÓN DE INFORMACIÓN COMPLEMENTARIA</p>	<p>Una vez se tienen las condiciones correctas para el envío de los datos del formulario se confirma el registro mediante la interacción del usuario con un mensaje de confirmación.</p> <p>Al momento de registrar la información complementaria, se debe verificar que todos los datos requeridos en el formulario de carácter obligatorio sean llenados, además de validar correctamente los tipos de datos ingresados en cada campo (textos, números, fotos, o documentos pdf si correspondiera) de no ser así se debe notificar para que se completen.</p>	Si
<p>REGISTRO Y ACTUALIZACIÓN DE</p>	<p>Al momento de registrar la información académica de cada postulante, se debe</p>	Si

INFORMACIÓN ACADÉMICA	<p>verificar que todos los datos requeridos en el formulario de carácter obligatorio sean llenados, además de validar correctamente los tipos de datos ingresados en cada campo (textos, números, fotos, o documentos pdf si correspondiera) de no ser así se debe notificar para que se completen.</p> <p>Una vez se tienen las condiciones correctas para el envío de los datos del formulario se confirma el registro mediante la interacción del usuario con un mensaje de confirmación.</p>	
REGISTRO Y ACTUALIZACIÓN DE ESTUDIOS REALIZADOS	<p>Al momento de registrar información sobre los estudios realizados de cada postulante, se debe verificar que todos los datos requeridos en el formulario de carácter obligatorio sean llenados, además de validar correctamente los tipos de datos ingresados en cada campo (textos, números, fotos, o documentos pdf si correspondiera) de no ser así se debe notificar para que se completen.</p> <p>Una vez se tienen las condiciones correctas para el envío de los datos del formulario se confirma el registro mediante la interacción del usuario con un mensaje de confirmación.</p>	Si
REGISTRO Y ACTUALIZACIÓN DE SEGURO DE SALUD Y DECLARACIONES JURADAS	<p>Al momento de registrar los datos de seguros de salud de cada postulante y las declaraciones, se debe verificar que todos los datos requeridos en el formulario de carácter obligatorio sean llenados, además de validar correctamente los tipos de datos ingresados en cada campo (textos, números, fotos, o documentos pdf si correspondiera) de no ser así se debe notificar para que se completen.</p>	Si

<p>REGISTRO Y ACTUALIZACIÓN DE EXPERIENCIA LABORAL</p>	<p>Una vez se tienen las condiciones correctas para el envío de los datos del formulario se confirma el registro mediante la interacción del usuario con un mensaje de confirmación.</p> <p>Al momento de registrar la experiencia laboral de cada postulante, se debe verificar que todos los datos requeridos en el formulario de carácter obligatorio sean llenados, además de validar correctamente los tipos de datos ingresados en cada campo (textos, números, fotos, o documentos pdf si correspondiera) de no ser así se debe notificar para que se completen.</p>	Si
<p>REGISTRO Y ACTUALIZACIÓN DE INFORMACIÓN DE DEPENDIENTES</p>	<p>Una vez se tienen las condiciones correctas para el envío de los datos del formulario se confirma el registro mediante la interacción del usuario con un mensaje de confirmación.</p> <p>Al momento de registrar la información de dependientes de cada postulante, se debe verificar que todos los datos requeridos en el formulario de carácter obligatorio sean llenados, además de validar correctamente los tipos de datos ingresados en cada campo (textos, números, fotos, o documentos pdf si correspondiera) de no ser así se debe notificar para que se completen.</p>	Si
<p>REGISTRO Y ACTUALIZACIÓN DE INFORMACIÓN DE DISCAPACIDAD</p>	<p>Una vez se tienen las condiciones correctas para el envío de los datos del formulario se confirma el registro mediante la interacción del usuario con un mensaje de confirmación.</p> <p>Si el postulante presentara alguna discapacidad, se debe verificar que todos los datos requeridos en el formulario de carácter obligatorio sean llenados, además de validar</p>	Si

<p>MODULO DE REVISIÓN DE DOCUMENTACIÓN</p>	<p>correctamente los tipos de datos ingresados en cada campo (textos, números, fotos, o documentos pdf si correspondiera) de no ser así se debe notificar para que se completen. Una vez se tienen las condiciones correctas para el envío de los datos del formulario se confirma el registro mediante la interacción del usuario con un mensaje de confirmación. Una vez los formularios se encuentren llenados para cada postulante, se podrán visualizar en su módulo correspondiente un listado de todos los postulantes, esta vista principal es donde se visualizan los datos registrados en una tabla ordenada. Esta tabla debe permitir realizar búsquedas, filtrar la información y ordenar la información correspondiente a los postulantes, además se podrá visualizar cada documento subido y cada formulario de cada uno de ellos, para su posterior revisión y actualización de estado: siendo estos estados aprobados o rechazados, además de añadir un campo para observaciones si es que las hubiera en cada formulario.</p>	Si
------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----

4.5.2 Pruebas de Caja Negra

Las pruebas de caja negra evalúan la funcionalidad externa del software sin considerar su estructura interna, asegurando su calidad y usabilidad. En ese contexto se realizaron las siguientes pruebas.

Tabla 28

Casos de Prueba de Caja Negra.

ENTRADAS	SALIDAS	RESULTADOS OBTENIDOS
<p>DATOS INTRODUCIDOS EN EL INICIO DE SESIÓN: USUARIO: MANOLITO CONTRASEÑA: “ “</p>	<p>“Credenciales no válidos.”</p>	<p>Si uno de los datos introducidos no se encuentra registrado en la base de datos, se le notificará el error y se le negará el acceso al sistema.</p>
<p>DATOS INTRODUCIDOS EN EL INICIO DE SESIÓN: USUARIO: “ ” CONTRASEÑA: “123“ O USUARIO: MANOLITX CONTRASEÑA: “123“</p>	<p>“Credenciales no Validos”</p>	<p>Si uno de los datos introducidos no se encuentra registrado en la base de datos, se le notificará el error y se le negará el acceso al sistema.</p>
<p>DATOS INTRODUCIDOS EN EL INICIO DE SESIÓN: USUARIO: MANOLITO CONTRASEÑA: “123“</p>	<p>“Credenciales correctas, bienvenido al sistema”</p>	<p>Ambos campos son válidos en la base de datos, por lo tanto, el usuario puede ingresar al sistema.</p>
<p>LLENADO Y CONFIRMACIÓN DEL ENVÍO DEL FORMULARIO DE REGISTRO O ACTUALIZACIÓN CON TODOS LOS CAMPOS DE ENTRADA COMPLETOS.</p>	<p>- Módulo de Usuarios: Campos de entrada correctamente validados, mensaje de confirmación aceptado. Mensaje: El usuario ha sido Creado / Actualizado correctamente</p>	<p>En cada módulo del sistema, una vez que los datos han sido validados correctamente, se procede a almacenar la información en la base de datos correspondiente. Este almacenamiento se realiza de manera segura y conforme a las políticas de</p>

-
- Módulo de Información privacidad y seguridad de Personal: ¡Excelente! La datos. Además, se generan Información Personal ha registros de actividad que sido ingresada y permiten el seguimiento y validada con éxito. Sus auditoría del proceso. datos personales han La validación exitosa sido también puede creados/actualizados desencadenar otros correctamente en el procesos automatizados sistema. como la generación de
 - Módulo de Información credenciales de acceso o la Complementaria: Se actualización de perfiles y confirma que la registros existentes. Información Complementaria ha sido completada adecuadamente. Esta información adicional ha sido creada/actualizada con precisión.
 - Módulo de Información Académica: La Información Académica ha sido registrada satisfactoriamente. Agradecemos su diligencia en proporcionar un historial académico completo, el cual ha sido creado/actualizado en la base de datos.
 - Módulo de Estudios Realizados: Los detalles
-

de sus Estudios Realizados han sido validados y confirmados. Esta valiosa información educativa ha sido creada/actualizada exitosamente.

- Módulo de Seguro de Salud y Declaraciones Juradas: La información del Seguro de Salud y Declaraciones Juradas ha sido procesada y aceptada. Todos los datos relevantes han sido creados/actualizados para su seguridad y conformidad legal.
 - Módulo de Experiencia Laboral: Su Experiencia Laboral ha sido verificada y confirmada. Nos complace informarle que su experiencia profesional ha sido creada/actualizada en nuestro registro.
 - Módulo de Información de Dependientes: La información sobre sus Dependientes ha sido recibida y validada correctamente. Los
-

**LLENADO E INTENTO
DE ENVÍO DEL
FORMULARIO DE
REGISTRO O
ACTUALIZACIÓN CON
2 CAMPOS DE
ENTRADA
OBLIGATORIOS
INCOMPLETOS**

-
- detalles proporcionados han sido registrados/actualizados para reflejar adecuadamente sus circunstancias familiares.
- Módulo de Información de Discapacidad: Hemos registrado con éxito la información relacionada con la Discapacidad. Aseguramos que todas las adaptaciones necesarias han sido consideradas y los datos han sido registrados/actualizados en nuestro sistema.
 - Módulo de Usuarios: Si un formulario se encuentra Alerta del campo o de los campos vacíos, bloqueo del botón de envío de datos. Si un formulario se encuentra incompleto, el sistema bloqueará el envío del mismo hasta que todos los campos obligatorios sean llenados. Mensaje: Nombre del campo: incompleto, por favor llene los campos obligatorios. Esto es para asegurar la integridad y la completitud de la información requerida. Los campos incompletos serán resaltados o marcados para guiar al usuario a través de las correcciones necesarias. Si el formulario no se envía, la información no se
-

Personal no han sido procesará ni almacenará, y el completados. Por favor, usuario no podrá avanzar en revise y llene cada el proceso hasta que campo requerido para complete todos los pasos asegurar la precisión de requeridos.

sus datos personales antes de continuar con el proceso de registro.

- Módulo de Información Complementaria:

Parece que se han omitido detalles importantes en la sección de Información Complementaria. Es crucial que proporcione toda la información solicitada para que podamos comprender mejor su perfil y necesidades.

- Módulo de Información Académica: La sección de Información Académica contiene campos vacíos que son obligatorios para el registro. Por favor, ingrese su historial académico completo, incluyendo instituciones y calificaciones, para completar esta parte del formulario.
-

-
- Módulo de Estudios Realizados: La sección de Estudios Realizados no está completa. Necesitamos que detalle todos los estudios que ha realizado, ya que esta información es fundamental para evaluar su trayectoria educativa.
 - Módulo de Seguro de Salud y Declaraciones Juradas: La información requerida en el módulo de Seguro de Salud y Declaraciones Juradas es esencial para cumplir con las normativas legales y proteger su bienestar. Por favor, asegúrese de completar todos los campos sin excepción.
 - Módulo de Experiencia Laboral: Observamos que falta información en la sección de Experiencia Laboral. Es importante que proporcione un resumen detallado de su experiencia laboral previa, incluyendo roles
-

y responsabilidades, para validar su experiencia en el campo.

- Módulo de Información de Dependientes: La sección dedicada a la Información de Dependientes está incompleta. Esta información es vital para garantizar que todos los beneficios y consideraciones sean aplicados correctamente a sus dependientes.

- Módulo de Información de Discapacidad: No ha proporcionado información completa en el módulo de Información de Discapacidad. Es imperativo llenar todos los campos requeridos para asegurar el acceso y las adaptaciones necesarias según sus necesidades específicas.

**VALIDACIÓN DE LOS
FORMULARIOS
REGISTRADOS**

- Formularios aprobados: Durante la validación, se Si todos los campos de presentan dos posibles los formularios cumplen escenarios para los con los criterios de formularios registrados:

-
- validación, y son aprobación o rechazo. Cada aprobados por los formulario es Revisores, el formulario cuidadosamente revisado es aprobado. Esto para asegurar que cumpla significa que la solicitud con los requisitos de ingreso del postulante establecidos. Si los ha sido aceptada y formularios son aprobados, puede proceder al se asigna un estado de siguiente paso en el 'aprobado' y el postulante proceso de admisión o avanza en el proceso. Si un inscripción. formulario no es aprobado,
- 1 o más Formularios no se asigna un estado de 'no aprobados En caso de aprobado' junto con una lista que uno o más detallada de las formularios no cumplan observaciones que deben con los estándares de ser atendidas por el validación debido a postulante. datos faltantes o erróneos, se registran las observaciones correspondientes. Al postulante se le notifica sobre las inconsistencias y se le brinda la oportunidad de actualizar y corregir su información.
-

4.5.3 Pruebas de Estrés

Se llevó a cabo pruebas de estrés en el módulo de destino para evaluar su rendimiento bajo condiciones de carga extrema. El objetivo es identificar cuellos de botella, determinar el

límite de capacidad y asegurar que el módulo puede manejar altos niveles de tráfico de manera eficiente. A continuación, se describe el proceso detallado para realizar estas pruebas:

Figura 62

Módulo Inicial de Acceso y Registro Personal

The screenshot shows the 'I. DATOS PERSONALES' form in the SIRP application. The form is divided into several sections:

- PERSONAL DATA:**
 - NOMBRE COMPLETO: ERNESTO PERALTA MAMANI
 - DOCUMENTO: CI
 - NUMERO DE DOCUMENTO: 5015261
 - EXPEDIDO: LP
 - ESTADO CIVIL: SOLTERO/A
 - PAIS DE NACIMIENTO: BOLIVIA
 - NACIONALIDAD: BOLIVIANA
- DATE AND LOCATION:**
 - FECHA DE NACIMIENTO: 13 de mayo de 2002
 - LOCALIDAD DE NAC.: NUESTRA SE ORA DE LA PAZ
 - PROVINCIA DE NAC.: MURILLO
 - DEPARTAMENTO DE NAC.: LA PAZ
- Documentos PDF:**

#	PDF	Fecha
1	DocumentosPersonales1.pdf	14/12/2023
2	DocumentosPersonales2.pdf	15/12/2023
3	DocumentosPersonales3.pdf	16/12/2023
4	DocumentosPersonales4.pdf	17/12/2023
5	DocumentoKardex.pdf	14/12/2023
- Imágenes:**

#	IMG	Fecha
1	foto.joep	15/12/2023
2	fotodePerfil.joep	14/12/2023
3	Ernesto_Peralta.PNG	20/06/2024
- Other details:**
 - Fecha de Registro: 25 de septiembre de 2024
 - Button: Editar Registros

Nota: A continuación, se muestran los resultados obtenidos con la herramienta Apache JMeter, considerando las siguientes consultas por un periodo de 1 minuto en este módulo de inicio:

Tabla 29

Resultados de Pruebas de Estrés

N°	Muestra	Media	Min	Max	Desviación	% error	Rendimiento
1	40	266	0	844	65.67	5.83 %	3.21/min
2	100	257	0	844	70.74	4.17 %	11.1/min
3	200	50	0	844	71.52	2.39 %	15.5/min
4	300	241	0	844	67.83	1.79 %	22.1/min
5	500	236	0	844	63.16	1.97 %	33.3/min

Donde:

- **Muestra:** es la cantidad de datos para el caso de ejecución.

- **Media:** es el tiempo promedio transcurrido de un conjunto de resultados.
- **Min:** es el tiempo transcurrido más bajo para la muestra.
- **Max:** es el tiempo transcurrido más largo para la muestra.
- **Desviación estándar:** es la variabilidad del tiempo transcurrido de la muestra.
- **Error %:** indica el porcentaje de solicitudes con errores.
- **Rendimiento:** mide las solicitudes por segundo, minuto y hora.

Se concluye que los errores al momento de las consultas son relativamente bajos en comparación con el tiempo total transcurrido durante las pruebas. Esto indica que la mayoría de las solicitudes se completaron con éxito dentro de un margen de tiempo razonable. Además, la desviación estándar es aceptable, lo que sugiere que el tiempo de respuesta de las consultas es consistente y predecible, con pocas variaciones significativas entre las solicitudes individuales. En resumen, el módulo demuestra una buena capacidad para manejar altos niveles de tráfico con una tasa de error mínima y un rendimiento estable.

INGENIERÍA DE SISTEMAS
UNIVERSIDAD PÚBLICA DE EL ALTO

CAPITULO V
CONCLUSIONES Y
RECOMENDACIONES



**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO

5. Capítulo 5: Conclusiones y Recomendaciones

5.1 Introducción

En este capítulo final, se presentan las conclusiones derivadas del análisis y desarrollo del sistema, así como una serie de recomendaciones basadas en los hallazgos obtenidos. Las conclusiones resumen los logros alcanzados y las lecciones aprendidas a lo largo del proyecto, mientras que las recomendaciones proporcionan directrices para futuras mejoras y prácticas óptimas para mantener la eficiencia, seguridad y efectividad del sistema. Estas observaciones y sugerencias son esenciales para asegurar la continuidad y evolución positiva del proyecto, garantizando su alineación con los objetivos institucionales y las necesidades de los usuarios.

5.2 Conclusiones

Basado en los objetivos y el desarrollo del sistema de información web descrito, se puede hacer las siguientes conclusiones:

- Mediante la digitalización y categorización de documentos, se logró reducir la acumulación y el entrapapelado, facilitando un acceso más rápido y ordenado a la información crucial para los funcionarios.
- La implementación de procesos automatizados para la revisión y llenado de información, junto con módulos segmentados e intuitivos, contribuyó significativamente a minimizar errores en el registro y verificación de datos. Esto mejoró la precisión de la información, evitando decisiones erróneas basadas en datos incorrectos.
- La integración de un sistema de archivo digital permitió almacenar y recuperar fácilmente documentos importantes como archivos PDF e imágenes, agilizó los procesos de contratación y ayudo a que no se genere duplicidad de información innecesaria.

- La adopción de una base de datos NoSQL, como MongoDB, garantizó la integridad, disponibilidad a largo plazo y accesibilidad de la documentación de respaldo. Esta estructura robusta facilitó la gestión de registros precisos y confiables, esenciales para futuras verificaciones y auditorías.
- La creación de una plataforma web personalizada proporcionó una interfaz intuitiva para el registro y visualización de información relevante sobre funcionarios. Esto promovió una gestión eficiente y accesible de los datos, mejorando la productividad institucional al facilitar el acceso rápido y seguro a la información necesaria.

En conclusión, el Sistema de Información Web desarrollado ha alcanzado con éxito el objetivo central de centralizar, gestionar y controlar la documentación de los funcionarios. Este avance significativo ha optimizado de manera efectiva el proceso de seguimiento previo a la contratación, asegurando la integridad y la fiabilidad de los datos. Mediante la implementación de tecnologías modernas como Tailwind CSS para una interfaz personalizada, MongoDB como base de datos NoSQL para persistir datos y Redux para mejorar la experiencia del usuario, se ha logrado mejorar sustancialmente la experiencia del usuario y la eficiencia del sistema. Además, la utilización de OpenUp ha sido crucial para la elaboración integral del proyecto debido a su metodología estructurada que facilita la planificación, desarrollo y entrega de software de manera eficiente, organizada y con enfoque iterativo.

5.3 Recomendaciones

Para garantizar la efectividad y seguridad de un sistema de información web que centraliza y gestiona documentación relevante de funcionarios, es crucial considerar las siguientes recomendaciones:

- Dado que el sistema almacena documentos de gran relevancia, es esencial implementar medidas estrictas de seguridad para el resguardo del servidor. Esto incluye el uso de firewalls, sistemas de detección de intrusos y cifrado de datos.

- Es importante socializar el sistema de manera constante, ya que el personal de la institución puede cambiar frecuentemente. Programas regulares de capacitación y actualización garantizarán que todos los usuarios estén familiarizados con el sistema y sus funcionalidades.
- Implementar políticas de seguridad claras y estrictas para regular el acceso y la gestión en las áreas del servidor, dado que cualquier acceso no autorizado podría comprometer la seguridad de los datos personales de los funcionarios. Estas políticas deben incorporar controles de acceso basados en roles, credenciales seguras para el acceso al servidor, y monitoreo continuo para resguardar la información sensible de manera efectiva.
- Mantener el software del sistema actualizado constantemente es vital para mitigar vulnerabilidades que puedan surgir con el tiempo. Las actualizaciones deben incluir parches de seguridad, mejoras de rendimiento y nuevas funcionalidades que optimicen la operatividad del sistema.

Siguiendo estas recomendaciones, se puede asegurar que el sistema de información no solo sea eficiente y efectivo, sino también seguro y confiable para la gestión de documentación crítica dentro de la institución.

BIBLIOGRAFÍA

- Ali Ticona, J. R. (2020). *DESARROLLO DE UN SISTEMA WEB DE CONTROL DE REGISTRO DE ARCHIVOS NOTARIALES* [Universidad Pública de El Alto].
<http://repositorio.upea.bo/jspui/handle/123456789/63>
- Arévalo Lizardo, M. E. (2010, diciembre 2). *INTRODUCCIÓN AL PATRÓN DE ARQUITECTURA POR CAPAS*.
<https://arevalomaria.wordpress.com/2010/12/02/introduccion-al-patron-de-arquitectura-por-capas/>
- AWS. (2023). *¿QUÉ ES JAVASCRIPT?* <https://aws.amazon.com/es/what-is/javascript/>
- Blancarte, O. (s. f.). *ARQUITECTURA EN CAPAS*. Recuperado 25 de octubre de 2023, de <https://reactiveprogramming.io/blog/es/estilos-arquitectonicos/capas>
- Calamani Salas, J. M. (2020). *SISTEMA DE INFORMACIÓN PARA EL REGISTRO Y SEGUIMIENTO ACADÉMICO* [Universidad Pública de El Alto].
<http://repositorio.upea.bo/jspui/handle/123456789/223>
- Carrión, J. (s. f.). *DIFERENCIA ENTRE DATO, INFORMACIÓN Y CONOCIMIENTO*. Recuperado 24 de octubre de 2023, de <http://iibi.unam.mx/voutssasmt/documentos/dato%20informacion%20conocimiento.pdf> <http://biblioteca.udgvirtual.udg.mx/jspui/handle/123456789/869>
- CLOUDFLARE. (2023). *¿QUÉ ES LA SEGURIDAD DE APLICACIONES WEB?*
<https://www.cloudflare.com/es-es/learning/security/what-is-web-application-security/>
- concepto. (2023a). *DATO*. <https://concepto.de/dato/#ixzz8GdXcg7rG>
- concepto. (2023b). *SISTEMA*. <https://concepto.de/sistema/#ixzz8GY3Tk7j6>
- Diario de QA. (2022, octubre 22). *TIPOS DE PRUEBAS - PRUEBAS DE CAJA BLANCA*. <https://www.diariodeqa.com/post/pruebas-de-caja-blanca>
- DocuWare. (s. f.). *GESTION DOCUMENTAL (DMS)*. Recuperado 27 de octubre de 2023, de <https://start.docuware.com/es/gestion-documental#section1>
- EcuRed. (s. f.). *COCOMO II*. Recuperado 27 de octubre de 2023, de https://www.ecured.cu/COCOMO_II#Caracter.C3.ADsticas
- Fernández Rojas, J. (2023). *HISTORIA DE LA CIUDAD DE EL ALTO*.
<https://www.elalto.gob.bo/historia/>
- Gimson, L. (2015). *DESARROLLO BASADO EN CONOCIMIENTO SIGUIENDO PRÁCTICAS ÁGILES*. Universidad Nacional de La Plata.
- Gobierno Autónomo Municipal de El Alto. (2023a). *MISION*.
<https://www.elalto.gob.bo/mision/>
- Gobierno Autónomo Municipal de El Alto. (2023b). *OBJETIVOS*.
<https://www.elalto.gob.bo/objetivos>

- Gobierno Autónomo Municipal de El Alto. (2023c). *VISIÓN*.
<https://www.elalto.gob.bo/vision/>
- Gobierno Autónomo Municipal de El Alto. (2024). *ORGANIGRAMA, LISTA DE FUNCIONARIOS EN LA INSTITUCIÓN*.
- Gómez Adriana, López María, Migani Silvina, & Otazú Alejandra. (s. f.). *COCOMO-UN MODELO DE ESTIMACION DE PROYECTOS DE SOFTWARE*. Recuperado 27 de octubre de 2023, de
<https://blogadmi1.files.wordpress.com/2010/11/cocomo01full.pdf>
- Hernández Aguilar, A. (2018). *DESARROLLO DE UN SISTEMA WEB DE CONTROL DOCUMENTAL (APLICADO A EMPRESA AUTOMOTRIZ)* [Facultad de Estudios Superiores Cuautitlán]. <https://hdl.handle.net/20.500.14330/TES01000769929>
- HOSTINGER TUTORIALES. (2023a). *QUÉ ES REACT: DEFINICIÓN, CARACTERÍSTICAS Y FUNCIONAMIENTO*.
<https://www.hostinger.es/tutoriales/que-es-react>
- HOSTINGER TUTORIALES. (2023b, abril 19). *¿QUÉ ES NODE.JS?*
https://www.hostinger.es/tutoriales/que-es-node-js#%C2%BFQue_es_Nodejs
- HubSpot. (2021). *QUÉ ES LA ARQUITECTURA EN CAPAS, VENTAJAS Y EJEMPLOS*. <https://blog.hubspot.es/webite/que-es-arquitectura-en-capas>
- Kinsta. (2022, diciembre 19). *¿QUÉ ES EXPRESS.JS?* <https://kinsta.com/es/base-de-conocimiento/que-es-express/>
- Kruchten. (1999). *THE RATIONAL UNIFIED PROCESS*.
- Medium. (s. f.). *IMPLEMENTANDO REDUX EN REACT.JS (REDUX TOOLKIT)*.
- Medium. (2023). *INTRODUCCIÓN A EXPRESS JS*.
<https://medium.com/@aarnlpezsosa/introducci%C3%B3n-a-express-js-a1ebe16dbcf4>
- modelosred. (s. f.). *MODELOS DE CALIDAD FURPS*. Recuperado 23 de octubre de 2023, de https://modelosred.fandom.com/es/wiki/Modelo_de_calidad_FURPS
- Muñoz Cañavate, A., & Chain Navarro, C. (1999). *LOS SISTEMAS DE INFORMACIÓN WEB COMO ELEMENTOS DE DIFUSIÓN DE INFORMACIÓN AL CIUDADANO. ANÁLISIS Y EVOLUCIÓN DE LAS PRESTACIONES OFRECIDAS*.
- OpenWebinars. (2019, octubre 28). *¿QUE ES MONGODB?*
<https://openwebinars.net/blog/que-es-mongodb/>
- OpenWebinars. (2023a). *¿QUÉ ES NODEJS Y PARA QUÉ SIRVE?*
<https://openwebinars.net/blog/que-es-nodejs/>
- OpenWebinars. (2023b). *¿QUÉ ES TAILWIND CSS?* <https://openwebinars.net/blog/que-es-tailwind-css-y-por-que-deberias-usarlo/>

- OriHhs. (s. f.). *MÉTODOS DE RECOLECCIÓN DE INFORMACIÓN*. Recuperado 28 de octubre de 2023, de https://ori.hhs.gov/education/products/sdsu/espanol/eg_info.htm
- Peláez, J. (s. f.). *ARQUITECTURA BASADA EN CAPAS*. Recuperado 25 de octubre de 2023, de <https://geeks.ms/jkpelaez/2009/05/30/arquitectura-basada-en-capas/>
- Peña Ayala, A. (2006). *INGENIERÍA DE SOFTWARE: Una Guía para Crear Sistemas de Información* (Primera Edición).
- Platzi. (s. f.). *PATRÓN ARQUITECTONICO DE CAPAS/LAYERS*. Recuperado 25 de octubre de 2023, de <https://platzi.com/tutoriales/1248-pro-arquitectura/5439-patron-arquitectonico-de-capas-layers/>
- React. (2023). *REACT*. <https://es.legacy.reactjs.org/>
- Ríos Salgado, S., Hinojosa Raza, C., & Delgado Rodriguez, R. (2013). *APLICACIÓN DE LA METODOLOGIA OPENUP EN EL DESARROLLO DEL SISTEMA DE DIFUSIÓN DE GESTIÓN DEL CONOCIMIENTO DE LA ESPE*.
- Rivera Cruz, O. O. (2021). *SISTEMA WEB DE GESTIÓN DE CLIENTES, REGISTRO DE PEDIDOS Y ENVÍOS CASO: FLORERÍA "DALIA"* [Universidad Mayor de San Andrés]. <http://repositorio.umsa.bo/xmlui/handle/123456789/29722>
- Ruiz Oblitas, R. I. (2022). *SISTEMA WEB PARA LA ADMINISTRACIÓN Y CONTROL DE REGISTROS DIARIOS PARA EL ESTUDIO JURÍDICO "VILLEGAS & ASOCIADOS"* [Universidad Mayor de San Andrés]. <http://repositorio.umsa.bo/xmlui/handle/123456789/29720>
- Salas Vega, N. (2021). *DESARROLLO DE UN SISTEMA DE EVALUACIÓN DE POSTULACIONES AL MAGÍSTER EN TECNOLOGÍAS DE LA INFORMACIÓN* [Universidad de Chile]. <https://repositorio.uchile.cl/handle/2250/181752>
- Significados. (2023). *¿QUÉ ES UN SISTEMA?* <https://www.significados.com/sistema/>
- Solano, H., & Torres, I. (2013). *ANÁLISIS DE FRAMEWORKS PARA EL DESARROLLO DE APLICACIONES MÓVILES EN LA PLATAFORMA ANDROID*.
- SOOluciona. (s. f.). *COCOMO II*. Recuperado 23 de octubre de 2023, de <https://sooluciona.com/diferencias-entre-cocomo-1-y-cocomo-2/>
- Testing iT. (2022, julio 8). *PRUEBAS DE ESTRÉS DE SOFTWARE: ¿QUÉ SON Y PARA QUÉ SIRVEN?* <https://www.testingit.com.mx/blog/pruebas-de-estres-de-software>
- Wikipedia. (2023, julio 11). *OPENUP*. <https://es.wikipedia.org/wiki/OpenUP>
- Yahuasi Apaza, E. (2020). *SISTEMA DE INFORMACIÓN WEB DE CONTROL ADMINISTRATIVO* [Universidad Pública de El Alto]. <http://repositorio.upea.bo/jspui/handle/123456789/183>
- ZAPTEST. (2023a). *PRUEBAS DE CAJA BLANCA: ¿QUÉ ES, CÓMO FUNCIONA, RETOS, MÉTRICAS, HERRAMIENTAS Y MÁS!*

<https://www.zaptest.com/es/pruebas-de-caja-blanca-que-es-como-funciona-retos-metricas-herramientas-y-mas>

ZAPTEST. (2023b). *PRUEBAS DE CAJA NEGRA: QUÉ SON, TIPOS, PROCESOS, ENFOQUES, HERRAMIENTAS Y MUCHO MÁS.*

<https://www.zaptest.com/es/pruebas-de-caja-negra-que-son-tipos-procesos-enfoques-herramientas-y-mucho-mas>

ZAPTEST. (2023c). *PRUEBAS DE ESTRÉS EN PRUEBAS DE SOFTWARE: QUÉ ES, TIPOS, PROCESOS, ENFOQUES, HERRAMIENTAS & ¡MÁS!*

<https://www.zaptest.com/es/pruebas-de-estres-en-pruebas-de-software-que-es-tipos-procesos-enfoques-herramientas-mas>



ANEXOS

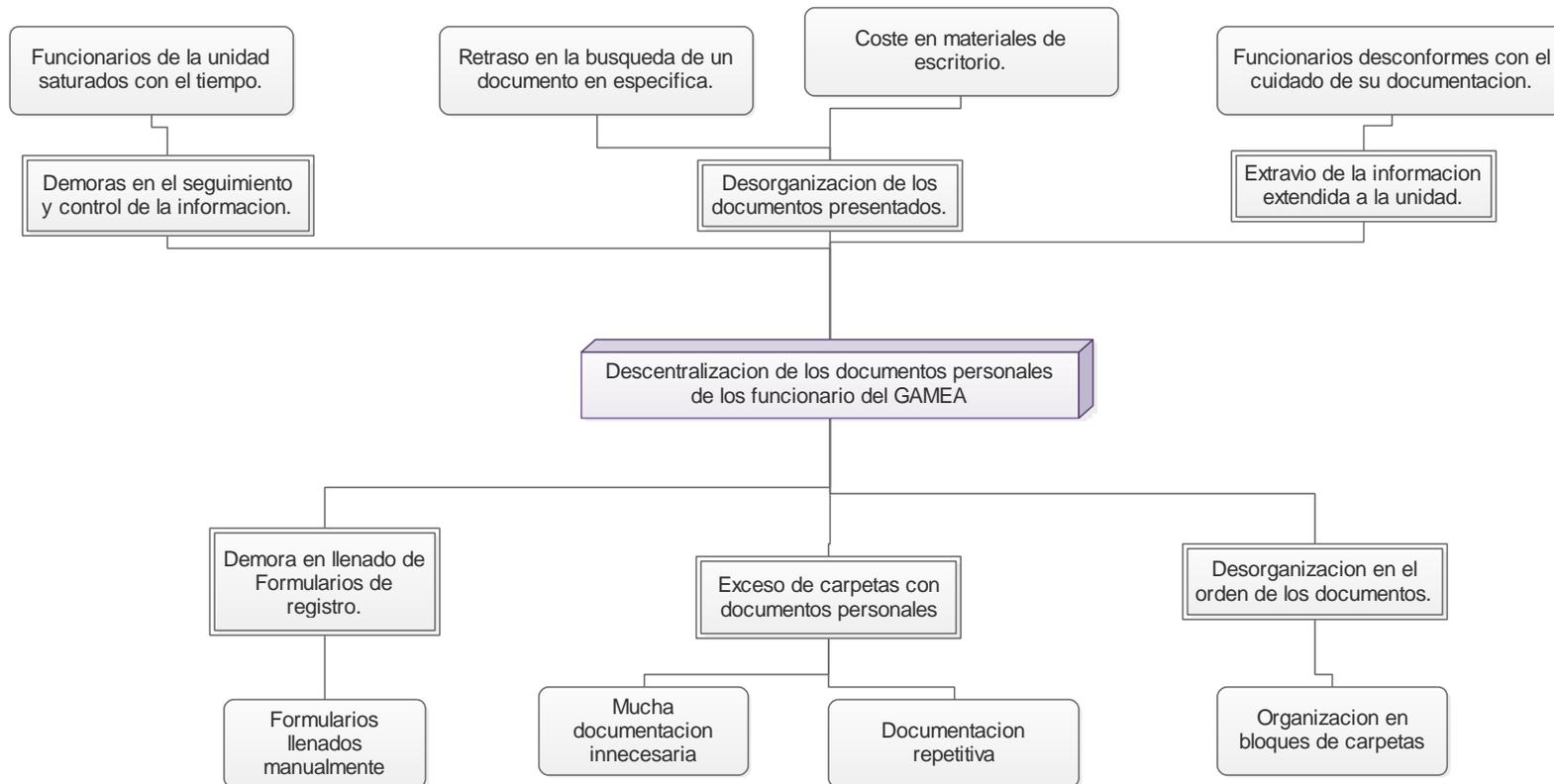


**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO

ANEXOS A

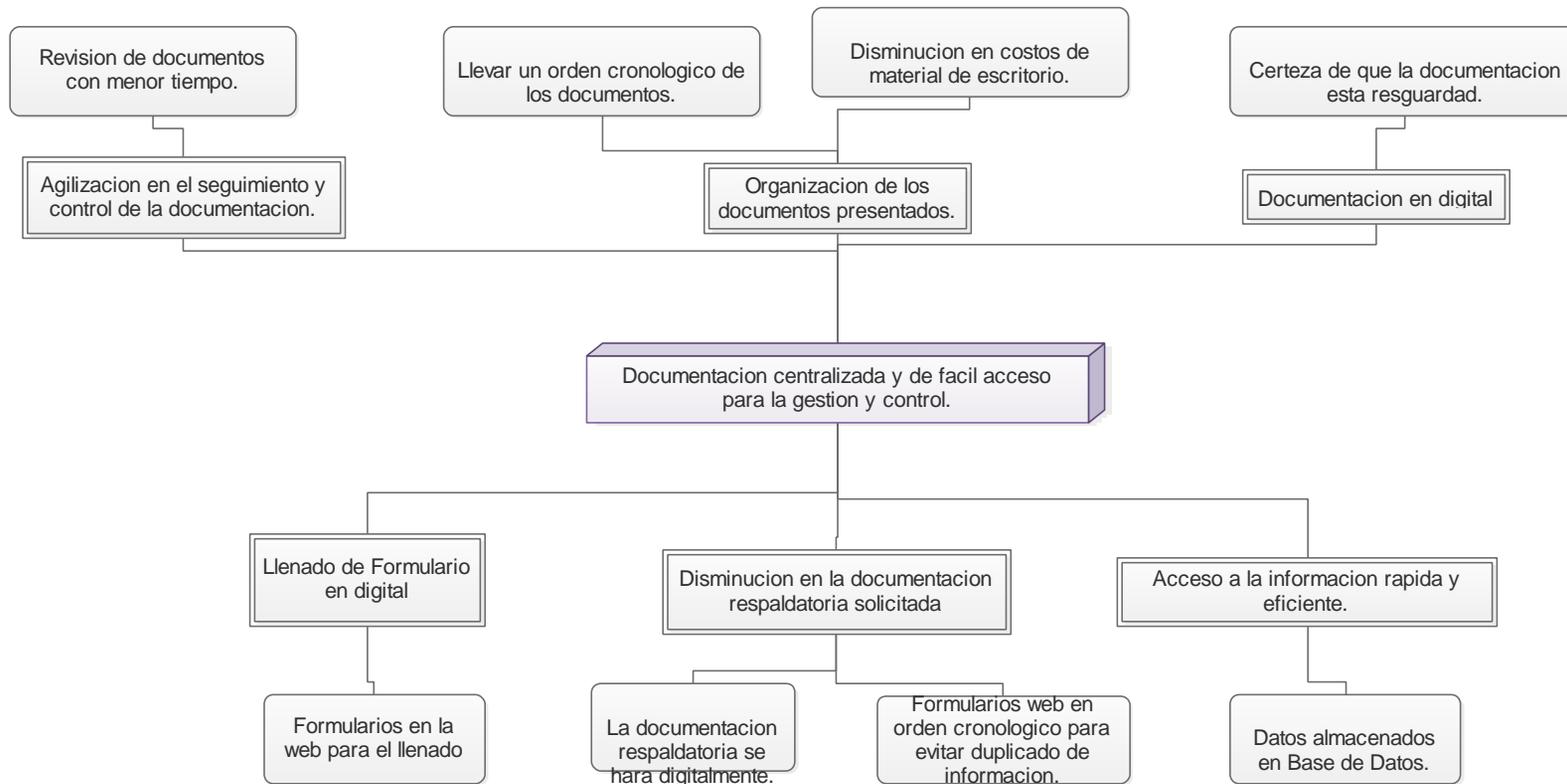
ÁRBOL DE PROBLEMAS, OBJETIVOS

ÁRBOL DE PROBLEMAS



Nota: Relación causal entre el problema central y sus causas y efectos.

ÁRBOL DE OBJETIVOS



Nota: Relación entre los objetivos principales y sus medios y fines.

ANEXOS B

FLUJO DE TRABAJO DE RECEPCIÓN DE DOCUMENTOS EN LA UNIDAD DE REGISTROS



El Alto, Junio de 2024

Señor:
Lic. Ing. William Roque Roque
DIRECTOR DE CARRERA
INGENIERÍA DE SISTEMAS
Presente. -

Ref.: AVAL DE CONFORMIDAD DE LA TUTORA METODOLÓGICA

Distinguido director de carrera:
Mediante la presente tengo a bien de comunicarle mi conformidad del Trabajo de Grado:

TÍTULO: Sistema de Información Web para la Gestión y Control de Documentación Personal

CASO: Unidad de Registros del Gobierno Autónomo Municipal de El Alto

MODALIDAD: Proyecto de grado

Univ. Juan Daniel Mamani Quispe

Registro Universitario: 200019297

Cédula de Identidad: 13606050 LP

Para su defensa pública y evaluación correspondiente a la materia de Taller de Grado II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Atentamente,

M. Sc. Lic. Ing. Fanny Helen Perez Mamani
TUTOR METODOLÓGICO
TALLER DE GRADO II



MANUAL DE USUARIO

SISTEMA DE INFORMACIÓN DE
REGISTROS PERSONALES

“SIRP”

G.A.M.E.A.

Versión V1

UNIDAD DE SISTEMAS

ÍNDICE

1. INTRODUCCIÓN.....	182
2. OBJETIVOS.....	182
3. REQUERIMIENTOS.....	182
3.1. <i>CONOCIMIENTOS BÁSICOS.....</i>	<i>182</i>
3.2. <i>REQUERIMIENTOS DE HARDWARE.....</i>	<i>182</i>
3.3. <i>REQUERIMIENTOS DE SOFTWARE.....</i>	<i>182</i>
3.4. <i>REQUERIMIENTOS DE RED.....</i>	<i>182</i>
4. GENERALIDADES.....	183
4.1. <i>INGRESO AL SISTEMA.....</i>	<i>183</i>
4.2. <i>INICIO DE SESIÓN.....</i>	<i>183</i>
4.3. <i>REGISTRO DE USUARIOS.....</i>	<i>183</i>
5. PAGINA PRINCIPAL.....	184
5.1. <i>INFORMACIÓN.....</i>	<i>184</i>
5.2. <i>COMUNICADOS.....</i>	<i>185</i>
5.3. <i>ESTADÍSTICAS.....</i>	<i>187</i>
6. ADMINISTRACIÓN.....	187
6.1. <i>ROLES.....</i>	<i>187</i>
6.2. <i>USUARIOS.....</i>	<i>189</i>
6.3. <i>COMUNICADOS.....</i>	<i>189</i>
7. FORMULARIOS.....	190

7.1.	<i>DATOS PERSONALES.</i>	190
7.2.	<i>DATOS COMPLEMENTARIOS.</i>	192
7.3.	<i>DATOS ACADÉMICOS.</i>	193
7.4.	<i>ESTUDIOS CURSADOS.</i>	194
7.5.	<i>SEGURO BIENES Y RENTAS.</i>	197
7.6.	<i>EXPERIENCIA LABORAL.</i>	198
7.7.	<i>DOCUMENTOS INSTITUCIONALES.</i>	200
7.8.	<i>FAMILIARES DEPENDIENTES.</i>	201
7.9.	<i>DATOS DE DISCAPACIDAD.</i>	202
8.	REVISIÓN.	204
9.	CERTIFICACIONES.	205
10.	REPORTES.	206

1 INTRODUCCIÓN.

El sistema SIRP “Sistema de información web para el control y gestión de documentación personal” es un sistema que permite tener un control sobre los datos del personal que ingresa a trabajar en la Alcaldía de El Alto.

2 OBJETIVOS.

Instruir al usuario en el uso correcto del sistema y la solución de los problemas que puedan suceder en la cada operación.

3 REQUERIMIENTOS.

3.1 CONOCIMIENTOS BÁSICOS.

Manejo de Navegadores de Internet.

3.2 REQUERIMIENTOS DE HARDWARE.

Computadora de escritorio o personal.

Conexión a internet.

3.3 REQUERIMIENTOS DE SOFTWARE.

Sistema Operativo: Windows (7 o superior)

Navegadores: Google Chrome (3.5 o superior), Opera (3.5 o superior),

Microsoft Edge (3.5 o superior), Mozilla Firefox (3.5 o superior), o navegadores de distribuciones Linux.

3.4 REQUERIMIENTOS DE RED.

Conexión a la intranet de la Alcaldía de El Alto.

4 GENERALIDADES.

4.1 INGRESO AL SISTEMA.

En la url de su navegador ingrese el siguiente enlace: <https://sirp.elalto.gob.bo/>.

4.2 INICIO DE SESIÓN.

En la ventana de la imagen 2 es donde el usuario deberá ingresar sus datos de autenticación para ingresar al sistema.



Imagen 2: Login (Inicio de sesión) del Sistema SIRP

Si los datos ingresados son los correctos, se le permitirá el acceso al sistema como se puede observar en la imagen 4 y si no son los correctos permanecerá en la ventana de Login.

4.3 REGISTRO DE USUARIOS.

En la ventana de Login se encuentra la opción para que los nuevos usuarios se puedan registrar en el sistema, esta sección de registro requiere que se llenen ciertos datos personales para registrar al usuario, como se puede observar en

la imagen 3 (recordar que cada campo solicitado es importante para el registro correspondiente, por ende solo cuando todos los campos estén llenos se le permitirá registrarse).

Numero de Documento 13590658 Complemento 10

Password *****

Confirmar Password *****

Registrarse

Iniciar sesion

Imagen 3: Ventana para el registro de usuarios.

5 PAGINA PRINCIPAL.

5.1 INFORMACIÓN.

Esta es la ventana principal del sistema, como se puede observar en la imagen 4, es una sección completa dedicada a guías sobre como registrar o subir documentación o contenido audiovisual al sistema.

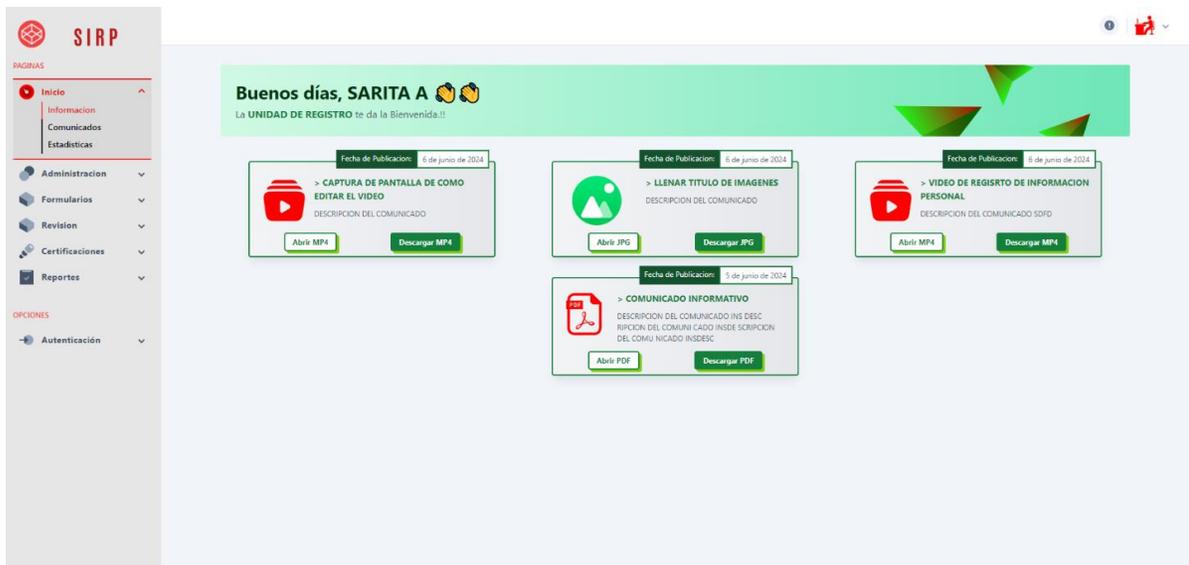


Imagen 4: Sección de Información.

5.2 COMUNICADOS.

En esta sección se visualizan todos los comunicados ya registrados ordenados por la última fecha en la que fueron modificados, teniendo las opciones de visualizarlos y descargarlos.

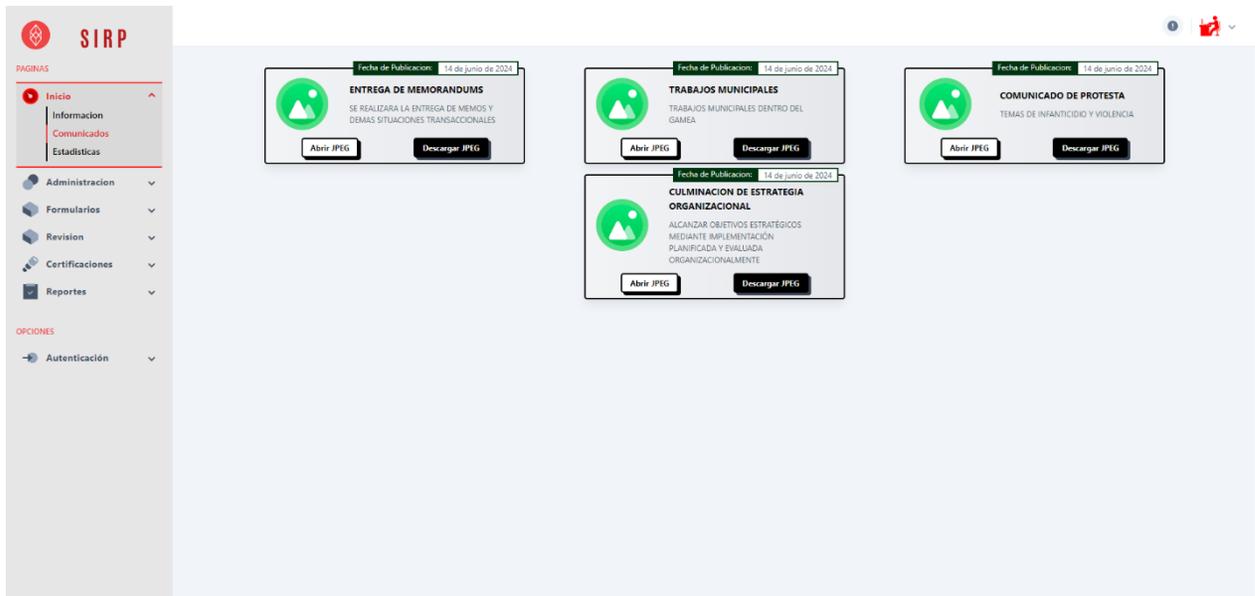


Imagen 5: Vista de la sección de comunicados

- **Crear Comunicados:** En esta sección se indican los campos necesarios para registrar un nuevo comunicado, estos campos deben ser rellenados y validados antes de ser registrados.

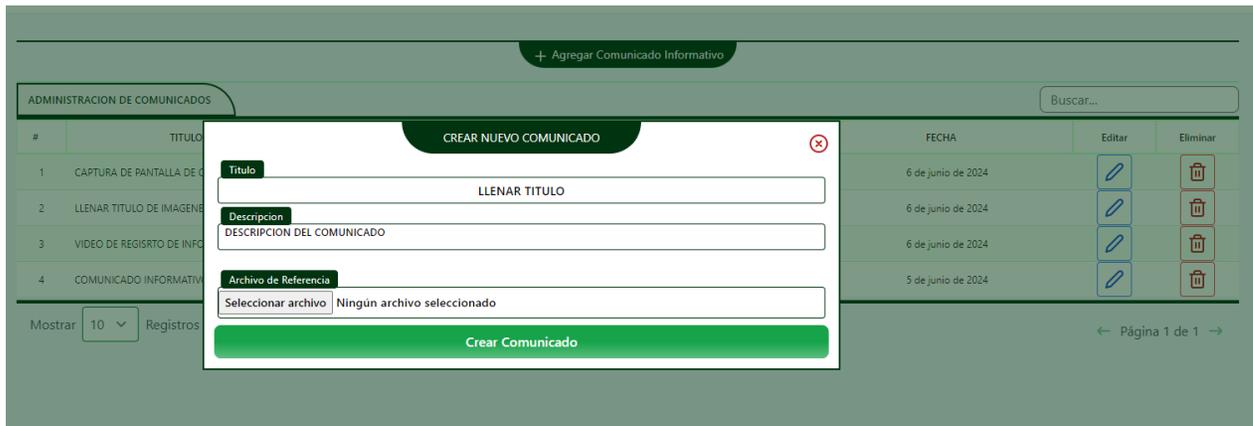


Imagen 6: Crear nuevo comunicado informativo

- **Editar Comunicados:** También se cuenta con la opción de editar y actualizar los comunicados registrados previamente, estos cambios se reflejarán en todo el sistema, de igual forma que en el Registro, todos los campos deben ser rellenados y validados antes de ser registrados.

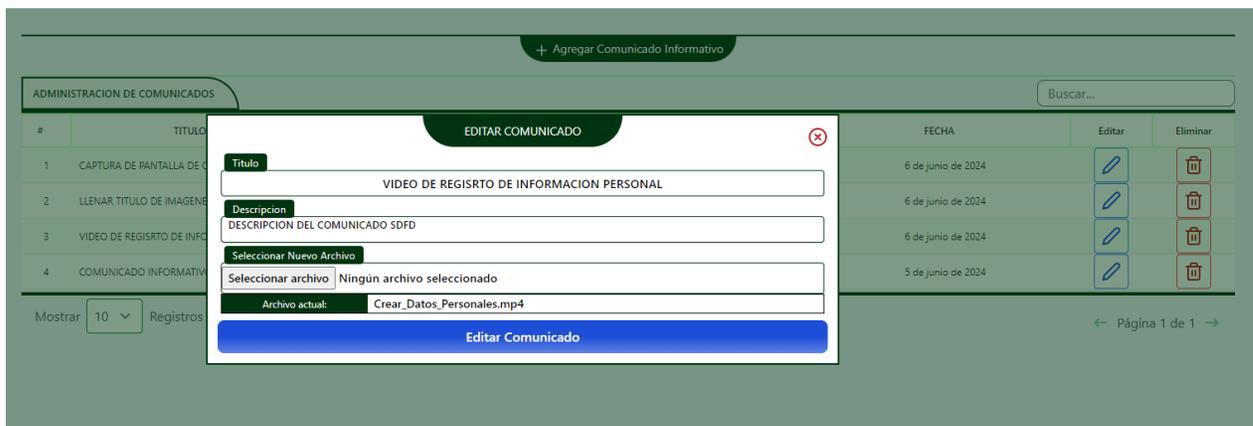


Imagen 7: Editar comunicado informativo

5.3 ESTADÍSTICAS.

En esta sección se visualizarán todos los gráficos estadísticos incluidos en el sistema, como puede observarse en la imagen 8.

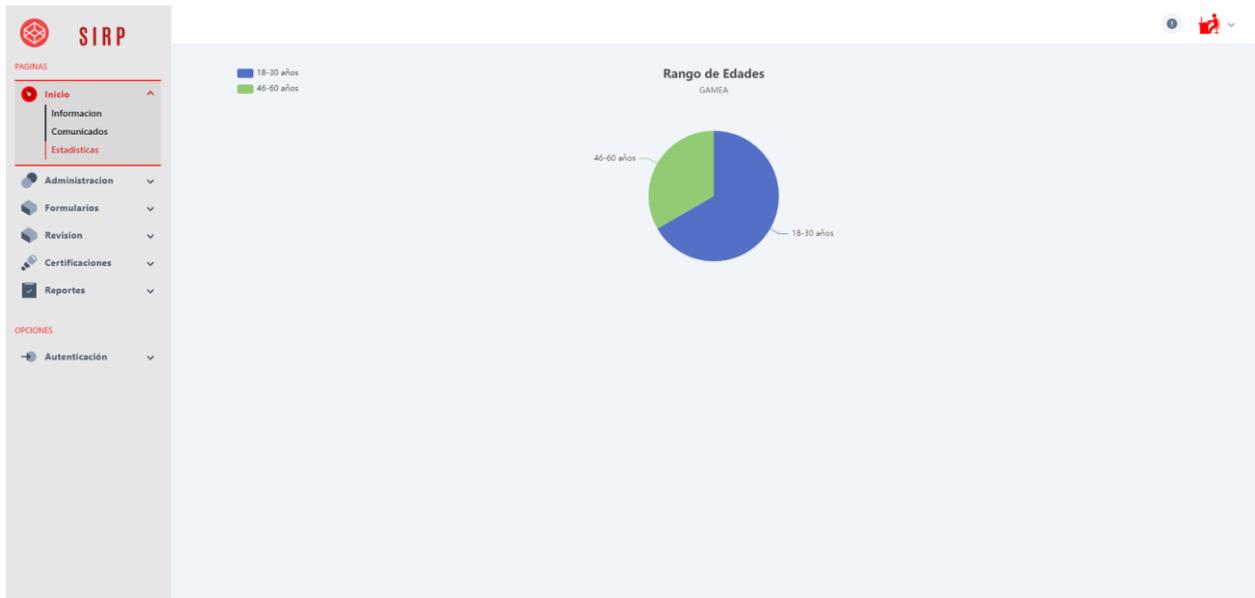


Imagen 8: Sección de estadísticas

6 ADMINISTRACIÓN.

6.1 ROLES.

Esta sección está dedicada a la gestión de los accesos a cada módulo del sistema mediante una tabla de datos en los cuales se tienen los roles del sistema definidos y se pueden gestionar el acceso a los módulos del sistema para cada rol, además teniendo la posibilidad de actualizar estos datos o eliminarlos.

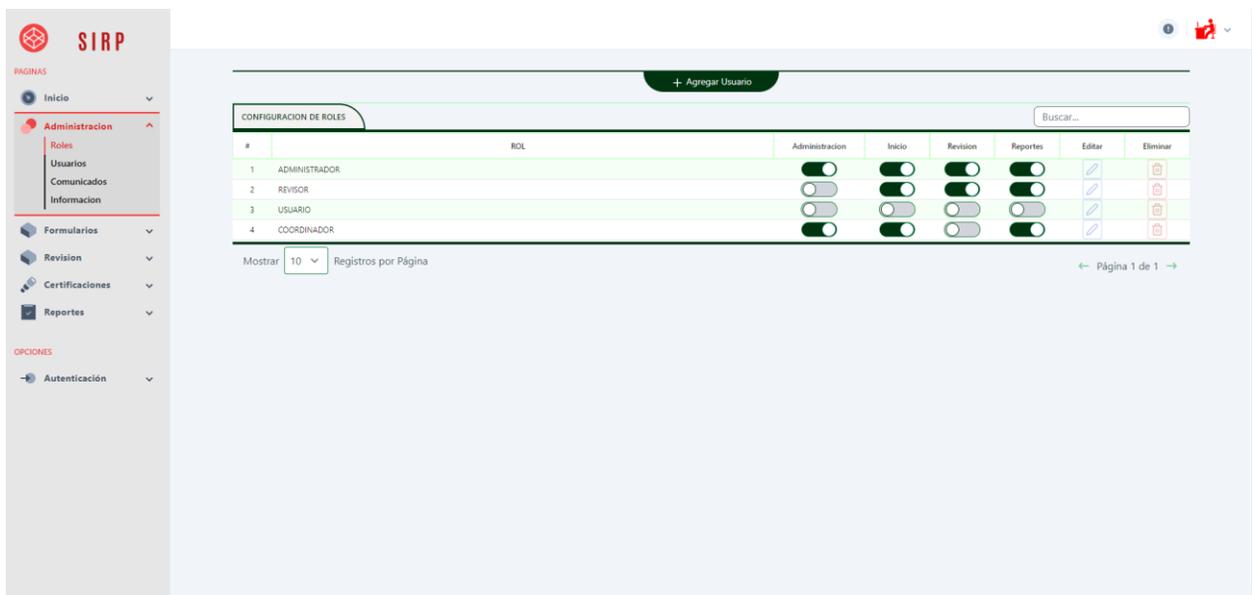


Imagen 9: Sección de roles del sistema

- **Crear Roles:** Esta ventana le permitirá añadir cualquier nuevo rol que se determine a futuro en el sistema.

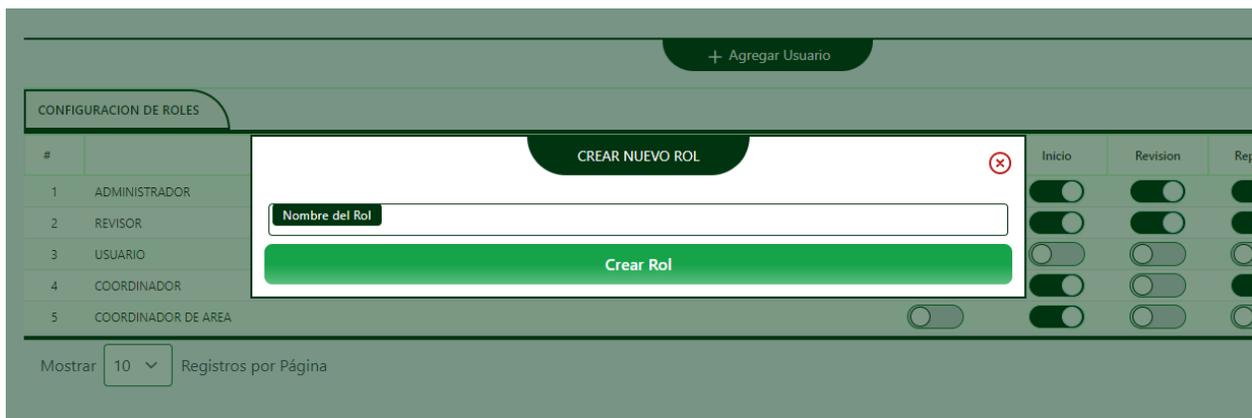


Imagen 10: Ventana para el registro de roles del sistema

- **Editar Roles:** También se cuenta con la opción de editar y actualizar los roles registrados previamente, estos cambios se reflejarán en todo el sistema, de igual forma que en el Registro, todos los campos deben ser rellenados y validados antes de ser registrados.



Imagen 11: Actualización de los roles del sistema

6.2 USUARIOS.

Esta sección está dedicada a la gestión de los usuarios del sistema mediante una tabla de datos en los cuales se tienen los roles del sistema definidos y se pueden gestionar el estado de cada uno.

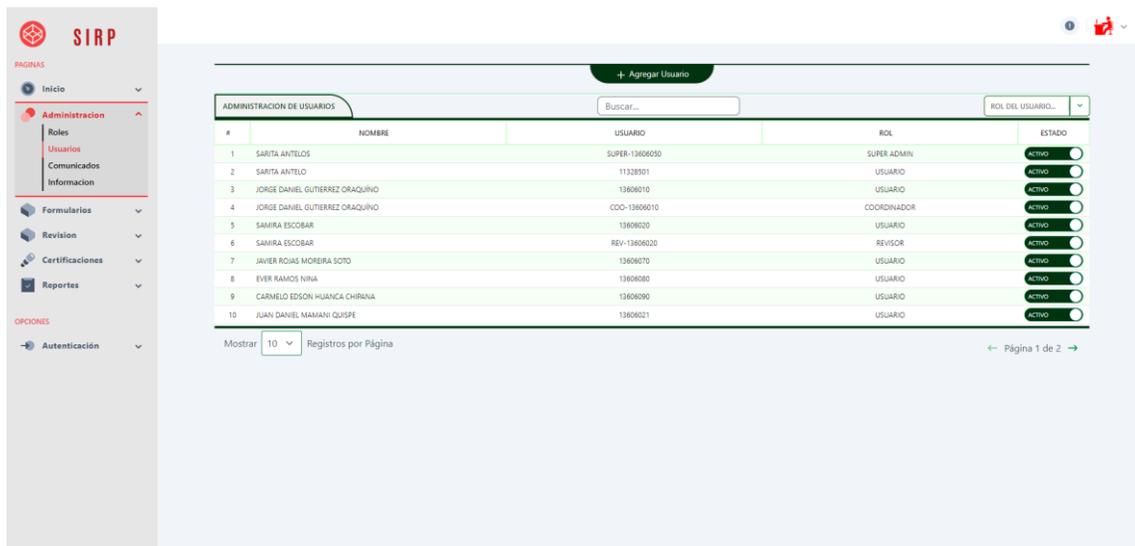


Imagen 12: Tabla de registros de los usuarios del sistema

6.3 COMUNICADOS.

Esta sección está dedicada a la gestión de los comunicados institucionales ya registrados en el sistema, con la posibilidad de registrar, actualizar y/o eliminar estos registros.

Administración de Comunicados

#	TÍTULO	DESCRIPCIÓN	PDF	FECHA	Editar	Eliminar
1	ENTREGA DE MEMORANDUMS	SE REALIZARA LA ENTREGA DE MEMOS Y DEMAS S...	Transaccionales.jpg	14 de junio de 2024		
2	TRABAJO MUNICIPALES	TRABAJO MUNICIPALES DENTRO DEL GAMEA	Trabajos Municipales.jpg	14 de junio de 2024		
3	COMUNICADO DE PROTESTA	TEMAS DE INFANTICIDIO Y VIOLENCIA	comunicado de protesta.jpg	14 de junio de 2024		
4	CULMINACION DE ESTRATEGIA ORGANIZACIONAL	ALCANZAR OBJETIVOS ESTRATEGICOS MEDIANTE L...	Comunicado Interno.jpg	14 de junio de 2024		

Mostrar 10 Registros por Página

← Página 1 de 1 →

Imagen 13: Tabla de registros de los comunicados institucionales del sistema

7 FORMULARIOS.

7.1 DATOS PERSONALES.

Esta sección está dedicada a la gestión de los datos personales de los postulantes registrados en el sistema, además teniendo la posibilidad de actualizar estos datos.

- **Registro de Datos Personales:** En esta ventana se tienen los campos necesarios para registrar a un nuevo postulante, siendo que se debe validar estos datos antes de ser almacenados.

I. DATOS PERSONALES

NOMBRE COMPLETO: SARITA ANTELOS | DOCUMENTO: CI | NUMERO DE DOCUMENTO: 11328501 | EXPEDIDO: LP | ESTADO CIVIL: SOLTERO/A | PAIS DE NACIMIENTO: BOLIVIA | NACIONALIDAD: BOLIVIANA | FECHA DE NACIMIENTO: 6 de diciembre de 2023

PROVINCIA DE NAC: MURILLO | DEPARTAMENTO DE NAC: LA PAZ

Documentos PDF			Imágenes		
#	PDF	Fecha	#	IMG	Fecha
1	1_Documentos_Personales_113406090.pdf	11/12/2023	1	imgPerfil.jpg	11/12/2023

Fecha de Registro: 11 de diciembre de 2023

[Editar Registro](#)

Imagen 14: Formulario de registro de datos personales

- **Actualización de Datos Personales:** En esta ventana se puede actualizar si correspondiera los datos registrados de los postulantes, siendo que se deben validar estos datos antes de ser almacenados.

I. DATOS PERSONALES

Nombres: SARITA ANTELOS | Primer Apellido: ANTELOS | Segundo Apellido: | Estado Civil: SOLTERO/A

Documento: CI | Numero de Documento: 11328501 | Complemento:

Apellido de Casado: | Expedido: | Expedido/Extrajero: | Pais de Nacimiento: BOLIVIA | Nacionalidad: BOLIVIANA

Fecha de Nacimiento: 06/12/2023 | Localidad: | Provincia: MURILLO | Departamento: LA PAZ

Foto de Identidad: [Seleccionar archivo](#) | Ningún archivo seleccionado

Documentos PDF			Imágenes		
#	PDF	Fecha	#	IMG	Fecha
1	1_Documentos_Personales_113406090.pdf	11/12/2023	1	imgPerfil.jpg	11/12/2023

El PDF que ingrese debe contener los siguientes archivos

- Carnet de identidad Digitalizado. (Obligatorio)
- Certificado de Nacimiento. (Obligatorio)
- Certificado de Matrimonio. (Si corresponde)

PDF Datos Personales: [Seleccionar archivo](#) | Ningún archivo seleccionado

[Actualizar Registro](#)
[Cancelar Edición](#)

Imagen 15: Formulario de actualización de datos personales

7.2 DATOS COMPLEMENTARIOS.

- **Registro de Datos Complementarios:** En esta ventana se tienen los campos necesarios para registrar información complementaria para cada postulante, siendo que se debe validar estos datos antes de ser almacenados.

The screenshot displays the 'II. DATOS COMPLEMENTARIOS' form in the SIRP system. The form is divided into several sections:

- Dirección de Domicilio:** Includes fields for ZONA-URBANIZACION (FRANZ TAMAYO), CALLE (ISMAEL SOTO MAYOR), NUMERO DE DOMICILIO (1474), DISTRITO (14), MUNICIPIO (EL ALTO), EMAIL PERSONAL (juandanielgoteparedor@gmail.com), and NUMERO DE CELULAR (74033698).
- Servicio Militar:** Includes fields for SEXO (MASCULINO), GRUPO SANGUINEO (ORH+), NUMERO DE LIBRETA (151878), NUMERO DE MATRICULA (A1587558), AÑO (2016), and TIPO DE SERVICIO (POR EXENCIÓN AUXILIAR A).
- Licencia de Conducir:** Includes fields for LICENCIA DE CONDUCIR (SI), NUMERO DE LICENCIA (11313145), and CATEGORIA (A).
- Familiar de Referencia:** Includes fields for NOMBRE DE FAMILIAR (JORGE DANIEL MQ), PARENTESCO (HE), and NUMERO DE CELULAR (74069369).

Below these sections is a table for 'Documentos PDF':

#	PDF	Fecha
1	2_Documentos_Complementarios_13406090.pdf	11/12/2023

The form also shows the 'Fecha de Registro' as 11 de diciembre de 2023 and an 'Editar Registros' button.

Imagen 16: Formulario de registro de datos complementarios

- **Actualización de Datos Complementarios:** En esta ventana se puede actualizar si correspondiera los datos registrados de los postulantes, siendo que se deben validar estos datos antes de ser almacenados.

II. DATOS COMPLEMENTARIOS

Datos de Referencia

Zona/Urbanización: FRANZ TAMAYO | Calle o Avenida: ISMAEL SOTO MAYOR | Numero de Domicilio: 1474 | Distrito: 14 | Municipio: EL ALTO

Edificio (Opcional): | Email Personal: juandanielquispeperedo@gmail.com | Email Trabajo (Opcional): | Numero Celular: 74033698

Servicio Militar (Si Corresponde)

Sexo: M F O | Grupo Sanguineo: ORH+ | Nro. Libreta Militar: 151878 | Matricula: A1587558 | Año: 2016 | Tipo de Servicio Militar: POR EXENCION AUXILIAR A

Licencia de Conducir (Opcional)

Licencia de Conducir: SI NO | Numero de Licencia: 1131345 | Categoria: CATEGORIA A

Familiar de Referencia

Nombre y Apellido: JORGE DANIEL MQ | Parentesco: HERMANO/A | Celular: 74069369

#	PDF	Fecha
1	2_Documentos_Complementarios_13406090.pdf	11/12/2023

El PDF que ingrese debe contener los siguientes archivos

- Croquis de Domicilio. (Obligatorio)
- Libreta de Servicio Militar. (Obligatorio en Varones)
- Licencia de Conducir. (Opcional)
- Cedula de Identidad digitalizada del Familiar. (Obligatorio)

PDF Datos Complementarios

Seleccionar archivo | Ningún archivo seleccionado

Actualizar Registro

Cancelar Edición

Imagen 17: Formulario de actualización de datos complementarios

7.3 DATOS ACADÉMICOS.

- **Registro de Datos Académicos:** En esta ventana se tienen los campos necesarios para registrar información académica para cada postulante, siendo que se debe validar estos datos antes de ser almacenados.

III. DATOS ACADÉMICOS

NIVEL DE ESTUDIO: EDUCACIÓN TÉCNICA Y TECNOLÓGICA | IDIOMA NATIVO: AYMARA | LEE: SI | ESCRIBE: SI | HABLA: NO

#	Curso	Institucion	Inicio	Conclusion
1	POLÍTICA DE ALIMENTACIÓN Y NUTRICIÓN ESCOLAR EN EL ESTADO PLURINACIONAL DE BOLIVIA	ESCUELA DE GESTION PÚBLICA PLURINACIONAL	06/12/2023	15/12/2023
2	CURSO DE LA LENGUA AIMARA Y FORTALECIMIENTO DE LA CULTURA	ESCUELA DE GESTION PÚBLICA PLURINACIONAL	11/01/2024	25/01/2024

#	PDF	Fecha
1	3_Documentos_Academicos_13406090.pdf	11/12/2023

Fecha de Registro: 11 de diciembre de 2023

Editar Registros

Imagen 18: Formulario de registro de datos académicos

- **Actualización de Datos Académicos:** En esta ventana se puede actualizar si correspondiera los datos registrados de los postulantes, siendo que se deben validar estos datos antes de ser almacenados.

MIS DATOS ACADÉMICOS

Nivel Académico Alcanzado
 EDUCACIÓN TÉCNICA Y TECNOLÓGICA | Certificado de Competencia | Otro

Conocimientos en Idioma Nativo (Obligatorio)
 Idioma: AYMARA | Conocimiento: Lee, Escribe, Habla

Conocimientos Obligatorios
 Nombre de la Capacitación: SELECCIONAR... | Institución: SELECCIONAR... | Fecha de Inicio: dd/mm/aaaa | Fecha de Conclusión: dd/mm/aaaa

Cursos Institucionales

#	Curso	Institucion	Inicio	Conclusión	Editar	Eliminar
1	POLÍTICA DE ALIMENTACIÓN Y NUTRICIÓN ESCOLAR EN EL ESTADO PLURINACIONAL DE BOLIVIA	ESCUELA DE GESTIÓN PÚBLICA PLURINACIONAL	06/12/2023	15/12/2023	[Editar]	[Eliminar]
2	CURSO DE LA LENGUA AIMARA Y FORTALECIMIENTO DE LA CULTURA	ESCUELA DE GESTIÓN PÚBLICA PLURINACIONAL	11/01/2024	25/01/2024	[Editar]	[Eliminar]

Documentos PDF

#	PDF	Fecha
1	1. Documento_Academico_13490205.pdf	11/12/2023

El PDF que Ingrese debe contener los siguientes Documentos.

- Título del Grado Académico Alcanzado. (Obligatorio)
- Certificado de Idioma Nativo. (Obligatorio)
- Ley Nro. 1178 - Ley de Administración y Control Gubernamentales. (Obligatorio)
- Responsabilidad por la Función Pública. (Obligatorio)

PDF Datos Académicos
 Seleccionar archivo | Ningún archivo seleccionado

Actualizar Registro
 Cancelar Edición

Imagen 19: Formulario de actualización de datos académicos

7.4 ESTUDIOS CURSADOS.

- **Registro de Datos Cursados:** En esta ventana se tienen los campos necesarios para registrar información sobre los estudios cursados para cada postulante, siendo que se debe validar estos datos antes de ser almacenados.

IV. ESTUDIOS REALIZADOS

Títulos Obtenidos

#	Título	Carrera	Institución
> 1	LICENCIATURA EN INFORMATICA Y TELECOMUNICACIONES	INGENIERIA DE SISTEMAS	UNIVERSIDAD PUBLICA DE EL ALTO

Idiomas Cursados

#	Idioma	Institución
> 1	INGLES TECNICO I	UPEA
> 2	UPEA	UPEA

Cursos Cursados

#	Curso	Institución
> 1	MANTENIMIENTO EN COMPUTADORAS	INTITULO BERLIN
> 2	SERVICIOS WEB CAM	UNIVERSIDAD PUBLICA DE EL ALTO

Documentos PDF

#	PDF	Fecha
1	4_Documento_Estudios_13406090.pdf	11/12/2023

Fecha de Registro: 11 de diciembre de 2023

[Editar Registros](#)

Imagen 20: Formulario de registro de estudios cursados

- **Actualización de Datos Cursados:** En esta ventana se puede actualizar si correspondiera los datos registrados de los postulantes, siendo que se deben validar estos datos antes de ser almacenados.

IV. ESTUDIOS CURSADOS / EN PROCESO (OPCIONAL)

Títulos Obtenidos

Título Obtenido: Carrera:

Institución / Universidad: Sigla de la Universidad: Lugar / Ciudad: País:

Diploma Académico: SI NO Revalidado: SI NO Institución que lo revalida: Número de Diploma:

Título Profesional: SI NO Revalidado: SI NO Institución que lo revalida: Número de Título:

[Agregar Título](#)

#	Título	Carrera	Institución	Editar	Eliminar
> 1	LICENCIATURA EN INFORMATICA Y TELECOMUNICACIONES	INGENIERIA DE SISTEMAS	UNIVERSIDAD PUBLICA DE EL ALTO	✎	🗑️

Conocimiento en Idiomas

Idioma: Institución: Sigla de la Institución: Capacidad: Lee Escribe Habla

Fecha de Inicio: Fecha de Conclusión:

[Agregar Idioma](#)

#	Idioma	Institución	Editar	Eliminar
> 1	INGLES TECNICO I	UPEA	✎	🗑️
> 2	UPEA	UPEA	✎	🗑️

Imagen 21: Formulario de actualización de datos cursados

IV. ESTUDIOS CURSADOS / EN PROCESO (OPCIONAL)

Títulos Obtenidos

Editar Título Profesional

Título Obtenido	LICENCIATURA EN INFORMÁTICA Y TELECOMUNICACIONES	Carrera	INGENIERIA DE SISTEMAS
Institución / Universidad	UNIVERSIDAD PUBLICA DE EL ALTO	Sigla de la Universidad	UPEA
Lugar / Ciudad	LA PAZ	País	BOLIVIA
Diploma Académico	SI <input checked="" type="radio"/> NO <input type="radio"/>	Revalidado	SI <input checked="" type="radio"/> NO <input type="radio"/>
Institución que lo revalida	UNIVERSIDAD PUBLICA DE EL ALTO	Numero de Diploma	45287
Título Profesional	SI <input checked="" type="radio"/> NO <input type="radio"/>	Revalidado	SI <input checked="" type="radio"/> NO <input type="radio"/>
Institución que lo revalida	UNIVERSIDAD PUBLICA DE EL ALTO	Numero de Título	64846

Cancelar Guardar

#	Idioma	Institucion	Editar
1	INGLES TECNICO I	UPEA	

Imagen 22: Formulario para la edición del título profesional

Conocimiento en Idiomas

Editar Idioma

Idioma	INGLES TECNICO I	Institucion	UPEA
Sigla de la Institucion	UPEA	Capacidad	<input checked="" type="checkbox"/> Lee <input type="checkbox"/> Escribe <input type="checkbox"/> Habla
Fecha de Inicio	12/04/2023	Fecha de Conclusion	22/04/2023

Cancelar Guardar

Carga Horaria (Horas)	Fecha de Inicio	Fecha de Conclusion
	dd/mm/aaaa	dd/mm/aaaa

Imagen 23: Formulario de actualización del idioma

#	Curso	Institucion
> 1		
> 2		

EDITAR CURSO

Nombre de la Capacitacion
MANTENIMIENTO EN COMPUTADORAS

Institucion INTITULO BERLIN **Sigla de la Institucion** UPEA

Carga Horaria (Horas) 120 **Fecha de Inicio** 12/04/2023 **Fecha de Conclusion** 30/04/2023

Cancelar
Guardar

El PDF que ingrese debe contener los siguientes Documentos.

- Titulos que registro previamente. **(Originales)**
- Certificado de los Idiomas Cursados. **(Originales)**
- Certificados de los Cursos Realizados. **(Originales)**

PDF Estudios Cursados

Seleccionar archivo Ningún archivo seleccionado

Imagen 24: Formulario de edición para los cursos

7.5 SEGURO BIENES Y RENTAS.

- **Registro de Seguros, Bienes y Rentas:** En esta ventana se tienen los campos necesarios para registrar datos sobre el seguro, bienes y rentas para cada postulante, siendo que se debe validar estos datos antes de ser almacenados.

SIRP

PAGINAS

- Inicio
- Administración
- Formularios**
 - Datos Personales
 - Datos Complementarios
 - Datos Academicos
 - Estudios Cursados
 - Seguro Bienes y Rentas**
 - Experiencia Laboral
 - Documentos
 - Institucionales
 - Familiares Dependientes
 - Datos de Discapacidad
- Revision
- Certificaciones
- Reportes

OPCIONES

- Autenticación

V. SEGURO, BIENES Y RENTAS

TIPO DE CONTRATO	NOMBRE DEL SEGURO	SEGURO SOCIAL	TITULAR DEL SEGURO	MATRICULA DE SEGURO	NUMERO DEL EMPLEADOR	TIPO DE SEGURO	ANTERIOR AFP	CIA/NUM	CAS
ITEM	CAJA NACIONAL DE SALUD	SEGURO NACIONAL	PADRE	99-115-MQJR	152878445	PUBLICO	GASTORA PUBLICA	1000265945	NO
ANTIGUEDAD CAS	CUENTA BANCARIA (BANCO UNION)								
4	10000965187								

Declaraciones Juradas

#	Numero de Declaracion	Antes	Durante	Despues
1	545498	06/05/2023	12/05/2023	28/05/2023

Documentos PDF

#	PDF	Fecha
1	5_Documentos_Seguro_y_Bienes_11409090.pdf	11/12/2023

Fecha de Registro: 11 de diciembre de 2023

Editar Registros

Imagen 25: Formulario de registro para los datos del seguro, bienes y rentas

- **Actualización de Datos del Seguro, Bienes y Rentas:** En esta ventana se puede actualizar si correspondiera los datos registrados de los postulantes, siendo que se deben validar estos datos antes de ser almacenados.

V. SEGURO, BIENES Y RENTAS

Seguro Social

Seguro Social: SEGURO NACIONAL Anterior AFP: GESTORA PUBLICA NIA / CUA: 1000265945

CAS: SI NO Años: 4 Numero de Cuenta en Banco Union: 10000065187

Seguro de Salud

Personal: ITEM Tipo de Seguro: PRIVADO PUBLICO Nombre del Seguro Medico: CAJA NACIONAL DE SALUD

Titular: PADRE Matricula de Asegurado: 99-115-MQJI Numero del Empleador: 152878445

Documentos PDF

#	PDF	Fecha
1	5. Documentos Seguro y Bienes.13406090.pdf	11/12/2023

El PDF que ingrese debe contener los siguientes Documentos.

- Títulos que registro previamente. (Originales)
- Certificado de los Idiomas Cursados. (Originales)
- Certificados de los Cursos Realizados. (Originales)

PDF de Respaldos: Seleccionar archivo | Ningún archivo seleccionado

Actualizar Registro

Cancelar Edición

Imagen 26: Formulario de actualización para los datos del seguro, bienes y rentas

7.6 EXPERIENCIA LABORAL.

- **Registro de Experiencia Laboral:** En esta ventana se tienen los campos necesarios para registrar la experiencia laboral para cada postulante, siendo que se debe validar estos datos antes de ser almacenados.

VI. EXPERIENCIA LABORAL

Experiencia General

#	Institucion	Cargo	Fecha	Conclusion
1	PIL ANDINA SRL	TECNICO I EN SISTEMAS	12/01/2023	12/11/2023

Experiencia Laboral Dentro del GAMEA

#	Institucion	Cargo	Inicio	Conclusion
1	UNIDAD DE SISTEMAS	TECNICO I - DESARROLLADOR DE SISTEMAS	25/11/2023	27/12/2023

Documentos PDF

#	PDF	Fecha
1	6_Documentos_Laboral_13406090.pdf	11/12/2023

Fecha de Registro: 11 de diciembre de 2023

Editar Registros

Imagen 27: Formulario de registro para la experiencia laboral

- **Actualización de la Experiencia Laboral:** En esta ventana se puede actualizar si correspondiera los datos registrados de los postulantes, siendo que se deben validar estos datos antes de ser almacenados.

VI. EXPERIENCIA LABORAL

Experiencia Laboral en el GAMEA (Opcional)

Instancia: Cargo Desempeñado: Fecha de Inicio: Fecha de Conclusion:

Agregar Experiencia

#	Institucion	Cargo	Inicio	Conclusion	Editar	Eliminar
1	UNIDAD DE SISTEMAS	TECNICO I - DESARROLLADOR DE SISTEMAS	25/11/2023	27/12/2023		

Documentos PDF

#	PDF	Fecha
1	6_Documentos_Laboral_13406090.pdf	11/12/2023

El PDF que ingrese debe contener los siguientes Documentos.

- Contratos realizados externamente. (Originales)
- Contratos realizados en la Institucion. (Originales)

PDF Experiencia Laboral

Seleccionar archivo | Ningún archivo seleccionado

Actualizar Registro

Cancelar Edición

Imagen 28: Formulario de actualización para la experiencia laboral

#	Institucion	Cargo	Fecha	Conclusion
1	UNIDAD DE SISTEMAS	TECNICO I - DESARROLLADOR DE SISTEMAS	25/11/2023	27/12/2023

EDITAR EXPERIENCIA EXTERNA ✕

Institucion PIL ANDINA SRL	Tipo de Institucion <input type="radio"/> PRIVADO <input checked="" type="radio"/> PUBLICO <input type="radio"/> OTRO
Otro (especificar)	Cargo Desempeñado TECNICO I EN SISTEMAS
Referente al Cargo que Postula <input checked="" type="radio"/> SI <input type="radio"/> NO	Pais BOLIVIA
Fecha de Inicio 12/01/2023 📅	Fecha de Conclusion 12/11/2023 📅

Cancelar
Guardar

Imagen 29: Formulario de actualización para la experiencia externa

#	Institucion	Cargo	Fecha	Conclusion
1	UNIDAD DE SISTEMAS	TECNICO I - DESARROLLADOR DE SISTEMAS	25/11/2023	27/12/2023

EDITAR EXPERIENCIA LABORAL ✕

Instancia UNIDAD DE SISTEMAS	Cargo Desempeñado TECNICO I - DESARROLLADOR DE SISTEMAS
Fecha de Inicio 25/11/2023 📅	Fecha de Conclusion 27/12/2023 📅

Cancelar
Guardar

Imagen 30: Formulario de actualización para la experiencia interna

7.7 DOCUMENTOS INSTITUCIONALES.

- **Registro de Documentos Institucionales:** En esta ventana se tienen los campos necesarios para registrar los documentos institucionales para cada postulante, siendo que se debe validar estos datos antes de ser almacenados.

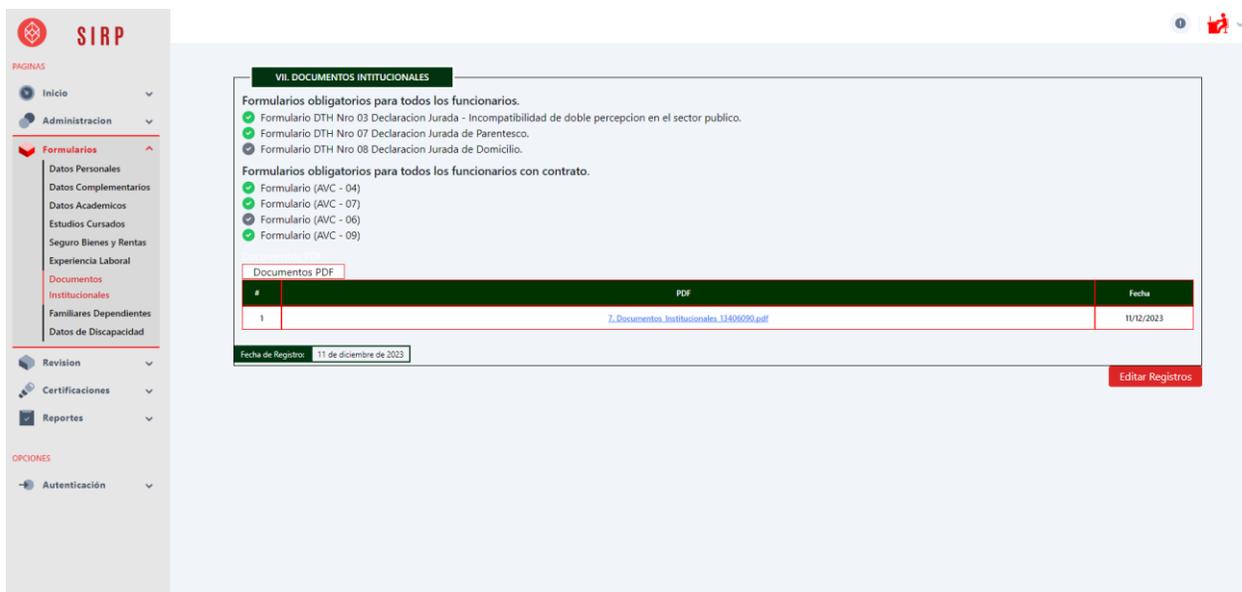


Imagen 31: Formulario de registro para los documentos institucionales

7.8 FAMILIARES DEPENDIENTES.

- **Registro de Datos de Familiares Dependientes:** En esta ventana se tienen los campos necesarios para registrar los datos de los familiares dependientes para cada postulante, siendo que se debe validar estos datos antes de ser almacenados.

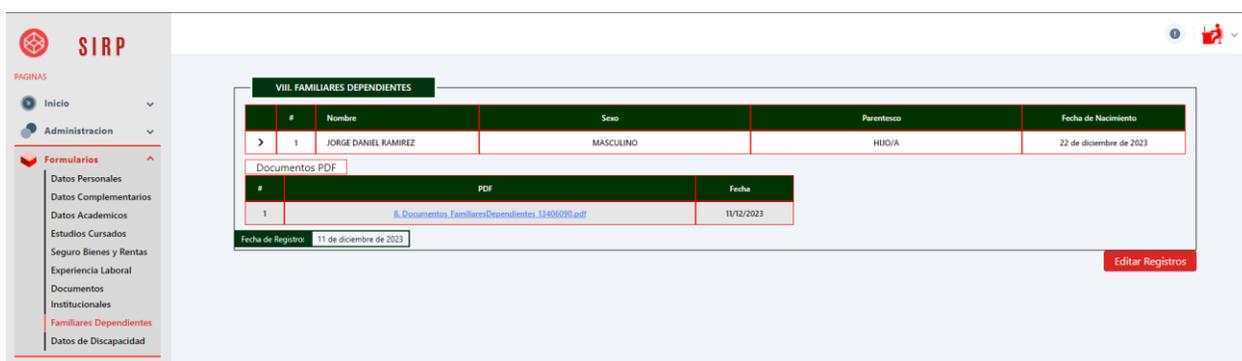


Imagen 32: Formulario de registro para los familiares dependientes

- **Actualización de Datos de Familiares Dependientes:** En esta ventana se puede actualizar si correspondiera los datos registrados de los postulantes, siendo que se deben validar estos datos antes de ser almacenados.

VIII. FAMILIARES DEPENDIENTES

Dependientes (Obligatorio)

Nombre y Apellido: [] Sexo: M F O Parentesco: HIJO/A

Fecha de Nacimiento: [dd/mm/aaaa] Tipo de Documento: CI Numero de Documento: [] Expedido: LA PAZ

Documento de Dependencia: SELECCIONAR... Otro (Especificar): CNAC Numero del Documento: [] **Agregar Dependiente**

#	Nombre	Sexo	Parentesco	Fecha de Nacimiento	Editar	Eliminar
1	JORGE DANIEL RAMIREZ	MASCULINO	HIJO/A	22 de diciembre de 2023		

Documentos PDF

#	PDF	Fecha
1	8_Documentos_familiaresDependientes_13406096.pdf	10/12/2023

El PDF que ingrese debe contener los siguientes Documentos.

- El certificado de relacion de Dependencia. (Originales)

PDF Certificados

Seleccionar archivo: Ningún archivo seleccionado

Actualizar Registro

Cancelar Edición

Imagen 33: Formulario de actualización para los familiares dependientes

7.9 DATOS DE DISCAPACIDAD.

- **Registro de Datos de Discapacidad:** En esta ventana se tienen los campos necesarios para registrar los datos sobre la discapacidad si es que la tuviera el postulante, siendo que se debe validar estos datos antes de ser almacenados.

IX. DATOS DE DISCAPACIDAD

Titular

TIPO DE CONTRATO DEL FUNCIONARIO/A: CONTRATO NUMERO DE CARNET DE DISCAPACIDAD: 5844584-I

Documentos PDF

#	PDF	Fecha
1	9_Documentos_Discapacidad_13406096.pdf	11/12/2023

Fecha de Registro: 11 de diciembre de 2023

Editar Registros

Imagen 34: Formulario de registro para los datos sobre discapacidad

- **Actualización de Datos de Discapacidad:** En esta ventana se puede actualizar si correspondiera los datos registrados de los postulantes, siendo que se deben validar estos datos antes de ser almacenados.

IX. DATOS DE DISCAPACIDAD

Discapacitado Titular

Tipo Contrato: CONTRATO | Numero de Carnet IBC: 5844584-1

Formulario de Discapacidad (Tutor)

Parentesco: -- | Numero de Carnet IBC: | Numero de Cedula de Identidad: | [Agregar Familiar](#)

Documentos PDF

#	PDF	Fecha
1	9_Documentos_Discapacidad_13406090.pdf	11/12/2023

El PDF que ingrese debe contener los siguientes Documentos.

- CASO TITULAR: Introducir en PDF la Carnet de Discapacidad del Discapacitado. **(Original)**
- CASO TUTOR: Introducir en PDF la Cedula de Identidad del Discapacitado. **(Original)**
- CASO TUTOR: Introducir en PDF la Carnet de Discapacidad del Discapacitado. **(Original)**

PDF de Discapacidad

Seleccionar archivo | Ningun archivo seleccionado

[Actualizar Registro](#)

[Cancelar Edicion](#)

Imagen 35: Formulario de actualización para los datos sobre discapacidad

Formulario de Discapacidad (Tutor)

Parentesco: ... | Numero de Carnet IBC: | Numero de Cedula de Identidad: | [Agregar Familiar](#)

EDITAR INFORMACION

Parentesco: MADRE | Numero de Carnet IBC: 158877 | Numero de Cedula de Identidad: 125155

[Cancelar](#) [Guardar](#)

Documentos PDF

#	PDF	Fecha
1	9_Documentos_Discapacidad_13406090.pdf	11/12/2023

El PDF que ingrese debe contener los siguientes Documentos.

Imagen 36: Formulario de actualización para los datos sobre el tutor del postulante

8 REVISIÓN.

#	Nombre del Formulario	Visualizar	Habilitar
1	Datos Personales.	<input checked="" type="checkbox"/>	Habilitar Edición
2	Datos Complementarios.	<input type="checkbox"/>	Habilitar Edición
3	Datos Académicos.	<input type="checkbox"/>	Habilitar Edición
4	Estudios Cursados.	<input type="checkbox"/>	Habilitar Edición
5	Datos de Seguro Bienes y Rentas.	<input type="checkbox"/>	Habilitar Edición
6	Experiencia Laboral.	<input type="checkbox"/>	Habilitar Edición
7	Documentos Solicitados.	<input type="checkbox"/>	Habilitar Edición
8	Familiares Dependientes.	<input type="checkbox"/>	Habilitar Edición
9	Datos de Discapacidad.	<input type="checkbox"/>	Habilitar Edición

Imagen 37: Vista principal de los postulantes registrados para su correspondiente revisión

#	Fecha de Revisión	Revisado
1	11/12/2023	CUMPLE
2	11/12/2023	NO CUMPLE

#	Nombre del Formulario	Visualizar	Habilitar
1	Datos Personales.	<input checked="" type="checkbox"/>	Habilitar Edición
2	Datos Complementarios.	<input type="checkbox"/>	Habilitar Edición
3	Datos Académicos.	<input type="checkbox"/>	Habilitar Edición
4	Estudios Cursados.	<input type="checkbox"/>	Habilitar Edición
5	Datos de Seguro Bienes y Rentas.	<input type="checkbox"/>	Habilitar Edición
6	Experiencia Laboral.	<input type="checkbox"/>	Habilitar Edición
7	Documentos Solicitados.	<input type="checkbox"/>	Habilitar Edición
8	Familiares Dependientes.	<input type="checkbox"/>	Habilitar Edición
9	Datos de Discapacidad.	<input type="checkbox"/>	Habilitar Edición

Imagen 38: Ventana con la información registrada de cada postulante para su revisión

9 CERTIFICACIONES.



Imagen 39: Vista principal sobre las certificaciones

GOBIERNO AUTÓNOMO MUNICIPAL DE EL ALTO				
FICHA KARDEX				
I. DATOS PERSONALES				
NOMBRE COMPLETO	PRIMER APELLIDO	SEGUNDO APELLIDO	ESTADO CIVIL	APELLIDO DE CASADIA
SARITA	ANTELOS		SOL	
DOCUMENTO	NUMERO DE DOCUMENTO	COMPLEMENTO	EXPEDIDO	EXPEDIDO EXTRANJERO
CI	11328501		LP	
PAIS DE NACIMIENTO		NACIONALIDAD	FECHA DE NACIMIENTO	
BOLIVIA		BOLIVIANA	8 de diciembre de 2023	
LOCALIDAD DE NACIMIENTO		PROVINCIA DE NACIMIENTO	DEPARTAMENTO DE NACIMIENTO	
MURILLO			LA PAZ	
Fecha de Creación: 11 de diciembre de 2023				
I. DATOS COMPLEMENTARIOS				
ZONA/URBANIZACION	CALLE/AVENIDA	NUMERO DE DOMICILIO	DISTRITO	MUNICIPIO
FRANZ TAMAYO	ISMAEL SOTO MAYOR	1474	14	EL ALTO
EDIFICIO	EMAIL PERSONAL	EMAIL TRABAJO	NUMERO DE CELULAR	
-	juanramirez.suarez@gmail.com	-	74033098	
SEXO	GRUPO SANGUINEO	NUMERO DE LIBRETA MILITAR	AÑO DE SERVICIO	TIPO DE SERVICIO
MASCULINO	ORH+	151878	2016	POR EXENCIÓN AUXILIAR A
LICENCIA DE CONDUCIR		NUMERO DE LICENCIA	CATEGORIA	
SI		11313145	A	
NOMBRE DE FAMILIAR DE REFERENCIA		PARENTESCO	CELULAR	
JORGE DANIEL MQ		HERMANO/A	74062369	
Fecha de Creación: 11 de diciembre de 2023				
I. DATOS ACADÉMICOS				
NIVEL ACADÉMICO		CERTIFICADO DE COMPETENCIA		
EDUCACIÓN TÉCNICA Y TECNOLÓGICA		OTRO (Desarrollar)		
IDIOMA		LEE	ESCRIBE	HABLA
AYMARÁ		SI	SI	NO
CURSADOS INSTITUCIONALES				
#	CURSO	INSTITUCION	FECHA DE INICIO	FECHA DE CONCLUSION
1.	ALIMENTACION_NUTRICION	EGPP	06-12-2023	15-12-2023
2.	LENGUA_AYMARA	EGPP	11-01-2024	25-01-2024
Fecha de Creación: 11 de diciembre de 2023				

Imagen 40: Documento PDF de ejemplo generado por el sistema



Imagen 41: Vista principal donde se visualiza los requisitos necesarios para las declaraciones.

10 REPORTES.

The screenshot shows the 'Documentos Presentados' report table. It has a search bar with '11328501' and a 'Buscar' button. The table lists 9 documents with their respective PDF links and submission dates.

#	PDF	Fecha
1	1.Documentos_Personales_13406090.pdf	11 de diciembre de 2023
2	2.Documentos_Complementarios_13406090.pdf	11 de diciembre de 2023
3	3.Documentos_Academicos_13406090.pdf	14 de junio de 2024
4	4.Documento_Estudios_13406090.pdf	14 de junio de 2024
5	5.Documentos_Seguros_y_Bienes_13406090.pdf	14 de junio de 2024
6	6.Documentos_Laboral_13406090.pdf	14 de junio de 2024
7	7.Documentos_Institucionales_13406090.pdf	14 de junio de 2024
8	8.Documentos_FamiliaresDependientes_13406090.pdf	14 de junio de 2024
9	9.Documentos_Discapacidad_13406090.pdf	14 de junio de 2024

Imagen 42: Tabla principal para los documentos subidos al sistema

The screenshot shows the 'Revisiones Realizadas' report table. It has a search bar with '11328501' and a 'Buscar' button. The table lists 4 reviews with their status, observations, reviewer, and date.

#	FORMULARIO	ESTADO	OBSERVACIONES	REVISOR	Fecha
1	Datos Personales	CUMPLE	SIN OBSERVACIONES	SARITA A	11 de diciembre de 2023
2	Datos Personales	NO CUMPLE	ESTE FORMULARIO NO CUMPLE CON LOS DATOS INGRESADOS	SARITA A	11 de diciembre de 2023
3	Datos Complementarios	NO CUMPLE	ESTE FORMULARIO NO CUMPLE CON LOS DATOS INGRESADOS	SARITA A	14 de junio de 2024
4	Estudios Curados	NO CUMPLE	ESTE FORMULARIO NO CUMPLE CON LOS DATOS INGRESADOS	SARITA A	14 de junio de 2024

Imagen 42: Tabla principal para la visualización del estado de los postulantes revisados.