



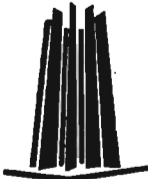
UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

**FACULTAD DE ESTUDIOS SUPERIORES
ARAGÓN**

**“SISTEMA DE FACTURACIÓN PARA
PROVEEDORES DE SERVICIOS DE INTERNET”**

**T R A B A J O E S C R I T O
EN LA MODALIDAD DE SEMINARIOS
Y CURSOS DE ACTUALIZACIÓN Y
CAPACITACIÓN PROFESIONAL
QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN
P R E S E N T A :
V I C T O R N O E C R U Z
P I N E D A**

ASESOR: M. en C. MARCELO PÉREZ MEDEL



MÉXICO, 2005.

0350971



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

E-BILLING
SISTEMA DE FACTURACIÓN PARA PROVEEDORES DE SERVICIOS DE INTERNET

DIPLOMADO PARA DESARROLLO DE SISTEMAS WEB

TABLA DE CONTENIDO

INTRODUCCIÓN.....	5
▪ Diplomado “Desarrollo de Sistemas Web”.....	5
▪ Caso práctico “E-BILLING”.....	7
CAPÍTULO I PROPUESTA.....	8
1.1 Presentación.....	9
▪ 1.1.1 Nombre del Proyecto.....	9
▪ 1.1.2 Descripción del Proyecto.....	9
▪ 1.1.3 Resumen.....	9
▪ 1.1.4 Antecedentes.....	10
▪ 1.1.5 Objetivos.....	11
▪ 1.1.6 Alcance del Proyecto.....	11
▪ 1.1.7 Entregables.....	11
1.2 Análisis de Información.....	12
▪ 1.2.1 Descripción del Producto.....	12
▪ 1.2.2 Descripción del Proceso.....	13
▪ 1.2.3 Organización.....	14
▪ 1.2.4 Sistema Informático Actual.....	15
1.3 Diagnóstico.....	16
▪ 1.3.1 Producto.....	16
▪ 1.3.2 Proceso.....	16
▪ 1.3.3 Sistema Informático Actual.....	16
▪ 1.3.4 Organización.....	16
1.4 Estrategia de solución.....	17
1.5 Plan detallado de Actividades.....	17
1.6 Riesgos y Beneficios.....	17
▪ 1.6.1 Riesgos.....	17
▪ 1.6.2 Beneficios.....	17
CAPÍTULO II ESPECIFICACIÓN DE REQUERIMIENTOS.....	18
2.1 Introducción.....	19
▪ 2.1.1 Reseña del Producto.....	19
2.2 Especificaciones funcionales.....	19
▪ 2.2.1 Necesidades acordes al Usuario.....	19

▪ 2.2.2 Actores	20
▪ 2.2.3 Diagrama de Clases	21
▪ 2.2.4 Diagrama de Casos de Uso	23
▪ 2.2.5 Descripción de Casos de Uso	24
2.3 Requerimientos no funcionales	26
▪ 2.3.1 Uso del Sistema	26
▪ 2.3.2 Seguridad	26
▪ 2.3.3 Disponibilidad	26
▪ 2.3.4 Mantenimiento y Crecimiento	26
2.5 Requerimientos de medio ambiente (Desarrollo e Implementación)	27
▪ 2.5.1 Hardware	27
▪ 2.5.2 Software	27
CAPÍTULO III MODELO DE ANÁLISIS	28
3.1 Introducción	29
3.2 Objetivo	29
3.3 Conceptos Básicos	29
▪ 3.3.1 Diagramas de Interacción	29
▪ 3.3.2 Diagramas de Secuencia	29
▪ 3.3.3 Diagramas de Colaboración	30
3.4 Desarrollo	31
▪ 3.4.1 Diagramas de Secuencia	31
▪ 3.4.2 Construcción de los Diagramas de Secuencia del Sistema	32
▪ 3.4.3 Construcción del Diagrama de Colaboración	35
3.5 Conclusiones	36
CAPÍTULO IV MODELO DE DISEÑO	37
4.1 Definiciones	38
4.2 Perspectiva de un proceso unificado	38
4.3 Objetivos	39
4.4 Desarrollo	39
▪ 4.4.1 Diseño - Implementación	40
▪ 4.4.2 Componentes	41
▪ 4.4.3 Clases de Diseño	42
▪ 4.4.4 Realización de caso de uso - diseño	43
▪ 4.4.5 Interfaces	43

▪ 4.4.6 Flujo de trabajo de diseño	44
▪ 4.4.7 Flujo de trabajo general.....	45
▪ 4.4.8 Diseño de Componentes	46
▪ 4.4.9 Diseño de Base de Datos.....	47
CAPÍTULO V MODELO DE IMPLEMENTACIÓN	48
5.1 Introducción	49
CAPÍTULO VI MODELO DE PRUEBAS	50
6.1 Introducción	51
6.2 Objetivos	53
6.3 Descripción de la propuesta	54
6.4 Generación de casos de prueba a partir de casos de uso	54
▪ 6.4.1 Generación de escenarios de uso	55
▪ 6.4.2 Identificación de casos de prueba	58
▪ 6.4.3 Identificación de valores a probar.....	61
6.5 Conclusiones modelo de pruebas.....	65
BIBLIOGRAFÍA.....	66
URL.....	66

INTRODUCCIÓN

▪ Diplomado “Desarrollo de Sistemas Web”

El presente trabajo fue elaborado dentro del diplomado “Desarrollo de Sistemas Web”. A través de los seis módulos impartidos se adquirieron los conocimientos necesarios para desarrollar aplicaciones WEB, utilizando las mejores prácticas y estándares.

A continuación se describe un breve resumen de los módulos estudiados y la manera en la que se aplicó al caso práctico:

1. Fundamentos para el desarrollo orientado al WWW

Una introducción a Internet, historia del DNS, introducción al modelo de tres capas, UNIX y KDE (Entorno que facilita el manejo del sistema operativo UNIX),

Se determinaron los elementos necesarios para un proyecto de desarrollo WWW

Sistema operativo	Red Hat Linux
Servidor de bases de datos	MySQL
Servidor de aplicación	JBOSS o Jakarta Tomcat
Servidor FTP	Proftpd
Servidor de administración remota	SSH
Maquina virtual Java	Sun Java SDK
Modelador entidad relacion	Case Studio
Entorno de desarrollo	Eclipse
Entorno de diseño html	Dream Weaver

2. Análisis y diseño con UML

Una introducción al análisis y diseño de software.

- Requisitos: Tiene como meta definir y delimitar la funcionalidad del sistema software
- Análisis: Se desarrolla la estructura lógica del sistema la cual debe ser estable, robusta, mantenible y extensible
- Diseño: Tiene como propósito extender la arquitectura general de análisis. En esta etapa se definen las decisiones estratégicas sobre como se organiza la funcionalidad del sistema en torno al ambiente de implementación
- Implementación: Toma el resultado del modelo de diseño para generar el código fuente
- Integración: Permite mantener la modularidad en los subsistemas, esto significa que inicialmente los subsistemas se desarrollan de manera independiente, llegando el momento para su integración final.
- Pruebas: es el responsable de revisar la calidad del sistema siendo desarrollado
- Documentación: Existen diferentes tipos de documentos que deben ser generados como apoyo al sistema. Cada uno de estos documentos tiene

diferentes objetivos y está dirigidos a distintos tipos de personas, desde los usuarios no técnicos hasta los desarrolladores más técnicos. Los siguientes son algunos de los documentos o manuales más importantes:

- i. Manual del Usuario, que le permite a un usuario comprender como utilizar el sistema.
 - ii. Manual del Programador, que le permite a un desarrollador entender los aspectos de diseño considerados durante su implementación.
 - iii. Manual del Operador, que le permite al encargado de operar el sistema comprender que pasos debe llevar a cabo para que el sistema funcione bajo cierta configuración de ambiente particular.
 - iv. Manual del Administrador, que le permite al encargado de administrar el sistema comprender aspectos más generales como son los modelos de requisitos y análisis.
- Mantenimiento: se basa en generar nuevos desarrollos pero tomando como punto de partida el sistema ya existente
 - Métodos y Metodologías: UML

3. Introducción a la programación orientada a objetos con JAVA

Se aprendieron los conceptos básicos de la programación orientada a objetos como:

- Objeto
- Atributo
- Clase
- Diagrama de Objetos, Diagrama de Clases
- Operaciones
- Accesos
- Métodos
- Polimorfismo
- Asociación e Implementación con JAVA

4. Capa de usuario

Se refiere a la capa de presentación basándose en el modelo de tres capas para aplicaciones WEB y dentro de sus principales objetivos está el administrar las peticiones y respuestas del usuario, proveer y delegar las llamadas a la capa de negocio y la validación de datos de entrada mediante la interfaz de usuario

5. Capa de Datos

Se introdujo a SQL (lenguaje de consulta estructurado) es un lenguaje de base de datos normalizado, utilizado por los diferentes motores de bases de datos para realizar determinadas operaciones sobre los datos o estructuras de los mismos

Se comprendió el modelo y estructura de datos relacional, así como las reglas de integridad. Finalmente, JDBC que es la API que permite una conectividad independiente entre el lenguaje de programación JAVA y una gran variedad de bases de datos

Las principales características de JDBC:

- Establece conexión con la base de datos y acceso tabular a la fuente de datos.
- Envío de sentencias SQL
- Proceso de resultado

6. Reglas de Negocio

Se introdujo al patrón MVC (MODEL VIEW CONTROLLER) que es un patrón de diseño que permite dividir a las aplicaciones en tres partes:

- Controlador
- Modelo
- Vistas

Las ventajas de implementar este modelo es la separación total entre lógica de negocio y presentación.

▪ Caso práctico “E-BILLING”

A través del curso del diplomado, se desarrolló un caso práctico denominado “E-BILLING” que es un sistema de información que permite llevar el seguimiento y control de ventas, facturación y cuentas por cobrar para ISPs (Proveedores de Servicios de Internet)

E-BILLING es un sistema que en un principio estaba definido bajo una arquitectura de dos capas (nivel de aplicación y nivel de base de datos) y tenía deficiencias de funcionalidad.

El objetivo de llevar a cabo este caso práctico es aplicar los conocimientos adquiridos para migrar el sistema de una arquitectura de dos capas a una arquitectura de tres capas, haciendo un análisis de requisitos, modelando el diseño e implementando la solución para incrementar la funcionalidad del sistema.

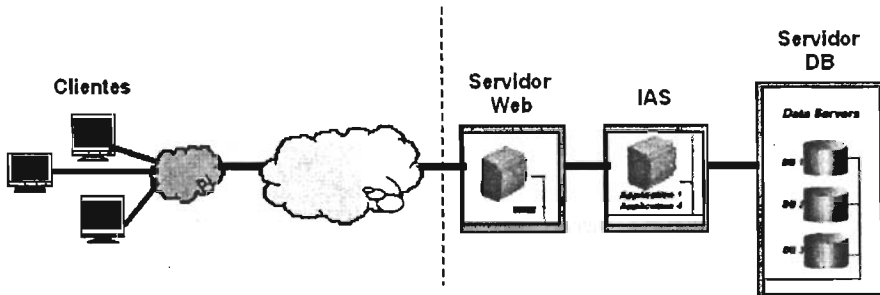


Figura 1 Modelo de tres capas

La figura uno muestra el modelo de tres capas, presentación, reglas de negocio y datos. La capa de presentación se puede definir como la interfaz que el usuario ve cuando ejecuta un programa. La capa de negocio es el pegamento entre la capa de presentación y la capa de datos. En la capa de negocios se realizan todos los procesos que son exclusivos de la empresa.

CAPÍTULO I PROPUESTA

1.1 PRESENTACIÓN

▪ 1.1.1 Nombre del Proyecto

E-BILLING, Sistema de facturación para ISPs (Proveedores de Servicios de Internet)

▪ 1.1.2 Descripción del Proyecto

Análisis, diseño y programación de un sistema de información que permita a los ISPs y a sus clientes dar seguimiento y control de la facturación y cobranza de sus servicios

▪ 1.1.3 Resumen

El presente documento contiene los resultados del estudio realizado por Victor Noé Cruz Pineda al ISP NUESTROSITE.COM para evaluar el impacto de implementar un nuevo sistema de información y desarrollar un óptimo plan de actividades encaminadas a lograr una implementación exitosa

En el diagnóstico se identifican y presentan las causas de los problemas con los que el ISP se enfrenta, diseñando medidas que permitan mejorar su desempeño.

Este documento pretende ser una herramienta para comprender y confirmar la problemática, considerando posibles soluciones que apoyen al cambio y al crecimiento estratégico con dirección.

El diagnóstico se inició con entrevistas en donde cada usuario comentó acerca de las actividades que desempeña y responsabilidades asignadas, proporcionando diagramas, procedimientos, reportes, etc. que utilicé para el desarrollo de sus tareas.

Posteriormente la información proporcionada se ordena y analiza identificando las causas y consecuencias de la problemática, finalmente se formulan propuestas de cambio y recomendaciones

Al finalizar el diagnóstico fue posible crear medidas de control que permitieron solucionar la problemática y mejorar los resultados.

▪ 1.1.4 Antecedentes

PANORAMA GENERAL DE LA COMPAÑÍA

Nuestrosite.com es una empresa mexicana dedicada a proveer soluciones para Internet con cinco años en el mercado se ha especializado en ofrecer servicios como: registro de nombres de dominio, renta de servidores dedicados y compartidos, venta de certificados SSL, desarrollo y diseño de sitios Web.

Actualmente está expandiendo su gama de productos y servicios entre los cuales destacan la consultoría, el desarrollo y la capacitación para la implementación de sistemas de información hechos a la medida y sistemas estándar desarrollados por Nuestrosite.com.

Para la administración, control y seguimiento de sus operaciones Nuestrosite.com cuenta y requiere de la siguiente estructura informática:

Dos Subsistemas que interactúan entre sí (Ya existentes y óptimos):

- 1) **ISP-Config:** Configura de manera automatizada los servidores que ofrecen servicios LDAP, DNS, HTTP, FTP y Correo electrónico.
- 2) **E-Panel:** Panel de control en dónde el cliente administra sus datos y cuentas de correo

Un tercer subsistema **REQUERIDO** (Existente con deficiencias en la funcionalidad) con las siguientes características:

- a. Administración de Datos Generales del Cliente
- b. Facturación de Ordenes de Venta
- c. Aplicación de cobros a facturas
- d. Consulta de facturas y saldos
- e. Rastreo de envíos de facturas

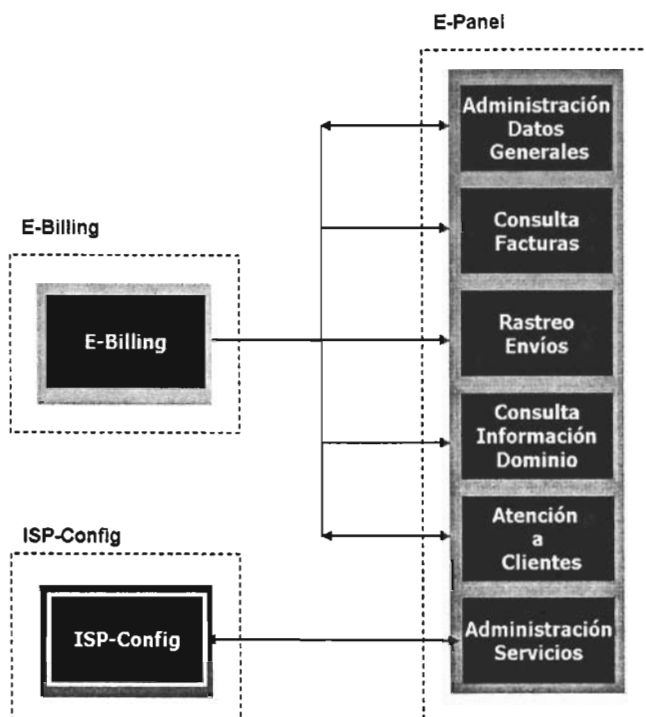


Figura 1.1.4 Funcionamiento operativo sistemático de la organización

▪ 1.1.5 Objetivos

1. Identificar las problemáticas del sistema de facturación que actualmente funciona en NUESTROSITE.COM
2. Hacer un nuevo modelo que resuelva las problemáticas detectadas siguiendo una metodología y estándares
3. Desarrollo del programa del modelo diseñado

▪ 1.1.6 Alcance del Proyecto

Eliminar las deficiencias e incrementar la funcionalidad del sistema E-BILLING

▪ 1.1.7 Entregables

A continuación se indican cada uno de los artefactos que serán generados y utilizados por el proyecto y que constituyen los entregables

1. Especificación de requerimientos
2. Modelo de negocio
3. Aplicación WEB mejorada del Sistema Informático E-BILLING

1.2 ANÁLISIS DE INFORMACIÓN

▪ 1.2.1 Descripción del Producto

Existen básicamente cuatro productos o servicios

1. Hospedaje WEB
2. Registro de Dominios
3. Certificados Digitales
4. Servicios Informáticos

El hospedaje Web consiste en rentarles a los clientes un espacio en un servidor dedicado o compartido con servicios de FTP, Base de Datos, Correo Electrónico y Página WEB.

El registro de dominios consiste en registrar el nombre de un dominio en Internet.

El certificado digital consiste en otorgar una llave criptográfica que permite intercambiar de manera segura información de un servidor a otro.

Los servicios informáticos se refieren a servicios varios que ofrece el ISP como son consultoría especializada, desarrollo de aplicaciones específicas, configuración de hardware, entre otros.

Los productos y servicios que ofrece el ISP a sus clientes tienen características especiales que se muestran a continuación:

1. Periodicidad o vigencia; para el caso del Hospedaje WEB, Registro de Dominios y Certificados Digitales. La problemática que presenta el sistema actual es que únicamente permite facturar los productos por vigencia adelantada debido a un mal diseño de la aplicación y algunos clientes pagan dichos productos una vez vencida su vigencia esto ocasiona un descontrol para el departamento de facturación y cuentas por cobrar.
2. Facturación de productos y servicios; Los clientes pueden adquirir cualquier producto o servicio que ofrece el ISP sin necesidad de tener asignado un dominio específico. Debido al mal diseño del sistema actual no se puede vender ningún producto o servicio si el cliente no tiene asignado un dominio porque la llave para facturar es el nombre del dominio.
3. Asignación de dominios; Un cliente puede adquirir varios dominios y cada dominio ser administrado de manera independiente. En el sistema actual se tiene la problemática de que sólo se permite un dominio por cliente lo que ocasiona que para asignar varios dominios a un cliente se tiene que dar de alta el cliente tantas veces como dominios haya comprado al ISP

1.2.2 Descripción del Proceso

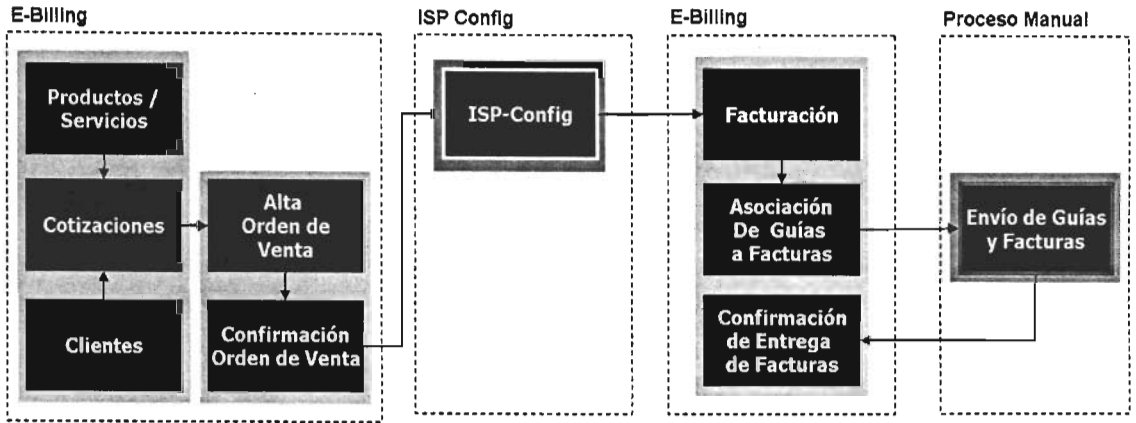


Figura 1.2.2.1 Descripción del Proceso

1. Llama el cliente solicitando un servicio
2. Ventas NS indica al cliente el monto del depósito que debe efectuar
3. El cliente notifica el depósito
4. Ventas NS verifica el depósito
5. Ventas NS verifica que exista la cuenta (dominio) como cliente, si no existe, lo da de alta en E-BILLING y en el ISP-CONFIG
6. El cliente complementa la información de envío y facturación para el dominio en E-PANEL en caso de ser necesario
7. Ventas NS registra los servicios contratados para el dominio en E-BILLING sección Cobros Programados
8. Ventas NS imprime las Facturas a partir de los Cobros Programados en E-BILLING sección Impresión de Facturas.
9. Ventas NS asigna Folios de Facturas a partir de Facturas impresas No Asignadas en E-BILLING sección Asignación de Folios de Factura
10. Ventas NS captura los depósitos de clientes a Facturas en E-BILLING
11. Ventas NS agrupa manualmente las Facturas por cliente y asocia un número de guía diferente por grupo de Facturas.

12. Ventas NS captura el número de guía correspondiente para cada Factura en E-BILLING

13. Ventas NS entrega a Mensajería las Facturas originales con las guías para envío

14. Ventas NS entrega a CXC NS las copias de las Facturas

1.2.3 Organización

La organización en NUESTROSITE.COM presenta los siguientes problemas:

1. El personal tiene asignada una función básica y varias adicionales que no necesariamente están ligadas
2. Existen en algunas áreas vacíos o ambigüedades en la definición de autoridad y responsabilidad
3. La coordinación entre las actividades es confusa y en algunos casos se omite

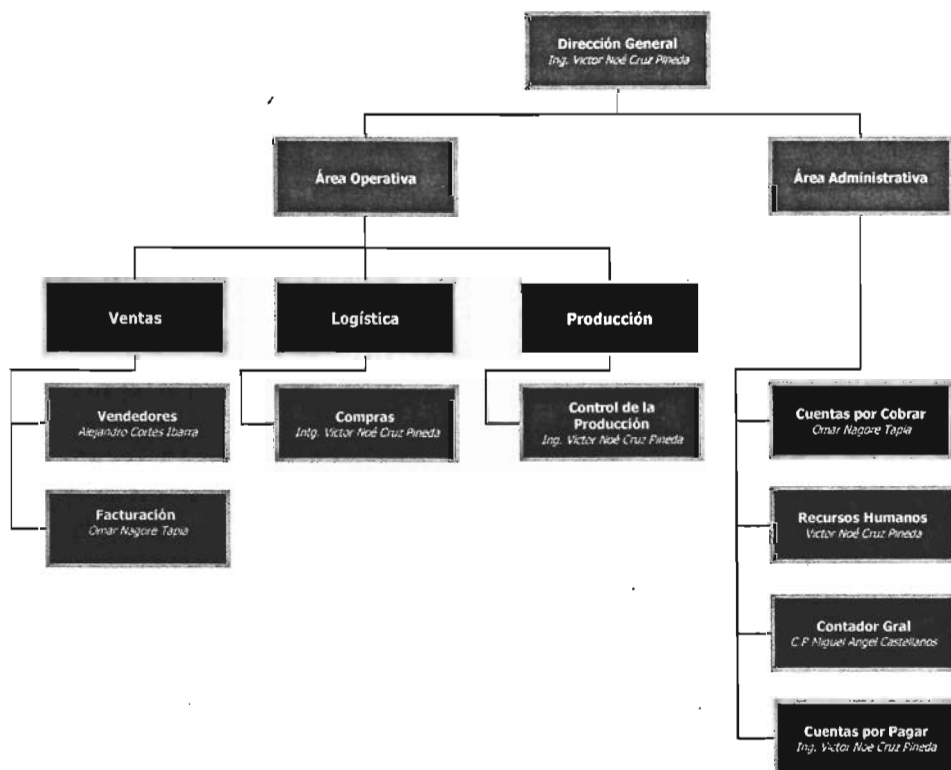


Figura 1.2.3.1 Organigrama NUESTROSITE

▪ 1.2.4 Sistema Informático Actual

El sistema informático actual no cuenta con las funcionalidades necesarias en el módulo de facturación y cuentas por cobrar. Existe la necesidad de incrementar la funcionalidad para que cubra con las especificaciones del modelo de negocio pero este incremento no se puede llevar a cabo debido a que no se cuenta con una estructura bien definida de dicho sistema, es decir, se desarrolló sin la aplicación de estándares y metodologías y no se cuenta con la definición del modelo de negocio y ninguna documentación, esto ocasiona que al querer incrementar la funcionalidad no se optimicen los procesos del modelo y únicamente se van agregando parches para resolver la problemática de manera inmediata.

1.3 DIAGNÓSTICO

▪ 1.3.1 Producto

Debido a las características de sus productos y servicios, el ISP se ve en la necesidad de modificar la forma de facturar los mismos. Es necesario fortalecer el proceso de facturación añadiendo funcionalidad para cubrir las problemáticas detectadas.

▪ 1.3.2 Proceso

Las problemáticas que se detectaron en el proceso actual son las siguientes:

1. Se registra el cliente tantas veces como dominios tenga contratados
2. Si el cliente no tiene un dominio registrado no es posible vender le ningún servicio
3. Se imprime una factura por servicio de cobro programado, es decir, se imprimen facturas de una sola partida aunque el cliente tenga pendiente por facturar varios servicios

▪ 1.3.3 Sistema Informático Actual

Se requiere rediseñar el modelo de negocio tomando en cuenta toda la funcionalidad anterior y los nuevos requerimientos detectados, aplicando estándares y metodología para contar con la documentación necesaria para permitir el crecimiento del sistema conforme a los nuevos requerimientos del ISP.

▪ 1.3.4 Organización

Existe una organización informal que es necesario formalizar para definir perfectamente bien las líneas de autoridad y coordinación; la definición de puestos para la asignación de responsabilidades; y la definición de procedimientos para el cumplimiento de la secuencia de las actividades.

Las líneas de autoridad permiten establecer una relación única entre un puesto de responsabilidad y los puestos de apoyo con sentido descendente para la comunicación de órdenes y en forma ascendente para la comunicación de informes.

Las líneas de coordinación permiten establecer una o varias relaciones de coordinación entre puestos de igual nivel de responsabilidad o de apoyo con sentido en las dos direcciones para comunicar información necesaria para el desempeño del trabajo. Las líneas de autoridad y coordinación deben de ser únicas y claras.

Es de suma importancia que al final del proyecto y antes de iniciar las capacitaciones del nuevo sistema, los manuales de organización, políticas y procedimientos deberán estar terminados y se deberán dar a conocer al personal involucrado en el uso del sistema.

1.4 ESTRATÉGIA DE SOLUCIÓN

A continuación se presenta la estrategia de solución apoyada en los siguientes elementos:

1. Estrategia de crecimiento de NUESTROSITE.COM
2. Ambiente organizacional: Estructuración de manuales de organización, políticas y procedimientos operativos
3. Infraestructura del sistema de información
4. Aplicación Informática E-BILLING en base de datos MySQL y lenguaje de programación JAVA
5. Usuarios del sistema

1.5 PLAN DETALLADO DE ACTIVIDADES

Actividad	SEMANAS																								TOTAL	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24		24
I Análisis de Requerimientos																										
II Diseño del Modelo																										
III Desarrollo de la aplicación																										

1.6 RIESGOS Y BENEFICIOS

▪ 1.6.1 Riesgos

El riesgo que existe al redefinir procesos y responsabilidades es que el personal (los usuarios) se resista al cambio y omitan algunas tareas definidas y esto ocasione que el sistema no funcione como se pretende al rediseñarlo.

▪ 1.6.2 Beneficios

El aplicar un proceso de reingeniería al modelo actual de la empresa ayudará a redefinir los procesos contemplando las nuevas necesidades y permitirá crecer la aplicación conforme al crecimiento del ISP

El sistema E-Billing "mejorado" al desarrollarse en una nueva plataforma bajo estándares agregará además de las funcionalidades necesarias, funcionalidades adicionales como son la seguridad y una mejor interfaz de aplicación para el usuario final.

CAPÍTULO II ESPECIFICACIÓN DE REQUERIMIENTOS

2.1 INTRODUCCIÓN

▪ 2.1.1 Reseña del Producto

E-BILLING, Sistema de Información para ISPs (Proveedores de Servicios de Internet) que cuenta con los siguientes módulos

- Configuración del Sistema: en este módulo se administran (altas, bajas y cambios) los catálogos necesarios para operar, así como la seguridad correspondiente a los accesos para usuarios del sistema
- Ventas: a través de este módulo se administran las solicitudes de los clientes (ordenes de venta), la facturación de ordenes de venta, y la administración de los envíos de facturas para los clientes (asociación de guías y rastreo)
- Cuentas por Cobrar: este módulo consiste en administrar los pagos de los clientes aplicados a las facturas
- Consultas: a través de este módulo los clientes podrán visualizar sus estados de cuenta (facturas y cobros aplicados)

2.2 ESPECIFICACIONES FUNCIONALES

▪ 2.2.1 Necesidades acordadas al Usuario

E-BILLING, tiene cuatro "USUARIOS TIPO" o roles que corresponden a los módulos que se desarrollarán para el sistema

- Administrador del sistema
- Vendedor
- Operador
- Cobrador
- Cliente

El rol define las distintas competencias que existen en el sistema. Se definen independientemente de las personas físicas a las cuales se les va a asignar dichos roles. Una persona puede tener más de un rol

Los objetivos por rol son los siguientes:

- El administrador del sistema debe contar con los catálogos necesarios para y administrar el sistema
- El vendedor requiere introducir en el sistema sus procesos que realiza:
 - ◊ Alta de las solicitudes de los clientes (ordenes de venta)
 - ◊ Impresión de facturas
 - ◊ Asociación de envíos de factura a clientes (guías)
- El cobrador requiere de una consulta de las cuentas por cobrar pendientes para darles seguimiento y una interfaz de captura para aplicar los pagos realizados por los clientes

- El cliente requiere de una interfaz de consulta para visualizar su estado de cuenta

▪ 2.2.2 Actores

Un actor es algo con comportamiento como una persona (identificada por un rol), un sistema informatizado u organización que realiza algún tipo de interacción con el sistema

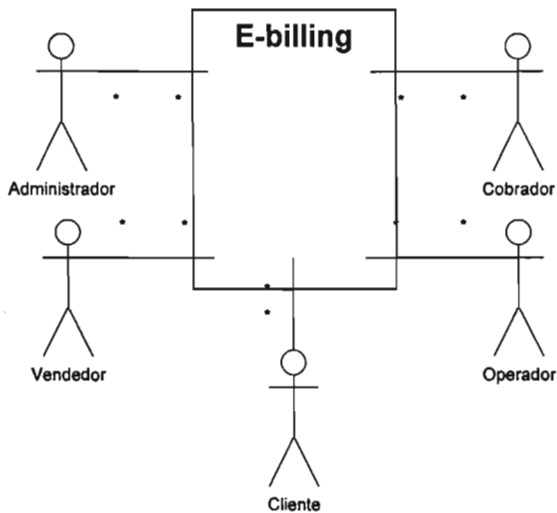
Notación;



Los actores en este caso son roles:

- Administrador del sistema
- Vendedor
- Operador
- Cobrador
- Cliente

Delimitación del sistema en base a los actores



▪ 2.2.3 Diagrama de Clases

Una clase describe un conjunto de objetos con:

- propiedades semejantes
- comportamientos semejantes
- relaciones comunes con otros objetos

Una clase es una fábrica de objetos. Los objetos contienen datos (propiedades) y ejecutan funciones (métodos), los objetos se relacionan. En las clases son definidos los métodos y las propiedades de los objetos.

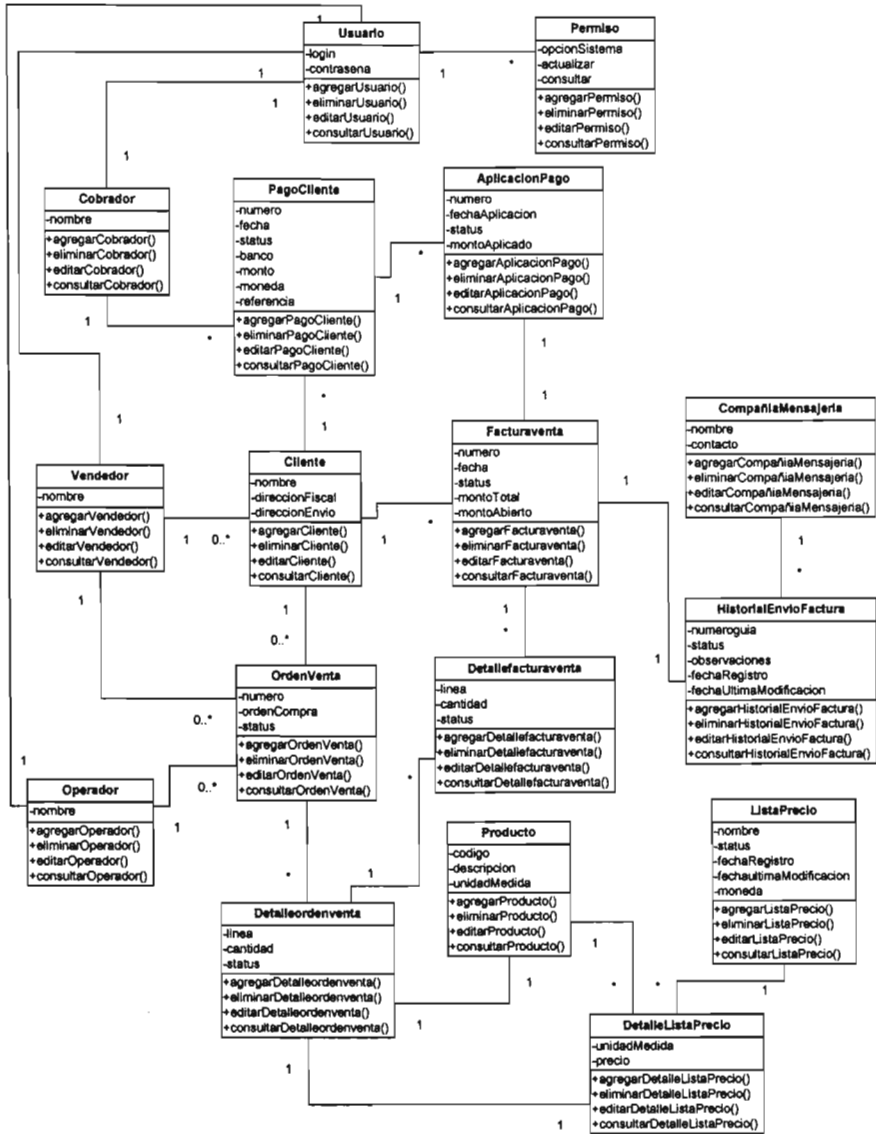
Un diagrama de clases:

1. Representa la estructura estática del sistema
2. Muestra los elementos que componen el sistema (clases)
3. Identifica como los datos y las funciones están distribuidas por las clases, y cuales relaciones existen entre ellas

Finalidad del diagrama de clases:

1. Después de identificados los requisitos que el sistema deberá atender (modelado de casos de uso), es necesario identificar los elementos que componen el sistema
2. Esos objetos son agrupados en clases por medio de un modelo gráfico

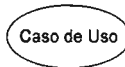
E-BILLING Diagrama de Clases



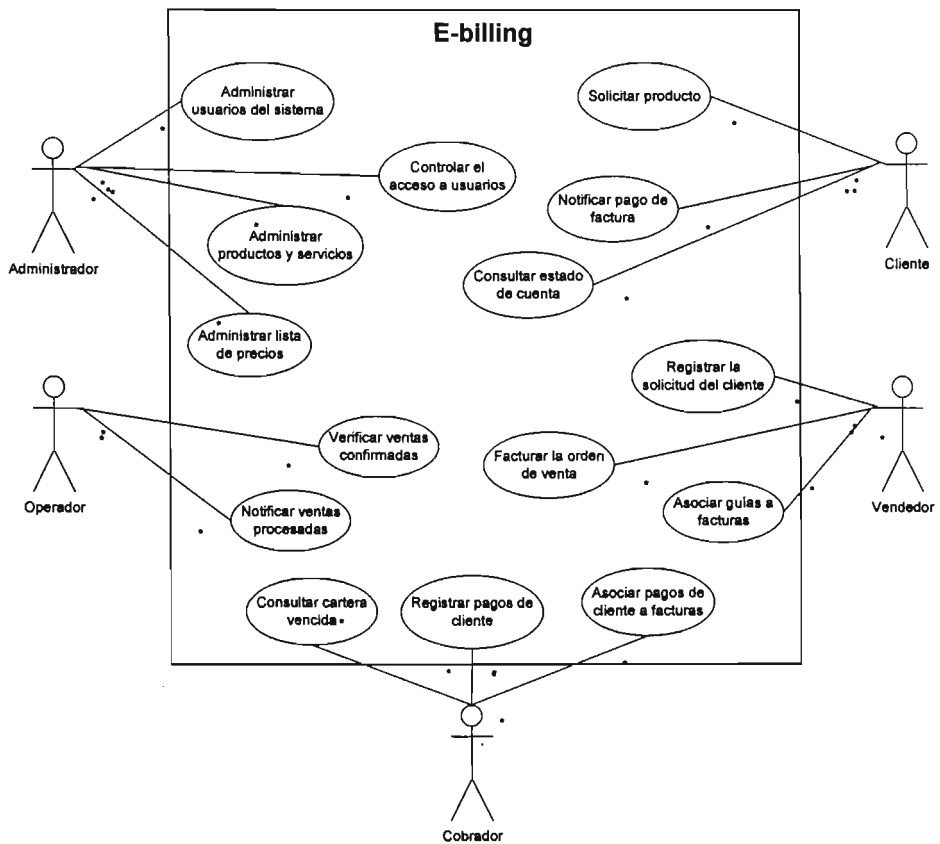
2.2.4 Diagrama de Casos de Uso

Un caso de uso es una operación o tarea específica que se realiza tras una orden de algún agente externo, sea desde una petición de un actor, o bien desde la invocación de otro caso de uso.

Notación:



A continuación se muestra el diagrama de casos de uso correspondiente al sistema E-billing



▪ 2.2.5 Descripción de Casos de Uso

▪ Administrador del sistema

CU 01 Administrar (alta baja y cambios) el catálogo usuarios del sistema

CU 02 Controlar el acceso de los usuarios al sistema (Contraseñas y permisos para las funcionalidades del sistema)

CU 03 Administrar (alta baja y cambios) el catálogo de productos y servicios

CU 04 Administrar (alta baja y cambios) el catálogo de listas de precios

CU 05 Administrar (alta baja y cambios) el catálogo de clientes

▪ Cliente

CU 06 Enviar orden de compra al vendedor para solicitar el producto o servicio

CU 07 Notificar pago de factura

CU 08 Consultar estado de cuenta

▪ Vendedor

CU 09 Registrar la solicitud del cliente (Orden de Venta)

CU 10 Verificar los saldos vencidos del cliente.

Esta verificación se hace para determinar si es factible seguir vendiendo al cliente, es decir, si no ha excedido su límite de crédito o tiene algún problema en cuentas por cobrar. No se realiza para clientes nuevos

CU 11 Facturar la orden de venta

CU 12 Asociar guías a facturas

▪ Operador

CU 13 Verificar ventas confirmadas para procesarlas

CU 14 Notificar mediante un cambio de status que la orden de venta se proceso

▪ Cobrador

CU 15 Consultar cartera vencida para notificar a los clientes de su retraso

CU 16 Registrar pagos de cliente

CU 17 Asociar los pagos de cliente a las facturas

2.3 REQUERIMIENTOS NO FUNCIONALES

▪ 2.3.1 Uso del Sistema

El uso del sistema debe permitir a los ISVs mejorar la capacidad para procesar transacciones con rapidez y eficiencia

Esta capacidad se mejora en tres formas:

1. Aumentar la velocidad de procesamiento
2. Aumento en el volumen de transacciones procesadas
3. Obtención de información de manera rápida y oportuna

▪ 2.3.2 Seguridad

Se requiere seguridad en la información para protegerla contra el acceso desautorizado o la modificación de información si está en fase de almacenamiento, procesamiento o tránsito. La seguridad debe ser preservada durante la operación y el mantenimiento para asegurar el buen funcionamiento del sistema

▪ 2.3.3 Disponibilidad

Se requiere que el sistema tenga fiabilidad ya que es un elemento muy importante de su calidad general.

La fiabilidad del software se define en términos estadísticos como la probabilidad de operación libre de fallos de un programa de computadora en un entorno determinado y durante un tiempo específico.

El fallo es cualquier falla de concordancia con los requisitos del software. Existen grados de fallo. Los fallos pueden ser simplemente desconcertantes o ser catastróficos. Puede que un fallo sea corregido en segundos mientras que otro lleve semanas o incluso meses. La corrección de un fallo puede llevar a la introducción de otros errores que, finalmente, lleven a más fallos.

Por esta razón es importante tomar como requisito que el sistema sea lo más aproximado a libre de fallos para tener la disponibilidad requerida que es "en todo momento"

▪ 2.3.4 Mantenimiento y Crecimiento

Se requiere utilizar un enfoque orientado a objetos para tener mejoramientos de calidad a lo largo de vida de la aplicación, esto para facilitar el mantenimiento y la creación de nuevas versiones que extiendan el programa

2.5 REQUERIMIENTOS DE MEDIO AMBIENTE (DESARROLLO E IMPLEMENTACIÓN)

En base a la experiencia de otras implementaciones, se detalla una lista de requerimientos que cumplen con las expectativas de funcionalidad, costo y beneficio.

▪ 2.5.1 Hardware

Servidor de base de datos y de aplicación

- Procesador Intel Xeon 2.4 Mhz
- Memoria ECC2 GB RAM
- 2 Discos duro SCSI 73 GB
- Controladora SCSI RAID RAID 1
- Sistema Operativo Red Hat Enterprise Linux 4
- Tarjeta de red 1000 mbps

▪ 2.5.2 Software

- Sistema operativo Red Hat LINUX
- Manejador de base de datos MySQL
- Máquina virtual y Java SDK
- Servidor de aplicación Jakarta Tomcat
- Servidor de correo postfix
- Servidor de servicios FTP Proftpd
- Servidor de servicios SSH OpenSSH
- Modelador entidad relación Case Studio
- Entorno de desarrollo Eclipse
- Entorno de diseño html Dream Weaver
- Reglas y políticas de seguridad firewall

CAPÍTULO III MODELO DE ANÁLISIS

3.1 INTRODUCCIÓN

Cuando ya se ha desarrollado y aceptado el modelo de requisitos, se inicia el desarrollo del modelo de análisis siguiendo el modelo de casos de uso, cuyo objetivo es comprender y generar una arquitectura de objetos para el sistema con base en lo especificado en el modelo de requisitos. Durante esta etapa no se considera el ambiente de implementación, modelando el sistema bajo condiciones ideales. Durante el modelo de diseño el sistema tendrá que adaptarse a las condiciones de implementación deseadas.

Es importante enfatizar que el modelo de análisis es una representación conceptual, correspondiente al problema y modelo de requisitos, en término de clase de objetos. Cada una de estas clases contribuye de manera especial a lograr la arquitectura deseada.

3.2 OBJETIVO

El desarrollar el modelo de análisis tiene como objetivo generar una especificación detallada del sistema de información que satisfaga las necesidades de los usuarios y sirva de base para el posterior diseño del sistema.

La participación activa de los usuarios es una condición imprescindible para el análisis del sistema de información ya que constituye una garantía de que los requisitos identificados son comprendidos e incorporados al sistema y, por tanto, de que éste será aceptado

3.3 CONCEPTOS BÁSICOS

▪ 3.3.1 Diagramas de Interacción

En los diagramas de interacción se muestra un patrón de interacción entre objetos. Hay dos tipos de diagramas de interacción, ambos basados en la misma información, pero cada uno enfatizando un aspecto en particular:

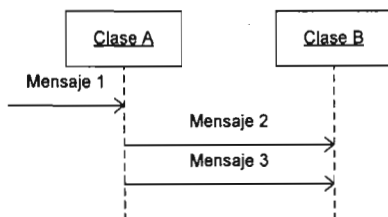
- Diagramas de secuencia
- Diagramas de colaboración

▪ 3.3.2 Diagramas de Secuencia

Un diagrama de secuencia muestra una interacción ordenada según la secuencia temporal de los eventos. En particular, muestra los objetos participantes en la interacción y los mensajes que cambian ordenados según una secuencia en el tiempo. El eje vertical representa el tiempo, y en el eje horizontal se colocan los objetos y

actores participantes en la interacción, sin un orden prefijado. Cada objeto o actor tiene una línea vertical, y los mensajes se representan mediante flechas entre los distintos objetos. El tiempo fluye de arriba abajo. Se pueden 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

Los diagramas de secuencia se usan para mostrar la interacción entre los usuarios, las pantallas y las instancias de los objetos en el sistema. Proveen un mapa secuencial del paso de los mensajes entre los objetos a lo largo del tiempo.



▪ 3.3.3 Diagramas de Colaboración

Un diagrama de colaboración muestra una interacción organizada basándose en los objetos que toman parte en la interacción y en los enlaces entre los mismos (en cuanto a la interacción se refiere). A diferencia de los diagramas de secuencia, los diagramas de colaboración muestran las relaciones entre los roles de los objetos. La secuencia de los mensajes y los flujos de ejecución concurrentes deben determinarse explícitamente mediante números de secuencia. En cuanto a la representación, un diagrama de colaboración muestra a una serie de objetos con los enlaces entre los mismos, y con los mensajes que se intercambian dichos objetos. Los mensajes son flechas que van junto al enlace por el que circula y con el nombre del mensaje y los parámetros (si los tiene) entre paréntesis. Cada mensaje lleva un número de secuencia que denota cual es el mensaje que le precede, excepto el mensaje que inicia el diagrama, que no lleva número de secuencia

3.4 DESARROLLO

▪ 3.4.1 Diagramas de Secuencia

EL diagrama de secuencia es uno de los diagramas más efectivos para modelar interacción entre objetos de un sistema. Un diagrama de secuencia se modela para cada caso de uso. Mientras el diagrama de caso de uso permite el modelado de una vista "business" del escenario, el diagrama de secuencia contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario, y mensajes pasados entre los objetos

Típicamente se examina la descripción de un caso de uso para determinar que objetos son necesarios para la implementación del escenario

Un diagrama de secuencia muestran los objetos que intervienen en el escenario con líneas discontinuas verticales, y los mensajes pasados entre los objetos como vectores horizontales. Los mensajes se dibujan cronológicamente desde la parte superior del diagrama

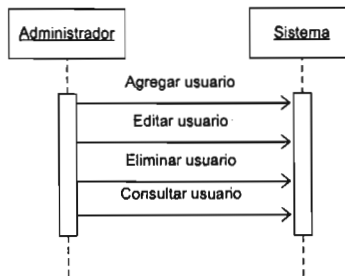
En cada caso de uso se muestra una interacción de actores con el sistema. En esta interacción los actores generan eventos, solicitando al sistema operaciones.

▪ 3.4.2 Construcción de los Diagramas de Secuencia del Sistema

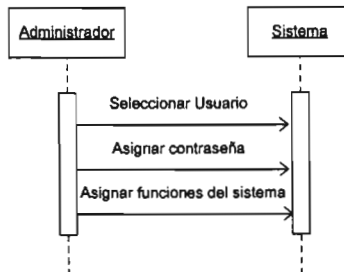
A continuación se muestran los diagramas de secuencia generados a partir de los casos de uso por actor o rol para la actualización de catálogos generales

Administrador del sistema

CU 01 Administrar el catálogo de usuarios del sistema



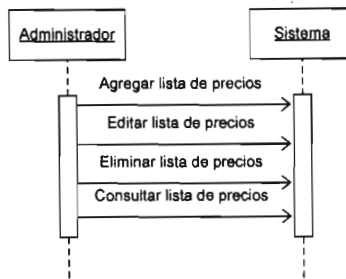
CU 02 Controlar el acceso de los usuarios al sistema



CU 03 Administrar el catálogo de productos y servicios



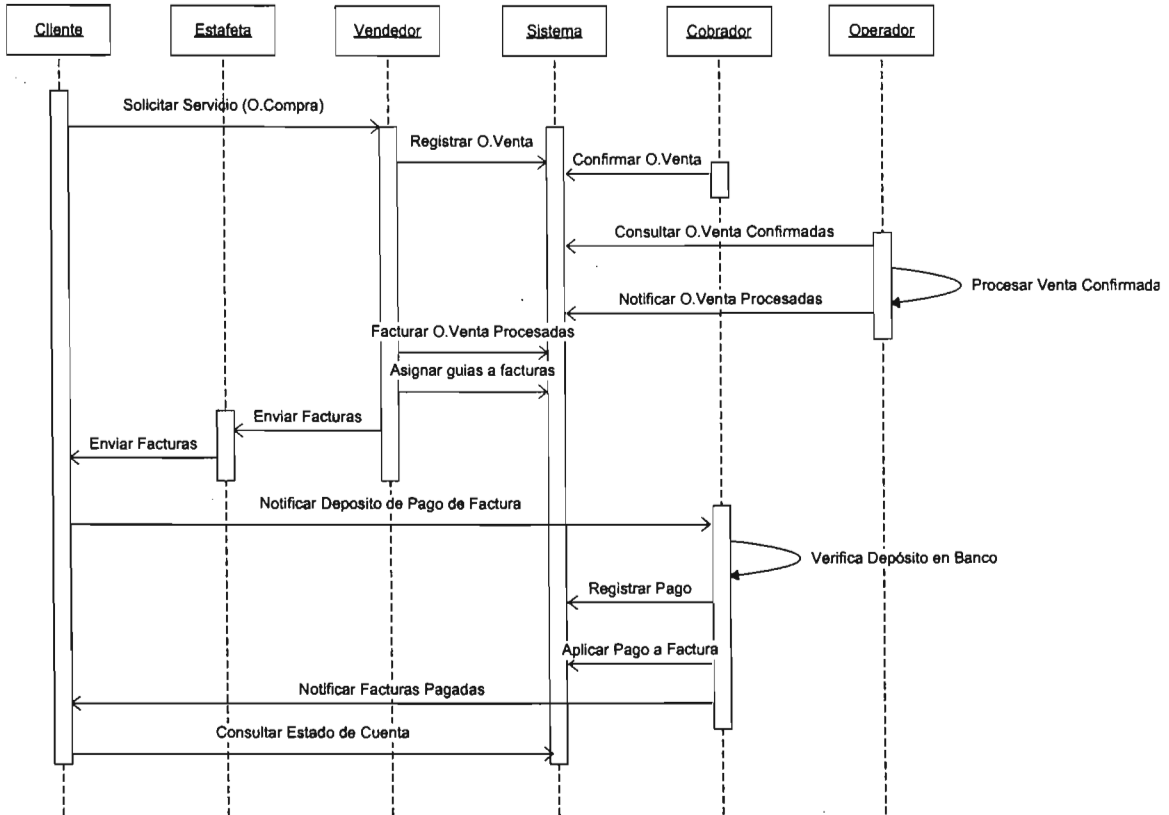
CU 04 Administrar el catálogo de productos y servicios



CU 05 Administrar el catálogo de clientes

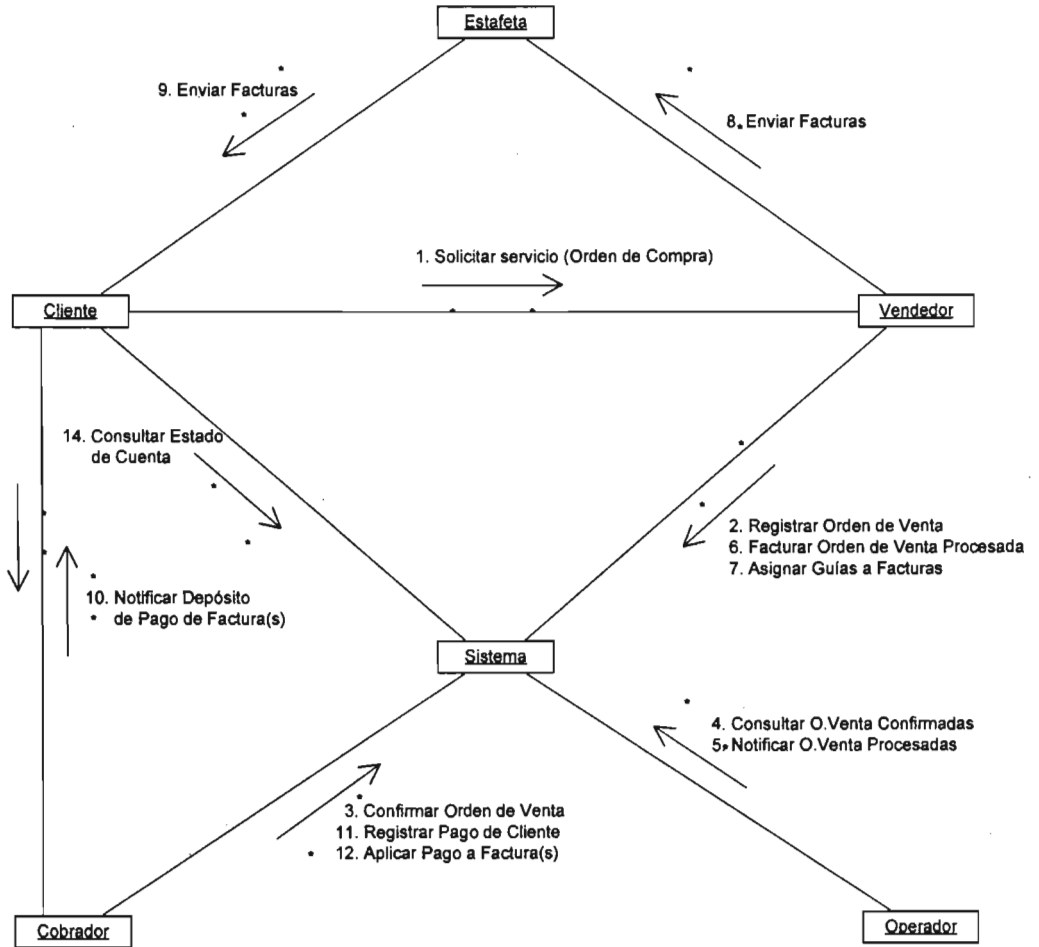


A continuación se muestra el diagrama de secuencia general del sistema E-BILLING en correspondencia a los casos de uso generados en el modelo de requerimientos



3.4.3 Construcción del Diagrama de Colaboración

A continuación se muestran el diagrama de colaboración del sistema E-BILLING en correspondencia a los casos de uso generados en el modelo de requerimientos



3.5 CONCLUSIONES

En una organización o Empresa, el análisis y Diseño de Sistemas, es el proceso de estudiar su Situación con la finalidad de observar como trabaja y decidir si es necesario realizar una mejora; el encargado de llevar a cabo estas tareas es el analista de sistemas.

Antes de comenzar con el desarrollo de cualquier proyecto, se conduce un estudio de Sistemas para detectar todos los detalles de la situación actual de la empresa. La información reunida con este estudio sirve como base para crear varias estrategias de Diseño. Los administradores deciden que estrategias seguir. Los Gerentes, empleados y otros usuarios finales que se familiarizan cada vez mas con el uso de computadoras están teniendo un papel muy importante en el desarrollo de sistemas.

Todas las organizaciones son Sistemas que actúan de manera reciproca con su medio ambiente recibiendo entradas y produciendo salidas. Los Sistemas que pueden estar formados por otros Sistemas de denominan Sub-sistemas y funcionan para alcanzar los fines de su Implantación.

CAPÍTULO IV MODELO DE DISEÑO

4.1 DEFINICIONES

- Etapa del ciclo de vida en la que los diseños de la arquitectura, componentes software, interfaces y datos son creados, documentados y verificados para que satisfagan los requisitos [IEEE Glosario Estándar]
- Descompone el sistema en sus auténticos componentes (arquitecturales) constituyentes, y después los descompone iterativamente... Define los algoritmos de cada módulo que serán implementados como código [Davis]
- (Flujo de trabajo) fundamental cuyo propósito principal es el de formular modelos que se centran en los requisitos no-funcionales y el dominio de la solución, y que prepara para la implementación y pruebas del sistema [RUP]

4.2 PERSPECTIVA DE UN PROCESO UNIFICADO

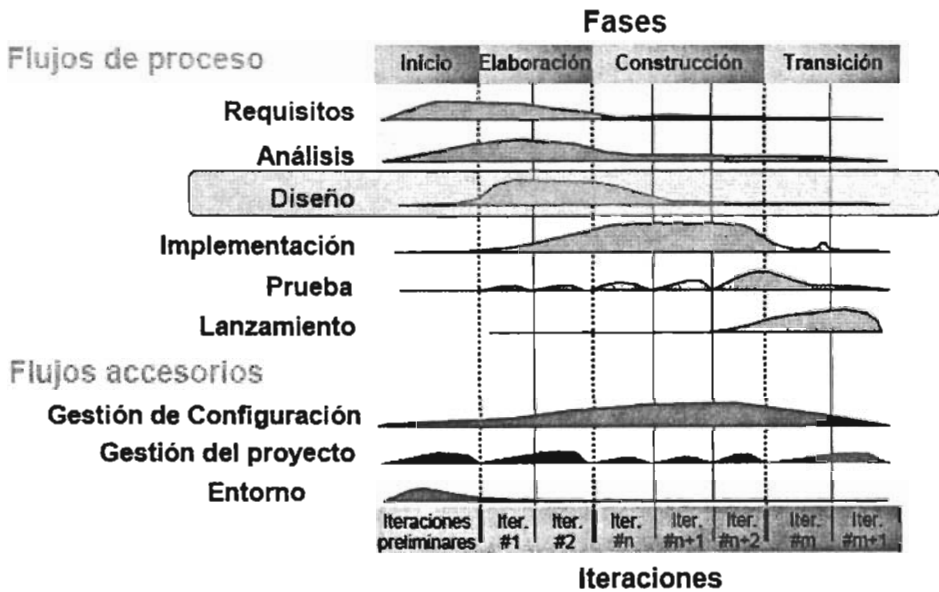


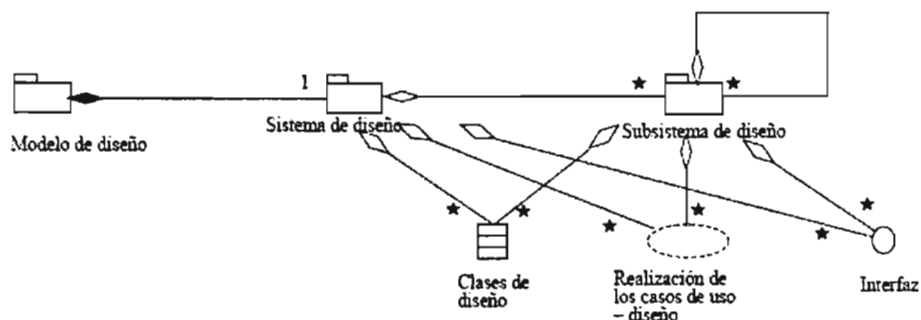
Figura 4.2.1 Etapas de la metodología RUP

4.3 OBJETIVOS

1. Comprender los aspectos relacionados con los requisitos no funcionales y restricciones relacionadas con los lenguajes de programación, componentes reutilizables, sistemas operativos, tecnologías de distribución y concurrencia, tecnologías de interfaz de usuario, tecnologías de gestión de transacciones, etc.
2. Crear un punto de partida para actividades de implementación subsiguientes
3. Ser capaces de descomponer los trabajos de implementación en partes más manejables que puedan ser llevadas a cabo por diferentes equipos de desarrollo, teniendo en cuenta la posible concurrencia

4.4 DESARROLLO

El modelo de diseño es un modelo orientado al objeto que describe la realización física de los casos de uso, centrándose en los requisitos funcionales, no funcionales y restricciones del entorno de implementación



▪ 4.4.1 Diseño - Implementación

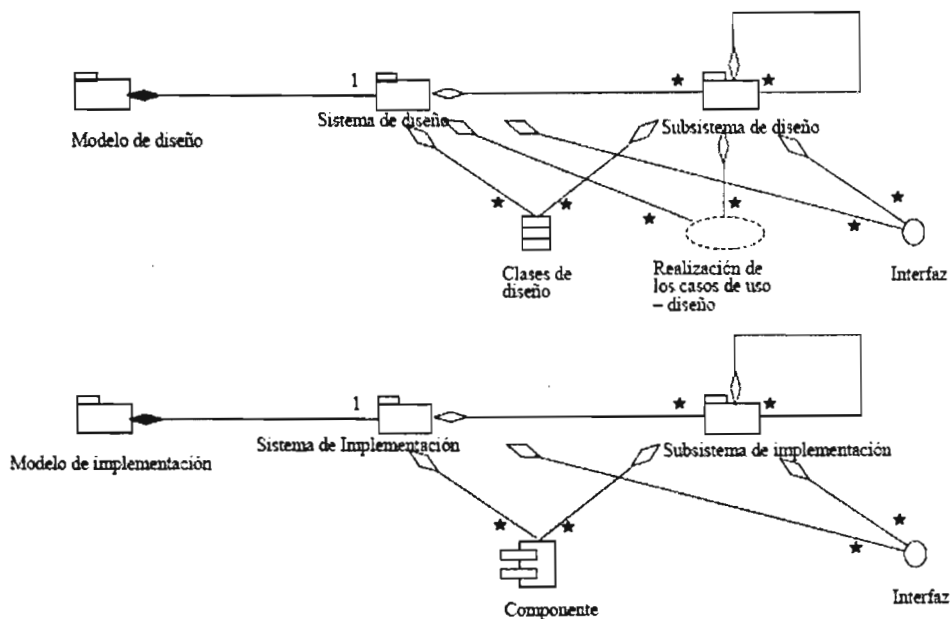


Figura 4.4.1.1 Modelo de diseño VS Modelo de implementación

4.4.2 Componentes

Parte física y reemplazable de un sistema que es conforme a un conjunto de interfaces y proporciona la realización de dicho conjunto

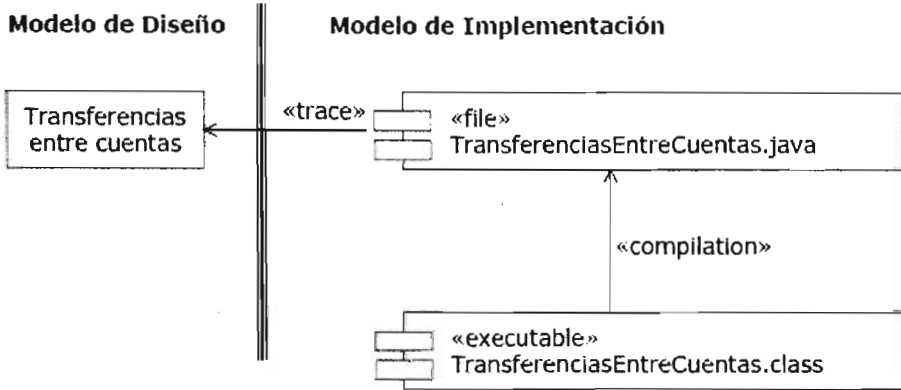


Figura 4.4.2.1 Conexión Modelo de diseño – Modelo de implementación

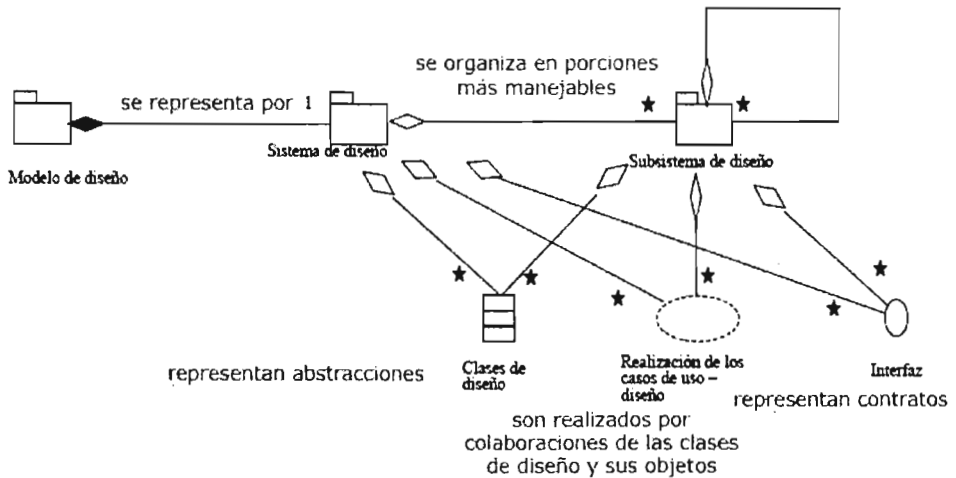


Figura 4.4.2.2 Modelo de implementación

▪ 4.4.3 Clases de Diseño

Es una abstracción '*sin costuras*' de una clase o construcción similar en la implementación del sistema

1. Se especifican utilizando la sintaxis del lenguaje de programación elegido
2. Se especifica la visibilidad de los atributos y operaciones *-public, protected, private-*
3. Se implementan las relaciones: referencias entre objetos
4. Se especifican los métodos: en lenguaje natural o pseudocódigo, que puede ser utilizado como comentario en la implementación del método
5. A menudo aparece con un estereotipo que se corresponde con una construcción de lenguaje de programación
6. Puede realizar interfaces si tiene sentido en el lenguaje de programación
7. Puede activarse; las clases activas aparecerán en el modelo de proceso, en el que se detallará su distribución y concurrencia

▪ 4.4.4 Realización de caso de uso - diseño

Es una colaboración en el modelo de diseño que describe cómo se realiza un caso de uso específico en términos de clases de diseño

1. Descripción textual de flujo de eventos
2. Diagrama de clases con las clases participantes
3. Diagramas de interacción que muestran la realización de escenarios en términos de interacción entre objetos de diseño
4. También gestiona requisitos no funcionales

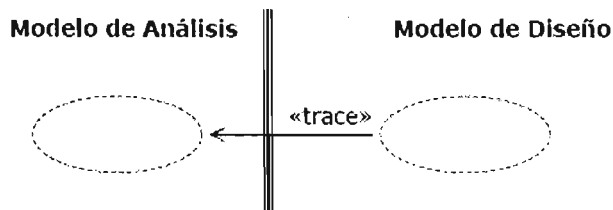


Figura 4.4.4.1 Conexión entre el modelo de análisis y el modelo de diseño

▪ 4.4.5 Interfaces

Se utilizan para especificar las operaciones que proporcionan las clases de diseño

1. Una clase de diseño que realice una interfaz debe proporcionar, también, los métodos que realicen las operaciones de la interfaz
2. Constituyen una forma de separar la especificación de funcionalidad de operaciones, de su implementación en términos de métodos

▪ 4.4.6 Flujo de trabajo de diseño

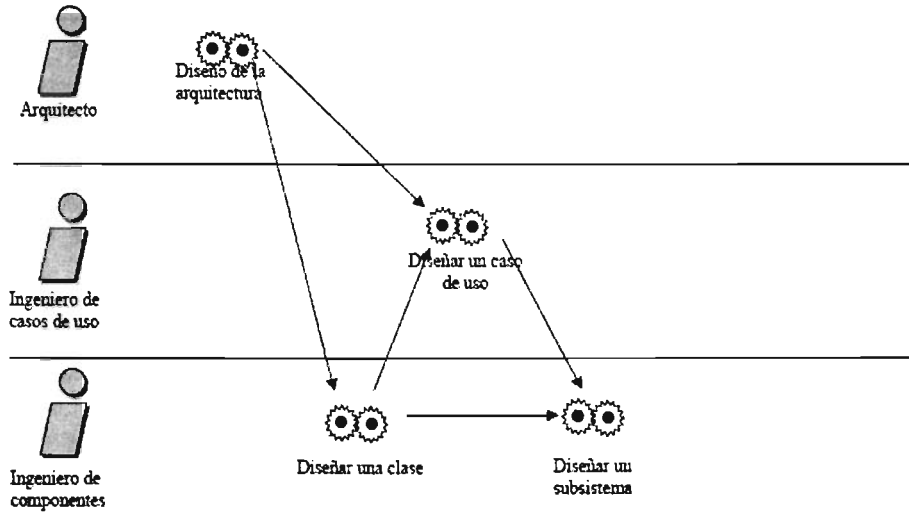


Figura 4.4.6.1 Etapas en el flujo de trabajo de diseño

4.4.7 Flujo de trabajo general

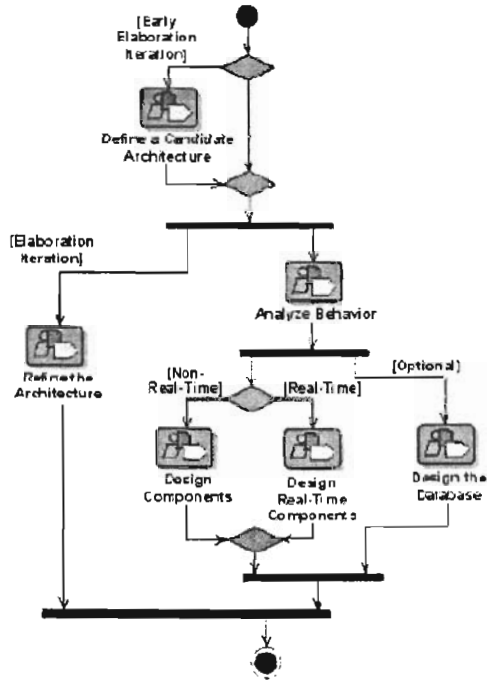


Figura 4.4.7.1 Etapas de flujo de trabajo general

4.4.8 Diseño de Componentes

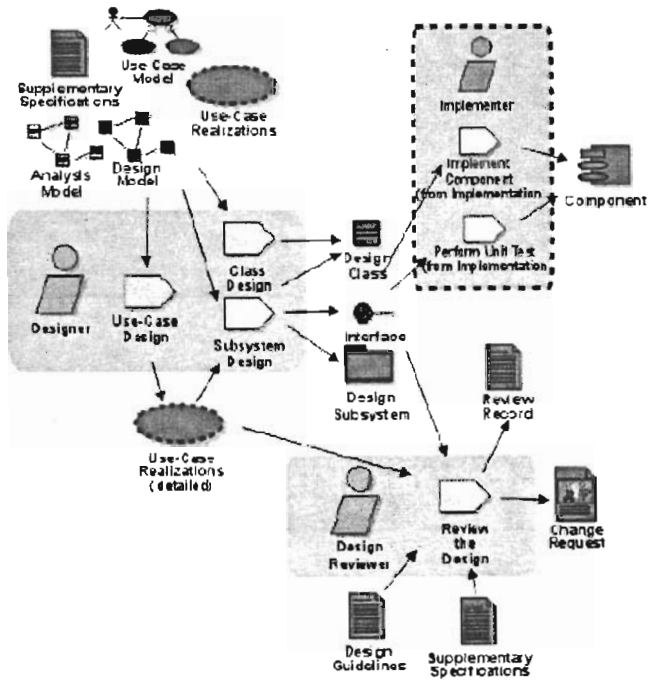


Figura 4.4.8.1 Etapas en el diseño de componentes

▪ 4.4.9 Diseño de Base de Datos

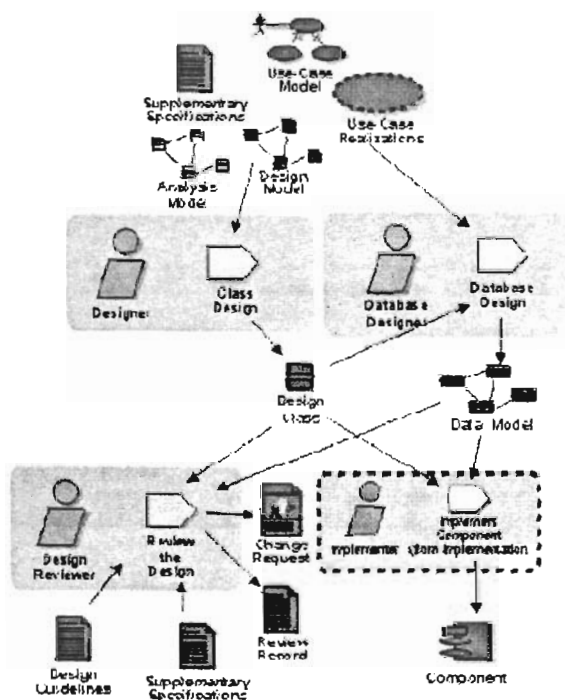


Figura 4.4.9.1 Implementación del modelo de bases de datos

CAPÍTULO V MODELO DE IMPLEMENTACIÓN

5.1 INTRODUCCIÓN

Este modelo es una colección de componentes y los subsistemas que los contienen. Estos componentes incluyen: archivos de código fuente y todo tipo de archivos necesarios para la iplantación y despliegue del sistema (Este modelo es sólo una versión preliminar que se obtendrá al final de la fase de desarrollo, posteriormente tendrá refinamiento).

Los archivos se entregarán como adjuntos en el cd referencia 5.1

CAPÍTULO VI MODELO DE PRUEBAS

6.1 INTRODUCCIÓN

Hoy en día, debido al aumento del tamaño y la complejidad del software, el proceso de prueba se ha convertido en una tarea vital en el desarrollo de cualquier sistema informático. Son muchas las clasificaciones de los tipos de prueba que se pueden realizar. Una clasificación posible se muestra en la siguiente tabla

Tipo de pruebas	Fase de realización	Descripción
Pruebas Unitarias	Durante la construcción del sistema	Prueban el diseño y el comportamiento de cada uno de los componentes del sistema una vez construidos
Pruebas de Integración	Durante la construcción del sistema	Comprueban la correcta unión de los componentes del sistema entre sí a través de sus interfaces, y si cumplen con la funcionalidad establecida
Pruebas de Sistema	Después de la construcción del sistema	Prueban a fondo el sistema, comprobando su funcionalidad e integridad globalmente, en un entorno lo más parecido posible al entorno final de producción
Pruebas de Aceptación	Después de la implantación en el entorno de producción	Verifican que el sistema cumple con todos los requisitos indicados y permite que los usuarios del sistema den el visto bueno definitivo
Pruebas de Regresión	Después de realizar modificaciones al sistema	El objetivo es comprobar que los cambios sobre un componente del sistema, no generan errores adicionales en otros componentes no modificados

Las pruebas unitarias y de integración no pueden comenzar hasta que no se dispone de componentes del sistema ya construidos, al igual que las pruebas de implantación y aceptación que tampoco pueden comenzar hasta que se disponga del sistema completo y se instale en su entorno de producción. Sin embargo, la planificación de las pruebas del sistema puede comenzar antes de que el sistema esté terminado.

Como el objetivo de las pruebas de sistema es comprobar que todo lo que está desarrollando cumple con los requisitos del sistema, la planificación de estas pruebas y el diseño de los casos de prueba pueden comenzar tan pronto como empiecen a estar disponibles las especificaciones funcionales del sistema.

La planificación y diseño de las pruebas de sistema en las primeras fases de desarrollo, cuando aún se realiza la especificación de requisitos, permite encontrar errores, omisiones, inconsistencias y sobreespecificaciones en los requisitos funcionales cuando aún es fácil y económico corregirlas, ya que el coste de eliminar defectos aumenta a

medida en que aumenta el tiempo que transcurre entre la aparición del defecto y su detección.

6.1.1 Descripción de las Pruebas de Software

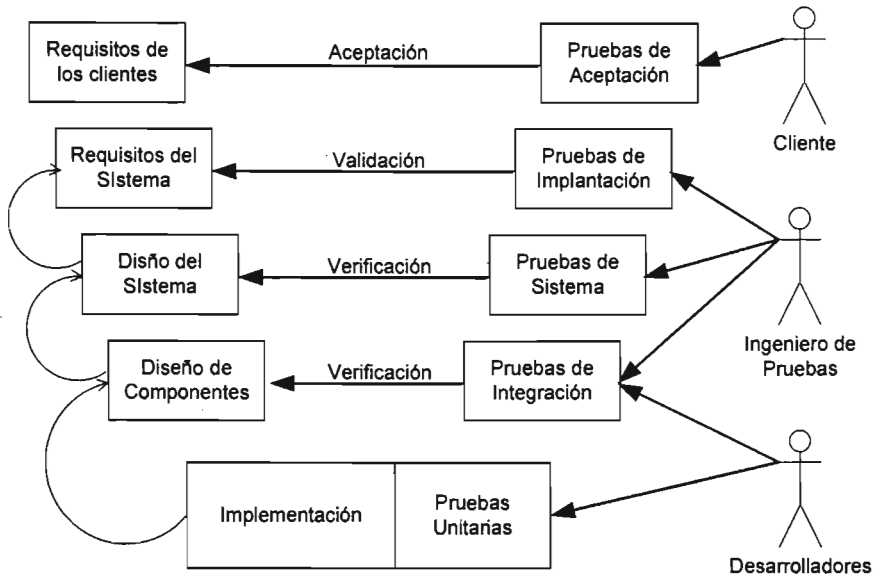


Figura 6.1.1.1 Tipos de prueba en software

Para que el proceso de prueba del sistema sea eficaz debe de integrarse junto con el proceso de desarrollo y, como cualquier otra fase de dicho proceso, debe realizarse de manera sistemática, minimizando el factor experiencia o intuición. Esto se puede conseguir a través de metodologías que guíen el proceso de desarrollo de pruebas de sistema a partir de requisitos funcionales

6.2 OBJETIVOS

El control de calidad de cualquier producto debe comprobar que la calidad del producto final funcione correctamente de acuerdo con sus especificaciones y en colaboración con otros sistemas y bases de datos.

Un aspecto crucial del control de la calidad son las pruebas del software y, dentro de éstas, las pruebas del sistema.

Cuando hablamos de pruebas de software nos referimos a la verificación dinámica del comportamiento de un sistema. Esta verificación está basada en la observación de un conjunto seleccionado de ejecuciones controladas o casos de prueba

Las pruebas de sistema son aquellas que se aplican al producto final, las cuales permiten detectar en que puntos el producto no cumple sus especificaciones. El objetivo de estas pruebas es probar a fondo el sistema, comprobando su funcionalidad e integridad globalmente, en un entorno lo más parecido posible al entorno final de producción.

El objetivo de las pruebas del sistema es comprobar que el sistema software que se está desarrollando cumple con la funcionalidad recogida en casos de uso o escenarios. En este trabajo se describe, analiza y aplica una propuesta para la obtención de casos de prueba para la realización de pruebas del sistema a partir de los requisitos del sistema

6.3 DESCRIPCIÓN DE LA PROPUESTA

En este apartado vamos a describir brevemente una propuesta llamada "GENERACIÓN DE CASOS DE PRUEBA A PARTIR DE CASOS DE USO" que permite obtener un conjunto de casos de prueba para la realización de pruebas del sistema a partir de los requisitos funcionales del sistema E-BILLING.

6.4 GENERACIÓN DE CASOS DE PRUEBA A PARTIR DE CASOS DE USO

Esta propuesta desarrolla un método basado en tres pasos para obtener un conjunto de casos de prueba del sistema a partir de casos de uso en UML.

Paso	Descripción	Resultado
1	Generar escenarios de uso	Todos los posibles caminos de ejecución de cada caso de uso. Cada camino es un escenario de uso
2	Identificación de casos de prueba	Conjunto de casos de prueba a partir de los escenarios anteriores
3	Identificar los valores a probar	Valores de prueba asociados a cada caso de prueba anterior

Para generar los casos de prueba, primero se generan todos los posibles escenarios o caminos de ejecución de cada caso de uso.

En la siguiente ilustración se muestra un ejemplo genérico en dónde se aprecia un camino de ejecución principal, flecha de color oscuro, y cuatro caminos de ejecución alternativos, uno que provoca la finalización de la ejecución y tres que hacen retroceder la ejecución a un paso anterior.

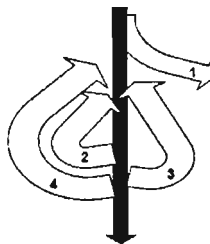


Figura 6.4.1 Ejemplo genérico. Varias alternativas de ejecución en un proceso

Después se identifican los casos de prueba a partir de estos escenarios, y por último se identifican los valores a probar de cada caso de prueba.

Tomando como punto de partida los casos de uso y su descripción no formal obtendremos al final una lista de casos de prueba, con los valores que deben probar y los resultados esperados para cada caso de uso

La parte más importante de un caso de uso para la generación de un caso de prueba es el camino de ejecución. Este camino se divide en dos en el patrón: el camino principal o secuencia normal y los caminos alternativos o excepciones.

El camino principal son los pasos que da el sistema si no surge ningún imprevisto ni error, mientras que los caminos alternativos son las variaciones que pueden surgir en distintos puntos del camino principal a causa de errores, rectificaciones, etc. A cada uno de estos caminos lo llamaremos escenarios de casos de uso. Todos los escenarios de caso de uso posible serán utilizados como base para crear las pruebas.

Un caso de prueba será un conjunto de entradas con datos de prueba, unas condiciones de ejecución, y unos resultados esperados. Para generar los casos de prueba aplicamos los tres puntos de la propuesta:

En el primer punto identificamos todas las combinaciones posibles de caminos de ejecución del caso de uso, es decir, todas las combinaciones posibles entre el camino principal y los caminos alternativos y le asignamos un nombre. Cada combinación será un escenario de uso. Todos los caminos posibles se recogen a continuación

▪ **6.4.1 Generación de escenarios de uso**

Escenario de caso de uso	Comienzo	Excepciones	
Control de acceso al sistema			
1. Acceso correcto	Secuencia Normal		
2. Falló de página (servidor no disponible o no carga adecuadamente)	Secuencia Normal	Excepción 1	
3. Nombre o contraseña en blanco	Secuencia Normal	Excepción 2	
4. Nombre no existente	Secuencia Normal	Excepción 3	
5. Contraseña no válida (no corresponde con el nombre de usuario)	Secuencia Normal	Excepción 4	
Administrar los usuarios del sistema			
6. Alta usuario nuevo con datos obligatorios completos	Secuencia Normal		
7. Alta usuario nuevo con los datos obligatorios incompletos	Secuencia Normal	Excepción 5	
8. Alta usuario existente	Secuencia Normal	Excepción 6	
Controlar el acceso de los usuarios al sistema (Contraseñas y permisos)			
9. Asignación de contraseña y permisos	Secuencia Normal		
10. Asignación de contraseña en blanco	Secuencia Normal	Excepción 7	
11. No asignación de	Secuencia Normal	Excepción 8	

permisos a funciones del sistema	Normal		
Administrar el catálogo de productos y servicios			
12. Alta de artículo nuevo con datos obligatorios completos	Secuencia Normal		
13. Alta de artículo nuevo con datos obligatorios incompletos	Secuencia Normal	Excepción 9	
14. Alta de artículo existente	Secuencia Normal	Excepción 10	
15. Baja de artículo ligado en el catálogo de listas de precios	Secuencia Normal	Excepción 11	
16. Baja de artículo ligado en el catálogo de Líneas de Orden de Venta	Secuencia Normal	Excepción 12	
17. Baja de artículo ligado en el catálogo de Líneas de Facturas de venta	Secuencia Normal	Excepción 13	
18. Baja de artículo ligado en el catálogo de existencias	Secuencia Normal	Excepción 14	
Administrar el catálogo de listas de precios			
19. Alta de nueva lista de precios con datos obligatorios completos	Secuencia Normal		
20. Alta de Detalle de listas de precios con artículos existentes y precio	Secuencia Normal		
21. Alta de nueva lista de precios con datos obligatorios incompletos	Secuencia Normal	Excepción 15	
22. Alta de lista de precios existente	Secuencia Normal	Excepción 16	
23. Alta de detalle de lista de precios artículo no existente	Secuencia Normal	Excepción 17	
24. Alta de detalle de lista de precios artículo existente sin precio	Secuencia Normal	Excepción 18	
Administrar el catálogo de clientes			
25. Alta de nuevo cliente con datos obligatorios completos	Secuencia Normal		
26. Alta de nuevo cliente con datos obligatorios incompletos	Secuencia Normal	Excepción 19	
27. Alta de cliente existente	Secuencia Normal	Excepción 20	
Registrar las solicitudes de los clientes (Ordenes de Venta)			
28. Alta de nueva solicitud con datos de venta	Secuencia Normal		

completos			
29. Alta de nueva solicitud con datos obligatorios incompletos	Secuencia Normal	Excepción 21	
30. Alta de solicitud existente	Secuencia Normal	Excepción 22	
Confirmar Ordenes de Venta (Cambio de Status)			
31. Confirmar orden de venta con status registrada	Secuencia Normal		
32. Confirmar orden de venta previamente confirmada o facturada	Secuencia Normal	Excepción 23	
Notificar Ordenes de Venta procesadas			
33. Cambio de status a ordenes de venta confirmadas	Secuencia Normal		
34. Cambio de status a ordenes de venta registradas o facturadas	Secuencia Normal	Excepción 24	
Registro de pagos de cliente			
35. Registro de nuevo pago de cliente	Secuencia Normal		
36. Registro de pago de cliente existente	Secuencia Normal	Excepción 25	
37. Asociación de pago no asignado a factura por cobrar	Secuencia Normal		
38. Asociación de pago asignado a factura por cobrar	Secuencia Normal	Excepción 26	
39. Asociación de pago no asignado a factura saldada	Secuencia Normal	Excepción 27	
40. Asociación de pago asignado a factura saldada	Secuencia Normal	Excepción 28	

- **6.4.2 Identificación de casos de prueba**

I Valor Inválido

V Valor Válido

O Vacío no se indica ningún valor

N/A El valor que tenga es irrelevante

Escenario de caso de uso	Valor	Resultados Esperados
Control de acceso al sistema		
1. Acceso correcto	V	Carga la página de Inicio
2. Falló de página (servidor no disponible o no carga adecuadamente)	I	Mensaje de error
3. Nombre o contraseña en blanco	I	Mensaje de error
4. Nombre no existente	I	Mensaje de error
5. Contraseña no válida (no corresponde con el nombre de usuario)	I	Mensaje de error
Administrar los usuarios del sistema		
6. Alta usuario nuevo con datos obligatorios completos	V	
7. Alta usuario nuevo con los datos obligatorios incompletos	O	Mensaje de error
8. Alta usuario existente	I	Mensaje de error
Controlar el acceso de los usuarios al sistema (Contraseñas y permisos)		
9. Asignación de contraseña y permisos	V	
10. Asignación de contraseña en blanco	I	Mensaje de error
11. No asignación de permisos a funciones del sistema	O	Mensaje de error
Administrar el catálogo de productos y servicios		
12. Alta de artículo nuevo con datos obligatorios completos	V	
13. Alta de artículo nuevo con datos obligatorios incompletos	O	Mensaje de error
14. Alta de artículo existente	I	Mensaje de error
15. Baja de artículo ligado en el catálogo de listas de precios	I	Mensaje de error

16. Baja de artículo ligado en el catálogo de Líneas de Orden de Venta	I	Mensaje de error
17. Baja de artículo ligado en el catálogo de Líneas de Facturas de venta	I	Mensaje de error
18. Baja de artículo ligado en el catálogo de existencias	I	Mensaje de error
Administrar el catálogo de listas de precios		
19. Alta de nueva lista de precios con datos obligatorios completos	V	
20. Alta de Detalle de listas de precios con artículos existentes y precio	V	
21. Alta de nueva lista de precios con datos obligatorios incompletos	I	Mensaje de error
22. Alta de lista de precios existente	I	Mensaje de error
23. Alta de detalle de lista de precios artículo no existente	I	Mensaje de error
24. Alta de detalle de lista de precios artículo existente sin precio	O	Mensaje de error
Administrar el catálogo de clientes		
25. Alta de nuevo cliente con datos obligatorios completos	V	
26. Alta de nuevo cliente con datos obligatorios incompletos	I	Mensaje de error
27. Alta de cliente existente	I	Mensaje de error
Registrar las solicitudes de los clientes (Ordenes de Venta)		
28. Alta de nueva solicitud con datos de venta completos	V	
29. Alta de nueva solicitud con datos obligatorios incompletos	I	Mensaje de error
30. Alta de solicitud existente	I	Mensaje de error
Confirmar Ordenes de Venta (Cambio de Status)		
31. Confirmar orden de venta con status registrada	V	
32. Confirmar orden de venta previamente confirmada o facturada	I	Mensaje de error
Notificar Ordenes de Venta procesadas		
33. Cambio de status a ordenes de venta	V	

**ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA**

confirmadas		
34. Cambio de status a ordenes de venta registradas o facturadas	I	Mensaje de error
Registro de pagos de cliente		
35. Registro de nuevo pago de cliente	V	
36. Registro de pago de cliente existente	I	Mensaje de error
37. Asociación de pago no asignado a factura por cobrar	V	
38. Asociación de pago asignado a factura por cobrar	I	Mensaje de error
39. Asociación de pago no asignado a factura saldada	I	Mensaje de error
40. Asociación de pago asignado a factura saldada	I	Mensaje de error

▪ **6.4.3 Identificación de valores a probar**

Escenario de caso de uso	Valor a probar	Resultados Esperados
Control de acceso al sistema		
1. Acceso correcto	HTTP ok user admin password XX	Carga la página de Inicio
2. Falló de página (servidor no disponible o no carga adecuadamente)	Error 505	Mensaje de error
3. Nombre o contraseña en blanco	""	Mensaje de error. Se solicita autenticación de nuevo
4. Nombre no existente	User adm.	Mensaje de error. Se solicita autenticación de nuevo
5. Contraseña no válida (no corresponde con el nombre de usuario)	User admin. Password XY	Mensaje de error. Se solicita autenticación de nuevo
Administrar los usuarios del sistema		
6. Alta usuario nuevo con datos obligatorios completos	User ID irmis Llenar campos obligatorios	
7. Alta usuario nuevo con los datos obligatorios incompletos	User ID irmis Dejar campos obligatorios en blanco	Mensaje de error. Introduzca los datos completos
8. Alta usuario existente	User ID admin.	Mensaje de error Usuario existente
Controlar el acceso de los usuarios al sistema (Contraseñas y permisos)		
9. Asignación de contraseña y permisos	User ID irmis Password XXX Acceso a Clientes, Ordenes de Venta	
10. Asignación de contraseña en blanco	User ID irmis Password ""	Mensaje de error Contraseña Inválida
11. No asignación de permisos a funciones del sistema	User ID irmis Sin acceso a funciones	Mensaje de error. Seleccione las funciones de acceso al usuario
Administrar el catálogo de productos y servicios		
12. Alta de artículo nuevo con datos obligatorios completos	Artículo: 1000 Con datos obligatorios completos	Ok
13. Alta de artículo nuevo con datos obligatorios incompletos	Artículo 1000 con datos obligatorios incompletos	Mensaje de error. Complete los campos marcados como obligatorios
14. Alta de artículo existente	Artículo 1000	Mensaje de error. Artículo

		existente
15. Baja de artículo ligado en el catálogo de listas de precios	Artículo 1000	Mensaje de error. Artículo existente en listas de precios
16. Baja de artículo ligado en el catálogo de Líneas de Orden de Venta	Artículo 1000	Mensaje de error. Artículo existente en ordenes de venta
17. Baja de artículo ligado en el catálogo de Líneas de Facturas de venta	Artículo 1000	Mensaje de error. Artículo existente en facturas de venta
18. Baja de artículo ligado en el catálogo de existencias	Artículo 1000	Mensaje de error. Artículo con existencias
Administrar el catálogo de listas de precios		
19. Alta de nueva lista de precios con datos obligatorios completos	Lista: PreciosBase con campos obligatorios completos	Ok
20. Alta de Detalle de listas de precios con artículos existentes y precio	Lista: PreciosBase Artículo 1000 \$10	Ok
21. Alta de nueva lista de precios con datos obligatorios incompletos	Lista: PreciosBase con campos obligatorios incompletos	Mensaje de error. Complete los datos marcados como obligatorios
22. Alta de lista de precios existente	Lista: PreciosBase	Mensaje de error. Lista de precios existente
23. Alta de detalle de lista de precios artículo no existente	Lista PreciosBase Artículo 100 \$10	Mensaje de error. Artículo no existe
24. Alta de detalle de lista de precios artículo existente sin precio	Lista PreciosBase Artículo 1000	Mensaje de error. Artículo debe asignarle precio de venta
Administrar el catálogo de clientes		
25. Alta de nuevo cliente con datos obligatorios completos	Cliente C001 RFC MOPO771220 datos obligatorios completos	Ok
26. Alta de nuevo cliente con datos obligatorios incompletos	Cliente C001 RFC MOPO771220 datos obligatorios incompletos	Mensaje de error. Complete los datos marcados como obligatorios
27. Alta de cliente existente	Cliente C002	Mensaje de error. Cliente

	RFC MOPO771220	existente
Registrar las solicitudes de los clientes (Ordenes de Venta)		
28. Alta de nueva solicitud con datos de venta completos	OV: OV001 OC: 1000 Cliente C001 campos obligatorios completos	Ok
29. Alta de nueva solicitud con datos obligatorios incompletos	OV001 campos obligatorios incompletos	Mensaje de error. Complete los datos marcados como obligatorios
30. Alta de solicitud existente	OV: OV002 OC: 1000 Cliente C001	Mensaje de error. Orden de Venta existente (Valida orden de compra y cliente)
Confirmar Ordenes de Venta (Cambio de Status)		
31. Confirmar orden de venta con status registrada	OV: OV001 Status Registrada	Ok
32. Confirmar orden de venta previamente confirmada o facturada	OV: OV001 Status Confirmada o Facturada	Mensaje de error. No existe ese numero de orden por confirmar
Notificar Ordenes de Venta procesadas		
33. Cambio de status a ordenes de venta confirmadas	OV: OV001 Status: Procesada	Ok
34. Cambio de status a ordenes de venta registradas o facturadas	OV: OV001 Status: Registrada o Facturada	Mensaje de error. No existe ese número de orden por procesar
Registro de pagos de cliente		
35. Registro de nuevo pago de cliente	Pago 1 Cliente C001 Status Registrado	Ok
36. Registro de pago de cliente existente	Pago 1 Cliente C001	Mensaje de error. Pago de cliente existente
37. Asociación de pago no asignado a factura por cobrar	Pago 1 Cliente C001 Factura F001 Status Asignado	Ok
38. Asociación de pago asignado a factura por cobrar	Pago 1 Cliente C001 Status Asignado	Mensaje de error. El pago ya fue asignado a una factura
39. Asociación de pago no asignado a factura saldada	Pago 2 Cliente C001 Factura F001	Mensaje de error. La factura a la cual quiere asignar el pago está saldada

	pagada	
--	--------	--

Con el conjunto escenarios de uso y valores de prueba obtenidos se podrán realizar fácilmente las pruebas del sistema que garanticen que toda la funcionalidad recogida en los casos de uso ha sido correctamente implementada. De esta manera se ha obtenido el catálogo con las pruebas necesarias para garantizar la calidad del sistema desarrollado

6.5 CONCLUSIONES MODELO DE PRUEBAS

La planificación de las pruebas del sistema en las primeras fases del desarrollo ayudan a validar los requisitos funcionales.

Esta planificación debe empezar tan pronto como se comience a disponer de los requisitos funcionales. La obtención de los casos de prueba del sistema puede sistematizarse e integrarse en el proceso de desarrollo, en paralelo a la fase de especificación de requisitos

En el presente trabajo se propuso utilizar el método "Generación de casos de prueba a partir de casos de uso", ya que a través de este modelo es posible describir los escenarios de prueba para cada caso de uso, en términos de valores concretos de entrada, los valores concretos que se deben esperar como salida, la secuencia de acciones para obtenerlos, los requerimientos de disponibilidad y desempeño que deben cumplirse.

Se inicia la definición de los casos de prueba, como pruebas de caja negra una vez definidos los casos de uso. Luego se incorpora un mayor nivel de detalle a partir del modelo de diseño. El modelo de casos de prueba es uno de los elementos fundamentales para hacer efectivo el esfuerzo de testing.

BIBLIOGRAFÍA

UML Gota a Gota. Autor Martín Fowler. Editorial (Addison Wesley) México 1999

Desarrollo y Gestión de Proyectos Informáticos. Autor Steve McConnell. Editorial McGraw Hill, España 1997

Construcción de Software Orientado a Objetos. Autor Bertrand Meyer. Editorial Prentice Hall España 1999

Análisis y Diseño de Sistemas de Información. Autor Jeffrey L Whitten y Lonnie D Bentley. Editorial McGraw Hill, Colombia 1997

Core Servlets and Java Server Pages. Autor Marty Hall. Editorial Prentice Hall y Sun Microsystems 2002

Hibernate in Action. Autores Christian Bauer y Gavin King. Editorial Manning Publications 2005

Managing Software Requirements. Autores Dean Leffingwell y Don Widdring. Editorial Addison Wesley 1999

Applying UML and Patterns, Second Edition. Autores Craig Larman. Editorial Prentice Hall 2001

URLS

<http://www.agiledata.org/essays/mappingObjects.html>

<http://www.rational.com/uml>

<http://www.omg.org/technology/documents/formal/uml.htm>

<http://jakarta.apache.org/tomcat/>

<http://dev.mysql.com/doc/mysql/en/index.html>

<http://www.hibernate.org>

<http://struts.apache.org>

<http://www.springframework.org>