



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA



Sistema Telefónico Automático para Consultas de Deudas y Fechas de Pago

Andrea Solange Freire Morán ⁽¹⁾ Eduardo Arturo López Yaguana ⁽²⁾ Gabriel Astudillo Brocel ⁽³⁾

Facultad de Ingeniería en Electricidad y Computación (FIEC)

Escuela Superior Politécnica del Litoral (ESPOL)

Campus Gustavo Galindo, Km 30.5 vía Perimetral

Apartado 09-01-5863. Guayaquil, Ecuador

afreire@fiec.espol.edu.ec ⁽¹⁾, ealopez@fiec.espol.edu.ec ⁽²⁾

Escuela Superior Politécnica del Litoral (ESPOL) ⁽³⁾, Ingeniero en Telecomunicaciones ⁽³⁾,

gastudil@fiec.espol.edu.ec ⁽³⁾

Resumen

El proyecto expuesto en esta tesis es una de las muchas soluciones para darle a los usuarios de una institución comercial o educativa información de sus movimientos en este caso de sus deudas de una forma ágil, rápida y sencilla con una llamada telefónica y usando un campo clave para identificarse en este caso el número de cédula.

Cuando un usuario realice una llamada a la central telefónica este escuchará un menú y tendrá que ingresar su identificador; de acuerdo a lo que el presione se pueden dar varios casos, los cuales se explican en la descripción del proyecto.

Esta es una solución a muy bajo costo para las instituciones puesto que todo lo que se utiliza para su implementación es software libre.

Se utiliza para la implementación de esta solución como servidor de Voz sobre IP Asterisk, para hacer el código que permite usar la opción Interfaz de Enlace de Asterisk (AGI) que tiene Asterisk se utiliza el lenguaje php, se usa una librería de php "php-agi" que maneja todas las opciones de Asterisk para que puedan ser usadas más fácilmente en el código php y para almacenar los datos se usa el motor para la base de datos Mysql.

Palabras Claves: asterisk, agi, php, php-agi.

Abstract

The project described in this thesis is one of many solutions to give users for commercial or educational institutions information of their movements in this case of debt in an agile, quick and easy way with a phone call and using a key field to identify in this case the identity card number.

When a user makes a call to the telephone exchange this will hear a menu and must enter its ID, according to what it press can take several cases, which are explained in the description of the project.

This is a very low cost solution for institutions since all that is used for implementation is open source.

Asterisk is used to implement this solution as VoIP server, to make the code that allows the option to use Asterisk Gateway Interface (AGI) that have Asterisk, PHP language is used, use a php library "php-agi" that handles all Asterisk options so they can be used more easily in php code and MySQL database is used as data storage engine.



1. Introducción

Las tecnologías avanzan a una velocidad muy acelerada y cambian constantemente, podemos ver el cambio que están teniendo actualmente los sistemas analógicos a sistemas digitales, tomando como tema particular las centrales telefónicas y una de las tecnologías emergentes Voz sobre IP sobre la cual se pueden adaptar todos los servicios de las centrales telefónicas analógicas.

Por ejemplo, una institución comercial o de educación la cual requiere una forma más eficaz para que sus usuarios consulten su información en la empresa en este caso sus deudas; este proyecto consiste en la implementación de un sistema de consultas de deudas y fechas de pago por teléfono.

Tomando en cuenta la arquitectura de red de la empresa podemos adecuar un servicio de Voz sobre IP de varias formas: una Arquitectura Distribuida, Centralizada o Mixta; para la implementación de esta solución usaremos la arquitectura centralizada que consiste en un servidor que tiene toda la información de la red de Voz sobre IP de la institución.

Se asume que en la empresa hay instalado un servidor de voz sobre IP asterisk, sino lo hay, se lo pondrá en funcionamiento y para implementar la solución se utilizara un código en php usando la librería php-agi y base de datos mysql que son de código abierto.

2. Metodología

Los pasos para la realización de este proyecto fueron:

1. Instalación de la base de datos mysql.
2. Iniciar el servicio mysql con los respectivos permisos para su uso.
3. Instalación de php 5.
4. Instalación del programa Asterisk sobre una distribución Linux para el proyecto se utilizo la distribución CentOS 5.
5. Configuración de teléfonos IP con usuarios SIP para pruebas.
6. Instalación y configuración de softphone Zoiper.
7. Diseño e implementación del sistema usando php-agi.

3. Descripción del Proyecto

El proyecto consistió en la implementación de un sistema de consultas de deudas y fechas de pago en un servidor de Voz sobre IP como asterisk.

Este sistema funciona de la siguiente manera:

El usuario llama a la central y cuando le contestan escuchara un menú pidiéndole que digite su número de cédula para identificarse o que presione las teclas 999 seguidas de la tecla numeral para salir del sistema; si el usuario no presiona nada volverá a reproducirse la grabación hasta que se ejecute una de las dos acciones.

Una vez ingresado el número de cédula este puede tener tres tipos de respuesta. El primero es un mensaje indicándole que su cédula no está registrada si el número digitado no se encuentra en la base de datos.

El segundo es un mensaje indicándole que no tiene deudas si el número ingresado no registra ninguna deuda en la base de datos. El tercero, cuando la cédula ingresada tenga deudas, se escuchara el valor de la deuda y la fecha de pago seguida de un mensaje pidiéndole que presione la tecla 1 para confirmar que el usuario ha escuchado el mensaje de la deuda y así sucesivamente si el usuario tiene más de una deuda.

4. Asterisk y php agi

Asterisk es un PBX por software de código abierto, creado por Digium, Inc. y una base de usuarios y desarrolladores en continuo crecimiento. Digium desarrolla código fuente y hardware de telefonía de bajo costo que funciona con asterisk. Asterisk funciona en Linux y otras plataformas Unix con o sin hardware que conecte tu servidor con la red global tradicional de telefonía, la PSTN. 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. En php existe una clase con varias funciones previamente definidas que hacen más fácil la tarea de elaborar códigos PHP para Asterisk. Lo único que hay que hacer es descargarla, descomprimirla e instalarla (copiarla) dentro del directorio `/var/lib/asterisk/agi-bin/`, para poder utilizar las funciones que en esta clase se definen, se debe incluir la clase en el código mediante la sentencia:

```
require "/var/lib/asterisk/agi-bin/phpagi-1.12/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.



5. Implementación del proyecto

5.1. Hardware

Este sistema cuenta con un servidor y un teléfono que soporta voz sobre IP.

El servidor que se utilizó es una computadora con las siguientes características:

- Procesador arquitectura x86 de Intel Core Duo de 2.8 GHZ
- Tarjeta de Red de 10/100 Mbps.
- Ram de 2 GB.
- Disco Duro de 80 GB.
- Tarjeta analógica Marca Digium TDM410p 4 puertos.

Utilizamos el teléfono de voz sobre IP Grandstream GXP2000, del cual utilizamos sus conectores RJ-45 para conectar el teléfono a la computadora y a la red.

5.2. Software

En el servidor se instalaron los sistemas operativos y programas:

- Sistema Operativo -Centos 5.2.
- Software IP PBX-Asterisk 1.4.
- Sistema de Gestión de Bases de datos-MySQL.
- Paquete de desarrollo-Java Development Kit 1.5.
- Paquete de desarrollo-php 5.2

5.3. Instalación

Para la realización de esta tesis se procedió de la siguiente manera:

Actualizar el sistema operativo.

Instalar los paquetes para poder instalar asterisk: gcc, openssl, bison, ncurses, openssl-devel.

Instalar los paquetes de la base de datos: mysql-server mysql-devel mysql.

Instalar los paquetes de php: php php-mysql.

Descargar, descomprimir e instalar los paquetes de asterisk: Libpri Dahdilinux Dahditools Asterisk AsteriskAddons .

Todos estos paquetes se los guarda en /usr/src .

Luego de esto se comienza a extraer por medio de tar -xvzf.

En el servidor se instalaron

Se configura y se instala con el comando ./configure y el comando make.

5.4. Estructura de la base de datos

Este sistema cuenta con una base de datos almacenada en MySQL. Esta base de datos consiste en dos tablas:

1. Clientes
2. Deudas

Por tanto que cada cliente está interesado en conocer cuantas deudas tiene y las fechas de pago de cada deuda, los datos se encuentran almacenados en estas dos tablas.

Un cliente puede estar relacionado con ninguna o muchas deudas, mientras que una deuda siempre se relaciona con un cliente.

5.5. Lenguaje de Descripción de Datos del Sistema.

A continuación se detalla la creación de la base de datos de este sistema, todos estos comandos son ejecutados desde la línea de comandos del terminal de MySQL.

La contraseña del usuario root está en blanco para ingresar a MySQL.

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

La base de datos tiene por nombre "SistemaConsultas" y se deberá emplear el comando:

```
create database SistemaConsultas;
```

Se crea la tabla Clientes:

```
create table Clientes
```

```
(  
  IDCliente int not null auto_increment,  
  Nombres varchar(40) not null,  
  Apellidos varchar(40) not null,  
  telefono varchar(40),  
  celular varchar(40),  
  edad int,  
  sexo varchar(40),  
  domicilio varchar(200),  
  cédula varchar(40) not null,  
  estado int not null,  
  primary key (IDCliente)  
)ENGINE=INNODB;
```

Se crea la tabla Deudas:

```
create table Deudas
```

```
(  
  IDDeuda int not null auto_increment,  
  IDCliente int not null,  
  monto double(7,2),  
  FechaPago date,  
  descripcion varchar(200),  
  cancelado int not null,  
  estado int not null,  
  Confirmacion int,  
  primary key (IDDeuda),  
  foreign key (IDCliente) references Clientes  
  (IDCliente) on delete cascade  
)ENGINE=INNODB;
```

El sistema tiene la restricción de que la deuda de un cliente está en el rango entre 0 a 99999.99 dólares, y cada deuda tiene una referencia a un cliente, si un



cliente se elimina de la base de datos, automáticamente se eliminan del sistema las deudas de ese cliente.

5.6. Configuración de Archivos en Asterisk

5.6.1. 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.

```
[general]
rtcachefriends=yes      qualify=yes
rtupdate=no             calltransfer=no
rtautoclear=yes        callforwarding=no
srvlookup=yes          [901]
disallow=all           type=friend
allow=alaw             secret=901
allow=ulaw             qualify=yes
allow=gsm              nat=yes
language=es           host=dynamic
callwaiting=no        canreinvite=no
                      context=incoming
```

5.6.2. Configuración de los archivos system.conf y chan_dahdi.conf.

Estos archivos se usan para la configuración de los canales dahdi, que son los que se emplean para el uso de la tarjeta analógica tdm410p de 4 puertos. Para nuestro caso, usaremos el puerto 4 como fxo con señalización fxs. El archivo system.conf se ubica en la ruta /etc/dahdi y el archivo chan_dahdi.conf se encuentra en /etc/asterisk.

```
chan_dahdi.conf
[channels]
usecallerid=yes        echotraining=yes
hidecallerid=no       immediate=no
callwaiting=no        group=1
threewaycalling=yes  context=internal
transfer=yes          signaling=fxs_ks
echocancel=yes       channel => 4

system.conf
fxsks=4
echocanceller=mg2,4
loadzone=us
defaultzone=us
```

5.6.3. Configuración de los archivos extension.conf

Se encuentra en la ruta /etc/asterisk y consiste en el plan de marcado de este sistema.

La función principal de este archivo fue de invocar al código donde se encuentra almacenado el programa principal.

; contexto general sirve para establecer los parámetros generales

; del plan de marcado.

```
[general]
```

```
autofallthrough=no
```

```
clearglobalvars=no
```

; contexto "internal" sirve para especificar el procedimiento a ;realizarse para comunicarse con alguna de las extensiones en el interior de la PBX.

```
[internal]
```

```
exten => 789,1,AGI(proyecto.php)
```

5.7. Descripción general del programa

El sistema recibe una llamada, para lo cual si la extensión marcada es "789" el archivo extensions.conf invoca al código "proyecto.php".

El sistema contesta la llamada y se reproduce un mensaje de bienvenida donde se le indica al cliente que ingrese su número de cédula o si quiere salir del sistema que digite "999". El sistema espera a que el cliente ingrese su cédula y ejecuta una consulta a la base de datos para verificar si el cliente está registrado y si tiene deudas. Una vez que la base de datos retorna los datos de las deudas y fechas de pago del cliente, se reproducen unas grabaciones que le indican al cliente la cantidad de deudas que tiene, el monto y la fecha de pago de cada deuda. Por cada deuda se reproduce una grabación que le indica al cliente que digite la tecla "1" para confirmar que el cliente escuchó la grabación. Si digitó "1" se graba en la base de datos, en la tabla de Deudas, en el campo Confirmación un "1", que indica que el cliente si escuchó la grabación, de lo contrario el campo Confirmación permanece con un valor de "0", que quiere decir que el cliente no ha confirmado que ha escuchado una de sus deudas. Si el cliente desea salir del sistema puede digitar "999" y con esto se reproduce un mensaje de despedida y se termina la llamada.

Las grabaciones fueron realizadas con Asterisk y se encuentran en la ruta /var/lib/asterisk/sounds.

Las voces que vienen pregrabadas y que están en español también se encuentran en la ruta /var/lib/asterisk/sounds.

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

5.8. Casos del sistema

- El cliente no digite ningún número: El sistema esperará siete segundos para que el cliente ingrese



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA



su cédula, sino ingresa ningún número, volverá a reproducir la grabación inicial.

- El cliente ingresa una cédula no registrada en la base de datos: El sistema hará la consulta a la base de datos, pero al no encontrarse registrada su cédula se reproducirá una grabación indicando que el cliente no se encuentra registrado en el sistema. Luego se reproduce la grabación inicial.
- El cliente ingresa su cédula que sí se encuentra registrada en el sistema pero no tiene deudas: El sistema hará la consulta a la base de datos, entonces el sistema reproduce una grabación que le indica al cliente que no posee deudas. Luego se reproduce la grabación inicial.
- El cliente ingresa su cédula que sí está registrada y también posee deudas: Mediante grabaciones, el sistema le indica al cliente la cantidad de deudas que posee, luego le indica deuda por deuda, el monto de la deuda y la fecha a pagar. Por cada deuda el sistema le pide al cliente que digite un "1" para confirmar que escuchó la deuda, ésta confirmación se la registra en la base de datos, pero si el cliente no escuchó la deuda, no se altera la base de datos. Luego se reproduce la grabación inicial.

5.9. Código fuente

```

1 #!/usr/bin/php -q
2 <?php
3 set_time_limit(30);
4 error_reporting(E_ALL);
5 require ('phpagi-2.14/phpagi.php');
6 //creo mi conexion
7 $con=mysql_connect('localhost','root','labtelecom09')
8 or die(mysql_error());
9 //selecciono mi bd
10 mysql_select_db('SistemaConsultas',$con)or die(mysql_error());
11
12 $agi = new AGI();
13 $agi->answer();
14 $agi->exec(Playback,"Bienvenida");
15 do{
16     $result = $agi->get_data("Menu", 7000,10);
17     $nummarcado = $result['result'];
18 //Verifico si el número digitado es la tecla para salir del sis
19 if($nummarcado!='999' && $nummarcado!= null)
20 {
21     $sql="SELECT * FROM Clientes WHERE cédula='$nummarcado'";
22     $res=mysql_query($sql,$con) or die('ERROR SQL');
23     $r1=mysql_fetch_array($res);
24
25     if($r1['cédula']==$nummarcado)
26     {
27         $sql="SELECT * FROM Deudas WHERE
28 IDCliente='".$r1['IDCliente']."' and cancelado=0";
29 $res2=mysql_query($sql,$con)or die (mysql_error());
30 if(mysql_num_rows($res2)== 0)
31 {
32     $agi->exec(Playback,"SinDeudas");
33 }
34 //si llega a este if es porque si esta la CI y tiene deuda
35 else
36 {
37     $agi->exec(Playback,"UstedTiene");
38 $CantDeudas = mysql_num_rows($res2);
39 $agi->say_number($CantDeudas);
40 $agi->exec(Playback,"Deudas");
41
42     while ($r2=mysql_fetch_array($res2))
43     {
44         $IDDeuda=$r2['IDDeuda'];

```

```

45
46     $agi->exec(Playback,"UstedDebe");
47     $monto=$r2['monto'];
48     $fechap=$r2['FechaPago'];
49
50     $montos=explode("-", $monto);
51     $fechas=explode("-", $fechap);
52
53     $agi->say_number($montos[0]);
54     $agi->exec(Playback,"DolaresCon");
55
56
57     if($montos[1]=='00')
58     {
59         $agi->say_number(0);
60     }
61
62     else
63     {
64         $agi->say_number($montos[1]);
65     }
66
67     $agi->exec(Playback,"Centavos");
68     $agi->exec(Playback,"ydebepagarantesde");
69     $agi->say_number($fechas[2]);
70     $agi->exec(Playback,"de");
71     switch($fechas[1])
72     {
73         case 1: $cad="Enero"; break;
74         case 2: $cad="Febrero"; break;
75         case 3: $cad="Marzo"; break;
76         case 4: $cad="Abril"; break;
77         case 5: $cad="Mayo"; break;
78         case 6: $cad="Junio"; break;
79         case 7: $cad="Julio"; break;
80         case 8: $cad="Agosto"; break;
81         case 9: $cad="Septiembre"; break;
82         case 10: $cad="Octubre"; break;
83         case 11: $cad="Noviembre"; break;
84         case 12: $cad="Diciembre"; break;
85     }
86     $agi->exec(Playback, $cad);
87     $agi->exec(Playback,"del");
88     $agi->say_number($fechas[0]);
89 //Mensaje de Confirmacion
90
91     $confirmacion = $agi->get_data("MensajeConfirmacion", 7000,1);
92     $num = $confirmacion['result'];
93     if($num==1)
94     {
95         $sql2="update Deudas set Confirmacion=1 WHERE IDDeuda=
96 $IDDeuda";
97         $res=mysql_query($sql2,$con) or die ('ERROR SQL');
98     }
99 }
100
101 }
102 //Si no existe la cédula digitada en la base se reproduce una
103 //grabación que no está registrado
104 {
105     else{ $agi->exec(Playback,"CédulaNoReg");
106 }
107 }while($nummarcado!='999');
108 $agi->exec(Playback, "Despedida");
109 $agi->hangup();
110 ?>

```

5.10. Funciones sobre el plan de marcado

error_reporting(E_ALL);-Crea un log con todos los errores ocurridos durante la ejecución del código.

\$agi = new AGI();-Crea una nueva instancia de la clase agi.

require ('phpagi-2.14/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->get_data("ArchivodeAudio", tiempoEspera,MaxDigitos);-Captura la extensión marcada luego de reproducirse un archivo de audio dentro de un tiempo de espera, el tercer parámetro indica la cantidad máxima de dígitos que se pueden ingresar.

agi->say_number(\$Numero);-Reproduce en audio el número que se le ha enviado.

\$arreglo=explode("Caracter",\$cadena);-Se le envía una cadena y retorna un arreglo de los elementos de la cadena separados por el carácter que se envía como primer parámetro.

\$agi->hangup();-Termina la llamada.

5.11. Descripción de archivos de audio

Las grabaciones fueron realizadas utilizando Asterisk, con la función "Record" y llamando a la extensión 999. La función "Playback" nos permite escuchar lo que acabamos de grabar. Estos archivos están almacenados en la ruta /var/lib/asterisk/sounds y fueron grabados de la siguiente manera:

```
exten => 999,1,Record(NombredelArchivo.gsm)
exten => 999,2,Playback(${RECORDED_FILE})
exten => 999,3,Hangup()
```

Los siguientes son los archivos de audio grabados utilizando Asterisk, todos tienen el formato .gsm:

- Bienvenida
- SinDeudas
- UstedTiene
- Deudas
- UstedDebe
- DolaresCon
- Centavos
- ydebepagarantesde
- de
- del
- MensajeConfirmacion
- CédulaNoReg
- Despedida

Otros archivos de audio que se emplean en este proyecto vienen pregrabados en Asterisk, y los utiliza la librería "phpagi.php" junto con la función "text-to-sound" para reproducir texto a audio.

5.12. Sistema de administración de la base de datos.

Se creó un sistema que administra la base de datos donde se encuentra almacenada toda la información concerniente a los clientes y sus deudas. Este sistema puede ingresar, modificar, consultar y eliminar los registros de los clientes y sus deudas. Este sistema fue hecho en Java utilizando la librería "mysql-connector-java5.1.5-bin.jar" y lo pueden utilizar los administradores del sistema.

IDDeuda	Monto	Fecha de Pago	Descripcion	Cancelado	Confirmacion
12	350.0	2010-01-01	registro de P...	Si	No
13	487.05	2010-10-07	Registro del ...	Si	No

Figura 1. Consulta de deudas de un cliente

Cliente

Nombres:

Apellidos:

Edad:

Telefono:

Celular:

Domicilio:

Cedula:

Sexo:

Figura 2. Modificación de los datos de un cliente

6. Funcionamiento y Pruebas del Proyecto

6.1. Activación del Servicio de Mysql

Para que el sistema pueda hacer las consultas pertinentes, MySQL debe estar activado:

```
service mysqld start -u root -p
```

6.2. Ingreso de usuarios y deudas

Antes de utilizar el sistema, ingresamos dos usuarios a la base de datos desde el programa que administra la base de datos del sistema.

Para este ejemplo se van a ingresar dos deudas a un usuario, y el otro usuario quedará registrado en la base de datos pero sin deudas.

Con ambos usuarios registrados y con las deudas registradas para uno de los usuarios ahora procedemos a probar el sistema.

Figura 3. Ingreso de un usuario a la base de datos

Figura 4. Ingreso de una deuda a un usuario

6.3. Activación de asterisk

```

Asterisk Console on 'localhost.localdomain' (pid 4011)
Asterisk Console on 'localhost.localdomain' (pid 4011) | root@localhost:~
-- Registered application 'VoiceMailMain'
-- Registered application 'MailboxExists'
-- Registered application 'VMAuthenticate'
-- Registered custom function 'MAILBOX_EXISTS'
-- Manager registered action VoicemailUsersList
app_voicemail.so => (Comedian Mail (Voicemail System))
-- Registered file format vox, extension(s) vox
format_vox.so => (OggSPEEX_VOX (OggSPEEX) File Format)
-- Parsing /etc/asterisk/codecs.conf: -- Found
-- codec_g726: using generic PLC
-- Registered translator 'g726slin' from format g726 to slin, cost 1000
-- Registered translator 'slin_g726' from format slin to g726, cost 999
-- Registered translator 'g726aal2to1in' from format g726aal2 to slin, cost 1
-- Registered translator 'linlog726aal2' from format slin to g726aal2, cost 1
-- Registered translator 'g726aal2log726' from format g726aal2 to g726, cost 1
-- Registered translator 'g726log726aal2' from format g726 to g726aal2, cost 1
codec_g726.so => (ITU G.726-32kbps G726 Transcoder)
-- Registered custom function 'SMAI'
func_smai.so => (SMAI-3 computation (smaiplan function))
-- Manager registered action PlayDTMF
-- Registered application 'SendDTMF'
app_senddtmf.so => (Send DTMF digits Application)
Asterisk Ready:
-- Parsing /etc/asterisk/call.conf: -- Found
[CLI] [Apr 9 16:18:21] NOTICE[4023]: chan_sip.c:20257 sip_poke_noanswer: Peer '901' is now UNREACHABLE! Last qualify: 0
  
```

Figura 5. Activación de Asterisk

6.4. Llamada al sistema consultas

Se marca la extensión 789 y el sistema contestará la llamada. Luego el sistema reproduce el mensaje de bienvenida y espera siete segundos para que el usuario digite su cédula.

El sistema reproduce grabaciones de audio a los distintos casos de ingreso de cédulas que se den, a continuación se mostrará el reporte que emite Asterisk a cada uno de estos casos.

6.5. Respuesta del sistema al ingreso de una cédula registrada con deudas

Una vez que hemos llamado al sistema e ingresado una cédula registrada, el sistema busca las deudas del cliente y reproduce las grabaciones pertinentes de acuerdo al resultado.

```

Asterisk Console on 'localhost.localdomain' (pid 2821)
Asterisk Console on 'localhost.localdomain' (pid 2821) | root@localhost:~
-- Using SIP RTP CoS mark 5
-- Executing [2@internal:1] AGI('SIP/901-08a03908', 'proyecto.php') in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/proyecto.php
-- AGI Script Executing Application: (Playback) Options: (Bienvenida)
--<SIP/901-08a03908> Playing 'Bienvenida.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'Menu.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (UstedTiene)
--<SIP/901-08a03908> Playing 'UstedTiene.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'digits/2.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (Deudas)
--<SIP/901-08a03908> Playing 'Deudas.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'digits/1.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (UstedDebe)
--<SIP/901-08a03908> Playing 'UstedDebe.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'digits/2.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (DolaresCon)
--<SIP/901-08a03908> Playing 'DolaresCon.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'digits/0.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (Centavos)
--<SIP/901-08a03908> Playing 'Centavos.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (ydebepagarantesde)
--<SIP/901-08a03908> Playing 'ydebepagarantesde.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'digits/4.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (de)
--<SIP/901-08a03908> Playing 'de.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (Abril)
--<SIP/901-08a03908> Playing 'Abril.gsm' (language 'en')
  
```

Figura 6. Respuesta del sistema al ingreso de una cédula en el servidor

6.6. Confirmación del usuario

Cuando el usuario ha escuchado la deuda, el sistema reproduce una grabación indicándole que digite uno si escuchó la grabación. Cuando el usuario confirma que escuchó la deuda, se graba en la base de datos que esa deuda sí fue escuchada.

```

Asterisk Console on 'localhost.localdomain' (pid 2821)
Asterisk Console on 'localhost.localdomain' (pid 2821) | root@localhost:~
--<SIP/901-08a03908> Playing 'DolaresCon.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'digits/3.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'digits/hundred.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'digits/40.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (Centavos)
--<SIP/901-08a03908> Playing 'Centavos.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (ydebepagarantesde)
--<SIP/901-08a03908> Playing 'ydebepagarantesde.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'digits/12.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (de)
--<SIP/901-08a03908> Playing 'de.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (Marzo)
--<SIP/901-08a03908> Playing 'Marzo.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (del)
--<SIP/901-08a03908> Playing 'del.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'digits/2.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'digits/thousand.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'digits/10.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'MensajeConfirmacion.gsm' (language 'en')
--<SIP/901-08a03908> Playing 'Menu.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (Despedida)
--<SIP/901-08a03908> Playing 'Despedida.gsm' (language 'en')
[Apr 9 14:54:29] NOTICE[2833]: chan_sip.c:18872 handle_request_subscribe: Received SIP subscribe for peer without mailbox: 901
--<SIP/901-08a03908> AGI Script proyecto.php completed, returning -1
[CLI]
  
```

Figura 7. Confirmación del usuario

```

mysql> select * from Deudas;
+-----+-----+-----+-----+-----+-----+-----+
| IDDeuda | IDCliente | monto | FechaPago | descripcion | cancelado | estado | C
onfirmacion |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | 1 | 2.000 | 0800-00-00 | Registro de Voip | 0 | 1 | 1
| 2 | 1 | 250.540 | 2010-03-12 | Registro de JSP | 0 | 1 | 1
| 3 | 2 | 350.540 | 2010-02-12 | Registro de .ASP | 1 | 1 | 1
| 4 | 2 | 2.050 | 2010-05-06 | Registro de Mantenimiento | 0 | 1 | 1
| 5 | 2 | 2.500 | 2010-11-30 | Registro de Servidores | 0 | 1 | 1
+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.01 sec)
mysql>
  
```

Figura 8. Confirmación del usuario en la base de datos

```

-- Using SIP RTP CoS mark 5
-- Executing [2@internal:1] AGI("SIP/901-089f43d0", "proyecto.php") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/proyecto.php
-- AGI Script Executing Application: (Playback) Options: (Bienvenida)
--SIP/901-089f43d0> Playing 'Bienvenida.gsm' (language 'en')
--SIP/901-089f43d0> Playing 'Menu.gsm' (language 'en')
[Apr 9 13:20:51] NOTICE[2833]: chan_sip.c:18872 handle_request_subscribe: Received SIP subscribe
for peer without mailbox: 901
-- AGI Script Executing Application: (Playback) Options: (CedulaNoReg)
--SIP/901-089f43d0> Playing 'CedulaNoReg.gsm' (language 'en')
--SIP/901-089f43d0> Playing 'Menu.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (Despedida)
--SIP/901-089f43d0> Playing 'Despedida.gsm' (language 'en')
--SIP/901-089f43d0> AGI Script proyecto.php completed, returning -1
[Apr 9 13:21:51] NOTICE[2833]: chan_sip.c:18872 handle_request_subscribe: Received SIP subscribe
for peer without mailbox: 901
[Apr 9 13:22:51] NOTICE[2833]: chan_sip.c:18872 handle_request_subscribe: Received SIP subscribe
for peer without mailbox: 901
  
```

Figura 10. Respuesta del sistema a una cédula no registrada

6.7. Respuesta del sistema a una cédula registrada sin deudas

El sistema reproducirá una grabación indicando que el usuario no posee deudas pero si está registrado.

```

-- Using SIP RTP CoS mark 5
-- Executing [2@internal:1] AGI("SIP/901-089f43d0", "proyecto.php") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/proyecto.php
-- AGI Script Executing Application: (Playback) Options: (Bienvenida)
--SIP/901-089f43d0> Playing 'Bienvenida.gsm' (language 'en')
--SIP/901-089f43d0> Playing 'Menu.gsm' (language 'en')
[Apr 9 14:15:38] NOTICE[2833]: chan_sip.c:18872 handle_request_subscribe: Received SIP subscribe
for peer without mailbox: 901
-- AGI Script Executing Application: (Playback) Options: (SinDeudas)
--SIP/901-089f43d0> Playing 'SinDeudas.gsm' (language 'en')
--SIP/901-089f43d0> Playing 'Menu.gsm' (language 'en')
-- AGI Script Executing Application: (Playback) Options: (Despedida)
--SIP/901-089f43d0> Playing 'Despedida.gsm' (language 'en')
--SIP/901-089f43d0> AGI Script proyecto.php completed, returning -1
  
```

Figura 9. Respuesta del sistema a una cédula registrada sin deudas

6.8. Respuesta del sistema a una cédula no registrada

El sistema reproducirá una grabación indicando que el usuario no posee deudas pero si está registrado.

6.9. Fin de la llamada

La llamada termina cuando el usuario digita 999 o cuelga el teléfono.

7. Conclusiones

- La implementación de un sistema telefónico automatizado de consultas de uso sencillo que satisfaga las necesidades del usuario, es fácil de realizar con las funciones para crear IVR's que nos ofrece asterisk.
- Se pueden desarrollar funciones a medida de una institución usando la interfaz de desarrollo de asterisk AGI, con el uso de la librería phpagi se facilita en gran manera el desarrollo de aplicaciones complejas sobre Asterisk, permitiéndonos trabajar con un motor de base de datos tan utilizado como lo es MySQL.
- El uso del software libre nos proporciona una gran variedad de herramientas y librerías, tiempos menores de desarrollo lo que nos permite la implementación de sistemas útiles, eficientes, robustos, adaptables a la institución, escalables y a un bajo costo.
- Para administrar la base de datos que maneja el sistema de consultas se necesita desarrollar una aplicación que permita poder ejecutar todas las operaciones sobre ella.



8. Recomendaciones

- Para utilizar este sistema con una red externa es necesario poner la extensión del sistema en el contexto “incoming” y en el archivo chan_dahdi.conf especificar que se trabaja con el contexto “incoming”.
- Utilizar el sistema operativo CentOS puesto que es bastante estable y es dedicado a los servidores.
- Levantar los servicios que se utilizan con este sistema, tales como MySQL, como administrador para poder acceder a los datos.
- Descargar las voces en español que trabajan con este sistema y colocarlas en la ruta var/lib/asterisk/sounds para su correcto funcionamiento.
- Renombrar la carpeta que contiene los archivos de audio de las voces en inglés.
- Ubicar la ruta donde se encuentre almacenado el compilador de php.
- Este sistema puede ser modificado en un corto o mediano plazo según como vayan cambiando los requerimientos de los usuarios para hacer distintas consultas. Por ejemplo, se podría añadir al sistema la opción de consultar los seminarios que están próximos a ser dictados, consultar cuándo se van a dictar y su costo y hasta cuándo se puede pagar. También se puede añadir la opción de reservar un registro de un seminario al digitar una tecla e inmediatamente se le indicaría al usuario hasta cuando está vigente su cupo.

9. Referencias

- [1] Matthew Asham, Pagina oficial de la clase phpagi, <http://phpagi.sourceforge.net>, 2004.
- [2] Integración de Asterisk usando AGI y AMI, http://www.asteriskguide.com/mediawiki/index.php/Integraci%C3%B3n_de_Asterisk_usando_AGI_y_AMI, 2010.
- [3] Loris Santamaria, Introducción a Asterisk y la telefonía IP, <http://www.solucionesit.com.ve/tisol/files/PresentacionAsterisk.pdf>, 2010.
- [4] Portal de informática, internet, tecnologías y web, <http://www.alegsa.com.ar/>, 2009.
- [5] Introducción al Asterisk, http://www.asteriskguide.com/mediawiki/index.php/Introducci%C3%ADn_al_Asterisk, 2009.
- [6] Universidad de Quevedo, Sistemas de Telecomunicaciones, <http://www.uteq.edu.ec/facultades/empresariales/infor>

matica/tutoriales/siscomunicacionesii2004/materia/e1yt1.htm, 2004.

[7] Proyecto de grado profesional de un estudiante de la UNAD, Diferencia entre puertos FXS y FXO, <http://proyectopica.wordpress.com/2008/11/27/diferencia-entre-puertos-fxs-y-fxo/>, 2008.

[8] Kioskea, Firmas electrónicas, <http://es.kioskea.net/contents/crypto/signature.php3>, 2010.

[9] Entender VoIP (Voz sobre IP), <http://www.informatica-hoy.com.ar/voz-ip-voip/Entender-VoIP-Voz-sobre-IP.php>, 2010.

[10] VOIP-Info.org, Asterisk AGI, <http://www.voip-info.org/wiki/view/Asterisk+AGI>, 2003.

[11] Wikipedia, Asterisk, <http://es.wikipedia.org/wiki/Asterisk>, 2010.

[12] asterisk_agi[VoIP en español], Asterisk AGI, http://voip.megawan.com.ar/doku.php/asterisk_agi, 2006.

[13] Van Meggelen J., Smith J. y Madsen L., Asterisk The Future of Telephony, Editorial O'Reilly Media, 2005.