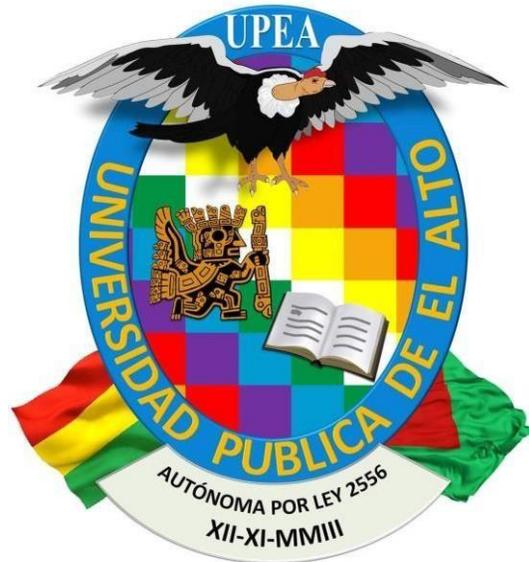


UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

“SISTEMA DE MONITOREO PARA EL PROCESO DE CONTROL DE PROYECTOS VIA WEB”

CASO: GOBIERNO AUTÓNOMO MUNICIPAL DE BATALLAS

Para Optar al Título de Licenciatura en Ingeniería de Sistemas
MENCIÓN: GESTIÓN Y PRODUCCIÓN

Postulante: Jorge Luis Mamani Sonco
Tutor Metodológico: M. Sc. Ing. Maricel Yarari Mamani
Tutor Revisor: Ing. Gabriel Reynaldo Sirpa Huayhua
Tutor Especialista: M. Sc. Ing. Hernán Quispe Conurana

EL ALTO – BOLIVIA

2023

DEDICATORIA

A Dios, Tú has sido mi guía constante y mi fuerza en cada paso de mi vida. Me has bendecido con innumerables dones y oportunidades, y te estoy eternamente agradecido por tu amor incondicional y por nunca abandonarme.

A mi amada esposa, no hay palabras suficientes para describir cuánto aprecio y valoro tu amor, apoyo y comprensión. Tú has estado a mi lado en cada alegría y en cada desafío.

A mi padre Isidro, tú has sido mi modelo a seguir y mi guía en la vida. Tus enseñanzas, tu sabiduría y tu apoyo incondicional han sido fundamentales para mi crecimiento personal y profesional.

A mi madre Dominga, tú eres mi fuente de amor incondicional y mi mayor defensora. Tu cariño, tus cuidados y tu sacrificio han sido la base de todo lo que he logrado.

A mis Hermanos, ustedes son mi apoyo incondicional y mis aliados en la vida.

A mis Tutores por su invaluable guía y por compartir conmigo su vasto conocimiento durante todo el proceso de desarrollo y finalización del mismo.

AGRADECIMIENTO

A Dios, En primer lugar, quiero elevar mi más sincero agradecimiento hacia Ti. Tú has sido mi guía constante en cada paso de mi vida, y no puedo expresar con palabras cuánto aprecio tu amor incondicional y tu protección. Has sido mi fortaleza en los momentos difíciles y mi fuente de inspiración en los momentos de alegría. Gracias por estar siempre presente, por escuchar mis oraciones y por bendecirme con tus infinitas bendiciones. Mi gratitud hacia Ti es eterna.

A mi esposa Aida Quispe Laruta, deseo expresar mi más profundo agradecimiento por tu amor incondicional y tu apoyo inquebrantable.

A mis padres, Isidro Mamani Calcina y Dominga Sonco de Mamani, Quiero expresar mi profundo agradecimiento por todo lo que han hecho por mí a lo largo de mi vida. Gracias por su amor incondicional, su dedicación y su sacrificio.

A mis hermanos, Juan, José Antonio y Adrián, Quiero expresar mi más sincero agradecimiento amistad y apoyo incondicional

A mis tutores M.Sc.Ing. Maricel Yarari Mamani, M. Sc. Ing. Hernán Quispe Conurana e Ing. Gabriel Reynaldo Sirpa Huayhua, expresarles mi agradecimiento por su apoyo, colaboración, paciencia para concluir el presente trabajo de grado.

RESUMEN

En la actualidad las tecnologías de información, son una pieza fundamental para el desarrollo de una institución ya sea pública o privada. En el presente proyecto de grado veremos todo el proceso para llevar a cabo la implementación del proyecto.

El presente documento se divide por capítulos, que lo describiremos a continuación:

En el Capítulo I, veremos una breve introducción del documento, los antecedentes de la institución, los antecedentes afines al proyecto de grado, tanto internacionales, nacionales y locales, los planteamientos de problemas donde veremos la problemática que nos llevó al desarrollo e implementación del presente proyecto, los objetivos a realizarse para la implementación del presente proyecto, la justificación, la metodología que se empleará, las herramientas de desarrollo, entre otros puntos importantes.

En el Capítulo II, veremos el marco teórico, en donde se revisa, conceptos sobre los sistemas, ingeniería de requerimientos, ingeniería de software, la metodología, herramientas de desarrollo de software, las métricas de calidad, estimación de costos, seguridad del software. La conceptualización nos ayudará a contextualizar el trabajo y a establecer la conexión entre el proyecto actual y el conocimiento existente, contribuyendo así al avance del campo de estudio.

En el Capítulo III, veremos la aplicación de la metodología UWE.

En el Capítulo IV, veremos las métricas de calidad y la seguridad del software.

En el capítulo V, se hace la conclusión del trabajo respondiendo a cada uno de los objetivos específicos planteados y se remienda sobre el uso del sistema y los módulos que se podrán adicionar en una siguiente versión del sistema.

ÍNDICE GENERAL

| | |
|--|----------|
| CAPITULO I | 1 |
| MARCO PRELIMINAR | 1 |
| 1. MARCO PRELIMINAR | 2 |
| 1.1 INTRODUCCIÓN | 2 |
| 1.2 ANTECEDENTES | 3 |
| 1.2.1 Antecedentes Institucionales | 3 |
| 1.2.1.1. Visión..... | 3 |
| 1.2.1.2. Misión | 3 |
| 1.2.2 Antecedentes afines al proyecto de grado | 4 |
| 1.2.2.1. Antecedentes Internacionales | 4 |
| 1.2.2.2. Antecedentes Nacionales..... | 4 |
| 1.2.2.2. Antecedentes Locales | 5 |
| 1.3 PLANTEAMIENTO DEL PROBLEMA | 5 |
| 1.3.1 Problema Principal | 6 |
| 1.3.2 Problemas secundarios | 7 |
| 1.3.3 Formulación del problema | 8 |
| 1.4 OBJETIVOS | 8 |
| 1.4.1 Objetivo General | 8 |
| 1.4.2 Objetivos Específicos | 8 |
| 1.5 JUSTIFICACIÓN | 9 |
| 1.5.1 Justificación Técnica | 9 |
| 1.5.2 Justificación Económica | 9 |
| 1.5.3 Justificación Social | 10 |
| 1.6 METODOLOGÍA..... | 10 |
| 1.6.1 Metodología de Desarrollo | 10 |
| 1.6.2 Métricas de Calidad 9126..... | 11 |
| 1.6.3 Costos | 11 |
| 1.6.4 Seguridad | 12 |
| 1.7 METODOS DE RECOLECCIÓN DE DATOS..... | 12 |
| 1.7.1 Técnicas de Investigación | 12 |

| | |
|--|-----------|
| 1.7.1.1. La entrevista..... | 12 |
| 1.7.1.2. Encuesta y cuestionarios..... | 12 |
| 1.8 HERRAMIENTAS DE DESARROLLO..... | 12 |
| 1.8.1 herramientas de desarrollo Back End | 13 |
| 1.8.1.1. Php..... | 13 |
| 1.8.1.2. Laravel 10..... | 13 |
| 1.8.1.3. MariaBD | 13 |
| 1.8.2 herramientas de desarrollo Front End | 13 |
| 1.8.2.1. Bootstrap | 13 |
| 1.8.2.2. HTML5..... | 14 |
| 1.8.2.3. CSS3..... | 14 |
| 1.8.2.3. JavaScript..... | 14 |
| 1.9 LÍMITES Y ALCANCES..... | 14 |
| 1.9.1 Alcances..... | 15 |
| 1.9.2 Limites | 15 |
| 1.10 APORTES | 16 |
| CAPITULO II | 17 |
| MARCO TEÓRICO..... | 17 |
| 2. MARCO TEÓRICO..... | 18 |
| 2.1 SISTEMA..... | 18 |
| 2.2 MONITOREO | 19 |
| 2.3 PROCESO | 20 |
| 2.4 SISTEMA WEB | 21 |
| 2.5 INTERNET | 22 |
| 2.6 INGENIERÍA DE REQUERIMIENTOS | 22 |
| 2.6.1 Concepción | 24 |
| 2.6.2 Indagación..... | 25 |
| 2.6.3 Elaboración | 25 |
| 2.6.4 Negociación..... | 26 |
| 2.6.4 Especificación | 27 |
| 2.6.5 Validación..... | 27 |

| | |
|--|----|
| 2.6.6 Administracion de los requerimientos..... | 28 |
| 2.7 INGENIERIA DE SOFTWARE | 28 |
| 2.7.1 Proceso del desarrollo del software | 32 |
| 2.7.2 Modelos de proceso prescriptivo..... | 34 |
| 2.7.2.1. Modelo de la cascada..... | 34 |
| 2.7.2.2. Modelo de proceso incremental | 35 |
| 2.7.2.3. Modelo de proceso evolutivo..... | 35 |
| 2.8 METODOLOGÍA..... | 36 |
| 2.8.1 Metodología UWE | 36 |
| 2.8.1.1. Características de UWE | 37 |
| 2.8.1.2. Actividades de modelado UWE | 38 |
| 2.8.1.2.1. Modelo de caso de uso | 38 |
| 2.8.1.2.2. Modelo conceptual | 39 |
| 2.8.1.2.3. Modelo de navegación | 40 |
| 2.8.1.2.4 Modelo de presentación | 43 |
| 2.9 HERRAMIENTAS DE DESARROLLO DEL SOTFWARE | 44 |
| 2.9.1 Herramientas de desarrollo front end..... | 44 |
| 2.9.1.1. Html 5..... | 44 |
| 2.9.1.2. Css 3 | 45 |
| 2.9.1.3. Javascript | 45 |
| 2.9.1.4. Bootstrap 5 | 46 |
| 2.9.2 herramientas de desarrollo back end | 46 |
| 2.9.2.1. Php..... | 46 |
| 2.9.2.2. MariaDB | 47 |
| 2.9.2.3. Laravel 10..... | 47 |
| 2.10 MÉTRICAS DE CALIDAD | 48 |
| 2.10.1 Calidad del Software | 48 |
| 2.10.2 Norma ISO/IEC 9126..... | 50 |
| 2.10.2.1. Calidad interna o externa | 51 |
| 2.11 ESTIMACIÓN DE COSTOS PARA EL DESARROLLO DEL PROYECTO..... | 53 |
| 2.11.1. El modelo COCOMO II | 53 |

| | |
|--|-----------|
| 2.12 SEGURIDAD DEL SOFTWARE | 55 |
| 2.12.1 Norma ISO/IEC 27001 | 56 |
| CAPÍTULO III | 58 |
| MARCO APLICATIVO..... | 58 |
| 3. MARCO APLICATIVO | 59 |
| 3.1 CAPTURA, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS..... | 59 |
| 3.1.1 Obtención de requerimientos | 59 |
| 3.1.2 Identificación de actores..... | 60 |
| 3.3 INGENIERIA DE REQUERIMIENTOS | 61 |
| 3.3.1 Requerimientos funcionales | 62 |
| 3.3.2 Requerimientos No funcionales..... | 64 |
| 3.4 DESCRIPCIÓN DE ACTORES | 65 |
| 3.5 DIAGRAMA DE CASOS DE USO | 66 |
| 3.5.1 Caso de uso General del sistema | 66 |
| 3.5.2 Caso de Uso – Ingreso al Sistema | 68 |
| 3.5.3 Caso de Uso – Administración de Usuarios | 69 |
| 3.5.4 Caso de Uso – Administración de Roles | 71 |
| 3.5.5 Caso de Uso – Monitoreo de Proyectos | 72 |
| 3.5.6 Caso de Uso – Directorio de Empresas | 74 |
| 3.5.8 Caso de Uso – Gestión de Recursos | 76 |
| 3.5.9 Caso de Uso – Avance del cronograma de actividades | 77 |
| 3.5.10 Caso de Uso – Cronograma del Proyecto | 79 |
| 3.5.11 Caso de Uso – Tablero de Indicadores | 80 |
| 3.5.12 Caso de Uso – Cuenta | 81 |
| 3.5.13 Caso de Uso – Proyectos Aprobados..... | 83 |
| 3.5.14 Caso de Uso – Gestión de Poa | 84 |
| 3.5.15 Caso de Uso – Actividades | 86 |
| 3.6 DIAGRAMA DE CLASES | 87 |
| 3.7 MODELO DE BASE DE DATOS | 89 |
| 3.8 MODELO DE NAVEGACIÓN | 90 |
| 3.8.1 Modelo de Navegación: Administrador..... | 90 |

| | |
|---|------------|
| 3.8.2 Modelo de Navegación: Supervisor de Proyectos | 91 |
| 3.8.3 Modelo de Navegación: Alcalde | 91 |
| 3.8.4 Modelo de Navegación: Secretaria | 92 |
| 3.8.5 Modelo de Navegación: Dirigente | 92 |
| 3.9 MODELO DE PRESENTACIÓN..... | 93 |
| 3.9.1 Modelo de Presentación: LOGIN (Inicio de sesión)..... | 93 |
| 3.9.2 Modelo de Presentación: Página principal | 93 |
| 3.9.3 Modelo de Presentación: Administración de usuarios..... | 94 |
| 3.9.4 Modelo de Presentación: Administración de roles..... | 95 |
| 3.9.5 Modelo de Presentación: Administración de recursos..... | 95 |
| 3.9.6 Modelo de Presentación: Directorio de empresas | 96 |
| 3.9.7 Modelo de Presentación: Monitoreo de proyectos | 96 |
| 3.9.8 Modelo de Presentación: Cronograma de actividades | 97 |
| 3.9.9 Modelo de Presentación: Avance de actividades | 97 |
| 3.9.10 Modelo de Presentación: Dashboard | 98 |
| 3.9.11 Modelo de Presentación: Gestión de Poa | 98 |
| 3.9.12 Modelo de Presentación: Proyectos Aprobados..... | 99 |
| 3.10 IMPLEMENTACIÓN DEL SISTEMA | 99 |
| 3.10.1 Interfaz de Inicio de Sesión | 99 |
| 3.10.2 Funcionalidad General | 100 |
| 3.10.3 Módulos que Integran el Sistema | 101 |
| 3.11 PRUEBAS DE SOFTWARE | 106 |
| 3.11.2 Pruebas de Caja Negra | 106 |
| 3.11.1 Pruebas de Caja Blanca..... | 110 |
| CAPÍTULO IV | 113 |
| MÉTRICAS DE CALIDAD, SEGURIDAD Y ESTIMACIÓN DE COSTOS..... | 113 |
| 4.1 MÉTRICAS DE CALIDAD | 114 |
| 4.1.1 Norma ISO/IEC 9126..... | 114 |
| 4.1.1.1. Funcionalidad | 114 |
| 4.1.1.2. Confiabilidad..... | 119 |
| 4.1.1.3. Mantenibilidad | 120 |

| | |
|--|------------|
| 4.1.1.4. Portabilidad | 122 |
| 4.1.1.5. Usabilidad..... | 123 |
| 4.2 SEGURIDAD DEL SOFTWARE | 125 |
| 4.2.1 Seguridad Física | 125 |
| 4.2.2 Seguridad Lógica | 126 |
| 4.3 ESTIMACION DE COSTOS PARA EL DESARROLLO DEL SOFTWARE | 131 |
| 4.3.1 Costo del desarrollo del software | 132 |
| 4.3.2 Costo de elaboración de proyecto..... | 135 |
| 4.3.3 Costo total del proyecto..... | 136 |
| CAPITULO V | 137 |
| CONCLUSIONES Y RECOMENDACIONES | 137 |
| 5.1 CONCLUSIONES..... | 138 |
| 5.2 RECOMENDACIONES | 139 |
| BIBLIOGRAFÍA..... | 141 |
| ANEXOS..... | 144 |

ÍNDICE DE TABLAS

| | |
|--|-----|
| Tabla 1 <i>Ecuaciones por tipo de modelo cocomo II: básico e intermedio</i> | 54 |
| Tabla 2 <i>Constante de coste modelo básico</i> | 55 |
| Tabla 3 <i>Obtención de requisitos</i> | 59 |
| Tabla 4 <i>Lista de Actores</i> | 60 |
| Tabla 5 <i>Requerimientos Funcionales</i> | 62 |
| Tabla 6 <i>Requerimientos no funcionales</i> | 64 |
| Tabla 7 <i>Descripción de los actores de casos de uso</i> | 65 |
| Tabla 8 <i>Descripción Caso De Uso Ingreso al Sistema</i> | 68 |
| Tabla 9 <i>Caso de Uso – Administración de Usuario</i> | 70 |
| Tabla 10 <i>Caso de uso de administración de roles</i> | 71 |
| Tabla 11 <i>Caso de uso monitoreo de proyectos</i> | 73 |
| Tabla 12 <i>Caso de Uso de Directorio de Empresas</i> | 75 |
| Tabla 13 <i>Caso de uso gestión de recursos</i> | 76 |
| Tabla 14 <i>Caso de uso Avance de actividades</i> | 78 |
| Tabla 15 <i>Cronograma del proyecto</i> | 79 |
| Tabla 16 <i>Tablero de indicadores</i> | 81 |
| Tabla 17 <i>Cuenta</i> | 82 |
| Tabla 18 <i>Caso de uso Gestión POA</i> | 85 |
| Tabla 19 <i>Actividades</i> | 86 |
| Tabla 20 <i>Tabla de rendimientos de módulos</i> | 108 |
| Tabla 21 <i>Prueba de caja negra Inicio de sesión</i> | 108 |
| Tabla 22 <i>Validación de un formulario</i> | 109 |
| Tabla 23 <i>Punto de función: entrada de datos</i> | 115 |
| Tabla 24 <i>Punto de función entradas para el cálculo de funcionalidad</i> | 115 |
| Tabla 25 <i>Ajustes de complejidad del punto función</i> | 116 |
| Tabla 26 <i>Información requerida para el IMS</i> | 121 |
| Tabla 27 <i>Evaluación de preguntas para el cálculo de usabilidad</i> | 124 |
| Tabla 28 <i>Niveles de acceso y seguridad Modulo Administración de Usuarios</i> | 127 |
| Tabla 29 <i>Niveles de acceso y seguridad Modulo de Gestión de Roles</i> | 128 |
| Tabla 30 <i>Niveles de acceso y seguridad Modulo de Gestión de POA</i> | 128 |

| | |
|---|-----|
| Tabla 31 <i>Niveles de acceso y seguridad Módulo Gestión de Recursos</i> | 129 |
| Tabla 32 <i>Niveles de acceso y seguridad Modulo Proyectos Aprobados</i> | 129 |
| Tabla 33 <i>Niveles de acceso y seguridad Módulo de Empresas</i> | 129 |
| Tabla 34 <i>Niveles de acceso y seguridad Módulo de Monitoreo de Proyectos</i> | 130 |
| Tabla 35 <i>Niveles de acceso y seguridad Modulo Cronograma de Actividades</i> | 130 |
| Tabla 36 <i>Niveles de acceso y seguridad Modelo de Actividades</i> | 131 |
| Tabla 37 <i>Coeficientes y exponentes</i> | 132 |
| Tabla 38 <i>Factor LCD/PF de lenguajes de programación</i> | 133 |
| Tabla 39 <i>Costo de elaboración de proyecto</i> | 135 |
| Tabla 40 <i>Costo total del proyecto</i> | 136 |

ÍNDICE DE FIGURAS

| | | |
|------------------|---|----|
| Figura 1 | <i>Organigrama del gobierno autónomo municipal de batallas.....</i> | 3 |
| Figura 2 | <i>Capas de ingeniería de software.....</i> | 30 |
| Figura 3 | <i>Representación gráfica de los casos de uso.....</i> | 39 |
| Figura 4 | <i>Modelo de contenido UWE.....</i> | 40 |
| Figura 5 | <i>Modelo de navegación UWE.....</i> | 41 |
| Figura 6 | <i>Estereotipos de Iconos para el modelo de navegación de UWE.....</i> | 41 |
| Figura 7 | <i>Modelo de estructura de navegación de UWE.....</i> | 42 |
| Figura 8 | <i>Modelo de presentación de UWE.....</i> | 43 |
| Figura 9 | <i>Estereotipos e Iconos para el modelo de presentación UWE.....</i> | 44 |
| Figura 10 | <i>Vista de las características y sub características de la calidad interna y externa según el estándar iso/iec 9126.....</i> | 52 |
| Figura 11 | <i>Calidad en uso como componente del modelo propuesto por el estándar iso/iec 9126.....</i> | 53 |
| Figura 12 | <i>Diagrama de caso de uso general del sistema.....</i> | 67 |
| Figura 13 | <i>Caso de uso ingreso al sistema.....</i> | 68 |
| Figura 14 | <i>Caso de uso administración de usuarios.....</i> | 69 |
| Figura 15 | <i>Caso de uso administración de roles.....</i> | 71 |
| Figura 16 | <i>Caso de uso monitoreo de proyectos.....</i> | 72 |
| Figura 17 | <i>Caso de uso directorio de empresas.....</i> | 74 |
| Figura 18 | <i>Caso de uso gestión de recursos.....</i> | 76 |
| Figura 19 | <i>Caso de uso avance de actividades.....</i> | 77 |
| Figura 20 | <i>Caso de uso cronograma del proyecto.....</i> | 79 |
| Figura 21 | <i>Caso de uso tablero de indicadores.....</i> | 80 |
| Figura 22 | <i>Caso de uso cuenta.....</i> | 81 |
| Figura 23 | <i>Caso de uso Proyectos Aprobados.....</i> | 83 |
| Figura 24 | <i>Caso de uso Gestión de POA.....</i> | 84 |
| Figura 25 | <i>Caso de uso Actividades.....</i> | 86 |
| Figura 26 | <i>Diagrama de clases.....</i> | 88 |
| Figura 27 | <i>Modelo de base de datos.....</i> | 89 |
| Figura 28 | <i>Modelo de navegación – administrador.....</i> | 90 |

| | |
|--|-----|
| Figura 29 <i>Modelo de navegación – supervisor de proyectos</i> | 91 |
| Figura 30 <i>Modelo de navegación – alcalde</i> | 91 |
| Figura 31 <i>Modelo de navegación – secretaria</i> | 92 |
| Figura 32 <i>Modelo de navegación – dirigente</i> | 92 |
| Figura 33 <i>Modelo de presentación: inicio de sesión</i> | 93 |
| Figura 34 <i>Modelo de presentación: página principal</i> | 94 |
| Figura 35 <i>Modelo de presentación: administración de usuarios</i> | 94 |
| Figura 36 <i>Modelo de presentación: administración de roles</i> | 95 |
| Figura 37 <i>Modelo de presentación: administración de recursos</i> | 95 |
| Figura 38 <i>Modelo de presentación: directorio de empresas</i> | 96 |
| Figura 39 <i>Modelo de presentación: monitoreo de proyectos</i> | 96 |
| Figura 40 <i>Modelo de presentación: Cronograma de actividades</i> | 97 |
| Figura 41 <i>Modelo de presentación: Avance de actividades</i> | 97 |
| Figura 42 <i>Modelo de presentación: dashboard</i> | 98 |
| Figura 43 <i>Modelo de Presentación: Gestión de Poa</i> | 98 |
| Figura 44 <i>Modelo de Presentación: Proyectos Aprobados</i> | 99 |
| Figura 45 <i>Interfaz de Inicio de sesión</i> | 100 |
| Figura 46 <i>Pantalla de Inicio</i> | 100 |
| Figura 47 <i>Módulo de Usuarios</i> | 101 |
| Figura 48 <i>Módulo de Roles</i> | 101 |
| Figura 49 <i>Módulo de Recursos</i> | 102 |
| Figura 50 <i>Módulo de Empresas</i> | 102 |
| Figura 51 <i>Módulo de Proyectos</i> | 103 |
| Figura 52 <i>Módulo de Tablero de indicadores</i> | 103 |
| Figura 53 <i>Módulo de Avances</i> | 104 |
| Figura 54 <i>Reportes</i> | 104 |
| Figura 55 <i>Módulo de Cuenta</i> | 105 |
| Figura 56 <i>Módulo de Poa</i> | 105 |
| Figura 57 <i>Módulo de Proyectos Aprobados</i> | 106 |
| Figura 58 <i>Prueba de autenticación – inicio de sesión</i> | 107 |
| Figura 59 <i>Validación de campo en blanco</i> | 109 |

| | |
|---|-----|
| Figura 60 <i>Diagrama de flujo: Registro de usuarios</i> | 110 |
| Figura 61 <i>Grafo de métrica: Registro de usuarios</i> | 111 |

CAPITULO I

MARCO PRELIMINAR

1. MARCO PRELIMINAR

1.1 INTRODUCCIÓN

En la actualidad, podemos ver que los municipios desempeñan un papel fundamental en el proceso de transformación del país, especialmente en la gestión y administración de proyectos. A medida que un municipio crece, las técnicas tradicionales utilizadas en las operaciones administrativas se vuelven cada vez más inadecuadas. Por lo tanto, es crucial adaptarse a los cambios tecnológicos en el manejo de la información para mejorar la toma de decisiones. Para lograrlo, es necesario utilizar las tecnologías de la información y las comunicaciones, las cuales están teniendo un gran impacto en la estructura y el funcionamiento de las instituciones públicas y privadas.

Las tecnologías de información y las comunicaciones permiten la administración efectiva de información en pequeñas organizaciones, lo que genera una competencia adicional. Además, brindan la oportunidad de entregar productos y servicios a través de nuevos medios corporativos, lo que aumenta la calidad y la rentabilidad. Estas nuevas tecnologías están afectando casi todos los aspectos de la vida económica, especialmente la organización y dirección de las instituciones.

En los últimos años, se ha desarrollado el concepto de Gobierno Electrónico, el cual busca modernizar la gestión pública mediante mejores prácticas, mayor control y transparencia. El Gobierno Autónomo Municipal de Batallas es un ejemplo de esto, ya que busca proporcionar a los ciudadanos acceso flexible y oportuno a toda la información relacionada con los proyectos sin necesidad de acudir a las dependencias del gobierno.

El objetivo de este proyecto es implementar un Sistema de Información que permita la interacción entre el Gobierno Autónomo Municipal de Batallas y los comuneros, facilitando el registro de proyectos, el seguimiento del tiempo de ejecución y el cumplimiento de los cronogramas de acuerdo con la planificación.

1.2 ANTECEDENTES

1.2.1 Antecedentes Institucionales

El municipio de Batallas ciudad comercial perteneciente a la tercera sección de la provincia los andes del departamento de La paz, según el INE (2019) la población realizada el año 2012 es de 17,426 habitantes, 9 cantones y cuenta con 43 comunidades los cuales son dependientes de la Municipio de Batallas.

1.2.1.1. Visión

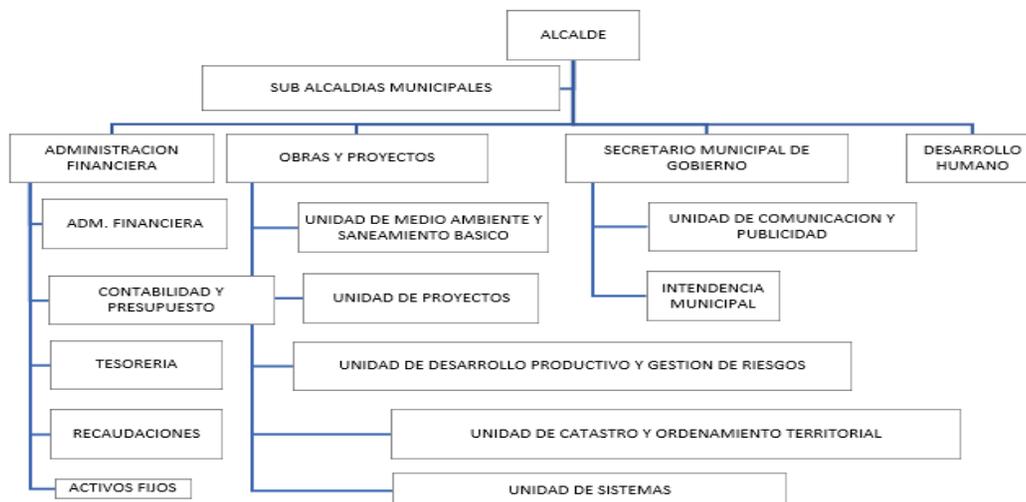
El Gobierno Autónomo Municipal de Batallas, es de ofrecer apoyo económico y de desarrollo humano convirtiéndose así en un eje articulador del desarrollo regional.

1.2.1.2. Misión

Contribuir a la satisfacción de las necesidades colectivas y garantizar la integración y participación de los ciudadanos, en la planificación y el desarrollo.

Figura 1

Organigrama del gobierno autónomo municipal de batallas



Nota. Tomado del *Manual de Organización y funciones del gobierno autónomo municipal de batallas*, GAMB, 2015.

1.2.2 Antecedentes afines al proyecto de grado

1.2.2.1. Antecedentes Internacionales

- El trabajo realizado por (Uchuquicaña 2017), “Implementación de un sistema de información orientado al control y seguimiento de productos” la siguiente tesis de grado fue desarrollado en la empresa publigarment S.A.C. en Lima - Perú, como su principal objetivo tiene el de implementar un sistema informático web que permita integrar ventas, producción y logística. Para el desarrollo del sistema se utilizó la metodología RUP.
- Se cita el trabajo realizado por (Mendoza 2017), “Implementación de sistemas web para la gestión y control de los procesos de la unidad de titulación”, Universidad salesiana, sede Guayaquil, como objetivo principal es implementar un sistema que le permita gestionar y administrar el proceso de titulación. La metodología que uso para el desarrollo del sistema web es la METRICA V3.
- El presente trabajo realizado por (García 2018), “Análisis, diseño e implementación de un aplicativo web para el seguimiento y control de pedidos”, tesis de grado realizado en la empresa Lima tours S.A.C. como principal objetivo tiene Analizar, diseñar e implementar un aplicativo web. Para esta investigación se utilizó la metodología RUP.

1.2.2.2. Antecedentes Nacionales

- El trabajo realizado por (CUSSI 2016), “SISTEMA WEB DE SEGUIMIENTO Y CONTROL DE LA INFORMACIÓN DE CLIENTES”, proyecto de grado realizado en la empresa LINXSS.R.L. como objetivo principal tiene el de brindar información oportuna y actualizada a gerencia general, área contable y vendedores. Para la realización del sistema se utilizó la metodología ágil KANBAN.

1.2.2.2. Antecedentes Locales

- Se cita el trabajo realizado por (Choque 2020), “Sistema de información para el seguimiento y control de ejecución de proyectos municipales”, proyecto desarrollado en el municipio de Catacora, el principal objetivo facilite toda la información que se genere a cerca del registro de proyectos, tiempo de ejecución de un proyecto, cumplimiento de cronogramas de acuerdo a la planificación de las poas. Para este proyecto utilizaron la metodología SCRUM.
- Se referencia el trabajo realizado por (Rivero 2017), “Sistema web de control y seguimiento de obras municipales”, en Sub alcaldía de Ovejuyo D-I municipio de Palca, el objetivo del proyecto es para dar solución a los problemas suscitados por los vecinos en las gestiones pasadas con los incumplimientos de la ejecución de proyectos, ocasionados por la gran cantidad de proyectos que existen en la Sub - alcaldía. Para el desarrollo de este proyecto dieron uso de la metodología UWE.
- (Sangalli 2020), “Sistema de control y seguimiento de proyectos educativos”, el proyecto fue realizado en el municipio de Puerto Pérez, como finalidad tiene tener información actualizada y en el momento oportuno ya que es una necesidad primordial en cualquier institución municipal y que esta sea explorada desde que cualquier dispositivo que se conecte a internet. Para el proyecto desarrollado se utilizó la metodología UWE.

1.3 PLANTEAMIENTO DEL PROBLEMA

En el ámbito de la gestión de proyectos, es imprescindible contar con herramientas eficientes y efectivas que permitan monitorear el avance de proyectos de cualquier índole. En la actualidad, con la creciente digitalización y la necesidad de trabajar de forma remota, se requiere disponer de herramientas accesibles a través de la web para garantizar una gestión efectiva.

El control de proyectos implica el seguimiento de diferentes actividades, recursos, plazos y costos, así como la identificación y resolución de posibles desviaciones. Sin embargo, muchos sistemas tradicionales de gestión de proyectos carecen de la capacidad de ser accesibles desde cualquier lugar con conexión a internet, lo que dificulta la colaboración y la toma de decisiones en tiempo real.

Por lo tanto, el problema radica en la necesidad de desarrollar un sistema para el control de proyectos que sea accesible a través de la web y que pueda permitir el seguimiento en tiempo real del avance de las tareas, el registro de las actividades realizadas, la asignación de recursos y la generación de informes de estado. Además, debe facilitar la comunicación y colaboración entre los miembros del equipo, independientemente de su ubicación geográfica.

1.3.1 Problema Principal

La realidad que atraviesa el Gobierno Autónomo Municipal de Batallas en el manejo de información de proyectos es caótica, ya que no cuentan con un sistema que les brinde información en tiempo real, para brindar una respuesta inmediata a las autoridades de las comunidades que concurren con mucha frecuencia a exigir el avance de sus proyectos y también solicitan proyectos para sus comunidades.

El manejo de información para el monitoreo de proyectos en el Gobierno Autónomo Municipal de Batallas es de manera tradicional, ya que trabajan de manera individual en Herramientas de oficina digital que vendrían a ser los procesadores de texto, hojas de cálculo, archivos físicos, de la misma manera no cuentan con una herramienta que visualice toda esta información, de manera que para obtener la información requerida tienen que ir de unidad en unidad para solicitar la información solicitada. Esta información, es posteriormente centralizada por un operador para la emisión de informes, reportes estadísticos y consiguientemente la toma de decisiones

lo cual es un proceso bastante demoroso que genera disconformidad en los dirigentes ya que tienen que volver a visitar las instalaciones de la alcaldía una y otra vez.

En resumen, el Gobierno Autónomo Municipal de Batallas presenta carencias en el registro de los proyectos que gestiona y en el seguimiento de su avance. Además, no dispone de información actualizada sobre las empresas que llevan a cabo los proyectos, ni sobre las comunidades que se benefician de ellos. Esto genera desorganización y falta de disponibilidad de información, lo cual se agrava por la ausencia de un sistema informático que gestione todos estos datos.

1.3.2 Problemas secundarios

- La descentralización de la información concerniente a los proyectos en ejecución es caótica, ocasionando la demora de la información requerida.
- El monitoreo de proyectos de manera tradicional, genera pérdida de datos e informes equívocos.
- El no tener antecedentes e información de las empresas que realizan los proyectos, generan el retraso de los proyectos.
- No tener la información centralizada de las comunidades pertenecientes al Gobierno Autónomo Municipal de Batallas, no permite que exista un reporte confiable.
- La inexistencia de un registro digitalizado de los recursos de cada proyecto en ejecución de cada gestión, ocasiona la desinformación para las autoridades entrantes.
- No existe un mecanismo que muestre las actividades pendientes, en desarrollo y concluidos de cada proyecto, generando el retraso de los proyectos.
- La generación de informes imprecisos e inexactos plantea problemas significativos en la toma de decisiones informadas, asignación de recursos, una planificación deficiente y una falta de comprensión real del estado de los proyectos o las operaciones.

1.3.3 Formulación del problema

Teniendo en cuenta los argumentos mencionados en el planteamiento general, se ha originado una pregunta que se relaciona directamente con el problema principal en cuestión.

¿Cómo se podría optimizar el proceso tradicional del manejo de información y del monitoreo de avance de los proyectos vía web en el Gobierno Autónomo Municipal de Batallas y de esa manera no generar descontentos en la comunidad perteneciente al municipio?

1.4 OBJETIVOS

1.4.1 Objetivo General

Desarrollar un sistema de información para monitorear el proceso de control de proyectos, vía web en el Gobierno Autónomo Municipal de Batallas, que permita realizar un seguimiento a la gestión de proyectos, y proporcione información precisa, actualizada y confiable que sea útil para la toma de decisiones inmediatas.

1.4.2 Objetivos Específicos

- Centralizar la información pertinente a los proyectos que ya se encuentran en ejecución, que evite el desorden caótico de la información requerida.
- Desarrollar un módulo que permita gestionar de manera efectiva y centralizada todas las empresas que estén vinculadas a los proyectos en ejecución, para tener un antecedente de las mismas.
- Recopilar información completa y actualizada de todas las comunidades que conforman el municipio, para registrar en la base de datos, de manera que los datos obtenidos sean precisos acerca de las comunidades beneficiadas con los proyectos en ejecución. Logrando tener un registro completo de las

comunidades involucradas en los proyectos y una mejor comprensión de las necesidades específicas de cada una de ellas, lo que permitiría una gestión más efectiva y una toma de decisiones más precisa en relación a los proyectos.

- Desarrollar un módulo de recursos relacionados con los proyectos en ejecución, para una buena información de la gestión del POA.
- Desarrollar un módulo de gestión de cronograma de actividades, mejorando la visualización de los proyectos pendientes, en desarrollo y los concluidos logrando una buena ejecución y conclusión de proyectos.
- Implementar módulos de reportes a cada uno de los módulos, para tener la facilidad de ubicar los reportes, evitando la confusión de descargar el reporte no requerido.

1.5 JUSTIFICACIÓN

1.5.1 Justificación Técnica

El siguiente proyecto se justifica técnicamente por el uso de tecnología de comunicación y de medios para efectuar el análisis, diseño, implementación y mantenimiento haciendo uso de las TIC y de esta manera implementar tecnología al Gobierno Autónomo Municipal de Batallas. La implementación de esta tecnología será de mucha utilidad, ya que agilizará los procesos demorosos tradicionales que se utilizan en esta institución.

El Gobierno Autónomo Municipal de Batallas cuenta con equipos de computación e internet para el uso del sistema web.

1.5.2 Justificación Económica

El sistema a implementarse en el Gobierno Autónomo Municipal de Batallas económicamente beneficiará tanto a la alcaldía y tanto como a los dirigentes de las

comunidades beneficiadas, menos gasto de compra de papeles, reducción de gasto de pasajes y el software desarrollado será gratuito para el control y seguimiento de proyectos. De la misma manera el software a implementarse esta desarrollado con herramientas gratuitas.

1.5.3 Justificación Social

El desarrollo del software de control y seguimiento de proyectos será de muy gran utilidad para la sociedad perteneciente al Gobierno Autónomo Municipal de Batallas para que los encargados de gestión de proyectos, lo realizaran de una manera más ordenada y responsable del seguimiento de proyectos.

1.6 METODOLOGÍA

1.6.1 Metodología de Desarrollo

“UWE es un enfoque de ingeniería de software para el dominio web que tiene como objetivo cubrir todo el ciclo de vida del desarrollo de aplicaciones web. El aspecto clave que distingue a UWE es la confianza en los estándares.” (LMU - Universidad Ludwig Maximilian de Múnich, 2016).

Según LMU - Universidad Ludwig Maximilian de Múnich (2016) la metodología UWE está dividido en 5 modelos que son los siguientes:

- Modelo de requisitos
- Modelo de contenido
- Modelo de navegación
- Modelo de presentación
- Modelo de proceso

1.6.2 Métricas de Calidad 9126

Según Pressman (2010) indica:

Incluso los desarrolladores de software más experimentados estarán de acuerdo en que obtener software de alta calidad es una meta importante. Pero, ¿cómo se define la calidad del software? En el sentido más general se define¹ como: Proceso eficaz de software que se aplica de manera que crea un producto útil que proporciona valor medible a quienes lo producen y a quienes lo utilizan. (p. 340).

1.6.3 Costos

Según Sharma (2011) define:

COCOMO II es un modelo de coste objetivo para la planificación y ejecución de proyectos de software. Es un ingrediente importante para el manejo proyectos de software o líneas de negocio de software. Un modelo de costos proporciona un marco para comunicar las decisiones comerciales entre las partes interesadas de un esfuerzo de software. (p. 1).

Según Calero (2010) nos dice que se divide en tres partes que lo mencionaremos a continuación:

- Básico
- Intermedio
- Detallado.

1.6.4 Seguridad

ISO 27001 es una norma internacional que permite el aseguramiento, la confidencialidad e integridad de los datos y de la información, así como de los sistemas que la procesan.

1.7 METODOS DE RECOLECCIÓN DE DATOS

1.7.1 Técnicas de Investigación

1.7.1.1. La entrevista

“Las entrevistas son la técnica de recolección más utilizada, y de hecho son prácticamente inevitables en cualquier desarrollo. En las entrevistas se pueden identificar claramente tres fases [Piattini]: preparación, realización y análisis.” (Grau, 2016, pág. 10).

1.7.1.2. Encuesta y cuestionarios

Una encuesta es un procedimiento dentro de los diseños de una investigación descriptiva en el que el investigador recopila datos mediante el cuestionario previamente diseñado, sin modificar el entorno ni el fenómeno donde se recoge la información ya sea para entregarlo en forma de tríptico, gráfica o tabla. (Grau, 2016, pág. 10).

1.8 HERRAMIENTAS DE DESARROLLO

Para el desarrollo del presente proyecto aremos uso de las herramientas de desarrollo tanto para el lado del cliente y como para el lado del servidor, a continuación, se menciona y define las herramientas a usarse.

1.8.1 herramientas de desarrollo Back End

Son herramientas para el desarrollo por el lado del servidor y los mencionaremos a continuación con las que nosotros trabajaremos para el desarrollo del presente sistema.

1.8.1.1. Php

“PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML” (Php, 2022).

1.8.1.2. Laravel 10

Laravel es un marco de aplicación web con una sintaxis expresiva y elegante. Un marco web proporciona una estructura y un punto de partida para crear su aplicación, lo que le permite concentrarse en crear algo increíble mientras nos preocupamos por los detalles. (Laravel-LLC, 2022, párr. 1).

1.8.1.3. MariaBD

“MySQL es un sistema de gestión de base de datos relacional (RDBMS) de código abierto, basado en lenguaje de consulta estructurado (SQL). MySQL se ejecuta en prácticamente todas las plataformas, incluyendo Linux, Mac y Windows” (Oracle-Corporation, 2022).

1.8.2 herramientas de desarrollo Front End

1.8.2.1. Bootstrap

“Bootstrap es un framework CSS de código abierto que favorece el desarrollo web de un modo más sencillo y rápido. Incluye plantillas de diseño basadas en

HTML y CSS con la que es posible modificar tipografías, formularios, botones, tablas, navegaciones, menús desplegables y otros” (Arimerics, 2022).

1.8.2.2. HTML5

“HTML5 es un lenguaje de marcado que se utiliza para el desarrollo de páginas de Internet. Se trata de las siglas que corresponden a HyperText Markup Language, es decir, Lenguaje de Marcas de Hipertexto”. (Flores, 2015).

1.8.2.3. CSS3

CSS3 son las siglas en inglés para «hojas de estilo en cascada» (Cascading Style Sheets). Básicamente, es un lenguaje que maneja el diseño y presentación de las páginas web, es decir, cómo lucen cuando un usuario las visita. Funciona junto con el lenguaje HTML que se encarga del contenido básico de las páginas. (HubSpot, 2022).

1.8.2.3. JavaScript

Javascript es un lenguaje de programación utilizado para crear pequeños programitas encargados de realizar acciones dentro del ámbito de una página web. Con Javascript podemos crear efectos especiales en las páginas y definir interactividades con el usuario. El navegador del cliente es el encargado de interpretar las instrucciones Javascript y ejecutarlas para realizar estos efectos e interactividades, de modo que el mayor recurso, y tal vez el único, con que cuenta este lenguaje es el propio navegador. (Alvarez & Gutierrez, 2007).

1.9 LÍMITES Y ALCANCES

A continuación, se hace mención sobre los límites y alcances del sistema propuesto.

1.9.1 Alcances

El siguiente sistema contendrá los siguientes módulos:

- Contará con un módulo de gestión de proyectos: En donde se podrá realizar reportes de todos los proyectos y de cada proyecto, actualizar estado, crear, editar, eliminar, buscar datos según nombre de comunidad.
- Contará con un módulo de gestión de recursos: Las funciones que tendrá, crear, editar, actualizar estado, eliminar, filtrar por comunidades y reportes.
- Contará con un módulo de administración de usuarios: registrar nuevo usuario, editar, eliminar, asignar rol, estado, reporte.
- Contará con un módulo de gestión de empresas: se tendrá la lista de todas las empresas activas e inactivas en donde se puede dar de alta nueva empresa, editar, actualizar estado, eliminar, reporte.
- Contará con un módulo de gestión de cronograma de actividades: se podrá crear, editar, eliminar, actualizar estado, actualizar imagen, visualizar imagen, reporte.
- Contará con un módulo de gestión de cuentas: El usuario podrá Actualizar su cuenta:
- Contará con un módulo de información estadístico de los proyectos realizados de cada comunidad.
- Contará con un módulo de cerrar sesión: el usuario podrá cerrar sesión y de esta manera salir del sistema.

1.9.2 Limites

El presente sistema no podrá realizar:

- El sistema desarrollado se limita a que será solo para El GAMB.

- Se limita solo al control y Monitoreo de proyectos aprobados por la sesión del Honorable Concejo del GAMB.
- Gestión de recursos humanos.
- Gestión de administración contable.
- No contemplará a la gestión de administración de inventarios y de la misma manera se limita a ser un software libre.

1.10 APORTES

El aporte principal es la implementación del sistema de monitoreo de proyectos que será de gran aporte en el Gobierno Autónomo Municipal de Batallas Automatizando y centralizando los procesos manuales y de ofimática en el proceso de seguimiento de los proyectos aprobados por la sesión de concejo, brindando una mejor atención y servicio a la población del municipio de Batallas.

CAPITULO II

MARCO TEÓRICO

2. MARCO TEÓRICO

En el presente capítulo se describirá los principios y conceptos básicos para el desarrollo del presente proyecto, sobre las herramientas a utilizar dentro del desarrollo del proyecto teóricamente para poder conocer y aprender sobre las funcionalidades de cada una de ellas, sobre la calidad y seguridad para el software y la metodología a implementarse.

2.1 SISTEMA

Un sistema es un conjunto de elementos interconectados que trabajan juntos para lograr un objetivo común. En informática, un sistema se refiere a un conjunto de componentes de hardware y software que trabajan juntos para proporcionar una funcionalidad específica.

En informática, también se habla de sistemas distribuidos, que son sistemas que se ejecutan en múltiples computadoras interconectadas a través de una red. Los sistemas distribuidos se utilizan para proporcionar servicios a gran escala, como el procesamiento de transacciones financieras, el almacenamiento y la recuperación de datos y el procesamiento de grandes volúmenes de datos.

Los sistemas también pueden funcionar como un manual de trabajo más práctico y simplificado. Dependerá de la forma en que se necesite utilizar un sistema y sus distintos niveles para la elaboración de un trabajo que conlleva distintos tipos de logros, pero con la misma Causa. La causa puede ser el apoyo a una función empresarial o la ampliación de un producto que puede ser vendido para generar ingresos. Producto que puede comprarse para generar ingresos.

Un sistema es un conjunto ordenado de factores que pueden estar interrelacionados y tener interacción entre sí para alcanzar un objetivo. Para examinar un sistema, es muy necesario reconocer sus componentes y las relaciones entre esos

componentes. En concreto, un sistema es un conjunto de elementos que, de forma ordenada, se relacionan entre sí que contribuyen a un objetivo determinado.

2.2 MONITOREO

Monitorear, se refiere al proceso de supervisar continuamente el estado o progreso de algo.

En informática, el monitoreo se utiliza para supervisar y controlar el rendimiento, la disponibilidad y la seguridad de los sistemas informáticos.

El monitoreo es una herramienta de gestión para controlar el avance de los proyectos en ejecución, según la idea básica de comparar su desempeño efectivo con los planes, y medir los resultados reales en función de los previstos.

El monitoreo es parte integrante de la ejecución de proyectos, no siendo más que una forma para controlar el avance de los proyectos en forma eficaz y proporcionar información sistemática, uniforme y fiable sobre el avance de los mismos.

Una vez establecido, el monitoreo le ahorra tiempo y trabajo al gerente del proyecto, y facilita el control y preparación de los reportes correspondientes. No aumenta la complejidad del proyecto, sino que permite hacerlos objeto de una gestión más sistemática y manejable. (OPS, 1999, p. 1).

Monitoreo, es un proceso permanente el cual consiste en hacer revisiones de un proyecto, para de esta manera hacer cumplir los cronogramas programados.

En resumen, el monitoreo es el proceso de supervisar continuamente el estado o progreso de algo. El monitoreo se utiliza para detectar y solucionar problemas que se presentan en el contexto, así como para asegurarse de que los todo esté disponible y funcionando correctamente.

2.3 PROCESO

Un proceso es un seguimiento de pasos a seguir hasta conseguir lo deseado en cualquier ámbito.

En un sistema de información, un proceso es un programa en ejecución que se encuentra en una instancia particular de su ejecución. Es decir, un proceso es un conjunto de tareas que se realizan dentro de un programa en un momento determinado. Los procesos son la base del funcionamiento de los sistemas operativos modernos y se utilizan para administrar la ejecución de los programas y recursos del sistema.

Un proceso es un conjunto de actividades, acciones y tareas que se ejecutan cuando va a crearse algún producto del trabajo. Una actividad busca lograr un objetivo amplio (por ejemplo, comunicación con los participantes) y se desarrolla sin importar el dominio de la aplicación, tamaño del proyecto, complejidad del esfuerzo o grado de rigor con el que se usará la ingeniería de software. Una acción (diseño de la arquitectura) es un conjunto de tareas que producen un producto importante del trabajo (por ejemplo, un modelo del diseño de la arquitectura). Una tarea se centra en un objetivo pequeño, pero bien definido (por ejemplo, realizar una prueba unitaria) que produce un resultado tangible. (Pressman, 2010, p.12).

Proceso, es una secuencia de pasos dispuestos con algún tipo de sentido común que se especializa en alcanzar algún resultado particular. Expone mecanismos de comportamiento que los individuos diseñan para mejorar la productividad de alguna cosa, para establecer un orden o para deshacerse de algún tipo de problema.

En resumen, el proceso es una secuencia esencial en la gestión de proyectos, ya que establecen una secuencia lógica de acciones para lograr los objetivos deseados. Al seguir procesos bien definidos, se promueve la eficiencia, la consistencia

y la calidad en la ejecución de actividades, lo que contribuye al éxito global del proyecto.

2.4 SISTEMA WEB

Un sistema web es un tipo de sistema de software que se ejecuta en un servidor y se accede a través de un navegador web. Un sistema web está diseñado para proporcionar servicios o funcionalidades a los usuarios a través de internet o una intranet.

La web ha venido avanzando con pasos agigantados estos últimos años, comenzando con una Web de Hipertexto que hizo posible enlazar y conectar los documentos compuestos por texto, imágenes, videos y otros.

En la actualidad todos estamos familiarizados con el sistema web, el sistema está basado en una arquitectura cliente-servidor. El cliente es el encargado de iniciar la comunicación por medio de un navegador o explorador (Google, Firefox, y otros) que interpreta y visualiza la información suministrada por el servidor. El servidor está a la escucha de nuevos clientes que realicen peticiones para suministrar las páginas solicitadas o procesar la información recibida por el cliente.

Según Berzosa (2022) explica que, un sistema de navegacion web es la interfaz, en otras palabras tambien infiere que es la informacion que presenta al usuario, presentado mediante diseños grafico o textuales.

Los sistemas web pueden ser muy variados en cuanto a su propósito y complejidad. Por ejemplo, un sistema web puede ser una simple página web con información estática, como una página de inicio de una empresa. También puede ser una aplicación web que permite a los usuarios realizar tareas específicas, como la gestión de proyectos o la gestión de inventarios.

2.5 INTERNET

Internet es una red global de ordenadores interconectados que se comunican entre sí utilizando un conjunto de protocolos estándar de comunicación de datos. Es una red de redes que permite a los usuarios conectarse y compartir información, recursos y servicios en todo el mundo.

Internet se compone de millones de servidores, routers y otros dispositivos de red que están interconectados alrededor del mundo. Los usuarios pueden acceder a internet a través de dispositivos como ordenadores, smartphones, tabletas, consolas de juegos y otros dispositivos conectados a la red.

Según, Milenium (2022) define:

El Internet, algunas veces llamado simplemente "La Red", es un sistema mundial de redes de computadoras, un conjunto integrado por las diferentes redes de cada país del mundo, por medio del cual un usuario en cualquier computadora puede, en caso de contar con los permisos apropiados, acceder información de otra computadora y poder tener inclusive comunicación directa con otros usuarios en otras computadoras. (párr. 4).

En resumen, internet es una red global de ordenadores interconectados que permite a los usuarios conectarse y compartir información, recursos y servicios en todo el mundo. Ofrece una amplia variedad de servicios y recursos, y se basa en una serie de protocolos de comunicación de datos que permiten a los dispositivos de red comunicarse entre sí.

2.6 INGENIERÍA DE REQUERIMIENTOS

La ingeniería de requerimientos, es como una lista de requisitos indispensables a cumplirse, para que el sistema a implementarse tenga una buena calidad. Ya que gracias a la ingeniería de requerimientos podemos saber lo que en realidad requiere el usuario.

El diseño y construcción de software de computadora es difícil, creativo y sencillamente divertido. En realidad, elaborar software es tan atractivo que muchos desarrolladores de software quieren ir directo a él antes de haber tenido el entendimiento claro de lo que se necesita. Argumentan que las cosas se aclararán a medida que lo elaboren, que los participantes en el proyecto podrán comprender sus necesidades sólo después de estudiar las primeras iteraciones del software, que las cosas cambian tan rápido que cualquier intento de entender los requerimientos en detalle es una pérdida de tiempo, que las utilidades salen de la producción de un programa que funcione y que todo lo demás es secundario. Lo que hace que estos argumentos sean tan seductores es que tienen algunos elementos de verdad. Pero todos son erróneos y pueden llevar un proyecto de software al fracaso. (Pressman, 2010, p. 102).

Pressman (2010) nos explica sobre la importancia de la ingeniería de requerimientos:

La ingeniería de requerimientos proporciona el mecanismo apropiado para entender lo que desea el cliente, analizar las necesidades, evaluar la factibilidad, negociar una solución razonable, especificar la solución sin ambigüedades, validar la especificación y administrar los requerimientos a medida de que se transforman en un sistema funcional. Incluye siete tareas diferentes: concepción, indagación, elaboración, negociación, especificación, validación y administración. Es importante notar que algunas de estas tareas ocurren en paralelo y que todas se adaptan a las necesidades del proyecto. (p. 102).

Podemos inferir a la ingeniería de requerimientos como el proceso de recopilar, analizar y verificar las necesidades que requiere el usuario para un sistema a desarrollar.

2.6.1 Concepción

La concepción en informática se refiere al proceso de definición y planificación de un proyecto de software. Es el primer paso en el ciclo de vida del software y es fundamental para el éxito del proyecto. La concepción implica la identificación de los objetivos del proyecto, la definición de los requisitos y la planificación del proceso de desarrollo del software.

Pressman (2010) nos dice lo siguiente:

¿Cómo inicia un proyecto de software? ¿Existe un solo evento que se convierte en el catalizador de un nuevo sistema o producto basado en computadora o la necesidad evoluciona en el tiempo? No hay respuestas definitivas a estas preguntas. En ciertos casos, una conversación casual es todo lo que se necesita para desencadenar un trabajo grande de ingeniería de software. Pero en general, la mayor parte de proyectos comienzan cuando se identifica una necesidad del negocio o se descubre un nuevo mercado o servicio potencial. Los participantes de la comunidad del negocio (por ejemplo, los directivos, personal de mercadotecnia, gerentes de producto, etc.) definen un caso de negocios para la idea, tratan de identificar el ritmo y profundidad del mercado, hacen un análisis de gran visión de la factibilidad e identifican una descripción funcional del alcance del proyecto. Toda esta información está sujeta a cambio, pero es suficiente para desencadenar análisis con la organización de ingeniería de software. (p. 103).

En el momento de pensar en el proyecto, se establece la comprensión básica del problema, los individuos que quieren una solución, el carácter de la solución deseada, además de la eficacia de la comunicación y la colaboración preliminares entre las otras partes interesadas y el equipo del proyecto. De la conversación y la colaboración iniciales entre las demás partes interesadas y el equipo de software.

En resumen, la concepción en informática es el proceso de definición y planificación de un proyecto de software. Es un paso fundamental en el ciclo de vida del software y tiene un impacto significativo en el éxito del proyecto.

2.6.2 Indagación

La indagación, se refiere al proceso de investigación y exploración de una o varios datos para poder obtener información mas relevante y útil en un contexto específico. Se trata de un proceso que involucra la investigación, el análisis y la evaluación de diferentes soluciones, para resolver un problema o necesidad del usuario.

Pressman (2010) aclara que la indagación no están sencillas como parece:

En verdad que parece muy simple: preguntar al cliente, a los usuarios y a otras personas cuáles son los objetivos para el sistema o producto, qué es lo que va a lograrse, cómo se ajusta el sistema o producto a las necesidades del negocio y, finalmente, cómo va a usarse el sistema o producto en las operaciones cotidianas. Pero no es simple: es muy difícil. (p. 103).

Pressman. (2010), También explica las complejidades que existen:

- **Problemas de alcance:** Sistemas más definidos o usuarios finales indecisos.
- **Problemas de entendimiento:** Usuario indeciso genera una mala comunicación y un entendimiento malo.
- **Problemas de volatilidad:** Según la tecnología avanza los requerimientos cambian. (p. 103).

2.6.3 Elaboración

La elaboración simplemente es la creación de un producto en donde realiza los escenarios que interactuará con el usuario.

La elaboración está motivada por la creación y mejora de escenarios de usuario que describan cómo interactuará el usuario final (y otros actores) con el sistema. Cada escenario de usuario se enuncia con sintaxis apropiada para extraer clases de análisis, que son entidades del dominio del negocio visibles para el usuario final. Se definen los atributos de cada clase de análisis y se identifican los servicios que requiere cada una de ellas. Se identifican las relaciones y colaboración entre clases, y se producen varios diagramas adicionales. (Pressman, 2010, p. 103).

En resumen, la elaboración en informática es el proceso de construcción y desarrollo de un software o sistema informático, que sigue a la etapa de concepción y precede a la implementación y mantenimiento. Es una etapa fundamental en el ciclo de vida del software y requiere la programación, integración y pruebas del software para asegurar que funcione de acuerdo a los requisitos y especificaciones definidos.

2.6.4 Negociación

La negociación, se refiere al proceso de llegar a un acuerdo entre dos o más partes en relación a un proyecto, contrato o servicio informático. Este proceso puede involucrar la discusión y negociación de diversos términos y condiciones, incluyendo el precio, el alcance del proyecto, los plazos y la calidad del servicio.

Según Pressman. (2010) nos explica:

Estos conflictos deben reconciliarse por medio de un proceso de negociación. Se pide a clientes, usuarios y otros participantes que ordenen sus requerimientos según su prioridad y que después analicen los conflictos. Con el empleo de un enfoque iterativo que da prioridad a los requerimientos, se evalúa su costo y riesgo, y se enfrentan los conflictos internos; algunos requerimientos se eliminan, se combinan o se modifican de modo que cada parte logre cierto grado de satisfacción. (p. 104).

La negociación, es un proceso importante para llegar a un acuerdo entre dos o más partes en relación a un proyecto. Este proceso involucra la discusión y negociación de diversos términos y condiciones, y puede ocurrir en diferentes contextos y situaciones.

2.6.4 Especificación

La especificación, se refiere al proceso de definir y documentar los requisitos y funcionalidades de un software o sistema informático. Es una etapa fundamental en el ciclo de vida del software, que sigue a la concepción y precede a la elaboración, implementación y mantenimiento.

Pressman (2010), nos explica:

En el contexto de los sistemas basados en computadora (y software), el término especificación tiene diferentes significados para distintas personas. Una especificación puede ser un documento escrito, un conjunto de modelos gráficos, un modelo matemático formal, un conjunto de escenarios de uso, un prototipo o cualquier combinación de éstos. (p. 104).

La especificación es el proceso de definir y documentar los requisitos y funcionalidades de un software o sistema informático para que en futuras tenga un un buen mantenimiento de las funcionalidades del software.

2.6.5 Validación

La validación es la verificación y comprobación de un sistema para ver si cumple con los requisitos especificados y ver sus funcionalidades para el que esta diseñado, de acuerdo a las normativas implementadas para su desarrollo.

Pressman (2010), nos indica:

La calidad de los productos del trabajo que se generan como consecuencia de la ingeniería de los requerimientos se evalúa durante el paso de validación. La validación de los requerimientos analiza la especificación a fin de garantizar que todos ellos han sido enunciados sin ambigüedades; que se detectaron y corrigieron las inconsistencias, las omisiones y los errores, y que los productos del trabajo se presentan conforme a los estándares establecidos para el proceso, el proyecto y el producto. (p. 105).

Podemos decir que, la verificación es para garantizar el correcto funcionamiento del producto desarrollado.

2.6.6 Administracion de los requerimientos

La administración de los requerimientos, es para dar un seguimiento y identificar, a los posibles cambios que se pueden realizar en el proyecto en cualquier momento.

Según Pressman (2010) señala que, “la administración de los requerimientos es el conjunto de actividades que ayudan al equipo del proyecto a identificar, controlar y dar seguimiento a los requerimientos y a sus cambios en cualquier momento del desarrollo del proyecto” (p. 105).

Podemos decir que, gracias a la administracion de los requerimientos podemos estar al tanto, con los cambios que se puedan realizar en el proyecto en desarrollo.

2.7 INGENIERIA DE SOFTWARE

La ingeniería de software, es el encargado de la producción de software desde el inicio del desarrollo del software hasta el mantenimiento del producto desarrollado.

Según Sommerville (2005), nos conceptualiza: “La ingeniería del software es una disciplina de la ingeniería que comprende todos los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de este después de que utiliza” (p. 6).

Según Sommerville (2005) existen dos frases clave:

- Disciplina de la ingeniería: los ingenieros hacen que las cosas funcionen. Aplican teorías, métodos y herramientas donde sean convenientes, pero las utilizan de forma selectiva y siempre tratando de descubrir soluciones a los problemas, aun cuando no existan teorías y métodos aplicables para resolverlos.
- Todos los aspectos de producción de software: la ingeniería del software no solo comprende los procesos técnicos del desarrollo de software, sino también con actividades tales como la gestión de software de alta calidad. (p. 6).

Según Pressman (2010), explica cómo se estructura:

La ingeniería de software es una tecnología con varias capas. Como se aprecia en la figura 2, cualquier enfoque de ingeniería (incluso la de software) debe basarse en un compromiso organizacional con la calidad. La administración total de la calidad, Six Sigma y otras filosofías similares¹⁰ alimentan la cultura de mejora continua, y es esta cultura la que lleva en última instancia al desarrollo de enfoques cada vez más eficaces de la ingeniería de software.

El fundamento en el que se apoya la ingeniería de software es el compromiso con la calidad. (p. 12).

Figura 2

Capas de ingeniería de software



Nota: Tomado del libro de *Ingeniería de Software. Un enfoque práctico* (p. 12), por Pressman, 2020, McGraw-Hill Editorial.

Según Pressman (2010) fundamenta sobre las capas de de proseso de la ingenieria de software:

El fundamento para la ingeniería de software es la capa proceso. El proceso de ingeniería de software es el aglutinante que une las capas de la tecnología y permite el desarrollo racional y oportuno del software de cómputo. El proceso define una estructura que debe establecerse para la obtención eficaz de tecnología de ingeniería de software. El proceso de software forma la base para el control de la administración de proyectos de software, y establece el contexto en el que se aplican métodos técnicos, se generan productos del trabajo (modelos, documentos, datos, reportes, formatos, etc.), se establecen puntos de referencia, se asegura la calidad y se administra el cambio de manera apropiada.

Los métodos de la ingeniería de software proporcionan la experiencia técnica para elaborar software. Incluyen un conjunto amplio de tareas, como comunicación, análisis de los requerimientos, modelación del diseño, construcción del programa, pruebas y apoyo. Los métodos de la ingeniería de software se basan en un conjunto de principios fundamentales que gobiernan cada área de la tecnología e incluyen actividades de modelación y otras técnicas descriptivas.

Las herramientas de la ingeniería de software proporcionan un apoyo automatizado o semiautomatizado para el proceso y los métodos. Cuando se integran las herramientas de modo que la información creada por una pueda ser utilizada por otra, queda establecido un sistema llamado ingeniería de software asistido por computadora que apoya el desarrollo de software (p. 12).

Según la página oficial de UNIR (2021) describe sobre la ingeniería de software como una disciplina de esta manera:

La ingeniería del software es una disciplina que implica el uso de estructuras, herramientas y técnicas para construir programas informáticos.

Así mismo, incluye el análisis previo de la situación, la redacción del proyecto, la creación del software y las pruebas necesarias para garantizar el correcto funcionamiento del software antes de poner el sistema en funcionamiento.

Esta ingeniería aborda todas las fases del ciclo de vida de desarrollo de cualquier tipo de sistema de información y es aplicable a una amplia gama de ámbitos de la informática y la ciencia de los ordenadores, como el diseño de compiladores, sistemas operativos y tecnologías de Intranet/Internet: La empresa, la investigación científica, la medicina, la fabricación, la logística, la banca, el control del tráfico y la meteorología son sólo algunos de los campos en los que puede trabajar. Podemos deducir que la ingeniería de software es un proceso legal para el desarrollo de software en donde el pedido del cliente o usuario se lo genera en requerimientos, y estos se van transformando en diseño que luego se emplea código que se prueba, luego se documenta y se certifica para su uso en producción (párr. 1).

Podemos deducir que, la ingeniería de software es el uso de técnicas y métodos para la construcción de un software, desde el inicio hasta el mantenimiento del producto desarrollado.

2.7.1 Proceso del desarrollo del software

El proceso de desarrollo de software es el conjunto de actividades y tareas que se realizan para crear un software de calidad y cumplir con los requisitos del usuario. Pressman (2010), define como la participación de todos los participantes involucrados en el sistema, a continuación veremos su definición:

Un proceso es un conjunto de actividades, acciones y tareas que se ejecutan cuando va a crearse algún producto del trabajo. Una actividad busca lograr un objetivo amplio (por ejemplo, comunicación con los participantes) y se desarrolla sin importar el dominio de la aplicación, tamaño del proyecto, complejidad del esfuerzo o grado de rigor con el que se usará la ingeniería de software. Una acción (diseño de la arquitectura) es un conjunto de tareas que producen un producto importante del trabajo (por ejemplo, un modelo del diseño de la arquitectura). Una tarea se centra en un objetivo pequeño pero bien definido (por ejemplo, realizar una prueba unitaria) que produce un resultado tangible. (p, 12).

Según Pressman (2010), explica del como esta estructurado el proceso del desarrollo del software:

La estructura del proceso establece el fundamento para el proceso completo de la ingeniería de software por medio de la identificación de un número pequeño de actividades estructurales que sean aplicables a todos los proyectos de software, sin importar su tamaño o complejidad. Además, la estructura del proceso incluye un conjunto de actividades sombrilla que son aplicables a través de todo el proceso del software. Una estructura de proceso general para la ingeniería de software consta de cinco actividades:

Comunicación. Antes de que comience cualquier trabajo técnico, tiene importancia crítica comunicarse y colaborar con el cliente (y con otros

participantes). Se busca entender los objetivos de los participantes respecto del proyecto, y reunir los requerimientos que ayuden a definir las características y funciones del software.

Planeación. Cualquier viaje complicado se simplifica si existe un mapa. Un proyecto de software es un viaje difícil, y la actividad de planeación crea un “mapa” que guía al equipo mientras viaja. El mapa llamado plan del proyecto de software define el trabajo de ingeniería de software al describir las tareas técnicas por realizar, los riesgos probables, los recursos que se requieren, los productos del trabajo que se obtendrán y una programación de las actividades.

Modelado. Ya sea usted diseñador de paisaje, constructor de puentes, ingeniero aeronáutico, carpintero o arquitecto, a diario trabaja con modelos. Crea un “bosquejo” del objeto por hacer a fin de entender el panorama general cómo se verá arquitectónicamente, cómo ajustan entre sí las partes constituyentes y muchas características más. Si se requiere, refina el bosquejo con más y más detalles en un esfuerzo por comprender mejor el problema y cómo resolverlo. Un ingeniero de software hace lo mismo al crear modelos a fin de entender mejor los requerimientos del software y el diseño que los satisfará.

Construcción. Esta actividad combina la generación de código (ya sea manual o automatizada) y las pruebas que se requieren para descubrir errores en éste.

Despliegue. El software (como entidad completa o como un incremento parcialmente terminado) se entrega al consumidor que lo evalúa y que le da retroalimentación, misma que se basa en dicha evaluación. (p. 13).

Las actividades mencionadas por Pressman (2010), son las mas usadas durante la creacion de sistemas web, ya sean los sistemas pequeños y sencillos o grandes de gran complejidad.

2.7.2 Modelos de proceso prescriptivo

Un modelo prescriptivo es un tipo de modelo que se utiliza en la toma de decisiones y que se basa en recomendaciones específicas para lograr un objetivo determinado. Este tipo de modelo se utiliza para proporcionar soluciones concretas y detalladas a un problema específico.

Pressman (2010) describe:

Los modelos de proceso prescriptivo fueron propuestos originalmente para poner orden en el caos del desarrollo de software. La historia indica que estos modelos tradicionales han dado cierta estructura útil al trabajo de ingeniería de software y que constituyen un mapa razonablemente eficaz para los equipos de software. Sin embargo, el trabajo de ingeniería de software y el producto que genera siguen “al borde del caos”. (p, 33).

Se utilizan para tomar decisiones en áreas como la planificación de la producción, la asignación de recursos, la optimización de procesos para que todo vaya según lo planificado.

2.7.2.1. Modelo de la cascada

En el modelo de la cascada consiste en que cada fase tiene que ser completada antes de pasar a la siguiente fase para que todo vaya en orden.

Hay veces en las que los requerimientos para cierto problema se comprenden bien: cuando el trabajo desde la comunicación hasta el despliegue fluye en forma razonablemente lineal. Esta situación se encuentra en ocasiones cuando deben hacerse adaptaciones o mejoras bien definidas a un sistema ya existente (por ejemplo, una adaptación para software de contabilidad que es obligatorio hacer debido a cambios en las regulaciones gubernamentales).

También ocurre en cierto número limitado de nuevos esfuerzos de desarrollo, pero sólo cuando los requerimientos están bien definidos y tienen una estabilidad razonable. (Pressman, 2010, p. 33).

Podemos decir que, para avanzar a la siguiente fase, tenemos que terminar primero la fase anterior para avanzar con el desarrollo del proyecto en desarrollo.

2.7.2.2. Modelo de proceso incremental

El modelo de proceso incremental, nos dice que tenemos que terminar una fase para avanzar a la siguiente fase.

El modelo incremental combina elementos de los flujos de proceso lineal y paralelo estudiados. El modelo incremental aplica secuencias lineales en forma escalonada a medida que avanza el calendario de actividades. Cada secuencia lineal produce “incrementos” de software susceptibles de entregarse de manera parecida a los incrementos producidos en un flujo de proceso evolutivo (Pressman, 2010, p. 33).

2.7.2.3. Modelo de proceso evolutivo

El modelo de proceso evolutivo, es iterativo, se le conoce por su manera de realizar versiones cada vez más sofisticadas del producto desarrollado.

Según Pressman(2010) el modelo de proceso evolutivo se le conoce como: “Los modelos evolutivos son iterativos. Se caracterizan por la manera en la que permiten desarrollar versiones cada vez más completas del software. En los párrafos que siguen se presentan dos modelos comunes de proceso evolutivo” (p. 39).

Pressman (2010), infiere que existen dos tipos de prototipos los cuales son:

Hacer prototipos. Es frecuente que un cliente defina un conjunto de objetivos generales para el software, pero que no identifique los requerimientos detallados para las funciones y características. En otros casos, el desarrollador tal vez no esté seguro de la eficiencia de un algoritmo, de la adaptabilidad de un sistema operativo o de la forma que debe adoptar la interacción entre el humano y la máquina. En estas situaciones, y muchas otras, el paradigma de hacer prototipos tal vez ofrezca el mejor enfoque.

El modelo espiral. Propuesto en primer lugar por Barry Boehm [Boe88], el modelo espiral es un modelo evolutivo del proceso del software y se acopla con la naturaleza iterativa de hacer prototipos con los aspectos controlados y sistémicos del modelo de cascada. Tiene el potencial para hacer un desarrollo rápido de versiones cada vez más completas. (p. 39).

2.8 METODOLOGÍA

La metodología, es un método para realizar un buen desarrollo del software, es de suma importancia para conseguir un buen trabajo, y existen una variedad de metodologías según el tipo de proyecto, para este proyecto nos estaremos enfocando en la metodología UWE, ya que es la que mejor se adapta para el presente proyecto.

2.8.1 Metodología UWE

“UWE es un enfoque de ingeniería de software para el dominio web que tiene como objetivo cubrir todo el ciclo de vida del desarrollo de aplicaciones web. El aspecto clave que distingue a UWE es la confianza en los estándares” (LMU – Ludwig-Maximilians-Universität München, 2016, párr. 1).

Para el señor Aurazo (2017) la metodología uwe es:

UWE (UML-Based Web Engineering) es una propuesta basada en UML y en el proceso unificado para modelar aplicaciones web. Esta propuesta está formada por una notación para especificar el dominio (basada en UML) y un modelo para llevar a cabo el desarrollo del proceso de modelado. Los sistemas adaptativos y la sistematización son dos aspectos sobre los que se enfoca UWE. (p. 1).

De la misma manera Minguez & García (2007) señala:

UWE UML (UML - Based Web Engineering) Es una metodología basada en el proceso unificado UML. Es una herramienta que permite identificar de mejor manera una aplicación Web, cubre todo el ciclo de vida de las aplicaciones Web. Su proceso de desarrollo es por fases: la fase de captura de requisitos, la fase de análisis y diseño de contenidos, la fase de modelo navegacional y la fase de implementación. Es una herramienta para modelar aplicaciones web, utilizadas en la ingeniería web, presentado especial atención en sistematización y personalización (sistemas adaptativos). UWE es una propuesta basada en el proceso unificado y UML, pero adaptados a la web. En requisitos separa las fases de captura, definición y validación. Hace además una clasificación y un tratamiento especial dependiendo del carácter de cada requisito. (p. 3).

La metodología UWE, es una metodología ágil y esta basado en el proceso unificado y UML para desarrollar las aplicaciones web, engloba todo el ciclo de vida de las aplicaciones o sistemas web. Esta metodología se utiliza para desarrollar sistemas o aplicaciones web libianas, que se puede desarrollar en un corto tiempo.

2.8.1.1. Características de UWE

Según Villarroel (2020) Sus características más relevantes en los que se fundamenta UWE, mencionaremos a continuación:

- Una de las características de UWE es el uso de una notación estándar, para todos los modelos Lenguaje de modelado unificado UML.
- Definición de métodos: UWE presenta una definición de los pasos para la construcción de los diferentes modelos.
- Especificación de Restricciones: en la metodología UWE, se recomienda el uso de restricciones en su desarrollo (p. 37).

2.8.1.2. Actividades de modelado UWE

Definiéremos cada uno de los modelos UWE.

2.8.1.2.1. Modelo de caso de uso

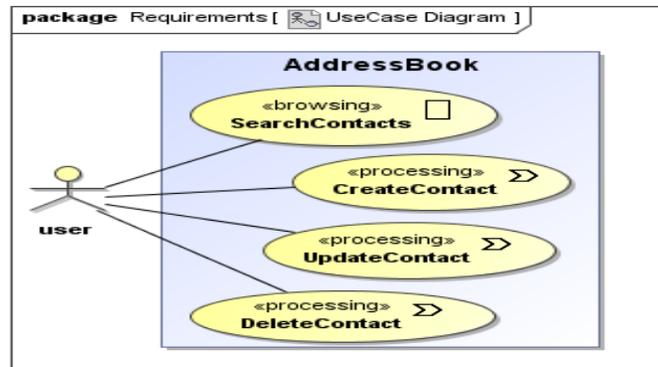
Según Villarroel (2020) define, “para describir los requerimientos funcionales de una aplicación se puede usar un modelo de caso de uso. Este describe un trozo de comportamiento de la aplicación sin revelar su estructura interna” (p. 38).

Villarroel (2020), lo describe en los siguientes puntos.

- Un caso de uso es una técnica de modelado usada para describir lo que debería hacer un sistema nuevo o lo que hace un sistema que ya existe.
- Los componentes primarios de un modelo de casos de uso (case-use model) son los casos de uso (use cases), los actores y el sistema modelado.
- Los casos de uso son descripciones funcionales del sistema; describen cómo los actores pueden usar un sistema (p. 38).

Figura 3

Representación gráfica de los casos de uso



Nota: tomado de UWE - UML - Based Web Engineering (párr. 1), por Ludwing-Maximilians, 2016, UWE.

2.8.1.2.2. Modelo conceptual

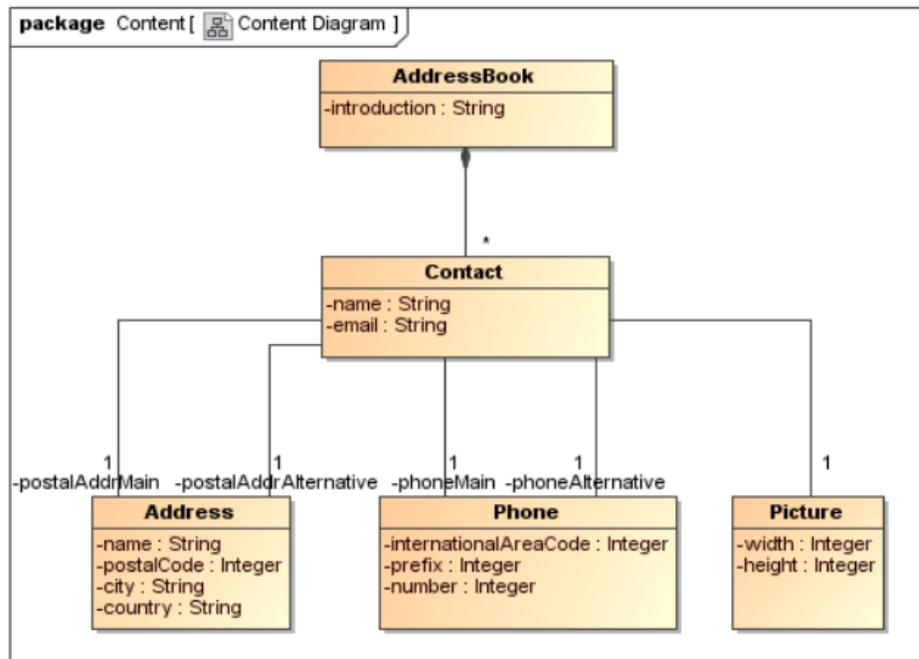
La descripción del modelo conceptual según Villarroel (2020) es:

Este modelo especifica cómo se encuentran relacionados los contenidos del sistema, es decir define la estructura de los datos que se encuentran alojados en el sitio web. Su objetivo es construir un modelo conceptual del dominio de la aplicación considerando los requisitos reflejados en los casos de uso. (p. 38).

El diagrama de clases de dominio lo podemos visualizar en la siguiente figura(4).

Figura 4

Modelo de contenido UWE



Nota: tomado de UWE - UML - Based Web Engineering (párr. 1), por Ludwing-Maximilians, 2016, UWE.

2.8.1.2.3. Modelo de navegación

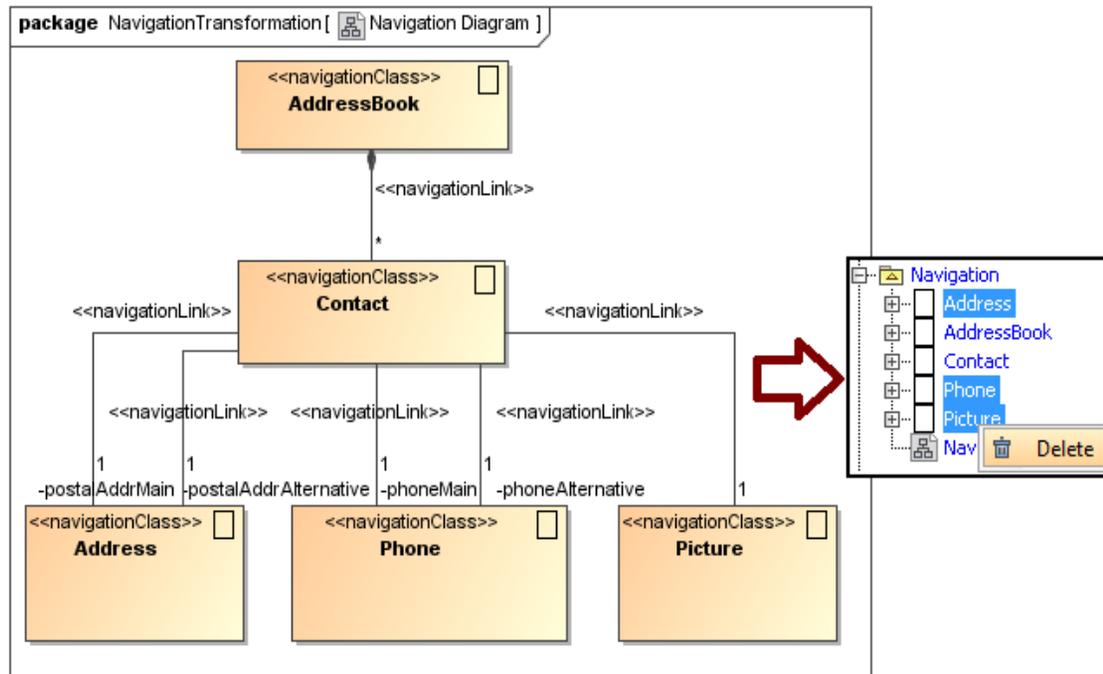
La descripción según Villarroel (2020) nos explica:

En un sistema para la web es útil saber cómo están enlazadas las páginas. Ello significa que necesitamos un diagrama conteniendo nodos (nodes) y enlaces (links). Se obtienen el modelo de espacio de navegación y modelo de estructura de navegación, que muestra cómo navegar a través del espacio de navegación. Se obtienen diagramas de clases que representan estos modelos. (p. 39).

Los podemos apreciar en la siguiente figura (5):

Figura 5

Modelo de navegación UWE



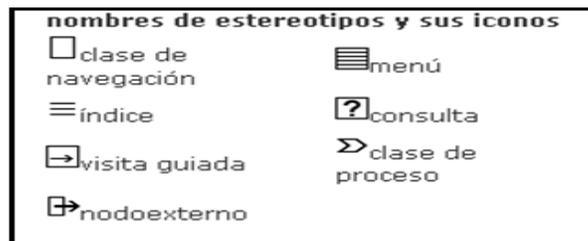
Nota: tomado de UWE - UML - Based Web Engineering (párr. 1), por Ludwing-Maximilians, 2016, UWE.

Según Villarroel (2020) la face de navegación se divide en dos areas:

- Modelo del espacio de navegación: basada en lo estructurado en la fase de conceptualización, es decir en los diagramas de clases. El modelo de navegación se lo realiza con los siguientes estereotipos.

Figura 6

Estereotipos de Iconos para el modelo de navegación de UWE



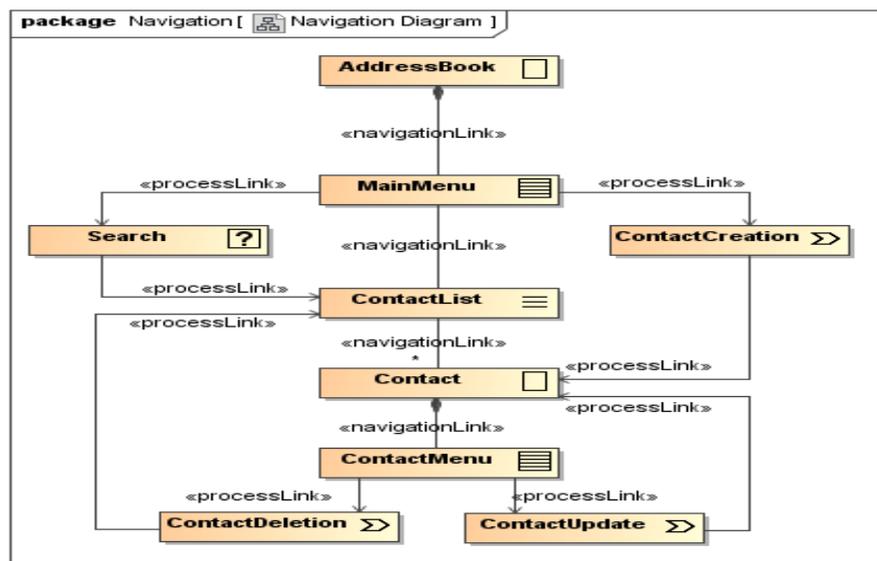
Nota: tomado de UWE - UML - Based Web Engineering (párr. 1), por Ludwing-Maximilians, 2016, UWE.

- Modelo de la estructura de navegación: Muestra la forma de navegar ante el espacio de navegación. Están constituidas por menús, índices, visitas guiadas, y formularios.
- Los índices es la colección de objetos permitiendo una navegación directa.
- Las visitas guiadas compuesta por grupo de referencias, permitiendo una navegación secuencial.
- Un menú es un elemento parte de la navegación con un número específico de conexiones a otros objetos.
- Un formulario facilita al usuario ingresar información para completar las condiciones de selección de objetos pertenecientes a las colecciones de índices y visitas guiadas.

Este refinamiento consiste en mejorar el modelo de espacio de navegación añadiendo estructuras de acceso como: índices, guías de ruta, consultas y menús.(p. 40). Y lo podremos ver en la siguiente figura(7):

Figura 7

Modelo de estructura de navegación de UWE



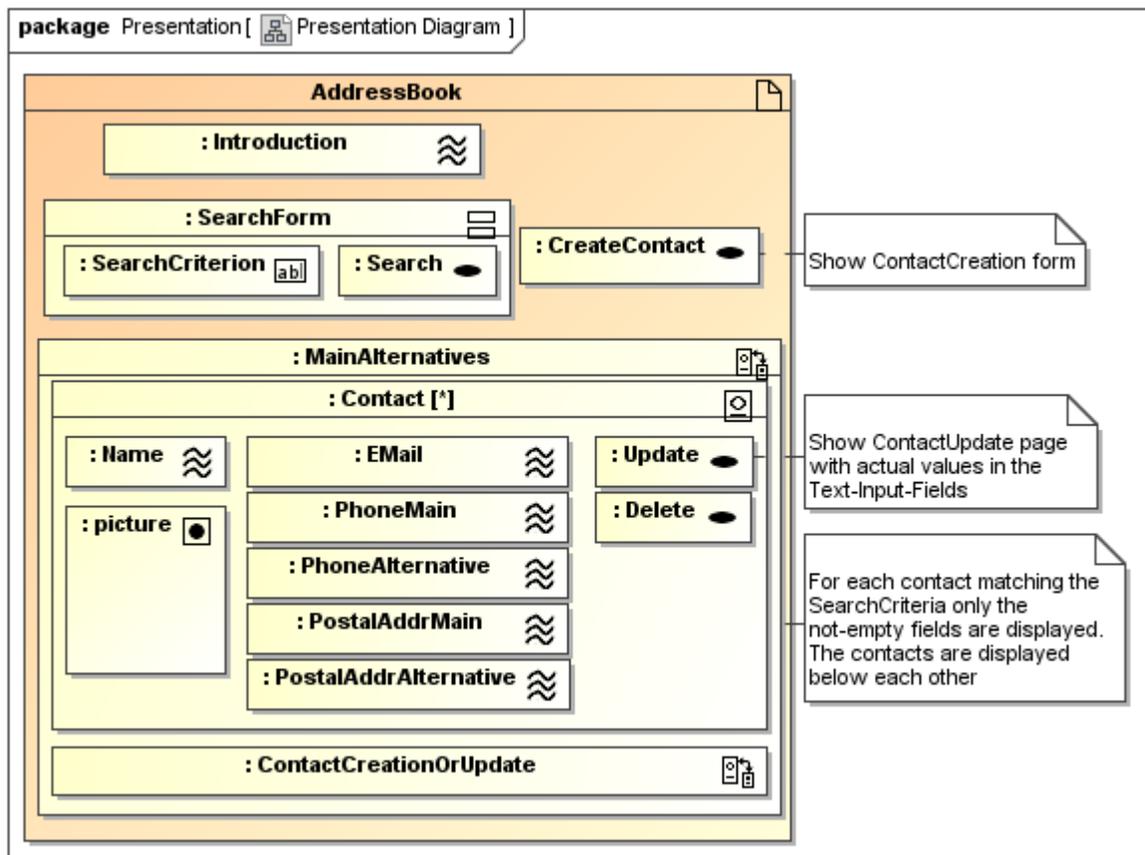
Nota: tomado de UWE - UML - Based Web Engineering (párr. 1), por Ludwing-Maximilians, 2016, UWE.

2.8.1.2.4 Modelo de presentación

Basándose en el modelo anterior y en información adicional recopilada durante el análisis de requerimientos se elabora el modelo de presentación, cuyo propósito es el diseño abstracto de interfaces de usuario. Se obtienen el modelo de espacio de navegación y modelo de estructura de navegación, que muestra cómo navegar a través del espacio de navegación. Se obtienen diagramas de clases que representan estos modelos (Villarroel, 2020, p. 41).

Figura 8

Modelo de presentación de UWE

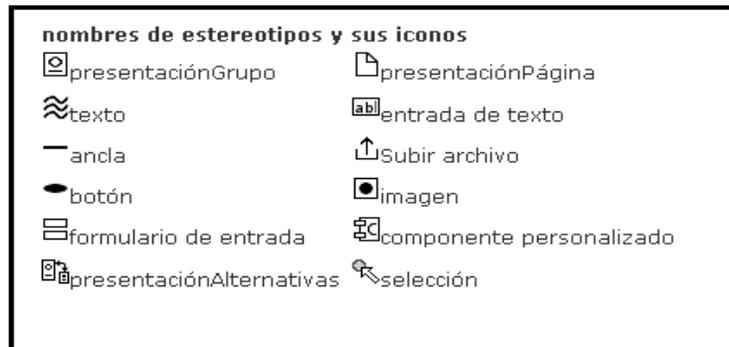


Nota: tomado de UWE - UML - Based Web Engineering (párr. 1), por Ludwing-Maximilians, 2016, UWE.

Este modelo presenta las siguientes características:

Figura 9

Estereotipos e Iconos para el modelo de presentación UWE



Nota: tomado de *UWE - UML - Based Web Engineering* (párr. 1), por Ludwing-Maximilians, 2016, UWE.

2.9 HERRAMIENTAS DE DESARROLLO DEL SOTFWARE

Las herramientas de desarrollo del software son los medios con los que se desarrollara el siguiente proyecto, tanto tangibles e intangibles.

2.9.1 Herramientas de desarrollo front end

Las herramientas de desarrollo front end es la parte que tiene interaccion con el cliente o usuario y lo mencionaremos a continuacion.

2.9.1.1. Html 5

Según Gauchat (2012):

HTML5 provee básicamente tres características: estructura, estilo y funcionalidad. Nunca fue declarado oficialmente pero, incluso cuando algunas APIs (Interface de Programación de Aplicaciones) y la especificación de CSS3 por completo no son parte del mismo, HTML5 es considerado el producto de la combinación de HTML, CSS y Javascript. Estas tecnologías son altamente dependientes y actúan como una sola unidad organizada bajo la especificación de HTML5. HTML está a cargo de la estructura, CSS presenta esa estructura y

su contenido en la pantalla y Javascript hace el resto que (como veremos más adelante) es extremadamente significativo.

Más allá de esta integración, la estructura sigue siendo parte esencial de un documento. La misma provee los elementos necesarios para ubicar contenido estático o dinámico, y es también una plataforma básica para aplicaciones. Con la variedad de dispositivos para acceder a Internet y la diversidad de interfaces disponibles para interactuar con la web, un aspecto básico como la estructura se vuelve parte vital del documento. Ahora la estructura debe proveer forma, organización y flexibilidad, y debe ser tan fuerte como los fundamentos de un edificio. (p. 1).

2.9.1.2. Css 3

Gauchat (2017) nos indica:

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. (p. 83).

2.9.1.3. Javascript

Gauchat (2017) señala:

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. (p. 241).

2.9.1.4. Bootstrap 5

Según el blog de 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. (párr. 2).

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 3, el framework se volvió más compatible con el desarrollo web responsive.

2.9.2 herramientas de desarrollo back end

Las herramientas de back end son los que ayudan a procesar la entrada desde el front end, es el que trabaja todo internamente sin que el usuario o el cliente lo vea y lo mencionaremos en seguida.

2.9.2.1. Php

SiteGround (2020):

PHP es el lenguaje de desarrollo web más popular. Es gratis, de código abierto y del nivel de servidor (el código se ejecuta en el servidor). PHP es un lenguaje a nivel de servidor de código abierto que es utilizado para crear páginas web dinámicas. Puede ser enlazado dentro de HTML. PHP se utiliza generalmente en conjunción con una base de datos MySQL en servidores web Linux/UNIX. Es probablemente el lenguaje de scripting más popular. (párr. 1).

2.9.2.2. MariaDB

Según HostingPlus (2020) explica sobre lo que es MariaDB:

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. (párr. 3).

2.9.2.3. Laravel 10

Altube (2021) indica:

Laravel es un framework de PHP para ayudarnos en un tipo de desarrollo sobre aplicaciones escritas en este lenguaje de programación. Este framework o más bien podría llamarlo compañero de ahora en adelante, nos ayuda en muchas cosas al desarrollar una aplicación, por medio de sus sistema de paquetes y de ser un framework del tipo MVC (Modelo-Vista-Controlador) da como resultado que podamos “despreocuparnos” (por así decirlo) en ciertos aspectos del desarrollo, cómo instanciar clases y métodos para usarlos en muchas partes de nuestra aplicación sin la necesidad de escribirlo y repetirlos muchas veces con lo que eso conlleva a la hora de modificar algo en el código. (párr. 2).

De la misma manera según la documentación oficial de Laravel LLC (2022) indica define a laravel como:

Laravel es un marco de aplicación web con una sintaxis expresiva y elegante. Un marco web proporciona una estructura y un punto de partida para crear su aplicación, lo que le permite concentrarse en crear algo increíble mientras nos preocupamos por los detalles.

2.10 MÉTRICAS DE CALIDAD

2.10.1 Calidad del Software

La calidad del software se refiere a la medida en que un producto de software cumple con los requisitos y expectativas del cliente, así como con los estándares y las normas de calidad establecidos. La calidad del software es un aspecto crucial en el desarrollo de software, ya que un producto de baja calidad puede tener efectos negativos en la satisfacción del cliente, la reputación de la empresa y los costos de mantenimiento.

Incluso los desarrolladores de software más experimentados estarán de acuerdo en que obtener software de alta calidad es una meta importante. Pero, ¿cómo se define la calidad del software? En el sentido más general se define como: Proceso eficaz de software que se aplica de manera que crea un producto útil que proporciona valor medible a quienes lo producen y a quienes lo utilizan. (Pressman, 2010, p. 340).

Según Pressman (2010) nos indica que existen tres importantes puntos en la calidad del software que lo mencionaremos a continuación:

- Un proceso eficaz de software establece la infraestructura que da apoyo a cualquier esfuerzo de elaboración de un producto de software de alta calidad. Los aspectos de administración del proceso generan las verificaciones y equilibrios que ayudan a evitar que el proyecto caiga en el caos, contribuyente clave de la mala calidad. Las prácticas de ingeniería de software permiten al desarrollador analizar el problema y diseñar una solución sólida, ambas actividades críticas de la construcción de software de alta calidad. Por último, las actividades sombrija, tales como administración del cambio y revisiones técnicas, tienen tanto que ver con la calidad como cualquier otra parte de la práctica de la ingeniería de software.
- Un producto útil entrega contenido, funciones y características que el usuario final desea; sin embargo, de igual importancia es que entrega estos activos en forma confiable y libre de errores. Un producto útil siempre satisface los requerimientos establecidos en forma explícita por los participantes. Además, satisface el conjunto de requerimientos (por ejemplo, la facilidad de uso) con los que se espera que cuente el software de alta calidad.
- Al agregar valor para el productor y para el usuario de un producto, el software de alta calidad proporciona beneficios a la organización que lo produce y a la comunidad de usuarios finales. La organización que elabora el software obtiene valor agregado porque el software de alta calidad requiere un menor esfuerzo de mantenimiento, menos errores que corregir y poca asistencia al cliente. Esto permite que los ingenieros de software dediquen más tiempo a crear nuevas aplicaciones y menos a repetir trabajos mal hechos. La comunidad de usuarios obtiene valor agregado porque la aplicación provee una capacidad útil en forma tal que agiliza algún proceso de negocios. El resultado final es 1) mayores utilidades por el producto de software, 2) más rentabilidad cuando una aplicación apoya un proceso de negocios y 3) mejor disponibilidad de información, que es crucial para el negocio. (p. 341).

2.10.2 Norma ISO/IEC 9126

La norma iso/iec 9126, evalua el producto software desarrollado, evañlua la calidad del desarrollo del software.

La norma ISO/IEC 9126, tiene como fin cuantificar los productos de software, esta norma nos indica las características de la calidad del software y fue elaborado para cubrir las necesidades de error que genera. La norma fue diseñada en los siguientes factores: calidad de proceso, calidad del producto, calidad del software y calidad de uso. Por otro lado, presenta dos modelos de calidad, el primero referido a la calidad interna y externa y el segundo modelo referido a la calidad en uso; a continuación se describe cada uno de ellos. (Acosta, et al., 2017, p. 76).

La norma ISO/IEC 9126 esta dividido en cuatro partes, las cuales estan encargados de direccionar, las metricas externas, metricas internas, la calidad en las metricas de uso y expedido.

Según Acosta, et al. (2017) las características definidas en la norma son las siguientes:

- **Usabilidad:** Se refiere a la capacidad de un producto software de ser entendido, aprendido y usado por los usuarios bajo condiciones específicas.
- **Funcionabilidad:** se refiere a la capacidad de un producto software de proporcionar funciones que ejecuten las necesidades explicitas e implicitas de los usuarios cuando el software es utilizado bajo condiciones específicas.
- **Confiabilidad:** se refiere a la capacidad de un producto software de mantener un nivel especificado de rendimiento cuando es usado bajo condiciones específicas.

- **Eficiencia:** se refiere a la relacion entre el grado de rendimiento de un producto y la cantidad de recursos (tiempo de cómputo, espacio de memoria, etc.) usados bajo ciertas condiciones.
- **Mantenimiento:** se refiere a la capacidad de un producto de software de ser modificado y probado.
- **Portatibilidad:** se refiere a la capacidad de un producto software de ser transferido de un ambiente operacional a otro. (p. 241).

Acosta, et al. (2017) indica que la “La norma ISO/IEC 9126 presenta en su definicion dos vistas, estas son la calidad externa e interna y la calidad de uso”. (p. 241).

2.10.2.1. Calidad interna o externa

Ambos conceptos estan relacionados;sin embargo, la obtencion de uno cualquiera de los dos no garantiza el logro del otro, son complementarios y su compocición genera el concepto complemento. (Acosta, et al., 2017, p. 241).

Según Acosta, et al. (2017) la calidad interna y externa se define como:

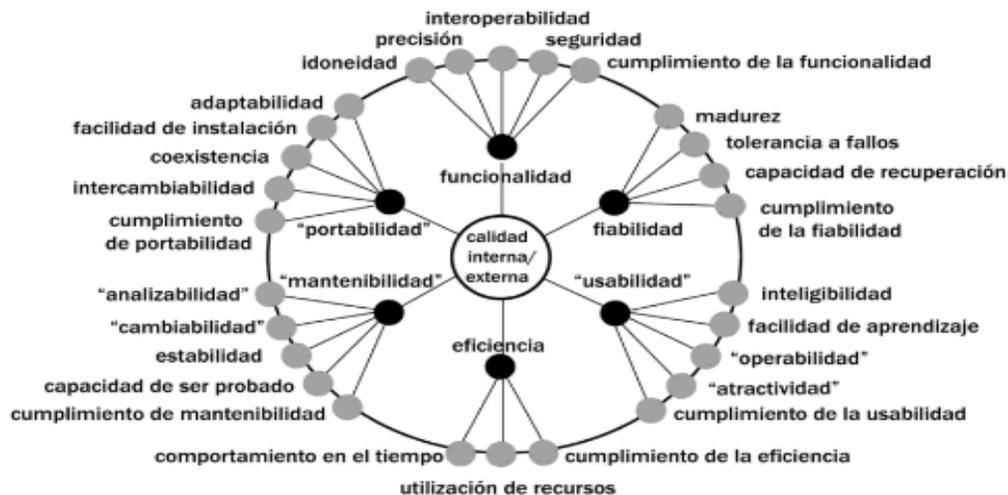
- **Calidad interna:** este concepto esta ligado a la vision de los desarrolladores del producto de software. Puede ser medida y calificada a travez de los atributos de componenetes y su documentacion generados durante el ciclo de la vida del desarrollo: i) Especificacion de requeriminetos, ii) Definicion de la arquitectura y diseño detallado, iii) Archivos de codigo fuente, etc. Estas metricas pueden obtenerse a lo largo de todo el ciclo de la vida como un indice de la evolucion de la calidad del producto en el proyecto.
- **Calidad externa:** este concepto esta ligado a la vision de los usuarios y clientes. Pueden ser medida y calificada por medio del comportamiento del codigo

ejecutable en un sistema de software en el ambiente operacional. Estas metricas se obtienen durante los primeros tiempos de la vida operativa del sistema en evaluacion a través de pruebas y encuestas de usuarios. (p. 241).

Las características, subcaracterísticas y sus correspondientes vinculos, lo podemos observar en la siguiente figura(10):

Figura 10

Vista de las características y sub características de la calidad interna y externa según el estándar iso/iec 9126



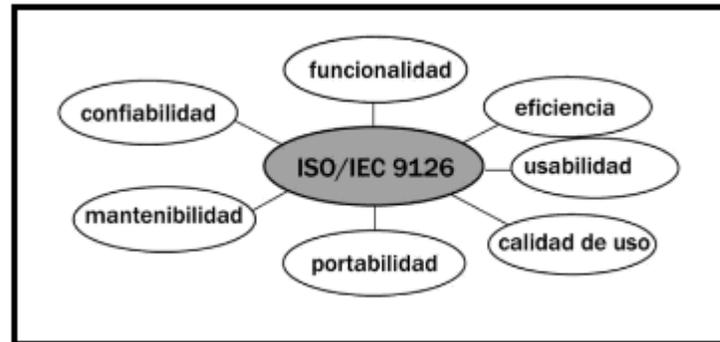
Nota: tomado de Estandares para la calidad de Software (p. 242), por Acosta, et al., 2017.

- **Calidad en uso:** el estandar ISO/IEC 9126 define calidad en uso como “la capacidad de un producto de software de facilitar a usuarios específicos alcanzar metas específicas con eficiencia, productividad, seguridad y satisfaccion en un contexto especifico de uso”. Resalta tambien que “calidad en uso es la vision de calidad de los usuarios de un ambiente contenido software medida sobre los resultados de usar el software en le ambiente, antes que sobre las propiedades del software en si mismo”.(p. 242).

En la siguiente figura los podemos observar la vista de calidad en uso como un componente complementario.

Figura 11

Calidad en uso como componente del modelo propuesto por el estándar iso/iec 9126



Nota: tomado de Estandares para la calidad de Software (p. 242), por Acosta, et al., 2017.

2.11 ESTIMACIÓN DE COSTOS PARA EL DESARROLLO DEL PROYECTO

Aston (2022) define:

Dentro de la gestión de proyectos, la estimación es simplemente un cálculo aproximado del esfuerzo y el costo que puede tomar completar un proyecto. No es una garantía del costo final; en cambio, proporciona al cliente una cotización o guía (a menudo un cálculo aproximado del alcance) de cuánto podría costar hacer o entregar algo para que puedan brindar el presupuesto para comenzar con el proyecto. Las estimaciones presupuestarias se realizan antes de que se inicie oficialmente un proyecto. (párr. 17).

2.11.1. El modelo COCOMO II

Según Pressman (2010) define que COCOMO II en realidad es una jerarquía de modelos de estimación que aborda las áreas siguientes:

- **Modelo de composición de aplicación.** Se usa durante las primeras etapas de la ingeniería de software, cuando son primordiales la elaboración de

prototipos de las interfaces de usuario, la consideración de la interacción del software y el sistema, la valoración del rendimiento y la evaluación de la madurez de la tecnología.

- **Modelo de etapa temprana de diseño.** Se usa una vez estabilizados los requisitos y establecida la arquitectura básica del software.
- **Modelo de etapa postarquitectónica.** Se usa durante la construcción del software. (p. 609).

Pressman (2010), explica sobre la estimación de costos: “Como todos los modelos de estimación para software, los modelos COCOMO II requieren información sobre dimensionamiento. Como parte de la jerarquía del modelo, están disponibles tres diferentes opciones de dimensionamiento: puntos objeto, puntos de función y líneas de código fuente”. (Pressman, 2010, p. 609).

Tabla 1

Ecuaciones por tipo de modelo cocomo II: 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) | $(P) = E/T$ | $(P) = E/T$ |

Nota: Fuente: Ingeniería de software

En donde donde:

- **E:** Esfuerzo requerido por el proyecto expresado en persona-mes.
- **T:** Tiempo requerido por el proyecto expresado en meses.
- **P:** Número de personas requeridas para el proyecto.
- **a, b, c y d:** Constantes con valores definidos según cada sub-modelo.
- **KLDC:** Cantidad de líneas de código distribuidas en miles.

Según Villarroel (2020) cada modelo se subdivide en modo, los mismos son:

- Modo orgánico: es un pequeño grupo de programadores experimentados desarrollando proyectos de software en un entorno familiar. El tamaño del software varía desde unos pocos miles de líneas (tamaño pequeño) a unas docenas de miles (tamaño medio).
- Modo semi - libre o semi - acoplado: Corresponde a un esquema intermedio entre modo orgánico y el rígido, el grupo de desarrollo puede incluir una mezcla de personas experimentadas y no experimentadas.
- Modo rígido o empotrado: El proyecto tiene fuertes restricciones, que pueden estar relacionadas con la funcionalidad y/o pueden ser técnicas. El problema a resolver es único, siendo difícil basarse en la experiencia puesto que puede no haberla. (p. 81).

En la siguiente tabla (2), se puede observar los coeficientes del proyecto de software según a los tres modos definidos anteriormente.

Tabla 2

Constante de coste modelo básico

| Proyecto del software | a _b | b _b | c _b | d _b |
|-----------------------|----------------|----------------|----------------|----------------|
| Orgánico | 2.4 | 1.05 | 2.5 | 0.38 |
| Semiseparado | 3.0 | 1.12 | 2.5 | 0.35 |
| Encajado | 3.6 | 1.20 | 2.5 | 0.32 |

Nota: Ingeniería de software

2.12 SEGURIDAD DEL SOFTWARE

Según Pressma (2010) la seguridad del software se lo define como:

La seguridad del software es una actividad del aseguramiento del software que se centra en la identificación y evaluación de los peligros potenciales que podrían afectarlo negativamente y que podrían ocasionar que falle todo el

sistema. Si los peligros se identifican al principio del proceso del software, las características de su diseño se especifican de modo que los eliminen o controlen. (p. 378).

2.12.1 Norma ISO/IEC 27001

ISO/IEC 27001 es un reconocido marco internacional de las mejores prácticas para un sistema de gestión de seguridad de la información. Le ayuda a identificar los riesgos para su información importante y pone en su lugar los controles apropiados para ayudarle a reducir el riesgo. (IBNORCA, 2022, p. 2).

Según Lone (2022) la norma ISO 27001 es:

La norma ISO/IEC 27001 es actualmente la norma internacional más reconocida para los sistemas de gestión de la seguridad de la información.

- Ayuda a las organizaciones a establecer la política y los objetivos de gestión de la seguridad de la información y a comprender cómo se pueden gestionar los aspectos importantes, aplicar los controles necesarios y establecer objetivos claros para mejorar la seguridad de la información.
- Permite a una organización gestionar su obligación de cumplir con los requisitos legales aplicables, como el GDPR (junto con la norma ISO 27701) y comprobar periódicamente el estado de cumplimiento. Esto permite una mejora continua del sistema para garantizar la protección y abordar las vulnerabilidades.
- Adopta un enfoque integral de la seguridad de la información. Los activos que necesitan protección van desde la información digital, los documentos en papel y los activos físicos (ordenadores y redes) hasta los conocimientos de los empleados individuales. Las cuestiones que hay que abordar van

desde el desarrollo de la competencia del personal a la protección técnica contra el fraude informático.

- La norma ISO 27001 está diseñada para ser compatible y armonizada con otras normas reconocidas de sistemas de gestión. Por lo tanto, es ideal para su integración en los sistemas y procesos de gestión existentes. (p. 4).

Lone (2022) nos explica sobre las ventajas del norma ISO/IEC 27001:

La norma adopta un enfoque global de la seguridad de la información y la protección de los activos. La norma ISO/IEC 27001 le ayudará a proteger su información en función de los siguientes principios:

- La confidencialidad garantiza que la información sólo sea accesible para las personas autorizadas a tener acceso.
- La integridad salvaguarda la exactitud y la exhaustividad de la información y los métodos de tratamiento.
- La disponibilidad garantiza que los usuarios autorizados tengan acceso a la información y a los activos asociados cuando sea necesario.
- La protección técnica contra el fraude informático. (p. 5).

La norma iso/iec 27001, trata de la seguridad de la información de un prototipo software, identifica el software de riesgos de información.

CAPÍTULO III

MARCO APLICATIVO

3. MARCO APLICATIVO

En el presente capítulo veremos, como interactúa los actores con la aplicación y así mismo veremos de forma detallada los procesos que se realizan para el desarrollo de la aplicación web y todo lo concerniente a los fundamentos técnicos para el análisis y diseño del proyecto a implementarse en Gobierno Autónomo Municipal de Batallas, la metodología a emplearse para todo el proceso del desarrollo del proyecto es la metodología ágil UWE, ya circunstanciado en capítulo II.

3.1 CAPTURA, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

La captura, análisis y especificación de requisitos es una parte fundamental del proceso de ingeniería de requisitos. En esta etapa, se recopilan y analizan los requisitos del sistema, se identifican las necesidades de los usuarios y se especifican los objetivos del sistema.

3.1.1 Obtención de requerimientos

La obtención de requisitos es fundamental para la recopilación de información y de esta manera realizar un buen desarrollo del presente proyecto a desarrollarse y en la siguiente tabla veremos la actividad que se realizó:

Tabla 3

Obtención de requisitos

| Tareas | Descripción |
|------------|---|
| Entrevista | La entrevista se realizó con el siguiente personal del GAMB <ul style="list-style-type: none">• Alcalde• Secretaria• Unidad de proyectos• Supervisor de proyectos• Dirigentes |

| | |
|---------------|---|
| Observación | Se pudo observar algunas dificultades en el proceso monitoreo y control de proyectos tanto por parte de funcionarios y dirigentes de comunidades. |
| Documentación | No se me permitió revisar la documentación por no ser un funcionario de la alcaldía. |

Nota: en la tabla se menciona a los funcionarios que se entrevistó.

3.1.2 Identificación de actores

La identificación de actores fue el que nos dio acceso a reconocer a los funcionarios encargados en el proceso de monitoreo y control de proyectos del Gobierno Autónomo Municipal de Batallas con el objetivo de formar los casos de uso.

Todo lo mencionado lo demostraremos en la siguiente tabla para un mejor entendimiento, sobre la relación de cada uno de los actores con las actividades que realizan en el municipio.

Tabla 4

Lista de Actores

| Actor | Descripción |
|---------|--|
| Alcalde | Es el funcionario a cargo de dar el visto bueno para la apertura de los proyectos, monitorea, controla el avance del proyecto, también es al quien acuden primero para presentar sus proyectos, en concreto es el funcionario al que más le hacen las peticiones o solicitudes frecuentemente. |

| | |
|-------------------------|--|
| Secretaria | El funcionario realiza la centralización de los informes para realizar los reportes sobre los proyectos para dar informe al alcalde. |
| Unidad de proyectos | Es el funcionario que se encarga de realizar el respectivo registro, monitoreo y clasificación de los tipos de proyectos que existen y también es encargado de revisar la documentación y dar vía libre a los proyectos. |
| Supervisor de proyectos | Es el funcionario que tiene el cargo de hacer el monitoreo y control de proyectos, también realiza informes sobre el avance del proyecto según el cronograma establecido de cada proyecto. |
| Dirigentes | Son Dirigentes representantes de las Comunidades pertenecientes al Gobierno Autónomo Municipal de Batallas los cuales dan el respectivo seguimiento sobre el proyecto. |

Nota: Se describe y explica la función de cada actor.

3.3 INGENIERIA DE REQUERIMIENTOS

La ingeniería de requisitos es una disciplina dentro de la ingeniería de software que se ocupa de la identificación, análisis, documentación y verificación de los requisitos del sistema. Estos requisitos son la base para el diseño, implementación y prueba del sistema, y son esenciales para asegurar que el sistema cumpla con las necesidades y expectativas de los usuarios y otros interesados.

3.3.1 Requerimientos funcionales

Los requerimientos funcionales son una lista de características, funcionalidades, y objetivos que deben ser cumplidos por un sistema o software para satisfacer las necesidades y expectativas de sus usuarios y/o clientes. Estos requerimientos describen las funcionalidades específicas que un sistema debe proveer y cómo debe comportarse en diferentes situaciones, como, por ejemplo, cómo se deben realizar determinadas operaciones, cómo se deben manejar los datos, y cómo debe interactuar con los usuarios. En la siguiente tabla (5) podremos visualizar los requerimientos funcionales.

Tabla 5

Requerimientos Funcionales

| Rol | Descripción | Función |
|------|----------------------------|---|
| R1-1 | Administración de Usuarios | Ingresar al sistema según sus funciones asignados, Administrador (unidad de proyectos), alcalde, secretaria, supervisor de proyectos y autoridades. |
| R1-2 | Administración de roles | Crear, listar actualizar y eliminar roles, la administración de roles permite la asignación de roles a cada funcionario según su función. |
| R1-3 | Administración de usuarios | Crear, listar, actualizar y eliminar o dar de baja a los usuarios que ya no son parte de la gestión. |
| R1-4 | Monitoreo de proyectos | Añadir nuevo proyecto para el monitoreo, se creará si tiene una empresa asignada y de lo |

| | | |
|-------|---------------------------|---|
| | | contrario no se va a poder, listar, actualizar y eliminar o dar de baja datos de todas las categorías de proyectos. |
| R1-5 | Directorio de empresas | Registrar datos de la empresa que ejecute un proyecto. Crear, listar, actualizar y eliminar. |
| R1-6 | Gestión de Poas | Añadir la POA actual para una determinada comunidad. Crear, listar, actualizar y eliminar. Añadir las fases de un proyecto mediante un cronograma en donde se podrá hacer seguimiento por imágenes. |
| R1-7 | Cronograma de actividades | Crear, listar, actualizar, eliminar y actualizar estado Añadir las actividades de cada fase en donde se podrá hacer el monitoreo mediante imágenes. |
| R1-8 | Actividades | |
| R1-9 | Tablero de indicadores | Datos estadísticos de la cantidad de proyectos por comunidad y cantón de la misma manera también se puede visualizar los datos estadísticos de cantidad de categorías por cantones, ver y reporte. |
| R1-10 | Cuenta | Permite personalizar la cuenta de un usuario. |
| R1-11 | Cerrar Sesión | Permite la acción de salir del sistema. |

| | | |
|-------|-----------------------|--|
| R1-12 | Proyectos aprobados | Añadir un nuevo proyecto aprobado por Honorable consejo. Crear, Editar, Eliminar Listar, Reporte, Buscar. |
| R1-13 | Avance de actividades | Se podrá visualizar de forma general todas las actividades que están en modo, Pendiente, desarrollo y concluidos. Las acciones a realizar son realizar el reporte del mismo |

Nota: Se describe las funcionalidades que realizara cada actor.

3.3.2 Requerimientos No funcionales

Los requisitos no funcionales son aquellos que describen las características del sistema que no están relacionadas directamente con su funcionamiento o comportamiento, sino que se refieren a otros aspectos como su calidad, rendimiento, seguridad, usabilidad y otros.

En la tabla (6) podemos visualizar los requerimientos no funcionales.

Tabla 6

Requerimientos no funcionales

| Rol | Función |
|------|---|
| R2-1 | El presente sistema debe ser compatible y funcionar en cualquier navegador como, por ejemplo, Chrome, Opera, Mozilla y los demás navegadores. |
| R2-2 | Mantenimiento adecuado del servidor. |
| R2-3 | Soporte y mantenimiento periódico del sistema para su buen y correcto funcionamiento. |

Nota: Describe el comportamiento y funcionamiento al sistema.

3.4 DESCRIPCIÓN DE ACTORES

En la tabla (7) veremos la descripción de cada uno de los actores involucrados en el monitoreo de proyectos.

Tabla 7

Descripción de los actores de casos de uso

| Rol | Descripción |
|-------------------------------------|---|
| Alcalde | <p>Es el encargado de monitorear y controlar los proyectos.</p> <p>Su participación del actor en el sistema es mínima ya que solo puede ingresar al sistema a solo ver el cronograma de actividades y al tablero de indicadores.</p> <p>El actor necesita solo necesita ver la información para tener una toma de decisión inmediata ya que todos se entrevistan directamente con el alcalde.</p> |
| Secretaria | <p>Es el actor que tiene la función de generar reportes.</p> <p>Su participación en el sistema es mínima ya que solo podrá generar reportes para mantener al tanto al alcalde.</p> |
| Administrador (Unidad de Proyectos) | <p>Es el encargado de administrar el registro de usuarios asignándolos sus respectivos roles a cada usuario y responsable de todos los proyectos.</p> <p>El actor tiene acceso a casi en la mayoría de módulos del sistema puede crear, actualizar e eliminar datos en todo el sistema.</p> |

Es el actor que requiere mayor información teniendo un monitoreo y control de los proyectos de las actividades en el municipio de Batallas.

Es el encargado de hacer el monitoreo y control de proyectos.

Supervisor de proyectos El actor tiene una mínima participación en el sistema ya que solo puede registrar, actualizar el cronograma de actividades y también puede generar reportes para su informe.

Dirigentes El dirigente tiene una mínima participación en el sistema solo puede ingresar a ver el listado de recursos, ver el cronograma de actividades, ver el tablero de actividades y generar algunos reportes de los módulos asignados.

Nota: Se describe el grado de participación de cada actor en el sistema.

3.5 DIAGRAMA DE CASOS DE USO

Los diagramas de casos de uso son una técnica de modelado utilizada en la ingeniería de software para representar las interacciones entre los usuarios y un sistema. Estos diagramas describen cómo se utiliza el sistema desde la perspectiva de los usuarios, es decir, qué funciones o acciones pueden realizar en el sistema y cómo interactúan con él.

Por medio del diseño de casos de usos representaremos el análisis de requerimientos del sistema, visualizaremos las interacciones de cada uno de los actores con el sistema.

3.5.1 Caso de uso General del sistema

En la figura (12), se puede visualizar la interacción que tendrá cada actor con el sistema.

Figura 12

Diagrama de caso de uso general del sistema

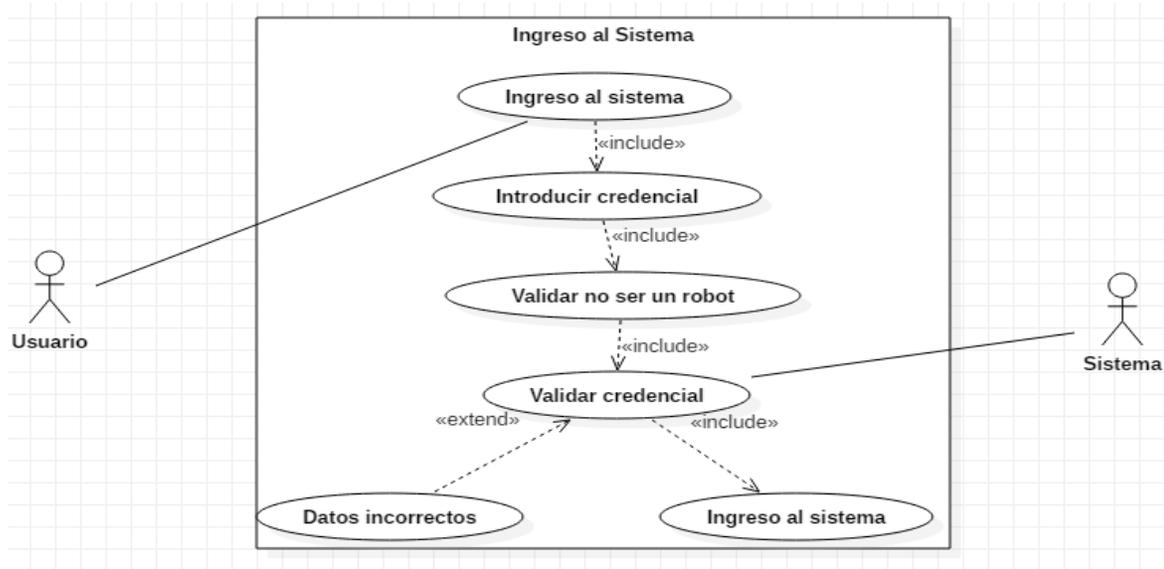


Nota: Elaboración Propia.

3.5.2 Caso de Uso – Ingreso al Sistema

Figura 13

Caso de uso ingreso al sistema



Nota: Elaboración Propia.

En la tabla (8) describiremos el ingreso al sistema:

Tabla 8

Descripción Caso De Uso Ingreso al Sistema

| Caso de Uso | Ingreso al Sistema |
|-------------|---|
| Actores | Administrador, Supervisor de proyectos, alcalde, secretaria y dirigentes. |
| Tipo | Primario esencial, Secundarios |
| Descripción | El usuario introduce su usuario y contraseña y validar que no es un robot para iniciar sesión en el sistema. El sistema valida las credenciales, si son correctos ingresa al sistema, caso contrario no. |

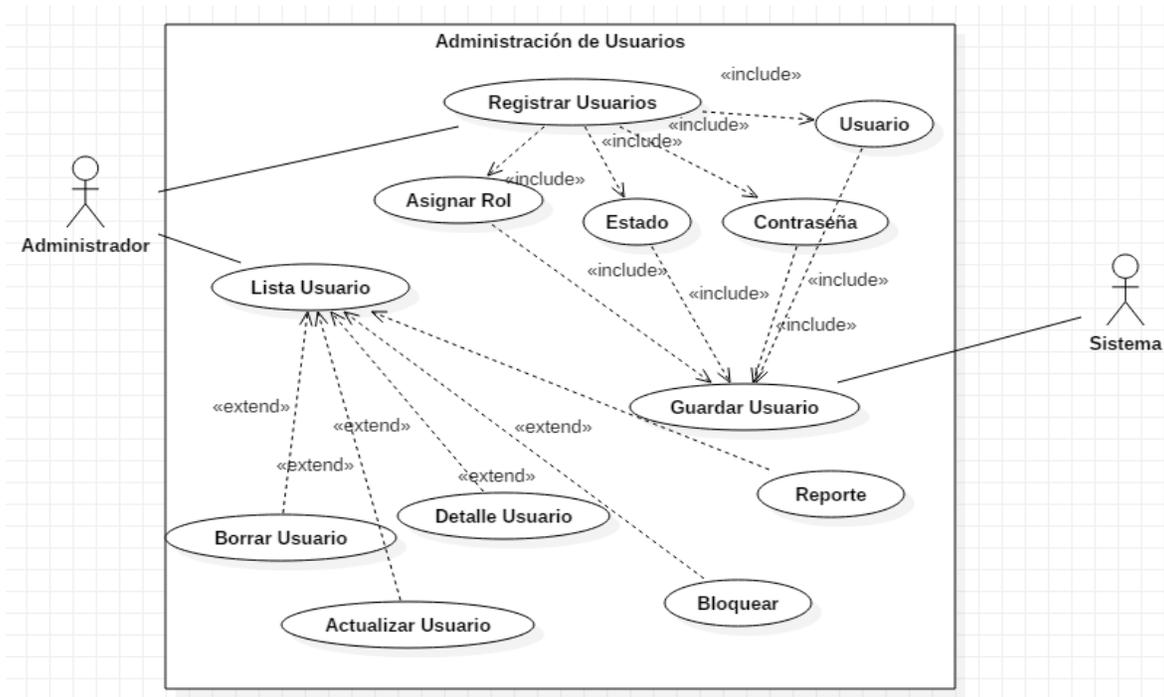
| | |
|---------------|---|
| | Los actores deben estar registrados en la base de datos. |
| Precondición | Los actores deben cumplir con los con los procesos de inicio de sesión. |
| Flujo normal | <ul style="list-style-type: none"> • El Administrador, Supervisor de proyectos, alcalde, secretaria, dirigentes, ingresan sus credenciales, marcan el captcha y oprimen el botón de ingreso al sistema. • El sistema verifica y valida los datos. |
| Postcondición | Datos correctos, ingreso al sistema caso contrario datos incorrectos, ingresar los datos correctos. |

Nota: Explica el comportamiento del sistema en el inicio de sesión. Elaboración propia.

3.5.3 Caso de Uso – Administración de Usuarios

Figura 14

Caso de uso administración de usuarios



Nota: Elaboración Propia

En la tabla (9) describiremos la Administración de usuarios.

Tabla 9*Caso de Uso – Administración de Usuario*

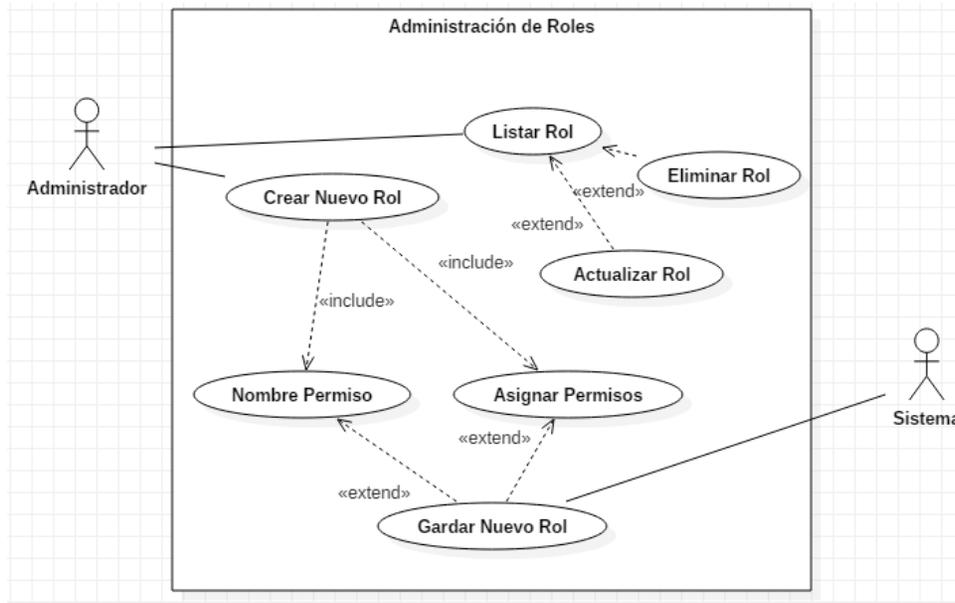
| Caso de Uso | Administración de Usuarios |
|---------------|---|
| Actores | Administrador, encargado de la unidad de proyectos. |
| Tipo | Primario Esencial |
| Descripción | El administrador registra y asigna el rol de cada usuario en base a su función en el sistema. También puede actualizar, eliminar, imprimir reportes, bloquear a exfuncionarios. |
| Precondición | Los actores deben estar registrados en la base de datos. Los actores deben ingresar al sistema mediante el proceso de inicio de sesión. |
| Flujo normal | <ul style="list-style-type: none">• El administrador ingresa al sistema.• El sistema muestra la primera interfaz gráfica, en donde se visualiza las opciones que incluye.• El administrador selecciona la opción de usuarios.• El sistema visualiza el listado de usuarios, la opción de crear nuevos usuarios, la opción de descargar reportes.• El administrador crear, edita, actualiza elimina.• El sistema guarda los cambios realizados. |
| Postcondición | Se creó, actualizó, eliminó y se guarda en la base de datos exitosamente. |

Nota: Explica el funcionamiento del sistema en el módulo de usuarios. Elaboración propia.

3.5.4 Caso de Uso – Administración de Roles

Figura 15

Caso de uso administración de roles



Nota: Elaboración propia.

En la tabla (10) describiremos la administración de roles:

Tabla 10

Caso de uso de administración de roles

| Caso de uso | Administración de roles |
|--------------|--|
| Actores | Administrador, Encargado de la Unidad de Proyectos. |
| Tipo | Primario Esencial |
| Descripción | El administrador crea y añade roles asignándoles sus respectivos permisos a cada rol de la misma manera que puede actualizar el rol y eliminar el rol. |
| Precondición | Los actores deben estar registrados en la base de datos. Los actores deben ingresar al sistema mediante el proceso de inicio de sesión. |

Flujo normal

- El administrador ingresa al sistema.
- El sistema muestra la primera interfaz gráfica, en donde se visualiza las opciones que incluye.
- El administrador selecciona la opción de roles.
- El sistema visualiza el listado de roles, la opción de crear nuevos roles, editar y eliminar rol.
- El administrador crear, edita, actualiza y elimina.
- El sistema guarda los cambios realizados.

Postcondición

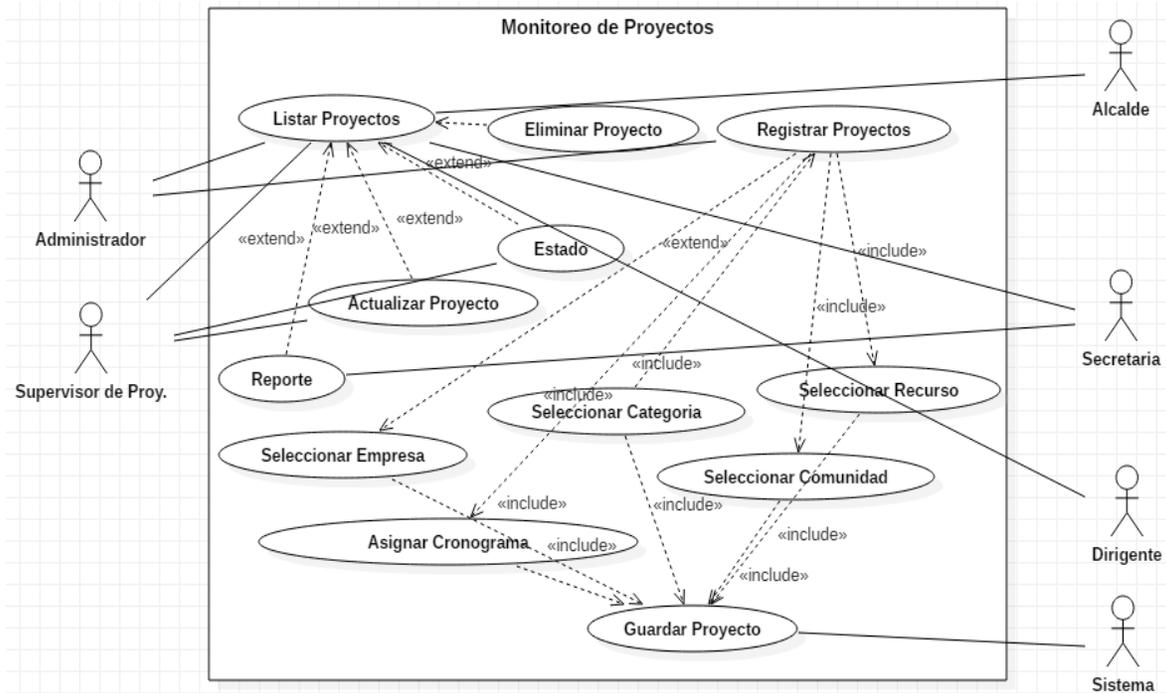
Se creó, actualizó, eliminó y se guarda en la base de datos exitosamente.

Nota: explica las funcionalidades del módulo de roles. Elaboración propia.

3.5.5 Caso de Uso – Monitoreo de Proyectos

Figura 16

Caso de uso monitoreo de proyectos



Nota: Elaboración propia

En la tabla (11) describiremos el Monitoreo de Proyectos:

Tabla 11

Caso de uso monitoreo de proyectos

| Caso de uso | Monitoreo de proyectos |
|--------------|--|
| Actores | Administrador, encargado de la unidad de proyectos, supervisor de proyectos, secretaria, alcalde y dirigente. |
| Tipo | Primario esencial y secundarios. |
| Descripción | El actor, registra y asigna y selecciona el recurso, empresa, comunidad, categoría y de la misma manera asigna Cronograma de actividades. También actualiza, Elimina e imprime reporte. Los actores deben estar registrados en la base de datos. |
| Precondición | Los actores deben ingresar al sistema mediante el proceso de inicio de sesión. <ul style="list-style-type: none"> • Los actores ingresan al sistema. • El sistema muestra la primera interfaz gráfica, en donde se visualiza las opciones que incluye de acuerdo a sus roles. • Los actores seleccionan la opción de Monitoreo de proyectos. |
| Flujo normal | <ul style="list-style-type: none"> • El sistema visualiza el listado de monitoreo de proyectos, la opción de crear nuevos roles, editar y eliminar según el rol asignado. • El administrador crear, edita, actualiza y elimina. • El supervisor de proyectos, actualiza. • La secretaria realizara reporte. • El dirigente ve el listado de proyectos. • El sistema guarda los cambios realizados. |

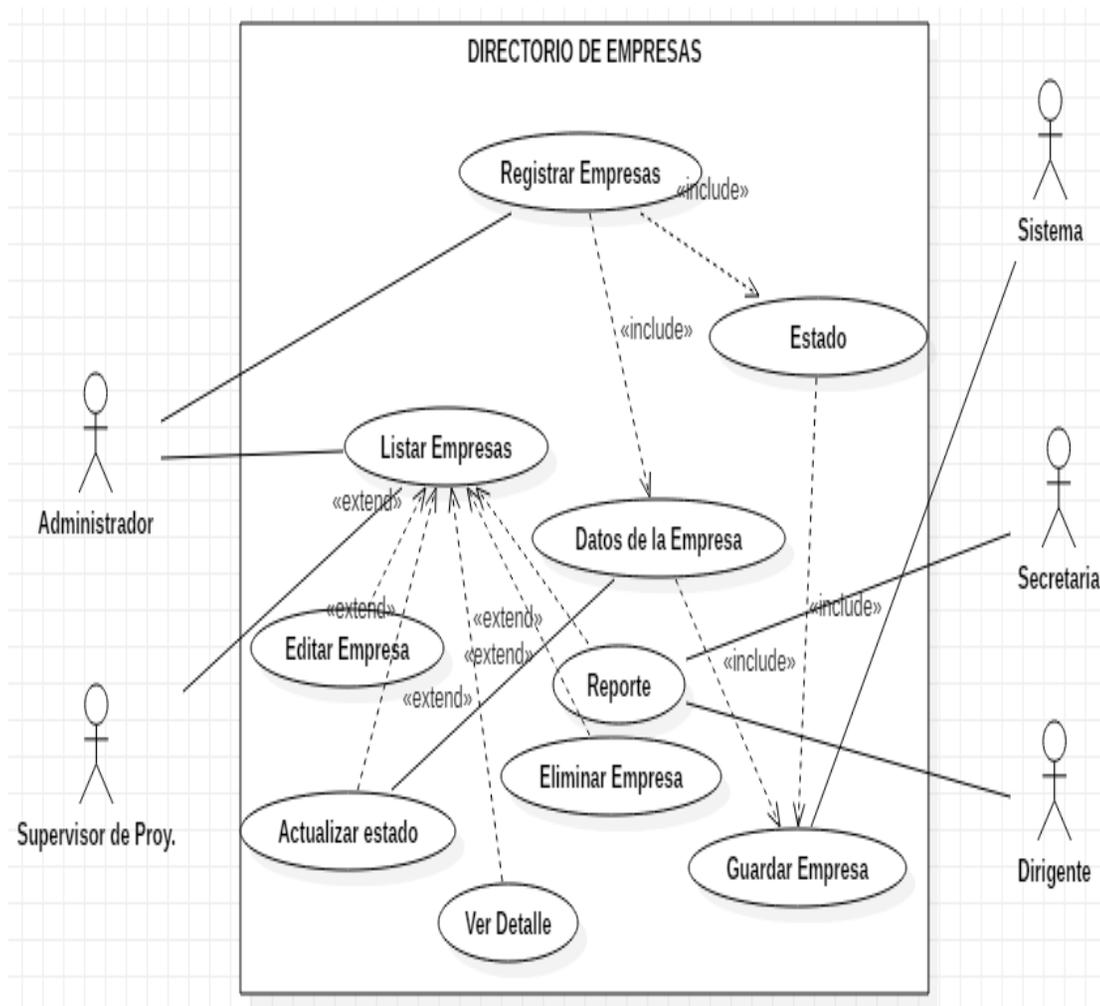
Postcondición Se creó, actualizó, eliminó, reporte y se guarda en la base de datos exitosamente.

Nota: Describe las funcionalidades del módulo de proyectos.

3.5.6 Caso de Uso – Directorio de Empresas

Figura 17

Caso de uso directorio de empresas



Nota: Elaboración propia

En la tabla (12) describiremos Directorio de Empresas:

Tabla 12*Caso de Uso de Directorio de Empresas*

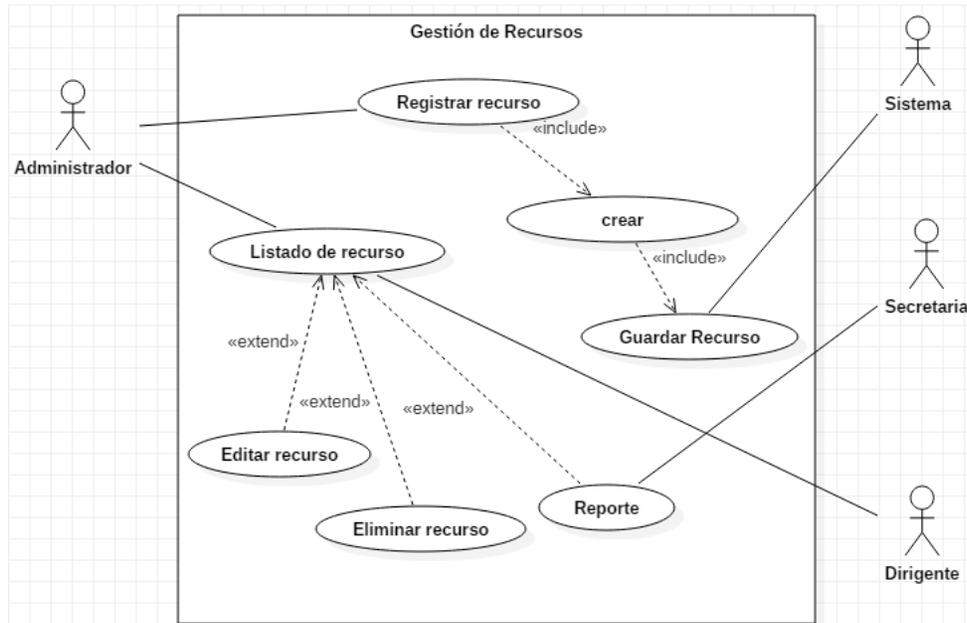
| Caso de uso | Directorio de empresas |
|---------------|--|
| Actores | Administrador, encargado de la unidad de proyectos, supervisor de proyectos, secretaria y dirigente. |
| Tipo | Primario esencial y Secundarios. |
| Descripción | El actor primario podrá, Registrar y listar empresas, también realiza la acción de editar y eliminar Empresa. Los actores secundarios podrán realizar reportes. Los actores deben estar registrados en la base de datos. |
| Precondición | Los actores deben ingresar al sistema mediante el proceso de inicio de sesión. <ul style="list-style-type: none">• Los actores ingresan al sistema.• El sistema muestra la primera interfaz gráfica, en donde se visualiza las opciones que incluye de acuerdo a sus roles.• Los actores seleccionan la opción de directorio de empresas. |
| Flujo normal | <ul style="list-style-type: none">• El sistema visualiza el listado de directorio de empresas, la opción de crear nuevos roles, editar y eliminar según el rol asignado.• El administrador crear, edita, actualiza y elimina.• El supervisor de proyectos, visualiza y realiza reporte.• La secretaria realizara reportes.• El dirigente visualiza listado de proyectos.• El sistema guarda los cambios realizados. |
| Postcondición | Se creó, actualizó, eliminó, reporte y se guarda en la base de datos exitosamente. |

Nota: Explica las funciones del módulo de empresas. Elaboración propia.

3.5.8 Caso de Uso – Gestión de Recursos

Figura 18

Caso de uso gestión de recursos



Nota: Elaboración propia

En la tabla (13) describiremos la Gestión de Recursos:

Tabla 13

Caso de uso gestión de recursos

| Caso de uso | Gestión de recursos |
|--------------|---|
| Actores | Administrador, Encargado de la unidad de proyectos, secretaria, encargado de informar al alcalde y dirigente. |
| Tipo | Primario esencial y Secundarios |
| Descripción | El actor Primario esencial registra, Edita, actualiza estado, Elimina e imprime el reporte. |
| Precondición | El actor secundario ingresa al módulo y puede imprimir el reporte. |
| | Los actores deben estar registrados en la base de datos. |

Los actores deben ingresar al sistema mediante el proceso de inicio de sesión.

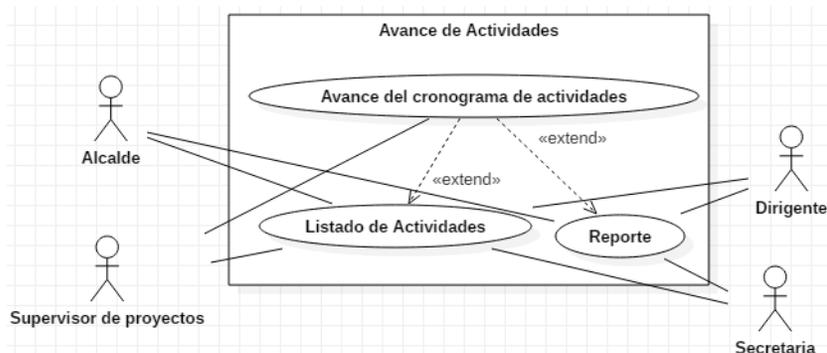
- Los actores ingresan al sistema.
 - El sistema muestra la primera interfaz gráfica, en donde se visualiza las opciones que incluye de acuerdo a sus roles.
 - Los actores seleccionan la opción de gestión de recursos.
- Flujo normal
- El sistema visualiza el listado de gestión de recursos, la opción de crear nuevos recursos, editar y eliminar según el rol asignado.
 - El administrador crear, edita, actualiza y elimina.
 - La secretaria realizara reportes.
 - El dirigente visualiza listado de proyectos.
 - El sistema guarda los cambios realizados.
- Postcondición
- Se creó, actualizó, eliminó, reporte y se guarda en la base de datos exitosamente.

Nota: Se describe las funcionalidades del módulo recursos. Elaboración propia.

3.5.9 Caso de Uso – Avance del cronograma de actividades

Figura 19

Caso de uso avance de actividades



Nota: Elaboración propia.

En la tabla (14) describiremos la Avance del cronograma de actividades:

Tabla 14

Caso de uso Avance de actividades

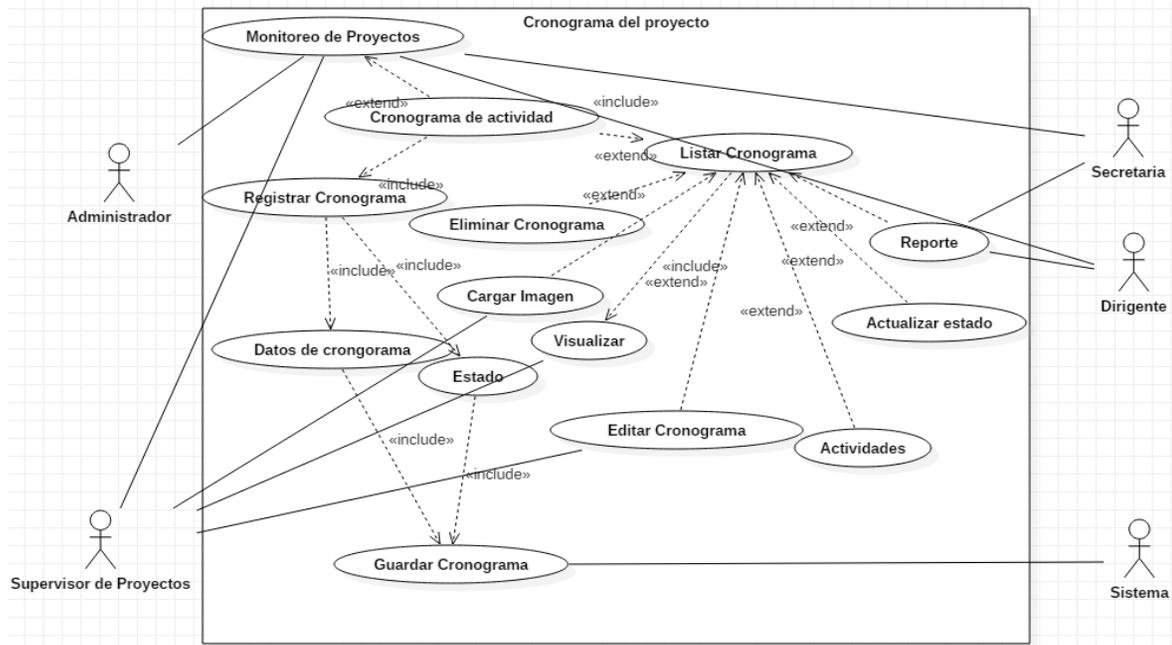
| Caso de uso | Avance de actividades |
|---------------|--|
| Actores | Alcalde, Dirigente, Supervisor de proyectos y secretaria. |
| Tipo | Secundarios |
| Descripción | Los actores secundarios ingresan al módulo, verificar el avance de los proyectos y pueden imprimir reporte del mismo. |
| Precondición | Los actores deben estar registrados en la base de datos. |
| Flujo normal | Los actores deben ingresar al sistema mediante el proceso de inicio de sesión. <ul style="list-style-type: none"> • Los actores ingresan al sistema. • El sistema muestra la primera interfaz gráfica, en donde se visualiza las opciones que incluye de acuerdo a sus roles. • Los actores seleccionan la opción de avance de actividades. • El sistema visualiza el listado de avance de actividades, la opción de reporte. • La secretaria realizara reportes. • El dirigente visualiza listado de proyectos. |
| Postcondición | Se descarga el reporte exitosamente. |

Nota: Se describe las funciones de cada actor en el sistema.

3.5.10 Caso de Uso – Cronograma del Proyecto

Figura 20

Caso de uso cronograma del proyecto



Nota: Elaboración propia

En la tabla (15) *Cronograma del proyecto*:

Tabla 15

Cronograma del proyecto

| Caso de uso | Cronograma del proyecto |
|--------------|--|
| Actores | Administrador, supervisor de proyectos, dirigente y secretaria. |
| Tipo | Primario esencial y Secundarios. |
| Descripción | El actor primario crea, edita, elimina, reporte y agrega imagen. El actor supervisor de proyectos actualiza y los actores dirigente y secretaria pueden generar reportes. Los actores deben estar registrados en la base de datos. |
| Precondición | Los actores deben ingresar al sistema mediante el proceso de inicio de sesión. |

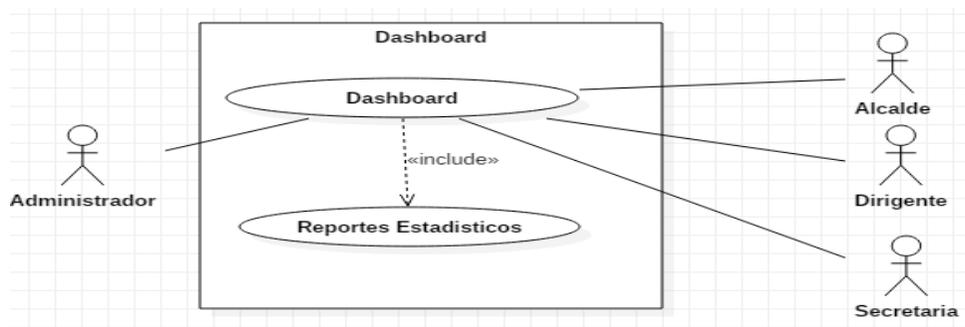
- Los actores ingresan al sistema.
 - El sistema muestra la primera interfaz gráfica, en donde se visualiza las opciones que incluye de acuerdo a sus roles.
 - Los actores seleccionan la opción de Monitoreo de proyectos e ingresan a la opción de cronograma de actividad.
- Flujo normal
- El sistema visualiza el listado de cronogramas de actividades, la opción de crear, editar y eliminar según el rol asignado.
 - El administrador crear, edita, actualiza y elimina.
 - El supervisor de proyectos, actualiza.
 - La secretaria realizara reporte.
 - El dirigente ve el listado de proyectos.
 - El sistema guarda los cambios realizados.
- Postcondición
- Se creó, actualizó, eliminó, reporte y se guarda en la base de datos exitosamente.

Nota: Describe las funciones que tendrá cada actor en cada módulo. Elaboración propia.

3.5.11 Caso de Uso – Tablero de Indicadores

Figura 21

Caso de uso tablero de indicadores



Nota: Elaboración propia.

En la tabla (16) describiremos el Tablero de Indicadores:

Tabla 16

Tablero de indicadores

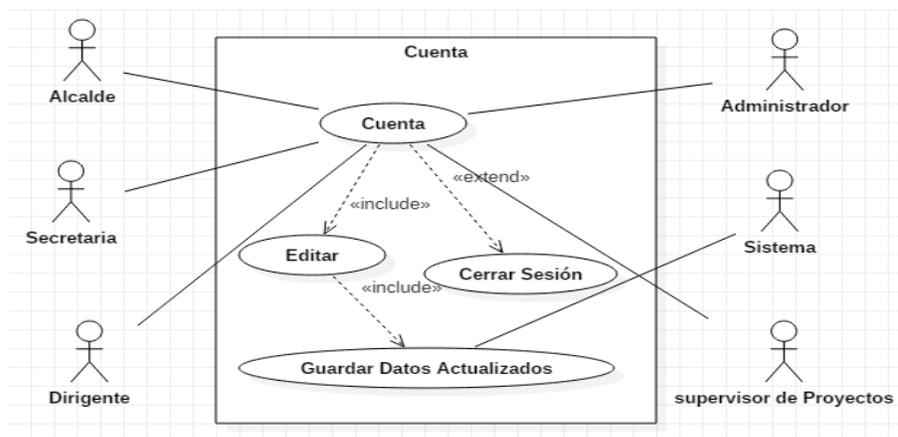
| Caso de uso | Dashboard |
|---------------|--|
| Actores | Administrador, Dirigente, alcalde y secretaria. |
| Tipo | Primario esencial y secundarios |
| Descripción | Los actores pueden visualizar y generar reporte. Los actores deben estar registrados en la base de datos. |
| Precondición | Los actores deben ingresar al sistema mediante el proceso de inicio de sesión. <ul style="list-style-type: none"> • Los actores ingresan al sistema. • El sistema muestra la primera interfaz gráfica. |
| Flujo normal | <ul style="list-style-type: none"> • Los actores seleccionan la opción de Dashboard. • El sistema visualiza datos estadísticos. • Secretaria y dirigente genera reporte. |
| Postcondición | realiza la descarga del reporte exitosamente. |

Nota: describe en datos estadísticos la información de proyectos.

3.5.12 Caso de Uso – Cuenta

Figura 22

Caso de uso cuenta



Nota: Elaboración propia.

En la tabla (18) describiremos la Cuenta:

Tabla 17

Cuenta

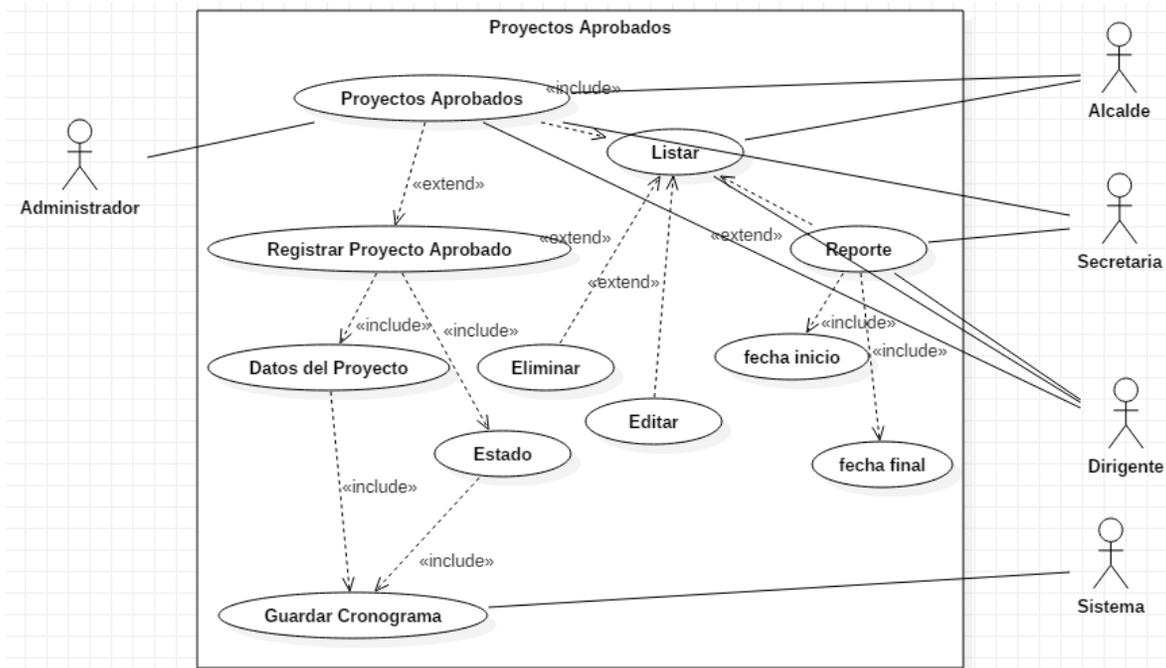
| Caso de uso | Cuenta |
|---------------|---|
| Actores | Administrador, supervisor de proyectos, alcalde secretaria, dirigente. |
| Tipo | Primario esencial y Secundarios |
| Descripción | Administrador, supervisor de proyectos, alcalde, secretaria y dirigente, realizan la acción de editar sus cuentas y podrán salir de la cuenta. |
| Precondición | Los actores deben estar registrados en la base de datos. Los actores deben ingresar al sistema mediante el proceso de inicio de sesión. |
| Flujo normal | <ul style="list-style-type: none"> • Los actores ingresan al sistema. • El sistema muestra la primera interfaz gráfica, en donde se visualiza las opciones que incluye de acuerdo a sus roles. • Los actores seleccionan del logo. • El sistema despliega la opción de perfil y cerrar sesión. • Los actores seleccionan perfil o cerrar sesión. • Actualizar datos. • El sistema guarda los cambios realizados o el sistema sierra la sesión. |
| Postcondición | Se actualizó y se guarda en la base de datos o salió del sistema exitosamente. |

Nota: Se describe las acciones que realizaran los actores en el módulo cuenta.

3.5.13 Caso de Uso – Proyectos Aprobados

Figura 23

Caso de uso Proyectos Aprobados



Nota: Elaboración propia

En la Tabla (19) describiremos la figura expuesta:

| Caso de Uso | Proyectos Aprobados |
|--------------|--|
| Actores | Administrador, supervisor de proyectos, alcalde secretaria, dirigente. |
| Tipo | Primario esencial y Secundarios |
| Descripción | Administrador tiene todos los privilegios, supervisor de proyectos puede actualizar, secretaria genera reportes, alcalde y dirigente podrá ver y listar. Los actores deben estar registrados en la base de datos. |
| Precondición | Los actores deben ingresar al sistema mediante el proceso de inicio de sesión. |

- Los actores ingresan al sistema.
- El sistema muestra la primera interfaz gráfica, en donde se visualiza las opciones que incluye de acuerdo a sus roles.
- Los actores seleccionan la opción de proyectos aprobados.
- El sistema visualiza el listado de proyectos aprobados, la opción de crear, editar y eliminar según el rol asignado.
- El administrador crear, edita, actualiza y elimina.
- La secretaria realizara reporte.
- El dirigente ve el listado de proyectos aprobados.
- El sistema guarda los cambios realizados.

Flujo normal

Postcondición

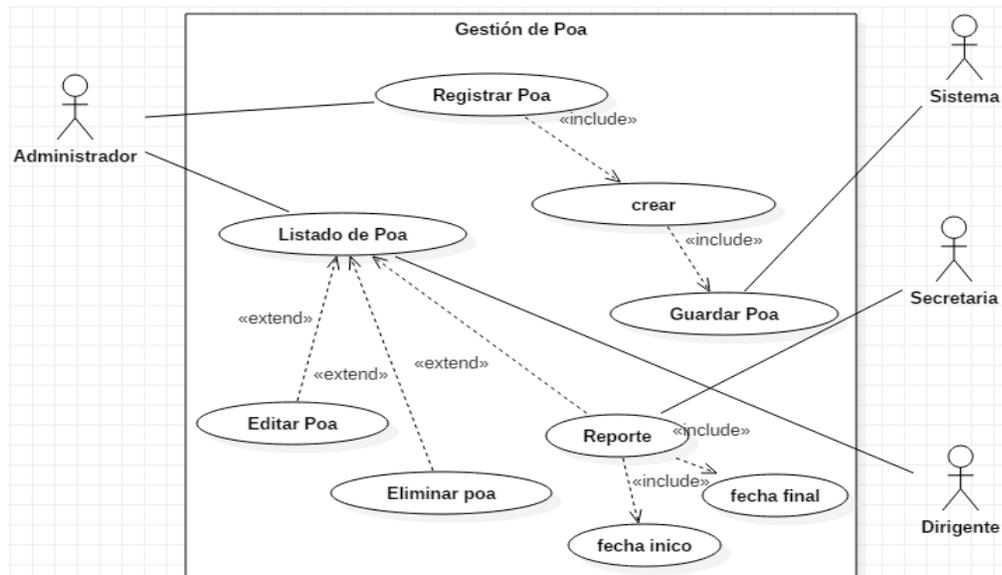
Se creó, actualizó, eliminó, reporte y se guarda en la base de datos exitosamente.

Nota: Se describe las funciones de cada actor en el sistema.

3.5.14 Caso de Uso – Gestión de Poa

Figura 24

Caso de uso Gestión de POA



Nota: Elaboración propia.

En la tabla (18) describiremos la Gestión de POA:

Tabla 18

Caso de uso Gestión POA

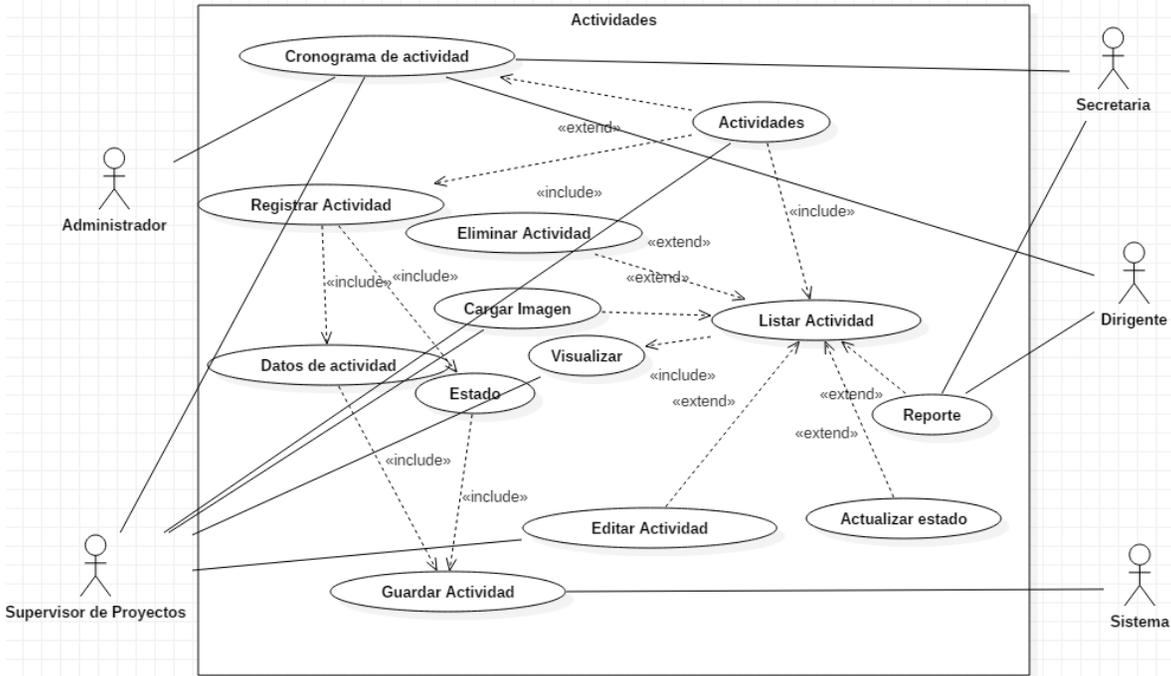
| Caso de uso | Gestión de POA |
|---------------|---|
| Actores | Administrador, supervisor de proyectos, alcalde secretaria, dirigente. |
| Tipo | Primario esencial y Secundarios |
| Descripción | El actor Primario esencial registra, Edita, actualiza estado, Elimina e imprime el reporte. El actor secundario ingresa al módulo y puede imprimir el reporte. Los actores deben estar registrados en la base de datos. |
| Precondición | Los actores deben ingresar al sistema mediante el proceso de inicio de sesión. |
| Flujo normal | <ul style="list-style-type: none"> • Los actores ingresan al sistema. • El sistema muestra la primera interfaz gráfica, en donde se visualiza las opciones que incluye de acuerdo a sus roles. • Los actores seleccionan la opción de gestión de POA. • El sistema visualiza el listado de gestión de POA, la opción de crear nuevos recursos, editar y eliminar según el rol asignado. • El administrador crear, edita, actualiza y elimina. • La secretaria realizara reportes. • El dirigente visualiza listado de gestión de POA. • El sistema guarda los cambios realizados. |
| Postcondición | Se creó, actualizó, eliminó, reporte y se guarda en la base de datos exitosamente. |

Nota: Se describe las funcionalidades del módulo de POA. Elaboración Propia.

3.5.15 Caso de Uso – Actividades

Figura 25

Caso de uso Actividades



Nota: Elaboración Propia

En la tabla (15) *Actividades*:

Tabla 19

Actividades

| Caso de uso | Actividades |
|-------------|---|
| Actores | Administrador, supervisor de proyectos, dirigente y secretaria. |
| Tipo | Primario esencial y Secundarios. |
| Descripción | El actor primario registra las actividades. Los actores secundarios, supervisor de proyectos es el que da el monitoreo a los proyectos por lo tanto hace el seguimiento y actualiza los estados, actualiza la imagen y puede editar para la observación. |

| | |
|---------------|---|
| | El dirigente y la secretaria pueden visualizar la imagen y sacar reportes. |
| | Los actores deben estar registrados en la base de datos. |
| Precondición | Los actores deben ingresar al sistema mediante el proceso de inicio de sesión. <ul style="list-style-type: none"> • Los actores ingresan al sistema. • El sistema muestra la primera interfaz gráfica, en donde se visualiza las opciones que incluye de acuerdo a sus roles. • Los actores seleccionan la opción de Monitoreo de proyectos e ingresan a la opción de cronograma de actividad y luego a actividades. |
| Flujo normal | <ul style="list-style-type: none"> • El sistema visualiza el listado de de actividades, la opción de crear, editar y eliminar según el rol asignado. • El administrador crear, edita, actualiza y elimina. • El supervisor de proyectos, actualiza. • La secretaria realizara reporte. • El dirigente ve el listado de Actividades. • El sistema guarda los cambios realizados. |
| Postcondición | Crea, actualiza, elimina, reporte y se guarda en la base de datos exitosamente. |

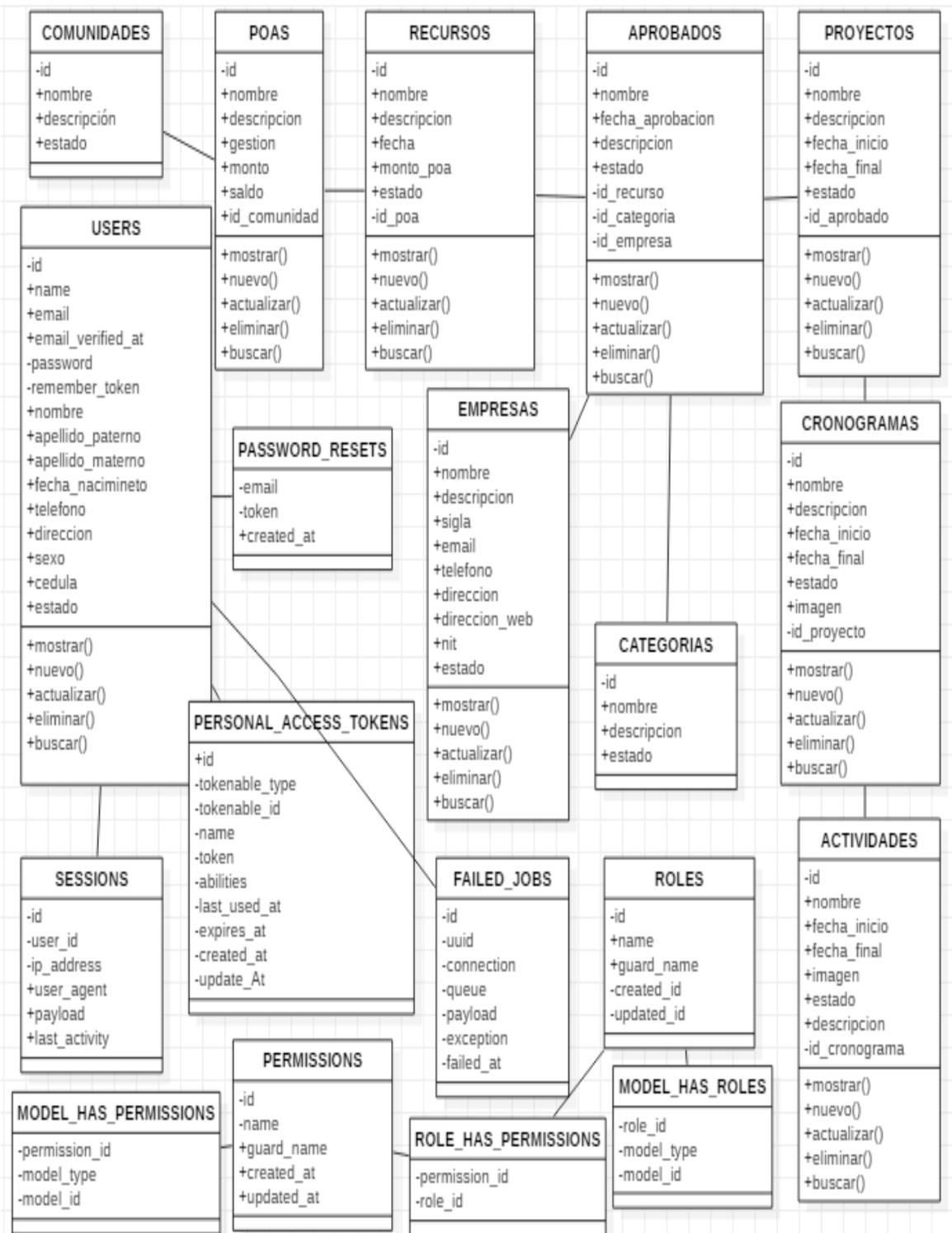
Nota: Describe las funciones que tendrá cada actor en el módulo actividad. Elaboración propia.

3.6 DIAGRAMA DE CLASES

Este diagrama de clase nos permite diseñar y tener una buena comprensión de la estructura de una base de datos, como veremos a continuación.

Figura 26

Diagrama de clases



Nota: Elaboración Propia.

3.7 MODELO DE BASE DE DATOS

A continuación, veremos las bases de datos del sistema.

Figura 27

Modelo de base de datos



Nota: Elaboración propia.

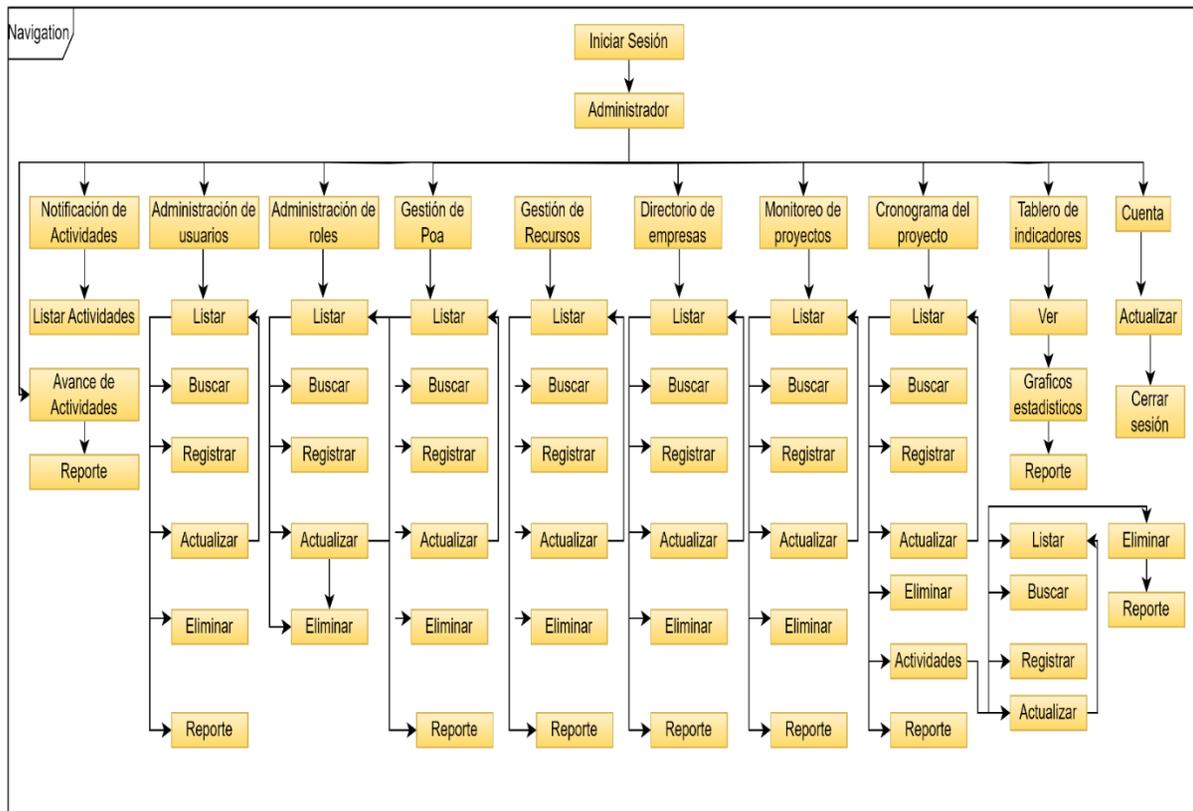
3.8 MODELO DE NAVEGACIÓN

El modelo de navegación se refiere a la forma en que las personas navegan por un sitio web o aplicación. Esencialmente, describe cómo las personas se mueven a través de las diferentes páginas, secciones y funcionalidades de un sitio para lograr sus objetivos. Seguidamente se realiza el modelo de navegación de cada actor en figuras en el cual podremos apreciar con más claridad la acción que cada usuario que ejecuta en el sistema de navegación.

3.8.1 Modelo de Navegación: Administrador

Figura 28

Modelo de navegación – administrador

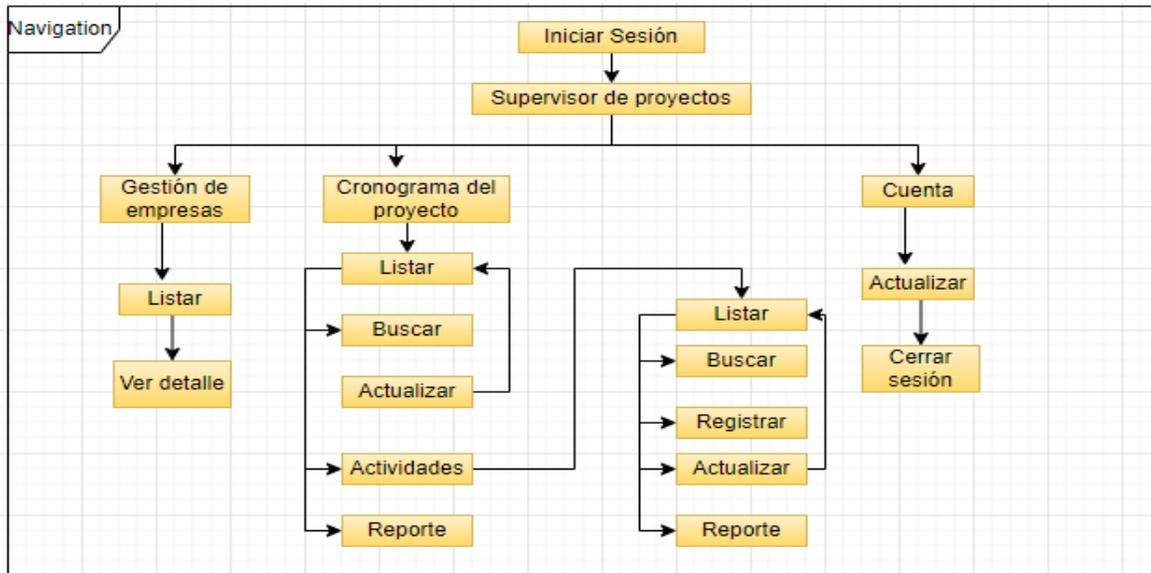


Nota: Elaboración propia.

3.8.2 Modelo de Navegación: Supervisor de Proyectos

Figura 29

Modelo de navegación – supervisor de proyectos

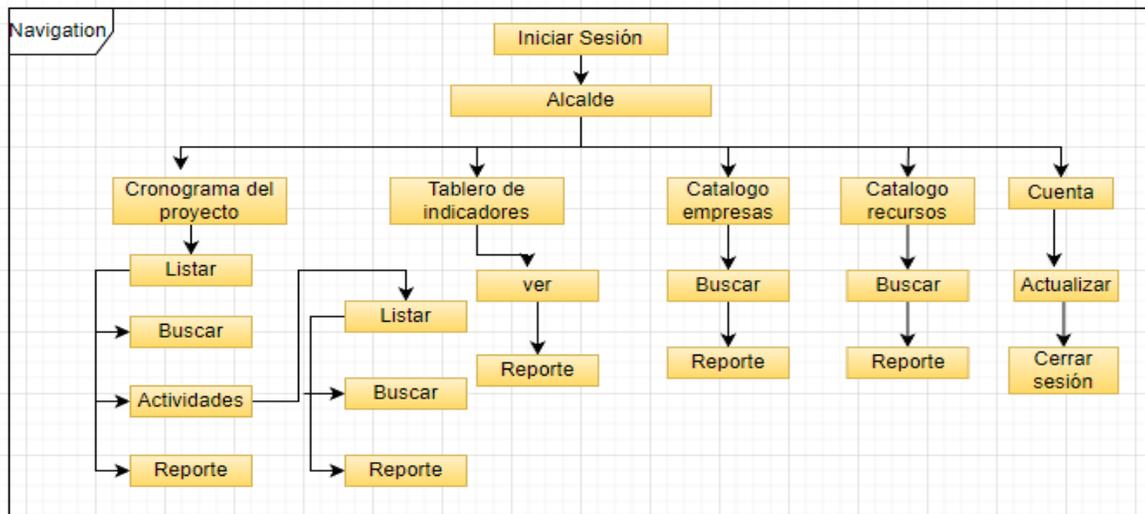


Nota: Elaboración propia.

3.8.3 Modelo de Navegación: Alcalde

Figura 30

Modelo de navegación – alcalde

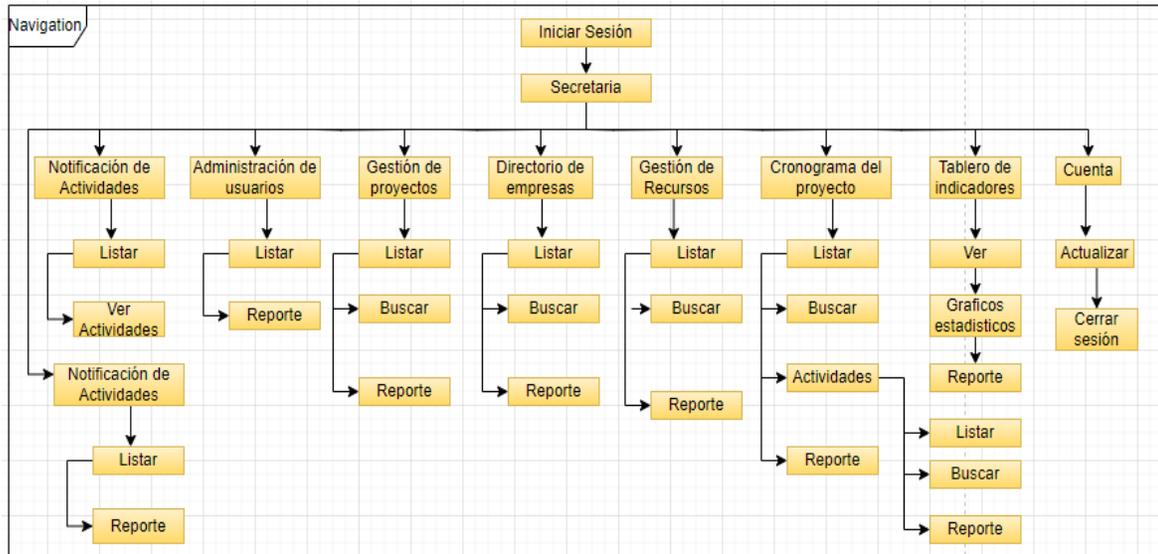


Nota: Elaboración propia.

3.8.4 Modelo de Navegación: Secretaria

Figura 31

Modelo de navegación – secretaria

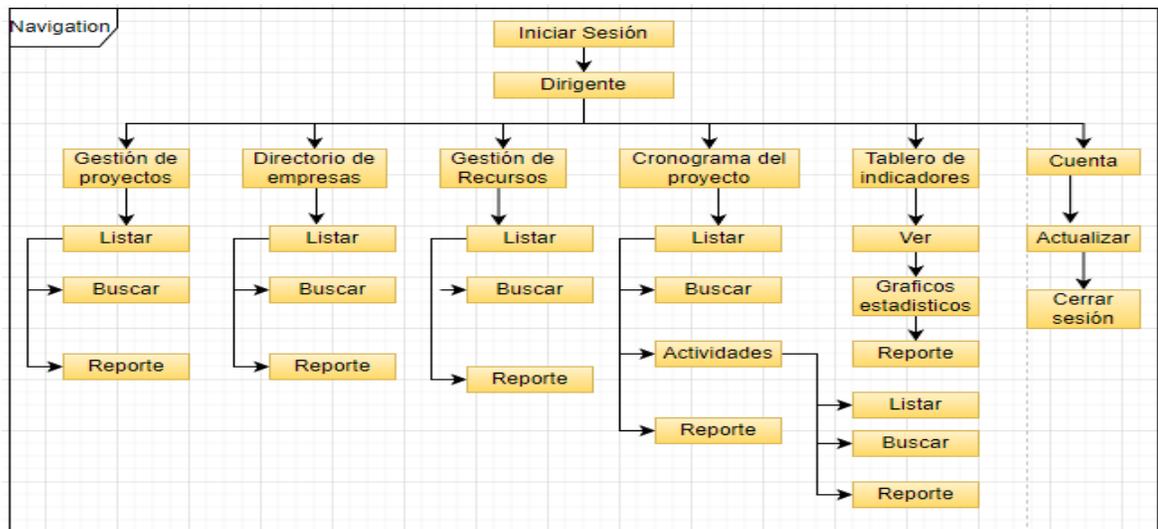


Nota: Elaboración propia.

3.8.5 Modelo de Navegación: Dirigente

Figura 32

Modelo de navegación – dirigente



Nota: Elaboración propia.

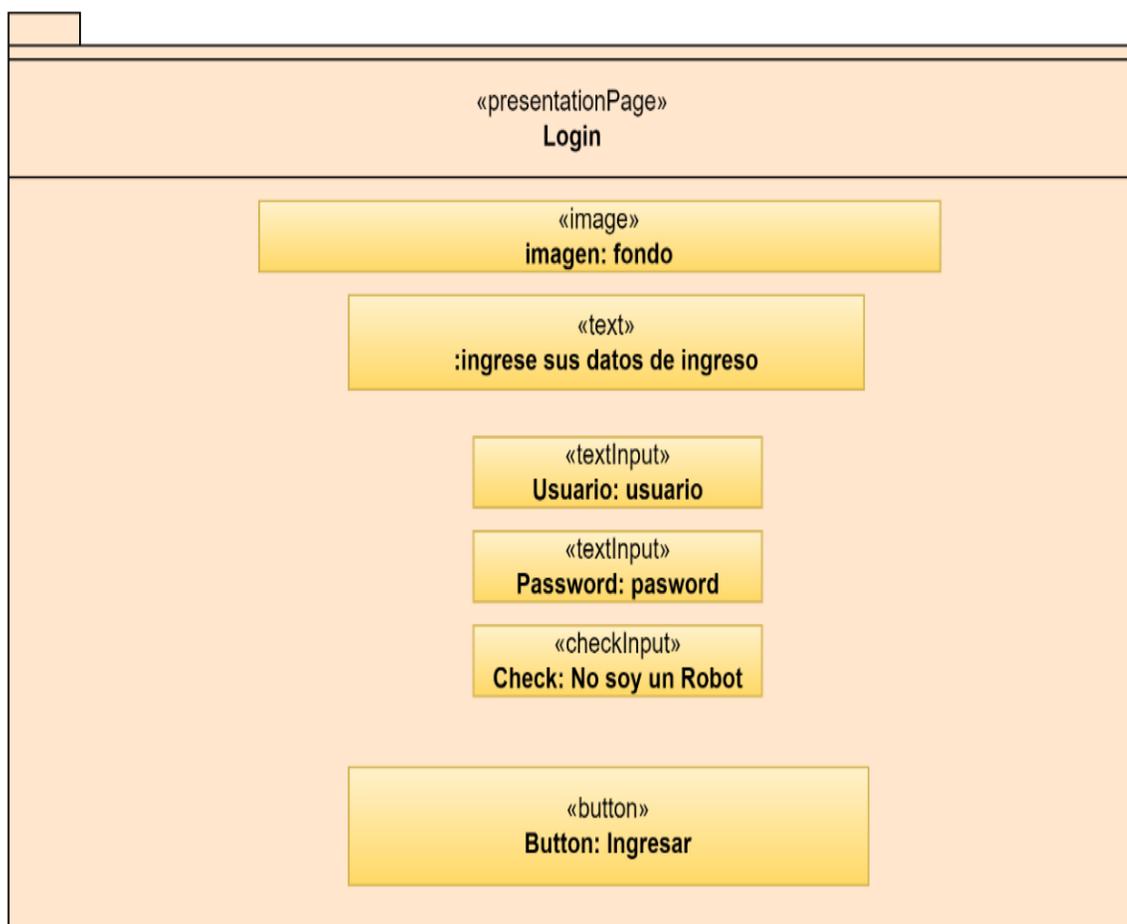
3.9 MODELO DE PRESENTACIÓN

En esta parte se observa los modelos de presentación que incorpora el sistema de cada módulo, se presentan en forma de captura de pantalla.

3.9.1 Modelo de Presentación: LOGIN (Inicio de sesión)

Figura 33

Modelo de presentación: inicio de sesión

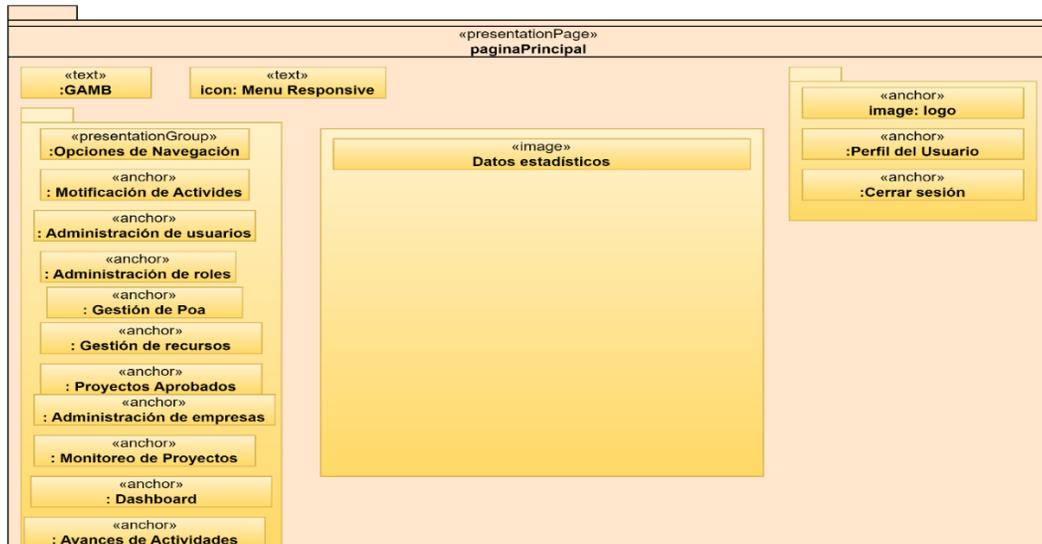


Nota: Elaboración propia.

3.9.2 Modelo de Presentación: Página principal

Figura 34

Modelo de presentación: página principal

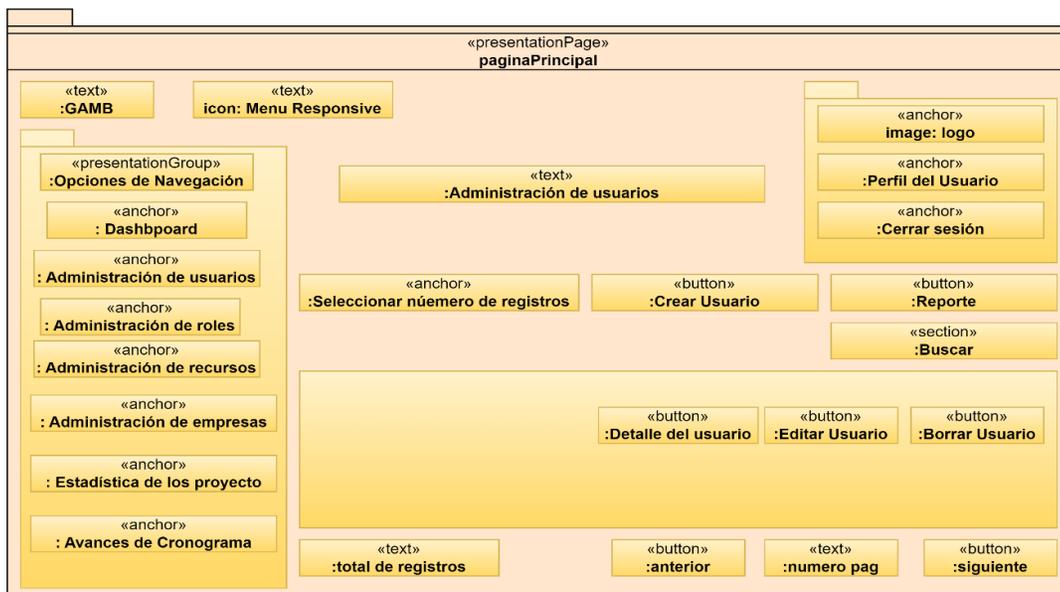


Nota: Elaboración propia.

3.9.3 Modelo de Presentación: Administración de usuarios

Figura 35

Modelo de presentación: administración de usuarios

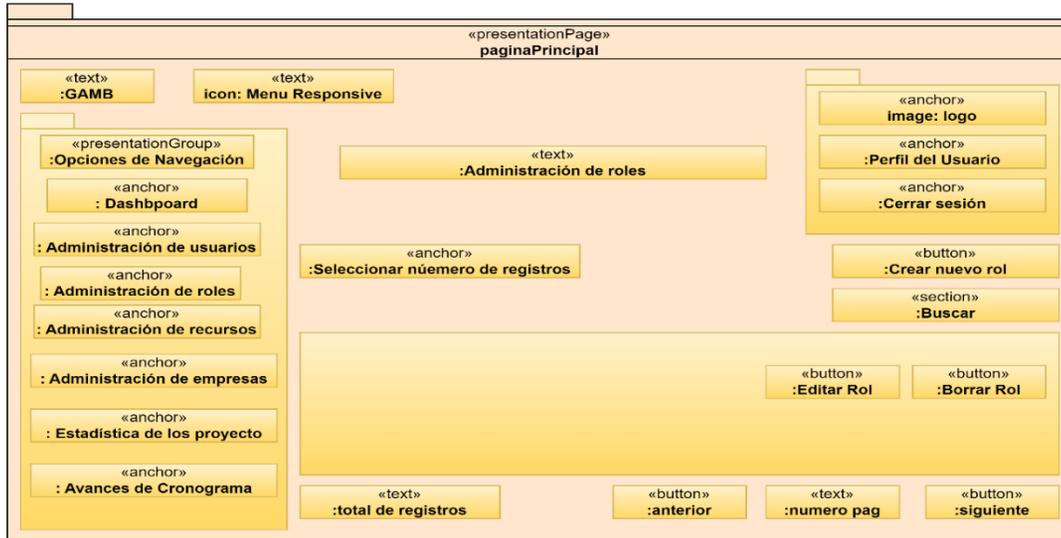


Nota: Elaboración propia.

3.9.4 Modelo de Presentación: Administración de roles

Figura 36

Modelo de presentación: administración de roles

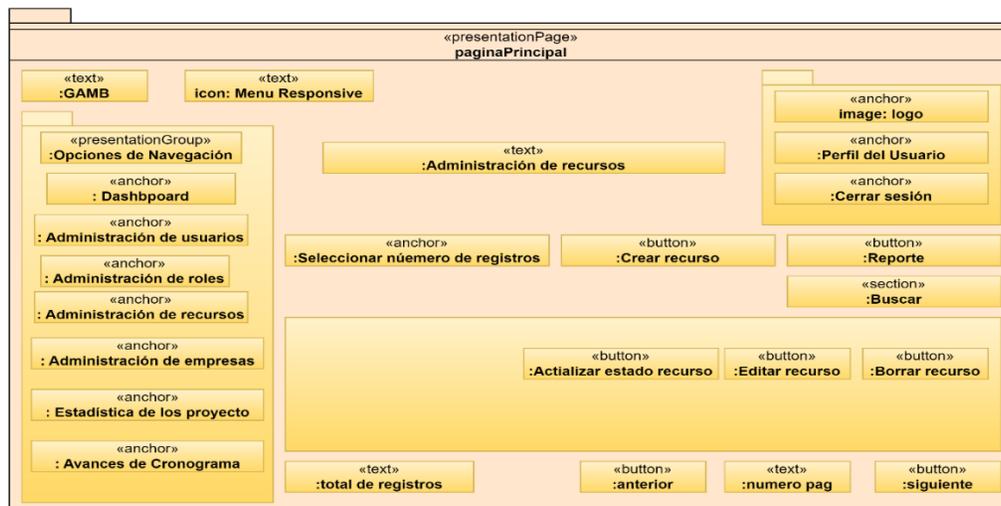


Nota: Elaboración propia.

3.9.5 Modelo de Presentación: Administración de recursos

Figura 37

Modelo de presentación: administración de recursos

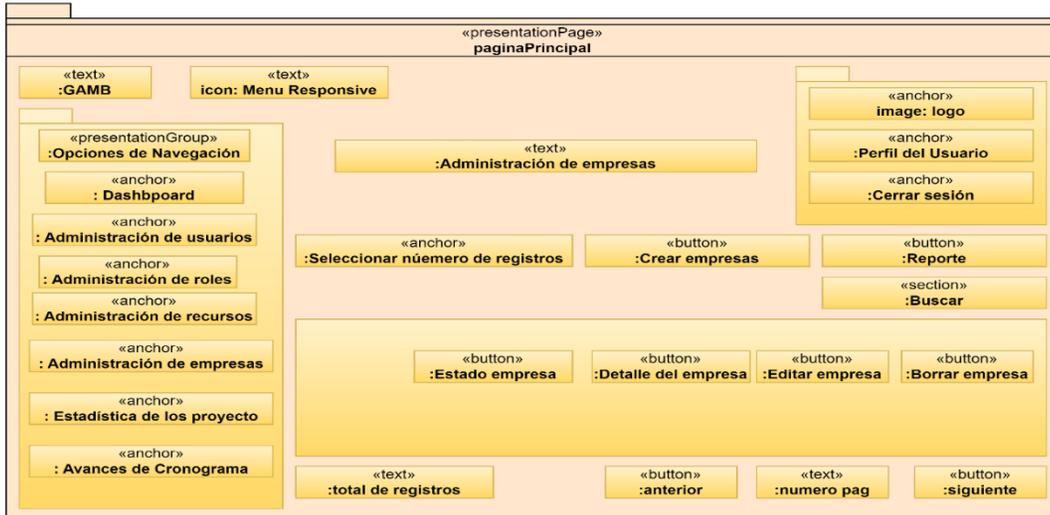


Nota: Elaboración propia.

3.9.6 Modelo de Presentación: Directorio de empresas

Figura 38

Modelo de presentación: directorio de empresas

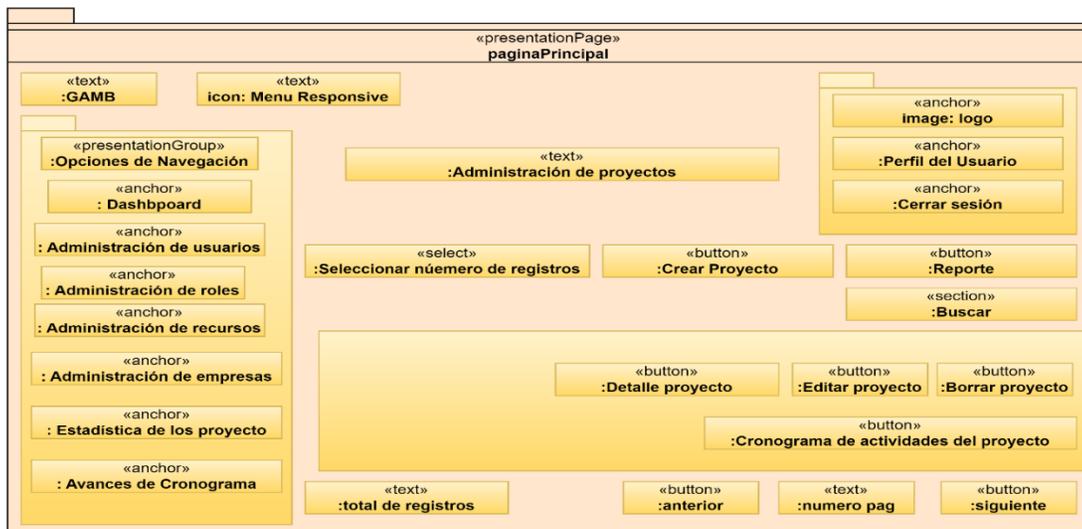


Nota: Elaboración propia.

3.9.7 Modelo de Presentación: Monitoreo de proyectos

Figura 39

Modelo de presentación: monitoreo de proyectos

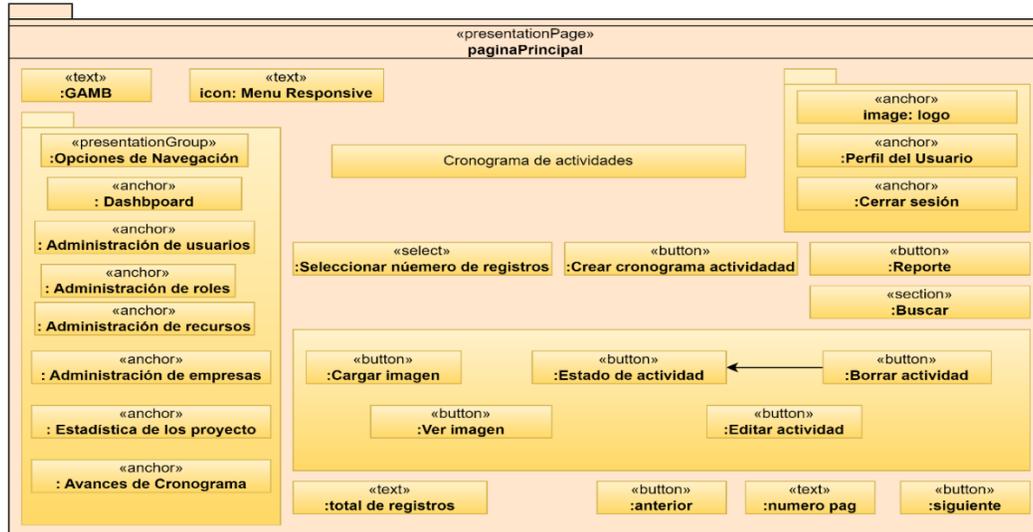


Nota: Elaboración propia.

3.9.8 Modelo de Presentación: Cronograma de actividades

Figura 40

Modelo de presentación: Cronograma de actividades

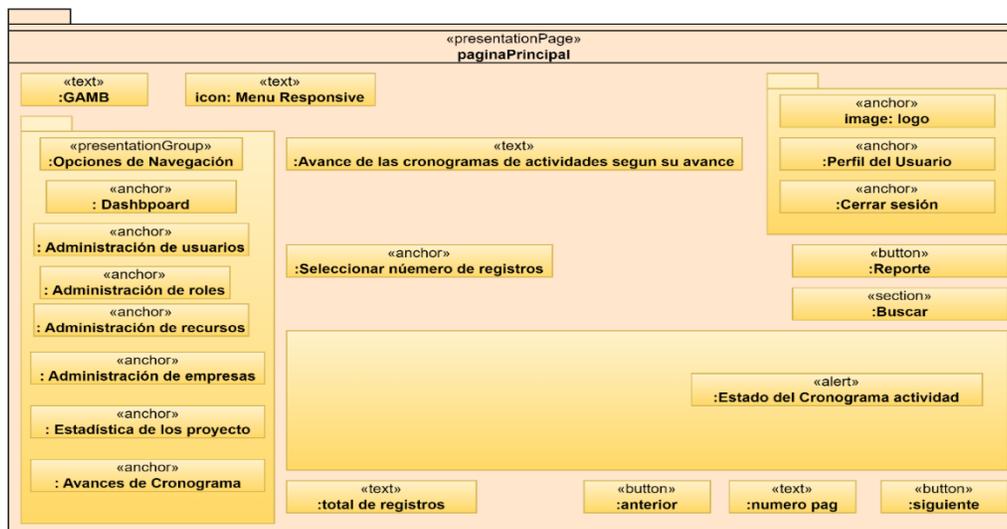


Nota: Elaboración propia.

3.9.9 Modelo de Presentación: Avance de actividades

Figura 41

Modelo de presentación: Avance de actividades

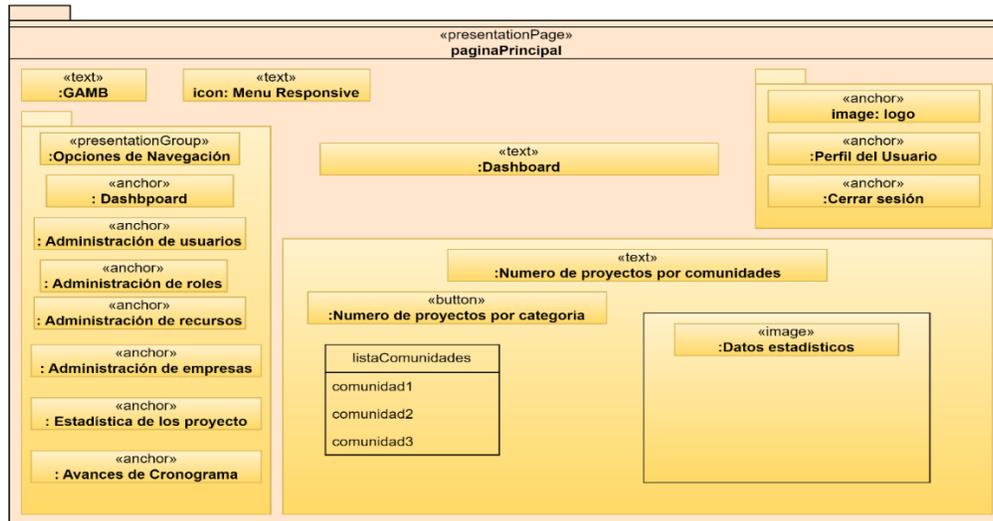


Nota: Elaboración propia.

3.9.10 Modelo de Presentación: Dashboard

Figura 42

Modelo de presentación: dashboard

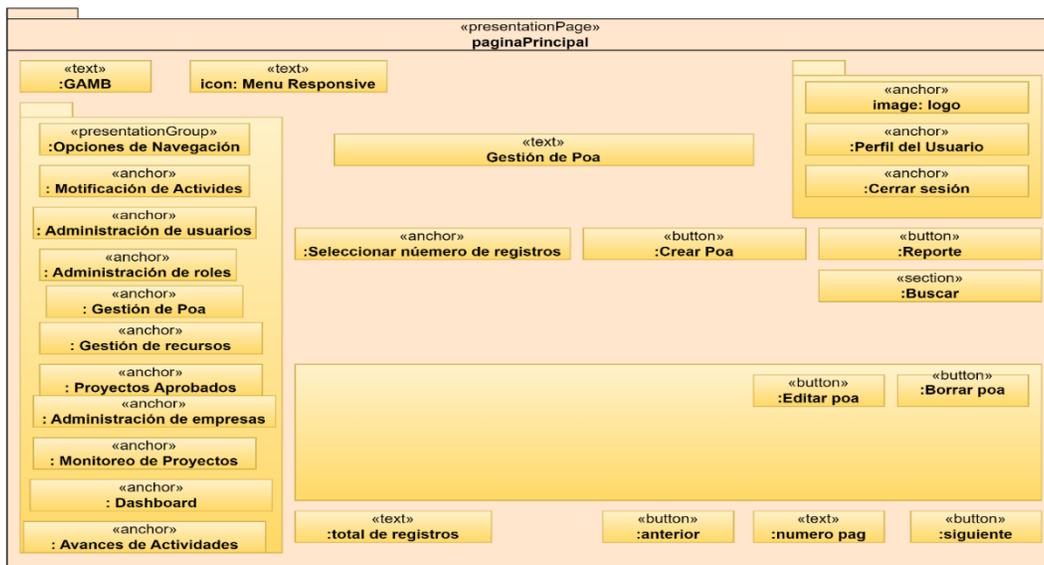


Nota: Elaboración propia.

3.9.11 Modelo de Presentación: Gestión de Poa

Figura 43

Modelo de Presentación: Gestión de Poa

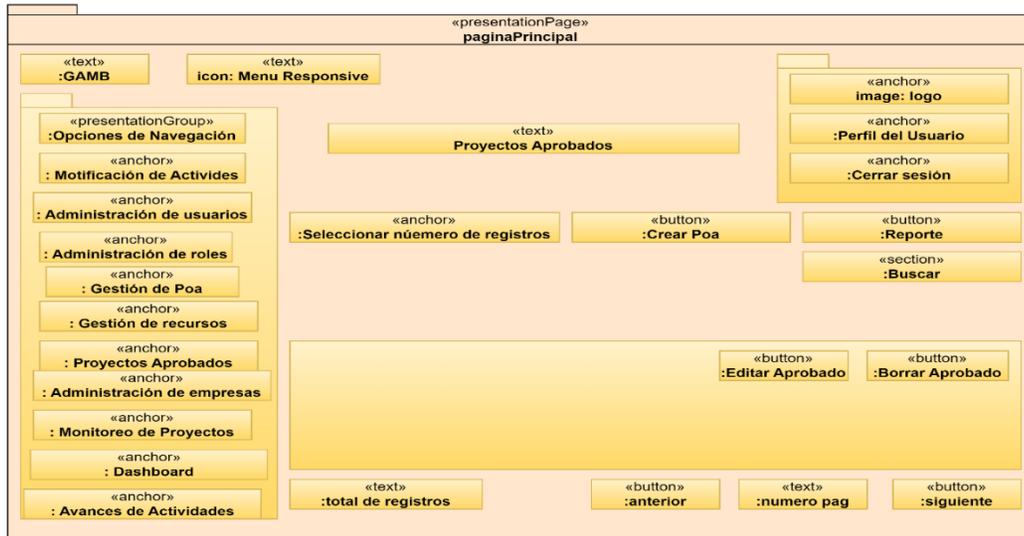


Nota: Elaboración propia.

3.9.12 Modelo de Presentación: Proyectos Aprobados

Figura 44

Modelo de Presentación: Proyectos Aprobados



Nota: Elaboración propia.

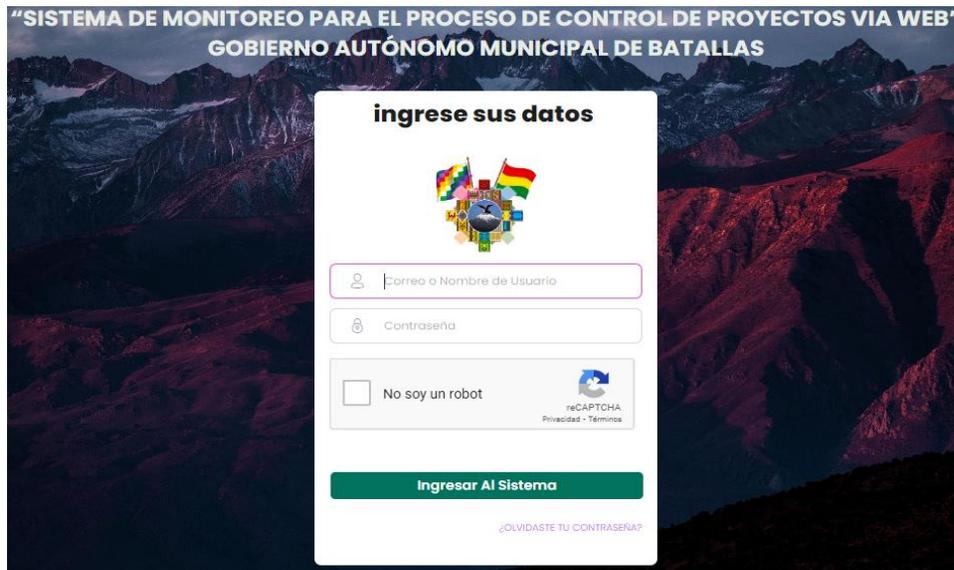
3.10 IMPLEMENTACIÓN DEL SISTEMA

3.10.1 Interfaz de Inicio de Sesión

En la captura de pantalla adjunta se puede observar el formulario de inicio, el cual proporciona acceso al menú de inicio. En la figura podemos apreciar la vista del inicio de sesión.

Figura 45

Interfaz de Inicio de sesión

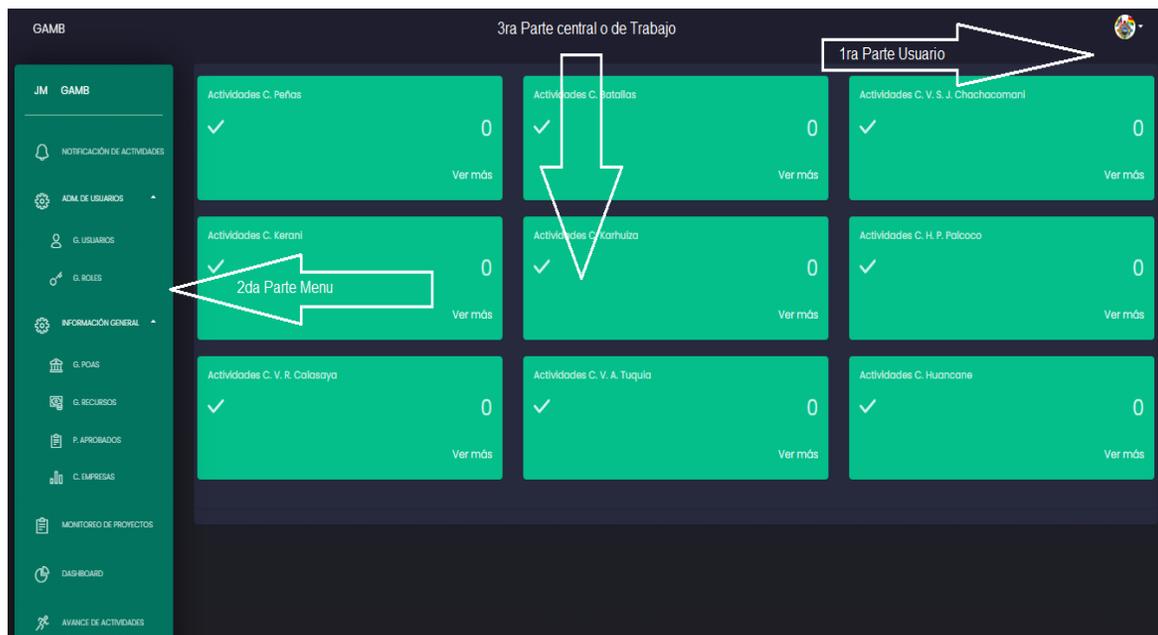


Nota: Elaboración propia

3.10.2 Funcionalidad General

Figura 46

Pantalla de Inicio



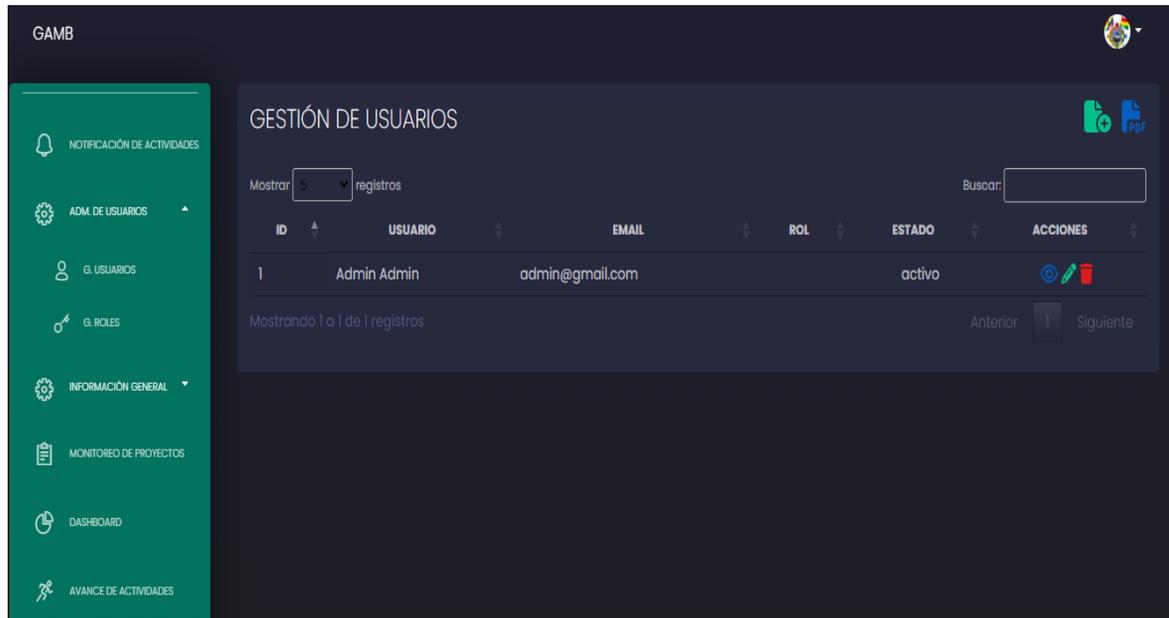
Nota: Elaboración propia.

3.10.3 Módulos que Integran el Sistema

MÓDULO DE USUARIOS

Figura 47

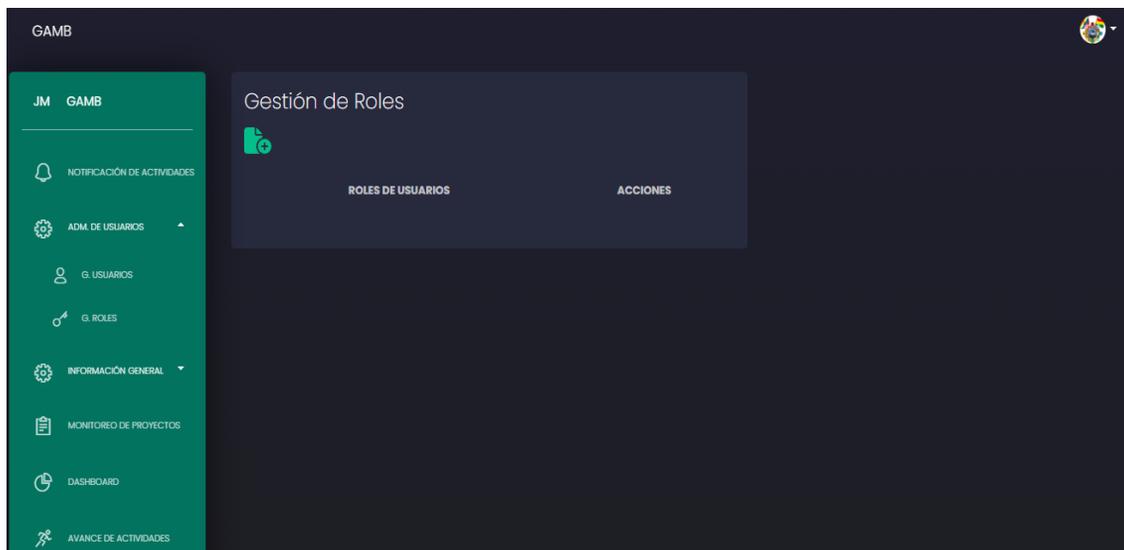
Módulo de Usuarios



MÓDULO DE ROLES

Figura 48

Módulo de Roles



MÓDULO DE RECURSOS

Figura 49

Módulo de Recursos

The screenshot displays the 'Módulo de Recursos' interface. On the left is a green sidebar with navigation options: JM GAMB, NOTIFICACIÓN DE ACTIVIDADES, ADM. DE USUARIOS, INFORMACIÓN GENERAL, G. POAS, G. RECURSOS, P. APROBADOS, C. EMPRESAS, MONITOREO DE PROYECTOS, and DASHBOARD. The main content area is titled 'RECURSOS DESIGNADOS DEL POA 2023 PARA PROYECTOS'. It includes a search bar, date filters for 'Fecha Inicio' and 'Fecha Final' (both set to 'dd/mm/aaaa'), and a 'Mostrar' dropdown set to '5 registros'. Below the search bar is a table with columns: NOMBRE, FECHA DE ASIGNACIÓN, MONTO ASIGNADO BS., ESTADO, OBSERVACIÓN, and ACCIONES. The table is currently empty, displaying the message 'No hay datos disponibles en la tabla' and 'Mostrando 0 a 0 de 0 registros'. Navigation links 'Anterior' and 'Siguiete' are visible at the bottom right of the table area.

MÓDULO DE EMPRESAS

Figura 50

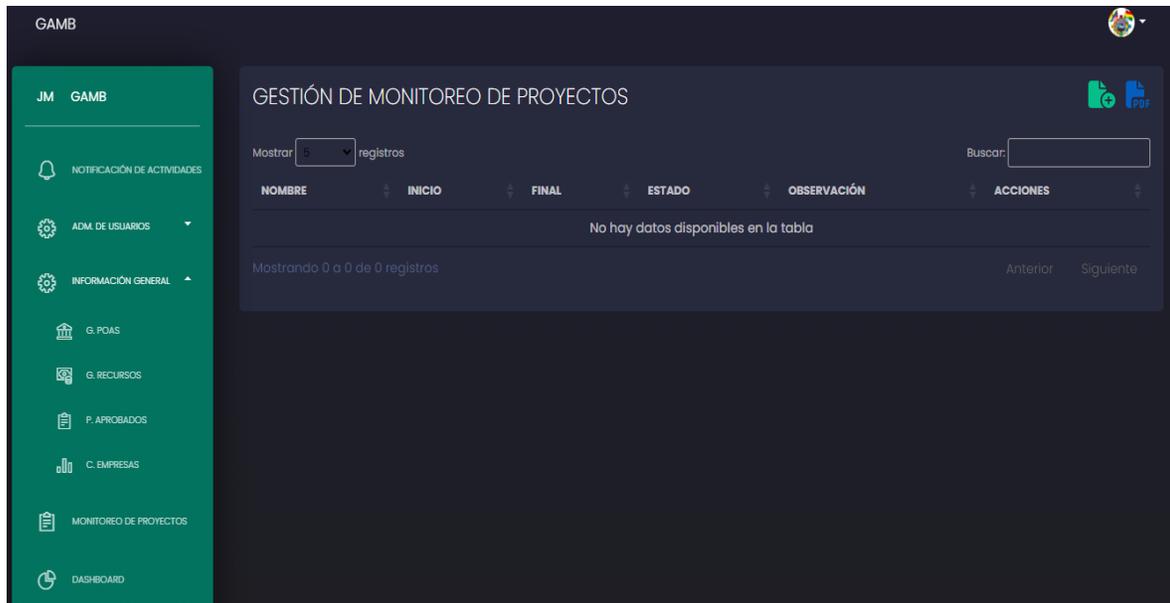
Módulo de Empresas

The screenshot displays the 'Módulo de Empresas' interface. On the left is a green sidebar with navigation options: JM GAMB, NOTIFICACIÓN DE ACTIVIDADES, ADM. DE USUARIOS, INFORMACIÓN GENERAL, G. POAS, G. RECURSOS, P. APROBADOS, C. EMPRESAS, MONITOREO DE PROYECTOS, and DASHBOARD. The main content area is titled 'DIRECTORIO DE EMPRESAS'. It includes a search bar, a 'Mostrar' dropdown set to '5 registros', and a 'Buscar' input field. Below the search bar is a table with columns: NOMBRE, EMAIL, TELEFONO, ESTADO, and ACCIONES. The table is currently empty, displaying the message 'No hay datos disponibles en la tabla' and 'Mostrando 0 a 0 de 0 registros'. Navigation links 'Anterior' and 'Siguiete' are visible at the bottom right of the table area.

MÓDULO DE MONITOREO PROYECTOS

Figura 51

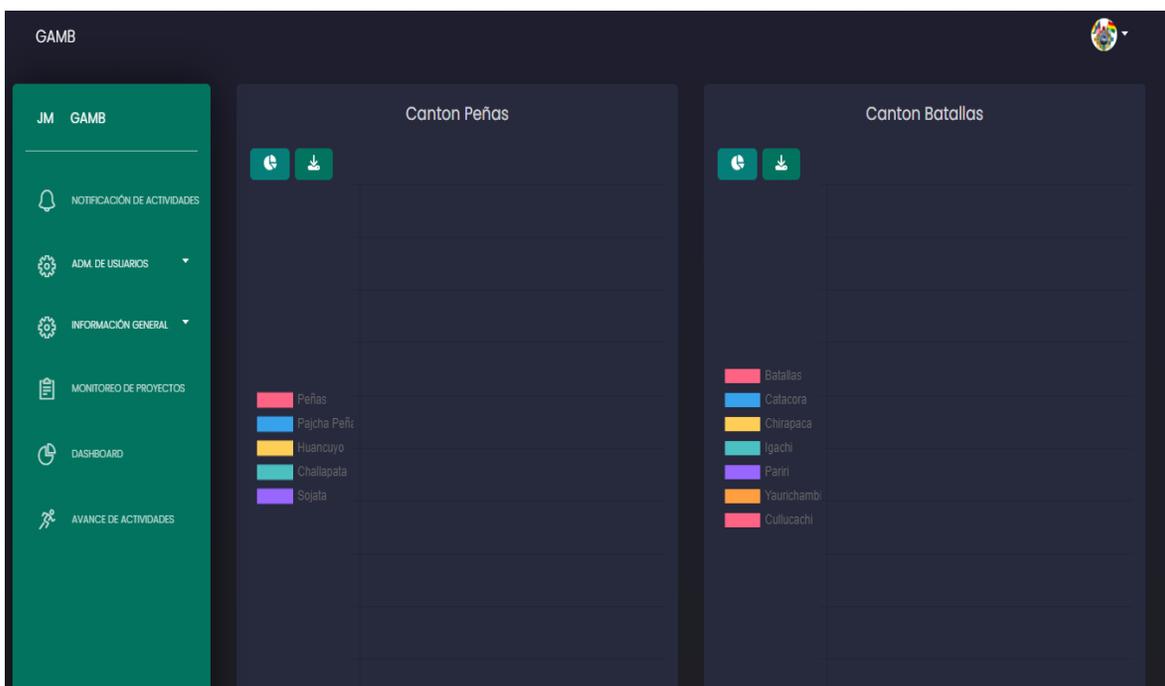
Módulo de Proyectos



MÓDULO DE TABLERO DE INDICADORES

Figura 52

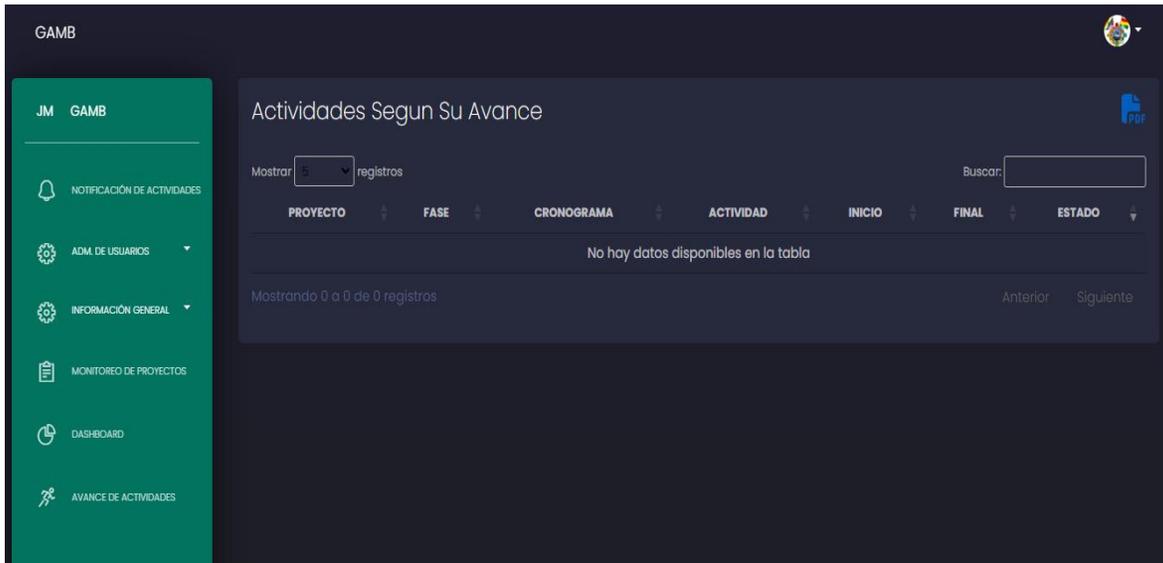
Módulo de Tablero de indicadores



MÓDULO DE AVANCES

Figura 53

Módulo de Avances



REPORTES

Figura 54

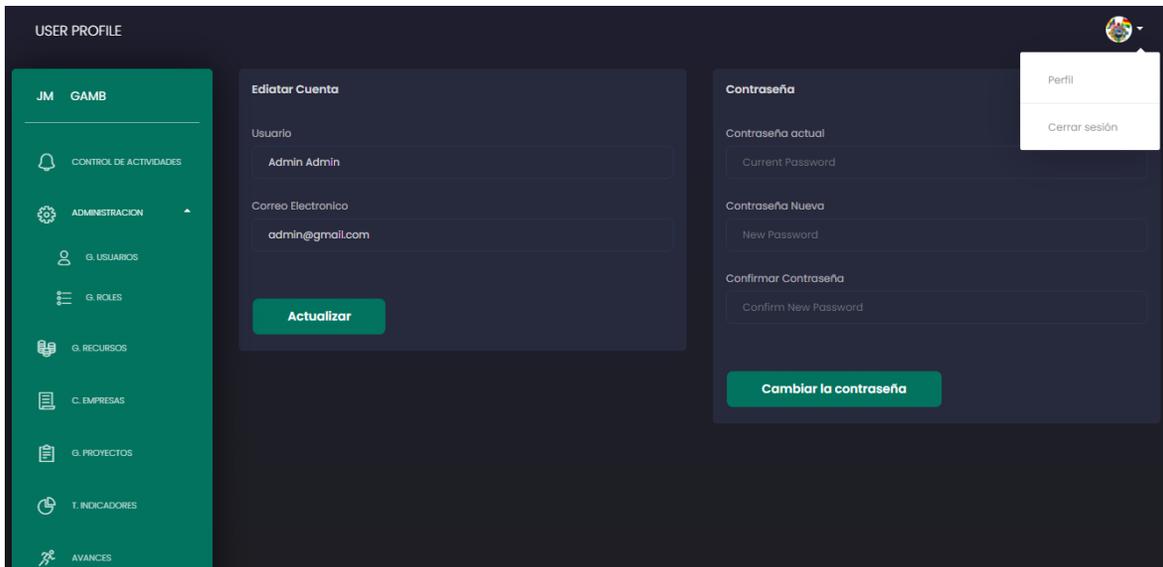
Reportes



MÓDULO DE CUENTA

Figura 55

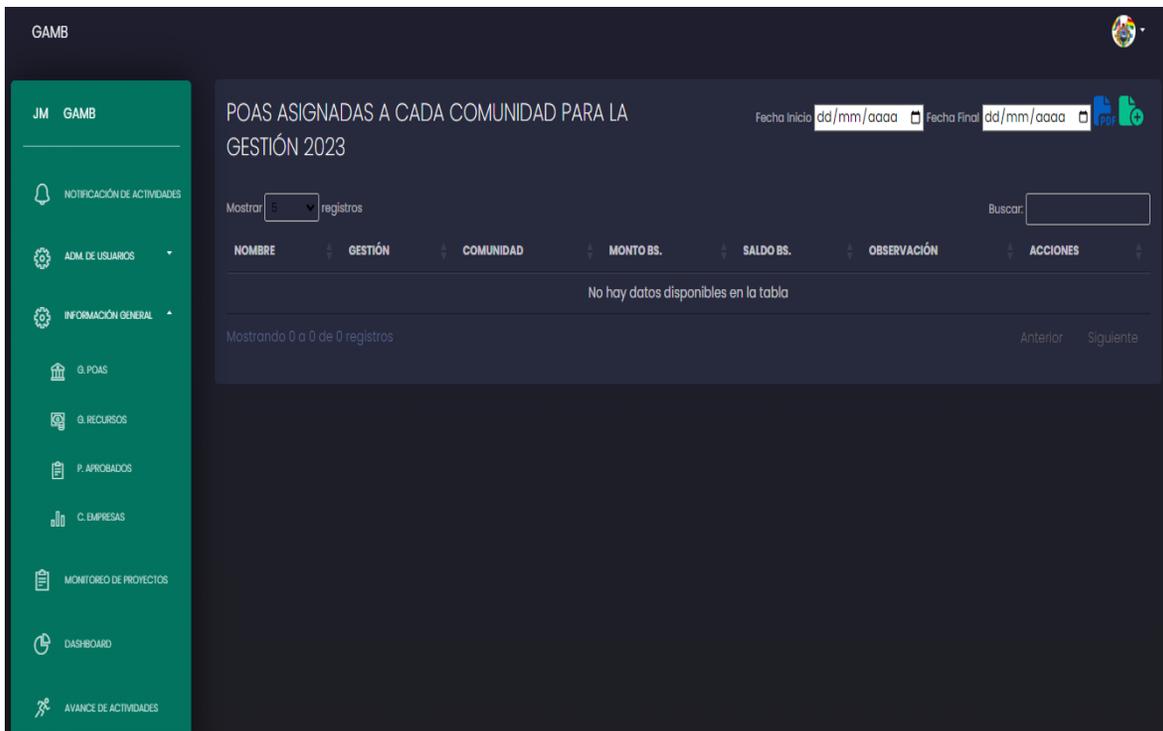
Módulo de Cuenta



MÓDULO DE POAS

Figura 56

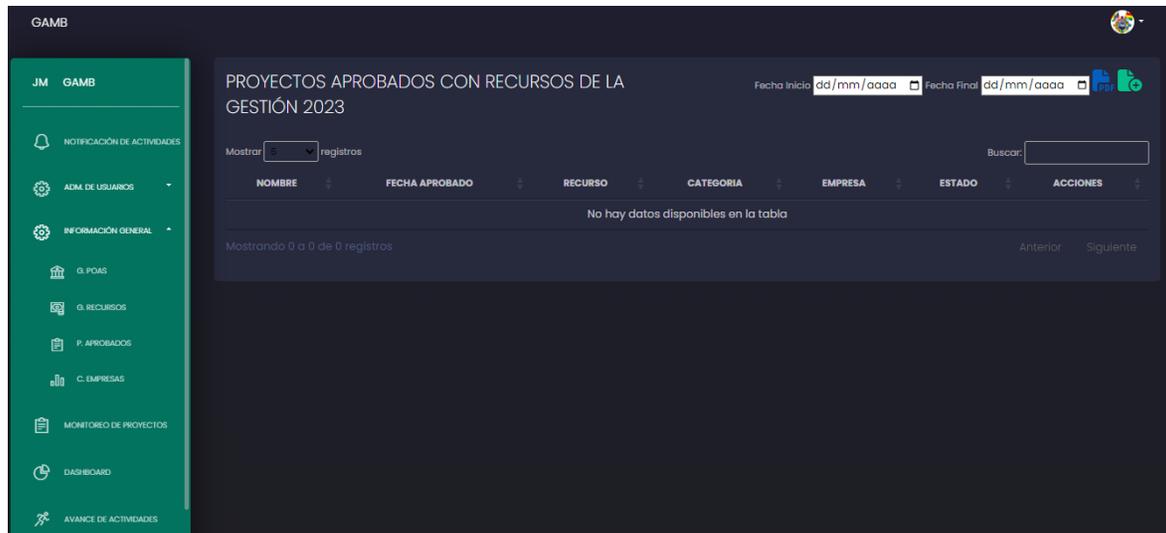
Módulo de Poa



MÓDULO DE ROYECTOS APROBADOS

Figura 57

Módulo de Proyectos Aprobados



3.11 PRUEBAS DE SOFTWARE

3.11.2 Pruebas de Caja Negra

En este tipo de prueba se realizan a la interfaz gráfica del sistema, y no al código fuente, mostrando los posibles errores que se podrían presentar de una forma en la que esta no se vea de forma poco amigable, sin embargo, mantenga el mismo formato, una de las técnicas de las que se hará uso para esta es la de partición de equivalencia. Para la siguiente se realizará la prueba al inicio de sesión al sistema, dando como posibles resultados el ingreso correcto al sistema o la negación al ingreso.

La prueba de caja negra consiste en hacer pruebas de diferentes combinaciones de datos de entrada, para de esta manera verificar el correcto funcionamiento del sistema desarrollado.

Para realizar la prueba de caja negra tomaremos en cuenta los módulos más relevantes del software.

Módulo de autenticación.

Figura 58

Prueba de autenticación – inicio de sesión



Nota: Elaboración propia.

En la figura (58) podemos observar, cuando los datos son ingresados de manera correcta el usuario tendrá un ingreso exitoso al sistema donde podrá visualizar la página de inicio, de lo contrario si no ingresa los datos correctos y no marca el recaptcha le saldrá los errores indicando la falla de la inserción de datos y no podrá ingresar al sistema teniendo un ingreso fallido.

Tabla de rendimiento de módulos.

Para la tabla de rendimientos de módulos, tomaremos los módulos más relevantes del sistema.

Tabla 20

Tabla de rendimientos de módulos

| Módulos | Tiempo usado (sistema propuesto) |
|----------------------------|-------------------------------------|
| Administración de usuarios | 3,6 minutos |
| Gestión de rol | 1.2 minutos |
| Gestión de POA | 2,1 minutos |
| Gestión de recursos | 1,8 minutos |
| Catálogo de empresas | 3,5 minutos |
| Proyectos aprobados | 10 minutos |

Nota: Elaboración propia

Tabla de acceso de usuarios.

Tabla 21

Prueba de caja negra Inicio de sesión

| Entradas | | Salidas | Resultados |
|----------|------------|---------------------------------|--|
| Usuario | Contraseña | | |
| | | Ingrese el usuario y contraseña | El sistema valida que no se ingresen campos en Blanco. |
| Admin11 | **** | Ingrese datos correctos | El sistema verifica las credenciales |
| Admin | ***** | Página de inicio de sesión. | Datos correctos, ingresa al sistema. |

Nota: Elaboración propia.

Validaciones de Inicio de Sesión

Figura 59

Validación de campo en blanco.



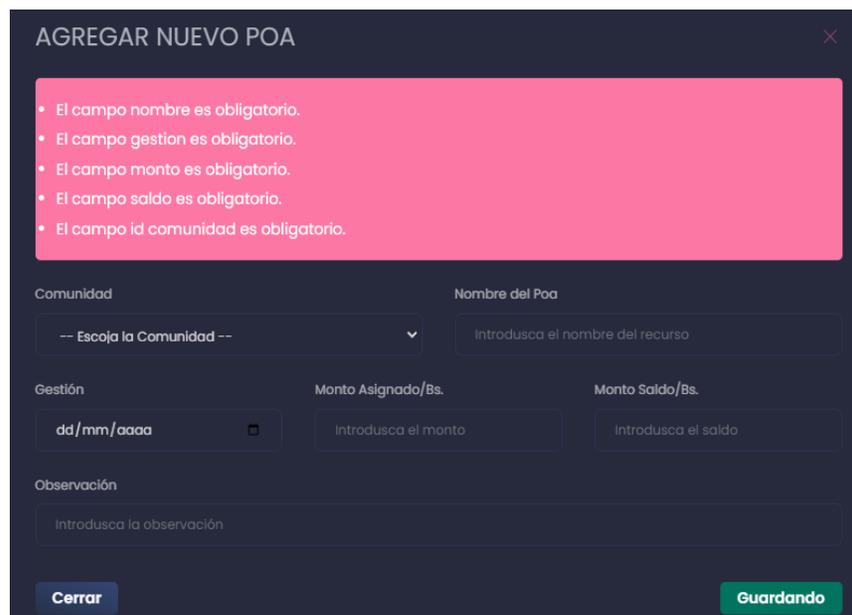
The image shows a login form with the title "ingrese sus datos". It features a header with flags and a globe. The form includes a text input for "Correo o Nombre de Usuario", a password input field with a "Completa este campo" error message, a "No soy un robot" checkbox, and a reCAPTCHA logo. A green "Ingresar Al Sistema" button is at the bottom, along with a link for "¿OLVIDASTE TU CONTRASEÑA?".

Nota: Elaboración propia.

Formulario de un formulario

Tabla 22

Validación de un formulario.



The image shows a form titled "AGREGAR NUEVO POA" with a validation error message. The error message is: "El campo nombre es obligatorio. El campo gestion es obligatorio. El campo monto es obligatorio. El campo saldo es obligatorio. El campo id comunidad es obligatorio." The form includes fields for "Comunidad" (a dropdown menu), "Nombre del Poa" (a text input), "Gestión" (a date input), "Monto Asignado/Bs." (a text input), "Monto Saldo/Bs." (a text input), and "Observación" (a text input). There are "Cerrar" and "Guardando" buttons at the bottom.

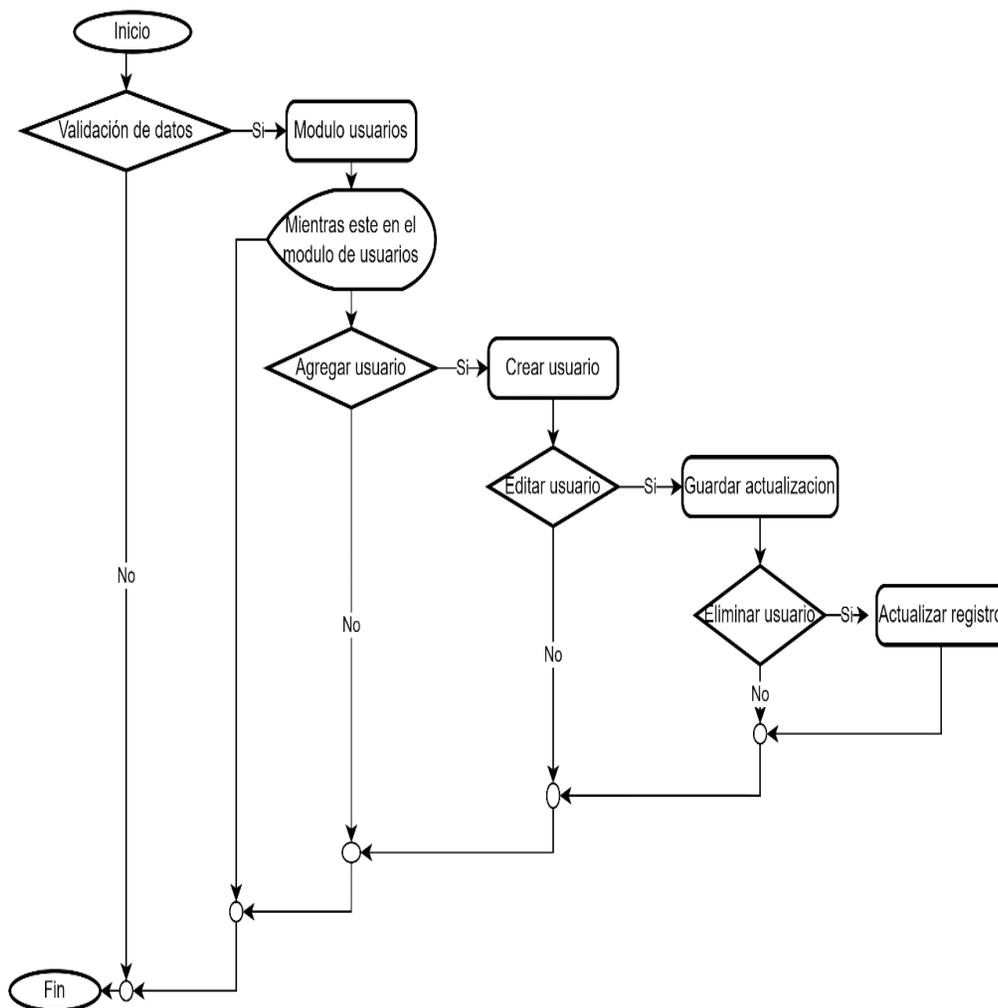
Nota: Elaboración propia.

3.11.1 Pruebas de Caja Blanca

La prueba de caja blanca, es para garantizar las funcionalidades internas de cada módulo. Tomaremos un solo modulo para la aplicar la mencionada prueba, en cual se describe de forma general la funcionalidad de los módulos que conforman el sistema.

Figura 60

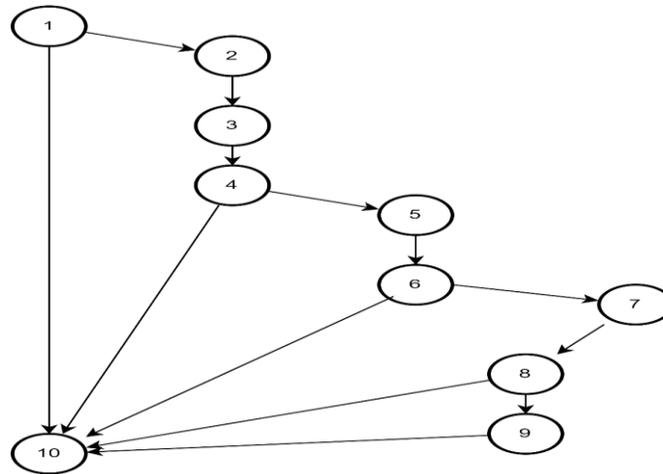
Diagrama de flujo: Registro de usuarios



Nota: Elaboración propia

Figura 61

Grafo de métrica: Registro de usuarios



Nota: Elaboración propia.

Descripción de cada nodo:

Donde:

1. Validación de datos.
2. Módulo de administración de usuarios.
3. Mientras este en el módulo administración de usuarios.
4. Agregar nuevo usuario
5. Crear usuario.
6. Editar usuario.
7. Guardar cambios.
8. Eliminar usuario.
9. Actualizar registro.

Como podemos observar en la figura (47), tenemos los nodos y aristas: en donde el número de nodos es igual a 10 y el número de aristas es igual a 13.

Para determinar la complejidad de ciclo matica, aremos uso de la siguiente ecuación:

$$V(G) = A - N + 2$$

En donde $A = 13$ y $N = 10$, reemplazamos datos en la ecuación:

$$V(G) = 13 - 10 + 2$$

$$\underline{V(G) = 5}$$

Camino 1: 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10

Camino 2: 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 10

Camino 3: 1 – 2 – 3 – 4 – 5 – 6 – 10

Camino 4: 1 – 2 – 3 – 4 – 10

Camino 5: 1 – 10

CAPÍTULO IV
MÉTRICAS DE CALIDAD,
SEGURIDAD Y
ESTIMACIÓN DE COSTOS

4.1 MÉTRICAS DE CALIDAD

La evaluación de la calidad del software se llevará a cabo utilizando la métrica ISO 9126, la cual establece que cualquier aspecto de la calidad del software puede describirse en términos de seis características fundamentales: funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad.

4.1.1 Norma ISO/IEC 9126

El propósito fundamental de esta norma es lograr el nivel de calidad requerido para satisfacer las expectativas del cliente. Se lleva a cabo una evaluación en dos áreas: el producto final entregado y los procesos utilizados durante su desarrollo.

4.1.1.1. Funcionalidad

Punto función es una medida que se enfoca en la funcionalidad y procesos de desarrollo del software. Se determina mediante una serie de actividades que incluyen calcular los siguientes números:

- **Números de entrada de usuario:** Se refieren al conteo a la cantidad de entradas por el usuario que se inician sesión en la aplicación.
- **Número de salida del usuario:** Hace referencia a la cantidad de salidas generadas por el software, como informes, pantallas, mensajes de error, y otros tipos de resultados que son mostrados al usuario.
- **Números de peticiones al usuario:** En este caso, se cuentan las peticiones realizadas al usuario como parte de la interacción del software, las cuales son generadas en respuesta a alguna entrada del usuario y que requieren su intervención para seguir el flujo de ejecución.

- **Número de archivos:** Refiere a la cantidad de archivos maestros lógicos presentes en el sistema, es decir, aquellos archivos que contienen información estructurada y son utilizados para el procesamiento de datos.
- **Número de interfaces externas:** Aquí se cuentan todas las interfaces legibles por la máquina, como archivos de datos en cinta o discos, que son utilizados para transmitir información a otros sistemas externos.

Según lo mencionado tenemos los siguientes resultados:

Tabla 23

Punto de función: entrada de datos

| | |
|----------------------|----|
| Entradas de usuario | 30 |
| Salidas de usuario | 20 |
| Consultas de usuario | 25 |
| Número de archivos | 23 |
| Interfaces externas | 7 |

Nota: se describe el número de entrada de datos.

A continuación, rellenaremos los puntos de función en tabla (20), para realizar el calcularlo con los datos obtenidos en la siguiente tabla (19), tomando en cuenta un factor de ponderación medio.

Tabla 24

Punto de función entradas para el cálculo de funcionalidad

| Parámetros de medición | Cuenta | Factor de ponderación medio | Total |
|-------------------------------|--------|-----------------------------|-------|
| Número de entradas de usuario | 27 | 6 | 162 |
| Número de salidas del usuario | 17 | 7 | 119 |

| | | | |
|---------------------------------|----|----|-----|
| Número de consultas del usuario | 22 | 6 | 132 |
| Número de archivos | 19 | 12 | 228 |
| Número de interfaces | 4 | 9 | 36 |
| Cuenta Total | | | 677 |

Nota: Son datos sobre número de peticiones al sistema. Elaboración propia.

La fórmula utilizada para calcular los puntos de función es la siguiente:

$$PF = Cuenta\ Total * (Grado\ de\ confiabilidad + tasa\ de\ error * \sum = Fi)$$

A continuación, en los siguientes puntos se explicará más detalladamente cada variable de la ecuación señalada:

- **PF** = Medida de Funcionalidad.
- **Cuenta Total** = Refiere a la suma de valor de las entradas, salidas, peticiones, interfaces externas y archivos.
- **Grado de confiabilidad** = hace referencia a la confiabilidad estimada del sistema.
- **tasa de error** = Es la estimación subjetiva de la probabilidad de error en el dominio de la información es del 1%.
- **Fi** = Los valores de ajuste de complejidad son números que se obtienen de la tabla 20 y que proporcionan respuestas a las preguntas de la siguiente tabla 21.

Tabla 25

Ajustes de complejidad del punto función

| | | | | | | | |
|-------------|--------|---------|---------|---------|---------|----------|--|
| Importancia | 0 % | 20 % | 40 % | 60 % | 80 % | 100 % | |
|-------------|--------|---------|---------|---------|---------|----------|--|

| Escala | Sin importancia | Incidental | Moderado | Medio | Significativo | Esencial | Fi |
|--|-----------------|------------|----------|-------|---------------|----------|----|
| 1. ¿Requiere el sistema copias de seguridad y recuperación fiable? | | | | | | X | 5 |
| 2. ¿Se requiere comunicación de datos? | | | X | | | | 2 |
| 3. ¿Existe funciones de procesamiento distribuido? | | | | | | X | 5 |
| 4. ¿Es crítico el rendimiento? | | | | X | | | 3 |
| 5. ¿Será ejecutado el sistema en un entorno operativo existente y frecuentemente utilizado? | | | | | | X | 5 |
| 6. ¿Requiere el sistema entrada de datos interactivos? | | | | | X | | 4 |
| 7. ¿Requiere la entrada de datos interactivos que las transiciones de entrada se lleven a cabo sobre múltiples o variadas operaciones? | | | X | | | | 2 |
| 8. ¿Se actualiza los archivos de forma interactiva? | | | | X | | | 3 |
| 9. ¿Son complejas las entradas, las salidas, los archivos o peticiones? | | | X | | | | 2 |
| 10. ¿Es complejo el procesamiento interno? | | | X | | | | 2 |
| 11. ¿Se ha diseñado el código para ser reutilizable? | | | | | | X | 5 |

| | | | | | | | | |
|---|--|------------------|---|--|--|---|---|--|
| 12. ¿Están incluidos en el diseño la conversión y la instalación? | | | | | | X | 5 | |
| 13. ¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes plataformas? | | | X | | | | 2 | |
| 14. ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizados por el usuario? | | | | | | X | 5 | |
| Total | | $\sum (Fi) = 50$ | | | | | | |

Nota: En la tabla se puede observar el grado de importancia las encuestas. Elaboración propia.

Ya teniendo todos los datos reemplazamos los valores y calculamos el valor de la variable PF en la siguiente formula:

$$PF = Cuenta\ Total * (Grado\ de\ confiabilidad + tasa\ de\ error * \sum = Fi)$$

$$Cuenta\ Total = 677$$

$$Grado\ de\ confiabilidad = 0.65$$

$$Tasa\ de\ error = 0.01$$

$$\sum (Fi) = 50$$

Reemplazando datos tenemos el siguiente resultado:

$$PF = 677 * (0.65 + 0.01 * 50)$$

$$PF = 778.55$$

Luego si consideramos el máximo valor de ajuste de complejidad como $\sum(Fi) = 65$ se obtiene el siguiente resultado:

$$PF = 677 * (0.65 + 0.01 * 65)$$

$$PF = 880.1$$

Por tanto, si $\sum(F_i)$ es considerada como 100%, entonces la relación obtenida entre los puntos no dará el siguiente resultado:

$$Funcionalidad = \frac{778.55}{880.1} * 100$$
$$\underline{Funcionalidad = 88.46\%}$$

Entonces, se puede concluir que la funcionalidad o capacidad del sistema es del 88%, lo que indica que el software opera de manera satisfactoria. Una alta funcionalidad indica que el software cumple con las expectativas y necesidades de los usuarios en términos de su propósito y características, lo que puede contribuir a una experiencia de usuario positiva y a una mayor eficiencia en la realización de tareas.

4.1.1.2. Confiabilidad

La confiabilidad, refiere a la capacidad del software para mantener un nivel consistente de rendimiento o funcionamiento dentro de un período determinado, y bajo las condiciones definidas para su uso. En otras palabras, un software confiable es aquel que es capaz de operar sin interrupciones, errores o fallos significativos, y que es capaz de mantener su nivel de calidad a lo largo del tiempo y bajo distintas circunstancias. La confiabilidad es un factor crítico para la satisfacción del usuario y para el éxito a largo plazo de cualquier proyecto de software.

A continuación, mediremos la confiabilidad del software con la siguiente ecuación:

$$R(t) = (Funcionalidad) * e^{-\lambda t}$$

Seguidamente explicaremos las variables de la ecuación:

$$R(t) = Confiabilidad del sistema$$
$$Funcionalidad = 88\%$$

$$\lambda = 0.01 \text{ (es el error 1 en cada 6 ejecuciones)}$$

$$t = 12 \text{ meses}$$

Ya teniendo todos los datos reemplazamos las variables en la ecuación:

$$R(t) = (\text{Funcionalidad}) * e^{-\lambda t}$$

$$R(12) = (88\%) * e^{-\frac{1}{6} * 12}$$

$$R(12) = 0.12 \Rightarrow 0.12 * 100 = 12\%$$

Luego reemplazamos en la siguiente ecuación de no hallar una falla obtenemos el siguiente resultado:

$$P^*(T > t) = 1 - R^*(t)$$

$$P^*(T > t) = 1 - 0.119 \quad P^*(T > t) = 0.881$$

$$P^*(T > t) = 0.881 * 100$$

$$P^*(T > t) = 88.1 \%$$

$$\underline{\text{Confiabilidad} = 88.1\%}$$

Por consiguiente, la confiabilidad del sistema es del 88.1%, lo que implica que, durante un periodo de 12 meses, el software mantendrá un nivel de rendimiento adecuado. Esto indica que se espera que el sistema opere de manera confiable y constante durante un período prolongado, lo que puede contribuir a una experiencia de usuario positiva y a una mayor eficiencia en la realización de tareas.

4.1.1.3. Mantenibilidad

Para determinar el índice de madurez del software (IMS), que mide la mantenibilidad del sistema, se tendrán en cuenta los cambios que se han producido en cada versión del producto. Para este propósito, se utilizará la siguiente ecuación para calcular el IMS:

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

En donde a continuación detallaremos cada uno de las variables de la ecuación:

- **Mt** = Número de módulos en la versión actual.
- **Fa** = Número de módulos en la versión actual que se ha cambiado.
- **Fc** = Número de módulos en la versión actual que se han añadido.
- **Fd** = Número de módulos en la versión actual que se han eliminado.

La tabla siguiente tabla (24) proporciona los valores necesarios para calcular el IMS del sistema en cuestión. Los valores obtenidos en esta tabla son fundamentales para medir la mantenibilidad del software y para evaluar la calidad de las diferentes versiones del producto. Es importante tener en cuenta estos valores para poder realizar una evaluación precisa del software y determinar las posibles mejoras que se pueden aplicar en el futuro.

Tabla 26

Información requerida para el IMS

| Información | Valores obtenidos |
|-------------|-------------------|
| Mt | 6 |
| Fc | 1 |
| Fa | 0 |
| Fd | 0 |

Nota: Obtenidos en la tabla son para aplicar mejoras en el sistema en el futuro. Elaboración propia.

Seguidamente calculamos el índice de madurez del software sustituyendo los valores en la la ecuación:

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

$$IMS = \frac{[6 - (1 + 0 + 0)]}{6}$$

$$IMS = 0.83$$

Para entender mejor la interpretación del resultado multiplicamos por 100, obteniendo el resultado en porcentajes:

$$\text{Mantenibilidad} = 0.83 * 100$$

$$\text{Mantenibilidad} = 83\%$$

Según los datos obtenidos, se puede deducir que la mantenibilidad del sistema es del 83%, lo que indica que se encuentra en un rango satisfactorio. Esta medida es una indicación del nivel de facilidad con el que se pueden realizar las tareas de mantenimiento y corrección en el software. Una Mantenibilidad alta es fundamental para garantizar una experiencia de usuario sin interrupciones y para evitar posibles errores o fallos en el sistema. Por lo tanto, contar con una Mantenibilidad satisfactoria es un aspecto importante a considerar para asegurar la calidad del software y la satisfacción del usuario.

4.1.1.4. Portabilidad

La portabilidad de un software hace referencia al esfuerzo que se requiere para transferir el programa de un entorno, ya sea de hardware o software, a otro entorno. Esta es una característica muy valorada en todo software, ya que permite su uso en diferentes sistemas y entornos sin necesidad de hacer grandes cambios o modificaciones.

Para determinar la portabilidad lo hallaremos con la siguiente ecuación:

$$P = [1 - \left(\frac{EP}{EI}\right)]$$

A continuación, explicaremos todas las variables de la ecuación mencionada:

- $P = Prtabilidad$
- $EP = Esfuerzo para portar$
- $EI = Esfuerzo para implemetar$

Digamos que el esfuerzo para portar e implementar la aplicación es de 10 y 50, respectivamente, dentro de un parámetro de 1 al 100.

Por tanto, reemplazando los datos en la ecuación obtenemos los siguiente:

$$P = [1 - \left(\frac{EP}{EI}\right)]$$

$$P = [1 - \left(\frac{10}{50}\right)]$$

$$P = 0.8$$

Para un buen entendimiento del resultado obtenido lo multiplicamos por 100 y así obteniendo un resultado en porcentaje:

$$P = 0.8 * 100$$

$$Portabilidad = 80\%$$

Por lo que podemos deducir, que es portable.

4.1.1.5. Usabilidad

Para evaluar la usabilidad o facilidad de uso (FU) de un sistema, se utiliza una ecuación específica que considera diversos factores relevantes. Esta ecuación es una herramienta útil para medir y comparar la calidad de diferentes sistemas desde la perspectiva del usuario.

$$FU = \frac{[\frac{\sum x_i}{n} * 100]}{n}$$

A continuación, calcularemos x_i y $\sum x_i$ en la siguiente tabla 22 haciendo uso de la escala de evaluaciones:

Tabla 27

Evaluación de preguntas para el cálculo de usabilidad

| Nº | Preguntas | Evaluación (x_i) |
|----|--|-------------------------|
| 1 | ¿El sistema satisface los requerimientos de manejo de información? | 4 |
| 2 | ¿Las salidas del sistema están de acuerdo a sus requerimientos? | 4 |
| 3 | ¿Cómo considera el ingreso de datos del sistema? | 4 |
| 4 | ¿Cómo considera los formularios que elabora el sistema? | 4 |
| 5 | ¿El sistema facilita el trabajo que realiza? | 5 |
| 6 | Total $\sum x_i$ | 21 |

Nota: Son datos para el cálculo de usabilidad. Elaboración propia.

Reemplazando los datos en la ecuación obtendremos:

$$FU = \frac{[\frac{\sum x_i}{n} * 100]}{n}$$

$$FU = \frac{[\frac{21}{5} * 100]}{5}$$

$$FU = 84$$

$$\underline{Usabilidad = 84\%}$$

Entonces el resultado obtenido nos indica que la usabilidad es de 84%, por tanto, nos da a entender que es de fácil uso.

4.2 SEGURIDAD DEL SOFTWARE

La norma ISO/IEC 27001 evalúa y corrige la implementación según los estándares de la norma, con el objetivo de cumplir con las normas y lograr una mejora continua, que reducen los riesgos de incidentes de la información.

4.2.1 Seguridad Física

Ya que es inevitable el acceso físico a las máquinas sobre las que realizan sus labores se puso medidas y controles que se implementan para proteger los componentes físicos de un sistema informático, como los servidores, equipos de red, dispositivos de almacenamiento y otros dispositivos relacionados.

- **Control de acceso:** Se limita el acceso a las áreas de servidor o de almacenamiento de datos a solo aquellos que tengan una necesidad legítima de acceder a ellos.
- **Seguridad de la red:** se cuenta con protección de las redes de comunicación y los sistemas de almacenamiento de datos mediante la implementación de cortafuegos, detección de intrusiones y medidas de prevención de malware.
- **Protección contra incendios y desastres naturales:** también se cuenta con la implementación de medidas de seguridad física para prevenir daños a los sistemas de información en caso de incendios, inundaciones o terremotos.
- **Respuesta a incidentes:** Cuenta con un plan de acción en caso de incidentes de seguridad, como intrusiones o pérdidas de datos.

4.2.2 Seguridad Lógica

El sistema cuenta con las siguientes medidas de seguridad en el inicio de sesión:

- **Contraseñas seguras:** Se hace uso de algoritmos de hashing seguros para almacenar las contraseñas de los usuarios de forma segura en la base de datos. Esto garantiza que las contraseñas no se almacenen en texto plano y que sean difíciles de descifrar en caso de una brecha de seguridad.

```
use Illuminate\Support\Facades\Hash;  
  
'password' => Hash::make($data['password']),
```

- **Protección contra ataques de fuerza bruta:** El sistema incorpora mecanismos para protegerse contra ataques de fuerza bruta, como el bloqueo temporal de una cuenta después de varios intentos fallidos de inicio de sesión.
- **Middleware de autenticación:** El sistema proporciona middleware proporcionados por laravel, que se puede aplicar a rutas específicas o grupos de rutas para garantizar que solo los usuarios autenticados tengan acceso a esas rutas.

```
Route::group(['middleware' => 'auth'], function () {  
    Route::resource('user', 'App\Http\Controllers\UserController', ['except' => ['show']]);  
    Route::get('profile', ['as' => 'profile.edit', 'uses' => 'App\Http\Controllers\ProfileController@edit']);  
    Route::put('profile', ['as' => 'profile.update', 'uses' => 'App\Http\Controllers\ProfileController@update']);  
    Route::put('profile/password', ['as' => 'profile.password', 'uses' => 'App\Http\Controllers\ProfileController@password']);  
});
```

- **Protección contra ataques CSRF:** Incluye protección automática contra ataques de falsificación de solicitudes entre sitios (CSRF), lo que evita que se realicen acciones no autorizadas en nombre de un usuario.

```
$.ajaxSetup({
  headers: {
    'X-CSRF-TOKEN': $('meta[name="csrf-token"]').attr('content')
  }
});
```

4.2.3 Seguridad de copia de seguridad

El respaldo de base de datos implementado en nuestro sistema es una medida proactiva que garantiza la disponibilidad y protección de la información almacenada. A través de este proceso, se realizan copias de seguridad periódicas de la base de datos, capturando todos los datos y la estructura necesarios para su posterior restauración.

El proceso de respaldo se lleva a cabo utilizando la funcionalidad proporcionada por el paquete "spatie/laravel-backup". Este paquete nos permite programar y automatizar el respaldo de la base de datos de manera eficiente. Utilizamos una configuración específica que incluye la definición de la frecuencia de respaldo, el almacenamiento de las copias de seguridad y la retención de los archivos de respaldo.

4.2.4 Niveles de acceso y seguridad

A continuación, se muestra la tabla de los niveles de acceso que tiene cada usuario para el sistema.

Tabla 28

Niveles de acceso y seguridad Modulo Administración de Usuarios

| Usuarios | Operaciones | | | | |
|----------------------------------|-------------|-------|----------------|-----------|----------|
| | Altas | Bajas | Modificaciones | Consultas | Reportes |
| Administrador (Súper Usuario) | X | X | X | X | X |
| Supervisor de proyectos | X | | X | X | |

| | | | | |
|------------|--|--|---|---|
| Alcalde | | | X | |
| Secretaria | | | X | X |
| Dirigente | | | X | |

Nota: Elaboración propia.

Tabla 29

Niveles de acceso y seguridad Modulo de Gestión de Roles

| Usuarios | Operaciones | | | | |
|----------------------------------|-------------|-------|----------------|-----------|----------|
| | Altas | Bajas | Modificaciones | Consultas | Reportes |
| Administrador (Súper Usuario) | X | X | X | X | X |
| Supervisor de proyectos | | | | | |
| Alcalde | | | | | |
| Secretaria | | | | | |
| Dirigente | | | | | |

Nota: Elaboración propia.

Tabla 30

Niveles de acceso y seguridad Modulo de Gestión de POA

| Usuarios | Operaciones | | | | |
|----------------------------------|-------------|-------|----------------|-----------|----------|
| | Altas | Bajas | Modificaciones | Consultas | Reportes |
| Administrador (Súper Usuario) | X | X | X | X | X |
| Supervisor de proyectos | | | | | |
| Alcalde | | | | X | |
| Secretaria | | | | X | X |
| Dirigente | | | | X | |

Nota: Elaboración propia

Tabla 31*Niveles de acceso y seguridad Módulo Gestión de Recursos*

| Usuarios | Operaciones | | | | |
|----------------------------------|-------------|-------|----------------|-----------|----------|
| | Altas | Bajas | Modificaciones | Consultas | Reportes |
| Administrador (Súper Usuario) | X | X | X | X | X |
| Supervisor de proyectos | | | | | |
| Alcalde | | | | X | |
| Secretaria | | | | X | X |
| Dirigente | | | | X | |

Nota: Elaboración propia.

Tabla 32*Niveles de acceso y seguridad Modulo Proyectos Aprobados*

| Usuarios | Operaciones | | | | |
|----------------------------------|-------------|-------|----------------|-----------|----------|
| | Altas | Bajas | Modificaciones | Consultas | Reportes |
| Administrador (Súper Usuario) | X | X | X | X | X |
| Supervisor de proyectos | | | | | |
| Alcalde | | | | X | |
| Secretaria | | | | X | X |
| Dirigente | | | | X | |

Nota: Elaboración propia.

Tabla 33*Niveles de acceso y seguridad Módulo de Empresas*

| Usuarios | Operaciones | | | | |
|----------|-------------|-------|----------------|-----------|----------|
| | Altas | Bajas | Modificaciones | Consultas | Reportes |

| | | | | | |
|----------------------------------|---|---|---|---|---|
| Administrador (Súper Usuario) | X | X | X | X | X |
| Supervisor de proyectos | | | | X | X |
| Alcalde | | | | X | |
| Secretaria | | | | X | X |
| Dirigente | | | | X | X |

Nota: Elaboración propia

Tabla 34

Niveles de acceso y seguridad Módulo de Monitoreo de Proyectos

| Usuarios | Operaciones | | | | |
|----------------------------------|-------------|-------|----------------|-----------|----------|
| | Altas | Bajas | Modificaciones | Consultas | Reportes |
| Administrador (Súper Usuario) | X | X | X | X | X |
| Supervisor de proyectos | | | X | X | X |
| Alcalde | | | | X | |
| Secretaria | | | | X | X |
| Dirigente | | | | X | X |

Nota: Elaboración propia.

Tabla 35

Niveles de acceso y seguridad Modulo Cronograma de Actividades

| Usuarios | Operaciones | | | | |
|----------------------------------|-------------|-------|----------------|-----------|----------|
| | Altas | Bajas | Modificaciones | Consultas | Reportes |
| Administrador (Súper Usuario) | X | X | X | X | X |
| Supervisor de proyectos | X | | X | X | |
| Alcalde | | | | X | |

| | | | | |
|------------|--|--|---|---|
| Secretaria | | | X | X |
| Dirigente | | | X | X |

Nota: Elaboración propia

Tabla 36

Niveles de acceso y seguridad Modelo de Actividades

| Usuarios | Operaciones | | | | |
|----------------------------------|-------------|-------|----------------|-----------|----------|
| | Altas | Bajas | Modificaciones | Consultas | Reportes |
| Administrador (Súper Usuario) | X | X | X | X | X |
| Supervisor de proyectos | X | | X | X | |
| Alcalde | | | | X | |
| Secretaria | | | | X | X |
| Dirigente | | | | X | |

Nota: Elaboración propia.

4.3 ESTIMACION DE COSTOS PARA EL DESARROLLO DEL SOFTWARE

Hay varios enfoques para calcular los costos de desarrollo de software, los cuales básicamente consisten en establecer una fórmula matemática que relacione el esfuerzo y el tiempo de desarrollo del software. Para calcular el costo de software del presente proyecto, aremos uso del modelo Cocomo II. El modelo mencionado está estructurado jerárquicamente de la siguiente manera:

Las ecuaciones de Cocomo básico son conformados de 3 ecuaciones, a continuación, veremos las ecuaciones mencionadas:

$$E = a(KLDC)^b, \left(\frac{\text{personas}}{\text{mes}}\right)$$

$$T = c(E)^d, (\text{mes})$$

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

E = continuación, se explicara las variables de las ecuaciones mencionadas.

D = Esfuerzo requerido por el proyecto, en meses.

P = Número de personas requeridas por el proyecto.

a, b, c y d = Constantes con valores definidos, según cada sub – modelo.

KLDC = Cantidad de líneas de código, en miles.

Para las constantes aremos uso los datos que podremos apreciar en la tabla (35) en donde a y c es el coeficiente y b y d es el exponente:

Tabla 37

Coeficientes y exponentes

| Proyecto de software | a | b | c | d |
|----------------------|-----|------|-----|------|
| Orgánico | 2.4 | 1.05 | 2.2 | 0.38 |
| Semi-acoplado | 3.0 | 1.12 | 2.5 | 0.35 |
| Empotrado | 3.6 | 1.20 | 2.5 | 0.32 |

Nota: Son constantes para hallar el esfuerzo. Elaboración propia.

4.3.1 Costo del desarrollo del software

Para llevar a cabo el cálculo del desarrollo de software, se utilizará como punto de partida el valor del punto de función no ajustado que se obtuvo en el capítulo previo. Es importante tener en cuenta este valor para asegurar una estimación precisa y fiable de los costos y esfuerzos necesarios para el desarrollo del software.

$$PF = 778.55$$

El resultado de PF se lo convertirá a KLDC (Kilos Líneas de Código), para la conversión de aremos uso los datos de la siguiente tabla (24).

Tabla 38*Factor LCD/PF de lenguajes de programación*

| Lenguaje | Factor LDC/PF |
|-------------|---------------|
| Ensamblador | 320 |
| C | 150 |
| Cobol | 105 |
| Pascal | 91 |
| PHP | 29 |
| C++ | 64 |
| Visual C# | 34 |
| SQL | 12 |

Nota: Obtenido de QSM, 2020.

Con el dato de la tabla (25), se calculará las líneas de código del software haciendo uso de la siguiente ecuación:

$$LCD = PF * \frac{Factor\ LDC}{PF}$$

$$LDC = 778.55 * 29$$

$$PF = \underline{22577.95}$$

Y para obtener en (KLDC) se realizará la conversión dividiendo entre 1000 de la siguiente manera:

$$KLCD = \frac{LDC}{1000}$$

$$KLCD = \frac{22577.95}{1000}$$

$$\underline{KLCD = 22.58}$$

Luego de obtener KLDC, se calculará el esfuerzo necesario para el desarrollo del software. Y para esto haremos uso de la siguiente ecuación:

$$E = a * (KLDC)^b, \left(\frac{\text{personas}}{\text{mes}}\right)$$

La ecuación mencionada nos ayudara a hallar el esfuerzo, seguidamente reemplazamos los datos a las variables de la ecuación:

$$E = 2.4 * (22.58)^{1.05}$$

$$E = 63.3, \left(\frac{\text{personas}}{\text{mes}}\right)$$

Para hallar el tiempo requerido tomamos los datos de la tabla (23), donde c = 2.2 y d = 0.38 y reemplazando los datos en la siguiente ecuación tenemos:

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

$$T = 2.2 * (63.3)^{0.38}$$

$$\underline{T = 10.6 \cong 10 (\text{meses})}$$

El resultado obtenido nos indica que para el desarrollo del proyecto se requiere un lapso de 10 meses.

Ahora para calcular el personal requerido para el desarrollo del proyecto se calcula con la ayuda de la siguiente ecuación:

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

$$P = \frac{63.3}{10.6}, (\text{personas})$$

$$\underline{P = 5.9, (\text{personas}) \cong 6 (\text{desarrolladores})}$$

Para hallar el salario de un desarrollador, tomaremos en cuenta el siguiente salario de un desarrollador que oscila entre 500 (\$):

$$\text{Costo del software por persona} = P * \text{salario de un desarrollador}$$

$$\text{Costo del software por persona} = 6 * 500 (\$)$$

$$\underline{\text{Costo del software por persona} = 3000 (\$)}$$

Realizaremos la conversión de \$ a Bs. Tomando en cuenta que dólar en la actualidad esta \$ = 6.91.

$$\text{Costo total de desarrollo} = \text{costo total del software desarrollado} * (\$)\text{Actual}$$

$$\text{Costo total de desarrollo} = 3000 * 6.91$$

$$\underline{\text{Costo total de desarrollo} = 20730 (\text{Bs.})}$$

4.3.2 Costo de elaboración de proyecto

En concreto, los costos de elaboración del proyecto corresponden a los gastos asociados con la investigación y el análisis del sistema en la fase de recopilación de datos. Estos costos se presentan en una tabla que puede incluir diferentes elementos relacionados con el proceso de planificación.

Tabla 39

Costo de elaboración de proyecto

| Detalle | Importe (Bs.) |
|-------------------------------|---------------|
| Análisis y diseño del sistema | 500 |
| Bibliografía | 600 |
| Material de escritorio | 800 |
| Internet | 1000 |
| Otros | 450 |
| Total | 3350 |

Nota: describe el costo y el costo total de elaboración del proyecto. Elaboración propia.

4.3.3 Costo total del proyecto

El costo total se calcula sumando el costo del software desarrollado y el costo de elaboración del proyecto, los cuales se encuentran desglosados en la tabla (26).

Tabla 40

Costo total del proyecto

| Detalle | Importe (Bs.) |
|---|---------------|
| Costo total del software desarrollado | 20730 |
| Costo total de elaboración del proyecto | 3350 |
| Total | 24080 |

Nota: Costo total del proyecto. Elaboración propia.

Por tanto, el costo total del desarrollo del presente proyecto es de 24080 Bs desde el inicio del proyecto hasta su conclusión.

CAPITULO V
CONCLUSIONES Y
RECOMENDACIONES

5.1 CONCLUSIONES

Ya cumplido con los objetivos del proyecto denominado “SISTEMA DE MONITOREO PARA EL PROCESO DE CONTROL DE PROYECTOS VIA WEB” para el Gobierno Autónomo Municipal de Batallas, se ha logrado alcanzar el objetivo principal establecido según los requerimientos del municipio que a continuación veremos más detallado:

- Se logró la centralización de la gestión de proyectos que se encuentran en ejecución, evitando el desorden caótico de la información requerida para la realización de reportes e informes.
- Se logró centralizar la información de todas las empresas que están relacionadas con la realización de los proyectos.
- Se logró recopilar y registrar en la base de datos la información completa y actualizada de todas comunidades según al cantón que pertenecen y que conforman el municipio de Batallas, para un registro completo de todas las comunidades y para comprender mejor las necesidades específicas de cada una de las comunidades.
- Se optimizo la búsqueda de información de recursos para un buen reporte e informe de los recursos gestionados para la ejecución del proyecto.
- Se optimizó el monitoreo y control de proyectos, mediante los cronogramas de actividades y actividades cumpliendo fechas de inicio, fechas de finalización y mediante imágenes.
- Se mejoró la visualización de las actividades en desarrollo, en ejecución y pendiente para un buen control de proyectos.

- Implementar módulos de reportes a cada uno de los módulos, para tener la facilidad de ubicar los reportes, evitando la confusión de descargar el reporte no requerido.
- Se logró mejorar la optimización de generación de reportes de todos los módulos.

El sistema implementado ha superado el desafío de descentralización de datos, proporcionando una solución efectiva para la generación de informes y la toma de decisiones. Ahora, contamos con una base de datos centralizada, informes automatizados y acceso oportuno a la información, lo que ha mejorado significativamente nuestra capacidad para tomar decisiones informadas de manera rápida y precisa.

Hacer uso de un método ágil en el desarrollo de sistemas web es de suma importancia, ya que proporciona una estructura y un enfoque claros para lograr un desarrollo exitoso del sistema.

5.2 RECOMENDACIONES

- Se recomienda la revisión periódica por cierto periodo de tiempo para la eficiencia y un funcionamiento adecuado del sistema.
- Recomendar a la institución tener cuidado a la asignación de roles.
- Se debe realizar Backus mensualmente de la base de dato.
- Proporcionar capacitación a los nuevos administradores con el fin de que adquieran las habilidades necesarias para llevar a cabo operaciones del sistema de manera efectiva y lograr una correcta gestión del mismo.

- Se sugiere a la institución que adopte, utilice y gestione el sistema de acuerdo con las instrucciones proporcionadas.

Tras la realización del proyecto grado, se sugieren implementar los siguientes módulos:

- Desarrollar e implementar módulos que brinden seguimiento financiero a los departamentos de finanzas, licitaciones y contrataciones.
- Se recomienda implementar módulos que brinden seguimiento licitaciones.

BIBLIOGRAFÍA

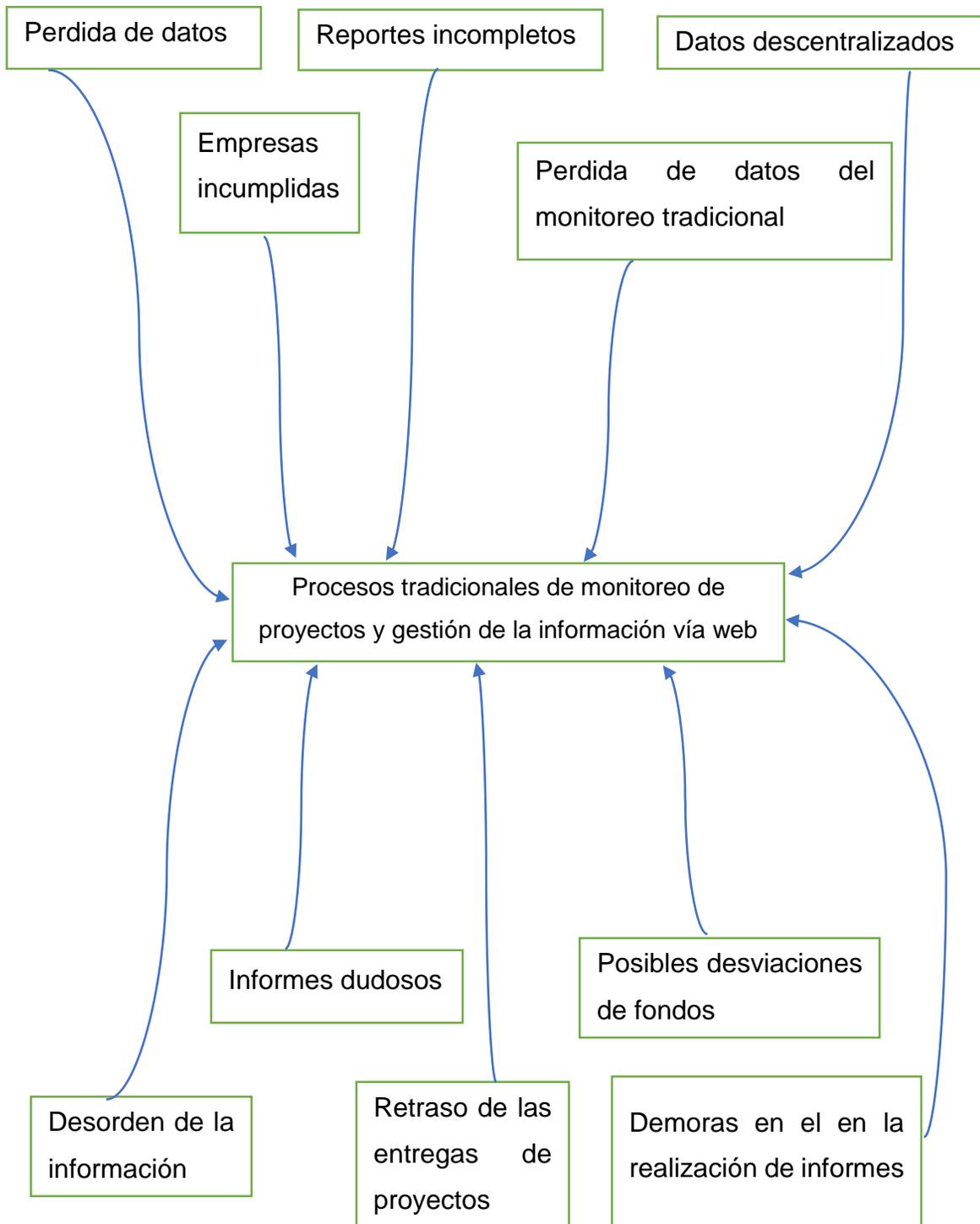
- Acosta, N. J., Espinel, L. A., & García, L. J. (2017). Estándares para la calidad de software. *TIA*, 5(1), 75 - 84.
<https://doi.org/https://revistas.udistrital.edu.co/index.php/tia/article/view/8388/pdf>
- Altube, V. R. (31 de Marzo de 2021). *Blog de programación y sistemas*. OpenWebinars: <https://openwebinars.net/blog/que-es-laravel-caracteristicas-y-ventajas/>
- Arias Chaves, M. (7 de Julio de 2006). Ingenieria de requerimientos y su importancia en el desarrollo de proyectos de software. *Revista de las sedes regionales*(VI(10)), 1 - 13.
- Aston, B. (3 de Marzo de 2023). *DPM*. DPM: <https://thedigitalprojectmanager.com/es/temas/guia-estimacion-presupuesto-costos/>
- Aurazo, E. (30 de Marzo de 2017). *KUPDF*. KUPDF.inc: https://kupdf.net/download/estudio-de-uwe-metodologia-de-desarrollo-web_58dc375bdc0d604d128971b2_pdf
- Berzosa, V. (17 de Marzo de 2023). *Quees*. ¿Qué es un sistema web según autores, libros e internet?: <https://quees.com/sistema-web/>
- Boconsaca Armijos, S. V. (2020). *SISTEMA DE INFORMACIÓN WEB PARA LA GESTIÓN ADMINISTRATIVA Y CONTROL DE RUTAS DE LA COMPAÑÍA DE TRANSPORTE TRANSBALAOTUR S.A.* UNIVERSIDAD AGRARIA DEL ECUADOR FACULTAD DE CIENCIAS AGRARIAS CARRERA DE INGENIERÍA EN COMPUTACIÓN EN INFORMÁTICA, Milagro, Ecuador.
- Calero, W. (7 de Octubre de 2010). *Ingenieria de Software*. <http://ingenieraupoliana.blogspot.com/2010/10/cocomo.html>
- Choque Tito, E. H. (s.f.). *SISTEMA DE INFORMACIÓN PARA EL SEGUIMIENTO Y CONTROL DE EJECUCIÓN DE PROYECTOS MUNICIPALES CASO: GOBIERNO AUTÓNOMO MUNICIPAL DE CATACORA*. Universidad Pública de el Alto - Ingenieria de Sistemas, El Alto, Bolivia.

- Cussi Merlo, H. A. (s.f.). *Sistema web de seguimiento y control de la información de clientes*. Universidad Mayor de San Andrés, La Paz, Bolivia.
- García Jara, E. B. (2018). *Análisis, diseño e implementación de un aplicativo web para el seguimiento y control de pedidos en la empresa Lima Tours S.A.C.* Universidad Peruana de las Américas, Lima, Perú.
- Gauchat, J. D. (2017). *El Gran Libro de HTML5, CSS3 y JavaScript* (Vol. 3). Barcelona, España: Marcombo.
- HostingPlus. (14 de Diciembre de 2020). *Hostingplus.pe*. Qué es MariaDB y cuáles son sus características: <https://www.hostingplus.pe/blog/que-es-mariadb-y-cuales-son-sus-caracteristicas/>
- IBNORCA. (20 de Junio de 2022). *ibnorca*. <https://www.ibnorca.org/es/certificaciones/gestion-de-la-seguridad-de-la-informacion-isoiec-27001>
- Instituto Nacional de Estadística. (Julio de 2019). *Catálogo ANDA*. <https://anda.ine.gob.bo/index.php/catalog/8>
- Laravel LLC. (17 de Marzo de 2023). *laravel*. <https://laravel.com/docs/10.x/installation>
- LATINCLOUD. (10 de Marzo de 2023). *Latincloud Blog*. <https://latincloud.com/blog/que-es-bootstrap-5/>
- LMU – Ludwig-Maximilians-Universität München. (10 de Agosto de 2016). UWE – UML-based Web Engineering: <https://uwe.pst.ifi.lmu.de/>
- LMU – Ludwig-Maximilians-Universität München. (10 de Agosto de 2016). UWE – UML-based Web Engineering. <https://uwe.pst.ifi.lmu.de/index.html>
- Lone, C. (10 de 25 de 2022). DNV. Det Norske Veritas: <https://www.dnv.com.mx/news/iso-publica-una-nueva-version-de-iso-iec-27001-234321>
- Mendoza Rivilla, J. E. (2017). *Implementación de sistema web para la gestión y control de los procesos de la unidad de titulación de la carrera de Ingeniería de Sistemas de la Universidad Salesiana*. Universidad Politécnica Salesiana Ecuador, Guayaquil, Ecuador.
- Milenium. (20 de Junio de 2022). *Milenium*. Qué es internet: <https://www.informaticamilenium.com.mx/es/temas/que-es-internet.html>

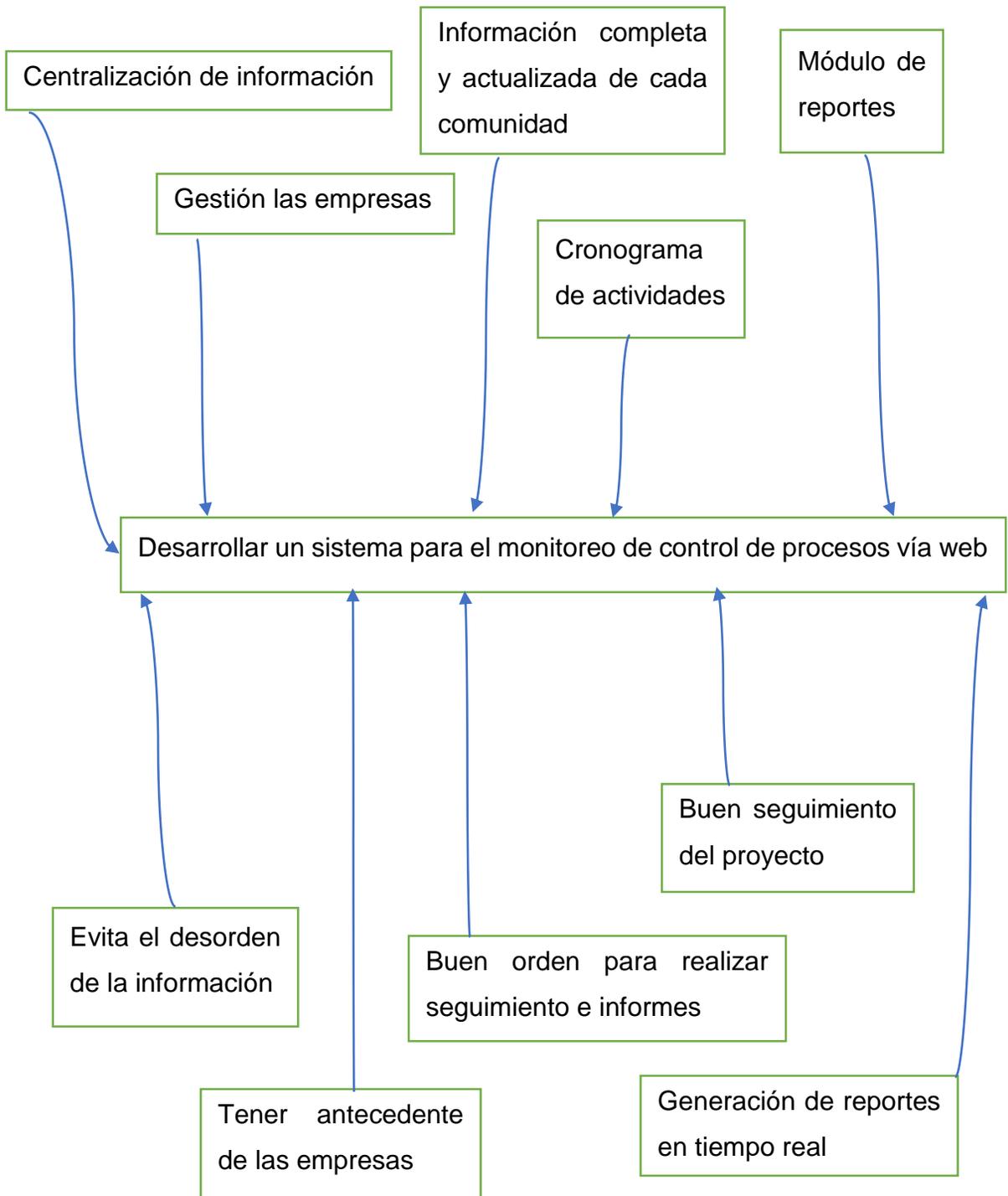
- OPS. (1999). *Manual de Monitoreo de la Ejecucion de Proyectos*. Washington, EEUU.
- Pressman, R. S. (2010). *Ingeniería de software: un enfoque práctico* (7 ed.). Mexico: McGraw Hill.
- Rivero, G. F. (s.f.). *Sistema web de control y seguimiento de obras municipales para la subalcaldía de ovejuyo d-i municipio* .
- Sangalli Paco, A. (2020). *Sistema de control y seguimiento de proyectos esducativos*. Universidad Pública de El Alto, El Alto, Bolivia.
- Sommerville, I. (2005). *Ingenieria de Software* (Vol. 7). Madrid, España: PearsonEducación S.A.
- Uchuquicaña Ranilla, J. I. (2017). *Implementacion de un sistema de información orientado al la empresa publigarment S.A.C. en Lima-2017*. Lima, Peru.
- UNIR. (6 de Abril de 2021). *La Universidad en Internet*. Ingeniería de software: qué es, objetivos y funciones del ingeniero:
<https://mexico.unir.net/ingenieria/noticias/ingenieria-de-software-que-es-objetivos/>
- Villarroel Apaza, V. (2020). *Sistema web para la gestion de procesis de pasantía y practicas profecionales*. Universidad Pública de El Alto, El Alto, Bolivia.

ANEXOS

ANEXO A: ARBOL DE PROBLEMAS



ANEXO B: ARBOL DE OBJETIVOS



ANEXO B: CUESTIONARIO DE LAS ENTREVISTAS

Nombres y apellidos:

Cargo:

Preguntas a funcionarios de la GAMB.

¿Existe un sistema de información para el monitoreo de proyectos?

R.-

¿Qué tipos de proyectos manejan?

R.-

¿Los proyectos son entregados a tiempo?

R.-

¿La información sobre los proyectos es inmediata?

R.-

¿De qué manera se manipula la información?

R.-

¿Quiénes son los encargados de monitorear los proyectos?

R.-

¿Quiénes son los encargados de registrar los proyectos aprobados?

R.-.

¿Quién realiza los informes?

R.-

Preguntas a Dirigentes de Comunidades.

¿Cuántas veces visitan a la alcaldía para solicitar informe?

R.-

¿Qué información requieren?

R.-

¿A quién solicitan el informe?

R.-

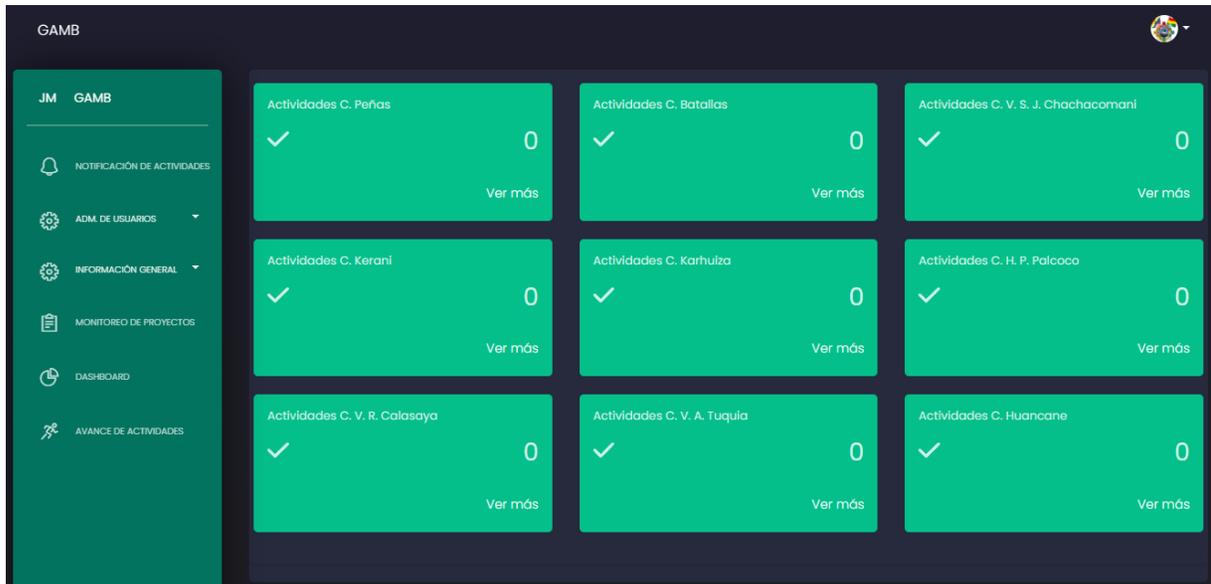
¿Tienen preocupación por posibles desvíos sobre la gestión de recursos?

R.-

ANEXO D: Manual de usuario

Manual De Usuario

SISTEMA DE MONITOREO PARA EL PROCESO DE CONTROL DE PROYECTOS VIA WEB CASO: GOBIERNO AUTÓNOMO MUNICIPAL DE BATALLAS



1. INTRODUCCIÓN

En la actualidad, el uso de sistemas informáticos se ha vuelto fundamental para la gestión eficiente de la información en diversas organizaciones. En el caso del Gobierno Autónomo Municipal de Batallas, es esencial contar con un sistema web para el monitoreo de proyectos, que permita gestionar de manera efectiva los proyectos municipales que están en ejecución.

Según lo descrito se desarrolló, el SISTEMA DE MONITOREO PARA EL PROCESO DE CONTROL DE PROYECTOS VIA WEB, que centraliza y optimiza los procesos tradicionales de monitoreo y manejo de información.

2. OBJETIVO

El objetivo principal del manual de usuario es orientar sobre el uso del sistema.

3. TIPOS DE USUARIOS

- Administrador (Súper Usuario), tiene todos los privilegios y permisos al sistema.
- Supervisor de proyectos, Actualizar en módulo de monitoreo de proyectos.
- Alcalde, ver los módulos de POA, recursos, proyectos aprobados y proyectos.

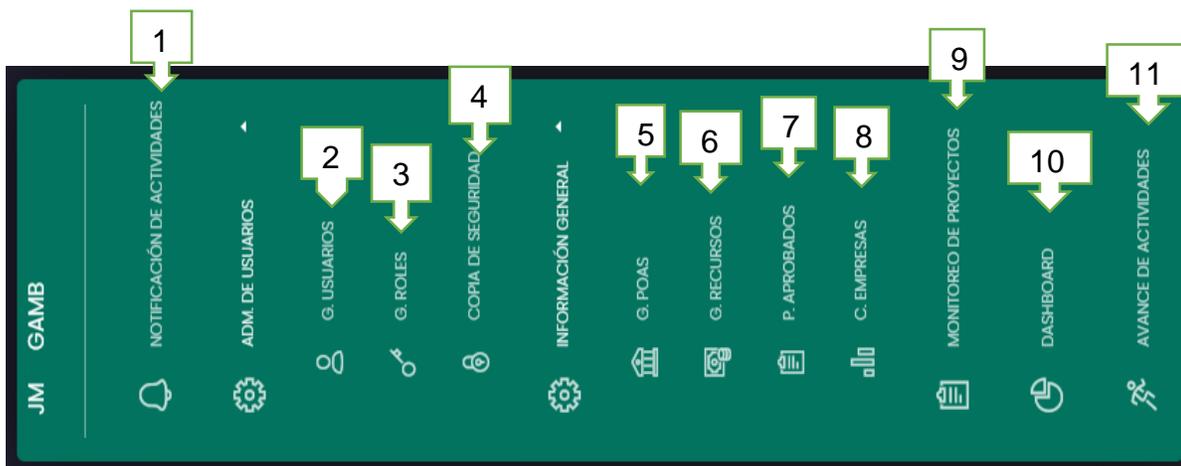
- Secretaria y Dirigente, ver y reportes, módulo de POA, recursos, proyectos aprobados, empresas, monitoreo de proyectos, dashboard y avance de actividades.

4. INGRESO AL SISTEMA

4.1 Pantalla de Inicio de sesión



1. Ingrese su nombre de usuario o correo electrónico.
2. Ingrese la contraseña.
3. Marcar no soy un robot.
4. Clic en el botón Ingresar al sistema, y el sistema mostrara la página de inicio según sus roles.



1. Módulo de Notificación de actividades.
2. Módulo de Administración de usuarios
3. Módulo de Administración de roles.
4. Módulo de Copia de seguridad
5. Módulo de Gestión de POA
6. Módulo de Recursos.
7. Módulo de Proyectos aprobados.
8. Módulo de Directorio de empresas
9. Módulo de Proyectos aprobados
10. Módulo de Dashboard
11. Módulo de Avance de Actividades

5. ADMINISTRACIÓN DE USUARIOS

Este módulo permite administrar (Agregar, Editar y eliminar), cabe mencionar que al crear un usuario se debe asignar un rol y el estado en activo.

The screenshot shows a user management interface titled 'GESTIÓN DE USUARIOS'. It features a table with columns for ID, USUARIO, EMAIL, ROL, ESTADO, and ACCIONES. Two users are listed: 'Admin Admin' (ID 1) and 'Edwin' (ID 2). The interface includes a search bar, a dropdown for 'Mostrar 5 registros', and a 'Buscar:' field. Callouts with arrows point to specific actions: 'Crear Usuarios' points to a plus icon, 'Generar reporte' points to a document icon, 'Ver detalle' points to a magnifying glass icon, 'Editar Usuario' points to a pencil icon, and 'Eliminar' points to a trash can icon.

| ID | USUARIO | EMAIL | ROL | ESTADO | ACCIONES |
|----|-------------|-----------------------------|-----|--------|----------|
| 1 | Admin Admin | admin@gmail.com | | | |
| 2 | Edwin | george.kingleorik@gmail.com | | | |

Agregar un nuevo usuario

DATOS PERSONALES

Nombre

Apellido Paterno

Apellido Materno

Cédula

Dirección

Fecha de Nacimiento

Telefono

Género Masculino Femenino

CREAR CUENTA

Nombre de Usuario

Correo Electrónico

Confirmar Contraseña

Roles

Estado

Guardar **Cancelar**

Selecctionar Activo o Inactivo

Selecctionar Rol

5. ADMINISTRACION DE ROLES

Gestión de Roles

 ← Crear Nuevo rol

| ROLES DE USUARIOS | ACCIONES |
|----------------------------|---|
| Alcalde |   |
| Secretaria |   |
| Administrador |  |
| Dirigente |   |
| Supervisor de Proyecto |   |
| Secretaria - Administrador |   |

Editar Rol

Eliminar Rol

Agregar Rol

Nombre del Rol:
Alcalde

Marcar para asignar el rol

Presionar el botón para guardar el rol creado

Guardar

7. GESTIÓN DE POAS

Ingresar fecha de inicio y fecha final para generar reportes

Fecha Inicio dd/mm/aaaa Fecha Final dd/mm/aaaa PDF

Mostrar 10 registros

Buscar:

| NOMBRE | GESTIÓN | COMUNIDAD | MONTO BS. | SALDO BS. | OBSERVACIÓN | ACCIONES |
|-------------------------------------|------------|--------------|-----------|-----------|-------------|------------------------|
| Poa de la comunidad de Peñas | 2023-06-22 | Peñas | | | | Crear Nuevo POA |
| Poa de la comunidad de Pajcha Peñas | 2023-06-22 | Pajcha Peñas | 120000.00 | 39381 | | |
| Poa de la comunidad de Huancuyo | 2023-06-22 | Huancuyo | 90000.00 | 90000 | | |
| Poa de la comunidad de Challapata | 2023-06-22 | Challapata | 95000.00 | 95000 | | |
| Poa de la comunidad de Sojata | 2023-06-22 | Sojata | 122587.00 | | | Editar POA |
| Poa de la comunidad de Batallas | 2023-06-22 | Batallas | 350798.00 | 350798 | | |
| Poa de la comunidad de Catacora | 2023-06-22 | Catacora | 78450.00 | | | Eliminar POA |
| Poa de la comunidad de Chirapaca | 2023-06-22 | Chirapaca | 180920.00 | 180920 | | |
| Poa de la comunidad de Igachi | 2023-06-22 | Igachi | 145456.00 | 145456 | | |
| Poa de la comunidad de Yaurichambi | 2023-06-22 | Yaurichambi | 70456.00 | 70456 | | |

Mostrando 1 a 10 de 43 registros

Anterior 1 2 3 4 5 Siguiente

Agregar un nuevo POA

The screenshot shows a form titled "AGREGAR NUEVO POA" with the following fields and annotations:

- Comunidad:** A dropdown menu with the text "-- Escoja la Comunidad --". An annotation box says "Seleccionar Comunidad" with an arrow pointing to the dropdown.
- Gestión:** A date input field with the placeholder "dd/mm/aaaa".
- Monto Asignado/Bs.:** A text input field with the placeholder "Introduzca el monto". An annotation box says "Solo ingresar el Monto a asignar" with an arrow pointing to this field.
- Monto Saldo/Bs.:** A text input field with the placeholder "Introduzca el saldo".
- Observación:** A text input field with the placeholder "Introduzca la observación".
- Buttons:** "Cerrar" (left), "Oprimir para guardar Nuevo" (center, with an arrow pointing right), and "Guardar" (right).

8. GESTIÓN DE RECURSOS

The screenshot shows the "RECURSOS PROYECTOS" management interface with the following elements and annotations:

- Form Fields:** "Fecha Inicio" and "Fecha Final" (both with "dd/mm/aaaa" placeholders). An annotation box says "Ingresar fecha de inicio y fecha final para generar reportes" with an arrow pointing to the date fields.
- Buttons:** "Crear Nuevo recurso" (with an annotation box), "Buscar:" (with a search input), "PDF" (with a download icon), and a plus icon.
- Table:** A table with columns: NOMBRE, FECHA DE ASIGNACIÓN, MONTOS, ESTADO, and ACCIONES. It contains 4 rows of resource data.
- Annotations:** "Editar recursos" (with an arrow pointing to the edit icon in the actions column) and "Eliminar recursos" (with an arrow pointing to the delete icon in the actions column).
- Footer:** "Mostrando 1 a 4 de 4 registros", "Anterior", "1", and "Siguiente".

Agregar Nuevo Recurso

AGREGAR NUEVO

Escoja la POA Existente

Si al escoger el POA ya no crea, es porque ya no tiene POA y se le notificara en una alerta

Escoja una de las Poas

-- Escoja la Poa --

Nombre del Proyecto

Introduzca el nombre del recurso

Monto Poa /Bs.

Introduzca el monto

Fecha de Asignación

dd/mm/aaaa

Estado

Activo

Observación

Introduzca el Observación

Cerrar

Oprimir para guardar Recurso

Guardar

9. PROYECTOS APROBADOS

PROYECTOS APROBADOS CON RECURSOS DE LA GESTIÓN 2023

Fecha Inicio dd/mm/aaaa Fecha Final dd/mm/aaaa

Mostrar 5 registros

Buscar:

| NOMBRE | FECHA APROBADO | EMPRESA | ESTADO | ACCIONES |
|--|-------------------------------|---|---------|---|
| Construcción Modulo Policial Peñas | Poa de la comunidad de Peñas, | Soci Com | |   |
| Refacción Aulas Colegio Tu Katari | | Sin Asignar | vigente |   |
| Mejoramiento de Ganado Ovino | | Sin Asignar | vigente |   |
| Construcción muro Perimetral Cementerio - Pajcha Peñas | 2023-05-04 | Poa de la comunidad de Pajcha Peñas, Asignación 2 con un monto de 3682.00 Bs. | |   |

Mostrando 1 a 4 de 4 registros

Anterior 1 Siguiente

Agregar un nuevo proyecto aprobado

Editar Proyecto aprobado

Eliminar Proyecto Aprobado

Si no tiene asignado una empresa no se podrá crear para hacer el Monitoreo de proyectos

Agregar nuevo proyecto aprobado

AGREGAR NUEVO PROYECTO APROBADO

Ingresar la fecha de aprobación de del proyecto aprobado

Nombre del Proyecto Aprobado

Fecha de Aprobación del Proyecto

Seleccionar Recurso

Las empresas en se asignan en la actualización, pero si tuviera, se le asigna al crear.

Fecha: dd/mm/aaaa

Recurso: -- Escoga la Recurso --

Categoría: -- Escoga la Categoría --

Empresa: -- Escoga la Empresa --

Estado: Vigente

Seleccione la categoría a la que pertenece

Cerrar

Oprimir para guardar los datos

Guardar

10. DIRECTORIO DE EMPRESAS

DIRECTORIO DE EMPRESAS

Crear Nueva Empresa

Mostrar 5 registros

Generar reporte

Buscar:

| NOMBRE | EMAIL | ESTADO | ACCIONES |
|------------------------------------|-------------------|--------|--------------------------------|
| SOCIEDAD CONSTRUCTORA CONSTEC S... | | | Ver más detalles de la empresa |
| Empresa Constructora ASOCIADOS | asociatesAT@gmail | | Editar la empresa |

Mostrando 1 a 2 de 2 registros

Eliminar empresa

11. MONITOREO DE PROYECTOS

GESTIÓN DE MONITOREO DE PROYECTOS

Crear nuevo Proyecto

Mostrar 5 registros

Reporte de todos los proyectos en general

Editar Proyecto

Eliminar Proyecto

Este botón redirige a Cronograma de Actividades

Generar Reporte del general del proyecto

Anterior 1 Siguiente

 al oprimir el botón Cronograma de Actividades redirige a la siguiente página:

CRONOGRAMA DE ACTIVIDADES

PROYECTO: Construcción Modulo Policial Peñas

Agregar cronograma actividad

Generar reportes

Mostrar 5 registros

Ver imagen

Subir imagen

Nueva Página Actividades

Editar Cronograma Actividad

Editar Cronograma actividad

Mostrando 1 a 5 de 6 registros

Anterior 1 2 Siguiente



al oprimir el botón se redirigirá a la página de ACTIVIDADES:

ACTIVIDADES

FASE: 1 ACTIVIDAD: Obras Preliminares

Mostrar registros

Buscar:

| NOMBRE | INICIO | FINAL | IMAGEN | ESTADO | ACCIONES |
|--------------------------------|------------|------------|--------|------------|----------|
| Limpieza del Terreno | 2023-05-25 | | | Desarrollo | |
| Replanteo y Nivelación de Obra | 2023-05-27 | 2023-06-01 | | | |

Mostrando 1 a 2 de 2 registros

Anterior | Siguiente

Annotations:

- Agregar nueva actividad (points to + icon)
- Generar Reporte (points to PDF icon)
- Ver imagen (points to image icon in first row)
- Subir imagen (points to image upload icon in first row)
- Editar actividad (points to edit icon in second row)
- Eliminar nueva (points to delete icon in second row)
- Ver Detalle (points to refresh icon in second row)

11. BASHBOARD

Canton Peñas

Canton Batallas

Generar reporte (points to report icon)

Datos Estadísticos (points to statistics icon)

Número de Proyectos por Categorías

- Peñas
- Pajcha Peñas
- Huancuyo
- Challapata
- Sojata
- Educacion
- Productivos
- Vivienda y Urbanizacion
- Sociales-Comunitarios

Education 1

En esta página se visualiza datos estadísticos de número de proyectos por comunidades y numérelo de proyectos por categoría.

Canton Villa

Canton Kerani

12. AVANCE DE ACTIVIDADES

Actividades Segun Su Avance

Mostrar registros

Generar Reportes 

Lista de las actividades según su estado: Pendiente, Desarrollo y concluido.

| PROYECTO | FASE | INICIO | FINAL | ESTADO |
|------------------------------------|---------|------------|------------|------------|
| Construcción Modulo Policial Peñas | FASE: 1 | 2023-05-25 | 2023-05-27 | desarrollo |
| Construcción Modulo Policial Peñas | FASE: 1 | 2023-05-27 | 2023-06-01 | desarrollo |
| Construcción Modulo Policial Peñas | FASE: 2 | 2023-06-01 | 2023-06-08 | desarrollo |
| Construcción Modulo Policial Peñas | FASE: 2 | 2023-06-08 | 2023-06-28 | desarrollo |
| Construcción Modulo Policial Peñas | FASE: 2 | 2023-06-28 | 2023-07-05 | desarrollo |

Mostrando 1 a 5 de 18 registros

Anterior 1 2 3 4 Siguiente

13. RESPALDO DE SEGURIDAD

PANEL DE COPIA DE SEGURIDAD

Volver a la página principal [Crear copia de seguridad](#)

Retornar a la página principal [Copia de seguridad general](#)

| DISCO | SIN AMENAZA | CANTIDAD DE ARCHIVOS | MÁS RECIENTE | ALMACENAMIENTO USADO |
|-------|---|----------------------|--------------|----------------------|
| local |  | 6 | | 23.03 MB |

Historial de Copia de Seguridad

| RUTA | FECHA DE CREACIÓN | TAMAÑO | ACCIONES |
|---|---------------------|---------|---|
| Laravel/backup-only-files-2023-06-26-07-41-10.zip | 2023-06-26 07:41:10 | | Descargar copia de seguridad Descargar Eliminar |
| Laravel/backup-only-files-2023-06-26-06-29-54.zip | 2023-06-26 06:29:57 | 5.19 MB | Descargar Eliminar |
| Laravel/backup-only-files-2023-06-26-06-28-31.zip | 2023-06-26 06:28:34 | 5.19 MB | Descargar Eliminar |
| Laravel/backup-2023-06-25-15-16-33.zip | 2023-06-25 15:16:34 | | Eliminar copia de seguridad Eliminar |
| Laravel/2023-05-20-15-10-36.zip | 2023-05-20 15:10:36 | 3.72 MB | Descargar Eliminar |
| Laravel/2023-05-18-11-47-10.zip | 2023-05-18 11:47:10 | 3.72 MB | Descargar Eliminar |

Para ver los datos de la copia de seguridad que se descargara en un archivo.zip tiene que realizar los siguientes pasos:

- 1.- Descomprimir el archivo.zip.
- 2.- Se descomprimirá un archivo.sql.
- 3.- Luego se tendrá que descargar e instalar XAMPP o LARAGON.
- 4.- Crear la base de datos con el mismo nombre.
- 5.- Luego importar el achivo.sql. o jalar directamente el archivo a PhpMyAdmin.
- 6.- Luego presionar en restaurar.
- 7.- De esa manera obtendrá los datos.

ANEXO D: MANUAL TECNICO

Este documento proporciona instrucciones detalladas para que cualquier usuario con conocimientos básicos en sistemas pueda instalar la aplicación web en el Gobierno Autónomo Municipal de Batallas.

Es esencial tener en consideración que este manual técnico menciona los requisitos mínimos de hardware, software y las versiones de las aplicaciones necesarias para llevar a cabo la instalación adecuada del sistema.

1. Requerimiento mínimo de hardware.

- Memoria RAM: Mínimo: 4 Gigabytes (GB)
- Espacio de disco duro mínimo: 5Gb.

2. Requerimiento mínimo de software

- Privilegios de administrador
- Sistema Operativo: recomendable en Linux para la parte de administración del sistema.
- Para el uso de sistema: todo los Windows, Linux, Mac y Android.

3. Herramientas de desarrollo

- **Php 8.1**
- **NodeJS 18.16**
- **Laravel 10**
- **Composer 2.5.5**
- **Zip – php**
- **Maria DB**

ANEXO D: AVALES

El Alto, junio de 2023

Señor:

M. Sc. Ing. Maricel Yarari Mamani

**DIRECTOR
CARRERA INGENIERIA DE SISTEMAS – UPEA**

Presente. –

REF.: AVAL DE CONFORMIDAD

Distinguido tutor metodológico:.

Tengo a bien dirigirme a su persona para comunicarle mi conformidad del trabajo final del Proyecto de Grado, titulado **“SISTEMA DE MONITOREO PARA EL PROCESO DE CONTROL DE PROYECTOS VÍA WEB”** caso: **GOBIERNO AUTÓNOMO MUNICIPAL DE BATALLAS**, elaborado por el universitario: **MAMANI SONCO JORGE LUIS** con **C.I. 135456** y **R.U. 200000133**, para su defensa pública y evaluación correspondiente a la materia de Taller de Grado II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de sistemas de la Universidad Pública de El Alto.

Sin otro particular me despido de usted.

Atentamente:



M. Sc. Ing. Hernán Quispe Conurana
TUTOR ESPECIALISTA

El Alto, junio de 2023

Señor:

M. Sc. Ing. David Carlos Mamani Quispe

**DIRECTOR
CARRERA INGENIERIA DE SISTEMAS – UPEA**

Presente. –

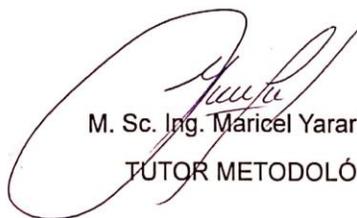
REF.: AVAL DE CONFORMIDAD

Distinguido director de carrera:

Tengo a bien dirigirme a su persona para comunicarle mi conformidad del trabajo final del Proyecto de Grado, titulado **“SISTEMA DE MONITOREO PARA EL PROCESO DE CONTROL DE PROYECTOS VÍA WEB”** caso: **GOBIERNO AUTÓNOMO MUNICIPAL DE BATALLAS**, elaborado por el universitario: **MAMANI SONCO JORGE LUIS** con **C.I. 135456** y **R.U. 200000133**, para su defensa pública y evaluación correspondiente a la materia de Taller de Grado II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de sistemas de la Universidad Pública de El Alto.

Sin otro particular me despido de usted.

Atentamente:



M. Sc. Ing. Maricel Yarari Mamani
TUTOR METODOLÓGICO

El Alto, junio de 2023

Señor:

M. Sc. Ing. Maricel Yarari Mamani

**DIRECTOR
CARRERA INGENIERIA DE SISTEMAS – UPEA**

Presente. –

REF.: AVAL DE CONFORMIDAD

Distinguido tutor metodológico:

Tengo a bien dirigirme a su persona para comunicarle mi conformidad del trabajo final del Proyecto de Grado, titulado **“SISTEMA DE MONITOREO PARA EL PROCESO DE CONTROL DE PROYECTOS VÍA WEB”** caso: **GOBIERNO AUTÓNOMO MUNICIPAL DE BATALLAS**, elaborado por el universitario: **MAMANI SONCO JORGE LUIS** con **C.I. 135456** y **R.U. 200000133**, para su defensa pública y evaluación correspondiente a la materia de Taller de Grado II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de sistemas de la Universidad Pública de El Alto.

Sin otro particular me despido de usted.

Atentamente:



Ing. Gabriel Reynaldo Sirpa Huayhua

TUTOR REVISOR



GOBIERNO AUTÓNOMO MUNICIPAL DE BATALLAS

Órgano Legislativo

Fundado el 17 de Septiembre de 1970 - 3ra. Sección - Provincia Los Andes
La Paz - Estado Plurinacional de Bolivia

Batallas
Tuquia
Pañas
Chachacomani
Kerani
Karhuiza
Calasaya
Huancané

El Alto, junio de 2023

Señor (a):

M. Sc. Ing. Maricel Yarari Mamani

TUTOR METODOLÓGICO

TALLER DE GRADO II

Presente. –

REF: CONFORMIDAD DEL DESARROLLO DEL PROYECTO DE GRADO

Distinguido tutor metodológico:

Mediante la presente tengo a bien comunicarle mi conformidad del Proyecto de Grado titulado: "SISTEMA DE MONITOREO PARA EL PROCESO DE CONTROL DE PROYECTOS VÍA WEB" caso: GOBIERNO AUTÓNOMO MUNICIPAL DE BATALLAS por el Sr. JORGE LUIS MAMANI SONCO con C.I. 10035456 L.P. y R.U. 200000133.

De tal forma cabe recalcar que el "SISTEMA DE MONITOREO PARA EL PROCESO DE CONTROL DE PROYECTOS VÍA WEB" se implementó y satisface los requerimientos de la institución, de esta forma se dio cumplimiento de los objetivos del presente.

En cuanto certifico, en honor a la verdad, para fines consiguientes del interesado para su defensa pública y evaluación correspondiente a la materia de Taller de Grado II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de sistemas de la Universidad Pública de El Alto.

Atentamente:


Edwin Hualpa Layme
RESIDENTE DEL CONCEJO
GOBIERNO AUTÓNOMO MUNICIPAL DE BATALLAS
PROV. LOS ANDES - LA PAZ - BOLIVIA

