

UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

SISTEMA DE INFORMACIÓN PARA EL REGISTRO Y CONTROL DE
DOCUMENTOS AMBIENTALES

CASO: (Dirección de Gestión y Control Ambiental – Gobierno Autónomo
Municipal de El Alto)

Para optar al título de Licenciatura en Ingeniería de Sistemas

Mención: INFORMÁTICA Y COMUNICACIONES

Postulante: Univ. Mariela Condori Calle

Tutor Metodológico: Ing. Marisol Arguedas Balladares

Tutor Especialista: Lic. Jesús Juan Rocha Vera

Tutor Revisor: Ing. Rubén David Quispe Amaru

EL ALTO – BOLIVIA

2020

DEDICATORIA

A ti Dios mío por darme la vida y permitirme llegar a esta etapa tan importante de mi vida profesional.

A ti papá Eustaquio C. por tu apoyo inmenso, siempre estuviste alentándome para salir adelante. A ti mamita querida Leonora C. por tu inmenso amor de madre.

A mi esposo Hernán por todo el apoyo, comprensión y por estar a mi lado.

A mis hijos Fernando y Rosa, a ustedes va dedicado este trabajo, porque ustedes son mi razón de vivir.

A mis hermanos Eddy y Carlos por sus palabras de aliento y apoyo...

Agradecimiento

A mi tutor especialista Lic. Jesús J. Rocha Vera, por el apoyo constante y las recomendaciones realizadas.

A mis tutores: Ing. Marisol Arguedas Balladares e Ing. David Rubén Quispe Amaru por las recomendaciones y observaciones realizadas.

A la Arq. Gabriela Tenorio Directora de la Dirección de Gestión y Control Ambiental, por su comprensión para que pueda implementar el presente proyecto.

A mis compañeros, amigos de mi vida universitaria con las cuales compartí momentos inolvidables.

RESUMEN

El presente proyecto contiene el proceso de desarrollo del Sistema de Información para El Registro y Control de Documentos Ambientales de las Unidades Industriales que se encuentran en el Municipio de El Alto, tiene como finalidad brindar información rápida, oportuna, fiable. Que permitirá un monitoreo adecuado.

Capítulo I. Marco Preliminar.

Este capítulo muestra un panorama general del proyecto propuesto, que contiene los antecedentes, la problemática, los objetivos, el alcance y la metodología a utilizar.

Capítulo II. Marco Teórico.

Este capítulo contiene el fundamento teórico del proyecto, describiendo la Ley del medio ambiente y el reglamento ambiental para el sector industrial manufacturero RASIM, la metodología AUP y las herramientas que fundamenta el desarrollo del proyecto.

Capítulo III. Marco Aplicativo.

Este capítulo muestra el diagnóstico de la situación actual, el análisis de requerimientos descritos mediante diagramas UML (diagramas de casos de uso) los procesos más importantes, el diseño y desarrollo del sistema aplicando los pasos de la metodología AUP. Seguidamente se encuentran las métricas y políticas de seguridad.

Capítulo IV: Conclusiones y Recomendaciones.

En este capítulo se muestra las conclusiones y recomendaciones que se obtuvieron durante el desarrollo del proyecto.

ÍNDICE CONTENIDO

CAPÍTULO I. MARCO PRELIMINAR

	Pág.
1.1. Introducción.....	1
1.2. Antecedentes.....	2
1.2.1. Antecedentes de la Institución.....	
1.2.1.1. Misión.....	
1.2.1.2. Visión.....	2
1.2.2. Antecedentes Académicos.....	2
1.3. Planteamiento del problema.....	3
1.3.1. Descripción del problema.....	3
1.3.2. Problema principal.....	4
1.3.3. Problemas secundarios.....	
1.4. Objetivos.....	4
1.4.1. Objetivo general.....	4
1.4.2. Objetivos específicos.....	5
1.5. Justificación.....	5
1.5.1. Justificación Técnica.....	5
1.5.2. Justificación Económica.....	5
1.5.3. Justificación Social.....	6
1.6. Metodología y técnicas.....	
1.6.1. Metodología Proceso Unificado Ágil (AUP).....	6
1.6.2. Metodología UML based Web Engineering (UWE).....	7
1.6.2.1. Las fases de la UWE.....	7
1.6.3. Técnicas.....	8
1.6.3.1. Técnica de recopilación de datos.....	8
1.6.3.2. Técnicas de seguridad.....	8
1.7. Límites y alcances.....	8
1.7.1. Límites.....	8
1.7.2. Alcances.....	9

CAPÍTULO II. MARCO TEÓRICO.

2.1. Sistema.....	10
2.2. Sistema de información.....	10
2.3. Ley del Medio Ambiente.....	10
2.3.1. Reglamento Ambiental para el Sector Industrial Manufacturero (RASIM).....	11

2.3.2.	Categorías de las industrias.....	11
2.3.3.	Instrumentos de Regulación De Alcance Particular (IRAP).....	12
2.3.4.	Registro Ambiental Industrial (RAI).....	12
2.3.5.	Estudio de Evaluación Impacto Ambiental y Plan de Manejo Ambiental (EEIA - PMA).....	13
2.3.6.	Descripción de Proyecto y Plan de Manejo Ambiental (DP - PMA)...	14
2.3.7.	Manifiesto Ambiental Industrial, Plan de Manejo Ambiental (MAI - PMA).....	14
2.3.8.	Vigencia del Plan de Manifiesto Ambiental (PMA).....	15
2.3.9.	Informe Ambiental Industrial (IIA).....	15
2.4.	Métodos, Técnicas e Instrumentos.....	15
2.4.1.	Observación.....	15
2.4.2.	Cuestionario.....	16
2.4.3.	Entrevista.....	16
2.5.	Ingeniería de requerimientos.....	16
2.6.	Ingeniería de software.....	17
2.7.	Proceso Unificado Ágil (AUP).....	18
2.7.1.	Ciclo de vida del Proceso Unificado Ágil (AUP).....	18
2.7.2.	Fases de Proceso Unificado Ágil (AUP).....	19
2.7.2.1.	Inicio (Concepción).....	19
2.7.2.2.	Elaboración.....	20
2.7.2.3.	Construcción.....	20
2.7.2.4.	Transición.....	20
2.7.3.	Disciplinas de AUP.....	21
2.7.4.	Principios de AUP.....	22
2.8.	UWE (UML – Base Web Engineering).....	22
2.8.1.	Actividades de modelado de UWE.....	23
2.8.2.	Modelo Lógico-Conceptual.....	23
2.8.3.	Modelo Navegación.....	23
2.8.4.	Modelo de Presentación.....	23
2.8.5.	Escenario Web.....	23
2.8.6.	Fases de UWE.....	24
2.8.6.1.	Análisis y requisitos.....	24
2.8.6.2.	Diseño Conceptual.....	25
2.8.6.3.	Diseño Navegacional.....	26
2.8.6.4.	Diseño de Presentación.....	27
2.9.	Herramientas.....	28
2.9.1.	Herramientas de Frontend.....	28
2.9.1.1.	Marcas de Hipertexto (HTML5).....	28
2.9.1.2.	CSS3.....	28
2.9.1.3.	Javascript.....	28

2.9.2.	Herramienta Backend.....	29
2.9.2.1.	PHP.....	29
2.9.2.2.	Gestor de base de datos MySQL.....	29
2.10.	Ingeniería de software.....	29
2.11.	Método de prueba de software.....	30
2.11.1.	Caja blanca.....	31
2.11.2.	Caja negra.....	32
2.12.	Calidad del software.....	33
2.12.1.	Norma ISO 9126.....	33
2.12.1.1.	Funcionalidad.....	34
2.12.1.2.	Confiabilidad.....	34
2.12.1.3.	Usabilidad.....	34
2.12.1.4.	Eficacia.....	35
2.12.1.5.	Mantenibilidad.....	35
2.12.1.6.	Portabilidad.....	35
2.13.	Prueba de seguridad.....	35
2.13.1.	Norma ISO 27001.....	35
2.13.1.1.	Encriptación.....	36
2.13.1.2.	Autenticación y autorización.....	36
2.14.	Estimación de costos.....	36
2.14.1.	Método COCOMO.....	37
2.14.2.	Características.....	37
2.14.3.	Niveles.....	38
2.14.4.	Modos de desarrollo.....	39

CAPÍTULO III.MARCO APLICATIVO

3.1.	Fase inicio.....	40
3.1.1.	Análisis de la situación actual.....	40
3.1.3.	Identificación de actores.....	41
3.1.4.	Modelado de negocio.....	43
3.1.4.1.	Modelo de casos de uso del negocio.....	43
3.1.5.	Ingeniería de requerimientos.....	45
3.1.5.1.	Requerimientos funcionales y no funcionales.....	45
3.2.	Fase Elaboración.....	46
3.2.1.	Modelo de caso de uso UWE.....	46
3.2.2.	Descripción de caso de uso.....	47
3.2.3.	Modelo de Clases.....	61
3.2.4.	Modelo Navegacional.....	62
3.2.5.	Modelo de diseño.....	63
3.2.6.	Modelo de Presentación.....	64

3.3.	Fase construcción.....	70
3.3.1.	Diseño de interfaz web.....	70
3.4.	Fase de Transición.....	76
3.4.1.	Prueba de Software.....	76
3.4.1.1.	Prueba de caja blanca.....	76
3.4.1.2.	Prueba de caja negra.....	78
3.5.	Pruebas de calidad.....	80
3.5.1.	Norma ISO 9126.....	80
3.5.1.1.	Funcionalidad.....	80
3.5.1.2.	Confiabilidad.....	83
3.5.1.3.	Usabilidad.....	84
3.5.1.4.	Eficacia.....	86
3.5.1.5.	Portabilidad.....	87
3.6.	Pruebas de seguridad	87
3.6.1.	Norma ISO 27001.....	87
3.6.1.1.	Encriptación.....	88
3.6.1.2.	Autenticación y autorización.....	88
3.7.	Estimación de Costos.....	89

CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES

4.1.	Conclusiones.....	93
4.2.	Recomendaciones.....	94

BIBLIOGRAFÍA.....	95
-------------------	----

ÍNDICE DE TABLAS Y FIGURAS

TABLAS

Tabla N° 2.1 Modelo COCOMO Básico.....	38
Tabla N° 3.2 Fases de UWE	40
Tabla N° 3.3: Descripción de Actores.....	42
Tabla N° 3.4 Descripción de Caso de Uso - Administrador de Usuario.....	49
Tabla N° 3.5 Descripción de Caso de Uso - Registro Técnico	51
Tabla N° 3.6 Descripción Caso de Uso - Registro CAEB	53
Tabla N° 3.7 Descripción Caso de Uso - Registro Unidad Industrial	55
Tabla N° 3.8 Descripción de Caso de Uso - Registro de Documento Ambiental... 56	
Tabla N° 3.9 Descripción Caso de Uso - Registro Sanción.....	58
Tabla N° 3.10 Descripción de Caso de Uso - Consulta Información	59
Tabla N° 3.11 Descripción Caso de Uso - Consulta Estado de la Unidad Industrial	60
Tabla N° 3.12 Prueba de caja negra	79
Tabla N° 3.13 Cálculo de Punto de Fusión.....	81
Tabla N° 3.14 Cálculo de Punto de Fusión.....	82
Tabla N° 3.15 Escala de Valores.....	85
Tabla N° 3.16 Evaluación de Preguntas para Calcular la Usabilidad	85
Tabla N° 3.17 Facilidad de Mantenimiento.....	86
Tabla N° 3.18 Conteo de cada Valor de Dominio	89
Tabla N° 3.19 Cálculo de Ajuste de Complejidad.....	90
Tabla N° 3.20 Métrica de Desarrollo de Software – Según herramientas de Desarrollo	91
Tabla N° 3.21 Coeficientes.....	92

FIGURAS

Figura N° 2.1 Ciclo de vida Del Proceso Unificado Ágil (AUP)	19
Figura N° 2.2 Fases del AUP	20
Figura N° 3.3 Modelo de UWE	24
Figura N° 2.4 Análisis de Casos de Uso.....	25
Figura N° 2.5 Diagrama de Clase.....	26
Figura N° 2.6 Modelo Navegacional	27
Figura N° 2.7 Diseño de Presentación	27
Figura N° 2.8 Gafico de Flujo	31
Figura N° 2.9 Norma ISO	34
Figura N° 3.10 Diagramas de Caso de Uso Del Negocio	44
Figura N° 3.11 Casos de Uso del Sistema	47
Figura N° 3.12 Caso de Uso - Administrador de Usuario	48
Figura N° 3.13 Caso de Uso - Registro Técnico.....	50
Figura N° 3.14 Caso de Uso - Registro de Clasificación de Actividades Económicas de Bolivia (CAEB)	52
Figura N° 3.15 Caso de Uso - Registro de Unidad Industrial	54
Figura N° 3.16 Registro Documento Ambiental	56
Figura N° 3.17 Caso de Uso Genera Reporte	57
Figura N° 3.18 Descripción Caso de Uso - Registra Sanción.....	57
Figura N° 3.19 Caso de Uso Consulta Información	58
Figura N° 3.20 Caso de Uso Consulta de Estado de Unidad Industrial.....	60
Figura N° 3.21 Diagrama de Clase.....	61
Figura N° 3.22 Modelo Navegacional	62
Figura N° 3.23 Modelo Entidad Relación.....	63
Figura N° 3.24 Modelo de Presentación - Inicio de Sesión al Sistema.....	64
Figura N° 3.25 Página Principal.....	65
Figura N° 3.26 Página de Búsqueda	66
Figura N° 3.27 Registro Ambiental Industrial.....	67
Figura N° 3.28 Reportes.....	68
Figura N° 3.29 RAI´s por Categoría.....	69

Figura N° 3.30 Pantalla de Bienvenida al Sistema	70
Figura N° 3.31 Autenticación al Sistema	71
Figura N° 3.32 Pantalla de Búsqueda	71
Figura N° 3.33 Pantalla de Datos Generales de Unidad Industrial	72
Figura N° 3.34 Pantalla de Ubicación Geográfica de las Unidades Industriales ...	72
Figura N° 3.35 Pantalla de Usuarios	73
Figura N° 3.36 Pantalla de Nuevo Usuario.....	73
Figura N° 3.37 Pantalla de Registro de CAEB	73
Figura N° 3.38 Pantalla de Registro de Unidades Industriales.....	74
Figura N° 3.39 Pantalla Principal de Reportes	74
Figura N° 3.40 Pantalla de Reporte de RAI's por Categoría	75
Figura N° 3.41 Pantalla de Reportes de Unidades Industriales por Distrito	75
Figura N° 3.42 Grafo del Sistema.....	76
Figura N° 3.43 Encriptación.....	88
Figura N° 3.44 Autenticación y Autorización	88

CAPÍTULO I. MARCO PRELIMINAR

1.8. Introducción.

Los sistemas de información son un conjunto de elementos interrelacionados con el propósito de prestar atención a las demandas de información de una organización, y proporcionar un mejor apoyo a la toma de decisiones y desarrollo de acciones. (EconLink, 2019)

Las tecnologías de la información y la comunicación (TIC's) facilitan la mejora de los procesos administrativos, que buscan reducir costes a través de un mejor aprovechamiento de los recursos disponibles, que permiten adelantar y agilizar los procesos de comunicación e información.

La Dirección de Gestión y Control Ambiental dependiente de la Secretaría Municipal de Servicios Municipales del Gobierno Autónomo Municipal de El Alto, realiza el control y monitoreo a las unidades industriales de forma semi-automatizada, el cual genera demora en la toma de decisiones.

El presente proyecto tiene por objeto desarrollar un Sistema de Información de Registro y Control de Documentos Ambientales. De forma que se tenga el registro de los documentos de esta manera obtener la información precisa de cada unidad industrial que presente la documentación respectiva, así brindar información relevante en tiempo real.

El sistema, se desarrollará utilizando la metodología AUP (Procesos Unificado Ágil) para la elaboración del sistema y UWE (UML-based Web Engineering) basada en UML para el desarrollo de aplicación web, así mismo utilizarán las herramientas MySQL como servidor de base de datos, lenguaje de programación PHP; bajo la estructura (MVC) del Modelo, Vista, Controlador.

1.9. Antecedentes.

1.9.1. Antecedentes de la Institución.

La Dirección de Gestión y Control Ambiental dependiente del Gobierno Autónomo Municipal de El Alto, se encuentra en la zona Ciudad Satélite avenida del Policía, Edificio ESSEC (Ex Embajada del Perú).

La Dirección de Gestión y Control Ambiental promueve el desarrollo sostenible con equidad y justicia social mediante acciones de regulación, control, monitoreo y evaluación de fuentes de contaminación ambiental generadas por las unidades industriales existentes en el Municipio de El Alto tomando en cuenta el cumplimiento de su documentación ambiental precautelando los derechos de la población en su conjunto.

1.9.1.1. Misión.

Forjar y fortalecer los criterios e instrumentos de gestión ambiental, de manera oportuna, real y eficaz en todos los sectores sociales e institucionales que aseguren la óptima protección, conservación y aprovechamiento de los recursos naturales, estructurando así una política ambiental incluyente en el marco del desarrollo sostenible.

1.9.1.2. Visión.

Un Municipio habitable y sano con mejor calidad de vida, y una ciudadanía comprometida con el medio ambiente, buscando un equilibrio entre el desarrollo productivo integral con equidad social.

1.9.2. Antecedentes Académicos.

Pensando la temática que trata el actual proyecto y al instante de realizar una exploración bibliográfica en nuestro medio se han detallado algunos trabajos afines, que se describen a continuación:

En primer lugar, se menciona El trabajo realizado por (Núñez 2008), que titula **“Simulación del Impacto Ambiental en el Departamento de La Paz”**, lo cual se basa en el software de simulación VESIM, lo que contribuye con la información sustentada de los daños ambientales y cómo afectan a los cambios climáticos en años venideros.

La relación con el presente proyecto es el uso de diferentes herramientas para el desarrollo y el diseño. Como lenguaje de programación se utiliza PHP y como lenguaje de Base de Datos MySQL.

Se describe también el trabajo realizado por (Cruz, Gallego, Gonzales 2009), titulado **“Sistema de Evaluación de Impacto Ambiental”**, el cual utiliza la metodología de lógica borrosa (FUZZY LOGIC) que ofrece reportes de información, de los resultados del impacto ambiental utilizando Lenguaje de Marcado Extensible (XML).

La diferencia que existe con el presente proyecto de grado es la descripción de los módulos que comprende su diseño, se utilizara AUP y UWE.

También se cita el trabajo realiza por: Pineda (2010), titulada **“Implementación del Sistema de Gestión Ambiental en la Empresa G.L. Ingenieros S.A.”**, que utilizando la norma ISO 14001, para el procedimiento de neutralización de enjuagues, de neutralización de PH y Floculación/Sedimentación.

Con relación al proyecto se diferencia por la sistematización de la información y documentación de los documentos ambientales a las unidades industriales de la ciudad de El Alto.

1.10. Planteamiento del problema.

1.10.1. Descripción del problema.

Los documentos ambientales de las unidades industriales que ingresan a la Dirección de Gestión y Control Ambiental, son los siguientes:

- Registro Ambiental Industrial (RAI)
- Instrumento de Regulación de Alcance Particular (IRAP)
 - Descripción de Proyecto (DP)
 - Manifiesto Ambiental Industrial (MAI)
 - Plan de Manejo Ambiental (PMA)
- Informe Ambiental Anual (IAA)
- Sancionatorios

1.10.2. **Problema principal.**

Realizado un análisis de los documentos ambientales, por lo cual se ha diseñado un diagrama de causa – efecto (diagrama de Ishikawa, ver ANEXO A) que ha permitido identificar la problemática a resolver, con base a este se plantea el siguiente problema:

El proceso de control y registro de los documentos ambientales de las unidades industriales existentes en el municipio de la ciudad de El Alto no es lo suficientemente adecuado debido a que se procesa de manera semi-automatizada lo cual genera demora en el procesamiento de los datos, emite información no confiable, ocasionando dificultad en la toma de decisiones.

1.10.3. **Problemas secundarios.**

- El registro semi-automatizado de los documentos ambientales no son confiables como requerimientos solicitados por los técnicos de la dirección de gestión y control ambiental.
- El control de vigencia y caducidad de los documentos ambientales es inadecuado debido a la gran cantidad de información que se registra de manera semi-automatizada.
- El control semi-manual de las sanciones de las diferentes unidades industriales lo cual provoca un monitoreo erróneo.
- La ubicación geográfica de las unidades industriales no es correcta, lo que dificulta la identificación de los predios en los monitoreos programados.
- La falta de información por parte del representante legal, del estado de sus documentos ambientales ocasiona que las industrias comentan infracciones administrativas.

1.11. **Objetivos.**

1.11.1. **Objetivo general.**

Desarrollar un Sistema de Información para El Registro y Control de Documentos Ambientales de las Unidades Industriales de la Ciudad de El Alto, que brinde información rápida, oportuna, fiable y que ayude a un adecuado control de documentos ambientales.

1.11.2. **Objetivos específicos.**

- Automatizar el procesamiento de los documentos ambientales para que los técnicos utilicen como requerimientos confiables.
- Sistematizar la información respecto a estado de vigencia o caducidad de los documentos ambientales, que son otorgadas a las unidades industriales.
- Automatizar el control de las sanciones para realizar la inspección correspondiente a la industria de forma correcta.
- Automatizar la información respecto a las coordenadas geográficas de las unidades industriales para un control eficiente.
- Implementar el acceso a cada unidad industrial para que pueda verificar el estado de sus documentos ambientales.
- Elaborar reportes de vigencia y caducidad de documentos ambientales de las industrias para realizar toma de decisiones.
- Elaborar un mapeo de la ubicación geográfica de las unidades industriales por categorías y distritos.

1.12. **Justificación.**

1.12.1. **Justificación Técnica.**

Se justifica técnicamente el presente proyecto, porque contribuye con una herramienta para un buen control y monitoreo de las unidades industriales, la dirección cuenta con equipos de computación para la elaboración e implementación del sistema y la optimización de solicitud de información de cada unidad industrial.

1.12.2. **Justificación Económica**

El proyecto se justifica económicamente porque se desarrollará utilizando herramientas de software libre lo cual no genera costos adicionales con relación a las licencias, el personal tiene acceso a internet, mediante un equipo de computación así también pretende, disminuir la cantidad de documentación física, menos consumo de papel, reduce el tiempo de búsqueda, optimizando el tiempo del personal técnico y legal.

1.12.3. **Justificación Social.**

El proyecto se justifica socialmente porque permitirá al personal de la dirección de gestión y control ambiental acceder a la información permitiendo un adecuado desarrollo de sus actividades laborales. Las unidades industriales se beneficiarán con la información oportuna del estado de sus documentos ambientales.

1.13. **Metodología y técnicas.**

Se utilizará el Proceso Unificado Ágil (AUP) este método describe de manera simple y fácil de entender aplicaciones de software utilizando conceptos que aún se mantienen en RUP, así también se utilizará la Ingeniería Web basada en UML, UML based Web Engineering (UWE).

1.13.1. **Metodología Proceso Unificado Ágil (AUP)**

El proceso unificado ágil (AUP) es una versión simplificada del Proceso Unificado de Rational (RUP), desarrollada por Scott Ambler. Describe una manera simple y fácil de entender la forma de desarrollar de software de aplicación de negocios usando técnicas y conceptos ágiles. (Perez Escarzo, 2016)

AUP consta de cuatro fases los cuales son:

- **Inicio:** El objetivo de esta fase es obtener una comprensión común cliente equipo de desarrollo del alcance del nuevo sistema y definir una o varias arquitecturas candidatas para el mismo.
- **Elaboración:** El objetivo es que el equipo de desarrollo profundice en la comprensión de los requisitos del sistema y en validar la arquitectura.
- **Construcción:** Durante la fase de construcción el sistema es desarrollado y probado al completo en el ambiente de desarrollo.
- **Transición:** el sistema se lleva a los entornos de preproducción donde se somete a pruebas de validación y aceptación y finalmente se despliega en los sistemas de producción.

1.13.2. Metodología UML based Web Engineering (UWE)

UWE es un proceso del desarrollo para aplicaciones Web enfocado sobre el diseño sistemático, la personalización y la generación semi-automática de escenarios que guíen el proceso de desarrollo de una aplicación Web. UWE describe una metodología de diseño sistemática, basada en las técnicas de UML, la notación de UML y los mecanismos de extensión de UML.

1.13.2.1. Las fases de la UWE

UWE cubre todo el ciclo de vida de este tipo de aplicación centrado además su atención en aplicaciones personalizadas o adaptativas.

Las fases o etapas a utilizar son:

1) Captura, análisis y especificación de requisitos: En simple palabras y básicamente, durante esta fase, se adquieren, reúnen y especifican las características funcionales y no funcionales que deberá cumplir la aplicación web.

Trata de diferente forma las necesidades de información, las necesidades de navegación, las necesidades de adaptación y las de interfaz de usuario, así como algunos requisitos adicionales. Centra el trabajo en el estudio de los casos de uso, la generación de los glosarios y el prototípico de la interfaz de usuario.

2) Diseño del sistema: Se basa en la especificación de requisitos producido por el análisis de los requerimientos (fase de análisis), el diseño define cómo estos requisitos se cumplirán, la estructura que debe darse a la aplicación web.

3) Codificación del software: Durante esta etapa se realizan las tareas que comúnmente se conocen como programación; que consiste, esencialmente, en llevar a código fuente, en el lenguaje de programación elegido, todo lo diseñado en la fase anterior.

4) Pruebas: Las pruebas se utilizan para asegurar el correcto funcionamiento de secciones de código.

5) La Instalación o Fase de Implementación: es el proceso por el cual los programas desarrollados son transferidos apropiadamente al computador destino, inicializados, y, eventualmente, configurados; todo ello con el propósito de ser ya utilizados por el usuario final.

1.13.3. **Técnicas.**

1.13.3.1. **Técnica de recopilación de datos.**

➤ **Entrevista.**

Se realizó entrevistas al personal de Archivo y al jefe de unidad de control y monitoreo dependiente de la Dirección de Gestión y Control Ambiental. Para recabar información para la elaboración del proyecto.

➤ **Cuestionario.**

Se realizó un cuestionario para el relevamiento de información al encargado de archivo, jefe de unidad y monitoreo ambiental, personal del área técnica y legal, para así ver la situación actual.

1.13.3.2. **Técnicas de seguridad.**

➤ **Uso del password.**

Para la autenticación de los usuarios que tienen el sistema (administrador, Director/Jefe de Unidad, Técnicos y Asesor Legal), se hará uso de SHA1 para la identificación de los mismos.

➤ **Encriptación de datos**

Es de gran importancia guardar las contraseñas encriptados mediante un algoritmo de encriptación para hacer imposible a alguien que acceda al sistema, para el presente proyecto se aplicará el SHA1.

1.14. Límites y alcances

1.14.1. **Límites**

➤ La implementación del Sistema se realizará en el área de Archivos de la Dirección de Gestión y Control Ambiental.

- No ofrecerá información de seguimiento, del trámite de registros de los documentos ambientales.
- Solo se hará la digitalización del informe de los documentos ambientales aprobados.
- El acceso al sistema será autorizado según el nivel de usuario que corresponda.

1.14.2. **Alcances.**

- Acceso a la información oportuna y fiable para el personal del área técnica y legal.
- Información instantánea del estado de los documentos ambientales de las unidades industriales.
- Clasificación de las unidades industriales por rubro.
- Clasificación de las unidades industrial por categoría.
- Clasificación de las unidades industriales por distrito.
- Datos estadísticos con relación a la cantidad de unidades industriales en el municipio de El Alto.
- Ubicación geográfica de las Unidades Industriales.

CAPÍTULO II. MARCO TEÓRICO

El marco teórico viene a ser la base principal para comprender todas las tareas relacionadas con las necesidades y condiciones a satisfacer para la realización del software.

2.15. Sistema.

Un sistema es una colección de componentes interrelacionados que trabajan conjuntamente para cumplir algún objetivo. (Sommerville I. , 2005, pág. 36)

Sistema es un conjunto ordenado de normas y procedimientos que regulan el funcionamiento de un grupo o colectividad.

2.16. Sistema de información.

Sistema de información se puede definir como: un conjunto formal de procesos que, operando sobre una colección de datos estructurada según las necesidades de la empresa, recopilan elaboran y distribuyen la información (o parte de ella) necesaria para las operaciones de dicha empresa y para las actividades de dirección y control correspondientes (toma de decisiones), que permitan desempeñar su actividad de acuerdo a su estrategia de negocio.(Ciencia, 2004, pág. 6)

Los sistemas de información son un conjunto de procesos que, operando sobre los datos estructurados de unión a las necesidades de la empresa, recopila, elabora y distribuyen selectivamente la información.

2.17. Ley del Medio Ambiente.

La Ley N° 1333 del 23 de abril de 1992, de medio ambiente tiene por objeto la protección y conservación del medio ambiente y los recursos naturales, regulando las acciones del hombre con relación a la naturaleza y promoviendo el desarrollo sostenible con la finalidad de mejorar la calidad de vida de la población. (Manufacturero, 1992, pág. 4)

2.17.1. Reglamento Ambiental para el Sector Industrial Manufacturero (RASIM).

El RASIM prioriza a esfuerzos orientados a la prevención y reducción de la contaminación ambiental originado por la generación de residuos: por otro lado, incorpora el empleo de incentivos para promover la productividad y competitividad del sector, siempre y cuando se tenga como objetivo la protección del medio ambiente. El manual de Aplicación del Reglamento Ambiental para el Sector Industrial Manufacturero es el documento de interpretación oficial del RASIM, dirigido a orientar y facilitar de manera práctica y detallada su correcta aplicación por parte de los funcionarios públicos. (Viceministerio de Industria, Reglamento Ambiental para el Sector Industrial Manufacturero, 2003, pág. 7)

El RASIM se aplica a toda actividad económica del sector industrial manufacturero, ya sea que se encuentre en su etapa de proyecto, implementación y/o en operación, que tiene como principal actividad transformar materias primas e insumos en productos terminados. Esta transformación puede ser en calidad y/o cantidad. (Viceministerio de Industria, 2003, pág. 7)

1) Transformación en calidad.

La transformación en calidad de una materia prima genera un producto totalmente diferente a la inicial. Ejemplo: la transformación de piel a una prenda de vestir. (Viceministerio de Industria, 2003, pág. 19)

2) Transformación en cantidad.

La transformación en cantidad implica que el producto resultante, solo cambia en alguna de sus características físicas. Ejemplo: envasa azúcar contenida en bolsas de 50 kilogramos y la fracciona en bolsas de 5 kilogramos. (Viceministerio de Industria, 2003, pág. 19)

2.17.2. Categorías de las industrias

El RASIM clasifica a las industrias según su riesgo de contaminación en tres categorías las cuales son:

- **Categoría 1 y 2.-** Industrias con mayor riesgo de contaminación.
- **Categoría 3.-** Industrias con riesgo medio de contaminación.
- **Categoría 4.-** Industrias con poco significado de contaminación.

Las categorías 1,2 y 3 tienen que cumplir con todo el RASIM, la categoría 4 está exenta de presentar varios documentos ambientales, pero los principios básicos y el resto de disposiciones del RASIM deben ser cumplidas por todas las industrias.(RASIM, 2004, pág. 5)

2.17.3. Instrumentos de Regulación De Alcance Particular (IRAP).

Los Instrumentos de Regulación de Alcance Particular son documentos (registros, estudios, planes, etc.) que tienen por objetivo permitir que las industrias evalúen sus problemas ambientales y planifiquen las acciones medidas para prevenir y mitigar los impactos ambientales que pueden estar generando, en el marco de lo que establece el reglamento. (Viceministerio de Industria, 2003, pág. 29)

- **Regulación de Alcance Particular.**

Son documentos ambientales que deben elaborar y presentar las industrias. Dependiendo de su categoría en ellos se señala como utilizar de mejor manera el agua, la energía y las materias primas, así protección al medio ambiente y las personas.(RASIM, 2004, pág. 8)

2.17.4. Registro Ambiental Industrial (RAI).

Se crea el Registro Ambiental Industrial (RAI) como instrumento de regulación de alcance particular para el registro y la categorización de las unidades industriales del sector manufacturero. (Ramirez, pág. 12)

- **Registro.** - toda unidad industrial en proyecto o en operación deberá registrarse en la instancia ambiental del gobierno municipal (IAGM) donde se proyecte localizar o localice su actividad productiva, mediante el formulario de Registro Ambiental Industrial (RAI).(Ramirez, pág. 12)
 - a) La unidad industrial en proyecto deberá registrarse antes de iniciar cualquier actividad física de instalación.

- b)** La unidad industrial en operación deberá registrarse en el plazo máximo de dos años a partir de la puesta en vigencia del presente reglamento.
- **Actualización.** - la industria debe informar por escrito y con carácter previo a la IAGM para la actualización de su Registro Ambiental Industrial, en los siguientes casos: (Ramirez, pág. 13)
Inicio de las operaciones de una industria en proyecto, cambio de razón social, cambio de representante legal, cierre de su actividad industrial.
- **Modificación.** - la industria debe solicitar a la IAGM la modificación de su registro, en los siguientes casos:(Ramirez, pág. 14)
 - a)** Diversificación del rubro de la producción que implique el cambio de subclase, será considerada como una industria en proyecto.
 - b)** Ampliación de la capacidad instalada de una industria en operación, se considera la nueva capacidad instalada para establecer sus nuevas obligaciones.
- **Traslado.** - en caso de traslado de una industria en operación deberá cumplir con el cierre de la actividad según lo establecido, y deberá registrarse nuevamente.(Ramirez, pág. 14)
- **Vigencia y renovación.** - el RAI de una unidad industrial tendrá una vigencia de cinco (5) años a partir de su registro inicial o renovación por modificación. Con antelación de treinta (30) días a su vencimiento el representante legal deberá renovar su RAI.(Ramirez, pág. 14)

2.17.5. Estudio de Evaluación Impacto Ambiental y Plan de Manejo Ambiental (EEIA - PMA)

Se entiende por estudio/evaluación de impacto ambiental (EIA) al conjunto de procedimientos administrativos, estudios y sistemas técnicos que permiten estimar los efectos que implementación una determinada industria.(Ramirez, pág. 14)

La industria en proyecto de categorías 1 y 2, no podrá iniciar actividad física alguna de instalación sin su respectiva Declaración de Impacto Ambiental (DIA), que se constituye en su licencia ambiental. La DIA aprueba el EEIA, autoriza la implementación del proyecto y acepta el compromiso del representante legal de

ejecutar su Plan de Manejo Ambiental (PMA) para los primeros 5 años a partir de su otorgación. (Ramirez, págs. 14,15)

Un EEIA contiene, datos del proyecto, resumen, descripción del proyecto, línea base o estado inicial del medio ambiente, identificación y evaluación de impactos, procedimientos para la consulta pública. (RASIM, 2004, pág. 11)

Un PMA contiene, datos generales, plan de prevención y mitigación, plan de aplicación y seguimiento ambiental, análisis de riesgos.(RASIM, 2004, pág. 11)

2.17.6. Descripción de Proyecto y Plan de Manejo Ambiental (DP - PMA)

Con el objeto de documentar la planificación para el cumplimiento de las disposiciones, la industria en proyecto de categoría 3, deberá elaborar una descripción del proyecto y un plan de manejo ambiental (PMA), las industrias de categoría 3, no podrá iniciar actividad física alguna de instalación sin el certificado de aprobación de la descripción del proyecto y PMA. (Ramirez, pág. 18)

La instancia ambiental del gobierno municipal (IAGM) revisa la DP – PMA en 15 días verificando que cumplan con los contenidos descritos. Si no se cumplen se solicitará aclaraciones por una solo vez. Si la IAGM incumple los plazos previstos, la industria podrá acceder al silencio administrativo positivo y dar por aprobado sus documentos enviando una carta notariada.(RASIM, 2004, pág. 13)

2.17.7. Manifiesto Ambiental Industrial, Plan de Manejo Ambiental (MAI - PMA)

Las industrias en operación de categorías 1,2 y categoría 3, deberán elaborar un manifiesto ambiental industrial (MAI) y un plan de manejo ambiental (PMA), en un plazo no mayor a 60 días a partir de la puesta en vigencia. Los plazos de presentación del MAI y PMA, establecidos en el cronograma, no podrán exceder los tres (3) años. Cumplido el plazo de tres años las industrias que no presentaron su MAI y PMA estarán obligadas a realizar una auditoría ambiental y presentar un PMA para regularizar su funcionamiento.(Ramirez, pág. 20)

2.17.8. Vigencia del Plan de Manifiesto Ambiental (PMA).

El plan de manejo ambiental (PMA) tiene vigencia de cinco (5) años, al cabo de los cuales deberá actualizarse. La declaratoria de impacto ambiental (DIA), declaratoria de adecuación ambiental (DAA) certificado estarán vigentes en tanto este vigente el PMA aprobado.(Ramirez, pág. 23)

Por una sola vez se presenta el EEIA, DP y MAI. Cada cinco (5) años el PMA debe actualizarse 60 días antes de cumplir los 5 años de vigencia. En el caso de la DIA, la DAA y el CA se pueden suspender cuando: no se hubiese actualizado el PMA, si existiera diversificación y/o ampliación y no se ha informado la modificación de su RAI, incumplimiento de las medidas previstas en el PMA aprobado, si se comprueban impactos no identificados con el PMA, como resultado de una sanción.(RASIM, 2004, pág. 14)

2.17.9. Informe Ambiental Industrial (IIA).

Todas las industrias de categoría 1,2 y categoría 3 que tengan su PMA aprobado deberán elaborar y presentar el IIA.El IIA se presenta para realizar el seguimiento de las medidas de prevención, mitigación y control de los impactos negativos planificados en el PMA.(RASIM, 2004, pág. 15)

2.18. Métodos, Técnicas e Instrumentos.

2.18.1. Observación.

La observación científica es aquella que utiliza hipótesis expresa y manifiesta, a pesar de que se puedan obtener observaciones científicas por azar o no conexas con objetivos de investigación. (Metodo de Observacion, pág. 2)

El método de observación es un elemento fundamental del proceso de investigación; para ello se apoya en la investigación para obtener un número de datos.

2.18.2. Cuestionario.

Un cuestionario es un formato estructurado de recogida de información a través de la cumplimentación de una serie predeterminada de preguntas. (Lorenzo, 2012, pág. 3)

El método de cuestionario es un sistema de preguntas ordenadas con sentido lógico y psicológico, un plan de recolección de datos.

El cuestionario es el instrumento del método de encuesta para obtener la información pertinente que permite contrastar el modelo de análisis. (Lopez Roldan & Fachelli, 2015, pág. 23)

Los cuestionarios son un documento formado por un conjunto de preguntas que deben estar escritas de manera coherente, y organizadas.

2.18.3. Entrevista.

La entrevista se define como una conversación que se propone con un fin determinado distinto al simple hecho de conversar. (Diaz Bravo, Torruco Garcia, Martinez Hernandez, & Varela Ruiz, 2013, pág. 2)

Las entrevistas, son interpersonal concretada entre el investigador y el sujeto de estudio a fin de obtener.

El proceso administrativo es el conjunto de fases o etapas sucesivas a través de las cuales se efectúa la administración misma que se interrelacionan y forman un proceso integral.

2.19. 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 que se transforman en un sistema funcional. (Pressman, 2010, pág. 102)

La principal tarea que cumple la Ingeniería de Requerimientos es poder recoger las necesidades de los clientes que ayude a resolver algún problema en el proyecto.

➤ **Requisitos Funcionales.**

Son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que este debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares. En algunos casos, los requerimientos funcionales del sistema también pueden declarar explícitamente lo que el sistema no debe hacer. (Sommerville I. , 2005, pág. 123)

Los requerimientos funcionales son aquellos que proporcionan funcionalidad al sistema.

➤ **Requisitos No Funcionales.**

Son restricciones de los servicios o funciones ofrecidos por el sistema. Incluye restricciones de tiempo, sobre el proceso de desarrollo y estándares. Los requerimientos no funcionales a menudo se aplican al sistema en su totalidad. (Sommerville I. , 2005, págs. 123-124)

Son aquellos requerimientos que no proporcionan funcionalidad al sistema.

La ingeniería de requerimientos se utiliza para definir todas las actividades involucradas en el descubrimiento, documentación y mantenimiento de los requerimientos para un producto de software determinado, donde es muy importante tomar en cuenta que el aporte de ingeniería de requerimientos vendrá a ayudar a determinar la viabilidad de llevar a cabo el software.

2.20. **Ingeniería de software.**

La ingeniería de software es una disciplina o área de la informática que ofrece métodos y técnicas para desarrollar y mantener software de calidad que resuelve problemas de todo tipo. (Carrera Universitaria, 2019)

La ingeniería de 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 se utiliza.

La ingeniería de software incluye procesos, métodos y herramientas que permiten elaborar a tiempo y con calidad sistemas complejos basados en computadoras. (Pressman S. R., 2010, pág. 21)

2.21. Proceso Unificado Ágil (AUP).

El Proceso Unificado Ágil de Scott Ambler o Agile Unified Process (AUP) es una versión simplificada del Proceso Unificado de Rational (RUP). Este describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen validas en RUP. (Aurelio, 2017)

El proceso unificado ágil nos describe de una manera más simple y fácil de entender el desarrollo del software.

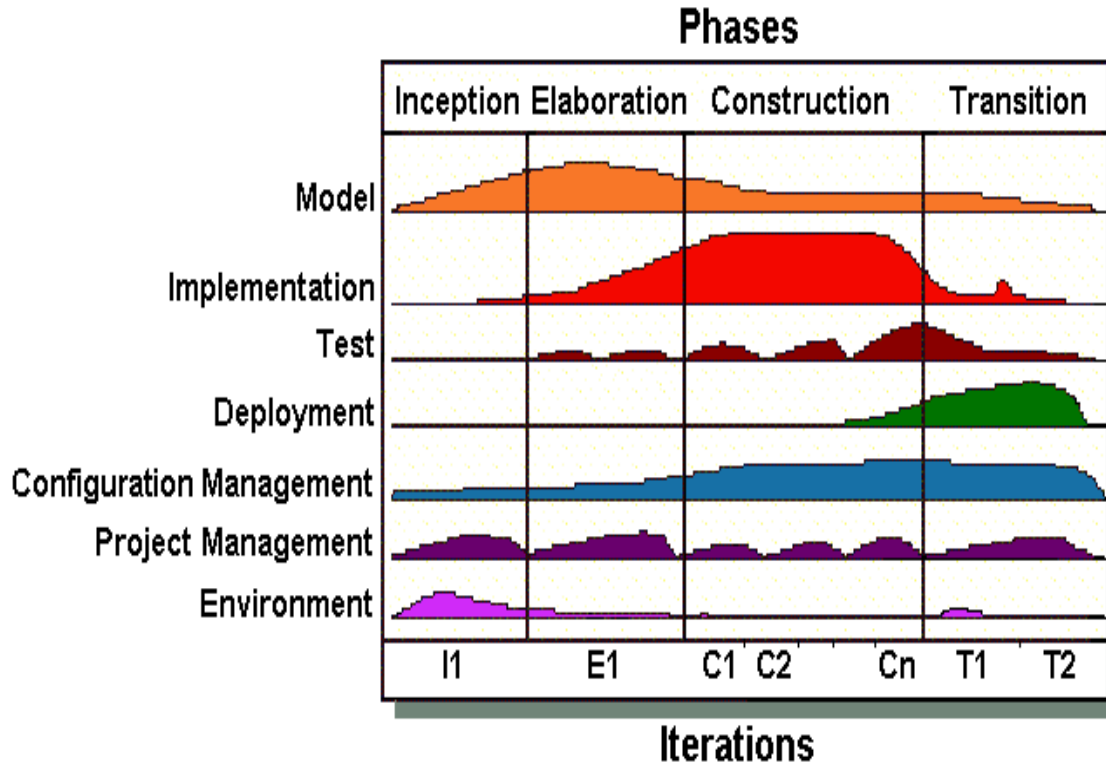
AUP brinda un revestimiento en serie (ejemplo, una secuencia lineal de actividades de ingeniería de software) que permite que el equipo visualice el flujo general del proceso de un proyecto de software. Sin embargo, dentro de cada actividad, el equipo repite con objeto de alcanzar la agilidad y entregar tan rápido como sea posible incrementos de software significativos a los usuarios finales. (Pressman, 2010, pág. 75)

El proceso unificado ágil AUP es un marco de trabajo conceptual de la ingeniería de software que suscita iteraciones en el desarrollo a lo largo de todo el ciclo de vida del proyecto.

2.21.1. Ciclo de vida del Proceso Unificado Ágil (AUP).

Al igual que en RUP, el Proceso Unificado Ágil establece cuatro fases que pasan de manera sucesiva y acaban con hitos.

Figura N° 2.1Ciclo de vida Del Proceso Unificado Ágil (AUP)



Fuente: (https://www.ecured.cu/Agile_Unified_Process)

2.21.2. Fases de Proceso Unificado Ágil (AUP).

Al igual que en RUP, en AUP establecen cuatro fases que transcurren de manera consecutiva y que acaban con hitos claros.

2.21.2.1. Inicio (Concepción).

El objetivo de esta fase es obtener una comprensión común cliente equipo de desarrollo del alcance del nuevo sistema y definir una o varias arquitecturas candidatas para el mismo. (Cordero L., pág. 4)

En la fase de inicio especificamos criterio del proyecto, definimos los requisitos estimamos los recursos necesarios también definimos los documentos del proyecto.

2.21.2.2. **Elaboración.**

El objetivo es que el equipo de desarrollo profundice en la comprensión de los requisitos del sistema y en validar la arquitectura. (Camacho, 2013)

En esta fase realizamos la planificación del proyecto, modelo de procesos, y una arquitectura básica.

2.21.2.3. **Construcción.**

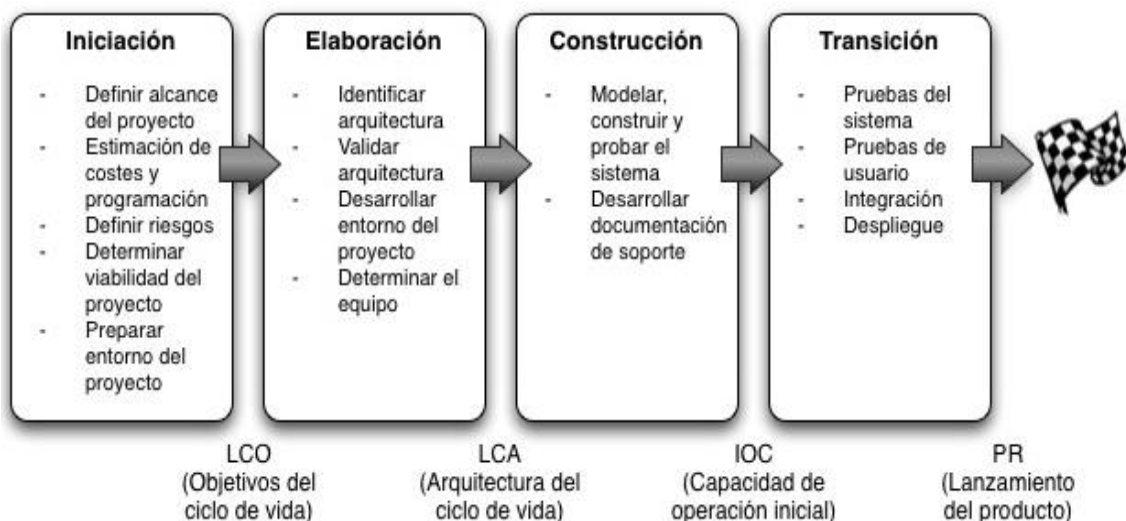
Durante la fase de construcción el sistema es desarrollado y probado al completo en el ambiente de desarrollo. (Camacho, 2013)

En esta fase realizamos el análisis, diseño, implementación y pruebas del proyecto.

2.21.2.4. **Transición.**

El sistema se lleva a los entornos de preproducción donde se somete a pruebas de validación y aceptación y finalmente se despliega en los sistemas de producción. (Camacho, 2013)

Figura N° 2.2 Fases Del AUP



Fuente: (Torrecilla, 2012)

2.21.3. Disciplinas de AUP.

Las actividades que el equipo desarrollo se debe realizar para construir, validar y entregar un software que satisfaga las necesidades de los involucrados.

➤ **Modelo.**

El objetivo es entender la organización, el problema de dominio que se aborda en el proyecto.

➤ **Implementación.**

El objetivo de esta disciplina es transformar su modelo en código ejecutable y realizar un nivel básico de las pruebas.

➤ **Prueba.**

Es realizar una valoración objetiva para garantizar la calidad y comprobando que se cumplan los requisitos.

➤ **Despliegue.**

El objetivo de esta disciplina es la presentación y ejecución del sistema y que el mismo está a disposición de los usuarios finales.

➤ **Gestión de configuración.**

Su objetivo es la gestión de acceso a herramientas de su proyecto.

➤ **Gestión de proyecto.**

Es dirigir actividades que se lleva a cabo en el proyecto incluye la gestión de riesgos, coordinar con el personal para asegurarse de que se entrega a tiempo y dentro del presupuesto.

➤ **Entorno.**

El objetivo es apoyar en la orientación y herramientas estén disponibles para el equipo según sea necesario.

Los equipos del AUP echan típicamente lanzamientos del progreso en el final de cada iteración. El primero puede tomar más tiempo que el segundo; eso se debe que en primera instancia hay más retraso a resolver que en las instancias futuras, lo que propone AUP es que a medida que el proyecto avanza las iteraciones se vuelven más cortas. (Camacho, 2013)

2.21.4. Principios de AUP.

AUP se basa en los siguientes principios.

➤ **El personal sabe lo que está haciendo.**

La gente no va a leer detallado el proceso de documentación, pero algunos desean una orientación de alto nivel y/o una información de vez en cuando.

➤ **Simplicidad.**

Se describe usando poca cantidad de páginas.

➤ **Agilidad.**

Se adjunta a los valores y a los principios de la Agile Alianza.

➤ **Herramienta de la independencia.**

Se puede utilizar cualquier conjunto de herramientas, pero se recomienda herramientas simples.

2.22. UWE (UML – Base Web Engineering).

UWE cubre todo el ciclo de vida de este tipo de aplicaciones, centrado además su atención en aplicaciones personalizadas. Esta metodología distingue entre la tarea de requisitos, definir y validar los requisitos. (Koch & Escalona, 2002)

UWE es desarrollado para aplicación web enfocado en el diseño sistemático, basado en UML.

UWE es un proceso del desarrollo para aplicaciones Web enfocado sobre el diseño sistemático, la personalización y la generación semiautomática de escenarios que guíen el proceso de desarrollo de una aplicación Web. Es una herramienta que nos permitirá modelar aplicaciones web, utilizada en la ingeniería web, presentado especialmente atención en sistematización y personalización. UWE es una propuesta basada en el proceso unificado y UML, pero adaptados a la web. (Galiano, 2012)

UWE es una propuesta basada en el proceso unificado y UML, pero orientado a la web.

2.22.1. Actividades de modelado de UWE.

Las actividades base de modelado de UWE son el análisis de requerimientos, el modelo conceptual, el modelo navegacional y el modelo de presentación. A estos modelos se pueden sumar otros modelos como son el modelo de interacción y la visualización de escenarios web. (Galiano, 2012)

Modelo que propone UWE está compuesto por etapas o sub-modelos

- Modelo de Casos de uso
- Modelo de Contenidos
- Modelo de Navegación
- Modelo de Proceso
- Modelo de Presentación

2.22.2. Modelo Lógico-Conceptual.

La construcción de este modelo lógico-conceptual se debe llevar a cabo de acuerdo con los casos de uso que se define en la especificación de requerimientos.

2.22.3. Modelo Navegación.

El modelo de navegación especifica los objetos que serán visitados por los usuarios a través de la aplicación.

2.22.4. Modelo de Presentación.

Los modelos de presentación son una presentación esquemática de los objetos visibles al usuario.

2.22.5. Escenario Web.

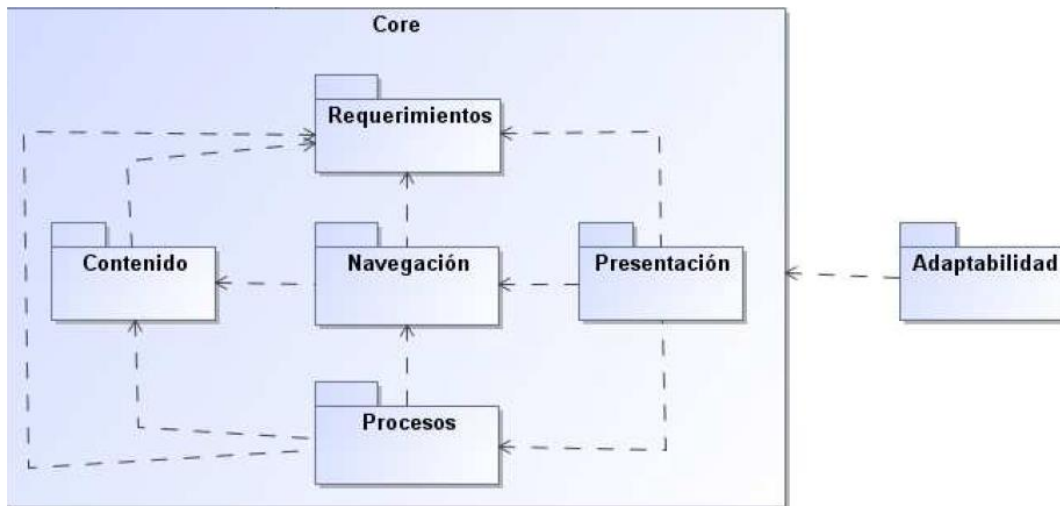
El escenario web permite detallar del modelo de navegación especifica los eventos de las situaciones y definen condiciones e incluye las acciones que son realizadas.

2.22.6. Fases de UWE.

UWE cubre todo el ciclo de vida de este tipo de aplicaciones centrado además su atención en aplicaciones personalizadas o adaptativas. (Galiano, 2012)

UWE provee diferentes modelos que permiten describir una aplicación web desde varios puntos de vista abstractos, en este sentido, los actores se utilizan para modelar los usuarios de la aplicación web.

Figura N° 3.3 Modelo de UWE



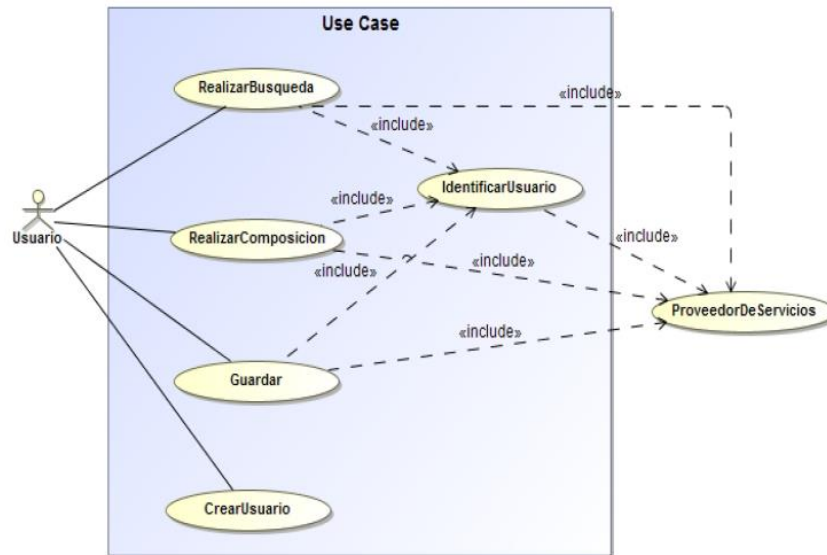
Fuente: (Narváz, Baldeon, Hinojosa, Matinez, 2008)

2.22.6.1. Análisis y requisitos.

Durante esta fase, se adquieren, reúnen y especifican las características funcionales y no funcionales que deberá cumplir la aplicación web. (Galiano, 2012)

Realizamos la identificación de los requisitos, y UWE se especifican mediante el modelo de requerimientos, que involucran a los casos de uso con UML.

Figura N° 2.4 Análisis de Casos de Uso



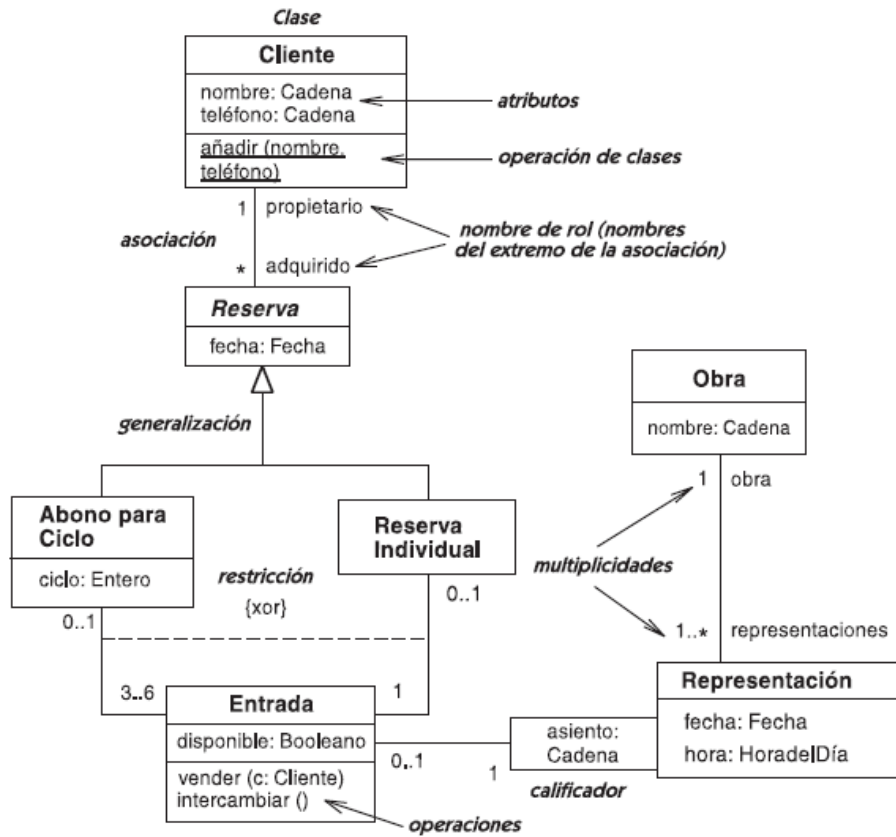
Fuente: (Nieves, Ucan, Menéndez, 2014)

2.22.6.2. Diseño Conceptual.

El objetivo del modelo de contenido es proporcionar una especificación visual de la información en el dominio relevante para la aplicación web. Se presenta el diagrama de clases para el modelo de contenidos. (Nieves Guerrero, Ucan Pech, & Menendez Dominguez, 2014, pág. 4)

El diseño conceptual comprende el modelo de dominio que al igual que los casos debe cumplir las funcionalidades requeridas por el sistema a desarrollar.

Figura N° 2.5 Diagrama de Clase



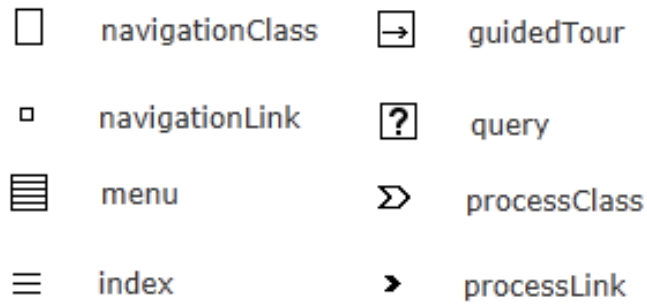
Fuente: (Booch, Jacobson, Rumbaugh)

2.22.6.3. Diseño Navegacional.

En una aplicación para la web es útil saber cómo están enlazadas las páginas. Ello significa que se requiere un diagrama de navegación con nodos y enlaces. (Nieves Guerrero, Ucan Pech, & Menendez Dominguez, 2014)

En el diseño navegacional es necesario conocer la relación y enlaces entre las páginas web, es por eso que describimos a través de diagrama de navegación del sistema.

Figura N° 2.6 Modelo Navegacional















Fuente: (Nieves, Ucan, Menendez, 2014)

2.22.6.4. Diseño de Presentación

El modelo de presentación ofrece una visión abstracta de la interfaz de usuario de una aplicación web. Se basa en el modelo de navegación y en los aspectos concretos de la interfaz de usuario. (Nieves Guerrero, Ucan Pech, & Menendez Dominguez, 2014, pág. 5)

El diseño de presentación se basa en el modelo de estructura de navegacional, pueden interpretarse con interfaces del sistema web, para ello se tiene estereotipos o iconos que ayudan al diseño de diagrama de presentación.

Figura N° 2.7 Diseño de Presentación

Nombres de Estereotipos y sus Iconos	
 grupo de presentación	 página de presentación
 texto	 entrada de texto
 ancla	 fileUpload
 botón	 imagen
 formulario	 componente de cliente
 alternativas de presentación	 selección

Fuente: (Informatics, 2016)

2.23. Herramientas.

2.23.1. Herramientas de Frontend.

El frontend es desarrollar la interfaz gráfica de usuario, buscando una experiencia de uso bien valorada por el usuario final, siendo en algunos casos necesario hacer investigación, estudio y pruebas para llegar a este fin. (Valdivia Caballero, 2016, pág. 5)

El frontend usa tecnologías y lenguajes de estilo del lado del usuario para su estructuración, el maquetado, las animaciones del sistema web.

2.23.1.1. Marcas de Hipertexto (HTML5).

HTML5 crea mejores prácticas de codificación. La separación de código de presentación y de estructura es más adecuada en el desarrollo de programación actual.

HTML5 es un lenguaje que se compone por elementos que permiten definir la estructura del documento. Estos elementos son los que nos posibilitan determinar cómo estará armada la página y sus secciones. (Damian, 2011, pág. 66)

2.23.1.2. CSS3.

CSS3 es un lenguaje de hojas de estilo creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas. (Eguiluz Perez, 2008, pág. 5)

También CSS permite hacer webs más elaboradas y más dinámicas, con mayor separación entre estilos y contenidos.

2.23.1.3. Javascript.

Los programas que se realiza, no necesitaran ser compiladas, los lenguajes de Scripting son lenguajes interpretados. Esto en la práctica significa que cuando trabajas con JavaScript escribimos nuestro programa y podremos ejecutarlo de forma directa. (Rives, 2011, pág. 15)

Javascript es un lenguaje con muchas posibilidades, utilizado para crear pequeños programas que luego son insertados en una página web y en programas más grandes, orientados a objetos mucho más complejos.

2.23.2. Herramienta Backend.

En el contexto del desarrollo de aplicaciones están implicados las actividades realizadas del lado del servidor; es decir, las tareas de base de datos y los servidores de aplicaciones que el usuario no puede visualizar en el explorador de internet. (Valdivia Caballero, 2016, pág. 5)

El Backend son datos de un software que no es directamente accesible para los usuarios.

2.23.2.1. PHP.

PHP es un lenguaje diseñado para crear contenido HTML. PHP puede ser ejecutado de tres formas: en un servidor web, a través de la línea de comandos, o mediante un cliente GUI.(Armando, 2018, pág. 7)

El lenguaje PHP es un lenguaje de programación de estilo clásico, con esto quiero decir que es un lenguaje de programación con variables, sentencias condicionales, bucles, funciones.

2.23.2.2. Gestor de base de datos MySQL.

MySQL es un sistema de gestión de base de datos que ofrece los mecanismos para añadir, acceder y procesar los distintos datos almacenados en una base de datos.

2.24. Ingeniería de software.

La ingeniería de 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 se utiliza. (Sommerville I. , 2005, pág. 23)

La ingeniería de software es una aplicación de enfoque sistemático, disciplinado y cuantificable al desarrollo, operacional y mantenimiento de software.

El software se ha convertido en el elemento clave de la evolución de los sistemas y productos basados en computadoras, así como en una de las tecnologías más importantes en el ámbito mundial, la ingeniería de software es proporcionar un marco general para construir software con una calidad mucho mayor. (Pressman, 2010, pág. 21)

La ingeniería de software es un enfoque sistémico de progresos, es la rama de la ingeniería que emplea los principios de ciencia de la computación, también requiere un conjunto de conceptos, para poder realizar el ciclo de vida del software.

2.25. Método de prueba de software.

El desarrollo de software implica la realización de una serie de actividades predisuestas a incorporar errores.

Según (Pressman, 2005). La meta de probar es encontrar errores, y una buena prueba es aquella que tiene una alta probabilidad de encontrar uno. Por tanto, un sistema basado en computadora o un producto debe diseñarse e implementarse teniendo en mente la “comprobabilidad”. Al mismo tiempo, las pruebas en si misma deben mostrar un conjunto de características que logren la meta de encontrar la mayor cantidad de errores con el mismo esfuerzo.

Hay una de dos maneras de probar cualquier producto construido la primera es si se conoce la función específica para la que se diseñó el producto, se aplican pruebas, que demuestren que cada función es plenamente operacional, mientras se buscan los errores en cada función; la segunda si se conoce el funcionamiento interno del producto, se aplican pruebas para asegurarse de que “todas las piezas encajan”; es decir, que las operaciones internas se realizan de acuerdo a las especificaciones, y que se han probado todos los componentes internos de manera adecuada al primer enfoque de prueba se le denomina prueba de caja negra; al segundo, prueba de caja blanca.

El software se prueba para descubrir errores que se cometieron de manera inadvertida conforme se diseñó y construyó. (Pressman, 2010, pág. 383)

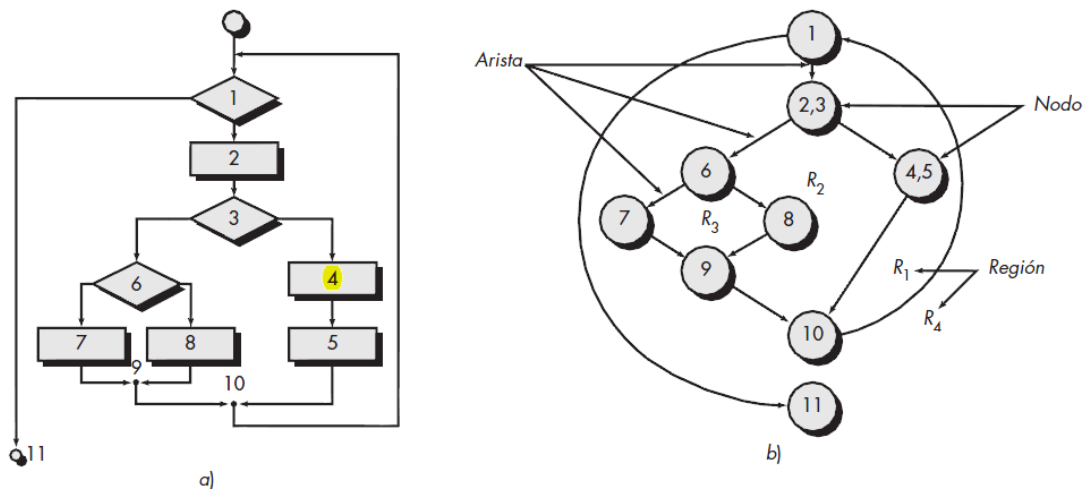
2.25.1. Caja blanca.

Este tipo de prueba es un diseño de casos de prueba que usa la estructura de control del diseño procedimental para derivar los casos de pruebas.

Según (Pressman, 2010, pág. 414) la prueba de caja blanca, en ocasiones llamada pruebas de caja de vidrio, es una filosofía de diseño de casos de pruebas que usan las estructuras de control descrita como parte del diseño a nivel de componentes para derivar casos de pruebas. Al usar los métodos de prueba de caja blanca, puede derivar casos de prueba que:

- 1) Garanticen que todas las rutas independientes dentro de un módulo se revisaron al menos una vez.
- 2) Revisen todas las decisiones lógicas en sus lados verdadero y falso.
- 3) Ejecuten todos los bucles en sus fronteras y dentro de sus fronteras y operativas.
- 4) Revisen estructuras de datos internas para garantizar su validez.

Figura N° 2.8 Gafico de Flujo



Fuente: (Pressman S. R., 2010)

2.25.2. Caja negra.

Las pruebas de caja negra se realizan sobre la interfaz del software, no tomando en cuenta el comportamiento interno y la estructura del programa.

Las pruebas de caja negra, también llamadas pruebas de comportamiento, se enfocan en los requerimientos funcionales del software; es decir, las técnicas de pruebas de caja negra le permiten derivar conjuntos de condiciones de entradas que revisaran por completo todos los requerimientos funcionales para un programa. (Pressman, 2010, pág. 423)

Las pruebas de diseño responden las siguientes preguntas:

- ¿Cómo se prueba la validez funcional?
- ¿Cómo se prueban el comportamiento y el rendimiento del sistema?
- ¿Qué clases de entrada harán buenos casos de prueba?
- ¿El sistema es particularmente sensible a ciertos valores de entrada?
- ¿Cómo se aíslan las fronteras de una clase de datos?
- ¿Qué tasas y volumen de datos puede tolerar el sistema?
- ¿Qué efectos tendrán sobre la operación del sistema algunas combinaciones específicas de datos?

En primera instancia se debe comprender los objetos modelados entre el software y la relación entre ellos, a partir de ello se crea una gráfica de objetos importantes y sus relaciones.

Para esto existen algunos métodos de comprobación los cuales son:

- Método de prueba basado en grafos: en este método se debe entender los objetos (objetos de datos, objetos de programa tales como módulos o colecciones de sentencias del lenguaje de programación) que se modelan en el software y las relaciones que conectan a estos objetos. Una vez que se ha llevado a cabo esto, el siguiente paso es definir una serie de pruebas que verifiquen que todos los objetos tienen entre ellos las relaciones esperadas.
- Partición equivalente: el objetivo de partición equivalente es reducir el posible conjunto de casos de prueba en uno más pequeño, un conjunto manejable que evalúe bien el software. Se toma un riesgo porque se escoge no probar

todo. Así que se necesita tener mucho cuidado al escoger las clases.
(Pressman, 2005)

2.26. Calidad del software.

Son un conjunto de reglas generadas para la creación de productos de software con calidad, que si se siguen correctamente pueden garantizar que los proyectos darán como resultado la satisfacción del cliente.

La medición es un elemento clave en cualquier proceso de ingeniería, las medidas se emplean para comprender mejor los atributos de los modelos que se crean y evaluar la calidad de los productos de la ingeniería o de los sistemas que se construyen. (Pressman, 2010, pág. 484)

2.26.1. Norma ISO 9126.

La norma ISO 9126 permite especificar y evaluar la calidad del software desde diferentes criterios asociados con adquisición, requerimiento, desarrollo, uso, evaluación, soporte, mantenimiento, aseguramiento de la calidad y auditoria de software. (Borbon Ardila, 2013)

La norma ISO9126 evalúa el software debido a las necesidades de un modelo que expresa la calidad del software.

Para la medición de calidad del software se seguirá el estándar ISO 9126, basado en seis características principales: confiabilidad, funcionalidad, facilidad de uso, eficiencia, mantenibilidad y portabilidad.

Figura N° 2.9 Norma ISO



Fuente: (Largo, Mazo, 2005)

2.26.1.1. **Funcionalidad.**

La funcionalidad es la capacidad del software de cumplir y proveer las funciones para satisfacer las necesidades explícitas e implícitas cuando es utilizado en condiciones específicas. (Largo Garcia & Marin Mazo, 2005, pág. 15)

Para este propósito se establece los siguientes atributos: adecuación, exactitud, interoperabilidad, conformidad, seguridad.

La funcionalidad es la existencia de un conjunto de funciones y propiedades específicas.

2.26.1.2. **Confiabilidad.**

Según (Informatics, 2016, pág. 2) aquí se agrupan un conjunto de atributos que se refieren a la capacidad del software de mantener su nivel de ejecución bajo condiciones normales en un periodo establecido. La sub-características que el estándar sugiere son: Nivel de madurez, tolerancia en fallas, Recuperación.

2.26.1.3. **Usabilidad.**

Consiste de un conjunto de atributos que permiten evaluar el esfuerzo necesario que deberá invertir el usuario para utilizar en sistema: comprensibilidad, facilidad de aprender, operabilidad. (Informatics, 2016, pág. 2)

2.26.1.4. **Eficacia.**

El grado en que el software emplea en forma óptima los recursos de sistemas, como lo indican los siguientes subatributos; comportamiento en el tiempo, comportamiento de los recursos. (Pressman, 2010, pág. 488)

La eficacia se desempeña de acuerdo al número de recursos utilizados según las condiciones planteadas.

2.26.1.5. **Mantenibilidad.**

La facilidad con la que repara el software de acuerdo con los siguientes subatributos: facilidad de análisis, facilidad de cambio, estabilidad y facilidad de prueba. (Pressman, 2010, pág. 488)

La mantenibilidad del software tiene la cualidad de ser modificado, mejoras al sistema.

2.26.1.6. **Portabilidad.**

La facilidad con que se lleva el software de un entorno a otro según los siguientes subatributos: adaptabilidad, facilidad para instalarse, cumplimiento, facilidad para reemplazarse.

2.27. **Prueba de seguridad.**

2.27.1. **Norma ISO 27001.**

La norma ISO 27001 puede desarrollarse un Sistema de Gestión de Seguridad de la Información que permite evaluar todo tipo de riesgos o amenazas susceptibles de poner en peligro la información de una organización tanto propia como datos de terceros. (ISOTools, SGSI, 2019)

La norma ISO 27001 tiene puntos en común con dos normas ISO las cuales son: la norma ISO 22301 de continuidad de negocio y la ISO/IEC 20000 de gestión de servicio TI (tecnología de información).

Las medidas de seguridad que debe contar el sistema son la encriptación y uso de variables de sesión. Es decir, el usuario debe contar con un usuario y contraseña se creará una variable de sesión y finalizará las tareas correspondientes.

2.27.1.1. Encriptación.

El SHA1 es una función hash criptográfica ampliamente utilizado, que genera un 160 bit (20 bytes) hash a partir de cualquier valor de entrada. Esto se utiliza para calcular un valor de comprobación única para todos los datos digitales de no más de 264 -1 bit (=2 exbibyte) de longitud y es la base para la creación de una firma digital. (Jose, 2013)

El Sha1 permite realizar la encriptación de la información del usuario y contraseña para ocultarla.

2.27.1.2. Autenticación y autorización.

La autenticación define proceso por el que el usuario se identifica en el equipo donde se alojan los recursos a los que quiere acceder. La autenticación sigue el proceso mediante que el usuario obtiene los privilegios necesarios para poder tratar la información.(ISOTools, La Norma Iso 27001, 2019)

La autenticación y la autorización utiliza variables de sesión se inician cuando el sistema verifica que el usuario y contraseña son correctas además que el usuario no esté de baja.

2.28. Estimación de costos.

La estimación del software debe realizarse de forma objetiva e intentando predecir lo mejor posible el coste de desarrollo de software. Si el coste del proyecto se calcula dentro de un presupuesto para un cliente, entonces tendremos que decidir cómo se relacionan ambos conceptos. (Sommerville I. , 2005, pág. 569)

Se hallan técnicas para la estimación de costos, para ello se requiere experiencia. El manejo de costo principal para el software es sin duda el tamaño del producto.

La estimación comienza con una descripción del ámbito del problema. Luego este se descompone en un conjunto de problemas más pequeño y cada uno de estos se estima, usando como guías datos históricos y experiencias, la complejidad y el riesgo del problema se consideran antes de realizar una estimación final. (Pressman, 2010, pág. 593)

2.28.1. **Método COCOMO.**

La estimación de costos es un aspecto importante en un proyecto de desarrollo web, donde se toma en cuenta los costos de tiempo y esfuerzo; para este propósito se utilizará el método de estimación de costos COCOMO II.

El modelo COCOMO es un modelo empírico que se obtuvo recopilando datos de varios proyectos grandes estos datos fueron analizados para describir las fórmulas que mejor se ajustaban a las observaciones. Estas fórmulas vinculan el tamaño del sistema y del producto, factores del proyecto y el equipo con el esfuerzo necesario para desarrollar el sistema. (Sommerville I. , 2005, pág. 578)

COCOMO es un modelo diseñado para la estimación de costos, del número de meses que tomara para desarrollar un producto de software.

2.28.2. **Características.**

- Está basado en modelos de estimaciones matemáticas.
- Está bien documentado, además es de dominio público y le apoyan las herramientas comerciales.
- Está orientada al producto final, no a fases internacionales.
- Se basa en la cantidad de líneas de código del proyecto.
- Se ha utilizado y evaluado permanentemente.
- Tiene un buen antecedente, desde su primera versión (1981) hasta su última versión reciente COCOMO II (2000).

2.28.3. Niveles.

a) Básico.

Calcula el esfuerzo y el costo del desarrollo de software en función del tamaño del programa, expresado en las líneas estimadas de código (LDC).

Las ecuaciones del COCOMO básico tienen la siguiente fórmula:

$$E = a_b KLD^{b_b}$$

$$D = C_b E^{d_b}$$

Donde E es el esfuerzo aplicado en personas – mes, D es el tiempo de desarrollo en meses cronológicas y KLDC es el número estimado de líneas de código distribuidas (en miles) para el proyecto. Los coeficientes $a_b y^{b_b}$ con valores constantes se muestran en la tabla.(Caldero, 2010)

Tabla N° 2.1 Modelo COCOMO Básico

PROYECTO DE SOFTWARE	a_b	d_b	C_b	b_b
Orgánico	24	1.05	2.5	0.38
Semiacoplado	3.0	1.12	2.5	0.35
Empotrado	3.6	1.20	2.5	0.32

Fuente: (Caldero, 2010)

b) Intermedio.

Calcular el esfuerzo del desarrollo del software en función del tamaño del programa y de un conjunto de “conductores de costo” que incluyen la evaluación subjetiva del producto, del hardware, del personal y de los atributos del proyecto.(Caldero, 2010)

$$E = (a)(KLDC)^b(EAF); \quad T = (c)(E)^d$$

Donde:

E = Esfuerzo aplicado en hombre – mes.

T = Tiempo de desarrollo en meses.

KLDC = Número de miles de líneas de código estimado para el proyecto.

EAF = Factor de ajuste del esfuerzo.

Para calcular el esfuerzo (E) primero necesitamos hallar la variable KLDC, que responde a la siguiente relación:

$$KLDC = (PF \times \text{lineas de codigo por cada PF})/1000$$

c) Avanzado.

Incorpora todas las características de la versión intermedia y lleva a cabo una evaluación del impacto de los conductores de costos en cada fase (análisis, diseño, etc.) del transcurso de la ingeniería de software.

2.28.4. Modos de desarrollo.

a) Orgánico.

Aquellos productos de software que son relativamente pequeños y sencillos, donde trabajan pequeños equipos que poseen buena experiencia en la aplicación sobre un conjunto de requerimiento poco rígido.

b) Semiacoplado.

Son los proyectos de software intermedios hablando de tamaño y complejidad, en donde los equipos tienen diversos niveles de experiencia, y además deben satisfacer requerimientos poco o medio regido.

c) Empotrado.

Son los proyectos de software que deben ser desarrollados en un conjunto de hardware, software y restricciones operativas muy restringidas.

CAPÍTULO III. MARCO APLICATIVO

En el presente capítulo se realizará el análisis y diseño del “Sistema de Información de Registro y Control de Documentos Ambientales de las unidades industriales”, hace uso de metodología y técnicas definidas en el capítulo anterior “Marco Teórico”, sigue los lineamientos del Proceso Unificado Ágil (AUP), utiliza el lenguaje de modelado UML-Based Web Engineering (UWE).

Los mismos son desarrollados acorde a la metodología AUP en sus fases de: Inicio, Elaboración, Construcción y Transición.

Tabla N° 3.2 Fases de UWE

FASE	APLICACIÓN
Inicio	Análisis de requerimientos, detallar la arquitectura en base a la visión cliente y equipo de desarrollo.
Elaboración	Profundiza los requerimientos del sistema y validar la arquitectura. En base a los diseños.
Construcción	Codificación del software. Y pruebas de caja blanca y caja negra.
Transición	Liberar el sistema.

Fuente: (Elaboración Propia)

3.3. Fase inicio.

En esta fase se define y delimita el alcance del proyecto, identificando los riesgos asociados al proyecto y proponer una visión general de la estructura del software a desarrollarse.

3.1.2. Análisis de la situación actual.

En el diagrama de caso de uso (Figura N° 3.10) se observa los procedimientos que se efectúan en la Dirección de Gestión y Control Ambiental para el manejo de files, se realiza el proceso de recepción de documentación ambiental donde

interviene directamente la secretaria que después es derivada a la Unidad de Prevención, Control y Monitoreo Ambiental dependiente de la DGCA donde un técnico de la misma procede a la revisión de la documentación, una vez realizado el informe positivo sobre la misma es aprobado por el Director, una vez aprobado, así mismo si la Unidad Industrial tuviera infracción administrativa el Técnico del Área Legal realiza una Resolución Administrativa sancionatoria inmerso en la documentación, la Secretaria se encarga de entregar una copia de la documentación al Representante Legal, como También a el Área de Archivo.

3.1.2. Identificación de actores.

Las siguientes tablas referentes muestran la descripción de caso de uso, de los procesos de iteración que tienen los actores con el sistema, especificando las funciones que desempeña cada actor en la dirección.

Tabla N° 3.3: Descripción de Actores

ACTOR	DESCRIPCIÓN
DIRECTOR	Encargado de aprobar el documento, previo informe de la unidad técnica, que ingresa de la unidad industrial en cumplimiento al reglamento Ambiental del Sector Industrial Manufacturero (RASIM).
SECRETARIA	Es la encargada de recepcionar el documento ambiental para después asignar un número de trámite y luego derivar a la unidad técnica.
TÉCNICO	Es el encargado de realizar la revisión del documento ambiental e inspeccionar la unidad industrial, posteriormente realiza un informe.
TÉCNICO ÁREA LEGAL	Encargado de realizar la resolución administrativa sancionatoria considerando si la unidad industrial tiene alguna falta en cuanto al incumplimiento del reglamento RASIM.
ENCARGADO DE ARCHIVO	Es el encargado de registrar todos los datos de la unidad industrial y posteriormente archivar el documento, previamente realiza la revisión de los documentos en físico el cual debe cumplir tanto el informe con su acta de inspección, así como también su correspondiente nota ya sea de aprobación o de observaciones.

Fuente: (Elaboración propia)

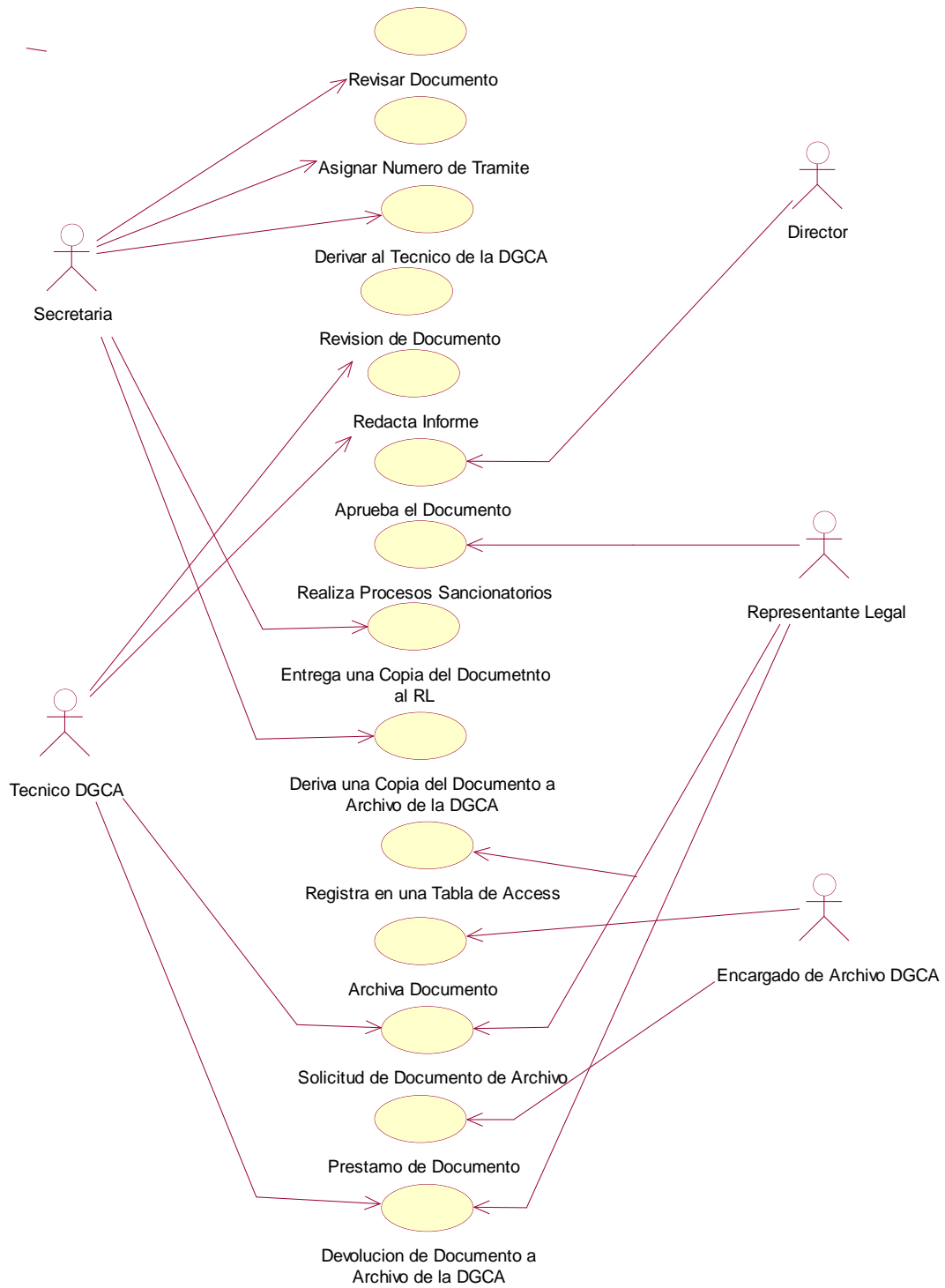
3.1.6. Modelado de negocio.

El modelado de negocio permitirá comprender mejor los procesos de funcionamiento de la documentación Ambiental de la Dirección de Gestión y Control Ambiental.

3.1.6.1. Modelo de casos de uso del negocio.

En el diagrama de caso de uso se observa los procedimientos que se efectúan en la Dirección de Gestión y Control Ambiental para el manejo de la files, se realiza el proceso de recepción de documentación ambiental donde interviene directamente la secretaria que después es derivada a la Unidad de Prevención, Control y Monitoreo Ambiental dependiente de la DGCA donde un técnico de la misma procede a la revisión de la documentación, una vez realizado el informe positivo sobre la misma es aprobado por el Director, una vez aprobado, así mismo si la Unidad Industrial tuviera infracción administrativa el Técnico del Área Legal realiza una Resolución Administrativa sancionatoria inmerso en la documentación, la Secretaria se encarga de entregar una copia de la documentación al Representante Legal, como También a el Área de Archivo.

Figura N° 3.10 Diagramas de Caso de Uso Del Negocio



Fuente: (Elaboración Propia)

3.1.7. Ingeniería de requerimientos.

El modelo de requerimientos describe los requisitos que el sistema va a cumplir. Incluye requerimientos técnicos, requerimientos de usuario, requerimiento a nivel de negocio.

3.1.7.1. Requerimientos funcionales y no funcionales.

a) Descripción de requerimiento a nivel negocio

El requerimiento que tiene la Dirección de Gestión y Control Ambiental es contar con un sistema de información para el registro y control de documentos ambientales.

b) Descripción de requerimiento a nivel usuario

R1.- Registrar datos de los funcionarios de la Dirección de Gestión y Control Ambiental.

R2.- Administrar los cargos de cada funcionario, en el cual se podrá adicionar, modificar y eliminar cargos.

R3.- Registrar la documentación ambiental de la unidad industrial.

R4.- Controlar la vigencia o caducidad de la documentación de la unidad industrial.

R5.- Realizar el registro de procesos sancionatorios de las unidades industriales.

R6.- Registrar modificación de datos de la unidad industrial.

c) Descripción de requerimientos a nivel de sistema

S1.- El Sistema registrará datos personales de los funcionarios.

S2.- El sistema registrará todos los datos de las unidades industriales.

S3.- El sistema permitirá realizar modificación y eliminación de algún cargo laboral.

S4.- El sistema mostrará un aviso cuando la vigencia del documento ambiental está por vencer o necesita ser actualizada.

S5.- El sistema realizará la modificación o actualización de datos de la unidad industrial.

S6.- El sistema mostrará datos estadísticos del registro de nuevas unidades industriales.

3.4. Fase Elaboración.

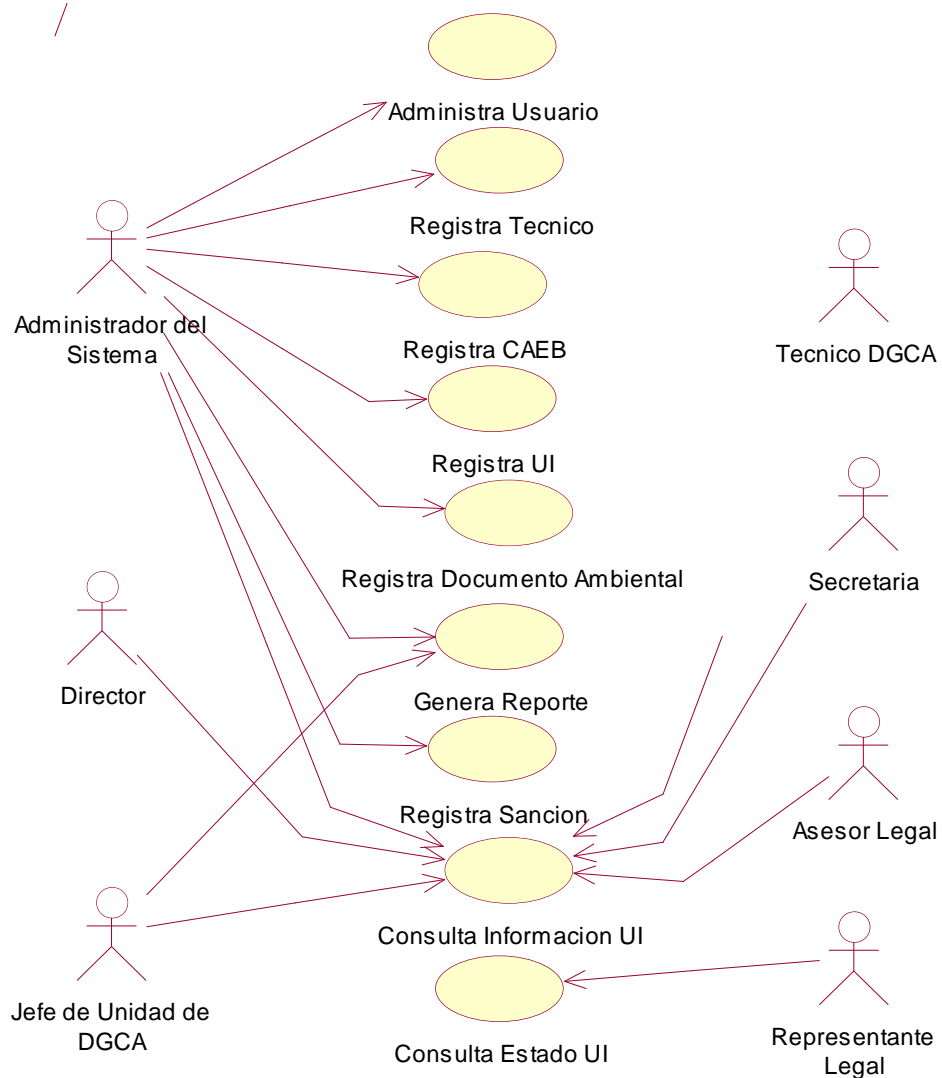
La fase de elaboración se profundiza en la comprensión de los requisitos del sistema, mediante el modelo de análisis que comprende los casos de uso y con el modelo de diseño se valida la arquitectura.

3.7.1. Modelo de caso de uso UWE.

Se realiza un análisis de los casos de uso que representan actores del sistema, hacer la descripción de cada uno de ellos. Es así que se tiene los siguientes diagramas de caso de uso.

➤ **Diagrama de caso de uso de alto nivel**

Figura N° 3.11 Casos de Uso del Sistema



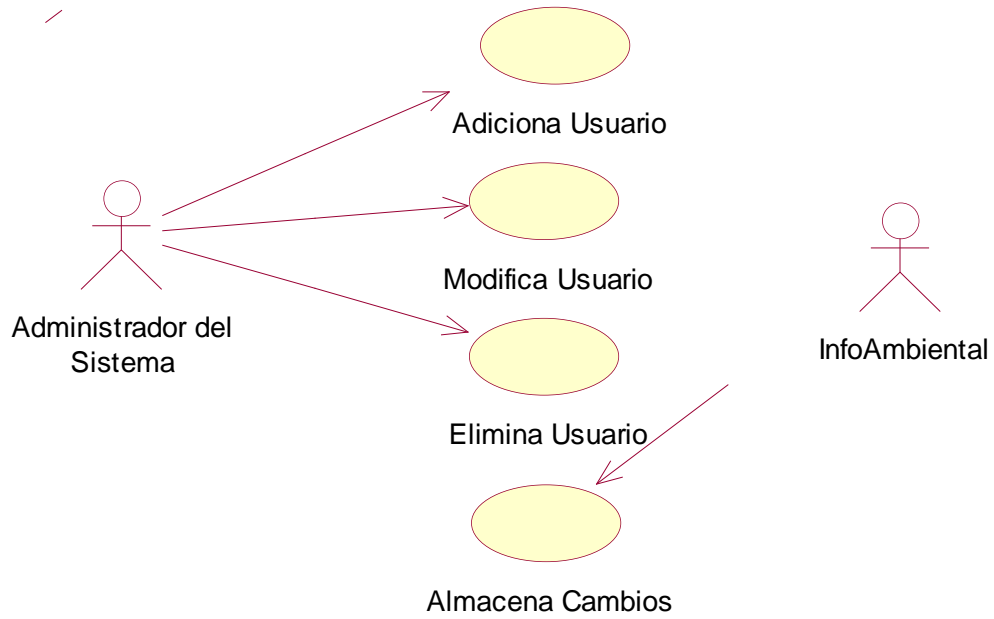
Fuente: (Elaboración Propia)

3.7.2. Descripción de caso de uso.

A continuación, se mostrarán los casos de uso, de los procesos reflejados en el diagrama de casos de uso del sistema.

➤ **Caso de Uso: Administrador de Usuario**

Figura N° 3.12 Caso de Uso - Administrador de Usuario



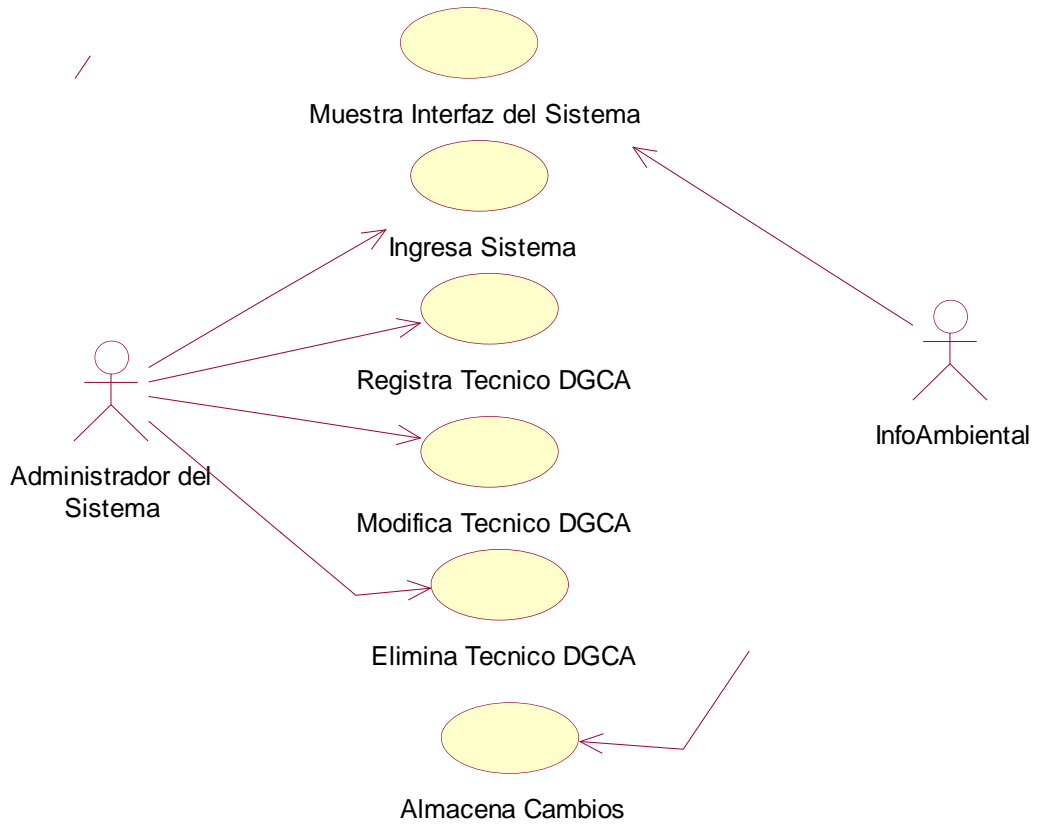
Fuente: (Elaboración Propia)

Tabla N° 3.4 Descripción de Caso de Uso - Administrador de Usuario

Caso de Uso	Administrador de Usuario
Autor	
Descripción	El administrador realizara el registro, modificación y eliminación de los usuarios
Actores	Administrador
Precondiciones	El administrador debe estar logueado en el sistema
Flujo normal	<ol style="list-style-type: none"> 1. El administrador ingresa al sistema 2. El administrador realiza el registro de usuario 3. El administrador realiza la modificación de usuario 4. El administrador realiza la eliminación de usuario 5. El sistema realiza el almacenamiento de los cambios realizados 6. Salir del sistema
Flujo alternativo	Si se realiza algún error de registro, modificación y eliminación el sistema informara dicho error
Post Condiciones	El sistema guardara todos los cambios

Fuente: (Elaboración Propia)

Figura N° 3.13 Caso de Uso - Registró Técnico



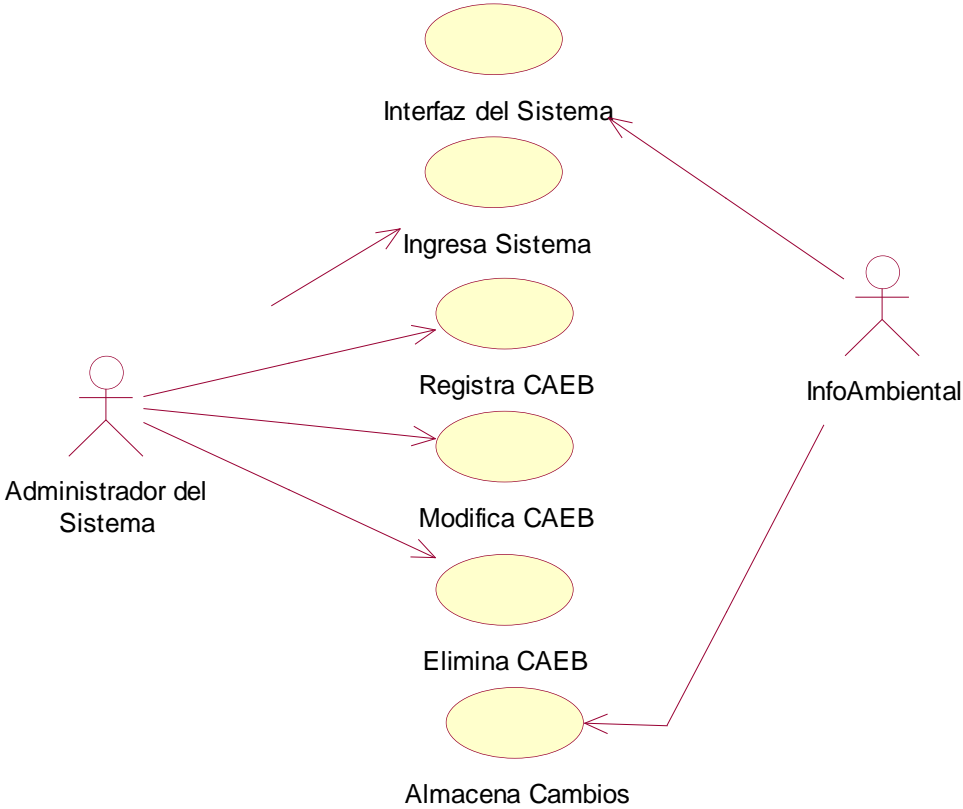
Fuente: (Elaboración Propia)

Tabla N° 3.5 Descripción de Caso de Uso - Registro Técnico

Caso de Uso	Registro Técnico
Autor	
Descripción	El administrador realizara el registro, modificación y eliminación de los técnicos
Actores	Administrador
Precondiciones	El técnico debe estar registrado, para acceder al sistema
Flujo normal	<ol style="list-style-type: none"> 1. El sistema muestra la interfaz al administrador 2. El administrador ingresa al sistema 3. El administrador elige la opción registrar técnico 4. El administrador realiza la modificación del técnico 5. El administrador realiza la eliminación del técnico 6. Se almacenará todos los cambios realizados en el sistema
Flujo alternativo	Si se realiza algún error de registro, modificación y eliminación el sistema informará dicho error
Post Condiciones	El sistema guardará todos los cambios

Fuente: (Elaboración Propia)

Figura N° 3.14 Caso de Uso - Registro de Clasificación de Actividades Económicas de Bolivia (CAEB)



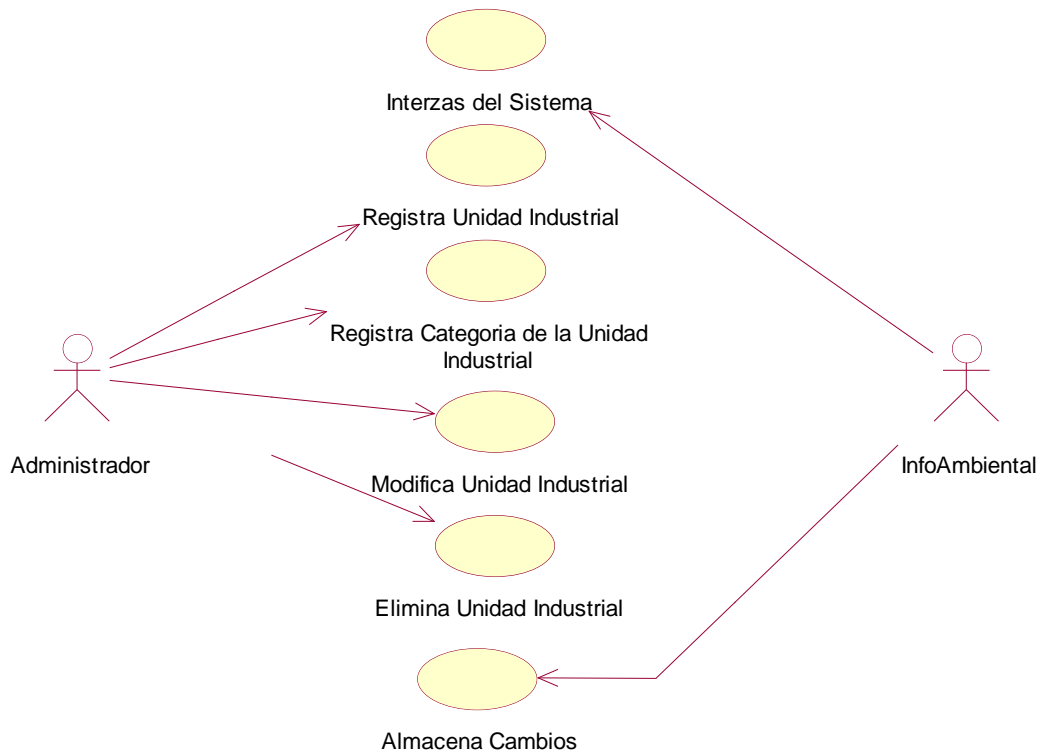
Fuente: (Elaboración Propia)

Tabla N° 3.6 Descripción Caso de Uso - Registro CAEB

Caso de Uso	Registro CAEB
Autor	
Descripción	El administrador ingresa al sistema y realizará el registro, modificación y actualización de los CAEB, para así también realizar reportes sobre la cantidad de CAEB registrados
Actores	Administrador
Precondiciones	El administrador debe estar logueado en el sistema
Flujo normal	<ol style="list-style-type: none"> 1. El sistema muestra la interfaz para interactuar 2. El administrador ingresa al sistema 3. El administrador elige la opción registrar CAEB 4. El administrador realiza la modificación del CAEB 5. El administrador realiza la eliminación de CAEB 6. El sistema almacena los cambios realizados
Flujo alternativo	Si se realiza algún error de registro, modificación o actualización el sistema informará dicho error
Post Condiciones	El sistema guardará todos los cambios

Fuente: (Elaboración Propia)

Figura N° 3.15 Caso de Uso - Registro de Unidad Industrial



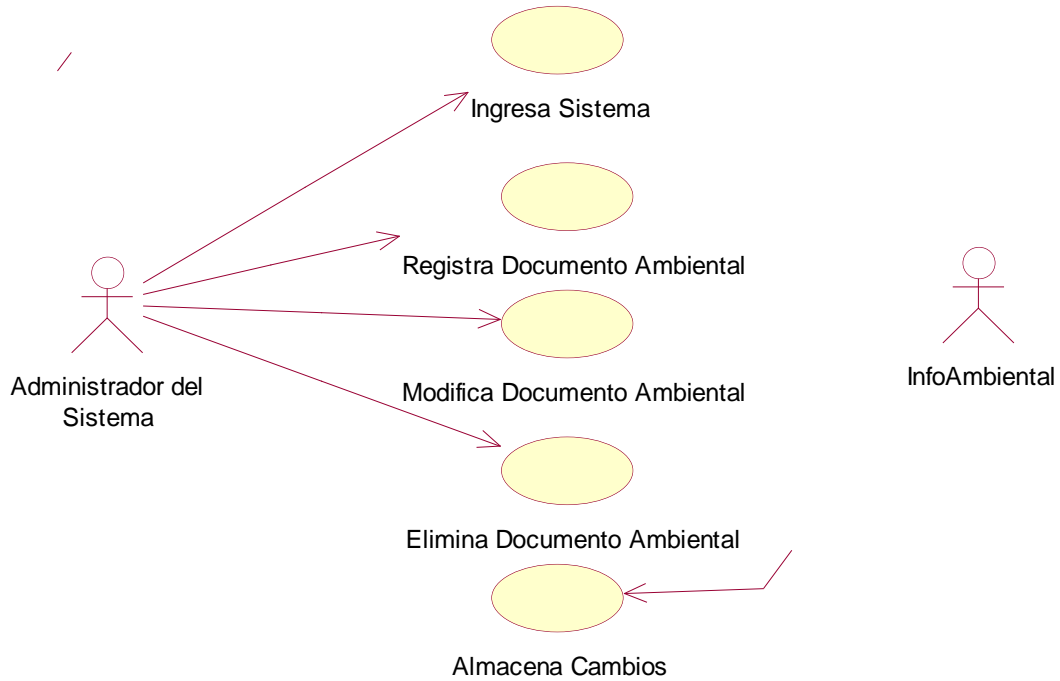
Fuente: (Elaboración Propia)

Tabla N° 3.7 Descripción Caso de Uso - Registro Unidad Industrial

Caso de Uso	Registro Unidad industrial
Autor	
Descripción	El administrador ingresa al sistema y realizará el registro, modificación y renovación de la unidad industrial
Actores	Administrador
Precondiciones	El administrador debe estar logueado en el sistema
Flujo normal	<ol style="list-style-type: none"> 1. El sistema muestra la interfaz en el cual se interactuará 2. El administrador ingresa al sistema 3. Realiza el respectivo registro de la unidad industrial 4. Podrá realizar el registro de categoría a la cual corresponde la unidad industrial 5. El administrador modificara y actualizar los datos de la unidad industrial 6. El sistema realizará el almacenamiento de los cambios realizados
Flujo alternativo	Si se realiza algún error de registro, modificación o actualización el sistema informará dicho error
Post Condiciones	El sistema guardará todos los cambios

Fuente: (Elaboración Propia)

Figura N° 3.16 Registro Documento Ambiental



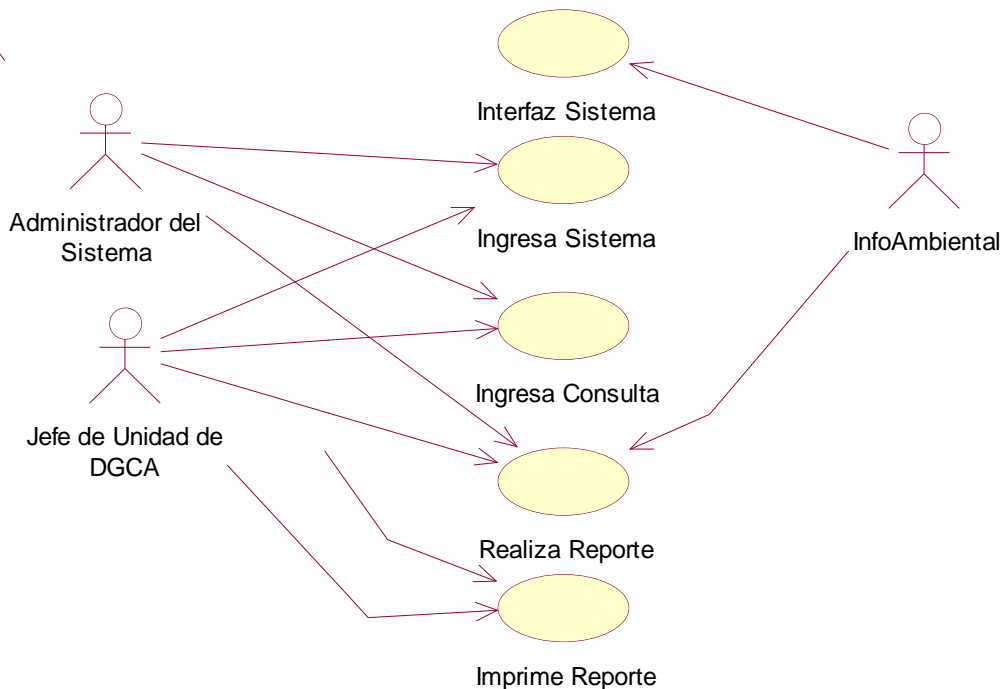
Fuente: (Elaboración Propia)

Tabla N° 3.8 Descripción de Caso de Uso - Registro de Documento Ambiental

Caso de Uso	Registro Documento Ambiental
Autor	
Descripción	El administrador ingresa al sistema y realizará el registro, modificación y actualización del documento ambiental
Actores	Administrador
Precondiciones	El administrador debe estar logueado en el sistema
Flujo normal	<ol style="list-style-type: none"> 1. El administrador ingresa al sistema 2. El administrador realizará el registro, modificación y actualización de los documentos ambientales
Flujo alternativo	Si se realiza algún error de registro, modificación o actualización el sistema informará dicho error
Post Condiciones	El sistema guardará todos los cambios

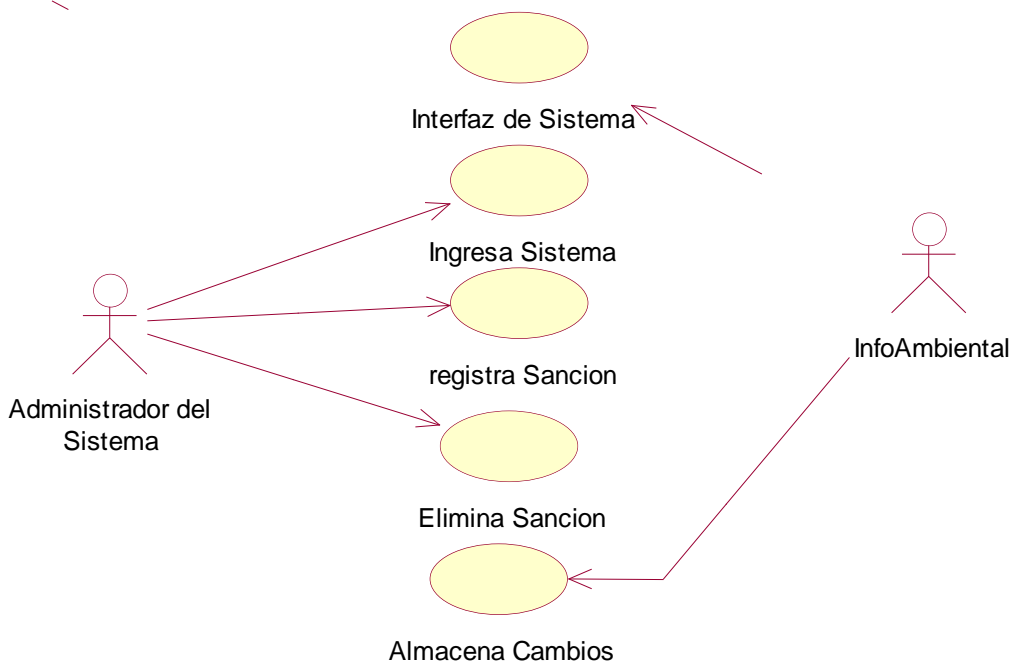
Fuente: (elaboración propia)

Figura N° 3.17 Caso de Uso Generará Reporte



Fuente: (Elaboración Propia)

Figura N° 3.18 Descripción Caso de Uso - Registra Sanción



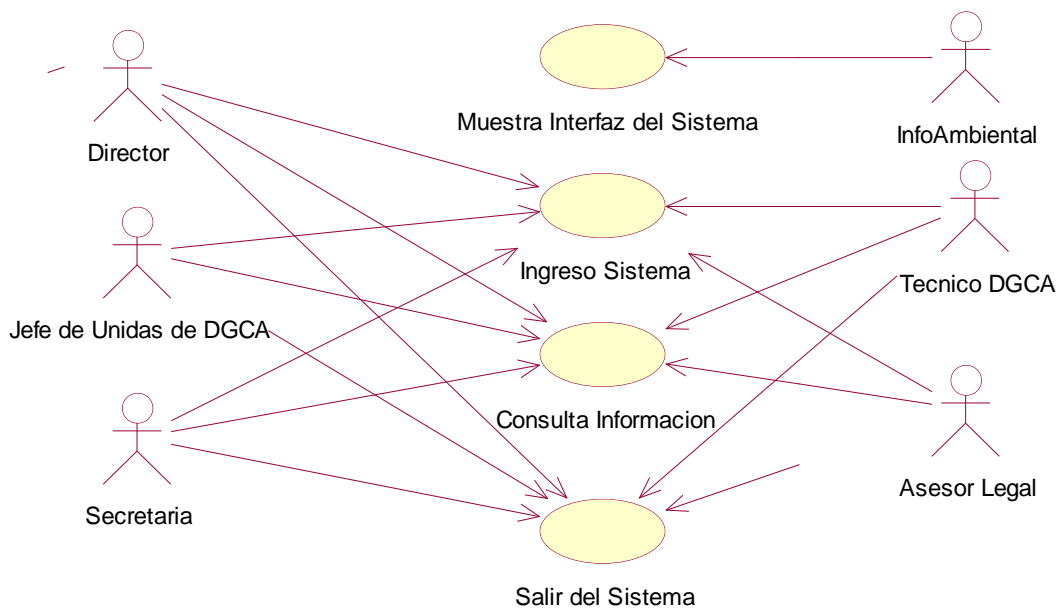
Fuente: (Elaboración Propia)

Tabla Nº 3.9 Descripción Caso de Uso - Registro Sanción

Caso de Uso	Registro de sanciones
Autor	
Descripción	El administrador ingresa al sistema y realizará el registro guarda la sanción ingresada
Actores	Administrador
Precondiciones	El administrador debe estar logueado en el sistema
Flujo normal	<ol style="list-style-type: none"> 1. El sistema muestra la interfaz 2. El administrador ingresa al sistema 3. El administrador realizará el registro. 4. El administrador elimina la sanción
Flujo alternativo	Si se realiza algún error de registro, actualización el sistema informará dicho error
Post Condiciones	El sistema guardará todos los cambios

Fuente: (Elaboración Propia)

Figura Nº 3.19 Caso de Uso Consulta Información



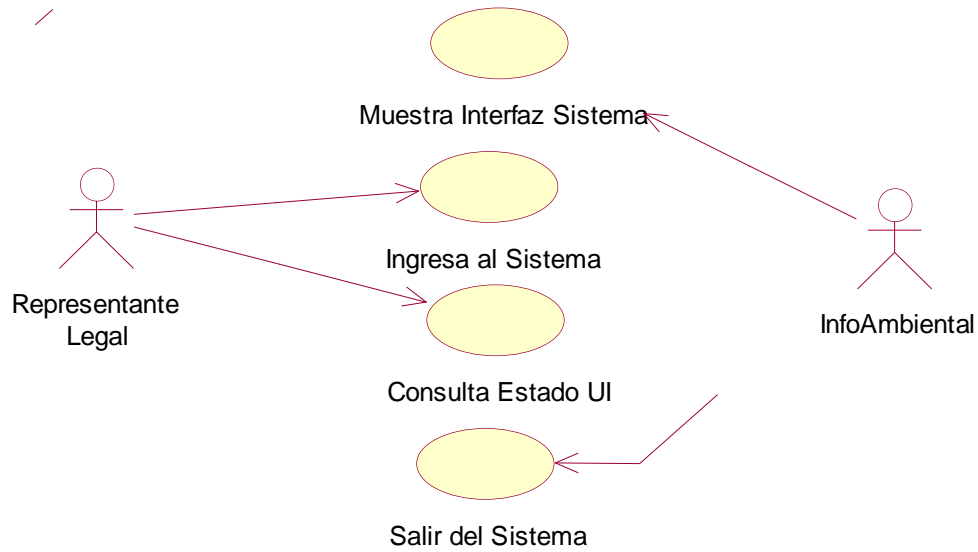
Fuente: (elaboración propia)

Tabla N° 3.10 Descripción de Caso de Uso - Consulta Información

Caso de Uso	Consulta de Información
Autor	
Descripción	El Director, Jefe de Unidad de DGCA, Secretaria, Técnico DGCA y el Área Legal podrán realizar las respectivas consultas de acuerdo a su interés
Actores	Director, Jefe de Unidad de DGCA, Secretaria, Técnico de DGCA y el Área Legal
Precondiciones	Los servidores públicos deberán ingresar su usuario y password para ingresar al sistema
Flujo normal	<ol style="list-style-type: none"> 1. El sistema muestra la interfaz en la cual se interactuará 2. Los servidores públicos deberán ingresar al sistema 3. Realizar la respectiva consulta de la información requerida 4. Salir del sistema
Flujo alternativo	Los servidores públicos sólo podrán realizar consultas
Post Condiciones	Los servidores públicos observan la información solicitada

Fuente: (Elaboración Propia)

Figura N° 3.20 Caso de Uso Consulta de Estado de Unidad Industrial



Fuente: (Elaboración Propia)

Tabla N° 3.11 Descripción Caso de Uso - Consulta Estado de la Unidad Industrial

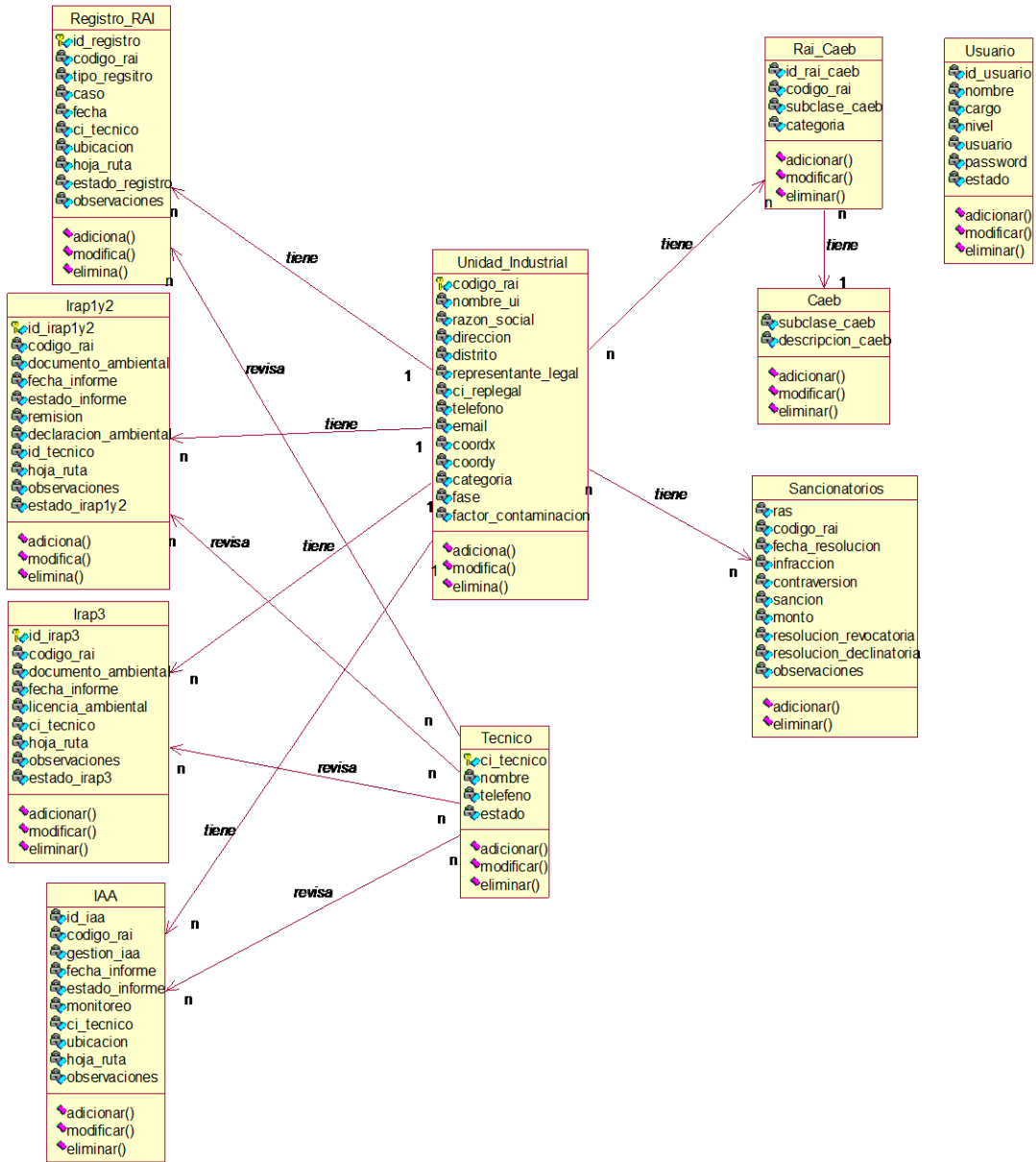
Caso de Uso	Consulta Estado de la Unidad Industrial
Autor	
Descripción	El Representante Legal podrán realizar la consulta del estado de la unidad industrial
Actores	Representante Legal
Precondiciones	El representante legal debe ingresar su usuario y password para ingresar al sistema
Flujo normal	<ol style="list-style-type: none"> 1. Se muestra la interfaz del sistema 2. El representante legal ingresar al sistema 3. Realizar la respectiva consulta del estado de la unidad industrial 4. Salir del sistema
Flujo alternativo	El representante legal realizar consulta de la unidad industrial
Post Condiciones	El representante legal observan la información solicitada

Fuente: (Elaboración Propia)

3.7.3. Modelo de Clases

El modelo de clases lo describimos las clases identificadas, atributos y métodos.

Figura N° 3.21 Diagrama de Clase

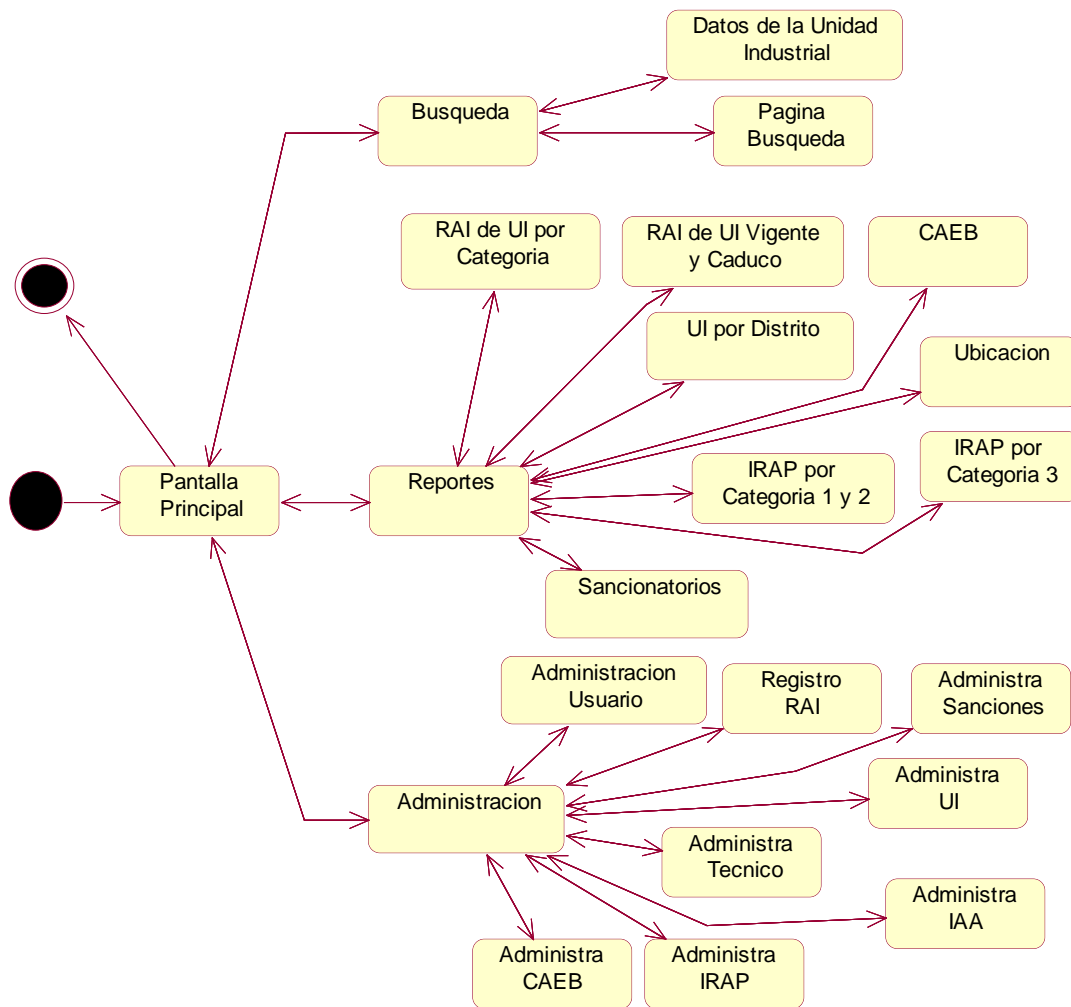


Fuente: (Elaboración Propia)

3.7.4. Modelo Navegacional

En la siguiente figura observamos el espacio navegacional que serán visitados mediante la navegación de usuarios.

Figura Nº 3.22 Modelo Navegacional

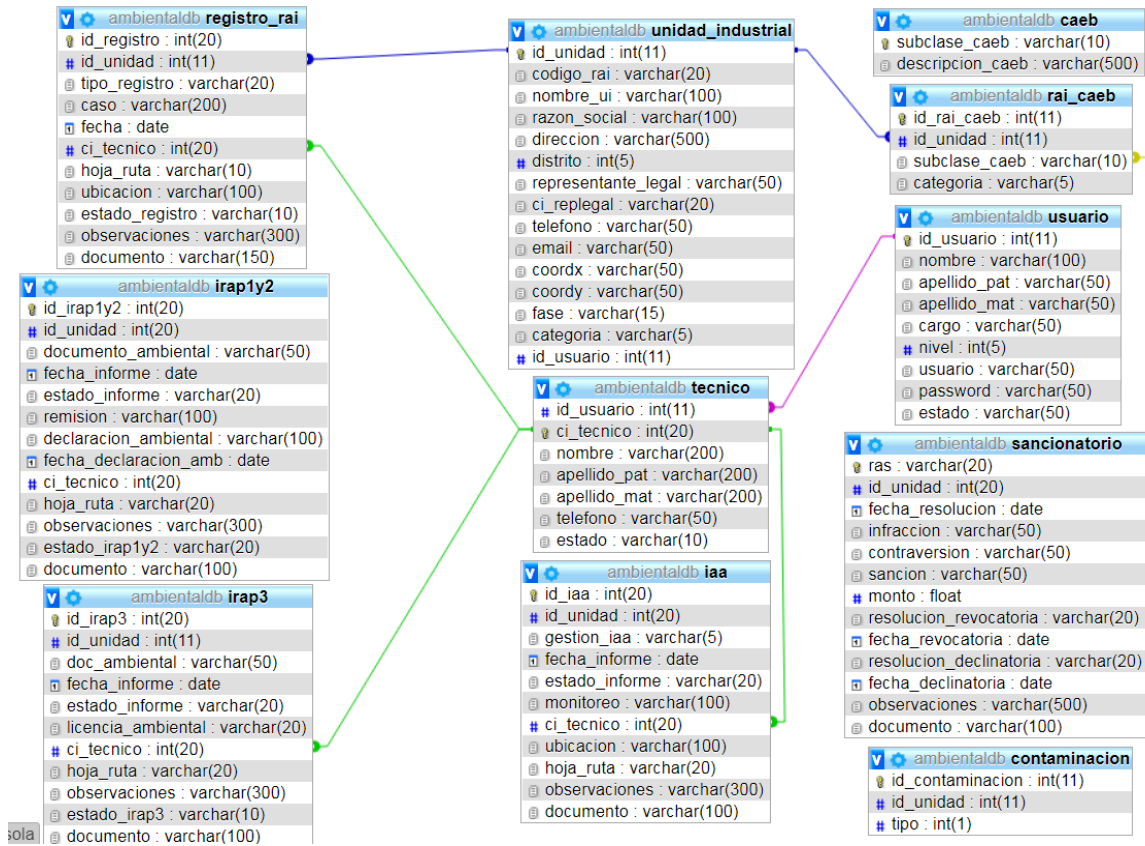


Fuente: (Elaboración Propia)

3.7.5. Modelo de diseño.

Se realiza a continuación el modelo entidad relación.

Figura N° 3.23 Modelo Entidad Relación

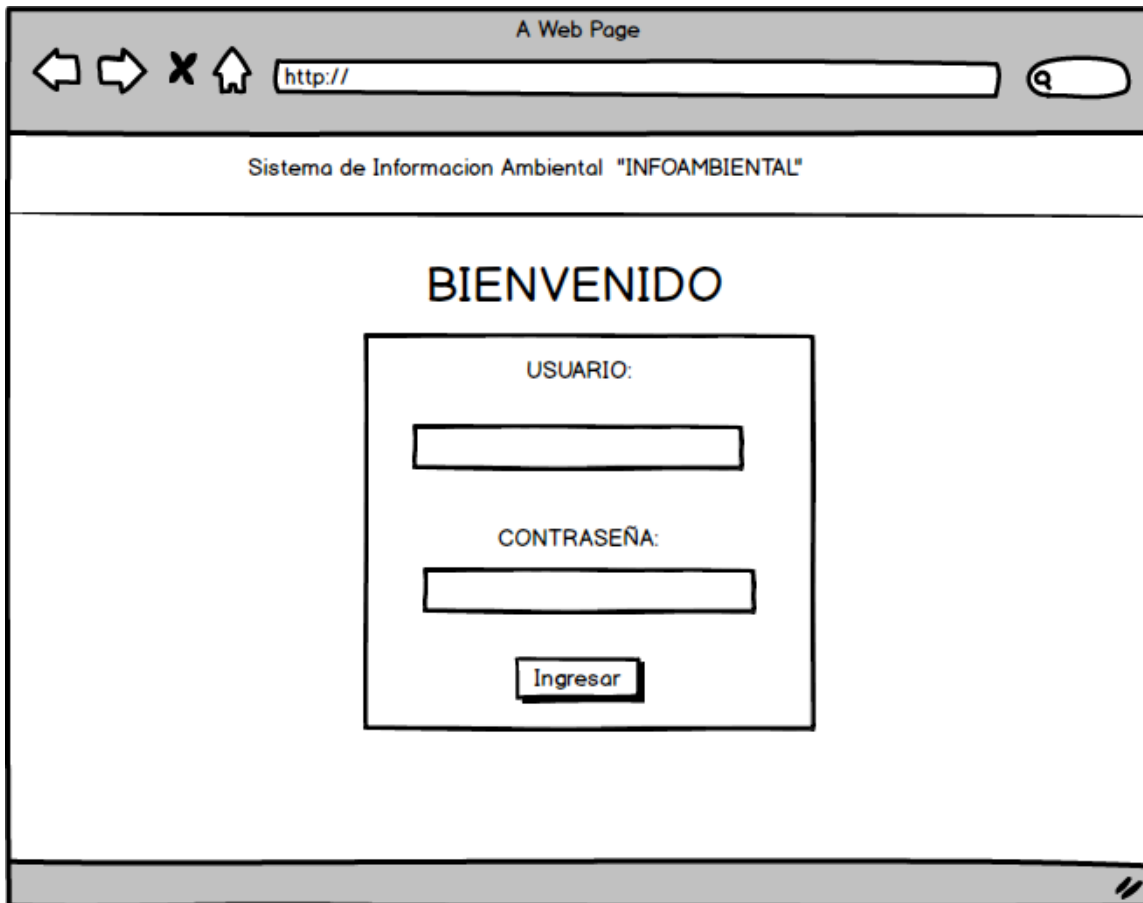


Fuente: (Elaboración Propia)

3.7.6. Modelo de Presentación.

El modelo de presentación muestra el cómo se verá el sistema, con una descripción secuencial.

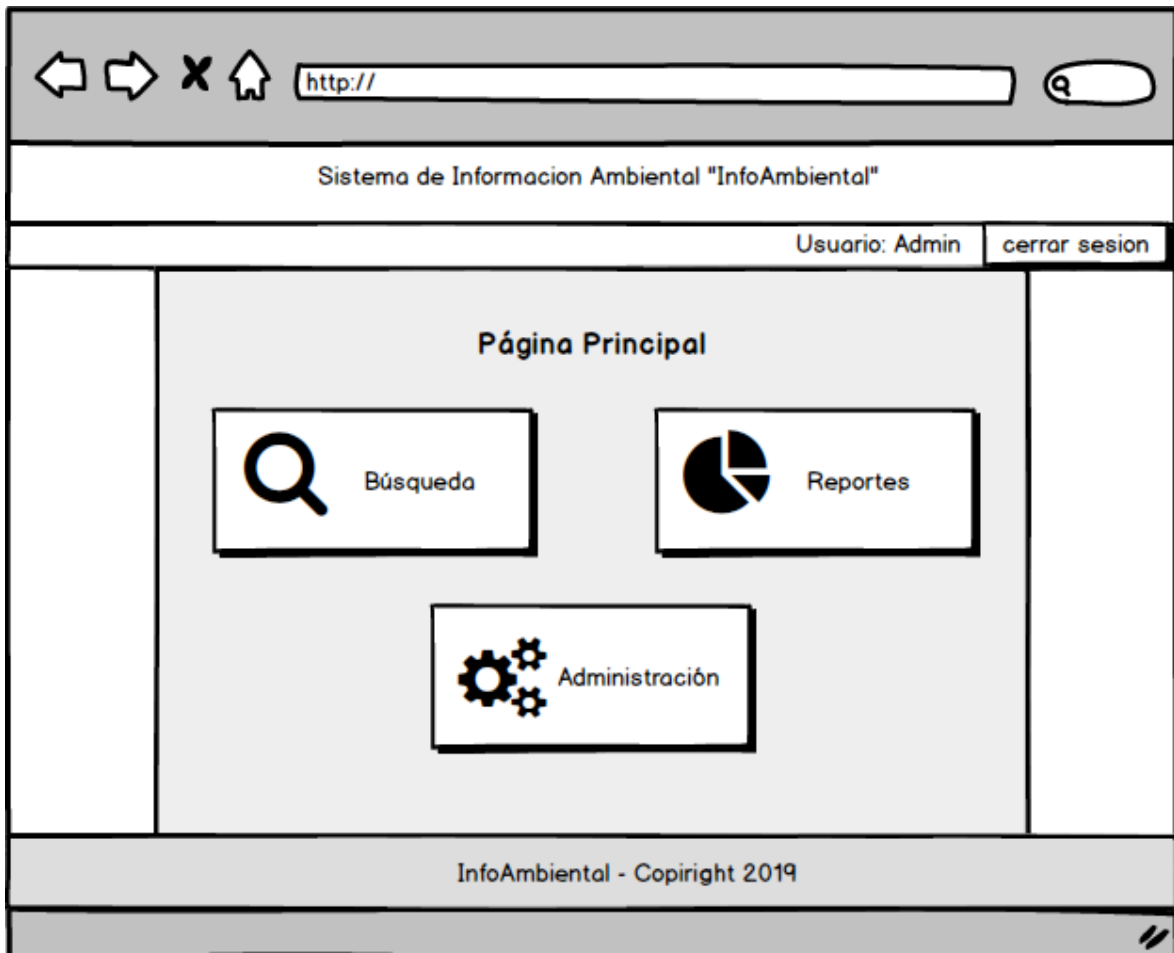
Figura N° 3.24 Modelo de Presentación - Inicio de Sesión al Sistema



Fuente: (elaboración propia)

La figura 13, muestra la interfaz de ingreso al sistema, donde el usuario introduce su usuario y contraseña.

Figura N° 3.25 Página Principal



Fuente (Elaboración Propia)

La figura 14, muestra la interfaz de la página principal, donde se podrá acceder a las secciones de búsqueda, reportes y administración.

Figura N° 3.26 Página de Búsqueda

Sistema de Informacion Ambiental "InfoAmbiental"

> Inicio Usuario: Admin cerrar sesion

Q Búsqueda

Código RAI:

Cod. RAI	Unidad Industrial	Rep Legal	Dirección	Acción
0201040020	La Francesa SRL	Mario A. Yaffar	C. a Viacha #1000	<input type="button" value="Ver"/>

Unidad Industrial:

Representante Legal:

C.I. Rep. Legal

Dirección:

InfoAmbiental - Copiright 2019

Fuente: (Elaboración Propia)

La figura 15, muestra la interfaz de búsqueda, con referencia a cinco criterios de búsqueda: por código RAI, por nombre de Unidad Industrial, por nombre de Representante Legal, CI de Representante Legal y por dirección de Unidad Industrial.

Figura N° 3.27 Registro Ambiental Industrial

The screenshot shows a web browser window with the URL 'http://'. The page title is 'Sistema de Informacion Ambiental "InfoAmbiental"'. The navigation bar includes 'Inicio', 'Busqueda', 'Usuario: Admin', and 'cerrar sesion'. The main content area displays the following information:

Código RAI: 0201040020
 Unidad Industrial: La Francesa SRL
 Representante Legal: Mario A. Yaffar
 C.I. Rep. Legal: 474454 LP
 Dirección: C. \ a Viacha # 1000

CAEB - Rubro

SubClase	Descripción	Categoría
15412	Elaboración de Galletas	4

Registro RAI

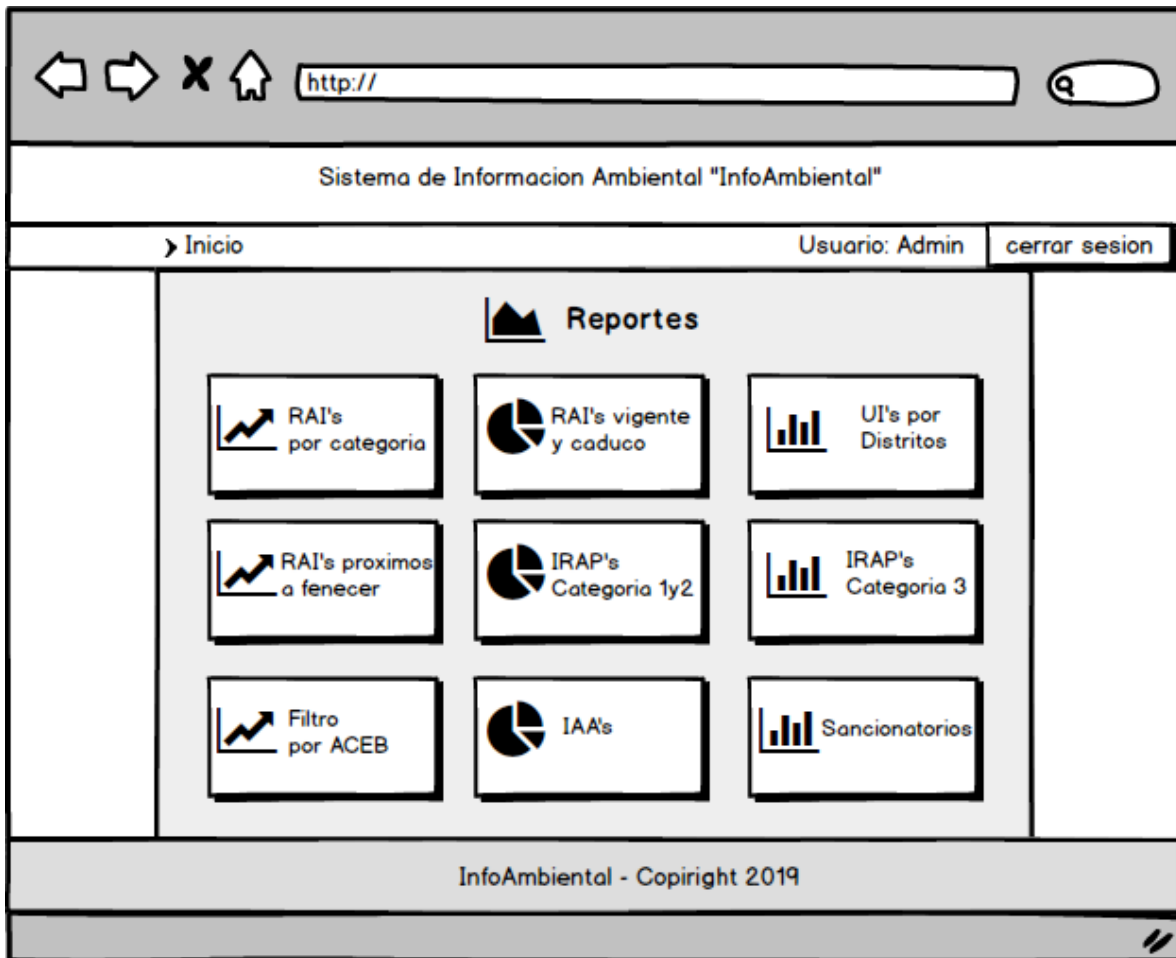
Tipo de Reg.	Razón	Fecha	Técnico	Ubicación
Nuevo	Por primera vez	09/06/2003	Rafael Irusta	Archivo

InfoAmbiental - Copiright 2019

Fuente: (Elaboración Propia)

La figura 15, muestra la interfaz del Registro Ambiental Industrial de la industria seleccionada, donde se muestra los datos generales, el rubro y los registro RAI de la Industria.

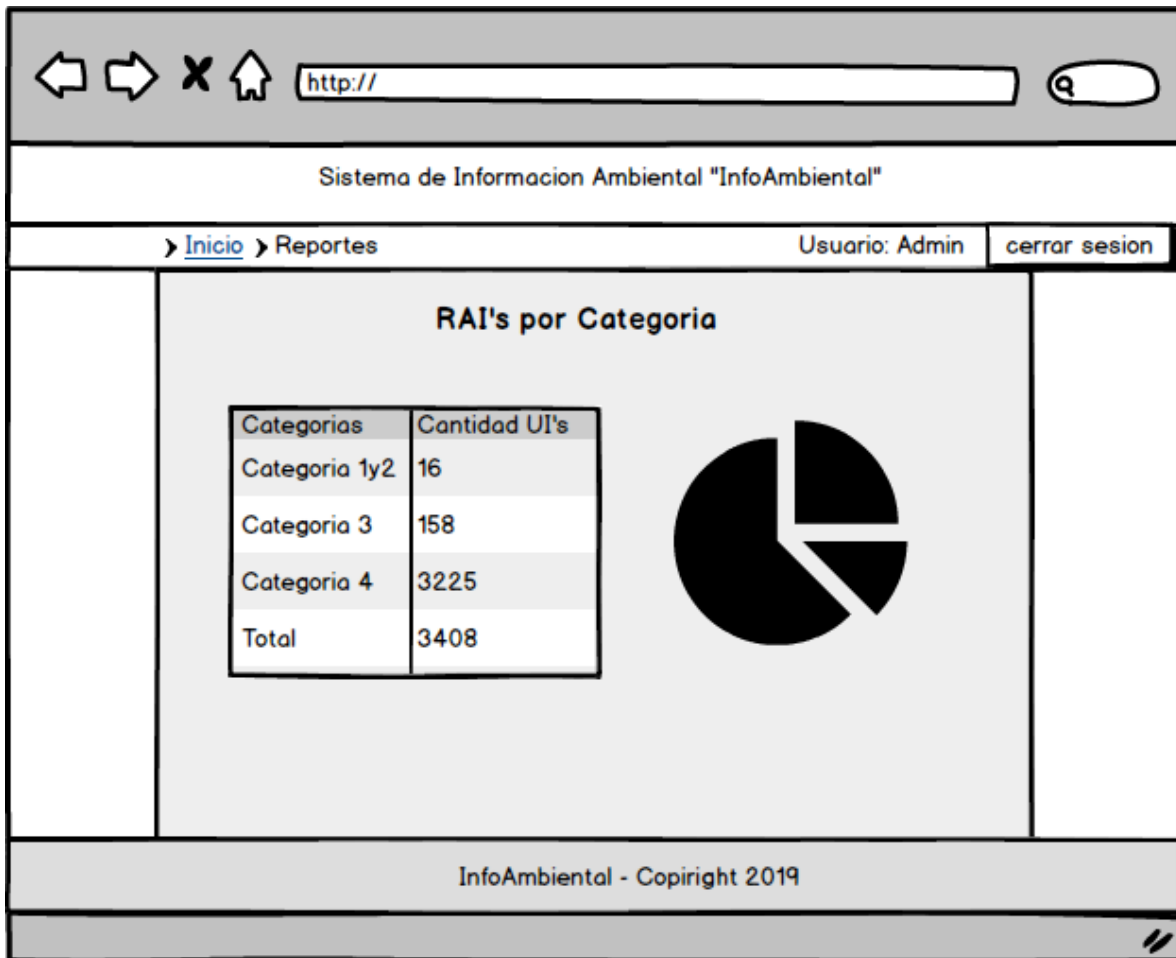
Figura N° 3.28 Reportes



Fuente: (Elaboración Propia)

La figura 17, muestra la interfaz de datos estadísticos de las Unidades Industriales.

Figura N° 3.29 RAI's por Categoría



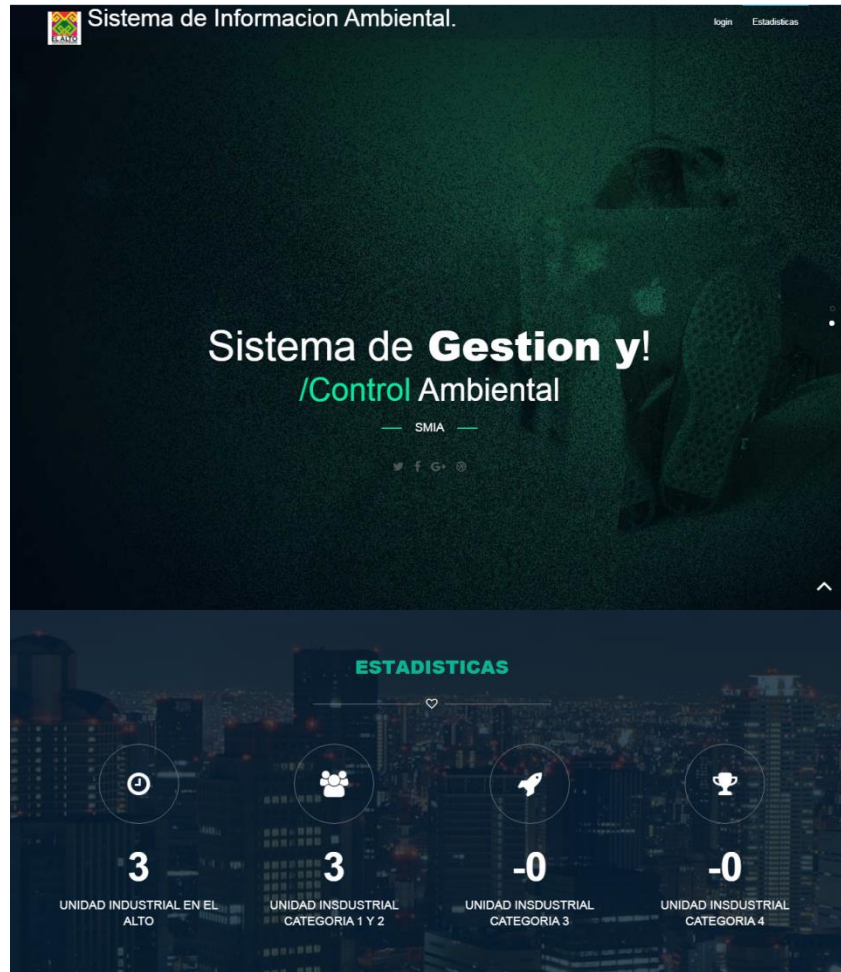
Fuente: (Elaboración Propia)

La figura 18, muestra la interfaz de RAI's por categoría el cual nos mostrará datos generales de todas las industrias registradas.

3.8. Fase construcción.

3.8.1. Diseño de interfaz web.

Figura N° 3.30 Pantalla de Bienvenida al Sistema



Fuente: (Elaboración Propia)

Figura N° 3.33 Pantalla de Datos Generales de Unidad Industrial

Datos generales

Codigo RAI:	22222	FASE ACTIVIDAD:	Proyecto
UNIDAD INDUSTRIAL:	HORNO DE PANIFICACIÓN "SAMO"	RAZON SOCIAL:	CARLOS SAMO SAMO
EMAIL:	samo@mail.com	TELEFONO:	2123456
DIRECCION:	Calle Jose Chacon (Entre Esq Alvarez Plata-Nistaus) N° 50 Zona Ballivan 1ra Sección	DISTRITO:	1
REPRESENTANTE LEGAL:	Carlos Samo Samo	CEDULA DE IDENTIDAD:	2460545 LP
COORDENADA X:	-16 500821645844482	COORDENADA Y:	-68 2210421562195
		ESTADO DE REGISTRO:	Solicitado
		CATEGORIA FINAL:	1

Caeb-Rubros de la actividad

Codigo CAEB	Descripción	Categoría
1	sadsfds	1

Registro Ambiental Industrial (RAI)

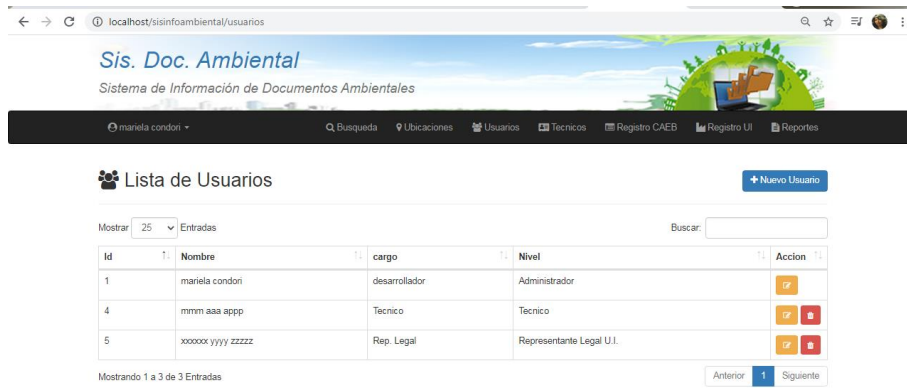
Fuente: (Elaboración Propia)

Figura N° 3.34 Pantalla de Ubicación Geográfica de las Unidades Industriales

Ubicaciones - Unidades Industriales

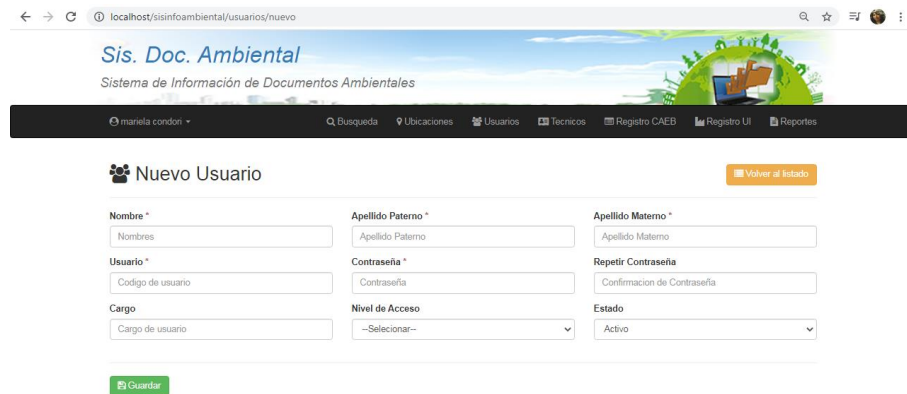
Fuente: (Elaboración Propia)

Figura N° 3.35 Pantalla de Usuarios



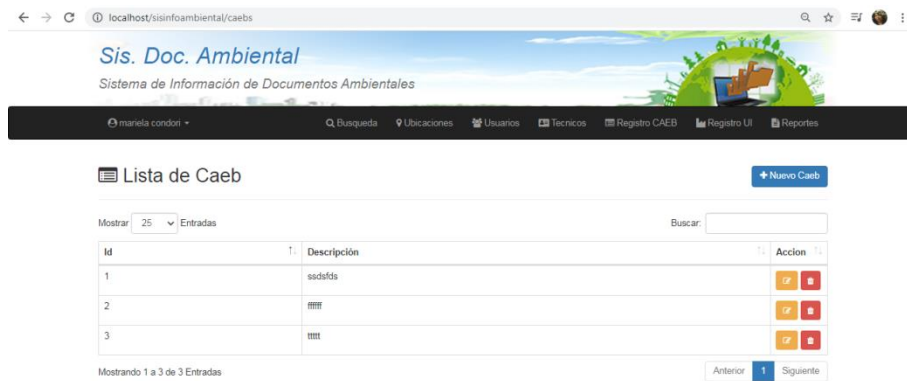
Fuente: (Elaboración Propia)

Figura N° 3.36 Pantalla de Nuevo Usuario



Fuente: (Elaboración Propia)

Figura N° 3.37 Pantalla de Registro de CAEB



Fuente: (Elaboración Propia)

Figura N° 3.38 Pantalla de Registro de Unidades Industriales

Lista de Unidad Industrial + Nueva Unidad

Mostrar 25 Entradas Buscar:

Codigo RAI	Unidad Industrial	Razon Social	Representante Legal	C.I. Rep. Legal	Accion
	Taller de Mecanica Hersa	adolfo santos	adolfo santos	9876541 LP	
22222	HORNO DE PANIFICACIÓN "SAMO"	CARLOS SAMO SAMO	Carlos Samo Samo	2460545 LP	
456	aaaa	bbbb	xxxxxx yyyy zzzzz	21345	
ddd	11134	222	55	66	

Mostrando 1 a 4 de 4 Entradas Anterior 1 Siguiente

Fuente: (Elaboración Propia)

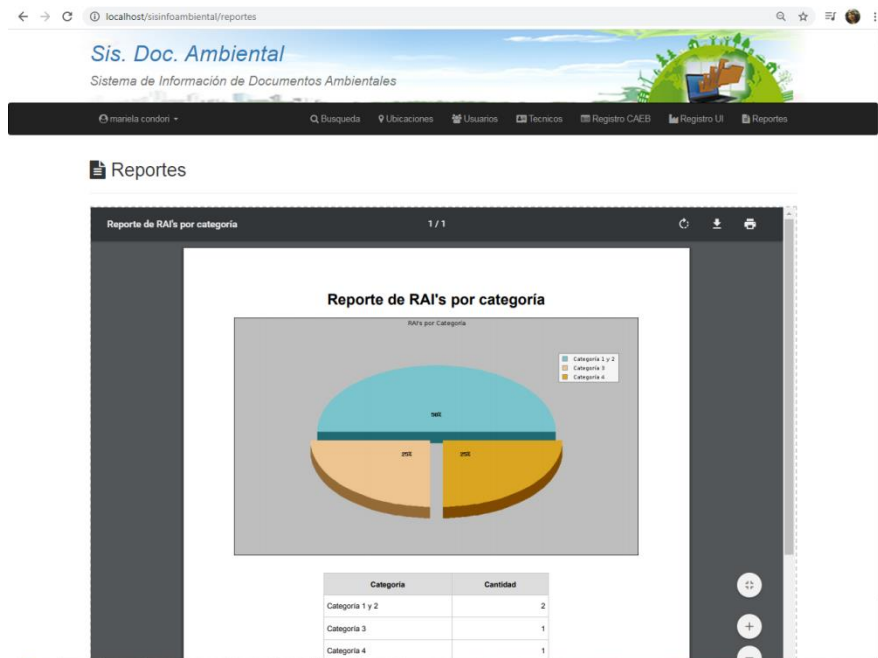
Figura N° 3.39 Pantalla Principal de Reportes

Reportes

- RAI's por categoria
- RAI's vigente y caduco
- UI's por distritos
- RAI's proximos a fenecer
- IRAP's categoria 1 y 2
- IRAP's categoria 3
- Filtro por caeb
- IAA's
- Sancionatorios

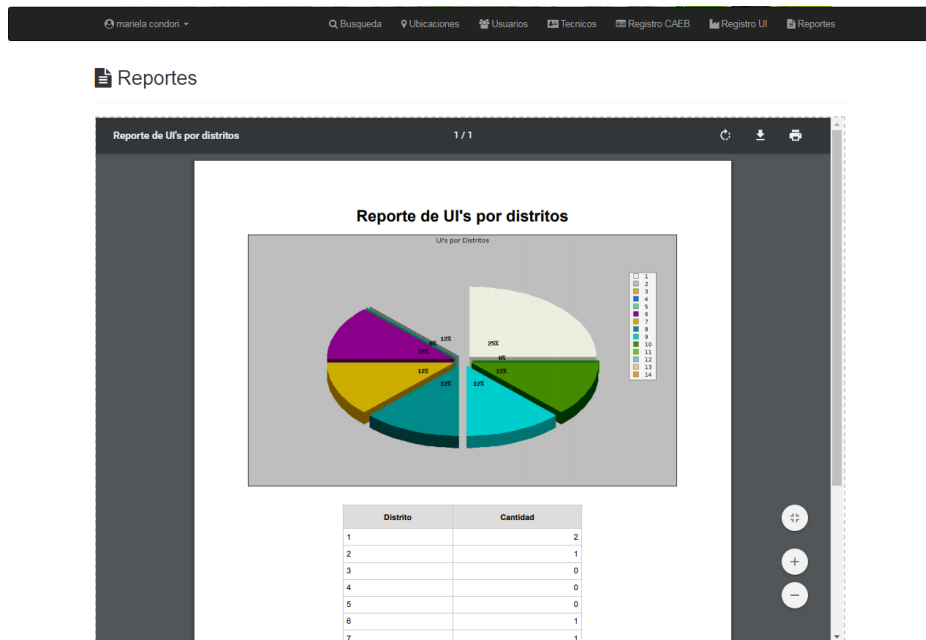
Fuente: (Elaboración Propia)

Figura N° 3.40 Pantalla de Reporte de RAI's por Categoría



Fuente: (Elaboración Propia)

Figura N° 3.41 Pantalla de Reportes de Unidades Industriales por Distrito



Fuente: (Elaboración Propia)

3.9. Fase de Transición.

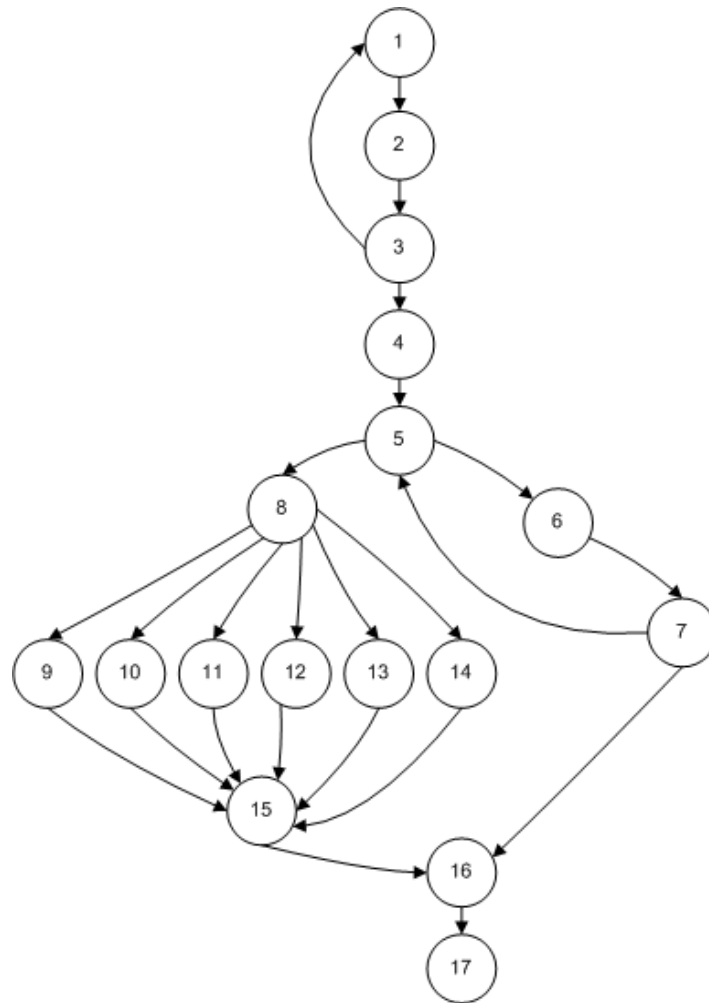
3.9.1. Prueba de Software.

En esta fase se realiza las pruebas de validación para determinar el correcto funcionamiento del sistema.

3.9.1.1. Prueba de caja blanca.

Permite obtener una medida de complejidad de un diseño procedimental, y utilizar esta medida para definir una serie de caminos básicos de ejecución, garantizando que cada camino se ejecute al menos una vez.

Figura N° 3.42 Grafo Del Sistema



Fuente: (Elaboración Propia)

Donde:

1. Inicio del sistema
2. Usuario y contraseña
3. Validar usuario y contraseña
4. Menú principal búsqueda
5. Menú administrador
6. Selecciona unidad industrial
7. Mostrar registro ambiental industrial
8. Menú administración
9. Usuario
10. Técnico
11. Clasificación de rubro (CAEB)
12. Registro unidad industrial
13. Reportes
14. Ubicación geográfica
15. Fin ciclo administración
16. Fin ciclo sistema
17. Fin sistema

Examinamos el grafo creado a partir de las características del sistema, se procede a determinar la complejidad ciclomática del grafo mediante:

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

Donde:

A = número de aristas

N = número de nodos

$$V(G) = 23 - 17 + 2 = 8$$

Por tanto, la complejidad ciclomatica es: $V(G) = 8$, esto significa que existe 8 caminos independientes.

Camino 1: 1, 2, 3, 4, 5, 8, 9, 15, 16,17

Camino 2: 1, 2, 3, 4, 5, 8, 10, 15, 16,17

Camino 3: 1, 2, 3, 4, 5, 8, 11, 15, 16, 17

Camino 4: 1, 2, 3, 4, 5, 8, 12, 15, 16,17

Camino 5: 1, 2, 3, 4, 5, 8, 13, 15, 16,17

Camino 6: 1, 2, 3, 4, 5, 8, 14, 15, 16,17

Camino 7: 1, 2, 3, 4, 5, 6, 7, 16,17

Camino 8: 1, 2, 3, 4, 5, 6, 16,17

3.9.1.2. Prueba de caja negra.

La prueba de caja negra está en probar cada una de las funciones del sistema.

Con esta prueba se busca las funciones que sean operativas.

Tabla N° 3.12 Prueba de caja negra

Entrada	
Aprobación	Registro de Usuario
	Registro de Unidades Industriales (UI)
	Registro de Clasificación de Actividades Económicas de Bolivia (CAEB)
	Registro de Documentos Ambientales
	Registro de Sanciones
Seguridad	Una vez teniendo usuarios y contraseñas se podrá acceder.
Funciones	
Software	Menú del administrador
	Menú para el director y/o jefe de unidad
	Menú para los técnicos, secretaria y asesor legal
	Menú para el representante legal
Salida	
Resultado	Reporte específicos
	Reporte de categorías por registro ambiental industrial
	Reporte del estado de las Unidades Industriales
	Reporte de Informes Ambiental Anual
	Reporte de sanciones
	Datos estadísticos del estado de las Unidades Industriales
	Datos estadísticos de Unidades Industriales por Distrito
	Mapeo general y específico de las Unidades Industriales
Seguridad	Son resultados que verán los usuarios al acceder con su usuario y contraseña

Fuente: (Elaboración Propia)

3.10. Pruebas de calidad.

3.10.1. Norma ISO 9126.

En el presente proyecto se aplicará las medidas indirectas planteadas por la norma ISO 9126, que se define por seis características:

- Funcionalidad
- Confiabilidad
- Usabilidad
- Eficiencia
- Facilidad de mantenimiento
- Portabilidad

3.10.1.1. Funcionalidad.

Para calcular la funcionalidad utilizaremos la métrica de punto de fusión, para ello determinamos las cinco características de dominio de información:

- **Números de entrada de usuario.** -Se cuenta cada entrada del usuario que proporcione al software.
- **Numero de salida de usuario.** -Se refiere a informes, pantallas, mensajes de error, etc.
- **Numero de petición de usuario.** -Se define como una entrada interactiva que resulta de algún tipo de respuesta en forma de salida iterativa.
- **Numero de archivo.** -Cuenta cada archivo maestro lógico, es decir que puede ser una parte de una base de datos o archivo independiente.
- **Numero de interfaz externa.** -Son las interfaces legibles por la máquina que utilizan para trasladar información a otro sistema.

Tabla N° 13 Cálculo de Punto de Fusión

Parámetros de Medición	Factores de ponderación		
	Cuenta	Medio	Total
N° de Entradas de Usuario	17	4	68
N° de Salidas de Usuario	17	5	85
N° de Peticiones de Usuario	10	4	40
N° de Archivos en Operación	10	10	100
N° de Interfaces Externos	1	7	7
Cuentas Total			300

Fuente: (Elaboración propia)

Para calcular el punto de fusión se utilizará la siguiente ecuación.

$$PF = Cuenta\ Total * (Grado\ de\ Confiabilidad + tasa\ de\ error * \sum f_i)$$

Donde:

PF = Medida de Funcionalidad

Cuenta Total = Es la suma de valor de las entradas, salida, peticiones, interfaces externas y archivos.

Grado de Confiabilidad = Es la confiabilidad estimada del sistema.

Tasa de error = Probabilidad subjetiva estimada del dominio de la información.

$\sum f_i$ = Son valores de ajuste de complejidad según las respuestas destacadas en la siguiente tabla.

Tabla N° 3.14 Cálculo de Punto de Fusión

N°	Factores de Complejidad	Valor
1	¿Requiere el sistema copias de seguridad y de recuperación fiable?	5
2	¿Se requiere comunicaciones de datos?	4
3	¿Existen funciones procesamientos distribuidos?	4
4	¿Es crítico el rendimiento?	3
5	¿Sera ejecutado el sistema en un entorno operativo existente y frecuentemente utilizado?	4
6	¿Requiere el sistema entrada de datos interactiva?	5
7	¿Facilidad operativa?	4
8	¿Se actualizan los archivos maestros de forma interactiva?	5
9	¿Son complejos las entradas, las salidas, los archivos o las peticiones?	4
10	¿Es complejo el procedimiento interno?	4
11	¿Se ha diseñado el código para ser reutilizable?	5
12	Facilidad de instalación	4
13	¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	4
14	Facilidad de cambio	4
Total		59

Fuente:(Elaboración propia)

Reemplazamos la ecuación.

$$PF = 300 * (0.65 + 0.01 * 59)$$

$$PF = 372$$

Entonces:

$$PF_{Maxima} = Cuenta\ Total * \left[0.65 + 0.01 * \sum F_i \right]$$

El punto de fusión máximo que se puede alcanzar es:

$$PF = 300 * (0.65 + 0.01 * 70)$$

$$PF = 405$$

Resolviendo la curva normal

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

Por lo tanto:

$$Funcionalidad = \left(\frac{372}{405} \right) * 100$$

$$Funcionalidad = 91.85\%$$

Por lo tanto, la funcionalidad del sistema es de 91%

3.10.1.2. Confiabilidad.

La confiabilidad del software debe mantener su nivel de rendimiento bajo las condiciones establecidas por un periodo de tiempo, se mide de la siguiente manera.

$$R(t) = f * e^{\left(-\frac{\lambda}{10} * t\right)}$$

Donde:

R (t)= Confiabilidad del sistema

f = es la funcionalidad del sistema ya calculada: 0.91

λ = Error que puede tener el sistema 0.03 (3%)

t = Tiempo de operación del sistema: 12 meses

Reemplazamos la ecuación con los datos obtenidos.

$$F(t) = 0.91 * e^{(-\frac{8}{10} * 12)}$$

$$F(t) = 0.32 * 100\%$$

$$F(t) = 3.2\%$$

La probabilidad del sistema tenga errores es 3.2%, esto nos dice que la probabilidad de que el sistema no tenga errores es de:

$$P(T \geq t) = 1 - F(t)$$

$$P(T \geq t) = (1 - 0.032) * 100\%$$

$$P(T \geq t) = 96\%$$

La confiabilidad del sistema es del 96%, donde en 12 meses el sistema mantendrá un rendimiento óptimo.

3.10.1.3. Usabilidad.

La usabilidad o facilidad de uso, se calcula de la siguiente manera.

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

Donde:

xi: Sumatoria de valores

n: Numero de preguntas

N: Cantidad de personas

Para calcular la usabilidad del sistema realizamos una encuesta a los usuarios de la siguiente manera:

Tabla N° 3.15 Escala de Valores

Escala	Valor
Muy buena	5
Buena	4
Media	3
Malo	2
Muy bajo	1

Fuente:(Elaboración Propia)

Tabla N° 3.16 Evaluación de Preguntas para Calcular la Usabilidad

N°	Pregunta	Evaluación
1	¿El sistema es fácil de utilizar?	8
2	¿El sistema facilita el trabajo que usted realiza?	9
3	¿Cómo considera los formularios que elabora el sistema?	8
4	¿El sistema tiene la seguridad necesaria?	7
5	¿Cómo considera el ingreso de datos del sistema?	8
Total		40

Fuente:(Elaboración Propia)

Se realizó la encuesta a 8 personas, así calculamos la facilidad de uso.

$$FU = \frac{\left[\frac{40}{8} * 100 \right]}{8}$$

$$FU = 62.5$$

Por lo tanto, la usabilidad de uso es del 62%.

3.10.1.4. Eficacia.

La eficiencia es el conjunto de atributos que se relacionan con el software y de la cantidad de recursos usados. El sistema se considera eficiencia por la óptima utilización de los recursos

3.10.1.5. Facilidad de mantenimiento

Para la facilidad del mantenimiento se usará el índice de madurez del software se calcula de la siguiente manera:

$$IMS = \frac{[M_t - (F_a + F_c + F_d)]}{M_T}$$

Donde:

Mt = Numero de módulos en la versión actual.

Fc = Numero de módulos en la revisión actual que se ha cambiado.

Fa = Numero de módulos en la versión actual que se han añadido.

Fd = Numero de módulos en la versión actual que se han eliminado.

Donde se obtendrá los siguientes valores:

Tabla N° 3.17 Facilidad de Mantenimiento

Información	Valores obtenidos
Mt	10
Fc	0
Fa	3
Fd	0

Fuente:(Elaboración Propia)

Calculamos el índice de madurez del software los cuales son los resultados obtenidos en el sistema.

$$IMS = \frac{[10 - (0 + 3 + 0)]}{10}$$

$$IMS = 0.7 * 100\%$$

$$IMS = 70\%$$

Entonces el índice de madurez alcanzado es del 70%, este se encuentra en un rango satisfactorio.

3.10.1.6. Portabilidad.

La portabilidad se describe al esfuerzo necesario para trasladar el programa de entorno ya sea hardware y/o software a otro. La portabilidad se enfoca en tres aspectos los cuales son:

- Hardware del Servidor
- Sistema Operativo del Servidor
- Software del Servidor

El sistema es portable en sus diferentes entornos tanto hardware y software por lo que se considera portable el 100%.

3.11. Pruebas de seguridad.

La seguridad de la información se puede definir como un conjunto de medidas técnicas organizacionales y legales que puedan permitir a la organización proteger y asegurar su confidencialidad, integridad de la información que genera.

3.11.1. Norma ISO 27001.

El sistema debe de contar con las medidas de seguridad como la encriptación y uso de variables de sesión. La encriptación nos ayuda a encriptar datos como ser contraseñas, así también las variables de sesión ayudan en el acceso restringido a un sistema, es decir que el usuario debe de tener usuario y contraseña para la verificación existente del usuario se crea una variable de sesión y finaliza las

tareas correspondientes al final se destruye las variables de sesión creadas por el sistema.

3.11.1.1. Encriptación.

Para la encriptación se utilizará el algoritmo SHA1 el cual ya está incluido en PHP, SHA1 es de la familia de hash, que nos permite encriptar la contraseña y la verificación se realiza comparando encriptaciones con la finalidad de proteger las contraseñas con la salida de SHA1 de 160 bits (20 bytes).

Figura N° 3.43 Encriptación

```
136     $usuario = $this->input->post('usuario');
137     $contrasena = sha1($this->input->post('contrasena'));
138
```

Fuente: (Elaboración Propia)

3.11.1.2. Autenticación y autorización.

La autenticación y autorización van unidas por los accesos de usuario a distintos niveles de información. Este proceso realiza la autenticación de usuario tanto como los encargados o administradores del sistema, además permite comprobar la compatibilidad y el origen ya sea de un programa, una función, una secuencia o una persona.

Figura N° 3.44 Autenticación y Autorización

```
134     public function ingresar()
135     {
136         $usuario = $this->input->post('usuario');
137         $contrasena = sha1($this->input->post('contrasena'));
138         if ($this->usuario->verificar_usuario($usuario, $contrasena) {
139             if ($this->session->userdata('nivel')==7){
140                 redirect( base_url('inicio/rai_unidad/'.$this->session->userdata('id')) );
141             }else{
142                 redirect( base_url('unidades/busqueda') );
143             }
144         } else {
145             $this->session->set_flashdata('error', 'Error: Datos erróneos');
146             redirect( base_url('inicio') );
147         }
148     }
149
```

Fuente: (Elaboración Propia)

3.12. Estimación de Costos.

Para calcular el costo del proyecto se utilizará el modelo constructivo COCOMO II.

Se muestra el valor de dominio, para hallar el punto de función.

Tabla N° 3.18 Conteo de cada Valor de Dominio

Valor de dominio de información	Conteo	Factor ponderado			
		Simple	Promedio	Complejo	
Entradas externas (EE)	17	3	4	6	68
Salida externa (SE)	17	4	5	7	85
Consulta externa (CE)	10	3	4	6	60
Archivos lógicos internos (ALI)	10	7	10	15	100
Archivos de interfaz externos (AIE)	1	5	7	10	7
Conteo total					320

Fuente:(Elaboración Propia)

Calculamos el ajuste de complejidad del sistema se debe valorar según los parámetros de evaluación.

Tabla N° 3.19 Cálculo de Ajuste de Complejidad

N°	Escala	Valor
1	¿El sistema requiere copia de seguridad y de recuperación fiable?	5
2	Se requiere comunicación de datos	4
3	Existen funciones de procesamiento distribuido	2
4	Es crítico el rendimiento	3
5	Se ejecutará el sistema operativo con un entorno operativo existente y fuertemente utilizado	4
6	Requiere el sistema entrada de datos interactiva	3
7	Facilidad operativa	4
8	Se actualiza los archivos maestros de forma interactiva	4
9	Son complejas las entradas, las salidas, los archivos a las peticiones	4
10	El procesamiento interno es complejo	4
11	El código es reutilizable	4
12	Facilidad de instalación	5
13	El sistema soporta múltiples instalaciones en diferentes sitios	4
14	La aplicación se diseña para facilitar el cambio y su uso por parte del usuario	4
$\sum (F_i)$		54

Fuente:(Elaboración Propia)

El punto de función (PF), reemplazamos valores en la siguiente relación:

$$PF = (Cuenta\ Total * (0.65 + 0.01 * \sum (F_i)))$$

$$PF = 320 * (0.65 + 0.01 * 54)$$

$$PF = 380.8$$

Ahora convertimos los PF a miles de líneas de código.

Tabla N° 3.20 Métrica de Desarrollo de Software – Según herramientas de Desarrollo

Lenguaje	LDC/PF
Ensamblador	320
C	150
COBOL	105
C++	64
Visual Basic	32
SQL	12

Fuente:(Elaboración Propia)

Calculamos KLDC:

$$KLDC = \frac{PF * LDC}{1000}$$

$$KLDC = \frac{380.8 * 12}{1000}$$

$$KLDC = 4.56$$

Las ecuaciones del COCOMO II tienen la siguiente forma:

Se aplicará la formula básica de esfuerzo, tiempo calendario y personal requerido.

$$E = a_b * (KLDC)^{b_b} \text{ esfuerzo de perona por mes}$$

$$D = c_b * (E)^{d_b} \text{ Tiempo de desarrollo en mes}$$

Tabla N° 3.21 Coeficientes

Proyecto de Software	a_b	b_b	c_b	d_b
Orgánico	3.2	1.05	2.5	0.38
Semi-acoplado	3	1.12	2.5	0.35
Empotrado	3.6	1.20	2.5	0.32

Fuente:(Elaboración Propia)

Como el sistema es un proyecto intermedio, en tamaño y complejidad, se elige Semi-acoplado.

Donde:

$$E = a_b * KLDC^{b_b}$$

$$E = 3 * 4.6^{1.12}$$

$$E = 16.5$$

$$D = c_b * (E)^{d_b}$$

$$D = 2.5 * (16.5)^{0.35}$$

$$D = 6.66$$

El personal requerido se obtiene con la siguiente fórmula:

$$\text{Número Programadores} = \frac{E}{D}$$

$$\text{Número Programadores} = \frac{16.5}{6.6}$$

$$\text{Número Programadores} = 2.5 = 3$$

Suponiendo que el salario aproximado de un programador es de 300, el costo será:

$$\text{Costo Software} = \text{Número de Programadores} * 300$$

$$\text{Costo Software} = 3 * 300$$

$$\text{Costo Software} = 900 \text{ \$us}$$

CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES.

4.1. Conclusiones.

El presente proyecto llegó a las siguientes conclusiones:

- El sistema de información para el registro y control de documentos ambientales fue desarrollado, haciendo el uso de la metodología UWE con el que se realizó el análisis, diseño y desarrollo.
- El sistema utiliza niveles de usuario activando sesión en cada ingreso al sistema.
- Se desarrolló una interfaz de trabajo amigable para la Dirección de Gestión y Control Ambiental, que le permite el registro de datos de las unidades industriales para posteriormente realizar consultas que brinde información rápida y fiable.
- Permite obtener información sobre el estado de los documentos ambientales. El cual brinda la información a los técnicos para realizar una inspección a la unidad industrial.
- El sistema genera reportes e informes de datos específicos de las unidades industriales.
- El sistema genera datos estadísticos de la cantidad de unidades industriales por distrito y categoría.
- El sistema muestra la ubicación geográfica de todas las unidades industriales.

4.2. Recomendaciones.

Después de realizado el proyecto se recomienda lo siguiente:

- Se recomienda al administrador realizar el mantenimiento de los datos para que la información tenga valor y uso de manera que no exista algún tipo de errores, para la obtención de reportes.
- Para el mantenimiento y configuración del sistema se requiere personal con conocimientos informáticos en análisis y diseño, base de datos MySQL y conocimiento de plataformas de desarrollo PHP.
- Se recomienda descargar un backup de la base de datos y verificar el acceso de los usuarios para tener una mayor seguridad.

BIBLIOGRAFÍA

- Armando, A. (2018). *Programacion PHP*.
- Aurelio, G. (23 de Julio de 2017). *Metodología*. Obtenido de Metodología para el Desarrollo y Mantenimiento de Software y Sistemas de Informacion: <https://metodologia.es/aup/>
- Borbon Ardila, I. N. (12 de Marzo de 2013). *Evaluacion de Software*. Obtenido de Norma de Evaluacion ISO 9126: <http://actividadreconocimiento-301569-8.blogspot.com/2013/03/norma-de-evaluacion-isoiec-9126.html>
- Caldero, W. (7 de Octubre de 2010). *Ingenieria de Software* . Obtenido de COCOMO: <http://ingenieraupoliana.blogspot.com/2010/10/cocomo.html>
- Camacho, D. (30 de Octubre de 2013). *Ciclo de Vida de la AUP*. Obtenido de <http://danielcamachosanchez.blogspot.com/2013/10/ciclo-de-vida-de-la-aup-proceso>
- CarreraUniversitaria. (2019). *Ingenieria de Software*. Obtenido de <https://micarrerauniversitaria.com/c-ingenieria/ingenieria-de-software/>
- Ciencia, U. -D. (2004). *Sistemas de Informacion. Tecnologías de la informacion*. Barcelona.
- Cordero L., J. L. (s.f.). *Metodologias Agiles*. La Paz, El Alto - Bolivia.
- Damian, L. (2011). *HTML*. Buenos Aires: User Digital Desing.
- Diaz Bravo, L., Torruco Garcia, U., Martinez Hernandez, M., & Varela Ruiz, M. (2013). *Metodologia de Investigación en Educación Médica*. Mexico D. F.
- EconLink. (27 de 08 de 2019). *Sistema de Informacion*. Obtenido de Sistema de Informacion: <https://www.econlink.com.ar/sistemas-informacion/definicion>
- Eguiluz Perez, J. (2008). *Introduccion a CSS*.
- Galiano, L. (3 de Noviembre de 2012). *Metodologia UWE aplicada*. Obtenido de Informe de la Metodologia Aplicada: <http://elproyectodeluisgaliano.blogspot.com/2012/11/metodologia-uwe-aplicada-mi-solucion.html>
- Informatics, I. f. (10 de 08 de 2016). *UWE-UML-based Web Engineering*. Obtenido de <https://uwe.pst.ifi.lmu.de/teachingTutorialPresentationSpanish.html>
- ISOTools. (2019). *La Norma Iso 27001*. Obtenido de Aspectos Clave de sus Diseño e Implementacion: <http://info.isotools.org/ebook-iso-27001?hsCtaTracking=ae0d87df-3148-4ae7-871c-c60b085059af%7C29a0d704-6f6d-4053-ae3d-b9676df57f7a>

- ISOTools. (22 de Agosto de 2019). SGSI. Obtenido de ISO 27001 : <https://www.pmg-ssi.com/2019/08/iso-27001-aspectos-claves-y-relacion-con-las-normas-iso-22301-e-iso-iec-20000/>
- Jose, D. (18 de Abril de 2013). *Seguridad en Redes*. Obtenido de SHA1: <https://statusexcessu.wordpress.com/2013/04/18/sha-1/>
- Koch, N., & Escalona, M. J. (2002). *Ingenieria de Requisitos en Aplicaciones para la Web*. Sevilla.
- Largo Garcia, C. A., & Marin Mazo, E. (2005). *Guia Tecnica para Evaluacion de Software*.
- Lopez Roldan, P., & Fachelli, S. (2015). *Metodologia de la Investigacion Social Cuantitativa*. Barcelona, España: <http://ddd.uab.cat/record/129382>.
- Lorenzo, J. C. (2012). *Cuestionarios*.
- Manufacturero, R. A. (1992). *Ley de Medio Ambiente*. Bolivia.
- Metodo de Observacion. (s.f.). *Tema 4: El Metodó Observación*. Obtenido de Introduccion de la Psicología: <http://www4.ujaen.es/~eramirez/Descargas/tema4>
- Nieves Guerrero, C., Ucan Pech, J., & Menendez Dominguez, V. (2014). *UWE en Sistemas de reconocimiento de Objetos de Aprendizaje*. Merida, Yucatan, Mexico.
- Perez Escarzo, G. (29 de Noviembre de 2016). *Metodologia AUP*. Obtenido de Proceso Unificado Agil: <https://es.scribd.com/document/332680589/METODOLOGIA-AUP>
- Pressman. (2005). *Ingenieria de Software*. En R. S. Pressman, *Ingenieria de software* (pág. 383). Mexico: Marcela I. Rocha Martínez.
- Pressman, S. R. (2010). *Ingenieria de Software*. Nueva York: Marcela Teraza Zapata.
- Ramirez, J. Q. (s.f.). *Decreto Supremo No. 26736*. Bolivia.
- RASIM. (2004). *Reglamento Ambiental para el Sector Industrial Manufacturero*. Bolivia.
- Rives, A. P. (2011). *Manual de JavaSript*. Madrid: cep.
- Sommerville, I. (2005). *Ingenieria de Software*. Madrid: Pearson Educación.
- Sommerville, I. (2005). *Ingenieria del Software*. Madrid España: Pearson Educacion S.A.

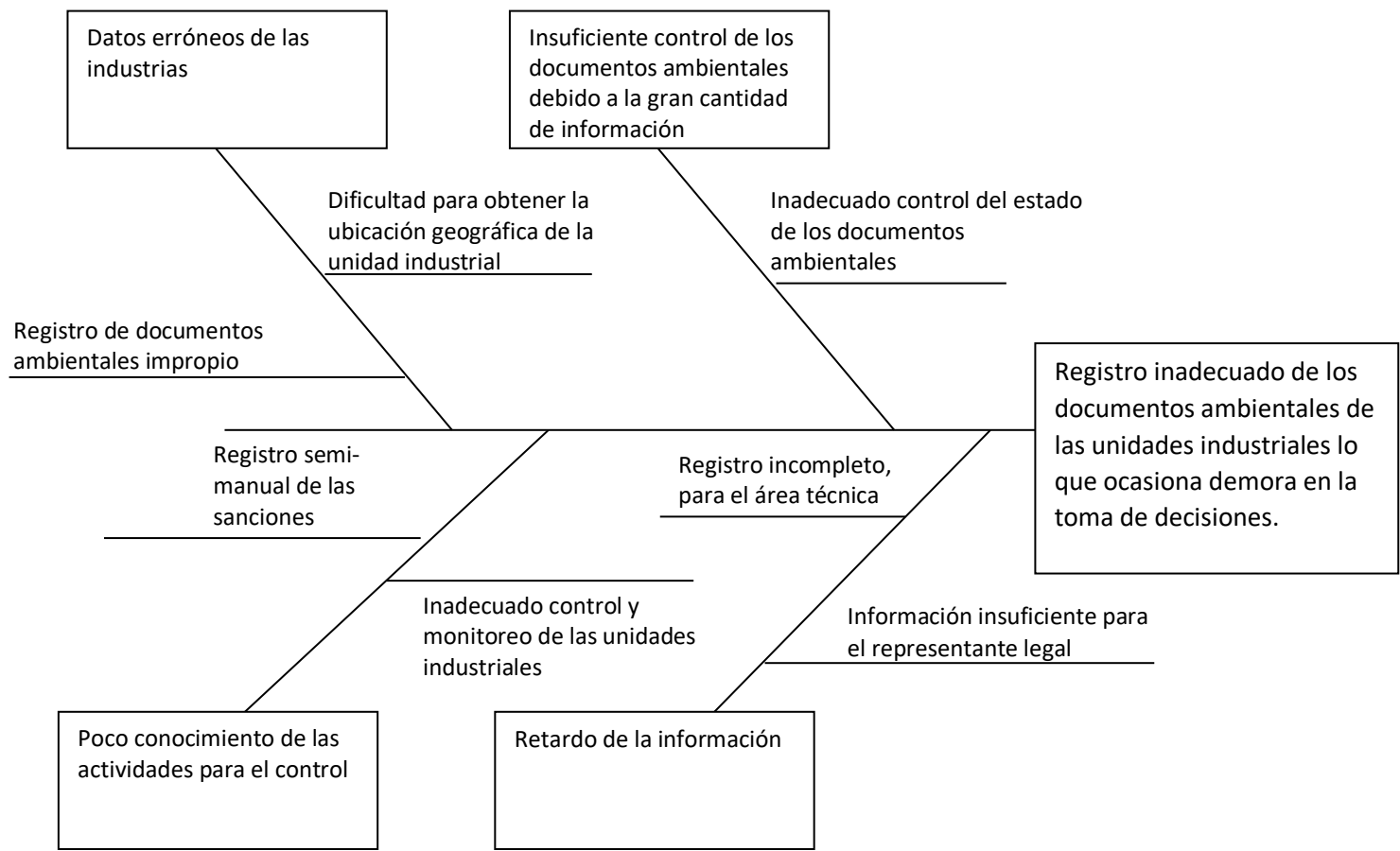
Valdivia Caballero, J. J. (2016). *Modelo de Procesos para el Desarrollo del Frontend de Aplicaciones Web*. Lima-Peru.

Viceministerio de Industria, C. y. (2003). *Reglamento Ambiental para el Sector Industrial Manufacturero*. La Paz, Bolivia.

Viceministerio de Industria, C. y. (2003). *Reglamento Ambiental para el Sector Manufacturero*. La Paz, Bolivia.

ANEXOS

Anexo "A" Arbol de Problemas



Anexo "B" Árbol de Objetivos

