

UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

“SISTEMA WEB PARA CONTROL Y SEGUIMIENTO DE CORRESPONDENCIA”

CASO: GOBIERNO AUTÓNOMO MUNICIPAL DE AYATA

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

Postulante: Sabino Flores Lipa

Tutor Metodológico: Lic. Ing. Helen Fanny Suntura Escobar

Tutor Especialista: Lic. Fredy Alanoca Coareti

Tutor Revisor: M. Sc. Lic. Zara Yujra Cama

EL ALTO – BOLIVIA

2024

DECLARACIÓN JURADA DE AUTENTICIDAD Y RESPONSABILIDAD

Yo, Sabino Flores Lipa, estudiante con C.I. 6902847 LP mediante la presente declaro de manera pública que la propuesta del TRABAJO DE GRADO titulada "SISTEMA WEB PARA EL CONTROL Y SEGUIMIENTO DE CORRESPONDENCIA. Caso: Gobierno Autónomo Municipal de Ayata." 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, noviembre de 2024

Univ. Sabino Flores Lipa.
C.I. 6902847 L.P.
flores.onibas@gmail.com

DEDICATORIA

Este proyecto de grado está dedicado:

Con sincera gratitud, dedico este proyecto de grado, en primer lugar, a Dios, quien con su guía y bendiciones me ha permitido alcanzar este significativo logro en mi formación profesional.

A mis amados padres, cuyo amor incondicional, apoyo constante y firme confianza en mí han sido la base sólida que me ha impulsado a avanzar y superar cada reto en este camino.

Ha todas estas personas que nos ha apoyado y han hecho posible que pueda cumplir mis objetivos.

AGRADECIMIENTOS

Expreso mi profundo agradecimiento a Dios, quien me ha otorgado el valioso don de la perseverancia, permitiéndome alcanzar mis metas.

Agradezco de manera sincera a mi familia, quienes han brindado un apoyo incondicional a lo largo de mi camino. Reconozco y agradezco a la Universidad por haberme dado la oportunidad de crecer y desarrollarme como individuo.

Mis palabras de gratitud se dirigen también a los dedicados docentes que, a lo largo de los años, han sido una inspiración y un modelo a seguir para mí

RESUMEN

Dentro del Gobierno Autónomo Municipal de Ayata, los procedimientos para gestionar, controlar y realizar el seguimiento de la correspondencia han mejorado gracias al Sistema Web para el Control y Seguimiento de Correspondencia. El municipio recibe entre 20 y 30 solicitudes diarias, lo que generaba desorganización en la información y una clasificación inadecuada de los documentos. Esta situación causaba insatisfacción entre los comunarios y las instituciones que remitían correspondencia, debido a las demoras en las respuestas y la falta de seguimiento adecuado.

Tras el desarrollo del nuevo sistema, los procesos de registro, seguimiento, distribución y control de correspondencia han mejorado, eliminando la desorganización y aumentando la eficiencia en la atención al público. El personal ahora dispone de herramientas adecuadas que permiten un seguimiento preciso de cada solicitud, lo que ha reducido los casos de documentos extraviados y ha agilizado la atención a las solicitudes.

Este sistema ha permitido al Gobierno Autónomo Municipal de Ayata mejorar su capacidad para gestionar las solicitudes de manera oportuna y eficiente, contribuyendo al crecimiento de la institución. La automatización de procesos también ha facilitado la generación de consultas y reportes estadísticos, mejorando la transparencia y el proceso de toma de decisiones.

El proyecto fue desarrollado utilizando la metodología OOHDM y cumplió con todos los objetivos específicos, como mejorar la gestión administrativa y fomentar la innovación tecnológica dentro de la institución. Se emplearon herramientas como MariaDB, PHP y JavaScript, garantizando una solución eficiente que ha resuelto los problemas identificados en la gestión de la correspondencia.

Palabras Claves: Correspondencia, Seguimiento, Control, Automatización, Innovación

ABSTRACT

Within the Autonomous Municipal Government of Ayata, the procedures for managing, controlling, and tracking correspondence have improved thanks to the Web System for Control and Tracking of Correspondence. The municipality receives between 20 and 30 daily requests, which generated disorganization in the information and improper classification of documents. This situation caused dissatisfaction among the community members and the institutions sending correspondence due to delays in responses and a lack of proper follow-up.

After the development of the new system, the processes of registration, tracking, distribution, and control of correspondence have improved, eliminating disorganization and increasing efficiency in public service. The staff now has the appropriate tools that allow for precise tracking of each request, which has reduced the number of lost documents and expedited the processing of requests.

This system has enabled the Autonomous Municipal Government of Ayata to improve its ability to manage requests in a timely and efficient manner, contributing to the growth of the institution. The automation of processes has also facilitated the generation of queries and statistical reports, improving transparency and decision-making processes.

The project was developed using the OOHDM methodology and met all specific objectives, such as improving administrative management and fostering technological innovation within the institution. Tools such as MariaDB, PHP, and JavaScript were used, ensuring an efficient solution that has addressed the issues identified in the management of correspondence.

Keywords: Correspondence, Tracking, Control, Automation, Innovation

CONTENÍDO

	Página
1. CAPÍTULO I MARCO PRELIMINAR.....	2
1.1. INTRODUCCIÓN.....	2
1.2. ANTECEDENTES.....	3
1.2.1. Antecedentes Institucionales.....	3
1.2.2. Antecedentes Afines al Proyecto de Grado.....	5
1.3. PLANTEAMIENTO DEL PROBLEMA.....	7
1.3.1. Problema Principal.....	8
1.3.2. Problemas Secundarios.....	8
1.3.3. Formulación del Problema.....	9
1.4. OBJETIVOS.....	9
1.4.1. Objetivo General.....	9
1.4.2. Objetivos Específicos.....	9
1.5. JUSTIFICACIÓN.....	10
1.5.1. Justificación Técnica.....	10
1.5.2. Justificación Económica.....	10
1.5.3. Justificación Social.....	10
1.6. METODOLOGÍA DE DESARROLLO.....	11
1.6.1. Metodología OOHDM (Método de Diseño Hipermedia Objeto Orientado).....	11
1.7. MÉTRICAS DE CALIDAD.....	12
1.7.1. Control de Calidad ISO 25000.....	12
1.8. ESTIMACIÓN DE COSTOS COCOMO.....	12
1.9. SEGURIDAD ISO/IEC 27001.....	13
1.10. PRUEBAS DE SOFTWARE.....	13
1.10.1. Pruebas de Caja Blanca.....	13

1.10.2. Pruebas de Caja Negra	14
1.10.3. Pruebas de Estrés	14
1.10.4. Pruebas de Accesibilidad	14
1.11. HERRAMIENTAS	15
1.11.1. Lenguajes de Programación.....	15
1.11.2. Plantillas y Librerías.....	15
1.11.3. Base de Datos	16
1.11.4. Servidor Local.....	16
1.11.5. Entorno de Desarrollo.....	16
1.11.6. Modelado de Diagramas	16
1.12. ALCANCES Y LIMITES	17
1.12.1. Alcances.....	17
1.12.2. Limites.....	17
1.13. APORTES.....	18
2. CAPÍTULO II MARCO TEÓRICO	20
2.1. INTRODUCCIÓN	20
2.2. CONCEPTOS BÁSICOS.....	20
2.2.1. Sistema	20
2.2.2. Web	21
2.2.3. Sistema web	21
2.2.4. Correspondencia	22
2.2.5. Seguimiento	23
2.2.6. Control.....	24
2.2.7. Control y Seguimiento	25
2.2.8. Hojas de Ruta	25
2.3. METODOLOGÍA DE DESARROLLO	26
2.3.1. Metodología OOHDM.....	26

2.3.2. Fases de la Metodología OOHDM	26
2.4. ARQUITECTURA DE SOFTWARE (MVC)	30
2.4.1. Modelo	31
2.4.2. Vista	31
2.4.3. Controlador	32
2.5. HERRAMIENTAS	34
2.5.1. Herramientas de Desarrollo Front-End	34
2.5.2. Herramientas de Desarrollo Back End	37
2.5.3. Funciones del lenguaje PHP	37
2.5.4. Base de Datos	37
2.5.5. Servidor Local	38
2.5.6. Modelado de Diagramas	38
2.6. COSTOS	39
2.6.1. Cocomo	39
2.7. MÉTRICAS DE CALIDAD	45
2.7.1. Norma ISO/IEC 25010	46
2.8. SEGURIDAD	54
2.9. PRUEBAS DE SOFTWARE	55
2.9.1. Pruebas de Caja Blanca	56
2.9.2. Pruebas de Caja Negra	56
2.9.3. Pruebas de Estrés	58
2.9.4. Pruebas de Accesibilidad	58
3. CAPÍTULO III MARCO APLICATIVO	60
3.1. INTRODUCCIÓN	60
3.2. APLICACIÓN DE LA METODOLOGÍA OOHDM	60
3.2.1. FASE I: Obtención de Requerimientos	61
3.2.2. FASE II: Diseño Conceptual	80

3.2.3. FASE III: Diseño navegacional.....	88
3.2.4. FASE IV: Diseño de Interfaz Abstracta.....	93
3.2.5. FASE V: Implementación.....	101
4. CAPÍTULO IV CALIDAD, SEGURIDAD Y COSTOS.....	111
4.1. INTRODUCCIÓN	111
4.2. MÉTRICAS DE CALIDAD - ESTÁNDAR ISO/IEC 25000.....	111
4.2.1. Funcionalidad.....	112
4.2.2. Confiabilidad	116
4.2.3. Usabilidad	118
4.2.4. Eficiencia.....	120
4.2.5. Mantenibilidad.....	122
4.2.6. Portabilidad.....	123
4.3. ESTIMACIÓN DE COSTO COCOMO.....	124
4.3.1. Modelo Semi Acoplado.....	125
4.3.2. Costo del Software	125
4.3.3. Ecuaciones de COCOMO Intermedio	126
4.3.4. Esfuerzo Requerido del Desarrollo	128
4.3.5. Tiempo de Desarrollo Requerido del Sistema	128
4.3.6. Número de Desarrolladores	129
4.3.7. Costo Total	130
4.4. PRUEBAS DE SOFTWARE.....	131
4.4.1. Prueba de Caja Blanca.....	131
4.4.2. Prueba de Caja Negra	134
4.4.3. Pruebas de Estrés	139
4.4.4. Pruebas de Accesibilidad	141
4.5. SEGURIDAD DEL SISTEMA – ISO 270001.....	142
4.5.1. Encriptación	142

4.5.2. Autenticación Segura.....	143
4.5.3. Gestor de Base de Datos	143
5. CAPÍTULO V CONCLUSIONES Y RECOMENDACIONES	145
5.1. CONCLUSIONES	145
5.2. RECOMENDACIONES.....	146
BIBLIOGRAFÍA.....	147
ANEXOS.....	151

ÍNDICE DE FIGURAS

	Página
Figura 1. Organigrama GAMA.....	5
Figura 2. Teoría de un Sistema.....	21
Figura 3. Fase de la Metodología OOHDM	27
Figura 4. Casos de Usos	28
Figura 5. Modelo Vista Controlador.....	33
Figura 6. ISO/IEC 25010	46
Figura 7. Pruebas de Caja Blanca	56
Figura 8. Prueba de Caja Negra	57
Figura 9. Descripción de Actores	64
Figura 10. Relación de Usuarios	65
Figura 11. Caso de uso General del sistema	66
Figura 12. Casos de Uso secretaria.....	67
Figura 13. Caso de Uso Administrador	72
Figura 14. Diagrama de Caso de Uso - Responsable de la Unidad	77
Figura 15. Caso de Uso - Seguimiento	79
Figura 16. Modelo Conceptual	80
Figura 17. Diseño de Base de Datos	81
Figura 18. Diagrama Navegacional administrador	89
Figura 19. Diagrama Navegacional – secretario(o)	90
Figura 20. <i>Diagrama Navegacional - jefe de la Unidad</i>	91
Figura 21. Diagrama Navegacional Remitente	92
Figura 22. Diseño de Pantalla Principal	93
Figura 23. Diseño de Pantalla de Inicio	94

Figura 24. Inicio de Sesión	94
Figura 25. Inicio de Sesión Administrador.....	95
Figura 26. Menú Principal Administrador	95
Figura 27. Diseño de Inicio de Sesión – Rol secretaria	96
Figura 28. Diseño Panel de Navegación de secretaria	96
Figura 29. Diseño de Inicio de Sesión – Rol Jefes de Unidad	97
Figura 30. Diseño Interfaz de Menú Principal de Jefe de Unidad	97
Figura 31. Diseño Interfaz de Búsqueda.....	98
Figura 32. Diseño Interfaz Seguimiento Correspondencia	98
Figura 33. Control de Correspondencia	99
Figura 34. Diseño Interfaz Modal para Opción Salir	99
Figura 35. Diseño de Interfaz Menu Navegacional.....	100
Figura 36. Panel de Administración del Servidor	103
Figura 37. Captura de Pantalla Principal	104
Figura 38. Captura de Pantalla Mensaje de Bienvenida	104
Figura 39. Interfaz de Búsqueda de Correspondencia	105
Figura 40. interfaz de Mensaje de Advertencia - Hoja de Ruta no Encontrada	105
Figura 41. Control de Correspondencia	105
Figura 42. Interfaz de Seguimiento de Correspondencia	106
Figura 43. Interfaz de Acceso para Usuarios Registrados	106
Figura 44. Interfaz de Panel de Usuario de Administrar	107
Figura 45. interfaz de Panel de secretaria	108
Figura 46. Interfaz Formulario de registro de nueva correspondencia	108
Figura 47. Hoja de Ruta Generada	108
Figura 48. Interfaz Listado de Registros	109
Figura 49. Interfaz panel de jefe de la Unidad	109

Figura 50. Interfaz - Opción de Derivar a Otras Áreas	109
Figura 51. Diagrama del Prueba de caja blanca.....	132
Figura 52. Prueba de Caja Negra Iniciar Sesión	134
Figura 53. Alertas a Iniciar.....	135
Figura 54. Caja Negra Ingreso al Sistema	135
Figura 55. Prueba de Caja Negra – Agregar Personal.....	136
Figura 56. Prueba de Caja Negra – Registro de Correspondencia.....	136
Figura 57. Prueba de Caja Negra – Registro Exitoso.....	137
Figura 58. Generación de Hoja de Ruta	137
Figura 59. Prueba de Caja Negra – Mensaje de Advertencia	137
Figura 60. Caja Negra Seguimiento.....	138
Figura 61. Caja Negra Control Correspondencia	138
Figura 62. Configuración para Prueba de Estres	139
Figura 63. Evaluando Pruebas de Estrés.	140
Figura 64. Prueba de Accesibilidad de WAVE	141
Figura 65. Seguridad del Sistema	142
Figura 66. Autenticación Segura	143

ÍNDICE DE TABLAS

	Página
Tabla 1. Tabla de Valores de Modelo Básico.....	¡Error! Marcador no definido.
Tabla 2. Factor de Ponderación.....	48
Tabla 3. Índices de Usabilidad.....	48
Tabla 4. Fases de la Metodología OOHDm.....	60
Tabla 5. Descripción del Proceso de Requerimiento	61
Tabla 6. Requerimientos no Funcionales	63
Tabla 7. Detalles de Casos Uso - Iniciar Sesión	67
Tabla 8. Destalles de Caso de Uso - Consultar Estado de Correspondencia	68
Tabla 9. Detalles de Caso de Uso - Gestionar Tipo de Documento	68
Tabla 10. Detalles de Caso de Uso - Generar Reportes.....	69
Tabla 11. Detalles de Caso Uso - Correspondencia	69
Tabla 12. Detalles Caso Uso - Iniciar Sesión	72
Tabla 13. Detalles de Caso de Uso - Gestionar Cargos	73
Tabla 14. <i>Detalles de Caso de Uso - Gestionar Personal</i>	73
Tabla 15. Detalles de Caso de Uso - Gestionar Unidad	74
Tabla 16. Detalles Caso de Uso - Gestionar Usuarios.....	74
Tabla 17. Caso de Uso – Gestionar Tipo de Documento.....	75
Tabla 18. Detalles Caso de Uso – Verificar Estado de Correspondencia.....	75
Tabla 19. Detalles de Caso de Uso - Generar Reportes.....	76
Tabla 20. Detalles de Caso de Uso - Buscar Correspondencia.....	76
Tabla 21. Detalles Caso de Uso - Iniciar Sesión	77
Tabla 22. Detalles Caso de Uso - Gestionar Correspondencia	78
Tabla 23. Detalles Caso de Uso - Buscar Remitente.....	79

Tabla 24. Descripción de Tabla Cargo.....	82
Tabla 25. Descripción Tabla Documento	82
Tabla 26. Descripción Tabla Historial.....	83
Tabla 27. Descripción Tabla Personal	84
Tabla 28. Descripción Tabla Tipo Documento	86
Tabla 29. Descripción Tabla Unidad	86
Tabla 30. Descripción Tabla Usuario	86
Tabla 31. Tabla de Iconos	88
Tabla 32. Factores de Ponderación.....	113
Tabla 33. Índice de Usabilidad.....	113
Tabla 34. Indicadores de Facilidades de Uso	118
Tabla 35. Usabilidad del Sistema	119
Tabla 36. Escala de Valores de Eficiencia.....	120
Tabla 37. Evaluación de Eficiencia.....	120
Tabla 38. Indicadores de Mantenibilidad	122
Tabla 39. Resultados Obtenidos - Norma ISO 25000.....	124
Tabla 40. Estimación de Costos	¡Error! Marcador no definido.
Tabla 41. Complejidad y Eficiencia del Lenguaje.....	125
Tabla 42. Formulas Para el Costo	126
Tabla 43. Tabla de Atributos Factores del Esfuerzo	127
Tabla 44. Valores de Caja Negra Login.....	134
Tabla 45. Prueba de Caja Negra Ingreso el Sistema.....	135

ÍNDICE DE ECUACIONES

	Página
(4) Ecuación Personas Por Mes Según Modelo De Cocomo Intermedio	43
(5) Ecuación Tiempo de Proyecto Según Modelo Intermedio	44
(6) Ecuación Costo Personal	44
(7) Ecuación de Costo Total del Proyecto.....	44
(8) Fórmula de Ajuste	49
(9) Punto Función Máxima	50
(10) Formula de la Funcionalidad	50
(11) Probabilidad de Fallas.....	51
(12) Probabilidad de Trabajo Sin Fallas.....	51
(13) Función de la Confiabilidad del Sistema MT	51
(14) Función para Determinar la Usabilidad MT	52
(15) Fórmula de la Eficiencia MT	53
(16) Cálculo del Nivel de Madurez del Sistema.....	53
(17) Fórmula de la Portabilidad MT.....	54
(18) Ecuación de Ajuste	114
(19) Función de la Confiabilidad del Sistema	116
(20) Función para Determinar la Usabilidad	118
(21) Fórmula Eficiencia	121
(22) Ecuación para el Índice de Madurez del Sistema	122
(23) Fórmula Portabilidad.....	123
(24) Ecuación Esfuerzo	128
(25) Ecuación Tiempo Requerido.....	129
(26) Ecuación Número de Desarrolladores	129

(27) Ecuación Costo del Software.....	130
(28) Ecuación Complejidad Ciclomática	133

LISTADO DE ABREVIATURAS

FLDC: Factor de conversión de líneas de código entregado.

TDES: Tiempo empleado en el desarrollo.

PF: Puntos de función.

PHP: Preprocesador de Hipertexto.

OOHDM: Método de Diseño Hipermedia Objeto Orientado).

CSS: Cascading Style Sheets

HTML: Lenguaje de Marcado de Hipertexto.

HTTP: Protocolo de Transferencia de Hipertexto.

LDC: Líneas de Código.

IDE: Entorno de Desarrollo Integrado

MVC: Modelo Vista Controlador

ISO: Organización Internacional de Normalización

IEC: Comisión Electrotécnica Internacional

JS: JavaScript

FAE: Factor de Ajuste de Esfuerzo

MS: Mili segundos

CAPÍTULO I

MARCO

PRELIMINAR

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



1. CAPÍTULO I MARCO PRELIMINAR

1.1. INTRODUCCIÓN

El Gobierno Autónomo Municipal de Ayata es una institución que representa a la segunda sección de la provincia Muñecas. Está encargado de gestionar recursos propios para la ejecución de proyectos, programas y atención a los habitantes de ese municipio, así como a la población en general. Para ser atendido, el remitente o instituciones deben dejar cartas, solicitudes e informes, lo que se denomina correspondencia.

Todo el procedimiento sigue una serie de pasos para aprobar o archivar la correspondencia, y actualmente se lleva a cabo de manera manual. Primeramente, se recibe la correspondencia, se verifica, se llena un formulario (estos son proporcionados por un proveedor externo, lo que representa gastos innecesarios para la institución) con datos del remitente y de la correspondencia, se registra en un libro de acta y se distribuye al área correspondiente. Todo este proceso es realizado por la secretaría.

En la actualidad, el Gobierno Autónomo Municipal de Ayata cuenta con acceso a internet y equipos de computación suficientes. En este contexto, surge la necesidad de desarrollar un sistema web que facilite el registro, control y seguimiento de la correspondencia. Este sistema no solo optimizará la gestión de documentos dentro de la institución, sino que también para una mayor eficiencia y transparencia en la comunicación y el seguimiento de la correspondencia. En un mundo cada vez más digitalizado, este sistema se convierte en una herramienta esencial para el Gobierno Autónomo Municipal de Ayata en su búsqueda de mejorar el servicio a la comunidad y adaptarse a los avances tecnológicos.

El desarrollo de este sistema brindará numerosos beneficios tanto para el Gobierno Autónomo Municipal de Ayata como para los usuarios y ciudadanos en general. Entre los beneficios esperados se encuentran la reducción de tiempos y costos asociados a la gestión

de la correspondencia, el aumento de la eficiencia en la atención de solicitudes, la mejora en la calidad de los servicios ofrecidos y la transparencia en los procesos administrativos.

1.2. ANTECEDENTES

1.2.1. *Antecedentes Institucionales*

El Gobierno Autónomo Municipal de Ayata es una institución que desempeña un papel fundamental en la administración y gestión de los asuntos locales. A lo largo de su historia, ha enfrentado diversos desafíos en relación a la eficiencia y control de la correspondencia que recibe. La falta de un sistema informático adecuado ha generado problemas como extravío de documentos, demoras en la entrega y dificultades para realizar un seguimiento efectivo.

1.2.1.1. Visión

La visión del Municipio se traduce en el desarrollo e industrialización agropecuaria, con integración vial tele comunicativa, respetando la biodiversidad, normas y procedimientos propios, promoviendo además la calidad de educación, salud y turismo, equitativo en todos sus grupos generacionales.

1.2.1.2. Misión

El Gobierno Autónomo Municipal de Ayata es una unidad territorial político y administrative, organizada y autónoma, que contribuye al desarrollo endógeno y a la satisfacción de las necesidades colectivas, garantizando la participación de todos/as habitantes en el proceso de planificación para el desarrollo productivo agroindustrial, tecnológico, integral y sostenible, que promueva el vivir bien.

1.2.1.3. Objetivos institucionales

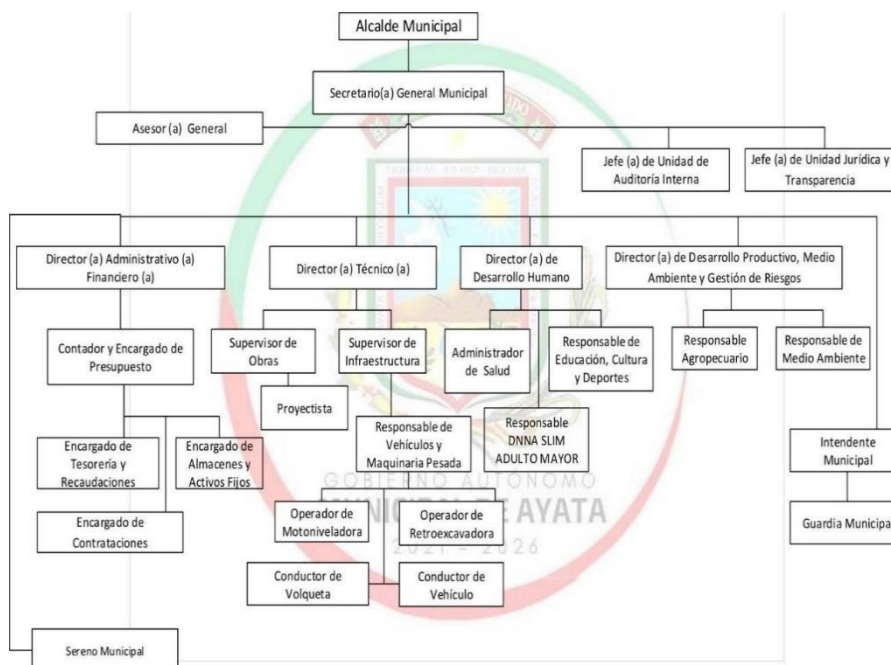
- Planificar y ejecutar los procesos de Desarrollo Integral, Humano, Económico y Social del Municipio, garantizando resultados satisfactorios, para “VIVIR BIEN”.
- Formular políticas destinadas al Desarrollo Social y Económico, con prioridad en la Salud, Educación, Producción y el mejoramiento de infraestructura.
- Formular Políticas y Normas para la Gestión Municipal.
- Promover Programas y Proyectos de Pre-Inversión e Inversión gestionando o coadyuvando el Financiamiento, como ente planificador, ejecutor y coordinador de la Segunda Sección Municipal.

1.2.1.4. Organigrama

“El Gobierno Autónomo Municipal de Ayata, está enmarcado en las Normas Legales Vigentes sobre el control, contratación y calificación de personal dependiente que está comprendida de la siguiente manera:

El G.A.M.A. establece la siguiente estructura general:

Figura 1.
Organigrama GAMA



Nota: Organigrama del GAMA, GAMA, 2020, Planta Editorial de la Alcaldía

1.2.2. Antecedentes Afines al Proyecto de Grado

1.2.2.1. Antecedentes Internacionales

Según Barreto (2017), “implementación de un sistema de trámite documentario para la mejora del proceso de atención de mesa de partes en la Municipalidad del Centro Poblado de Santa María de Huachipa” desarrollando e impulsando la mejora de servicios de dicha entidad; producto de una reforma en su proceso de tramitación. El sistema simplifica los procedimientos para mejorar la tramitación documental y asegura la calidad del servicio a los ciudadanos, basándose en la tupa.

Según Wilson (2022), en su trabajo titulado “Implementación de un sistema web de Trámite Documentario Para La Municipalidad Distrital De Ignacio Escudero – Sullana 2021”, La MDIES ejecuta en forma manual el sistema de trámite documentario, decir, la administración de los documentos en trámite, desde el registro hasta su resolución y

derivación al área de archivo. El sistema de registro, búsqueda y emisión de reportes o documentos solicitados en el trámite se realizan en forma desordenada, con procesos redundantes y una deficiente gestión de seguimiento de la hoja de ruta de cada documento en trámite. Por otro lado, se observa que el actual sistema de trámite documentario no aprovecha las tecnologías de información, los parámetros vigentes del estado y la cultura institucional, que son factores para instituir un óptimo flujo documental. Con el actual sistema de trámite documentario, el tiempo del registro y atención a los ciudadanos es muy alto; la baja cantidad de expedientes atendidos por día es bajo debido a la falta de personal generando acumulación diaria de documentos en trámite, sin ningún tipo de control para su seguimiento. La duplicidad en el número y el tras papeleo de expedientes hace lenta la búsqueda de documentos en trámite generando malestar en los ciudadanos que a diario realizan y solicitan diferentes documentos.

Según Bombón (2020), en su trabajo titulado “Sistema Informático Para Dar Seguimiento A Documentos Y Trámites En El Ilustre Municipio De Cevallos” La carencia de un sistema de control de documentos y trámites en el Municipio de Cevallos es evidente desde ya hace mucho tiempo pues al ingresar un documento no existe un respaldo de haberlo recibido así también si ya se está dando el tratamiento adecuado o está aún en circulación o lo que puede ser peor la pérdida del mismo.

1.2.2.2. Antecedentes Nacionales

Según Yana (2020), “Sistema de Gestión de Correspondencia vía web caso: área ciencias económicas, financieras y administrativas Universidad Pública De El Alto” El presente proyecto plantea un sistema de Gestión de Correspondencia vía web de gestión administrativa para los servicios de atención del solicitante de dicha correspondencia, además contienen módulos de administración de usuarios, por lo cual permiten al usuario satisfacer la necesidad de consultas, generar reportes y entablar una interfaz amigable para los usuarios.

Según lo planteado por Yucra (2016), "Sistema Web de Control y Seguimiento de Documentación Caso: Departamento de Infraestructura de la U.M.S.A." tiene como objetivo colaborar al personal del departamento para mejorar el tiempo y esfuerzo de los procesos de trabajo sobre los documentos mediante el uso de un sistema web de información, el sistema cuenta con un módulo de administración principal, que contempla la administración de los usuarios junto con su estructura organizacional y los accesos del sistema, esto permitirá configurar los permisos de acceso y acciones de los usuarios de acuerdo al rol que tenga asignado cada personal.

Según lo expuesto por Tola (2015), "Sistema De Seguimiento Y Control De Correspondencia" Caso: Enlace Consulting El proyecto Sistema de Seguimiento y Control de Correspondencia está diseñado para la Empresa Enlace Consulting Servicios multidisciplinarios, destinado a realizar el seguimiento y control de los procesos que se desarrollan en la recepción y entrega de la correspondencia que administra dicha empresa, además de proporcionar a sus clientes una mejor atención a la hora de realizar sus consultas, la empresa tiene como uno de sus hitos llegar a ser reconocida por todos sus clientes por su calidad eficiencia y eficacia en el trabajo que realiza además de proporcionar a sus clientes otros tipos de servicios como traducciones, validación de datos, servicio de traductores.

1.3. PLANTEAMIENTO DEL PROBLEMA

El Gobierno Autónomo Municipal de Ayata (GAMA), como institución pública, tiene la responsabilidad de gestionar de manera eficiente la correspondencia que recibe para atender las solicitudes de la población. Sin embargo, los procesos no están delimitados y mucho menos automatizados, lo cual ha generado inconvenientes, como demoras en la atención al público, desorganización en la clasificación de documentos y dificultades para realizar el seguimiento de la correspondencia.

La gestión manual actual implica un mayor consumo de tiempo y recursos, además de incrementar el riesgo de extravío de documentos importantes. Esto afecta la capacidad de la institución para brindar un servicio ágil y preciso, limitando la transparencia y la efectividad en la administración de la correspondencia.

1.3.1. Problema Principal

La gestión manual de la correspondencia en el Gobierno Autónomo Municipal de Ayata dificulta los procesos de registro, seguimiento, distribución y control, generando demoras, errores humanos, pérdida de información y limitando la capacidad de atender de manera eficiente las solicitudes de la población, así como de generar reportes y consultas automatizadas que fortalezcan la toma de decisiones.

1.3.2. Problemas Secundarios

- Extravío de documentos: La falta de un método adecuado para el registro y seguimiento de la correspondencia ha generado extravíos de documentos, dificultando su localización y recuperación, lo que pone en riesgo la pérdida de información importante.
- Demoras en las respuestas a los remitentes: El tiempo y los recursos necesarios para ubicar la información de una correspondencia específica causan demoras en la atención a los solicitantes.
- Desorganización de la correspondencia: La correspondencia no está organizada adecuadamente, lo que dificulta su ubicación en las unidades de trabajo correspondientes.
- La falta de control sobre el estado de los documentos: La ausencia de un sistema adecuado de control impide conocer el estado actual de los documentos.

1.3.3. Formulación del Problema

¿De qué manera se puede mejorar el control y seguimiento de la correspondencia en el Gobierno Autónomo Municipal de Ayata para contribuir a una gestión más eficiente, agilizando los procesos de registro, distribución y monitoreo de la correspondencia?

1.4. OBJETIVOS

1.4.1. Objetivo General

Desarrollar un Sistema Web para el Control y Seguimiento de Correspondencia que mejore los procesos de registro, seguimiento, distribución y control en el Gobierno Autónomo Municipal de Ayata, facilitando la atención al público y la generación automatizada de consultas y reportes estadísticos.

1.4.2. Objetivos Específicos

- Implementar un módulo que permita el registro de las correspondencias recibidas, asignando automáticamente un número único de hoja de ruta y facilitando así su seguimiento.
- Aplicar un interfaz amigable y de fácil acceso a la búsqueda de correspondencias y permita una respuesta rápida y eficiente a los remitentes.
- Desarrollar un módulo de distribución y organización que clasifique la correspondencia a las unidades correspondientes, mejorando el orden y accesibilidad de la información.
- Implementar un módulo de control de estado de correspondencia a fin de permitir a los usuarios conocer el estado y la ubicación de cada correspondencia.

- Generar reportes estadísticos de los estados de las correspondencias, ubicación en las unidades, además de informes sobre los tiempos en el procesamiento de los casos.

1.5. JUSTIFICACIÓN

1.5.1. Justificación Técnica

El Gobierno Autónomo Municipal de Ayata cuenta con equipos de hardware y software, así como acceso a Internet WiFi. La implementación de un sistema web permitirá automatizar procesos, centralizar la información de la correspondencia, agilizar la respuesta a los remitentes y mejorar la imagen de la institución. Esto se logrará utilizando recursos existentes.

1.5.2. Justificación Económica

Con el desarrollo de sistema web para control y seguimiento de correspondencia se minimizará el tiempo que se destinaba a la búsqueda de información requerida de los remitentes, también se reducirá los gastos que se realizaban en la compra de materiales de escritorio, lo que representa un ahorro en los recursos económicos de la institución.

1.5.3. Justificación Social

EL desarrollo del sistema web de control y seguimiento de correspondencia fortalecerá la imagen del Gobierno Autónomo Municipal de Ayata. Al mejorar la eficiencia en la gestión de las correspondencias recibidas, se demostrará el compromiso de la alcaldía con la atención ágil y oportuna de las necesidades de la comunidad. Esto generará una percepción positiva entre los ciudadanos y contribuirá a fortalecer la confianza en la institución. Asimismo, al proporcionar un acceso fácil y transparente a la información, se fomentará la participación ciudadana y se brindará un mayor sentido de cercanía entre la alcaldía y la población. En conjunto, estos aspectos contribuirán a una imagen sólida y favorable del Gobierno Autónomo Municipal de Ayata.

1.6. METODOLOGÍA DE DESARROLLO

Para Para la realización del proyecto se utilizará la metodología de desarrollo OOHDM es una metodología orientada a la elaboración y desarrollo de aplicaciones multimedia, pero en la actualidad se ha estado utilizando mucho más en el desarrollo de aplicaciones Web.

OOHDM ha estado evolucionando bastante desde su nacimiento y uno de sus atributos clave es que está basado en el paradigma de programación orientada a objetos que es muy útil en el desarrollo de aplicaciones y sistemas de gran magnitud

1.6.1. Metodología OOHDM (Método de Diseño Hipermedia Objeto Orientado)

El método OOHDM es una metodología de desarrollo propuesta por Rossi y Schwabe (ROSSI 1996) para la elaboración de aplicaciones multimedia y tiene como objetivo simplificar y a la vez hacer más eficaz el diseño de aplicaciones hipermedia. OOHDM está basada en HDM, en el sentido de que toma muchas de las definiciones, sobre todo en los aspectos de navegación, planteadas en el modelo de HDM. Sin embargo, OOHDM supera con creces a su antecesor, ya que no es simplemente un lenguaje de modelado, sino que define unas pautas de trabajo, centrado principalmente en el diseño, para desarrollar aplicaciones multimedia de forma metodológica (Soliz y Morales, 2014).

1.6.1.1. Facetas de la metodología OOHDM

- Obtención De Requerimientos
- Diseño Conceptual
- Diseño Navegacional
- Diseño De Interfaces Abstractas
- Implementación

1.7. MÉTRICAS DE CALIDAD

La ingeniería del software se diferencia de otras áreas, al no estar basada en leyes cuantitativas básicas, en su lugar se realiza un conjunto de medidas conocidas como métricas, las cuales proporcionan una referencia de la calidad de algún producto de software (Pressman, 2010, p.414).

1.7.1. Control de Calidad ISO 25000

La ISO 25000 proporciona una guía para el uso de la nueva serie de normas internacionales denominadas Sistemas y Requisitos de Calidad del Software y Evaluación (SQuaRE). El objetivo de ISO 25000 es proporcionar una visión general de los contenidos de SQuaRE, modelos de referencia y definiciones comunes, así como la relación entre los documentos. La última revisión se produjo en 2014, por lo que el próximo año será revisada y actualizada.

Los requisitos de la ISO 25000 se centran además en dos procesos principales: especificación de requisitos de calidad del software y evaluación de la calidad del software, soportada por el proceso de medición de calidad del software (Crespo, 2018 p 278).

1.8. ESTIMACIÓN DE COSTOS COCOMO

Una de las tareas de mayor importancia en la planificación de proyectos de software es la estimación, la cual consiste en determinar, con cierto grado de certeza, los recursos de hardware y software, costo, tiempo y esfuerzo necesarios para el desarrollo de los mismos. COCOMO, este modelo permite realizar estimaciones en función del tamaño del software, y de un conjunto de factores de costo y de escala, se engloba en el grupo de los modelos algorítmicos que tratan de establecer una relación matemática la cual permite estimar el esfuerzo y tiempo requerido para desarrollar un producto (Fernández, 2016,p.145).

1.9. SEGURIDAD ISO/IEC 27001

Para el fin de preservar la información, se ha demostrado que no es suficiente la implantación de controles y procedimientos de seguridad realizados frecuentemente sin un criterio común establecido, en torno a la compra de productos técnicos y sin considerar toda la información esencial que se debe proteger.

La Organización Internacional de Estandarización (ISO), a través de las normas recogidas en ISO / IEC 27000, establece una implementación efectiva de la seguridad de la información empresarial desarrolladas en las normas ISO 27001 / ISO 27002. (Alonso, 2015).

1.10. PRUEBAS DE SOFTWARE

Las pruebas son básicamente un conjunto de actividades dentro del desarrollo de software. Dependiendo del tipo de pruebas, estas actividades podrán ser implementadas en cualquier momento de dicho proceso de desarrollo. Existen distintos modelos de desarrollo de software, así como modelos de pruebas. A cada uno corresponde un nivel distinto de involucramiento en las actividades de desarrollo. En nuestro caso nos enfocaremos más a la funcionalidad y estructura de nuestro proyecto por (Aguilar, 2015, p.245).

1.10.1. Pruebas de Caja Blanca

La prueba de caja blanca, en ocasiones llamada prueba de caja de vidrio, es una filosofía de diseño de casos de prueba que usa la estructura de control descrita como parte del diseño a nivel de componentes para derivar casos de prueba (Pressman, 2010, p.441)

Al usar los métodos de prueba de caja blanca, puede derivar casos de prueba que:

- Garanticen que todas las rutas independientes dentro de un módulo se revisaron al menos una vez
- Revisen todas las decisiones lógicas en sus lados verdadero y falso

- Ejecuten todos los bucles en sus fronteras y dentro de sus fronteras operativas.
- Revisen estructuras de datos internas para garantizar su validez.

1.10.2. Pruebas de Caja Negra

Las pruebas de caja negra, también llamadas pruebas de comportamiento, se enfocan en los requerimientos funcionales del software; es decir, las técnicas de prueba de caja negra le permiten derivar conjuntos de condiciones de entrada que revisarán por completo todos los requerimientos funcionales para un programa. Las pruebas de caja negra no son una alternativa para las técnicas de caja blanca. En vez de ello, es un enfoque complementario que es probable que descubra una clase de errores diferente que los métodos de caja blanca. (Pressman, 2010, p.423).

1.10.3. Pruebas de Estrés

Las pruebas de estrés se diseñan para simular situaciones de carga del mundo real. Conforme aumenta el número de usuarios simultáneos de la aplicación o el número de transacciones en línea o la cantidad de datos descargados o subidos estas pruebas de rendimiento ayudarán a responder preguntas sobre el rendimiento de la aplicación. (Pressman, 2023, p. 210)

1.10.4. Pruebas de Accesibilidad

Las WAVE es una herramienta de evaluación desarrollada para ayudar a los diseñadores y desarrolladores web a identificar problemas de accesibilidad en sus sitios web, proporcionando retroalimentación visual directa sobre los elementos de accesibilidad y su cumplimiento con los estándares establecidos"(WebAIM, 2023).

1.11. HERRAMIENTAS

Para el desarrollo de sistema web para el control seguimiento de correspondencia, se han seleccionado diversas herramientas y tecnologías, cada una con una función específica en la estructura y funcionalidad del proyecto. A continuación, se detallan las principales:

- Crear y administrar la base de datos, los usuarios y los permisos.
- Diseñar y ejecutar las instrucciones SQL.
- Realizar las consultas requeridas por el usuario.

1.11.1. Lenguajes de Programación

- PHP: Es el lenguaje de programación principal utilizado para desarrollar la lógica del sistema y manejar la interacción con la base de datos. (Lerdorf, 1995,p.86)
- HTML5, CSS y JavaScript: Se emplean para construir la interfaz de usuario, proporcionando estructura (HTML5), estilo (CSS) e interactividad (JavaScript). (W3C, 2023)

1.11.2. Plantillas y Librerías

- AdminLTE: Esta plantilla de administración basada en Bootstrap se utiliza para mejorar y unificar el aspecto visual de la interfaz del sistema, permitiendo un diseño profesional y consistente. (AdminLTE, 2023)
- jQuery y AJAX (Asynchronous JavaScript and XML): Estas herramientas facilitan la comunicación asíncrona entre el cliente y el servidor, mejorando la interactividad y la experiencia del usuario al hacer la interfaz más dinámica. (jQuery, 2023; Mozilla, 2023)

1.11.3. Base de Datos

- MariaDB: Sistema de gestión de bases de datos utilizado para almacenar toda la información relevante sobre la correspondencia y su seguimiento. (MariaDB Foundation, 2023)
- Navicat: Herramienta de administración de bases de datos que permite gestionar y consultar los datos de forma visual, facilitando el control y la organización de la base de datos. (PremiumSoft, 2023)

1.11.4. Servidor Local

- Apache: Servidor web que permite ejecutar el sistema en un entorno local. Funciona en conjunto con XAMPP para el desarrollo y pruebas del sistema. (Apache Software Foundation, 2023)
- XAMPP: Plataforma de desarrollo que incluye Apache, proporcionando un entorno de servidor local para ejecutar y probar la aplicación web. (Apache Friends, 2023)

1.11.5. Entorno de Desarrollo

- Visual Studio Code: Editor de código principal utilizado para escribir y editar el código del sistema, con soporte para múltiples lenguajes y extensiones que facilitan el desarrollo. (Microsoft, 2023)

1.11.6. Modelado de Diagramas

- Enterprise Architect: Herramienta utilizada para el modelado de diagramas UML, facilitando la visualización y documentación de la arquitectura del sistema, así como su estructura interna. (Sparx Systems, 2023)

1.12. ALCANCES Y LIMITES

1.12.1. Alcances

El Sistema Web de Control y Seguimiento de Correspondencia tendrá los siguientes alcances:

- **Registro de Correspondencia:** Se implementará una funcionalidad que permitirá a la secretaria realizar el registro de la correspondencia recibida. Esta acción incluirá la captura de información relevante, como remitente, destinatario, asunto, fecha de recepción y la asignación de una hoja de ruta para su seguimiento.
- **Asignación de Correspondencia:** El asesor general tendrá la capacidad de derivar la correspondencia registrada a las diferentes áreas. Esta funcionalidad facilitará la distribución eficiente de las tareas de revisión y respuesta.
- **Derivación de Correspondencia:** Los jefes de área podrán recibir, derivar o archivar la correspondencia según corresponda. Esta funcionalidad asegurará que cada unidad gestione la correspondencia de manera efectiva, de acuerdo con sus responsabilidades y competencias.
- **Módulo de Búsqueda:** Se implementará un módulo de búsqueda que permitirá a los usuarios buscar correspondencia ingresando el número de la hoja de ruta. Esta funcionalidad facilitará el seguimiento de la correspondencia y permitirá a los usuarios acceder rápidamente a la información relevante.

1.12.2. Límites

El desarrollo del sistema web de control y seguimiento de correspondencia, se establecen los siguientes límites:

- No se digitalizan los documentos físicos: El sistema no contempla la digitalización de la correspondencia en papel. La gestión de documentos se realizará en formato físico, y su registro en el sistema se llevará a cabo de manera manual.
- Falta de firma digital: El sistema no incorpora un mecanismo de firma digital para validar documentos. En situaciones donde se requiera una validación, será necesario que los jefes o responsables firmen los documentos de forma manual.
- Alcance limitado a la correspondencia interna: Este sistema se diseñará principalmente para el manejo de correspondencia interna dentro del Gobierno Autónomo Municipal de Ayata, y no incluirá funcionalidades para gestionar la correspondencia externa o interacciones con otras entidades.

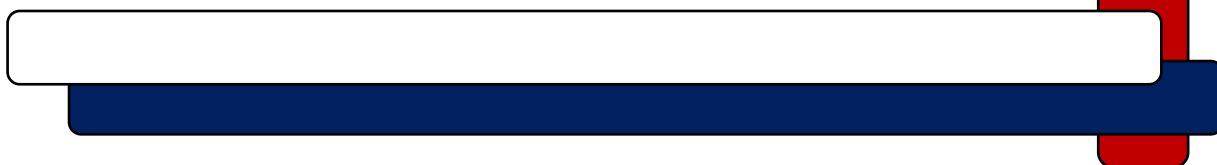
1.13. APORTES

El sistema mejora significativamente el control de la correspondencia al automatizar los procesos de registro, seguimiento y distribución, lo que permite reducir errores, evitar extravíos y garantizar un acceso rápido y eficiente a la información. Esto contribuye a optimizar la gestión administrativa del Gobierno Autónomo Municipal de Ayata y a brindar una atención más oportuna a la ciudadanía.

Además, centraliza y organiza los datos relacionados con la correspondencia, facilitando la generación de reportes y estadísticas que fortalecen la toma de decisiones estratégicas, promoviendo una administración más transparente, ágil y efectiva.

CAPÍTULO II MARCO TEÓRICO

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



2. CAPÍTULO II MARCO TEÓRICO

2.1. INTRODUCCIÓN

El Sistema Web para el Control y Seguimiento de Correspondencia del Gobierno Autónomo Municipal de Ayata permitirá mantener la información y funciones accesibles a los usuarios según sus privilegios, a través de un navegador web. Esta característica agilizará las actividades del personal de la unidad, optimizando el manejo de la correspondencia y reduciendo su carga de trabajo, al asignar y derivar algunas de las tareas actuales hacia otros responsables del proceso

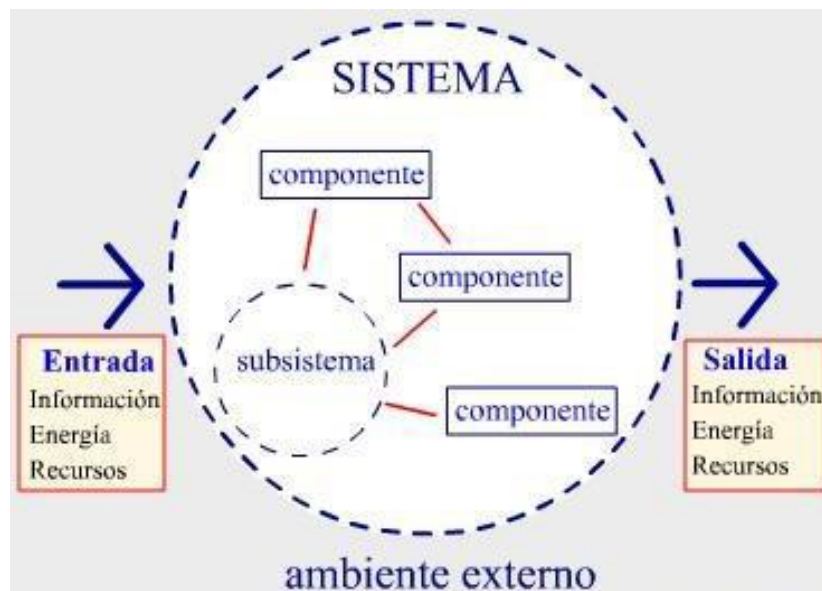
2.2. CONCEPTOS BÁSICOS

2.2.1. *Sistema*

Un sistema es un conjunto de elementos interrelacionados que trabajan de manera coordinada para lograr un propósito común. Aunque cada elemento puede operar de forma independiente, su función individual se encuentra integrada en una estructura más amplia que permite el funcionamiento del sistema como un todo(Lujan, 2019,p.77).

Un sistema es un módulo ordenado de elementos que se encuentran interrelacionados y que interactúan entre sí, se utiliza tanto para definir a un conjunto de procesos como también para relacionar los objetos reales dotados en la organización, Los sistemas reales comprenden intercambios de energía, información o materia con su entorno. Las células y la biosfera son ejemplos de sistemas naturales.

Los sistemas también pueden servir como guía de trabajo más práctico y simplificado como el que se muestra en la figura., también dependerá de la manera en que necesites emplear un sistema y sus diferentes fases, para la elaboración de un trabajo que conlleve diferentes metas, pero con un mismo propósito.

Figura 2.*Teoría de un Sistema*

Nota. *Funcionalidad del sistema, Pressma, 2018, conceptos básicos de un sistema.*

2.2.2. Web

La World Wide Web (WWW), también conocida simplemente como web, es un sistema interconectado de documentos y otros recursos, accesibles a través de Internet, que están identificados por URLs y son accesibles mediante un navegador web. Estos recursos pueden incluir texto, imágenes, videos, y otros elementos multimedia, y se accede a ellos principalmente utilizando el protocolo HTTP (HyperText Transfer Protocol). La web permite la interactividad entre usuarios y sistemas, y ha revolucionado la forma en que se comparte la información globalmente (Berners, 1999).

2.2.3. Sistema web

Un sistema web se define como una aplicación informática que opera a través de una red, generalmente internet, permitiendo a los usuarios interactuar con interfaces gráficas mediante navegadores web para realizar diversas tareas.

Por otro lado, (Kroski ,2008) menciona que "los sistemas web permiten un acceso flexible y centralizado a la información, siendo clave para la digitalización de procesos en instituciones y organizaciones".

El sistema Web o también denominado aplicaciones Web se define como aplicaciones de software que se puede usar en un servicio Web por medio de internet o de una intranet desde un navegador. (System, 2022)

2.2.4. Correspondencia

La correspondencia es un elemento importante en las organizaciones diariamente, ya que la misma nos permite enviar y recibir documentos y no sólo se pueden enviar documentos si no también paquetes ahorrando tiempo. Dicho contenido presenta información importante acerca de la correspondencia, la carta y como debe de elaborarse un sobre, el cual es un valioso instrumento en el ámbito empresarial. El sobre y la redacción de la carta juegan un papel muy importante y fundamental en este proceso de comunicación, siguiendo los pasos estipulados se puede lograr hacer un buen uso de la correspondencia.

Es el trato recíproco entre dos personas mediante el intercambio de cartas, tarjetas, telegramas, catálogos, folletos, etc. En las empresas se considera que la correspondencia es el alma del comercio y la industria. Es un medio de comunicación usado por el hombre desde hace muchos años para comunicarse

entre dos personas o individuos que están a larga distancia o cerca con un motivo muy variado. A lo largo del tiempo se ha perfeccionado sus normas y estilos, hasta llegar a nuestros días que existe el e-mail, que es la forma más rápida de enviar y asegurarse que la información llegue al destinatario. (Andino, 2001,p.149)

2.2.4.1. Importancia de la Correspondencia

Es un importante instrumento de comunicación escrita, es la parte intermedia entre las relaciones cliente proveedor y el lazo que une la mayor parte de transacciones comerciales.

De ella depende el desarrollo de las operaciones comerciales; el éxito de un negocio, una venta, por su eficiencia y rapidez han a que las empresas aumenten el volumen de sus ventas (Fernandes,2020,p.235).

2.2.4.2. División de la correspondencia

Correspondencia comercial: Se refiere a la cruzada entre comerciantes, industrias, banqueros, etc. Y su finalidad es promover y agilizar las diversas transacciones comerciales.

Correspondencia Familiar y amistosa: La finalidad de esta correspondencia es muy variada cualquier asunto particular entre ella.

Correspondencia Oficial: Se refiere a la correspondencia cruzada entre los distintos organismos y oficinas de los gobiernos nacionales, municipales, estatales (Boccaletti, 2006, p.65).

2.2.5. Seguimiento

El seguimiento consiste en el análisis y recopilación sistemáticos de información a medida que avanza un trámite. Su objetivo es mejorar la eficacia y efectividad de un trámite. Se basa en metas establecidas y actividades planificadas durante las distintas fases del trabajo de planificación. Ayuda a que se siga una línea de trabajo, y además permite a la administración conocer cuando algo no está funcionando(Kerzner, 2017,p.208)

Las medidas de control deben ser estandarizables. Las variables seleccionadas deben tener capacidad para registrar las propiedades de los sistemas afectados que se desea

valorar, ser indicadores del estado del conjunto de un proceso. La escala de las mediciones debe ser la apropiada.

El seguimiento (también denominado 'acompañamiento' o 'monitoreo') y la evaluación son como dos caras de una misma moneda. Es la forma que tenemos de saber si hemos aprovechado o no nuestro tiempo en el desarrollo de un proyecto. El seguimiento es un proceso continuo, mientras que la evaluación generalmente se hace al finalizar una etapa o al llegar al final del proyecto. De todas formas, ambos elementos marchan unidos, puesto que el seguimiento es una forma de ir evaluando día a día el proyecto y en definitiva nos servirá para llegar al momento de la evaluación con más información, además de permitirnos la realización de ajustes periódicos (Prieto, 2017, p.60).

2.2.6. Control

El término control es de uso frecuente y se utiliza al expresar que algo o alguien ha sido objeto de una corroboración, examen, verificación, inspección, revisión o supervisión. En relación con esto, controlar implica esencialmente la medición y posterior corrección de las actividades de la organización para que ellas estén alineadas con los planes y los objetivos fijados por la dirección. Para el autor, el control es "básicamente información para permitir la corrección oportuna de las desviaciones de las acciones reales con respecto al plan. El control puede ser definido como el conjunto de actividades que tiene como objetivo hacer realidad una serie de hechos deseados" (Zuani, 2005, p. 375).

Se define al control "como la medición y corrección del desempeño para garantizar que los objetivos de la empresa y los planes diseñados para alcanzarlos se logren" (Wehrich, 2007, p. 372).

El control “es el proceso que consiste en supervisar las actividades para garantizar que se realicen según lo planeado y corregir cualquier desviación significativa” (Coulter, 2014, p. 266).

2.2.7. Control y Seguimiento

El seguimiento consiste básicamente en el análisis de la información, mediante proyecciones relacionadas con los tiempos que son generadas en un proyecto, para la identificación temprana de riesgos y desviaciones respecto al plan. Por su parte el control comprende el desarrollo de las actuaciones para conseguir que lo planificado y esperado ocurra. Por lo tanto, controlar no significa sólo identificar, sino que la esencia del control supone indagar en las causas que puedan conllevar a definir las acciones e implementarlas de manera que sus efectos lleven a minimizar riesgos o peligros dentro de un proyecto establecido. (Sanz, 2016,p.178)

Los procesos que ayudan en gestionar labores de una magnitud amplia, por lo general siempre coinciden con los pasos que se establecen antes de dar forma al proyecto, pero este se basa en lo que se conoce como control y seguimiento (Junta de andalucia, 2015) “El procedimiento de Seguimiento y Control, establece el conjunto de acciones que se llevarán a cabo para la comprobación de la correcta ejecución de las actividades del proyecto establecidas en la planificación del mismo”. Su propósito es proporcionar un entendimiento del progreso del proyecto de forma que se puedan tomar las acciones correctivas apropiadas cuando la ejecución del proyecto se desvíe significativamente de su planificación.

2.2.8. Hojas de Ruta

Una hoja de ruta es un plan que establece a grandes rasgos la secuencia de pasos para alcanzar un objetivo. Se especifican tiempo y recursos necesarios. Se debe entender como recursos al envío y recepción de correspondencia a la totalidad de área hasta

transformarlo en otro y que debe pasar en un determinado tiempo establecido por el requerimiento de alcance de objetivo (Edwin 2017, p.45):

2.3. METODOLOGÍA DE DESARROLLO

Las metodologías de desarrollo de software se utilizan en el ámbito de la programación, entre otros, con el objetivo de trabajar en equipo de manera organizada. Estas metodologías han ido evolucionando a lo largo del tiempo, pasando de ser un mero trámite de organización a ser una base importantísima a la hora de desarrollar software de una manera productiva y eficaz. (Zaens, 2010, p. 7).

2.3.1. Metodología OOHDM

Es OOHDM (Método de Diseño Hipermedia Objeto Orientado) es una metodología de desarrollo propuesta por (ROSSI 1996) para la elaboración de aplicaciones multimedia y tiene como objetivo simplificar y a la vez hacer más eficaz el diseño de aplicaciones hipermedia. OOHDM está basada en HDM, en el sentido de que toma muchas de las definiciones, sobre todo en los aspectos de navegación, planteadas en el modelo de HDM. Sin embargo, OOHDM supera con creces a su antecesor, ya que no es simplemente un lenguaje de modelado, sino que define unas pautas de trabajo, centrado principalmente en el diseño, para desarrollar aplicaciones multimedia de forma metodológica (Morales, 2014, p.250)

2.3.2. Fases de la Metodología OOHDM.

“OOHDM comprende cuatro actividades diferentes, a saber, diseño conceptual, diseño de navegación, diseño e implementación de interfaz abstracta. Se realizan en una combinación de estilos de desarrollo incrementales, iterativos y basados en prototipos. Durante cada actividad, se crea o enriquece un conjunto de modelos orientados a objetos que describen preocupaciones de diseño particulares a partir de iteraciones anteriores” (Schwabe, 1998, p. 1).

Sin embargo, a través del tiempo, diversos autores señalan la necesidad de incluir una actividad adicional inicial: la obtención de requerimientos.

Esta metodología está compuesta por cinco fases:

Obtención de Requerimientos

Diseño conceptual

Diseño navegacional

Diseño de interfaces abstractas

Implementación

Figura 3.

Fase de la Metodología OOADM



2.3.2.1. Obtención de Requerimientos

En esta fase se identifican los actores y las tareas o funciones que cada uno podrá realizar. Posteriormente se identifica la información que los actores deberán proveer al sistema, así como el resultado que esperan obtener del mismo. El artefacto con el que se diseña esta fase son los diagramas de casos de uso.

De acuerdo con Rumbaugh (2007), el objetivo de los diagramas de casos de uso es “Modelar la funcionalidad de un sistema tal como lo perciben los agentes externos”, denominados actores, que interactúan con el sistema desde un punto de vista particular.

Un caso de uso es una unidad de funcionalidad expresada como una transacción entre los actores y el sistema. (Booch, 2007, pág. 31).

Figura 4.

Casos de Usos



Nota. Casos de usos, Booch, 2007, introducción a los Casos de Usos.

2.3.2.2. Diseño Conceptual

En esta fase se construye el modelo orientado a objetos que represente el dominio de la aplicación y el resultado de esta fase nos arroja el modelo de clases con sus relaciones. El artefacto con el que se diseña esta fase es el diagrama de clases.

“En OOADM, el esquema conceptual está construido por clases, relaciones y subsistemas. Las clases son descritas como en los modelos orientados a objetos tradicionales. Sin embargo, los atributos pueden ser de múltiples tipos para representar perspectivas diferentes de las mismas entidades del mundo real” (Mercerat, 2001, p. 4).

2.3.2.3. Diseño Navegacional

En esta fase se debe diseñar la aplicación tomando en cuenta las tareas que el usuario va a realizar sobre el sistema, lo anterior basado en el modelo de clases generado en la fase anterior. Como resultado se obtendrán las clases navegacionales que se componen de

enlaces (que se derivan de las relaciones), nodos (que representan las vistas), y las estructuras de acceso. En esta fase se pueden utilizar técnicas de modelado orientadas a objetos, así como patrones de diseño.

Ya que no existe un diagrama UML específico para la implementación de esta fase, para la construcción del diagrama de clases navegacionales es factible utilizar un diagrama similar al implementado en la fase anterior, pero haciendo uso de las extensiones Web para UML (WAE), establecidas por Jim Conallen (1999): “Una extensión de UML se expresa en términos de estereotipos, valores etiquetados y restricciones. Combinados, estos mecanismos nos permiten extender la notación de UML, lo que nos permite crear nuevos tipos de bloques de construcción que podemos usar en el modelo. El estereotipo, una extensión del vocabulario del lenguaje, nos permite adjuntar un nuevo significado semántico a un elemento modelo. Los estereotipos se pueden aplicar a casi todos los elementos del modelo y generalmente se representan como una cadena entre un par de comillas angulares: «». Sin embargo, también se pueden representar con un nuevo icono” (Conallen, 1999, p. 4).

Por otra parte, el diagrama de contexto navegacional, supone el camino o ruta que determinado actor puede realizar para cumplir una acción en particular. Esto se puede diseñar en forma particular para cada actor de acuerdo a los permisos que se le otorguen, o de forma general indicando las distintas rutas existentes en todo el sistema Web.

2.3.2.4. Diseño de Interfaz Abstracta

Esta fase comprende la definición de objetos que serán perceptibles para el usuario. La separación entre el diseño navegacional y el diseño de interfaz abstracta permitirá construir diferentes interfaces para el mismo modelo navegacional. En esta fase se utilizan los Modelos de Vistas Abstractas de Datos, cuya función es representar las entradas y salidas de la interfaz que interactúan con el usuario, también llamados ADV por sus siglas en inglés (Abstract Data

View), cuya función es representar las entradas y salidas de la interfaz que interactúan con el usuario.

“Los ADV son objetos en el sentido de que tienen un estado y una interfaz, donde la interfaz se puede ejercitar a través de mensajes (en particular, eventos externos generados por el usuario). Los ADV son abstractos en el sentido de que solo representan la interfaz y el estado, y no la implementación” (Schwabe & Rossi, 1998, pág. 14).

2.3.2.5. Implementación

Una vez concluidas las fases mencionadas anteriormente, únicamente es necesario llevar los objetos a un lenguaje concreto de programación, para obtener así la implementación ejecutable de la aplicación.

2.4. ARQUITECTURA DE SOFTWARE (MVC)

MVC es una propuesta de arquitectura del software utilizada para separar el código por sus distintas responsabilidades, manteniendo distintas capas que se encargan de hacer una tarea muy concreta, lo que ofrece beneficios diversos.

MVC se usa inicialmente en sistemas donde se requiere el uso de interfaces de usuario, aunque en la práctica el mismo patrón de arquitectura se puede utilizar para distintos tipos de aplicaciones. Surge de la necesidad de crear software más robusto con un ciclo de vida más adecuado, donde se potencie la facilidad de mantenimiento, reutilización del código y la separación de conceptos.

MVC es un "invento" que ya tiene varias décadas y fue presentado incluso antes de la aparición de la Web. No obstante, en los últimos años ha ganado mucha fuerza y seguidores gracias a la aparición de numerosos frameworks de desarrollo web que utilizan el patrón MVC como modelo para la arquitectura de las aplicaciones web (Gonzales, 2021, p. 10).

2.4.1. Modelo

Es la capa donde se trabaja con los datos, por tanto, contendrá mecanismos para acceder a la información y también para actualizar su estado. Los datos los tendremos habitualmente en una base de datos, por lo que en los modelos tendremos todas las funciones que accederán a las tablas y harán los correspondientes selects, updates, inserts

El modelo es el responsable de:

Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.

Define las reglas de negocio (la funcionalidad del sistema). Un ejemplo de regla puede ser: "Si la mercancía pedida no está en el almacén, consultar el tiempo de entrega estándar del proveedor".

Lleva un registro de las vistas y controladores del sistema.

Si estamos ante un modelo activo, notificará a las vistas los cambios que en los datos pueda producir un agente externo (por ejemplo, un fichero por lotes que actualiza los datos, un temporizador que desencadena una inserción.).

2.4.2. Vista

Las vistas, como su nombre nos hacen entender, contienen el código de nuestra aplicación que va a producir la visualización de las interfaces de usuario, o sea, el código que nos permitirá renderizar los estados de nuestra aplicación en HTML. En las vistas nada más tenemos los códigos HTML y PHP que nos permite mostrar la salida.

En la vista generalmente trabajamos con los datos, sin embargo, no se realiza un acceso directo a éstos. Las vistas requerirán los datos a los modelos y ellas se generará la

salida, tal como nuestra aplicación requiera. Este presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario.

Las vistas son responsables de:

Recibir datos del modelo y los muestra al usuario.

Tienen un registro de su controlador asociado (normalmente porque además lo instancia).

Pueden dar el servicio de "Actualización ()", para que sea invocado por el controlador o por el modelo (cuando es un modelo activo que informa de los cambios en los datos producidos por otros agentes).

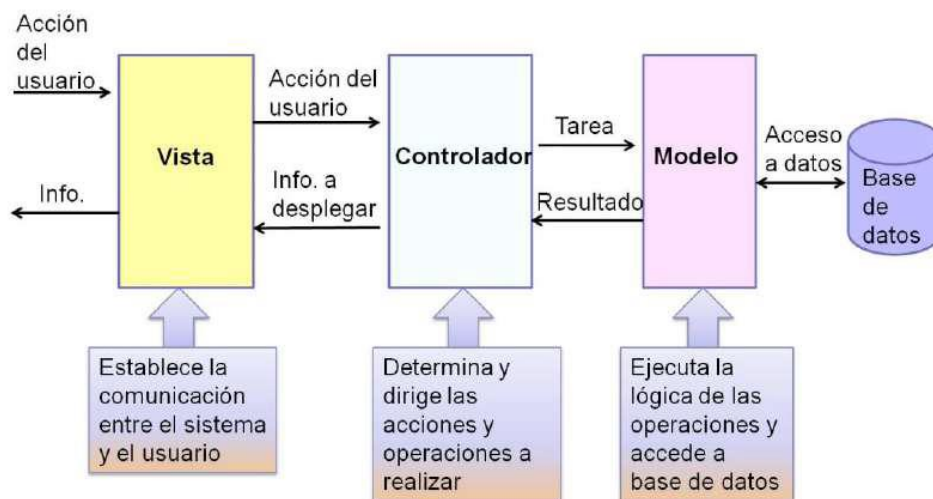
2.4.3. Controlador

Contiene el código necesario para responder a las acciones que se solicitan en la aplicación, como visualizar un elemento, realizar una compra, una búsqueda de información, etc.

En realidad, es una capa que sirve de enlace entre las vistas y los modelos, respondiendo a los mecanismos que puedan requerirse para implementar las necesidades de nuestra aplicación. Sin embargo, su responsabilidad no es manipular directamente datos, ni mostrar ningún tipo de salida, sino servir de enlace entre los modelos y las vistas para implementar las diversas necesidades del desarrollo.

A continuación, encontrarás un diagrama que te servirá para entender un poco mejor cómo colaboran las distintas capas que componen la arquitectura de desarrollo de software en el patrón MVC.

Figura 5.

Modelo Vista Controlador**Nota (característico en un esquema MVC)**

Veamos paso a paso cómo sería el flujo de trabajo característico en un esquema MVC.

- El usuario **realiza una solicitud** a nuestro sitio web. Generalmente estará desencadenada por acceder a una página de nuestro sitio. Esa solicitud le llega al controlador.
- El **controlador comunica tanto con modelos como con vistas**. A los modelos les solicita datos o les manda realizar actualizaciones de los datos. A las vistas les solicita la salida correspondiente, una vez se hayan realizado las operaciones pertinentes según la lógica del negocio.
- Para producir la salida, en ocasiones las **vistas pueden solicitar más información a los modelos**. En ocasiones, el controlador será el responsable de solicitar todos los datos a los modelos y de enviarlos a las vistas, haciendo de puente entre unos y otros. Sería corriente tanto una cosa como la otra, todo depende de nuestra implementación; por eso esa flecha la hemos coloreado de otro color.

- **Las vistas envían al usuario la salida.** Aunque en ocasiones esa salida puede ir de vuelta al controlador y sería éste el que hace el envío al cliente, por eso he puesto la flecha en otro color.

2.5. HERRAMIENTAS

2.5.1. *Herramientas de Desarrollo Front-End*

HTML5 ofrece principalmente tres características: estructura, estilo y funcionalidad. Aunque no fue declarado oficialmente, se considera que HTML5 es el resultado de la combinación de HTML, CSS y JavaScript, a pesar de que algunas APIs (Interfaces de Programación de Aplicaciones) y la especificación completa de CSS3 no forman parte de este. Estas tecnologías son altamente interdependientes y funcionan como una unidad organizada bajo la especificación de HTML5.

HTML se encarga de la estructura, CSS presenta esa estructura y su contenido en la pantalla, y JavaScript añade la funcionalidad (como veremos más adelante), lo cual es extremadamente significativo. Más allá de esta integración, la estructura sigue siendo un componente esencial de un documento, ya que proporciona los elementos necesarios para ubicar contenido estático o dinámico, además de ser una plataforma básica para aplicaciones.

Dada la variedad de dispositivos para acceder a Internet y la diversidad de interfaces disponibles para interactuar con la web, un aspecto fundamental como la estructura se vuelve vital. Esta debe proporcionar forma, organización y flexibilidad, siendo tan sólida como los fundamentos de un edificio, según (Gauchat 2012,p.82).

CSS es un lenguaje que facilita instrucciones que podemos usar para asignar estilos a los elementos HTML, como colores, tipos de letra, tamaños, etc. Los estilos se deben definir con CSS y luego asignar a los elementos hasta que logramos el diseño visual que queremos para nuestra página. (Gauchat, 2017p. 83).

JavaScript es un lenguaje de programación que se usa para procesar información y manipular documentos. Al igual que cualquier otro lenguaje de programación, JavaScript provee instrucciones que se ejecutan de forma secuencial para indicarle al sistema lo que queremos que haga (realizar una operación aritmética, asignar un nuevo valor a un elemento, etc.). Cuando el navegador encuentra este tipo de código en nuestro documento, ejecuta las instrucciones al momento y cualquier cambio realizado en el documento se muestra en pantalla. (Gauchat, 2017 p. 241).

- **Bootstrap**

Según Latincloud (2023) explica sobre Bootstrap: Bootstrap es un conjunto de herramientas de código abierto para desarrollos web del tipo responsive, con HTML, CSS y JavaScript. Esas herramientas permiten darle forma a tu sitio web por medio del uso de sus librerías CSS y JavaScript. Además, incorpora distintos componentes: ventanas modales, botones, menús, cuadros, formularios.

Bootstrap es un buen instrumento para crear interfaces de usuario sencillas, y completamente adaptables a todo tipo de dispositivos y pantallas, cualquiera sea su formato. Desde Bootstrap 4, el framework se volvió más compatible con el desarrollo web responsive.

- **Jquery**

Es una biblioteca de JavaScript rápida y concisa que simplifica el documento HTML, manejo de eventos, animación y las interacciones AJAX para el desarrollo web. jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio (Fuentes, 2013, p.45).

- **Ajax**

AJAX no es un lenguaje de programación. Es una técnica para crear mejores y más rápidas aplicaciones Web usando sus mismas tecnologías (HTML, CSS, JavaScript y XML). Dado que usa funcionalidad nativa de los navegadores no requiere plug-in. Usa el objeto de los navegadores XMLHttpRequest, que permite hacer requerimientos al servidor sin refresco completo de la página (Bazán,2020).

- **Visual Studio Code**

es un editor de código fuente ligero pero potente que se ejecuta en el escritorio y está disponible para Windows, macOS y Linux. Viene con soporte incorporado para JavaScript, TypeScript y Node.js, y cuenta con un rico ecosistema de extensiones para otros lenguajes (como C++, C#, Java, Python, PHP y Go) y tiempos de ejecución (como .NET y Unity) (Neerajnandwana-msft, 2020, p. 58).

Multiplataforma: Esta característica es fundamental en cualquier aplicación de desarrollo. Visual Studio Code está disponible para Windows, GNU/Linux y macOS.

Extensiones: He mencionado varias veces el término "extensiones" porque es uno de los puntos fuertes del editor. Visual Studio Code es potente, en gran parte, gracias a las extensiones. Estas nos permiten personalizar y agregar funcionalidad adicional de forma modular y aislada. Por ejemplo, podemos programar en diferentes lenguajes, agregar nuevos temas al editor y conectarnos a otros servicios. Las extensiones mejoran nuestra experiencia sin afectar el rendimiento del editor, ya que se ejecutan en procesos independientes (Flores, 2022, p.58).

AdminLTE: Esta plantilla de administración basada en Bootstrap se utiliza para mejorar y unificar el aspecto visual de la interfaz del sistema, permitiendo un diseño profesional y consistente. (AdminLTE, 2023)

2.5.2. Herramientas de Desarrollo Back End

PHP, acrónimo de Hypertext Preprocessor, es un lenguaje de programación orientado a objetos, de código abierto muy popular para el desarrollo web que puede ajustarse en HTML. Es considerado como uno de los lenguajes más potentes y populares para el desarrollo web y de aplicaciones. (González, 2006,p.56)

Las páginas de PHP contienen HTML con código incrustado que cumple una función y está encerrado entre las etiquetas especiales de comienzo y final.

Lo que distingue a PHP es su extrema simplicidad para programadores, desarrolladores o ingenieros en un nivel principiante, pero al mismo tiempo ofrece una gran variedad de características avanzadas para los profesionales en niveles más desarrollados.

2.5.3. Funciones del lenguaje PHP

- Programación del lado del servidor: ideal para páginas webs de prueba antes de la publicación en la web.
- Programación a través de la línea de comandos: los scripts PHP pueden crearse sin necesidad de un navegador o servidor, sólo se requiere de un analizador PHP. Los scripts son ideales para las tareas regulares de una web.
- Aplicaciones de escritorio: a pesar de ser una función que se utiliza con menor frecuencia, el lenguaje PHP también es adecuado para el desarrollo de aplicaciones de escritorio gracias a su nivel de complejidad.

2.5.4. Base de Datos

MariaDB es un sistema de gestión de bases de datos que está muy relacionado con MySQL, ya que fue desarrollado por uno de los desarrolladores, Michael "Monty" Widenius. El

objetivo de su desarrollo fue el de mantener el software de gestión de base de datos en un modelo de software libre.

El sistema de gestión de bases de datos MariaDB incorpora las distintas funciones características de MySQL añadiendo algunas mejoras, como la posibilidad de ejecutar consultas complejas y almacenarlas directamente en caché, la nueva gestión de conexiones a BD, la posibilidad de acceder a cluster de datos (interesante para el trabajo en la nube) o el soportar la utilización de jerarquías de graphs y estructuras más complejas.

En cuanto a seguridad y rendimiento, MariaDB incorpora mejoras, estando siempre en constante evolución gracias a la aportación de una gran comunidad que se encuentra tras de ella. (HostingPlus ,2020 párr. 3).

Navicat: Herramienta de administración de bases de datos que permite gestionar y consultar los datos de forma visual, facilitando el control y la organización de la base de datos. (PremiumSoft, 2023)

2.5.5. Servidor Local

- Apache: Servidor web que permite ejecutar el sistema en un entorno local. Funciona en conjunto con XAMPP para el desarrollo y pruebas del sistema. (Apache Software Foundation, 2023)
- XAMPP: Plataforma de desarrollo que incluye Apache, proporcionando un entorno de servidor local para ejecutar y probar la aplicación web. (Apache Friends, 2023)

2.5.6. Modelado de Diagramas

Enterprise Architect: Herramienta utilizada para el modelado de diagramas UML, facilitando la visualización y documentación de la arquitectura del sistema, así como su estructura interna. (Sparx Systems, 2023)

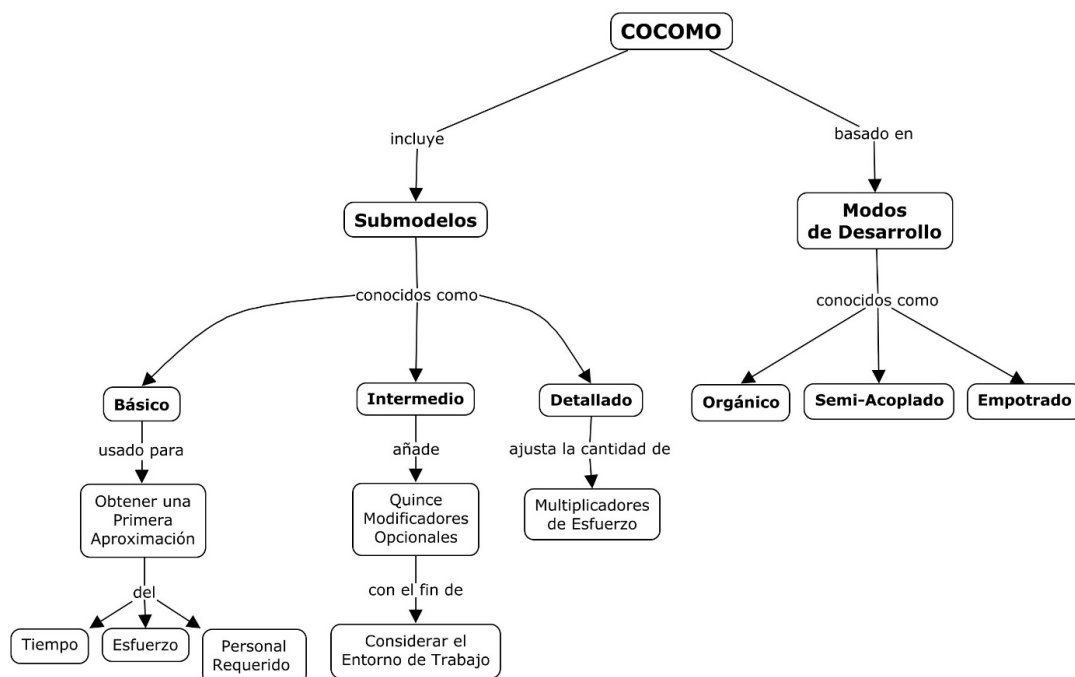
2.6. COSTOS

El objetivo de la estimación de costo de software consiste en estimar el tamaño, el esfuerzo, la complejidad y el costo del proyecto de software para poder encontrar la mejor decisión de desarrollo y asegurar que el gasto se encuentre de acuerdo con lo presupuestado. Por todo lo anterior, la estimación de costos de software es importante para la planificación, programación, presupuesto y establecimiento del precio indicado al desarrollo del software (Alarcón, 2023, p. 14).

2.6.1. Cocomo

(COCOMO, por su acrónimo del idioma inglés CONstructive COst MOdel) (Boehm, 1981). COCOMO es un modelo de formulación matemática con un fuerte componente de base empírica, principalmente utilizado para estimación de costos en los proyectos de software (Garita, 2014). En la Figura 6 se muestran los principales conceptos relacionados con el modelo COCOMO. Este modelo, propuesto por Barry W. Boehm, fue introducido a finales de los años 70 y comienzos de los 80 del siglo pasado en su trabajo, *Software Engineering Economics* (Boehm, 1981). Entre otras características, el modelo COCOMO está orientado a la magnitud del producto final, está basado en estimaciones matemáticas, mide el “tamaño” del proyecto y utiliza las líneas de código como unidad de medida. Dos de los aspectos fundamentales del modelo COCOMO son los submodelos y los modos de desarrollo.

Figura 6.
Conceptos de Cocomo



Nota: Conceptualización básica de COCOMO, *Fulvio, 2018*

Los submodelos son tres: básico, intermedio y detallado. Por su parte, los modos de desarrollo son también tres: orgánico, semi-acoplado y empotrado. En la Tabla 1 se muestra el esquema de modos de desarrollo de software con sus principales características que ayudan a elegir el tipo de modo de desarrollo para un proyecto en particular. En esta Tabla 2, en el caso del tamaño, se consideran las líneas de código fuente del software en unidades de miles de líneas de código (KDLC, por sus siglas en inglés).

Tabla 1.
Esquema de modos de desarrollo de software

Modo de desarrollo	de	Requisitos	Tamaño	Complejidad	Personas	Experiencia
Orgánico		Poco rígidos	Pequeño (<50KLDC)	Pequeña	Pocas	Mucha
Semiacoplado		Poco/medio	Medio (50 a 300KLDC)	Medio	Medio	Medio
Empotrado		Alto	Grande (>300KLDC)	Alta	Alta	Poca

2.6.1.1. Ecuaciones del Modelo COCOMO

El Estos modos de desarrollo permiten utilizar cuatro valores constantes. En la Tabla 2 se muestran los modos de desarrollo y los valores constantes respectivos. Estos valores constantes, codificados aquí como “a”, “b”, “c” y “d”, son propuestos por el modelo COCOMO para complementar las ecuaciones de cálculo usadas en el modelo. (Dialnet, 2021, p. 7).

Tabla 2.
Valores constantes por modo de desarrollo

Modo de desarrollo	COCOMO Básico	COCOMO Intermedio	b	c	d
	a	A			
Orgánico	2.4	3.2	1.0 5	2.50	0.3 8
Semiacoplado		3.0	1.1 2		0.3 5
Empotrado	3.6	2.8	1.2 0		0.3 2

Las ecuaciones incluidas son las utilizadas para los submodelos básico e intermedio. Estas ecuaciones se utilizan para calcular el esfuerzo nominal en personas/mes (E), tiempo estimado en meses (T) y personal requerido (P). No se incluyen las ecuaciones para el submodelo detallado, por razones de espacio dentro del desarrollo de la propuesta del enfoque pedagógico descrita en este trabajo. En la Tabla 4, se muestran las ecuaciones para esfuerzo

nominal en personas/mes (E), tiempo estimado en meses (T) y personal requerido (P) así como los multiplicadores de esfuerzo (ME), utilizados solo en la ecuación de esfuerzo del submodelo intermedio.

Tabla 3.
Ecuaciones por tipo de modelo COCOMO: Básico e intermedio

Ecuación	Submodelo básico	Submodelo intermedio
Esfuerzo (E)	$(E) = a * (KLDC)^b$	$(E) = a * (KLDC)^b * ME$
Tiempo (T)	$(T) = c * (E)^d$	$(T) = c * (E)^d$
Personal (P)	$(P) = E/T$	$(P) = E/T$

2.6.1.2. Modelos de COCOMO Intermedio

Los multiplicadores de esfuerzo, utilizados en la ecuación de esfuerzo del submodelo intermedio, son quince agrupados en cuatro grandes categorías: atributos de producto, atributos de computador, atributos personales y atributos del proyecto. En la Figura se muestran los multiplicadores de esfuerzo. Cada uno de estos multiplicadores de esfuerzo, tiene una valoración que se clasifica en una escala de 6 valores desde “muy bajo”, “bajo”, “nominal”, “alto”, “muy alto” y “extraordinariamente alto”. Estos multiplicadores de esfuerzo ajustan el valor real del esfuerzo. Dos ejemplos nos ayudan a entender el objeto de los multiplicadores de esfuerzo. Primero, en caso de que no se desee ajustar el esfuerzo nominal, se debe utilizar el valor nominal (es decir, el valor “1”) para cualquier multiplicador de esfuerzo. Segundo; si, por el contrario, se desea hacer algún ajuste en particular al esfuerzo nominal por razones de, en este caso, poca experiencia en la aplicación, se podría utilizar el multiplicador AEXP en valor nominal “muy bajo” (1,29).

Figura 7.
Ecuaciones por tipo de modelo COCOMO: Básico e intermedio

Multiplicadores de esfuerzo (ME)			Valoración					
			Muy bajo	Bajo	Nominal	Alto	Muy alto	Extr. alto
Atributos del producto								
1.	RELY	Fiabilidad requerida del software	0,75	0,88	1.00	1,15	1,40	
2.	DATA	Tamaño de la base de datos		0,94	1.00	1,08	1,16	
3.	CPLX	Complejidad del producto	0,70	0,85	1.00	1,15	1,30	1,65
Atributos de la computadora								
4.	TIME	Restricciones del tiempo de ejecución			1.00	1,11	1,30	1,66
5.	STOR	Restricciones del almacenamiento princ.			1.00	1,06	1,21	1,56
6.	VIRT	Inestabilidad de la máquina virtual		0,87	1.00	1,15	1,30	
7.	TURN	Tiempo de respuesta del computador		0,87	1.00	1,07	1,15	
Atributos del personal								
8.	ACAP	Capacidad del analista	1,46	1,19	1.00	0,86	0,71	
9.	AEXP	Experiencia en la aplicación	1,29	1,13	1.00	0,91	0,82	
10.	PCAP	Capacidad de los programadores	1,42	1,17	1.00	0,86	0,70	
11.	VEXP	Experiencia en S.O. utilizado	1,21	1,10	1.00	0,90		
12.	LEXP	Experiencia en el lenguaje de progr.	1,14	1,07	1.00	0,95		
Atributos del proyecto								
13.	MODP	Uso de prácticas de programación modernas	1,24	1,10	1.00	0,91	0,82	
14.	TOOL	Uso de herramientas software	1,24	1,10	1.00	0,91	0,83	
15.	SCED	Restricciones en la duración del proy.	1,23	1,08	1.00	1,04	1,10	

Nota: (Boehm, 1981).

(1) Ecuación Personas Por Mes Según Modelo De Cocomo Intermedio

$$E = a * KLDC^b * ME, \text{ en personas/mes} \quad (4)$$

Donde:

$$a = 3.2 \text{ Coeficiente Cocomo Intermedio}$$

$$b = 1.05$$

$$c = 2.5$$

$$d = 0.38$$

KLDC = líneas de código

(2) Ecuación Tiempo de Proyecto Según Modelo Intermedio

$$T = c(E)^d, \text{ en meses} \quad (5)$$

Donde:

(3) Ecuación Costo Personal

$$P = \frac{E}{T}, \text{ en personas} \quad (6)$$

Donde:

E = Esfuerzo

T = Tiempo requerido

P = Costo personal

(4) Ecuación de Costo Total del Proyecto

$$C_{sof} = S_{mes} * P * T \quad (7)$$

Donde:

Smes = Sueldo mes

P = Número de personas

T = Tiempo requerido

Csof = Costo total del proyecto

2.7. MÉTRICAS DE CALIDAD

Las Métricas de Calidad proporcionan una indicación de cómo se ajusta el software, a los requerimientos implícitos y explícitos del cliente (Abud, 2012, p.46).

Hablar de calidad del software implica la necesidad de contar con parámetros que permitan establecer los niveles mínimos que un producto de este tipo debe alcanzar para que se considere de calidad. El problema es que la mayoría de las características que definen al software no se pueden cuantificar fácilmente; generalmente, se establecen de forma cualitativa, lo que dificulta su medición, ya que se requiere establecer métricas que permitan evaluar cuantitativamente cada característica dependiendo del tipo de software que se pretende calificar. (Abud, 2012.p.44)

El objetivo principal de la ingeniería del software es producir un producto de alta calidad. Para lograr este objetivo, los ingenieros del software deben utilizar mediciones que evalúen la calidad del análisis y los modelos de desafío, el código fuente, y los casos de prueba que se han creado al aplicar la ingeniería del software. Para lograr esta evaluación de la calidad en tiempo real, el ingeniero debe utilizar medidas técnicas que evalúan la calidad con objetividad, no con subjetividad. (Abud, 2012, p.67)

En la mayoría de los desafíos técnicos, las métricas nos ayudan a entender tanto el proceso técnico que se utiliza para desarrollar un producto, como el propio producto. El principio, podría parecer que la necesidad de la medición es algo evidente después de todo lo que nos permite cuantificar y por consiguiente gestionar de forma más efectiva (Pavón, 2012, p.86).

- Calidad se basa en la satisfacción del cliente interno y externo, la totalidad de funciones y características de un bien o servicio que atañan a su capacidad para satisfacer necesidades expresas o implícitas

- Métricas hace referencia a la medición del software en base a parámetros predeterminados, como puede ser el número de líneas de código de que consta o el volumen de documentación asociada.

2.7.1. Norma ISO/IEC 25010

La familia de normas ISO/IEC 25010 ISO/IEC 25000, conocida como SQuaRE (System and Software Quality Requirements and Evaluation), es una familia de normas que tiene por objetivo la creación de un marco de trabajo común para evaluar la calidad del producto software. (ISO25000, 2020)

La ISO/IEC 25000 es la evolución de otras normas, especialmente de las normas ISO/IEC 9126, que describe las particularidades de un modelo de calidad del producto software, y la ISO/IEC 14598, que abordaba el proceso de evaluación de productos software. Esta familia de normas ISO/IEC 25000 se encuentra compuesta por cinco divisiones.

Figura 8.
ISO/IEC 25010

CALIDAD DEL PRODUCTO SOFTWARE								
ADECUACIÓN FUNCIONAL	EFICIENCIA DE DESEMPEÑO	COMPATIBILIDAD	CAPACIDAD DE INTERACCIÓN	FIABILIDAD	SEGURIDAD	MANTENIBILIDAD	FLEXIBILIDAD	PROTECCIÓN
COMPLETITUD FUNCIONAL	COMPORTAMIENTO TEMPORAL	COEXISTENCIA	RECONOCIBILIDAD DE ADECUACIÓN	AUSENCIA DE FALLOS	CONFIDENCIALIDAD	MODULARIDAD	ADAPTABILIDAD	RESTRICCIÓN OPERATIVA
CORRECCIÓN FUNCIONAL	UTILIZACIÓN DE RECURSOS	INTEROPERABILIDAD	APRENDIZABILIDAD	DISPONIBILIDAD	INTEGRIDAD	REUSABILIDAD	ESCALABILIDAD	IDENTIFICACIÓN DE RIESGOS
PERTINENCIA FUNCIONAL	CAPACIDAD		OPERABILIDAD	TOLERANCIA A FALLOS	NO-REPUDIO	ANALIZABILIDAD	INSTALABILIDAD	PROTECCIÓN ANTE FALLOS
			PROTECCIÓN FRENTE A ERRORES DE USUARIO	RECUPERABILIDAD	RESPONSABILIDAD	CAPACIDAD DE SER MODIFICADO	REEMPLAZABILIDAD	ADVERTENCIA DE PELIGRO
			INVOLUCRACIÓN DEL USUARIO		AUTENTICIDAD	CAPACIDAD DE SER PROBADO		INTEGRACIÓN SEGURA
			INCLUSIVIDAD		RESISTENCIA			
			ASISTENCIA AL USUARIO					
			AUTO-DESCRIPTIVIDAD					

Nota. ISO/IEC 25000, Pavon, 2020, Norma ISO/IEC 25000

2.7.1.1. ISO/IEC 25010

La calidad del producto software se puede interpretar como el grado en que dicho producto satisface los requisitos de sus usuarios aportando de esta manera un valor. Son

precisamente estos requisitos (funcionalidad, rendimiento, seguridad, mantenibilidad, etc.) los que se encuentran representados en el modelo de calidad, el cual categoriza la calidad del producto en características y sus características.

El modelo de calidad del producto definido por la ISO/IEC 25010 se encuentra compuesto por las nueve características de calidad que se muestran.

2.7.1.1.1. Adecuación Funcional

En este grupo se conjunta una serie de atributos que permiten calificar si un producto de software maneja en forma adecuada, el conjunto de funciones que satisfagan las necesidades para las cuales fue diseñado.

Para el cálculo funcional se debe determinar cinco características de dominios de información los valores de información se definen de la siguiente manera.

- Número de entradas de usuario. Se cuenta cada entrada de usuario que proporciona diferentes datos orientados a la aplicación. Las entradas se deberían diferenciar de las peticiones, las cuales se cuentan de forma separada.
- Número de salidas de usuario. Se cuenta cada salida que proporciona al usuario información orientada a la aplicación. En este contexto la salida se refiere a informes, pantallas, mensajes de error, etc.
- Número de peticiones de usuario. Una petición se define como una entrada interactiva que produce la generación de alguna respuesta del software inmediata en forma de salida interactiva. Se cuenta cada petición por separado.
- Número de archivos. Se cuenta cada archivo maestro lógico (esto es, un grupo lógico de datos que puede ser una parte de una gran base de datos o un archivo independiente).

- Número de interfaces externas. Se cuentan todas las interfaces legibles por la máquina (por ejemplo: archivos de datos de cinta o disco) que se utilizan para transmitir información a otro sistema.

Tabla 4.*Factor de Ponderación*

VALOR DOMINIO DE INFORMACIÓN	CONTEO	FACTOR DE PONDERACIÓN			TOTAL
		SIMPLE	PROMEDIO	COMPLEJO	
Entradas externas (EE)	x	3	4	6	X
Salidas externas (SE)	x	4	5	7	X
Consultas externas (CE)	x	3	4	6	X
Archivos logicos internos (ALI)	x	7	10	15	X
Archivos de interfaz externos (AIE)	x	5	7	10	
TOTAL, PFNA					x

Tabla 5.*Índices de Usabilidad*

Factores de complejidad	Sin influencia	Inicial	Moderado	Medio	Significativo	Esencial	FI
	0	1	2	3	4	5	
¿Requiere el sistema copias de seguridad y de recuperación fiables?							
¿Se requiere comunicación de datos?							
¿Existen unciones de procesamiento distribuido?							
¿Es crítico el rendimiento?							

¿Se ejecutará el sistema con un entorno operativo existente y fuertemente utilizado?						
¿Requiere el sistema entrada de datos interactiva?						
Facilidad Operativa						
¿Se actualiza los archivos maestros de forma interactiva?						
¿Son complejas las entradas, las salidas, los archivos o las peticiones?						
Procesamiento interno complejo						
Diseño de código reutilizable						
Facilidad de Instalación						
Facilidad de cambios						
¿Soporta múltiples instalaciones en diferentes sitios?						
Factor ajuste de complejidad						total

Fórmula para ajustar puntos de función, que considera factores como la complejidad técnica, el rendimiento y la usabilidad.

(5) Fórmula de Ajuste

$$PF = \text{conteo total} \left[0.65 + 0.01 * \sum fi \right] \quad (8)$$

Donde:

$\sum (fi)$: Sumatoria de los valores de los factores de ajuste.
 PF : Ajuste

Con el límite superior establecido en la suma de los factores de ajuste de acuerdo a los parámetros propuestos, se realiza el cálculo del valor siguiente en la secuencia, este proceso es fundamental para mantener la coherencia en la evaluación de la complejidad del sistema.

(6) Punto Función Máxima

$$PFmax = Cuenta Total \left[0.65 + (0.01 * \sum fi) \right] \quad (9)$$

Donde:

$\sum (fi)$: = Sumatoria de los valores de los factores de ajuste.

PFmax: = Punto función máxima.

Con estos factores podemos encontrar el valor de la funcionalidad.

(7) Formula de la Funcionalidad

$$Funcionalidad = \frac{PF}{PF max} * 100 \quad (10)$$

Donde:

PF : Punto función.

$PF max$: Punto función máxima.

2.7.1.1.2. Confiabilidad

La confiabilidad de un sistema se refiere a su capacidad para mantener un nivel de funcionamiento adecuado durante un período de prueba establecido que implica considerar aspectos como la madurez del sistema. Aplicaremos las fórmulas correspondientes, teniendo en cuenta las probabilidades de fallas y éxitos debido a varios factores.

- Comportamiento temporal. Grado en que un producto o realiza sus funciones de forma que el tiempo de respuesta y el ratio de rendimiento cumple los requisitos especificados.
- Utilización de recursos. Grado en que la cantidad y tipos de recursos utilizados por el producto al llevar a cabo su función bajo condiciones determinadas no exceden lo especificado.
- Capacidad. Grado en que el producto cumple los requisitos relativos a límites máximos para un parámetro (ítems almacenados, usuarios concurrentes, ancho de banda de comunicaciones).

Considerando que se tiene en cuenta que:

(8) Probabilidad de Fallas

$$P(T \leq t) \Rightarrow \text{Probabilidad de fallas} \quad (11)$$

Probabilidad de trabajo sin fallas

(9) Probabilidad de Trabajo Sin Fallas

$$P(T > t) = 1 - F(t) \Rightarrow \text{Probabilidad de trabajo sin fallas} \quad (12)$$

Se considera la siguiente función al calcular la confiabilidad del sistema.

(10) Función de la Confiabilidad del Sistema MT

$$F(t) = f * e^{-\mu * t} \quad (13)$$

Donde:

f	=	Funcionalidad del sistema
μ	=	Probabilidad de error del sistema
t	=	Tiempo de prueba del sistema
$F(t)$	=	Confiabilidad

2.7.1.1.3. Usabilidad

Al evaluar la usabilidad de un sistema, es importante considerar el factor humano, asegurando que el software cumpla con los requisitos y expectativas del usuario, por lo tanto, se lleva a cabo una evaluación mediante encuestas dirigidas a los usuarios del sistema.

Estas encuestas permiten recopilar información valiosa sobre la experiencia del usuario, su satisfacción y la facilidad de uso percibida que se utiliza luego para calcular la usabilidad del sistema, aplicando una fórmula diseñada específicamente para medir la eficacia y la experiencia del usuario en la interacción con el software.

(11) Función para Determinar la Usabilidad MT

$$FU = \left(\frac{\sum \text{valor}}{n} * 100 \right) / 5 \quad (14)$$

Donde:

FU	=	Función usabilidad
$\sum \text{valor}$	=	Sumatoria de los valores de la usabilidad del sistema
n	=	Cantidad de valores de la usabilidad del sistema

2.7.1.1.4. Eficiencia

Capacidad del producto software para que el usuario interactúe mediante su interfaz intercambiando información para completar determinadas tareas.

Para determinar la eficiencia, se emplea la fórmula siguiente.

(12) Fórmula de la Eficiencia MT

$$E = \frac{\sum x_i}{n} * \frac{100}{n} \quad (15)$$

Donde:

$\sum x_i$ = Sumatoria de los valores de eficiencia

n = Número de preguntas

E = Eficiencia

2.7.1.1.5. Mantenibilidad

Esta métrica evalúa la cantidad de trabajo requerido para realizar cambios en el sistema, ya sea para corregir errores o agregar nuevas funcionalidades. El estándar IEEE94 recomienda utilizar el índice de madurez del sistema como indicador de su estabilidad. Por lo tanto, la ecuación para este índice es la siguiente:

(13) Cálculo del Nivel de Madurez del Sistema

$$IMS = \frac{[Mt - (Fa + Fc + Fd)]}{Mt} \quad (16)$$

Donde:

Mt = Número de módulos total de la versión actual

Fa = Número de módulos de la versión actual que se añadieron.

Fc = Número de módulos de la versión actual que se cambiaron.

Fd = Número de módulos de la versión anterior que se eliminaron en la versión actual

IMS = Nivel de madurez del sistema

2.7.1.1.6. Portabilidad

Se evalúa la oportunidad para adaptar el software a diferentes ambientes sin necesidad de aplicarle modificaciones, es decir que el software se puede trasladar de un entorno a otro.

Para poder medir la portabilidad del sistema usaremos la siguiente formula que indica el grado de portabilidad que tiene un software.

$$GP = 1 - (ET/ER)$$

Donde:

ET: Es la medida de los recursos necesarios para llevar el sistema a otro entorno.

ER: Es la medida de los recursos necesarios para crear el sistema en el entorno residente.

Si $GP > 0$, la portabilidad es más rentable que el re-desarrollo

Si $GP = 1$, la portabilidad es perfecta

Si $GP < 0$, el re-desarrollo es más rentable que la portabilidad.navegador.

(14) Fórmula de la Portabilidad MT

$$Portabilidad = 1 - \left(\frac{\text{número de día para aportar el sistema}}{\text{número de días para implementar el sistema}} \right) \quad (17)$$

2.8. SEGURIDAD

La También ISOTools Excellence (2020), afirma que la norma ISO 27001 es una solución de mejora continua en base a la cual puede desarrollarse un Sistema de Gestión de Seguridad de la Información (SGSI) que permita evaluar todo tipo de riesgos o amenazas susceptibles de poner en peligro la información de una organización tanto propia como datos de terceros.

La norma ISO 27001 es una solución de mejora continua en base a la cual puede desarrollarse un Sistema de Gestión de Seguridad de la Información (SGSI) que Seguridad permita evaluar todo tipo de riesgos o amenazas susceptibles de poner en peligro la información de una organización tanto propia como datos de terceros. Por otro lado, también

permite establecer los controles y estrategias más adecuadas para eliminar o minimizar dichos peligros.

Es un estándar de buenas prácticas de seguridad informática que tiene las siguientes características:

- **Confidencialidad de datos:** Tiene como propósito garantizar la seguridad al momento de ingresar a la información, no permite divulgar la información a personas ajenas a la institución o empresa, con ello se busca conseguir una seguridad óptima que solo los usuarios registrados puedan ingresar.
- **Disponibilidad de datos:** tiene como propósito que los datos o información del sistema estén disponibles para los usuarios en lo posible las 24 horas del día
- **Integridad de datos:** Hace referencia a los datos que no pueden ser alterados por ningún tipo de personal. Teniendo como referencia al rol que pertenezca el usuario y la jerarquía en la institución.

Como beneficio de la ISO 27001 demuestra una garantía de los controles internos y que cumpla los requisitos de la seguridad, demuestra una ventaja competitiva al cumplir requisitos y demostrar a los usuarios la seguridad de la información es primordial.

2.9. PRUEBAS DE SOFTWARE

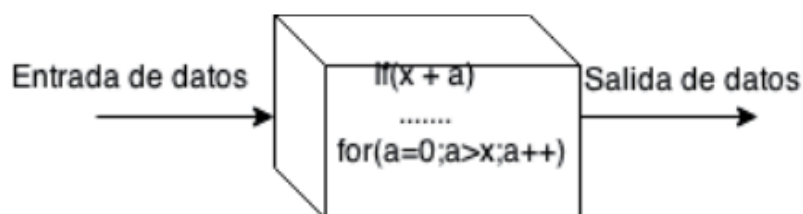
La meta de probar es encontrar errores, y una buena prueba es aquella que tiene una alta probabilidad de encontrar uno. Por tanto, un sistema basado en computadora o un producto debe diseñarse e implementarse teniendo en mente la "comprobabilidad". Al mismo tiempo, las pruebas en sí mismas deben mostrar un conjunto de características que logren la meta de encontrar la mayor cantidad de errores con el mínimo esfuerzo (Pressman, p. 412).

2.9.1. Pruebas de Caja Blanca

El También a veces definida como prueba de “caja de cristal” o “caja transparente”, es una técnica de diseño de casos de prueba que usa la estructura de control para obtener los casos de prueba.

Figura 9.

Pruebas de Caja Blanca



Nota. Fundamentos y Técnicos, Sánchez, 2015, Software de Caja Blanca

Dentro de esta estructura de control podemos encontrar la estructura de un componente, puede ser sentencias de decisiones, caminos distintos de código.

Se deriva casos de prueba que se:

Garantizan que todas las rutas del código se revisan al menos una vez

Revisa todas las condiciones lógicas

Ejecuta todos los bucles de sus fronteras y dentro de sus fronteras.

Revisan estructuras de datos

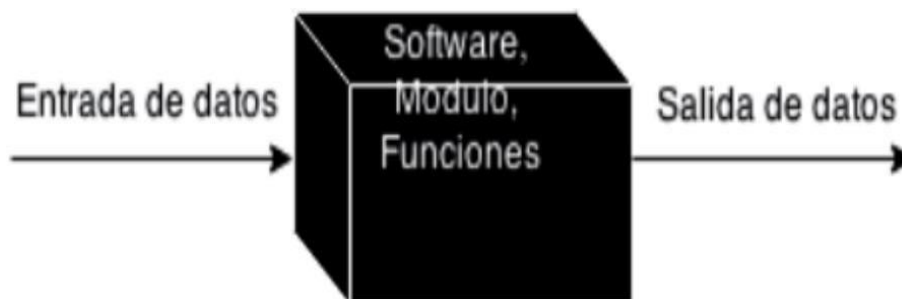
2.9.2. Pruebas de Caja Negra

Llamada pruebas de comportamiento. Son las que utilizan el análisis de especificación, tanto funcional como no funcional, sin tener en cuenta la estructura interna del programa para

diseñar los casos de pruebas. A diferencia de las pruebas de caja blanca, estas suelen realizarse durante las últimas etapas de las pruebas. (Maldonado 2021, p.86).

Figura 10.

Prueba de Caja Negra



Nota. *Fundamentos y Técnicos, Sánchez Peño, 2015, software de caja negra*

Las técnicas más comunes son:

Partición de equivalencia: Es un método de caja negra que divide en dominio en entrada de un programa en clase de datos que pueden derivarse casos de prueba. Se espera que el programa tenga un comportamiento común.

Según su técnica consta de dos pasos:

Identificar las clases de equivalencia

Identificar los casos de prueba

Análisis de valores límite: completan a la partición equivalente, se debe verificar y poner mucha atención en los límites deben estar correctamente definidos y programados. Esta técnica nos lleva a elegir los casos de prueba que ejercen los valores límites.

Transición de estado: Nos indica que todos los sistemas se mueven por transiciones de un paso a otro, se puede guiar de transiciones válidas e inválidas.

Tablas de decisión: Consideran que para encontrar el resultado esperado deben estar en conjunto varias condiciones. (Zaptest, 2024, p. 7).

2.9.3. Pruebas de Estrés

Las pruebas de estrés de software también conocidas como pruebas de resistencia, son utilizados durante la etapa de testing, con el objetivo de probar los límites de un sistema y prever escenarios de riesgo ante cargas extremas (Acosta, 2021, p.59).

Si el sistema no se encuentra en condiciones óptimas responderá con errores o comportamientos anormales, como fallos de código o bloqueos de información es por eso que es indispensable realizar las pruebas de estrés para evitar cuando ya se ponga en producción.

2.9.3.1. Tipos de prueba de estrés

De aplicación: Se efectuarán ante posibles errores asociados al bloqueo de información y fallas en la conexión.

Simétricas: Son aplicables cuando es necesario probar el funcionamiento de dos o más sistemas que operan bajo el mismo servidor.

Exploratorias: Se ejecutan cuando el sistema es susceptible a fallos en condiciones inesperadas, como la saturación de la plataforma.

2.9.4. Pruebas de Accesibilidad

Las "WAVE es una herramienta de evaluación desarrollada para ayudar a los diseñadores y desarrolladores web a identificar problemas de accesibilidad en sus sitios web, proporcionando retroalimentación visual directa sobre los elementos de accesibilidad y su cumplimiento con los estándares establecidos" (WebAIM, 2023).

CAPÍTULO III

MARCO

APLICATIVO

INGENIERÍA
DE SISTEMAS
UNIVERSIDAD PÚBLICA DE EL ALTO



3. CAPÍTULO III MARCO APLICATIVO

3.1. INTRODUCCIÓN

En este capítulo, se llevará a cabo la aplicación, análisis y diseño del Sistema Web para el Control y Seguimiento de Correspondencia del Gobierno Autónomo Municipal de Ayata, en el departamento de La Paz.

3.2. APLICACIÓN DE LA METODOLOGÍA OOHDM

La metodología OOHDM comprende cinco fases descritas a continuación.

Tabla 6.

Fases de la Metodología OOHDM

FASE	NOMBRE	PRODUCTOS	DESCRIPCIÓN
I	OBTENCIÓN DE REQUERIMIENTOS	Casos de uso	Se elabora un documento que describe los requerimientos funcionales y no funcionales del sistema, además de identificar los actores (usuarios o sistemas externos que interactúan con el sistema).
II	DISEÑO CONCEPTUAL	Clases, subsistemas, relaciones, atributos	Se modela la semántica del dominio de la aplicación.
III	DISEÑO NAVEGACIONAL	Nodos, enlaces, estructuras de acceso contextos navegacionales, transformaciones de navegación	Se toma en cuenta el perfil del usuario y las tareas que realizará. Se pone especial atención en la facilidad de uso. Se diseña la estructura de navegación de la aplicación.

IV	DISEÑO DE INTERFAZ ABSTRACTA	Elementos de la interfaz abstracta y reacciones a eventos externos	Se modelan los elementos visibles. Se describen los componentes para la interfaz de navegación. Se define la apariencia de los elementos de la interfaz.
V	IMPLEMENTACIÓN	Aplicación en funcionamiento	Se implementa el sistema y se lleva a cabo su puesta en producción.

3.2.1. FASE I: Obtención de Requerimientos

La fase de obtención de requerimientos es muy importante para el desarrollo del sistema, ya que en esta etapa se identifican y definen las funciones que el software debe cumplir para satisfacer las necesidades del Gobierno Autónomo Municipal de Ayata. En esta fase, se establecen las tareas principales que el sistema deberá realizar y las limitaciones que deberá respetar.

El análisis de los requerimientos se llevó a cabo a través de reuniones con el personal de la Alcaldía, lo que permitió obtener información detallada sobre los procesos actuales de gestión de correspondencia y las mejoras necesarias. Gracias a esta colaboración, los requerimientos reflejan con precisión las necesidades y expectativas reales de los usuarios a continuación veremos la actividad que se realizó:

Tabla 7.

Descripción del Proceso de Requerimiento

Tareas	Descripción
Entrevista	La entrevista se realizó con el personal del GAMA: Alcalde Secretaria Área técnica Área financiera Encargados de la unidad Remitentes

documentación	Se revisaron los documentos relacionados con el proceso de correspondencia que se manejan dentro de la institución, incluyendo el tipo de presentación, los registros y su seguimiento.
Observación	Se observó el proceso y manejo de la correspondencia por parte de la secretaria, los jefes de cada unidad, y la forma en que los remitentes hacen el seguimiento de la correspondencia, así como las funciones que desempeñan cada uno de los actores involucrados.

3.2.1.1. Análisis de Requerimientos

En esta etapa, se identifican los requerimientos funcionales y no funcionales necesarios para el desarrollo del sistema. Los requerimientos funcionales especifican las acciones que el sistema debe ser capaz de realizar para satisfacer las necesidades de los usuarios, mientras que los requerimientos no funcionales abordan aspectos como la usabilidad, rendimiento y seguridad del sistema. A continuación, se presenta una tabla con los requerimientos funcionales para cada tipo de usuario:

Tabla 6.

Requerimientos Funcionales

USUARIO	REQUERIMIENTOS FUNCIONALES
Administrador	Acceder a sistema
	Gestionar cargos
	Gestionar personal.
	Gestionar unidades.
	Gestionar usuarios.
	Gestionar tipo de documento.
	Consultar el estado de la correspondencia.
	Generar reportes.

Secretaria(o)	Acceder a sistema Gestionar correspondencia. Gestionar tipo de documentos. Consultar el estado de correspondencia. Generar reportes.
Responsable de Unidad	Acceder a sistema Recibir correspondencia física y confirmar en el sistema. Derivar correspondencia a otra unidad o responsable. Archivar/concluir correspondencia Consultar el estado de la correspondencia asignada a su unidad.
Remitente	Consultar el estado de su correspondencia

Además de los requerimientos funcionales, es importante considerar los requerimientos no funcionales que garantizan la calidad y el rendimiento del sistema. Estos aspectos son esenciales para asegurar que el sistema no solo cumpla con las funciones esperadas, sino que también ofrezca una experiencia de usuario satisfactoria y sea resistente ante diferentes condiciones de uso. A continuación, se presenta una tabla con los requerimientos no funcionales para cada tipo de usuario:

Tabla 8.

Requerimientos no Funcionales

Rol	- Requerimientos No Funcionales
Administrador	Seguridad en el acceso a datos sensibles. Disponibilidad del sistema 24/7. Respuesta rápida en la generación de reportes.



	Escalabilidad para soportar un número creciente de usuarios y datos.
Secretaria(o)	Interfaz de usuario intuitiva y fácil de usar. Accesibilidad desde dispositivos móviles. Tiempo de respuesta rápido en el registro y consulta de correspondencia
Responsable de Unidad	de Confiabilidad en la entrega y derivación de correspondencia. Protección de datos durante el manejo de la correspondencia. Disponibilidad de soporte técnico.
Remitente	Acceso seguro para verificar el estado de su correspondencia. Disponibilidad continua del sistema para consultas. Interfaz amigable y fácil de navegar.


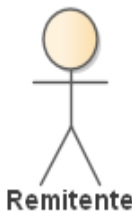
La tabla detalla los requerimientos no funcionales establecidos para el proyecto

3.2.1.1. Identificación de actores y tareas

Figura 11.

Descripción de Actores

Actor	Rol	Descripción:
 Administrador	Administrador	Es el usuario que tiene el control total del sistema. Administra todos los módulos y genera reportes sobre el funcionamiento y uso del sistema.
 Secretario(a)	Secretaria(o)	Se encarga de registrar la correspondencia, generar hojas de ruta, derivar documentos, recibir y hacer seguimiento de la correspondencia. También realiza búsquedas y obtiene reportes relacionados.

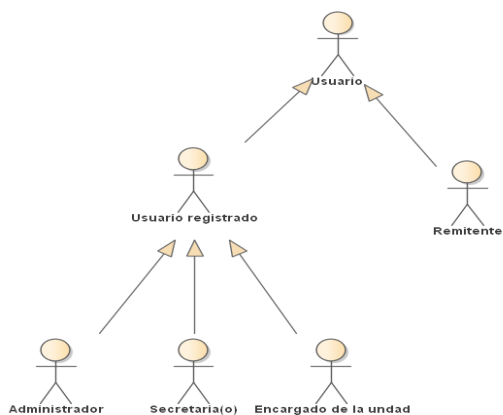
	Responsable de Unidad	Recibe la correspondencia de manera física y confirma su recepción en el sistema. Deriva a la unidad competente con un instructivo y sello físico, y registra o archiva la correspondencia en el sistema. También elabora respuestas según sea necesario.
	Remitente	Cualquier persona que envía notas, solicitudes, invitaciones, avisos o facturas. Los remitentes pueden ser personal de la Alcaldía, autoridades de comunidades y representantes de otras instituciones. Al dejar la correspondencia en la secretaria, se registra y se genera un número de hoja de ruta para su seguimiento. Puede consultar el estado de su correspondencia en cualquier momento ingresando el número de hoja de ruta en el sistema.

3.2.1.2. Relaciones de generalización entre actores

Los actores identificados están clasificados de acuerdo a su rol y autenticación en el sistema.

Figura 12.

Relación de Usuarios

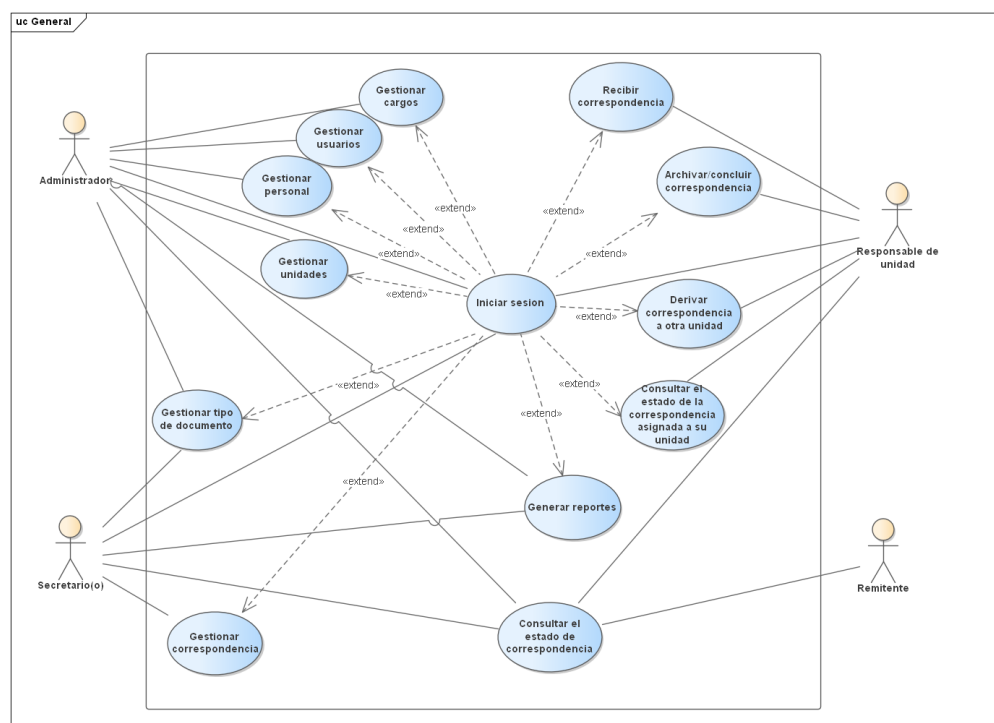


3.2.1.3. Especificaciones de Casos de Uso

En esta sección, se utilizará el modelo UML (Lenguaje de Modelado Unificado) para especificar los Casos de Uso del sistema. Los Casos de Uso describen cómo los diferentes actores interactúan con el sistema, detallando las funciones y procesos que se llevan a cabo. Esto ayuda a entender mejor los requerimientos del sistema y a asegurar que se cubran todas las necesidades de los usuarios.

Figura 13.

Caso de uso General del sistema



3.2.1.4. DESCRIPCIÓN DE LOS CASOS DE USO ESPECÍFICOS

Luego de determinar las funcionalidades generales a continuación, se describen las funcionalidades específicas descrita mediante los casos de uso.

Figura 14.

Casos de Uso secretaria

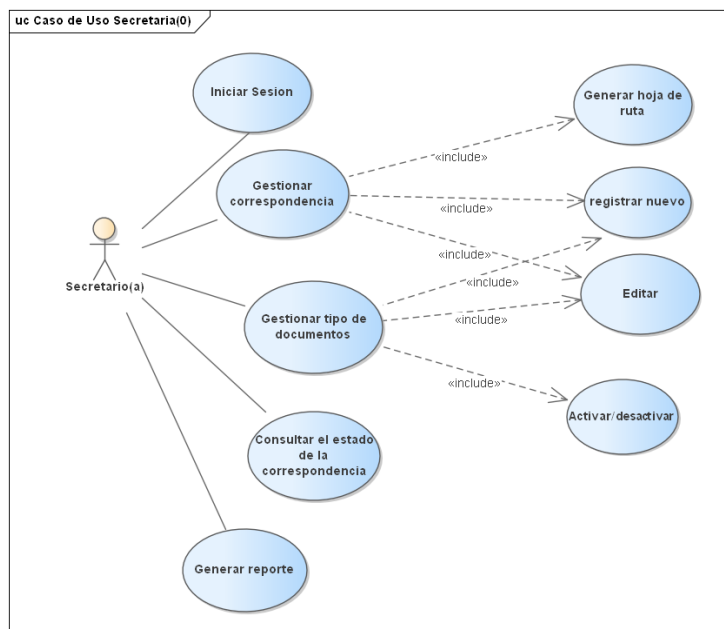


Tabla 9.

Detalles de Casos Uso - Iniciar Sesión

Caso de Uso	Iniciar Sesión
Actores	Secretaria
Propósito	Permitir que la secretaria acceda al sistema mediante sus credenciales.
Resumen	La secretaria introduce su usuario, contraseña y CAPTCHA para acceder al sistema.
Precondiciones	La secretaria debe tener un usuario y contraseña válidos registrados en el sistema.
Flujo Principal	<ol style="list-style-type: none"> 1. La secretaria introduce usuario, contraseña y CAPTCHA. 2. El sistema valida las credenciales. 3. Si las credenciales son correctas, la secretaria accede al sistema. 4. En caso de error, se contabiliza un intento fallido. 5. Con tres intentos fallidos, el sistema bloquea la cuenta temporalmente.
Postcondiciones	La secretaria puede ver y utilizar las funcionalidades del sistema tras el acceso exitoso.

Flujo Alternativo	A1: Restablecer Contraseña (en caso de olvido).
Excepciones	E1: Error de CAPTCHA.

Tabla 10.*Destalles de Caso de Uso - Consultar Estado de Correspondencia*

Caso de Uso	Consultar el Estado de la Correspondencia
Actores	Secretaria
Propósito	Permitir a la secretaria ver el estado de la correspondencia registrada.
Resumen	La secretaria consulta el estado de la correspondencia usando filtros como remitente, fecha, tipo de documento y numero de hoja de ruta.
Precondiciones	La secretaria debe haber iniciado sesión.
Flujo Principal	<ol style="list-style-type: none"> 1. La secretaria accede a la opción de consulta de estado. 2. Introduce los criterios de búsqueda (número de hoja de ruta). 3. El sistema muestra el estado y ubicación de la correspondencia según.
Postcondiciones	La secretaria visualiza el estado actual de la correspondencia.
Excepciones	E1: No se encuentra.

Tabla 11.*Detalles de Caso de Uso - Gestionar Tipo de Documento*

Caso de Uso	Gestionar Tipo de Documento
Actores	Secretaria
Propósito	Administrar los tipos de documentos en el sistema.
Resumen	La secretaria puede agregar, editar o activar/desactivar tipos de documentos.
Precondiciones	La secretaria debe haber iniciado sesión y tener permisos para modificar tipos de documentos.
Flujo Principal	<ol style="list-style-type: none"> 1. La secretaria selecciona la opción para gestionar tipos de documentos. 2. Puede agregar un nuevo tipo, editar uno existente o activar/desactivar uno. 3. El sistema guarda los cambios realizados.
Postcondiciones	Los tipos de documentos se actualizan en el sistema.

Flujo Alternativo	A1: Intento de editar un tipo de documento no permitido.
Excepciones	E1: Error de validación de datos.
Tabla 12.	
<i>Detalles de Caso de Uso - Generar Reportes</i>	
Caso de Uso	Generar Reportes
Actores	Secretaria
Propósito	Permitir la generación de reportes sobre la correspondencia gestionada.
Resumen	La secretaria selecciona criterios para generar reportes específicos sobre la correspondencia.
Precondiciones	La secretaria debe haber iniciado sesión.
Flujo Principal	<ol style="list-style-type: none"> 1. La secretaria selecciona la opción de generación de reportes. 2. Elige entre generar un reporte de todos los registros o filtrar por fecha de inicio y fecha de fin. 3. Selecciona el formato del reporte (PDF, Excel o impresión directa). 4. El sistema genera el reporte en el formato y criterio seleccionados.
Postcondiciones	El reporte es generado y queda disponible para la secretaria en el formato seleccionado.
Excepciones	E1: No se encuentran datos para el reporte solicitado.

Tabla 13.*Detalles de Caso Uso - Correspondencia*

Caso de Uso	Gestionar Correspondencia
Actores	Secretaria
Propósito	Registrar y gestionar la correspondencia recibida en el sistema, ingresando los datos del remitente como nombre completo, apellidos, cargo, número de celular, fecha del documento, asunto, número de fojas, área de destino y tipo de documento. Además, generar automáticamente una hoja de ruta con un número único para el seguimiento y localización de la correspondencia.
Resumen	La secretaria registra en el sistema la correspondencia de los remitentes, genera automáticamente hojas de ruta con números

	específicos para su seguimiento, y tiene la opción de editar la información de correspondencia y registrada.
Precondiciones	<p>1. La secretaria debe estar previamente registrada en el sistema por el administrador con todos sus datos completos. 2. La secretaria debe tener asignada un área específica, así como un usuario y contraseña para acceder al sistema.</p> <p>2. La secretaria debe tener asignada un área específica, así como un usuario y contraseña para acceder al sistema.</p>
Flujo Principal	<p>1. La secretaria ingresa al sistema introduciendo su usuario, contraseña y un código CAPTCHA.</p> <p>2. Si la secretaria introduce un usuario correcto y se equivoca en la contraseña, el sistema contabiliza el intento fallido.</p> <p>- Regla de intentos fallidos: Se permiten hasta tres intentos fallidos. Si se alcanza el límite, la cuenta se bloquea temporalmente.</p> <p>3. Si la secretaria introduce incorrectamente tanto el usuario como la contraseña, el sistema muestra una alerta indicando que el usuario no está registrado en la base de datos, sin contabilizar el intento como fallido.</p> <p>4. En caso de haber olvidado la contraseña, la secretaria puede seleccionar la opción "¿Desea restablecer contraseña?"</p> <p>- Recuperación de contraseña:</p> <p>- La secretaria introduce su correo electrónico registrado en el sistema.</p> <p>- El sistema envía un correo con un enlace para restablecer la contraseña.</p> <p>- La secretaria sigue el enlace y establece una nueva contraseña.</p> <p>5. Si el usuario, contraseña y CAPTCHA son correctos, la secretaria accede al sistema y puede comenzar a gestionar la correspondencia.</p>
Postcondiciones	<p>- Si se realiza un intento fallido de contraseña, se registra en el sistema.</p> <p>- Si se alcanza el límite de intentos fallidos, la cuenta de la secretaria se bloquea temporalmente.</p>

	<ul style="list-style-type: none">- Al ingresar exitosamente, la secretaria puede ver el módulo de correspondencia y realizar registros o ediciones según lo requiera.
Flujo Alternativo	<ul style="list-style-type: none">- A1: Intento fallido de inicio de sesión- Si el usuario es correcto pero la contraseña es incorrecta, se muestra un mensaje de error y se contabiliza el intento fallido.- Si el número de intentos fallidos llega a tres, se bloquea el acceso a la cuenta y se muestra un mensaje indicando que debe recuperarse la contraseña.- A2: Recuperación de contraseña- Si la secretaria selecciona "¿Desea restablecer contraseña?", se le solicita que introduzca su correo electrónico.- El sistema verifica el correo y, si es válido, envía una nueva contraseña temporal para que la secretaria pueda restablecer su acceso.
Excepciones	<ul style="list-style-type: none">- E1: Error de CAPTCHA- Si la secretaria introduce el CAPTCHA incorrectamente, el sistema muestra un mensaje de error y permite un nuevo intento.

Figura 15.

Caso de Uso Administrador

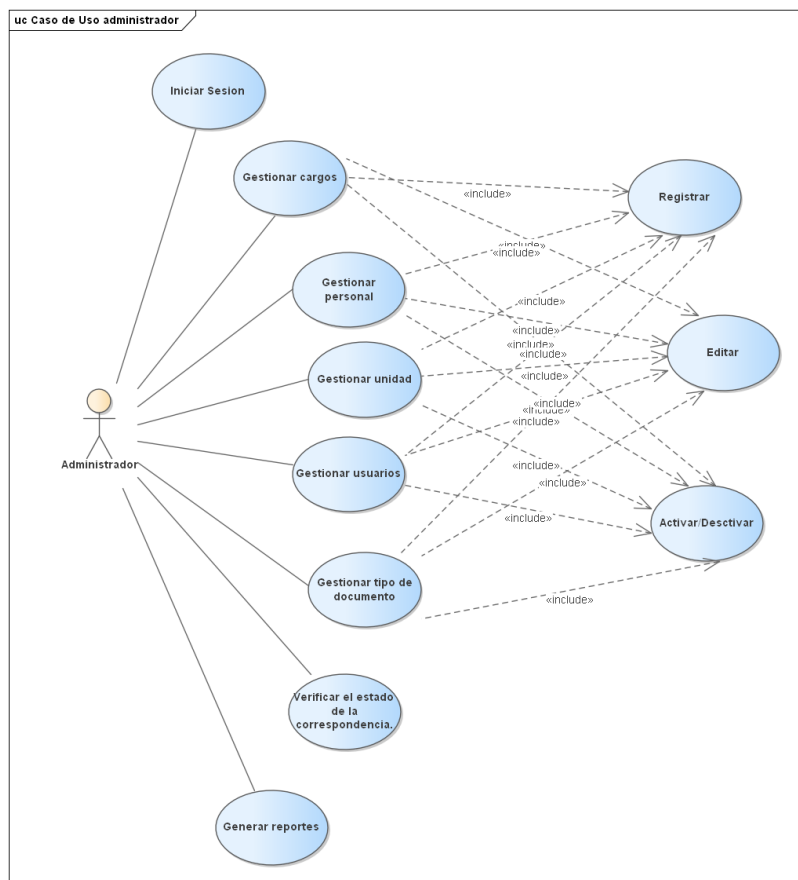


Tabla 14.

Detalles Caso Uso - Iniciar Sesión

Caso de Uso	Iniciar Sesión
Actor	Administrador
Precondición	El Administrador debe estar registrado en el sistema con un correo válido.
Flujo Básico	El Administrador introduce su número de usuario, contraseña y CAPTCHA. Si el número de usuario es incorrecto, el sistema muestra el mensaje "Usuario no existe en la base de datos". Si la contraseña es incorrecta, el sistema contabiliza un intento fallido, y si se alcanzan tres intentos fallidos, el usuario es bloqueado temporalmente. Si el CAPTCHA es incorrecto, el sistema muestra un mensaje de error, pero no contabiliza los intentos fallidos. Si el Administrador olvida su contraseña, puede hacer clic en "Olvidé mi contraseña", y el sistema le pedirá su

	correo electrónico para enviarle un enlace de restablecimiento de contraseña.
Flujos Alternativos	Si el número de usuario o la contraseña son incorrectos, se muestra un mensaje de error. Si se alcanza el máximo de intentos fallidos (3), el sistema bloquea al Administrador temporalmente.
Postcondición	El Administrador accede al sistema y se muestra el panel de control si las credenciales son correctas. Si el Administrador fue bloqueado, no podrá acceder hasta que se desbloquee.

Tabla 15.*Detalles de Caso de Uso - Gestionar Cargos*

Caso de Uso	Gestionar Cargos
Actor	Administrador
Precondición	El Administrador debe estar logueado en el sistema.
Flujo Básico	El Administrador puede registrar, editar, activar o desactivar cargos.
Flujos Alternativos	Si se intenta registrar un cargo con el mismo nombre, el sistema muestra un mensaje de error.
Postcondición	El cargo es gestionado correctamente (registrado, editado, activado o desactivado).
Include	Registrar Nuevo Cargo, Editar Cargo, Activar/Desactivar Cargo

Tabla 16.*Detalles de Caso de Uso - Gestionar Personal*

Caso de Uso	Gestionar Personal
Actor	Administrador
Precondición	El Administrador debe estar logueado en el sistema.
Flujo Básico	El Administrador puede registrar, editar, activar o desactivar personal.
Flujos Alternativos	Si se intenta registrar un empleado con datos duplicados, el sistema muestra un error.
Postcondición	El personal es gestionado correctamente (registrado, editado, activado o desactivado).
Include	Registrar Nuevo Personal, Editar Personal, Activar/Desactivar Personal

Tabla 17.*Detalles de Caso de Uso - Gestionar Unidad*

Caso de Uso	Gestionar Unidad
Actor	Administrador
Precondición	El Administrador debe estar logueado en el sistema.
Flujo Básico	El Administrador puede registrar, editar, activar o desactivar unidades.
Flujos Alternativos	Si se intenta registrar una unidad con el mismo nombre, el sistema muestra un mensaje de error.
Postcondición	La unidad es gestionada correctamente (registrada, editada, activada o desactivada).
Include	Registrar Nueva Unidad, Editar Unidad, Activar/Desactivar Unidad

Tabla 18.*Detalles Caso de Uso - Gestionar Usuarios*

Caso de Uso	Gestionar Usuarios
	Administrador
Precondición	El Administrador debe estar logueado en el sistema.
Flujo Básico	El Administrador puede registrar, editar, activar o desactivar usuarios.
Flujos Alternativos	Si se intenta registrar un usuario con el mismo correo o nombre de usuario, el sistema muestra un error.
Postcondición	El usuario es gestionado correctamente (registrado, editado, activado o desactivado).
Include	Registrar Nuevo Usuario, Editar Usuario, Activar/Desactivar Usuario

Tabla 19.*Caso de Uso – Gestionar Tipo de Documento*

Caso de Uso	Gestionar Tipo de Documento
Actor	Administrador
Precondición	El Administrador debe estar logueado en el sistema.
Flujo Básico	El Administrador puede registrar, editar, activar o desactivar tipos de documento.
Flujos Alternativos	Si se intenta registrar un tipo de documento con el mismo nombre, el sistema muestra un error.
Postcondición	El tipo de documento es gestionado correctamente (registrado, editado, activado o desactivado).
Include	Registrar Nuevo Tipo de Documento, Editar Tipo de Documento, Activar/Desactivar Tipo de Documento

Tabla 20.*Detalles Caso de Uso – Verificar Estado de Correspondencia*

Caso de Uso	Verificar Estado de Correspondencia
Actor	Administrador
Precondición	El Administrador debe estar logueado en el sistema.
Flujo Básico	El Administrador accede al módulo de correspondencia y puede consultar el estado de la correspondencia por número de hoja de ruta o por remitente.
Flujos Alternativos	Si la correspondencia no está registrada, el sistema muestra un mensaje de error.
Postcondición	El Administrador puede verificar el estado actual de la correspondencia.

Tabla 21.*Detalles de Caso de Uso - Generar Reportes*

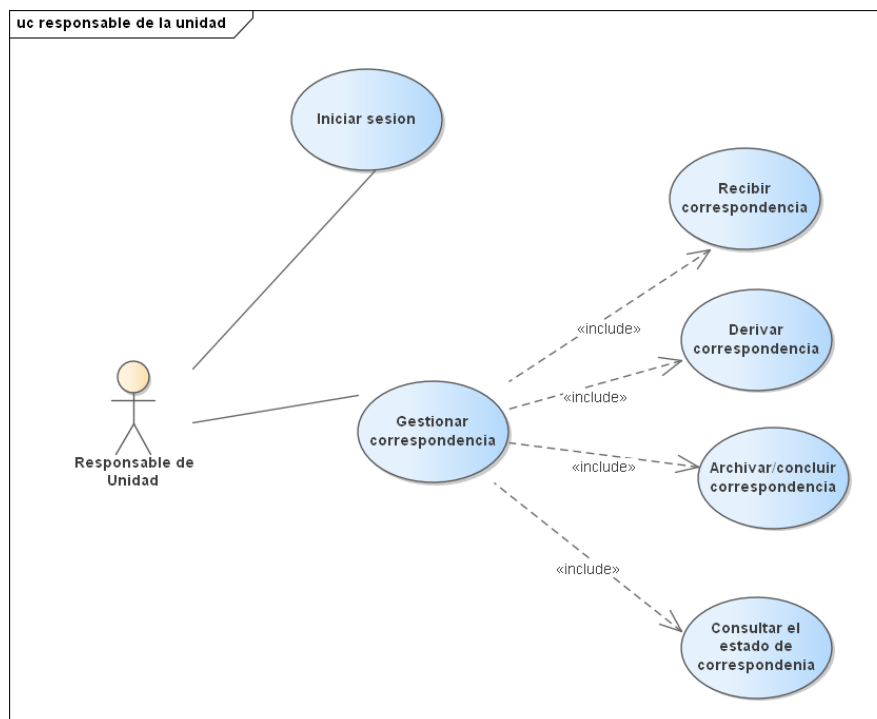
Caso de Uso	Generar Reportes
Actor	Administrador
Precondición	El Administrador debe estar logueado en el sistema.
Flujo Básico	El Administrador accede al módulo de generación de reportes, selecciona el tipo de reporte (por área, estado, etc.), y genera el reporte en formato PDF o Excel.
Flujos Alternativos	Si no hay datos que coincidan con los filtros seleccionados, el sistema muestra un mensaje de advertencia.
Postcondición	El reporte es generado y mostrado en pantalla o enviado para descarga.

Tabla 22.*Detalles de Caso de Uso - Buscar Correspondencia*

Caso de Uso	Buscar Correspondencia
Actor	Cualquier usuario (Administrador, jefe de Unidad, Remitente)
Precondición	No es necesario estar logueado en el sistema.
Flujo Básico	El usuario ingresa el número de la hoja de ruta en el módulo de búsqueda. El sistema muestra la información relacionada con ese número de hoja de ruta.
Flujos Alternativos	Si no se encuentra la hoja de ruta, el sistema muestra un mensaje de error indicando que no se encontró la correspondencia.
Postcondición	El usuario visualiza la correspondencia asociada con el número de hoja de ruta proporcionado.

Figura 16.

Diagrama de Caso de Uso - Responsable de la Unidad

**Tabla 23.**

Detalles Caso de Uso - Iniciar Sesión

Caso de Uso	Iniciar Sesión
Actor	Jefe de Unidad
Precondición	El jefe de Unidad debe estar registrado en el sistema con un correo válido.
Flujo Básico	El jefe de Unidad introduce su número de usuario, contraseña y CAPTCHA. Si el número de usuario es incorrecto, el sistema muestra el mensaje "Usuario no existe en la base de datos". Si la contraseña es incorrecta, el sistema contabiliza un intento fallido, y si se alcanzan tres intentos fallidos, el usuario es bloqueado temporalmente. Si el CAPTCHA es incorrecto, el sistema muestra un mensaje de error, pero no contabiliza los intentos fallidos. Si el jefe de Unidad olvida su contraseña, puede hacer clic en "Olvidé mi contraseña", y el sistema le pedirá su correo electrónico para enviarle un enlace de restablecimiento de contraseña.

Flujos Alternativos	Si el número de usuario o la contraseña son incorrectos, se muestra un mensaje de error. Si se alcanza el máximo de intentos fallidos (3), el sistema bloquea al jefe de Unidad temporalmente.
Pos condición	El jefe de Unidad accede al sistema y se muestra el panel de control si las credenciales son correctas. Si el jefe de Unidad fue bloqueado, no podrá acceder hasta que se desbloquee.

Tabla 24.

Detalles Caso de Uso - Gestionar Correspondencia

Caso de Uso	Gestionar Correspondencia
Actor	Jefe de Unidad
Precondición	El jefe de Unidad debe estar loqueado en el sistema.
Flujo Básico	El jefe de Unidad puede recibir, derivar, archivar/concluir y consultar el estado de la correspondencia.
Flujos Alternativos	Si la correspondencia no está registrada, el sistema muestra un mensaje de error.
Postcondición	La correspondencia es gestionada correctamente según la acción realizada (recibida, derivada, archivada/concluida o consultada).
Include	Recibir Correspondencia, Derivar Correspondencia, Archivar/Concluir Correspondencia, Consultar Estado de Correspondencia

Figura 17.

Caso de Uso - Seguimiento

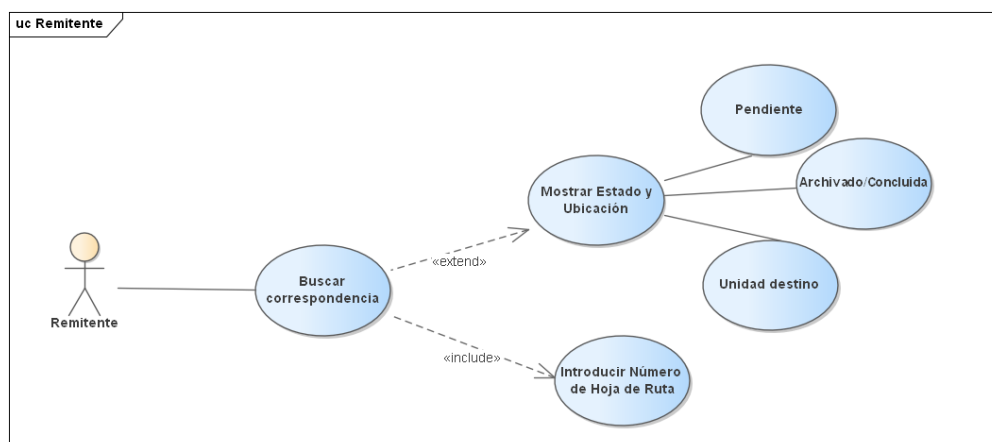


Tabla 25.

Detalles Caso de Uso - Buscar Remitente

Caso de Uso	Buscar Correspondencia
Actores	Remitente, Admirador, Jefe de la unidad, Secretaria
Precondición	no necesita estar logueado en el sistema para buscar la correspondencia. Solo necesita conocer el número de hoja de ruta.
Flujo Básico	ingresa el número de hoja de ruta de la correspondencia que desea consultar. El sistema muestra el estado actual de la correspondencia (Pendiente, Archivado/Concluido) y su ubicación o área asociada.
Flujos Alternativos	Si el número de hoja de ruta ingresado es incorrecto o no existe, el sistema muestra un mensaje de error informando que no se encuentra la correspondencia.
Postcondición	El Remitente visualiza el estado (Pendiente, Archivado/Concluido) y la ubicación del área asociada a la correspondencia consultada.
Include	Introducir número de hoja de ruta
Extend	Mostrar estado de correspondencia (Pendiente, Archivado/Concluido) y área de ubicación

3.2.2. FASE II: Diseño Conceptual

El modelo conceptual está compuesto por clases, atributos, operaciones y relaciones entre las entidades identificadas en la aplicación.

Figura 18.

Modelo Conceptual

s Modelo Conceptual

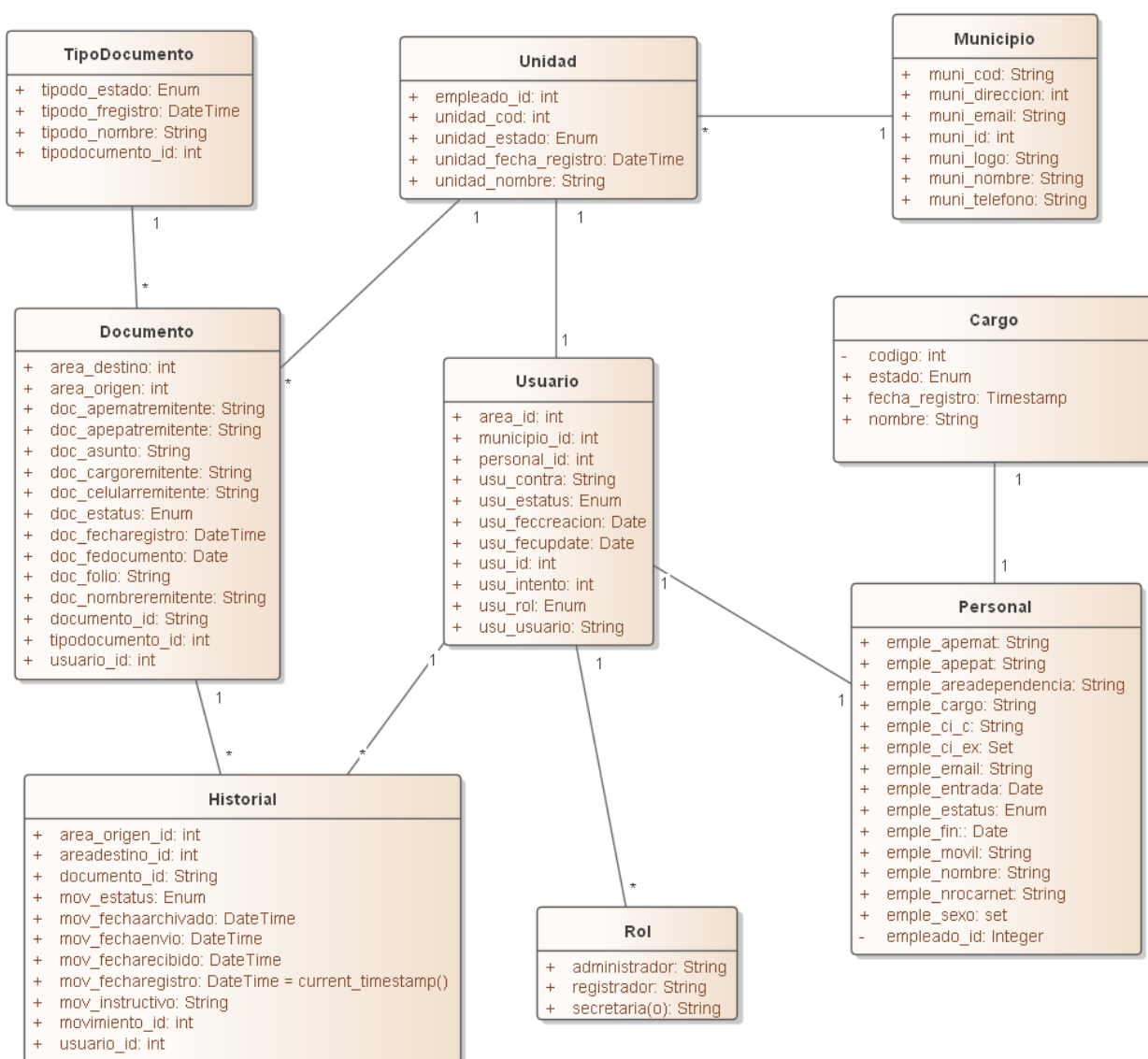


Figura 19.

Diseño de Base de Datos

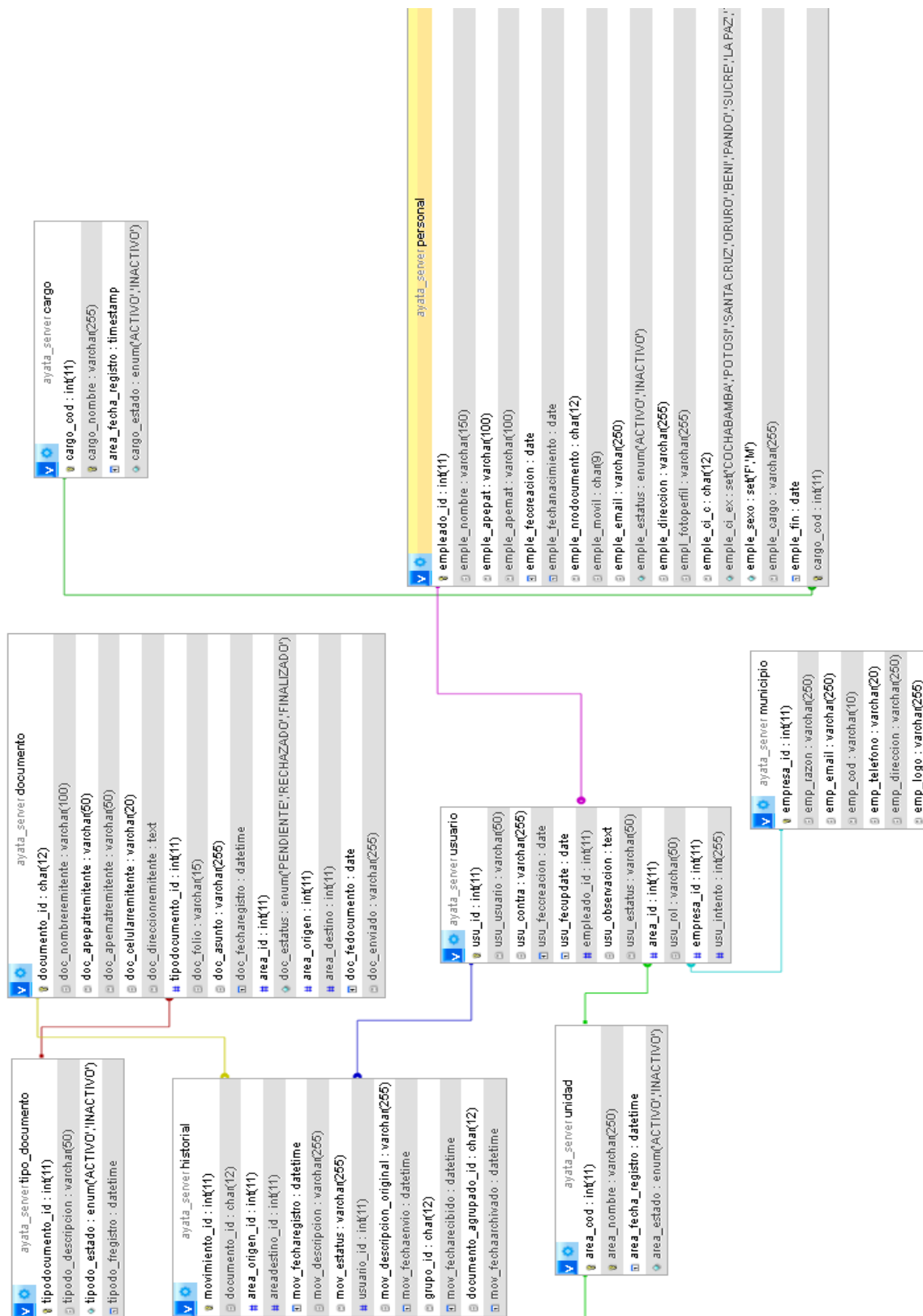


Tabla 26.*Descripción de Tabla Cargo*

Columna	Tipo	Comentarios
cargo_cod	int(11)	Código único del cargo, clave primaria de la tabla.
cargo_nombre	varchar(255)	Nombre del cargo, permite hasta 255 caracteres.
area_fecha_registro	timestamp	Fecha y hora de registro del cargo; se genera automáticamente en el momento de inserción.
cargo_estado	enum('ACTIVO', 'INACTIVO')	Estado del cargo, indica si está activo o inactivo en el sistema.

Tabla 27.*Descripción Tabla Documento*

Columna	Tipo	Comentarios
documento_id	char(12)	Identificador único del documento, compuesto de hasta 12 caracteres.
doc_nombre_remitente	varchar(100)	Nombre del remitente del documento, hasta 100 caracteres.
doc_apepatr_emitente	varchar(50)	Apellido paterno del remitente del documento, hasta 50 caracteres.
doc_apemat_remitente	varchar(50)	Apellido materno del remitente del documento, hasta 50 caracteres.
doc_celularr_emitente	varchar(9)	Número de celular del remitente, hasta 9 caracteres.
doc_direccion_remitente	text	Dirección completa del remitente.
tipodocumento_id	int(11)	Identificador del tipo de documento, clave foránea que se relaciona con la tabla de tipos.
doc_folio	varchar(15)	Número de folio del documento, hasta 15 caracteres.

doc_asunto	varchar(255)	Asunto o descripción breve del documento, hasta 255 caracteres.
doc_fecha_registro	datetime	Fecha y hora en que se registró el documento.
area_id	int(11)	Identificador del área responsable del documento, clave foránea que se relaciona con la tabla de áreas.
doc_estatus	enum('PENDIENTE','FINALIZADO')	Estado actual del documento: pendiente, finalizado.
area_origen	int(11)	Área de origen del documento, clave foránea.
area_destino	int(11)	Área de destino del documento, clave foránea.
doc_fecha_documento	date	Fecha del documento (la fecha específica mencionada en el documento).
doc_enviado	varchar(255)	Estado o información del envío del documento, hasta 255 caracteres.

Tabla 28.*Descripción Tabla Historial*

Columna	Tipo	Comentarios
movimiento_id	int(11)	Identificador único del movimiento, clave primaria de la tabla.
documento_id	char(12)	Identificador del documento asociado al movimiento, clave foránea relacionada con la tabla de documentos.
area_origen_id	int(11)	Identificador del área de origen del movimiento, clave foránea que se relaciona con la tabla de áreas.

area_destino_id	int(11)	Identificador del área de destino del movimiento, clave foránea relacionada con la tabla de áreas.
mov_fecha_registro	datetime	Fecha y hora en que se registró el movimiento.
mov_descripcion	varchar(255)	Descripción del movimiento, hasta 255 caracteres.
mov_estatus	varchar(255)	Estado del movimiento, hasta 255 caracteres.
usuario_id	int(11)	Identificador del usuario que realizó el movimiento, clave foránea relacionada con la tabla de usuarios.
mov_descripcion_original	varchar(255)	Descripción original del movimiento, hasta 255 caracteres.
mov_fechaenvio	datetime	Fecha y hora en que se envió el movimiento.
grupo_id	char(12)	Identificador del grupo al que pertenece el movimiento, clave foránea relacionada con la tabla de grupos.
mov_fecha recibido	datetime	Fecha y hora en que el movimiento fue recibido.
documento_agrupado_id	char(12)	Identificador de un documento agrupado, clave foránea para vincular documentos relacionados.
mov_fechaarchivado	datetime	Fecha y hora en que el movimiento fue archivado.

Tabla 29.*Descripción Tabla Personal*

Columna	Tipo	Comentarios
empleado_id	int(11)	Identificador único del empleado, clave primaria de la tabla.
emple_nombre	varchar(150)	Nombre completo del empleado, hasta 150 caracteres.
emple_apepat	varchar(100)	Apellido paterno del empleado, hasta 100 caracteres.

emple_apemat	varchar(100)	Apellido materno del empleado, hasta 100 caracteres.
emple_fecreacion	date	Fecha en que se creó el registro del empleado.
emple_fechanacimiento	date	Fecha de nacimiento del empleado.
emple_nrodocumento	char(12)	Número de documento de identidad del empleado, hasta 12 caracteres.
emple_movil	char(9)	Número de teléfono móvil del empleado, hasta 9 caracteres.
emple_email	varchar(250)	Dirección de correo electrónico del empleado, hasta 250 caracteres.
emple_estatus	enum('ACTIVO', 'INACTIVO')	Estado actual del empleado en el sistema, activo o inactivo.
emple_direccion	varchar(255)	Dirección residencial del empleado, hasta 255 caracteres.
empl_fotoperfil	varchar(255)	URL o ruta de la foto de perfil del empleado, hasta 255 caracteres.
emple_ci_c	char(12)	Número de cédula de identidad del empleado, hasta 12 caracteres.
emple_ci_ex	set('COCHABAMBA', 'POTOSI', 'SANTA CRUZ', 'ORURO', 'BENI', 'PANDO', 'SUCRE', 'LA PAZ', 'TARIJA')	Departamento de emisión de la cédula de identidad.
emplesexo	set('F', 'M')	Sexo del empleado: F para femenino, M para masculino.
emple_cargo	varchar(255)	Cargo o puesto del empleado en la organización, hasta 255 caracteres.
emple_fin	date	Fecha en que el empleado terminó su relación con la empresa, si aplica.
cargo_cod	int(11)	Código del cargo, clave foránea relacionada con la tabla de cargos.

Tabla 30.*Descripción Tabla Tipo Documento*

Columna	Tipo	Comentarios
tipodocumento_id	int(11)	Código auto incrementado del tipo de documento, clave primaria.
tipodo_descripcion	varchar(50)	Descripción del tipo de documento, hasta 50 caracteres.
tipodo_estado	enum('ACTIVO', 'INACTIVO')	Estado actual del tipo de documento: activo o inactivo en el sistema.
tipodo_registro	datetime	Fecha y hora en que se registró el tipo de documento.

Tabla 31.*Descripción Tabla Unidad*

Columna	Tipo	Comentarios
area_cod	int(11)	Código único del área, clave primaria de la tabla.
area_nombre	varchar(250)	Nombre del área, permite hasta 250 caracteres.
area_fecha_registro	datetime	Fecha y hora en que el área fue registrada en el sistema.
area_estado	enum('ACTIVO', 'INACTIVO')	Estado actual del área: indica si está activa o inactiva en el sistema.

Tabla 32.*Descripción Tabla Usuario*

Columna	Tipo	Comentarios
usu_id	int(11)	Identificador único del usuario, clave primaria de la tabla.





usu_usuario	varchar(50)	Nombre de usuario utilizado para iniciar sesión, hasta 50 caracteres.
usu_contra	varchar(255)	Contraseña del usuario, en formato encriptado, hasta 255 caracteres.
usu_feccreacion	date	Fecha en que se creó el registro del usuario.
usu_fecupdate	date	Fecha de la última actualización del registro del usuario.
empleado_id	int(11)	Identificador del empleado asociado, clave foránea de la tabla de empleados.
usu_observacion	text	Observaciones o notas adicionales sobre el usuario.
usu_estatus	varchar(50)	Estado del usuario, que puede indicar su disponibilidad o situación en el sistema, hasta 50 caracteres.
area_id	int(11)	Identificador del área en la que el usuario está asignado, clave foránea de la tabla de áreas.
usu_rol	varchar(50)	Rol del usuario en el sistema (administrador, secretaria y Registrador), hasta 50 caracteres.
muni_id	int(11)	Identificador de la empresa a la que el usuario pertenece, clave foránea de la tabla de municipio.
usu_intento	int(255)	Número de intentos fallidos de inicio de sesión del usuario, útil para controlar bloqueos.

3.2.3. FASE III: Diseño navegacional

El diseño navegacional se realizó utilizando la herramienta Enterprise Architect y el diagrama de modelado web. Las clases del modelo conceptual se transformaron en objetos de navegación, mientras que los escenarios se convirtieron en contextos de navegación. Los objetos de navegación identificados para el diseño de clases y contextos navegacionales son:

Tabla 33.

Tabla de Iconos

Simbología	Estereotipo	Significado
	Página (lado cliente)	Entorno visual al cual puede acceder el cliente.
	Página (lado servidor)	Proceso de una página del servidor lado del servidor.
	Conjunto de marcos	Opciones seleccionables o visualizables con el propósito de organizarlos en un orden específico.
	Formulario	Formulario de datos que se encuentra en una página.

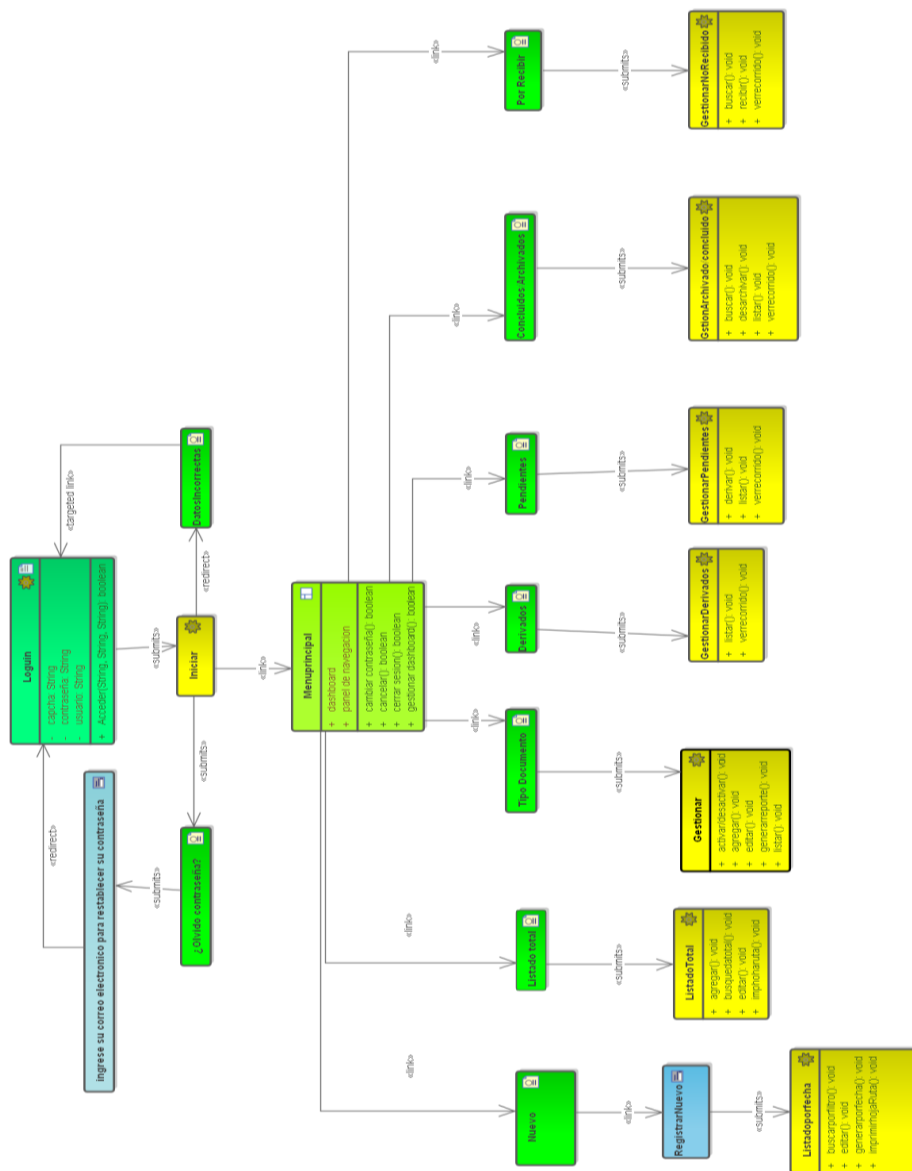
3.2.3.1. Modelo de Navegación del Usuario Administrador

El modelo de navegación del usuario administrador ha sido diseñado utilizando la herramienta Enterprise Architect. Este modelo visualiza los diferentes objetos de navegación y las relaciones entre ellos, proporcionando una visión clara de cómo el administrador interactúa con el sistema. A continuación, se presenta el diagrama de navegación correspondiente:

3.2.3.2. Modelo de Navegación del usuario secretaria(o)

El modelo de navegación del usuario secretaria(o) ha sido diseñado utilizando la herramienta Enterprise Architect. Este modelo representa cómo la secretaria(o) interactúa con el sistema, permitiendo realizar las tareas correspondientes de manera eficiente. A continuación, se presenta el diagrama de navegación que muestra las acciones y contextos asociados a este rol:

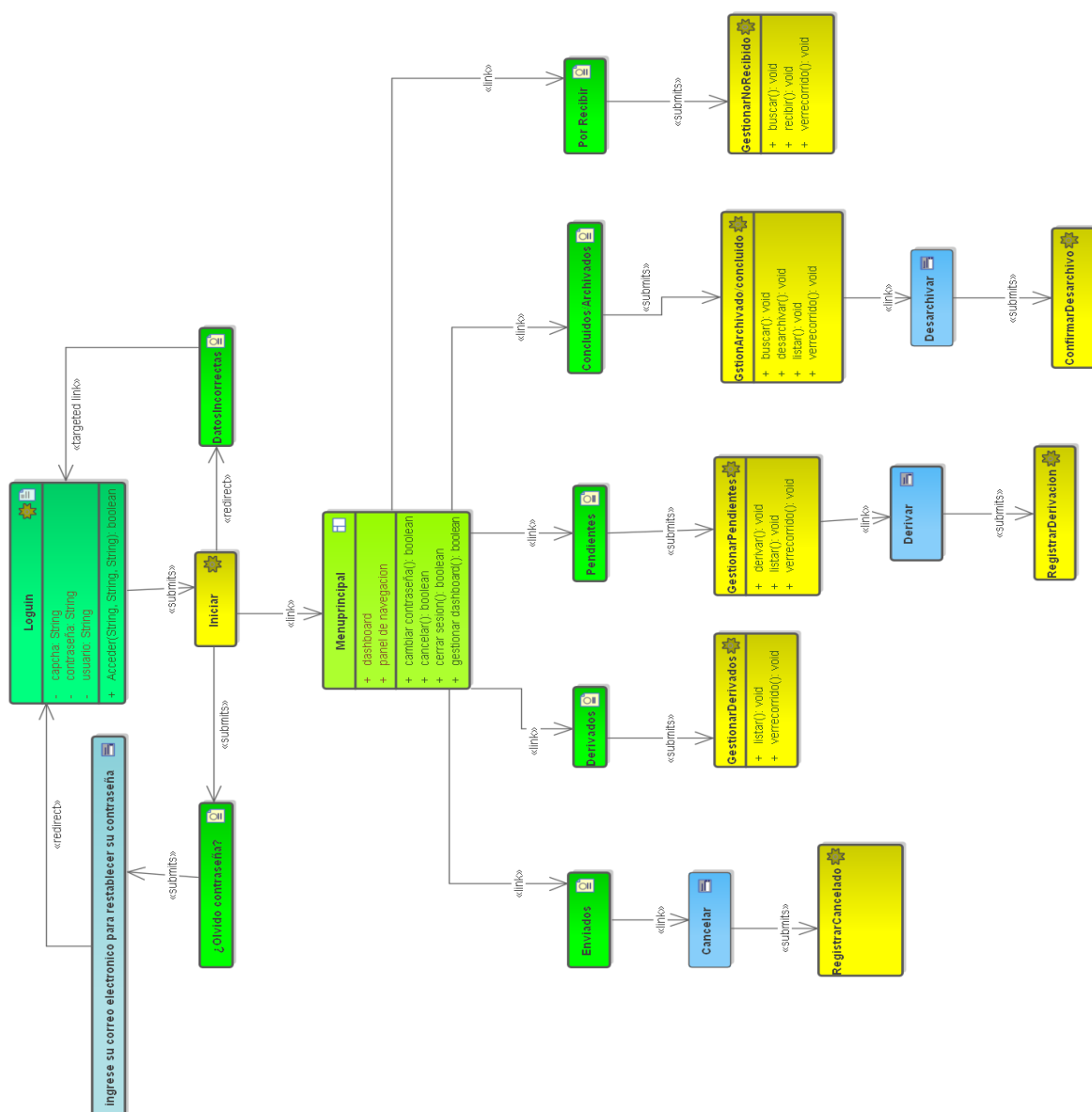
Figura 21.
Diagrama Navegacional – secretaria(o)



3.2.3.3. Modelo de Navegación del usuario jefe de la unidad

El modelo de navegación del usuario jefe de la unidad ha sido diseñado utilizando la herramienta Enterprise Architect. Este modelo refleja las acciones que el jefe de la unidad puede realizar dentro del sistema, así como los contextos de navegación específicos para su rol. A continuación, se presenta el diagrama de navegación correspondiente:

Figura 22.
Diagrama Navegacional - jefe de la Unidad

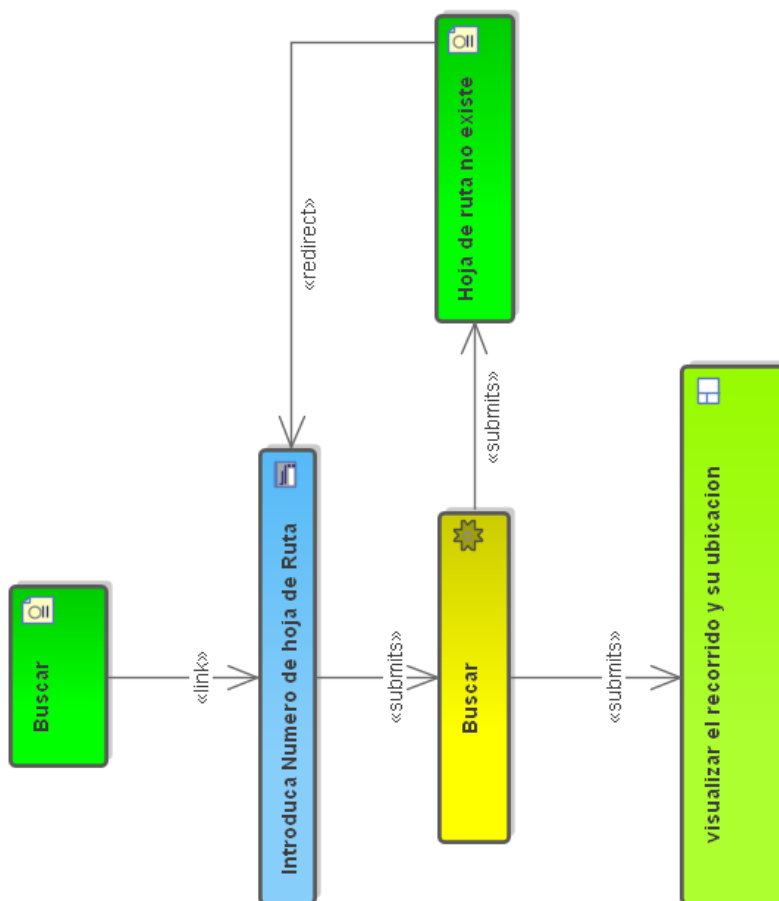


3.2.3.4. Modelo de Navegación del remitente

El modelo de navegación del usuario remitente ha sido diseñado utilizando la herramienta Enterprise Architect. Este modelo visualiza las interacciones que el remitente tiene con el sistema, permitiéndole realizar las tareas necesarias de manera eficiente. A continuación, se presenta el diagrama de navegación correspondiente:

Figura 23.

Diagrama Navegacional Remitente

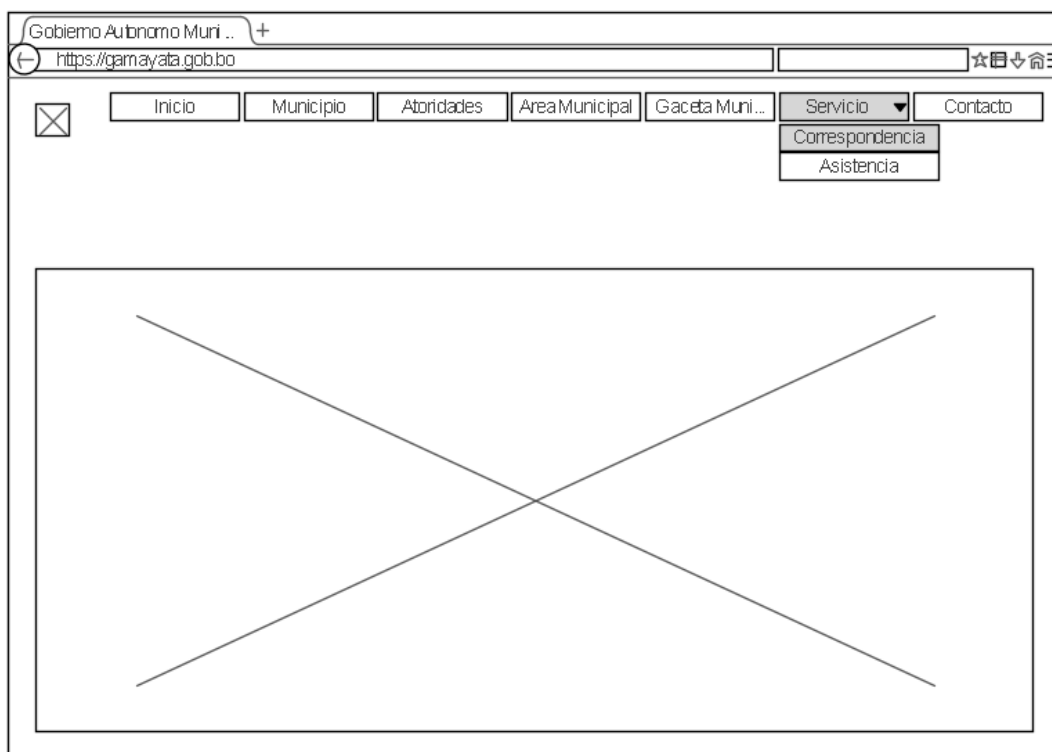


3.2.4. FASE IV: Diseño de Interfaz Abstracta

Al igual que los modelos anteriores, este diseño fue realizado con la herramienta de modelado Enterprise Architect, utilizando específicamente el modelo "Webpage Wireframe Diagram".

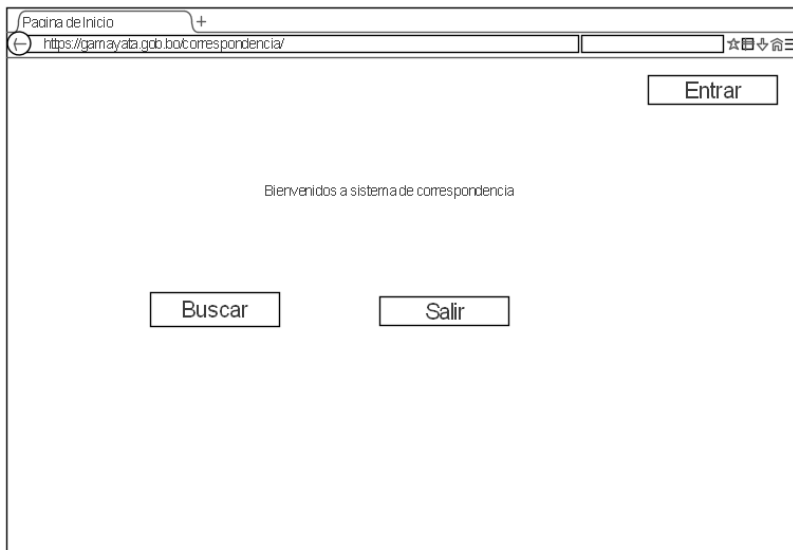
Figura 24.

Diseño de Pantalla Principal



3.2.4.1. Gestión de correspondencia

Los usuarios registrados y los remitentes pueden acceder a las siguientes opciones, según la acción que deseen realizar:

Figura 25.*Diseño de Pantalla de Inicio*

3.2.4.2. Opcion de Entrar

Para usuarios registrados, incluyendo administrador, secretaria y jefe de unidad

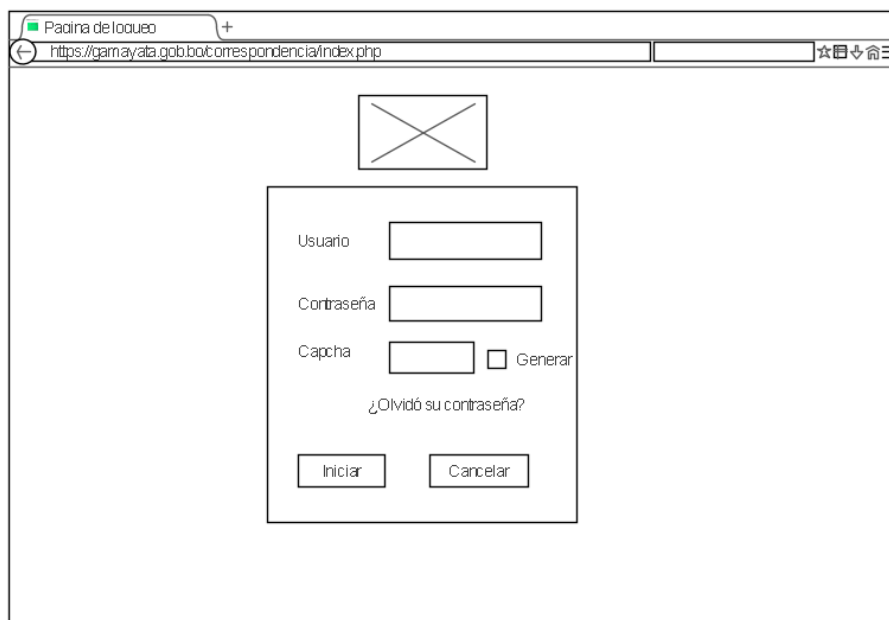
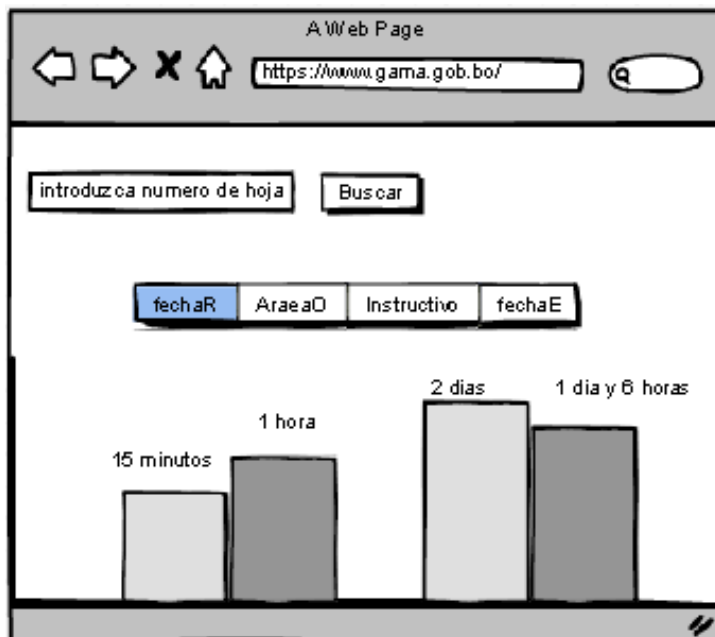
Figura 26.*Inicio de Sesión*

Figura 35.*Control de Correspondencia*

3.2.4.7. Opción salir

La opción "Salir" redirige a la pantalla principal de la página web del municipio.

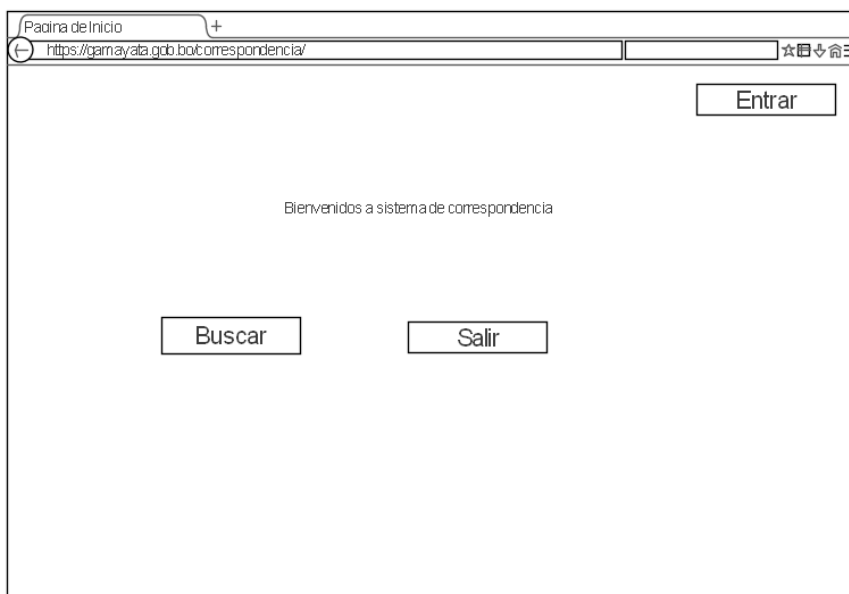
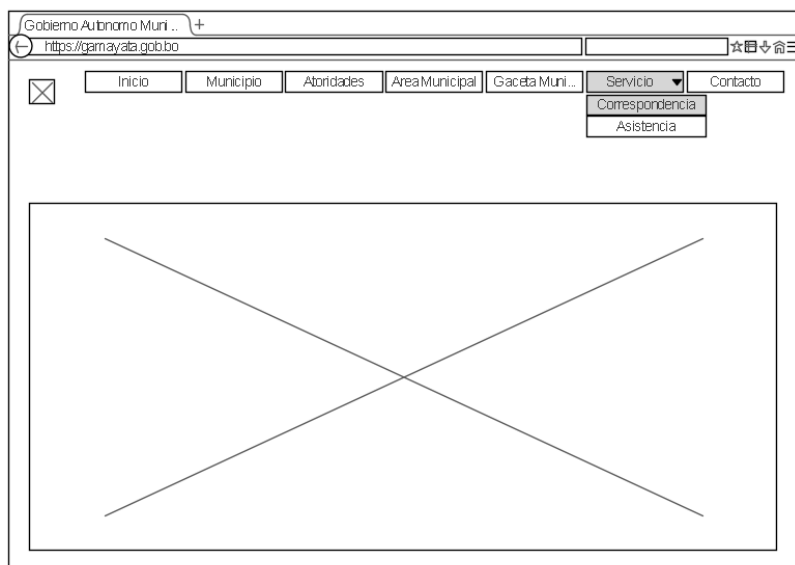
Figura 36.*Diseño Interfaz Modal para Opción Salir*

Figura 37.*Diseño de Interfaz Menu Navegacional*

3.2.5. FASE V: Implementación

3.2.5.1. Implementación del Sistema

En esta fase, se realiza la implementación del sistema en un entorno de producción. El objetivo es garantizar que el sistema esté disponible y funcionando correctamente en un servidor adecuado, siguiendo los estándares de hardware y software necesarios. Se realizan las configuraciones finales, las pruebas de funcionamiento y la implementación en la red, asegurando que el sistema esté operativo para los usuarios finales.

3.2.5.1.1. Requerimientos de Hardware

Para asegurar el correcto funcionamiento del sistema, se utilizó el siguiente hardware:

Laptop: Procesador Intel Core i3.

Memoria RAM: 4 GB.

Disco Duro: 750 GB.

3.2.5.1.2. Requerimientos de Software

El sistema fue desarrollado utilizando las siguientes herramientas y tecnologías:

Servidor Local: XAMPP (con Apache, PHP 8.* y MariaDB).

Editor de Código: Visual Studio Code.

Gestión de Bases de Datos: Navicat para MariaDB.

Modelo de Desarrollo: MVC (Modelo-Vista-Controlador).

Plantillas y Frameworks: AdminLTE para la interfaz de usuario, Bootstrap 4 para el diseño responsivo.

3.2.5.1.3. Librerías y Lenguajes:

Frontend: HTML, CSS, JavaScript (con jQuery).

Interactividad: AJAX, JSON, SweetAlert2 (para mensajes emergentes).

Navegador: Google Chrome para pruebas y desarrollo.

A continuación, se presenta la descripción del proceso de implementación.

3.2.5.2. Proceso de Implementación

Para la implementación del sistema, gestioné personalmente la adquisición de un plan de hosting a través de un proveedor externo, ya que el Gobierno Autónomo Municipal de Ayata no contaba previamente con un servicio de hospedaje web. La institución proporcionó los fondos necesarios para el proyecto, permitiendo que se cubrieran los costos del hosting de 6 GB y el dominio gamayata.gob.bo. Además, realicé los trámites y coordinaciones para obtener el dominio institucional.gob.bo, asegurando así que el sistema cumpla con los requisitos oficiales y esté accesible para los usuarios.

3.2.5.3. Subida a Producción

Una vez realizados los pagos y aprobados los trámites para el dominio.gob.bo, el sistema fue subido al servidor de producción. Actualmente, el sistema está en línea y accesible para los usuarios a través del dominio oficial: <https://gamayata.gob.bo>.

3.2.5.3.1. Configuración del Servidor

En el servidor de producción, se configuró correctamente el entorno de ejecución del sistema con los siguientes componentes:

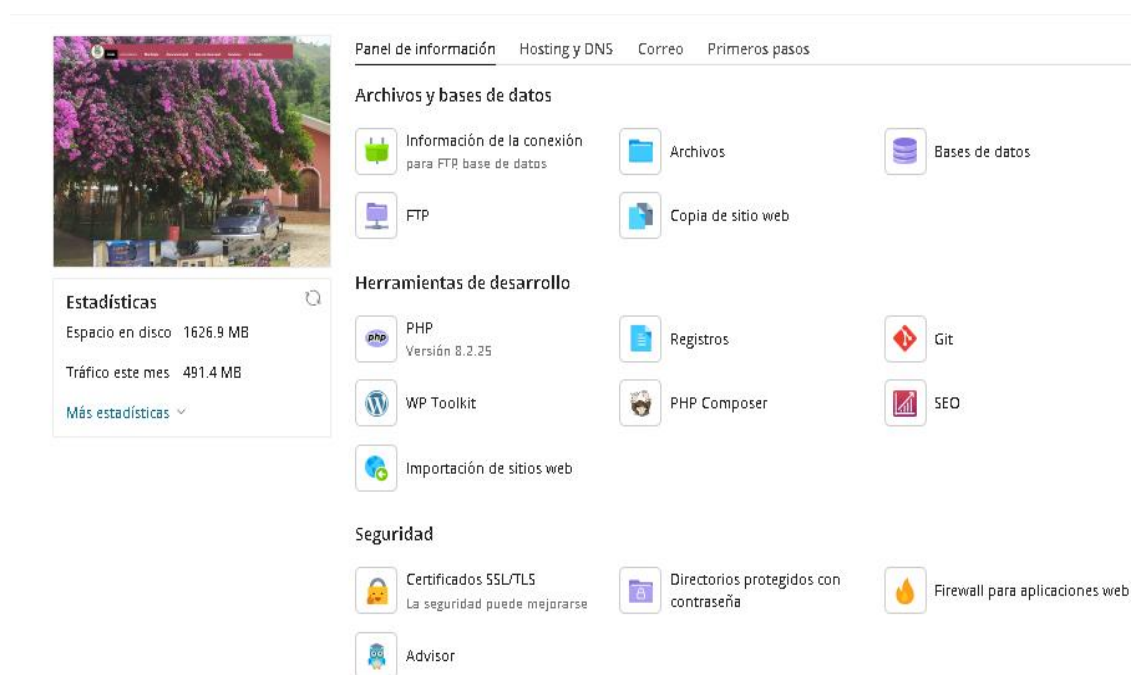
PHP 8.2.25.

MariaDB para la gestión de la base de datos.

Configuración de Apache para servir el sitio web.

Figura 38.

Panel de Administración del Servidor



3.2.5.3.2. Capturas de Pantalla y Demostración Visual

En esta sección, se presentan capturas de pantalla del sistema web de correspondencia, mostrando cómo los distintos tipos de usuarios (secretaria, Administrador, jefe de la Unidad, Remitente) interactúan con el sistema en función de su rol. La demostración incluye el acceso, las secciones principales, y la navegación dentro del módulo de correspondencia.

3.2.5.3.3. Página Principal del Sistema de Correspondencia

Buscar: Permite a los remitentes consultar el estado de su hoja de ruta.

Entrar: Para usuarios autenticados que gestionan la correspondencia según su rol.

Salir: Redirige al usuario a la página principal del sistema.

Figura 39.

Captura de Pantalla Principal

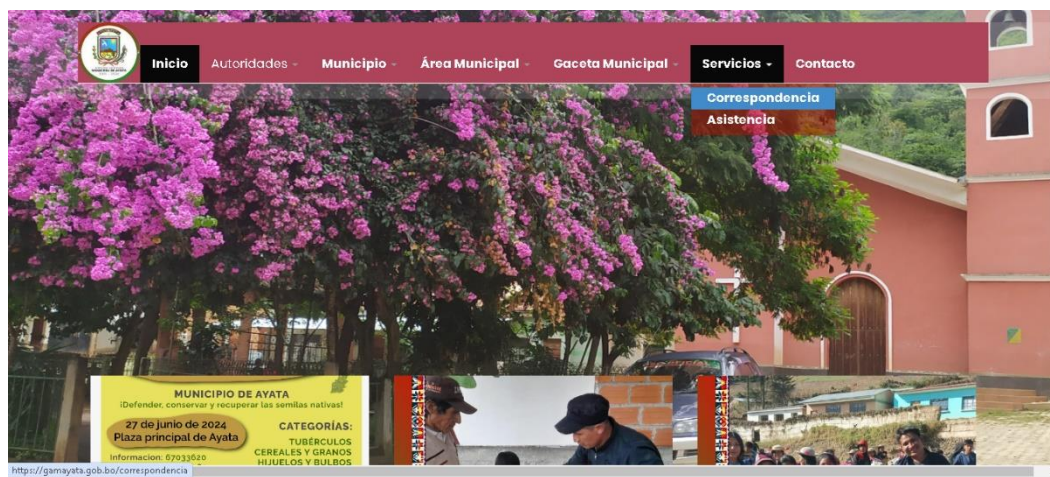


Figura 40.

Captura de Pantalla Mensaje de Bienvenida



Figura 41.
Interfaz de Búsqueda de Correspondencia

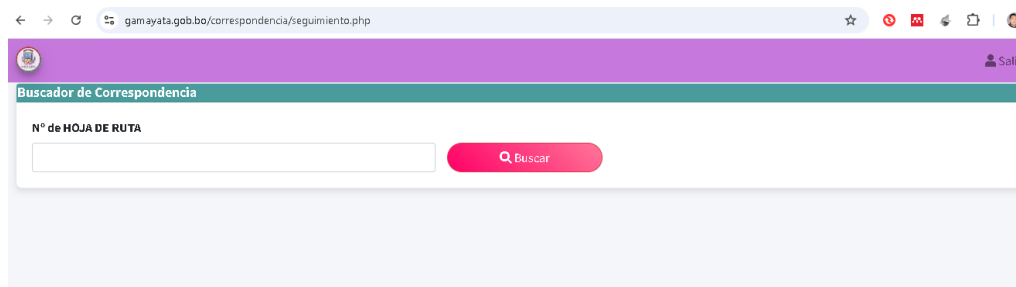


Figura 42.
interfaz de Mensaje de Advertencia - Hoja de Ruta no Encontrada

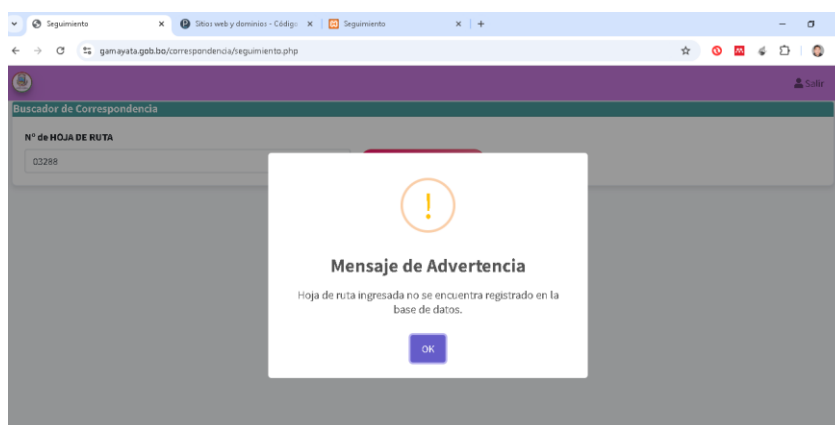


Figura 43.
Control de Correspondencia

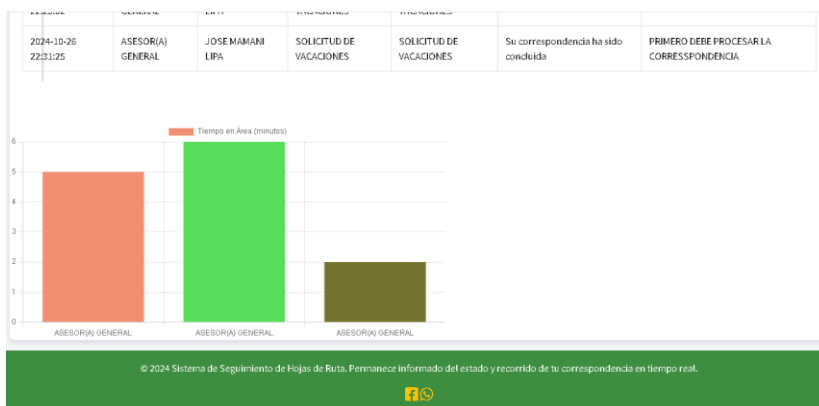


Figura 44.**Interfaz de Seguimiento de Correspondencia**

22-10-2024 10:29

FECHA HORA RECIBIDO	REMITIDO AL DPTO	PERSONAL ASIGNADO	ASUNTO DE TRAMITE	INSTRUCTIVO	ESTADO DE CORRESPONDENCIA	FECHA-HORA DESPACHADO
2024-10-22 10:29:58	RESPONSABLE DE TRANSPARENCIA Y LUCHA CONTRA LA COR	VICTOR VILLARROEL ALVAREZ	SOLICITUD DE CERTIFICADO DE TRABAJO	SOLICITUD DE CERTIFICADO DE TRABAJO	No Recibido	PRIMERO DEBE PROCESAR LA CORRESPONDENCIA

1,0
0,9

■ Tiempo en Área (minutos)

Figura 45.**Interfaz de Acceso para Usuarios Registrados**

Usuario

Contraseña

uat

Genera

Ingresa el texto de la imagen

¿Olvidó su contraseña?

Iniciar Cancelar

Usuarios registrados: Opciones para visualizar y gestionar toda la correspondencia del sistema según su privilegio.

Figura 46.
Interfaz de Panel de Usuario de Administrar



Figura 47.
interfaz de Panel de secretaria

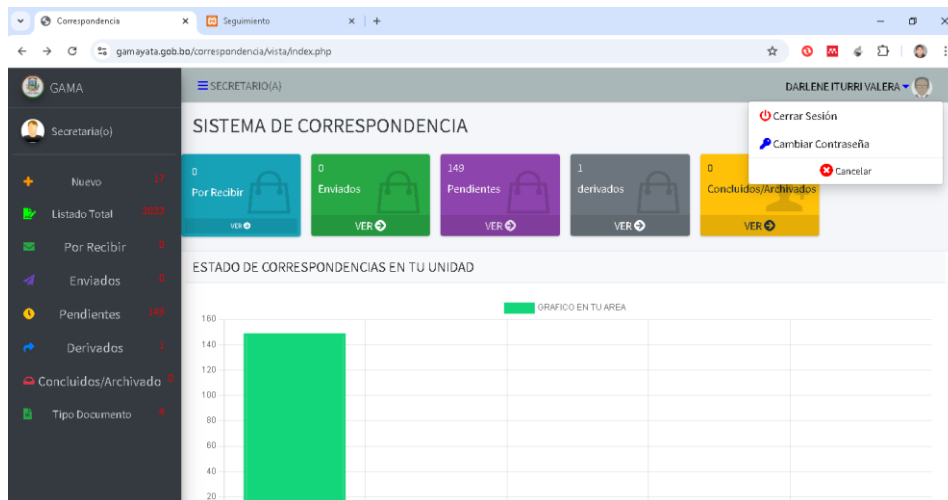


Figura 48.
Interfaz Formulario de registro de nueva correspondencia

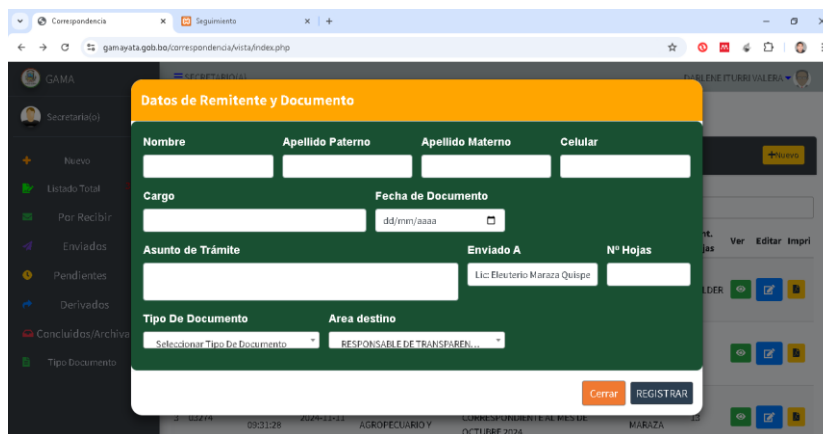


Figura 49.
Hoja de Ruta Generada



Figura 50.
Interfaz Listado de Registros

Numero Documento	Fecha en Recepción	Fecha de Documento	Remite	Asunto	Enviado A:	Cant. Hojas	Ver	Editar	Impri
03272	2024-11-12 09:24:40	2024-11-11	ARQ. MIGUEL S. MAGUERO CENTELLAS SUPERVISOR DE OBRA	SOLICITUD DE CERTIFICACION PRESUPUESTARIA - CONST. CASA CULTURAL PALHUAYA	LIC: ELEUTERIO MARAZA QUISPE	FOLDER			
03273	2024-11-12 09:27:30	2024-11-11	ING. ADALID MAMANI LAGUNA PROYECTISTA - SUPERVISOR DE OBRAS	SOLICITUD DE PAGO DEL MES DE OCTUBRE 2024	LIC: ELEUTERIO MARAZA QUISPE	13			
03274	2024-11-12 09:31:28	2024-11-11	JUVENAL CHURA MOLINA TECNICO AGROPECUARIO Y MEDIO AMBIENTE	SOLICITUD DE PAGO CORRESPONDIENTE AL MES DE OCTUBRE 2024	LIC: ELEUTERIO MARAZA QUISPE	13			

Figura 51.
Interfaz panel de jefe de la Unidad



Figura 52.
Interfaz - Opción de Derivar a Otras Áreas

DERIVAR O ARCHIVAR ARCHIVO00203

Fecha Registro: 2024-02-15 23:16:04 | Accion: DERIVAR

Area de Origen: ASESORÍA GENERAL | Area Destino: Seleccionar Área

Anexar Documento: Seleccionar archivo | Ningún archivo seleccionado

Instructivo:

Close | Registrar

La fase de implementación culmina con el sistema en línea, operando bajo el dominio institucional gamayata.gob.bo

CAPÍTULO IV

CALIDAD, SEGURIDAD Y COSTOS

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



4. CAPÍTULO IV CALIDAD, SEGURIDAD Y COSTOS

4.1. INTRODUCCIÓN

En este capítulo se presenta detalladamente la calidad, el costo y la seguridad del presente proyecto a través de diversas métricas, estas métricas son cruciales para asegurar que el sistema cumpla con los requisitos técnicos y las expectativas del usuario.

Las métricas específicas que se utilizan para evaluar cada uno de estos aspectos, proporcionando una base sólida para medir y controlar el desempeño del sistema que ayudará a determinar que el software opera de manera efectiva y cumple con los estándares establecidos, las métricas de costo permitirán analizar la eficiencia en la gestión de recursos y el presupuesto, mientras que las métricas de seguridad evaluarán la capacidad del sistema para proteger en contra de las amenazas y vulnerabilidades.

Al detallar estas métricas, el capítulo ofrecerá una comprensión integral de cómo evaluar y asegurar la excelencia del sistema en todos estos ámbitos clave.

4.2. MÉTRICAS DE CALIDAD - ESTÁNDAR ISO/IEC 25000

Para el presente Proyecto de Grado, se han implementado métricas de calidad que permiten evaluar diversos parámetros del sistema. Estas métricas son fundamentales para especificar de manera ordenada las características y atributos del software. Entre los diversos modelos y criterios de calidad para productos de software, uno de los más destacados es el ISO 25000. Este estándar ofrece un marco integral para evaluar la calidad del software y se utiliza para medir varios aspectos, incluyendo la funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad del sistema, de acuerdo al siguiente detalle:

4.2.1. Funcionalidad

En este grupo se conjunta una serie de atributos que permiten calificar si un producto de software maneja en forma adecuada, el conjunto de funciones que satisfagan las necesidades para las cuales fue diseñado.

Para el cálculo funcional se debe determinar cinco características de dominios de información los valores de información se definen de la siguiente manera.

Número de entradas de usuario. Se cuenta cada entrada de usuario que proporciona diferentes datos orientados a la aplicación. Las entradas se deberían diferenciar de las peticiones, las cuales se cuentan de forma separada.

Número de salidas de usuario. Se cuenta cada salida que proporciona al usuario información orientada a la aplicación. En este contexto la salida se refiere a informes, pantallas, mensajes de error, etc.

Número de peticiones de usuario. Una petición se define como una entrada interactiva que produce la generación de alguna respuesta del software inmediata en forma de salida interactiva. Se cuenta cada petición por separado.

Número de archivos. Se cuenta cada archivo maestro lógico (esto es, un grupo lógico de datos que puede ser una parte de una gran base de datos o un archivo independiente).

Número de interfaces externas. Se cuentan todas las interfaces legibles por la máquina (por ejemplo: archivos de datos de cinta o disco) que se utilizan para transmitir información a otro sistema.

Tabla 34.*Factores de Ponderación*

VALOR DOMINIO DE INFORMACION	CONTEO	FACTOR DE PONDERACION			TOTAL
		SIMPLE	PROMEDIO	COMPLEJO	
Entradas externas (EE)	4	-	4	-	16
Salidas externas (SE)	9	-	5	-	45
Consultas externas (CE)	8	-	4	-	32
Archivos lógicos internos (ALI)	9	-	10	-	90
Archivos de interfaz externos (AIE)	0	-	7	-	0
TOTAL, PFNA					183

En la siguiente tabla se muestra el factor de ajuste de complejidad basado en las respuestas de las siguientes preguntas evaluadas.

Tabla 35.*Indice de Usabilidad*

Factores de complejidad	Sin influencia	Inicial	Moderado	Medio	Significativo	Esencial	Fi
	0	1	2	3	4	5	
¿Requiere el sistema copias de seguridad y de recuperación fiables?						X	5
¿Se requiere comunicación de datos?					X		4
¿Existen unciones de procesamiento distribuido?					X		4
¿Es crítico el rendimiento?					X		4

¿Se ejecutará el sistema con un entorno operativo existente y fuertemente utilizado?					X	5
¿Requiere el sistema entrada de datos interactiva?				X		4
Facilidad Operativa				X		4
¿Se actualiza los archivos maestros de forma interactiva?					X	5
¿Son complejas las entradas, las salidas, los archivos o las peticiones?			X			3
¿Procesamiento interno complejo?				X		4
¿Diseño de código reutilizable?				X		4
¿Facilidad de Instalación?				X		4
¿Facilidad de cambios?				X		4
¿Soporta múltiples instalaciones en diferentes sitios?					X	5
Factor ajuste de complejidad						59

Fórmula para ajustar puntos de función, que considera factores como la complejidad técnica, el rendimiento y la usabilidad.

(15) Ecuación de Ajuste

$$PF = \text{Conteo total} * [0.65 + 0.01 * \sum fi] \quad (18)$$

Donde:

$\sum (f_i)$: Sumatoria de los valores de los factores de ajuste.

Conteo total : Número de parámetros del punto función

PF : Punto de Función

Se obtiene la sumatoria de ponderación de los índices de usabilidad.

0.65: Valor base mínimo que representa la complejidad inicial del sistema.

0.01: Escalador para ajustar la influencia de las características generales del sistema

$$\sum (f_i) = 59$$

$$PF = 183 * [0.65 + 0.01 * 59]$$

$$PF = 226.9$$

Con el límite superior establecido en la suma de los factores de ajuste máximo $\sum (f_i \text{ max}) = 70$, se realiza el cálculo del valor siguiente en la secuencia, este proceso es fundamental para mantener la coherencia en la evaluación de la complejidad del sistema, hallamos el punto función máximo.

$$PF_{max} = Cuenta\ Total \left[0.65 + (0.01 * \sum f_i \text{ max}) \right]$$

$$PF_{max} = 183 * [0.65 + (0.01 * 70)]$$

$$PF_{max} = 247.05$$

La medición de la funcionalidad se basa en comparar el valor máximo posible con el valor obtenido del punto de función ajustado, esta relación proporciona información crucial sobre la eficiencia y la complejidad del sistema evaluado.

$$Funcionalidad = \frac{PF}{PF_{max}} * 100$$

$$Funcionalidad = \frac{226.9}{247.05} * 100$$

$$\text{Funcionalidad} = 0.92 * 100$$

Funcionalidad = 92%

Entonces, la eficacia del sistema alcanza el 92 %, indicando su alta capacidad operativa con un riesgo mínimo de fallos (8 %),

4.2.2. Confiabilidad

La confiabilidad de un sistema se refiere a su capacidad para mantener un nivel de funcionamiento adecuado durante un período de prueba establecido que implica considerar aspectos como la madurez del sistema. Aplicaremos las fórmulas correspondientes, teniendo en cuenta las probabilidades de fallas y éxitos debido a varios factores.

Considerando que se tiene en cuenta que:

Probabilidad de fallas

$$F(t) = P(T \leq t)$$

Probabilidad de trabajo sin fallas

$$P(T > t) = 1 - F(t)$$

Se considera la siguiente función al calcular la confiabilidad del sistema.

(16) Función de la Confiabilidad del Sistema

$$R(t) = f * e^{-\mu * t} \quad (19)$$

Donde:

- f = Funcionalidad del sistema
- μ = Probabilidad de error del sistema
- t = Tiempo de prueba del sistema

$F(t)$ = Confiabilidad

Durante un periodo de prueba de 20 días, se registra una falla por cada 9 ejecuciones del sistema.

$$F(t) = 0.92 * e^{-\frac{1}{9} * 20}$$

$$F(t) = 0.92 * 0.111$$

$$F(t) = 0.101$$

$$F(t) = 0.101 * 100$$

$$F(t) = 11 \%$$

Este valor representa la probabilidad de fallas del sistema en un periodo de tiempo, por lo que la siguiente formula representa la probabilidad de que el sistema sea confiable.

Sustituyendo:

$$\text{Confiabilidad} = P(T \leq t) = 1 - F(t)$$

$$\text{Confiabilidad} = P(T \leq t) = 0.101$$

$$\text{Confiabilidad} = P(T \leq t) = 1 - 0.101$$

$$\text{Confiabilidad} = P(T \leq t) = 0.90 * 100$$

$$\text{Confiabilidad} = P(T \leq t) = 90 \%$$

Por lo tanto, para este software desarrollado, la confiabilidad se sitúa en un sólido 90%, verificada a lo largo de un periodo de prueba de 20 días, este resultado confirma la consistencia y fiabilidad del sistema en condiciones operativas durante este período de

evaluación establecido, cabe mencionar que en todo proceso se tiene un margen de error, que en lo recomendable sea mínimo.

4.2.3. Usabilidad

Al evaluar la usabilidad de un sistema, es importante considerar el factor humano, asegurando que el software cumpla con los requisitos y expectativas del usuario, por lo tanto, se lleva a cabo una evaluación mediante encuestas dirigidas a los usuarios del sistema.

Estas encuestas permiten recopilar información valiosa sobre la experiencia del usuario, su satisfacción y la facilidad de uso percibida que se utiliza luego para calcular la usabilidad del sistema, aplicando una fórmula diseñada específicamente para medir la eficacia y la experiencia del usuario en la interacción con el software.

(17) Función para Determinar la Usabilidad

$$FU = \left(\frac{\sum \text{valor}}{n} * 100 \right) / 5 \quad (20)$$

Donde:

\sum valor = Sumatoria de los valores de la usabilidad del sistema

n = Cantidad de valores de la usabilidad del sistema

FU = Usabilidad

Tabla 36.

Indicadores de Facilidades de Uso

ESCALA	VALOR
Muy bueno	5
Bueno	4
Regular	3
Malo	2
Pésimo	1

A continuación, en la siguiente tabla se muestra:

Tabla 37.

Usabilidad del Sistema

PREGUNTA	VALOR
¿No es complicado usar el sistema?	5
¿Las pantallas de vista fueron de su agrado?	4
¿Los datos obtenidos que vio fueron fáciles de comprender?	5
¿El sistema responde rápido a sus solicitudes?	5
¿El sistema facilita el trabajo para generar datos?	5
¿La respuesta del sistema es satisfactoria?	4
¿Le parece complicada las funciones del sistema?	4
¿El sistema reduce el tiempo de trabajo en generar informes?	5
¿Los resultados que proporciona el sistema facilitan el trabajo?	5
¿Durante el uso del sistema se produjo errores?	4
TOTAL	46

A partir de los datos recabados a través del cuestionario, llevamos a cabo un análisis exhaustivo para evaluar la usabilidad del sistema. Este proceso implica calcular diversos indicadores y métricas que reflejen la facilidad con la que los usuarios interactúan con el sistema.

$$FU = \left(\frac{46}{10} * 100 \right) / 5$$

$$FU = (4.4 * 100) / 5$$

Usabilidad = FU = 92%

Con esta información, podemos afirmar que la facilidad de uso del sistema alcanza un porcentaje del 92%, este dato proporciona una medida concreta de la experiencia del usuario y su capacidad para interactuar efectivamente con el sistema.

4.2.4. Eficiencia

La eficacia evalúa si el sistema utiliza eficientemente sus recursos. Para determinar su nivel, se considera una escala específica como referencia.

Tabla 38.

Escala de Valores de Eficiencia

ESCALA	VALOR
Excelente	5
Bueno	4
Aceptable	3
Deficiente	2
Pésimo	1

Tabla 39.

Evaluación de Eficiencia

PREGUNTA	PORCENTAJE
¿La distribución y estilo de la interfaz permite que un usuario introduzca con eficiencia las operaciones y la información?	5
¿Una secuencia de operaciones (o entrada de datos) puede realizarse con facilidad de movimientos?	5
¿Los datos de salida están presentados de modo que se entienden de inmediato?	4

¿Las operaciones jerárquicas están organizadas de manera que minimizan la navegación del usuario para hacer que alguna se ejecute?	4
¿Procesa y responde adecuadamente cuando realiza alguna consulta o búsqueda?	5
TOTAL	23

Para determinar la eficacia, se emplea la fórmula siguiente.

(18) Fórmula Eficiencia

$$E = \frac{\sum x_i}{n} * \frac{100}{n} \quad (21)$$

Donde:

$\sum x_i$ = Sumatoria de los valores de eficiencia

n = Número de preguntas

E = Eficiencia

Sustituyendo se tiene lo siguiente:

$$E = \frac{23}{5} * \frac{100}{5}$$

<i>Eficiencia = E = 92 %</i>

Con estos datos, determinamos que la eficiencia del sistema se sitúa en el 92%, lo que indica que los recursos del sistema se están utilizando de manera efectiva y óptima.

4.2.5. *Mantenibilidad*

Esta métrica evalúa la cantidad de trabajo requerido para realizar cambios en el sistema, ya sea para corregir errores o agregar nuevas funcionalidades. El estándar IEEE94 recomienda utilizar el índice de madurez del sistema como indicador de su estabilidad. Por lo tanto, la ecuación para este índice es la siguiente:

(19) Ecuación para el Índice de Madurez del Sistema

$$IMS = \frac{[Mt - (Fa + Fc + Fd)]}{Mt} \quad (22)$$

Donde:

- Mt* = Número de módulos total de la versión actual
- Fa* = Número de módulos de la versión actual que se añadieron.
- Fc* = Número de módulos de la versión actual que se cambiaron.
- Fd* = Número de módulos que se eliminaron en la versión anterior a la actual.
- IMS = Madurez del sistema

Tabla 40.

Indicadores de Mantenibilidad

DESCRIPCIÓN	VALOR
Mt	14
Fa	0
Fc	0
Fd	0

Sustituyendo en la ecuación se tiene:

$$IMS = \frac{[14 - (0 + 0 + 0)]}{14}$$

$$IMS = 1 * 100\%$$

IMS = 100%

Con la interpretación a este resultado establece un 100% de madures, lo que indica que no requiere de mantenimientos inmediatamente.

4.2.6. Portabilidad

Se evalúa la oportunidad para adaptar el software a diferentes ambientes sin necesidad de aplicarle modificaciones, es decir que el software se puede trasladar de un entorno a otro.

Para poder medir la portabilidad del sistema usaremos la siguiente formula que indica el grado de portabilidad que tiene un software.

$$GP = 1 - (ET/ER)$$

Donde:

ET: Es la medida de los recursos necesarios para llevar el sistema a otro entorno.

ER: Es la medida de los recursos necesarios para crear el sistema en el entorno residente.

Si $GP > 0$, la portabilidad es más rentable que el desarrollo

Si $GP = 1$, la portabilidad es perfecta

Si $GP < 0$, el desarrollo es más rentable que la portabilidad.

Entonces la factibilidad estimada para trasportar el sistema es de 1/3 días a otro entorno y la implementación del mismo en otro entorno es de 5 días.

(20) Fórmula Portabilidad

$$Portabilidad[GP] = 1 - \left(\frac{[ET]}{[ER]} \right) \quad (23)$$

Sustituyendo la fórmula

$$Portabilidad = 1 - \left(\frac{[1/3]}{5}\right)$$

$$Portabilidad = 0.93 * 100\%$$

$Portabilidad = 93,3\%$

Por lo que se concluye que el sistema tiene un grado de portabilidad del 93,3%.

Después de un análisis exhaustivo de cada parámetro según los criterios establecidos por la norma ISO 25000 se ha obtenido resultados que reflejan el rendimiento global del sistema, el cálculo detallado de cada aspecto relevante para asegurar que el sistema cumple con los estándares de calidad y funcionalidad esperados, proporcionando así una visión clara y detallada de su desempeño general.

Tabla 41.

Resultados Obtenidos - Norma ISO 25000

CRITERIOS EVALUADOS	RESULTADOS
Funcionalidad	92 %
Confiabilidad	90 %
Usabilidad	92 %
Eficiencia	92%
Mantenibilidad	100 %
Portabilidad	93.3 %
TOTAL	93.2%

4.3. ESTIMACIÓN DE COSTO COCOMO

COCOMO es un método que proporciona una estimación precisa de los costos asociados con el desarrollo de proyectos de software, este enfoque facilita la elaboración de

presupuestos, la planificación y la mejora en la ejecución del proyecto que evalúa el tamaño del software basándose en las líneas de código producidas durante el desarrollo lo que permite una estimación más exacta de los costos y ayuda a una planificación y a una adecuada asignación de recursos.

El cálculo de los costos del proyecto se realizará utilizando el modelo COCOMO que se divide en tres niveles: Orgánico, semi acoplado y empotrado, estos niveles se ajustan a diferentes tipos de software y etapas del desarrollo permitiendo una estimación precisa de los costos y el esfuerzo necesario para completar el proyecto con éxito (Galvan, 2014, p. 13).

4.3.1. Modelo Semi Acoplado

Se aplica a proyectos de complejidad y tamaño intermedios, con experiencia variable y restricciones moderadas.

4.3.2. Costo del Software

Para la estimación de costos del sistema ha sido desarrollado bajo las KLDC (Kilo-Líneas de Código) las que detallamos lo siguiente:

Para este proyecto basado en la funcionalidad obtenemos como base 489.9 de PF que nos servirá para calcular el tamaño de la programación PHP.

En la tabla se muestra la relación para convertir el valor de PF a KLDC

Tabla 42.

Complejidad y Eficiencia del Lenguaje.

LENGUAJE	NIVEL	FACTOR LDC/PF
C	2.5	128
JAVA	6	53
ANSI COBOL 74	3	107
ASP	9.00	36

PHP	11.00	29
VISUAL C++	9.50	34

Nota. Conversión de puntos función, Pressma, 2020, costos de software

Entonces, realizando los cálculos y escogiendo el valor del lenguaje de programación PHP de la tabla, tenemos:

$$LDC = PF * \text{Factor LDC}$$

$$LDC = 226,9 * 29$$

$$LDC = 6580$$

$$KLDC = 6580 / 1000$$

$$KLDC = 6.6$$

Ahora, para hallar el esfuerzo reemplazamos los valores obtenidos hasta ahora:

4.3.3. Ecuaciones de COCOMO Intermedio

Tabla 43.

Formulas Para el Costo

Variable	Ecuación	Tipo / Unidad
Esfuerzo requerido por el proyecto	$E = a * (KLDC)^b * FAE$	Persona/mes
Tiempo requerido por el proyecto	$T = c * (E)^d$	Mes
Número de personas requeridos para el proyecto	$P = \frac{E}{T}$	Personas
Costo Total	$CT = \text{SueldoMes} * P * T$	\$us

Para hallar los valores de FAE, se utilizará la tabla de atributos multiplicadores.

Tabla 44.

Tabla de Atributos Factores del Esfuerzo

Atributos	VALOR					
	Muy bajo	Bajo	Nominal	Alto	Muy alto	Extra alto
ATRIBUTOS DE SOFTWARE						
Fiabilidad	0,75	0,88	1,00	1,15	1,40	
Tamaño de Base de datos		0,94	1,00	1,08	1,16	
Complejidad	0,70	0,85	1,00	1,15	1,30	1,65
ATRIBUTOS DE HARDWARE						
Restricciones de tiempo de ejecución			1,00	1,11	1,30	1,66
Restricciones de memoria virtual			1,00	1,06	1,21	1,56
Volatilidad de la máquina virtual		0,87	1,00	1,15	1,30	
Tiempo de respuesta		0,87	1,00	1,07	1,15	
ATRIBUTOS DE PERSONAL						
Capacidad de análisis	1,46	1,19	1,00	0,86	0,71	
Experiencia en la aplicación	1,29	1,13	1,00	0,91	0,82	
Calidad de los programadores	1,42	1,17	1,00	0,86	0,70	
Experiencia en la máquina virtual	1,21	1,10	1,00	0,90		
Experiencia en el lenguaje	1,14	1,07	1,00	0,95		
ATRIBUTOS DEL PROYECTO						
Técnicas actualizadas de programación	1,24	1,10	1,00	0,91	0,82	

Utilización de herramientas de software	1,24	1,10	1,00	0,91	0,83
Restricciones de tiempo de desarrollo	1,22	1,08	1,00	1,04	1,10
TOTAL				0.843	

La tabla ilustra el cálculo del Factor de Ajuste de Esfuerzo (FAE) mediante la multiplicación de los valores de los atributos relevantes, este procedimiento ajusta la estimación del esfuerzo del proyecto al tener en cuenta factores como la complejidad del sistema de la experiencia del equipo y las condiciones del entorno, como resultado de este ajuste se obtiene un FAE final de 0.843.

4.3.4. Esfuerzo Requerido del Desarrollo

Se emplearán los coeficientes simples de la tabla para calcular el esfuerzo necesario, ya que se ajustan de manera óptima al proyecto el cual se establece así una relación directa entre los valores seleccionados y el cálculo del esfuerzo.

(21) Ecuación Esfuerzo

$$E = a * KLDC^b * FAE, \text{ en personas/mes} \quad (26)$$

Calculando:

$$E = 17.3 * 0.843 \text{ meses/persona}$$

$$E = 14,7 \text{ meses/persona}$$

4.3.5. Tiempo de Desarrollo Requerido del Sistema

Para determinar el tiempo de desarrollo del proyecto se utilizarán los datos obtenidos, incluyendo el factor de ajuste de esfuerzo (FAE) y el tamaño del proyecto en líneas de código.

(22) Ecuación Tiempo Requerido

$$T = c(E)^d, \text{ en meses} \quad (27)$$

Donde:

$$c : \text{ Coeficiente} \quad 2.50$$

$$d : \text{ Coeficiente} \quad 0.38$$

$$E : \text{ Esfuerzo} \quad 14,7$$

Calculando:

$$T = 2.50(14,7)^{0.38} \text{ meses}$$

$$T = 6,9 \text{ meses}$$

$$T = 7 \text{ meses}$$

4.3.6. Número de Desarrolladores

Se determinará el número de desarrolladores requeridos para el sistema, considerando el esfuerzo estimado del proyecto y el tiempo disponible, para asegurar una asignación adecuada de recursos y cumplimiento de plazos.

(23) Ecuación Número de Desarrolladores

$$P = \frac{E}{T}, \text{ en personas} \quad (28)$$

Donde:

$$E : \text{ Esfuerzo} \quad 14.7$$

$$T : \text{ Tiempo requerido} \quad 7$$

Calculando:

$$P = \frac{14.7}{7} \text{ personas}$$

$$P = 2.1 \text{ personas}$$

$$P = 2 \text{ personas}$$

4.3.7. Costo Total

Posteriormente se calculará el monto total a pagar por el software utilizado en este proyecto específico, el cálculo se basa en el salario mínimo nacional en Bolivia, establecido en 2,500 Bs, este valor sirve como referencia para estimar los costos asociados al software permitiendo una planificación financiera precisa y adaptada a la economía local del proyecto.

Considerando el salario mínimo, se podrán calcular de manera detallada los costos laborales y otros gastos relacionados, facilitando una gestión adecuada del presupuesto y asegurando que los recursos se utilicen de manera eficiente en el contexto económico del proyecto.

(24) Ecuación Costo del Software

$$C_{sof} = S_{mes} * P * T \quad (29)$$

Donde:

S_{mes} : Sueldo mes = 2500 Bs.

P : Número de personas = 2

T : Tiempo requerido = 7

Calculando:

$$C_{sof} = 2500 * 2 * 7$$

$$C_{sof} = 35.000 \text{ Bs}$$

$$C_{sof} = 35000 \text{ Bs}$$

Según los cálculos realizados se estima que se necesitarán dos programadores trabajando durante ocho meses para completar el desarrollo del software, durante este período el costo total del proyecto se proyecta en 5000 \$ considerando que el valor del dólar en Bolivia es de 7 Bs realizando una conversión del costo total del proyecto a moneda local, resultando en una estimación de 35000 Bs.-

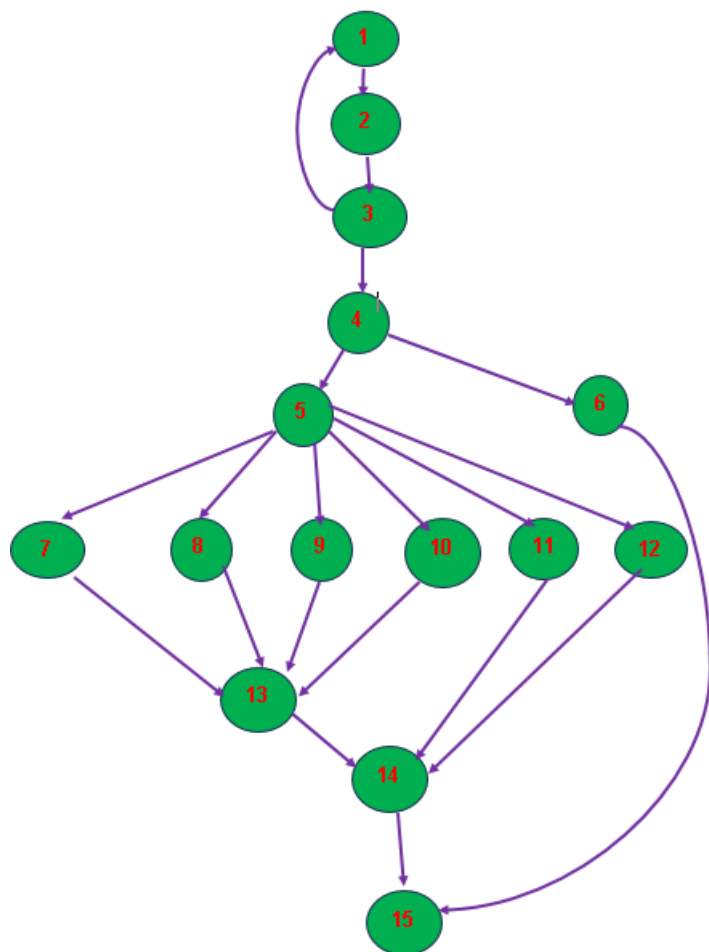
- Se requiere 2 desarrolladores
- Un tiempo de desarrollo de 7 meses
- Costo total estimado 5000 \$ -> 35000 Bs.

4.4. PRUEBAS DE SOFTWARE

Para el presente proyecto nos basaremos en las dinámicas que proponen las pruebas de caja blanca, pruebas de caja negra, pruebas de estrés, pruebas de accesibilidad y pruebas de usabilidad que ayudaran a verificar el funcionamiento del sistema.

4.4.1. Prueba de Caja Blanca

Se llevarán a cabo pruebas de caja blanca para evaluar las condiciones lógicas del portal web y revelar su estructura de datos, este enfoque ayudará a reducir errores y a comprender la complejidad lógica del sistema determinando el número de caminos posibles en el código excluyendo los detalles de entradas y salidas, y proporcionando una visión detallada del flujo lógico del software.

Figura 53.*Diagrama del Prueba de caja blanca*

Después de revisar el diagrama de del sistema y haber generado el grafo correspondiente, el siguiente paso es determinar la complejidad ciclomática, se aplicará la fórmula adecuada el cual permite cuantificar la complejidad del código al medir el número de caminos independientes a través del grafo.

Donde:

1. Inicio del sistema
2. Usuario y contraseña

3. Validar usuario y contraseña
4. Menú principal
5. Menú administrador
6. Cambiar contraseña
7. Gestión de correspondencia
8. Gestión de personal
9. Gestión de usuarios
10. Gestión de unidades
11. Gestión de Dashboard
12. Gestión de Reportes
13. Fin Panel de administración
14. Fin sistema
15. Cerrar sesión

(25) Ecuación Complejidad Ciclomática

$$V(G) = A - N + 2 \quad (24)$$

Donde:

$$A = \text{Número de aristas} = 21$$

$$N = \text{Número de nodos} = 15$$

$$V(G) = \text{Complejidad}$$

Calculamos:

$$V(G) = 21 - 15 + 2$$

$$\mathbf{V(G) = 8}$$

Identificar el conjunto esencial de trayectorias linealmente independientes, lo cual se requiere evaluar 8 caminos específicos, determinados por ciertas variables.

4.4.2. Prueba de Caja Negra

Las pruebas de caja negra se centran en verificar los requisitos funcionales del software sin examinar su código interno, evaluando la interfaz y asegurando que el sistema responda correctamente a las entradas y cumpla con las especificaciones, el objetivo es identificar errores relacionados con el uso práctico del sistema, mejorando así la funcionalidad.

Figura 54.

Prueba de Caja Negra Iniciar Sesión

Tabla 45.

Valores de Caja Negra Login

CAMPOS	TIPO DE DATOS	ENTRADA INVÁLIDA
Usuario	Cadena de texto	espacios en blanco.
Contraseña	Cadena de texto	Espacios en blanco.
Captcha	Imagen	verificar el código

Figura 55.
Alertas a Iniciar

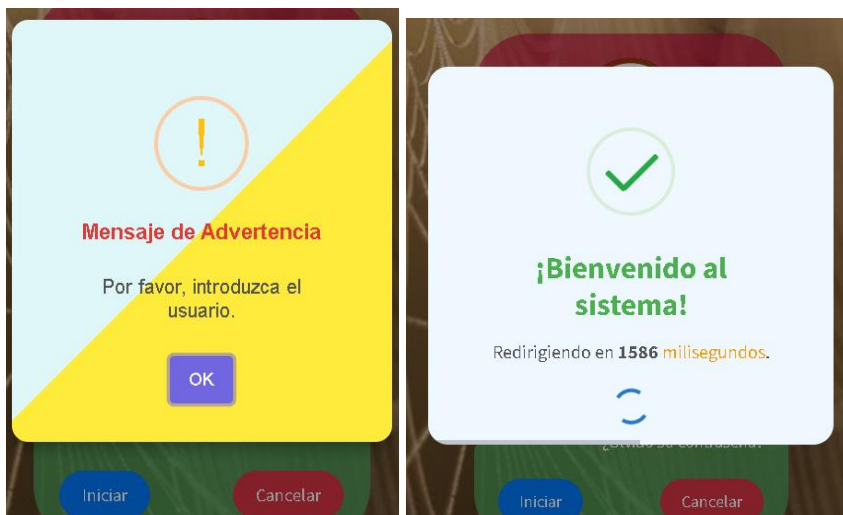


Figura 56.
Caja Negra Ingreso al Sistema

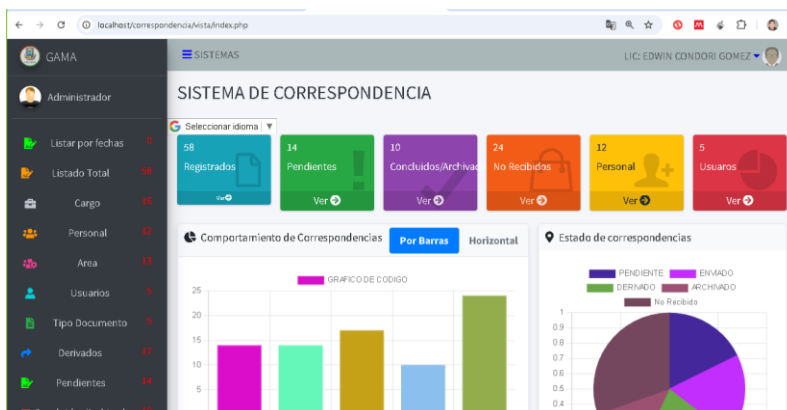


Tabla 46.
Prueba de Caja Negra Ingreso el Sistema

ENTRADA		RESULTADO
Usuario	SABINO	El sistema verifica que no queden campos vacíos y al introducir datos correctos, concede el acceso.
Contraseña	*****	
Captcha	FrG	

A continuación, se muestra la interfaz de acuerdo a la prueba de caja negra

Figura 57.

Prueba de Caja Negra – Agregar Personal

The screenshot shows a web browser window with the URL `localhost/correspondencia/Vista/index.php`. A modal window titled "REGISTRO DE PERSONAL" is open over a background interface. The modal contains the following fields:

- Nro Carnet:** Text input field.
- Nro Complementario:** Text input field.
- CI Expedido en:** Dropdown menu with "La Paz" selected.
- Nombres:** Text input field.
- Apellido Paterno:** Text input field.
- Apellido Materno:** Text input field.
- Sexo:** Dropdown menu with "M" selected.
- Fecha De Ingreso:** Date picker showing "dd/mm/aaaa".
- Celular:** Text input field.
- Cargo:** Dropdown menu with "SELECCIONAR CARGO" selected.
- Direccion:** Text input field.
- Email:** Text input field.

At the bottom right of the modal, there are two buttons: "Close" and "REGISTRAR".

Figura 58.

Prueba de Caja Negra – Registro de Correspondencia

The screenshot shows a web browser window with the URL `gamayata.gob.bo/correspondencia/Vista/index.php`. A modal window titled "Datos de Remitente y Documento" is open. It contains the following fields:

- Nombre:** Text input field.
- Apellido Paterno:** Text input field.
- Apellido Materno:** Text input field.
- Celular:** Text input field.
- Cargo:** Text input field.
- Asunto de Trámite:** Text input field.
- Tipo De Documento:** Dropdown menu with "Seleccionar Tipo De Docum" selected.
- Nº Hojas:** Text input field.

At the bottom right of the form, there are two buttons: "Cerrar" and "REGISTRAR".

Overlaid on top of the form is a "Mensaje de Advertencia" modal. It features a warning icon (exclamation mark in a circle) and the text "Mensaje de Advertencia" and "Llene todos los datos del Documento". There is an "OK" button at the bottom of this modal.

Figura 59.

Prueba de Caja Negra – Registro Exitoso

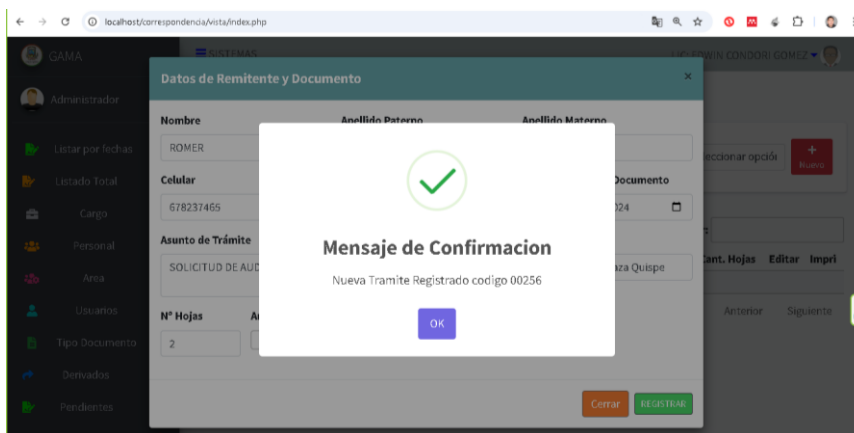


Figura 60.

Generación de Hoja de Ruta



Figura 61.

Prueba de Caja Negra – Mensaje de Advertencia

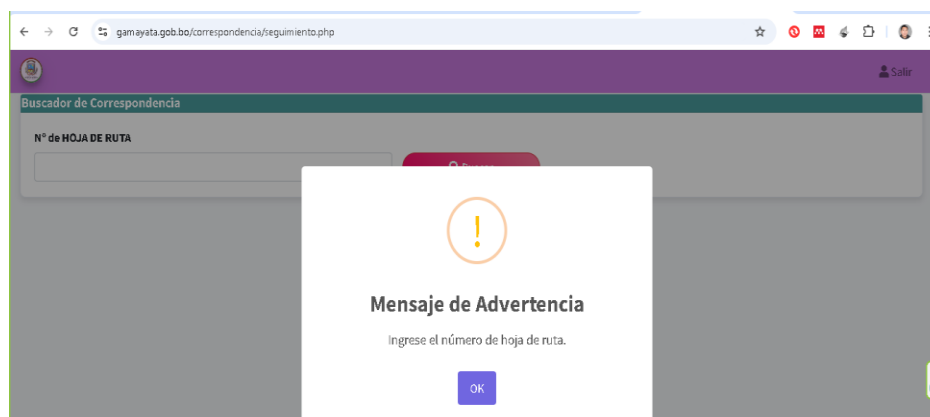


Figura 62.
Caja Negra Seguimiento

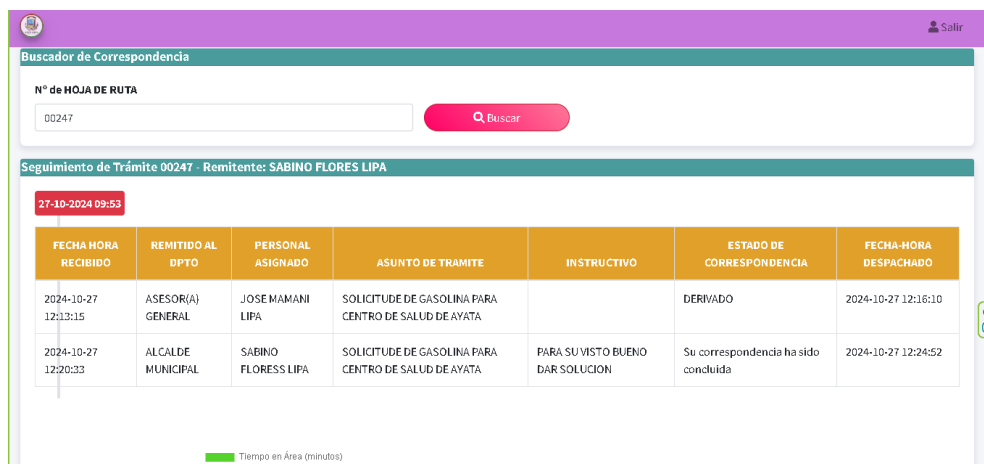
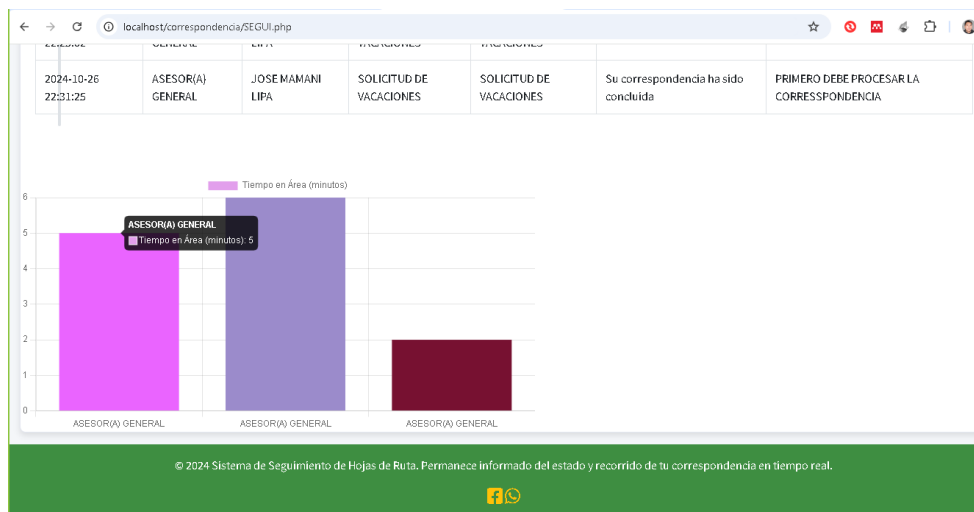


Figura 63.
Caja Negra Control Correspondencia



Después de llevar a cabo la prueba de caja negra se confirmó que esta funciona según lo previsto, garantizando así que se ingresen los datos necesarios para el proceso, esta validación asegura que no se omita información importante, cumpliendo con los requisitos funcionales establecidos para el correcto funcionamiento del registro, de esta manera el sistema responde adecuadamente a las entradas del usuario mostrando alertas.

4.4.3. Pruebas de Estrés

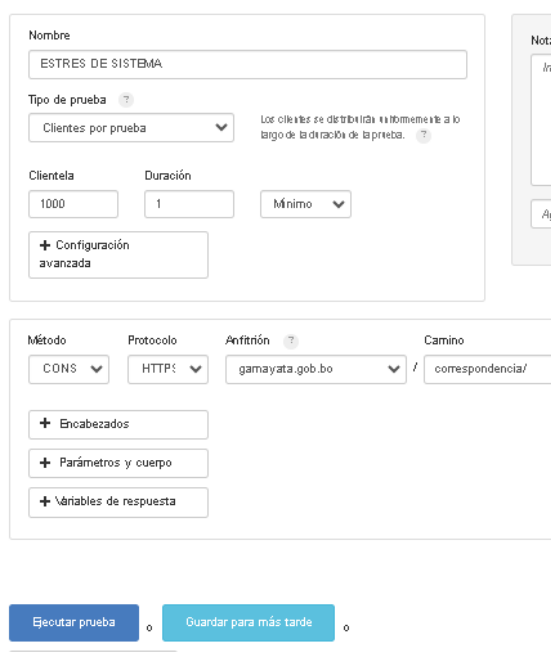
Las pruebas de estrés evalúan el comportamiento de una aplicación Web bajo condiciones de carga extrema llevando el sistema al límite de su capacidad operativa.

Este proceso permite identificar cómo responde la aplicación en diversos escenarios o situaciones límite, durante esta fase se realizaron pruebas de estrés específicamente en el módulo de reservas.

Para esta prueba, se empleó un servidor interno de la institución, caracterizado por ser de media gama en comparación con los servidores de alta gama, cabe mencionar que se hicieron pocas modificaciones en el sistema operativo, conforme a las requerimientos de la institución, a pesar de estas limitaciones, los resultados obtenidos estuvieron en línea con los promedios de rendimiento exigidos para el sistema, demostrando que el servidor y la configuración empleada fueron adecuadas para las necesidades de la prueba.

Figura 64.

Configuración para Prueba de Estres



Nombre
ESTRES DE SISTEMA

Tipo de prueba
Clientes por prueba
Los clientes se distribuirán uniformemente a lo largo de la duración de la prueba.

Clientela
1000

Duración
1
Mínimo

+ Configuración avanzada

Método
CONS

Protocolo
HTTP

Anfitrión
gamayata.gob.bo

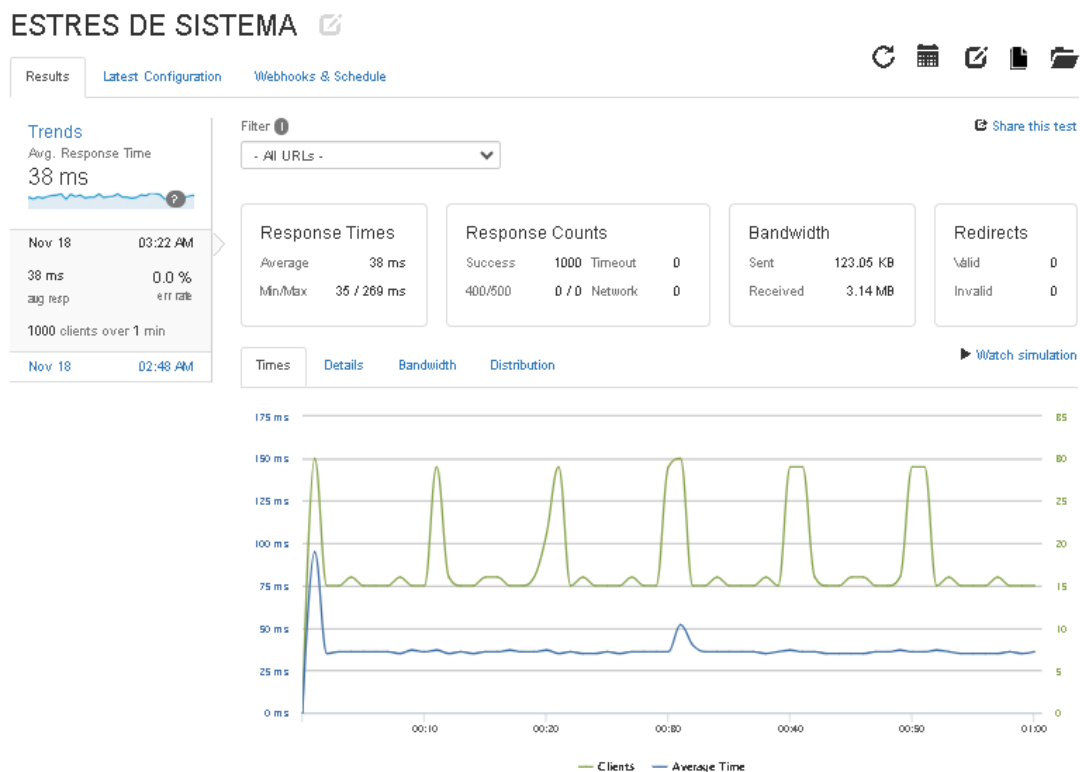
Camino
correspondencia/

+ Encabezados

+ Parámetros y cuerpo

+ Variables de respuesta

Ejecutar prueba Guardar para más tarde

Figura 65.*Evaluando Pruebas de Estrés.*

Respecto a las pruebas de estrés realizadas, se utilizó la herramienta de pruebas de carga, donde los resultados obtenidos muestran que, al realizar la prueba con 1000 usuarios simultáneos durante 1 minuto, el sistema presentó un tiempo de respuesta promedio de 38 ms, con un tiempo máximo de 269 ms. Además, la tasa de error fue de 0%, lo que significa que no hubo fallos ni problemas con las respuestas del servidor.

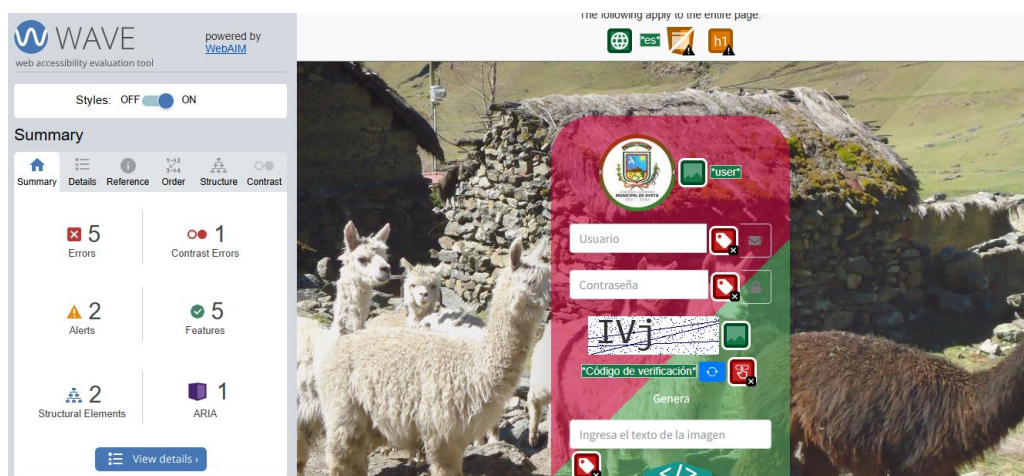
Esto indica que el sistema tiene un rendimiento sólido bajo condiciones de estrés, respondiendo a todas las solicitudes de los usuarios sin interrupciones o errores. Los 1000 clientes pudieron interactuar con el sistema sin que se presentaran problemas de tiempos de espera o sobrecarga, lo cual es una señal de que el servidor es capaz de manejar una carga considerable sin afectar la calidad del servicio.

4.4.4. Pruebas de Accesibilidad

Las pruebas de accesibilidad se enfocaron en garantizar que el sistema web del Gobierno Autónomo Municipal de Ayata sea fácil de usar para todos los usuarios. Se tomó en cuenta el uso de los colores favoritos de los usuarios, asegurando que los, tamaños de letra y contrastes sean apropiados para que el texto sea legible y los elementos de la interfaz sean fáciles de identificar. Además, se verificó que la interfaz sea compatible con diferentes dispositivos y navegadores, permitiendo una navegación fluida en diversas plataformas. El diseño se mantiene limpio y sencillo, asegurando que las funciones principales sean fácilmente localizables y accesibles para todos los usuarios.

Figura 66.

Prueba de Accesibilidad de WAVE



Según la prueba de accesibilidad se ve los resultados, cual está permitido en el rango de lo que se puede tener fallas en el sistema este margen de errores permite la funcionalidad del sistema con normalidad.

4.5. SEGURIDAD DEL SISTEMA – ISO 270001

Uno de los elementos fundamentales a considerar en este proyecto es la aplicación de medidas de seguridad. La norma ISO 27001 se encarga de evaluar y corregir el cumplimiento de dichas medidas, además de promover la mejora constante mediante un conjunto de controles destinados a mitigar el riesgo de incidentes de seguridad en el funcionamiento del sistema dentro de la institución. Para lograrlo, se consideran diversos tipos de seguridad, entre los cuales se incluyen.

4.5.1. Encriptación

Para la seguridad del password estará encriptado con el algoritmo HASH que es una encriptación de alta seguridad para el ingreso al sistema.

Figura 67.

Seguridad del Sistema

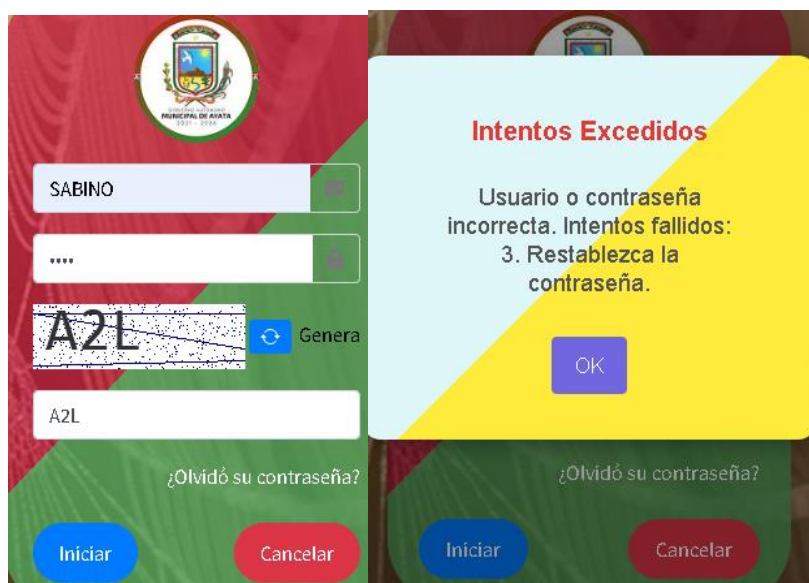
usu_usuario	usu_contra
admin	\$2y\$12\$raqIDtw6GxgHrO9mOeQOD.vZCwe3.WI/VS6BsdNCrnH...
SABINO	\$2y\$12\$raqIDtw6GxgHrO9mOeQOD.vZCwe3.WI/VS6BsdNCrnH...
ANDRES	\$2y\$12\$raqIDtw6GxgHrO9mOeQOD.vZCwe3.WI/VS6BsdNCrnH...
FLORES	\$2y\$12\$raqIDtw6GxgHrO9mOeQOD.vZCwe3.WI/VS6BsdNCrnH...

4.5.2. Autenticación Segura

Se utiliza un sistema de autenticación que requiere nombres de usuario, contraseñas encriptadas y código capcha para acceder al sistema, con diferentes niveles de acceso según los privilegios asignados.

Figura 68.

Autenticación Segura



4.5.3. Gestor de Base de Datos

- En cuanto a la forma de resguardo se realizó los siguientes puntos:
- Los respaldos o back-up de la base de datos se deberá realizar.
- La información en una institución es muy valiosa, por tanto, su resguardo es fundamental.

CAPÍTULO V

CONCLUSIONES Y

RECOMENDACIONES

INGENIERÍA
DE SISTEMAS
UNIVERSIDAD PÚBLICA DE EL ALTO



5. CAPÍTULO V CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

5.1.1. Mm Conclusiones

- Se implementó con éxito un módulo que permite el registro eficiente de las correspondencias recibidas. Este módulo asigna automáticamente un número único de hoja de ruta a cada correspondencia, facilitando su seguimiento desde el momento de su ingreso hasta su resolución final.
- Se desarrolló una interfaz de fácil acceso la búsqueda de correspondencias. Esto permite a los remitentes obtener respuestas rápidas y eficientes, incrementando la satisfacción de los usuarios y mejorando la gestión administrativa.
- El módulo de distribución y organización de correspondencias fue implementado satisfactoriamente. Este clasifica de forma automática las correspondencias según las unidades correspondientes, mejorando significativamente el orden, la organización y la accesibilidad de la información.
- Se implementó un módulo de control de estado que permite a los usuarios conocer en tiempo real el estado y la ubicación de cada correspondencia. Esto ha incrementado la transparencia en la gestión y ha reducido los tiempos de consulta.
- Se generaron reportes estadísticos detallados sobre el estado de las correspondencias, su ubicación en las diferentes unidades y los tiempos de procesamiento de los casos. Estos informes han facilitado la toma de decisiones estratégicas y han contribuido a la mejora continua de los procesos internos.

5.2. RECOMENDACIONES

En base a los resultados obtenidos durante la implementación de este sistema, se formulan las siguientes recomendaciones:

- Es importante capacitar a los usuarios del sistema de manera periódica para asegurar un uso adecuado y eficiente del sistema, especialmente en áreas relacionadas con el registro, seguimiento y gestión de correspondencia.
- Realizar pruebas de rendimiento periódicas para evaluar la capacidad del sistema a medida que crece el volumen de documentos y usuarios, garantizando que el sistema siga siendo eficiente bajo condiciones de alta carga.
- Se recomienda realizar un mantenimiento continuo del sistema, actualizando tanto el software como la infraestructura tecnológica para mantener la seguridad y la eficiencia.
- Refuerzo de la seguridad mediante la implementación de controles adicionales, como la autenticación de dos factores, para proteger la información sensible almacenada en el sistema.
- Continuar con la actualización de la interfaz de usuario, asegurándose de que se mantenga intuitiva y accesible, y evaluando la retroalimentación de los usuarios para hacer ajustes que mejoren aún más la experiencia.

BIBLIOGRAFÍA

- Alarcón, J. (2023). *Cocoma y punto de fusion*.
- Barish, R. A., & Arnold, T. (12 de Septiembre de 2022). *Mordeduras MSD*. Obtenido de <https://www.msdmanuals.com/es/hogar/traumatismos-y-envenenamientos/mordeduras-y-picaduras/mordeduras-de-animales>
- Bartholomew, D. (2014). *libro de MariaDB*. Packt Publishing.
- Brinck, T., & Gergle, D. (2002). *Usability for the Web: Designing Web Sites that Work*. Obtenido de <https://www.uxlumen.com/modelos-de-navegacion/>
- Bustamante, P. (2021). *Estimación de esfuerzo de trabajo y planificación de proyectos de desarrollo con restricciones de recursos*. Obtenido de repositorio UCHILE: <https://repositorio.uchile.cl/>
- Calero, W. (16 de Octubre de 2010). *Ingeniería de software*. Obtenido de Ingeniería de software: <https://ingenieraupoliana.blogspot.com/2010/10/cocoma.html>
- Clutton-Brock. (1999). *Una Historia Natural de los Mamíferos Domesticados*. Prensa de la Universidad de Cambridge.
- Flores, A. (20 de Febrero de 2024). *Ecosistemas*. Obtenido de <https://ecosistemas.win/que-animales-silvestres-y-domesticos/>
- Galvan, H. (24 de Septiembre de 2014). *SlideShare*. Obtenido de SlideShare: <https://es.slideshare.net/slideshow/cocoma-39479541/39479541>
- Gardey, J. (2021). *Definicion.de*. Obtenido de <https://definicion.de/web-2-0/>.
- Gómez, J. (8 de Diciembre de 2013). *Estimación de Costes con COCOMO 81 (II)*. Obtenido de Estimación de Costes con COCOMO 81 (II): <https://www.laboratorioti.com/2013/04/22/estimacion-de-costes-con-cocoma-81-ii/>
- Gonzales, E. (2021). *Desarrollo Web con PHP y MySQL*.

González, E. (2006).

https://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=492:ique-es-php-y-ipara-que-sirve-un-potente-lenguaje-de-programacion-para-crear-paginas-web-cu00803b&catid=70&Itemid=193.

Jaimes, J., & Mitzi, N. (2014). *Modelo de estimación del proyecto Cocomo*.

Lucena, P. (Junio de 2023). *Universidad CESUMA*. Obtenido de Universidad CESUMA:

https://www.cesuma.mx/blog/que-son-los-sistemas-de-informacion-y-por-que-son-necesarios.html#abh_posts

Marrugo, Y. (23 de Junio de 2010). *Modelo Cocomo*. Obtenido de

<https://es.slideshare.net/jedaro/modelo-cocomo-4593779>

Mendoza, L. E. (18 de Marzo de 2022). *Goconqr*. Obtenido de Goconqr:

https://www.goconqr.com/flashcard/35942991/metodologia-extreme-programming-xp#google_vignette

Miss, F. G., & Baena, R. (05 de 06 de 2019). *ENUMED.NET*. Obtenido de ENUMED:

<http://www.eumed.net/rev/ce/2019/norma-iso-eic.html>

Modelo Cocomo. (19 de Julio de 2008). Obtenido de

<https://acevedodelacru.wordpress.com/cocomo-ii/>

Osmosis Latina. (15 de Abril de 2024). *Osmosis Latina*. Obtenido de Osmosis Latina:

<https://www.osmosislatina.com/jmeter/basico.htm>

Parry, C. M. (2008). *Resistencia a los antimicrobianos en Salmonella: Una Perspectiva Global Enfermedades infecciosas clínicas*.

Platzi. (9 de septiembre de 2018). Obtenido de <https://platzi.com/blog/pruebas-esenciales-para-evaluar-el-rendimiento-de-software/>

Pressman, R. (2010). *Ingeniería de software*.

Risso, A., Ruiz Yanzi, M. V., & Volij, C. (12 de Diciembre de 2022). *evidencia.org*. Obtenido

de evidencia.org: <https://www.evidencia.org/index.php/Evidencia/article/view/7049>

Smith, J. S. (1994). *Virus de la Rabia en Microbiología Veterinaria*.

Tenter, A. M. (2000). *Toxoplasma gondii: de los Animales a los Humanos.*" *Revista Internacional de Parasitología*.

Thompson, R. C. (2019). *Zoonoses: Biology and Emerging Issues*.

Vargas, M. S. (25 de Junio de 2017). *Ingeniería basada en modelos*. Obtenido de Ingeniería basada en modelos: <http://marcelosalasvargas.blogspot.com/2017>

Wormser, G. P. (2006). *Guías de práctica de la sociedad de Enfermedades Infecciosas de Estados Unidos para el Tratamiento de la Enfermedad de Lyme*.

Zaptest. (07 de 03 de 2024). *Copyright 2024*. Obtenido de <https://www.zaptest.com/es/pruebas-de-caja-negra-que-son-tipos-procesos-enfoques-herramientas-y-mucho-mas>

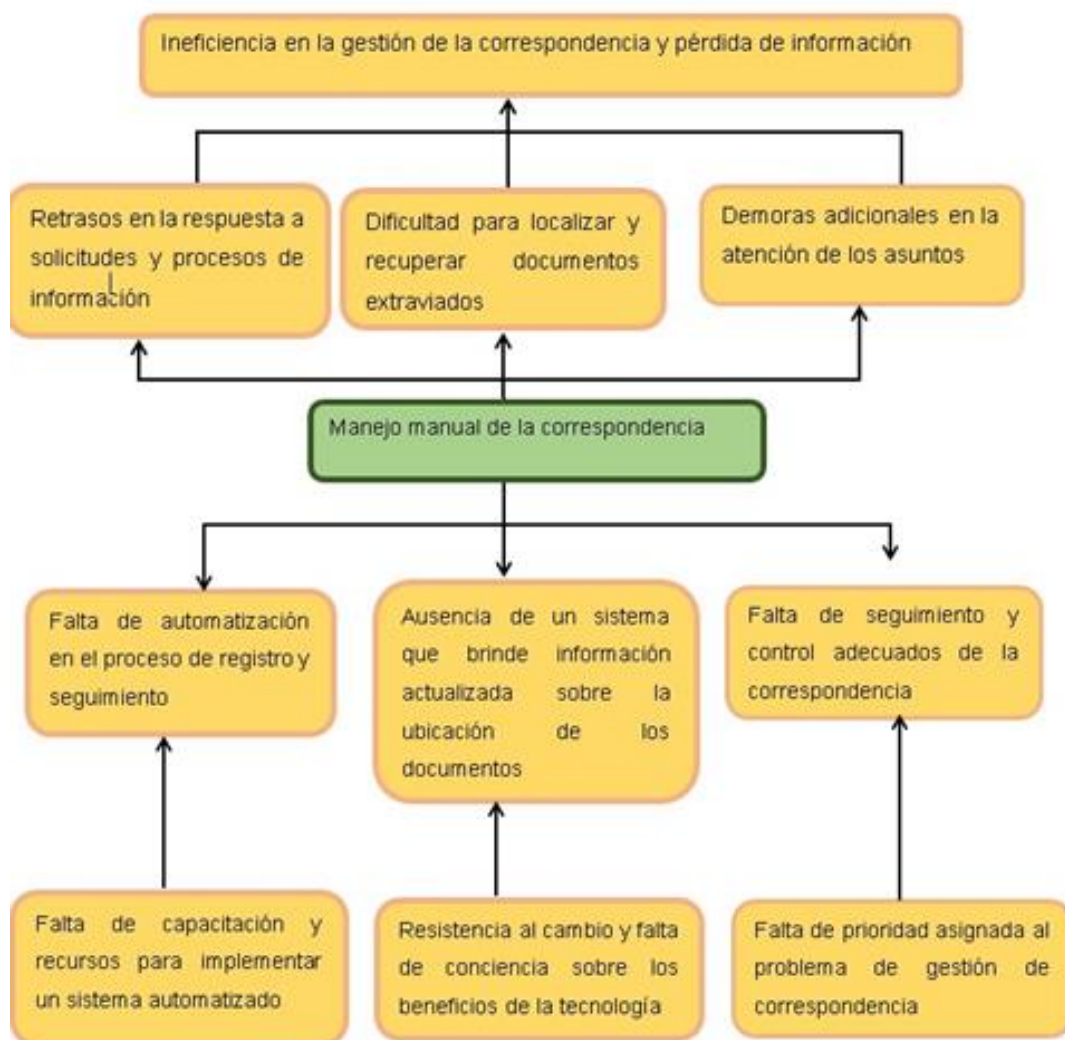
ANEXOS

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

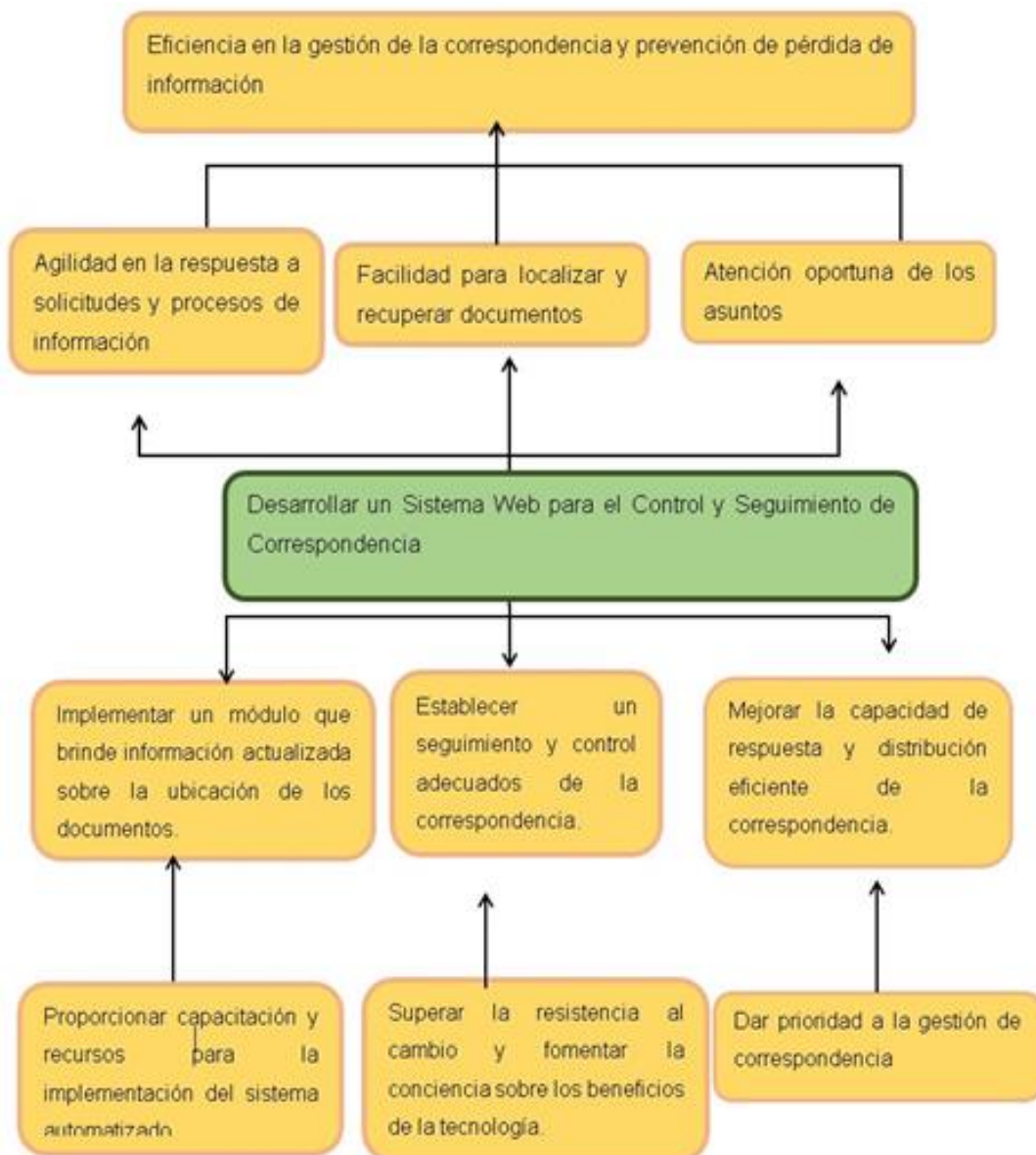


ANEXOS

Árbol de problemas



Árbol de objetivos



AVAL DE CONFORMIDAD
(TUTOR ESPECIALISTA)

El Alto, noviembre de 2024

Señor(a):

LIC.ING, HELEN FANNY SUNTURA ESCOBAR

TUTOR METODOLOGIA

TALLER DE GRADO II

Presente. –

REF, AVAL DE CONFORMIDAD

Distinguido tutor metodológico:

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

TITULO: SISTEMA WEB PARA EL CONTROL Y SEGUIMIENTO DE CORRESPONDENCIA.

CASO: GOBIERNO AUTONOMO MUNICIPAL DE AYATA

MODALIDAD: PROYECTO DE GRADO.

Univ. Sabino Flores Lipa.

Registro Universitario: 11001320.

Cedula de Identidad: 6902847.

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

Atentamente,

Lic. Fredy Alanoca Coareti
TUTOR ESPECIALISTA

AVAL DE CONFORMIDAD
(TUTOR REVISOR)

El Alto, noviembre de 2024

Señor(a):
LIC.ING, HELEN FANNY SUNTURA ESCOBAR
TUTOR METODOLOGIA
TALLER DE GRADO II

Presente. –

REF. AVAL DE CONFORMIDAD

Distinguido tutor metodológico:

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

TITULO: SISTEMA WEB PARA EL CONTROL Y SEGUIMIENTO DE
CORRESPONDENCIA.

CASO: GOBIERNO AUTONOMO MUNICIPAL DE AYATA

MODALIDAD: PROYECTO DE GRADO.

Univ. Sabino Flores Lipa.

Registro Universitario: 11001320.

Cedula de Identidad: 6902847.

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

Atentamente,

M. Sc. Lic. Zara Yujra Cama

TUTOR REVISOR

**AVAL DE CONFORMIDAD
(TUTOR METODOLÓGICO)**

El Alto, noviembre de 2024

Señor:
ING, WILLIAM ROQUE ROQUE
**DIRECTOR DE CARRERA
INGENIERIA DE SISTEMAS**
Presente. –

REF. AVAL DE CONFORMIDAD

Distinguido tutor metodológico:

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

TITULO: SISTEMA WEB PARA EL CONTROL Y SEGUIMIENTO DE
CORRESPONDENCIA.

CASO: GOBIERNO AUTONOMO MUNICIPAL DE AYATA

MODALIDAD: PROYECTO DE GRADO.

Univ. Sabino Flores Lipa.

Registro Universitario: 11001320.

Cedula de Identidad: 6902847.

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

Atentamente,

LIC.ING, HELEN FANNY SUNTURA ESCOBAR
TUTOR METODOLOGICO

TALLER DE GRADO II

MANUAL DE USUARIO ADMINISTRADOR

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



MANUAL DE ADMINISTRADOR

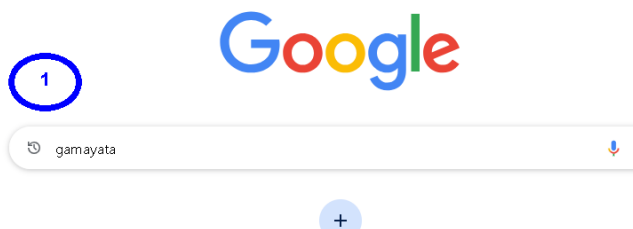
INTRODUCCIÓN.

El presente manual de administración pretende servir de guía para el uso adecuado del sistema de control y seguimiento de correspondencia del Gobierno Autónoma Municipal de Ayata.

OBJETIVO

Brindar información necesaria para poder realizar la administración y configuración del de los módulos del sistema.

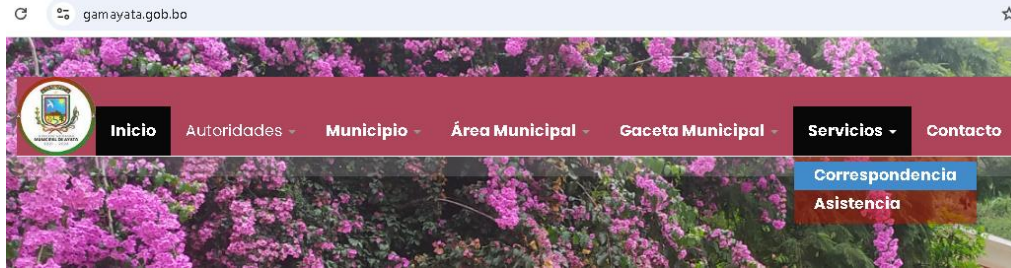
Pso 1 tener internet y un navegador



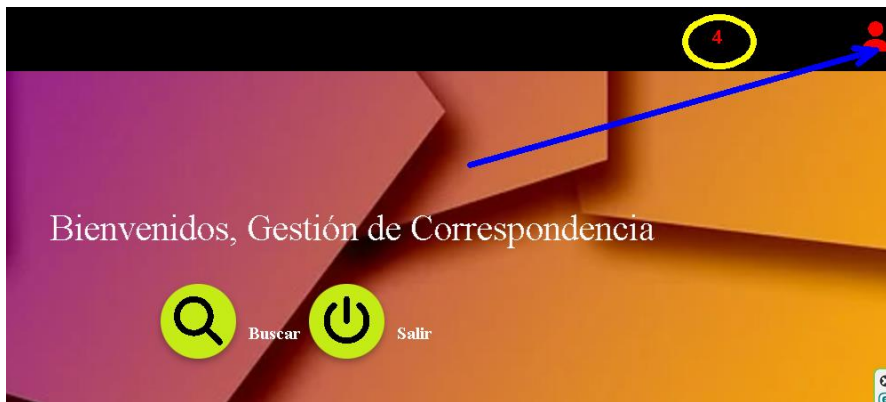
Paso 2 entrar a sitio web.



Paso 3 una vez ingresado a sitio web de municipio ir a servicios y correspondencia



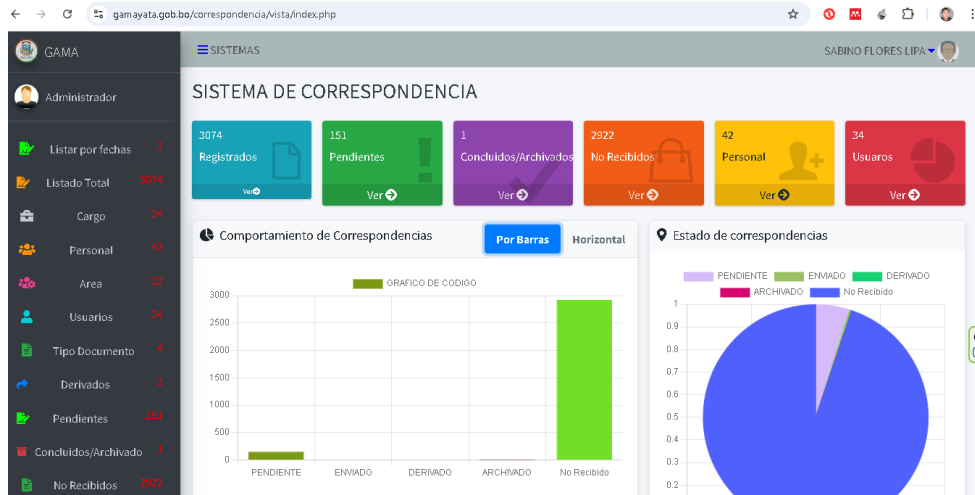
Paso 4 damos click en icono del usuario en la parte superior como indica la flecha



Paso 5 login introducir las credenciales para ingresar al sistema:



Paso 6 principal para el administrador, tiene opciones de consultar por gráficos .actualizar, listar insertar y sacar reportes en PDF Y EXCEL todos los módulos



gestión de usuario: creamos usuario y contraseña por primera vez par el nuevo personal luego puede gestionar para ello deben estar registrados en sistemas

The screenshot shows a 'REGISTRO DE USUARIO' modal form. It contains the following fields and options: 'USUARIO' (text input), 'CONTRASEÑA' (password input), 'PERSONAL' (dropdown menu with 'SABINO FLORES LIPA' selected), 'UNIDAD' (dropdown menu with 'CONCEJO MUNICIPAL' selected), and 'ROL' (dropdown menu with 'REGISTRADOR(A)' selected). At the bottom right, there are 'Close' and 'REGISTRAR' buttons.

Gestion de Personal

gabayata.gob.bo/correspondencia/vista/index.php

GAMA SISTEMAS SABINO FLORES LIPA

PERSONAL DE G.A.M.A GESTIONAR PERSONAL DEL GAMA

Home / Personal

INSE **ACTIVAR** **ESTATUS** **ACCION**

Mostrar 10 registros

Buscar: **BUSCAR**

#	Foto	Nro De Carnet	Nombre Completo	Movil	Email	Direccio	Estatus	Accion
1		690284700	SABINO FLORES LIPA	73012886	flores.onibas@gmail.com	EL ALTO	ACTIVO	
2		6772195	LUIS KAMA CARRIZALES	67890376	SABBINO@HOTMAIL.COM	EL ALTO	ACTIVO	
3		6858670	TEODORA LAURA OJEDA	73045634	TEODORA@GMAIL.COM	CUIBAJA AYATA	ACTIVO	
4		6772220	ELEUTERIO MARAZA QUISPE	74012461	ELUETERIO@GMAIL.COM	ALCALDE	ACTIVO	
5		10013137	NORHA CASTILLO QUISPE	0	HORA12@GMAIL.COM	TECNICA	INACTIVO	

generar listar imprimir hojas de ruta

gabayata.gob.bo/correspondencia/vista/index.php

GAMA SISTEMAS SABINO FLORES LIPA

GENERAR REPORTE POR FECHAS DE CORRESPONDENCIA REGISTRADAS

Listado de Correspondencia

Desde: 18/11/2024 Hasta: 18/11/2024 **Q** **Buscar**

Mostrar 10 registros

Buscar: **BUSCAR**

#	Numero Documento	Fecha en Recepcion	Fecha de Documento	Remite	Asunto	Enviado A:	Cant. Hojas	Ver	Editar	Impri
1	03327	2024-11-18 08:51:10	2024-11-15	ING EFRAIN MAMANI ANDRADE DIRECTOR DE PROYECTOS PRODUCTIVOS DE MEDIO AMBIENTE Y GESTION DE RIESGOS	SOLICITUD DE INFORME PARA ELEVAR /DECLARAR LA LEY AUTONOMICA MUNICIPAL DE DESASTRES NATURALES DEL EVENTO ADVERSO SEQUIA DEL MUNICIPIO DE AYATA	LIC: ELEUTERIO MARAZA QUISPE	10			
2	03328	2024-11-18 08:56:53	2024-11-15	ROLANDO OTOYA OTOYA RESPONSABLE DE ACTIVOS FLUJOS Y ALMACENES	SOLICITUD DE VERIFICACION EN MERCADO VIRTUAL E INICIO DE PROCESO PARA LA ADQUISICION DE PERFORADORAS DE PAPEL PARA EL EJECUTIVO DEL GOBIERNO AUTONOMO MUNICIPAL DE AYATA	LIC: ELEUTERIO MARAZA QUISPE	3			
				SANTIAGO MAMANI LAURA	ENTREGA DE PROVISIONAL Y	LIC:				

Todos los módulos tienen las mismas características insertar ,editar, desactivar/activar generar reportes en pdf y en Excel .

gamayata.gob.bo/correspondencia/avista/mpdf/REPORTE/Correspondencia.php

Correspondencia.php

Gobierno Autónomo Municipal de Ayata
 Ubicación: Provincia Muñecas-La Paz - Bolivia
 Teléfono: 22345678
 Correo: ayata@gmail.com
LISTADO DE CORRESPONDENCIAS

Nº	COD	REMITENTE	FECHA	CELULAR	DIRECCION	ASUNTO
1	03327	ING EFRAIN MAMANI ANDRADE	2024-11-18 08:51:10		DIRECTOR DE PROYECTOS PRODUCTIVOS DE MEDIO AMBIENTE Y GESTION DE RIESGOS	SOLICITUD DE INFORME PARA ELEVAR /DECLARAR LA LEY AUTONOMICA MUNICIPAL DE DESASTRES NATURALES DEL EVENTO ADVERSO SEQUIA DEL MUNICIPIO DE AYATA
						SOLICITUD DE

Generación de hojas de ruta de las correspondencia registradas por la secretaria cada módulo tiene la opción de generar listo de categorías que desea

gamayata.gob.bo/correspondencia/avista/mpdf/REPORTE/ticket_tramite.php?codigo=03327#zoom=100

ticket_tramite.php

Gobierno Autónomo Municipal de Ayata
 Provincia Muñecas-La Paz - Bolivia

La Paz: 03327

HOJA DE RUTA

Recibido en Recepcion: 18/11/24 08:51:10 Fecha del Documento: 15/11/24

Remite ING EFRAIN MAMANI ANDRADE Cargo DIRECTOR DE PROYECTOS PRODUCTIVOS DE MEDIO AMBIENTE Y GESTION DE RIESGOS

ASUNTO: SOLICITUD DE INFORME PARA ELEVAR /DECLARAR LA LEY AUTONOMICA MUNICIPAL DE DESASTRES NATURALES DEL EVENTO ADVERSO SEQUIA DEL MUNICIPIO DE AYATA

ENVIADO A: LIC: ELEUTERIO MARAZA QUISPES CANT. HOJAS 10

A	SU APROBACION	F	VISITARME
B	SU INFORMACION	G	DIFUNDIR
C	TOMAR NOTA/TRANSCRIBIR	H	CONFORME/DAR CURSO
D	SU REVISION	I	PREPARAR RESPUESTA
E	PROGRAMAR ENTREVISTA	J	REGISTRO/ARCHIVO

INGRESO-HORA DE RECEPCION	REMITIR AL DPTO:	SAIDA-HORA DESPACHADO

Generar en excel

The screenshot shows an Excel spreadsheet with the following content:

- Row 1: Gobierno Autónomo Municipal de Ayata
- Row 2: Ubicación: Provincia Muñecas
- Row 3: Teléfono: 22345678
- Row 4: Correo: ayata@gmail.com
- Row 5: LISTADO DE CORRESPONDENCIAS
- Row 6: (Empty)
- Row 7: (Empty)
- Row 8: Header row with columns: N°, COD, REMITENTE, FECHA, CELULAR DIRECCION, ASUNTO
- Row 9: Data row: 1, 03327, ING EFRAIN MAMANI ANDRADE, 2024-11-18 08:51:10, DIRECTOR DE PROYECTOS PRODUCTIVOS DE MEDIO AMBIENTE Y GESTION DE RIESGOS, SOLICITUD DE INFORME PARA ELEVAR /DECLARAR LA LEY AUTONOMICA MUNICIPAL DE DESASTRES NATURALES DEL EVENTO ADVERSO SEQUIA DEL MUNICIPIO DE AYATA
- Row 10: Data row: 9, (Empty), (Empty), (Empty), (Empty), SOLICITUD DE VERIFICACION EN MERCADO VIRTUAL E INICIO DE PROCESO

Puede cambiar su contraseña o cerrar la sesión

The screenshot shows a web application interface with the following elements:

- Header: GAMA, SISTEMAS, SABINO FLORES LIMA
- Left sidebar: Administrador, Listar por fechas (3), Listado Total (3074), Cargo (34), Personal (42), Area (33), Usuarios (34)
- Main content: Listado de Correspondencia
- Filters: Desde: 18/11/2024, Hasta: 18/11/2024, Buscar
- Actions: Cerrar Sesión, Cambiar Contraseña, Cancelar
- Table: Mostrando 10 registros. Header: #, Numero Documento, Fecha en Recepcion, Fecha de Documento, Remite, Asunto, Enviado A:, Cant. Hojas, Ver, Editar, Impri.
- Table Row 1: 1, 03327, 2024-11-18 08:51:10, 2024-11-15, ING EFRAIN MAMANI ANDRADE DIRECTOR DE PROYECTOS PRODUCTIVOS DE MEDIO AMBIENTE Y, SOLICITUD DE INFORME PARA ELEVAR /DECLARAR LA LEY AUTONOMICA MUNICIPAL DE DESASTRES NATURALES DEL EVENTO ADVERSO SEQUIA DEL, LIC: ELEUTERIO MARAZA, 10, Ver, Editar, Impri

Opción de actualizar contraseña

The screenshot shows a password change form with the following fields and buttons:

- Contraseña Actual: contraseña actual
- Nueva Contraseña: nueva contraseña
- Repetir Contraseña: repetir la nueva contraseña
- Buttons: Modificar, Cerrar

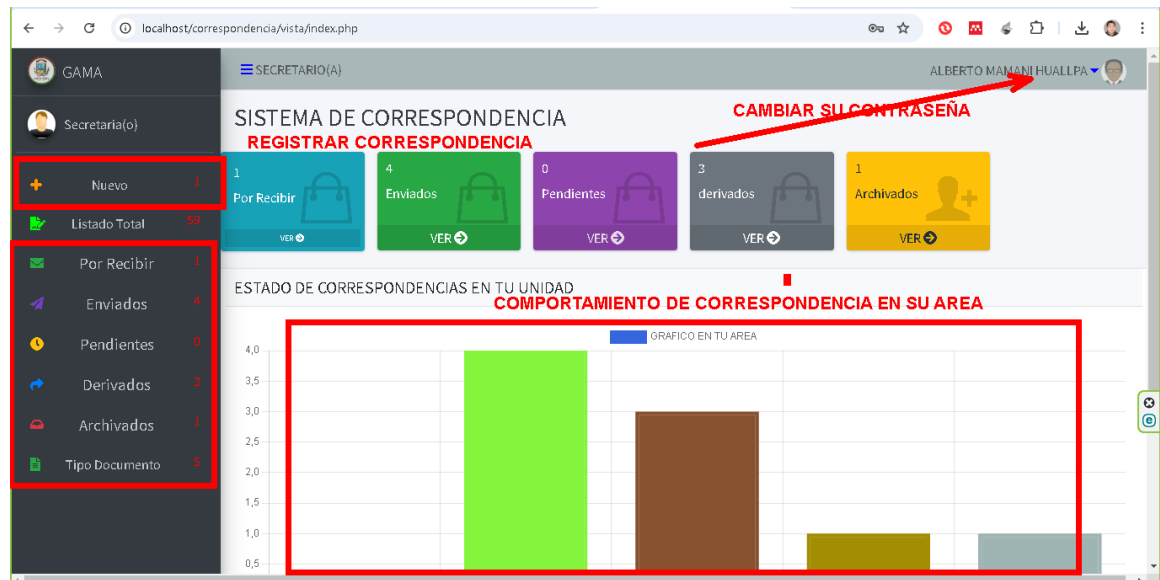
MANUAL DE USUARIO

SECRETARIA

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



para secretaria sigue los mismos pasos desde paso 1 hasta paso 4 del manual de administrador



Formulario de registro de correspondencia

Datos de Remitente y Documento			
Nombre	Apellido Paterno	Apellido Materno	Celular
Cargo	Fecha de Documento (dd/mm/aaaa)		
Asunto de Trámite	Enviado A (Lic: Eleuterio Maraza Quispe)	Nº Hojas	
Tipo De Documento	Area destino (RESPONSABLE DE TRANSPAREN...)		
Cerrar			REGISTRAR

Una vez registrado automáticamente le genera la siguiente Hoja de ruta.

Plantilla de hoja de ruta para adjuntar a cada correspondencia que entra

Gobierno Autónomo Municipal de Ayata
Provincia Multicéntrica - La Paz - Bolivia

La Paz: 03327

HOJA DE RUTA

Recibido en Recepción: 18/11/24 08:51:10 Fecha del Documento: 15/11/24

Remite: ING EFRAIN MAMANI ANDRADE Cargo DIRECTOR DE PROYECTOS PRODUCTIVOS DE MEDIO AMBIENTE Y GESTION DE RIESGOS

ASUNTO: SOLICITUD DE INFORME PARA ELEVAR /DECLARAR LA LEY AUTONOMICA MUNICIPAL DE DESASTRES NATURALES DEL EVENTO ADVERSO SEQUIA DEL MUNICIPIO DE AYATA

ENVIADO A: LIC: ELEUTERIO MARAZA QUISPPE CANT. HOJAS 10

A	SU APROBACION	F	VISITARME
B	SU INFORMACION	G	DIFUNDIR
C	TOMAR NOTA/TRANSCRIBIR	H	CONFORME/DAR CURSO
D	SU REVISION	I	PREPARAR RESPUESTA
E	PROGRAMAR ENTREVISTA	J	REGISTRO/ARCHIVO

INGRESO-HORA DE RECEPCION: REMITIR AL DPTO. SALIDA-HORA DESPACHADO

INSTRUCCIONES:

Listado por fechas buscar o generar reportes sobre correspondencia

SECRETARIO(A) DARLENE ITURRI VALERA

Listado de Correspondencia

Desde: 18/11/2024 Hasta: 18/11/2024

Mostrar 10 registros

#	Numero Documento	Fecha en Recepcion	Fecha de Documento	Remite	Asunto	Enviado A:	Cant. Hojas	Ver	Editar	Impri
1	03327	2024-11-18 08:51:10	2024-11-15	ING EFRAIN MAMANI ANDRADE DIRECTOR DE PROYECTOS PRODUCTIVOS DE MEDIO AMBIENTE Y GESTION DE RIESGOS	SOLICITUD DE INFORME PARA ELEVAR /DECLARAR LA LEY AUTONOMICA MUNICIPAL DE DESASTRES NATURALES DEL EVENTO ADVERSO SEQUIA DEL MUNICIPIO DE AYATA	LIC: ELEUTERIO MARAZA QUISPPE	10			
2	03328	2024-11-18 08:56:53	2024-11-15	ROLANDO OTOYA OTOYA RESPONSABLE DE ACTIVOS FLOS Y ALMACENES	SOLICITUD DE VERIFICACION EN MERCADO VIRTUAL E INICIO DE PROCESO PARA LA ADQUISICION DE PERFORADORAS DE PAPEL PARA EL EJECUTIVO DEL GOBIERNO AUTONOMO MUNICIPAL DE AYATA	LIC: ELEUTERIO MARAZA QUISPPE	3			
				SANTIAGO MAMANI LAURA	ENTREGA DE PROVISIONAL Y	LIC:				

Puede recibir correspondencia derivados de otras unidades ver estado de correspondencia la correspondencia

GAMA SECRETARIO(A) DARLENE ITURRI VALERA

HOJAS DE RUTAS PENDIENTES

Agrupar NuevoM

Mostrar 10 registros Buscar:

#	N° HR	Remitente	Area Origen:	Para su:	Cent. Hojas	Recibido	Estado	Plazo	Accion
	00258	LIC. JOSE MANUEL IRIGOYEN CABRERA	SECRETARIO(A) - DARLENE ITURRI VALERA	SOLICITUD DE PAGO DE SUELDO DEL MES DE ENERO 2024	6	RECIBIDO 02/02/2024 16:14:37	PENDIENTE	¡Tiempo agotado! Tienes -282 días con -19 horas y -17 minutos para responder o archivar la correspondencia	
	00259	ING. MARCIAL DELGADILLO VARGAS	SECRETARIO(A) - DARLENE ITURRI VALERA	INFORME Y SOLICITUD DE PAGO DEL MES DE ENERO 2024	5	RECIBIDO 02/02/2024 16:22:29	PENDIENTE	¡Tiempo agotado! Tienes -282 días con -19 horas y -9 minutos para responder o archivar la correspondencia	

Puede administrar su contraseña

Contraseña Actual Nuevo Contraseña Repetir Contraseña

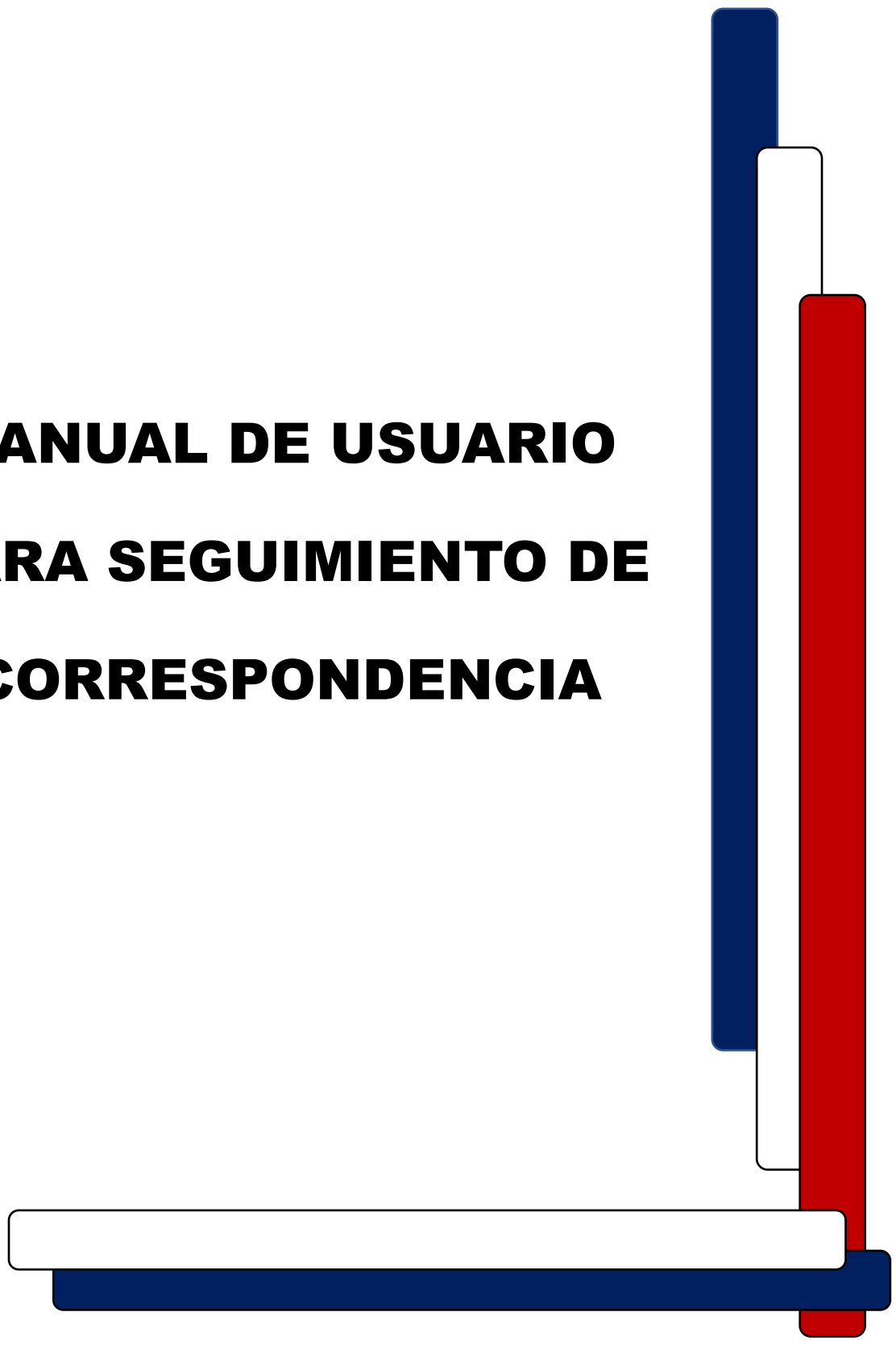
Contraseña actual nueva contraseña repetir la nueva contraseña

Modificar Cerrar

Puede visualizar el comportamiento de correspondencia por mes en un gráfico.



**MANUAL DE USUARIO
PARA SEGUIMIENTO DE
CORRESPONDENCIA**

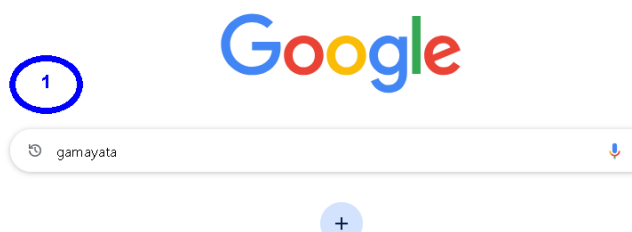


Para ver el comportamiento de correspondencia registradas deben seguir los siguientes pasos.

Primero debes tener a la mano el número de hoja de ruta de la correspondencia que se genera automáticamente cuando registramos al sistema.

los módulos del sistema.

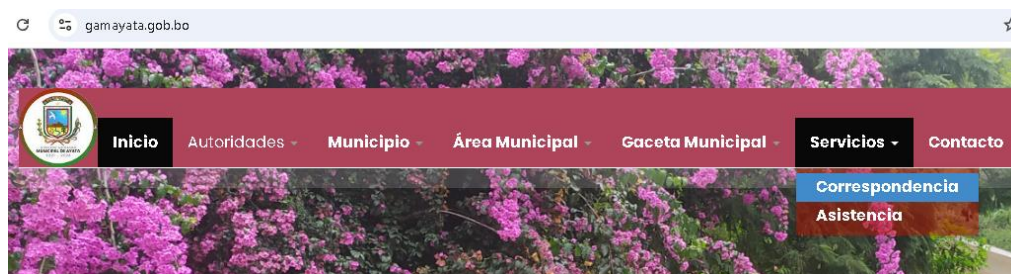
Pso 1 tener internet y un navegador



Paso 2 entrar a sitio web.



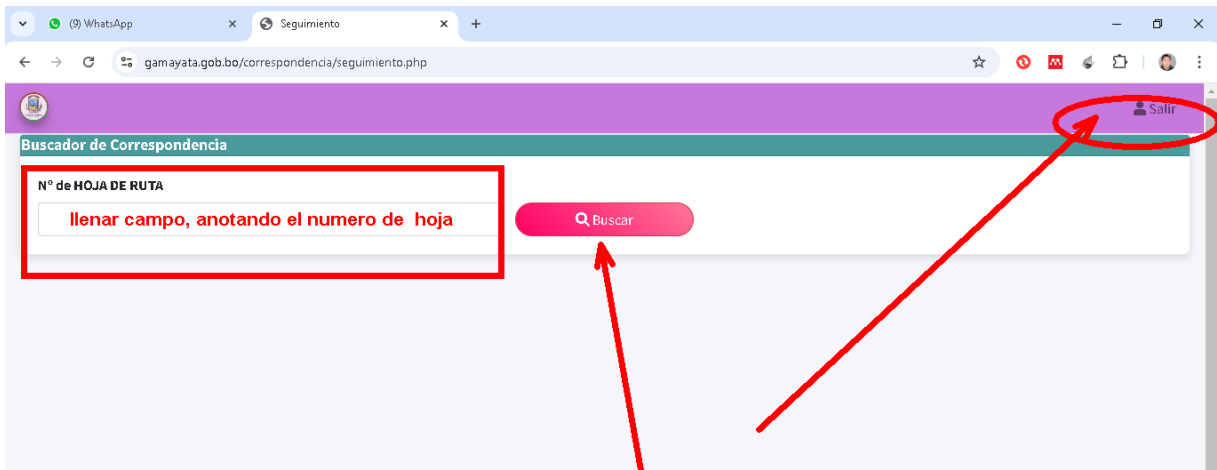
Paso 3 una vez ingresado a sitio web de municipio ir a servicios y correspondencia



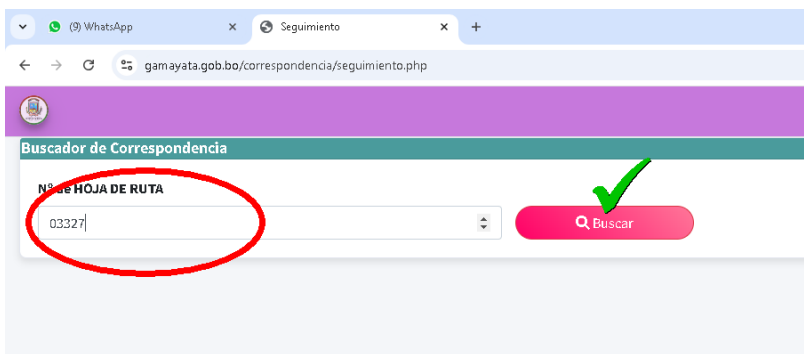
Paso 4 damos click en icono del usuario en la parte superior como indica la flechaha



Paso 5 dar click en icono buscar.



En este campo anotamos el numero de hoja de ruta como sigue en la siguiente imagen



En esta modulo podemos ver el historial de hoja de ruta por donde va pasando el estado que esta, la unidad que se encuentra con personal asignado de esa are y fechas de comportamiento

Buscador de Correspondencia

N° de HOJA DE RUTA

03327

Buscar

Salir

Seguimiento de Trámite 03327 - Remitente: ING EFRAIN MAMANI ANDRADE

18-11-2024 08:51

FECHA HORA RECIBIDO	REMITIDO AL DPTO	PERSONAL ASIGNADO	ASUNTO DE TRÁMITE	INSTRUCTIVO	ESTADO DE CORRESPONDENCIA	FECHA-HORA DESPACHADO
NO RECIBIO	RESPONSABLE DE TRANSPARENCIA Y LUCHA CONTRA LA COR	VICTOR VILLARROEL ALVAREZ	SOLICITUD DE INFORME PARA ELEVAR /DECLARAR LA LEY AUTONOMICA MUNICIPAL DE DESASTRES NATURALES DEL EVENTO ADVERSO SEQUIA DEL MUNICIPIO DE AYATA	null	No Recibido	PRIMERO DEBE PROCESAR LA CORRESPONDENCIA

Luego podemos ir a opción salir para volver a pantalla principal

Gobierno Autónomo Municipal

Inicio Autoridades Municipio Área Municipal Gaceta Municipal Servicios Contacto

Inicio

10:53 18/11/2024