



**Universidad de las Ciencias
Informáticas**

Facultad 2

**Título: Automatización de la obtención de la factura de los
abonados de ETECSA**

Trabajo de Diploma para optar por el título de Ingeniero en Ciencias
Informáticas

Autor

Frank García Albert

Tutor

Ing. Abel Olivares Castellanos

Co-Tutor

Ing. Yanerys Gourrie Fernández

Ciudad de La Habana, Junio 2011

"Año 53 de la Revolución"

Pero la juventud tiene que crear. Una juventud que no crea es una anomalía realmente.



DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo al centro de Telemática de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste, firmo la presente a los ____ días del mes de _____ del año _____

Frank García Albert
Firma Autor

Ing. Abel Olivares Castellanos
Firma del Tutor

Ing. Yanerys Gourrie Fernández
Firma del Co-Tutor

DATOS DE CONTACTO

Tutora: Ing. Abel Olivares Castellanos (aolivares@uci.cu)

Profesión: Ingeniero en Ciencias Informáticas.

Área: Facultad 2. Centro de Telemática.

Años de Graduado: 1.

Co-Tutora: Ing. Yanerys Gourrie Fernández(ygourrie@uci.cu)

Profesión: Ingeniero en Ciencias Informáticas.

Área: Facultad 2. Centro de Telemática.

Años de Graduada: 0

AGRADECIMIENTOS

Primeramente le agradezco al mayor de mis tesoros, mi mamá, que siempre ha estado ahí para mí, en las buenas y en las malas, a mi papá por apoyarme, gracias a ellos es que estoy hoy aquí, logrando un sueño que se hizo realidad.

A mis tutores Abel y Yanery por soportarme.

A Yohandy y a Roberto por su ayuda, que fue clave en mi tesis.

A Odaimy, Claudia y Alberto, y en especial a Claudia por las torticas del comedor.

A Ráiner, a Julio, a Yadier, a Elmer y muchos más que si se me quedan fuera de este papel no es porque no los recuerde.

A mis amistades del edificio 60, que nunca han perdido el sentido del humor, y espero que sigan así.

A esta Universidad que me ha formado como mejor persona y mejor profesional, para ser un Ingeniero de futuro.

DEDICATORIA

A mi mamá por ser la mejor madre del mundo, a mi papá por comprenderme, a mis hermanitos Raúl Alejandro, Franciel y Francelis.

A mis abuelos Nerida y Oracio, a mis tíos Rosa María y Carlo, a mis primos Rosita y Paulito.

A la gente de mi zona.

A mis tutores Abel y Yanery por apoyarme.

A todas mis amistades que sé que están muy contentos por mí.

RESUMEN

Debido al desarrollo vertiginoso de las comunicaciones en Cuba, al rápido ascenso de la cantidad de usuarios que utilizan la telefonía, tanto celular como convencional, la demanda de nuevos servicios y la necesidad de la automatización de los ya existentes, la Empresa de Telecomunicaciones de Cuba SA (ETECSA) desea poner en práctica diferentes servicios para mejorar la calidad de las prestaciones que brinda a sus usuarios. Entre las cuales se encuentra la creación de un sistema que garantice la automatización de la obtención de la factura de los abonados de ETECSA dándole esta tarea al proyecto “Comunicaciones Unificadas” del Centro de Telemática de la Facultad 2 de la Universidad de las Ciencias Informáticas (UCI) que en conjunto con ETECSA deberán crear un software capaz de satisfacer los servicios que debería brindar un sistema como el mencionado.

En el mundo actualmente hay varios sistemas de automatización de facturación de llamadas telefónicas. Luego de un estudio profundo de sistemas similares, extrayendo de cada uno sus mejores características y añadiendo otras que eran de interés para la nueva solución, se decidió implementar un sistema para garantizar la automatización de la obtención de la factura de los abonados de ETECSA, tomando en consideración los recursos disponibles para esto. Al software desarrollado se le realizaron las pruebas unitarias de aceptación requeridas, las cuáles cumplió satisfactoriamente, se validó además la factibilidad de la realización del proyecto con el modelo matemático COCOMO¹ II.

PALABRAS CLAVE

telefonía, automatización, facturación

¹ COCOMO: **CO**nstructive **CO**st **MO**del, Modelo Constructivo de Costes

ÍNDICE DE CONTENIDOS

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	6
1.1 Introducción.....	6
1.2 Comunicaciones Unificadas en el Mundo	6
1.2.1 Aironix	6
1.3 Asterisk.....	7
1.3.1 IVR como parte de Asterisk	7
1.4 Lenguajes, Herramientas y Tecnologías	8
1.4.1 Metodología de desarrollo: XP	9
1.4.2 Lenguaje de Modelado: BPMN.....	9
1.4.3 Lenguajes de Programación.....	10
1.4.4 Lenguaje de Etiquetado: HTML.....	11
1.4.5 Herramienta “BizAgi” para la modelación de Procesos del Negocio.....	12
1.4.6 Servidor web: Apache2.....	12
1.4.7 IDE phpDesigner 7 para la implementación del portal web	13
1.4.8 Herramienta Notepad++ 5.9 para la implementación de la IVR.	14
1.4.9 Gestor de Base de Datos: PostgreSQL 8.3	14
1.4.10 Servicios web.....	15
1.4.11 Marco de trabajo: Sauxe	15
1.5 Conclusiones.....	17
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA	18
2.1 Introducción.....	18
2.2 Descripción del proceso.....	18
2.3 Objeto de Automatización.....	18
2.4 Propuesta del Sistema.....	19
2.5 Modelación de Procesos del Sistema.....	22
2.6 Historias de Usuario	23
2.6.1 Muestra de Historia de Usuario(HU)	24
2.6.2 Historia de Usuario: Establecer Comunicación SAOF.....	24
2.6.3 Historia de Usuario: Registrar Teléfono	25
2.6.4 Historia de Usuario: Comunicar Factura Teléfono Local	25
2.6.5 Historia de Usuario: Comunicar Factura Teléfono Remoto	26
2.6.6 Historia de Usuario: Comunicar Detalles de Factura.....	27
2.6.7 Historia de Usuario: Realizar Reporte de Listado de Teléfonos	27
2.6.8 Historia de Usuario: Filtrar Reporte de Listado de Teléfonos	28
2.6.9 Historia de Usuario: Realizar Reporte de Utilización del Servicio	29
2.6.10 Historia de Usuario: Filtrar Reporte de Utilización del Servicio	30
2.6.11 Historia de Usuario: Realizar Reporte General Cantidad de Utilización del Servicio.....	31
2.6.12 Historia de Usuario: Filtrar Reporte General Cantidad de Utilización del Servicio	32
2.7 Características no Funcionales del Sistema	33
2.7.1 Apariencia o Interfaz Externa	33
2.7.2 Usabilidad	34
2.7.3 Rendimiento	34

2.7.4	Portabilidad	35
2.7.5	Seguridad	35
2.7.6	Hardware	35
2.7.7	Software	36
2.8	Conclusiones	36
CAPÍTULO 3: DISEÑO, IMPLEMENTACION DEL SISTEMA Y PRUEBA		37
3.1	Introducción	37
3.2	Arquitectura Propuesta	37
3.3	Estilo Arquitectónico del Marco de Trabajo Sauxe	38
3.4	Patrones de Diseño	39
3.4.1	GoF (Gang of Four)	39
3.4.2	GRASP	40
3.5	Diseño Físico de la Base de Datos	42
3.6	Tarjetas CRC (Cargo o Clase, Responsabilidad y Colaboración)	42
3.7	Tareas de Ingeniería	43
3.7.1	Muestra de Tarea de Ingeniería	43
3.7.2	Tarea de Ingeniería: Establecer Comunicación SAOF	43
3.7.3	Tarea de Ingeniería: Registrar Teléfono	44
3.7.4	Tarea de Ingeniería: Comunicar Factura Teléfono Local	44
3.7.5	Tarea de Ingeniería: Comunicar Factura Teléfono Remoto	45
3.7.6	Tarea de Ingeniería: Comunicar Detalles de Factura	46
3.7.7	Tarea de Ingeniería: Realizar Reporte de Listado de Teléfonos	46
3.7.8	Tarea de Ingeniería: Filtrar Reporte de Listado de Teléfonos	47
3.7.9	Tarea de Ingeniería: Realizar Reporte de Utilización del Servicio	47
3.7.10	Tarea de Ingeniería: Filtrar Reporte de Utilización del Servicio	48
3.7.11	Tarea de Ingeniería: Realizar Reporte General Cantidad de Utilización del Servicio	49
3.7.12	Tarea de Ingeniería: Filtrar por Hora	49
3.7.13	Tarea de Ingeniería: Filtrar por Día	50
3.7.14	Tarea de Ingeniería: Filtrar por Mes	51
3.8	Prueba	51
3.8.1	Pruebas Unitarias	52
3.8.2	Pruebas de Aceptación	54
3.9	Conclusiones	56
CAPITULO 4: ESTUDIO DE LA FACTIBILIDAD		57
4.1	Introducción	57
4.2	Planificación	57
4.2.1	Estimación de esfuerzo por Historia de Usuario	57
4.2.2	Iteraciones	58
4.2.3	Plan de duración de las Iteraciones	58
4.2.4	Plan de Entrega	59
4.3	Modelo matemático COCOMO II	59
4.4	Características del Proyecto	60
4.4.1	Entradas Externas	60
4.4.2	Salidas Externas	61

4.4.3	Consultas Externas.....	62
4.4.4	Archivos Lógicos Internos.....	62
4.4.5	Archivos de Interfaz Externos.....	62
4.4.6	Puntos de Función Desajustados.....	63
4.5	Cálculos de instrucciones fuentes, esfuerzo, tiempo de desarrollo, cantidad de hombres y costo..	64
4.5.1	Cálculo de instrucciones fuentes.....	64
4.5.2	Cálculo de esfuerzo nominal	64
4.6	Ajuste del esfuerzo nominal.....	66
4.6.1	Clasificado en categorías.	67
4.6.2	Cálculo del tiempo de desarrollo del software	68
4.6.3	Cálculo del costo total del proyecto.....	68
4.7	Resultados.....	69
4.8	Análisis de costo	70
4.9	Conclusiones.....	70
CONCLUSIONES		71
RECOMENDACIONES.....		72
BIBLIOGRAFÍA		73
ANEXO I CASOS DE PRUEBA DE ACEPTACIÓN.....		76
ANEXO II TABLAS DE COCOMO II.....		85

ÍNDICE DE TABLAS

Tabla 1 Resultados de pruebas realizadas al portal web	34
Tabla 2 Requerimientos de Hardware (Servidores).....	35
Tabla 3 Requerimientos de Hardware (Clientes).....	36
Tabla 4 Requerimientos de Software (Servidores)	36
Tabla 5 Requerimientos de Software (Clientes)	36
Tabla 6 Tarjeta CRC clase ListaTelefonosController.....	42
Tabla 7 Tarjeta CRC clase UtilServicioController	42
Tabla 8 Tarjeta CRC clase ReporteGeneralController	43
Tabla 9 Cantidad de pruebas unitarias realizadas.....	53
Tabla 10 Estimación de esfuerzos por Historia de Usuario.	57
Tabla 11 Plan de duración de las iteraciones.	58
Tabla 12 Duración total de las iteraciones.	59
Tabla 13 Duración total de las iteraciones.	59
Tabla 14 Entradas Externas	60
Tabla 15 Salidas Externas.....	61
Tabla 16 Consultas Externas.....	62
Tabla 17 Archivos Lógicos Internos	62
Tabla 18 Archivos de Interfaz Externos	63
Tabla 19 Puntos de función desajustados	63
Tabla 20 Características del Sistema.....	64
Tabla 21 Factores de escala	66
Tabla 22 Multiplicadores de esfuerzo.....	67
Tabla 23 Resultados	69

INTRODUCCIÓN

Con el auge de las telecomunicaciones en el mundo y la necesidad de crear sistemas más económicos, ligeros, confiables, rápidos y eficientes tanto las instituciones de software propietario como las de software libre han desarrollado y han implantado varios software, donde Asterisk (1.3) debido a la gran variedad de servicios que brinda y a su bajo costo ocupa un lugar importante entre todos ellos. Si se le une a estos beneficios el hecho de contar con un sistema telefónico como la Respuesta de Voz Interactiva o IVR (IVR del inglés: Interactive Voice Response), poder transmitir mediante VoIP² y Elastix un sistema operativo capaz de gestionar la mayoría de las funcionalidades de Asterisk entonces se cuenta con un servidor de comunicaciones unificadas que supera toda expectativa y así lo demuestra.

El desarrollo de la telefonía y el aumento exponencial de los servicios a brindar ha cambiado radicalmente la ideología a la hora de implementar e implantar los servicios. Ha sido necesario dirigirlos más hacia la automatización y a la comodidad del usuario que lo consume. Automatizar todo lo que pueda ser automatizado. ¿Por qué no lograr que cuando un usuario utilice un servicio que necesite de una interacción o de una respuesta en tiempo real, poder brindarla, en cualquier momento, las 24 horas del día, sin tener que depender del factor humano? IVR surgió para darle solución a este problema.

Cuba, aún con todas sus limitaciones, no se ha quedado atrás en materia de telefonía, se ha propuesto y ha logrado poco a poco implantar una variada gama de servicios y muchos otros están aún en desarrollo para su futura implantación. La Empresa de Telecomunicaciones de Cuba (ETECSA) junto a la Universidad de Ciencias Informáticas (UCI) ha aceptado el reto de implementar una gran variedad de servicios telefónicos entre los cuales se encuentra: un sistema de cobro revertido, un sistema de facturación y un despertador matutino.

“...La Empresa de Telecomunicaciones de Cuba, SA (ETECSA) es la entidad encargada de la prestación de los servicios públicos de telecomunicaciones mediante la operación, instalación, explotación, comercialización y mantenimiento de sus redes públicas en todo el territorio cubano...”. (1)

Actualmente en ETECSA se presta el servicio de Gestión Comercial al cual se accede mediante el número 112. A este servicio acceden los usuarios para realizar diferentes trámites enfocados a su teléfono. Este servicio es muy usado por la población ya que pueden solicitar extensiones para sus viviendas, solicitar

² VoIP, acrónimo del inglés Voice over IP (Voz sobre IP).

cambios de teléfono o de dirección, así como conocer el monto de la factura de su teléfono. Pero este servicio colapsa a partir del día 15 de cada mes ya que en esta fecha es que ETECSA emite el reporte de factura y muchos usuarios llaman solicitando el mismo. Las operadoras no pueden atender a todas las llamadas entrantes y la población no queda satisfecha con el servicio prestado ya que este es uno de los principales servicios brindados por el 112. ETECSA desea continuar incrementando los servicios que brinda a sus abonados, por el incremento de la telefonía fija en el país y por las ventajas que traería esto consigo, tanto materiales como de disponibilidad. Se hace necesario implementar un servicio que le permita a los abonados de ETECSA conocer el monto de su factura desde un teléfono fijo.

Teniendo en cuenta la **situación problemática** anterior se plantea como **problema a resolver**:

¿Cómo mejorar el proceso de obtención de la factura telefónica a los abonados de ETECSA?

A partir de este problema se define como **objeto de estudio** el proceso de gestión de información mediante IVRs para Asterisk y el **campo de acción** queda enmarcado en el proceso de gestión de información mediante IVRs para Asterisk en ETECSA.

El **objetivo general** de este trabajo es implementar un sistema que permita a los abonados de ETECSA conocer su factura desde un teléfono fijo.

Del mismo se derivan los siguientes **objetivos específicos**:

1. Desarrollar una IVR que permita informar al usuario del monto de su factura desde un teléfono fijo.
2. Desarrollar una aplicación web donde se realicen reportes del servicio.

A partir de los objetivos planteados la **Idea a Defender** es la siguiente:

Mediante la automatización de la obtención de la factura de los abonados de ETECSA se facilitará la obtención de la misma ya que los abonados podrán consultar las 24 horas del día el saldo consumido en el mes vencido.

Las **tareas investigativas** que se deben desarrollar para cumplir los objetivos son:

- ✓ Estudio y comprensión del funcionamiento de las IVRs para el desarrollo de las mismas.

- ✓ Estudio y análisis de Asterisk para poder desarrollar IVRs.

Para realizar las tareas se emplearán los siguientes **métodos**:

Métodos Teóricos:

Analítico - sintético: Este método permitirá analizar las bibliografías referentes al desarrollo de IVRs, facilitando de esta forma la extracción de los elementos más importantes. Además de que posibilitará construir el camino a seguir, a partir del análisis detallado de cada una de las IVRs analizadas.

Modelación: Este método resultará muy útil para la realización del portal web y de la IVR debido a la selección de la metodología que se utilizará, ya que en la mayoría de estas se hace muy necesaria la creación de varios modelos, pues estos permitirán una reproducción ampliada de la realidad, además de que posibilitará descubrir y estudiar nuevas relaciones y cualidades del objeto de estudio.

Método Empírico:

Observación: Este método permitirá obtener conocimiento acerca del comportamiento de las IVRs. Permitirá observar cómo funcionan, cómo se desarrollan así como las ventajas que brindan y sus especificidades de acuerdo al lugar donde se encuentren.

El trabajo está estructurado en 4 **capítulos**:

Capítulo 1 Fundamentación Teórica: En este capítulo se abordará la fundamentación teórica, en la cual se hace un estudio del software para llamadas telefónicas Asterisk. También se realizará el estudio de las herramientas y tecnologías utilizadas para el desarrollo de una IVR.

Capítulo 2 Características del Sistema: En este capítulo se realizará un análisis de las características del sistema a desarrollar, se hará hincapié en el objeto de automatización, las interfaces de usuario que integrarán el mismo, las Historias de Usuarios y las Características no Funcionales del Sistema.

Capítulo 3 Diseño, Implementación y Prueba: En este capítulo se hará hincapié en la arquitectura del sistema propuesta, se darán los detalles descriptivos de la solución propuesta y las pruebas realizadas al sistema. A través del análisis se obtendrá una visión del sistema, a través del diseño se verá cómo cumple el sistema sus objetivos y mediante las pruebas se comprobará cada funcionalidad del sistema.

Capítulo 4 Estudio de la factibilidad: En este capítulo se realizará una planificación de las iteraciones definidas así como un estudio de factibilidad para la realización del sistema propuesto mediante una estimación de tamaño, esfuerzo y planificación necesaria para llevar a cabo el mismo.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

Este capítulo se realiza un estudio de las Comunicaciones Unificadas en el mundo, Asterisk como parte inseparable de este, así como las facilidades y funcionalidades que le aportan las IVRs. Se definen las herramientas y tecnologías utilizadas para el desarrollo de una IVR y del portal web para un nuevo servicio a brindar por ETECSA y que en capítulos posteriores será la base para la construcción de la arquitectura de la misma.

1.2 Comunicaciones Unificadas en el Mundo

1.2.1 Auronix

Auronix es una empresa con 12 años de experiencia que cuenta con una amplia trayectoria en el sector de Tecnologías de la Informática y las Comunicaciones (TIC). Con una clara vocación por la innovación tecnológica, está enfocada en el desarrollo de productos para la automatización de la atención, gestión y grabación de llamadas. (2)

Auronix orienta sus conocimientos y capacidades hacia el desarrollo de soluciones globales a la medida de sus clientes. Todo esto través de su amplia red de Socios Comerciales y Distribuidores en todo el país, Centro y Sudamérica. Cuenta una base instalada de más de 2500 clientes en México. (2)

Auronix cuenta con una gama de innovadores productos:

- ✓ *Calixta–Suite para centros de contacto, incluye IVR, marcación predictiva, grabación de llamadas, manejo de email y chat.*
- ✓ *Ivervox–IVR*
- ✓ *Aurolog–Grabación de llamadas*
- ✓ *Aurodialer–Marcación automática*
- ✓ *Aurovox-CT–Mensajería unificada, servidor de fax. (2)*

1.3 Asterisk

“El Asterisk es un software completo en PBX³, actúa en el Linux y provee todas las configuraciones que esperas de un PBX y más. Asterisk hace VoIP en tres protocolos y puede interoperar con equipos de telefonía estándar básicas usando un hardware relativamente sin costo”. (3)

Este software se ha convertido en la evolución de las tradicionales centrales analógicas y digitales permitiendo integración con la tecnología más actual: VoIP. Asterisk provee servicios de buzón de voz con directorios, conferencias, IVR, llamadas en espera.

Quizás lo más interesante de Asterisk es que reconoce muchos protocolos VoIP como pueden ser SIP⁴, H.323⁵ y IAX⁶. Asterisk puede interoperar con terminales IP actuando como un registrador y como puerta de enlace entre ambos.

Ventajas:

- ✓ Tiene control de su sistema de telefonía.
- ✓ Rico y abundante en recursos.
- ✓ Es posible proveer contenido dinámico por teléfono.
- ✓ Plan de discado flexible y poderoso.
- ✓ Corre bajo Linux y es código abierto.

1.3.1 IVR como parte de Asterisk

“Una IVR basada en Asterisk permite la interacción con bases de datos internas del sistema (MySQL o PostgreSQL), externas como Oracle o Web Services. Acepta conexiones tanto de líneas analógicas y

³ PBX (Private Branch Exchange) central para extensiones telefónicas, sistema de teléfonos pertenecientes a una organización.

⁴ Protocolo de Inicialización de Sesión o SIP (Session Initiation Protocol) es un protocolo de señalización para conferencia, telefonía, presencia, notificación de eventos y mensajería instantánea a través de Internet.

⁵ H.323 es un estándar "paraguas" de la ITU (International Telecommunications Union) que describe una familia de protocolos usados para realizar el control de llamadas en una comunicación multimedia a través de redes conmutadas por paquetes. (39)

⁶ IAX es un protocolo de transporte (al igual SIP) que utiliza un único puerto UDP (4569) para el canal de señalización y en tiempo real del transporte (RTP) Protocolo de arroyos. (40)

digitales o troncales VoIP. La plataforma de hardware y software libre, dimensionada específicamente para cubrir con holgura la necesidad del cliente, es robusta y adaptable". (4)

1.3.1.1 Funcionamiento de IVR

La IVR le permite a un sistema telefónico interactuar con un usuario, por medio de una voz humana previamente grabada. La IVR es capaz de reconocer respuestas enviadas por el usuario mediante el teclado del teléfono. Este sistema es implementado la mayor parte de las veces en empresas que poseen gran flujo de llamadas telefónicas y que la demanda de la comunicación por esta vía es importantísima para el desarrollo de las mismas.

Las IVR consisten en un sistema telefónico que es capaz de recibir una llamada e interactuar con el humano a través de grabaciones de voz y el reconocimiento de respuestas simples. Cuando los usuarios llaman a una IVR, una serie de menús grabados les van guiando sobre las diferentes opciones/servicios que se van prestando. Los clientes hacen su elección contestando desde el teclado de su teléfono, en función de cada respuesta. La IVR realiza una serie de acciones sobre la base de datos. Entre éstas se incluye proporcionar determinado tipo de información recogida de la base de datos o ficheros de documentos y su lectura, traducidos a voz mediante sistemas de conversión/síntesis texto-voz.

Por su definición, los sistemas de IVR son de uso indispensable en los servicios de atención de llamadas, ya que proporcionan una descongestión necesaria en aquellos servicios de atención de llamadas saturados por consultas simples a bases de datos o información general.

1.4 Lenguajes, Herramientas y Tecnologías

En este epígrafe se detallarán las herramientas y tecnologías utilizadas para el desarrollo de la IVR y el portal web. Estas tecnologías y herramientas fueron seleccionadas de acuerdo a las ventajas que proporcionan para la implementación y puesta en marcha de la IVR y la aplicación web.

1.4.1 Metodología de desarrollo: XP

Para obtener un software con la calidad requerida debe llevarse a cabo una metodología de desarrollo, que guíe el Proceso Industrial del Software, para la realización de este software se eligió Extreme Programming (XP).

Básicamente, XP, busca dos objetivos: hacer un software con calidad y de la forma más rápida posible. De hecho, estos son los objetivos principales de cualquier metodología aplicada al desarrollo de software y a cualquier otra área en general.

Uno de los principales motivos por los cuales se escogió esta metodología es porque la línea de desarrollo de Comunicaciones Unificadas del centro Telemática, donde surgió este proyecto, establece como política que se utilice dicha metodología. Además son numerosas las ventajas que proporciona XP, entre las cuales podemos mencionar las siguientes:

Ventajas:

- ✓ *El cliente forma parte del equipo de desarrollo.*
- ✓ *Se consiguen productos usables con mayor rapidez.*
- ✓ *Se consigue integrar todo el trabajo con mucha mayor facilidad.*
- ✓ *Se atienden las necesidades del usuario con mayor exactitud. Esto se consigue gracias a las continuas versiones que se ofrecen al usuario, en las diferentes iteraciones definidas por el desarrollador y el cliente.*
- ✓ *Se obtiene código más simple y más fácil de entender, reduciendo el número de errores.*
- ✓ *Gracias al “refactoring” es más fácil el modificar los requerimientos del usuario. (5)*

1.4.2 Lenguaje de Modelado: BPMN

Business Process Modeling Notation o BPMN (en español Notación para el Modelado de Procesos de Negocio) es una notación gráfica común para cerrar la brecha de comunicación que frecuentemente se presenta entre el diseño de los procesos de negocio y su procesos públicos y privados, orquestación, coreografía, etc., así como conceptos avanzados de modelado (por ejemplo manejo de excepciones, compensación de transacciones, entre otros). (6)

Teniendo en cuenta que la metodología utilizada es XP y está dirigida a proyectos pequeños, se decidió seleccionar como lenguaje de modelado BPMN por su sencillez y su fácil aprendizaje, dado esto por el hecho de que los procesos tienen un bajo grado de complejidad, y que solo está dirigida a modelar procesos del negocio.

1.4.3 Lenguajes de Programación

Los lenguajes de programación permiten crear programas mediante un grupo de instrucciones, operadores y reglas de sintaxis, que son puestos en manos del programador para que este pueda comunicarse con los dispositivos de hardware y con el software existente en el ordenador.

1. **PHP**⁷: es un lenguaje de programación interpretado, diseñado para la creación de páginas web dinámicas. Es un lenguaje del lado del servidor pero actualmente puede ser utilizado en la creación de aplicaciones con interfaz gráfica usando las bibliotecas Qt⁸ o GTK+.⁹

Este lenguaje de programación fue utilizado para el desarrollo del portal web, específicamente en la implementación del negocio y del acceso a datos, por otra parte también es utilizado en el desarrollo de la IVR. El mencionado lenguaje es utilizado en Sauxe (1.4.11) debido a que este lo utiliza en estas dos capas (negocio y acceso a datos) para la implementación, además de presentar las siguientes ventajas:

- ✓ Está orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una Base de Datos.
- ✓ El código PHP es invisible al navegador y al usuario ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML¹⁰ al navegador. Esto hace que la programación en PHP sea segura y confiable.
- ✓ Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad.
- ✓ Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.

⁷ PHP acrónimo en inglés de Hypertext Preprocessor (procesador de hipertexto)

⁸ QT es una librería para la creación de interfaces gráficos.

⁹ GTK+ o The GIMP Toolkit es un conjunto de Bibliotecas Multiplataforma para desarrollar interfaces gráficas de usuario (GUI)

¹⁰ Hyper Text Markup Language (Lenguaje de Marca de Hipertextos).

- ✓ Permite aplicar técnicas de programación orientada a objetos.
- ✓ No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.

2. JavaScript: es un lenguaje de programación interpretado y multiplataforma, desarrollado para incrementar las funcionalidades del lenguaje HTML. Permite crear efectos especiales, interactuar con el visitante y funciones básicas que son soportadas por la mayoría de los navegadores que se utilizan en la actualidad.

Este lenguaje de programación fue utilizado para el desarrollo del portal web, específicamente en la implementación de las vistas o interfaces. El mencionado lenguaje es utilizado porque el marco de trabajo Sauxe (1.4.11) lo utiliza en esta capa (vista) para la implementación y visualización de las interfaces, además de presentar las siguientes ventajas:

- ✓ Permite elaborar aplicaciones web que simulen características de aplicaciones escritorios.
- ✓ Es un código interpretado por el navegador web.
- ✓ Es un código orientado a objetos.
- ✓ Es un código integrado a HTML.
- ✓ Reutilización de código de programación.

1.4.4 Lenguaje de Etiquetado: HTML

"Definiéndolo de forma sencilla, "HTML es lo que se utiliza para crear todas las páginas web de Internet". Más concretamente, HTML es el lenguaje con el que se "escriben" la mayoría de páginas web". (7)

HTML, es un lenguaje de programación muy sencillo que se utiliza para crear los textos y las páginas web. Está compuesto por etiquetas que definen la estructura y el formato del documento que verá el usuario a través de la web. Esas etiquetas son leídas por el navegador, es decir el programa que se utiliza para navegar, y que es el que ejecuta las funciones creadas en HTML permitiendo que puedan ser visibles en nuestra máquina. La principal ventaja que tiene HTML es la gran variedad de navegadores y exploradores que lo soportan. Debido a esto, se ha convertido en el formato más usado

para la transmisión de documentos electrónicos a través de Internet. Actualmente existe un gran número de navegadores por lo que a veces se complica que todos puedan interpretar el código HTML de la misma manera. Por esa razón es que quienes crean las páginas, chequean que pueda ser leída por los menos por los navegadores más conocidos.

1.4.5 Herramienta “BizAgi” para la modelación de Procesos del Negocio.

“BizAgi es un modelador de procesos que permite representar de forma esquemática todas las actividades y decisiones que se toman en el negocio. Con una interfaz que recuerda a Microsoft Office, BizAgi Process Modeler cumple con el estándar BPMN (Business Process Management Notation)”. (8)

El Modelador de Procesos BizAgi permite diagramar procesos de forma sencilla y muy rápida gracias a su interfaz amigable muy parecida a la de Microsoft Office. Este modelador permite validar los diagramas sin necesidad de esperar a que estén terminados completamente.

Se utiliza esta herramienta ya que es gratuita y en español. Permite modelar y visualizar los procesos, lo cual evita muchos dolores de cabeza a la hora de implementarlos, además utiliza el lenguaje de modelado seleccionado (BPMN).

“Para el correcto funcionamiento de BizAgi Process Modeler necesitas tener instalado en tu ordenador el componente Microsoft .NET Framework¹¹ 2.0 o superior”. (9)

1.4.6 Servidor web: Apache2

Apache es el servidor web por excelencia, su alto nivel de configuración, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa”. (10)

La popularidad de este software libre ampliamente reconocido en muchos ámbitos empresariales y tecnológicos es debido a las siguientes razones:

- ✓ Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- ✓ Apache es una tecnología de software libre.

¹¹ Framework: marco de trabajo

- ✓ *Es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades del servidor web Apache. Actualmente existen muchos módulos para Apache que son adaptables a este, y están ahí para que los instalemos cuando los necesitemos.*
- ✓ *Apache trabaja con gran cantidad de lenguajes script. Tiene todo el soporte que se necesita para tener páginas dinámicas.*
- ✓ *Apache te permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor.*
- ✓ *Tiene un alto nivel de configuración en la creación y gestión de logs. (10)*

Se utiliza este servidor web debido a que, además de presentar las ventajas antes mencionadas, se utiliza como paquete de instalación el WAMP5¹², donde Apache viene incluido, el cual se instala para un buen funcionamiento del marco de trabajo Sauxe (1.4.11).

1.4.7 IDE phpDesigner 7 para la implementación del portal web

“PHP Designer es un completo entorno de desarrollo y programación especialmente diseñado para los amantes de PHP, aunque también permite trabajar con comodidad en otros lenguajes de programación como HTML, XHTML¹³, CSS¹⁴ y SQL¹⁵”. (11)

“Cuenta con navegador de ficheros integrado, utilidades de corrección y autocompletado, soporte para proyectos, además de usar un práctico esquema de color para la sintaxis del código fuente que facilita enormemente la programación”. (11)

Se utiliza esta herramienta para la implementación del portal web debido a que soporta varios lenguajes como son HTML, JavaScript, PHP y CSS, además permite trabajar con el framework¹⁶ Ext JS. Permite la depuración de código y admite resaltado de sintaxis para un mejor trabajo con el mismo. Es de código libre y se puede descargar muy fácilmente desde la web.

¹² WAMP5 es un instalador para Windows; este instala una completa solución WAMP en su computador. WAMP sirve para: Windos – Apache – MySQL – PHP5

¹³ XHTML, acrónimo en inglés de eXtensible Hypertext Markup Language (lenguaje extensible de marcado de hipertexto)

¹⁴ CSS, acrónimo en inglés de Cascading Style Sheets (hojas de estilo en cascada)

¹⁵ SQL, acrónimo en inglés de Structured Query Language (lenguaje de consulta estructurado)

¹⁶ Framework: marco de trabajo.

1.4.8 Herramienta Notepad++ 5.9 para la implementación de la IVR.

“Basado en un potente componente de edición llamado Scintilla, Notepad++ está escrito en C++ y usa directamente la API¹⁷ de Win32¹⁸, lo que asegura una gran velocidad de ejecución y un menor tamaño del programa”. (12)

Se utiliza esta herramienta debido a que es un editor gratuito de software libre, ligero y su entorno de trabajo se adapta fácilmente a varios lenguajes de programación entre los que se encuentran: PHP y JavaScript. Se utiliza específicamente para la implementación de la IVR debido a que el código de la misma está es desarrollada en el lenguaje PHP. Funciona en entorno Windows y su uso está regulado por la licencia GPL¹⁹.

1.4.9 Gestor de Base de Datos: PostgreSQL 8.3

“PostgreSQL es un servidor de base de datos objeto relacional libre, ya que incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional, liberado bajo la licencia BSD²⁰. Como muchos otros proyectos de software libre, el desarrollo de PostgreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo, dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group)”. (13)

Cuenta con una serie de características, como son:

- ✓ *Documentación muy bien organizada, pública y libre, con comentarios de los propios usuarios.*
- ✓ *Comunidades muy activas, varias comunidades en castellano.*
- ✓ *Altamente adaptable a las necesidades del cliente. (13)*

Se decidió seleccionar PostgreSQL como gestor de base de datos porque el marco de trabajo Sauxe (1.4.11), para su correcto funcionamiento necesita de este gestor, además de presentar las siguientes ventajas:

¹⁷ API, acrónimo en inglés de Application Programming Interface (Interfaz de Programación de Aplicaciones)

¹⁸ Win32: Windows 32 bits

¹⁹ GPL, acrónimo en inglés de GNU General Public License (Licencia Pública General de GNU)

²⁰ BSD, acrónimo en inglés de Berkeley Software Distribution (Distribución de Software de Berkeley)

- ✓ *Puede ser utilizado, modificado y distribuido por cualquiera gratuitamente, para cualquier propósito ya sea con fines privados, comerciales o académicos.*
- ✓ *Corre en casi todos los principales sistemas operativos: Linux, Unix, Windows.*
- ✓ *Soporte nativo para los lenguajes más populares: PHP, C, C++, Perl, Python, etc. (13)*

Además de las facilidades antes mencionadas se decidió escoger este gestor porque es altamente escalable tanto en la cantidad de datos que puede manipular como en la cantidad de usuarios concurrentes que puede atender, lo cual se ajusta a las características de la IVR a desarrollar, la cual podrá atender una gran cantidad de llamadas simultáneamente.

1.4.10 Servicios web

Los servicios web, son servicios ofertados vía web, utilizados para transmitir y recibir datos por aplicaciones heterogéneas de diferentes empresas u organizaciones. Permiten una total independencia entre un servicio y su consumidor. De esta manera el cliente se abstrae de toda la tecnología que se utiliza detrás del servicio.

El lenguaje de especificación de interfaces utilizado en los servicios web es WSDL (*del inglés: Web Services Description Language*). WSDL permite especificar las operaciones y tipos de datos de un servicio web. Así, aunque el cliente y el servidor estén escritos en lenguajes distintos, pueden interactuar al utilizar un lenguaje neutral para comunicarse.

Esta tecnología se utiliza para obtener el monto de la factura del teléfono que llama, así como los detalles de la misma, y así poder informársela al usuario.

1.4.11 Marco de trabajo: Sauxe

Sauxe es el primer sistema integral de gestión cubano que sustituye la amplia gama de aplicaciones informáticas usadas para actividad empresarial en el país.

Este es el marco de trabajo sobre el cual se implementará el portal web para la visualización de los reportes del servicio. El mismo fue escogido ya que es un producto cubano, de software libre y como política del centro de Telemática las aplicaciones web se desarrollarán sobre el mismo.

Este marco incluye tres frameworks principales, los cuales son:

1. **Ext JS:** *“es una biblioteca de Java Script para el desarrollo de aplicaciones web interactivas usando tecnologías como AJAX²¹, DHTML²² y DOM²³. Fue desarrollada por Sencha. Originalmente construida como una extensión de la biblioteca YUI²⁴, en la actualidad puede usarse como extensión para las bibliotecas jQuery²⁵ y Prototype²⁶. Desde la versión 1.1 puede ejecutarse como una aplicación independiente”.* (14)

Dispone de un conjunto de componentes para incluir dentro de una aplicación web, como:

- ✓ Cuadros y áreas de texto.
- ✓ Campos para fechas.
- ✓ Campos numéricos.
- ✓ Combos.
- ✓ Pestañas.
- ✓ Menús al estilo de Windows.

Se utiliza este framework para la implementación de las interfaces o vistas, y así darle un ambiente verdaderamente agradable y atractivo a las mismas.

2. **Doctrine ORM²⁷:** El framework Doctrine es un potente mapeador de relaciones de objetos para PHP 5.2+, el cual le permite al usuario abstraerse del gestor de base de datos que se utiliza. Además es multiplataforma. Está inspirado en Hibernate, que es uno de los ORM más populares y grandes que existen. Brinda la posibilidad de escribir consultas de base de datos en un lenguaje propio llamado Doctrine Query Language (DQL).

Este framework se utiliza para la capa de acceso a datos, lo que permite abstraernos del gestor de base de datos con el cual se quiere comunicar.

²¹ AJAX, acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML)

²² DHTML: DynamicHTML (HTML Dinámico)

²³ DOM: Document Object Model

²⁴ YUI: Yahoo User Interface (Interfaz de Usuario de Yahoo)

²⁵ jQuery: framework de JavaScript

²⁶ Prototype: framework escrito en JavaScript para lograr páginas web dinámicas

²⁷ ORM, acrónimo del inglés object-relational mapping (mapeo de objetos a bases de datos)

3. Zend Framework:

Zend Framework (ZF) es un framework de código abierto para desarrollar aplicaciones web y servicios web con PHP 5. ZF es una implementación que usa código 100% orientado a objetos. La estructura de los componentes de ZF es algo único; cada componente está construido con una baja dependencia de otros componentes. Esta arquitectura débilmente acoplada permite a los desarrolladores utilizar los componentes por separado. A menudo se refiere a este tipo de diseño como "uso a voluntad". (15)

Los componentes de la biblioteca estándar de Zend Framework (ZF), aunque se pueden utilizar por separado, al combinarse conforman un robusto framework de aplicaciones web. ZF ofrece un gran rendimiento y una robusta implementación MVC, una abstracción de base de datos fácil de usar, validación y filtrado para que los desarrolladores puedan consolidar todas las operaciones usando de una manera sencilla la interfaz orientada a objetos. Se utiliza específicamente en la programación PHP.

1.5 Conclusiones

En este capítulo se tocaron temas esenciales para llevar a cabo la investigación, tales como el software Asterisk, las IVRs, el gestor de base de datos PostgreSQL y el marco de trabajo Sauxe haciendo énfasis en las necesidades que presenta nuestro servicio; arrojando como resultado la importancia que tiene llevar a cabo la puesta en marcha del servicio ya que aumentaría la amplia gama de servicios que brinda actualmente ETECSA y facilitaría la utilización de la telefonía en Cuba. Se detallaron la metodología, plataforma, tecnologías y herramientas de desarrollo utilizadas en la construcción del sistema.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

2.1 Introducción

En este capítulo se realizará un análisis de las características del sistema a desarrollar, se hará hincapié en el objeto de automatización, la arquitectura del sistema propuesta, las interfaces de usuario que integraran el mismo, las historias de usuarios y las características no funcionales del sistema.

2.2 Descripción del proceso

Actualmente si una persona desea conocer el monto de su factura por medio de un teléfono fijo, marca el número 112, le contesta una persona a la cual se le denomina operadora, esta operadora le brinda el siguiente mensaje: "Hola, buenas –días, tardes, noche-. Soy su guía. ¿En qué puedo ayudarle?", el usuario le comunica que desea conocer el monto de la factura de su teléfono, la operadora le pide al usuario el número de teléfono, este le comunica el mismo y la operadora le informa del monto de la factura del teléfono dado.

2.3 Objeto de Automatización

El proceso de la obtención de la factura de los abonados de ETECSA, así como crear reportes, los cuales podrán ser filtrados por diferentes criterios.

2.4 Propuesta del Sistema



Figura 1: Propuesta del sistema

Al ser efectuada una llamada a Asterisk mediante el número *112 se le reproduce el siguiente mensaje de voz al usuario: "Bienvenido al servicio de facturación de ETECSA". El sistema captura el id de la llamada entrante que corresponde con el id del teléfono que llama, si esta llamada proviene de una central digital este id se corresponde con el número de teléfono que realiza la llamada, si la llamada proviene de una central analógica el id se corresponde con una secuencia de caracteres los cuales no se corresponden con el número de teléfono que llama. El sistema le brinda al usuario dos opciones (1 y 2) mediante un menú de voz (menú de voz principal).

Si el usuario escoge la opción 1 y la llamada proviene de una central digital el sistema comprueba que este teléfono este registrado, si es así el sistema le pide que introduzca la clave secreta correspondiente al teléfono del que llama, el usuario introduce la clave secreta y el sistema le comunica el monto de su factura, luego el sistema le muestra al usuario un menú de voz que consta de 6 opciones (1, 2, 3, 4, 5, 6), si el usuario escoge la opción 1, 2, 3 o 4 se le comunicará el estado de la factura por concepto de llamadas locales, llamadas nacionales, llamadas internacionales o servicios suplementarios respectivamente, si el usuario escoge la opción 5 regresa al menú de voz principal, si escoge la opción 6 sale del servicio y se termina la comunicación. Si el teléfono no se encuentra registrado el sistema le advierte de ello y le pide que entre una clave secreta de 4 dígitos, la cual será su clave secreta, el usuario entra su clave secreta, el sistema le pide confirmación de la misma, el usuario confirma la clave secreta, el sistema registra el teléfono y la clave y se le informa que se ha registrado satisfactoriamente, y se retorna al menú de voz principal.

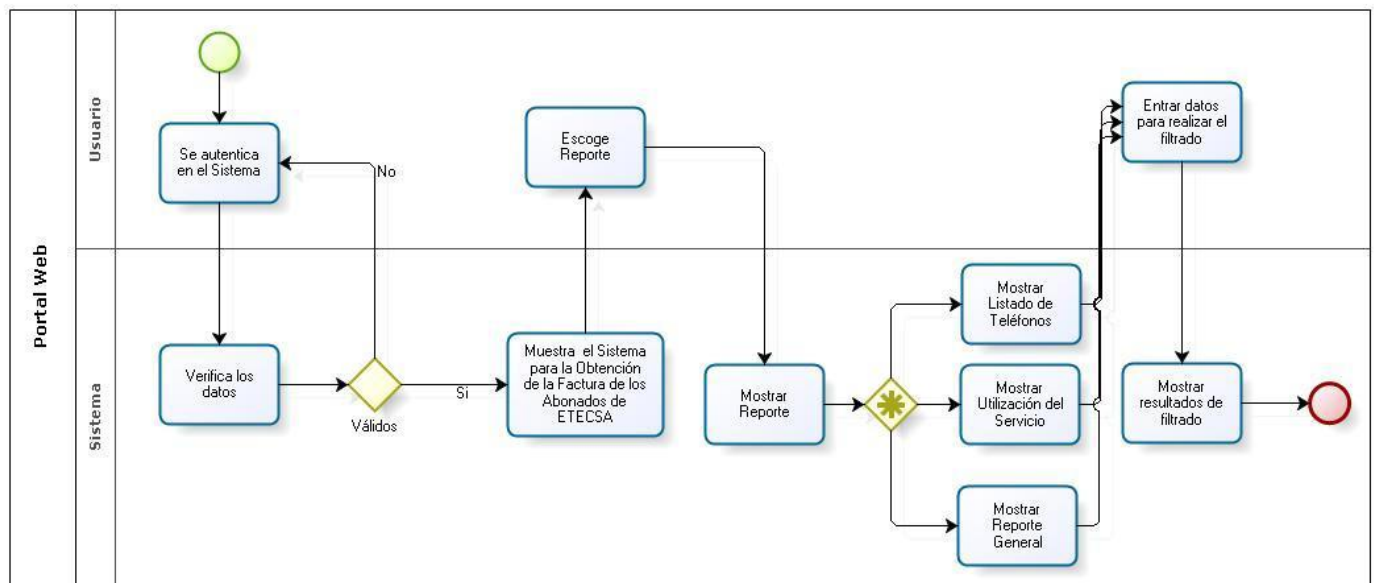
Si el usuario escoge la opción 1 y la llamada proviene de una central analógica el sistema le pide al usuario que entre su número de teléfono, el usuario entra el número de teléfono, si está registrado el sistema le pide que entre su clave secreta, el usuario entra su clave secreta y el sistema le comunica el monto de la factura de su teléfono, luego el sistema le muestra al usuario un menú de voz que consta de 6 opciones (1, 2, 3, 4, 5, 6), si el usuario escoge la opción 1, 2, 3 o 4 se le comunicará el estado de la factura por concepto de llamadas locales, llamadas nacionales, llamadas internacionales o servicios suplementarios respectivamente, si el usuario escoge la opción 5 regresa al menú principal, si escoge la opción 6 sale del servicio y se termina la comunicación. Si el teléfono no se encuentra registrado el sistema le advierte de ello y le pide que entre una clave secreta de 4 dígitos, la cual será su clave secreta, el usuario entra su clave secreta, el sistema le pide confirmación de la misma, el usuario confirma la clave secreta, el sistema registra el teléfono y la clave y se le informa que se ha registrado satisfactoriamente, y se retorna al menú de voz principal.

Si el usuario escoge la opción 2 el sistema le pide que entre el teléfono, el usuario entra el teléfono, si este se encuentra registrado el sistema le pide que entre la clave secreta correspondiente al teléfono entrado, el sistema le comunica el monto de la factura del teléfono, luego el sistema le muestra al usuario un menú de voz que consta de 6 opciones (1, 2, 3, 4, 5, 6), si el usuario escoge la opción 1, 2, 3 o 4 se le comunicará el estado de la factura por concepto de llamadas locales, llamadas nacionales, llamadas internacionales o servicios suplementarios respectivamente, si el usuario escoge la opción 5 regresa al

menú principal, si escoge la opción 6 sale del servicio y se termina la comunicación. Si el teléfono no se encuentra registrado el sistema le advierte de ello y le pide que entre una clave secreta de 4 dígitos, la cual será su clave secreta, el usuario entra su clave secreta, el sistema le pide confirmación de la misma, el usuario confirma la clave secreta, el sistema registra el teléfono y la clave, y se le informa que se ha registrado satisfactoriamente, y se retorna al menú de voz principal.

El sistema será capaz de generar y visualizar reportes del servicio mediante un portal web, en el cual el administrador podrá consultar la información relacionada con el uso del servicio, los reportes podrán ser filtrados por criterios previamente definidos en dependencia del reporte que sea, los reportes son:

- ✓ Listado de teléfonos que podrá ser filtrado por: teléfono y rango de fecha.
- ✓ Utilización del servicio que podrá ser filtrado por: teléfono, rango de fecha y rango de hora.
- ✓ Reporte general que podrá ser filtrado por: tipo de reporte y rango de fecha.



powered by
BizAgi
Process Modeler

Figura 3: Proceso del sistema portal web

2.6 Historias de Usuario

Las Historias de Usuario son la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas.

Durante esta fase, para el desarrollo del sistema se elaboraron 11 Historias de Usuario: Establecer Comunicación SAOF²⁸, Registrar Teléfono, Comunicar Factura Teléfono Local, Comunicar Factura Teléfono Remoto, Comunicar Detalles de Factura, Realizar Reporte de Listado de Teléfonos, Filtrar Reporte de Listado de Teléfonos, Realizar Reporte de Utilización del Servicio, Filtrar Reporte de Utilización del Servicio, Realizar Reporte General Cantidad de Utilización del Servicio, Filtrar Reporte General Cantidad de Utilización del Servicio, a continuación se explican cada una de ellas:

²⁸ Sistema Automatizado para la Obtención de la Factura de los Abonados de ETECSA.

2.6.1 Muestra de Historia de Usuario(HU)

Historia de Usuario	
Número: (Número de la historia de usuario incremental en el tiempo)	Nombre de Historia de Usuario: (El nombre de la historia de usuario sería para identificarlas mejor entre los desarrolladores y el cliente)
Modificación de Historia de Usuario Número: (si sufrió alguna modificación anterior)	
Usuario: (Involucrados en el desarrollo de la HU)	Iteración Asignada: (Número de la iteración)
Prioridad en negocio: (Alta / Media / Baja)	Puntos estimados: (El tiempo estimado que se demorará el desarrollo de la HU)
Riesgo en Desarrollo: (Alta / Media / Baja)	Puntos Reales: (El tiempo que se demoró en realidad el desarrollo de la HU)
Descripción: (Breve descripción de la HU)	
Observaciones: Señalamiento o advertencia del sistema.	

2.6.2 Historia de Usuario: Establecer Comunicación SAOF

Historia de Usuario	
Número: 1	Usuario: Usuario
Nombre historia: Establecer Comunicación SAOF	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Frank García Albert	
Descripción: El usuario teclea *112 para establecer comunicación con el servicio y el sistema reproduce un mensaje de Bienvenida, seguido de un menú de voz (menú principal) en el cual se le da dos opciones al usuario (1 y 2), si escoge la opción 1 se ejecuta la HU: Comunicar Factura Teléfono Local, si escoge la opción 2 se ejecuta la HU: Comunicar Factura Teléfono Remoto y si escoge una opción no válida el sistema le da la oportunidad de volver	

a realizar la entrada de la opción hasta un máximo de tres veces, si pasadas estas tres veces el número entrado sigue siendo incorrecto se termina la conexión y se cuelga.

Observaciones:

2.6.3 Historia de Usuario: Registrar Teléfono

Historia de Usuario	
Número: 2	Usuario: Usuario
Nombre historia: Registrar Teléfono	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Frank García Albert	
Descripción: El sistema le pide al usuario un PIN de 4 dígitos, el cual no se le deberá olvidar al usuario para la futura utilización del servicio. El usuario entra el PIN, si este no es válido se le da la opción al usuario de repetir la entrada hasta tres veces, si al cabo de estas 3 veces no entra un PIN válido se termina la conexión y se cuelga. Si el PIN es válido el sistema le pide confirmación del mismo, si este no coincide se le da la opción al usuario de repetir la entrada hasta tres veces, si al cabo de estas 3 veces no entra un PIN que coincida se termina la conexión y se cuelga. Si el PIN coincide se le reproduce al usuario un mensaje de confirmación y se ejecuta la HU: Establecer Comunicación SAOF.	
Observaciones:	

2.6.4 Historia de Usuario: Comunicar Factura Teléfono Local

Historia de Usuario	
Número: 3	Usuario: Usuario
Nombre historia: Comunicar Factura Teléfono Local	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto

Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Frank García Albert	
Descripción: El sistema comprueba que el <i>agi_callerid</i> ²⁹ sea válido, si no es válido se ejecuta la HU: Comunicar Factura Teléfono Remoto. Si el <i>agi_callerid</i> es válido entonces el sistema comprueba si está registrado, si no está registrado se ejecuta la HU: Registrar Teléfono, si está registrado el sistema le pide la entrada de su PIN, si el PIN es incorrecto se le da la oportunidad al usuario de entrarlo nuevamente, si pasados tres intentos el PIN sigue siendo incorrecto se cierra la conexión y se cuelga, si el PIN es correcto se le brinda al usuario la Factura del mes en curso y se ejecuta la HU: Comunicar Detalles de Factura.	
Observaciones:	

2.6.5 Historia de Usuario: Comunicar Factura Teléfono Remoto

Historia de Usuario	
Número: 4	Usuario: Usuario
Nombre historia: Comunicar Factura Teléfono Remoto	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Frank García Albert	
Descripción: El sistema le pide al usuario el número de teléfono, el usuario entra el número de teléfono, si el teléfono no es correcto se le da la oportunidad al usuario de volver a entrarlo, si pasados tres intentos el número de teléfono sigue siendo incorrecto se termina la conexión y se cuelga. Si el número de teléfono es correcto el sistema comprueba si está registrado, si no está registrado se ejecuta la HU: Registrar Teléfono, si está registrado se ejecuta la HU: Comunicar Factura Teléfono Local.	
Observaciones:	

²⁹ Identificador del teléfono que realiza la llamada, si la llamada viene de una central digital corresponde al número de teléfono, sino, es una cadena de caracteres la cual contiene una combinación de números y letras.

2.6.6 Historia de Usuario: Comunicar Detalles de Factura

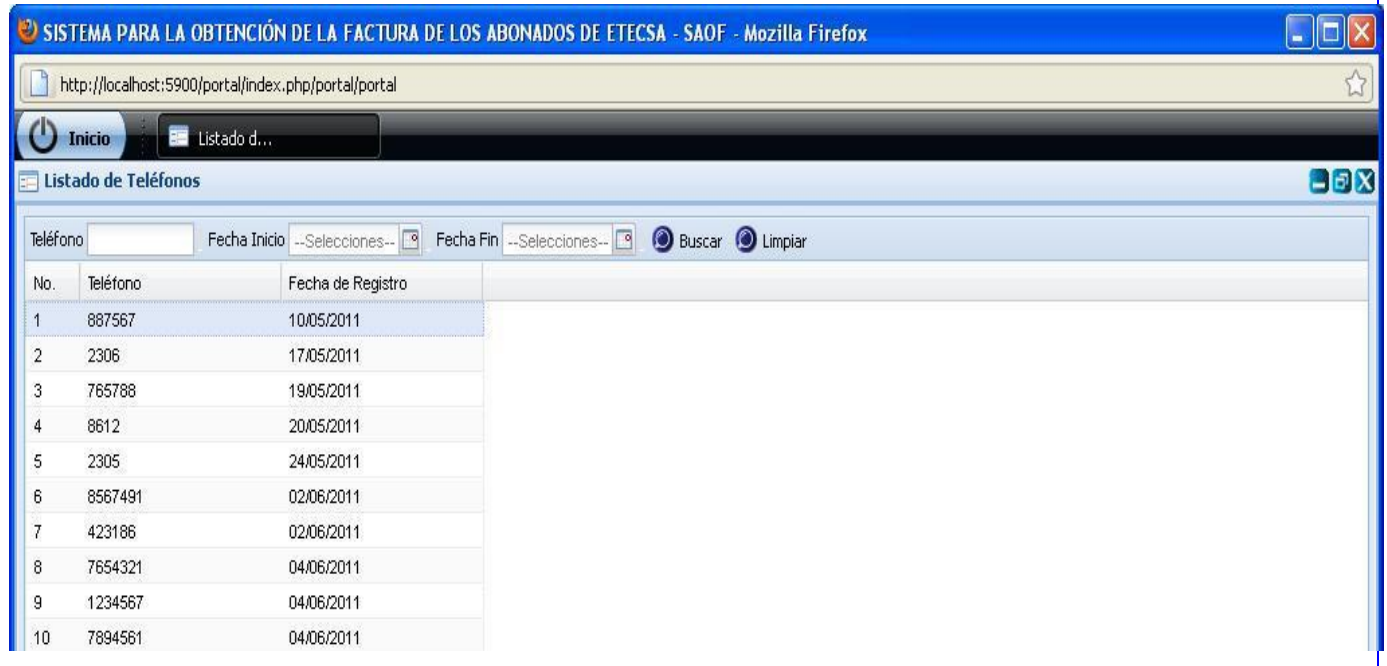
Historia de Usuario	
Número: 5	Usuario: Usuario
Nombre historia: Comunicar Detalles de Factura	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Frank García Albert	
<p>Descripción: El sistema reproduce un menú de voz (menú de detalles) el cual consta de 6 opciones (1, 2, 3, 4, 5 y 6). Si el usuario escoge la opción 1 el sistema muestra el monto de la Factura por concepto de llamadas locales, si el usuario escoge la opción 2 el sistema muestra el monto de la factura por concepto de llamadas nacionales, si el usuario escoge la opción 3 el sistema muestra el monto de la factura por concepto de llamadas internacionales, si el usuario escoge la opción 4 el sistema muestra el monto de la factura por concepto de servicios suplementarios, si el usuario escoge la opción 5 se ejecuta la HU: Establecer Comunicación SAOF, si el usuario escoge la opción 6 se reproduce el mensaje de despedida, se termina la conexión y se cuelga.</p>	
Observaciones:	

2.6.7 Historia de Usuario: Realizar Reporte de Listado de Teléfonos

Historia de Usuario	
Número: 6	Usuario: Administrador
Nombre historia: Realizar Reporte de Listado de Teléfonos	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Frank García Albert	
<p>Descripción: Se muestran los parámetros de los teléfonos:</p>	

- ✓ Número.
- ✓ Teléfono.
- ✓ Fecha de registro.

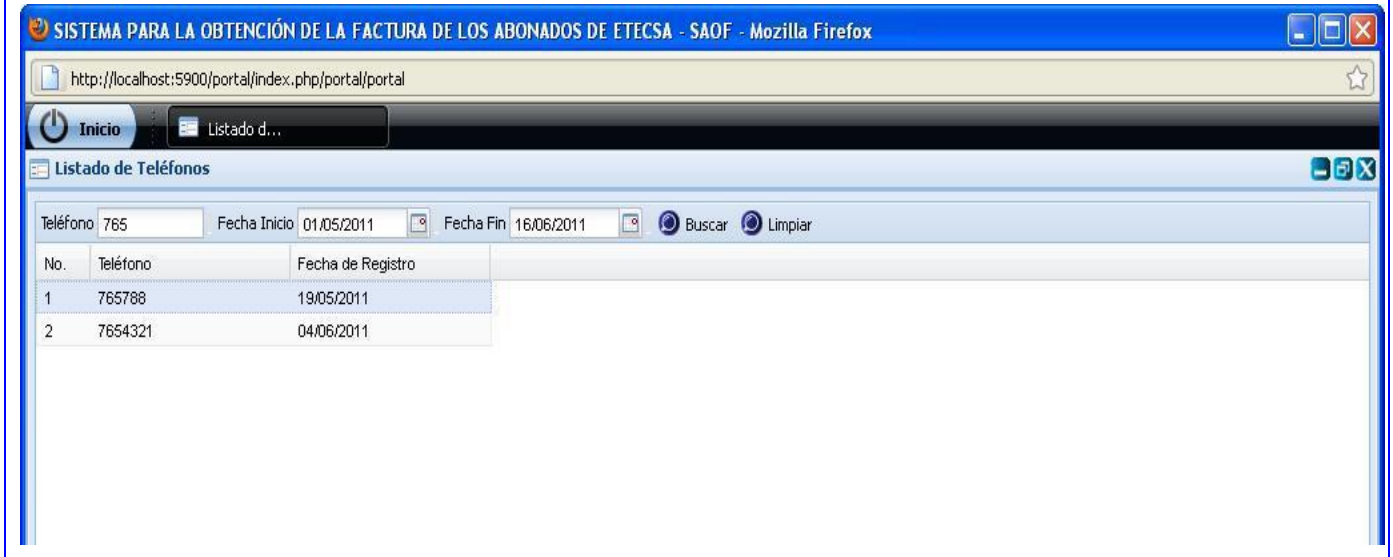
Observaciones:



2.6.8 Historia de Usuario: Filtrar Reporte de Listado de Teléfonos

Historia de Usuario	
Número: 7	Usuario: Administrador
Nombre historia: Filtrar Reporte de Listado de Teléfonos	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Frank García Albert	
Descripción: Permite filtrar el reporte por: <ul style="list-style-type: none"> ✓ Número de teléfono. ✓ Rango de fecha. 	

Observaciones: Esta Historia de Usuario solo podrá ejecutarse cuando la Historia de Usuario Realizar Reporte de Listado de Teléfonos esté ejecutada.



2.6.9 Historia de Usuario: Realizar Reporte de Utilización del Servicio

Historia de Usuario	
Número: 8	Usuario: Administrador
Nombre historia: Realizar Reporte de Utilización del Servicio	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Frank García Albert	
Descripción: Se muestran los parámetros de la utilización del servicio: <ul style="list-style-type: none"> ✓ Número. ✓ Teléfono. ✓ Saldo total. ✓ Llamadas locales. ✓ Llamadas nacionales. ✓ Llamadas internacionales. 	

- ✓ Servicios suplementarios.
- ✓ Fecha de utilización.
- ✓ Hora de utilización.

Observaciones:

The screenshot shows a web browser window titled "SISTEMA PARA LA OBTENCIÓN DE LA FACTURA DE LOS ABONADOS DE ETECSA - SAOF - Mozilla Firefox". The address bar shows "http://localhost:5900/portal/index.php/portal/portal". The page content includes a navigation bar with "Inicio" and "Utilizaci..." buttons. Below is a section titled "Utilización del Servicio" which contains a table with the following data:

No.	Teléfono	Fecha de Utilización	Hora de Utilización	Saldo Total	Llamadas Locales	Llamadas Nacionales	Llamadas Internacionales	Servicios Suplementarios
1	887567	11/05/2011	12:25:30	1	1	1	1	1
2	765788	19/05/2011	09:30:00	1	1	1	1	0
3	765788	19/05/2011	22:31:01	1	0	0	0	0
4	8612	20/05/2011	14:47:47	1	0	0	0	0
5	8612	20/05/2011	14:48:59	1	0	0	0	0
6	8612	20/05/2011	09:10:32	1	1	1	1	1
7	8612	20/05/2011	09:24:33	1	0	0	0	0
8	8612	27/05/2011	10:10:04	1	0	0	0	0
9	765788	28/05/2011	10:01:52	1	0	0	0	1
10	8612	28/05/2011	10:04:16	1	0	0	0	0

2.6.10 Historia de Usuario: Filtrar Reporte de Utilización del Servicio

Historia de Usuario	
Número: 9	Usuario: Administrador
Nombre historia: Filtrar Reporte de Utilización del Servicio	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Frank García Albert	
Descripción: Permite filtrar el reporte por: <ul style="list-style-type: none"> ✓ Teléfono. ✓ Rango de fecha. ✓ Rango de hora. 	

Observaciones: Esta Historia de Usuario solo podrá ejecutarse cuando la Historia de Usuario Realizar Reporte de Utilización del Servicio esté ejecutada.

No.	Teléfono	Fecha de Utilización	Hora de Utilización	Saldo Total	Llamadas Locales	Llamadas Nacionales	Llamadas Internacionales	Servicios Suplementarios
1	8612	20/05/2011	14:47:47	1	0	0	0	0
2	8612	20/05/2011	09:24:33	1	0	0	0	0
3	8612	20/05/2011	14:48:59	1	0	0	0	0
4	8612	20/05/2011	09:10:32	1	1	1	1	1
5	8612	27/05/2011	10:10:04	1	0	0	0	0
6	8612	28/05/2011	10:04:16	1	0	0	0	0
7	8612	02/06/2011	14:06:57	1	0	1	0	0
8	8612	02/06/2011	14:10:38	1	0	0	0	0
9	8612	13/06/2011	12:45:21	1	1	1	1	1

2.6.11 Historia de Usuario: Realizar Reporte General Cantidad de Utilización del Servicio

Historia de Usuario	
Número: 11	Usuario: Administrador
Nombre historia: Realizar Reporte General Cantidad de Utilización del Servicio	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 3
Programador responsable: Frank García Albert	
Descripción: Se muestran los parámetros del consumo por teléfono: <ul style="list-style-type: none"> ✓ Número. ✓ Hora. ✓ Saldo total. 	

- ✓ Llamadas locales.
- ✓ Llamadas nacionales.
- ✓ Llamadas internacionales.
- ✓ Servicios suplementarios.

Observaciones:



2.6.12 Historia de Usuario: Filtrar Reporte General Cantidad de Utilización del Servicio

Historia de Usuario	
Número: 11	Usuario: Administrador
Nombre historia: Filtrar Reporte General Cantidad de Utilización del Servicio	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración asignada: 3
Programador responsable: Frank García Albert	
Descripción: Se muestran los parámetros del consumo por teléfono: <ul style="list-style-type: none"> ✓ Tipo de reporte (los reportes pueden ser de los siguientes tipos: hora, día, mes). ✓ Rango de fecha. 	
Observaciones: Esta Historia de Usuario solo podrá ejecutarse cuando la Historia de Usuario Realizar Reporte	

General Cantidad de Utilización del Servicio esté ejecutada.

No.	Tiempo	Saldo Total	Llamadas Locales	Llamadas Nacionales	Llamadas Internacionales	Servicios Suplementarios
1	11:41:14	1	0	0	0	0
2	12:25:28	1	0	1	0	0
3	14:27:18	1	0	0	1	0
4	14:31:11	1	0	0	0	0
5	15:33:36	1	0	0	0	0

2.7 Características no Funcionales del Sistema

2.7.1 Apariencia o Interfaz Externa

El portal web tendrá una interfaz amigable y fácil de usar. La interfaz principal contará con una imagen de fondo en la se visualiza el nombre del producto, contará con una barra de pestañas de color negro, además cuenta con un botón “Inicio” de color azul mediante el cual se accede a los reportes dando click sobre el mismo, al dar click se le mostrará un menú “Facturación” dentro del cual aparecen tres submenús, cada uno de los cuales corresponde a un reporte distinto. En la parte superior/izquierda de cada reporte mostrado se encuentra el nombre del reporte con el cual se está interactuando. Cada reporte está formado por un grid³⁰ en el cual se mostrará la información correspondiente a cada reporte, encima del grid se mostrarán los criterios de filtrado y los botones Buscar y Limpiar los cuales serán de color azul, en la parte superior/derecha se encuentran los botones: minimizar, maximizar/restaurar y cerrar. En la parte inferior/izquierda se puede observar el número de la página en la que se encuentra. El contorno de las ventanas de los reportes será de color azul.

³⁰ grid: componente donde se visualizan la información del reporte

2.7.2 Usabilidad

A los administradores del portal web se les dará un adiestramiento básico en el uso de la aplicación. Estas personas tendrán un nivel de acceso amplio en la aplicación para poder visualizar los reportes y realizar el filtrado de los mismos.

2.7.3 Rendimiento

La eficiencia del producto estará determinada en gran medida por el aprovechamiento de los recursos que se disponen en el modelo cliente/servidor, la rapidez de las consultas de la base de datos y la velocidad del servidor Asterisk.

Por la parte del rendimiento de la IVR: “Según Digium, una planta telefónica Asterisk que se encuentra en un equipo que cuenta con un microprocesador Dual Intel Xeon, con 1.8 Ghz y 1 Gb³¹ RAM³² soporta 60 llamadas concurrentes codificando con el codec G.729³³”. (16)

Se le realizaron pruebas al portal web y las mismas arrojaron los siguientes resultados:

Tabla 1 Resultados de pruebas realizadas al portal web

Información a mostrar	Tiempo de retardo
Interfaz principal	4214 ms ³⁴
0 filas	1092 ms
36 filas	1115 ms
70 filas	1150 ms

Los resultados antes mencionados pueden variar con una variación de aproximadamente 40 ms.

³¹ Gb, acrónimo del inglés Giga byte (Unidad de medida)

³² RAM, acrónimo del inglés Random Access Memory (Memoria de Acceso Aleatorio)

³³ Códec G.729: algoritmo de compresión de datos de audio

³⁴ ms: milisegundos

2.7.4 Portabilidad

La planta telefónica Asterisk se recomienda que sea montada sobre la distribución de Linux Centus. El portal web puede estar montado tanto en una computadora con Sistema Operativo Linux o Windows. El servidor web y el servidor de base de datos pueden estar en la misma PC³⁵ sin ocasionar problema alguno.

2.7.5 Seguridad

El marco de trabajo Sauxe define niveles de acceso. Existe un superusuario³⁶ (instalación) protegido por contraseña. Se definió otro usuario que será el administrador del portal web (administrador) protegido por contraseña. El usuario administrador tendrá acceso solamente al módulo facturación, solo podrá visualizar los reportes y realizar el filtrado de los mismos. Se garantiza mediante validaciones, que el usuario que hace uso del portal web y realiza el filtrado de los reportes no introduzca caracteres extraños a través de los criterios de filtrado, y en caso de hacerlo se le advierte mediante un mensaje el error que ha cometido.

2.7.6 Hardware

Tabla 2 Requerimientos de Hardware (Servidores)

Servidores	Especificaciones
Servidor aplicaciones web	<ul style="list-style-type: none">• Procesador: 3.00 GHZ.• RAM: 1GB.• Disco duro: 160 GB.• UPS: 1.• Tarjeta de Red: 1.
Servidor de base de datos	<ul style="list-style-type: none">• Procesador: 3.00 GHZ.• RAM: 1GB.• Disco duro: 160 GB.• UPS: 1.• Tarjeta de Red: 1.

³⁵ PC: computadora personal

³⁶ superusuario: usuario que tendrá acceso a todos los modulos del marco de trabajo Sauxe y tiene permisos de modificación de los mismos

Tabla 3 Requerimientos de Hardware (Clientes)

Clientes	Especificaciones
PC cliente	<ul style="list-style-type: none">• Procesador: 1.40 GHZ.• RAM: 512 MB• Tarjeta de Red: 1.

2.7.7 Software

Tabla 4 Requerimientos de Software (Servidores)

Servidores	Especificaciones
Servidor aplicaciones web	<ul style="list-style-type: none">• Sistema Operativo: Ubuntu Server.• Servidor Web: Apache 2.0.• Librerías Adicionales: PHP 5.
Servidor de base de datos	<ul style="list-style-type: none">• Sistema Operativo: Ubuntu Server.• Sistema Gestor de Base de Datos: PostgreSQL 8.3.8.

Tabla 5 Requerimientos de Software (Clientes)

Clientes	Especificaciones
PC cliente	<ul style="list-style-type: none">• Sistema Operativo: Linux o Windows.• Navegador web: Mozilla Firefox v2.2 o superior.

2.8 Conclusiones

Tanto las características del sistema a desarrollar, los distintos requerimientos del software, darán como resultado un sistema amigable, adaptable y fácil de usar, se generaron las historias de usuarios necesarias para el correcto desarrollo del sistema.

CAPÍTULO 3: DISEÑO, IMPLEMENTACION DEL SISTEMA Y PRUEBA

3.1 Introducción

En el presente capítulo se describe el proceso de diseño a través de varios artefactos propuestos por la metodología XP. Como exige dicha metodología, el software ha sido implementado de forma iterativa, obteniendo al culminar de cada iteración un producto funcional que debe ser probado y mostrado al cliente para incrementar la visión de los desarrolladores con la opinión de este. Se exponen fundamentalmente las Tarjetas CRC, las tareas generadas por cada historia de usuario, así como las pruebas efectuadas sobre el sistema.

3.2 Arquitectura Propuesta

Diversas aplicaciones se ejecutan en un entorno cliente/servidor. Esto significa que los equipos clientes, en este caso los teléfonos que forman parte de la red telefónica, se comunican con un servidor, en este caso con el servidor Elastix, el cual proporciona servicios a los teléfonos clientes. Ejemplo de estos servicios son: teleconferencias, así como llamadas en espera.

Esta arquitectura es la que se utiliza en el servicio desarrollado. El cliente (teléfono) se comunica con el servidor (Elastix) mediante los protocolos SIP y RTP³⁷; mediante el protocolo SIP se envían las señales y mediante el protocolo RTP se transmite la voz.



Figura 4: Arquitectura Propuesta Cliente Servidor

Ventajas

³⁷ Protocolo en Tiempo Real. El objetivo de RTP es brindar un medio uniforme de transmisión sobre IP de datos que estén sujetos a las limitaciones de tiempo real (audio, video, etc.).

- ✓ Recursos centralizados: debido a que el servidor es el centro de la red, puede administrar los recursos que son comunes a todos los usuarios, por ejemplo: una base de datos centralizada se utilizaría para evitar problemas provocados por datos contradictorios y redundantes.
- ✓ Seguridad mejorada: ya que la cantidad de puntos de entrada que permite el acceso a los datos no es importante.
- ✓ Administración al nivel del servidor: ya que los clientes no juegan un papel importante en este modelo, requieren menos administración.
- ✓ Red escalable: gracias a esta arquitectura, es posible quitar o agregar clientes sin afectar el funcionamiento de la red y sin la necesidad de realizar mayores modificaciones.

3.3 Estilo Arquitectónico del Marco de Trabajo Sauxe

“Modelo Vista Controlador (MVC). Es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El estilo de llamada y retorno MVC, se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página”. (17)

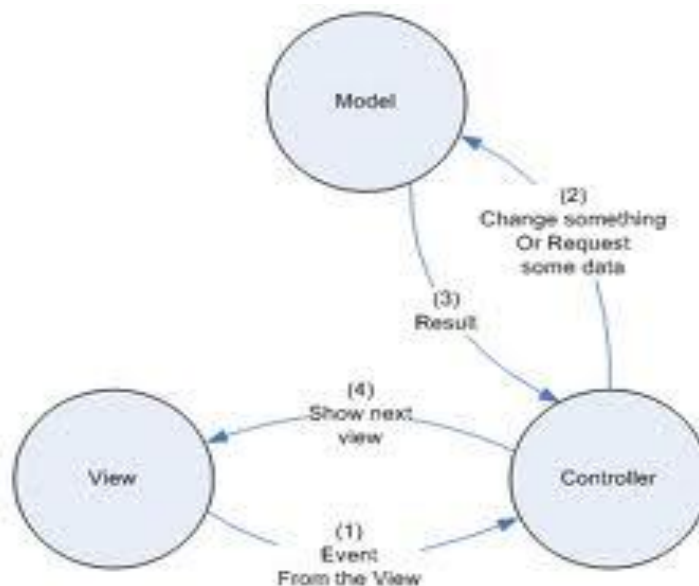


Figura 5: Estilo Arquitectónico (Modelo-Vista-Controlador)

Este estilo arquitectónico se utiliza porque el marco de trabajo utilizado (Sauxe) lo utiliza para su

implementación. En Sauxe se ve definido claramente cómo se separan en distintas capas las vistas (Vistas) de la capa de negocio (Controlador) así como de la capa de acceso a dato, donde las vistas comprenden a las interfaces que muestran la información que reciben, la cual es enviada desde las clases del negocio y envían la información entrada por el usuario hacia la capa de negocio, la capa de negocio (Controlador) comprende las clases controladoras que gestionan la información proveniente de las interfaces y envía esta información gestionada a las clases de acceso a dato, y envían la respuesta de la capa de acceso a datos hacia las interfaces después de ser gestionada, la capa de acceso a datos (Modelo) obtiene la información de la base de datos mediante consultas a partir de la información que recibió de las clases controladoras y esta información obtenida de las consultas realizadas se envía hacia las clases controladoras para su gestión.

3.4 Patrones de Diseño

“Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces. Un patrón de diseño es una solución a un problema de diseño. Para que una solución sea considerada un patrón debe poseer ciertas características. Una de ellas es que debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores. Otra es que debe ser reusable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias”. (18)

3.4.1 GoF (Gang of Four)

- 1 **Solitario:** dentro de los patrones de diseño GoF se utiliza el Solitario, por su utilidad a la hora de realizar las peticiones a la base de datos. Tiene como propósito garantizar que una clase sólo tenga una única instancia, proporcionando un punto de acceso global a la misma. Este patrón se ve implementado en la clase “Model”, que está en la siguiente dirección del marco Sauxe: PACSOFT\lib\ZendExt, donde se devuelve la conexión a la base de datos, la cual puede ser utilizada en cualquier momento sin necesidad de crear una nueva conexión. Se implementa también en las interfaces cuando se inicializan todos los mensajes mediante la función Ext.QuickTips.init().

Presenta las siguientes ventajas:

- ✓ *El acceso a la “instancia única” está más controlado, puesto que la clase Model encapsula su única instancia.*
- ✓ *Evita la creación de tantas variables globales que almacenan las instancias.*
- ✓ *Se puede crear una subclase de la clase “Model” y utilizarla si es necesario en tiempo de ejecución.*
- ✓ *Permite un número variable de instancias. (19)*

2 Fachada: *El patrón fachada se utiliza para proporcionar una interfaz unificada de alto nivel para un conjunto de clases en un subsistema, haciéndolo más fácil de usar. (20)*

En el marco de trabajo Sauxe se utiliza este patrón en las interfaces. Sauxe define una interfaz principal sobre la cual se pueden realizar diferentes funcionalidades, esta interfaz tiene elementos los cuales se pueden reutilizar en cualquier interfaz creada por el desarrollador.

3 Decorador: El patrón Decorador permite añadir funcionalidad dinámicamente a un objeto. Este patrón el marco de trabajo Sauxe lo utiliza en las interfaces en el trabajo con el framework Ext JS. La aplicación del mismo se ve reflejado por ejemplo cuando a un grid se le añade un scroll, también se evidencia cuando un mismo reporte contiene varias páginas con información.

3.4.2 GRASP

1 Bajo acoplamiento: Este patrón determina que deben existir la menor cantidad de dependencias entre las clases para que un cambio en una de estas clases afecte a la menor cantidad de clases. Dicho patrón se ve reflejado en el marco de trabajo Sauxe ya que cada clase depende solamente de las clases necesarias para lograr un buen funcionamiento del marco de trabajo. Por ejemplo cada clase del acceso a dato solo depende de las clases necesarias y suficientes para realizar su función.

2 Alta cohesión: *Cada elemento de nuestro diseño debe realizar una labor única dentro del sistema, no desempeñada por el resto de los elementos y auto-identificable. (21)*

Este patrón se ve reflejado en el marco de trabajo Sauxe ya que cada clase no tiene una sobrecarga de funciones a realizar, solo realizan las funciones necesarias y delegan en otras

clases las funciones que deben hacer estas. Un ejemplo de esto se evidencia en las clases de acceso a datos ya que cada una de estas clases solo manejan la información que les concierne a cada una de ellas.

3 Creador: *Se asigna la responsabilidad de que una clase B cree un Objeto de la clase A solamente cuando:*

- ✓ *B contiene a A.*
- ✓ *B es una agregación (o composición) de A.*
- ✓ *B almacena a A.*
- ✓ *B tiene los datos de inicialización de A (datos que requiere su constructor).*
- ✓ *B usa a A. (21)*

Este patrón se utiliza en la implementación de la IVR a la hora de instanciar la clase Conexión, que es la encargada de establecer la conexión con la base de datos, ya que cuando se crea el objeto de esta clase se le pasa los parámetros de inicialización, sin estos parámetros esta clase no podrá ser inicializada, o sea la IVR es la encargada de inicializar esta clase Conexión.

4 Experto: *La responsabilidad de realizar una labor es de la clase que tiene o puede tener los datos involucrados (atributos). Una clase, contiene toda la información necesaria para realizar la labor que tiene encomendada. (21)*

Cada una de las clases en el marco de trabajo Sauxe tiene los datos necesarios y suficientes para realizar su función dentro del sistema. Por ejemplo cada clase del negocio cuenta con los datos necesarios para gestionar la información que le corresponde y no otra información, así como las clases de acceso a dato solo manejarán los datos que le conciernen a ellas.

5 Controlador: *Asignar la responsabilidad de controlar el flujo de eventos del sistema, a clases específicas. (21)*

Este patrón, dentro del marco de trabajo Sauxe, se ve reflejado más claramente en las clases controladoras (Negocio), ya que cada una de estas controla y gestiona el flujo de datos que le corresponde a cada una de ellas. Este patrón está muy ligado al patrón alta cohesión debido a

que cada una de estas clases realiza la función para la que están destinadas, no tienen sobre carga de funciones.

3.5 Diseño Físico de la Base de Datos

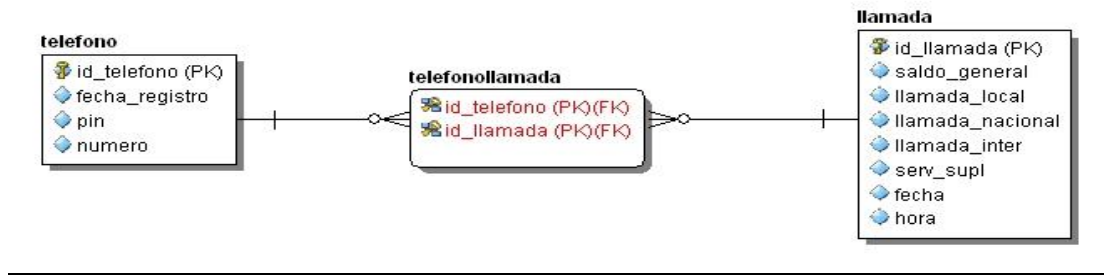


Figura 6: Diseño físico de la base de datos

3.6 Tarjetas CRC (Cargo o Clase, Responsabilidad y Colaboración)

El diagrama de tarjetas CRC es una técnica de modelado orientado a objetos que permite identificar las clases y sus responsabilidades. Las tarjetas CRC son una herramienta de ayuda al refinamiento de clases. Consiste en elaborar para cada clase una tarjeta con los siguientes datos:

- ✓ Nombre.
- ✓ Responsabilidades.
- ✓ Colaboraciones.

Las tarjetas determinan el comportamiento de cada actividad.

Tabla 6 Tarjeta CRC clase ListaTelefonosController

Clase: ListaTelefonosController	
Responsabilidad	Colaboración
Cargar Información de Teléfonos	Teléfono

Tabla 7 Tarjeta CRC clase UtilServicioController

Clase: UtilServicioController	
Responsabilidad	Colaboración
Cargar Información de Utilización del Servicio	Teléfono Llamada

Tabla 8 Tarjeta CRC clase ReporteGeneralController

Clase: ReporteGeneralController	
Responsabilidad	Colaboración
Cargar Información General de Utilización del Servicio	Llamada

3.7 Tareas de Ingeniería

Las tareas de la ingeniería son escritas por el equipo de desarrollo a partir de las historias de usuario elaboradas por el cliente, las cuales describen las tareas correspondientes a cada historia de usuario, dando un detalle más profundo para realizar una implementación de las mismas y estimando un tiempo más cercano a la realidad para cada una de ellas.

3.7.1 Muestra de Tarea de Ingeniería

Tarea de Ingeniería	
Número Tarea: [Los números deben ser consecutivos]	Número Historia de Usuario: [Número de la historia de usuario a la que pertenece la tarea]
Nombre Tarea: [Nombre que identifica a la tarea.]	
Tipo de Tarea : [Las tareas pueden ser de: Desarrollo, Corrección, Mejora, Otra(Especificar)]	Puntos Estimados: [Tiempo en semanas que se le asignará. (Estimado)]
Programador Responsable: [Nombre y Apellidos del programador]	
Descripción: [Breve descripción de la tarea.]	

3.7.2 Tarea de Ingeniería: Establecer Comunicación SAOF

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 1
Nombre Tarea: Establecer Comunicación SAOF	
Tipo de Tarea : Desarrollo	Puntos Estimados: 2

Programador Responsable: Frank García Albert

Descripción: El usuario para establecer comunicación con el servicio, debe teclear *112, una vez establecida la comunicación se captura el *agi_callerid*, inmediatamente el sistema reproduce un mensaje de bienvenida, acto seguido se le muestra el menú principal, donde se le brindan dos opciones (1 y 2), si escoge la opción 1 se le da paso a la Tarea de Ingeniería: Comunicar Factura Teléfono Local, si escoge la opción 2 se le da paso a la Tarea de Ingeniería: Comunicar Factura Teléfono Remoto. Si la opción escogida no es válida se le reproduce un mensaje de error al usuario dando la oportunidad de volver a seleccionar la opción deseada, después de 3 intentos incorrectos el sistema termina la conexión y cuelga.

3.7.3 Tarea de Ingeniería: Registrar Teléfono

Tarea de Ingeniería

Número Tarea: 2

Número Historia de Usuario: 2

Nombre Tarea: Registrar Teléfono

Tipo de Tarea : Desarrollo

Puntos Estimados: 1

Programador Responsable: Frank García Albert

Descripción: Se informa al usuario de que a continuación deberá entrar una clave secreta la cual no podrá olvidar, ya que será indispensable para sus futuras visitas al servicio, esta clave deberá ser de 4 dígitos. Se le pide al usuario que introduzca la clave, si esta clave no es de 4 dígitos y no es numérica se le muestra un mensaje al usuario alertándolo de ello y se le da nuevamente la oportunidad de introducir la clave, si pasado 3 intentos la clave sigue siendo incorrecta se termina la conexión y cuelga. Si la clave introducida por el usuario es correcta, se le pide confirmación de la misma pidiéndole que introduzca nuevamente su clave secreta, si esta confirmación no coincide con la clave original, se alerta al usuario de ello y se le da nuevamente la posibilidad de entrar su clave, si pasados 3 intentos la clave sigue siendo incorrecta se termina la conexión y cuelga. Si la confirmación es correcta se registra al usuario en la Base de Datos, se le da un mensaje de confirmación de su registro y se le da paso a la Tarea de Ingeniería: Establecer Comunicación SAOF, en caso de que existan problemas con su registro se le muestra un mensaje alertándolo de ello, y se le da la paso a la Tarea de Ingeniería: Registrar Teléfono.

3.7.4 Tarea de Ingeniería: Comunicar Factura Teléfono Local

Tarea de Ingeniería

Número Tarea: 3	Número Historia de Usuario: 3
Nombre Tarea: Comunicar Factura Teléfono Local	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Programador Responsable: Frank García Albert	
<p>Descripción: El sistema comprueba si el <i>agi_callerid</i> es un número y si está registrado en la Base de Datos de ETECSA mediante un web service, si el <i>agi_callerid</i> no es un número o el número no se encuentra registrado en la base de datos de ETECSA se le da paso a la Tarea de Ingeniería: Comunicar Factura Teléfono Remoto, si cumple con los requerimientos se le pide al usuario que entre su clave secreta, si la clave secreta no coincide con la clave real se le da la oportunidad al usuario de poder volver a entrarla nuevamente, si pasados 3 intentos la clave sigue siendo incorrecta se termina la conexión y se cuelga, si la clave es correcta se le muestra al usuario la información correspondiente al monto de su factura y se le da paso a la Tarea de Ingeniería: Comunicar Detalles de Factura.</p>	

3.7.5 Tarea de Ingeniería: Comunicar Factura Teléfono Remoto

Tarea de Ingeniería	
Número Tarea: 4	Número Historia de Usuario: 4
Nombre Tarea: Comunicar Factura Teléfono Remoto	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Programador Responsable: Frank García Albert	
<p>Descripción: Se le pide al usuario que entre el número de teléfono, se comprueba que este número de teléfono aparezca en la base de datos de ETECSA mediante un web service, si el número de teléfono no aparece se le muestra un mensaje de error alertando al usuario de ello y se le da la oportunidad de que puede entrar el número de teléfono nuevamente, si pasados 3 intentos el teléfono sigue teniendo problemas se termina la conexión y cuelga, si el teléfono si aparece se comprueba que esté registrado en el servicio, en caso que no esté registrado se le muestra al usuario un mensaje en el cual se le informa que debe estar registrado para conocer esta información y se le da paso a la Tarea de Ingeniería: Registrar Teléfono, si el teléfono está registrado se le da paso a la Tarea de Ingeniería: Comunicar Factura Teléfono Local.</p>	

3.7.6 Tarea de Ingeniería: Comunicar Detalles de Factura

Tarea de Ingeniería	
Número Tarea: 5	Número Historia de Usuario: 5
Nombre Tarea: Comunicar Detalles de Factura	
Tipo de Tarea : Desarrollo	Puntos Estimados: 2
Programador Responsable: Frank García Albert	
Descripción: El sistema le muestra un menú de voz al usuario el cual consta de 6 opciones (1, 2, 3, 4, 5 y 6), si el usuario escoge la opción 1 se le muestra la información correspondiente al monto de su factura por concepto de llamadas locales, si escoge la opción 2 se le muestra la información correspondiente al monto de su factura por concepto de llamadas nacionales, si escoge la opción 3 se le muestra la información correspondiente al monto de su factura por concepto de llamadas internacionales, si escoge la opción 4 se le muestra la información correspondiente al monto de su factura por concepto de servicios suplementarios, si escoge la opción 5 se le da paso a la Tarea de Ingeniería: Establecer Comunicación SAOF, si escoge la opción 6 se reproduce el mensaje de despedida, se termina la conexión y cuelga.	

3.7.7 Tarea de Ingeniería: Realizar Reporte de Listado de Teléfonos

Tarea de Ingeniería	
Número Tarea: 6	Número Historia de Usuario: 6
Nombre Tarea: Realizar Reporte de Listado de Teléfonos	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Programador Responsable: Frank García Albert	
Descripción: Se muestra la información de los teléfonos registrados en el servicio, el reporte está conformado por 3 columnas: <ul style="list-style-type: none">✓ No.: numero consecutivo que le corresponde.✓ Teléfono: número telefónico.✓ Fecha de registro: fecha en que se registró el teléfono.	

3.7.8 Tarea de Ingeniería: Filtrar Reporte de Listado de Teléfonos

Tarea de Ingeniería	
Número Tarea: 7	Número Historia de Usuario: 7
Nombre Tarea: Filtrar Reporte de Listado de Teléfonos	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Programador Responsable: Frank García Albert	
Descripción: La información mostrada puede ser filtrada por los siguientes parámetros: <ul style="list-style-type: none">✓ Teléfono: este campo debe contener solo números y debe estar registrado en el servicio, si no es así, se le muestra un mensaje al usuario alertándolo de ello.✓ Fecha Inicio: el valor de este campo debe ser menor o igual que el valor del campo fecha fin, si no es así, se lanza un mensaje de error alertando al usuario de ello.✓ Fecha Fin: el valor de este campo debe ser mayor o igual que el valor del campo fecha inicio, si no es así, se lanza un mensaje de error alertando al usuario de ello. A la hora de filtrar la información (Buscar) por fecha los dos campos, fecha inicio y fecha fin, deben estar llenos, si no es así, se lanza un mensaje de error alertando al usuario de ello. Si se presiona el botón Limpiar se retorna el reporte al estado inicial que tenía antes de ser filtrado.	

3.7.9 Tarea de Ingeniería: Realizar Reporte de Utilización del Servicio

Tarea de Ingeniería	
Número Tarea: 8	Número Historia de Usuario: 8
Nombre Tarea: Realizar Reporte de Utilización del Servicio	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Programador Responsable: Frank García Albert	
Descripción: Se muestra la información generada a partir de la utilización del servicio por los usuarios o abonados, el reporte está formado por 9 columnas: <ul style="list-style-type: none">✓ No.: número consecutivo que le corresponde.✓ Teléfono: número de teléfono que utiliza el servicio.✓ Fecha de utilización: fecha en que se utiliza el servicio.	

- ✓ Hora de utilización: hora en que se utiliza el servicio.
- ✓ Saldo total: este campo tendrá valor 1 si el usuario cuando utilizó el servicio quiso escuchar el monto de su factura total, o tendrá valor 0 en caso contrario.
- ✓ Llamadas locales: este campo tendrá valor 1 si el usuario cuando utilizó el servicio quiso escuchar el monto de su factura por concepto de llamadas locales, o tendrá valor 0 en caso contrario.
- ✓ Llamadas nacionales: este campo tendrá valor 1 si el usuario cuando utilizó el servicio quiso escuchar el monto de su factura por concepto de llamadas nacionales, o tendrá valor 0 en caso contrario.
- ✓ Llamadas internacionales: este campo tendrá valor 1 si el usuario cuando utilizó el servicio quiso escuchar el monto de su factura por concepto de llamadas internacionales, o tendrá valor 0 en caso contrario.
- ✓ Servicios suplementarios: este campo tendrá valor 1 si el usuario cuando utilizó el servicio quiso escuchar el monto de su factura por concepto de servicios suplementarios, o tendrá valor 0 en caso contrario.

3.7.10 Tarea de Ingeniería: Filtrar Reporte de Utilización del Servicio

Tarea de Ingeniería	
Número Tarea: 9	Número Historia de Usuario: 9
Nombre Tarea: Filtrar Reporte de Utilización del Servicio	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Programador Responsable: Frank García Albert	
<p>Descripción: La información mostrada puede ser filtrada por los siguientes parámetros:</p> <ul style="list-style-type: none"> ✓ Teléfono: este campo debe contener solo números y debe estar registrado en el servicio, si no es así, se le muestra un mensaje al usuario alertándolo de ello. ✓ Fecha inicio: el valor de este campo debe ser menor o igual que el valor del campo fecha fin, si no es así, se lanza un mensaje de error alertando al usuario de ello. ✓ Fecha fin: el valor de este campo debe ser mayor o igual que el valor del campo fecha inicio, si no es así, se lanza un mensaje de error alertando al usuario de ello. ✓ Hora inicio: el valor de este campo debe ser menor o igual que el valor del campo hora fin, si no es así, se lanza un mensaje de error alertando al usuario de ello. ✓ Hora fin: el valor de este campo debe ser mayor o igual que el valor del campo hora inicio, si no es así, se lanza un mensaje de error alertando al usuario de ello. <p>A la hora de filtrar la información (Buscar) por fecha los dos campos, fecha inicio y fecha fin, deben estar llenos, si no es así, se lanza un mensaje de error alertando al usuario de ello.</p>	

A la hora de filtrar la información (Buscar) por hora los dos campos, hora inicio y hora fin, deben estar llenos, si no es así, se lanza un mensaje de error alertando al usuario de ello.
Si se presiona el botón Limpiar se retorna el reporte al estado inicial que tenía antes de ser filtrado.

3.7.11 Tarea de Ingeniería: Realizar Reporte General Cantidad de Utilización del Servicio

Tarea de Ingeniería	
Número Tarea: 10	Número Historia de Usuario: 10
Nombre Tarea: Realizar Reporte General Cantidad de Utilización del Servicio	
Tipo de Tarea : Desarrollo	Puntos Estimados: 2
Programador Responsable: Frank García Albert	
<p>Descripción: Se muestra la información generada a partir de la utilización del servicio por los usuarios o abonados, el reporte está formado por 7 columnas:</p> <ul style="list-style-type: none"> ✓ No.: numero consecutivo que le corresponde. ✓ Tiempo: el valor de este campo va a variar en dependencia del tipo de reporte que se escoja (hora-días). ✓ Saldo total: contendrá la cantidad de veces que los usuarios escucharon el monto de su factura total. ✓ Llamadas locales: contendrá la cantidad de veces que los usuarios escucharon el monto de su factura por concepto de llamadas locales. ✓ Llamadas nacionales: contendrá la cantidad de veces que los usuarios escucharon el monto de su factura por concepto de llamadas nacionales. ✓ Llamadas internacionales contendrá la cantidad de veces que los usuarios escucharon el monto de su factura por concepto de llamadas internacionales. ✓ Servicios suplementarios: contendrá la cantidad de veces que los usuarios escucharon el monto de su factura por concepto de servicios suplementarios. <p>La interfaz se carga inicialmente con la información correspondiente al día actual ordenado por hora.</p>	

3.7.12 Tarea de Ingeniería: Filtrar por Hora

Tarea de Ingeniería

Número Tarea: 11	Número Historia de Usuario: 11
Nombre Tarea: Filtrar por Hora	
Tipo de Tarea : Desarrollo	Puntos Estimados: 2 días
Programador Responsable: Frank García Albert	
<p>Descripción: La información mostrada puede ser filtrada por los siguientes parámetros:</p> <ul style="list-style-type: none"> ✓ Reporte: para este tipo de reporte el valor de este campo será hora y la información que contendrán las columnas del reporte estará en dependencia de este tipo de reporte. ✓ Fecha inicio: el valor de este campo debe ser menor o igual que el valor del campo fecha fin, si no es así, se lanza un mensaje de error alertando al usuario de ello. ✓ Fecha fin: el valor de este campo debe ser mayor o igual que el valor del campo fecha inicio, si no es así, se lanza un mensaje de error alertando al usuario de ello. <p>A la hora de filtrar la información (Buscar) por fecha los dos campos, fecha inicio y fecha fin, deben estar llenos, si no es así, se lanza un mensaje de error alertando al usuario de ello.</p> <p>Si se presiona el botón Limpiar se retorna el reporte al estado inicial que tenía antes de ser filtrado.</p>	

3.7.13 Tarea de Ingeniería: Filtrar por Día

Tarea de Ingeniería	
Número Tarea: 12	Número Historia de Usuario: 11
Nombre Tarea: Filtrar por Día	
Tipo de Tarea : Desarrollo	Puntos Estimados: 2 días
Programador Responsable: Frank García Albert	
<p>Descripción: La información mostrada puede ser filtrada por los siguientes parámetros:</p> <ul style="list-style-type: none"> ✓ Reporte: para este tipo de reporte el valor de este campo será día y la información que contendrán las columnas del reporte estará en dependencia de este tipo de reporte. ✓ Fecha inicio: el valor de este campo debe ser menor o igual que el valor del campo fecha fin, si no es así, se lanza un mensaje de error alertando al usuario de ello. ✓ Fecha fin: el valor de este campo debe ser mayor o igual que el valor del campo fecha inicio, si no es así, se lanza un mensaje de error alertando al usuario de ello. <p>A la hora de filtrar la información (Buscar) por fecha los dos campos, fecha inicio y fecha fin, deben estar llenos, si</p>	

no es así, se lanza un mensaje de error alertando al usuario de ello.

Si se presiona el botón Limpiar se retorna el reporte al estado inicial que tenía antes de ser filtrado.

3.7.14 Tarea de Ingeniería: Filtrar por Mes

Tarea de Ingeniería	
Número Tarea: 13	Número Historia de Usuario: 11
Nombre Tarea: Filtrar por Mes	
Tipo de Tarea : Desarrollo	Puntos Estimados: 2 días
Programador Responsable: Frank García Albert	
Descripción: La información mostrada puede ser filtrada por los siguientes parámetros: <ul style="list-style-type: none">✓ Reporte: para este tipo de reporte el valor de este campo será mes y la información que contendrán las columnas del reporte estará en dependencia de este tipo de reporte.✓ Fecha inicio: el valor de este campo debe ser menor o igual que el valor del campo fecha fin, si no es así, se lanza un mensaje de error alertando al usuario de ello.✓ Fecha fin: el valor de este campo debe ser mayor o igual que el valor del campo fecha inicio, si no es así, se lanza un mensaje de error alertando al usuario de ello. A la hora de filtrar la información (Buscar) por fecha los dos campos, fecha inicio y fecha fin, deben estar llenos, si no es así, se lanza un mensaje de error alertando al usuario de ello. Si se presiona el botón Limpiar se retorna el reporte al estado inicial que tenía antes de ser filtrado.	

3.8 Prueba

Uno de los pilares fundamentales de XP es el proceso de pruebas, el cual anima a los desarrolladores a probar constantemente tanto como sea posible. Mediante esta filosofía se reduce el número de errores no detectados así como el tiempo entre la introducción de éste en el sistema y su detección. Esto contribuye a elevar la calidad de los productos desarrollados y la seguridad de los programadores a la hora de introducir cambios o modificaciones. La metodología XP divide las pruebas en dos grupos: pruebas unitarias, desarrolladas por los programadores, encargadas de verificar el código de forma automática y las pruebas de aceptación, destinadas a evaluar si al final de una iteración se obtuvo la funcionalidad requerida, además de comprobar que dicha funcionalidad sea la esperada por el cliente.

3.8.1 Pruebas Unitarias

Las pruebas unitarias son generalmente de caja negra, o sea que solo se observa el resultado obtenido de la llamada a las funciones a probar de acuerdo a los parámetros de entrada, si una función retorna al menos un valor no esperado se procede a su depuración paso a paso para detectar los errores que pueda contener. Al software se le realizaron las pruebas unitarias correspondientes a medida que se fue implementando el mismo, y los resultados de las mismas fueron satisfactorios.

Durante el transcurso del desarrollo de la aplicación se realizaron una serie de pruebas para verificar el buen funcionamiento de cada funcionalidad que se iba implementando y para corregir errores encontrados dentro del código. Prueba de ello es el siguiente fragmento de código:

```
telefono = new Ext.form.TextField({
width:80,
id:'telefono'}),

new Ext.Button({icon:perfil.dirImg+'buscar.png',iconCls:'btn',text:perfil.etiquetas.lbBtnBuscar,
handler:function(){
if((fechaDesde.getValue()==" && fechahasta.getValue()!=") || (fechaDesde.getValue()!=" &&
fechahasta.getValue()=="))
mostrarMensaje(1,'Por favor verifique nuevamente que hay campo(s) con valor(es) incorrecto(s)');
else if(fechaDesde.getValue()>fechahasta.getValue())
mostrarMensaje(1,'Por favor verifique nuevamente que la entrada de las Fechas es incorrecta');
else
{
buscarportel(telefono.getValue(), fechaDesde.getValue(), fechahasta.getValue());
}
}
```

El código antes expuesto permitía que el usuario introdujera en el criterio de filtrado “Teléfono” caracteres extraños como caracteres extraño lo que se pudo corregir con la siguiente expresión regular en JavaScript en la interfaz correspondiente y el siguiente segmento de código PHP en la clase controladora correspondiente a esta interfaz:

Expresión regular en JavaScript:

```
var validartelefono = /^[0-9]*$/;
```

```
telefono = new Ext.form.TextField({
width:80,
regex:validartelefono,
id:'telefono'}),

new Ext.Button({icon:perfil.dirIImg+'buscar.png',iconCls:'btn',text:perfil.etiquetas.lbBtnBuscar,
handler:function(){
if((fechaDesde.getValue()==' && fechahasta.getValue()!=') || (fechaDesde.getValue()!=' &&
fechahasta.getValue()=='))
mostrarMensaje(1,'Por favor verifique nuevamente que hay campo(s) con valor(es) incorrecto(s)');
else if(fechaDesde.getValue()>fechahasta.getValue())
mostrarMensaje(1,'Por favor verifique nuevamente que la entrada de las Fechas es incorrecta');
else
{
buscarportel(telefono.getValue(), fechaDesde.getValue(), fechahasta.getValue());
}
}
```

Segmento de código PHP en la clase controladora:

```
$tel = $this->_request->getPost("telefono");
$comptel = Telefono::compTel($tel);
if($this->validarTelefono($tel) == false)
throw new ZendExt_Exception('FAS01');
if($comptel == 0)
throw new ZendExt_Exception('FAS05');
```

Tabla 9 Cantidad de pruebas unitarias realizadas

Cantidad de pruebas realizadas	Cantidad de pruebas satisfactorias	Cantidad de errores encontrados	Cantidad de errores corregidos
30	25	5	5

Todos los errores se pudieron corregir a tiempo, por lo que todas las pruebas arrojaron un resultado satisfactorio.

3.8.2 Pruebas de Aceptación

Las pruebas de aceptación son pruebas de caja negra que se crean a partir de las historias de usuario. Durante las iteraciones las historias de usuarios seleccionadas serán traducidas a pruebas de aceptación. En ellas se especifican, desde la perspectiva del cliente, los escenarios para probar que una historia de usuario ha sido implementada correctamente. Una historia de usuario puede tener todas las pruebas de aceptación que necesite para asegurar su correcto funcionamiento. El objetivo final de éstas es garantizar que los requerimientos han sido cumplidos y que el sistema es aceptable. Una historia de usuario no se considera completa hasta que no ha pasado por sus pruebas de aceptación.

Durante el desarrollo de las tres iteraciones del proyecto se realizaron las pruebas de aceptación para las once historias de usuario que conforman el sistema. Todas las pruebas de aceptación arrojaron resultados satisfactorios con lo que el cliente expresa su conformidad y satisfacción con el sistema creado.

Cada caso de prueba está conformado por varias secciones las cuales se explicarán a continuación:

- ✓ **Código:** servirá como identificador de la prueba realizada, a su vez será sugerente al nombre de la prueba a la que hace referencia.
- ✓ **Número Historia de Usuario:** tendrá el número de la HU a la que hace referencia la prueba a realizar.
- ✓ **Nombre:** nombre que se le da a la prueba a realizar.
- ✓ **Descripción:** se describe la funcionalidad que se desea probar.
- ✓ **Condiciones de Ejecución:** mostrará las condiciones que deben cumplirse para poder llevar a cabo el caso de prueba.
- ✓ **Entradas / Pasos de Ejecución:** se hará la descripción de cada uno de los pasos seguidos durante el desarrollo de la prueba, se tendrá en cuenta cada una de las entradas que hace el usuario con el objetivo de ver si se obtiene el resultado esperado.
- ✓ **Resultado esperado:** se hará una breve descripción del resultado que se espera obtener con la prueba realizada.
- ✓ **Evaluación de la prueba:** acorde al resultado de la prueba realizada se emitirá una evaluación sobre la misma:
 1. Bien: cuando el resultado de la prueba es exactamente el esperado por el usuario.

2. Parcialmente bien: cuando el resultado no es completamente el esperado por el cliente o usuario de la aplicación y muestra resultados erróneos o fuera de contexto.
3. Mal: cuando el resultado de la prueba realizada genera un error de codificación en la aplicación o muestra como resultado elementos no deseados o fuera de contexto, trayendo como consecuencia que la funcionalidad requerida por el cliente no tenga resultado, lo que invalida también la HU.

Prueba #1.1 Establecer Comunicación SAOF: arrojó un resultado satisfactorio.

Prueba #2.1 Registrar Teléfono: arrojó un resultado satisfactorio.

Prueba #3.1 Comunicar Factura Teléfono Local: arrojó un resultado satisfactorio.

Prueba #4.1 Comunicar Factura Teléfono Remoto: arrojó un resultado satisfactorio.

Prueba #5.1 Comunicar Detalles de Factura: arrojó un resultado satisfactorio.

Prueba #6.1 Realizar Reporte Listado de Teléfonos: arrojó un resultado satisfactorio.

Prueba #7.1 Filtrar Reporte Listado de Teléfonos: arrojó un resultado satisfactorio.

Prueba #8.1 Realizar Reporte de Utilización del Servicio: arrojó un resultado satisfactorio.

Prueba #9.1 Filtrar Reporte de Utilización del Servicio: arrojó un resultado satisfactorio.

Prueba #10.1 Realizar Reporte General Cantidad de Utilización del Servicio: arrojó un resultado satisfactorio.

Prueba #11.1 Filtrar Reporte General Cantidad de Utilización del Servicio: arrojó un resultado satisfactorio.

Cada una de las pruebas de aceptación fue realizada por el cliente, el cual quedó muy satisfecho con el resultado obtenido de las mismas, debido esto a que todas las pruebas arrojaron un resultado satisfactorio.

Las pruebas de aceptación correspondiente a cada una de las HU se encuentran en el Anexo I.

3.9 Conclusiones

En el presente capítulo se abordó la fase de diseño de la herramienta, se obtuvo la metáfora que va a describir el sistema y se crearon las tarjetas CRC de todas las clases que intervendrán en la solución con lo que se hizo una exploración de las relaciones y colaboraciones entre ellas, lo cual contribuyó a lograr un buen diseño, se mostró la arquitectura propuesta. La arquitectura propuesta muestra que una empresa de poco presupuesto puede adquirir el hardware requerido para la instalación de la plataforma.

CAPITULO 4: ESTUDIO DE LA FACTIBILIDAD

4.1 Introducción

En la planificación del proceso de desarrollo de software es de vital importancia la estimación, la cual consiste en determinar con cierto grado de certeza los recursos necesarios para el desarrollo del mismo, ya sean recursos de hardware, software, esfuerzo, tiempo y costo. En el presente capítulo se detallan las tres iteraciones llevadas a cabo durante la etapa de construcción del sistema y se realizará un estudio de factibilidad para la realización del sistema propuesto mediante una estimación de tamaño, esfuerzo y planificación necesaria para llevar a cabo el mismo.

4.2 Planificación

4.2.1 Estimación de esfuerzo por Historia de Usuario

Se hizo un estudio con el objetivo de estimar el esfuerzo necesario para realizar cada una de las Historias de Usuario, los cuales se reflejan en la tabla que se muestra a continuación (**Tabla 17**):

Tabla 10 Estimación de esfuerzos por Historia de Usuario.

Historia de Usuario	Puntos Estimados
Establecer Comunicación SAOF	2
Registrar Teléfono	2
Comunicar Factura Teléfono Local	1
Comunicar Factura Teléfono Remoto	1
Comunicar Detalles de Factura	2
Realizar Reporte de Listado de Teléfonos	1
Filtrar Reporte de Listado de Teléfonos	1
Realizar Reporte de Utilización del Servicio	1
Filtrar Reporte de Utilización del Servicio	1
Realizar Reporte General Cantidad de Utilización del Servicio	2
Filtrar Reporte General Cantidad de Utilización del Servicio	2

4.2.2 Iteraciones

Se decidió dividir el trabajo en tres iteraciones, las cuales se detallan a continuación:

Primera Iteración: Se implementa la Historia de Usuario número uno, dos, tres, cuatro y cinco. Al concluir se contará con una primera versión (1.0) que será mostrada al cliente como versión de prueba, para comprobar si satisface sus expectativas.

Segunda Iteración: En esta iteración se implementa la Historia de Usuario número seis, siete, ocho y nueve, al terminar dicha iteración se contará con una nueva versión del producto (1.1), a la cual se le realizarán pruebas para verificar si cumple con las funcionalidades requeridas. El cliente observará dichas pruebas con el objetivo de realizar cambios en el sistema en proceso de desarrollo, si lo opina conveniente.

Tercera Iteración: En esta iteración se implementa la Historia de Usuario número diez y once, al finalizarla se contará con la primera versión del producto final (2.0) y se le harán pruebas al sistema completo para definir si cumple con todos los requerimientos necesarios.

La definición de las iteraciones y las funcionalidades que debían quedar implementadas al finalizar cada una de estas iteraciones, se realizó en conjunto con el cliente y se tuvieron en cuenta la complejidad de las funcionalidades y el grado de importancia de las mismas

4.2.3 Plan de duración de las Iteraciones

Como parte del ciclo de vida de cualquier proyecto que haga uso de XP se crea el plan de duración de las iteraciones, mostrando: las iteraciones involucradas en el proceso de desarrollo, las Historias de Usuario a desarrollar en cada iteración y el tiempo de duración estimado para cada una. El orden que poseen las Historias de Usuarios en la **Tabla 11**, será el orden a seguir para la implementación de las mismas. El tiempo de duración estimado de cada iteración se muestra en la **Tabla 12**.

Tabla 11 Plan de duración de las iteraciones.

Iteraciones	Orden de las Historias Usuario a implementar	Duración de la estimación
Iteración 1	Establecer Comunicación SAF.	2 semana
Iteración 1	Registrar Teléfono	2 semanas
Iteración 1	Comunicar Factura Teléfono Local	1 semanas
Iteración 1	Comunicar Factura Teléfono Remoto	1 semanas

Iteración 1	Comunicar Detalles de Factura	2 semanas
Iteración 2	Realizar Reporte de Listado de Teléfonos	1 semana
Iteración 2	Filtrar Reporte de Listado de Teléfonos	1
Iteración 2	Realizar Reporte de Utilización del Servicio	1 semana
Iteración 2	Filtrar Reporte de Utilización del Servicio	1 semana
Iteración 3	Realizar Reporte General Cantidad de Utilización del Servicio	2 semanas
Iteración 3	Filtrar Reporte General Cantidad de Utilización del Servicio	2 semanas

Tabla 12 Duración total de las iteraciones.

Iteraciones	Duración de la estimación
Iteración 1	8 semanas
Iteración 2	4 semanas
Iteración 3	4 semanas

4.2.4 Plan de Entrega

Tabla 13 Duración total de las iteraciones.

Iteraciones	Fecha de Finalización	Versión del Producto
1ra Iteración	4ta Semana de Marzo 2009	1.0
2da Iteración	4ta Semana de Abril 2009	1.1
3ra Iteración	3ra Semana de Mayo 2009	2.0

4.3 Modelo matemático COCOMO II.

COCOMO, propuesto y desarrollado por Barry Boehm, es uno de los modelos de estimación de costos mejor documentados y utilizados. El modelo permite determinar el esfuerzo y tiempo que se requiere en un proyecto de software a partir de una medida del tamaño del mismo expresada en el número de líneas de código que se estimen generar para la creación del producto software.

Consiste básicamente en la aplicación de ecuaciones matemáticas sobre los puntos de función sin ajustar o la cantidad de líneas de código (SLOC, Source Lines of Code) estimados para un proyecto. Estas

ecuaciones se encuentran ponderadas por ciertos factores de costo que influyen en el esfuerzo requerido para el desarrollo del software.

4.4 Características del Proyecto.

El primer paso a llevar a cabo para la estimación del proyecto consiste en la obtención de los puntos de función desajustados, los cuales están dados por la suma de cada una de las entradas, las salidas y las consultas externas del sistema, así como los archivos lógicos internos y de interfaz externos. A continuación se muestran cada una de estas características aplicadas al software en cuestión.

4.4.1 Entradas Externas.

Se definen como un proceso elemental mediante el cual ciertos datos cruzan la frontera del sistema desde afuera hacia adentro. En el caso particular de la aplicación propuesta se cuenta con tres entradas externas, especificadas en la siguiente tabla:

Tabla 14 Entradas Externas

Nombre de la entrada externa.	Cantidad de ficheros.	Cantidad de elementos de datos.	Clasificación (Simple, Media, Compleja)
Establecer Comunicación SAOF	1	2	Simple
Registrar Teléfono	1	2	Simple
Comunicar Factura Teléfono Local	1	1	Simple
Comunicar Factura Teléfono Remoto	1	1	Simple
Comunicar Detalles de Factura	1	1	Simple
Realizar Reporte de Listado de Teléfonos	1	0	Simple
Filtrar Reporte de Listado de Teléfonos	1	3	Simple
Realizar Reporte de Utilización del Servicio	1	0	Simple
Filtrar Reporte de Utilización del Servicio	1	5	Simple

Realizar Reporte General Cantidad de Utilización del Servicio	1	0	Simple
Filtrar Reporte General Cantidad de Utilización del Servicio	1	3	Simple
Total		18	Simple

4.4.2 Salidas Externas

Se definen como un proceso elemental con componentes de entrada y de salida mediante el cual datos simples y datos derivados (esto es, datos que se calculan a partir de otros datos) cruzan las fronteras del sistema desde adentro hacia afuera. Las salidas externas vinculadas al proyecto se describen a continuación.

Tabla 15 Salidas Externas

Nombre de la salida externa.	Cantidad de ficheros.	Cantidad de elementos de datos.	Clasificación (Simple, Media, Compleja)
Establecer Comunicación SAOF	1	1	Simple
Registrar Teléfono	1	2	Simple
Comunicar Factura Teléfono Local	1	2	Simple
Comunicar Factura Teléfono Remoto	1	1	Simple
Comunicar Detalles de Factura	1	2	Simple
Realizar Reporte de Listado de Teléfonos	1	1	Simple
Filtrar Reporte de Listado de Teléfonos	1	1	Simple
Realizar Reporte de Utilización del Servicio	1	1	Simple
Filtrar Reporte de Utilización del Servicio	1	1	Simple
Realizar Reporte General Cantidad de Utilización del Servicio	1	1	Simple

Filtrar Reporte General Cantidad de Utilización del Servicio	1	1	Simple
Total		14	Simple

4.4.3 Consultas Externas

Se definen como un proceso elemental con componentes de entrada y de salida donde un Actor del sistema rescata datos de uno o más Archivos Lógicos Internos o Archivos de Interfaz Externos. Los datos de entrada no actualizan ni mantienen ningún archivo (lógico interno o de interfaz externo) y los datos de salida no contienen datos derivados (es decir, los datos de salida son básicamente los mismos que se obtienen de los archivos). En este caso no fueron identificados.

Tabla 16 Consultas Externas

Nombre de la Petición.	Cantidad de ficheros.	Cantidad de elementos de datos.	Clasificación (Simple, Media, Compleja)
Total		0	

4.4.4 Archivos Lógicos Internos

Constituyen un grupo de datos relacionados lógicamente e identificables por el usuario, que residen enteramente dentro de los límites del sistema y se mantienen a través de entradas externas.

Tabla 17 Archivos Lógicos Internos

Nombre de la petición.	Cantidad de ficheros.	Cantidad de elementos de datos.	Clasificación (Simple, Media, Compleja)
Base Datos: pacsoft	1	3(tablas)	Simple.
Total		3	

4.4.5 Archivos de Interfaz Externos

Son un grupo de datos relacionados lógicamente e identificables por el usuario, que se utilizan solamente para fines de referencia. Los datos residen enteramente fuera de los límites del sistema y se mantienen

por las Entradas Externas de otras aplicaciones, es decir, cada Archivo de Interfaz Externo es un Archivo Lógico Interno de otra aplicación. En este caso fue identificado un Archivos de interfaz externa.

Tabla 18 Archivos de Interfaz Externos

Nombre de la interfaz externa.	Cantidad de ficheros.	Cantidad de elementos de datos.	Clasificación (Simple, Media. Compleja)
Servicio web	1	5	Simple.
Total		5	

4.4.6 Puntos de Función Desajustados

La siguiente tabla está basada en las características del sistema anteriormente expuestas, el producto de la cantidad existente de cada una de ellas y el peso correspondiente a las mismas dan como resultado final, los puntos de función desajustados pertenecientes al proyecto.

Tabla 19 Puntos de función desajustados

Elementos.	Simple.		Medio.		Complejo.		Subtotal.
	No.	Peso.	No.	Peso.	No.	Peso.	
Entradas externas.	18	3	0	4	0	6	54
Salidas externas.	14	4	0	5	0	7	56
Consultas externas.	0	3	0	4	0	6	0
Fichero lógico interno.	3	7	0	10	0	15	21
Fichero interfaz externo	5	5	0	7	0	10	25
Total (UFP):							156

Se encuentra la tabla de peso del factor de complejidad. [Anexo II]

4.5 Cálculos de instrucciones fuentes, esfuerzo, tiempo de desarrollo, cantidad de hombres y costo.

4.5.1 Cálculo de instrucciones fuentes

Una vez obtenida la cantidad total de puntos de función desajustados pertenecientes al proyecto, se procede a calcular la cantidad de instrucciones fuentes del mismo, para lo cual se utiliza la ecuación que se muestra seguidamente:

$$SLOC = UFP \times ratio$$

Donde:

SLOC: Cantidad de instrucciones fuente (12095).

UFP: Puntos de función desajustados. (156)

ratio: Conversión de puntos de función desajustados a líneas de código para el lenguaje PHP (59).

Partiendo de la ecuación anterior se obtiene el siguiente resultado:

$$SLOC = 156 \times 59 \quad SLOC = 9204 = 9.2 \text{ KSLOC}$$

Tabla 20 Características del Sistema

Características	Valor
Puntos de función desajustados.	156
Lenguaje (PHP).	59
Instrucciones fuentes por puntos de función.	9204 SLOC
Instrucciones fuentes.	9.2 KSLOC

4.5.2 Cálculo de esfuerzo nominal

Posteriormente se procede al cálculo del esfuerzo nominal, ecuación que se toma como base tanto en el método de diseño preliminar, al cual se hace alusión anteriormente, como en el modelo Post arquitectura, ambos definidos por COCOMO II.

$$PM_{Nominal} = Ax(Size)^B$$

$$PM_{Nominal} = 2.94 + (9.2)^{1.00}$$

$$PM_{Nominal} = 27.04 \text{ meses/hombre}$$

Donde:

$PM_{Nominal}$: Esfuerzo nominal requerido en meses – hombres.

$Size$: Tamaño estimado del software en Puntos de Función sin Ajustar. (KSLOC)

A : Constante que se utiliza para capturar los efectos multiplicativos en el esfuerzo requerido de acuerdo al crecimiento de tamaño del software. (2.94).

B : Constante denominada Factor escalar y su valor está dado por la resultante de los aspectos positivos sobre los negativos que presenta el proyecto.

$$B = 0.91 + 0.01 x (W_i)$$

$$B = 0.91 + 0.01 x 9.15$$

$$B = 1.002$$

Donde:

W_i : Variables escalares que indican las características que el proyecto presenta en lo que a su complejidad y entorno de desarrollo se refiere.

1 Precedentes (PREC)

El factor de precedencia (PREC) toma en cuenta el grado de experiencia previa en relación al producto a desarrollar, tanto en aspectos organizacionales como en el conocimiento del software y hardware a utilizar.

2 Flexibilidad de desarrollo (FLEX)

El factor de flexibilidad (FLEX) considera el nivel de exigencia en el cumplimiento de los requerimientos preestablecidos, plazos de tiempos y especificaciones de interfaz.

3 Cohesión del equipo (TEAM)

El factor de escala denominado Cohesión del Equipo tiene en cuenta las dificultades de sincronización entre los participantes del proyecto: usuarios, clientes y desarrolladores. Estas dificultades pueden surgir por diferencias culturales, dificultad en la conciliación de objetivos, falta de experiencia y familiaridad con el trabajo en equipo.

4 Solución de Riesgo (RESL)

Este factor involucra aspectos relacionados al conocimiento de los ítems de riesgo crítico y al modo de abordarlos dentro del proyecto.

5 Madurez del proceso (PMAT)

El procedimiento para determinar el PMAT es establecer el porcentaje de cumplimiento de cada una de las Áreas evaluando el grado de cumplimiento de las metas correspondientes.

La siguiente tabla muestra los valores asignados a cada una de estas variables:

Tabla 21 Factores de escala

Nombre.	Valor.	Justificación.
PREC	1.24	Existen varios proyectos similares a nivel internacional, pero no se encontró ninguno a nivel nacional.
FLEX	1.01	Cuenta con alta flexibilidad en cuanto a los requerimientos establecidos inicialmente.
TEAM	1.10	El equipo de desarrollo presenta una alta cohesión.
RESL	4.24	No se identificaron grandes riesgos.
PMAT	1.56	Se cuenta con la experiencia necesaria como para que el software cumpla con las funcionalidades requeridas.
Total(SF)	9.15	

Se encuentra la tabla de factores de escala. **[Anexo II]**.

4.6 Ajuste del esfuerzo nominal.

El esfuerzo calculado anteriormente es un valor nominal y debe ser ajustado en base a las características del proyecto para lo cual se tiene un conjunto de Multiplicadores de Esfuerzo (MEi) que representan las características del proyecto y expresan su impacto en el desarrollo total del producto de software.

4.6.1 Clasificado en categorías.

El esfuerzo nominal de desarrollo de un proyecto de software se ajusta para una mejor estimación mediante factores que se clasifican en cuatro áreas: Producto, Plataforma, Personal y Proyecto.

Los 7 Multiplicadores de Esfuerzo son:

Del Producto: Se refieren a las restricciones y requerimientos sobre el producto a desarrollar.

RCPX: Confiabilidad y Complejidad del producto.

RUSE: Requerimiento de Reusabilidad.

De la Plataforma: Estos factores analizan la complejidad de la plataforma subyacente.

PDIF: Dificultad de la Plataforma.

Del Personal: Estos factores están referidos al nivel de habilidad que posee el equipo de desarrollo.

PERS: Aptitud del Personal.

PREX: Experiencia del Personal.

Del Proyecto: Estos factores se refieren a las condiciones y restricciones bajo las cuales se lleva a cabo el proyecto.

FCIL: Facilidades.

SCED: Cronograma de Desarrollo Requerido.

Tabla 22 Multiplicadores de esfuerzo

Nombre	Valor	Justificación
RCPX	1.74	La complejidad del sistema es alta así como su confiabilidad.
RUSE	0.95	La parte del código que se pretende reutilizar es muy poca.
PDIF	1.00	Uso de memoria y almacenamiento normal, plataforma estable.
PREX	1.00	La experiencia del personal en cuanto a la utilización del lenguaje y herramientas es intermedia.
PERS	0.63	La aptitud del personal es alta.
FCIL	0.73	Se utilizaron entornos de desarrollo integrados, herramientas de modelación y automatización que facilitan el trabajo.
SCED	1.00	El tiempo estimado para el desarrollo del producto se empleó

		correctamente.
Total(EM)	0.76	

Se encuentra la tabla de multiplicadores de esfuerzo del modelo del diseño temprano. [Anexo I].

$$PM_{ajustado} = PM_{Nominal} \times (ME_t)$$

$$PM_{ajustado} = 27.04 \text{ meses/hombre} \times 0.76$$

$$PM_{ajustado} = 20.55 \text{ meses/hombre}$$

4.6.2 Cálculo del tiempo de desarrollo del software

El tiempo requerido para el desarrollo del proyecto está dado por la siguiente ecuación:

$$TDEV = C \times (PM_{ajustado})^F$$

$$TDEV = 3.64 \times 20.55^{0.26}$$

$$TDEV = 7.99$$

Donde:

C: Constante con valor 3.64.

$$PM_{ajustado} = 20.55 \text{ meses/hombre}$$

F: 0.26

Calcular

$$F = D + 0.2 \times 0.01 \times SF$$

$$F = 0.24 + 0.2 \times 0.01 \times 9.15$$

$$F = 0.26$$

Donde:

D : Constante cuyo valor es 0.24.

SF : Valor de los factores de escala.

4.6.3 Cálculo del costo total del proyecto

Para el cálculo del costo total correspondiente al proyecto en cuestión, COCOMO II propone la siguiente ecuación:

$$C = CHM \times PM$$

$$C = 338 \times 27.04$$

$$C = 9139,52$$

CALCULAR

Donde:

C: Costo total.

CHM: Costo teniendo en cuenta salario de todos los obreros, el cual se calcula por la siguiente ecuación.

$$CHM = CH \times sal$$

$$CHM = 3.38 \times 100.$$

$$CHM = 338$$

CALCULAR

Donde:

sal : Salario medio por cada trabajador.

CH : Cantidad de personas destinadas al proyecto, lo que es calculado a través de la ecuación:

$$CH = \frac{PM}{TDEV}$$

$$CH = \frac{27.04}{7.99}$$

$$CH = 3.38$$

4.7 Resultados

En la siguiente tabla se muestran los resultados obtenidos luego de haber efectuado todos los cálculos para determinar el costo y esfuerzo requeridos por el proyecto.

Tabla 23 Resultados

Cálculo	Valor
Esfuerzo	27.04 meses/hombres
Tiempo de desarrollo	8 meses

Cantidad de hombres	1 hombres
Salario medio	100 pesos
Costo	9139.52

4.8 Análisis de costo

El desarrollo de un producto siempre tiene un costo de producción, el cual debe ser justificado en base a los beneficios reportados por el mismo. El sistema que se propone en este trabajo no conlleva a grandes gastos, puesto que solo es influyente el salario de los desarrolladores, por lo cual se concluye que su implementación es factible. Esto se debe en gran medida a la utilización del componente api, y a las herramientas libres que no requieren el pago de alguna licencia.

4.9 Conclusiones

En el presente capítulo se elaboró un plan de entrega de cada iteración y se realizó un análisis de factibilidad de la solución propuesta, arribando a la conclusión de que es viable su desarrollo comparando los costos de producción con los beneficios reportados por su puesta en funcionamiento.

CONCLUSIONES

Con la implantación del nuevo producto se les permitirá a los usuarios o abonados de ETECSA conocer su factura desde un teléfono fijo. Se realizó una aplicación, mediante la cual un administrador será capaz de visualizar reportes y estadísticas de la utilización del servicio así como de los usuarios suscritos al servicio. La calidad del servicio, así como la cantidad de llamadas concurrentes que se puedan atender, estará dada únicamente por las prestaciones de hardware que posea el servidor y la cantidad de canales asignados al mismo.

Para la realización de la IVR fue necesario realizar un estudio de las características, cualidades, así como el funcionamiento de las principales IVRs a nivel internacional, para de esta forma obtener un producto final con la mejor calidad.

Con la construcción de la misma, se logró poner en práctica los conocimientos adquiridos (Ingeniería de Software, Programación, Telecomunicaciones, entre otros), además de conocimientos que fueron adquiriéndose durante el paso por la Universidad y el desarrollo del trabajo.

Después de terminados la IVR y el portal web, y realizadas las pruebas de modo satisfactorio se concluye que los objetivos propuestos para el presente proyecto han sido cumplidos satisfactoriamente, incluyéndose una serie de recomendaciones que deben tenerse en cuenta para el trabajo futuro.

RECOMENDACIONES

A continuación se lista una serie de recomendaciones con el objetivo de realizar un seguimiento a este trabajo:

- Implantar el servicio en ETECSA.
- Integrar el portal web de gestión de reportes a los demás sistemas desarrollados para ETECSA en el marco de trabajo Sauxe.
- Mejorar la IVR para que se pueda consumir el saldo de aquellos teléfonos que se cobran totalmente en Divisas.
- Darle la posibilidad al usuario de poder escuchar los mensajes en otro idioma (Inglés).

BIBLIOGRAFÍA

1. **Sierra Quesada, Msc Carlos E.** *LA GESTION DEL CONOCIMIENTO EN ETECSA*. Ciudad de la Habana : s.n., 2004.
2. **Auronix.** Auronix. [En línea] 2011. [Citado el: 5 de 12 de 2010.] <http://www.auronix.com.mx/>.
3. **Sheets, Kris, y otros, y otros.** *Asterisk en Español*. Abril del 2005.
4. **Voxdata Comunicaciones IP.** Voxdata Comunicaciones IP. *Sistemas IVR basados en Asterisk: Recomendaciones de diseño para una mejor atención al cliente*. [En línea] 2010. [Citado el: 14 de Enero de 2011.] <http://www.voxdata.com.ar/voxiivr.html>.
5. **Cortizo Pérez, José Carlos, Expósito Gil, Diego y Ruiz Leyva, Miguel.** Scribd. *eXtreme Programming*. [En línea] 2011. [Citado el: 20 de 1 de 2011.] <http://es.scribd.com/doc/51712448/1/EXTREME-PROGRAMMING>.
6. **Academic.** Los diccionarios y las enciclopedias sobre el Académico. *Business Process Modeling Notation*. [En línea] 2000 - 2010. <http://www.esacademic.com/dic.nsf/eswiki/194876>.
7. **Eguíluz Pérez, Javier.** *Introducción a XHTML*. s.l. : <http://www.librosweb.es/xhtml/index.html>, 17 de diciembre de 2008.
8. **Rowman, Danny.** Buenas Tareas. *Manual De Bizagi*. [En línea] 2009. [Citado el: 20 de 1 de 2011.] <http://www.buenastareas.com/ensayos/Manual-De-Bizagi/1362471.html>.
9. **WProgramas.com.** BizAgi Process Modeler. *BizAgi Process Modeler*. [En línea] 2009. [Citado el: 25 de 1 de 2011.] <http://bizagi-process-modeler.wprogramas.com/>.
10. **Ciberaula.** Una Introducción a APACHE. *Una Introducción a APACHE*. [En línea] 2010. [Citado el: 1 de 2 de 2011.] http://linux.ciberaula.com/articulo/linux_apache_intro/.
11. **Freira García, Ivan.** Mi Mundo al Revés. *PHPDesigner 2008 v6.2.4 Professional | Multilenguaje | Final | Full*. [En línea] 12 de 2010. [Citado el: 20 de 1 de 2011.] <http://ivanfreiregarcia.wordpress.com/2010/12/08/phpdesigner-2008-v6-2-4-professional-multilenguaje-final-full/>.
12. **Administrator.** Alójate.com. *Notepad ++ 5.9.1: Completo editor de texto y código fuente gratuito*. [En línea] 2 de 6 de 2011. [Citado el: 10 de 6 de 2011.] <http://www.alojate.com/blogs/9627/notepad-5-9-1-completo-editor-de-texto-y-codigo-fuente-gratuito/>.
13. **Aliaga Ibarra, Antonio y Miani Flores, Marcos Agustín.** I.E.S. San Vicente. *PostgreSQL*. [En línea] 21 de 1 de 2008. [Citado el: 5 de 2 de 2011.] <http://www.iessanvicente.com/colaboraciones/postgresql.pdf>.
14. **Nando.** Expresión Binaria. *Expresión Binaria*. [En línea] 18 de 3 de 2011. [Citado el: 10 de 2 de 2011.] <http://www.expresionbinaria.com/glosario/ext-js/>.
15. **doczf.mikaelkael.fr/1.10/en/.** Zend Framework 1.10.8 Manual. [En línea] 2005 - 2010. [Citado el: 20 de 2 de 2011.] <http://doczf.mikaelkael.fr/1.10/es/introduction.overview.html>.
16. **Dunayevich, Julián, y otros, y otros.** *Asterisk*. 2005.
17. **EcuRed.** Patrón Modelo Vista Controlador. *Patrón Modelo Vista Controlador*. [En línea] 9 de 6 de 2011. [Citado el: 12 de 6 de 2011.] http://www.ecured.cu/index.php/Patr%C3%B3n_Modelo_Vista_Controlador.
18. **BuenasTareas.** Identificación De Patrones De Diseño. *Identificación De Patrones De Diseño*. [En línea] 2011. [Citado el: 20 de 4 de 2011.] <http://www.buenastareas.com/ensayos/Identificaci%C3%B3n-De-Patrones-De-Dise%C3%B1o/873678.html>.
19. **Gamma, Erich, y otros, y otros.** *Patrones de Diseño*. Madrid : Pearson Educación S.A., 2003.

20. **Universidad del Valle.** Escuela de Ingeniería de Sistemas y Computación. *Patrones de Diseño*. [En línea] 2011. [Citado el: 10 de 5 de 2011.] <http://eisc.univalle.edu.co/cursos/web/material/750092M/80/DS2-CLASE-DISENO-GoF.pdf>.
21. **Canales Mora, Roberto.** Patrones de GRASP. [En línea] 22 de 12 de 2003. [Citado el: 20 de 5 de 2011.] <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=grasp>.
22. **BizAgi.** Bizagi Process Modeler. *Bizagi Process Modeler*. [En línea] 2011. http://www.bizagi.com/index.php?option=com_content&view=article&id=95&Itemid=107&lang=es..
23. **Palosanto Solutions.** Elastix. [En línea] 2006-2011. [Citado el: 4 de Mayo de 2011.] <http://www.elastix.org/es>.
24. ETECSA. *Empresa de Telecomunicaciones de Cuba S.A.* [En línea] 2010. [Citado el: 1 de Marzo de 2011.] <http://www.etecsa.cu/index.php?se=inicio>.
25. **Landívar, Edgar.** *Comunicaciones Unificadas con Elastix - Volumen 1 & 2*. 2008-2009.
26. **Van Meggelen, Jim, Madsen, Leif y Smith, Jared.** *Asterisk: El Futuro de la Telefonía*. s.l. : O'Reilly.
27. **Yanerys Gourrie, Gerardo Fernández.** *TeleIdentificador Personal: Plugin para el navegador WEB Mozilla Firefox y cliente de correo Mozilla Thunderbird para el servicio TIP*. La Habana : UCI, 2010.
28. **Digium, Inc.** Asterisk. [En línea] 2010. [Citado el: 05 de Abril de 2011.] <http://www.asterisk.org/>.
29. **VOIP-Info.org LLC.** VOIP-Info. [En línea] 2003-2010. [Citado el: 25 de Abril de 2011.] <http://www.voip-info.org/wiki/view/IVR>.
30. **Larman, Craig.** *UML y Patrones*. México : s.n., 1999.
31. **Visconti, Marcello.** *Fundamentos de la Ingeniería de Software*. 2006.
32. **Bourdon, Romain.** ¿Qué es WAMP5? [En línea] 2011. <http://aclibre.org/archives/Wamp5/index.html>.
33. **Jeff.** Entorno cliente/servidor. [En línea] 2008. <http://es.kioskea.net/contents/cs/csintro.php3>.
34. **Montero Puñales, Nicolás y Acosta Aparicio, Luis Eduardo.** *Sistema de Reportes y Facturación de PBX Asterisk*. Ciudad Habana : Universidad de las Ciencias Informáticas, 2009.
35. **Morgavi, Alfonsina y Kizskurno, Ernesto.** Pruebas de Aceptación, el camino hacia Clientes satisfechos. [En línea] 2008. <http://www.sg.com.mx/sg08/downloads/presentaciones/SG08/Ingenieria%20de%20software/SE02%20-%20Pruebas%20de%20Aceptacion.pdf>.
36. **Prieto, Félix.** Programación III.I.T.I. de Sistemas. [En línea] 2008/2009. http://www.infor.uva.es/~felix/datos/priii/tr_patrones-2x4.pdf.
37. **Salazar, Erick y Aponte, Anais.** Sistemas de Programas (CI-3711). [En línea] 1999. <http://ldc.usb.ve/~teruel/ci3711/patron3a/index.html#controlador>.
38. **Recursos VoIp.** Recursos VoIP. [En línea] 2011. <http://www.recursosvoip.com/protocolos/sip.php>.
39. **Universidad Nacional Autónoma de México.** Universidad Nacional Autónoma de México. [En línea] 2011. <http://vnoc.unam.mx/es/preguntas-frecuentes/7-videoconferencia-h323/44-protocolos-h323>.
40. **Ige, Salomon.** Protocolos de VoIP: IAX (El "Inter-Asterisk eXchange" Protocolo). [En línea] 2008. <http://voip-mundo.blogspot.com/2008/03/protocolos-de-voip-iax-el-inter.html>.
41. **Ares, Roberto.** Protocolos de señalización. [En línea] 2011. <http://www.monografias.com/trabajos16/telefonía-senalizacion/telefonía-senalizacion.shtml#protocolo>.
42. **Astersik-ES.** Asterisk-ES. [En línea] 2011. [Citado el: 10 de 1 de 2011.] http://comunidad.asterisk-es.org/index.php?title=Introduccion_a_Asterisk.

43. **EDITORIAL CURSOS LUIS BONILLA.** Introducción a JavaScript. [En línea] 2011.
<http://www.cursopaginasweb.net/curso-paginas-web/introduccion-javascript.htm>.
44. **Guardado, Iván.** Utilizando Doctrine como ORM en PHP. [En línea] 6 de 7 de 2010. [Citado el: 20 de 2 de 2011.] <http://web.ontuts.com/tutoriales/utilizando-doctrine-como-orm-en-php/>.

ANEXO I CASOS DE PRUEBA DE ACEPTACIÓN

Muestra de Prueba de Aceptación

Caso de Prueba Aceptación	
Código: identificador de la prueba realizada.	Número Historia de Usuario: número de la HU a la que hace referencia la prueba a realizar.
Nombre: nombre que se le da a la prueba a realizar.	
Descripción: describe la funcionalidad que se desea probar.	
Condiciones de Ejecución: condiciones que deben cumplirse para poder llevar a cabo el caso de prueba.	
Entrada/Pasos de ejecución: descripción de cada uno de los pasos seguidos durante el desarrollo de la prueba.	
Resultado Esperado: breve descripción del resultado que se espera obtener con la prueba realizada.	
Evaluación de la Prueba: evaluación de la prueba realizada.	

Prueba #1.1 Establecer Comunicación SAOF.

Caso de Prueba Aceptación	
Código: 1.1	Número Historia de Usuario: 1
Nombre: Establecer Comunicación SAOF.	
Descripción: Prueba para verificar la comunicación de los usuarios con el servicio.	
Condiciones de Ejecución: El servicio debe estar activado.	
Entrada/Pasos de ejecución: El usuario inserta desde su teléfono el código para acceder al servicio.	
Resultado Esperado: Reproducir el Mensaje de Bienvenida y acto seguido reproducir un menú de voz en el cual se le dan dos opciones al usuario (1 y 2).	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #2.1 Registrar Teléfono.

Caso de Prueba Aceptación	
Código: 2.1	Número Historia de Usuario: 2
Nombre: Registrar Teléfono.	
Descripción: Prueba para verificar que el usuario se registre satisfactoriamente.	
Condiciones de Ejecución: El usuario no puede estar registrado en el servicio.	
Entrada/Pasos de ejecución: El usuario entra su clave secreta y la confirmación de la misma.	
Resultado Esperado: El usuario se registra satisfactoriamente y reproducir un mensaje confirmando ello.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #3.1 Comunicar Factura Teléfono Local

Caso de Prueba Aceptación	
Código: 3.1	Número Historia de Usuario: 3
Nombre: Comunicar Factura Teléfono Local.	
Descripción: Prueba para verificar que se le muestra al usuario la información correspondiente al monto de su factura total satisfactoriamente.	
Condiciones de Ejecución: El usuario debe estar registrado en el servicio.	
Entrada/Pasos de ejecución: El usuario entra su clave secreta.	
Resultado Esperado: Se le muestra al usuario el monto de su factura total.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #4.1 Comunicar Factura Teléfono Remoto

Caso de Prueba Aceptación	
Código: 4.1	Número Historia de Usuario: 4

Nombre: Comunicar Factura Teléfono Remoto.
Descripción: Prueba para verificar que se le muestra al usuario la información correspondiente al monto de la factura total de un teléfono remoto satisfactoriamente.
Condiciones de Ejecución: El teléfono debe estar registrado en el servicio.
Entrada/Pasos de ejecución: El usuario entra el número de teléfono.
Resultado Esperado: Se le muestra al usuario el monto de la factura total del teléfono entrado.
Evaluación de la Prueba: Prueba satisfactoria.

Prueba #5.1 Comunicar Detalles de Factura

Caso de Prueba Aceptación	
Código: 5.1	Número Historia de Usuario: 5
Nombre: Comunicar Detalles de Factura.	
Descripción: Prueba para verificar que se le muestra al usuario la información detallada de su factura total, que se desglosa en: factura por llamadas locales, factura por llamadas nacionales, factura por llamadas internacionales, factura por servicios suplementarios, si desea regresar al Menú Principal o terminar la conexión.	
Condiciones de Ejecución: El teléfono debe estar registrado en el servicio y el usuario ya debe conocer su factura total.	
Entrada/Pasos de ejecución: El usuario debe escoger una opción de las mostradas, y en dependencia de la opción escogida se le brindará la información correspondiente a esta opción.	
Resultado Esperado: Se le muestra al usuario el monto de la factura por concepto de llamadas locales, llamadas nacionales, llamadas internacionales, servicios suplementarios, regresar al Menú Principal o terminar la conexión en dependencia de la opción escogida.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #6.1 Realizar Reporte de Listado de Teléfonos

Caso de Prueba Aceptación	
Código: 6.1	Número Historia de Usuario: 6

Nombre: Realizar Reporte de Listado de Teléfonos.
Descripción: Prueba para verificar la funcionalidad Realizar Reporte de Listado de Teléfonos.
Condiciones de Ejecución: El teléfono debe estar registrado en el servicio.
Entrada/Pasos de ejecución: En el menú Facturación se escoge la opción Listado de Teléfonos.
Resultado Esperado: Se muestra una interfaz con la información de los teléfonos registrados en el servicio.
Evaluación de la Prueba: Prueba satisfactoria.

Prueba #7.1 Filtrar Reporte de Listado de Teléfonos

Caso de Prueba Aceptación	
Código: 7.1	Número Historia de Usuario: 7
Nombre: Filtrar Reporte de Listado de Teléfonos.	
Descripción: Prueba para verificar la funcionalidad Filtrar Reporte de Listado de Teléfonos.	
Condiciones de Ejecución: El administrador llena los campos para realizar el filtrado.	
Entrada/Pasos de ejecución: Introducir el teléfono y presionar el botón Buscar.	
Resultado Esperado: Se muestra la información correspondiente, en dependencia del teléfono entrado.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #7.1 Filtrar Reporte de Listado de Teléfonos

Caso de Prueba Aceptación	
Código: 7.1	Número Historia de Usuario: 7
Nombre: Filtrar Reporte de Listado de Teléfonos.	
Descripción: Prueba para verificar la funcionalidad Filtrar Reporte de Listado de Teléfonos.	
Condiciones de Ejecución: El administrador llena los campos para realizar el filtrado.	
Entrada/Pasos de ejecución: Introducir Fecha Inicio y Fecha Fin, y presiona el botón Buscar.	
Resultado Esperado: Se muestra la información correspondiente, en dependencia del rango de fecha	

entrado.
Evaluación de la Prueba: Prueba satisfactoria.

Prueba #7.1 Filtrar Reporte de Listado de Teléfonos

Caso de Prueba Aceptación	
Código: 7.1	Número Historia de Usuario: 7
Nombre: Filtrar Reporte de Listado de Teléfonos.	
Descripción: Prueba para verificar la funcionalidad Filtrar Reporte de Listado de Teléfonos.	
Condiciones de Ejecución: El administrador llena los campos para realizar el filtrado.	
Entrada/Pasos de ejecución: Introducir el Teléfono, Fecha Inicio y Fecha Fin, y presiona el botón Buscar.	
Resultado Esperado: Se muestra la información correspondiente, en dependencia del teléfono y el rango de fecha entrado.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #8.1 Realizar Reporte de Utilización del Servicio

Caso de Prueba Aceptación	
Código: 8.1	Número Historia de Usuario: 8
Nombre: Filtrar Reporte de Utilización del Servicio.	
Descripción: Prueba para verificar la funcionalidad Realizar Reporte de Utilización del Servicio.	
Condiciones de Ejecución: Al menos un teléfono debe haber utilizado el servicio.	
Entrada/Pasos de ejecución: En el menú Facturación se escoge la opción Utilización del Servicio.	
Resultado Esperado: Se muestra la información de correspondiente a la utilización del servicio.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #9.1 Filtrar Reporte de Utilización del Servicio

Caso de Prueba Aceptación

Código: 9.1	Número Historia de Usuario: 9
Nombre: Filtrar Reporte de Utilización del Servicio.	
Descripción: Prueba para verificar la funcionalidad Filtrar Reporte de Utilización del Servicio.	
Condiciones de Ejecución: El administrador llena los campos para realizar el filtrado.	
Entrada/Pasos de ejecución: Introducir el Teléfono y presiona el botón Buscar.	
Resultado Esperado: Se muestra la información correspondiente, en dependencia del teléfono entrado.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #9.1 Filtrar Reporte de Utilización del Servicio

Caso de Prueba Aceptación	
Código: 9.1	Número Historia de Usuario: 9
Nombre: Filtrar Reporte de Utilización del Servicio.	
Descripción: Prueba para verificar la funcionalidad Filtrar Reporte de Utilización del Servicio.	
Condiciones de Ejecución: El administrador llena los campos para realizar el filtrado.	
Entrada/Pasos de ejecución: Introducir la Fecha Inicio, la Fecha Fin y presiona el botón Buscar.	
Resultado Esperado: Se muestra la información correspondiente, en dependencia del rango de fecha entrado.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #9.1 Filtrar Reporte de Utilización del Servicio

Caso de Prueba Aceptación	
Código: 9.1	Número Historia de Usuario: 9
Nombre: Filtrar Reporte de Utilización del Servicio.	
Descripción: Prueba para verificar la funcionalidad Filtrar Reporte de Utilización del Servicio.	

Condiciones de Ejecución: El administrador llena los campos para realizar el filtrado.
Entrada/Pasos de ejecución: Introducir la Hora Inicio, la Hora Fin y presiona el botón Buscar.
Resultado Esperado: Se muestra la información correspondiente, en dependencia del rango de hora entrado.
Evaluación de la Prueba: Prueba satisfactoria.

Prueba #9.1 Filtrar Reporte de Utilización del Servicio

Caso de Prueba Aceptación	
Código: 9.1	Número Historia de Usuario: 9
Nombre: Filtrar Reporte de Utilización del Servicio.	
Descripción: Prueba para verificar la funcionalidad Filtrar Reporte de Utilización del Servicio.	
Condiciones de Ejecución: El administrador llena los campos para realizar el filtrado.	
Entrada/Pasos de ejecución: Introducir el Teléfono, la Hora Inicio, la Hora Fin y presiona el botón Buscar.	
Resultado Esperado: Se muestra la información correspondiente, en dependencia del teléfono y el rango de hora entrado.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #9.1 Filtrar Reporte de Utilización del Servicio

Caso de Prueba Aceptación	
Código: 9.1	Número Historia de Usuario: 9
Nombre: Filtrar Reporte de Utilización del Servicio.	
Descripción: Prueba para verificar la funcionalidad Filtrar Reporte de Utilización del Servicio.	
Condiciones de Ejecución: El administrador llena los campos para realizar el filtrado.	
Entrada/Pasos de ejecución: Introducir el Teléfono, la Fecha Inicio, la Fecha Fin y presiona el botón Buscar.	
Resultado Esperado: Se muestra la información correspondiente, en dependencia del teléfono y el rango de fecha entrado.	

Evaluación de la Prueba: Prueba satisfactoria.

Prueba #9.1 Filtrar Reporte de Utilización del Servicio

Caso de Prueba Aceptación	
Código: 9.1	Número Historia de Usuario: 9
Nombre: Filtrar Reporte de Utilización del Servicio.	
Descripción: Prueba para verificar la funcionalidad Filtrar Reporte de Utilización del Servicio.	
Condiciones de Ejecución: El administrador llena los campos para realizar el filtrado.	
Entrada/Pasos de ejecución: Introducir la Fecha Inicio, la Fecha Fin, la Hora Inicio, la Hora Fin y presiona el botón Buscar.	
Resultado Esperado: Se muestra la información correspondiente, en dependencia del rango de fecha y el rango de hora entrado.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #9.1 Filtrar Reporte de Utilización del Servicio

Caso de Prueba Aceptación	
Código: 9.1	Número Historia de Usuario: 9
Nombre: Filtrar Reporte de Utilización del Servicio.	
Descripción: Prueba para verificar la funcionalidad Filtrar Reporte de Utilización del Servicio.	
Condiciones de Ejecución: El administrador llena los campos para realizar el filtrado.	
Entrada/Pasos de ejecución: Introducir el Teléfono, la Fecha Inicio, la Fecha Fin, la Hora Inicio, la Hora Fin y presiona el botón Buscar.	
Resultado Esperado: Se muestra la información correspondiente, en dependencia del teléfono, del rango de fecha y el rango de hora entrado.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #10.1 Realizar Reporte General Cantidad de Utilización del Servicio

Caso de Prueba Aceptación	
Código: 10.1	Número Historia de Usuario: 10
Nombre: Realizar Reporte General Cantidad de Utilización del Servicio.	
Descripción: Prueba para verificar la funcionalidad Realizar Reporte General Cantidad de Utilización del Servicio.	
Condiciones de Ejecución: Al menos un teléfono debe haber utilizado el servicio.	
Entrada/Pasos de ejecución: En el menú Facturación se escoge la opción Reporte General.	
Resultado Esperado: Se muestra la información de correspondiente a la cantidad de utilización del servicio del día actual ordenada por hora.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba #11.1 Filtrar Reporte General Cantidad de Utilización del Servicio

Caso de Prueba Aceptación	
Código: 11.1	Número Historia de Usuario: 9
Nombre: Filtrar Reporte de Utilización del Servicio.	
Descripción: Prueba para verificar la funcionalidad Filtrar Reporte General Cantidad de Utilización del Servicio.	
Condiciones de Ejecución: El administrador llena los campos para realizar el filtrado.	
Entrada/Pasos de ejecución: Introducir tipo de Reporte, la Fecha Inicio, la Fecha Fin y presiona el botón Buscar.	
Resultado Esperado: Se muestra la información correspondiente, en dependencia del tipo de reporte y del rango de fecha entrado.	
Evaluación de la Prueba: Prueba satisfactoria.	

ANEXO II TABLAS DE COCOMO II.

Tipo de función	Peso del Factor de Complejidad		
	Bajo	Promedio	Alto
Entradas Externas (Inputs)	3	4	8
Salidas Externas (Outputs)	4	5	7
Archivo Lógicos Internos (Archivos)	7	10	15
Archivos Externos de Interfase (Interfases)	5	7	10
Consultas Externas (Queries)	3	4	8

Tabla 10: Peso del Factor de Complejidad. [Boehm 1995/2]

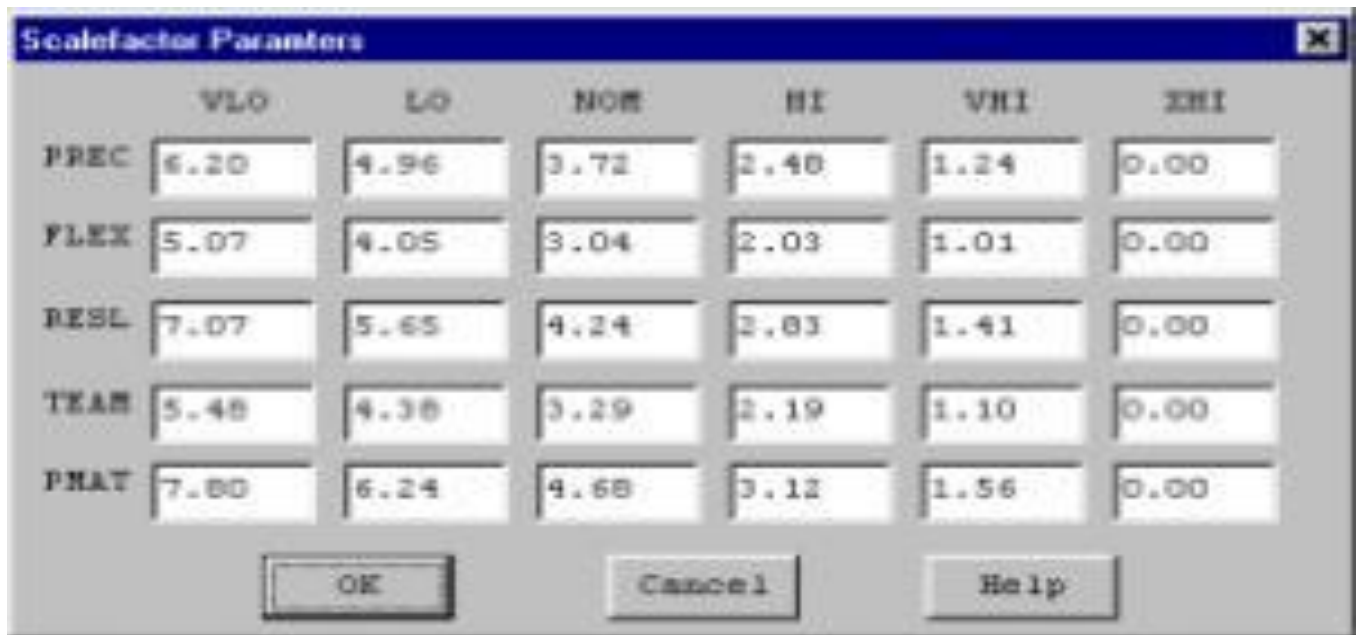


Figura 8: Factores de Escala. [COCOMO II.0]

	XLO	VLO	LO	NOB	HI	VHI	XHI
RCPX	0.73	0.81	0.98	1.00	1.30	1.74	2.38
RUSE	XXXX	XXXX	0.95	1.00	1.07	1.15	1.24
PDIF	XXXX	XXXX	0.87	1.00	1.29	1.81	2.61
PERS	2.12	1.62	1.26	1.00	0.63	0.63	0.50
PREX	1.59	1.33	1.12	1.00	0.67	0.71	0.62
FCIL	1.43	1.30	1.10	1.00	0.67	0.73	0.62
SCED	XXXX	1.43	1.14	1.00	1.00	1.00	XXXX
USR1	XXXX	1.00	1.00	1.00	1.00	1.00	XXXX
USR2	XXXX	1.00	1.00	1.00	1.00	1.00	XXXX

Figura 4: Multiplicadores de Esfuerzo del Modelo de Diseño Temprano. [COCOMO II.0]

