



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

**FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y
COMPUTACIÓN**

INFORME DE MATERIA DE GRADUACIÓN

**“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE VIDEO
CONFERENCIAS PRE-PAGADAS BASADAS EN EL PROTOCOLO SIP”**

Previa a la obtención del Título de:

INGENIERO EN TELEMÁTICA

PRESENTADA POR:

**MARIELA BELEN DIAZ BALLADARES
ANGELA MARIA INFANTE QUIROZ**

GUAYAQUIL - ECUADOR

2012

AGRADECIMIENTO

*A Dios, a nuestros padres,
a nuestros familiares
y a nuestro director que
ha sido un gran apoyo
en cada uno de los
avances y resultados
obtenidos, siendo bases
importantes para realizar
este trabajo.*

DEDICATORIA

A mi hermano por ser incondicional conmigo, a toda mi familia y a mi madre que me acompaña desde el cielo.

Mariela Belén Díaz Balladares.

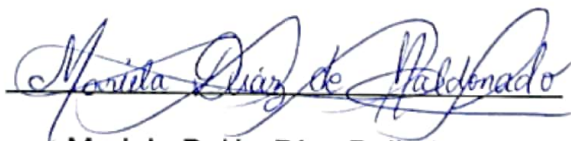
*A Dios que me ha dado la vida y fortaleza
para terminar este proyecto de graduación,
A mis Padres por estar ahí cuando más los necesité,
A mis hermanas Ma. Del Carmen y Kristy
quienes han sido mi guía a seguir.*

Angela María Infante Quiroz.

DECLARACIÓN EXPRESA

"La responsabilidad por los hechos, ideas y doctrinas expuestas en este trabajo de graduación, nos corresponden exclusivamente; y, el patrimonio intelectual de la misma, a la Escuela Superior Politécnica del Litoral"

(Reglamento de exámenes y títulos profesionales de la ESPOL)



Mariela Belén Díaz Balladares



Angela María Infante Quiroz

TRIBUNAL DE SUSTENTACIÓN

A handwritten signature in black ink, appearing to read 'Gabriel Astudillo', is written over a horizontal line.

Ing. Gabriel Astudillo

PROFESOR DE LA MATERIA DE GRADUACIÓN

A handwritten signature in blue ink, appearing to read 'Patricia Chávez', is written over a horizontal line.

Ing. Patricia Chávez

PROFESOR DELEGADO POR EL DECANO DE LA FACULTAD

RESUMEN

El objetivo de este proyecto consiste en el diseño e implementación de un sistema de video conferencias pre-pagadas basadas en el protocolo SIP que permite a las PYMES realizar sesiones entre sus colaboradores o con sus clientes ubicados en localidades remotas.

La necesidad de la creación de este sistema surge por que hoy en día existen PYMES que brindan servicios de consultoría o soporte de manera personal a sus clientes creando un costo extra en el traslado hacia el lugar de destino.

Es por ello que para poder solucionar este problema se ha decidido crear este sistema con la finalidad de que el cliente tenga el servicio de soporte o consultoría de manera remota por medio de video conferencia. Para el cobro del servicio se ha implementado la forma de prepago con el objetivo de que el cliente cancele previamente por el minuto que requiera de servicio, así dependiendo del saldo que tenga disponible podrá realizar o no la video conferencia para solicitar el soporte, y no tener un plan postpago, en donde puede ser el caso de que no se use todos los minutos disponibles.

INDICE GENERAL

AGRADECIMIENTO	ii
DEDICATORIA	iii
DECLARACIÓN EXPRESA	iv
TRIBUNAL DE SUSTENTACIÓN	v
RESUMEN	vi
INDICE GENERAL	vii
ABREVIATURAS	ix
INDICE DE FIGURAS	x
INDICE DE TABLAS	xii
INTRODUCCIÓN	xiii
CAPÍTULO 1	1
1 ANTECEDENTES	1
1.1 Identificación de la Problemática	1
1.2 Objetivos	2
1.2.1 Objetivo General.	2
1.2.2 Objetivos Específicos del Proyecto.	3
1.3 Metodología	4
1.4 Perfil de la Tesis	4
CAPÍTULO 2	6
2 INTRODUCCIÓN	6
2.1 ASTERISK.	6
Historia.....	7
2.2 Protocolos de señalización	8
2.3 Plan de Mercado.....	8
2.4 Canal	10
2.5 AGI.....	10
2.6 PHP - AGI	11
CAPÍTULO 3	13
3 IMPLEMENTACIÓN.....	13
3.1 Introducción.	13
3.2 Hardware	13
3.3 Software.....	14
3.3.1 Servidor	14
3.3.2 Instalación de Asterisk	15
3.3.3 Instalación del Servidor Web Apache.....	18
3.3.4 Instalación de MySQL	18
3.3.5 Instalación de PHP	19
3.4 Base de Datos del Sistema.....	20
3.4.1 Ingreso a mysql.....	20
3.4.2 Creación de la base de datos.....	21

3.4.3	Creación de las tablas.....	21
3.5	Configuraciones en Asterisk	22
3.5.1	Configuración del archivo sip.conf.....	23
3.5.2	Configuración del archivo cdr_mysql.conf	24
3.5.3	Configuración del archivo extconfig.conf	24
3.6	Programa Principal	25
3.6.1	Descripción General.....	25
3.6.2	Casos del sistema	27
3.6.3	Código Fuente.....	28
3.6.4	Descripción de las funciones utilizadas	29
	Funciones sobre la base de datos	29
3.6.5	Funciones sobre el plan de marcado	30
3.6.6	Aplicación Web.....	31
	Administrador:.....	33
	Cliente:.....	34
	Usuario Web:	35
	Extensión:	36
	Usuario Asterisk:.....	37
	Acreditar Saldo:	39
CAPÍTULO 4		41
4	FUNCIONAMIENTO Y PRUEBAS.....	41
4.1	Activación del Servicio de MySQL	41
4.2	Ingreso a la Aplicación Web	41
4.3	Activación de Asterisk.....	42
4.4	Configuración de cuenta SIP en EyeBeam.....	42
4.5	Llamada entre usuarios	45
	Conclusiones y recomendaciones	47
	APENDICE.....	49
	GLOSARIO	49
	ANEXOS	52
	BIBLIOGRAFÍA	55

ABREVIATURAS

AGI	Asterisk Gateway Interface
BSD	Distribución de Software Berkeley
CLI	Interface de Línea de Comando
CRM	Customer Relationship Management
DAHDI	Digium Asterisk Hardware Device Interface
GPL	General Public License
IAX	Inter-Asterisk eXchange
IETF	Grupo Especial sobre Ingeniería de Internet
ITU	Unión Internacional de Telecomunicaciones
MGCP	Media Gateway Control Protocol
PHP	Hypertext Pre-processor
PYMES	Pequeñas Y Medianas Empresas
SIP	Protocolo de Inicio de Sesión
SCCP	Skinny Call Control Protocol
VoIP	(Voice over IP), Voz sobre IP
WWW	World Wide Web

INDICE DE FIGURAS

Figura 3.1 Diagrama de flujo del sistema.....	27
Figura 3.2 Página de Autenticación de la Aplicación Web	32
Figura 3.3 Página principal de la Aplicación Web	33
Figura 3.4 Menú Administración para el usuario tipo administrador	34
Figura 3.5 Menú Procesos para el usuario tipo administrador	34
Figura 3.6 Menú Administración para el usuario tipo cliente.....	34
Figura 3.7 Formulario de consulta de Usuarios WEB	35
Figura 3.8 Formulario de creación de un Usuario WEB	35
Figura 3.9 Formulario de edición de un Usuario WEB	36
Figura 3.10 Opción de activación e inactivación de un Usuario WEB.....	36
Figura 3.11 Formulario de consulta de extensiones	37
Figura 3.12 Formulario de creación de una extensión	37
Figura 3.13 Opción de eliminación de una extensión	37
Figura 3.14 Formulario de consulta de usuarios Asterisk	38
Figura 3.15 Formulario de creación de un usuario Asterisk.....	38
Figura 3.16 Formulario de edición de un usuario Asterisk	38
Figura 3.17 Opción de activación e inactivación de un Usuario Asterisk	39
Figura 3.18 Formulario de consulta de acreditaciones de saldo	39
Figura 3.19 Formulario de acreditación de saldo a un Usuario Asterisk	39
Figura 4.1 Inicialización del servicio de MySQL.....	14
Figura 4.2 Activación de Asterisk.....	42

Figura 4.3 Acceso a la configuración de EyeBeam.....	43
Figura 4.4 Configuración de cuenta SIP en EyeBeam	44
Figura 4.5 Registro de usuario SIP 1500 en Asterisk	44
Figura 4.6 Llamada desde usuario 1600 al 1500	45
Figura 4.7 Finalización del AGI videoconferencia	46
Figura 4.8 Reproducción de mensaje de saldo insuficiente	46

INDICE DE TABLAS

Tabla 3.1 Características del Servidor	14
Tabla 3.2 Software instalado en el Servidor.....	14

INTRODUCCIÓN

Hoy en día hay varias entidades que ofrecen la posibilidad de implementar sistemas basados en voz sobre IP; unos más flexibles y otros más costosos; es por este motivo que se suelen escoger plataformas basadas en software libre.

Actualmente las redes nos permiten transmitir datos y voz, lo cual nos da la posibilidad de implementar un sistema basado en voz sobre IP usando Asterisk como una solución flexible y de bajo costo.

El sistema de gestión de videoconferencias pre-pagadas para PYMES que se propone, solucionara uno de los problemas de consultoría o soporte que normalmente se realizan de forma presencial, brindando la posibilidad de que el cliente pueda solicitar el servicio en cualquier momento y lugar a través de un video-teléfono que se conectara al servidor de VoIP de la empresa que brindara el servicio. El costo del servicio se lo maneja de forma de pre-pago, con lo que el cliente podrá adquirir minutos para realizar sus requerimientos y conservar un saldo disponible para el futuro.

El servidor donde se encuentra alojado el sistema utiliza el sistema operativo Linux, con la distribución Centos 5.6, Asterisk versión 1.8.5 y un conjunto de servicios que complementan la funcionalidad de Asterisk como son: Lenguaje

de programación PHP, el motor de base de datos MySQL y el Servidor de Aplicaciones Web Apache, el cual brindara la posibilidad de servir nuestra aplicación web en la que se podrá realizar algunas acciones como son: gestión de usuarios, extensiones, saldo y de configuraciones de video-teléfonos.

CAPÍTULO 1

1 ANTECEDENTES

En este primer capítulo se presenta de forma general la problemática que tienen hoy en día las PYMES que brindan un servicio de consultoría o soporte de manera presencial, y de qué manera nuestro sistema brindara un mejor acceso a los mismos. Además se detallará nuestros objetivos principales a alcanzar mediante la creación de nuestro Sistema de gestión de videoconferencias pre-pagadas.

1.1 Identificación de la Problemática

Actualmente las comunicaciones son una parte importante para el desarrollo y crecimiento de las empresas. Dentro de una empresa todo lo que represente rapidez y bajos costos es fácilmente adaptado para su uso dentro de la misma. Con el crecimiento de las redes de datos, la llegada del internet y las nuevas tecnologías facilito la forma y de cómo poder acceder a la información.

El avance tecnológico ha permitido la migración de sistemas como las centrales telefónicas analógicas, que han sido reemplazadas por centrales PBX de telefonía IP. Esto mejora los costos de implementación, permitiendo que una PYME pueda brindar un servicio basado en VoIP.

En los últimos años se han desarrollado varias soluciones VoIP usando software libre, la que más ha destacado es Asterisk, un sistema completo de comunicaciones sobre redes de datos que integra la telefonía, internet, fax y otros elementos, logrando destacar su interface de enlace AGI que permite extender las funcionalidades de Asterisk mediante el uso de lenguajes de programación como PHP, sirviendo de enlace entre las aplicaciones externas y el núcleo de Asterisk.

1.2 Objetivos

La implementación de nuestra solución tecnológica pretende alcanzar los siguientes objetivos.

1.2.1 Objetivo General.

- Diseñar e implementar un sistema de video conferencias pre-pagadas basadas en el protocolo

SIP que permite a las PYMES realizar sesiones entre sus colaboradores o con sus clientes ubicados en localidades remotas.

1.2.2 Objetivos Específicos del Proyecto.

- Brindar una solución tecnológica que satisfaga las necesidades actuales de las PYMES que brindan servicio de consultoría o soporte de manera presencial.
- Usar el AGI para el desarrollo de nuestro sistema.
- Usar la librería “php-agi” para el desarrollo del script de validación de inicio de una videoconferencia.
- Permitir la opción de gestión de usuarios y extensiones a través de una aplicación web.
- Brindar la opción de gestión de saldo para los clientes a través de una interfaz web.
- Ofrecer la opción de consulta de saldo de un cliente a través de una interfaz web.
- Dar la opción de consulta de la configuración de un video teléfono por parte del cliente a través de una interfaz web.

1.3 Metodología

Los pasos para la realización de este proyecto son:

1. Instalación del Sistema Operativo Linux distribución
CENTOS V 5.6
2. Instalación del Servidor de Aplicaciones WEB Apache.
3. Instalación del motor de base de datos MySQL.
4. Instalación de PHP.
5. Configuración de PHP con el Servidor de Aplicaciones WEB
Apache.
6. Instalación y configuración del modulo de PHP para la
conexión con el motor de la Base de Datos MySQL.
7. Instalación y configuración de Asterisk V 1.8.5
8. Instalación y configuración de softphone EyeBeam.

1.4 Perfil de la Tesis

En el capítulo 2 se presentan las bases teóricas para mayor comprensión de los conceptos tratados para la realización del proyecto.

En el capítulo 3 se presentan las especificaciones técnicas, la instalación del software y las configuraciones respectivas para el funcionamiento del proyecto.

En el capítulo 4 se presentan las pruebas realizadas con los diferentes casos que se pueden presentar en el proyecto.

CAPÍTULO 2

2 INTRODUCCIÓN

El crecimiento de la comunicación y el manejo de información han aumentado en los últimos años. Son por estos motivos, que el desarrollo de sistemas basados en VoIP ha incrementado en gran medida.

Este segundo capítulo detalla cada una de las tecnologías utilizadas en el desarrollo de nuestra solución tecnológica.

2.1 ASTERISK.

Asterisk es una solución PBX por software, originalmente desarrollado para el sistema operativo GNU/Linux, pero actualmente también se distribuye en versiones para los sistemas operativos BSD, Mac OS X, Solaris y Microsoft Windows, aunque la plataforma nativa (GNU/Linux) es la que cuenta con mejor soporte de todas, realizando las funciones de una central telefónica, de un sistema de buzones de voz, un entorno de llamadas para centro de llamadas, un sistema integrable con soluciones CRM, y casi cualquier otro servicio.

Partiendo de este concepto, se evidencia que Asterisk, no es una central telefónica convencional; tiene características que en anteriormente sólo

eran accesibles mediante la adquisición de módulos costosos razones por las cuales Asterisk ha alcanzado una gran acogida a nivel mundial, presentándose como una opción seria al momento de implementaciones de diversos sistemas telefónicos.

Historia

Asterisk fue concebido y desarrollado por Mark Spencer, por una necesidad personal ya que requería una central telefónica para la su pequeña empresa de soporte “Linux Support Services”.

Spencer pensó en adquirir un software pero se dio cuenta que estaba lejos de su presupuesto, así que decidió crear un programa y comenzó a codificar lo que hoy conocemos como Asterisk. En 1999, cuando tuvo un código digno de mostrar al mundo decidió liberarlo bajo licencia GPL.

Con el paso del tiempo, Mark se percató de que su software necesitaba interactuar con un hardware telefónico y tuvo conocimiento del proyecto Zaptel, un proyecto de código abierto creado por Jim Dixon, que tenía como objetivo el crear drivers abiertos para tarjetas telefónicas de computadora.

A partir de allí Asterisk y Zaptel caminarían caminaron de la mano; de tal modo que en la actualidad los dos proyectos son mantenidos por la misma compañía.

En el 2002 Linux Support Services se convirtió en Digium, y actualmente ofrece productos y servicios relacionados con Asterisk, además se encarga del desarrollo del producto.

2.2 Protocolos de señalización

Los protocolos de señalización de VoIP cumplen funciones similares a sus homólogos en la telefonía tradicional, por ejemplo establecer sesiones y controlar el progreso de la llamada. Existen algunos protocolos de señalización desarrollados por diferentes organismos, tales como la ITU, el IETF; algunos soportados por Asterisk son SIP, IAX (estos son los mas utilizados), H.323, MGCP y SCCP.

2.3 Plan de Mercado

El plan de mercado es el encargado de proveer el control sobre Asterisk y determina que acción se debe realizar en función del estado actual.

Las extensiones de Asterisk son diferentes a las extensiones tradicionales de una PBX, pueden incluir números y texto.

Inicialmente el archivo `extensions.conf` tiene un contexto general `[general]` y otro global `[globals]`. Ambos establecen opciones y comportamientos que serán considerados durante la ejecución del plan de marcado.

Dentro de los contextos principales en el archivo `extensions.conf`, se cuenta con un conjunto de contextos encerrados entre corchetes “`[]`” realizando procesos o métodos para ejecutar varias operaciones, por ejemplo los contextos que realizan las opciones de grabación y reproducción de voces, procesamiento de datos ingresados, ejecución de AGIs, evaluación de condiciones, entre otras.

Las aplicaciones constituyen un papel importante, ya que son éstas quienes

realizan diferentes acciones sobre un canal.

A continuación se enlistan algunas aplicaciones utilizadas:

- `Answer()`: contesta una llamada.
- `Hangup()`: cuelga una llamada.
- `Dial()`: realiza una llamada saliente.

- `Playback()`: reproduce un archivo de sonido.
- `Background()`: similar a `playback`, pero si el usuario presiona dígitos, la aplicación los captura e interrumpe la reproducción del audio, enviándolo a la extensión que corresponda.
- `GotIf()`: condiciona una expresión como verdadera o como falsa para determinar su destino.

2.4 Canal

Un canal es una conexión que trae una llamada a un PBX Asterisk. Un canal podría ser una conexión a un teléfono ordinario fijado a mano o a una línea telefónica ordinaria, o a una llamada lógica (como una llamada telefónica del Internet). Asterisk no hace ninguna distinción entre los canales del estilo de "FXO" y de "FXS", es decir, no distingue entre las líneas telefónicas y los teléfonos. Cada llamada se pone o se recibe en un canal distinto.

2.5 AGI

Es una interfaz que agrega funcionalidad a Asterisk para que interactúe con diversos lenguajes de programación como: Perl,

PHP, C, PASCAL, Bourne Shell, etc., y así poder hacer sistemas más complejos y de mayor utilidad.

- AGI puede controlar el plan de discado, llamado en `extensions.conf`.
- EAGI da a la aplicación la posibilidad para tener acceso y para controlar los canales de sonidos además de la interacción con el plan del discado.
- FastAGI se puede utilizar para hacer el procesamiento en una máquina remota por conexión de red.
- DEADAGI da el acceso a un canal muerto después de colgar.

2.6 PHP - AGI

Es una clase con varias funciones previamente definidas que hacen más fácil la tarea de elaborar códigos PHP para Asterisk. Existen al momento la versión 2.20.

Para usar esta librería se debe descargarla, descomprimirla y copiarla dentro del directorio `/var/lib/asterisk/agi-bin/`. Ahora, para poder utilizar las funciones que en esta clase se definen, se debe incluir la clase en el código mediante la sentencia:

```
require "phpagi-2.20/phpagi.php";
```

Con estas funciones es posible construir aplicaciones más complejas de forma sencilla. Se recomienda revisar las funciones disponibles dentro de cada clase en la documentación correspondiente dentro de su sitio oficial.

CAPÍTULO 3

3 IMPLEMENTACIÓN

En el presente capítulo se describe nuestra solución tecnológica, donde se detalla el proceso para la implementación del mismo, así como la instalación y configuración de cada uno de los componentes que lo conforman.

3.1 Introducción.

Esta solución tecnológica podrá ser utilizada por PYMES que actualmente brinden servicios de consultoría, asesoría o soporte de manera presencial que carezcan de un sistema basado en VoIP. Creando de esta manera la oportunidad de tener un nuevo canal de comunicación controlado en forma de pre-pago.

3.2 Hardware

La solución tecnológica cuenta con un servidor donde se encuentra el sistema y se van a procesar las solicitudes de videoconferencia por parte de los clientes o personal de la

empresa. El servidor que se utilizó es una computadora con las características detalladas en la Tabla 3.1.

Procesador	Intel Core Duo de 2.8 GHZ
Tarjeta de Red	10/100 Mbps
Memoria Ram	2 GB
Disco Duro	80 GB

Tabla 3.1 Características del Servidor

3.3 Software

3.3.1 Servidor

En el servidor se instalaron el sistema operativo y aplicaciones detalladas en la Tabla 3.2.

Sistema Operativo	Linux
Distribución	Centos 5.6
Arquitectura	X86
Software IP	Asterisk 1.8.5
Motor de Base de Datos	MySQL
Interprete de Lenguaje de Programación	PHP

Tabla 3.2 Software instalado en el Servidor

3.3.2 Instalación de Asterisk

Para instalar Asterisk en el servidor se requieren instalar algunas dependencias previamente, cuyo proceso es detallado a continuación:

1. Descargar el paquete estable de Asterisk 1.8.5.0 y copiarlo al directorio /usr/src

<http://downloads.asterisk.org/pub/telephony/asterisk/releases/asterisk-1.8.5.0.tar.gz>

2. Descargar el paquete de Asterisk Add-Ons 1.6.2.3 y copiarlo al directorio /usr/src

<http://downloads.asterisk.org/pub/telephony/asterisk/releases/asterisk-addons-1.6.2.3.tar.gz>

3. Se recomienda tener todos los paquetes actualizados utilizando el siguiente comando:

```
yum update
```

4. Actualizar y activar paquetes necesarios a través de los siguientes comandos

```
yum groupinstall "Development Tools"
```

```
yum install e2fsprogs-devel keyutils-libs-devel  
krb5-devel libogg libselinux-devel libsepol-devel  
libxml2 libxml2-devel libtiff-devel kernel-devel  
kernel-headers ncurses ncurses-devel audiofile-
```

```
devel libogg-devel openssl zlib zlib-devel curl gcc  
gcc-c++ libtermcap libtermcap-devel gtk2 gtk2-  
devel libvorbis libvorbis-devel vorbis-tools curl-  
devel libidn-devel festival festival-devel
```

5. Actualizar el Kernel a través del siguiente comando

```
yum update kernel
```

6. Siguiendo el LSB (Linux Standard Base) Asterisk deberá ser descomprimido en /usr/src/

```
tar -xvzf asterisk-1.8.5.0.tar.gz
```

```
tar -xvzf asterisk-addons-1.6.2.3.tar.gz
```

7. Ejecutar el siguiente grupo de comandos para la configuración e instalación de Asterisk Add-Ons:

```
cd asterisk-addons-1.6.2.3
```

```
make clean
```

```
./configure
```

```
make menuselect
```

```
make install
```

8. Ejecutar el siguiente grupo de comandos previa configuración de Asterisk:

```
cd asterisk-1.8.5.0
```

```
contrib/scripts/get_mp3_source.sh
```

```
cd ../../asterisk-1.8.5.0
```

`./configure`

`make menuselect`

9. En la configuración seleccionamos las siguientes opciones:

- Add-ons
 - `app_saycountpl`
 - `format_mp3`
- Resource Module
 - `res_rtp_asterisk`
- Core and Sound Packages
 - `CORE-SOUNDS-EN-ULAW`
 - `CORE-SOUNDS-EN-ALAW`
 - `CORE-SOUNDS-EN-GSM`
- Music On Hold File Packages
 - `MOH-EN-ULAW`
 - `MOH-EN-ALAW`
 - `MOH-EN-GSM`
- Extra Sound Packages
 - `EXTRAS-SOUNDS-EN-GSM`
 - `EXTRAS-SOUNDS-EN-ULAW`
 - `EXTRAS-SOUNDS-EN-ALAW`

10. Una vez realizada la configuración procedemos a realizar la instalación de Asterisk utilizando los siguientes comandos:

```
make  
make install  
make config
```

3.3.3 Instalación del Servidor Web Apache

A continuación se detalla el procedimiento para instalar el servidor de aplicaciones Web Apache.

1. Ejecutar el siguiente grupo de comandos

```
yum install httpd  
chkconfig --add httpd  
service httpd start
```

3.3.4 Instalación de MySQL

A continuación se detalla el procedimiento para instalar el motor de la Base de Datos MySQL.

1. Ejecutar el siguiente grupo de comandos

```
yum install mysql-server  
chkconfig --add mysqld  
service mysqld restart
```

mysql_secure_installation

3.3.5 Instalación de PHP

Se requiere la instalación del intérprete PHP versión 5 para la ejecución de los scripts AGI que están desarrollados bajo este lenguaje.

A continuación se detalla el procedimiento para la instalación:

1. Ejecutar el siguiente comando para la instalación de php y del modulo para la conexión con MySQL.

```
yum install php53 php53-mysql
```

2. Para que un script en php pueda ser ejecutado por Asterisk se debe cambiar los permisos de ejecución mediante el siguiente comando:

```
chmod 777  
<ruta_absoluta_o_nombre_archivo.php>
```

3. Para ejecutar un script en php en consola se debe ubicar en la ruta donde se encuentra el archivo o especificar la ruta absoluta en el comando y ejecutar el comando:

```
! /usr/bin/php -q <archivo.php>
```

4. Descargar la librería phpagi de la siguiente dirección:

```
http://sourceforge.net/projects/phpagi/files/phpagi/  
2.20/phpagi-2.20.tgz/download
```

5. Copiar la librería descargada de phpagi en la ruta
`/var/lib/asterisk/agi-bin/`

3.4 Base de Datos del Sistema

Este sistema cuenta con una base de datos almacenada en el motor MySQL, la cual está compuesta por las siguientes tablas:

- `usuario_web` (Aquí se almacenan los usuarios que tendrán acceso a la aplicación web)
- `extension` (Aquí se almacenan todas las extensiones que serán usados por los usuarios Asterisk)
- `usuario` (Aquí se almacenan los usuarios Asterisk por cada usuario web y extensión, así como también el saldo de cada uno de ellos)
- `cdr` (Aquí se almacena el registro de las llamadas realizadas por los usuarios Asterisk)

3.4.1 Ingreso a mysql

Para el acceso a mysql ingresamos el siguiente comando:

```
mysql -u root -p admin123
```

3.4.2 Creación de la base de datos

La base de datos tiene por nombre “videoconferencia_db” y se deberá emplear el siguiente comando:

```
create database videoconferencia_db;
```

3.4.3 Creación de las tablas

A continuación se detalla una breve explicación del contenido de cada una de las tablas que se creó en la base de datos.

usuario_web

Contiene todos los datos de los usuarios que accederán a la aplicación WEB, proporcionada para el sistema de videoconferencia.

extension

Información de las extensiones que son asignadas para que cada uno de los usuarios.

usuario

Datos de los usuarios que serán utilizados por Asterisk.

cdr

Registro en detalle de llamadas, que almacena datos pertenecientes al origen, destino, duración y otros de una llamada para la tarificación o el control en la PBX.

3.5 Configuraciones en Asterisk

Asterisk es mucho mas que una PBX normal y se puede hacer telefonía de nuevas formas. Puede conectar empleados trabajando en casa hacia la PBX en la oficina mediante una conexión de banda ancha, conecta oficinas mediante varias alternativas de VoIP, Internet o una red privada virtual, entrega correo de voz, integrada con una web y algún mail, puede construir aplicaciones interactivas de voz y mucho más. Además incluye cualidades que sólo se encuentran en productos top de la mensajería unificada.

Para poder hacer todo esto, Asterisk funciona mediante canales. Estos canales son drivers para distintos tipos de conexiones para protocolos de VoIP como SIP, IAX, MGCP y H.323.

Teléfonos y Softphones conectan a un canal, algunos de ellos se registran para dar a conocer que están en línea. Los canales también registran conexiones salientes a otro servidor VoIP.

Entender la configuración de Asterisk es algo así como aprender a programar, son eventos que van sucediendo, y que se tienen que ir organizando paso a paso, para que la PBX sepa que hacer. Ahora en el siguiente paso voy a dar a conocer los archivos de configuración y cuales fueron los que utilice para la PBX.

3.5.1 Configuración del archivo sip.conf

Este archivo de configuración se encuentra en la ruta `/etc/asterisk` y sirve para colocar los usuarios sip a registrar en la pbx Asterisk, así como para conectarse a un proveedor sip y establecer todo lo relacionado a este protocolo. A continuación la configuración que se agregó para activar video que será utilizado para nuestro proyecto.

sip.conf

[general]

videosupport=yes ; Para permitir video llamadas.

allow=g723 ; habilita un códec estándar de voz, proporciona una calidad de voz que cubre 300 Hz y 3400 Hz

Cabe recalcar que los usuarios se los está manejando a través de la base de datos por lo que aquí se ha definido el contexto general donde se especifica el soporte de videoconferencia.

3.5.2 Configuración del archivo cdr_mysql.conf

Este archivo de configuración se encuentra en la ruta /etc/asterisk y sirve para especificar que el registro de las llamadas que realizan los usuarios se los pueda guardar en una tabla de una base de datos. A continuación la configuración que se ha implementado para la realización de este proyecto:

cdr_mysql.conf

```
[global]
hostname=localhost
dbname=videoconferencia_db
table=cdr
password=admin123
user=root
port=3306
```

3.5.3 Configuración del archivo extconfig.conf

Este archivo de configuración se encuentra en la ruta /etc/asterisk y sirve para especificar que la gestión de los usuarios y extensiones de Asterisk se lo realice a través de una

base de datos. A continuación la configuración que se ha implementado para la realización de este proyecto:

ext_config.conf

[settings]

sipusers => mysql,videoconferencia_db,usuario; SIP user

sippeers => mysql,videoconferencia_db,usuario; SIP peers

extensions => mysql,videoconferencia_db,extension; SIP
extensions

3.6 Programa Principal

3.6.1 Descripción General

El proceso comienza cuando un usuario realiza una llamada a otro usuario o extensión para lo cual se invoca el AGI “videoconferencia.agi” donde se almacena el programa principal.

El programa principal se encarga de consultar si el usuario tiene saldo disponible para realizar la llamada. En el caso de tenerla realiza la llamada sin problemas hasta que su saldo se consume o termine su llamada; luego de finalizar la llamada se invoca a un procedimiento en mysql llamado “actualizar_saldo”

el cual se encarga de consultar la tabla cdr que es donde se registra las llamadas realizadas y su tiempo y con esta información actualiza el saldo del usuario. Para el caso de que no tuviera saldo el programa principal reproduce un mensaje indicando que no tiene saldo suficiente para realizar la llamada.

Para el mensaje se uso un audio pregrabado de Asterisk que menciona que no tiene saldo disponible para esta llamada, este puede ser localizado en la ruta `/var/lib/asterisk/sounds`.

Los archivos “videoconferencia.agi” y `phpagi.php` se encuentran en la ruta `/var/lib/asterisk/agi-bin` y se los hace ejecutables para poder utilizarlos.

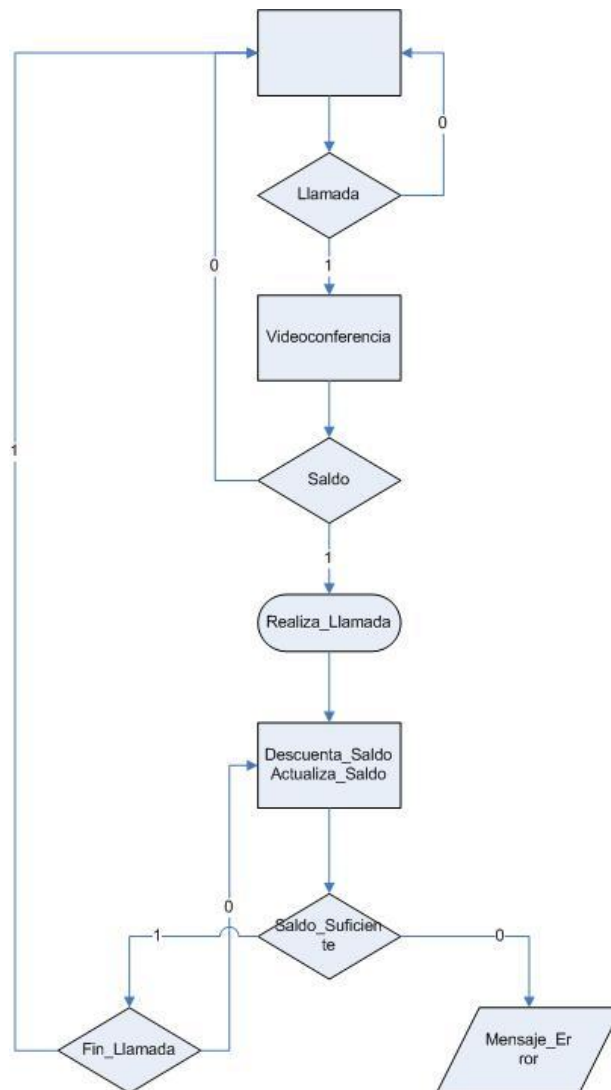


Figura 3.1 Diagrama de flujo del sistema.

3.6.2 Casos del sistema

Al momento que se realiza una llamada en el sistema. Se pueden producir los siguientes casos:

- **El cliente no tiene saldo suficiente para realizar la llamada**

Para este caso se reproduce un audio pregrabado llamado “insuficiente_saldo” el cual indica al cliente que no tiene saldo suficiente.

- **El cliente tiene saldo suficiente para realizar la llamada y finaliza antes de que se agote el saldo**

Para este caso inmediatamente se finaliza la llamada se invoca a un procedimiento almacenado en la base de datos de Mysql llamado “actualizar_saldo” el cual se encarga de consultar la tabla cdr y realiza de acuerdo al tiempo que tomo la llamada la respectiva actualización al campo saldo de la tabla usuario con lo que finaliza el proceso de actualización.

- **El cliente tiene saldo suficiente para realizar la llamada y finaliza hasta que se agote el saldo**

El proceso es similar al anterior con la particularidad de que el usuario estará con saldo 0.

3.6.3 Código Fuente

La implementación del código nombrado anteriormente, se lo puede observar en el ANEXO 1.

3.6.4 Descripción de las funciones utilizadas

La librería `phpagi` nos permite realizar distintas operaciones sobre el plan de marcado y sobre bases de datos. En este proyecto utilizamos distintas funciones que nos provee ésta librería las cuales detallamos a continuación:

Funciones sobre la base de datos

`mysql_connect('localhost', 'root', 'admin123')`

Función que nos permite conectar con la base de datos, el primer parámetro indica la ubicación de la base de datos, el segundo parámetro indica el nombre del usuario al cual pertenece la base de datos y el tercer parámetro indica la clave de ese usuario.

`mysql_select_db('videoconferencia_db',$con)`

Función que selecciona la base de datos con la cual vamos a trabajar, el primer parámetro indica el nombre de la base de datos y el segundo parámetro indica el nombre del puntero que tiene la conexión hacia la base de datos.

```
$row=mysql_query($sql,$con)
```

Función que ejecuta una petición, el primer parámetro es una cadena de caracteres que contiene la petición a ser ejecutada y el segundo parámetro indica el nombre del puntero que tiene la conexión hacia la base de datos.

```
$row=mysql_fetch_array($row)
```

Función que devuelve un arreglo de claves de cada una de las columnas de la base de datos la petición ejecutada.

3.6.5 Funciones sobre el plan de mercado

Lo que se muestra a continuación, detalla el uso de algunas de las aplicaciones de mercado que posee Asterisk, éstas fueron agregadas en el archivo “videoconferencia.agi” y phpagi.php se encuentran en la ruta /var/lib/asterisk/agi-bin.

```
$agi = new AGI()
```

Crea una nueva instancia de la clase agi.

```
require('include/phpagi.php')
```

Permite que las funciones en la librería phpagi sean usadas en el código principal.

```
$agi->answer()
```

Contesta la llamada.

```
$agi->exec(Playback,"ArchivodeAudio")
```

Reproduce una grabación.

```
$agi->hangup()
```

Termina la llamada.

3.6.6 Aplicación Web

Se creó una aplicación web desarrollada en PHP para la administración de los usuarios Asterisk, usuarios web, extensiones y para la acreditación de saldo.

En la se puede apreciar la página de autenticación de la aplicación donde el administrador o cliente puede acceder con su usuario y clave web.



The image shows a web application login interface. At the top left is a logo consisting of a blue circle with an orange dot in the center, surrounded by three orange curved lines above it, all within a grey speech bubble-like shape. To the right of the logo, the text "SGVCP" is displayed in a large, bold, grey font. Below "SGVCP", the text "SISTEMA DE GESTIÓN DE VIDEOCONFERENCIAS PRE-PAGADAS" is written in a smaller, bold, black font. Below the header, there is a white rounded rectangle containing the login form. The form has two input fields: the first is labeled "Usuario" and the second is labeled "Contraseña". Both fields are empty. At the bottom right of the form, there are two blue buttons with white text: "Ingresar" and "Limpiar".

Figura 3.2 Página de Autenticación de la Aplicación Web

En la Figura 3.3 se puede ver la página principal de la aplicación web una vez que el usuario ha realizado el proceso de autenticación correctamente.



Figura 3.3 Página principal de la Aplicación Web

Para el acceso al sistema se han definido dos tipos de usuarios, los cuales son detallados a continuación

Administrador:

El usuario de tipo administrador puede realizar las siguientes acciones, como se observan en las Figuras 3.4 y 3.5, como son:

- Consultar, crear, editar e inactivar usuarios que pueden acceder al sistema WEB.
- Consultar, crear y eliminar extensiones que son utilizadas por el software Asterisk.
- Consultar, crear y eliminar usuarios que son utilizados por el software Asterisk.
- Acreditar saldo a los usuarios de Asterisk.

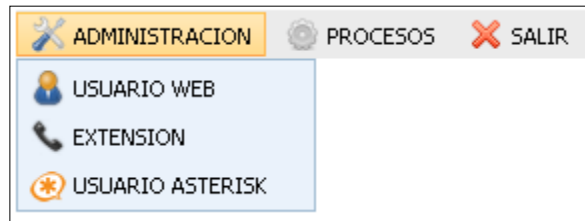


Figura 3.4 Menú Administración para el usuario tipo administrador

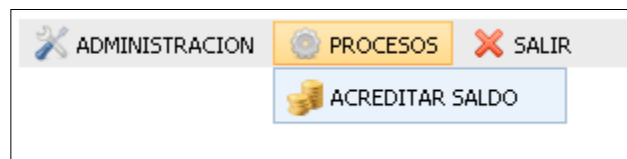


Figura 3.5 Menú Procesos para el usuario tipo administrador

Cliente:

El usuario de tipo cliente puede realizar las siguientes acciones, como se muestra en la Figura 3.6, como son:

- Consultar información respecto a la configuración de un softphone o videoteléfono IP.

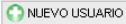


Figura 3.6 Menú Administración para el usuario tipo cliente

La aplicación WEB está compuesta por las siguientes opciones:

Usuario Web:

Permite al administrador consultar, crear, editar e inactivar los usuarios que accederán a la aplicación WEB, como se muestran en las Figuras 3.7, 3.8, 3.9 y 3.10 respectivamente.



DESCRIPCION:

POR:

NOMBRES

CONSULTAR

Cedula	Usuario	Nombres	Apellidos	Tipo	Estado	Opción
0926525585	admin	MARIELA BELEN	DIAZ BALLADARES	ADMINISTRADOR	ACTIVO	
0918781238	cliente	ANGELA MARIA	INFANTE QUIROZ	CLIENTE	ACTIVO	

Figura 3.7 Formulario de consulta de Usuarios WEB

USUARIO:

CLAVE:

CEDULA:

NOMBRES:

APELLIDOS:

TIPO:

ADMINISTRADOR

 CREAR USUARIO

 CANCELAR

Figura 3.8 Formulario de creación de un Usuario WEB

USUARIO:	admin
CLAVE:	
CEDULA:	0926525585
NOMBRES:	MARIELA BELEN
APELLIDOS:	DIAZ BALLADARES
TIPO:	ADMINISTRADOR ▼
✎ EDITAR USUARIO ✖ CANCELAR	

Figura 3.9 Formulario de edición de un Usuario WEB

Tipo	Estado	Opción
ADMINISTRADOR	ACTIVO	 
CLIENTE	ACTIVO	 

Figura 3.10 Opción de activación e inactivación de un Usuario WEB

Extensión:

Permite al administrador consulta, crear y eliminar las extensiones que serán utilizadas por el software Asterisk, como se muestran en las Figuras 3.11, 3.12 y 3.13 respectivamente.

ID	Extensión	Opción
1	1500	
2	1600	
10	1800	

Figura 3.11 Formulario de consulta de extensiones

Figura 3.12 Formulario de creación de una extensión

Extensión	Opción
1500	
1600	
1800	

Figura 3.13 Opción de eliminación de una extensión

Usuario Asterisk:

Permite al administrador consultar, crear, editar y eliminar los usuarios que serán utilizados por el software Asterisk, como se observan en las Figuras 3.14, 3.15, 3.16 y 3.17 respectivamente. Para el caso del cliente únicamente permite consultar su información.

NUEVO USUARIO

DESCRIPCION:

POR:

ID	Extensión	Nombre	Tipo	Usuario Asterisk	Clave Asterisk	Saldo	Usuario Web	Opción
1	1500	Ryaz Khan	ADMINISTRADOR	1500	1234	0	admin	 
2	1600	Ryaz Khan	CLIENTE	1600	1234	83	cliente	 

Figura 3.14 Formulario de consulta de usuarios Asterisk

USUARIO:

CLAVE:

EXTENSION:

NOMBRE:

USUARIO WEB:

Figura 3.15 Formulario de creación de un usuario Asterisk

USUARIO:

CLAVE:

EXTENSION:

NOMBRE:

USUARIO WEB:

Figura 3.16 Formulario de edición de un usuario Asterisk

Saldo	Usuario Web	Opción
0	admin	 
83	cliente	 

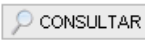
Figura 3.17 Opción de activación e inactivación de un Usuario Asterisk

Acreditar Saldo:

Permite al administrador consultar y acreditar saldo a los usuarios Asterisk, como se muestran en las Figuras 3.18 y 3.19 respectivamente.



USUARIO:



ID	Usuario	Saldo
1	1500	200
2	1600	200
10	1800	200

Figura 3.18 Formulario de consulta de acreditaciones de saldo

USUARIO:

SALDO:

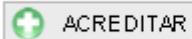
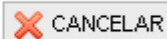



Figura 3.19 Formulario de acreditación de saldo a un Usuario Asterisk

CAPÍTULO 4

4 FUNCIONAMIENTO Y PRUEBAS

En este capítulo se describe el funcionamiento de la solución tecnológica y las pruebas que se realizaron.

4.1 Activación del Servicio de MySQL

Para que el sistema pueda hacer las consultas pertinentes, el servicio de MySQL debe estar activo, como se muestra en la Figura 4.1.

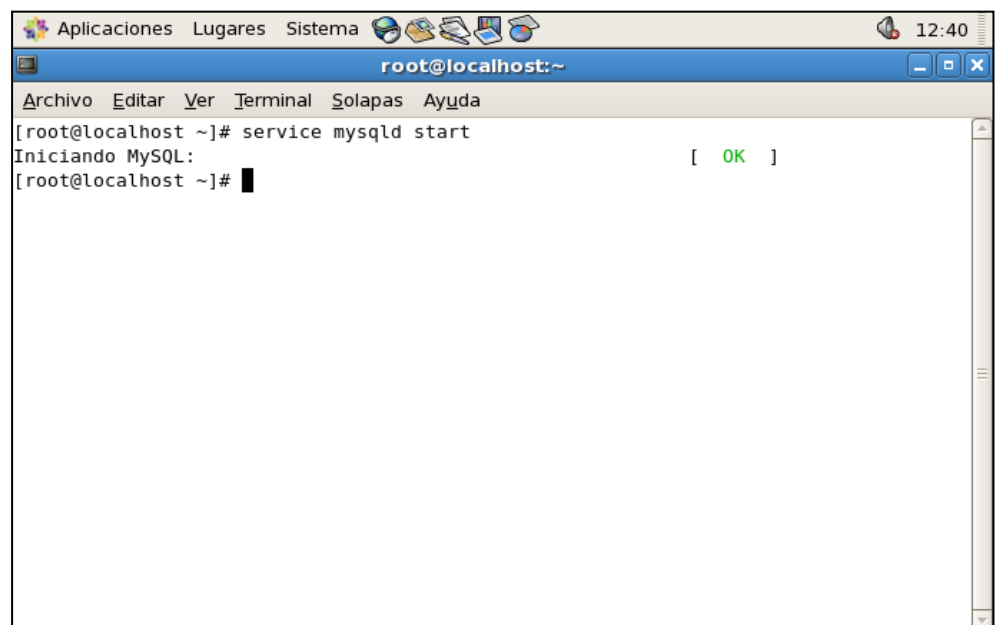


Figura 4.1 Inicialización del servicio de MySQL

4.2 Ingreso a la Aplicación Web

Accedemos a la Aplicación Web utilizando el usuario y clave de administrador como se puede ver en la Figura 3.1.

Una vez dentro del sistema nos dirigimos al menú Administración opción Usuario Web donde procedemos a crear dos usuarios web como se puede ver en la Figura 3.7.

Posterior a esta etapa procedemos a crear dos extensiones por lo que nos dirigimos al menú Administración opción Extensión como se puede ver en la Figura 3.11.

Al final procedemos a crear dos usuarios Asterisk por lo que nos dirigimos al menú Administración opción Usuario Asterisk como se puede ver en la Figura 3.14.

Una vez terminado el proceso de creación de los usuarios Asterisk, procedemos a acreditar saldo a ambos usuarios por lo que nos dirigimos al menú procesos opción Acreditar Saldo como se puede ver en la Figura 3.18.

Finalmente con la información ingresada se procede a realizar las pruebas con Asterisk.

4.3 Activación de Asterisk

A través del comando siguiente nos conectamos a la consola de Asterisk.

asterisk -rvvv

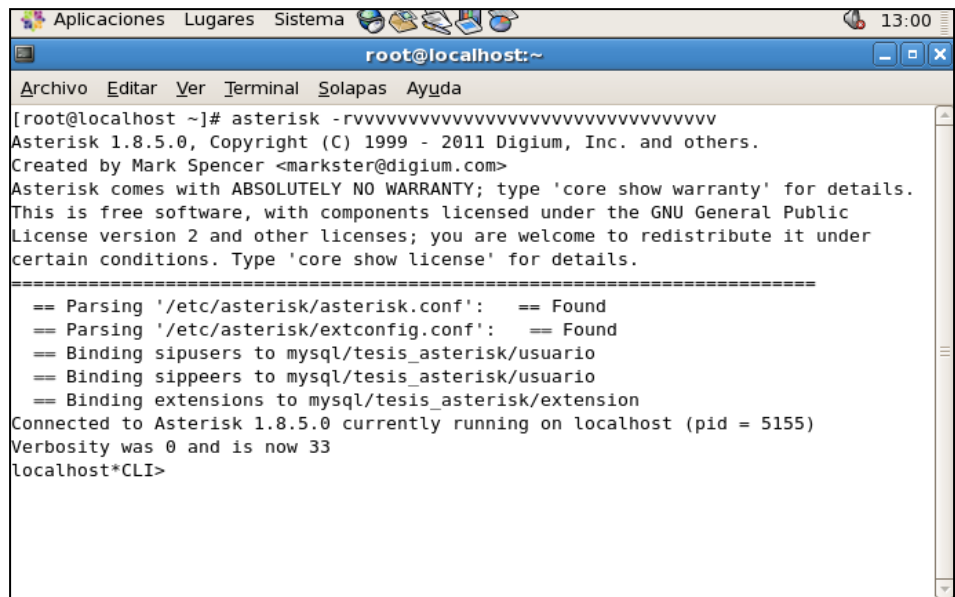


Figura 4.2 Activación de Asterisk

4.4 Configuración de cuenta SIP en EyeBeam

Para las pruebas se uso como softphone con soporte de video al EyeBeam. Una vez instalado el software y ejecutado, nos dirigimos con botón derecho sobre el softphone a la opción de Settings.



Figura 4.3 Acceso a la configuración de EyeBeam

Una vez dentro de las opciones de configuración, como se muestra en la Figura 4.4, del EyeBeam nos dirigimos a la parte de SIP Accounts en la opción Add a New SIP Account y configuramos al primer usuario, como se muestra en la Figura 4.5), con los siguientes datos:

Display Name: 1500

User name: 1500

Password: 1234

Authorization user name: 1500

Domain: 192.168.0.164 (IP del servidor de pruebas)

Choose Setting Category

- SIP Accounts
 - 192.168.0.164**
 - Add a New SIP Account
- Media
- System
- User Interface
- Diagnostics
- License Key

☒ Enable this SIP account

User Details

Display Name

User name

Password

Authorization user name

Domain

Domain Proxy

☒ Register with domain

☐ Use as Outbound Proxy

☐ Manual Override Host

SIP Listen Port

☐ Manual override

Clear proxy

Apply Revert OK Cancel

Figura 4.4 Configuración de cuenta SIP en EyeBeam

```
root@localhost:~  
Archivo Editar Ver Terminal Solapas Ayuda  
localhost*CLI>  
Disconnected from Asterisk server  
Executing last minute cleanups  
[root@localhost ~]# asterisk -rvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv  
Asterisk 1.8.5.0, Copyright (C) 1999 - 2011 Digium, Inc. and others.  
Created by Mark Spencer <markster@digium.com>  
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for details.  
This is free software, with components licensed under the GNU General Public  
License version 2 and other licenses; you are welcome to redistribute it under  
certain conditions. Type 'core show license' for details.  
=====
```

```
-- Parsing '/etc/asterisk/asterisk.conf': == Found  
-- Parsing '/etc/asterisk/extconfig.conf': == Found  
-- Binding sipusers to mysql/tesis_asterisk/usuario  
-- Binding sippeers to mysql/tesis_asterisk/usuario  
-- Binding extensions to mysql/tesis_asterisk/extension  
Connected to Asterisk 1.8.5.0 currently running on localhost (pid = 5155)  
Verbosity is at least 33  
-- Registered SIP '1500' at 192.168.0.137:7224  
localhost*CLI>
```

Figura 4.5 Registro de usuario SIP 1500 en Asterisk

Posterior al registro del primer usuario procedemos a registrar el segundo usuario en otro computador realizando el mismo proceso del primer usuario con los siguientes datos:

Display Name: 1600

User name: 1600

Password: 1234

Authorization user name: 1600

Domain: 192.168.0.164 (IP del servidor de pruebas)

4.5 Llamada entre usuarios

Una vez autenticados los usuarios procedemos a realizar una llamada desde el usuario con extensión 1600 al 1500, como se muestra en la Figura 4.6.

```
== Using SIP RTP CoS mark 5
-- Executing [1500@internal:1] AGI("SIP/1600-00000000", "videoconferencia.agi")
-- Launched AGI Script /var/lib/asterisk/agi-bin/videoconferencia.agi
-- AGI Script Executing Application: (Dial) Options: (SIP/1500,100,g|S83)
-- Setting call duration limit to 83.000 seconds.
== Using SIP RTP CoS mark 5
-- Called SIP/1500
localhost*CLI>
```

Figura 4.6 Llamada desde usuario 1600 al 1500

Como se puede ver en la imagen el numero marcado con un cuadro rojo representa el numero de segundos que el usuario tiene disponible como saldo, una vez que la llamada finaliza ya sea porque uno de los usuarios lo realizo o porque el saldo

termino automáticamente finaliza la llamada como se puede observar en la Figura 4.7.

```
-- Executing [1500@internal:1] AGI("SIP/1600-00000002", "videoconferencia.agi")
-- Launched AGI Script /var/lib/asterisk/agi-bin/videoconferencia.agi
-- AGI Script Executing Application: (Dial) Options: (SIP/1500,100,g|S(83))
-- Setting call duration limit to 83.000 seconds.
== Using SIP RTP CoS mark 5
-- Called SIP/1500
-- SIP/1500-00000003 is ringing
-- SIP/1500-00000003 answered SIP/1600-00000002
-- Locally bridging SIP/1600-00000002 and SIP/1500-00000003
-- <SIP/1600-00000002>AGI Script videoconferencia.agi completed, returning 4
== Spawn extension (internal, 1500, 1) exited non-zero on 'SIP/1600-00000002'
localhost*CLI>
```

Figura 4.7 Finalización del AGI videoconferencia

En el caso de encontrarse con cero inmediatamente lanza el mensaje de audio pregrabado que indica que no tiene saldo disponible como se puede ver en la Figura 4.8.

```
== Using SIP RTP CoS mark 5
-- Executing [1500@internal:1] AGI("SIP/1600-00000004", "videoconferencia.agi")
-- Launched AGI Script /var/lib/asterisk/agi-bin/videoconferencia.agi
-- AGI Script Executing Application: (Playback) Options: (insuficiente saldo)
-- <SIP/1600-00000004> Playing 'insuficiente_saldo.gsm' (language 'es')
-- <SIP/1600-00000004>AGI Script videoconferencia.agi completed, returning 4
== Spawn extension (internal, 1500, 1) exited non-zero on 'SIP/1600-00000004'
localhost*CLI>
```

Figura 4.8 Reproducción de mensaje de saldo insuficiente

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

- 1) La implementación de un sistema telefónico para la gestión de videoconferencias pre-pagadas realizadas entre las PYMES y sus clientes no presenta dificultades si se siguen las instrucciones detalladas en este trabajo.
- 2) El uso de la librería phpagi para el desarrollo de AGI's ayuda en gran medida puesto que permite personalizar fácilmente la AGI y permite trabajar con un motor de base de datos como es el MySQL.
- 3) La modularización del sistema brinda la oportunidad de incluir nuevos módulos así como también la posibilidad de conectar a otros motores de bases de datos de mayor capacidad.

Recomendaciones

- 1) Al momento de implementar la solución tecnológica se recomienda emplear el sistema operativo CentOS por su estabilidad, seguridad y que es especialmente diseñado para la implementación de servidores.
- 2) Analizar la implementación del concepto de una tarjeta donde, de acuerdo al valor, el usuario puede desde la web recargar su saldo sin que intervenga la central.

3) Analizar la opción de pasar saldo entre clientes.

APENDICE

GLOSARIO

Asterisk.- Software de código abierto liberado bajo licencia GPL que soporta todas las funcionalidades de las centralitas tradicionales/IP.

AGI.- Es una interfaz que sirve de pasarela para comunicarse con Asterisk similar al CGI usado por los servidores Web, que permite el uso de diferentes lenguajes de programación como el Perl, PHP, C entre otros.

AGI es llamado a partir del plan de discado definido en extensions.conf.

CentOS.- Community ENTERprise Operating System. CentOS es una distribución de Linux gratuita que está basada en la distribución Red Hat Enterprise Linux (RHEL).

Mysql.- Es un sistema de gestión de bases de datos (SGBD) multiusuario, multiplataforma y de código abierto.

ODBC.- Es un estándar de acceso a bases de datos que permite mantener independencia entre los lenguajes de programación, sistemas de bases de datos y los sistemas operativos.

Open source.- Denominación para aquellas aplicaciones que tienen su código fuente liberado. En general los programas de código abierto suelen ser libres aunque existen aplicaciones de código abierto que no son libres.

PBX.- Es una central telefónica privada que permite interconectar los teléfonos internos de una empresa. Tiene algunas funciones como son: seleccionar la línea saliente de forma automática, transferir llamadas, realizar conferencias, llevar un control de los números marcados entre otras.

PHP.- Lenguaje de programación interpretado usado generalmente en la creación de contenido dinámico web y aplicaciones para servidores, aunque también es posible crear aplicaciones gráficas utilizando la biblioteca GTK+. Generalmente los scripts en PHP se embeben en otros códigos como HTML. La interpretación y ejecución de los scripts PHP se realiza en el servidor mientras que el cliente (un navegador que solicita una página web) sólo recibe el resultado de la ejecución en HTML.

Permite la conexión a todo tipo de motores de base de datos como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite.

Softphone.- Software que realiza una simulación de teléfono en una computadora, permitiendo así la comunicación con otras computadoras que posean este mismo software, usando un VSP (Proveedor de Servicios de VoIP). Lo más comunes son: Skype, WengoPhone, Gizmo, XLite, Aim, SjPhone, y últimamente Zoiper.

Telefonía IP.- Hace referencia a comunicaciones telefónicas realizadas a través de redes TCP/IP. A diferencia de PSTN, que se compone de señales analógicas y digitales a través de una red con conmutación de circuitos, la telefonía IP utiliza conmutación de paquetes.

Voz sobre IP.- Tecnología que posibilita el uso de redes IPs como medio de transmisión de voz. El concepto consiste en convertir los paquetes de voz analógicos en paquetes digitales y hacerlos transitar por internet. Con un relativo ancho de banda (128Kbps ya se garantiza una buena calidad de voz, siendo de 256Kbps en adelante mejor) es posible usar la capacidad de transporte del protocolo IP para transmitir voz.

ANEXOS

ANEXO 1: CÓDIGO FUENTE

```
#!/usr/bin/php -q

<?php

    set_time_limit(100);

    require('include/phpagi.php');

    $agi = new AGI();

    $agi->answer();

    $extension_origen = $agi->request['agi_callerid'];

    $uniqueid = $agi->request['agi_uniqueid'];

    $extension_destino = $agi->request['agi_extension'];


    $db = 'videoconferencia_db';

    $dbuser = 'root';

    $dbpass = 'admin123';

    $dbhost = 'localhost';

    $saldo = "";

    mysql_connect($dbhost,$dbuser,$dbpass);

    mysql_select_db("$db"); //or die("could not open database");

    $row=mysql_query("select    saldo    from    usuario    where    name    =

'$extension_origen' LIMIT 1");
```

```
if (mysql_num_rows($row)==1){  
    $row=mysql_fetch_array($row);  
    $saldo = $row["saldo"];  
}  
if($saldo>0){  
    $agi->exec_dial("SIP",$extension_destino,"100","g|S(".$saldo.")");  
    mysql_query("CALL actualizar_saldo('$uniqueid','$extension_origen');");  
}else{  
    $agi->exec(Playback,"insuficiente_saldo");  
}  
  
    $agi->hangup();  
?>
```

BIBLIOGRAFÍA

- [1] **Digium, Inc.** *Asterisk*. [En línea] <http://www.asterisk.org>. [Citado el: 2 de Junio de 2011.]
- [2] **Craig Gurnik.** Install LAMP on CentOS 5.6. [En línea] <http://www.cgurnik.com/2011/04/06/install-lamp-linux-apache-mysql-php-on-centos-5-6/>. [Citado: 21 de Junio de 2011.]
- [3] **PHP AGI.** PHP The connection of PHP and Asterisk. *PHP AGI*. [En línea] <http://phpagi.sourceforge.net/>. [Citado el: 20 de Junio de 2011.]
- [4] **VOIP Info.** *VOIP Billing*. [En línea] <http://www.voip-info.org/wiki/view/VOIP+Billing>. [Citado el: 15 de Junio de 2011.]
- [5] **Sokol, Steven.** Beginning Asterisk. *Asterisk*. [En línea] <https://wiki.asterisk.org/wiki/display/AST/Beginning+Asterisk>. [Citado el: 5 de Junio de 2011.]
- [6] **Lono, Save.** *How to Query a Database from MySQL with Asterisk 1.6 ODBC*. [En línea] 25 de Agosto de 2009. <http://www.savelono.com/linux/asterisk-post/how-to-query-a-database-from-mysql-with-asterisk-16-odbc.html>. [Citado el: 25 de Junio de 2011.]
- [7] **VOIP.** *Asterisk phone xten eyeBeam*. [En línea] Enero 4, 2008. <http://www.voip-info.org/wiki/view/Asterisk+phone+xten+eyeBeam>. [Citado: Junio 25, 2011.]
- [8] **VOIP-Info.** *Asterisk. Asterisk*. [En línea] <http://www.voip-info.org/wiki/view/Asterisk>. [Citado el: 10 de Junio de 2011.]

- [9] **Phylevn.** Como implementar un AGI con PHP en un conmutador de VoIP sobre Asterisk. [En línea] http://phylevn.mexrom.net/index.php/blog/show/Como_implementar_un_AGI_con_PHP_en_un_conmutador_de_VoIP_sobre_Asterisk.html. [Citado el: 26 de Junio de 2011.]
- [10] **Landivar, Edgar.** *Ejemplo sencillo de script AGI.* [En línea] <http://es.scribd.com/doc/49810258/205/Ejemplo-sencillo-de-script-AGI>. [Citado el: 29 de Junio de 2011.]
- [11] **AsteriskGuru.** *Asterisk 1.4.0 CLI commands* . [En línea] http://www.asteriskguru.com/tutorials/cli_cmd_14.html. [Citado el: 2 de Julio de 2011.]
- [12] **VOIP.** Asterisk extensions from mysql. [En línea] 30 de Marzo de 2006. <http://www.voip-info.org/wiki/view/Asterisk+extensions+from+mysql>. [Citado el: 3 de Julio de 2011.]
- [13] **VOIP-Info.** *Asterisk cdr mysql.* [En línea] 24 de Agosto de 2010. <http://www.voip-info.org/wiki/view/Asterisk+cdr+mysql>. [Citado el: 5 de Julio de 2011.]
- [14] **VOIP.** Asterisk Gateway Interface (AGI). [En línea] 23 de Junio de 2011. <http://www.voip-info.org/wiki/view/Asterisk+AGI>. [Citado el: 10 de Julio de 2011.]
- [15] **Oliva, Juan.** Asterisk SIP Realtime , extensiones sip desde base de datos. [En línea] 1 de Mayo de 2010.

<http://jroliva.wordpress.com/2010/05/01/asterisk-sip-realtime-extensiones-sip-desde-base-de-datos/>. [Citado el: 15 de Julio de 2011.]