

UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA



PROYECTO DE GRADO

**“CONTROL Y GESTIÓN DE ACTIVOS FIJOS VIA WEB
CASO: GRUPO LARCOS INDUSTRIAL LTDA.”**

PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMÁTICA
MENCIÓN: INGENIERÍA DE SISTEMAS INFORMÁTICOS

Postulante: Edwin Enriquez Chambi Rojas

Tutor Metodológico: Lic. Grover Alex Rodríguez Ramírez

Asesor: Lic. Javier Hugo Reyes Pacheco

LA PAZ – BOLIVIA
2014



**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA**



LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.

LICENCIA DE USO

El usuario está autorizado a:

- a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
- b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
- c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.

Dedicatoria:

A mis padres Francisco y Dominga, quienes son la principal inspiración de superación y fortaleza en mi vida, a mis hermanos: Vilma y Walter, por su apoyo constante y confianza.

Con mucho cariño

Edwin Enríquez Chambi Rojas

Agradecimientos

A nuestro Dios sobre todas las cosas por concederme la vida y la salud para culminar una etapa más en mi vida.

Agradecer a mi mamá y papá por darme todo su apoyo y comprensión incondicional para que salga adelante, a mi hermana Vilma por su comprensión y apoyo, a mi hermano Walter por darme ánimos para continuar estudiando.

Al Lic. Grover Alex Rodríguez Ramírez docente tutor metodológico, por sus consejos y por guiarme a largo del desarrollo de este proyecto.

Al Lic. Javier Hugo Reyes Pacheco, docente Asesor, por brindarme su apoyo incondicional, tiempo dedicado y por su paciencia, que me ayudaron a culminar este proyecto.

A los docentes de la carrera de Informática.

Agradecer también a la empresa Grupo Larcos Industrial Ltda., especialmente al Ing. Jorge Cabero Tapia por darme la oportunidad de realizar este proyecto.

A mis amigos que siempre me han estado apoyando y ayudando.

De todo corazón muchas gracias a todos. .

RESUMEN

El presente proyecto fue desarrollado en la empresa Grupo Larcos Industrial LTDA., en la Gerencia de Operaciones, que realiza el control, seguimiento y la asignación de los activos fijos, de las tres sucursales que tiene la empresa

La Gerencia de Operaciones no cuenta con un sistema de información automático es decir todo lo realizan manualmente lo que ocasiona volúmenes de papelería con información.

Por lo mencionado anteriormente se desarrolló el Control y Gestión de activos fijos vía web, optimizando el trabajo en el tiempo de procesos y además llevar un control adecuado de la información.

El producto obtenido cuenta con todas las características requeridas por los usuarios, resultando una herramienta de ayuda para los procesos que se efectúan en la Gerencia de Operaciones.

Este proyecto fue realizado con la metodología de desarrollo de software del Proceso Unificado Ágil denotado por el acrónimo AUP para el análisis y diseño del sistema, para el modelado del sistema se utilizó la propuesta de Ingeniería Web basado en UML UWE, utilizando la herramienta case Magic Draw para representar los diferentes objetos o diagramas necesarios para el modelado del sistema.

La unidad de Sistemas de la empresa Grupo Larcos Industrial LTDA., trabaja sobre la plataforma Windows, es por este motivo que el sistema fue desarrollado con ASP.NET 2010, como lenguaje de programación se utilizó C# y como gestor de base de datos se utilizó SQL server 2008.

La calidad del sistema fue medida con la norma ISO 9126 con todas las características necesarias, también se efectuó una descripción de las amenazas del sistema y la aplicación de medidas de seguridad para el funcionamiento adecuado del sistema.

INDICE GENERAL

	Pág.
CAPITULO I.....	1
MARCO REFERENCIAL.....	1
1.1 Introducción.....	1
1.2 Antecedentes.....	2
1.2.1 Antecedentes de la Empresa.....	2
1.2.2 Antecedentes del Proyecto.....	3
1.3 Planteamiento de Problema.....	4
1.3.1 Problema Principal.....	4
1.3.2 Problemas Específicos.....	4
1.4 Objetivos.....	5
1.4.1 Objetivo Principal.....	5
1.4.2 Objetivos Específicos.....	5
1.5 Justificación.....	5
1.5.1 Justificación Económica.....	5
1.5.2 Justificación Social.....	6
1.5.3 Justificación Técnica.....	6
1.6 Alcances y Límites.....	7
1.6.1 Alcances.....	7
1.6.2 Límites.....	8
1.7 Aportes.....	8
1.8 Metodología.....	9

CAPITULO II.....	11
MARCO TEÓRICO.....	11
2.1 Marco Institucional.....	11
2.1.1 Organigrama.....	11
2.1.1.1 Gerencia de Operaciones.....	12
2.1.1.2 Unidad de Contabilidad.....	12
2.2 Metodología	12
2.2.1 Modelo en cascada.....	13
2.2.2 Método iterativo.....	14
2.2.3 Método evolutivo.....	15
2.2.4 Método ágil.....	16
2.2.5 Metodología ágil AUP.....	17
2.2.5.1 Estructura del proceso unificado Ágil (AUP).....	17
2.2.5.2 Disciplinas de AUP.....	18
2.2.5.3 Fases del AUP.....	19
2.2.5.4 Disciplina de Modelado.....	20
a) Flujo de Trabajo.....	21
b) Fase Por Fase.....	21
2.2.5.5 Disciplina de la implementación.....	23
a) Flujo de trabajo.....	23
b) Fase por Fase.....	23
2.2.5.6 Disciplina de Pruebas.....	24
a) Flujo de Trabajo.....	25
b) Fase Por Fase.....	25
2.2.5.7 Disciplina de administración de la configuración.....	26

	a) Flujo de trabajo.....	27
	b) Fase por Fase.....	27
2.2.5.8	Disciplina Administración de proyecto.....	28
	a) Flujo de trabajo.....	28
	b) Fase por Fase.....	29
2.2.5.9	Disciplina del entorno.....	30
	a) Flujo de trabajo.....	31
	b) Fase por Fase.....	31
2.3	Metodología de modelado UWE.....	33
2.3.1	UWE y su relación con UML.....	33
2.3.1.1	Modelo de casos de Uso.....	34
2.3.1.2	Modelo de contenido.....	34
2.3.1.3	Modelo de Navegación.....	35
2.3.1.4	Modelo de Presentación.....	36
2.3.1.5	Modelo de proceso.....	36
	a) Modelo de Estructura de Procesos.....	36
2.4	Lenguaje unificado de modelado UML.....	37
2.4.1	Vistas UML.....	38
2.4.1.1	Vistas de casos de uso.....	39
2.4.1.2	Vistas de Diseño.....	39
2.4.1.3	Vistas de Proceso.....	39
2.4.1.4	Vistas de Despliegue.....	39
2.4.1.5	Vistas de Implementación.....	39
2.5	Activos Fijos.....	40
2.5.1	Activos Fijos o Bienes de Uso.....	40

2.5.2	Codificación de Activos Fijos.....	40
2.5.3	Depreciación de bienes de uso.....	41
2.5.4	Métodos de Depreciación.....	42
2.5.4.1	Método de Línea Recta.....	42
2.5.4.2	Método de la suma de los dígitos del año.....	42
2.5.4.3	Método de la reducción de saldos.....	43
2.5.5	Baja de Activos Fijos.....	43
2.6	Tecnología de Software.....	44
2.7	Calidad de Software.....	46
2.7.1	Definición e Implementación de la Evaluación Elemental.....	46
2.7.2	Criterios elementales Absoluto con variable continua (a).....	47
2.7.3	Criterios de elementales Absoluto con variable discreta (b).....	48
2.7.4	Medición Elemental.....	48
2.8	Estudio de Costo Beneficio.....	53
2.8.1	Modelo Constructivo de Costos Cocomo.....	53
2.8.2	Modelo Basico.....	54
2.8.3	Modelo Intermedio.....	55
2.8.4	Modelo Detallado.....	57
2.9	Seguridad.....	58
2.9.1	Nivel Fisico.....	58
2.9.1.1	La protección contra la electricidad.....	58
2.9.1.2	La protección contra los ruidos eléctricos, los altibajos de tensión y los cortes de corriente.....	58
2.9.1.3	Protección contra suciedad.....	59
2.9.1.4	Seguridad contra incendios y agua.....	59

2.9.2 Nivel Logico.....	59
2.10 Políticas de Implementación.....	60
2.10.1 Corte y cambio.....	61
2.10.2 Operación paralela.....	61
2.10.3 Operación Piloto.....	61
2.10.4 Operación de fase.....	62
CAPITULO III.....	63
MARCO APLICATIVO.....	63
3.1 Introducción.....	63
3.2 Fase de Inicio.....	63
3.2.1 Modelado del Negocio.....	63
3.2.1.1 Modelado de casos de uso del Negocio.....	63
3.2.1.2 Descripción de actores del caso de uso de negocio.....	64
3.2.2 Modelado de Requerimientos.....	65
3.2.2.1 Descripción de requerimiento a nivel de negocios.....	65
3.2.2.2 Descripción de requerimientos a nivel de usuario.....	65
3.2.2.3 Descripción de requerimientos a nivel de sistema.....	66
3.2.2.4. Descripción de requerimientos a nivel técnico.....	66
3.3 Fase de Elaboración.....	67
3.3.1 Modelo de Análisis.....	67
3.3.1.1 Modelo de casos de uso.....	67
3.3.1.2 Diagrama de casos de uso de alto nivel.....	67
3.3.1.3 Descripción de casos de uso.....	68
a) Caso de uso: Administrar Empleado.....	68

b) Caso de uso: Administrar Usuario.....	70
c) Caso de uso: Seguimiento de Activos Fijos.....	71
d) Caso de uso: Administrar Activos Fijos.....	72
e) Casos de uso: Administrar Asignación.....	74
f) Caso de uso: Depreciación de Activos Fijos.....	75
g) Caso de uso: Baja de Activos Fijos.....	76
3.3.2 Modelo de diseño.....	77
3.3.2.1 Diagrama de clases.....	77
3.3.2.2 Modelo Entidad Relación.....	79
3.3.2.3 Modelo de Navegación.....	80
3.3.2.4 Modelo de Presentación.....	80
3.4 Fase de construcción.....	81
3.4.1 Diseño de interfaces.....	81
3.4.1.1 Autenticación.....	81
3.4.1.2 Pantalla principal.....	82
3.4.1.3 Registro de empleado.....	83
3.4.1.4 Registro de Grupo de Inventario.....	84
3.4.1.5 Registro de Asignación de Inventario.....	85
3.5 Fase de transición.....	86
3.6. Pruebas.....	86
3.6.1 Pruebas de aceptación.....	87
CAPITULO IV.....	90
MÉTRICAS DE CALIDAD.....	90
4.1 Introducción.....	90

4.2 Características Propuestas por WEB site QEM.....	90
4.2.1 Funcionalidad.....	91
4.2.2 Fiabilidad.....	92
4.2.3 Usabilidad.....	93
4.2.4 Eficiencia.....	94
4.2.5 Portabilidad.....	95
4.2.6 Mantenibilidad.....	96
CAPITULO V.....	99
EVALUACIÓN DE COSTO BENEFICIO.....	99
5.1 Introducción.....	99
5.2 Análisis de costos.....	99
5.2.1 Costo del software desarrollado.....	100
5.2.2 Costo de la implementación del proyecto.....	103
5.2.3 Costo de elaboración del proyecto.....	104
5.2.4 Costo total.....	104
5.3 Análisis de beneficios.....	104
5.4 Valor neto actual.....	105
5.5 Tasa interna de retorno.....	107
CAPITULO VI.....	108
SEGURIDAD DE SISTEMAS.....	108
6.1 Seguridad en el cliente.....	108
6.2 Seguridad en el servidor.....	108
6.3 Seguridad en la comunicación.....	109
6.4 Seguridad en la aplicación.....	110

CAPITULO VII.....	112
CONCLUSIONES Y RECOMENDACIONES.....	112
6.1 Conclusiones.....	112
6.2 Recomendaciones.....	113
FUENTES DE INFORMACIÓN.....	114
ANEXOS.....	117

INDICE DE FIGURAS

Figura 2.1: Organigrama.....	11
Figura 2.2: Ciclo de vida del cascada.....	13
Figura 2.3: Ciclo de vida del método iterativo.....	14
Figura 2.4: Fases y Disciplinas de AUP.....	17
Figura 2.5: Flujo de trabajo de la disciplina del modelado.....	21
Figura 2.6: Flujo de trabajo de la disciplina de la implementación.....	23
Figura 2.7: Flujo de trabajo de la disciplina de Pruebas.....	25
Figura 2.8: Flujo de trabajo de la disciplina de administración de la configuración.....	27
Figura 2.9: Flujo de trabajo de la disciplina de Administración de proyecto.....	28
Figura 2.10: Flujo de trabajo de la disciplina de entorno.....	31
Figura 2.11: Caso de uso UWE.....	34
Figura 2.12: Modelo de Contenido UWE.....	35
Figura 2.13: Modelo de navegación UWE.....	35
Figura 2.14: Modelo de presentación UWE.....	36
Figura 2.15: Modelo de Procesos UWE.....	37
Figura 2.16: Modelo de la arquitectura de un sistema mediante vistas.....	38
Figura 2.17: Codificación estándar de activos fijos.....	40
Figura 2.18 Características y sub características de medición web.....	49

Figura 2.19 Confiabilidad, características y atributos.....	52
Figura 2.20: Corte y Cambio.....	60
Figura 2.21: Operación Paralela.....	61
Figura 2.22: Operación Piloto.....	62
Figura 2.23: Operación de fase.....	62
Figura 3.1: Diagrama de casos de uso del negocio.....	64
Figura 3.2: Diagrama de caso de uso: Control y Gestión de activos fijos vía web...	68
Figura 3.3: Diagrama de caso de uso: Administrar Empleado.....	69
Figura 3.4: Diagrama de caso de uso: Administrar usuario.....	70
Figura 3.5: Diagrama de caso de uso: Control y Seguimiento de los Activos Fijos.	71
Figura 3.6: Diagrama de caso de uso: Administrar Activos Fijos.....	73
Figura 3.7: Diagrama de caso de uso: Administrar Asignación.....	74
Figura 3.8: Diagrama de caso de uso: Depreciación de Activos Fijos.....	75
Figura 3.9: Diagrama de caso de uso: Baja de Activos Fijos.....	76
Figura 3.10: Diagrama de clases: Control y gestión de activos fijos.....	78
Figura 3.11: Modelo Conceptual: Control y Gestión de Activos Fijos.....	79
Figura 3.12: Modelo de espacio de navegación.....	80
Figura 3.13: Modelo de presentación página maestra.....	81
Figura 3.14: Autenticación del sistema.....	82
Figura 3.15: Pantalla principal.....	83
Figura 3.16: Pantalla registro de empleado.....	84
Figura 3.17: Pantalla registro de inventario de activos fijos.....	85

Figura 3.18: Pantalla registro de asignación de activos fijos.....	86
---	----

INDICE DE TABLA

Tabla 2.1: Disciplina del AUP.....	19
Tabla 2.2: Fases del AUP.....	20
Tabla 2.3: Fases de la disciplina del modelado.....	23
Tabla 2.4: Fases de la disciplina de la implementación.....	24
Tabla 2.5: Fases de la disciplina de pruebas.....	26
Tabla 2.6: Fases de la disciplina de administración de la configuración.....	28
Tabla 2.7: Fases de la disciplina de administración de proyectos.....	30
Tabla 2.8: Fases de la disciplina de entorno.....	32
Tabla 2.9: Usabilidad.....	50
Tabla 2.10: Funcionalidad.....	51
Tabla 2.11: Confiabilidad.....	52
Tabla 2.12: Eficiencia.....	53
Tabla 2.13 Tabla de Constantes para Calcular Distintos aspectos de Costes.....	55
Tabla 3.1: Descripción de actores de caso de uso del negocio.....	65
Tabla 3.2: Descripción de casos de uso: Administración de personal.....	70
Tabla 3.3: Descripción de caso de uso: Administración de usuario.....	71
Tabla 3.4: Descripción de caso de uso: Control y Seguimiento de Activos Fijos.....	74
Tabla 3.5: Descripción de caso de uso: Administrar Activos Fijos.....	73

Tabla 3.6: Descripción de caso de uso: Administrar Asignación.....	75
Tabla 3.7: Descripción de caso de uso: Depreciación de Activos Fijos.....	76
Tabla 3.8: Descripción de caso de uso: Baja de Activos Fijos.....	77
Tabla: 3.9 Prueba de aceptación: Registro y Actualización de Productos.....	87
Tabla 3.10 Prueba de aceptación: Registro de Empleados.....	88
Tabla 3.11 Prueba de aceptación: Asignación de Activos Fijos.....	89
Tabla 3.12 Prueba de aceptación: Depreciación de Activos Fijos.....	89
Tabla 4.1: Resultado de la calidad del sistema Web. Funcionalidad.....	92
Tabla 4.2 Resultado de calidad del sistema Web. Fiabilidad.....	92
Tabla 4.3 Resultado de calidad del sistema Web. Usabilidad.....	94
Tabla 4.4 resultados de calidad del sistema Web. Eficiencia.....	94
Tabla 4.5: Resumen de Resultados.....	98
Tabla 5.1: Calculo de punto función no ajustado.....	100
Tabla 5.2: Calculo de punto función ajustada.....	101
Tabla 5.3 Conversión de puntos Función a KLDC.....	102
Tabla 5.4: Coeficientes A_b y B_b	102
Tabla 5.5: Costo de elaboración del proyecto.....	104
Tabla 5.6: Costo total del proyecto.....	104

MARCO REFERENCIAL

1.1 Introducción

La tecnología y los sistemas de información toman un lugar importante en los procesos administrativos dentro de las distintas instituciones empresariales, viéndose en la necesidad de automatizar muchos procesos que en muchos casos se los realiza manualmente con resultados poco satisfactorios.

La evolución de las tecnologías asociadas a la información, hace que nuestra sociedad este cada día más conectada electrónicamente. Las labores que tradicionalmente eran realizadas manualmente, ahora son realizadas por medio de sistemas de información.

Las redes informáticas es uno de los medios más eficientes para mantener la información actualizada ya que pueden compartir información entre varias computadoras y usuarios simultáneamente y de esta forma disponer de información oportuna que ayudara al mejor funcionamiento de la empresa. En la actualidad las empresas públicas y privadas necesitan tener medios tecnológicos para mantener su información de manera óptima y oportuna para la toma de decisiones.

Al referirnos a los Activos Fijos podemos encontrar muchos desafíos, estos se deben afrontar de manera correcta al realizar el control y seguimiento del mismo, tomando en cuenta los diversos factores.

Los Activos Fijos, sufren de méritos precisamente por el uso que se le da y por el paso del tiempo. A esta pérdida de valor de los bienes se les conoce como depreciación y en materia contable, dicha depreciación deberá reconocerse como un gasto, y en materia fiscal como una deducción autorizada. Independientemente del enfoque, contable o fiscal que se le dé a la pérdida de valor de los bienes. Es muy importante considerar dicha depreciación o deducción, toda vez que el vender un Activo Fijo es considerado, el

valor de la compra menos el demerito sufrido en un lapso de tiempo, nos dará un valor actual aproximado de lo que se supone el bien en cuestión debe costar, desde que se adquirió y hasta que el mismo sea enajenado.

De tal modo que la falta de un sistema informático es el motivo inicial del presente trabajo, en la empresa “GRUPO LARCOS INDUSTRIAL LTDA” con la implementación de un CONTROL Y GESTIÓN DE ACTIVOS FIJOS VÍA WEB que proporcionará mayor exactitud e información actualizada de los Activos Fijos de la Empresa, una herramienta útil para la asignación, control y a su vez tener información fresca y actualizada de la depreciación de los Activos Fijos.

1.2 Antecedentes

1.2.1 Antecedentes de la Empresa

“GRUPO LARCOS INDUSTRIAL” es una empresa que se creó con la finalidad de brindar soluciones industriales en: Automatización, Eléctrica y Mecánica para mejorar la industria de empresas Publicas y Privadas.

La Empresa GRUPO LARCOS INDUSTRIAL LTDA, cuenta con tres sucursales en tres ciudades importantes ubicadas en, La Paz av. Roma zona de Obrajes, Cochabamba av. Blanco Galindo Km 3 y Santa Cruz av. Virgen de Cotoca entre tercer anillo interno y externo. Con la finalidad de poder brindar la mayor cobertura en soluciones industriales acorde con las necesidades que demanda la industria.

En la actualidad la empresa no cuenta con una unidad de Activos Fijos, la Gerencia de Operaciones se encarga de procesar las solicitudes de compra o asignación de Activos Fijos de las distintas sucursales, que a su vez se apoya con la unidad de Contabilidad para el registro de los Activos Fijos. Todo el proceso es realizado de forma manual, en hojas de cálculo Excel e impresas para ser archivadas, ocasionando dificultades como ser:

- La demora en la Búsqueda de información

- Dificultades para el control y asignación de Activos Fijos
- Dificultades para realizar la depreciación Contable de Activos Fijos para el cierre de Gestión
- Dificultades para obtener un información de cuantos Activos Fijos fueron Activados o dados de Baja, en las diferentes sucursales

1.2.2 Antecedentes del Proyecto

En la carrera de Informática de la Universidad Mayor de San Andrés (U.M.S.A.) se verifico la existencia de proyectos de grado relacionados con el Control y Gestión de Activos Fijos vía Web, dichos proyectos están orientados a la automatización de los procesos manuales.

Se verifico los siguientes proyectos de grado:

- Sistema de registro y asignación de activos fijos para la SPVS¹, elaborado por Angélica Apaza Cocarico (2004), haciendo uso de la metodología de desarrollo de software RUP, para la parte de análisis, diseño, implementación y diagrama hace uso de la herramienta case architect y como lenguaje de modelado UML.
- Sistema de información y control de activos fijos – Ministerio de Gobierno², elaborado por James Williams Gutiérrez Flores (2005), haciendo uso de la metodología de desarrollo de software RUP, el modelo de análisis y diseño de sistema se lo realiza empleando el lenguaje de modelado UML.
- Sistema de codificación Seguimiento y control de Activos fijos ECOBOL³, desarrollado por Vargas Rolan (2004); el Objetivo del proyecto es precautelar el patrimonio de la empresa para su correcto manejo y administración acatando normas y leyes vigentes, incorpora un sistema de codificación, utilización de código de barras y pone énfasis en el cálculo de depreciación de activos.

¹ Apaza Cocarico Angélica. 2004

² Gutiérrez J. 2005

³ Vargas R. 2004

- Sistema de control de Activos Fijos hospital del niño⁴, desarrollado por Gonzalo Acarapi Apaza (2008); aplica la teoría de colas para el mantenimiento o reparación de activos y emplea la metodología orientada a objetos para el análisis y diseño.

1.3 Planteamiento del Problema

1.3.1 Problema Principal

La Empresa “GRUPO LARCOS INDUSTRIAL LTDA.” en la actualidad no cuenta con sistema información alguno que ayude a la gestión y control administrativo que proporcione información.

¿De qué manera la empresa “Grupo Larcos Industrial LTDA.” . Puede realizar un control de todos los Activos Fijos de forma precisa, confiable y actualizada. ?

1.3.2 Problemas Específicos

- No se tiene un inventario adecuado de los Activos Fijos, ocasionando demora en la búsqueda de información y pérdida de estos datos.
- No cuenta con información confiable los Activos Fijos distribuidos en las diferentes sucursales, ocasionando desinformación.
- No se cuenta con información precisa de la depreciación de los Activos Fijos, ocasionando la demora para el cierre de gestión.
- No se cuenta con información de la revalorización de los activos fijos, ocasionando gastos innecesarios para la compra de Activos Fijos.
- No existe un registro de los movimiento de los Activos Fijos, ocasionando la pérdida de la misma
- No se tienen un registro del estado de los Activos Fijos, generando la compra

⁴ Acarapi G. 2008

innecesaria.

- Se desconoce la asignación de los Activos Fijos, ocasiona por la falta de control

1.4 Objetivos

1.4.1 Objetivos Principal

“Desarrollar un sistema información web para el control y Gestión de Activos Fijos para la empresa GRUPO LARCOS INDUSTRIAL LTDA.”

Diseñar y desarrollar un sistema web de Control y Gestión, de tal manera que permita mejorar el control de los activos fijos de la Empresa “GRUPO LARCOS INDUSTRIAL LTDA.”, proporcionando la información de forma precisa, confiable y actualizada.

1.4.2 Objetivos Específicos

- Automatizar el registro de los inventarios de los Activos Fijos
- Realizar un módulo para obtener reporte de los Activos Fijos por sucursales
- Realizar un módulo para realizar la depreciación y generar reportes de los Activos Fijos
- Realizar un módulo para la revalorización de los Activos Fijos
- Automatizar el registro de los movimientos de los activos fijos
- Generar informes del estado de los Activos Fijos
- Automatizar la asignación de los Activos Fijos

1.5 Justificación

1.5.1 Justificación Económica

Con la implementación del sistema de información se obtendrá información confiable, rápida y actualizada para el control, al realizar el registro de activos fijos en

las sucursales, lo cual reducirá el tiempo en la búsqueda de la información requerida y aumentara la seguridad de información de la empresa.

Se justifica económicamente el proyecto porque proporcionara beneficios a corto y largo plazo que se observara a través del control y seguimiento, teniendo como resultado una reducción de compras innecesaria en las sucursales, permitiendo generar y consultar información en tiempo real obteniendo información confiable e inmediata, lo que implica un ahorro económico.

1.5.1 Justificación Social

La implementación del sistema de información web beneficia a la gerencia de operaciones en el control, seguimiento y asignación de los Activos Fijos, a la unidad de Contabilidad en la obtención de informes de depreciación actualizada Activos Fijos. Debido que el sistema les favorece en el mejoramiento de los procesos o tareas, permitiendo una distribución eficiente de bienes a los empleados, para su mejor rendimiento laboral.

1.5.3 Justificación Técnica

Se justifica técnicamente, porque la empresa GRUPO LARCOS INDUSTRIAL LTDA cuenta con las herramientas tecnológicas suficientes para implementar el sistema de información vía Web para el control y seguimiento de Activos Fijos. La modernización en el manejo de la información es indispensable para poder mejorar las actividades que se realizan.

Los componentes técnicos como tecnológicos que se están empleando en el desarrollo del software como por ejemplo computadoras, Internet y lenguajes de

programación impulsaran no solo al mejor tratamiento de información si no a la innovación tecnológica que nos permitirá diferenciarnos de las otras empresas.

Se empleará la tecnología Cliente/Servidor para compartir o distribuir la información de la Base de Datos que esté disponible en cualquier computador conectado a la red de Intranet, utilizando medidas de seguridad para que esta información pueda ser accedida solo por usuarios autorizados, además de usar una interfaz adecuada y fácil de operar por cualquier usuario.

1.6. Alcances y Limites

1.6.1 Alcances

El sistema de información vía Web para el control y gestión de activos fijos se implementará en la empresa GRUPO LARCOS INDUSTRIAL LTDA y en las sucursales con las que cuenta, el sistema funcionará en tres ciudades La Paz, Cochabamba y Santa Cruz

El sistema contara con los siguientes módulos:

- ❖ Módulo de registro de Inventarios
- ❖ Módulo de registro del personal de trabajo.
- ❖ Módulo de Asignación de Activos Fijos
- ❖ Módulo de registro de Sucursales
- ❖ Módulo de Depreciaciones y Bajas de Activos Fijos
- ❖ Módulo de la Revalorización de los Activos Fijos
- ❖ Consultas de existencias de Activos Fijos por sucursales

1.6.2. Limites

En la actualidad muchas empresas enfrentan problemas de diversa naturaleza con distintos grados de complejidad, que no necesariamente pueden ser solucionados a través de un sistema informático.

Los límites del presente proyecto son los siguientes:

- La asignación de los Activos Fijos será solo por el personal autorizado
- El Sistema no registra las facturas emitidas por la compra de un Activo Fijo
- El Sistema no realiza cotizaciones para la compra de los Activos Fijos
- El Sistema no registra a los Activos Fijos con códigos de Barra o QR

1.7. Aportes

Ofrecer un software que ayude al control y seguimiento, capaz de brindar información real, confiable y oportuna, que controle y gestione los activos fijos, siendo una herramienta de gran importancia.

De manera más específica, una vez implantado el sistema, tanto en la oficina central como en las sucursales, se pretende:

- Mejorar la comunicación entre sucursales ya que la transmisión de la información se realiza en un entorno de red (internet).
- Agilizar la asignación, con la información necesaria de los Activos Fijos disponibles.
- Conseguir información confiable y segura, ya que el sistema asegura la veracidad de la misma al implementar mecanismos de control.

De la misma manera el sistema, servirá también para la agilización de ciertos procesos manuales como el registro de inventario, asignación y la depreciación de los

Activos Fijos, generando informes oportunos y confiables mediante los diferentes módulos, que se utilizaran para la toma de decisiones.

Aplicación de la metodología ágil **Proceso Unificado Ágil AUP** que es una metodología de desarrollo de software que se basa en la simplicidad, la comunicación y la retroalimentación o reutilización del código como guía para la utilización e investigación de la misma.

Asimismo el sistema contribuirá con la aplicación del modelado web: **UWE** que es una notación visual para el diseño de aplicaciones Web complejos que usan datos intensivamente. Provee especificaciones gráficas formales para un proceso de diseño completo que puede ser asistido por herramientas de diseño visual.

1.8 Metodología

La metodología es la guía que nos va indicando que hacer y cómo actuar cuando se quiere obtener una investigación. Es posible decir que la metodología es aquel enfoque que permite observar un problema de una forma total, sistemática y disciplinada.

En cuanto a la obtención y desarrollo del sistema de información, se basara en una de las metodologías ágiles proceso Unificado Ágil denotado por el acrónimo AUP que se adecua mejor en la consecuencia del producto final. En cuanto a la herramienta de diseño se utilizara la propuesta de Ingeniería Web Basada en UML UWE es una herramienta detallada para el proceso de autoría de aplicaciones con una definición exhaustiva del proceso de diseño que debe ser utilizado.

Método de investigación que realizaremos en este trabajo en el método científico, que es un proceso destinado a explicar fenómenos físicos del mundo y permitan obtener, con estos conocimientos, aplicaciones útiles al hombre.

El tipo de investigación en principio será del tipo explicativo, ya que se efectúa sobre un tema u objeto desconocido o poco estudiado, por lo que sus resultados constituyen una visión aproximada de dicho objeto, es decir un nivel superficial de conocimiento. Después será una investigación descriptiva, ya que se conocerá las situaciones y actitudes predominantes a través de la descripción exacta de las actividades, objetos, procesos y personas. Se recogen los datos sobre la base de una hipótesis, luego se analiza minuciosamente los resultados, a fin de extraer generalizaciones significativas que contribuyan al conocimiento.



MARCO TEÓRICO

2.1 Marco Institucional

Visión.-

Constituirse en la empresa más importante del país, como proveedor estratégico de equipos y soluciones industriales para sus sectores productivos con un alto conocimiento técnico y comercial, contando con un personal comprometido con sus objetivos y valores.

Misión.-

Proveer adecuadamente equipos industriales y asesoramiento técnico a las unidades productivas del país, con asistencia de calidad y la difusión del conocimiento especializado involucrado.

2.1.1. Organigrama.-

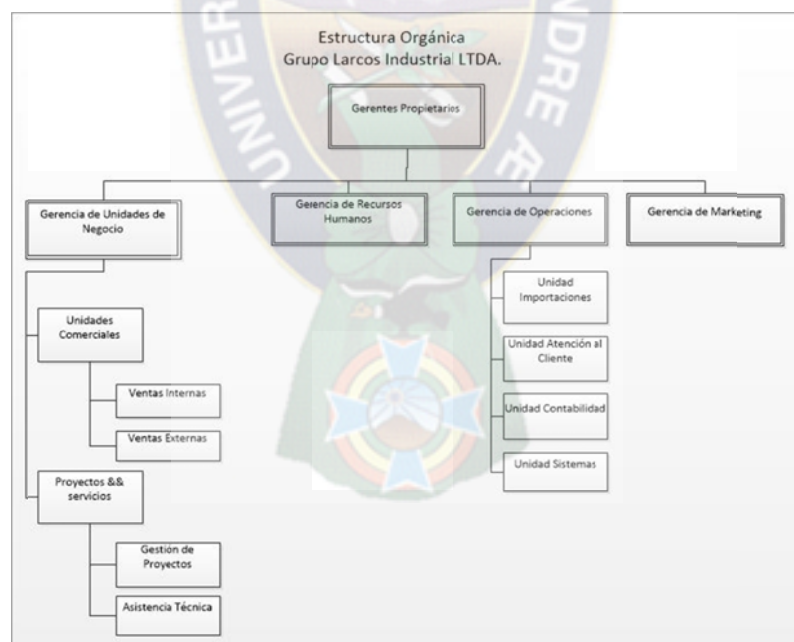


Figura 2.1: Organigrama
[Fuente: Grupo Larcos Industrial LTDA.]

2.1.1.1 Gerencia de Operaciones

La gerencia de Operaciones se encarga de:

- Verifica el funcionamiento óptimo de las unidades administrativas
- Controlar las comisiones de ventas y manos de obra de Vendedores y Técnicos
- Verifica o controlar las solicitudes de Compra de Activos Fijos de las distintas sucursales.
- Asigna Activos Fijos a los empleados de las distintas sucursales

2.1.1.2 Unidad de Contabilidad

La unidad de Contabilidad se encarga de:

- Verificar los estados financieros de la empresa
- Realiza la depreciación de los Activos Fijos
- Realiza la Baja de Activos Fijos, por obsolescencia o deterioro
- Realiza la Revalorización de Activos Fijos

2.2 Metodología

Metodología de desarrollo de software en ingeniería de software es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información. Además tiene como objetivo presentar un conjunto de técnicas tradicionales y modernas de modelado de sistemas que permitan desarrollar software de calidad, incluyendo heurísticas de construcción y criterios de comparación de modelos de sistemas.

En las metodologías de desarrollo de software se tiene los métodos: iterativos, evolutivos y ágiles.

2.2.1 Modelo en Cascada

El modelo en cascada [Pressman, 2006], algunas veces llamado el ciclo de vida clásico, sugiere un enfoque sistémico, secuencial hacia el desarrollo del software, que se inicia con la especificación de requerimientos del cliente y que continúa con la planeación, el modelado, la construcción y el despliegue para culminar en el soporte del software terminado.

El número de etapas suele variar, pero en general suelen ser:

- Análisis de requisitos del sistema
- Análisis de requisitos del software
- Diseño preliminar
- Diseño detallado
- Codificación y pruebas
- Explotación (u operación) y mantenimiento

Las características de este modelo son:

- Cada fase empieza cuando se ha terminado la anterior.
- Para pasar a la fase posterior es necesario haber logrado los objetivos de la previa.
- Es útil como control de fechas de entregas.
- Al final de cada fase el personal técnico y los usuarios tienen la oportunidad de revisar el progreso del proyecto.



Figura 2.2: Ciclo de vida del modelo en cascada
[Fuente: Lehman, M.M.]

2.2.2 Método Iterativo

Las etapas son las mismas que en el ciclo de vida en cascada y su realización sigue el mismo orden, pero corrige la problemática de la linealidad del modelo en cascada. Este modelo incremental fue desarrollado por Lehman [1984].

En cada paso sucesivo se agregan al sistema nuevas funcionalidades o requisitos que permiten el refinado a partir de una versión previa.

Este modelo es útil cuando la definición de los requisitos es ambigua e imprecisa, porque permite el refinamiento, o sea se pueden ampliar los requisitos y las especificaciones derivadas de la etapa anterior.

Uno de los problemas que se puede presentar es la detección de requisitos tardíamente, siendo su corrección tan costosa como en el caso de la cascada. Un gráfico que muestra cual es el desarrollo de este proceso es el siguiente.



Figura 2.3: Ciclo de vida del método iterativo
[Fuente: Lehman, M.M.]

2.2.3 Método Evolutivo

Los modelos evolutivos son iterativos. Se caracterizan por la forma en que permiten a los ingenieros del software desarrollar versiones cada vez más completas del software.

El uso de prototipos evolutivos se centra en la idea de ayudar a comprender los requisitos que plantea el usuario, sobre todo si este no tiene una idea muy acabada de lo que desea. También pueden utilizarse cuando el ingeniero de software tiene dudas acerca de la viabilidad de la solución pensada.

Esta versión temprana de lo que será el producto, con una funcionalidad reducida, en principio, podrá incrementarse paulatinamente a través de refinamientos sucesivos de las especificaciones del sistema, evolucionando hasta llegar al sistema final.

Al usar prototipos, las etapas del ciclo de vida clásico quedan modificadas de la siguiente manera:

- Análisis de requisitos del sistema
- Diseño, desarrollo e implementación del prototipo
- Prueba del prototipo.
- Refinamiento iterativo del prototipo
- Refinamiento de las especificaciones del prototipo
- Diseño e implementación del sistema final
- Explotación (u operación) y mantenimiento

Si bien el modelo de prototipos evolutivos, fácilmente modificables y ampliables es muy usado, en muchos casos pueden usarse prototipos descartables para esclarecer aquellos aspectos del sistema que no se comprenden bien.

2.2.4 Método Ágil

El desarrollo ágil de Software a un paradigma de Desarrollo de Software basado en procesos ágiles.

Los procesos ágiles de desarrollo de software, conocidos anteriormente como metodologías livianas, intentan evitar los tortuosos y burocráticos caminos de las metodologías tradicionales enfocándose en la gente y los resultados.

Según el manifiesto del método ágil se valora:

- Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas. La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.
- Desarrollar software que funciona más que conseguir una buena documentación. La regla a seguir es no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante. Estos documentos deben ser cortos y centrarse en lo fundamental.
- La colaboración con el cliente más que la negociación de un contrato. Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.
- Responder a los cambios más que seguir estrictamente un plan. La habilidad de responder a los cambios que puedan surgir a lo largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta sino flexible y abierta.

2.2.5 Metodología Ágil AUP

El Proceso Unificado Ágil (Agile UP) es un enfoque al desarrollo de software basado en el Rational Unified Process (RUP) de IBM. El ciclo de vida de Agile UP es serial en lo grande e iterativo en lo pequeño, liberando entregables incrementales en el tiempo.

2.2.5.1 Estructura del Proceso Unificado Ágil (AUP)

El Proceso Unificado Ágil está compuesto por cuatro fases, cada una de ellas se desarrolla mediante iteraciones, las cuales consisten en reproducir el ciclo de vida en cascada a menor escala.

El Eje Horizontal representa la parte dinámica del proceso en el tiempo, las iteraciones y las metas, el eje vertical representa la parte estática del proceso donde se describen los flujos de trabajo, requisitos, análisis, diseño, implementación y prueba. Las curvas son aproximaciones de hasta donde se llevan a cabo los flujos de trabajo en cada fase.

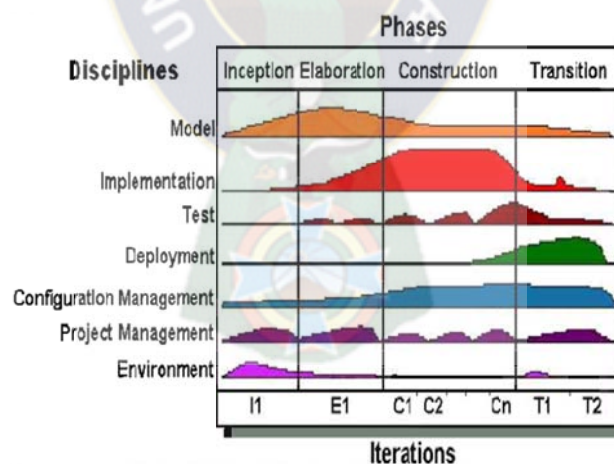


Figura 2.4: Fases y Disciplinas de AUP
[Fuente: Ambler, S., 2000]

2.2.5.2 Disciplinas de AUP

Las disciplinas son ejecutadas de una forma iterativa, definiendo las actividades, las cuales el equipo de desarrollo ejecuta para construir, validar y liberar software funcional, el cual cumple con las necesidades de los involucrados. En la siguiente tabla se puede ver las disciplinas del AUP.

Disciplina	Descripción
Modelado	La meta de ésta disciplina es entender el negocio de la organización, el dominio del problema que el proyecto aborda e identificar una solución viable para abordar el dominio del problema.
Implementación	La meta de ésta disciplina es transformar su modelo(s) en un código ejecutable y realizar una prueba de nivel básico en una unidad particular de prueba.
Pruebas	La meta de ésta disciplina es ejecutar una evaluación de los objetivos para asegurar la calidad. Esto incluye encontrar defectos, validar que el sistema función como fue diseñado y verificar que los requerimientos están completos.
Despliegue	La meta de ésta disciplina es planificar la entrega del sistema y ejecutar el plan para que el sistema esté disponible para los usuarios finales.
Administración de la Configuración	La meta de ésta disciplina es administrar el acceso a los entregables o productos del proyecto. Esto incluye no sólo el rastreo de versiones del producto en el tiempo, sino que también incluye controlar y administra los cambios que ocurran.
Administración de la Configuración	La meta de esta disciplina es dirigir las actividades que se llevan a cabo en el proyecto. Esto incluye administración del riesgo, la dirección de personas (asignar tareas, seguimiento de los procesos, etc.) y

	coordinar con los sistemas y personas fuera del alcance del proyecto para que el este termine a tiempo y dentro del presupuesto
Entorno	La meta de esta disciplina es dirigir las actividades que se lleva a cabo en el proyecto. Esto incluye administración del riesgo, la dirección de personas (asignar tareas, seguimiento de los procesos, etc.) y coordinar con los sistemas y personas fuera del alcance del proyecto para que el este termine a tiempo y dentro del presupuesto.

Tabla 2.1 Disciplina del AUP
[Fuente: Ambler. S., 2000]

2.2.5.3 Fases del AUP

Proceso unificado Ágil AUP está caracterizado por ser "serial en lo grande", contempla cuatro fases: inicio, elaboración, construcción y transición. Estas fases se muestran en la siguiente tabla y además podremos observar las metas e hitos de cada una de ellas, las fases del AUP se pueden mover en una forma serial.

Fases	Metas	Hitos
1. Inicio	Identificar el alcance inicial del proyecto, una arquitectura inicial del sistema y obtener un presupuesto inicial del proyecto y una aceptación de los involucrados.	Objetivos de Ciclo de Vida(LCO, por su siglas en inglés)
2.Elaboración	Probar arquitectura del sistema.	Arquitectura del ciclo de Vida (LCA, por su siglas en ingles)

3. Construcción	Construir un software funcional sobre una base regular e incremental, las cuales cumplan con las prioridades más importantes para los involucrados o usuarios del proyecto.	Capacidad Operacional Inicial (IOC, por su siglas en ingles)
4. Transición	Validar y desplegar el sistema en su ambiente de la producción	Liberación del Producto (PR, por su siglas en ingles)

Tabla 2.2 Fases del AUP
[Fuente: Ambler. S., 2000]

La metodología ágil AUP está caracterizada por ser serial en lo grande, algo que se ve a través de estas cuatro fases.

2.2.5.4 Disciplina de Modelado

El objetivo de esta disciplina es entender el negocio de la organización, el problema de dominio que se abordan en el proyecto, y determinar una solución viable para resolver el problema de dominio.

a) Flujo de Trabajo

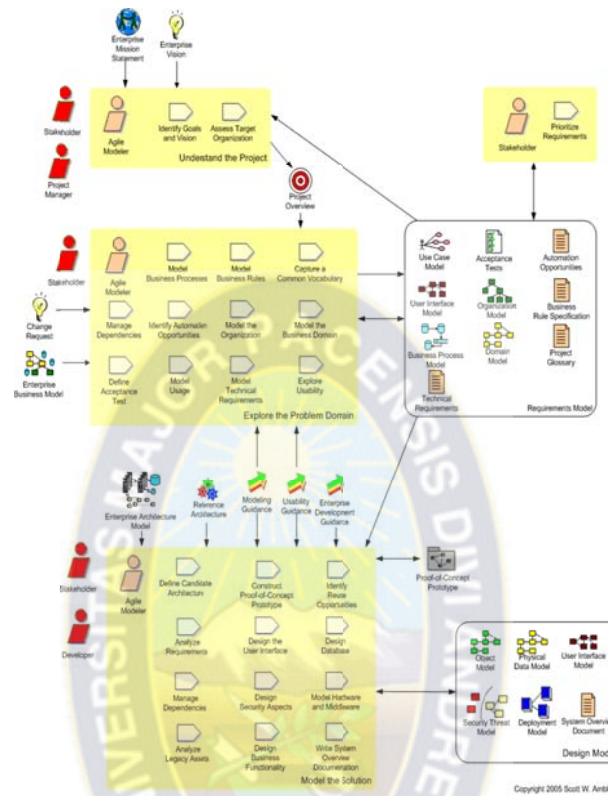


Figura 2.5: Flujo de trabajo de la Disciplina del Modelado
[Fuente: Ambler. S., 2005]

b) Fase Por Fase

Se observa las fases de la disciplina del modelado en la Tabla siguiente tabla.

Fases	Actividades
Inicio	<p>Modelado de requerimiento de alto nivel. Los interesados deben participar activamente en el modelo de requerimientos de alto nivel el cual define el alcance inicial para el proyecto y proporciona suficiente información para una estimación aproximada. Debería considerar:</p> <ul style="list-style-type: none"> ✓ Explorar el uso de Casos de Uso ✓ Identifique los procesos de negocio, para la creación de diagramas de flujo de datos. ✓ Identifique las entidades principales del negocio y sus relaciones trabajando con modelos de dominio liviano ✓ Identifique las principales reglas del negocio y requerimiento

	<p>técnicos.</p> <ul style="list-style-type: none"> ✓ Inicie el desarrollo de un glosario que describa términos importantes técnicos y del negocio. ✓ La comprensión de la estructura política dentro de su comunidad de partes interesadas a través del modelado de la organización.
Elaboración	<p>Identificar riesgos técnicos, sus necesidades de trabajo, en particular, sus casos de uso y requerimientos técnicos, ponen de manifiesto los posibles riesgos técnicos a su proyecto. Estos riesgos pueden incluir la introducción de nuevas tecnologías existentes, importante carga o estrés en su aplicación o sistemas actuales externos. La máxima prioridad debe ser abordar los riesgos por su esfuerzo de implementación en el desarrollo de un extremo a extremo del esqueleto del sistema Modelo de la Arquitectura. Como usted constituye el prototipo de la arquitectura necesitara modelar por lluvia de ideas algunos detalles para pensar pedazos de la arquitectura.</p>
Construcción	<p>Durante las iteraciones de la Construcción deberá trabajar cerca de los interesados del proyecto para entender sus necesidades. Aspectos importantes:</p> <ul style="list-style-type: none"> ✓ Participación activa de interesados y modelado inclusivo que usan técnicas y herramientas simples que son para su negocio. ✓ Si lo desea, puede profundizar en los detalles de sus casos de uso, quizá visualmente utilizando diagramas de flujo o diagramas de actividad UML, en vez de descripciones de texto. ✓ Explore las reglas del negocio y los requerimientos técnicos en la misma forma. ✓ Diagrama de secuencia de UML. Este diagrama representa la lógica dinámica dentro del código fuente. Son parte de su modelo de proyecto y usualmente se tiran lejos hasta que se tiene una buena herramienta CASE con compatibilidad para ingeniería inversa. Las pizarras son geniales herramientas para crear nuevos diagramas. ✓ Modelo de despliegue. Típicamente crear algún ordenamiento del diagrama de resumen representado en la arquitectura del sistema de despliegue. ✓ Diagrama de clases UML. Si va a hacer algún diagrama de clases use una herramienta de modelado que le permita generar código fuente. Su diagrama de clases debe estar basada en un modelo de dominio (si existe)
Transición	<p>Necesitará hacer algún modelado en el momento para tratar de entender las causas principales de un defecto. Finalice la documentación de resumen del sistema. El mejor momento para finalizar su documentación general del sistema cuando realmente</p>

	<p>está establecido.</p> <p>Realice su diseño crítico de decisiones, si lo documenta en la construcción, como una base desde la cual se construye este documento. Cualquier otra información importante que quiera en este documento es un resumen del alcance del sistema y de los diagramas de arquitectura críticos (ahora es cuando se deberá de colocar todos los diagramas de estilo libre y bocetos de pizarras en una herramienta de dibujo)</p>
--	--

Tabla 2.3: Fases de la Disciplina del Modelado

[Fuente: Ambler, S., 2000]

2.2.5.5 Disciplina de la implementación

El objetivo de esta disciplina es transformar el modelo en código ejecutable y llevar a cabo un nivel básico de las pruebas, en particular, la unidad de prueba.

a) Flujo de trabajo

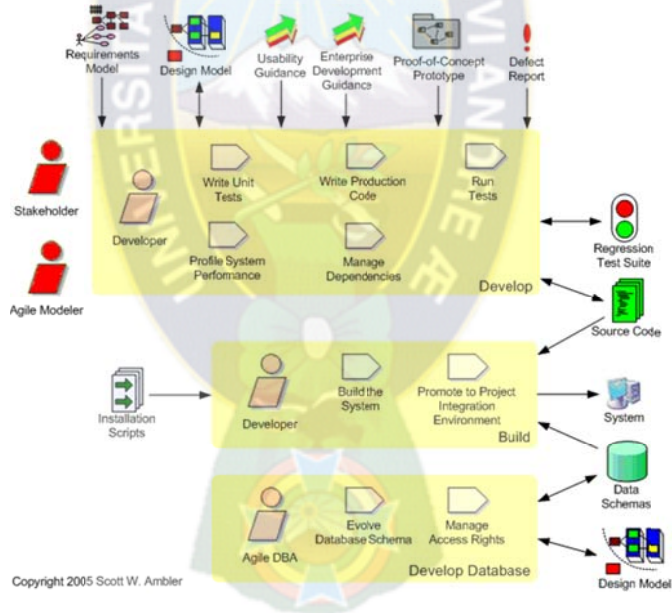


Figura 2.6: Flujo de trabajo de la Disciplina de la implementación

[Fuente: Ambler, S. 2005]

b) Fase por fase

Se observa las fases de la disciplina de la implementación en la siguiente tabla.

Fases	Actividades
-------	-------------

Inicio	Prototipo técnico, Es posible que tenga que picar un pequeño aspecto de un requisito con el fin de entender lo suficiente, lo que le permite estimar el esfuerzo requerido. Estos prototipos son típicamente pequeños.
Elaboración	Probar la arquitectura, Las actividades críticas dentro de la fase de elaboración es identificar la arquitectura potencial y luego probar que esta arquitectura funcione a través del desarrollo de la arquitectura del prototipo externo para su sistema, y a la vez mitigando gran parte de los riesgos técnicos en su proyecto. Los prototipos técnicos como son la calidad de producción de código que constituye el fundamento de su sistema.
Construcción	<p>Primeras Pruebas, Obtenga un acercamiento de la base del desarrollo dirigido por pruebas para todos los aspectos de la aplicación.</p> <p>Construya constantemente, Creaciones diarias son un buen comienzo, pero idealmente usted quiera construir su sistema cada vez que el código fuente cambie.</p> <p>Evolución de la lógica de dominio. Implemente su lógica del negocio en sus clases de negocio/dominio.</p> <p>Evolucionar las interfaces de usuario. La interface de usuario es el sistema para la mayoría de usuarios. Esfuércese por hacer su software tan usable como sea posible siguiendo las estrategias de diseño de interfaces de usuario y usabilidad.</p>
Transición	Corregir defectos. Concéntrese en la corrección de defectos encontrados como resultado de las pruebas.

Tabla 2.4: Fases de la Disciplina de la Implementación
[Fuente: Ambler, S. 2000]

2.2.5.6 Disciplina de Pruebas

El objetivo de esta disciplina es ejecutar una objetiva evaluación para asegurar la calidad. Esto incluye la detección de defectos, validaciones de que el sistema funciona como fue diseñado, y verificar que se cumplan los requerimientos.

a) Flujo de Trabajo

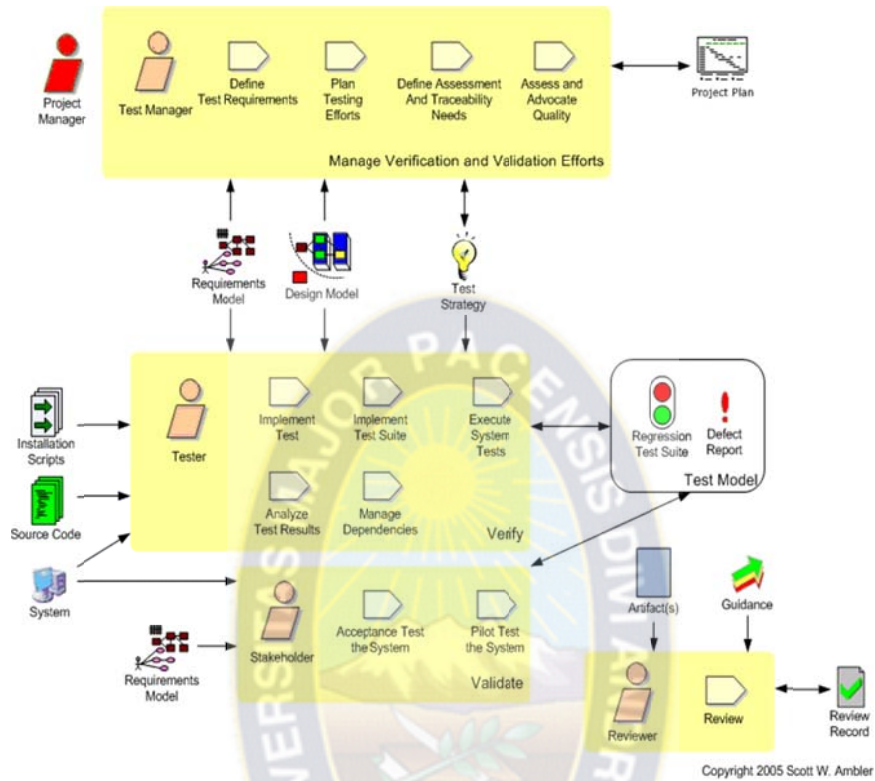


Figura 2.7: Flujo del trabajo de la Disciplina de Pruebas [Fuente: Ambler, S. 2005]

b) Fase Por Fase

Se observa las fases de la disciplina de pruebas en la siguiente tabla.

Fases	Actividades
Inicio	Planificación inicial de pruebas. Deben ser a muy alto nivel al principio. El objetivo principal es identificar cuántas pruebas necesita hacer, quien será el responsable de hacerlas, el nivel de participación requerido por los usuarios, y los tipos de herramientas y los entornos necesarios.

Elaboración	Validación de la Arquitectura. Usted debe tomar un enfoque de desarrollo controlado por pruebas para construir su prototipo técnico el cual compruebe la arquitectura de su sistema. Un aspecto importante de hito de revisión la validación de la arquitectura, que podría ser algo tan sencillo como presentar una visión general de la arquitectura y los resultados de sus esfuerzos de los prototipos para los interesados.
Construcción	Pruebas de software. Además de las unidades de prueba de los desarrolladores deberá hacer pruebas de instalación del script de despliegue o Liberación, sistema de pruebas de esfuerzos tales como la carga de pruebas de tensión y las pruebas de función, y sus pruebas
Transición	Validación del sistema. Usted se concentrará en las "grandes pruebas" de actividades tales como las del sistema. Validación de la documentación. Su Documentación del sistema y los materiales de capacitación necesitarán ser validados. Todo esto puede ser hecho por medio de las revisiones. Analice su modelo de pruebas. Va a tener que seguir ejecutando el paquete de pruebas de regresión y actualizarlo

Tabla 2.5: Fases de la Disciplina de Pruebas
[Fuente: Ambler, S., 2000]

2.2.5.7 Disciplina de administración de la configuración

La meta de esta disciplina es manejar el acceso a sus productos de trabajo de proyecto. Esta no sólo incluye el rastreo de versiones del trabajo del producto en el tiempo, sino que también el Control y administración de los cambio estos productos.

a) Flujo de trabajo

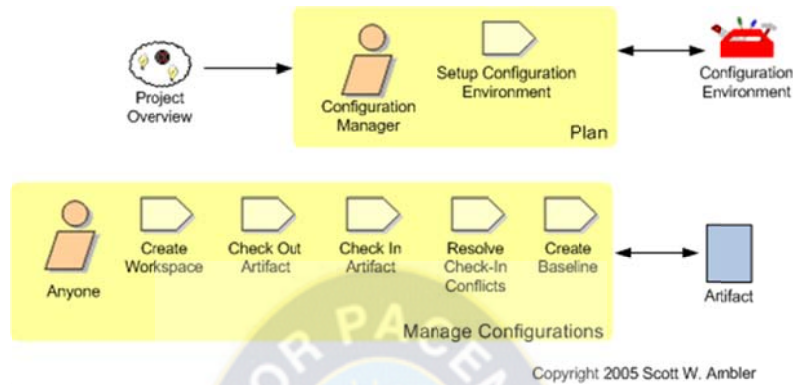


Figura 2.8: Flujo de trabajo de la disciplina de administración de la configuración
 [Fuente: Ambler, S. 2005]

b) Fase por fase

Se observa las fases de la disciplina de administración de la configuración en la siguiente tabla.

Fases	Actividades
Inicio	<p>Usted tiene que hacer varias cosas:</p> <ul style="list-style-type: none"> ✓ La estructura de directorios apropiada, la cual debe seguir los lineamientos corporativos, necesita ser creada para el equipo del proyecto. ✓ Los miembros del equipo del proyecto necesitan tener acceso al folder o directorios del proyecto. <p>Cualquier otro software del equipo del proyecto también necesita ser entrenados con los conceptos básicos de CM.</p> <ul style="list-style-type: none"> ✓ Los miembros del equipo del proyecto también necesitan ser entrenados con los conceptos básicos de CM. <p>Así como las herramientas necesarias</p> <ul style="list-style-type: none"> ✓ Su repositorio de CM necesitara ser instalado si este aún no ha instalado.

Elaboración	Poner todos los productos del proyecto sobre el control de CM.
Construcción	Poner todos los productos del proyecto sobre el control de CM.
Transición	Poner todos los productos del proyecto sobre el control de CM.

Tabla 2.6: Fases de la disciplina de administración de la configuración
[Fuente: Ambler, S. 2000]

2.2.5.8 Disciplina Administración de proyecto

El objetivo de esta disciplina es dirigir las actividades a lo largo del proyecto. Esto incluye la administración del riesgo, administración del personal (asignación de tareas, rastreo del progreso, etc.), y coordinación con personas y sistemas fuera del alcance del proyecto para asegurar su liberación a tiempo y dentro del presupuesto.

a) Flujo de trabajo

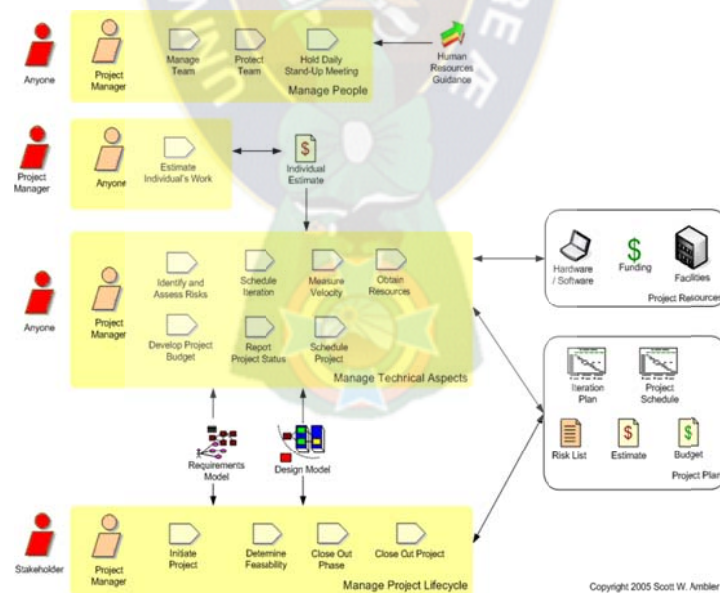


Figura 2.9: Flujo de trabajo de la disciplina de Administración de proyecto
[Fuente: Ambler, S. 2005]

b) Fase por Fase

Se observa las fases de la disciplina de administración de proyecto en la siguiente tabla.

Fases	Actividades
Inicio	<p>Inicie conformando el equipo. En este punto necesitará alguien con habilidades de modelado para trabaja con los usuarios para identificar los requerimientos iniciales del sistema y con las personas técnicas para identificar una arquitectura potencial.</p> <p>Crear relaciones con sus involucrados del proyecto. El soporte a los usuarios y la participación es crítica para su éxito.</p> <p>Desarrollar un cronograma de alto nivel para todo el proyecto. El cronograma del proyecto debe mostrar su proyecto organizado en iteraciones, indicar descripciones de los principales hitos, tareas con dependencias críticas, tanto para su equipo de trabajo como para otros equipos, y su fecha de finalización prevista.</p>
Elaboración	<p>Construya el equipo. Conforme su proyecto tome forma y crezca, necesitará agregar miembros al equipo. Durante esta fase necesitará personas con habilidades de análisis, desarrollo e implementación. Seguramente tendrá que capacitar a su equipo en las nuevas habilidades de desarrollo.</p> <p>Obtener recursos. Su equipo necesita financiación, instalaciones (por ejemplo salas y cubículos), hardware, software, y así sucesivamente para hacer su trabajo.</p> <p>Cerrar esta fase. Tendrá que ejecutar la revisión del ciclo de vida de la arquitectura, cuya principal finalidad es demostrar que su arquitectura funciona y que se está enfrentando correctamente los principales riesgos del proyecto proyectos</p>
Construcción	<p>Administre el equipo. Continúe desarrollando el equipo, manténgase protegiéndolos y proveyéndoles los recursos que necesitan.</p> <p>Manejo del riesgo. Continúe los esfuerzos de administración del riesgo.</p>

	<p>Actualizar su plan de proyecto. Durante la fase de construcción necesitará asegurar que tiene identificadas las principales dependencias involucradas en el desarrollo exitoso de su sistema. Debe considerar Las necesidades de sus equipos de operación y soporte, capacitación del usuario final, y el plan de pruebas al sistema piloto.</p> <p>Cerrar esta fase. Tendrá que ejecutar La revisión de la capacidad operativa inicial, cuya principal finalidad es demostrar que su equipo ha desarrollado un sistema que está potencialmente listo para implementarse en producción.</p>
Transición	<p>Administrar el equipo, incluye el equipo de desarrolladores, de pruebas e implementadores.</p> <p>Cerrar esta fase. Tendrá que ejecutar la revisión de los productos entregables, cuya principal finalidad es demostrar que su sistema ha pasado las pruebas y es aceptable para los involucrados.</p> <p>Iniciar el próximo ciclo del proyecto. Los sistemas se desarrollan y se ponen en producción de manera incremental. Durante la fase de transición del entregable N, deberá comenzar los primeros esfuerzos del entregable N+1.</p>

Tabla 2.7: Fases de la Disciplina de Administración de Proyectos
[Fuente: Ambler, S. 2005]

2.2.5.9 Disciplina del entorno

El objetivo de esta disciplina es soportar el resto del esfuerzo asegurando que el proceso apropiado, las guías (normas y directrices), y herramientas (hardware y software) estén disponibles para cuando el equipo las necesite.

a) Flujo de trabajo

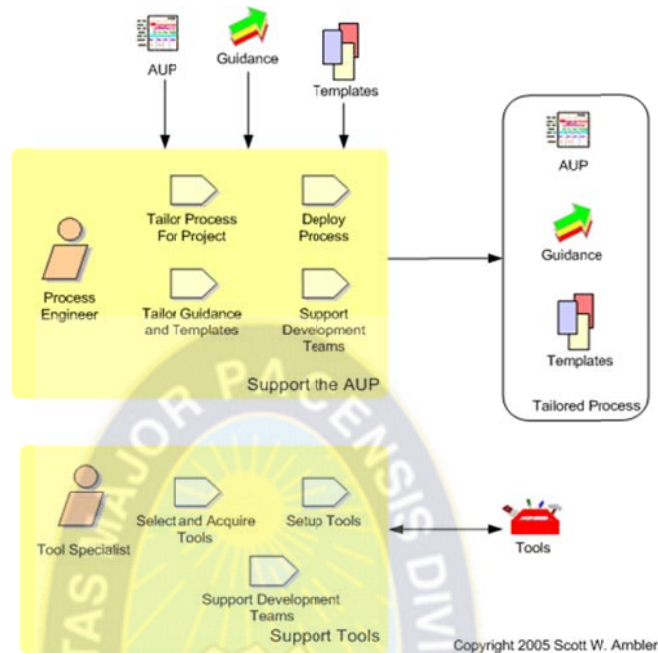


Figura 2.10: Flujo de Trabajo de la disciplina de entorno
[Fuente: Ambler, S. 2005]

b) Fase por fases

Se observa las fases de la disciplina de entorno en la siguiente tabla.

Fases	Actividades
Inicio	<p>Configure el entorno de trabajo. Esta será una tarea permanente ya que hay gente que se añade al equipo en el tiempo.</p> <p>Identifique la categoría del proyecto. Muchas organizaciones desarrollan varias versiones de sus procesos de software, por ejemplo uno para equipos pequeños, uno para reemplazar sistemas legales, otro para sistemas de plataforma comercial, etc. Esto brinda un punto de partida para ajustar la AUP a cumplir con las necesidades para cada proyecto porque han ocurrido muchos ajustes comunes.</p>

<p>Elaboración</p>	<p>Evolucionar el entorno de trabajo. Su proyecto progresa a medida que se entiende la evolución de los requisitos, la estrategia de La arquitectura, y su enfoque general. El resultado es que necesitará evolucionar su entorno instalando nuevas herramientas, o remover las herramientas que ya no necesita.</p> <p>Ajuste de los procesos de materiales. Se debe ajustar AUP para cumplir con las necesidades del equipo. Esto puede incluir materiales del proceso de AUP (por ejemplo esta página), se debe elegir entre escribir un documento corto que no se quiere hacer, o se debe simplificar eligiendo la cosa correcta en el momento correcto.</p>
<p>Construcción</p>	<p>Apoyar al equipo. Miembros del equipo del proyecto necesita ayuda para utilizar y 1 o la configuración de diversas herramientas para satisfacer sus necesidades. También se necesita ayuda para elegir las plantillas de la documentación adecuada y seguir la guía de su empresa.</p> <p>Establecer el ambiente de capacitaciones. A medida que progrese en el plan de despliegue o liberación se debe descubrir que se necesita entrenar al usuario, personal de soporte y el personal de operación. Este esfuerzo de capacitación debe requerir espacios de entrenamiento y versiones de entrenamiento del sistema disponibles, frecuentemente en la fase de Transición.</p>
<p>Transición</p>	<p>Configuración de las operaciones y soporte de los entornos. Personal de soporte, y algunas veces personal de operación, frecuentemente se necesita una versión del sistema configurada que se use para simular reportes de defectos en una forma segura.</p> <p>Recobrar licencias de software. A medida que su proyecto llega a la conclusión puede ser necesario desinstalar las licencias de software los equipos que ya no necesitan el software para que las licencias puedan estar disponibles a los demás dentro de su organización.</p>

Tabla 2.8: Fases de la Disciplina de Entorno
[Fuente: Ambler, S. 2005]

2.3 Metodología de modelado UWE

UWE (UML-Based Web Engineering) es una propuesta basada en UML y en el proceso unificado para modelar aplicaciones web.

Esta propuesta está formada por una notación para especificar el dominio (basada en UML) y un modelo para llevar a cabo el desarrollo del proceso de modelado. Los sistemas adaptativos y la sistematización son dos aspectos sobre los que se enfoca UWE.

Además de estar considerado como una extensión del estándar UML, también se basa en otros estándares como por ejemplo: XMI como modelo de intercambio de formato, MOF para el meta modelado, los principios de modelado de MOA, el modelo de transformación del lenguaje XML.

2.3.1 UWE y su relación con UML

UWE define una extensión del Lenguaje Unificado de Modelado (UML). Ésta, es considerada como una extensión ligera de peso e incluye en su definición tipos, etiquetas de valores y restricciones para las características específicas del diseño Web, las cuales, unidas a las definiciones de UML forman el conjuntos de objetos de modelado que se usarán para el desarrollo del modelo utilizado en UWE.

Las funcionalidades que cubren UWE abarcan áreas relacionadas con el Web como la navegación, presentación, los procesos de negocio y los aspectos de adaptación.

Una de las ventajas de que UWE extienda el estándar UML es la flexibilidad de éste para la definición de un lenguaje de modelado específico para el dominio web y sobretodo la aceptación universal de dicho estándar en el campo de la ingeniería del software.

Otra gran ventaja es que actualmente existen múltiples de herramientas CASE basadas en UML, con lo cual es relativamente sencillo su utilización y ampliación para utilizar los objetos de modelado definidos en UWE.

El modelo que propone UWE está compuesto por 6 etapas o sub-modelos:

2.3.1.1 Modelo de casos de Uso

Un Caso de Uso es una representación de una unidad discreta de trabajo realizada por un usuario (u otro sistema) usando el sistema en operación.

Se ejecuta en su totalidad o no se ejecuta nada, devolviendo algo de valor al usuario. Algunos ejemplos de casos de uso son Agregar_Pedido, Eliminar_Pedido, Modificar_Pedido, Registro_Usuario, Reportes, Dirección_libros entre otros. A continuación se puede observar un ejemplo de caso de uso

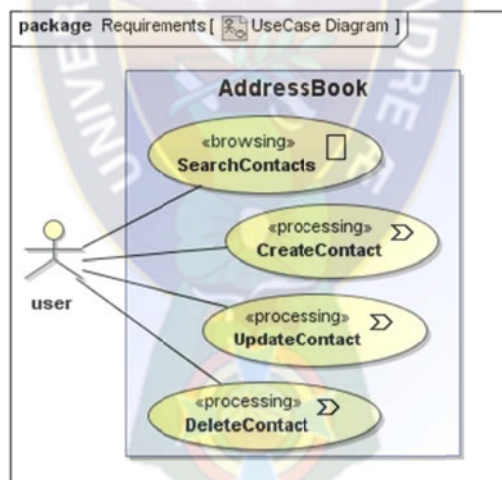


Figura 2.11: Caso de Uso UWE
[Fuente: Koch N., Kraus A., 2011]

2.3.1.2 Modelo de contenido

Este modelo especifica cómo se encuentra relacionados los contenidos del sistema, es decir, define la estructura de los datos que se encuentran alojados en el sitio

web. A continuación se muestra un ejemplo de este modelo contenido en la página web de UWE.

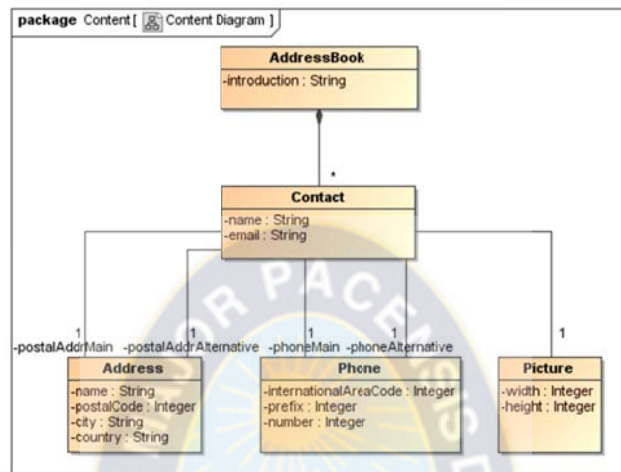


Figura 2.12: Modelo de Contenido UWE
[Fuente: Koch N., Kraus A., 2011]

2.3.1.3 Modelo de Navegación

Este modelo especifica la relación interna del sitio web, es decir cómo se relaciona cada página web con las demás, con lo cual, en definitiva es como se navega por el sitio web.

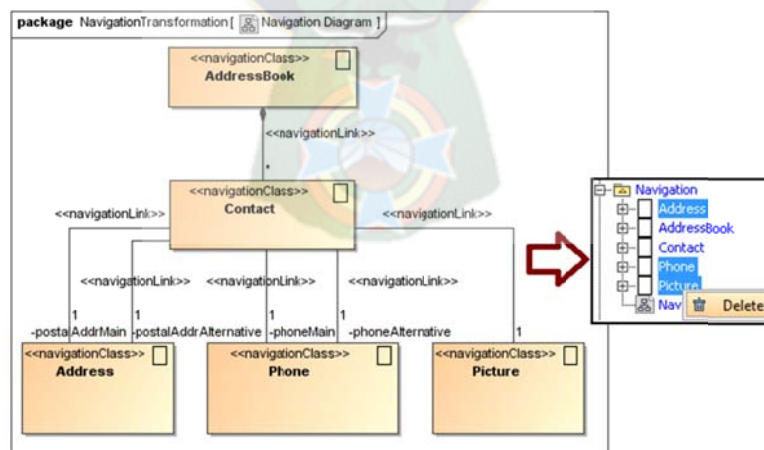


Figura 2.13: Modelo de navegación UWE
[Fuente: Koch N., Kraus A., 2011]

2.3.1.4 Modelo de Presentación

En este modelo se representan las clases de navegación y de procesos que pertenecen a cada página web. Estos son los elementos que introduce la metodología UWE en este modelo:



Figura 2.14: Modelo de presentación UWE
[Fuente: Koch kraus A., 2011]

2.3.1.5 Modelo de proceso

Este modelo especifica las acciones que realiza cada clase de proceso, en este modelo se incluye:

a) Modelo de Estructura de Procesos

Que define las relaciones entre las diferentes clases proceso. Un ejemplo de diagrama de clases de este modelo siguiendo el caso de la Agenda de contactos sería:

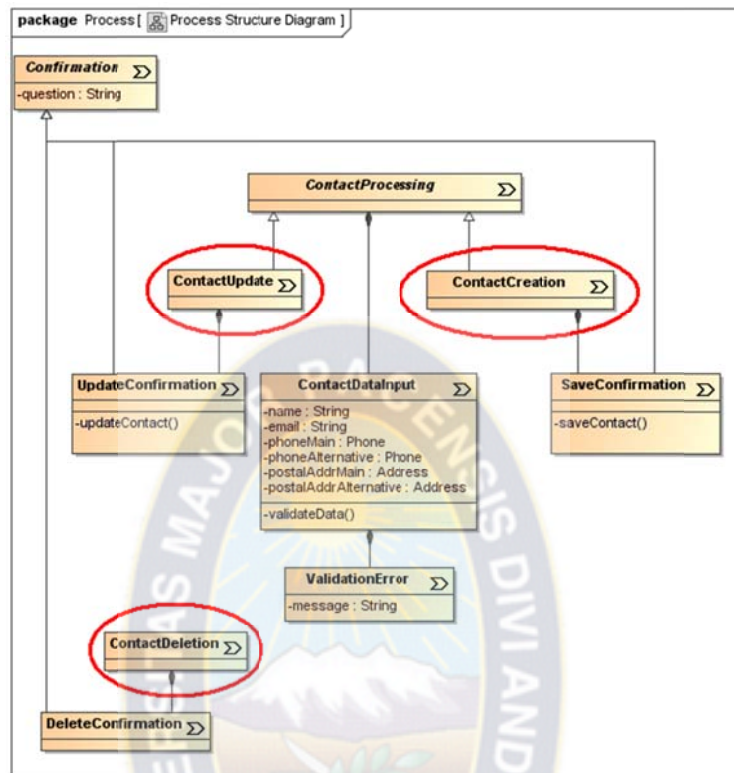


Figura 2.15: Modelo de Proceso UWE
 [Fuente: Koch N., Kraus A. 2011]

En este diagrama se puede ver que hay clases para definir 3 operaciones que necesita una confirmación.

Modelo de Flujo de Procesos: que especifica las actividades conectadas con cada proceso. Describe los comportamientos de una clase proceso. Lo que ocurre en detalle dentro de cada una.

2.4 Lenguaje unificado de modelado UML

UML es un lenguaje de modelado visual que permite especificar, visualizar, construir y documentar los componentes que forman parte de un sistema de software orientado a objetos.

Cualquier proceso software basado en UML debe contar con las siguientes características principales:

- La arquitectura debe estar basada en componentes
- Debe ser iterativo e incremental, centrándose en los aspectos críticos en las primeras iteraciones para minimizar riesgos para el futuro.
- Debe ser guiado por los requisitos (casos de uso), y preparando para identificar nuevos o para modificar requisitos a lo largo de todo el ciclo de vida.
- Tener un enfoque industrial para la producción de software: “capacidad de producir productos de alta calidad a bajo costo”
- La notación del modelo debe ser visual, facilitando la gestión de modelos, ayudando a mantener la consistencia entre los elementos del sistema y colaborando a mejorar la habilidad de equipo de desarrollo para manejar la complejidad del software.

2.4.1 Vistas UML

Una vista se usa para representar proyecciones del sistema relacionado con aspectos particulares funciones y no funcionales para el modelado de la arquitectura de un sistema. Las vistas son un conjunto de notaciones UML que modela construcciones que representa un aspecto de un sistema de distintas perspectivas con distintos diagramas, utilizando conjuntos separados de vistas.



Figura 2.16: Modelo de la arquitectura de un sistema mediante vistas
[Fuente: Alarcón. 2011]

2.4.1.1 Vista de casos de uso

Describe el comportamiento, con los diagramas de casos de uso y diagramas de actividad, que muestra la funcionalidad del sistema, es el hilo conductor de todo el proceso de desarrollo.

2.4.1.2 Vista de Diseño

Esta vista es aplicada durante la fase de diseño y desarrollo del sistema. Muestra el diseño del sistema con dos aspectos esenciales: el primer aspecto es la estructura, es decir los comportamientos que lo integran, para lo cual utiliza los diagramas de clases y objetos. El segundo aspecto es el comportamiento del sistema, es decir la dinámica de iteración de dichos componentes, utilizando diagramas de estado, secuencia, colaboración y actividad.

2.4.1.3 Vista de Procesos

Es utilizado por el grupo de desarrollo y consiste en diseñar los diagramas de componentes, Muestra la organización del código y demás archivos parte del sistema, archivos desarrollados o adquiridos, y la dependencia entre ellos.

2.4.1.4 Vista de Despliegue

Se utiliza una descripción de los nodos del sistema, que son las computadoras donde se ejecuta, y los dispositivos periféricos relevantes, utilizando diagramas de despliegue, Muestra la implantación del sistema en la arquitectura física, indicando donde están los ejecutables del sistema y como se comunican entre sí.

2.4.1.5 Vista de implementación

Es una combinación de la vista de Diseño, de procesos y despliegue. Muestra el manejo de los aspectos de afluencia en el sistema, especialmente los de comunicación y sincronización.

2.5 Activos Fijos

2.5.1 Activos Fijos o Bienes de Uso

Son aquellos bienes tangibles que se utilizan en la actividad de la empresa, que tengan una vida útil superior a un año y que no estén destinados a la venta, tales como: terrenos, edificios, muebles y enseres, vehículos, maquinaria y equipo, herramienta y equipos de computación.

Los bienes de uso (activos fijos) a diferencia de los Activos Fijos corrientes, no entran en la rotación comercial o industrial del negocio y por su naturaleza constituyen inversiones de carácter permanente que se encuentran al servicio de la empresa [FUN, 1998].

2.5.2 Codificación de Activos Fijos

Consiste en asignar un número consecutivo a un activo fijo que será la identificación del mismo durante toda su experiencia en los registros contables.

Una vez que se le asigna el número el plazo que sigue es pegarlo físicamente sobre el Activo Fijo, los que realizan esta operación es los encargados de Activos Fijos.

La codificación permitirá ver e identificar la ubicación, el destino del bien discriminado claramente un bien del otro, facilitando el recuento físico periódico con celeridad, véase la figura.

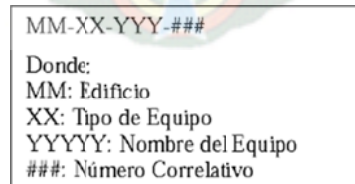


Figura 2.17: Codificación Estándar de Activos Fijos

Fuente: [Web3, 2007]

Para la codificación de mobiliario y equipo se utilizan códigos correlativos secuenciales únicos para cada Activos Fijos, porque la Empresa tiene infraestructura propia y sus dependencias se encuentran distribuidas en diferentes Sucursales.

2.5.3 Depreciación de Bienes de uso

Es la distribución del costo u otro valor básico del activo tangible, menos el valor de salvamento si hubiera, a través de la vida útil probables de la unidad y en forma sistemática y racional [FUN, 1998].

El importe de la depreciación no debe deducirse directamente del costo del activo, si no debe acreditarse a una cuenta complementaria como: “depreciación acumulada”, por dos razones que se indican a continuación:

- a) La depreciación contribuye una pérdida estimada del valor de un bien de uso (Activo Fijo) Tangible y su importe no es exacto, si no aproximado.
- b) Es el costo del Activo Fijo menos de depreciación acumulada, de esta manera el valor que refleja el balance General o Posición Financiera es el valor neto del Activo a una determinada fecha.

Calculo de depreciación: para efectuar el cálculo de la depreciación de un bien deben tomarse en cuenta lo siguiente [TER, 1997]:

- Costo actualizado del bien
- Valor de desecho (si corresponde)
- Años de vida útil estimados
- Disposición legal

- a. **Costo actualizado del bien.** El costo actualizado del bien está constituido por el precio de compra (según factura menos el IVA) más desembolsos incidentales (instalación, horarios de técnicos, prueba, etc.) más si corresponde el costo financiero (interés y comisiones sobre préstamos financieros) y adicionales la

expresión de valor en función a las fluctuaciones en la cotización del dólar Estadounidense [TER, 1997].

- b. Valor de desecho.** Conocido también como valor de salvamento a valor de rescate, es el importe mínimo de un bien después de cumplir los años de vida útil estimados, que se espera obtener al quedar fuera de servicio (por venta o baja) [TER, 1997].
- c. Años de vida útil estimados.** Constituyen el tiempo máximo de duración de los bienes nuevos, para prestar servicio, variando entre ellos por las características de cada uno de estos [TER, 1997].

2.5.4 Métodos de Depreciación

Se denomina método de depreciación a la mecánica utilizada para prorratear el costo actualizado del bien tomando en cuenta sus años de vida útil estimados, horas de trabajo, unidades de producción o cualquier otro parámetro aceptado por normas contable [TER, 1997].

2.5.4.1 Método de Línea Recta

Conocido también como método lineal o método porcentual, consiste en prorratear o distribuir el costo actualizado del bien en partes o fracciones iguales durante sus años de vida útil estimados.

$$D = \frac{C - VR}{n}$$

Dónde:

D = depreciación anual

C = costo

VR = Valor residual desecho

N = Años de vida Útil

2.5.4.2 Método de la suma de los dígitos del año

Este es un método de depreciación acelerada que busca determinar una mayor alícuota de depreciación en los primeros años de vida útil del activo.

La fórmula que se aplica es: $(\text{Vida útil}/\text{suma dígitos}) * \text{Valor activo}$

$$\frac{V(V + 1)}{2}$$

Dónde:

V = es la vida útil del activo

Donde se tiene que:

2.5.4.3 Método De La Reducción De Saldos

Este es otro método que permite la depreciación acelerada. Para su implementación, exige necesariamente la utilización de un valor de salvamento, de lo contrario en el primer año se depreciaría el 100% del activo, por lo perdería validez este método.

La fórmula a utilizar es la siguiente:

$$td = 1 - \left(\frac{Vs}{Va}\right)^{1/n}$$

Dónde:

n = es el la vida útil del activo

td = Tasa de depreciación

Vs = Valor de salvamento

Va = Valor activo

Como se puede ver, lo primero que se debe hacer, es determinar la tasa de depreciación, para luego aplicar esa tasa al valor no depreciado del activo o saldo sin de preciar

2.5.5 Baja de Activos Fijos

Se denomina baja de bienes de uso al retiro de estos por encontrarse en condiciones no aptas para prestar servicio útil a una empresa. Se puede originar tal

situación, por inclemencias climatológicas, por siniestros, por obsolescencia o por haber cumplido con su vida útil [TER, 1997].

Además, se deberá seguir el siguiente procedimiento

1. Actualizar valores hasta la fecha cuando se realiza la baja
2. Efectuar cálculos de la depreciación y su registro hasta la fecha cuando se realiza la baja.
3. Determinar el valor residual para efectuar cuenta de gasto cuando no se recupere o caso contrario cuenta de activo (por cobrar) si existe la posibilidad de recuperación.
4. Proceder a la presentación del registro contable

2.6 Tecnología de Software

Las herramientas necesarias para el desarrollo e implementación de cada iteración y entrega del sistema son las siguientes:

- Lenguaje de programación C# y ASP.NET.
- Servidor web IIS 7.0.
- Administrador de base de datos SQLSERVER 2008R2.
- Sistema operativo Windows Server 2008R2

C#.Net es un Lenguaje de programación, diseñado por Microsoft para su plataforma .NET. En los laboratorio de esta misma compañía a finales de la década de 1990 como parte de una estrategia global .NET. La primera versión alfa fue lanzada a mediados de 2000 y el diseñador encargado de su desarrollo fue Andrés Hejlsberg también es conocido como diseñador de lenguaje Turbo Pascal, C# combina múltiples lenguajes de amplia difusión como c, c++, java, visual Basic la creación de C# se basó en algunas herramientas que carecía java. Versiones estable de C# C# 1.0

ASP.NET es un modelo de desarrollo Web unificado que incluye los servicios necesarios para crear aplicaciones Web empresariales con el código mínimo. ASP.NET forma parte de .NET Framework y al codificar las aplicaciones ASP.NET tiene acceso a las clases en .NET Framework. El código de las aplicaciones puede escribirse en cualquier lenguaje compatible con el Common Language Runtime (CLR), entre ellos Microsoft Visual Basic y C#. Estos lenguajes permiten desarrollar aplicaciones ASP.NET que se benefician del Common Language Runtime, seguridad de tipos y herencia.

SQLSERVER 2008R2 es un Sistema Gestor de Bases de datos relacionales (SGBD) que además ahora en sus versiones más actuales cuenta con diferentes tipos de herramientas incorporadas en el programa, está basado en el lenguaje Transact-SQL y es capaz de poner grandes cantidades de información a muchos usuarios simultáneamente y de manera muy rápida. SQL Server 2008 es capaz de gestionar cualquier tipo de datos, en cualquier sitio y momento

Almacena datos de documentos estructurados, semi estructurados o no estructurados como imágenes, música y archivos directamente dentro de la base de datos. Se obtiene más rendimiento de los datos, poniendo a disposición servicios integrados como son consultas, búsquedas, sincronizaciones, informes y análisis.

IIS o Internet Information Services es un servidor web y un conjunto de servicios para el sistema operativo Microsoft Windows. Originalmente era parte del *Option Pack* para Windows NT. Luego fue integrado en otros sistemas operativos de Microsoft destinados a ofrecer servicios, como Windows 2000 o Windows Server 2003. Windows XP Profesional incluye una versión limitada de IIS. Los servicios que ofrece son: FTP, SMTP, NNTP y HTTP/HTTPS.2

Este servicio convierte a una PC en un servidor web para Internet o una intranet, es decir que en las computadoras que tienen este servicio instalado se pueden publicar páginas web tanto local como remotamente.

Se basa en varios módulos que le dan capacidad para procesar distintos tipos de páginas. Por ejemplo, Microsoft incluye los de Active Server Pages (ASP) y ASP.NET. También pueden ser incluidos los de otros fabricantes, como PHP o Perl.

2.7 Calidad de software

2.7.1 Definición e Implementación de la Evaluación Elemental

El modelo de evaluación WEB site QEM (Quality Evaluation Methodology o Metodología de Evaluación de Sitios Web) trata con actividades, modelos, técnicas heurísticas, y herramientas para determinar criterios de evaluación para cada atributo cuantificable y realizar el proceso de evaluación. Se consideran diferentes tipos de criterios de calidad elemental, escalas (y representación gráfica como escala de preferencia), valores, rangos críticos, y funciones para determinar la preferencia elemental, entre otros asuntos.

Una vez definidos y consensuados los criterios para medir cada atributo, se debe ejecutar el proceso de medición, es decir, la recolección de datos, el cómputo de las variables y las preferencias elementales, y la documentación de los resultados. A partir del árbol de requerimientos para cada atributo cuantificable A_i (u hoja del árbol) se debe asociar y determinar una variable X_i , que adopta un valor real a partir del proceso de medición. Además, para cada variable X_i computada, por medio de un criterio elemental, producirá una preferencia elemental IE_i . Este resultado final, elemental, se puede interpretar como el grado o porcentaje del requerimiento del usuario satisfecho para el atributo A_i .

Por definición el criterio elemental es una correspondencia del valor de la variable de calidad X_i en el valor de la preferencia (o indicador) elemental de calidad IE_i .

En este proceso se define una base de criterios para la evaluación elemental; se realiza proceso de medición, y puntaje elemental. Un criterio de evaluación elemental y se especifica cómo medir atributos cuantificables.

Donde el resultado final es una preferencia, el cual es interpretado como el grado o porcentaje del requerimiento elemental satisfecho. Para cada variable medida X_i , $i=1, \dots, n$ se define un función que representa al criterio elemental. Esta función es una correspondencia (mapeo) de los valores computados a partir del dominio empírico en el nuevo dominio numérico y se la denomina preferencia de calidad elemental (IE_i). Se asume que IE_i es el porcentaje de requerimiento satisfecho para A_i . Donde sí $IE_i=0\%$ denota una situación totalmente insatisfactoria, mientras que si $IE_i=100\%$ representa una situación totalmente satisfactoria, el puntaje elemental cae en un intervalo de aceptabilidad, es decir:

INSATISFACTORIO [0% - 40%]

MARGINAL de [40% - 60%]

SATISFACTORIO de [60% - 100%].

Existen dos tipos básicos de criterios elementales los cuales son: absolutos y relativos, dentro de los primeros se descompone en criterios con variables continuas, y criterios con variables discretas.

2.7.2. Criterios elementales Absoluto con variable continúa (a)

Criterio de variable única (CVU), es un criterio elemental común. Se asume que la variable es única y continua.

Criterio de Multi-variables Continuas (CMC), donde la variable X es resultante de algunas otras variables y constantes (el valor X corresponde a una métrica indirecta).

Criterio de Preferencia de calidad Directa (CDP), este criterio es subjetivo, basado en la experiencia y criterio de los evaluadores.

2.7.3. Criterios de elementales Absoluto con variable discreta (b)

Criterio Binario (CB), la variable binaria X se mapea en una preferencia elemental cuyas coordenadas son: $CrE(X_i) = \{(0, 0), (1, 100)\}$ En donde un valor de $X_i = 0$ se interpreta como la ausencia del atributo de calidad; en cambio un valor de $X_i = 1$, se interpreta como la presencia o disponibilidad del mismo. Este es el criterio utilizado para evaluar muchos de los atributos del árbol de requerimientos.

Criterio de Multi-nivel (CMN). Este criterio es una generalización del criterio binario. La variable discreta puede tomar más de dos valores, cada uno de los cuales se corresponde a una preferencia de calidad. La variable X se mapea en valores de preferencias cuyas coordenadas son: $CrE(X_i) = \{(0, 0), (1, 60), (2, 100)\}$.

En donde un valor de $X_i = 0$ se interpreta como la ausencia del atributo de calidad; en cambio un valor de $X_i = 1$, se interpreta como la presencia parcial de la versión sólo texto; y, finalmente, un valor de $X_i = 2$, se interpreta como la presencia total de la versión sólo texto para todo el sitio Web.

Criterio de Multi-variables discretas (CMD). Este criterio permite agrupar varias variables discretas y modelar el resultado en una única variable X . Sea el conjunto de variables discretas D_1, \dots, D_n , entonces se puede definir una variable compuesta X , también discreta, como función de las anteriores, a saber: $X = F(D_1, \dots, D_n)$, y $X \in \{X_1, \dots, X_n\}$

2.7.4 Medición Elemental

A fin de determinar el criterio elemental, se define el atributo y tipo de criterio elemental por medio de una tabla de referencia como herramienta de utilidad para

especificar información deseada de cada atributo X_i , correspondiente al árbol de requerimientos.

Para evaluar los resultados en forma global, es necesario realizar la estructura de agregación de preferencias parciales, es decir dar pesos a las distintas variables de calidad que tomamos en cuenta. Los pesos están dados en la metodología Web QEM [Olsina, 2000], utilizando las características apropiadas para un Sistema Web. Los valores son calculados según los resultados obtenidos en la evaluación elemental.

$$\text{Valor} = K_{Pi} * E_i; \text{ Donde } i = 1 \dots \text{Número de subpuntos } K_{Pi}=1$$

Ejemplo:

1.1.1 Esquema de organización global (Valor) = 1.1.1.1 Mapa del Sitio (Valor) * 0.2 + 1.1.1.2 Menú de contenidos (Valor)*0.4 + 1.1.1.3 Índice alfabético (Valor) * 0,4 = 0 * 0,2 + 100 * 0.4 + 0 * 0.4 = 40. A continuación la evaluación elemental y especificación de algunas características (Figura 2.15):



Figura 2.18 Características y sub características de medición web

[Fuente: slidefinder.net]

El resultado obtenido en la tabla 2.9 para la usabilidad tomando en cuenta las características y atributos de la interfaz de usabilidad como se ilustra en figura 2.13, representa el grado en que el software puede ser usado, comprensible, de una interfaz atractiva:

CÓDIGO	ATRIBUTO	CRITERIO ELEMENTAL	IEi(%)	Peso	Valor
1.	USABILIDAD				
1.1	Comprensibilidad global del sitio				
1.1.1	Esquema de organización global				
1.1.1.1	Mapa del sitio				
1.1.1.2	Menú de contenidos				
1.1.1.3	Índice alfabético				
1.2	Mecanismos de Ayuda y retroalimentación en línea				
1.2.1	Calidad de Ayuda				
1.2.1.1	Ayuda explicatorio Orientado al Usuario				
1.2.1.2	Ayuda de la búsqueda				
1.2.2	Indicador de última actualización				
1.2.2.1	Global todo el sitio Web				
1.2.2.2	Restringido por sub sitio o página				
1.2.3	Retroalimentación				

Tabla 2.9 : Usabilidad.
[Fuente: Olsina, 2000]

El porcentaje obtenido en la siguiente tabla 2.10 representa en qué grado el software satisface los requisitos funcionales, realizar búsquedas y navegación.

CÓDIGO	ATRIBUTO	CRITERIO ELEMENTAL	IEi(%)	Peso	Valor
2.	FUNCIONALIDAD				
2.1	Aspectos de Búsqueda y recuperación				
2.1.1	Mecanismos de búsqueda en el sitio				
2.1.1.1	Búsqueda restringida				
2.1.1.2	Búsqueda global				
2.2	Aspectos de Navegación y exploración				
2.2.1	Navegabilidad				
2.2.1.1	Orientación				
2.2.1.1.1	Indicador del Camino				
2.2.1.1.2	Etiqueta de posición actual				
2.2.1.2	Promedio de enlaces por páginas				
2.2.2	Objetos de control de navegación				
2.2.2.2	Nivel de desplazamiento				
2.2.2.2.1	Desplazamiento vertical				
2.2.2.2.2	Desplazamiento horizontal				
2.2.3	Predicción navegacional				
2.2.3.1	Enlace con título				
2.2.3.2	Calidad de la frase del enlace				

Tabla 2.10 Funcionalidad.

[Fuente: Olsina, 2000]

El resultado obtenido en la siguiente tabla 2.11 a través de la interfaz de la figura 2.14 indica en que porcentaje el software desarrollado es confiable, es decir, libre de fallos.

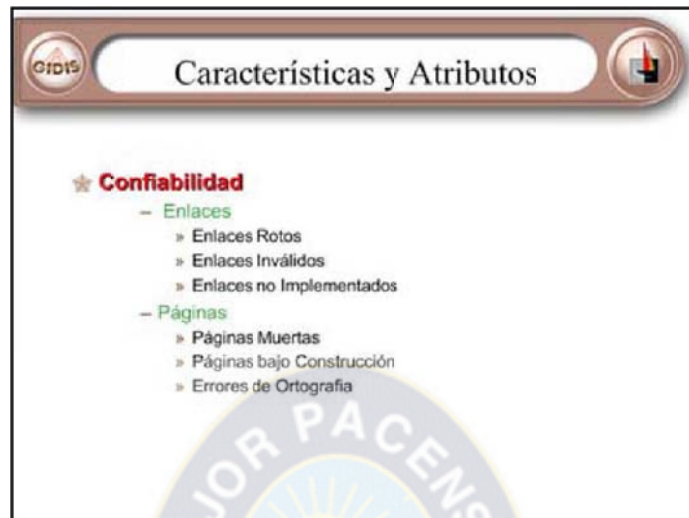


Figura 2.19 Confiabilidad, características y atributos
[Fuente: slidefinder.net]

CÓDIGO	ATRIBUTO	CRITERIO ELEMENTAL	IEi(%)	Peso	Valor
3.	CONFIABILIDAD				
3.1	No deficiencia				
3.1.1	Errores de enlaces				
3.1.1.1	Enlaces rotos				
3.1.1.2	Enlaces inválidas				
3.1.1.3	Enlaces no implementadas				
3.1.2	Errores o deficiencias varias				
3.1.2.1	Deficiencias o cualidades ausentes debido a diferentes navegadores(browsers)				
3.1.2.2	Deficiencia o resultados inesperados independientes de browsers.				
3.1.2.3	Nodos destinos (inesperadamente en construcción)				
3.1.2.4	Nodos muertos (sin enlaces de retorno)				

Tabla 2.11 Confiabilidad.
[Fuente: Olsina, 2000]

El resultado obtenido en la siguiente tabla 2.12 de eficiencia representa en qué grado se puede tener un tiempo de respuesta oportuno.

		CRITERIO ELEMENTAL	IEi(%)	Peso	Valor
4.	EFICIENCIA				
4.1	Performancia				
4.1.1	Páginas de acceso rápido				
4.2	Accesibilidad				
4.2.1	Accesibilidad de información				
4.2.1.1	Soporte versión solo texto				
4.2.1.2	Legibilidad al desactivar propiedad imagen				
4.2.1.2.	Imagen con titulo				
4.2.1.2.	Legibilidad Global				

Tabla 2.12 Eficiencia.

[Fuente: Olsina, 2000]

2.8 Estudio de Costo Beneficio

2.8.1 Modelo Constructivo de Costos COCOMO

El Modelo Constructivo de Costes (Constructive Cost Model) fue desarrollado por B. W. Boehm a finales de los 70 y comienzos de los 80, exponiéndolo detalladamente en su libro "Software Engineering Economics" (Prentice-Hall, 1981). COCOMO es una jerarquía de modelos de estimación de costes software que incluye sub modelos básico, intermedio y detallado. Para este proyecto utilizaremos el modelo básico.

Cocomo pertenece a la categoría de modelos de subestimaciones basados en estimaciones matemáticas. Está orientado a la magnitud del producto final, midiendo el "tamaño" del proyecto, en líneas de código principalmente. Las ecuaciones que se utilizan en los tres modelos son:

- $E = a(Kl)^b * m(X)$, en persona-mes
- $Tdev = c(E)ct$, en meses
- $P = E / Tdev$, en personas

Donde:

- E es el esfuerzo requerido por el proyecto, en persona-mes
- Tdev es el tiempo requerido por el proyecto, en meses
- P es el número de personas requerido por el proyecto
- a, b, c y d son constantes con valores definidos en una tabla, según cada sub modelo
- K1 es la cantidad de líneas de código, en miles.
- m(X) Es un multiplicador que depende de 15 atributos.

A la vez, cada sub modelo también se divide en modos que representan el tipo de proyecto, y puede ser:

- **modo orgánico**, un pequeño grupo de programadores experimentados desarrollan software en un entorno familiar. El tamaño del software varía desde unos pocos miles de líneas (tamaño pequeño) a unas decenas de miles (medio).
- **modo semilibre o semiencajado**, corresponde a un esquema intermedio entre el orgánico y el rígido el grupo de desarrollo puede incluir una mezcla de personas experimentadas y no experimentadas.
- **modo rígido o empotrado**, el proyecto tiene fuertes restricciones, que pueden estar relacionadas con la funcionalidad y/o pueden ser técnicas. El problema a resolver es único y es difícil basarse en la experiencia, puesto que puede no haberla.

2.8.2 Modelo básico

Se utiliza para obtener una primera aproximación rápida del esfuerzo, y hace uso de la siguiente tabla de constantes para calcular distintos aspectos de costes:

Modo	a	b	c	d
Orgánico	2.40	1.05	2.5	0.38
Semilibre	3.0	1.12	2.50	0.35
Rigido	3.60	1.20	2.50	0.32

Tabla 2.13 Tabla de Constantes para Calcular Distintos aspectos de Costes

Fuente: [Wikipedia, 2011]

Estos valores son para las fórmulas:

- Personas necesarias por mes para llevar adelante el proyecto (MM)= $a*(Klb)$
- Tiempo de desarrollo del proyecto (IDEV) = $c*(MMct)$
- Personas necesarias para realizar el proyecto (CosteH) = $MM/TDEV$
- Costo total del proyecto (CosteM) = CosteH * Salario medio entre los programadores y analistas.

Se puede observar que a medida que aumenta la complejidad del proyecto (modo), las constantes aumentan de 2.4 a 3.6, que corresponde a un incremento del esfuerzo del personal. Hay que utilizar con mucho cuidado el modelo básico puesto que se obvian muchas características del entorno.

2.8.3 Modelo intermedio

Este añade al modelo básico quince modificadores opcionales para tener en cuenta en el entorno de trabajo, incrementando así la precisión de la estimación. Para este ajuste, al resultado de la fórmula general se lo multiplica por el coeficiente surgido de aplicar los atributos que se decidan utilizar.

Se puede observar que los exponentes son los mismos que los del modelo básico, confirmando el papel que representa el tamaño mientras que los coeficientes de los modos orgánico y rígido han cambiado, para mantener el equilibrio alrededor del semilibre con respecto al efecto multiplicador de los atributos de coste.

Cada atributo se cuantifica para un entorno de proyecto. La escala es muy bajo - bajo - nominal - alto - muy alto - extremadamente alto. Dependiendo de la calificación de cada atributo, se asigna un valor para usar de multiplicador en la fórmula (por ejemplo, si para un proyecto el atributo DATA es calificado como muy alto, el resultado de la fórmula debe ser multiplicado por 1000).

El significado de los atributos es el siguiente, según su tipo:

De software

- REL Y: garantía de funcionamiento requerida al software. Indica las posibles consecuencias para el usuario en el caso que existan defectos en el producto. Va desde la sola inconveniencia de corregir un fallo (muy bajo) hasta la posible pérdida de vidas humanas (extremadamente alto, software de alta criticidad).
- DATA: tamaño de la base de datos en relación con el tamaño del programa. El valor del modificador se define por la relación: D / K , donde D corresponde al tamaño de la base de datos en bytes y K es el tamaño del programa en cantidad de líneas de código.
- CPLX: representa la complejidad del producto.

De hardware

- TIME: limitaciones en el porcentaje del uso de la CPU.
- STOR: limitaciones en el porcentaje del uso de la memoria.
- VIRT: volatilidad de la máquina virtual.
- TURN: tiempo de respuesta requerido.

De personal

- ACAP: calificación de los analistas.
- AEXP: experiencia del personal en aplicaciones similares.
- PCAP: calificación de los programadores.
- VEXP: experiencia del personal en la máquina virtual.
- LEXP: experiencia en el lenguaje de programación a usar.

De proyecto

- MODP: uso de prácticas modernas de programación.
- TOOL: uso de herramientas de desarrollo de software.
- SCED: limitaciones en el cumplimiento de la planificación.

2.8.4 Modelo Detallado

Presenta principalmente dos mejoras respecto al anterior:

Los factores correspondientes a los atributos son sensibles o dependientes de la fase sobre la que se realizan las estimaciones. Aspectos tales como la experiencia en la aplicación, utilización de herramientas de software, etc., tienen mayor influencia en unas fases que en otras, y además van variando de una etapa a otra.

Establece una jerarquía de tres niveles de productos, de forma que los aspectos que representan gran variación a bajo nivel, se consideran a nivel módulo, los que representan pocas variaciones, a nivel de subsistema; y los restantes son considerados a nivel sistema.

2.9 Seguridad

2.9.1 Nivel Físico

El lugar donde este colocado el servidor es sumamente importante para su estabilidad. El servidor necesita estar protegido contra distintos factores externos que pueden alterar el funcionamiento de la red.

Estos factores externos son: la electricidad estática, el calor, los ruidos eléctricos, los altibajos de tensión y los cortes de corriente.

2.9.1.1 La protección contra la electricidad

Se han de tomar algunas precauciones para proteger al servidor de las cargas estáticas, ya que el rendimiento de este afecta a toda la red.

No utilizar plásticos ni material sintético, ya que generan electricidad estática.

El calor y el frío excesivos son riesgos potenciales para el buen funcionamiento del servidor. Se ha de mantener la temperatura de la habitación del servidor entre 18° y 26° C, y asegurar una buena aeración.

2.9.1.2 La protección contra los ruidos eléctricos, los altibajos de tensión y los cortes de corriente.

Los ruidos eléctricos son causados por las inconsistencias del suministro de la corriente del ordenador. Para proteger al servidor contra los ruidos eléctricos, puede recurrirse a la instalación de una línea dedicada de suministro eléctrico.

No hay que conectar otros dispositivos a este suministro de corriente, porque pueden generar ruidos que anulen las ventajas de la protección ofrecida por la fuente de corriente dedicada. La conexión a la fuente de energía se ha de hacer con cable estándar de tres hilos, con el hilo de masa conectado a tierra.

2.9.1.3 Protección contra suciedad

Aquí interviene tanto la suciedad de la sala donde se encuentra el servidor como la suciedad que pueden generar los propios usuarios.

Hay que mantener la sala en un estado de perfecta limpieza para que evitar que el polvo pueda concentrarse dentro del servidor y altere su correcto funcionamiento.

Referente a los usuarios y administradores se ha de tener en cuenta que cualquier vertido de líquidos o de restos de alimentos sobre la pantalla y/o el teclado del servidor pueden producir distintos daños potenciales en el servidor.

2.9.1.4 Seguridad contra incendios y agua

De qué sirve tener bien protegido el servidor si no se cuenta con una buena protección contra incendios. En la sala donde se encuentre instalado el servidor debe haber detectores de humo de alta sensibilidad y un sistema contra incendios a base de gas a presión. Este sistema debe producir un aviso previo a su utilización porque provoca el apagado del incendio de forma inmediata por consumo del oxígeno de la sala y, por tanto, todas las personas deberán salir de dicha sala porque podrían correr riesgo de asfixia.

También deberá estar protegido el servidor contra peligros de inundaciones y goteras que podrían provocar cortocircuitos.

2.9.2 Nivel Lógico

Consiste en la aplicación de barreras y procedimientos que resguarden el acceso a los datos y sólo se permita acceder a ellos a las personas autorizadas para hacerlo.

Controles de Acceso

- Identificación y Autenticación
- Roles

- Transacciones
- Limitaciones a los Servicios
- Modalidad de Acceso
- Ubicación y Horario
- Control de Acceso Interno
- Control de Acceso Externo
- Administración

2.10 Políticas de Implementación

Las especificaciones del diseño del sistema sirven como base para la construcción del nuevo sistema. El proceso de activar el nuevo sistema de información y retirar el viejo, el cambio puede ser rápido o lento, dependiendo del método a usarse. Los cuatro métodos son Corte y Cambio, operación Paralela, operación piloto y operación de fase.

2.10.1 Corte y cambio

El cambio del sistema viejo al nuevo ocurre tan pronto el nuevo sistema es operacional. Usualmente es el menos costoso, puesto que se debe operar y mantener un sistema a la vez. Por otro lado, es el método con mayor riesgo, pues no se puede usar el sistema viejo para ayudar a resolver los problemas que ocurran, o para verificar que todos los detalles de los datos sean correctos. Este método se puede usar al implementar programas comprados, para sistema no críticos o cuando los dos sistemas (el viejo y el nuevo) no puedan existir.

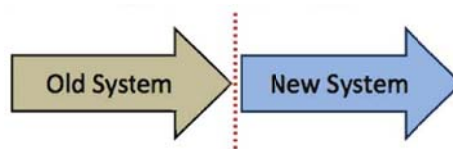


Figura 2.20: Corte y Cambio
[Fuente: Copley, S.]

2.10.2 Operación paralela

Ambos sistemas operan totalmente por un tiempo específico. Los datos son a ambos sistemas, y la salida generada por el nuevo sistema es comparado con su equivalente en el sistema viejo. Cuando el nuevo sistema opera correctamente, se deja de usar el sistema viejo. La ventaja mayor es su bajo riesgo, si el nuevo sistema no trabaja bien, se puede usar el sistema viejo hasta que se realicen los cambios apropiados. Sin embargo, este método es el más costoso, pues hay que darle servicios a ambos sistemas. Los usuarios deben trabajar en los dos sistemas y a veces hace falta ayuda temporánea. Este método no es práctico si los dos sistemas son incompatibles o realizan funciones diferentes.



Figura 2.21: Operación Paralela
[Fuente: Copley, S.]

2.10.3 Operación Piloto

Se implementa todo el nuevo sistema en un área de la compañía. El grupo que usa el nuevo sistema se conoce área o grupo piloto. El sistema viejo continúa operando en toda la organización, incluyendo el área piloto. Después de probar que el sistema trabaja correctamente en el área piloto, se implementa en toda la organización, normalmente usando el método corte y cambio. La operación piloto reduce el riesgo de fallas del sistema comparando con corte y cambio, y es menos costoso que la operación paralela.

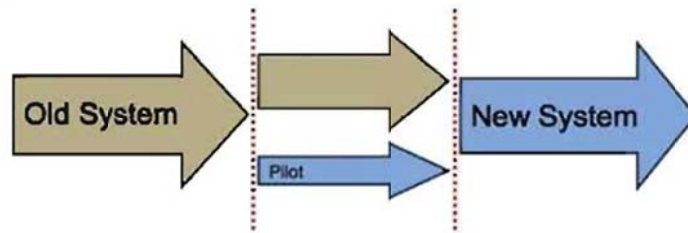


Figura 2.22: Operación Piloto
[Fuente: Copley, S.]

2.10.4 Operación de fase

Se implanta el nuevo sistema en fases, o módulos, en vez de implantar todo el sistema a la vez.

Cada subsistema se puede implantar usando uno de los otros métodos.

Una ventaja es que el riesgo se limita solo al módulo implementando. Este método no es posible si no se puede separar el sistema en módulos. No se debe confundir el método operación de fase con la operación piloto. En la operación de fase se da parte del sistema a todos los usuarios, mientras que la operación piloto provee el sistema completo, pero solo a un grupo de usuarios

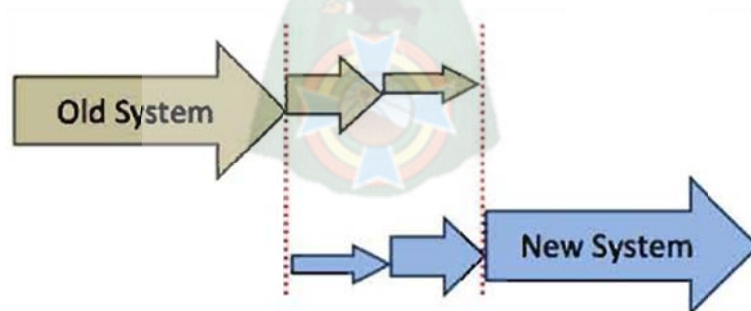


Figura 2.23: Operación de fase
[Fuente: Copley, S.]

3.1 Introducción

En este capítulo se utilizara el Proceso Unificado Ágil (AUP), que compone las siguientes fases: iniciación, elaboración, construcción de la aplicación y transición, siguiendo las diversas fases del modelo y desarrollo del modelo UWE (Ingeniería Web Basado en UML). Las fases de esta metodología son: análisis de requerimientos, diseño conceptual, diseño Navegación y diseño de presentación que nos presentan diversos diagramas y esquemas en un proceso interactivo e incremental dando apoyo al modelo de la aplicación.

3.2 Fase de Inicio

Esta fase tiene como propósito definir y acordar el alcance del proyecto, identificar los riesgos asociados al proyecto, proponer una visión muy general de la arquitectura de software. En esta fase el objetivo principal es el modelado de requerimientos de alto nivel. La fase inicio podría ser muy breve si se trata de un problemas conocido, o se ha decidido realizar el proyecto de todas formas.

3.2.1 Modelado del Negocio

El modelado de negocio permitirá comprender mejor los procesos de funcionamiento del control de los activos fijos.

3.2.1.1 Modelado de casos de uso del Negocio

En el diagrama de casos de uso de la figura 3.1 se observa las operaciones que se efectúan en la empresa Grupo Larcos Industrial Ltda., para la asignación, solicitar la compra, supervisar activos fijos, verificar el estado, verificar la existencia, ubicaciones y las depreciaciones de activos fijos, donde interviene directamente la gerencia de operaciones verificando la existencia de los activos fijos para asignar a los nuevos empleados que ingresan a la empresa. La unidad de contabilidad es la encargada de

realizar las depreciaciones de los activos fijos, verificando la fecha de compra del mismo, a su vez cuando determinar la baja de los activos fijos por obsolescencia o daños.

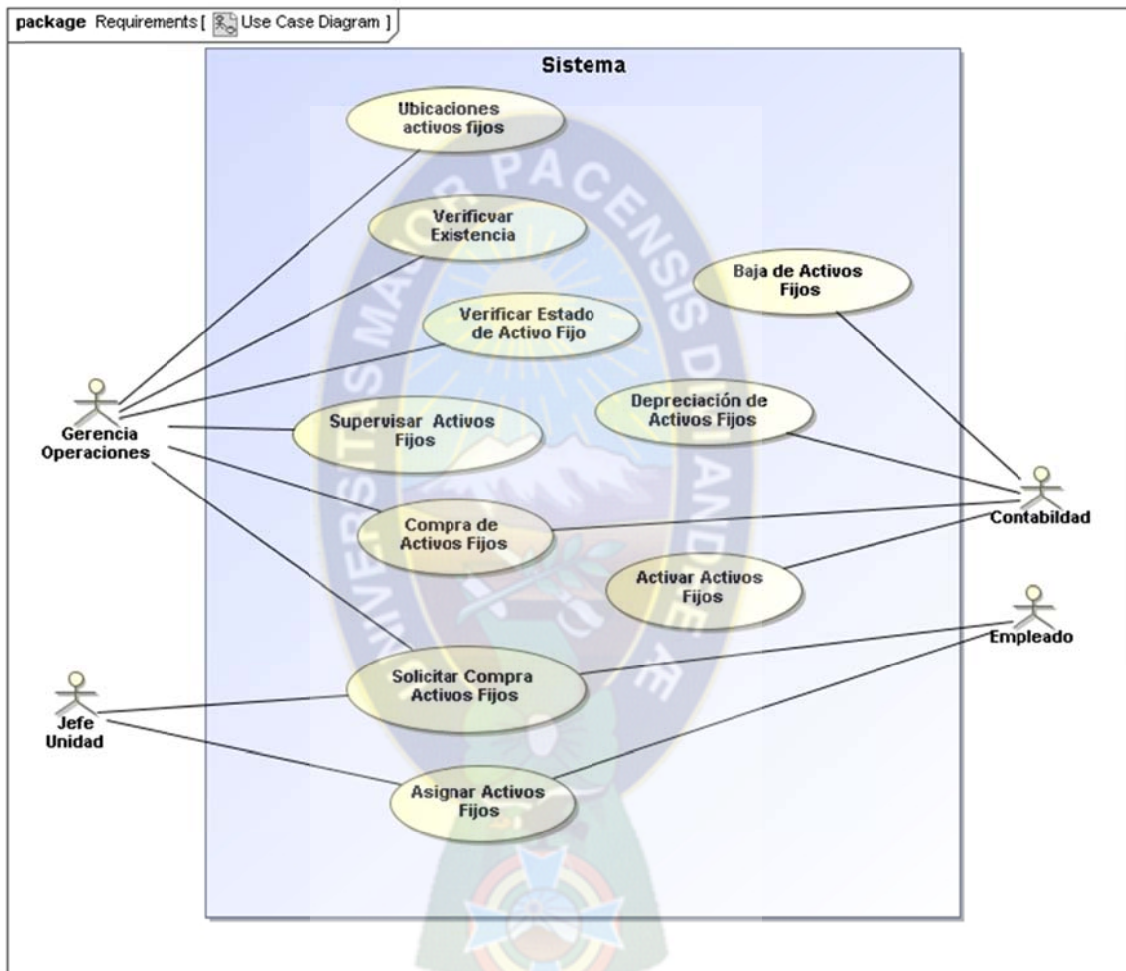


Figura 3.1: Diagrama de casos de uso del negocio
[Fuente: Elaboración Propia]

3.2.1.2 Descripción de actores del caso de uso de negocio

Descripción de los actores del diagrama de casos de uso del negocio

Actor	Descripción
Gerencia de Operaciones	Persona encargada de la gestión del sistema crea y administra las cuentas de usuarios, realiza el control y seguimiento, genera reportes.
Unidad Contabilidad	Persona encargada de realizar las depreciaciones, verificar estados, y bajas de los Activos Fijos
Jefes de Unidad	Personas encargadas de asignar los activos fijos a su personal, solicitar la compra de nuevos Activos Fijos.
Empleado	Persona que solicita Activos <u>Fijos</u>

Tabla 3.1: Descripción de actores de casos de uso del negocio
[Fuente Elaboración Propia]

3.2.2 Modelado de Requerimientos

Es el conjunto de técnicas y procedimientos que nos permiten conocer los elementos necesarios para definir un proyecto de software.

Es la etapa más crucial del desarrollo de un proyecto de software. Describe los requisitos que el sistema debe cumplir.

Consta de una variedad de productos de trabajo, incluyendo: requisitos técnicos, requerimientos de usuario, requerimientos a nivel de negocio. Los requerimientos son la parte final de la fase de inicio.

3.2.2.1 Descripción de requerimiento a nivel de negocios

Desarrollar un sistema web de seguimiento y control de activos fijos vía web para el personal de la empresa Grupo Larcos Industrial LTDA.

3.2.2.2 Descripción de requerimientos a nivel de usuario

U1.- Registro los datos personales de un empleado

U2.- Administrar el Inventario de los Activos Fijos, es decir que se pueda Adicionar, Eliminar y Modificar.

U3.- Asignaciones de los Activos Fijos.

U4.- Registro de Sucursales de la Empresa

U5.- Registro de Unidades de la Empresa

U6.- Depreciaciones de los Activos Fijos

U7.- Bajas de los Activos Fijos

U8.- Revalorización de Activos Fijos

U9.- Generar Reportes

U10.- Generar Boletas de Asignación de Activos Fijos

U11.- Reportes de Depreciaciones por Gestiones

U12.- Reportes de Bajas de Activos Fijos

U13.- Reportes de Asignación de Activos Fijos

3.2.2.3 Descripción de requerimientos a nivel de sistema

S1.- El sistema debe tener una interfaz para el registro de los datos personales del empleado.

S2.- El sistema tendrá una interfaz que permita elegir entre modificación y eliminación del Inventario de los activos fijos.

S3.- El sistema debe tener una interfaz que permita registrar las asignaciones de los activos fijos

S4.- El sistema de dar una alerta cuando el activo fijo haya terminado su vida útil

S5.- El sistema debe tener una interfaz que permita verificar las ubicaciones de los activos fijos

S6.- El sistema debe tener una interfaz que permita registrar las depreciaciones de los activos fijos

S7.- El sistema debe tener una interfaz que permita registrar las bajas de los activos fijos

S8.- El sistema debe tener una interfaz que permita generar boletas de asignación

3.2.2.4. Descripción de requerimientos a nivel técnico

Se tiene los requerimientos a nivel de hardware y software.

T1.- La codificación será realizada sobre la plataforma Microsoft, utilizando Visual Studio 2010

T2.- Para realizar los reportes se utiliza Crystal Reports.

T3.- Servidor de Aplicación IIS (Internet Information Services)

T4.- Servidor de Base de Datos SQL SERVER 2008.

3.3 Fase de Elaboración

En esta fase se determinaran las soluciones técnicas del proyecto. Durante la cual se elaboran los requisitos al nivel del diseño y por tanto, se realizan los modelos de casos de uso, casos de uso extendidos, diagramas de secuencia, estado, clases y diagrama de navegación

3.3.1 Modelo de Análisis

3.3.1.1 Modelo de casos de uso

Los casos de uso representan la iteración entre los usuarios y el sistema. Este modelo de casos de uso se desarrolla a lo largo de varias iteraciones añadiendo nuevos casos de uso y mejorando la descripción de los casos de uso que se crearon anteriormente.

3.3.1.2 Diagrama de casos de uso de alto nivel

Se describe clara y concisamente los procesos.

Después de realizar un análisis de los requerimientos, se han identificado los siguientes casos de uso y actores, que se puede observar en la Figura 3.2

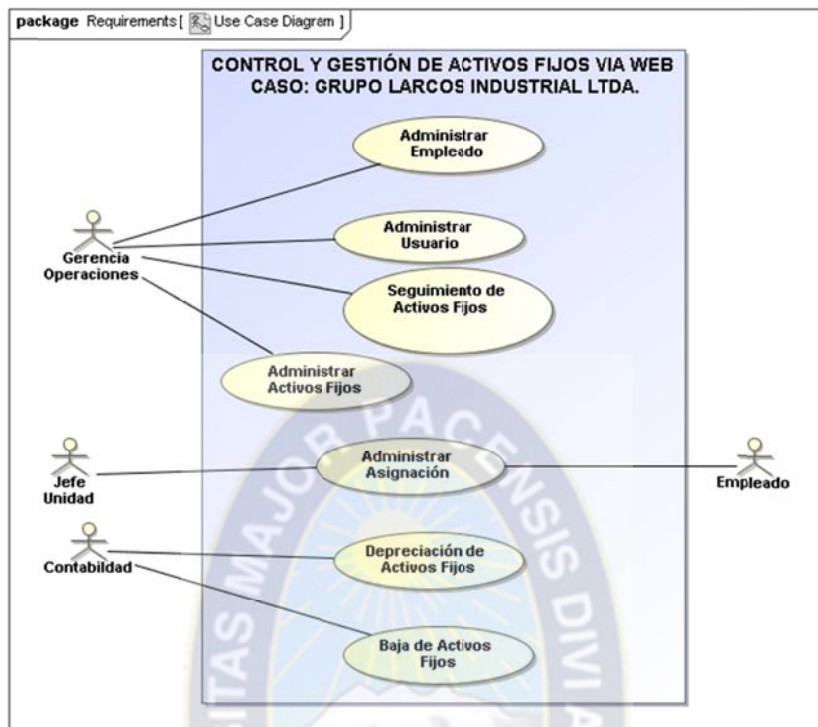


Figura 3.2: Diagrama de caso de uso: Control y Gestión de Activos Fijos vía Web
[Fuente: Elaboración Propia]

3.3.1.3 Descripción de casos de uso

A continuación, se describen los diagramas de casos de la Figura 3.2, estos diagramas proporcionan una guía para los siguientes flujos de trabajo y su descripción, desde el diseño hasta las pruebas.

a) Caso de uso: Administrar Empleado

La administración de empleado consiste en el registro de empleado que no es más que introducir todos los datos personales del empleado, la modificación de empleado que se realiza en caso de tener algún cambio de información después de haber realizado el registro y finalmente la eliminación de un empleado que se puede realizar cuando se despide a un empleado. La persona a cargo de todos estos procesos es el

personal de Gerencia de Operaciones que se labor consistirá en el manejo de la administración de Empleados.

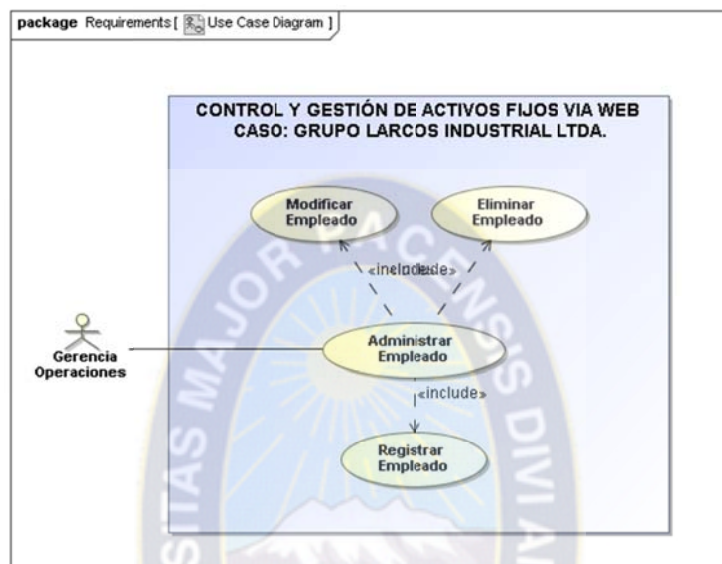


Figura 3.3: Diagrama de casos de uso: Administrar Empleado
[Fuente: Elaboración Propia]

Descripción del Diagrama de caso de Uso: Administrar Empleado	
Nombre :	Administrar Empleado
Autor :	Marco Méndez
Descripción :	Permite al encargado de personal registrar, modificar y eliminar empleado
Actores :	Personal de Gerencia de Operaciones
Precondiciones:	El actor debe ser usuario del sistema y tener el rol correspondiente.
Flujo Normal :	<ol style="list-style-type: none"> 1 Autenticación del actor en el sistema 2 Elegir la opción de adicionar, eliminar o modificar empleado y cambio de contraseña. 3 Salir del sistema.
Flujo alternativo :	En caso de cometer algún error en la adición, eliminación o modificación el sistema informara dicho error.

Post condición:

El sistema guarda los cambios en la administración de Empleado

Tabla 3.2: Descripción de casos de uso: Administración de personal
[Fuente: Elaboración Propia]

b) Caso de uso: Administrar Usuario

La administración de usuario consiste en la asignación de usuario que es la dar un rol a un empleado en el sistema, la modificación de usuario en caso de haber cometido algún error al habilitar a algún empleado y finalmente la eliminación de usuario que se puede realizar cuando se despide a un empleado.

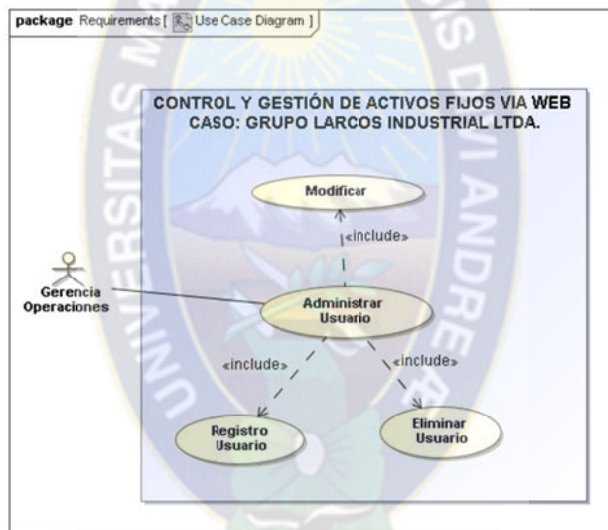


Figura 3.4: Diagrama de caso de uso: Administrar Usuario
[Fuente: Elaboración Propia]

Descripción del Diagrama de caso de Uso: Administrar Usuario	
Nombre :	Administrar Usuario
Autor :	Marco Méndez
Descripción :	Permite al Personal de la Gerencia de Operaciones registrar, modificar y eliminar usuario.
Actores :	Personal de Gerencia de Operaciones
Precondiciones:	El actor debe ser usuario del sistema y tener el rol de administrador
Flujo Normal :	

1 Autenticación del actor en el sistema 2 Elegir la opción de administración de usuario 3 Elegir la opción de adicionar, eliminar o modificar usuario. 4 Salir del sistema.
Flujo alternativo : En caso de cometer algún error en la administración de usuario el sistema informara dicho error.
Post condición: El sistema guarda los cambios en la administración de usuario

Tabla 3.3: Descripción de caso de uso: Administración de usuario
[Fuente: Elaboración Propia]

c) Caso de uso: Seguimiento de Activos Fijos

En el seguimiento de activos fijos, es un proceso que se realiza cada vez que se asigna, compra o se da de baja un activo fijo, se debe registrar cada acción que se realiza. La persona a cargo de todos estos procesos es el personal de Gerencia de Operaciones que se labor consistirá en el Seguimiento a los Activos Fijos.

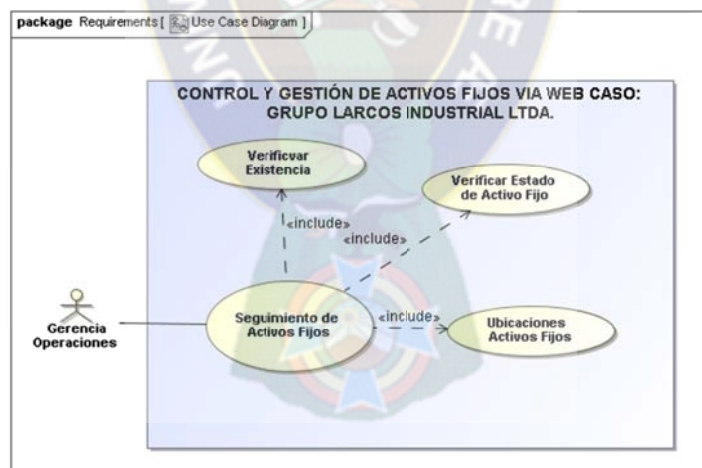


Figura 3.5: Diagrama de caso de uso: Control y Seguimiento de los Activos Fijos
[Fuente: Elaboración Propia]

Descripción del Diagrama de caso de Uso: Seguimiento de los Activos Fijos	
Nombre :	Seguimiento de Activos Fijos

Autor :	Marco Méndez
Descripción :	Permite al Personal de la Gerencia de Operaciones, Realizar control de todos los Activos Fijos, verificar estado, verificar existencia, ubicaciones
Actores :	Personal de Gerencia de Operaciones
Precondiciones:	El actor debe ser usuario del sistema y tener el rol respectivo
Flujo Normal :	<ol style="list-style-type: none"> 1 Autenticación del actor en el sistema 2 Elegir la opción de Seguimiento 3 Elegir la opción de: verificar Existencia, verificar Estado, Ubicaciones 4 Salir del sistema.
Flujo alternativo :	En caso de cometer algún error en el seguimiento de Activos Fijos el sistema informara dicho error.
Post condición:	El sistema genera Reporte

Tabla 3.4: Descripción de caso de uso: Control y Seguimiento de Activos Fijos
[Fuente: Elaboración Propia]

d) Caso de uso : Administrar Activos Fijos

La administración de Activos Fijos consiste en el registro que no es más que introducir todos los datos de los activos fijos, la modificación en caso de tener algún cambio de información después de haber realizado el registro y finalmente la eliminación que se puede realizar cuando el activo fijos sea obsoleto o defectuoso para darlo de baja. La persona a cargo de todos estos procesos es el personal de Gerencia de Operaciones.

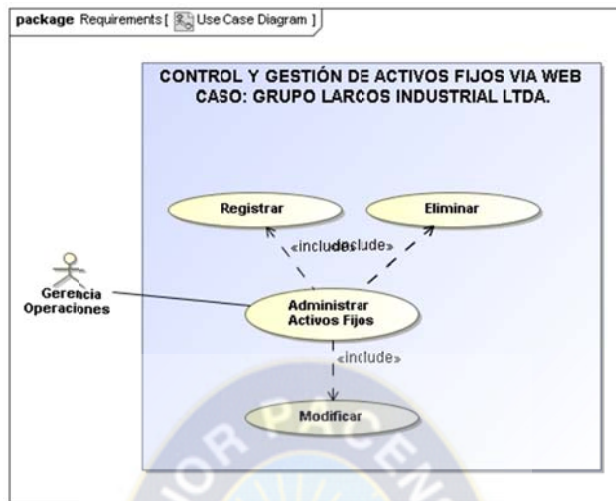


Figura 3.6: Diagrama de caso de uso: Administrar Activos Fijos
[Fuente: Elaboración Propia]

Descripción del Diagrama de caso de Uso: Administrar Activo Fijo	
Nombre :	Administrar Activo Fijo
Autor :	Marco Méndez
Descripción :	Permite al Personal de la Gerencia de Operaciones registrar, modificar y eliminar activos fijos.
Actores :	Personal de Gerencia de Operaciones
Precondiciones:	El actor debe ser usuario del sistema y tener el rol de administrador
Flujo Normal :	<ol style="list-style-type: none"> 1 Autenticación del actor en el sistema 2 Elegir la opción de administración de activos fijos 3 Elegir la opción de adicionar, eliminar o modificar 4 Salir del sistema.
Flujo alternativo :	En caso de cometer algún error en la administración de activos fijos el sistema informara dicho error.
Post condición:	El sistema guarda los cambios en la administración de activos fijos

Tabla 3.5: Descripción de caso de uso: Administrar Activos Fijos
[Fuente: Elaboración Propia]

e) Casos de uso : Administrar Asignación

Administrar la asignación consiste en verificar existencias, verificar el estado del activo fijo y el registro que no es más que introducir todos los datos de los activos fijos y del empleado a ser asignado, la modificación en caso de tener algún cambio de información después de haber realizado el registro. La persona a cargo de este proceso es el Jefe de Unidad.

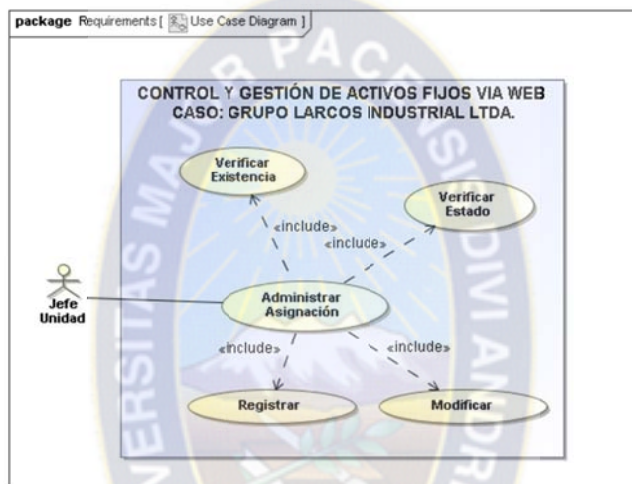


Figura 3.7: Diagrama de caso de uso: Administrar Asignación
[Fuente: Elaboración Propia]

Descripción del Diagrama de caso de Uso: Administrar Asignación	
Nombre :	Administrar Asignación
Autor :	Todos los Jefes de Unidad
Descripción : Permite a los Jefes de Unidad registrar, modificar y eliminar la asignación de activos fijos.	
Actores : Jefes de Unidad	
Precondiciones: El actor debe ser usuario del sistema y tener el rol respectivo	
Flujo Normal : 1 Autenticación del actor en el sistema 2 Elegir la opción de administrar Asignación 3 Elegir la opción de adicionar, eliminar o modificar	

4 Salir del sistema.
Flujo alternativo : En caso de cometer algún error en la administración de activos fijos el sistema informara dicho error.
Post condición: El sistema guarda los cambios en administrar asignación

Tabla 3.6: Descripción de caso de uso: Administrar Asignación
[Fuente: Elaboración Propia]

f) Caso de uso: Depreciación de Activos Fijos

La depreciación de activos fijos, es un proceso que se realiza cada vez que termina la gestión contable, se debe registrar la depreciación de todos los activos fijos, la modificación en caso de haber cometido algún error al registrar. La persona a cargo de todos estos procesos es el personal de Contabilidad.

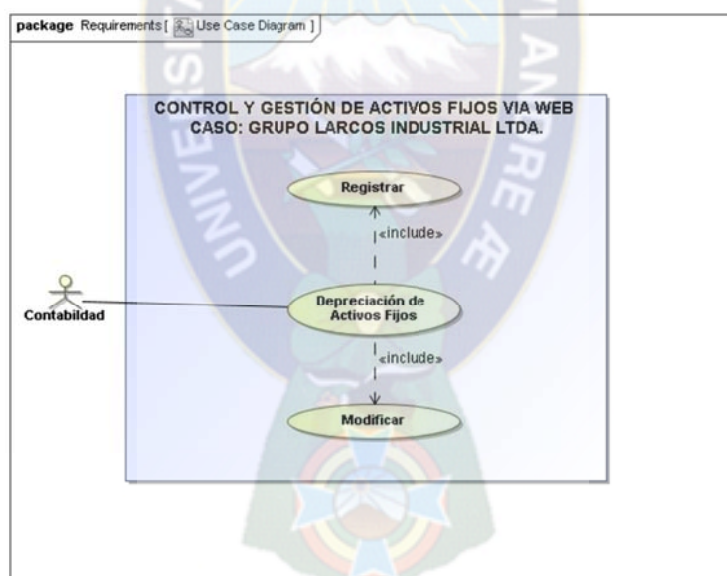


Figura 3.8: Diagrama de caso de uso: Depreciación de Activos Fijos
[Fuente: Elaboración Propia]

Descripción del Diagrama de caso de Uso: Depreciación de Activos Fijos	
Nombre :	Depreciación de Activos Fijos
Autor :	Contabilidad
Descripción : Permite al personal de contabilidad registrar y modificar la	

depreciación de los activos fijos.
Actores : Personal de Contabilidad
Precondiciones: El actor debe ser usuario del sistema y tener el rol respectivo
Flujo Normal : 1 Autenticación del actor en el sistema 2 Elegir la opción de Depreciación de Activos Fijos 3 Elegir la opción de registrar o modificar 4 Salir del sistema.
Flujo alternativo : En caso de cometer algún error en la Depreciación de activos fijos el sistema informara dicho error.
Post condición: El sistema guarda los cambios en la Depreciación de Activos Fijos

Tabla 3.7: Descripción de caso de uso: Depreciación de Activos Fijos
[Fuente: Elaboración Propia]

g) Caso de uso : Baja de Activos Fijos

La Baja de activos fijos, es un proceso que se realiza cada vez que un activo fijo termina su tiempo de vida útil y obsolescencia, se debe registrar todas las bajas de los activos fijos, la modificación en caso de haber cometido algún error al registrar. La persona a cargo de todo este proceso es el personal de Contabilidad.

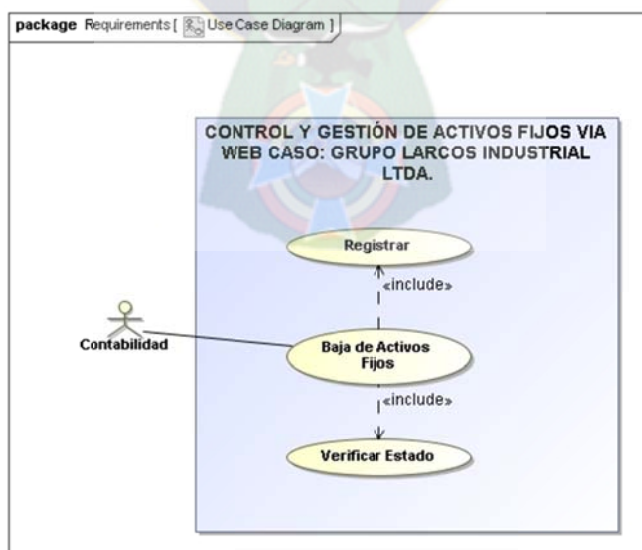


Figura 3.9: Diagrama de caso de uso: Baja de Activos Fijos
[Fuente: Elaboración Propia]

Descripción del Diagrama de caso de Uso: Baja de Activos Fijos	
Nombre :	Baja de Activos Fijos
Autor :	Contabilidad
Descripción :	Permite al personal de contabilidad registrar y verificar el estado antes de registrar la depreciación de los activos fijos.
Actores :	Personal de Contabilidad
Precondiciones:	El actor debe ser usuario del sistema y tener el rol respectivo
Flujo Normal :	1 Autenticación del actor en el sistema 2 Elegir la opción de Baja de Activos Fijos 3 Elegir la opción de registrar o modificar 4 Salir del sistema.
Flujo alternativo :	En caso de cometer algún error en la Baja de activos fijos el sistema informara dicho error.
Post condición:	El sistema guarda los cambios en Baja de Activos Fijos

Tabla 3.8: Descripción de caso de uso: Baja de Activos Fijos
[Fuente: Elaboración Propia]

3.3.2 Modelo de diseño

Diagramas descriptivos del diseño lógico, sin referencia al modo de implementación. Comprende diagramas de clases del software, diagrama de navegación y otros.

3.3.2.1 Diagrama de clases

Los diagramas se utilizan generalmente para facilitar el entendimiento de largas cantidades de datos y la relación entre diferentes partes de los datos.

Un diagrama de clases representa las clases que serán utilizadas dentro del sistema y las relaciones que existen entre ellas. Nos sirve para visualizar las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia, de uso y de convencimiento. Un diagrama de clases está compuesto por los siguientes elementos: clase: atributos, métodos y visibilidad Relaciones: Herencia, Composición, Agregación, Asociación y uso.

Este es el diagrama principal para el análisis y diseño, en la Figura 3.10 se representa las relaciones entre las clases, atributos y sus operaciones para representar la información del sistema. Después de haber realizado el análisis para la base de datos e identificar todas las entidades que intervienen en el sistema, se elabora el diagrama de clases.

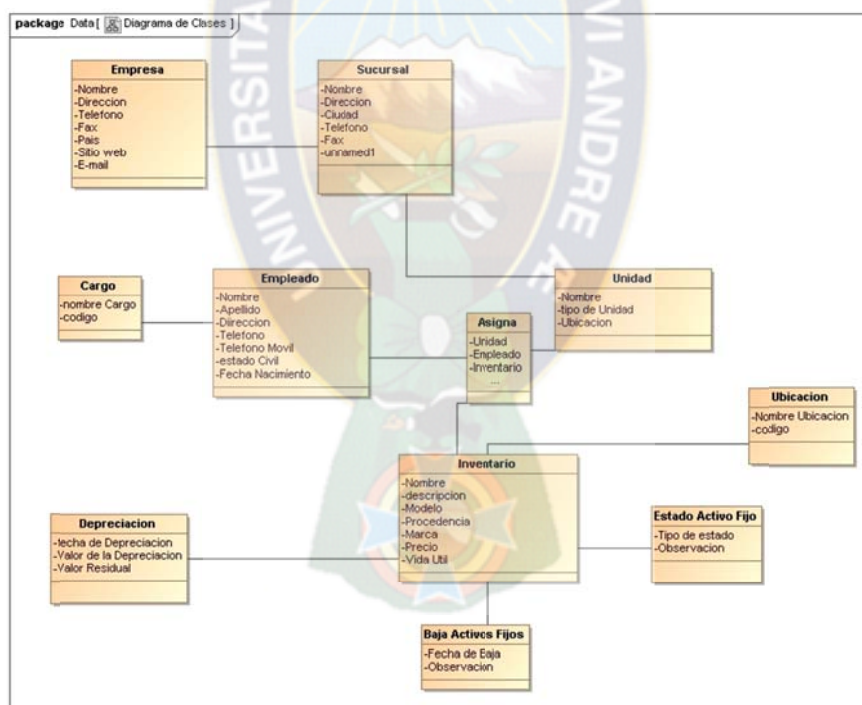


Figura 3.10: Diagrama de Clases: Control y Gestión de Activos Fijos
[Fuente: Elaboración Propia]

3.3.2.2 Modelo Entidad Relación

El almacenamiento de los datos en el sistema, se implementara en un gestor de base de datos relacional, que se muestra en la figura 3.11

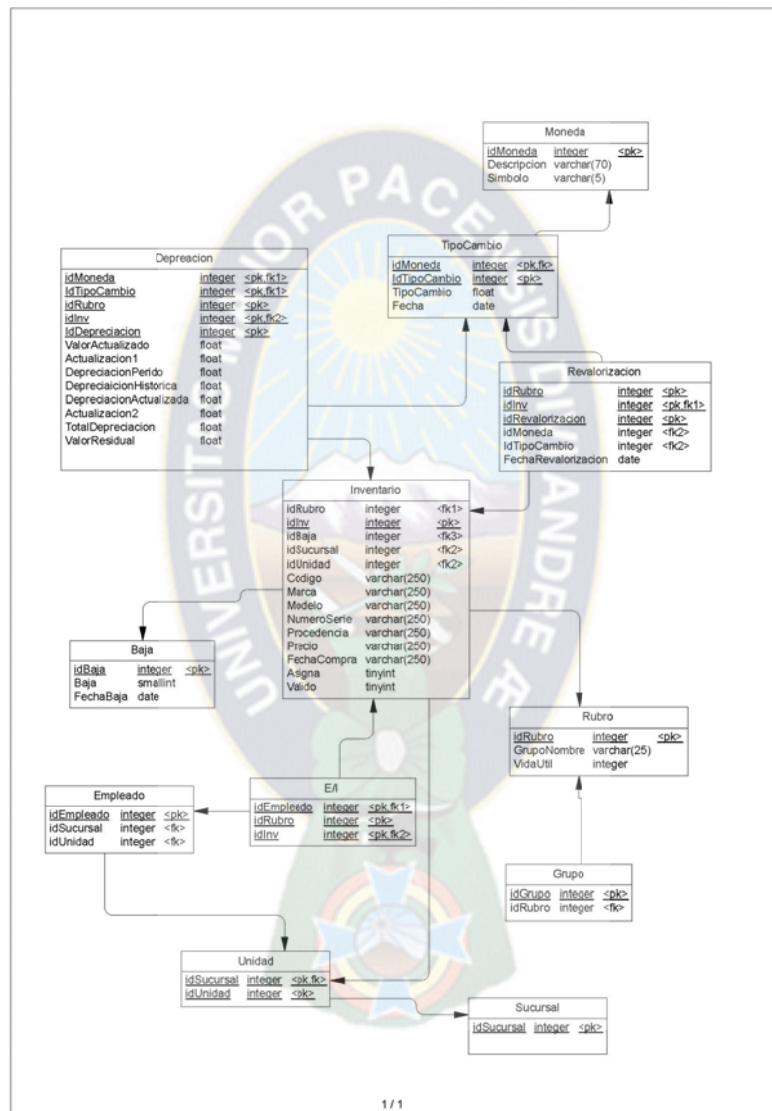


Figura 3.11: Modelo Conceptual: Control y Gestión de Activos Fijos
[Fuente: Elaboración Propia]

3.3.2.3 Modelo de Navegación

En la Figura 3.10 se observa el aspecto de navegación que especifica los objetos que pueden ser visitados mediante la navegación de los usuarios, clasificados por el rol que se les asigne.

En el diagrama de navegación se tiene tres usuarios: Gerencia de Operaciones, Contabilidad y Jefe de Unidad

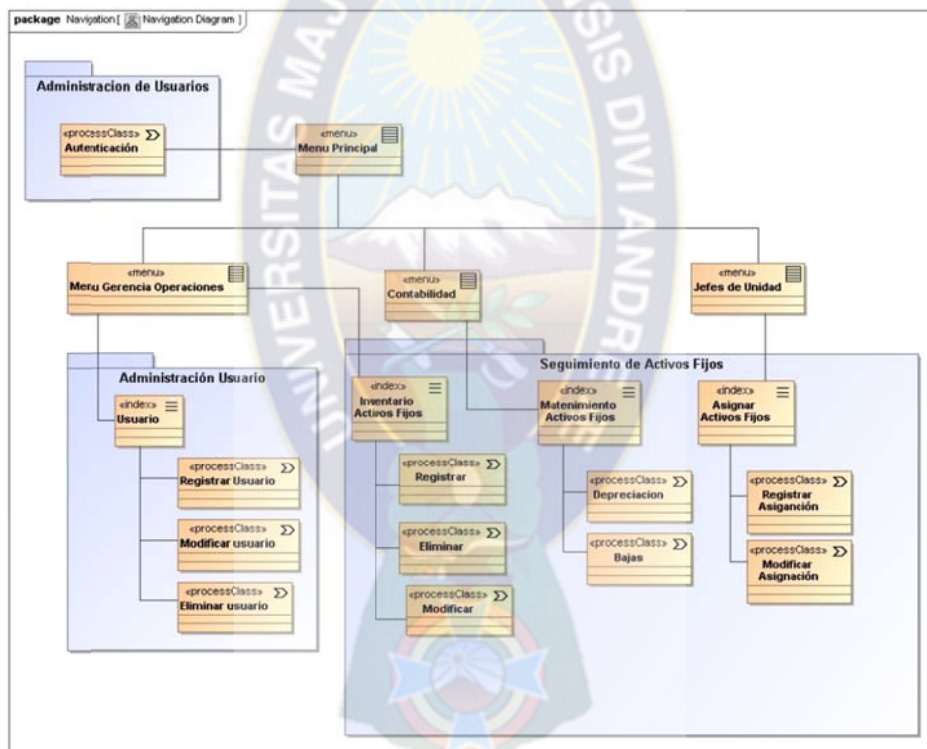


Figura 3.12: Modelo de Espacio de Navegación
[Fuente: Elaboración Propia]

3.3.2.4 Modelo de Presentación

El modelo de presentación nos va mostrando como se verá el sistema:

En la Figura 3.12 se puede observar la página principal después de haberse autenticado en el sistema. En la página principal tenemos un esquema del diseño de cómo se verá.

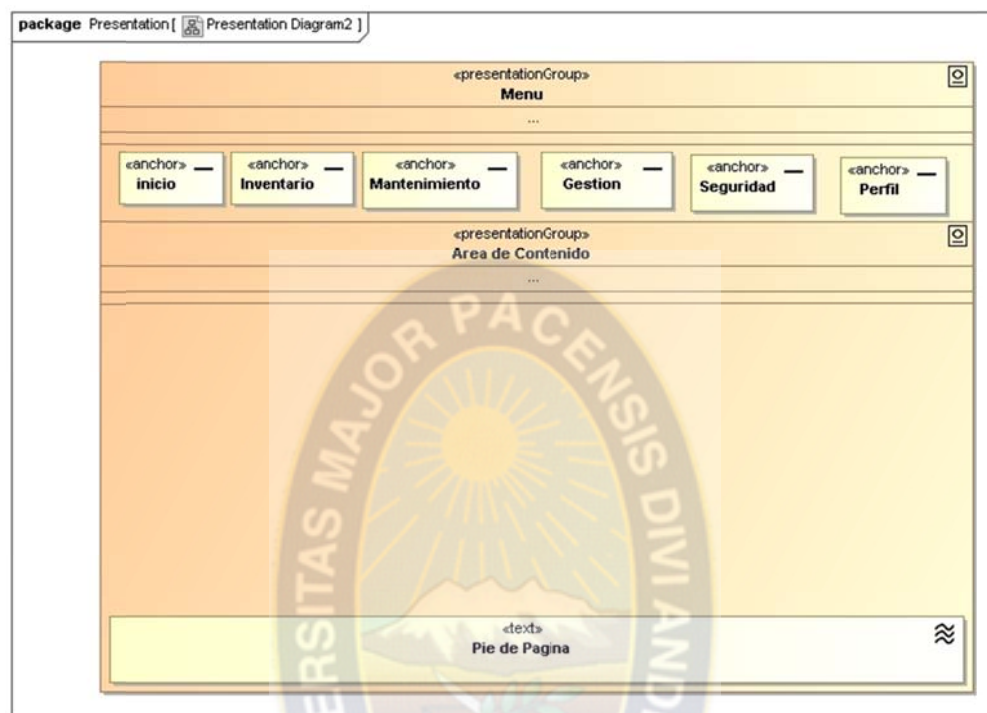


Figura 3.13: Modelo de presentación página maestra
[Fuente: Elaboración Propia]

3.4 Fase de construcción

El objetivo de esta fase consiste en desarrollar el sistema hasta el punto en que esté listo para pre producción de pruebas.

3.4.1 Diseño de interfaces

El diseño de la interfaz del sistema se desarrolló siguiendo el modelo de requerimientos y el modelo de diseño.

3.4.1.1 Autenticación

El empleado que quiera acceder al sistema debe tener su cuenta de usuario y contraseña, que son asignados por el administrador del sistema en este caso por la Gerencia de operaciones, que se muestra en la Figura 3.13

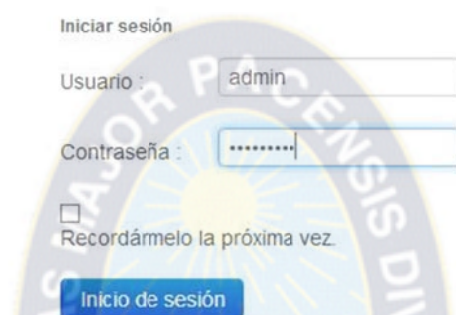


Figura 3.14: Autenticación del Sistema
[Fuente: Elaboración Propia]

3.4.1.2 Pantalla principal

Una vez que el usuario ha realizado el ingreso de su cuenta de usuario y contraseña, el sistema le permitirá el ingreso a la página principal, donde dependiendo del rol asignado al usuario pueda acceder a las diferentes opciones del menú del sistema, que se muestra en la figura 3.14



Figura 3.15: Pantalla Principal
[Fuente: Elaboración Propia]

3.4.1.3 Registro de empleado

Previamente se tiene que elegir la opción de registrar en el menú Gestión. En el registro de empleado se adicionan los datos personales, después de llenar toda la información se tiene la opción de finalizar para terminar el registro, continuar para registrar los datos y cancelar en caso de cometer algún error cancelar el registro.

The screenshot shows a web application interface for 'GRUPO LARCOS'. The main title is 'GESTIÓN DE ACTIVOS FIJOS'. A navigation bar includes 'Página Inicio', 'Inventario', 'Gestion de Inventario', 'Gestion de Grupo', 'Administración', and 'Reportes'. The 'EMPLEADOS' section contains a form with the following fields: Sucursal (dropdown), Unidad (dropdown), Codigo (text), Nombre (text), Apellido (text), Direccion (text), Teléfono (text), Celular (text), and Correo (text). Below the form are buttons for 'Consultar', 'Nuevo', 'Actualizar', 'Eliminar', and 'Limpiar'. A table below the form lists existing employees:

Ciudad	NombreUnidad	CodigoEmp	NombreEmp	ApellidoEmp
La Paz	Almacenes	E00001	Juan Carlos	Quenta
La Paz	Almacenes	E00002	Bruno	Tarqui
La Paz	Tesoreria	E00003	Juan Pablo	Flores
La Paz	Tesoreria	E00004	Israel	Ballesteros
Cochabamba	Contabilidad	E00005	Jimena	Huanca
Cochabamba	Operaciones	E00006	Jose	Pardo

Figura 3.16: Pantalla registro de empleado
[Fuente: Elaboración Propia]

3.4.1.4 Registro de Grupo de Inventario

Previamente se tiene que elegir la opción de registrar grupo de inventario en el menú Inventario. En el registro se adicionan los datos de los activos fijos, después de llenar toda la información se tiene la opción de finalizar para terminar el registro, continuar para registrar los datos y cancelar en caso de cometer algún error cancelar el registro.

GRUPO LARCOS
INDUSTRIAL S.A.S.

GESTIÓN DE ACTIVOS FIJOS

Página Inicio | Inventario | Gestion de Inventario | Gestion de Grupo | Administración | Reportes

REGISTRO DE INVENTARIO

Rubro : - Seleccione un Rubro -
 Estado : - Seleccione un Estado -
 Código * :
 Descripción * :
 Marca :
 Modelo :
 Numero de Serie :
 Fecha Compra * :
 Precio Compra * :
 Observaciones * :

Consultar | Nuevo | Actualizar | Eliminar | Limpiar

Código	Descripción	Modelo	Numero de serie	Precio Compra	Fecha Compra
INV-	Equipo de Computación de	Torre - TX110	S/N.973365410	5950	12/06/2008 0:00:00

Figura 3.17: Pantalla registro de inventario de activos fijos
 [Fuente: Elaboración Propia]

3.4.1.5 Registro de Asignación de Inventario

Previamente se tiene que elegir la opción de registrar asignar de inventario en el menú Inventario. En el registro se adicionan los datos de la asignación de activos fijos, después de llenar toda la información se tiene la opción de finalizar para terminar el registro.

Figura 3.18: Pantalla registro de asignación de activos fijos
[Fuente: Elaboración Propia]

3.5 Fase de transición

La fase de transición se enfoca en liberar el sistema a producción. Para esta fase se completa con la aprobación y visto bueno del diseño e implementación del sistema, por parte de los usuarios.

Para el presente trabajo se aplicara el método corte cambio por las siguientes razones:

- Toma menos tiempo y esfuerzo
- El nuevo sistema esta funcionamiento inmediatamente.

3.6. Pruebas

La fase de pruebas es una de las fases más importantes ya que nos permite verificar junto al cliente que se pudo atender los requerimientos especificados en las historias de usuario, también sirven como retroalimentación para ver que las historias de usuario que fueron implementadas en versiones anteriores y que necesitan ser

modificadas, mejoradas o simplemente descartadas. El tipo de pruebas realizadas en esta fase son pruebas de aceptación descritas a continuación.

3.6.1 Pruebas de aceptación

Este tipo de pruebas fueron realizadas para cada entrega del software en las distintas iteraciones que se tuvo, ya que fueron definidas en el reverso de cada historia de usuario.

Las siguientes tablas muestran todas las pruebas de aceptación requeridas por el cliente en cada historia de usuario, además de la iteración en la cual fueron solucionadas correctamente. *Registro y Actualización de Inventario*: esta prueba busca encontrar todo tipo de errores que se generan durante el proceso de registro y actualización de nuevos inventarios en el sistema, ya sea en el llenado del formulario de registro con datos no validos o también por la falta de datos obligatorios.

Prueba de Aceptación	
Numero: 1	Usuario: Administrador
Nombre Historia: Registro y Actualización de Inventario	
Descripción: <ul style="list-style-type: none"> • Llenar el formulario olvidado introducir algún dato obligatorio y verificar que se muestra en el mensaje de error • Introducir contraseña distintas y verificar el control de las mismas • Introducir todos los datos correctamente e informarle que se completó su registro 	
Test de Aceptación: Aceptado por el cliente en la primera Iteración	

Tabla: 3.9 Prueba de aceptación
Registro y Actualización de Productos

Registro de Empleados: esta prueba busca encontrar todo tipo de errores que se generen durante el proceso de registro, actualización y eliminación de empleados y listado, además de la verificación que la información generada es almacenada correctamente.

Prueba de Aceptación	
Numero: 2	Usuario: Administrador
Nombre Historia: Registro y Actualización de Empleados	
Descripción: <ul style="list-style-type: none"> • Mostrar mensajes para introducir los datos obligatorios en casos de no ser llenados los campos • Mostrar mensajes de confirmación en la actualización de datos • Mostrar alertas cuando el administrador este apunto de eliminar un registro de un empleado 	
Test de Aceptación: Aceptado por el cliente en la primera Iteración	

Tabla 3.10 Prueba de aceptación:
Registro de Empleados

Asignación de Activos Fijos: Esta prueba busca encontrar todo tipo de errores que se generen durante el proceso de selección y asignación de los activos fijos.

Prueba de Aceptación	
Numero: 3	Usuario: Administrador

Nombre Historia: Asignación de Activos Fijos
Descripción: <ul style="list-style-type: none"> • Mostrar mensajes para introducir los datos obligatorios en casos de no ser llenados los campos • Mostrar mensajes de confirmación en la asignación de activos fijos • Mostrar alertas cuando el administrador este apunto de eliminar un registro de un activo fijo
Test de Aceptación: Aceptado por el cliente en la primera Iteración

Tabla 3.11 Prueba de aceptación:
Asignación de Activos Fijos

Depreciación de Activos Fijos: Esta prueba busca encontrar todo tipo de errores que se generen durante el proceso de Depreciación de activos fijos.

Prueba de Aceptación	
Numero: 3	Usuario: Administrador
Nombre Historia: Depreciación de Activos Fijos	
Descripción: <ul style="list-style-type: none"> • Mostrar mensajes para Seleccionar los datos obligatorios en • Mostrar mensajes de confirmación para la depreciación de activos fijo • Mostrar reportes de Depreciación de los Activos Fijos 	
Test de Aceptación: Aceptado por el cliente en la primera Iteración	

Tabla 3.12 Prueba de aceptación:
Depreciación de Activos Fijos

MÉTRICAS DE CALIDAD

4.1 Introducción

La calidad del software, es el desarrollo de software basado en estándares con la funcionalidad y rendimiento total que satisfacen los requerimientos del cliente. A continuación se describen los factores de calidad con el objeto de evaluar la calidad del software.

A la conclusión del Sistema Control y Gestión de Activos Fijos vía Web se debe medir la calidad del producto, hay dos tipos de medición: directa e indirecta. En el presente proyecto se aplicará las medidas indirectas planteadas por la norma ISO 9126

4.2 Características Propuestas por WEB site QEM

WEB site QEM plantea un modelo normalizado que permite evaluar y comparar productos sobre la misma base.

Aquí la calidad se define por seis características:

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

Fiabilidad: Capacidad del software de mantener su nivel de ejecución bajo condiciones normales en un periodo de tiempo establecido.

Usabilidad: Consiste de un conjunto de atributos que permiten evaluar el esfuerzo necesario que deberá invertir el usuario para utilizar el sistema.

Portabilidad: Evalúa la oportunidad para adaptar el software a diferentes ambientes sin necesidad de aplicarle modificaciones.

Mantenibilidad: Es el esfuerzo necesario para diagnosticar las deficiencias o causas de fallas, o para identificar las partes que deberán ser modificadas.

Eficiencia: Esta característica permite evaluar la relación entre el nivel de funcionamiento del software y la cantidad de recursos usados

4.2.1 Funcionalidad

Las funciones desarrolladas deben satisfacer las necesidades declaradas o implícitas [ISO 9126: 1991]

CÓDIGO	ATRIBUTO	CRITERIO ELEMENTAL	IE i(%)	PESO	VALOR
1.	FUNCIONALIDAD				91.47
1.1	Aspectos de Búsqueda y Recuperación			0.8	90
1.1.1	Mecanismos de Búsqueda en el sitio				90
1.1.1.1	Búsqueda Restringida	CB	100	0.8	90
1.1.1.2	Búsqueda global	CB		0.4	0
1.2	Aspectos de Navegación y exploración			0.4	92.95
1.2.1	Navegabilidad			0.35	97
1.2.1.1	Orientación			0.7	100
1.2.1.1.1	Indicador del Camino	CB	100	0.6	60
1.2.1.1.2	Etiqueta de Posición actual	CB	100	0.4	40
1.2.1.2	Promedio de enlaces por pagina			0.3	90
1.2.2*	Objetos de control de Navegación			0.35	100
1.2.2.2	Nivel de desplazamiento				100
1.2.2.2.1	Desplazamiento Vertical	CB	100	0.5	50
1.2.2.2.2	Desplazamiento Horizontal	CB	100	0.5	50
1.2.3*	Predicción navegacional			0.3	80

1.2.3.1	Enlace con título	CDP	80	0.3	24
1.2.3.2	Calidad de Frase de Enlace	CDP	80	0.7	56

Tabla 4.1: Resultado de la calidad del sistema Web. Funcionalidad

[Fuente: Elaboración Propia]

De acuerdo a la tabla 4.1 se puede ver que el grado de funcionalidad del sistema es de un 91.47% es decir funcione debidamente.

4.2.2 Fiabilidad

Capacidad de un sistema para mantener su nivel de rendimiento.

CÓDIGO	ATRIBUTO	CRITERIO ELEMENTAL	IE i(%)	PESO	VALOR
2	Fiabilidad				96.4
2.1	No deficiencia				96.4
2.1.1*	Errores de enlace			0.7	100
2.1.1.1	Enlaces rotos	CMN	100	0.5	50
2.1.1.2	Enlaces invalidas	CMN	100	0.3	30
2.1.1.3	Enlaces no implementadas	CMN	100	0.2	20
2.1.2*	Errores o deficiencias varias			0.3	88
2.1.2.1	Deficiencias o cualidades ausentes debido a deficiencias	CMN	60	0.3	18
2.1.2.2	Deficiencia o resultados inesperados independientes de browser (p ej. Errores de búsqueda imprevistos, deficiencias con marcos (frames), etc.)	CMN	100	0.3	30
2.1.2.3	Nodos destinos (inesperados en construcción)	CMN	100	0.1	10
2.1.2.4	Nodos muertos (sin enlaces de retorno)	CMN	100	0.3	30

Tabla 4.2 Resultado de calidad del sistema Web. Fiabilidad

[Fuente: Elaboración Propia]

De acuerdo a la tabla 4.2 se puede ver que el grado de fiabilidad del sistema es de un 96.4%. Por lo tanto existe un 3.6% de probabilidad de que el sistema falle debido a incompatibilidad mínimas con los navegadores.

4.2.3 Usabilidad

Esfuerzo necesario para el uso y la valoración individual de tal uso, por parte de un conjunto de usuarios. [ISO 9126: 1991]

CÓDIGO	ATRIBUTO	CRITERIO ELEMENTAL	IE i(%)	PESO	VALOR
3.	USABILIDAD				94.46
3.1	Comprensibilidad global del sitio			0.90	90
3.1.1	Esquema de organización global				90
3.1.1.1	Mapa del sitio	CB	0	0.2	0
3.1.1.2	Menú de contenidos	CB	100	0.9	90
3.1.1.3	Índice alfabético	CB	0	0.4	0
3.2	Mecanismos de ayuda y retroalimentación en línea			0.3	93.4
3.2.1*	Calidad de ayuda			0.4	72
3.2.1.1	Ayuda exploratorio Orientado al usuario	CDP	100	0.9	90
3.2.1.2	Ayuda de la búsqueda	CDP	50	0.5	25
3.2.2*	Indicador de última actualización			0.2	0
3.2.2.1	Global todo el sitio Web	CB	0	0.5	0
3.2.2.2	Restringido por subsitio o pagina	CB	0	0.5	0
3.2.3*	Retroalimentación			0.4	100
3.2.3.1	Formulario de entrada	CDP	100	0.9	90
3.2.3.2	Reportes	CDP	100	0.9	90
3.3	Aspectos de interfaces y Estéticos			0.3	100
3.3.1	Cohesividad al agrupar los objetos	CDP	100	0.3	30

3.3.2	Permanencia y estabilidad en la presentación de los controles principales	CB	100	0.3	30
3.3.3	Uniformidad en el estilo del sitio	CB	100	0.3	30

Tabla 4.3 Resultado de calidad del sistema Web. Usabilidad

[Fuente: Elaboración Propia]

De acuerdo a la tabla 4.3 se puede ver que el grado de usabilidad del sistema es de un 94.46% es decir que pueda ser usado correctamente y un 5.53% de que existan dificultades en el uso.

4.2.4 Eficiencia

Es la relación entre el nivel de presentación de un sistema y volumen de recursos utilizados en condiciones declaradas. [ISO 9126: 1991].

CÓDIGO	ATRIBUTO	CRITERIO ELEMENTAL	IE i(%)	PESO	VALOR
4.	EFICIENCIA				90.47
4.1	Performance			0.9	90
4.1.1	Páginas de acceso rápido	CDP	100	0.9	90
4.2	Accesibilidad			0.4	92.5
4.2.1	Accesibilidad de información				92.8
4.2.1.1	Soporte de Versión solo texto	CDP	80	0.6	48
4.2.1.2	Legibilidad al desactivar propiedad imagen en el browser			0.7	88
4.2.1.2.1	Imagen con titulo	CB	100	0.6	60
4.2.1.2.2	Legibilidad Global	CDP	80	0.8	72

Tabla 4.4 resultados de calidad del sistema Web. Eficiencia

[Fuente: Elaboración Propia]

De acuerdo a la tabla 4.4 se puede ver que el grado de eficiencia del sistema es de un 90.47%, esto quiere decir que existe un buen tiempo de respuesta y accesibilidad en la información.

4.2.5 Portabilidad

Es la capacidad de un sistema para ser transferido de un entorno a otro. [ISO 9126: 1991]. En este caso, se refiere a la habilidad del software de ser transferido de un ambiente a otro, dado que la plataforma fue desarrollada en una herramienta bajo la plataforma Windows solo se garantiza el funcionamiento en servidores con sistema operativo Windows server 2008 en adelante.

Debido a que el sistema se encuentra desarrollado con las herramientas de Visual Studio 2010 Asp.net, se ejecuta en el servidor Web IIS. Se ejecuta en distintos navegadores que tenga el cliente.

El servidor requerido para el funcionamiento del sistema necesita los siguientes requisitos mínimos: un procesador core i3 de 3.0 Ghz, memoria Ram de 6 GB y un disco duro de 500 GB.

La portabilidad del sistema también está dada por la siguiente relación:

$$X = \frac{A}{B}$$

Dónde:

A: Número de casos de éxito de la operación de instalación por parte del usuario

B: Número total de operaciones de instalación que realizó el usuario.

Reemplazo los valores en la relación anterior tenemos:

$$X = \frac{3}{3} = 1$$

Por tanto existe un 100% de probabilidad de que el sistema sea instalado con éxito.

Como se puede ver a nivel de software y de hardware el sistema es portable

4.2.6 Mantenibilidad

Es el esfuerzo necesario para realizar modificaciones específicas. [ISO 9126: 1991]

Según el estándar IEEE 982 que el índice de madurez proporciona una indicación de la estabilidad del software. Si el índice de madurez se aproxima 1 se puede decir que el sistema comienza a estabilizarse.

De acuerdo a la siguiente relación se puede determinar el índice de madurez de software (IMS):

$$IMS = \frac{[Ma - (Fa + Fb + Fc)]}{Ma}$$

Dónde:

IMS = índice de madurez del software

Ma = Numero de módulos en la versión actual

Fa = Numero de módulos en la versión actual que ha sido modificados.

Fb = Numero de módulos en la versión actual que han sido añadidos

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

Entonces:

$$IMS = \frac{[9 - (0 + 0 + 0)]}{10}$$

$$IMS = 0.9$$

Por tanto el sistema tiene una estabilidad de un 90%

La calidad total del sistema estará dada por:

$$QT = (91.47 + 96.4 + 94.46 + 90.47 + 100 + 90) = 93.8$$

De acuerdo al intervalo de aceptabilidad de calidad global de la metodología Web Site QEM:

INSATISFACTORIO [0% - 40%]

MARGINAL [40% - 60%]

SATISFACTORIO [60% - 100%]

Se puede decir que el sistema se encuentra dentro de un nivel SATISFACTORIO con un 93.8% de calidad global.

La siguiente tabla 4.5 muestra un resumen de los datos obtenidos para la calidad de software.

Criterio	Valor Porcentual	Interpretación
Funcionalidad	91.47%	El sistema funcione debidamente
Fiabilidad	96.4%	La probabilidad de que el sistema falle debido a incompatibilidades, son mínimas con el usuario
Usabilidad	94.46%	Pueda ser usado correctamente y tiene un 5.53% de que existan dificultades en el uso.
Eficiencia	90.47%	Quiere decir que existe un buen tiempo de respuesta y accesibilidad en la información
Portabilidad	100%	Como se puede ver a nivel de software y de hardware el sistema es

		portable en diferentes plataformas.
Mantenibilidad	90%	Por tanto el sistema tiene una estabilidad de un 90%.

Tabla 4.5: Resumen de Resultados

[Fuente: Elaboración Propia]

De acuerdo con el resultado obtenido del sistema en calidad de software podemos afirmar que el sistema responde en un 93.8% en calidad y tiene un 6.2 % de error probable que significa que el sistema será una herramienta esencial para la empresa en las tecnologías de información.



EVALUACIÓN DE COSTO BENEFICIO

5.1 Introducción

Son una parte importante de todo análisis de sistemas. El propósito, es mostrar a los usuarios del nuevo sistema, al igual que a otros grupos de administradores de la organización, que los beneficios que se espera obtener con el nuevo sistema superan a los costos esperados.

En las secciones siguientes examinaremos diversos aspectos de los cálculos de costo beneficio

- Análisis de costos.
- Análisis de beneficios.
- Cómo expresar los ahorros.
- Análisis de riesgo.

5.2 Análisis de costos

Se deben calcular todos los costos anticipados asociados con el sistema. Para determinar el costo total del proyecto se tomará en cuenta los siguientes costos:

- Costo del Software desarrollado.
- Costo de la implementación del sistema.
- Costo de la elaboración del proyecto.

5.2.1 Costo del software desarrollado

Para determinar el costo del software desarrollado, se utilizará el modelo constructivo COCOMO II, orientado a los puntos de función.

Estimación de puntos de función:

Parámetro de medición	Cuenta	Factor de ponderación	Total
No de entradas de usuario	11	4	44
No de salidas de usuario	13	5	65
No de peticiones de usuario	10	4	40
Nº de archivos	11	10	110
No de interfaces externas	2	7	14
Cuenta total			273

Tabla 5.1: Calculo de punto función no ajustado

[Fuente:
Elaboración propia]

Calculo de valores de ajuste de la complejidad tornando los valores de la Tabla 5.2, en la cual se determina la complejidad.

$$\text{Factor de Ajuste} = (0.65 + 0.01 * 51) \text{ Factor de Ajuste} = 1.16$$

Factores de complejidad	Valor
Requiere el sistema copias de seguridad y de recuperación fiables.	5
Se requiere comunicación de datos.	5
Existen funciones de procesamiento distribuido.	2
Es crítico el rendimiento.	2
Se ejecutará el sistema con un entorno operativo existente y fuertemente utilizado.	4

Requiere el sistema entrada de datos interactiva.	3
Facilidad operativa.	4
Se actualiza los archivos maestros de forma interactiva.	2
Son complejos las entradas, las salidas, los archivos o las peticiones.	3
Procesamiento interno complejo.	4
Diseño del código reutilizable.	3
Facilidad de instalación	5
Soporta múltiples instalaciones en diferentes sitios	5
Facilidad de cambios	4
Total	51

Tabla 5.2: Calculo de punto función ajustada
[Fuente: Elaboración propia]

El cálculo de los puntos de función se basa en la fórmula:

$$PF = \text{Cuenta total} * \text{Factor de Ajuste}$$

$$PF = 273 * 1.16$$

$$PF = 316.68$$

Conversión de los puntos de función a KDLC.

Ahora convertiremos los PF a miles de líneas de código. Para ello veremos la Tabla 5.3.

Lenguaje	Nivel	Factor LDC / PF
C	2.5	128
Ansi Basic	5	64
Java	6	53
Ansi Cobol	3	107
Visual Basic	7	46
ASP	9	36

PHP	11	29
Visual C++	9.5	34

Tabla 5.3 Conversión de puntos Función a KLDC
[Fuente: Elaboración propia]

$$LDC = PF * \text{Factor LDC} / PF \quad LDC = 316.68 * 36$$

$$LDC = 11400,48$$

$$KLDC=11.40$$

Aplicando las fórmulas básicas del esfuerzo, tiempo calendario y personal requerido. Las ecuaciones del COCOMO tienen la siguiente forma:

$$E = a_b (KLDC)^{b_b}$$

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

Dónde:

E: Esfuerzo aplicado en personas por mes.

D: Tiempo de desarrollo en meses.

KLDC: Número estimado de líneas de código distribuidas (en miles)

Proyecto de software	Ab	Bb	Cb	Db
Orgánico	2.4	1.05	2.5	0.38
Semi-acoplado	3	1.12	2.5	0.35
Empotrado	3.6	1.2	2.5	0.32

Tabla 5.4: Coeficientes Ab y Bb
[Fuente: Elaboración propia]

En la Tabla 5.4 se muestra los tipos de proyecto de software. Como este es un proyecto intermedio, en tamaño y complejidad, se elige Semi-acoplado.

$$E = 3 * (11.40)^{1.12}$$

$$E = 45.79$$

$$D = 2.5 * (45.79)^{0.35}$$

$$D = 9.53$$

El personal requerido se obtiene con la siguiente fórmula: Numero

$$\text{Programadores} = E/D$$

$$\text{Numero Programadores} = 45.79/9.53$$

$$\text{Numero Programadores} = 4.80 = 5$$

EL salario de un programador aproximadamente es de 300 \$us., cifra que se tornará en cuenta para la estimación siguiente:

$$\text{Costo Software} = \text{Numero Programadores} * \text{Salario Programador}$$

$$\text{Costo Software} = 5 * 300$$

$$\text{Costo Software} = 1500 \$us$$

5.2.2 Costo de la implementación del proyecto

Como la empresa Grupo Larcos Industrial LTDA cuenta con sus servidores de aplicaciones y base de datos completamente instalados y configurados, por lo que el costo de implementación es cero para el sistema.

5.2.3 Costo de elaboración del proyecto

Se refieren a los costos de estudio del sistema, en la etapa de análisis estos costos se representan en la siguiente tabla:

Descripción	Costo total (\$us)
Análisis y diseño del proyecto	250
Bibliografía	40
Material de escritorio	30
Otros	30
Total	350

Tabla 5.5: Costo de elaboración del proyecto
[Fuente: Elaboración propia]

5.2.4 Costo total

El costo total es la suma del costo de software de desarrollo y el costo de elaboración del proyecto, que se puede observar en la siguiente tabla:

Descripción	Costo total (\$us)
Costo de software de desarrollo	1500
Costo de implementación	0
Costo de elaboración del proyecto	350
Total	1850

Tabla 5.6: Costo total del proyecto
[Fuente: Elaboración propia]

5.3 Análisis de beneficios

Es mucho más difícil calcular los beneficios de un nuevo sistema de información que calcular su costo. La labor más grande al llevar a cabo un cálculo de costo-beneficio será acorralar a los usuarios y hacer que identifiquen beneficios tangibles que pueden medirse y calcularse de manera cuantitativa.

Los beneficios para el presente proyecto son de tipo intangible. Para realizar el análisis de beneficios se utilizarán cinco criterios de evaluación. Los cuales permiten evaluar el sistema diseñado respecto al sistema anterior estos son:

- Incremento de velocidad en los procesos.
- Capacidad en el volumen de información.
- Control de procesos.
- Integración de la información.
- Información para la toma de decisiones.

5.4 Valor neto actual

El Valor neto Actual (VAN) es un indicador financiero que mide los flujos de los futuros ingresos y egresos que tendrá un proyecto, para determinar, si luego de descontar la inversión inicial, nos quedaría alguna ganancia. Si el resultado es positivo, el proyecto es viable.

Basta con hallar VAN de un proyecto de inversión para saber si dicho proyecto es viable o no. El VAN también nos permite determinar cuál proyecto es el más rentable entre varias opciones de inversión. Incluso, si alguien nos ofrece comprar nuestro negocio, con este indicador podemos determinar si el precio ofrecido está por encima o por debajo de lo que ganaríamos de no venderlo.

La fórmula del VAN es: $VAN = BNA - Inversión$

Donde el beneficio neto actualizado (BNA) es el valor actual del flujo de caja o beneficio neto proyectado, el cual ha sido actualizado a través de una tasa de descuento.

La tasa de descuento (TD) con la que se descuenta el flujo neto proyectado, es ella tasa de oportunidad, rendimiento o rentabilidad mínima, que se espera ganar; por lo tanto, cuando la inversión resulta mayor que el BNA (VAN negativo o menor que 0) es porque no se ha satisfecho dicha tasa. Cuando el BNA es igual a la inversión (VAN igual a 0) es porque se ha cumplido con dicha tasa. Y cuando el BNA es mayor que la inversión es porque se ha cumplido con dicha tasa y además, se ha generado una ganancia o beneficio adicional.

$VAN > 0$ → el proyecto es rentable.

$VAN = 0$ → el proyecto es rentable también, porque ya está incorporado ganancia

de la TD. $VAN < 0$ → el proyecto no es rentable.

Entonces para hallar el VAN se necesitan:

- Tamaño de la inversión.
- Flujo de caja neto proyectado.
- Tasa de descuento.

Veamos un ejemplo:

El proyecto tiene una inversión de 1200 \$us. y una tasa de descuento (TD) de 10%, por defecto.

Hallando el VAN:

$VAN = BNA - Inversión$

$$VAN = 1450 / (1 + 0.14)^1 + 1450 / (1 + 0.14)^2 + 1450 / (1 + 0.14)^3 + 1450 / (1 + 0.14)^4 + 1450 / (1 + 0.14)^5 - 1450$$

$$\text{VAN} = 1425.69 - 1450$$

$$\text{VAN} = 151.69$$

5.5 Tasa interna de retorno

La TIR es la tasa de descuento (TD) de un proyecto de inversión que permite que el BNA sea igual a la inversión (VAN igual a 0). La TIR es la máxima TD que puede tener un proyecto para que sea rentable, pues una mayor tasa ocasionaría que el BNA sea menor que la inversión (VAN menor que 0).

Entonces para hallar la TIR se necesitan:

- Tamaño de inversión.
- Flujo de caja neto proyectado. Veamos un ejemplo:

El proyecto tiene una inversión de 1200.

Para hallar la TIR hacemos uso de la fórmula del VAN, sólo que en vez de hallar el VAN (el cual reemplazamos por 0), estaríamos hallando la tasa de descuento:

$$\text{VAN} = \text{BNA} - \text{Inversión}$$

$$0 = 1200 / (1+i)^1 + 1200 / (1+i)^2 + 1200 / (1+i)^3 + 1200 / (1+i)^4 + 1200 / (1+i)^5 - 1200$$

$$i = 10\% \quad \text{TIR} = 10\%$$

Si esta tasa fuera mayor, el proyecto empezaría a no ser rentable, pues el BNA empezaría a ser menor que la inversión. Y si la tasa fuera menor (como en el caso del ejemplo del VAN donde la tasa es de 10%), a menor tasa, el proyecto sería cada vez más rentable, pues el BNA sería cada vez mayor que la inversión.

6.1 Seguridad en el cliente

Uno de los mecanismos de seguridad que se implementan son las validaciones por el lado del cliente.

Existen mecanismos de validación provistas por las herramientas que utilizamos para hacer la aplicación, en el caso de asp.net se tienen los controles de validación para la información introducida por el cliente, estas validaciones son realizadas antes de que la información introducida llegue al servidor, esto evita que se envíen datos incorrectos al servidor, además se ahorra tiempo, ya que si la información es incorrecta simplemente no se envía al servidor.

6.2 Seguridad en el servidor

La validación del lado del cliente no es suficiente, también deben realizarse otro tipo de controles por el lado del servidor, ya sea del servidor de aplicaciones o del servidor de base de datos.

- Seguridad en el servidor de aplicaciones.

Un servidor de aplicaciones proporciona muchos servicios y no todos son necesarios para el funcionamiento de la aplicación web.

Es conveniente deshabilitar lo que no se necesite en el servidor de aplicaciones, para ello debe configurarse adecuadamente el servidor de aplicaciones.

- Seguridad en el servidor de base de datos.

Existen muchos problemas a nivel de base de datos, uno de ellos es la Inyección SQL (lenguaje de consulta estructurada), son los ataques realizados contra bases de datos. En este caso, un usuario utiliza debilidades en el diseño de la base de datos o de la página web para extraer información o más aún, para manipular información dentro de la base de datos. Una forma de evitar este tipo de debilidades es restringiendo los caracteres que el usuarios introduce, por ejemplo las comillas, los punto y coma y otros.

También está el acceso no autorizado a la información. Esto podría solucionarse asignando correctamente los roles que tiene cada usuario en el sistema. El acceso de los usuarios a los formularios web es cargado desde la base de datos, cuando un usuario se autentica el sistema le da acceso sólo a los formularios asignados para él.

6.3 Seguridad en la comunicación

SSL es un protocolo diseñado y propuesto por Netscape Communications Corporation. Se encuentra en la pila OSI entre los niveles de TCP/IP y de los protocolos HTTP, FTP, SMTP, etc. Proporciona sus servicios de seguridad cifrando Los datos intercambiados entre el servidor y el cliente con un algoritmo de cifrado simétrico, típicamente el RC4 o IDEA, y cifrando la clave de sesión de RC4 o IDEA mediante un algoritmo de cifrado de clave pública, típicamente el RSA. La clave de sesión es la que se utiliza para cifrar los datos que vienen del y van al servidor seguro. Se genera una clave de sesión distinta para cada transacción, lo cual permite que aunque sea reventada por un atacante en una transacción dada, no sirva para descifrar futuras transacciones. MD5 se usa como algoritmo de hash.

Proporciona cifrado de datos, autenticación de servidores, integridad de mensajes y, opcionalmente, autenticación de cliente para conexiones TCP/IP.

Cuando el cliente pide al servidor seguro una comunicación segura, el servidor abre un puerto cifrado, gestionado por un software llamado Protocolo SSL Record, situado encima de TCP. Será el software de alto nivel, Protocolo SSL Handshake, quien utilice el Protocolo SSL Record y el puerto abierto para comunicarse de forma segura con el cliente.

6.4 Seguridad en la aplicación

El control de acceso de los usuarios es una parte fundamental para una aplicación web.

- Autenticación, determina si un usuario es quien dice ser.

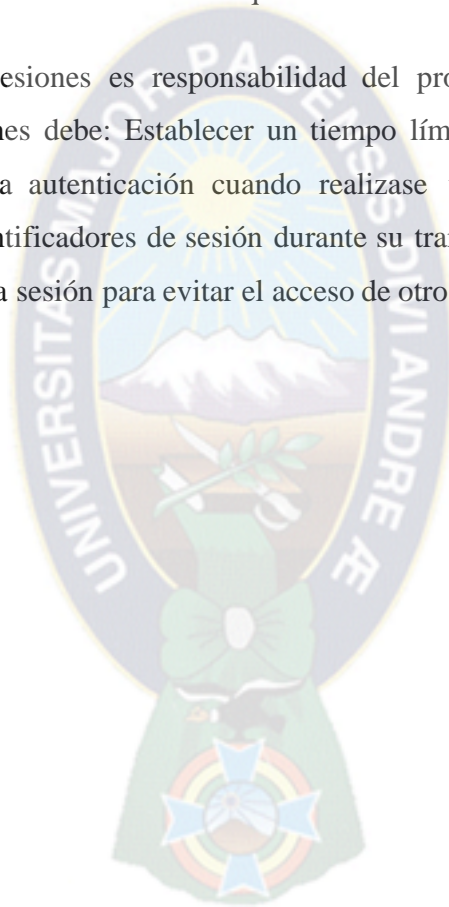
Autenticación HTTP básica, cuando se quiere ingresar a un formulario protegido el servidor devuelve un código "HTTP/1.1 401 Authorization required". El cliente debe enviar sus datos al servidor.

Esta autenticación es fácil de implementar, pero los datos viajan por la red sin encriptar, no se puede "cerrar sesión", la única forma es cerrar el navegador.

- Autenticación basada en la aplicación, en este caso la aplicación implementa su propio mecanismo de autenticación. Es más costosa pero es más flexible porque permite establecer diferentes permisos y niveles de acceso asignados al usuario.
- Passwords, se recomienda restringir los valores para los nombre de los usuarios.

Almacenar los passwords de forma segura protegiendo el acceso a la base de datos. Bloquear una cuenta cuando se detecta un número determinado de intentos de acceso incorrectos. Tener una política de recuperación de passwords en caso de olvido por parte del usuario.

- Sesiones, después de que el usuario se ha autenticado se debe mantener esta autenticación en cada conexión subsiguiente. Para esto se utilizan las variables de sesión, que permiten mantener el estado entre las diferentes peticiones HTTP. El procedimiento es el siguiente: Después de autenticarse el usuario recibe un identificador de sesión, este identificador es invisible y lo acompañará en cada petición. Este identificador se almacena en la máquina del cliente, mediante una cookie.
- La gestión de sesiones es responsabilidad del programador. Un sistema de gestión de sesiones debe: Establecer un tiempo límite de vida para la sesión, pedir una nueva autenticación cuando realizase una operación importante, proteger los identificadores de sesión durante su transición y destruir la cookie cuando finalice la sesión para evitar el acceso de otro usuario.



CONCLUSIONES Y RECOMENDACIONES

Durante el desarrollo del presente proyecto, se pudo evidenciar que la implementación de un sistema web basado en el control y gestión de los activos fijos no es una tarea sencilla pero es imprescindible adecuar los procesos que tienen para mejorar las perspectivas empresariales del mercado en la actualidad.

Finalmente este capítulo detalla las conclusiones y recomendaciones a las cuales llegaremos con la culminación del proyecto, así como las recomendaciones pertinentes.

6.1 Conclusiones

A la culminación del presente proyecto y conforme a las actividades definidas para el análisis e implementación del Sistema Control y Gestión de Activos Fijos vía Web caso: Grupo Larcos Industrial LTDA. Se concluye que:

- El producto obtenido cuenta con todas las características requeridas por los usuarios, resultando una herramienta de ayuda para los procesos que se efectúan
- Se logró centralizar y almacenar la información de los activos fijos, empleados para tener acceso inmediato.
- Las Depreciaciones se realizan automáticamente en el sistema, permitiendo ahorro de tiempo al momento de realizar un reporte
- El sistema lleva un control de todos los movimientos y cambios de los activos fijos, así de esta manera poder hacer un seguimiento.
- Con el módulo de asignación se logró obtener la información de los activos fijos asignados a los empleados, que indicara a que sucursal y unidad pertenece, también se podrá verificar en qué condiciones se encuentra los activos fijos

6.2 Recomendaciones

Para ampliar el presente proyecto de grado, se hacen las siguientes recomendaciones:

- Se recomienda implementar un módulo de registro de código QR
- Se deberá cambiar la contraseña semanalmente o mensualmente para dar mayor seguridad al sistema.
- Se recomienda enlazar el sistema web con el sistema contable para tener la unificación de la información.



FUENTES DE INFORMACIÓN

TEXTOS

- [Booch & Jacobson, 1999] Booch, G., J., R., & Jacobson. (1999). Proceso Unificado de Desarrollo de Software.
- [Fowler & Kendall, 1997] Fowler, M., & Kendall, S. (1997). UML Gota a Gota. México DF: Addison Wesley Longman
- [Terán, 2000] Terán Gandarillas, G. (1997). Contabilidad Básica 1ra Edición. La Paz Bolivia.
- [Kendall, 1997] Kendall, K. &. (1997). Análisis y Diseño de Sistemas. New York: Services of New England Inc.
- [Mendoza, 2008] Mendoza, M. S. (2008). Análisis Services. Perú: Megabyte.
- [Mínguez & García, 2000] Mínguez, D., & García, E. (2000). Metodología para el Desarrollo de Aplicaciones Web: UWE.
- [Piatini, 1999] Piatini, M. (1999). Fundamentos y Modelos Bases de Datos (2da ed.). México DF: Alfa omega.

[Prentice, 1999] Prentice, L. (1999). UML y Patrones Introducción al Análisis y Diseño Orientado a Objetos

[Pressman, 2006] Pressman, R. (2006). Ingeniería de Software un Enfoque Práctico (sexta edición ed.). México DF

[Rumbaugh, Jacobson & Booch, 2000] Rumbaugh, J., Jacobson, I. & Booch, G. (2000). El Lenguaje Unificado de Modelo. Madrid: Pearson.

SITIOS DE INTERNET

[AISI, 2007] Asociación de investigación en software inteligente AISI. (2007). Ingeniería Web. Recuperado el 4 de Agosto de 2011, de <http://www.AISI.html>.

[Ambler, 2000] Ambler, S. (12 de Marzo de 2010). Metodologías Proceso Ágil. Recuperado el 15 de Junio de 2011, de Metodologías del Proceso Ágil : www.agilernodeling.com/essays!agileModelingRUP.html.

[ICT, 2011] Academia Latinoamericana de Seguridad Informática. (28 de

Enero de 2013). Implementación y plan de seguridad. Recuperado el 1 de Noviembre de 2013, de <http://www.pirarnidedigital.com/Docurnentos/ICT/pdictseguridadinformaticaimplementacion>

[Copley, 2010] Copley, S. (25 de Enero de 2010). Implementación del Nuevo sistema. Recuperado el 5 de Noviembre de 2013, de <http://www.igcseict.info/theory/8/implernl>

[IEEE, 2011] IEEE, Norma IEEE 9000 estándar. Recuperado el 15 Noviembre del 2013, de <http://www.ieee.org>.

[Koch, 2011] Unit of programing and software engineering. (11 de Julio de 2013). UWE-UML- Based Web Engineering. Recuperado el 10 de Agosto de 2011, de <http://uwe.pst.ifi.lmu.de/examples.html>.

Anexo A

Análisis de Situación

Introducción.

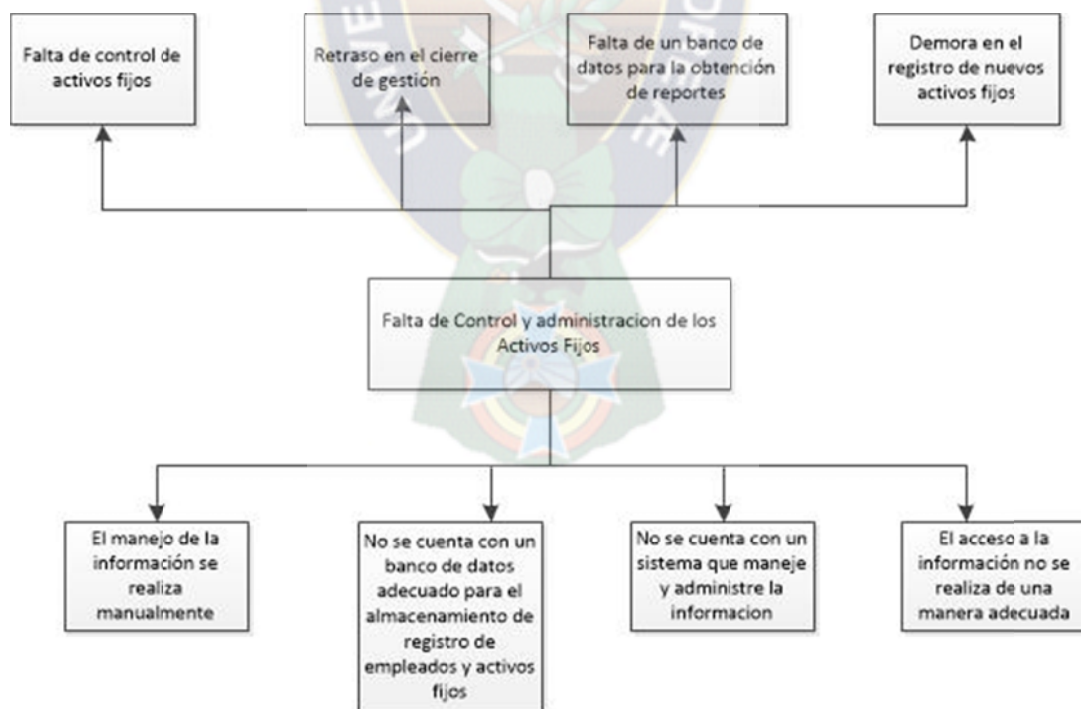
El tema a desarrollar es acerca del control y gestión de activos fijos vía web, para poder llevar un control y seguimiento del mencionado. Con el control y seguimiento del mismo se espera hacer llegar resultados fidedignos para la toma de decisiones para la empresa Grupo Larcos Industrial Ltda.

1. Titulo

"Control y Gestión de activos fijos vía web Caso: Grupo Larcos Industrial Ltda.".

2. Problema

2.1 Árbol de problemas



2.2 Problema Principal

Realizar el control y el seguimiento del personal.

2.3 Causas

- No se hace un control adecuado de permisos, faltas, vacaciones y otros.
- Trabajo manual en las liquidaciones.
- Falta de control de diversos cargos.

2.4 Efectos

- El incremento de sueldos y salarios no son automáticos.
- Mala toma de decisiones.
- No se tiene integración entre las áreas de liquidación y personal.

3. Objetivo

3.1 Árbol de Objetivos

