

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



PROYECTO DE GRADO
SISTEMA DE CONTROL Y SEGUIMIENTO DE ACTIVOS FIJOS PARA
LA EMPRESA AUTOVENTA INTERCAMBIO

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

POSTULANTE: UNIV.BEATRIZ CALLE LUNA
TUTOR: LIC. GROVER ALEX RODRÍGUEZ RAMÍREZ
REVISOR: LIC. CARLOS MULLISACA

LA PAZ- BOLIVIA

2011

DEDICATORIA

A Díos, La razón de mí existencia, por brindarme fortaleza y moldear mí vida, por permitirme alcanzar una vez más una de mis metas.

A mis padres T'oríbío Calle e Isabel Luna por su gran amor, paciencia y dedicación; por enseñarme el valor de la vida y encaminarme en ella, a mis hermanos por la comprensión y el apoyo que me Brindaron.

A mí tío Néstor y mis abuelos que a Cado de Dios me cuidan y me protegen.

A todos mis amigos quienes siempre estuvieron conmigo en las buenas y en las malas, por haber compartido grandes experiencias y sobre todo haber aprendido de ellas; esperando que con el andar de la vida siempre pueda contar con su amistad sincera como hasta ahora.

'Beatriz Calle Luna

AGRADECIEMIENTOS

Primeramente agradezco a Dios por que con su espíritu me ilumino para seguir adelante.

Agradecer a mí familia, quienes fueron impulsores para que mí persona pueda culminar con mis estudios.

Agradecer a mí docente Tutor Lic. Grove Rodríguez, por haber contribuido con la revisión y observación en el desarrollo de este proyecto.

Agradecer a mí docente "Revisor Lic. Carlos Mullisaca, por brindarme su tiempo, comprensión, paciencia, asesoramiento y revisión y durante todo el proceso de desarrollo hasta la culminación de este proyecto.

A mis amigos con quienes compartimos momentos inolvidables e hicimos el compromiso de culminar nuestros estudios universitarios de la mejor forma posible, a todos los mencionados anteriormente mis agradecimientos más sinceros.

Beatriz Calle Luna

RESUMEN

La Empresa Autoventa Intercambios dedicada a la venta, intercambio y renta de automotores cuenta con una sección de Activos Fijos que depende de la unidad Administrativa de la Dirección Administrativa y Financiera, que tiene la responsabilidad de administrar, clasificar, mantener y distribuir los activos fijos de la empresa.

La metodología adoptada para el desarrollo del software fue el Proceso Unificado de Rational RUP, que tiene como notación al Lenguaje de Modelado Unificado UML, que modela las distintas etapas para el desarrollo del nuevo sistema de software. Las herramientas utilizadas para la implementación son: el lenguaje de programación PHP y el gestor de base de datos MySQL.

Los módulos principales del nuevo sistema son: Registro, Asignación, Transferencia, Revalúo, Bajas, Inventarios, Historial de Mantenimiento, Historial de Transacciones, Adm. de parámetros, generación de actas y reportes de activos fijos.

Concluido el desarrollo del software se realizaron pruebas de funcionamiento, con las que se pudo constatar que el sistema responde a los requerimientos institucionales, posibilitando la reducción de tiempo y costos en la administración de activos fijos, además de la obtención de datos confiables.

ÍNDICE

<u>CONTENIDO</u>	<u>PAG.</u>
1. CAPITULO I: MARCO REFERENCIAL	1
1.1. INTRODUCCIÓN	2
1.2. ANTECEDENTES	3
1.2.1. Del Proyecto	3
1.2.2. De la Institución.....	3
1.3. PROBLEMÁTICA.....	4
1.3.1. Planteamiento del Problema	4
1.3.2. Definición del Problema.....	5
1.4. OBJETIVOS	5
1.4.1. Objetivo General.....	5
1.4.2. Objetivos Específicos.....	6
1.5. JUSTIFICACIONES	6
1.5.1. Justificación Técnica	6
1.5.2. Justificación Económica	7
1.5.3. Justificación Social	7
1.6. MÉTODOS Y TÉCNICAS	7
1.6.1. Metodología.....	7
1.6.2. Técnicas y herramientas.....	8
1.7. ALCANCES Y APORTES	8
1.7.1. Alcances.....	8
1.7.2. Aportes	9
2. CAPITULOII: MARCO TEÓRICO	10
2.1. INTRODUCCIÓN	11
2.2. METODOLOGIA EML	11
2.2.1. Los pasos clásicos del EML	11

a).	Análisis de Involucrados	11
b).	Análisis de Problemas.....	12
c).	Análisis de Objetivos.....	12
d).	Análisis de Alternativas	12
2.2.2.	Matriz de Marco Lógico	13
2.3.	LENGUAJE UNIFICADO DE MODELADO UML	13
2.3.1.	Vistas UML	14
a).	Vista de Casos de Uso.....	15
b).	Vista de Diseño	15
c).	Vista de Procesos	15
d).	Vista de Despliegue	15
e).	Vista de Implementación.....	15
2.4.	PROCESO UNIFICADO DE RATIONAL (RUP)	16
2.4.1.	Dirigido por Casos de Uso	16
2.4.2.	Centrado en la Arquitectura	17
2.4.3.	Iterativo e Incremental.....	18
2.4.4.	Vida del Proceso Unificado	18
a).	Fase de Inicio.....	20
b).	Fase de Elaboración	20
c).	Fase de Construcción	20
d).	Fase de Transición	22
2.4.5.	Flujos de Trabajo del Desarrollo de Software.....	23
a).	Modelado del Negocio	23
b).	Captura de Requerimientos.....	25
c).	Análisis del Sistema.....	28
d).	Diseño del Sistema	30

e).	Implementación del Sistema	31
f).	Pruebas del Sistema	32
g).	Despliegue del Sistema	33
2.5.	MÉTRICAS DE CALIDAD.....	34
2.5.1.	Funcionalidad	34
2.5.2.	Confiabilidad	35
2.5.3.	Mantenimiento	36
2.5.4.	Usabilidad.....	36
2.5.5.	Portabilidad.....	37
2.6.	ACTIVOS FIJOS	37
2.6.1.	Categorías importantes de los activos fijos	38
a).	Activos Fijos tangibles.....	38
b).	Activos Intangibles.....	38
c).	Recursos Naturales	39
2.6.2.	Clasificación General	39
a).	Bienes inmuebles.....	39
b).	Bienes muebles	39
2.6.3.	Clasificación Contable	40
a).	Activos No depreciables.....	40
b).	Activos depreciables	40
2.6.4.	Clasificación Especial	40
a).	Edificios	41
2.6.5.	Depreciación de Activos Fijos.....	41
2.6.6.	Revalorización de Activos Fijos	42
2.7.	ESTIMACIÓN DE COSTOS EN PROYECTOS SOFTWARE	43
2.8.	RECURSOS TECNOLÓGICOS	45

2.8.1. Pre procesador de Hipertexto PHP.....	45
2.8.2. Lenguaje de Consultas MySQL.....	47
3. CAPITULO III: MARCO APLICATIVO	49
3.1. INTRODUCCIÓN	50
3.2. FASE DE INICIO.....	50
3.2.1. Planificación de la Fase de Inicio	51
3.2.2. Modelo del Negocio	51
a). Descripción de Actores.	51
b). Modeladodel Negocio.....	52
3.2.3. REQUERIMIENTOS DEL SISTEMA.....	55
3.2.4. CASOS DE USO SISTEMA SCSAF	56
3.2.4.1 Identificación de Actores.....	57
3.2.4.2 Casos de uso del sistema..	58
3.2.5. EVALUACIÓN DE LA FASE DE INICIO	63
3.3.FASE DE ELABORACIÓN	63
3.3.1. Planificación de la Fase de Elaboración	63
3.3.2. Análisis de l Sistema.....	64
3.3.2.1 Diagramas de secuencias.....	64
3.3.3. Diseño del Sistema.....	67
3.3.3.1 Modelo de caso de uso Diseño Final.....	67
3.3.3.2 Modelo entidad-relación diagrama de clases.....	68
3.3.3.3 Modelode Datos.....	71
3.3.4. Evaluación de la Fase de Elaboración.....	71
3.5. FASE DE CONSTRUCCIÓN.....	71
3.4.1. Planificación de la Fase de Construcción	71
3.4.2 Implementación del Sistema.....	72
a). Diagrama de Componentes.....	73
b). Interfaz Gráfica de Usuario	74

3.4.3. Pruebas del Sistema	77
a). Prueba de la Caja Blanca.....	77
b). Prueba de la Caja Negra	86
3.4.4. Evaluación de la Fase de Construcción.....	87
3.5. FASE DE TRANSICIÓN.....	87
3.5.1. Planificación de la Fase de Transición	87
3.5.2. Despliegue del sistema	88
a). Diagrama de Despliegue	88
3.5.3. Evaluación de la Fase de Transición.....	91
4. CAPITULO IV: MÉTRICAS DE CAIDAD	92
4.1. INTRODUCCIÓN	93
4.2. USABILIDAD.....	94
4.3. MANTENIBILIDAD	95
4.4. FUNCIONALIDAD.....	96
4.5. CONFIABILIDAD	99
4.6. PORTABILIDAD...	101
5. CAPITULO V: EVALUCION DE COSTOS YBENEFICIOS	102
5.1. INTRODUCCIÓN.....	103
5.2. ESTIMACIONES DEL PROYECTO, MODELO COCOMO.....	103
5.2.1 ESTIMACIÓN DEL TAMA NO DEL SOFTWARE	103
5.2.2 ANÁLISIS DE COSTOS.....	106
5.2.2.1 COSTO DE LA FASE DE INICIO	106
5.2.2.2 COSTO DE LA FASE DE ELABORACIÓN.....	106
5.2.2.3 COSTO DE LA FASE DE CONSTRUCCIÓN.....	106
5.2.2.4 COSTO DE LA FASE DE TRANSICIÓN	107
5.2.3 ANÁLISIS DE RIESGOS.....	108
5.2.3.1 Proyección del Riesgo	108
6. CAPITULO VI SEGURIDAD DEL SISTEMA	110

6.1. INTRODUCCIÓN.....	111
6.2. CONFIDENCIALIDAD	111
6.3. INTEGRIDAD	111
6.4. DISPONIBILIDAD	112
6.5. AUTENTICIDAD	112
6.6. AUDITABILIDAD	112
7. CAPITULO VII CONCLUSIONES	113
8. CAPITULO VII RECOMENDACIONES.....	115
BIBLIOGRAFÍA	117
REFERENCIA BIBLIOGRÁFICA	118

ANEXOS

MATRIZ DE PROBLEMAS ÁRBOL DE
PROBLEMAS ÁRBOL DE OBJETIVOS
MATRIZ DE MARCO LÓGICO

ÍNDICE DE FIGURAS

<u>FIGURA</u>	<u>PAG.</u>
Figura 2.1: Modelo de la arquitectura de un Sistema mediante vistas.....	14
Figura 2.2: Los casos de uso integran el Trabajo	17
Figura 2.3: Evolución de la Arquitectura del Sistema	17
Figura 2.4: Proceso Iterativo e Incremental.....	18
Figura2.5: RUP en dos dimensiones.....	19
Figura 2.6: Representación Gráfica de un actor.....	26
Figura 2.7: Representación Gráfica de un Caso de Uso	26
Figura 2.8: Ejemplo: Diagrama de Estado de un Caso de Estudio	29
Figura 2.9: Ejemplo: diagrama de Secuencia de un Caso de estudio.....	30
Figura 2.10: Ejemplo: Diagrama de Clase de un Caso de Estudio	31
Figura 2.11: Ejemplo: Diagrama de Componentes de un Caso de Estudio....	32
Figura 2.12: Ejemplo: Diagrama de Despliegue de un Caso de Estudio	33
Figura 3.1: Casos de Uso del Modelo del Negocio: Adm. de Activos Fijos...	53
Figura 3.2: Diagrama de Caso de Uso: Sistema SIGSAF	58
Figura 3.3: Diagrama de Caso de Uso: Renta de Activos Fijos.....	59
Figura 3.4: Diagrama de Caso de Uso: Registro de Activos Fijos	60
Figura 3.5: Diagrama de Caso de Uso: Inventario de Activos Fijos	61
Figura 3.6: Diagrama de Caso de Uso: Mantenimiento de Activos Fijos...	62
Figura 3.7: Diagrama de Secuencia: Renta de Activos Fijos	65

Figura 3.8: Diagrama de Secuencia: Registro de Activos Fijos.....	66
Figura 3.9: Diagrama de Secuencia: Mantenimiento de Activos Fijos	66
Figura 3.10: Diagrama de Secuencia: Inventario de Activos Fijos.....	67
Figura 3.11: Diagrama de Caso de Uso Real: Sistema SCSAF	68
Figura 3.12: Modelo Entidad-Relación: Diagrama de Clases del Sistema....	69
Figura 3.13: Modelo Físico: Diseño de la Base de Datos del Sistema	70
Figura 3.14: Diagrama de Componentes del Sistema	73
Figura 3.15: Interfazde Usuario: Pagina Inicial.....	74
Figura 3.16: Interfazde Usuario: Registro de Nuevo Activo.....	75
Figura 3.17: Interfaz de Usuario: Baja de Activos.....	75
Figura 3.18: Interfaz de Usuario: Asignación de Activos Fijos.....	76
Figura 3.19: Interfaz de Usuario: Modificación de Activos Fijos	76
Figura 3.20: Diagrama de Flujo: Recepción Por.....	78
Figura 3.21: Grafo Flujo: Recepción Por	79
Figura 3.22: Diagrama de Flujo: Asignación y Liberación de Responsables. ...	82
Figura 3.23: Grafo Flujo: Asignación y Liberación de Responsables.....	83
Figura 3.24: Diagrama de Despliegue del Sistema	89

ÍNDICE DE TABLAS

<u>TABLA</u>	<u>PAG.</u>
Tabla 2.2: Descripción caso de uso expandido	27
Tabla 2.3: Coeficientes Cocomo	44
Tabla 3.1: Plan de actividades de la Fase de Inicio	51
Tabla 3.2: Funciones de Requerimientos del Sistema de Activos Fijos.....	56
Tabla 3.3: Relación entre actores y procesos en los cuales participan.....	57
Tabla 3.4: Caso de Uso expandido: Asignación de Activos Fijos.....	59
Tabla 3.5: Caso de Uso expandido: Registro de Activos Fijos	60
Tabla 3.6: Caso de Uso expandido: Inventario de Activos Fijos.....	61
Tabla 3.7: Caso de Uso expandido: Mantenimiento de Activos Fijos	62
Tabla 3.8: Plan de actividades de la Fase de Elaboración.....	64
Tabla 3.9: Plan de actividades de la Fase de Construcción.....	72
Tabla 3.10: Plan de actividades de la Fase de Transición.....	88
Tabla 4.1: Encuesta de Usabilidad del Sistema	94
Tabla 4.2: Cuenta Total con Factor de Ponderación Medio	97
Tabla 4.3: Valores de ajuste de complejidad.....	98
Tabla 5.1: Tabla del rango de Estimación	104
Tabla 5.2: Tabla de estimación del método LDC	105
Tabla 5.3: Tabla de Descripción de los costos de desarrollo del sistema	108



CAPITULO I

MARCO REFERENCIAL

1. MARCO REFERENCIAL

1.1. INTRODUCCION.

En la actualidad, los sistemas basados en computadoras se constituyen como las herramientas principales para satisfacer las constantes necesidades en las organizaciones. Debido a ello, surgen muchas inquietudes respecto a las formas de manejar estos instrumentos y aplicarlas, con el fin de mejorar la productividad y el logro de los objetivos, que les permite adecuarse al avance y seguir vigentes en los permanentes cambios que ocurren en nuestra sociedad.

La empresa Autoventa Intercambios que cuenta con una sección de Activos Fijos que se encarga de administrar las actividades y procedimientos relativos al ingreso, asignación, registro, mantenimiento, depreciación y control de los bienes (muebles, vehículos, equipos de computación, inmuebles).

Es así, que el presente proyecto de grado pretende resolver varios problemas que se observa al realizar el seguimiento y control físico de los bienes; aplicando técnicas y métodos para su diseño, con un lenguaje de codificación y plataforma de software de alto nivel para su implementación.

De esta manera el sistema nos permitirá obtener información periódica importante, respecto: las altas, bajas, devoluciones, transferencias y depreciaciones, ordenando y clasificando todos los activos fijos.

1.2. ANTECEDENTES.

Para elaborar el presente proyecto se reviso los siguientes antecedentes:

1.2.1. DEL PROYECTO.

Los siguientes trabajos tienen relación al proyecto propuesto:

- El proyecto “Seguimiento y Control Físico de Activos”, desarrollado por Navia (2000), que tiene como objetivo, dar solución a la problemática de desorganización a causa del procesamiento manual de la información y utiliza el método estructurado en el análisis y diseño del sistema.
- El proyecto “Sistemas de Información y Control de Activos Fijos”, desarrollado por Ticona (2000), que tiene por objetivo implementar un sistema informático que mejore la administración y control de Activos Fijos, promoviendo la racionalidad en la distribución de estos bienes a los servidores de la entidad para que cumplan sus funciones a cabalidad.
- “Sistema de información de Activos Fijos y Almacenes-Cámara Nacional de Comercio”, desarrollado por Choque (2003), cuyo propósito es desarrollar un sistema que proporcione información confiable, ágil y oportuna la administración y adoptar políticas institucionales sobre Activos fijos y almacenes

1.2.2. DE LA INSTITUCIÓN.

La Empresa Autoventa Intercambios dedicada a la venta, intercambio y renta de automotores cuenta con una sección de Activos Fijos que depende de la unidad Administrativa de la Dirección Administrativa y Financiera, que tiene la responsabilidad de administrar, clasificar,

mantener y distribuir los activos fijos de la empresa. Dicha Sección cuenta con un sistema de Información de Activos Fijos que fue desarrollado en FOX PRO, el mismo que proporciona información que no responde a las nuevas necesidades del personal y a las exigencias de las normas básicas del Sistema de Administración de Bienes y Servicios.

1.3. PROBLEMÁTICA

Existe deficiencia en el manejo y control de los Activos Fijos, con lo que la información de los mismos no es integra, lo cual dificulta en la actualización de la información de estos activos con referencia a esta asignación y devolución de los mismos en sus diferentes áreas de trabajo.

1.3.1. PLANTEAMIENTO DEL PROBLEMA

Se cuenta actualmente con un sistema de control de Activos Fijos y en base a la investigación previa realizada, Matriz de problemas Árbol de problemas (ver anexo A) de los cuales se detectaron los siguientes problemas:

- No se tiene un registro de los Activos prestados y/o rentados que tiene la empresa.
- Carencia de un sistema de codificación automático.
- No se cuenta con registro histórico de los activos que han sido revalorizados o transferidos.
- Errores de cálculos en la depreciación del activo fijo.
- No se cuenta con todos los requerimientos, normativas de manejo, y seguridad de los Activos Fijos, como especifica la norma Básica de Administración de Bienes y Servicios.

- Aplicaciones en FOX PRO no ofrecen entorno visual y amigable.
- Demora en la elaboración de reportes e informes para proporcionar a la toma de decisiones de los departamentos vinculados.
- No se cuenta con información detallada del activo fijo.

1.3.2. DEFINICIÓN DEL PROBLEMA.

Luego de considerar los problemas mencionados anteriormente se define el siguiente problema principal:

¿Cómo se puede realizar el seguimiento y control de Activos Fijos en la empresa Autoventa Intercambios que brinde información consistente y confiable, cumpliendo los requerimientos y normativas básicas del sistema de administración de bienes y servicios?

1.4. OBJETIVOS.

Para definir los objetivos se desarrollo un Árbol de objetivos (ver anexo A).

1.4.1 OBJETIVO GENERAL.

Desarrollar e implementar un sistema informático para el control y seguimiento de Activos Fijos para la empresa Autoventa Intercambios, el cual cumpla con los requerimientos y normativas básicas del sistema de administración de bienes y servicios.

1.4.2. OBJETIVOS ESPECÍFICOS.

- Diseñar un módulo de registro de activos fijos, bajas, modificaciones, revalorizaciones, transferencias, con la respectiva generación de documentos de respaldo si corresponde.
- Desarrollar un módulo para registrar a los clientes que rentan un automotor (activo fijo).
- Elaborar un algoritmo para obtener el valor neto de activos fijos, mejorando el cálculo de las depreciaciones, aplicando el método de la línea recta.
- Elaborar un módulo de asignación y liberación de responsable, que genere el acta de entrega y recepción de activos fijos.
- Desarrollar un módulo para generar informes que proporcione datos exactos del historial de transacciones de cada activo fijo (registro, bajas, modificaciones, transferencias, revaluó, liberaciones, mantenimiento).

1.5. JUSTIFICACIONES.

1.5.1. JUSTIFICACIÓN TÉCNICA.

La Sección de Activos Fijos cuenta con un ambiente de trabajo y equipos necesarios para la implementación del sistema, en el área de sistemas se cuenta con equipos de computación para el desarrollo del presente Proyecto.

1.5.2. JUSTIFICACIÓN ECONÓMICA.

Para obtener un informe más completo es necesario la búsqueda y ubicación de documentos lo cual invierte tiempo. El nuevo sistema permitirá incrementar beneficios con la mejora del manejo de la información, reduciendo la pérdida de tiempo, generar reportes a unidades que lo necesiten y así disminuir costos.

1.5.3. JUSTIFICACIÓN SOCIAL.

Se beneficiara el personal de la Sección de Activos Fijos que se encarga de interactuar con el sistema, con la mejora de sus actividades cotidianas relativos al ingreso, asignación, mantenimiento, registro y control de los Activos Fijos y favorecerá también a otras áreas que necesiten información y realicen consultas, como ser la Dirección Administrativa Financiera, el área de contabilidad.

1.6. METODOS Y TECNICAS.

Para cada etapa de desarrollo del software se implemento varios métodos y técnicas que se detallan a continuación.

1.6.1. METODOLOGÍA.

Para definir el problema y objetivo se utilizó la metodología de Enfoque de Marco Lógico.

Para el desarrollo del proyecto se utiliza el RUP (Proceso Unificado de Rational), como metodología para diseñar el ciclo de vida del desarrollo del sistema de software, por cumplir y ajustarse a las necesidades del Proyecto, adaptándose esta metodología a proyectos de diferente complejidad.

1.6.2. TÉCNICAS Y HERRAMIENTAS.

Las técnicas que se utilizan son:

- Para el modelo de análisis y diseño se utiliza UML (lenguaje de modelado unificado): Nos basaremos en su notación para la construcción de los diagramas para visualizar el sistema.

Las herramientas tecnológicas que se utilizaran en este proyecto son:

- Gestor de Base de Datos MySql, el lenguaje PHP.
- El sistema operativo Windows XP, y Windows Server NT.

1.7. ALCANCES Y APORTES.

1.7.1. ALCANCES.

El sistema de control y seguimiento de activos fijos para la empresa Autoventa Intercambios, será capaz de realizar las siguientes tareas:

- Registro de los Activos Fijos nuevos de forma dinámica.
- Asignación de Activos Fijos, a un responsable quien responda por el estado del activo fijo.
- Liberación a un usuario de la responsabilidad de un activo en común.
- Transferencia de Activos Fijos, de un responsable a otro.
- Depreciación de Activos Fijos, tomando en cuenta los activos fijos nuevos, existentes, revalorizados.
- Registro de la revalorización a un activo fijo.
- Baja del Activo fijo, por algún motivo fortuito o por su total depreciación.

- Generar reportes de los Activos Fijos para proporcionar a la toma de decisiones de los departamentos vinculados.

1.7.2. APORTES.

El principal aporte de este proyecto es que permita regularizar los procesos administrativos respecto al manejo de bienes y servicios en base a la normativa que existe.

A de más de:

- Automatización de los procedimientos manual de la unidad de activos fijos, lo que aumentara la velocidad en el procedimiento de la información y precisión en los resultados.
- Este sistema permitirá manejar mejor los inventarios permitiendo identificar de forma inmediata las características del bien, ubicación actual de uso, el responsable de su custodia, su depreciación a una determinada fecha y la emisión de reportes en forma oportuna.



CAPITULO II

MARCO TEORICO

2. MARCO TEÓRICO

2.1. INTRODUCCION

En el presente capítulo se revisa las definiciones empleadas de los métodos y herramientas que se utilizo para el desarrollo del proyecto.

2.2. METODOLOGÍA DE ML (MARCO LÓGICO)

El EML mantiene esencialmente un enfoque participativo en el que, de manera especial, se integran las personas y colectivos afectados por la futura intervención de desarrollo. Mientras los conceptos básicos del EML son relativamente simples, la calidad del producto depende directamente de la capacidad y experiencia de quienes están comprometidos con su aplicación.

2.2.1. LOS PASOS CLÁSICOS DEL EML.

Son los siguientes:

a) ANÁLISIS DE INVOLUCRADOS

- Identificar a todos los agentes sociales: grupos, instituciones, autoridades, etc. presentes en el área de intervención y que muestren problemas e intereses significativos.
- Investigar sus respectivos papeles, diferentes intereses, poder relativo y capacidad de participación.

b) ANÁLISIS DE PROBLEMAS

- Identificar los principales problemas existentes. Un problema es una situación negativa existente.
- De entre todos los problemas identificados, seleccionar en equipo el problema central para su análisis.
- Una vez consensuado el problema central, determinar cuáles son las causas directas que lo provocan, respondiendo a la pregunta ¿porqué? Así como sus efectos.
- Esquematizar esta relación causa-efecto a manera de árbol, colocando las causas por debajo del problema y los efectos por arriba.

c) ANÁLISIS DE OBJETIVOS

- Reformular todas las situaciones negativas en estados alcanzados positivos, que sean deseables y alcanzables.
- Verificar las principales relaciones medios/fines para asegurar la validez y la integridad de la jerarquía, respondiendo a la pregunta ¿cómo?
- Si es necesario, revisar el informe, añadir nuevos objetivos, borrar objetivos irreales o indeseables. Un problema que no puede ser convertido en objetivo se tendrán cuenta como amenaza.

d) ANÁLISIS DE ALTERNATIVAS

- Identificar aquellas estrategias consideradas como las más idóneas.
- Determinar los criterios que se consideren relevantes para evaluar las alternativas.
- Hacer una valoración, de preferencia cuantitativa para poder hacer una comparación.
- Seleccionar aquella alternativa que cumpla con los criterios establecidos de manera satisfactoria.

2.2.2. MATRIZ DE MARCO LÓGICO

¿Cómo se construye la Matriz de Marco Lógico?

- Traducir la información del árbol de objetivos a la matriz.
- Identificar posibles factores externos ó hipótesis.
- Construir los indicadores y sus fuentes de verificación.

2.3. UML (LENGUAJE UNIFICADO DE MODELADO)

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 preparado 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 modelado debe ser visual, facilitando la gestión de modelos, ayudando a mantener la consistencia entre los elementos del sistema y colaborando a mejorar la habilidad del equipo de desarrollo para manejar la complejidad del software.
- Debe existir un control de cambios de software para evitar un caos, especialmente en la comunicación entre los equipos de desarrollo, guardando la consistencia del desarrollo del sistema.

- Debe poderse evaluar continuamente la calidad de un sistema software con respecto a su funcionalidad, fiabilidad, rendimiento de la aplicación y del sistema.

2.3.1. VISTAS UML

Una vista se usa para representar proyecciones del sistema relacionados con aspectos particulares funcionales 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 separado de vistas (ver figura 2.1).

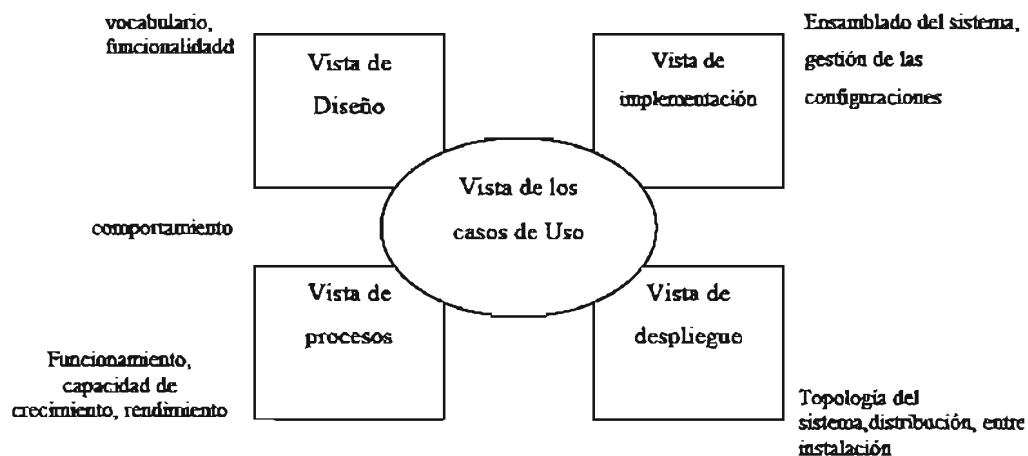


Fig. 2.1: Modelo de la arquitectura de un Sistema mediante vistas

Fuente: [ALARCON, 2000]

Los conjunto de Vista para modelar un sistema son:

a) 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.

b) 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 interacción de dichos componentes, utilizando diagramas de estado, secuencia, colaboración y actividad.

c) 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.

d) 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 dónde están los ejecutables del sistema y cómo se comunican entre sí.

e) VISTA DE IMPLEMENTACIÓN

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

2.4. RUP (PROCESO UNIFICADO DE RATIONAL)

Planteado por Kruchten (1996) para producir software de alta calidad, basado en componentes software interconectados a través de interfaces bien definidas.

Utiliza el UML (Lenguaje unificado de modelado), para preparar todos los esquemas de un sistema de software.

Es una metodología para desarrollo de software iterativo, centrado en arquitectura, dirigido por casos de uso, con proceso de Ingeniería de Software bien definido y bien estructurado, tiene roles claramente establecidos, con un ciclo de vida bien definido con hitos y puntos de decisión.

2.4.1. DIRIGIDO POR CASOS DE USO

Los casos de Uso, representan los requisitos funcionales del sistema, actuando como guía para la arquitectura del sistema, (que a su vez influye en la selección de los casos de uso), y como integrador de los flujos de trabajo (requisitos, análisis & diseño, implementación y pruebas, ver figura 2.2).



Fig. 2.2: Los casos de uso integran el Trabajo
Fuente [LET, 2006]

2.4.2. CENTRADO EN LA ARQUITECTURA

La arquitectura ayuda a los desarrolladores y clientes comprender el sistema, contribuyendo a organizar el desarrollo en subsistemas e interfaces bien definidas. En las fases iniciales lo que se hace es ir consolidando la arquitectura y se va modificando dependiendo de las necesidades del proyecto (ver figura 2.3).

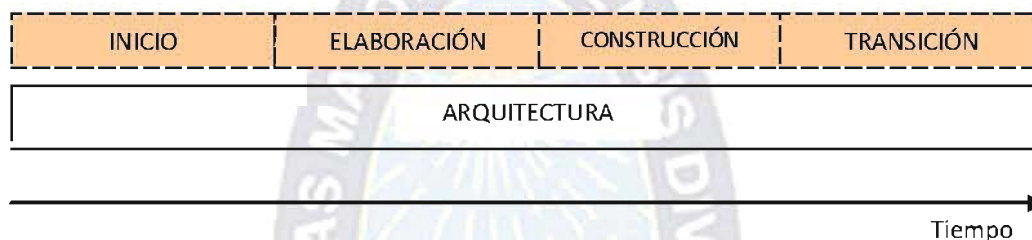


Fig. 2.3: Evolución de la Arquitectura del Sistema

Fuente: [LET, 2006]

La arquitectura de un producto Software se describe mediante vistas de diseño del sistema en construcción, incluyendo los aspectos estáticos y dinámicos más significativos del sistema.

El producto Software tiene tanto una función (casos de uso), como una forma (arquitectura), que evolucionan en paralelo, consolidando y/o modificando la arquitectura dependiendo de las necesidades del proyecto, hasta conseguir una arquitectura robusta en las fases finales del proyecto.

2.4.3. ITERATIVO E INCREMENTAL

Para el desarrollo de un producto software de gran esfuerzo, es práctico dividir el trabajo en partes más pequeñas o mini proyectos (ver figura 2.4).

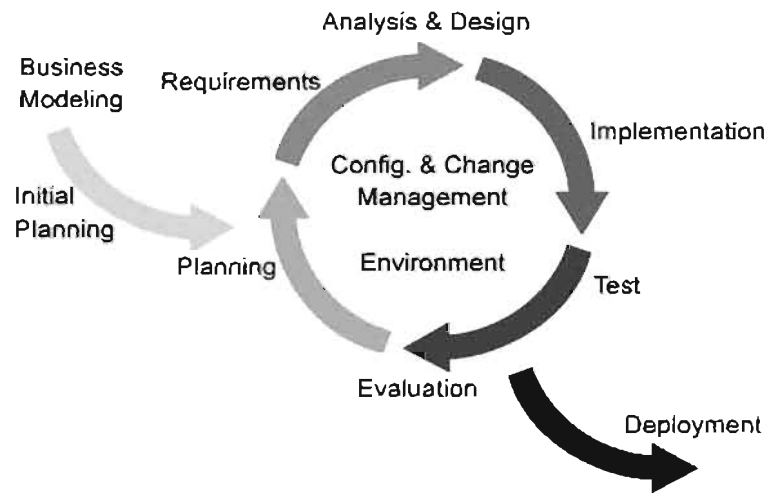


Fig. 2.4: Proceso Iterativo e Incremental

Fuente: [MARTINEZ, 2006]

Cada mini proyecto es una iteración, un recorrido más o menos completo a lo largo de todos los flujos de trabajo fundamentales, de la que se obtiene una versión interna, que resulta en un incremento al crecimiento del producto software.

2.4.4. VIDA DEL PROCESO UNIFICADO

Cada ciclo se divide en cuatro fases: y cada una de estas fases tiene que pasar por flujos de trabajo y estos son cinco, y a esto se le denomina iteración (ver figura 2.5).

El proceso Unificado se repite a lo largo de una serie de ciclos. Cada ciclo produce una nueva versión del sistema, y cada versión es un producto preparado para su entrega.

El producto terminado incluye los requisitos, casos de uso, especificaciones no funcionales y casos de prueba. Para llevar acabo el siguiente ciclo de manera eficiente, los desarrolladores necesitan todas las representaciones del producto software:

Un modelo de casos de uso, un modelo de análisis, un modelo de diseño, un modelo de implementación , un modelo de despliegue o de distribución, un modelo de prueba.

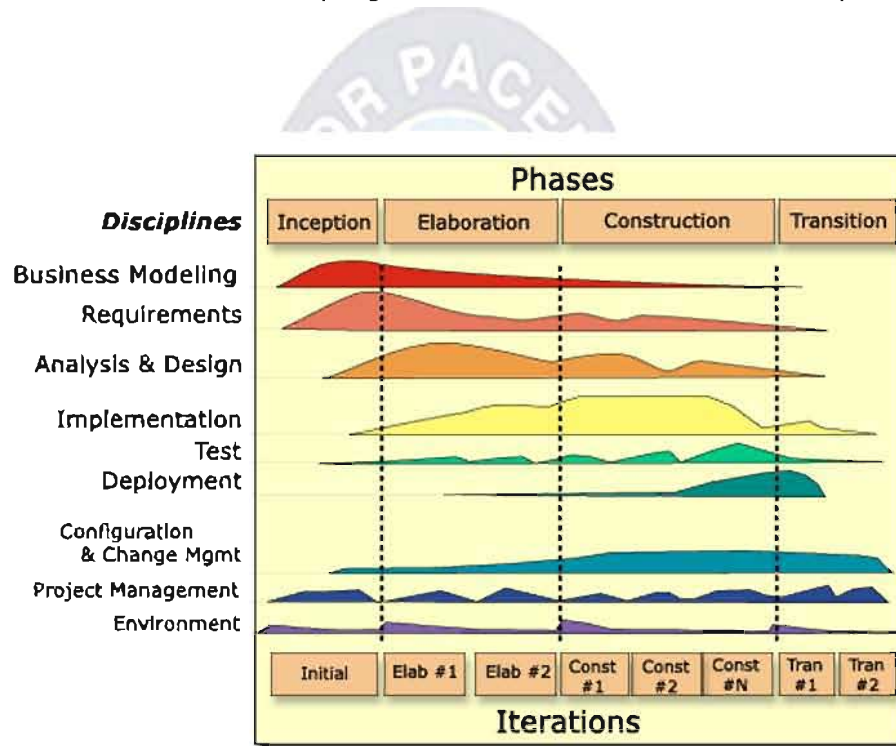


Fig. 2.5: RUP en dos dimensiones.

Fuente: [MARTINEZ, 2006]

FASES DE LA METODOLOGÍA RUP

Las fases de desarrollo de software son cuatro: Inicio, elaboración, construcción y transición. Cada fase se desarrolla mediante iteraciones genéricas. La iteración genérica se refiere a

que los cinco flujos de trabajo fundamentales se repite en cada iteración, precedidos por la planificación y seguidos por la evaluación.

En el plan de proyecto se deben planear las cuatro fases mediante la asignación del tiempo, hitos principales, planeación de iteraciones por fases.

Antes de iniciar un proyecto es necesario crear una lista de riesgos que se perciben en forma general y ordenarlos por su influencia en la planificación como en el desarrollo del proyecto para ser guiados por ellos y trabajar con ellos.

a) FASE DE INICIO

Para la planificación de la fase de inicio se debe empezar por llevar a cabo los siguientes pasos: reunir y organizar la información recogida antes de que el proyecto empiece, descubrir lo que le falta, para cuando se disponga de suficiente información presentar el plan de desarrollo de la primera iteración, también presentar los criterios de evaluación que indican cuando la iteración ha alcanzado sus objetivos, estos criterios se basan en: decidir el ámbito del sistema, resolver ambigüedades en los requisitos necesarios en esta fase, determinar una arquitectura candidata, mitigar los riesgos críticos.

El principal objetivo es de establecer el modelado del negocio, realizar un estudio de factibilidad del sistema propuesto para después decidimos si el sistema es viable o no, identificando todas las entidades externas con las cuales el producto va a interactuar, definiendo estas a una alto nivel, con la descripción de los casos de uso más significativos.

En la fase de inicio el desarrollo de una iteración transcurre a través de los flujos de trabajo fundamentales, debiendo avanzar sustancialmente en los componentes de: modelado del negocio y captura de requerimientos del sistema.

b) FASE DE ELABORACIÓN

El trabajo de una iteración durante la fase de elaboración discurre a lo largo de los flujos de trabajo fundamentales. En esta fase se utilizara los requerimientos funcionales para satisfacer las necesidades de los usuarios.

Esta fase ayuda a planificar con gran precisión la fase de construcción, concentrándose en el análisis y diseño del sistema hasta poder conseguir en sus últimas iteraciones algún prototipo que muestre el funcionamiento del sistema. Los objetivos principales de esta fase son: conseguir una arquitectura estable para guiar el sistema a lo largo de su vida futura, recopilar la mayor parte de los requisitos que aun queden pendientes, controlar la observación y control de los riesgos críticos y completar los detalles del plan del proyecto en el modelado del negocio si es que faltase completarse.

Al final de la fase de elaboración se empieza a planificar en forma más detallada la primera iteración de la fase de construcción y esbozar en términos más generales las iteraciones restantes.

c) FASE DE CONSTRUCCIÓN.

El trabajo de una iteración durante la fase de construcción discurre a lo largo de los flujos de trabajo. Aquí se detallan y analizan los requisitos restantes, pero la carga de trabajo en estos

dos flujos es relativamente leve o casi nula (análisis y requerimientos). La mayor parte del trabajo esta siendo realizado en los dos flujos siguientes (diseño e implementación).

El objetivo principal de esta fase es la capacidad de operación inicial. Es decir un producto listo para ser distribuido y ser sometido a pruebas. Esta fase requiere un mayor número de iteraciones que las fases anteriores, para cada iteración se selecciona algún Caso de Uso, se refina su análisis y diseño y se procede a su implementación y pruebas a diario, se empieza con la elaboración de material de apoyo al usuario. El hito que marca el fin de esta con la capacidad operacional parcial del producto que se haya considerado como crítica, lista para ser entregada a los usuarios para pruebas.

Los productos de esta fase son: El plan de proyecto para la fase de transición, el sistema software ejecutable, todos los artefactos incluyendo el modelo del sistema, la descripción de la arquitectura, el análisis y diseño del negocio que refleja la situación al final de la fase y una versión preliminar del manual de usuario.

d) FASE DE TRANSICIÓN.

El hito de esta fase es el lanzamiento del producto con la satisfacción de los usuarios. En este punto se establece si los objetivos fueron logrados.

El sistema ofrece la confianza suficiente como para operar en el entorno del usuario el cual debe cumplir los requisitos establecidos en las fases anteriores. El proyecto recibirá información para: determinar si el sistema hace lo que demanda su usuario y el negocio, descubrir riesgos inesperados, a notar problemas no resueltos, encontrar fallos, eliminar ambigüedades, centrarse en áreas en las que los usuarios muestren deficiencias y necesitan información o formación.

Comienza a menudo con la entrega de una versión beta del sistema mediante la distribución del producto software a una muestra representativa de la comunidad de usuario, la operación en paralelo con el sistema que se va a reemplazar y el entrenamiento de los usuarios y el personal de mantenimiento.

El producto de esta fase es el software, documentos legales, módulos del sistema, descripción completa y actualizada de la arquitectura, manuales y materiales de formación.

2.4.5. FLUJOS DE TRABAJO DEL DESARROLLO DE SOFTWARE

Los flujos de trabajo del desarrollo de software se presentan en las iteraciones que se desarrollan en las fases del Proceso Unificado y estas son: Modelado del Negocio, captura de requerimientos, análisis, diseño, implementación, pruebas y despliegue.

a) MODELADO DEL NEGOCIO

A este nivel de especificación lo que se pretende es comprender toda la actividad de la organización relacionada con el sistema a implementar, centrándose en “que” hace el sistema, en lugar de “como” lo hace. El modelado del negocio es una técnica que permite comprender los procesos de negocio de la organización, identificando y documentando la estructura y funcionamiento de la organización, mostrando la problemática de la organización como también el impacto que el sistema a desarrollar tendría sobre la organización.

Los pasos que se siguen en el modelado del negocio son:

- Identificar los roles implicados en los diferentes procesos del negocio, papeles que desempeñan los agentes o actores del sistema en cada proceso o caso de uso en que están involucrados.
- Identificar y delimitar los procesos de negocio y los procesos de mantenimiento existentes dentro de la organización, según objetivos de los mismos.
- Descripción textual del proceso de negocio.
- Especificar las informaciones que fluyen en cada diagrama y que serán de ayuda para construir el modelo conceptual.

Las tareas a realizar dentro de este flujo de trabajo son: modelo de casos de uso del negocio, descripción de los actores del negocio, descripción de los casos de uso del negocio y modelado del dominio.

- Modelo de casos de uso del negocio.

Representa la funcionalidad que ofrece el sistema en lo que se refiere a su interacción externa.

Un diagrama de casos de uso se emplea para visualizar el comportamiento del sistema, una parte de él o de una sola clase: y como se relaciona con su entorno, en este nivel no se distingue los procesos manuales de los automatizados.

- Descripción de los actores del negocio.

Representa terceros fuera del sistema desarrollado, identificando de esta manera el entorno externo del sistema, como ser: un proceso, una cosa que interactúe con el sistema, un subsistema, una clase o una persona externa.

- Descripción del modelo de casos de uso del negocio.

Es el desarrollo de un bosquejo inicial, explicando las actividades que realizan los actores, elaborando luego una descripción de lo que el sistema necesita hacer cuando interactúa con sus actores.

b) CAPTURA DE REQUERIMIENTOS DEL SISTEMA

En el proceso de desarrollo de un sistema, hay que identificar los requerimientos que el sistema debe cumplir, para satisfacer las necesidades de los usuarios finales, y lograr un acuerdo entre el equipo de desarrollo y el cliente.

Los requerimientos son todas aquellas funciones que deberá realizar el producto software, las mismas deben estar diferenciadas entre aquellas que son funciones evidentes para el usuario, de aquellas que son ocultas, se estructuran de forma natural mediante: diagramas de casos de uso de alto nivel, casos de uso expandidos.

Diagramas de casos de uso de alto nivel

El diagrama de casos de uso se representa como una caja rectangular con el nombre en su interior, los casos de uso están en el interior de la caja del sistema, y los actores fuera, y cada actor esta unida a los casos de uso en los que participa mediante una relación.

Actor.- es un rol que tiene un usuario o la labor que representa una persona, con respecto al sistema, un actor puede tener varios roles y un usuario concreto del sistema es una instancia de un actor.



Fig. 2.6: Representación Gráfica de un actor

Fuente: [LARMAN 1999]

Casos de uso.- fragmentos de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores, especifica una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores.



Fig. 2.7: Representación Gráfica de un Caso de Uso

Fuente: [LARMAN, 1999]

Relaciones entre casos de uso.-

- **Asociación**, es tipo de relación más básica que indica la invocación desde un actor o caso de uso a otra función (caso de uso), Dicha relación se denota con una flecha simple.

Asociación

- **Generalización**, cumple dos funciones, la de inclusión cuando un actor utiliza el caso de uso y la de herencia cuando un caso de uso es similar a otro en sus características.



Casos de uso expandidos

Describe los casos de uso con mayor detalle, enumerando paso a paso los eventos que se presenta durante una ocurrencia típica del caso de uso. El formato de la ficha de la descripción del caso de uso expandido es el siguiente:

Caso de Uso	(Nombre del Caso de uso)
Actores	(Usuarios involucrados con el caso de uso)
Propósito	(Función del caso de uso)
Resumen	(Narración breve de la funcionalidad del caso de uso y su interacción con los actores involucrados)
Tipo	(Tipo del caso de uso: Ej. : primario, secundario, abstracto)
Referencias Cruzadas	(Casos de uso o funciones del sistema, relacionados con el caso de uso en estudio)

Tabla 2.2: Descripción caso de uso expandido

Fuente: [AMO, 2006]

c) ANALISIS DEL SISTEMA

El análisis proporciona una visión general del sistema, el objetivo es conseguir una comprensión más precisa de los requisitos refinándolos y estructurándolos para lograr una descripción de los mismos, que sea fácil de mantener y que ayude a estructurar el sistema entero. El estudio del análisis es de mayor importancia durante la fase de elaboración, comprendiendo las siguientes tareas: definir diagramas de estado, diagramas de secuencia o diagramas de comunicación.

- Diagramas de Secuencias

Un diagrama de secuencia muestra una interacción ordenada de objetos según la secuencia temporal de eventos (ver figura 2.8).

- En la parte superior aparece los objetos que intervienen.
- La dimensión temporal se indica verticalmente (el tiempo transcurre hacia abajo).
- Las líneas verticales indican el período de vida de cada objeto.
- El paso de mensajes se indica con flechas horizontales u oblicuas (cuando existe demora entre el envío y la atención del mensaje).
- La realización de una acción se indica con rectángulos sobre las líneas de actividad del objeto que realiza la acción.
- Se puede colocar etiquetas (como restricciones de tiempo, descripciones de acciones, etc.) bien en el margen izquierdo o bien junto a las transiciones o actividades a las que se refieren.

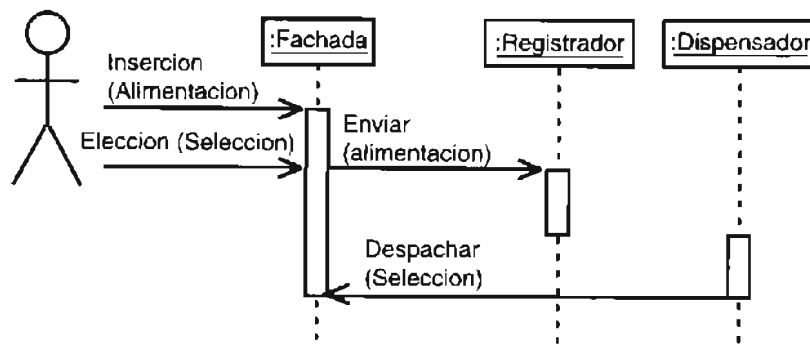


Fig. 2.8: Ejemplo: diagrama de Secuencia de un Caso de estudio

Fuente: [BERZAL, 2004]

- Diagramas de Estado

Los diagramas de estados son especialmente importantes para describir el comportamiento de un sistema reactivo (cuyo comportamiento está dirigido por eventos).

Los Diagramas de Estado muestran la secuencia de estados por los que pasa un caso de uso de un objeto a lo largo de su vida, o bien todo el sistema, en el diagrama de estado se indica que eventos hacen que se pase de un estado a otro y cuáles son las respuestas y acciones que genera (ver figura 2.9).

- un diagrama de estado es un grafo cuyos nodos son estados y cuyos arcos dirigidos son transiciones etiquetadas con los nombres de los eventos.
- Un estado se representa como una caja redondeada con el nombre del estado en su interior y las operaciones que realiza.
- Una transición se representa como una flecha desde el estado origen al estado destino.

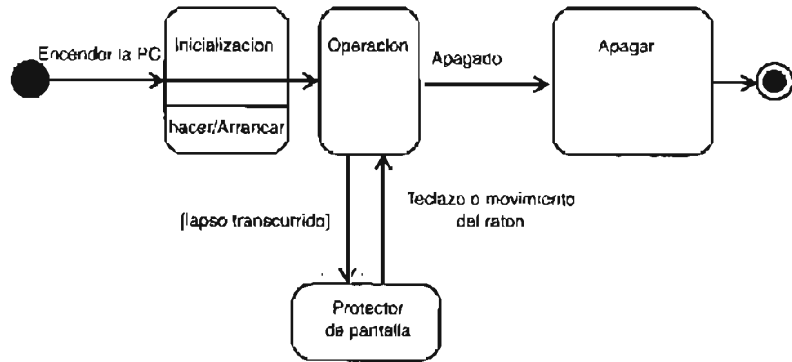


Fig. 2.9: Ejemplo: Diagrama de Estado de un Caso de Estudio

Fuente: [SCHMULLER, 2003]

d) DISEÑO DEL SISTEMA

El diseño del sistema es el centro de atención al final de la fase de elaboración y en el comienzo de la iteraciones de construcción. En este flujo de trabajo tomamos en cuenta las siguientes actividades: diseñar diagrama de clases. El diseño modela el sistema de tal forma que contenga todos los requisitos, de acuerdo al análisis realizado con anterioridad y de acuerdo a los requisitos establecidos por los usuarios.

- Diagramas de clase

Es el diagrama principal para el análisis y diseño, presentando las clases de sistema con sus relaciones estructurales y de herencia, incluyendo definición de atributos, especificando también operaciones, métodos, interfaces y dependencias.

Describe los tipos de objetos que hay en el sistema y las diversas clases de relaciones estáticas que existen entre ellos. En el diagrama de clases será donde definamos las características de cada una de las clases, interfaces, colaboraciones y relaciones de dependencia y generalización (ver figura 2.10).

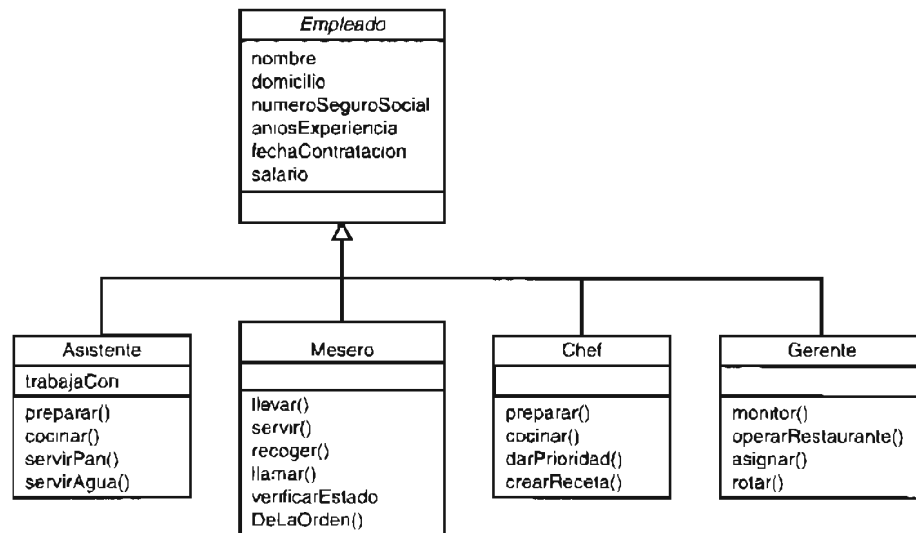


Fig. 2.10: Ejemplo: Diagrama de Clase de un Caso de Estudio

Fuente: [BERZAL, 2004]

e) IMPLEMENTACION DEL SISTEMA

En este flujo de trabajo diseñamos los diagramas de componentes e implementamos el código fuente del sistema. Se implementa el sistema en términos de componente, es decir, ficheros de código fuente, scripts, ficheros de código binario, ejecutables y similares. La implementación es la principal actividad durante las iteraciones de construcción.

- Diagrama de componentes

Un diagrama de componentes muestra la organización y las dependencias entre distintos componentes del sistema, el cual se representa como un conjunto de nodos y arcos entre estos (ver figura 2.11).

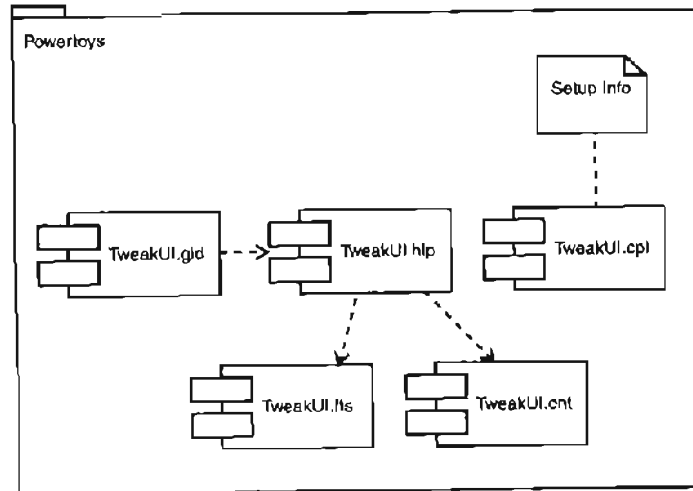


Fig. 2.11: Ejemplo: Diagrama de Componentes de un Caso de Estudio

Fuente: [BERZAL, 2004]

f) PRUEBAS DEL SISTEMA

Es importante identificar qué rangos de datos pueden alterar el comportamiento del programa y así definir zonas de trabajo. Es imprescindible pasar pruebas con al menos un dato de cada zona, tanto si el programa debe funcionar como si debe dar un mensaje de error. También es importante la validación de datos ingresados ya que podrían ocurrir desbordes y producir terribles errores del sistema.

Los datos que se utilizan para las pruebas, deben ser los más usuales por los clientes y si es posible se recomienda realizar con datos reales para ejercitar las pruebas, dichos datos son monocordes, es decir: si va con uno, va con todos.

g) DESPLIEGUE DEL SISTEMA

El despliegue del sistema especifica el modelado de un sistema cliente / servidor en un diagrama de Despliegue.

- Diagrama de Despliegue

El Diagrama de Despliegue representa los nodos de procesamiento y componentes de la configuración del sistema en tiempo de ejecución. Se debe tomar algunas decisiones sobre dónde colocar físicamente sus componentes software, qué cantidad de software debe residir en el cliente, etc. (ver figura 2.12). Para modelar un sistema cliente / servidor se deben realizar las siguientes tareas:

- Destacar los dispositivos relacionados con el comportamiento del sistema.
- Proporcionar señales visuales para esos procesadores y dispositivos a través de estereotipos.
- Identificar los nodos que representan los procesadores cliente y servidor.

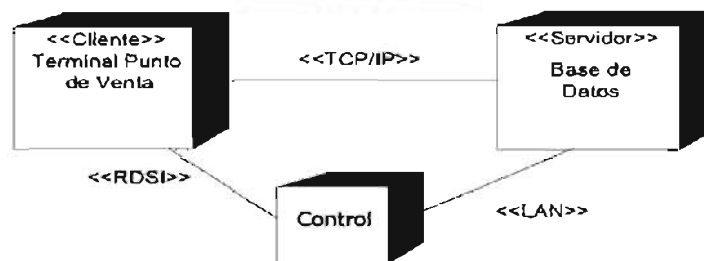


Figura 2.12: Ejemplo: Diagrama de Despliegue de un Caso de Estudio

Fuente: [BERZAL, 2004]

2.5. MÉTRICAS DE CALIDAD

La calidad de software es: la concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente.

El estándar ISO 9126 ha sido desarrollado en un intento de identificar los atributos clave de calidad para el software. El estándar identifica seis atributos de calidad: funcionalidad, usabilidad, facilidad de mantenimiento, portabilidad, confiabilidad y eficiencia.

2.5.1. FUNCIONALIDAD

Se trata de identificar la capacidad que el producto software tiene para proporcionar funcionalidades que satisfagan las necesidades especificadas.

Como no puede ser medida directamente, debe ser medido indirectamente de otras medidas directas como ser el punto función propuesto por Albretch.

Para el cálculo de esta métrica se determina los valores de las cinco características de dominio de información: número de entradas, salidas, peticiones de usuarios, archivos e interfaces de usuario. Asociando a estos dominios un valor de complejidad y calculando el Punto función con la siguiente relación:

$$PF = \text{Cuenta_total} * [0.65 + 0.01 * \sum Fi]$$

Cuenta_total : suma de todas las entradas.

0.65 : valor de ajuste de complejidad mínimo.

0.01 : factor de conversión con un error del 1 %.

2.5.2. CONFIABILIDAD

Es lo que se puede esperar que un programa lleve a cabo su función con exactitud requerida. Estadísticamente como la probabilidad de operación libre de fallos de un programa de computadora en un entorno determinado y un tiempo específico.

Para el cálculo de la confiabilidad del sistema se utiliza la siguiente ecuación:

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

F(t) : probabilidad de ejecución con fallas.

t : representa el periodo de tiempo en que se pone a prueba el sistema

1 - F(t) : probabilidad de ejecución sin fallas

Para hallar la probabilidad de falla del sistema F(t) en el periodo t, se utiliza la función exponencial dada por:

$$F(t) = f * e^{(-\lambda * t)}$$

f : funcionalidad del sistema.

λ : probabilidad de fallo en un periodo de tiempo.

t : tiempo total en el que se hace el cálculo de fallo.

2.5.3. MANTENIMIENTO.

La facilidad con que una modificación puede ser realizada. Esta indicada por los siguientes sub atributos: facilidad de análisis, facilidad de cambio establecido y facilidad de prueba.

Este factor de calidad viene dado según el estándar IEEE 982-1998 por la métrica del índice de madurez del software (IMS) que proporciona una indicación de la estabilidad del producto software (basada en cambio que ocurren en cada versión del producto). Se determina la siguiente fórmula:

$$IMS = (Mt - (Fa + Fc + Fd) / Mt)$$

Mt : número de módulos en la versión actual.

Fa : número de módulos en la versión actual que se han añadido.

Fc : número de módulos en la versión actual que se han cambiado.

Fd : número de módulos de la versión anterior, que se han borrado en la versión actual.

A medida que el IMS se aproxima a 1, el producto se empieza a estabilizar.

2.5.4. USABILIDAD

La usabilidad se determina calculando el porcentaje de una encuesta elaborada a un cierto número de usuarios finales.

Es el grado en el que el software es fácil de usar, se mide a través del esfuerzo necesario para aprender a utilizar el sistema, ya sea según su interfaz o manual de usuario.

2.5.5. PORTABILIDAD

La facilidad con el que el software puede ser llevado de un entorno a otro, o cambiar a un ambiente determinado. El grado de portabilidad del sistema esta dado por la siguiente ecuación:

$$GP = 1 - (\text{Costo de transportar} / \text{Costo de re-desarrollar})$$

GP: Grado de Portabilidad.

GP > 0, la portabilidad del sistema es más rentable que el re-desarrollo.

GP = 1, la portabilidad es perfecta.

GP < 1, re-desarrollar el sistema es más rentable que transportarlo.

2.6. ACTIVOS FIJOS.

Se considera Activo Fijo a aquellos bienes, que tienen una vida relativamente larga, físicamente tangible, no se venden pero pueden ser utilizados en la producción o

comercialización de bienes y servicios, para ser alquilado a terceros, o para fines administrativos y prestar servicio a la institución en el desarrollo específico de sus actividades.

Con el objetivo de lograr la racionalidad en la distribución, uso y salvaguarda de los activos fijos de la institución, la Sección de activos fijos trabaja en base a las normas básicas del Sistema de Administración de Bienes y Servicios, que constituyen un conjunto de elementos jurídicos, técnicos y administrativos que regulan el manejo de bienes de propiedad de la institución.

2.6.1. CATEGORÍAS IMPORTANTES DE LOS ACTIVOS FIJOS

Los renglones de planta y equipos se clasifican, con frecuencias, en los siguientes grupos:

a) ACTIVOS TANGIBLES

El término tangible denota sustancia física como es el caso de un terreno, un edificio o una máquina. Esta categoría se puede subdividir en dos clasificaciones claramente diferenciales:

- Propiedad de planta sujeta a depreciación. Se incluyen los activos fijos de vida útil limitada, tales como los edificios y equipos de oficinas.
- Terrenos. El único activo fijo que no está sujeto a depreciación es la tierra, que tiene un término limitado de existencia.

b) ACTIVOS INTANGIBLES

El termino activos Intangibles se usa para describir activos que se utilizan en la operación del negocio pero que no tiene sustancia física y no son corrientes. Como ejemplo están las patentes, los derechos del autor, las marcas registradas, las franquicias.

Los activos corrientes tales como las cuentas por cobrar o el arrendamiento prepagado, no están incluidos en la clasificación de intangibles, aunque carezcan de sustancia física.

c) RECURSOS NATURALES

Un lugar adquirido con el fin de extraer o remover parte de un recurso valioso tal como el petróleo, minerales o madera se clasifican como un recurso natural, no como terreno. Este tipo de activo se convierte gradualmente en inventario a medida que el recurso natural se extrae del lugar.

2.6.2. CLASIFICACIÓN GENERAL

Los activos fijos de una empresa se clasifican en dos grandes grupos principales: Bienes inmuebles y bienes muebles.

a) BIENES INMUEBLES.

Los Activos Fijos de una empresa se clasifican como bienes inmuebles cuando no son trasladables, es decir, que no se pueden mover de un lugar a otro.

b) BIENES MUEBLES.

Los Activos fijos de una empresa se clasifican como bienes muebles cuando son trasladables, es decir, que se pueden mover de un sitio a otro. La inmensa mayoría de los activos fijos de una empresa son bienes muebles, esto es, que se pueden trasladar o mover de un lugar a otro.

2.6.3. CLASIFICACIÓN CONTABLE

Para efectos contables los Activos Fijos, tanto muebles como inmuebles, se clasifican particularmente en dos grupos: activos no depreciables y activos depreciables.

a) ACTIVOS NO DEPRECIABLES.

Los activos no depreciables son aquellos que no sufren desgaste o demérito por el uso a que son sometidos y que por tanto no pierde un precio, al menos contablemente. Entre los activos no depreciables tenemos los terrenos, los cuales no sufren desgaste por el uso a que son sometidos, ni por el transcurso del tiempo, y por esta razón se consideran no depreciables. Esta es una teoría contable aceptada en todo el mundo.

b) ACTIVOS DEPRECIABLES.

La inmensa mayoría de los Activos Fijos de una empresa son depreciables, y son los que sufren desgaste o deterioro por el uso a que son sometidos o por el simple transcurso del tiempo.

2.6.4. CLASIFICACIÓN ESPECIAL

Existen algunos activos fijos que exigen alguna clasificación especial para su tratamiento contable. Uno de ellos, son los edificios.

a) EDIFICIOS

Los edificios físicamente conforman una sola unidad con el terreno sobre el cual están contruidos. Sin embargo para efectos contables es necesario separar uno del otro, al menos teóricamente, en cuentas separadas.

2.6.5. DEPRECIACIÓN DE ACTIVOS FIJOS

Con excepción de los terrenos, la mayoría de los activos fijos tienen una vida útil limitada ya sea por el desgaste resultante del uso, el deterioro físico causado por terremotos, incendios y otros siniestros, la pérdida de utilidad comparativa respecto de nuevos equipos y procesos o el agotamiento de su contenido. La disminución de su valor, causada por los factores antes mencionados, se carga a un gasto llamado depreciación.

La depreciación es un reconocimiento racional y sistemático del costo de los bienes, distribuido durante su vida útil estimada, con el fin de obtener los recursos necesarios para la reposición de los bienes, de manera que se conserve la capacidad operativa o productiva del ente público. Su distribución debe hacerse empleando los criterios de tiempo y productividad, mediante uno de los siguientes métodos: línea recta, suma de los dígitos de los años, saldos

decrecientes, número de unidades producidas o número de horas de funcionamiento, o cualquier otro de reconocido valor técnico (ver Anexo B), que debe revelarse en las notas a los estados contables.

La depreciación no es un proceso de valuación por el que se asigna a gastos el costo del activo de acuerdo con autovalúos realizados al fin de cada período. La depreciación es una asignación del costo del activo a gastos de acuerdo con su costo original. Un activo totalmente depreciado solamente significa que ha alcanzado el final de su vida útil estimada, es decir, que no registra más depreciación para el activo. Esto no quiere decir que el activo sea desechado o que ya no se use; la mayoría de veces, las empresas continúan utilizando los activos totalmente depreciados. La depreciación no significa que el negocio aparte efectivo para reemplazar los activos cuando lleguen a ser totalmente depreciados. La depreciación es simplemente parte del costo del activo que es enviado a gastos y no significa efectivo. La depreciación no implica un movimiento de efectivo pero sí afecta el efectivo de un negocio en el sentido de que constituye un gasto deducible para fines impositivos. Por lo tanto, la depreciación afecta el nivel de utilidades y el pago de impuestos. A un mayor nivel de depreciación, las utilidades son menores, y los impuestos correspondientes, también son menores.

Por otro lado, debe considerarse el valor residual final ó valor recuperable que será el que tendrá el bien cuando se discontinúe su empleo y se calcula deduciendo del precio de venta los gastos necesarios para su venta, incluyendo los costos de desinstalación y desmantelamiento, si estos fueran necesarios.

Importe original	+	Revalúos efectuados	-	Valor recuperable	=	Importe a depreciar
-------------------------	----------	----------------------------	----------	--------------------------	----------	----------------------------

2.6.6. REVALORIZACIÓN DE ACTIVOS FIJOS

Como consecuencia del proceso inflacionario, en algunos países de América Latina, las leyes permiten la revalorización de activos, unas veces, voluntaria y otras obligatoriamente bajo diferentes condiciones en cada país.

Por el método de revalorización de los Activos, los estados financieros expresados en moneda al costo histórico, son restituidos de conformidad a índices determinados por el gobierno. La medida más común usada se relaciona con los deflatores económicos o con la fijación de porcentajes específicos según la época de adquisición del activo a ser revaluado. Estas revalorizaciones abarcan especialmente a los bienes del Activo Fijo, pero en algunos casos, pueden referirse a todos los activos no monetarios.

El resultado del incremento afecta positivamente a alguna de las cuentas del patrimonio, que en la mayoría de los casos no puede ser distribuida como utilidad, y está sujeto a un régimen impositivo especial.

Sobre el valor actualizado, se aplicarán las tasas de depreciación correspondientes a los años de uso, para determinar el nuevo valor de activos netos. La diferencia favorable es abonada a una de las cuentas del patrimonio que se determina por Ley, o en su caso, a una cuenta de Beneficios o Ajustes por Revalorización que formará parte de las cuentas patrimoniales.

Para determinar los incrementos por el sistema porcentual, bastará dividir el índice del año actual por el índice de cada uno de los años y luego aplicar los porcentajes correspondientes.

2.7. ESTIMACIÓN DE COSTOS EN PROYECTOS SOFTWARE, MODELO COCOMO

Una metodología que se encarga de medir proyectos software es COCOMO. La metodología COCOMO (COConstructive COSt MOdel) se debe a Barry Boehm, y está orientada a líneas de código.

Hay una jerarquía de modelos COCOMO: básico, intermedio y avanzado, la cual se aplica a tres tipos diferentes de software:

1. **Orgánico:** proyectos relativamente sencillos, menores de 50.000 líneas de código. Se tiene experiencia en proyectos similares y se encuentra en un entorno estable.
2. **Semiacoplado:** proyectos intermedios en complejidad y tamaño. La experiencia en este tipo de proyectos es variable, y las restricciones intermedias.
3. **Empotrado:** proyectos bastante complejos, en los que apenas se tiene experiencia y en un entorno de gran innovación técnica. Se trabaja con unos requisitos muy restrictivos y de gran volatilidad.

Dado que sólo se va a emplear una variable para la estimación (la línea de código), se empleará COCOMO básico, ya que es un modelo univariable estático, con lo que se obtiene una **valoración objetiva** del esfuerzo realizado. Este proyecto será considerado como software orgánico, ya que posee menos de 50.000 líneas de código. La ecuación del esfuerzo de COCOMO básico tiene la siguiente forma:

$$E = \text{Esfuerzo} = a KLDC^b \text{ (persona x mes)}$$

donde *KLDC* es el número de líneas de código, distribuidas en millares, para el proyecto. La ecuación del tiempo de desarrollo es:

$$T = \text{Tiempo de duración del desarrollo} = c \text{ Esfuerzo}^d \text{ (meses)}$$

Por su parte los coeficientes *a*, *b*, *c* y *d* se obtienen empíricamente del estudio de una serie de proyectos, y sus valores son:

Proyecto de software	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>
Orgánico	2,4	1,05	2,5	0,38
Semiacoplado	3,0	1,12	2,5	0,35
Empotrado	3,6	1,20	2,5	0,32

Tabla 2.3: Coeficientes Cocomo

Fuente: [DANIELE, 2007]

2.8. RECURSOS TECNOLÓGICOS

2.8.1. PREPROCESADOR DE HIPERTEXTO PHP

PHP es un lenguaje de secuencia de comandos de servidor diseñado específicamente para la web. Dentro de una página web puede incrustar código PHP que se ejecutará cada vez que se visite una página. El código PHP es interpretado en el servidor Web y genera código HTML y otro contenido que el visitante verá.

PHP fue concebido en 1994 por Rasmus Lerdorf, inicialmente significaba Personal Home Page (Página de inicio personal) pero se modificó de acuerdo a la convención de designación del GNU.

Una de las ventajas de PHP es que está disponible para Microsoft Windows, para muchas versiones de Unix y para cualquier servidor Web completamente funcional. MySQL resulta igualmente versátil, dos configuraciones muy utilizadas son:

- Linux con el servidor Web Apache.
- Microsoft Windows XP con Microsoft Internet Information Server (IIS).

Sea cual sea el hardware, sistema operativo y servidor Web es recomendable el uso de PHP y MySQL por las siguientes cualidades que presenta con respecto a los actuales competidores:

a) ALTO RENDIMIENTO.

PHP es muy eficaz. Mediante el uso de un único servidor, puede servir millones de acceso al día, superando a sus competidores ampliamente en esta faceta.

b) INTERFACES PARA DIFERENTES SISTEMAS DE BASE DE DATOS.

PHP dispone de una conexión propia a todos los sistemas de base de datos. Además a través de MySQL puede conectarse directamente a las bases de datos de PostgreSQL, MSQL, Oracle, dbm, FilePro, Hyperwave, Informix, InterBase y Sybase entre otras, PHP5 también cuenta con una interfaz SQL incorporada a un archivo plano, denominada SQLite.

c) BIBLIOTECAS INCORPORADAS PARA MUCHAS TAREAS WEB HABITUALES.

PHP incorpora una gran cantidad de funciones integradas para realizar útiles tareas relacionadas con la Web.

d) BAJO COSTE.

PHP es gratuito. Puede descargar la última versión de <http://www.php.net> sin coste.

e) FACILIDAD DE APRENDIZAJE Y USO.

La sintaxis de PHP se basa en otros lenguajes de programación, principalmente en C y perl.

f) PORTABILIDAD.

PHP está disponible para una gran cantidad de sistemas operativos diferentes, puede escribir código PHP en todos los sistemas operativos gratuitos del tipo Unix, como Linux y FreeBSD, versiones comerciales de Unix, como Solaris e IRIX o en las diferentes versiones de Microsoft Windows. Su código funcionará sin necesidad de aplicar ninguna modificación a los diferentes sistemas que ejecute PHP.

g) DISPONIBILIDAD DE CÓDIGO ABIERTO.

Dispone de acceso al código fuente de PHP. A diferencia de los productos comerciales y de código cerrado, si desea modificar algo o agregar un elemento al programa, puede hacerlo con total libertad.

h) COMPATIBILIDAD CON EL ENFOQUE ORIENTADO A OBJETOS.

Desde la versión 5 de PHP, se cuenta con completas funciones orientadas a objetos. Como por ejemplo herencia, atributos y métodos privados y protegidos, clases y métodos abstractos, interfaces, constructores y destructores, como también un comportamiento de iteraciones incorporado.

2.8.2. LENGUAJE DE CONSULTAS MYSQL

MySQL es un sistema para la administración de bases de datos relacionales (RDBMS) rápido y sólido. Las bases de datos permiten almacenar, buscar, ordenar y recuperar datos de forma eficiente. El servidor de MySQL, controla el acceso a los datos para garantizar el

uso simultáneo de varios usuarios, para proporcionar acceso a dichos datos y para asegurarse de que sólo obtienen acceso a ellos los usuarios con autorización. Por lo tanto, MySQL es un servidor multiusuario y de subprocesamiento múltiple. MySQL lleva disponible desde 1996 pero su nacimiento se remonta a 1979.

MySQL se distribuye bajo un sistema de licencias dual. Puede utilizarlo bajo una licencia de código abierto (GPL), que es gratuita mientras cumpla las condiciones de la misma.

Entre los competidores principales de MySQL, se puede citar a PostgreSQL, Microsoft SQL Server y Oracle. MySQL cuenta con muchas ventajas, entre las cuales se encuentran:

a) RENDIMIENTO

MySQL es muy rápido. Existe una diferencia de velocidad abismal con respecto a los productos de la competencia según indicadores comparativos en el sitio web <http://web.mysql.com>

b) BAJO COSTE

MySQL está disponible de manera gratuita, bajo una licencia de código abierto, o por un precio reducido en forma de licencia comercial.

c) FACILIDAD DE USO

Si ha utilizado otros RDBMS, no hay problemas para adaptarse a este sistema. MySQL resulta además más sencillo de configurar que otros productos similares.

c) PORTABILIDAD

MySQL se puede utilizar en una gran cantidad de sistemas Unix diferentes así como bajo Microsoft Windows.

d) CÓDIGO FUENTE

Se puede obtener y modificar el código fuente de MySQL, garantizando la continuidad en el futuro.

En MySQL 5.x podemos apreciar importantes cambios como los procedimientos almacenados y la compatibilidad con cursores.





CAPITULO III

MARCO APLICATIVO

3. MARCO APLICATIVO

3.1. INTRODUCCIÓN

En este capítulo se define el marco de trabajo y las tareas que se requieren para desarrollar, construir e implementar el Sistema, se tiene como finalidad poner en práctica todo lo mencionado en los capítulos anteriores.

Para diseñar el ciclo de vida de desarrollo del sistema se empleara el RUP (Proceso Unificado de Rational), que implementa el desarrollo iterativo e incremental como estrategia o paradigma de Ingeniería de Software, buscando obtener un producto de calidad minimizando los riesgos y evolucionando el producto en cada una de las fases y flujos de trabajo sobre el cual itera.

En los siguientes puntos ampliaremos cada una de las fases de la metodología RUP, desarrollando todos los flujos de trabajo que se iteran sobre estos.

3.2. FASE DE INICIO

En esta fase nos centraremos en el estudio de factibilidad del sistema y los flujos de trabajo: modelado del negocio y captura de requerimientos del sistema.

3.2.1. PLANIFICACIÓN DE LA FASE DE INICIO

En la tabla 3.1 se presenta la planificación para la fase de Inicio del proyecto detallando la fecha de inicio, fin y aprobación de cada actividad generada.

Actividad generada durante la Fase de Inicio	Semana	Inicio	Fin	Aprobación
Modelado del Negocio				
Modelo de Casos de Uso del Negocio y Descripción de los actores del Negocio	2 - 5	14-09-10	09-10-10	09-10-10
Requerimientos				
Modelo de Casos de Uso	2 - 5	14-09-10	09-10-10	09-10-10
Especificación de Casos de Uso	4 - 5	28-09-10	09-10-10	09-10-10
Especificaciones Adicionales	4 - 5	28-09-10	09-10-10	09-10-10

Tabla. 3.1: Plan de Actividades de la Fase de Inicio.

Fuente: [Elaboración Propia]

3.2.2. MODELO DEL NEGOCIO DEL SISTEMA ACTUAL

Para el modelado del negocio, se han identificado

a) DESCRIPCIÓN DE LOS ACTORES.

Jefe de la Unidad Administrativa y Financiera.- Representa a la parte ejecutiva de la gestión de activos fijos dentro la institución, toma de decisiones en adquisiciones, transferencias, bajas y revalorizaciones.

Jefe del Área de Contabilidad.- Parte operativa de la Unidad Administrativa y financiera, encargada del cálculo y la revisión de las depreciaciones de los activos fijos.

Encargado de Almacenes.- Parte operativa del manejo de bienes y servicios, encargado del salvaguarda de activos fijos no asignados y dados de baja.

Encargado de Activos fijos.- Parte operativa del manejo y administración de activos fijos, responsable de la clasificación, registro, codificación, asignación, transferencias, revalorización, salvaguarda y generación de reportes de activos fijos.

Responsables.- Este actor representa a los funcionarios que trabajan dentro la empresa, los cuales tiene activos fijos a su responsabilidad.

Usuarios.- Este actor representa a los clientes que rentan un activo fijo específicamente automotores y tienen bajo su responsabilidad este activo.

b) MODELADO DEL NEGOCIO DEL SISTEMA ACTUAL: GESTIÓN DE ACTIVOS FIJOS.

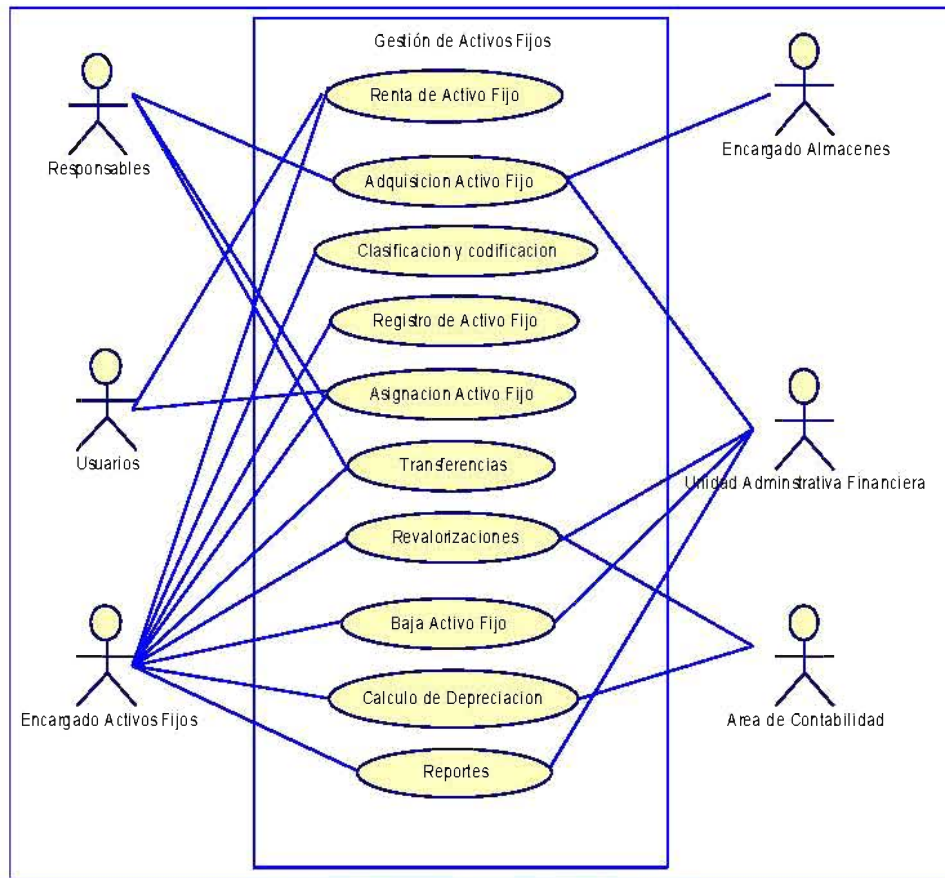


Fig. 3.2: Casos de Uso del Modelo del Negocio del Sistema Actual: Gestión de Activos Fijos.

Fuente: [Elaboración Propia]

c) DESCRIPCIÓN DE LOS CASOS DE USO DEL MODELADO DEL NEGOCIO DEL SISTEMA ACTUAL

Renta de Activos Fijos.- Este proceso consiste en la renta de un activo fijo a un cliente.

Adquisición de Activos Fijos.- Este proceso consiste en la adquisición de un activo fijo, tras la solicitud de un usuario a la Unidad Administrativa financiera, que da curso a la

adquisición del bien, para luego depositarlo en Almacenes hasta que el encargado de Activos Fijos lo registre, clasifique y asigne al usuario solicitante.

Clasificación y Codificación de Activos Fijos.- El encargado de Activos Fijos es el responsable de clasificar el activo según normas y reglamentos de las NB-SABS, y codificarlo según la clasificación realizada.

Registro de Activos Fijos.- El encargado de activos fijos registra los datos del Activo Fijo manualmente.

Asignación de Activos Fijos.- Se registra los datos de asignación manualmente, se transcribe un acta de entrega y recepción de activos fijos, se realiza la entrega física de los activos asignados ya sea al Responsable o al Cliente y se archiva la copia del acta.

Transferencias.- Este proceso consiste en la transferencia de un activo de un usuario a otro, por parte de el encargado de activos fijos, se verifica el registro del activo en el acta de entrega y recepción, para luego transcribir un acta de transferencia al nuevo responsable del activo, se le entrega el activo físicamente.

Revalorizaciones.- Amparado bajo una resolución este proceso consiste en dar un valor agregado a un activo fijo conteniendo algún motivo, gestionada por la Unidad Administrativa y financiera; el valor nuevo es calculado por el área de contabilidad, para que luego el encargado de activo fijos actualice su respectivo registro.

Baja de Activos Fijos.- Tras la depreciación total del activo fijo o por algún motivo, amparado bajo resolución generalmente se realiza la baja, gestionada por la Unidad

Administrativa y financiera, para luego pasar a su destrucción física, si amerita el caso, posterior registro.

Cálculo de Depreciaciones.- El área de contabilidad realiza el cálculo de las depreciaciones, para luego transcribir un reporte de depreciaciones.

Reportes.- Este proceso consiste en transcribir y dar un informe según se requiera de los activos con los que cuenta la empresa.

3.2.3. REQUERIMIENTOS DEL SISTEMA

Para la captura de requerimientos de aquellas funciones que deberá realizar el producto software, mismas que deben estar diferenciadas en categorías entre aquellas que son funciones evidentes para el usuario, de aquellas que son ocultas. Hay que identificar todos los requerimientos que el sistema debe cumplir, con el objetivo de satisfacer las necesidades de los usuarios finales.

Estos requerimientos son:

Ref #	Función	Categoría
R 1	Registrar la asignación de un activo fijo a un responsable en una determinada unidad.	Evidente
R 2	Registrar la liberación de un activos fijo de un responsable.	Evidente
R 3	Generar acta de Entrega y Recepción de activos fijos para un determinado responsable	Evidente
R 4	Registrar la transferencia de responsabilidad de un activo de un usuario a otro.	Evidente
R 5	Generar acta de Entrega y Recepción de activos fijos para un determinado cliente	Evidente
R 6	Registrar la asignación de un activo fijo a un cliente por motivo de Renta.	Evidente
R 7	Registrar la aceptación de transferencia de activos fijos.	Evidente

R 8	Generar acta de Transferencia con Vo. Bo. Encargado de activos fijos.	Evidente
R 9	Registrar el Informe técnico de mantenimiento.	Evidente
R 10	Registrar login, fecha y hora para cada proceso de gestión y transacciones que se registré para cada activo.	Oculto
R 11	Generar reporte de historial de transacciones.	Evidente
R 12	Registrar el inventariado de activos fijos que se realiza a un responsable determinado.	Evidente
R 13	Generar reporte de Inventario de activos Fijos de un determinado responsable.	Evidente
R 14	Generar reporte de Usuarios	Evidente
R 15	Generar reporte de Unidades y/o Oficinas.	Evidente
R 16	Generar reporte de Responsables	Evidente
R 17	Generar reporte de activos Fijos.	Evidente
R 18	Generar reporte de Incorporación por compra de activos fijos.	Evidente
R 19	Generar reporte de Revalúo de activos fijos.	Evidente
R 20	Generar reporte de activos fijos, por grupo y/o auxiliar contable.	Evidente
R 21	Generar reporte de activos fijos, por Unidad y/o responsable.	Evidente
R 22	Generar reporte de transferencias	Evidente
R 23	Generar reporte de activos dados de baja	Evidente
R 24	Generar reporte de historial de mantenimiento de un activo	Evidente
R 25	Generar reporte de activos no asignados.	Evidente
R 26	Registrar y codificar el nuevo activo fijo, que es incorporado a la institución, se incluye el grupo y auxiliar contable, fuente y organismo financiador.	Evidente
R 27	Registro de la Baja del activo fijo.	Evidente
R 28	Registro de la Revalorización del activo fijo.	Evidente
R 29	Reduce cantidades del inventario cuando se da de baja un activo.	Oculto
R 30	Mostrar los días consumidos, factor actual, valor actual, depreciación de la gestión, depreciación acumulada, valor neto del activo registrado.	Evidente
R 31	Generar código tentativo de un activo fijo.	Oculto
R 32	El usuario debe introducir su login y password para utilizar el sistema.	Evidente
R 33	Registrar y modificar los datos personales de los usuarios del sistema.	Evidente
R 34	Dar de bajo a usuarios del sistema.	Evidente
R 35	Registrar el password encriptado, de un determinado usuario.	Oculto

Tabla. 3.6: Funciones de Requerimientos del Sistema de Activos Fijos.

Fuente: [Elaboración Propia]

3.2.4. CASOS DE USO SISTEMA SCSAF

Describiremos las interacciones del sistema con su entorno, identificando los actores, que representan las actividades desempeñadas por los usuarios del sistema y los casos de uso corresponden a la funcionalidad que el sistema ofrece a los usuarios.

3.2.4.1. IDENTIFICACIÓN DE ACTORES.-

- **Encargado de Activos Fijos:** Realiza la gestión, control, seguimiento y salvaguarda de activos fijos (revalúo, bajas, liberación, asignación a Responsables, asignación a Usuarios, transferencia, aceptar transferencia), con la generación de sus respectivas actas, revisa, modifica y registra a los nuevos activos que se incorporan a la institución, registra y modifica parámetros para generar reportes, registra y genera reporte de inventarios.
- **Responsable:** Este actor tiene el rol de generar reportes de los activos del cual es responsable, transferir los activos a su cargo, aceptar transferencia de los activos que se le transfiere, generando la respectiva acta de conformidad, y registrar el mantenimiento de los activos del cual responde, generando el informe técnico del mantenimiento.
- **Usuario:** Este actor que es un cliente que ha rentado un activo (automotor) puede generar reportes a detalle de los activos que tiene a su cargo.

Actor	Proceso
Encargado de Activos Fijos	<ul style="list-style-type: none"> - Asignación y Liberación de responsables. - Asignación y Liberación de Usuarios. - Transferencia. - Aceptar Transferencias. - Revalúo. - Baja de activos fijos - Generación de actas. - Registro de activos fijos. - Registro y modificación de parámetros. - Registro y generación de reporte de inventarios.
Responsable	<ul style="list-style-type: none"> - Transferir. - Aceptar Transferencia. - Registrar y generar informe técnico de mantenimiento.
Usuarios	<ul style="list-style-type: none"> - Generar reportes a detalle de activos a cargo.

Tabla. 3.7: Relación entre actores y procesos en los cuales participan.

Fuente: [Elaboración Propia]

3.2.4.2. CASOS DE USO DEL SISTEMA.-

Se identifican los casos de uso luego de haber identificado el rol de los actores y las funciones en las cuales participan, sintetizando en un caso de uso las funciones principales que debe contar el Sistema de Control y Seguimiento de Activos Fijos.

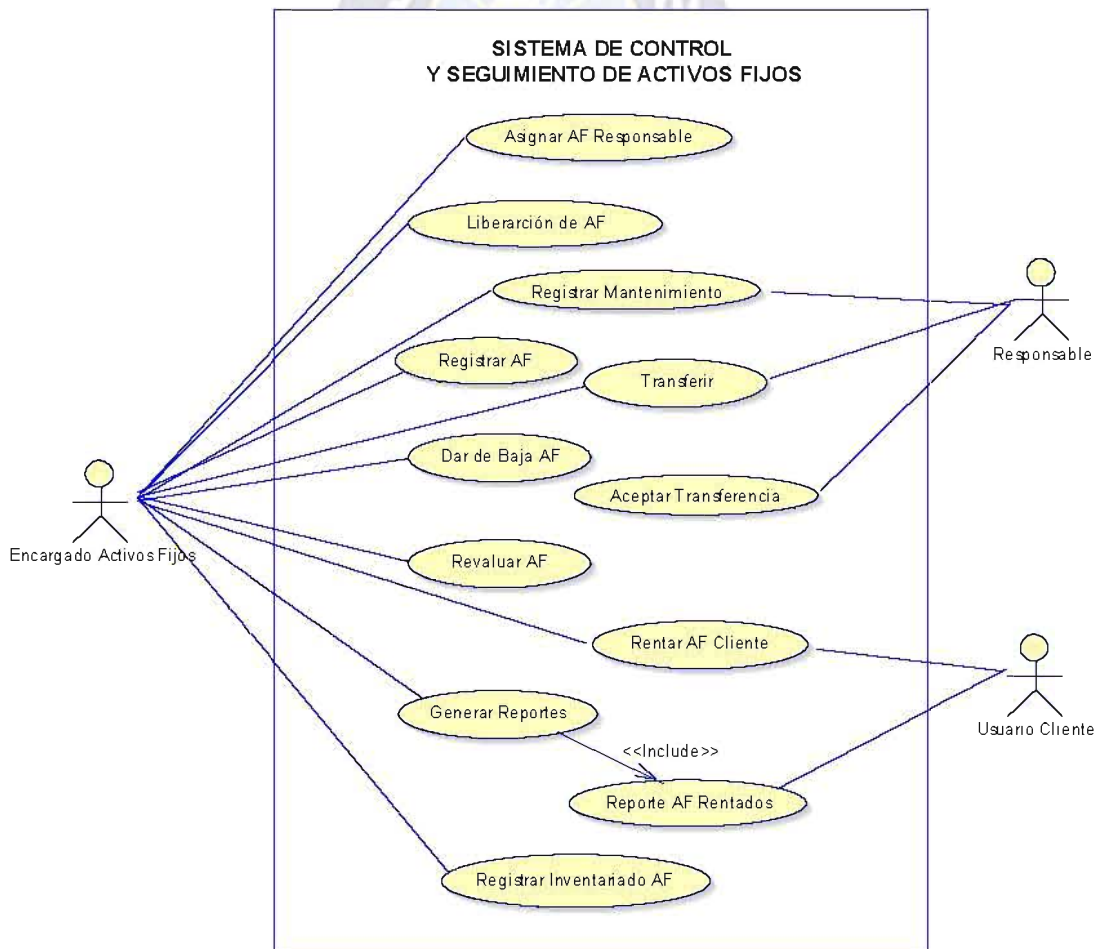


Fig. 3.3: Diagrama de Caso de Uso: Sistema SCSAF.

Fuente: [Elaboración Propia]

3.2.4.3. CASOS DE USO EXPANDIDOS DEL SISTEMA.-

Caso de Uso expandido: Renta de Activos Fijos

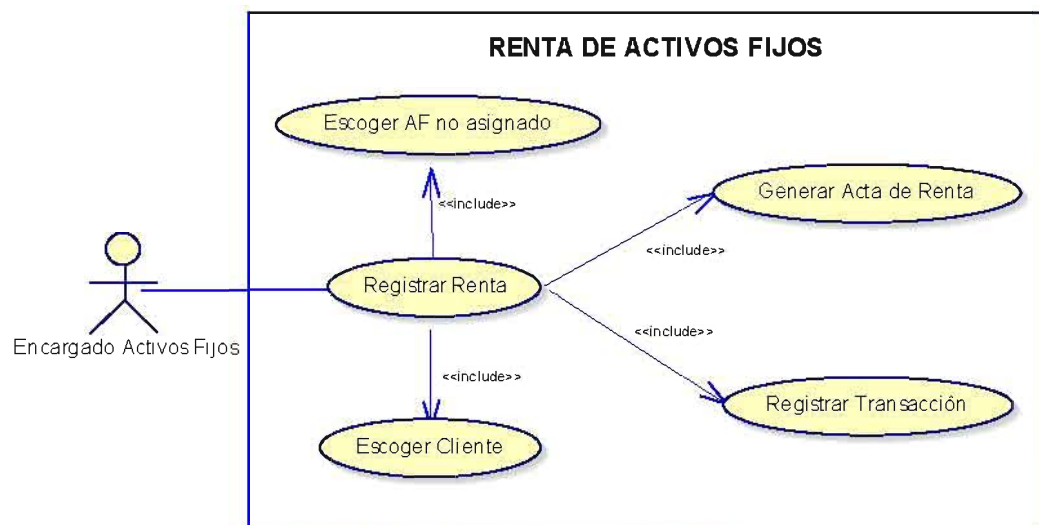


Fig. 3.4: Diagrama de Caso de Uso: Renta de Activos Fijos.

Fuente: [Elaboración Propia]

Descripción del Caso de Uso expandido: Renta de Activos Fijos

Caso de Uso :	Renta de Activos Fijos.
Actores :	Encargado de Activos Fijos.
Propósito :	Asignar un Activo Fijo a un determinado Usuario Cliente.
Admin. AF	Registra la asignación de un activo a un cliente en calidad de renta, y a un empleado en calidad de responsable.
Sistema	Solicita Nro. de Documento de respaldo de la asignación del Activo del cliente o del Empleado.
Admin. AF	Ingresa el Nro. de Documento de respaldo y Acepta la Asignación del Activo Fijo
Sistema	Guarda el registro de la asignación del Cliente o Responsable, Esta operación se registra en el Historial de Transacciones.
Tipo :	Primario
Referencias cruzadas :	R 1, R 2, R 6, R 26, R 27, R 32.

Tabla. 3.8: Caso de Uso expandido: Asignación de Activos Fijos.

Fuente: [Elaboración Propia]

Caso de Uso expandido: Registro de Activos Fijos

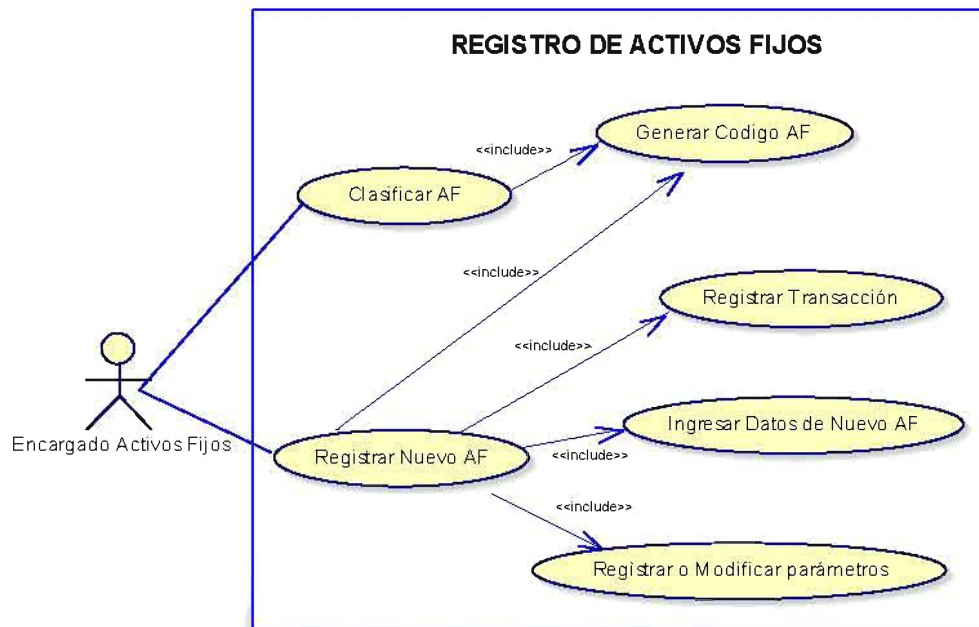


Fig. 3.5: Diagrama de Caso de Uso: Registro de Activos Fijos.

Fuente: [Elaboración Propia]

Descripción del Caso de Uso expandido: Registro de Activos Fijos

Caso de Uso :	Registro de Activos Fijos
Actores :	Encargado de Activos Fijos.
Propósito :	Registrar un Activo Fijo.
Admin. AF	Registra el ingreso de un Activo Fijo, clasificándolo por su grupo y auxiliar contable y otros datos del Activo Fijo.
Sistema	Genera el código tentativo del Activo Fijo.
Admin. AF	Acepta el Registro y el código tentativo generado o lo modifica.
Sistema	Guarda el Registro del Activo Fijo, Esta operación se registra también en el Historial de Transacciones.
Tipo :	Primario
Referencias cruzadas :	R 10, R 26, R 32.

Tabla. 3.9: Caso de Uso expandido: Registro de Activos Fijos.

Fuente: [Elaboración Propia]

Caso de Uso expandido: Inventario de Activos Fijos

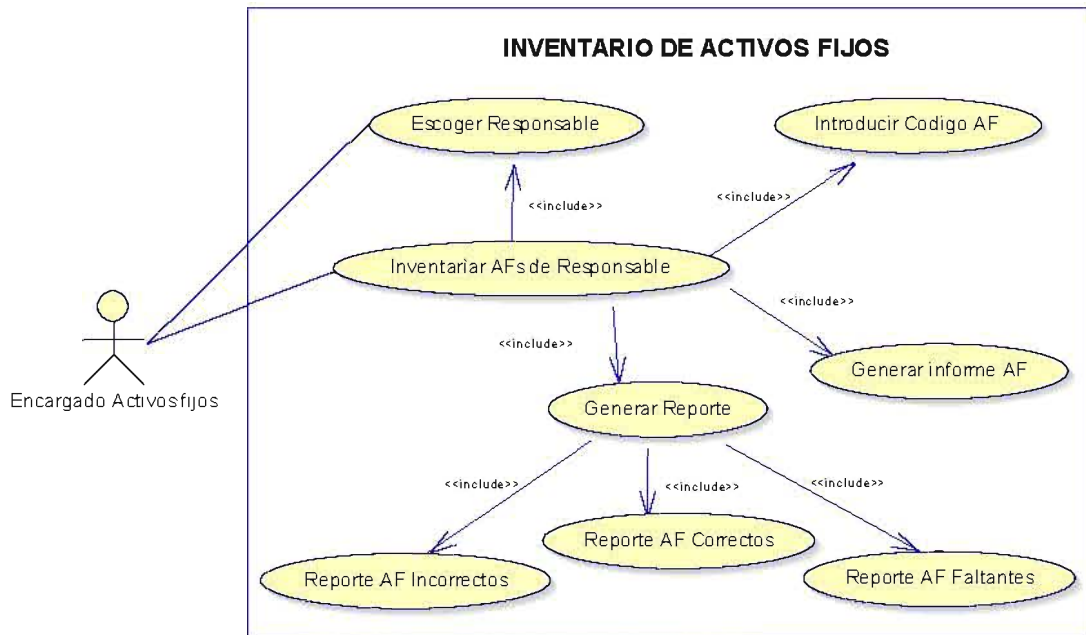


Fig. 3.6: Diagrama de Caso de Uso: Inventario de Activos Fijos.

Fuente: [Elaboración Propia]

Descripción del Caso de Uso expandido: Inventario de Activos Fijos

Caso de Uso :	Inventario de Activos Fijos
Actores :	Encargado de Activos Fijos.
Propósito :	Inventariar los activos fijos asignados a un determinado responsable o cliente.
Admin. AF	Registra el inventario del Activo Fijo. Escoge al cliente y / o responsable del Activo Fijo.
Sistema	Genera un informe parcial para cada activo.
Admin. AF	Introduce los códigos de los Activos Fijos a Inventariar que se encuentra en custodia del cliente y / o responsable.
Sistema	Al terminar el inventariado se genera el reporte de Activos fijos Incorrectos, Activos fijos Correctos y Activos fijos Faltantes.
Tipo :	Primario
Referencias cruzadas :	R 1, R 3, R 5, R 12, R 13, R 32.

Tabla. 3.10: Caso de Uso expandido: Inventario de Activos Fijos.

Fuente: [Elaboración Propia]

Caso de Uso expandido: Mantenimiento de Activos Fijos

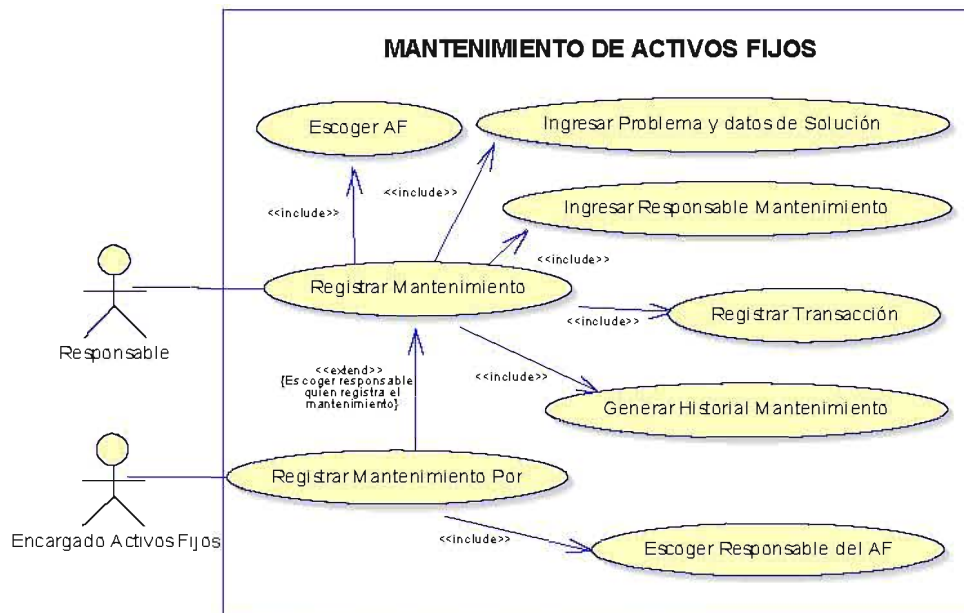


Fig. 3.7: Diagrama de Caso de Uso: Mantenimiento de Activos Fijos.

Fuente: [Elaboración Propia]

Descripción del Caso de Uso expandido: Mantenimiento de Activos Fijos

Caso de Uso :	Mantenimiento de Activos Fijos
Actores :	Encargado de Activos Fijos, Responsable.
Propósito :	Registrar el mantenimiento que se realiza a un Activo Fijo.
Responsable	Solicita y registra los mantenimientos menores que se realiza a los activos asignados a su persona, previa autorización del Encargado de AF. escogiendo el activo e ingresando los datos del problema, solución y responsable del mantenimiento
Admin. AF	Registra los mantenimientos en general que se realiza a un activo, escogiendo al responsable del activo fijo.
Sistema	Registra el Mantenimiento del Activo fijo Genera el historial de mantenimiento del activo fijo. Registra esta operación en el Historial de Transacciones.
Tipo :	Primario
Referencias cruzadas :	R 9, R 10, R 11, R 24, R 32.

Tabla. 3.11: Caso de Uso expandido: Mantenimiento de Activos Fijos.
Fuente: [Elaboración Propia]

Los Casos de Uso: Asignación, Baja, Liberación, Registro, Recepción, Revalorización, Reportes, Transferencias de Activos Fijos, se encuentran desarrollados en el Anexo C.

3.2.5. EVALUACIÓN DE LA FASE DE INICIO

En la Fase de Inicio se avanzó satisfactoriamente, de acuerdo a lo planificado al inicio de la presente fase, se estableció el modelado del negocio y desarrollo sustancialmente la captura de requerimientos del sistema. Como conclusión al estudio de factibilidad realizado, se puede establecer que el sistema propuesto es viable.

3.3. FASE DE ELABORACIÓN

En esta fase transcurrirá a través de los flujos de trabajo: análisis y diseño del sistema, se implementarán los módulos de Asignación, Liberación de Activos Fijos por petición expresa del Usuario Final (Encargado de Activos Fijos). Además se desarrollará las interfaces de Usuario.

3.3.1. PLANIFICACIÓN DE LA FASE DE ELABORACIÓN

En la tabla 3.12 se presenta la planificación para la fase de Elaboración del proyecto detallando la fecha de inicio, fin y aprobación de cada actividad generada.

Actividad generada durante la Fase de Elaboración	Semana	Inicio	Fin	Aprobación o Hito
Análisis / Diseño				
Revisar el modelo de Análisis / Diseño	6 - 14	12-10-10	11-12-10	11-12-10
Revisar el modelo de Datos	6 - 14	12-10-10	11-12-10	11-12-10
Implementación				
Prototipos de Interfaces de Usuario	15 - 25	14-12-10	Siguiente fase	Siguiente fase
Modelo de Implementación	15 - 25	14-12-10	Siguiente fase	Siguiente fase

Tabla. 3.12: Plan de actividades de la Fase de Elaboración

Fuente: [Elaboración Propia]

3.3.2. ANÁLISIS DEL SISTEMA

El modelado del análisis y diseño consiste en obtener una visión del sistema modelando los requisitos funcionales planteados en la captura de requerimientos. En esta fase, se usó el diagrama de secuencias para conseguir una comprensión más precisa de los requisitos, refinándolos y estructurándolos, logrando una descripción detallada del funcionamiento de cada módulo del sistema.

3.3.2.1. DIAGRAMAS DE SECUENCIA

Desarrollados los caso de uso expandidos, elaboramos los diagramas de secuencia relacionados a dichos casos de uso, en los que se realizan las operaciones que requieren los actores del sistema.

Diagrama de Secuencia: Renta de Activos Fijos

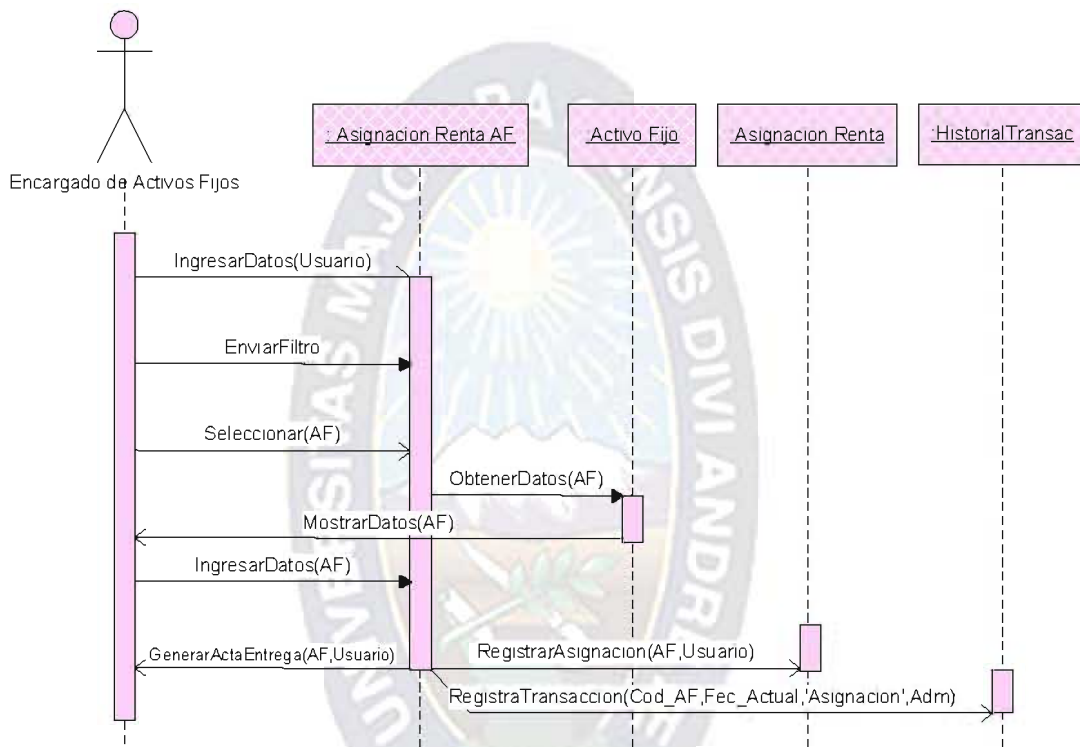


Fig. 3.8 Diagrama de Secuencia: Renta de Activos Fijos.

Fuente: [Elaboración Propia]

Diagrama de Secuencia: Registro de Activos Fijos.

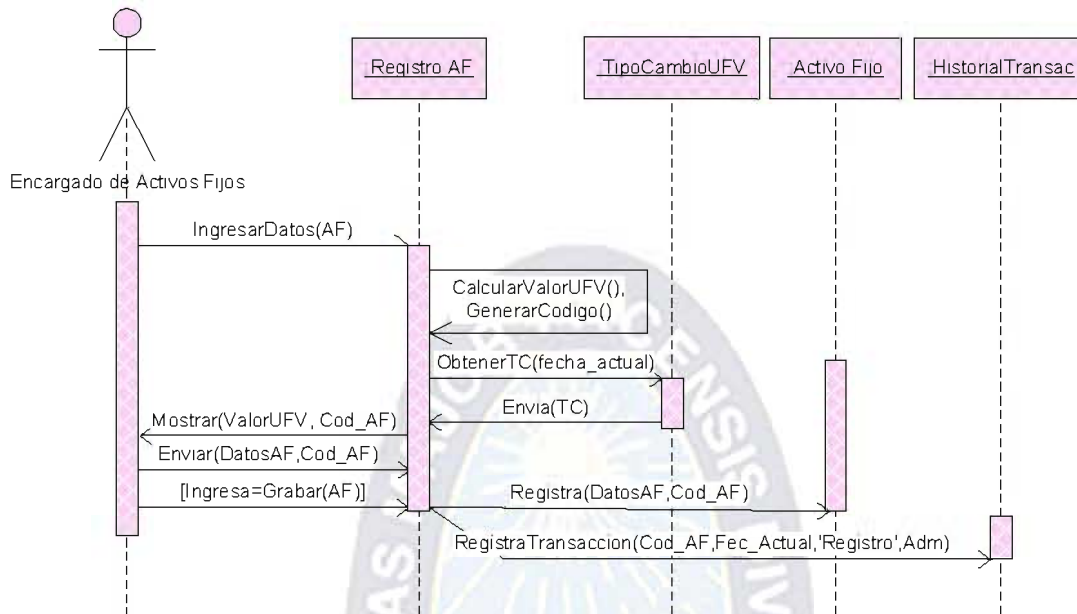


Fig. 3.9: Diagrama de Secuencia: Registro de Activos Fijos.

Fuente: [Elaboración Propia]

Diagrama de Secuencia: Mantenimiento de Activos Fijos.

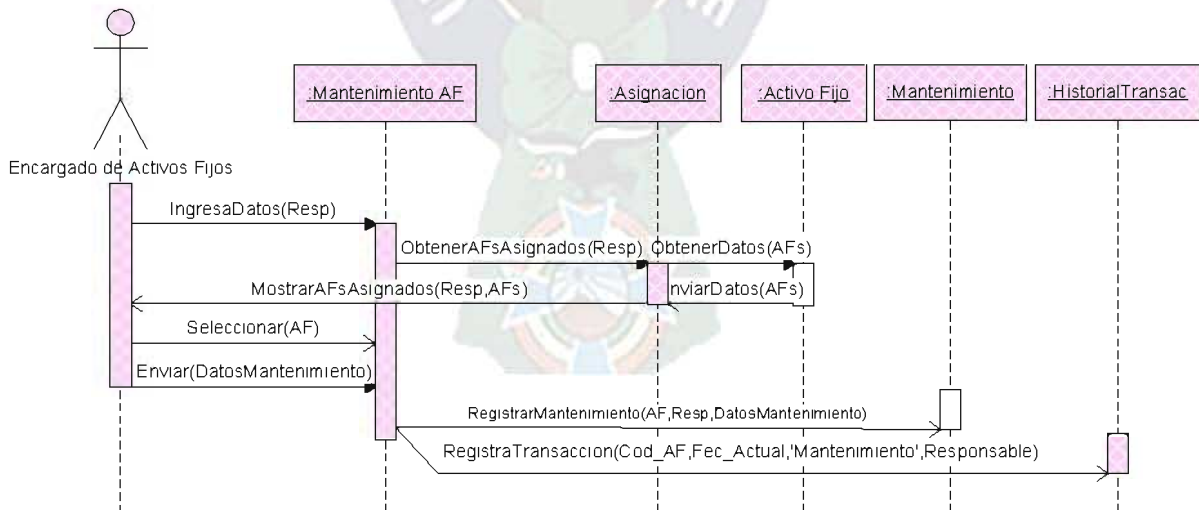


Fig. 3.10: Diagrama de Secuencia: Mantenimiento de Activos Fijos.

Fuente: [Elaboración Propia]

Diagrama de Secuencia: Inventario de Activos Fijos.

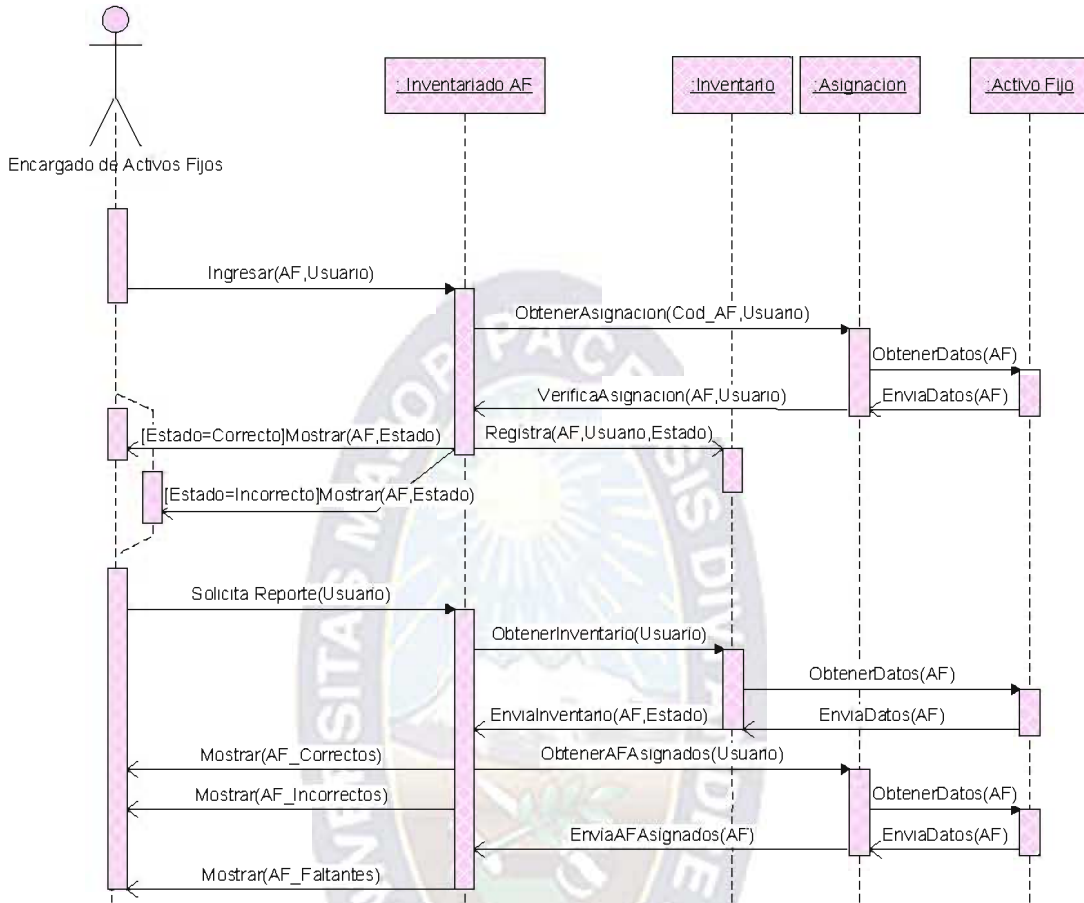


Fig. 3.11: Diagrama de Secuencia: Inventario de Activos Fijos.

Fuente: [Elaboración Propia]

El resto de estos diagramas se encuentra en el Anexo C.

3.3.3. DISEÑO DEL SISTEMA

3.3.3.1. MODELO DE CASO DE USO PARA DISEÑO FINAL:

Dentro del Modelo de Caso de Uso para diseño final se presenta el Diagrama de Caso de Uso real siguiente:

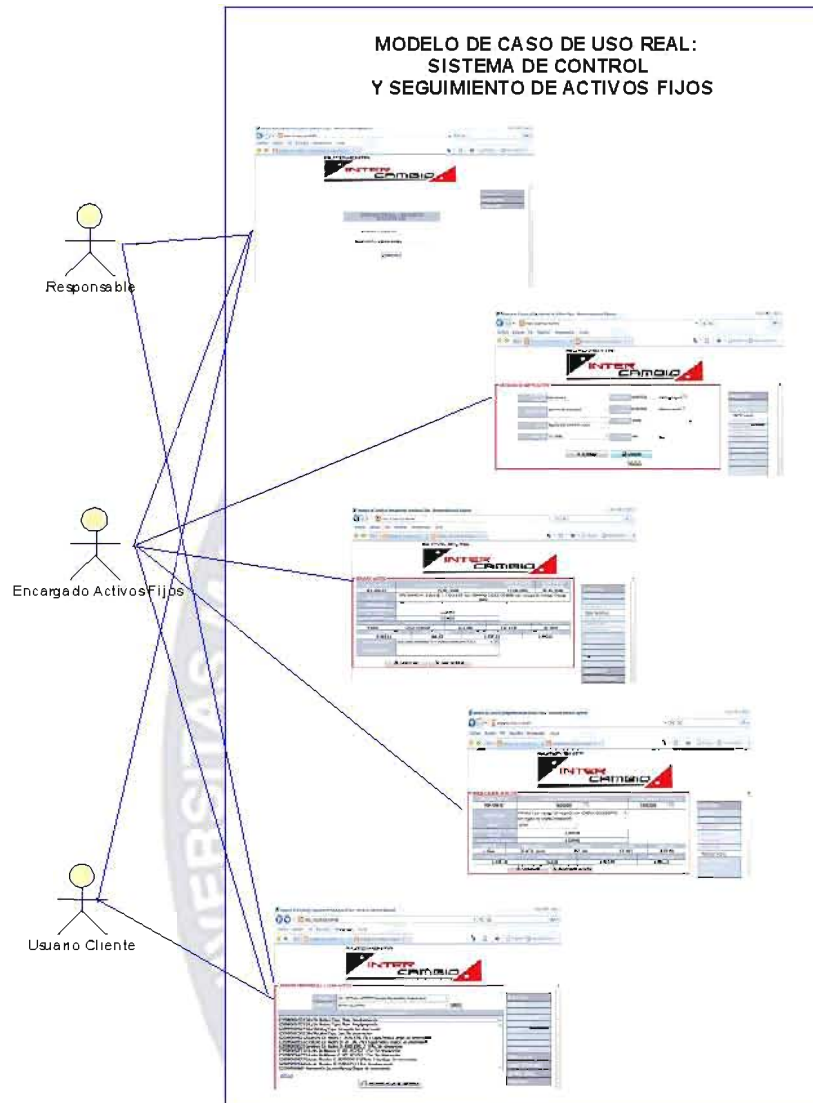


Fig. 3.12: Diagrama de Caso de Uso Real: Sistema SCSAF.

Fuente: [Elaboración Propia]

3.3.3.2. MODELO ENTIDAD-RELACION DIAGRAMA DE CLASES

En el Modelo de Entidad-Relación se presenta el diagrama de Clases (Fig. 3.13) en la que se diseñó las asociaciones, clases asociadas, atributos que se requiere para construir la

base de datos, también se incluye todos los métodos analizados en los diagramas de secuencia.



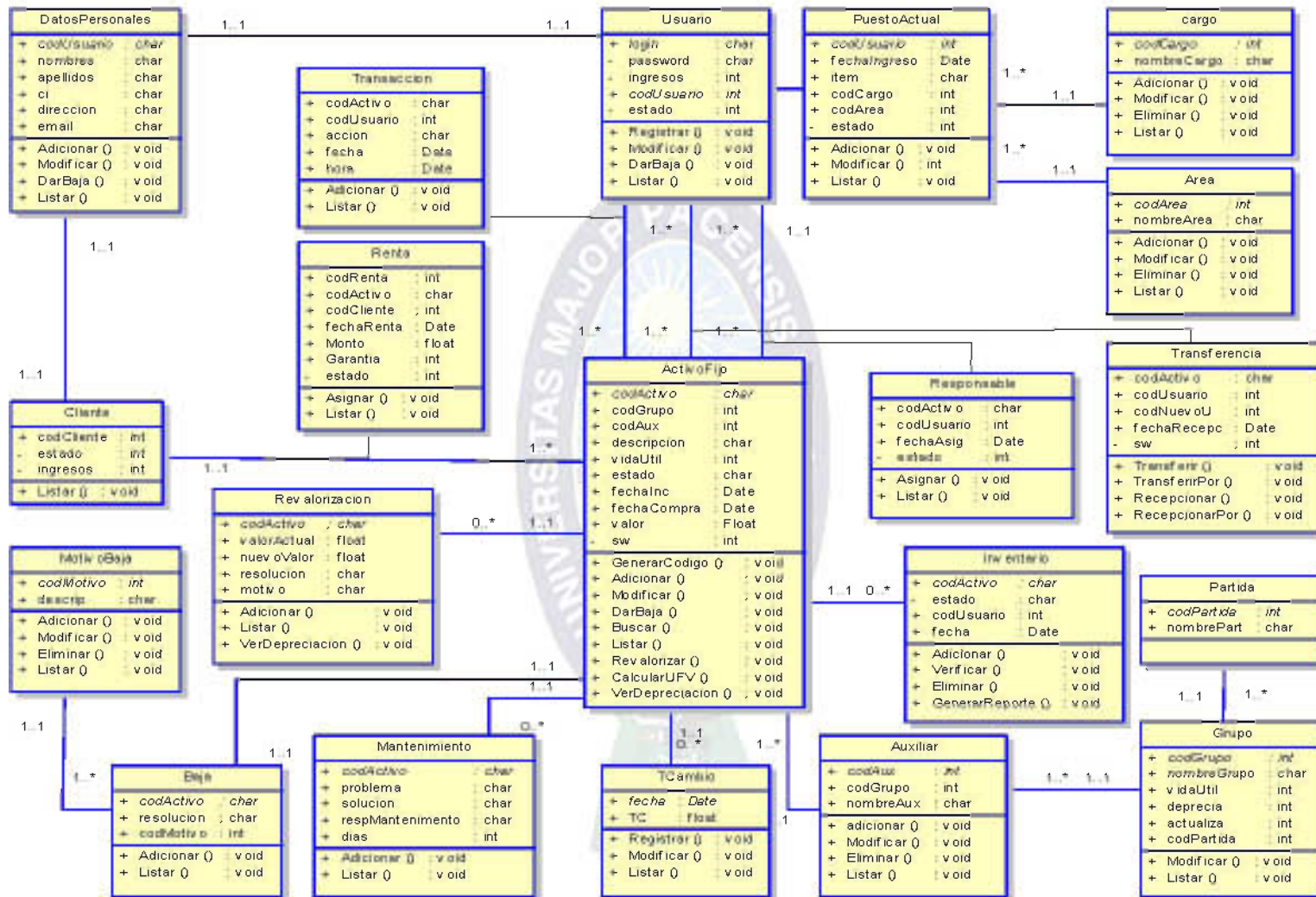


Fig. 3.13: Modelo Entidad-Relacion: Diagrama de Clases del Sistema.
Fuente: [Elaboración Propia]

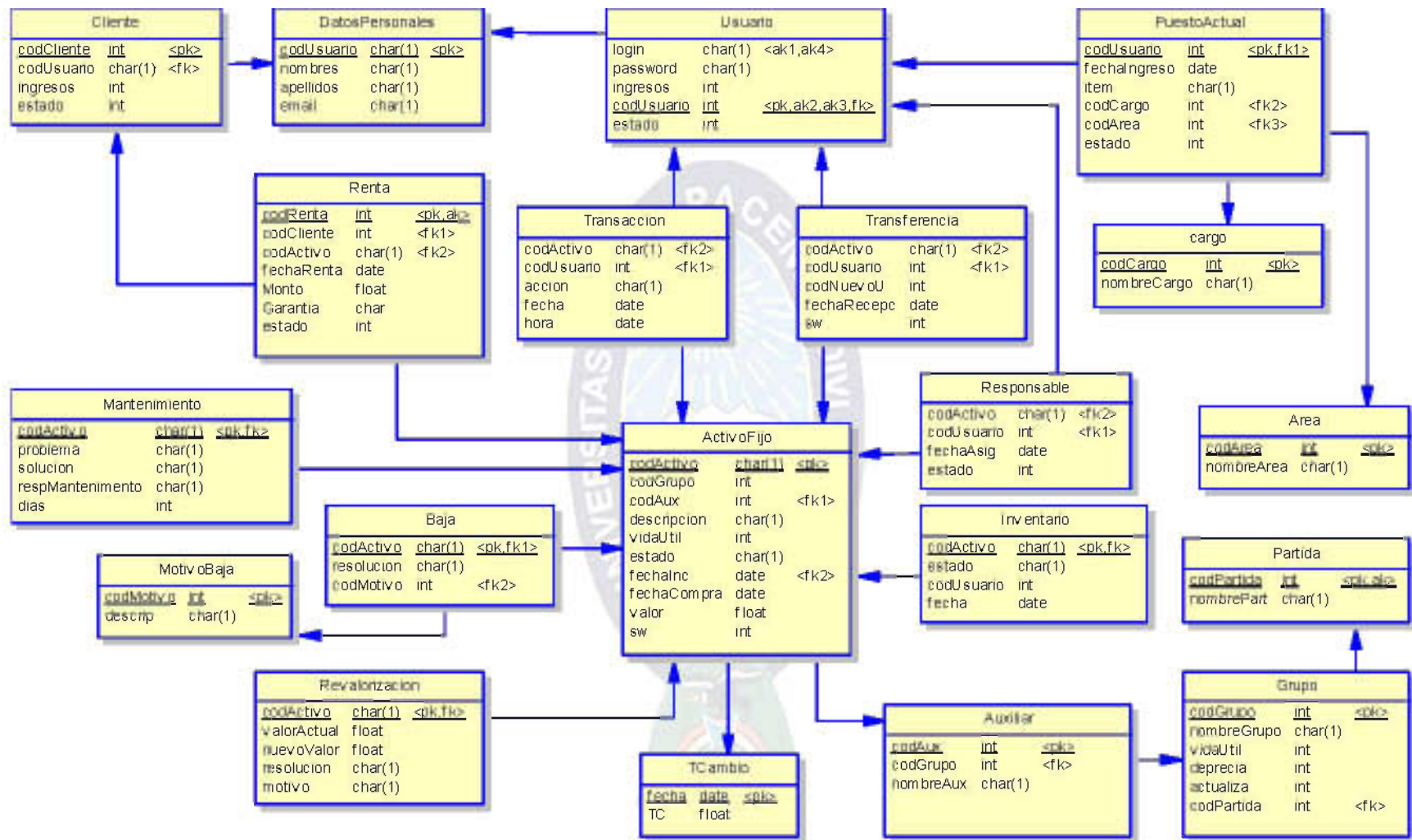


Fig. 3.14:Modelo Fisico: Diseño de la Base de Datos del Sistema.

Fuente: [Elaboración Propia]

3.3.3.3. MODELO DE DATOS

El almacenamiento de los datos en nuestro sistema, se implementará en un gestor de base de datos relacional, por lo tanto utilizaremos las siguientes correspondencias para definir su estructura física, independientemente de que sean implementados en una base de datos orientada a objetos o relacional. Definimos la estructura física de datos que utilizará el sistema, (ver fig. 3.14) a partir del modelo entidad-relación del diseño de clases (ver fig. 3.13),

- Las clases persistentes y sus atributos hacen referencia a las entidades lógicas y sus atributos.
- Las asociaciones entre clases, a relaciones entre entidades.
- Las clases asociadas, a las relaciones con atributos y llaves foráneas.

3.3.4. EVALUACIÓN DE LA FASE DE ELABORACIÓN

En la Fase de Elaboración se avanzó satisfactoriamente de acuerdo a lo planificado al inicio de la presente fase, se realizó el estudio del análisis y diseño del sistema con los diagramas de secuencia y el diagrama de clases, obteniendo el modelo de datos.

3.4. FASE DE CONSTRUCCIÓN

3.4.1. PLANIFICACIÓN DE LA FASE DE CONSTRUCCIÓN

En la tabla 3.12 se presenta la planificación para la fase de Construcción del proyecto detallando la fecha de inicio, fin y aprobación de cada actividad generada.

Actividad generada durante la Fase de Construcción	Semana	Inicio	Fin	Aprobación o Hito
Casos de Uso Negociados.				
Asignación de AF	15	14-12-10	18-12-10	21-12-10
Liberación de AF	15	14-12-10	18-12-10	21-12-10
Renta de AF	16	21-12-10	24-12-10	29-01-11
Administración de Usuarios	17 - 18	28-12-10	08-01-11	29-01-11
Registro de AF	19	11-01-11	15-01-11	29-01-11
Baja de AF	20	18-01-11	22-01-11	29-01-11
Transferencia de AF	21	25-01-11	29-01-11	29-01-11
Recepción de AF	22	01-02-11	05-02-11	Siguiente fase
Revalorización de AF	23	08-02-11	12-02-11	Siguiente fase
Mantenimiento de AF	23	08-02-11	12-02-11	Siguiente fase
Inventariado de AF	24	15-02-11	19-02-11	Siguiente fase
Generar Reportes	25	22-02-11	26-02-11	Siguiente fase
Pruebas				
Casos de Pruebas Funcionales	16 - 25	21-12-10	26-02-11	Revisar en cada hito aprobado

Tabla. 3.12: Plan de actividades de la Fase de Construcción.

Fuente: [Elaboración Propia]

3.4.2. IMPLEMENTACIÓN DEL SISTEMA

En esta Fase se desarrollarán el diagrama de componentes, interfaz de usuario y casos de prueba de los módulos del sistema. La implementación describe cómo los elementos del modelo de diseño se implementarán en términos de componentes como: páginas html, ficheros de código fuente, ejecutables, etc.

a) DIAGRAMA DE COMPONENTES

Los componentes son típicamente los archivos de implementación en el ambiente de desarrollo. El diagrama de componentes describe los componentes de software y sus dependencias entre sí, garantizando que se implemente su funcionalidad correcta.

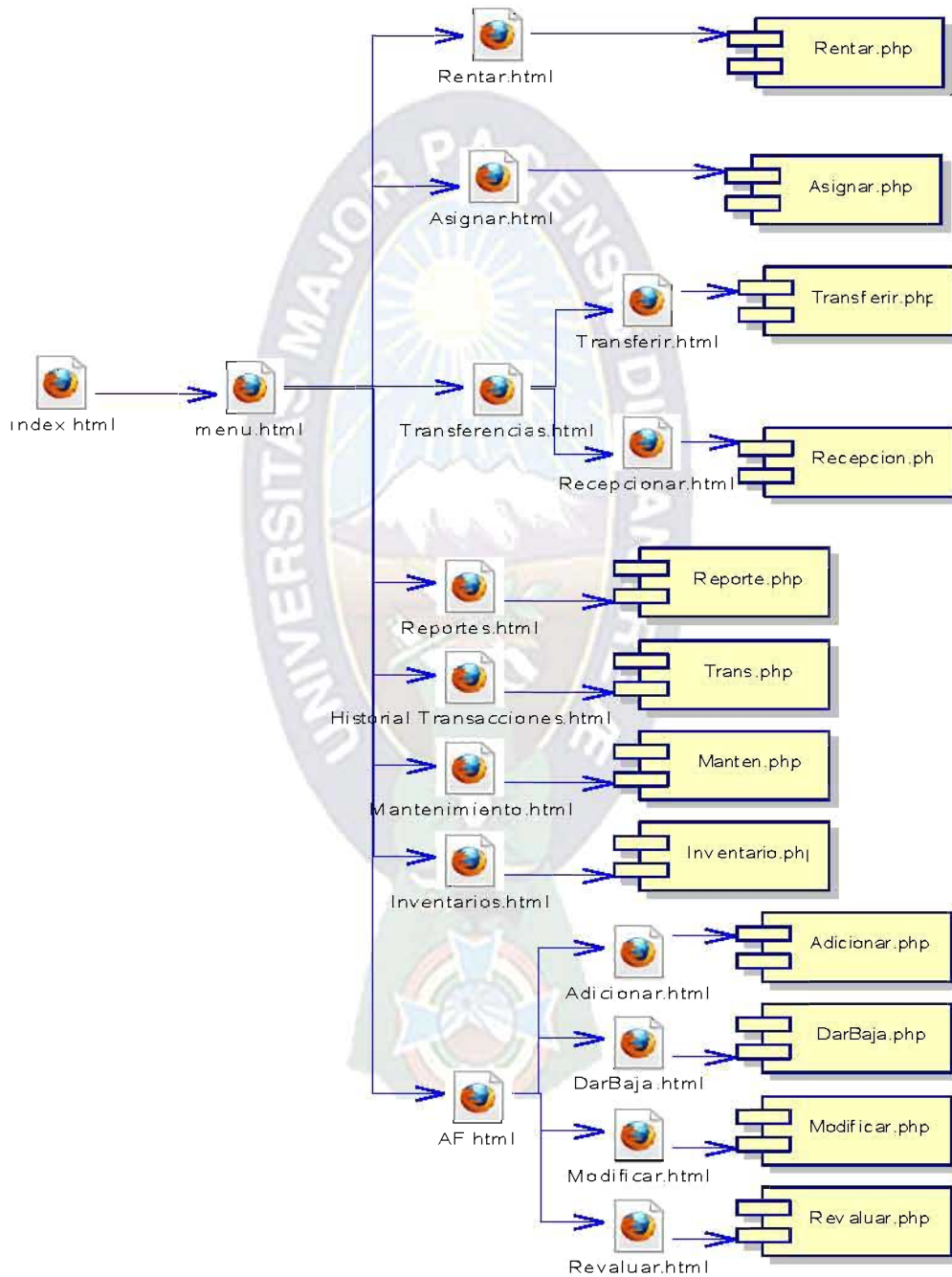


Fig. 3.14: Diagrama de Componentes del Sistema.

Fuente: [Elaboración Propia]

b) INTERFAZ GRÁFICA DE USUARIO

El objetivo del diseño de la Interfaz Gráfica de Usuario es facilitar la manipulación del sistema, de una manera agradable, de forma que pueda ser utilizado por personas sin amplia experiencia en el manejo de sistemas informáticos.

A continuación se presentan algunas interfaces gráficas de usuario diseñadas de la aplicación final.

La figura 3.15 es el menú inicial del sistema.

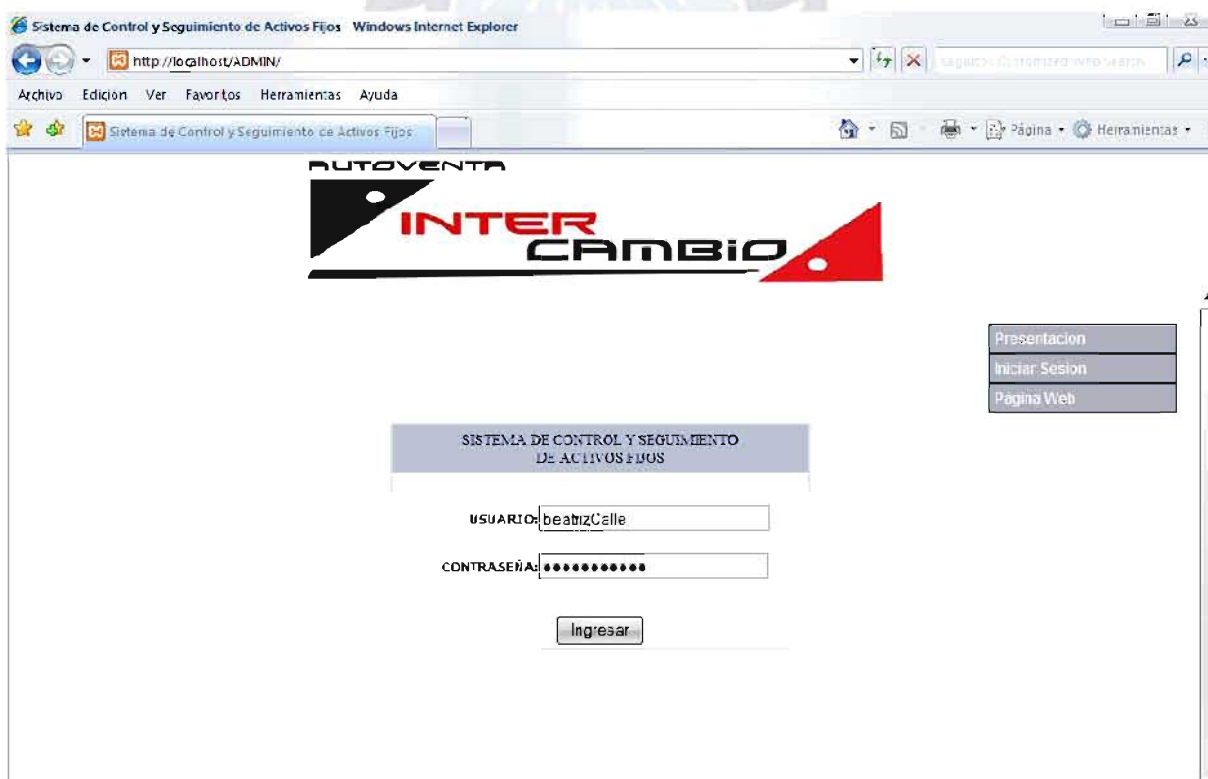


Fig. 3.15: Interfaz de Usuario: Pagina Inicial.

Fuente: [Elaboración Propia]

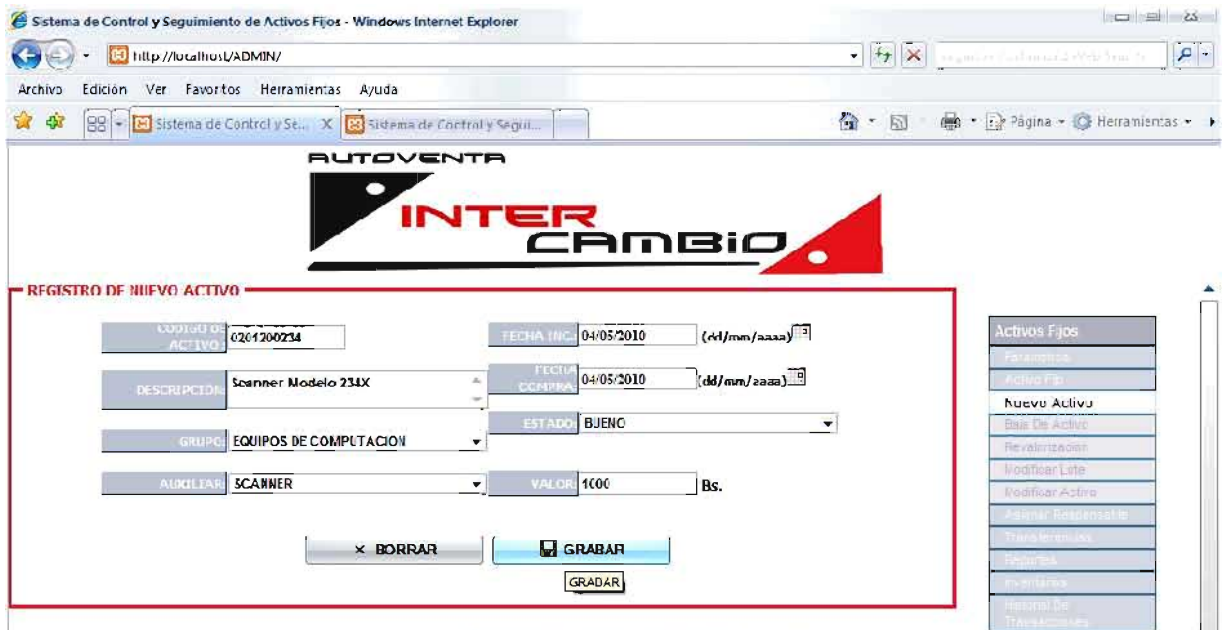


Fig. 3.16: Interfaz de Usuario: Registro de Nuevo Activo.

Fuente: [Elaboración Propia]

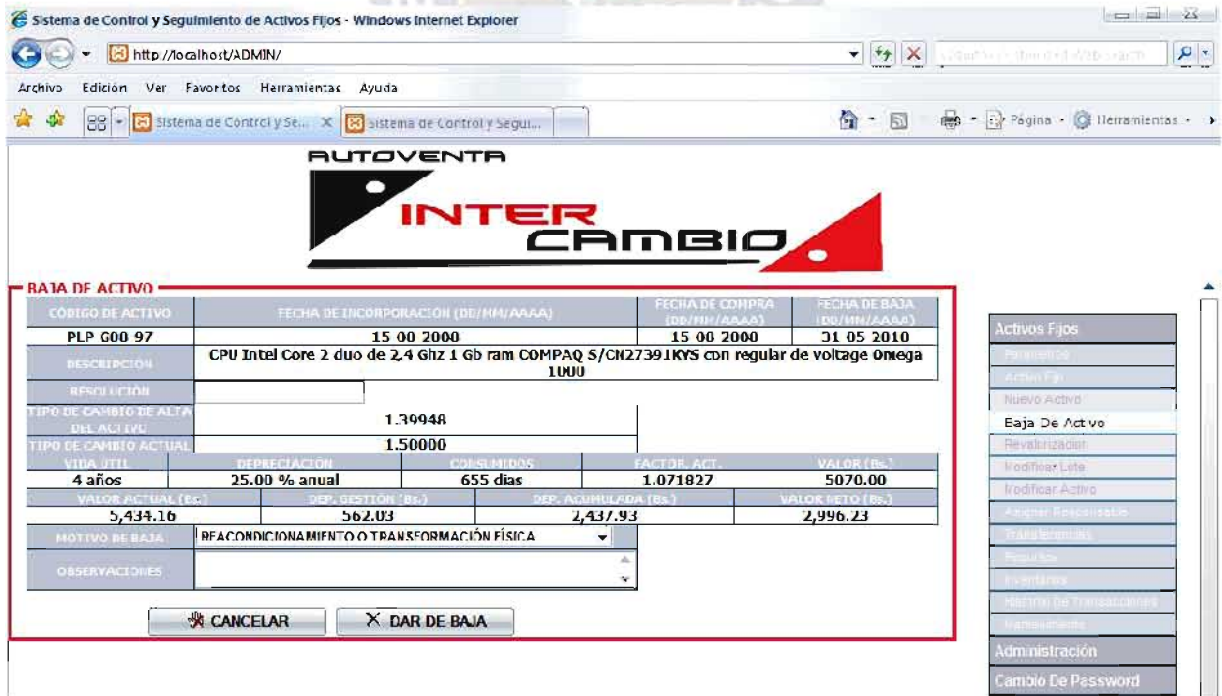


Fig. 3.17: Interfaz de Usuario: Baja de Activos.

Fuente: [Elaboración Propia]

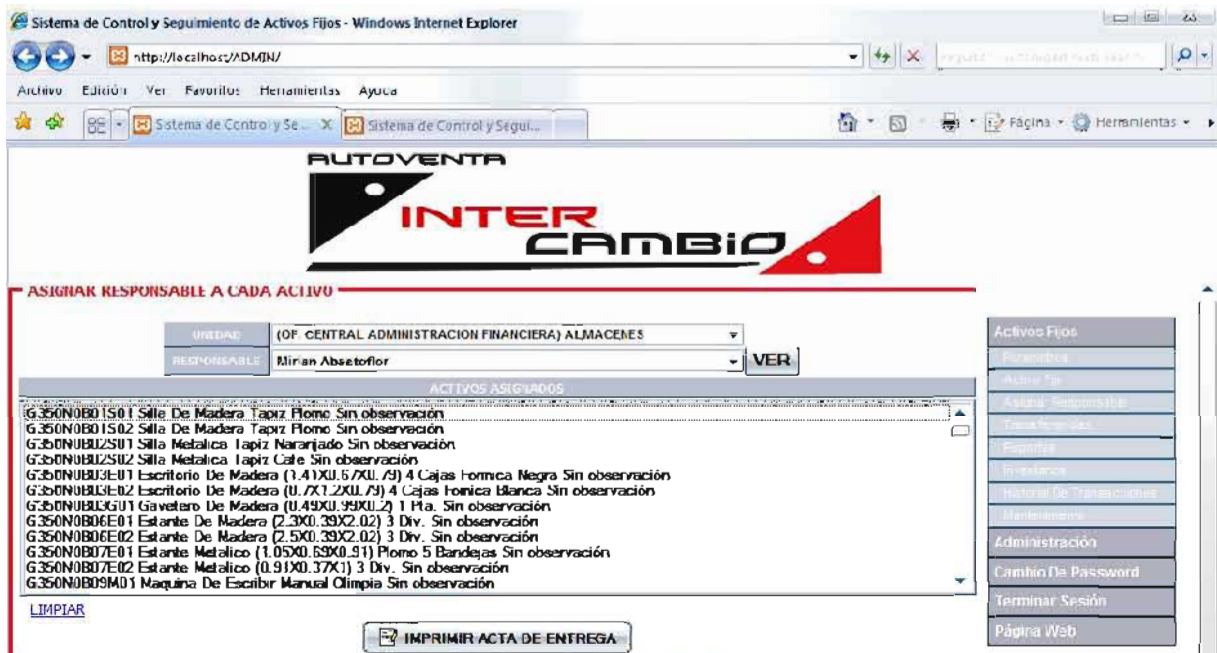


Fig. 3.18: Interfaz de Usuario: Asignación de Activos Fijos.

Fuente: [Elaboración Propia]

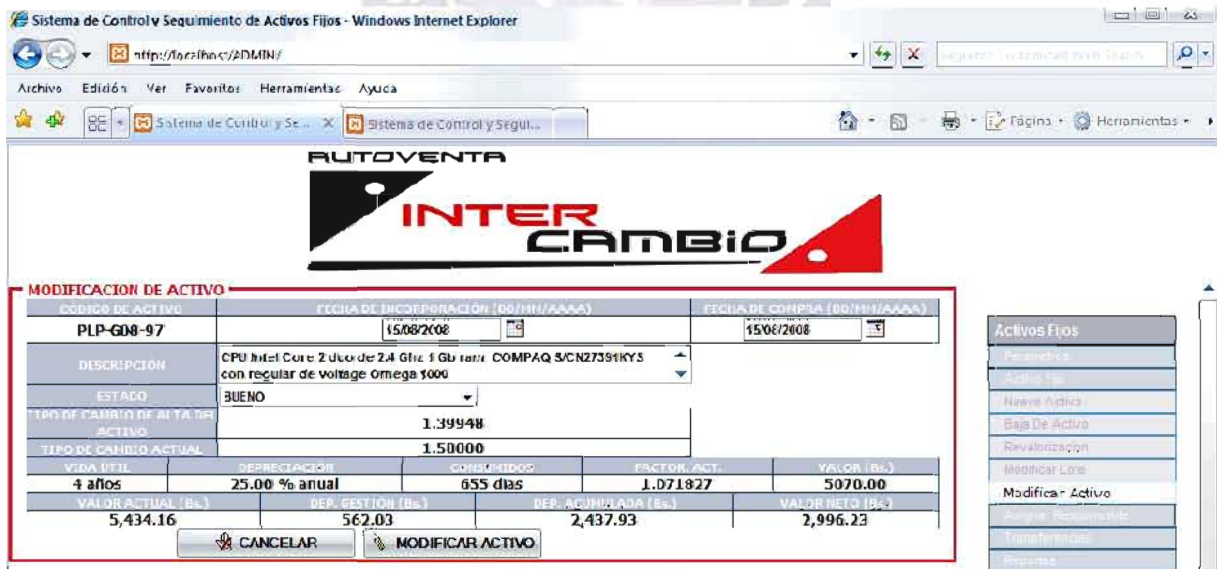


Fig. 3.19: Interfaz de Usuario: Modificación de Activos Fijos.

Fuente: [Elaboración Propia]

El resto de las Interfaces de Usuario mencionadas en los diagramas de secuencia, se presentan en el Anexo D.

3.4.3. PRUEBAS DEL SISTEMA

Según Pressman y considerando a la prueba parte de la Ingeniería de Software, existen dos tipos de pruebas complementarias: Prueba de Caja Blanca y Prueba de Caja Negra.

a) PRUEBA DE LA CAJA BLANCA

Para obtener una medida de la complejidad lógica de un diseño procedimental y utilizar esta medida como guía para la definición de un conjunto básico de caminos de ejecución se realiza la prueba del camino básico que es una técnica de prueba de caja blanca, que permite al diseñador de casos de prueba calcular el valor de la complejidad ciclomática que define el número de caminos independientes del conjunto básico de un programa y proporciona el límite superior para el número de pruebas que se debe realizar en un módulo.

La complejidad ciclomática es una métrica del software que proporciona una medida cuantitativa de la complejidad lógica de un programa.

Para calcular el número de módulos en los cuales realizar la prueba de caja blanca nos basamos en el Principio de Pareto: aproximadamente en el 20% de los módulos de un sistema, existe 80% de error en su implementación.

$$\text{N}^\circ \text{ de módulos del Sistema} = 10$$

$$\text{N}^\circ \text{ de módulos a Revisar} = 10 \cdot (20\%) = 2$$

Prueba: Recepcionar Por

La figura 3.21 muestra el Grafo Flujo del módulo Recepción Por que se realizó con la ayuda del Diagrama de Flujo que se muestra en la figura 3.20.

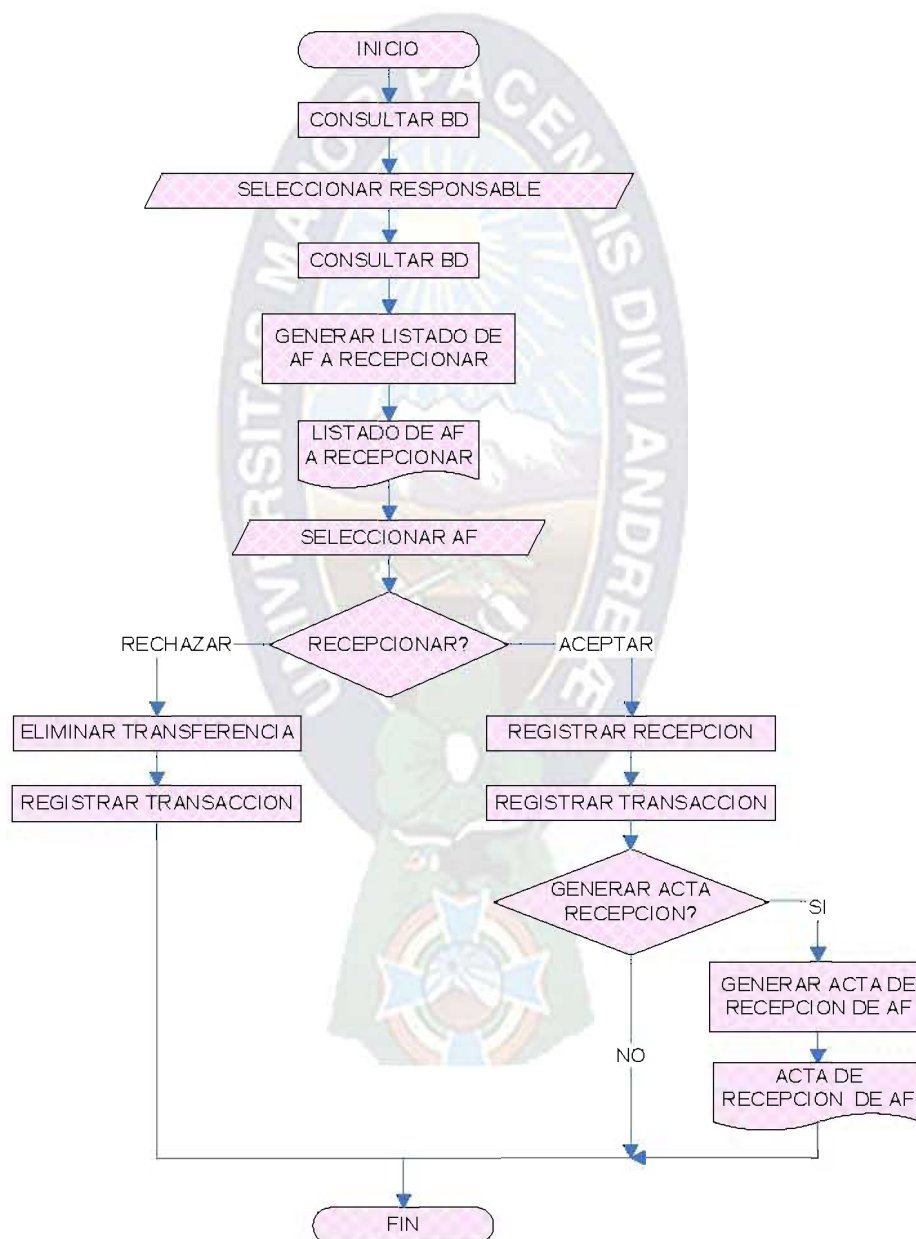


Fig. 3.20: Diagrama de Flujo: Recepción Por.

Fuente: [Elaboración Propia]

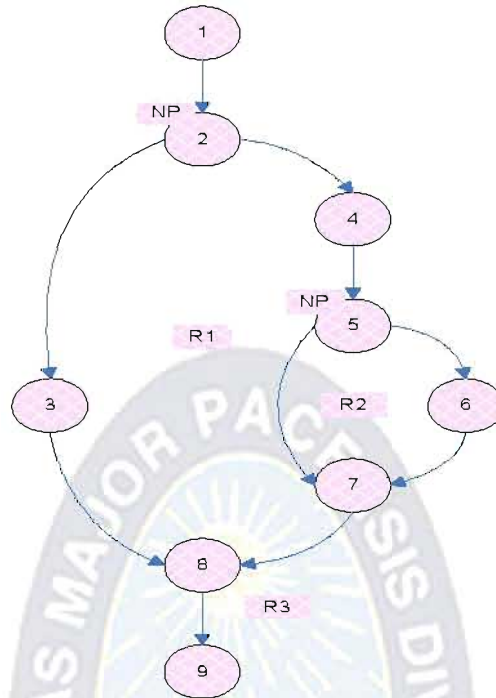


Fig. 3.21: Grafo Flujo: Recepción Por.

Fuente: [Elaboración Propia]

Datos:

Nº de Nodos, $N = 9$

Nº de Aristas, $A = 10$

Nº de Regiones = 3

Nº de Nodo Predicados NP = 2

Complejidad Ciclomática:

$$V(G) = \{R1, R2, R3\} = 3 \text{ //Por Regiones}$$

$$V(G) = A - N + 2 = 10 - 9 + 2 = 3 \text{ //Por Nodos y Aristas}$$

$$V(G) = NP + 1 = 2 + 1 = 3 \text{ //Por Nodo Predicados}$$

La complejidad ciclomática es 3, por lo tanto existen 3 caminos independientes para el proceso de Recepcionar Por.

Caminos Independientes:

C1 = 1, 2, 4, 5, 6, 7, 8, 9.

C2 = 1, 2, 4, 5, 7, 8, 9.

C3 = 1, 2, 3, 8, 9.

Caso de Prueba: Camino 1

Seleccionar (Responsable=Jorge Chavez)

Seleccionar (CodigoAF=0100200156)= Seleccionar **Aceptar**

RegistrarRecepción(CodAF= 0100200156, Responsable=Jorge Chavez)

RegistrarTransacción(AdmAF, 2009-12-18, 11:30, Recepción Por)

MostrarListado (AF_Asignados_A=Jorge Chavez)= GenerarActa **SI**

GenerarActa(AF_Recepcionados_De=Jorge Chavez)

ImprimirActa(AF_Recepcionados_De=Jorge Chavez)

Salir(Recepción Por)

Caso de Prueba: Camino 2

Seleccionar (Responsable=Jorge Chavez)

Seleccionar (CodigoAF=0100200157)= Seleccionar **Aceptar**

RegistrarRecepción(CodAF= 0100200157, Responsable=Jorge Chavez)

RegistrarTransacción(AdmAF, 2009-12-18, 11:35, Recepción Por)

MostrarListado (AF_Asignados_A=Jorge Chavez)= GenerarActa **NO**

Salir(Recepción Por)

Caso de Prueba: Camino 3

Seleccionar (Responsable=Jorge Chavez)

Seleccionar (CodigoAF=0100200158)= Seleccionar **Rechazar**

EliminarTransferencia(CodAF= 0100200158, Destinatario=Jorge Chavez)

RegistrarTransacción(AdmAF, 2009-12-18, 11:40, RechazarTransferenciaPor)

Salir(Recepción Por)

Se concluye que para todos los casos de prueba del Grafo Flujo: Recepción Por, existe un camino independiente y libre de bucles al infinito.

Prueba: Asignación y Liberación de Responsable

La figura 3.22 se muestra el Diagrama de Flujo con el cual se ayuda a desarrollar el Grafo Flujo del módulo Asignación de Responsables que se presenta en la figura 3.23.

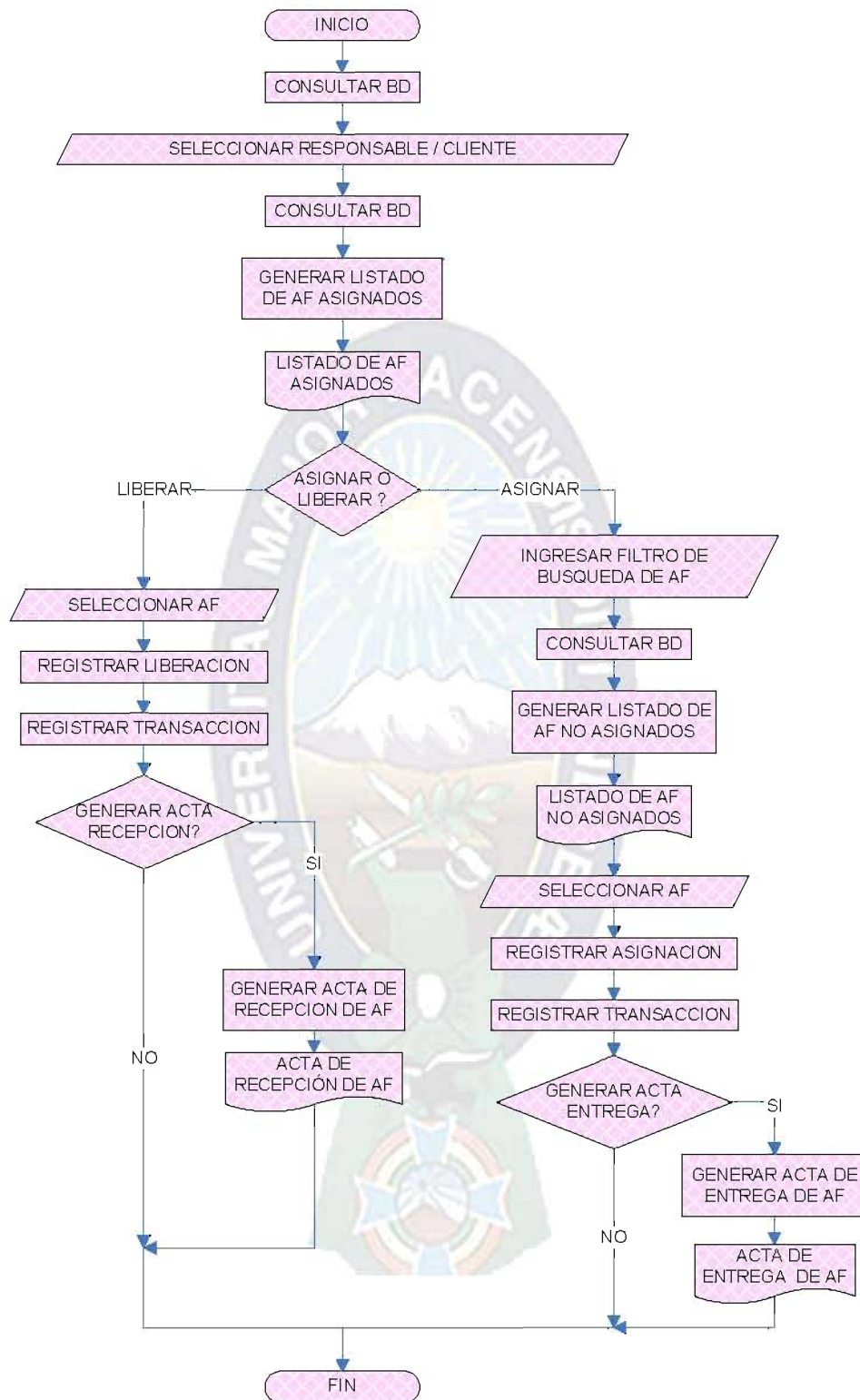


Fig. 3.22: Diagrama de Flujo: Asignación y Liberación de Responsables.

Fuente: [Elaboración Propia]

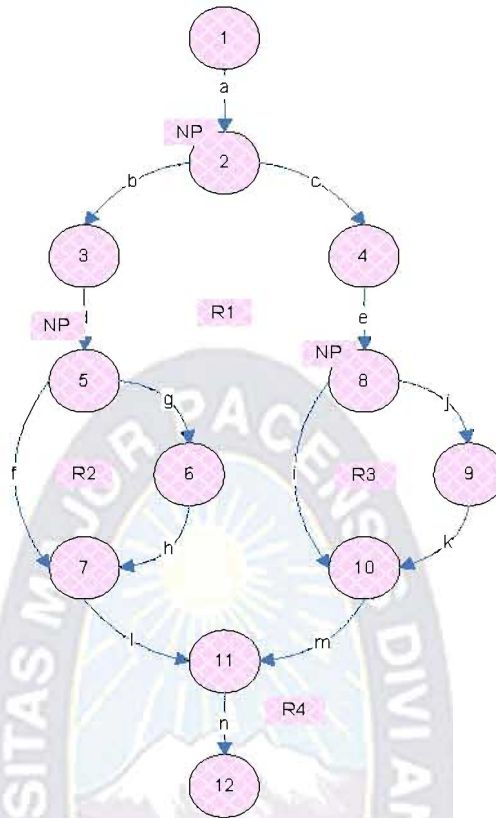


Fig. 3.23: Grafo Flujo: Asignación y Liberación de Responsables.

Fuente: [Elaboración Propia]

Datos:

Nº de Nodos, $N = 12$

Nº de Aristas, $A = 14$

Nº de Regiones = 4

Nº de Nodo Predicados NP = 3

Complejidad Ciclomática:

$V(G) = \{R1, R2, R3, R4\} = 4$ //Por Regiones

$$V(G) = A - N + 2 = 14 - 12 + 2 = 4 \text{ //Por Nodos y Aristas}$$

$$V(G) = NP + 1 = 3 + 1 = 4 \text{ //Por Nodo Predicados}$$

La complejidad ciclomática es 4, por lo tanto existen 4 caminos independientes para el proceso de Asignación y Liberación de Responsables.

Caminos Independientes:

C1 = 1, 2, 4, 8, 9, 10, 11, 12.

C2 = 1, 2, 4, 8, 10, 11, 12.

C3 = 1, 2, 3, 5, 6, 7, 11, 12.

C4 = 1, 2, 3, 5, 7, 11, 12.

Caso de Prueba: Camino 1

Seleccionar (Responsable=Jorge Chavez)

GenerarListado (AF_Asignados_A=Jorge Chavez)

MostrarListado (AF_Asignados_A=Jorge Chavez)= Selección Asignar

SeleccionarFiltroBusqueda(CodigoAF=0100200159)

Seleccionar(AF_No_Asignado= 0100200159)

RegistrarAsignación(CodAF= 0100200159, Responsable=Jorge Chavez)

RegistrarTransacción(AdmAF, 2009-12-18, 11:00, Asignación)

MostrarListado (AF_Asignados_A=Jorge Chavez)= GenerarActa

GenerarActa(AF_Asignados_A=Jorge Chavez)

ImprimirActa(AF_Asignados_A=Jorge Chavez)

Salir(Asignación y Liberación de Responsables)

Caso de Prueba: Camino 2

Seleccionar (Responsable=Jorge Chavez)

GenerarListado (AF_Asignados_A=Jorge Chavez)

MostrarListado (AF_Asignados_A=Jorge Chavez)= Selección Asignar

SeleccionarFiltroBusqueda(CodigoAF=0100200160)

Seleccionar(AF_No_Asignado= 0100200160)

RegistrarAsignación(CodAF= 0100200160, Responsable=Jorge Chavez)

RegistrarTransacción(AdmAF, 2009-12-18, 11:10, Asignación)

MostrarListado (AF_Asignados_A=Jorge Chavez)= NO GenerarActa

Salir(Asignación y Liberación de Responsables)

Caso de Prueba: Camino 3

Seleccionar (Responsable=Jorge Chavez)

GenerarListado (AF_Asignados_A=Jorge Chavez)

MostrarListado (AF_Asignados_A=Jorge Chavez)= Selección Liberar

Seleccionar(AF_Asignado= 0100200159)

RegistrarLiberación(CodAF= 0100200159, Responsable=Jorge Chavez)

RegistrarTransacción(AdmAF, 2009-12-18, 11:15, Liberación)

MostrarListado (AF_Asignados_A=Jorge Chavez)= GenerarActa

GenerarActa(AF_Asignados_A=Jorge Chavez)

ImprimirActa(AF_Asignados_A=Jorge Chavez)

Salir(Asignación y Liberación de Responsables)

Caso de Prueba: Camino 4

Seleccionar (Responsable=Jorge Chavez)

GenerarListado (AF_Asignados_A=Jorge Chavez)

MostrarListado (AF_Asignados_A=Jorge Chavez)= Selección Liberar

Seleccionar(AF_Asignado= 0100200160)

RegistrarLiberación(CodAF= 0100200160, Responsable=Jorge Chavez)

RegistrarTransacción(AdmAF, 2009-12-18, 11:25, Liberación)

MostrarListado (AF_Asignados_A=Jorge Chavez)= NO GenerarActa

Salir(Asignación y Liberación de Responsables)

Se concluye que para todos los casos de prueba del Grafo Flujo: Asignación y Liberación de AF, existe un camino independiente y libre de bucles.

b) PRUEBA DE CAJA NEGRA

Este tipo de prueba nos permite encontrar errores de las siguientes categorías:

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores de rendimiento.
- Errores de inicialización y/o terminación.

En el presente proyecto las pruebas de caja negra se realizaron en la fase de construcción y transición, para luego realizar sus respectivas correcciones, hasta llegar a los resultados esperados.

En este tipo de pruebas se demuestra la funcionalidad operativa del sistema, realizando pruebas de ejecución del programa, verificando el funcionamiento correcto de cada uno de los módulos del sistema, es un enfoque complementario a las técnicas de la caja blanca que intenta descubrir diferentes tipos de errores.

3.4.4. EVALUACIÓN DE LA FASE DE CONSTRUCCIÓN

En la Fase de Construcción se avanzo satisfactoriamente de acuerdo a lo planificado al inicio de la presente fase, se desarrollo el diagrama de componentes, se finalizo la implementación de todos los módulos del sistema realizando pruebas de caja blanca y caja negra.

3.5. FASE DE TRANSICIÓN

3.5.1. PLANIFICACIÓN DE LA FASE DE TRANSICIÓN

En la tabla 3.13 se presenta la planificación para la fase de Transición del proyecto, detallando la fecha de inicio, fin y aprobación de cada actividad generada.

Actividad generada o modificada durante la Fase de Transición	Semana	Inicio	Fin	Aprobación o Hito
Despliegue				
Modelo de Despliegue	26 - 29	01-03-11	26-03-11	26-03-11

Tabla. 3.13: Plan de actividades de la Fase de Transición

Fuente: [Elaboración Propia]

3.5.2. DESPLIEGUE DEL SISTEMA

En esta Fase se desarrollará la verificación de la implementación del sistema y se documentara en el diagrama de despliegue.

a) DIAGRAMA DE DESPLIEGUE

En el diagrama de despliegue se sitúa el software en el hardware que lo contiene, este diagrama indica la situación física de los componentes lógicos desarrollados (ver fig. 3.24).

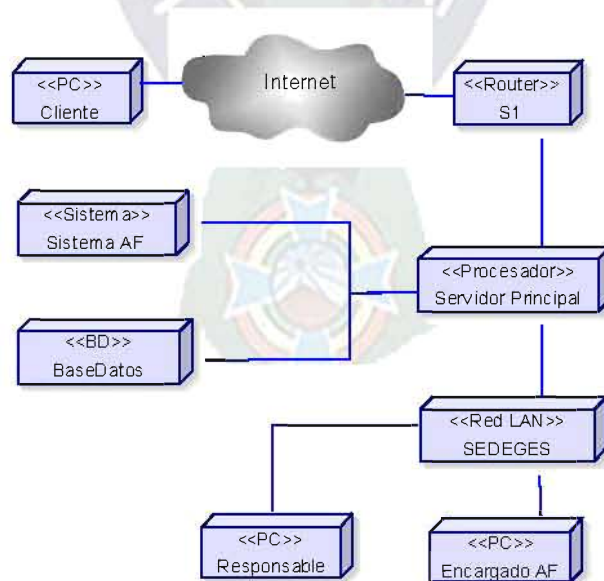


Fig. 3.24: Diagrama de Despliegue del Sistema.

Fuente: [Elaboración Propia]

Para el correcto despliegue del Sistema de Control y Seguimiento de Activos Fijos para la empresa Autoventa Intercambio se necesita realizar la siguiente especificación de Hardware y Software.

Especificación de Hardware.-

Requerimientos mínimos de hardware para el funcionamiento del Sistema.

Para el Servidor:

- HP ProLiant ML350 G4p, Dual Option SCSI – Tower Server.
- Procesador: Intel Xeon 3.40 GHz/800 MHz, 2M (up to 2 supported).
- Disco Duro: 160 GB Pluggable Ultra320 SCSI 15.000 rpm.
- RAM: 1 GB REG PC2-3200 (1x512).
- UPS Sistema de alimentación ininterrumpida M/Tripplete.
- Monitor SVGA.
- Teclado.
- Mouse.

Para las Terminales:

- CPU Pentium III (1.4 GHz/333 MHz) o superior.
- RAM: 128 MB.
- Disco Duro: 40 GB.
- Monitor SVGA.
- Teclado.

- Mouse.

Especificación de Software.-

Requerimientos mínimos de software para el funcionamiento del Sistema.

Para el Servidor:

- Sistema Operativo Windows 2000 Server.
- Lenguaje de desarrollo de aplicaciones PHP 5.2.5 para Windows.
- Gestor de Base de Datos MySQL 5.0 para Windows.
- Servidor de Páginas http Apache 2.2 para Windows.
- Internet Explorer 6.0
- FlashPlayer 6.0

Para las Terminales:

- Sistema Operativo Windows XP ServicePack 2.
- Internet Explorer 6.0
- FlashPlayer 6.0

3.5.3. EVALUACIÓN DE LA FASE DE TRANSICIÓN

En esta Fase de Transición se avanzó satisfactoriamente de acuerdo a lo planificado al inicio de la presente fase, se desarrolló el diagrama de despliegue especificando Hardware y Software requerido para el correcto funcionamiento del sistema.



CAPITULO IV

METRICAS DE CALIDAD

4. CALIDAD DE SOFTWARE

4.1. INTRODUCCIÓN.

Para valorar la calidad de los productos software o los sistemas que se desarrolla, se proporciona la información adecuada sobre los datos necesarios referentes a la calidad del producto, permitiendo una visión profunda sobre el cumplimiento de los objetivos del proyecto.

La ingeniería de software no está basada en leyes cuantitativas básicas, en su lugar, se hace uso de un conjunto de medidas indirectas conocidas como métricas, las cuales proporcionan una referencia de la calidad de algún producto software.

Entre las varias técnicas para medir la calidad de software se encuentra el estándar ISO 9126, que identifica los siguientes atributos clave de calidad para el software:

- Usabilidad.
- Mantenibilidad.
- Funcionalidad.
- Confiabilidad.
- Portabilidad.

4.2. USABILIDAD

Usabilidad es el esfuerzo necesario para aprender a utilizar el sistema. El sistema actual cuenta con una interfaz amigable e intuitiva lo cual hace fácil su utilización.

Se realizaron encuestas a 10 usuarios finales, sobre el manejo de el sistema para medir la usabilidad (ver tabla 4.1).

Pregunta	Respuestas		Porcentaje
	SI	NO	
¿Son Complicadas las respuestas del sistema?	2	8	80%
¿Son complicadas las funciones del sistema?	1	9	90%
¿Los resultados que proporciona el sistema le facilitan su trabajo?	9	1	90%
¿El sistema tiene interfaces dinámicas y agradables a la vista?	10	0	100%
¿Son satisfactorias las respuestas del sistema?	10	0	100%
¿Es difícil aprender a manejar el sistema?	2	8	80%
¿Entiende y controla las peticiones que el sistema solicita?	8	2	80%
¿Utiliza el sistema con facilidad?	9	1	90%
PROMEDIO			88.7%

Tabla. 4.1: Encuesta de Usabilidad del Sistema.

Fuente: [Elaboración Propia]

Por lo tanto se concluye que el sistema tiene un grado de usabilidad del 88.7%.

4.3. MANTENIBILIDAD

Para hallar una métrica de la calidad del mantenimiento del sistema, nos basaremos en el índice de madurez de software (IMS) que nos proporciona una indicación de la estabilidad de un producto de software.

Calculamos el índice de Madurez de software con la siguiente relación:

$$IMS = [M_t - (F_c + F_a + F_e)] / M_t$$

Donde:

M_t : número de módulos total de la versión actual.

F_c : número de módulos de la versión actual que se cambiaron.

F_a : número de módulos de la versión actual que se añadieron.

F_e : número de módulos de la versión anterior que se eliminaron en la versión actual.

Se tiene que el índice de madurez de software es:

$$IMS = [10 - (1 + 1 + 0)] / 10$$

$$IMS = 0.80$$

$$IMS = 80 \%$$

Por lo tanto el sistema empieza a estabilizarse en un 80%.

4.4. FUNCIONALIDAD

Para calcular la medida de funcionalidad denominada punto función cuantificamos el tamaño y la complejidad del sistema en términos de las funciones de usuario, determinando un valor específico para cada característica.

Determinaremos las cinco características del dominio de información:

Entradas de Usuario: Representamos las entradas de control del usuario que proporciona diferentes datos a la aplicación.

Salidas de Usuario: Representa cada salida de información referente a la aplicación (informes, pantallas, mensajes de error).

Consultas de Usuario: Representa a cada combinación única existente de entrada-salida, donde una entrada genera una salida.

Archivos: Representa a cada archivo parte de la base de datos o a un archivo independiente.

Interfaces Externas: Interfaces legibles por el ordenador, que son utilizados para transmitir la información a otro sistema.

La siguiente tabla muestra las cinco características con factor de ponderación medio para el cálculo del punto función.

Parámetro de Medida	Cuenta	Factor de Ponderación			Total
		Simple	Medio	Complejo	
Nº de Entradas de usuario	84	3	4	6	336
Nº de Salidas de usuario	118	4	5	7	590
Nº de Consultas del usuario	81	3	4	6	324
Nº de Archivos	25	7	10	15	250
Nº de Interfaces Externas	3	5	7	10	21
CUENTA TOTAL					1515

Tabla. 4.2: Cuenta Total con Factor de Ponderación Medio.

Fuente: [Elaboración Propia]

Según la importancia de las características ambientales del sistema calculamos en función a esta los valores de ajuste de complejidad F_i ($i = 1, \dots, 14$), necesarios para el cálculo del punto función (ver Tabla 4.3).

<i>i</i>	Características del Sistema	Valor
1	¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	4
2	¿Se ha diseñado el código para ser reutilizable?	4
3	¿Existen funciones de procesamiento distribuido?	0
4	¿Se ejecutara el sistema en un entorno operativo existente o fuertemente utilizado?	3
5	¿Requiere el sistema entrada de datos interactivos?	4
6	¿Se requiere que la entrada de datos interactivo, que las transacciones de entrada de datos se lleven acabo sobre múltiples pantallas y operaciones?	3
7	¿Es crítico el rendimiento?	3
8	¿Se utilizan los archivos maestros de forma interactiva?	4
9	¿Son complejas las entradas, salidas, archivos o las peticiones?	4
10	¿Es complejo el procesamiento interno?	5
11	¿Se requiere comunicación de datos?	5
12	¿Están incluidas en el diseño la conversión y la instalación?	3
13	¿Requiere el sistema copias de seguridad?	5
14	¿Se han diseñado las aplicaciones para facilitar los cambios y para ser fácilmente utilizada por el usuario?	5
FACTOR DE AJUSTE DE COMPLEJIDAD ($\sum F_i$)		52

Tabla. 4.3: Valores de ajuste de complejidad.

Fuente: [Elaboración Propia]

Calculamos el Punto Función con la siguiente relación:

$$PF = \text{CuentaTotal} * (0.65 + 0.01 * \sum Fi)$$

$$PF = 1515 * (0.65 + 0.01 * 52)$$

$$PF = 1773$$

Hallamos el Punto Función máximo para comparar los valores de funcionalidad del sistema:

$$PF_{MAX} = 1515 * (0.65 + 0.01 * 70)$$

$$PF_{MAX} = 2045$$

La Funcionalidad real es:

$$\text{FUNCIONALIDAD} = \frac{1773}{2045} * 100\% = 87\%$$

Por lo tanto el sistema tiene una capacidad del 87% para proporcionar funcionalidades que satisfagan las necesidades específicas.

4.5. CONFIABILIDAD

La métrica de Confiabilidad se tiene encontrando la probabilidad de operación libre de fallos de un programa en un entorno determinado y durante un tiempo específico.

Observamos el trabajo hasta que se produzca un fallo en el instante t , hallamos la probabilidad de falla con una variable aleatoria continua T , en una función exponencial. La relación es la siguiente:

$$\text{Probabilidad de hallar una falla: } P(T \leq t) = F(t)$$

$$\text{Probabilidad de no hallar una falla: } P(T > t) = 1 - F(t)$$

$$\text{Con } F(t) = F_c * (e^{-\lambda/6 * 12})$$

Donde :

$F_c=0.87$: Funcionalidad del sistema.

$\lambda=1$: Tasa de fallos en 6 ejecuciones dentro de un mes.

Se realiza el cálculo para la confiabilidad durante los próximos 12 meses:

$$F(t) = F_c * (e^{-\lambda/6 * 12})$$

$$F(t) = 0.87 * (e^{-1/6 * 12})$$

$$F(t) = 0.135$$

La Probabilidad de Hallar una falla es de un 15% durante los próximos 12 meses.

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

$$P(T > t) = 1 - 0.135$$

$$P(T > t) = 0.864$$

La Probabilidad de No hallar una falla es de un 86.4% durante los próximos 12 meses, lo cual es una probabilidad aceptable y confiable para la utilización del sistema.

4.6. PORTABILIDAD

La métrica de Portabilidad es cualificable por lo que representa el esfuerzo necesario para transferir el programa de un entorno de sistema hardware y/o software a otro. La Portabilidad del sistema se presenta cuando se cumple en los siguientes niveles:

a) Nivel Software

El Instalador del Sistema se puede distribuir en medios magnéticos, el software PHP, MySQL, Apache, se puede instalar bajo los sistemas operativos Windows, o Linux, por lo que el Producto Software es fácilmente portable y gracias a PhpMyAdmin se puede migrar la base de datos fácilmente a Latex, OpenDocument, PDF, SQL, XML, Access, Excel, Word,.

b) Nivel Hardware

El Sistema es portable cuando se cumpla los siguientes requerimientos mínimos de hardware:

- CPU Pentium III (1.4 GHz/333 MHz) o superior.
- RAM: 128 MB.
- Disco Duro: 2 GB.
- Monitor SVGA.
- Lector CD-ROM.
- Teclado.
- Mouse.



CAPITULO V

EVALUACION DE COSTOS Y BENEFICIOS

5. EVALUACION DE COSTOS Y BENEFICIOS

5.1. INTRODUCCIÓN.

Su principal propósito es realizar una evaluación de todo el proyecto además de analizar los riesgos y costos que implicarán su desarrollo. El estudio de factibilidad del Proyecto en el caso del Negocio proporciona la información necesaria, desde el punto de vista del negocio.

5.2 ESTIMACIONES DEL PROYECTO, MODELO COCOMO

5.2.1 ESTIMACIÓN DEL TAMAÑO DEL SOFTWARE

Contando con la experiencia y la información de proyectos históricos y considerando el número de componentes estándar, como son los módulos del sistema (reportes, pantallas, etc.) se puede estimar el tamaño del software midiendo por líneas de código (LDC), para cada subsistema o módulo del proyecto, a los cuales se detallan a continuación:

- Seguimiento de la Renta de Activos.
- Gestión del Activos Fijos (Registro, bajas, revalorización).
- Asignar Responsables a Activos.
- Transferencias de Activos Fijos.
- Seguimiento del mantenimiento de Activos Fijos.
- Inventarios.
- Reportes de Activos asignados y no asignados.
- Reportes de Activos rentados.
- Reportes por Grupo y/o Auxiliar contable.

- Reportes por Responsables y/o Ubicación.

El rango de estimaciones de LCD de la función de análisis geométrico de tres dimensiones, según la técnica de estimación de tres puntos de LDC es:

MÓDULO	RANGO DE ESTIMACIÓN [LDC]		
	OPTIMISTA (de 10 a 300)	MAS PROBABLE (de 301 a 750)	PESIMISTA (de 751 a 1000)
Seguimiento de la Renta de Activos	300	750	950
Gestión de Activos	300	750	950
Asignar Responsables	230	680	800
Transferencias	200	500	800
Mantenimiento	230	680	800
Inventarios	200	500	780
Reportes de Asignados y no Asignados	200	500	800
Reportes de Activos rentados.	200	600	780
Reportes por Grupo y/o Auxiliar	200	500	780
Reportes por Responsable y/o Ubicación	200	500	780

Tabla. 5.1: Tabla del rango de Estimación.

Fuente: [Elaboración Propia]

Se calcula el valor esperado para cada uno de los módulos del sistema, aplicando la siguiente ecuación:

$$VE = (S_{opt} + 4S_m + S_{pess}) / 6$$

Donde:

VE : Valor esperado de las variables de estimación.

S_{opt} : Estimación optimista.

S_m : Estimación más probable.

S_{pess} : Estimación pesimista.

Los valores esperados por módulo se presentan en la Tabla 5.2, reflejando un total de 5720 líneas de código para implementar el sistema de Control y Seguimiento de Activos Fijos para la empresa Autoventa Intercambios.

MÓDULO	LDC ESTIMADA
Seguimiento de la Renta de Activos	708
Gestión de Activos	708
Asignar Responsables	625
Transferencias	500
Mantenimiento	625
Inventarios	497
Reportes de Activos Asignados y no Asignados	500
Reportes de Activos rentados	563
Reportes por Grupo y/o Auxiliar	497
Reportes por Responsable y/o Ubicación	497
TOTAL	5720

Tabla. 5.2: Tabla de estimación del método LDC.

Fuente: [Elaboración Propia]

5.2.2 ANÁLISIS DE COSTOS

El costo que implica la implementación del sistema de Control y Seguimiento de Activos Fijos para la empresa Autoventa Intercambios esta previsto dentro de los gastos a largo plazo de la misma empresa, contando así con un presupuesto establecido, ajustamos nuestro desarrollo al presupuesto ya establecido.

5.2.2.1 COSTO DE LA FASE DE INICIO.

El tiempo en el cual se realizara el estudio de factibilidad, modelado de negocio, captura de requerimientos, desarrollados en la fase de Inicio será estimado llevarse a cabo en un plazo de 4 semanas con un salario mensual de 5000 Bs.

5.2.2.2 COSTO DE LA FASE DE ELABORACIÓN.

El tiempo que representa realizar el análisis y diseño del sistema, que se desarrolla en la fase de Elaboración será estimado a llevar a cabo en el lapso de 8 Semanas con un salario mensual de 5000 Bs. El costo total en esta fase alcanza a los 10000 Bs.

5.2.2.3 COSTO DE LA FASE DE CONSTRUCCIÓN.

Siguiendo el Modelo de Costos de Construcción, teniendo en cuenta que el software a desarrollarse es un tipo de software orgánico, con un entorno estable y contando con la experiencia en proyectos de similar magnitud, se sabe que la productividad promedio es de

94 LDC por día y considerando la estimación de líneas de código LDC obtenida en la Tabla 5.2, se calculó por este modelo de estimación que:

$$\begin{aligned}\text{Esfuerzo :} \quad E &= 2.4 (\text{KLDC})^{1.05} \\ E &= 2.4 (5.720)^{1.05} \\ E &= 14.98 \text{ [p/m]} \approx 15 \text{ [p/m]}\end{aligned}$$

$$\begin{aligned}\text{Duración:} \quad D &= 2.5 E^{0.38} \\ D &= 2.5 (15)^{0.38} \\ D &= 6.99 \text{ [meses]} \approx 7 \text{ [meses]}\end{aligned}$$

Con estos cálculos, estimamos 1 mes para la fase de Inicio, 2 meses para la fase de Elaboración, 3 meses para la fase de Construcción y 1 mes para la fase de Transición, estos 3 meses de la fase de construcción nos representan un costo de 15000 Bs.

5.2.2.4 COSTO DE LA FASE DE TRANSICIÓN

En la fase de Transición es donde se realiza las capacitaciones a los usuarios y la instalación del producto final, el tiempo estimado para este trabajo es de 1 mes con un costo de 5000 Bs.

Puesto que se esta usando software libre y se cuenta con los equipos de computación requeridos, (servidor, computadoras terminales, equipo de redes) para desarrollar e implementar el presente proyecto, no se tuvo que realizar otro gasto extra, se cuenta también con un dominio propio en Internet.

La siguiente tabla resume los costos para el desarrollo e implementación del sistema de Control y Seguimiento de Activos Fijos para la Empresa Autoventa Intercambio.

DESCRIPCION	TIEMPO [semanas]	COSTO [Bs.]
Fase de Inicio	4	5000
Fase de Elaboración	8	10000
Fase de Construcción	13	15000
Fase de Transición	4	5000
Costo de Software y herramientas adicionales	-	0
Costo de Hardware	-	0
TOTAL		35000

Tabla. 5.3: Tabla de Descripción de los costos de desarrollo del sistema.

Fuente: [Elaboración Propia]

5.2.3 ANÁLISIS DE RIESGOS

5.2.3.1 Proyección del Riesgo

Los riesgos del proyecto se presentan en la siguiente tabla, con los valores de impacto que se divide en cuatro categorías: 1(catastrófico), 2(crítico), 3(marginal), 4 (despreciable).

RIESGOS	PROBABILIDAD	IMPACTO
Cambio de requisitos por parte del Usuario	80%	2
Resistencia de los usuarios al sistema	70%	2
Falta de capacitación al Usuario	40%	3
Cambios de Personal	30%	4
Falta de experiencia con las herramientas de desarrollo	60%	2
La tecnología no alcanza las expectativas	60%	2
Interacción de Software con Hardware no probada	50%	2
Personal del Proyecto sin experiencia	40%	3

Tabla. 5.4: Tabla de Riesgos del Sistema.

Fuente: [Elaboración Propia]



CAPITULO VI

SEGURIDAD DEL SISTEMA

6. SEGURIDAD DEL SISTEMA

6.1. INTRODUCCIÓN.

Su principal propósito es documentar la seguridad del sistema basado en el UNE-ISO / IEC 17779:2002 (Códigos de buenas prácticas de seguridad para la gestión de la Información), en cuyos principales puntos se encuentran: Confidencialidad, Integridad, Disponibilidad, Autenticidad y Auditabilidad.

6.2. CONFIDENCIALIDAD.

Para la confidencialidad de la información se definen Permisos a los menús del sistema, teniendo cada usuario diferentes permisos, según el perfil de Acceso que tenga, por ejemplo un Usuario con perfil de acceso de Responsable de AF solo podrá ver los AF asignados a su persona.

6.3. INTEGRIDAD.

Para la Integridad se han definido Perfiles de Acceso con roles específicos para cada nivel de Usuario, garantizando que solo el personal autorizado a un cierto menú tenga Acceso a modificar la información, por ejemplo un Usuario con perfil de acceso de Adm. de AF es el único que podría modificar la información de algún AF.

Si se tratara de modificar la información directamente a la base de datos, no se tendría generado el registro de historial de transacciones respectivo, con lo que se podría verificar la integridad de los datos.

6.4. DISPONIBILIDAD.

Para asegurar la disponibilidad del sistema se implanta el uso de variables de sesión, permitiendo el acceso a la información autorizada sin restricciones, una vez realizada la autenticidad del usuario, sin embargo, una vez que no se realiza ninguna transacción durante un determinado periodo de tiempo, se niega el uso del sistema.

Se propone realizar backups mensualmente para evitar la pérdida fortuita de información, como también el uso necesario del UPS, asegurando la disponibilidad al sistema.

6.5. AUTENTICIDAD.

Este atributo nos permite asegurar el origen de la Información, es decir validar la identidad del Usuario Emisor, verificando su login y un password que se registra y almacena encriptado en la base de datos, para luego darle acceso a los menús permitidos.

6.6. AUDITABILIDAD.

Este atributo nos permite determinar que acciones o transacciones se han ejecutado en el sistema, quien lo ha ejecutado, cuando se ha llevado a cabo, y cual es el activo afectado. Para esto se lleva un registro de las acciones ejecutadas sobre activos fijos, almacenados en el Historial de Transacciones.



CAPITULO VII CONCLUSIONES

7. CONCLUSIONES

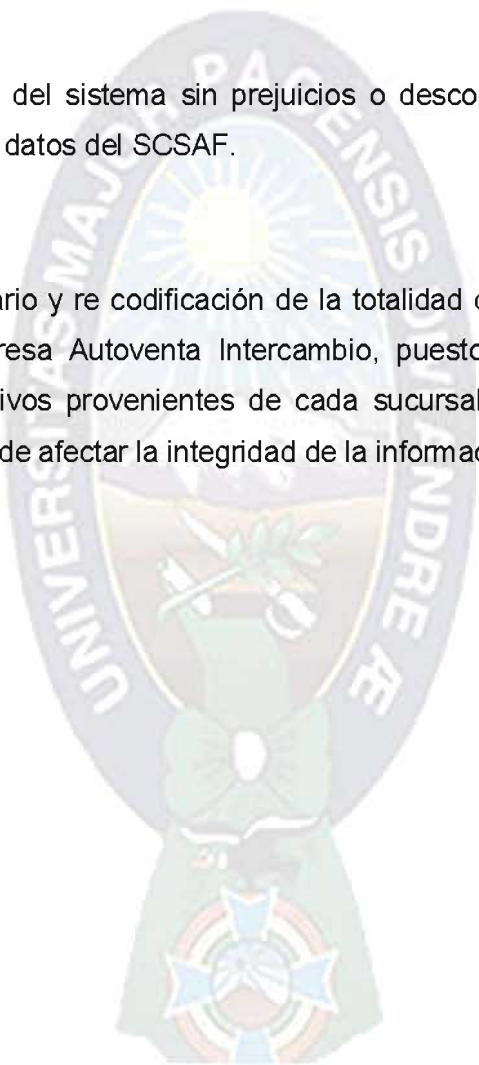
- El sistema de Gestión y Control de Activos Fijos para la empresa Autoventa Intercambio, llego a su conclusión de una manera satisfactoria, cumpliendo con todos los requisitos especificados en la fase de Inicio, cumpliendo su objetivo principal.
- La Implementación del sistema SIGSAF mejora el desempeño de los funcionarios de la Sección de Activos Fijos de la empresa Autoventa, reduciendo la carga de trabajo de los mismo, como también mejora la administración de Activos Fijos que se asigna a clientes ofreciendo confiabilidad en el manejo de la información correspondiente a procesos de gestión de Activos Fijos.
- Al finalizar el desarrollo del producto software, se logra implementar la seguridad para garantizar el manejo de la Información en cuanto a Confiabilidad, Integridad, Disponibilidad, Autenticidad y Auditabilidad se refiere.
- Se logra darle escalabilidad al sistema gracias al manejo de Permisos a Perfiles de Acceso y a la parametrización de Depreciaciones, Grupos contables, Auxiliares Contables, Partidas Presupuestarias, Fuentes de Financiamiento, Organismos Financiadores, Unidades y Cargos. Con la posibilidad de crecimiento o cambio de acuerdo a normas de auditoria del Ministerio de Hacienda y Ministerio de Trabajo.



CAPITULO VIII RECOMENDACIONES

8. RECOMENDACIONES

- Se recomienda el uso del sistema también en las sucursales que tiene la empresa AutoVenta Intercambio, Implementando el Sistema en la sucursal de El Alto, y la sucursal de la Ciudad de Santa Cruz, integrándolos a través de Internet.
- Concienciar el uso del sistema sin prejuicios o desconfianza sobre la integridad y confiabilidad de los datos del SCSAF.
- Realizar un Inventario y re codificación de la totalidad de los activos fijos que ahora administra la empresa Autoventa Intercambio, puesto que existe varios tipos de codificación de activos provenientes de cada sucursal con la que cuenta la dicha empresa y que puede afectar la integridad de la información.





BIBLIOGRAFIA

REFERENCIA BIBLIOGRÁFICA

- [BID, 2005]** BID (Banco Interamericano de Desarrollo)
Curso básico de Marco Lógico para el diseño y la conceptualización de proyectos.
- [JACOBSON, 2000]** IVAR JACOBSON, GRADY BOOCH, JAMES RUMBAUGH
El Proceso Unificado De Desarrollo De Software
Editorial ADDISIN WESLEY.
- [ALARCON, 2000]** RAUL ALARCON
Diseño Orientado a objetos con UML
Grupo EIDOS, Consultoría y Documentación Informática.
- [MARTINEZ, 2006]** ALFONZO MARTINEZ MARTINEZ
Rational Unified Process.
Universidad Autónoma Metropolitana.
- [LARMAN, 1999]** CRAIG LARMAN
UML y PATRONES (Introducción al Análisis y Diseño Orientado a Objetos)
Editorial Prentice Hall.

[SCHMULLER, 2003]

JOSEPH SCHMULLER

Aprendiendo UML en 24 Horas.

Editorial Prentice Hall.

[BERZAL, 2004]

FERNANDO BERZAL

Unified Modeling Language.

Curso Programación Orientada a Objetos

REFERENCIA WEB

[AMO, 2006]

ALONSO AMO.

Metodología RUP.

<http://s.fi.upm.es/mdp/si/metodorup.pdf>

Fecha de ingreso 20/06/06

[DANIELE, 2007]

MARCELA DANIELE.

Modelo de Construcción de Costos (COCOMO)

<http://pisuerga.inf.ubu.es/si/Docencia/TFC/ITIG/cruzadn/Memoria/index.htm>

Fecha de ingreso 27/08/2007

[IMAGEAN, 2008]

IMAGEAN DART.

Diseño de Interfaces.

http://www.imageandart.com/tutoriales/web_design/disenio_inter_faces/index.html

Fecha de ingreso 09/09/2008.





ANEXOS



ANEXO A: DESARROLLO DE MARCO LÓGICO

MATRIZ DE PROBLEMAS

ARBOL DE PROBLEMAS

ARBOL DE OBJETIVOS

MATRIZ DE MARCO LOGICO

MATRIZ DE PROBLEMAS.

	PROBLEMA	CAUSA	EFFECTO	INVOLUCRADO
P1	Carencia de un sistema de codificación automático.	La clasificación para la codificación de los activos fijos es manual	Pérdida de tiempo en la codificación de activos fijos.	Sección de Activos Fijos.
P2	No se cuenta con registro histórico de los activos que han sido revalorizados o transferidos.	No existen algunos datos necesarios en los formularios pre impresos que se usan.	No existen documentos legales de respaldo confiables.	Sección de Activos Fijos, funcionarios.
P3	Errores de cálculos en la depreciación del activo fijo.	El cálculo de depreciación manualmente trae errores de redondeo, y existe errores de cálculo por día, en el sistema actual.	Los reportes contables no son exactos ni confiables.	Unidad Administrativa y financiera. Sección de Activos Fijos.
P4	No se cuenta con todos los requerimientos, normativas de manejo, y seguridad de los Activos Fijos, como especifica la norma Básica de Administración de Bienes y Servicios.	Proporciona información limitada y no cumple con todos los requerimientos, ni normativas de manejo.	Existe una demora en la elaboración de reportes el mismo que tiene información incompleta.	Unidad Administrativa y financiera. Sección de Activos Fijos.
P5	Aplicaciones en FOX PRO no ofrecen entorno visual y amigable.	El sistema desarrollado en FOX PRO, el mismo que responde a limitadas tareas.	No responde a las nuevas necesidades del personal y a la exigencia de las normas básicas.	Sección de Activos Fijos, funcionarios.
P6	Demora en la elaboración de reportes e informes para proporcionar a la toma de decisiones de los departamentos vinculados.	Los reportes actuales no son detallados.	Los funcionarios fácilmente no precisan responsabilidades.	Sección de Activos Fijos, funcionarios.
P7	No se cuenta con información detallada del activo fijo.	No se elabora el procedimiento manual de seguimiento de activos fijos, por falta de disponibilidad de tiempo.	No existe informes de apoyo para la toma de decisiones en la unidad Administrativa	Unidad Administrativa y financiera. Sección de Activos Fijos.

MARCO LOGICO

	RESUMEN NARRATIVO	INDICADORES	MEDIOS DE VERIFICACION	SUPUESTOS
FIN	Proporcionar un Sistema de Control y seguimiento de Activos Fijos.	La eficiencia en los procesos de control y seguimiento de activos fijos se incrementa visiblemente.	Verificable mediante el registro de pruebas de resultados en archivos, formularios obtenidos en el sistema.	-Se tiene un conocimiento de los procesos del control y seguimiento de activos fijos.
PROPOSITO	Diseñar e implementar un sistema informático que mejore el control y seguimiento de Activos Fijos para la empresa Automóviles Intercambios en base a la Normas Básicas del Sistema de Administración de Bienes y Servicios.	-Contar con un sistema Integrado de Activos Fijos, a partir de julio 2011. -Contar con reportes desde que se encuentre el sistema en vigencia. -Reducción del tiempo de entrega de reportes, actas y formularios en un 50%.	Software desarrollado y funcionando.	-Interés político. -Presupuesto.
PRODUCTO	-Desarrollar un modulo de registro de activos fijos, bajas, modificaciones, revalorizaciones, transferencias, con la respectiva generación de documentos de respaldo si corresponde. -Elaborar un algoritmo para obtener el valor neto de activos fijos, mejorando el cálculo de las depreciaciones, aplicando el método de la line recta. -Elaborar un modulo de asignación y liberación de responsable, que genere el acta de entrega y recepción de activos fijos. -Desarrollar un modulo para generar informes que proporcione datos exactos del historial de transacciones de cada activo fijo (registro, bajas, modificaciones, transferencias, revaluó, liberaciones, mantenimiento).	-Un prototipo de Sistema de Control de Activos Fijos, desarrollado bajo la metodología RUP. -Un Manual de usuario elaborado según el sistema puesto en funcionamiento desde JULIO 2011.	-Reportes impresos de control y seguimiento de activos fijos. -Software presentado. -Discos de instalación. -Manual de Usuario presentado.	-Presupuesto.
ACTIVIDADES	-Estudio y recopilación de información del manejo de activos fijos. -Entrevista con el personal de la sección de activos fijos. -Analizar y diseñar el sistema informático de automatización para el control y seguimientos de activos fijos. -Codificación del sistema. -Elaboración del Manual. -Pruebas e Implementación del Sistema.	-Entrevista con el personal toma un tiempo de 2 a 3 veces por semana.	-Informes de correcciones y funcionamiento. -Documentación recolectada. -Documentos del proyecto y programa ejecutable.	-Acceso a toda la documentación necesaria a todo el estudio. -Contar con los medios necesarios para la realización del análisis y diseño del sistema. -Contar con el conocimiento necesario sobre el lenguaje de programación para la codificación del sistema. -Contar con el Software y hardware necesarios para la implementación del sistema.

