

# “Diseño e Implementación de un Sistema Telefónico Interactivo que me Permita Automatizar la Asignación y Consulta de Citas Médicas”

Cristina Alvarado Tumbaco  
Stephanie Kravarovich Granados  
Ing. Gabriel Astudillo Brocel  
Facultad de Ingeniería en Electricidad y Computación  
Escuela Superior Politecnica de Litoral (ESPOL)  
Campus Gustavo Galindo, Km 30.5 Vía perimetral  
Apartado 09-01-5863. Guayaquil – Ecuador  
[mcalvara@espol.edu.ec](mailto:mcalvara@espol.edu.ec)  
[krava1987@hotmail.com](mailto:krava1987@hotmail.com)  
[gastudil@espol.edu.ec](mailto:gastudil@espol.edu.ec)

## Resumen

*Este proyecto consistió en una solución para los centros médicos u hospitales, donde se busca otorgar una atención de manera eficaz y eficiente al momento, que los pacientes soliciten una cita, para esto se implemento una PBX cuyo software libre instalado es llamado Asterisk, su función es como la central telefónica en donde se receptaron las llamadas que en conjunto con el sistema IVR se presentó un menú sencillo, con opciones que debió ser escogida por el paciente, haciendo uso de un teléfono digital, de esta forma se alcanzó a disminuir las llamadas perdidas, como también se consiguió la disponibilidad de uso a este sistema a un servicio de 24 horas del día y los 7 días de la semana.*

*Los requerimientos de hardware no se basó en mayores gastos, un ejemplo es el del teléfono IP el cual fue remplazado por el Softphone también perteneciente al grupo de software libre y al servidor le fue instalado el sistema operativo Linux, de la distribución conocida como CENTOS.*

**Palabras Claves:** IVR (Interactive Voice Response), Software Libre, Linux, Centos, Centro Médicos.

## Abstract

*This project consisted of a solution to the medical centers or hospitals, where it seeks to provide care effectively and efficiently at the time that patients request an appointment for that which was implemented a free software PBX called Asterisk is installed, its function is like the telephone where the calls will be received in conjunction with the IVR system was introduced a simple menu with options should be chosen by the patient, using a digital phone, in this way managed to reduce calls lost, and also won the availability of use this system to service 24 hours a day, 7 days a week.*

*The hardware requirements are not based on higher expenses, one example is the IP phone which was replaced by Softphone also belonging to the group of free software and the server it was installed Linux operating system, distribution known as CENTOS.*

**Keywords:** IVR(Interacitve Voice Respons), Software open source, Centos, Medical Center

## 1. Introducción

En la actualidad, el avance tecnológico provoca una variedad de cambios en todo el mundo, con el fin

de satisfacer en sus necesidades cotidiana al ser humano, en este proyecto se tendrá como objetivo brindar asistencia y facilitar la solicitud de una cita médica evitando de esta forma que el paciente se deba acercar previamente al centro médico para obtener un turno, solo bastará que el usuario realice una llamada en un teléfono digital y siga las instrucciones que indica el menú.

La implementación y el Diseño se fundamenta en un sistema telefónico interactivo en el cual tendrá como objetivo el de gestionar las citas, disminuir las llamadas perdidas como también las llamadas en espera y aumentar la eficacia en la atención, otorgando así una mejora en el servicio.

El proyecto se realiza tomando en consideración el factor económico, donde se justifica la instalación de un sistema operativo OPENSOURCE como lo es CENTOS5.4 en donde se le agrega las funcionalidades de una central telefónica (PBX), un software libre llamado ASTERISK, en él se desarrollará la configuración de un IVR (Interactive voice response) que permite exponer un sencillo menú, el cual guiará paso a paso al usuario y así poder acceder a sus diversas opciones con solo presionar uno o varios botones de un teléfono digital y antes de que finalice su llamada tendrá como resultado la confirmación de su cita repitiendo la fecha y la hora asignada

## 2. Metodología

- Instalación de Centos 5.6
- Instalación de la base de datos MySQL.
- Instalación de php 5
- Instalación de Asterisk
- Configuración de teléfono IP con un usuario SIP
- Implementación del programa que maneja la base de datos
- Diseño e implementación del sistema usando php-agi

## 3. Implementación

Cabe indicar que Asterisk es uno de los proyectos de software libre más exitosos, ya que se pueden hacer cambios a la plataforma para adaptarlo a las necesidades requeridas, con lo cual, se puede decir que es un sistema escalable y se aplica a cualquier estándar y a casi todos los protocolos; debido a esto podemos decir que las centrales telefónicas tradicionales quedarían a un segundo plano, ya que serían reemplazadas por un computador.

El computador que reemplazaría a la central telefónica analógica mediría los costos que estos varían de acuerdo con el tipo de hardware que es requerido para soportar la gran concurrencia de llamadas telefónicas que atendería nuestro sistema telefónico, pero estos gastos siempre serán menores que implementar una central telefónica tradicional.

## 3.1. Hardware

Al momento de seleccionar las características de las herramientas basadas en hardware esto, solo dependerá de las funcionalidades a la que se le va a integrar al sistema, para así llegar a un correcto desarrollo y ambiente eficiente con sus requerimientos.

### 3.1.1 Servidor

Es una computadora donde se incorpora el sistema PBX Asterisk, el mismo que permite el proceso de las llamadas de los usuarios a la clínica.

A continuación se muestra en la tabla 1 las características utilizadas para la implementación de este proyecto:

Procesador	Intel Pentium 4 de 3GHZ
Memoria RAM	2048 MB DDR 400 RAM
Disco Duro	250GB
Tarjeta de Red	Fastethernet 10/100 Mbps

Tabla 1 Características del Servidor

### 3.1.2 Teléfono IP



Figura 1: Teléfono IP

Basado en los estándares abiertos de la industria (compatible con Asterisk), identificador de llamada, llamada en espera, transferencia de llamada, servidor web para la administración y configuración, registro de llamadas, cancelación de eco (G.168), asegura una calidad de audio de alta fidelidad, 2 puertos Ethernet 10/100 Mbps, del cual se utiliza conectores RJ-45 para conectar el teléfono a la computadora y a la red

## 3.2 Software

### 3.2.1 Servidor

En la tabla 2 se detalla los sistemas y las aplicaciones instalados en el servidor para este proyecto:

Sistema Operativo	Linux
Distribución	Centos 5.6
Arquitectura	X86

Software IP PBX	Asterisk Versión 1.4
Base de Datos	MySQL
Software Adicional	PHPMYADMIN 2.11

Tabla 2 Características del Servidor – Software

### 3.3 Base de Datos

Este sistema contiene una base de datos implementada en MySQL, la misma que se compone de 6 tablas:

- Cita
- Paciente
- Registro
- Usuarios
- Especialidad
- Horaturn

En el cual el paciente tendrá como objetivo solicitar su cita médica en la especialidad que escoja y con su respectivo turno automáticamente asignado mediante la fecha seleccionada previamente, cuyos datos se encuentran localizados en las tablas.

### 3.4 Configuración de Asterisk

#### 3.4.1 Configuración del archivo sip.conf

Este archivo de configuración se encuentra localizado en la ruta /etc/asterisk y sirve para registrar los usuarios SIP con las extensiones que usaran la PBX, el mismo se encuentra estructurado en dos bloques, la parte general donde se definen las extensiones, y el bloque final donde configuramos nuestras extensiones internas y externas.

```
[general]
rtcachefriends=yes
rtupdate=no
rtautoclear=yes
srvlookup=yes
disallow=all
allow=alaw
allow=ulaw
allow=gsm
language=es
calltransfer=no
callforwarding=no
qualify=yes
```

```
[2442268]
type=friend
secret=2442268
qualify=yes
nat=no
host=dynamic
canreinvite=no
context=internal
```

#### 3.4.2 Configuración del archivo extensions.conf

Está ubicado en la ruta /etc/asterisk, Este archivo comprende en el plan de marcado y es el archivo más importante de Asterisk, su función principal es invocar al código donde se encuentra almacenado el programa principal, aquí encontramos dos bloques de código el contexto general y el internal:

El general es donde establecemos los parámetros generales del plan de marcado.

El contexto internal establece el procedimiento a realizarse con alguna extensión en el interior de la PBX.

```
[general]
autofallthrough=no
clearglobalvars=no
```

```
[globals]
ANDRES=SIP/801
```

```
[internal]
exten=>2442268,1,set(CHANNEL(language)=es)
exten=>2442268,2,background(bienvenida0)
exten=>2442268,3,Wait(4)
exten=>1,1,Goto(agente,801,1)
exten=>2,1,Goto(cita,701,1)
exten=>2442268,4,hangup
```

```
[cita]
exten=>701,1,AGI(citasmedicas.php)
exten=>701,2,hangup
```

```
[agente]
exten=>801,1,Dial(${ANDRES},10,r)
exten=>801,2,playback(vm-nobodyavail)
exten=>801,3,hangup
exten=>801,102,playback(tt-allbusy)
exten=>801,103,hangup
```

## 4. Funcionamiento y Prueba

### 4.1. Inicio del servidor Mysql

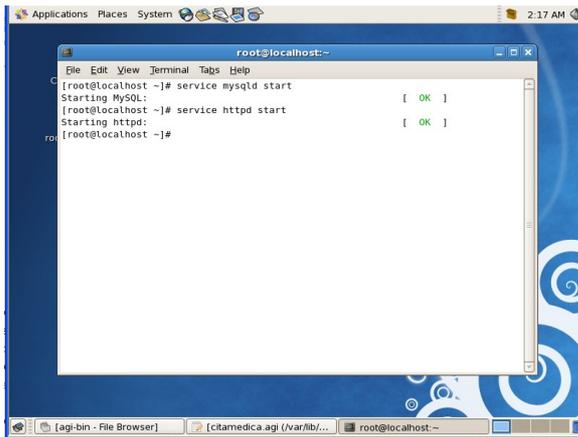


Figura 2: Iniciando Mysql

Para poder hacer uso de la base de datos donde se tiene toda la información de los pacientes y el personal de la institución médica se debe previamente ingresar el comando que permita iniciar el servicio MySQL, tal como se muestra en la Figura 2.

## 4.2 Inicio de Asterisk

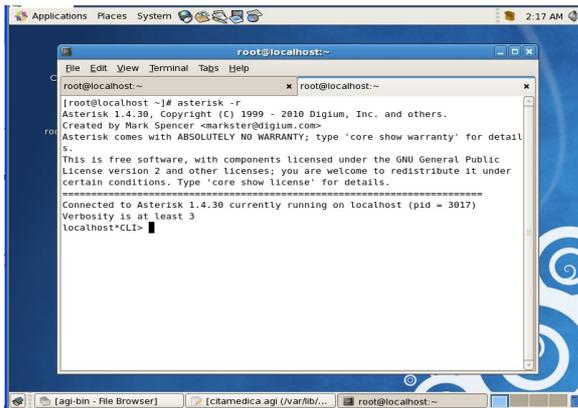


Figura 3: Iniciando Asterisk

Para poder iniciar el sistema principal de este proyecto se hace uso del comando Asterisk -r y deberá de presentarse como lo muestra la Figura 3.

## 4.3 Llamada al Sistema de Citas Medicas

El Paciente realizara la llamada al centro médico al número de teléfono 2442268, al marcar este número escuchara un mensaje de bienvenida seguido con opciones que deberá digitar para poder continuar con su petición, si el paciente digita 1, la llamada es transferida a un agente quien se encargará de ayudar al paciente a: consultar su cita, ingresar o actualizar información personal etc., si digita la opción 2 la llamada pasa inmediatamente a la solicitud de cita en el que debe ingresar su número cédula, si no está

registrada deberá volver a ingresar nuevamente caso contrario deberá volver a llamar y escoger la opción 1 para que un agente lo ayude, pero si está registrada se le presenta un menú donde escogerá la especialidad a la que desee asistir, continuando con el proceso se le pide digitar la fecha en un formato indicado previamente, permitiendo así asignarle una turno automáticamente escogido por el sistema todo esto depende de que si la fecha ingresa es correcta o esté disponible, finalmente se repetirá los datos para confirmar la hora y la especialidad, si está de acuerdo digitara 5 para ser almacenada a la base de datos del centro médico, caso contrario la llamada será colgada y la información descartada.

### 4.3.1. Ingreso de una Cedula Registrada

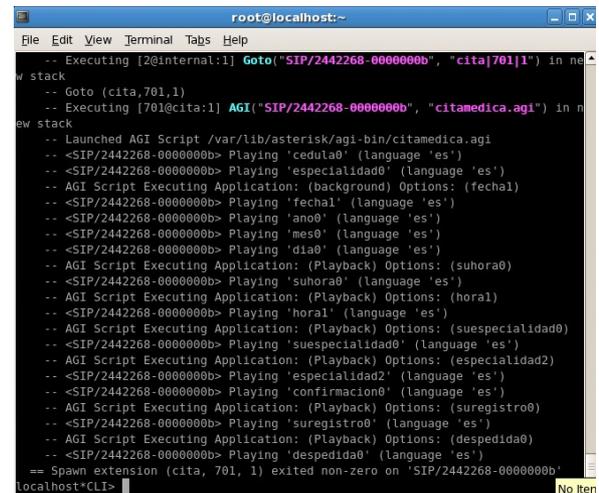


Figura 4: Ingreso de Cedula Registrada

Se realiza la búsqueda de la cédula registrada en la base de datos y reproduce la grabación para poder proceder con el siguiente paso como el de escoger especialidad, digitar fecha y confirmar, como se muestra en la Figura 4

### 4.3.2. Ingreso de una Cedula No Registrada

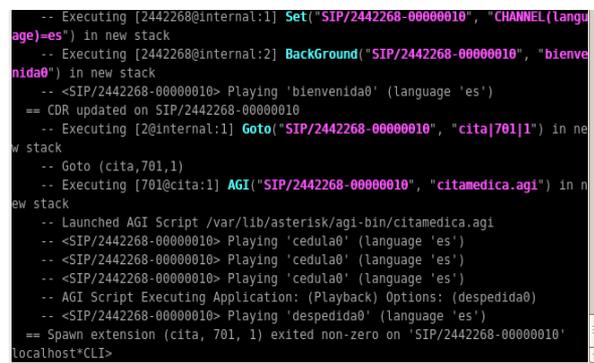


Figura 5: Ingreso de Cedula No Registrada

Si al no encontrar la cédula en la base de datos esta reproduce nuevamente la grabación indicándole

que ingrese su número de cédula o digite 9 para salir del sistema, tal como se muestra en la Figura 5.

### 4.3.3. Confirmación del Usuario

```
-- AGI Script Executing Application: (Playback) Options: (suhora0)
-- <SIP/2442268-0000000b> Playing 'suhora0' (language 'es')
-- AGI Script Executing Application: (Playback) Options: (horal)
-- <SIP/2442268-0000000b> Playing 'horal' (language 'es')
-- AGI Script Executing Application: (Playback) Options: (suespecialidad0)
-- <SIP/2442268-0000000b> Playing 'suespecialidad0' (language 'es')
-- AGI Script Executing Application: (Playback) Options: (especialidad2)
-- <SIP/2442268-0000000b> Playing 'especialidad2' (language 'es')
-- <SIP/2442268-0000000b> Playing 'confirmacion0' (language 'es')
-- AGI Script Executing Application: (Playback) Options: (suregistro0)
-- <SIP/2442268-0000000b> Playing 'suregistro0' (language 'es')
-- AGI Script Executing Application: (Playback) Options: (despedida0)
-- <SIP/2442268-0000000b> Playing 'despedida0' (language 'es')
== Spawn extension (cita, 701, 1) exited non-zero on 'SIP/2442268-0000000b'
```

Figura 6: Confirmación del Usuario

Se repite los datos ingresado en el transcurso de la llamada a lo que respecta sólo a la especialidad escogida y la hora asignada, si el paciente está de acuerdo con el resultado, se le confirmara al sistema al digitar 5, luego se reproduce la grabación indicando que su registro fue exitoso y se procede a colgar la llamada. Como se muestra en la Figura 6.

El paciente deberá obligatoriamente digitar el 5 para que su información pueda ser almacenada en la base de datos de la clínica caso contrario pasara a ser descartada automáticamente.

### 4.3.4. Fecha mal ingresada o Cita no Disponible

```
-- Executing [2442268@internal:1] Set("SIP/2442268-00000012", "CHANNEL(Language)-es") in new stack
-- Executing [2442268@internal:2] Background("SIP/2442268-00000012", "bienvenida0") in new stack
-- <SIP/2442268-00000012> Playing 'bienvenida0' (language 'es')
== CDR updated on SIP/2442268-00000012
-- Executing [2@internal:1] Goto("SIP/2442268-00000012", "cita[701]1") in new stack
-- Goto (cita,701,1)
-- Executing [701@cita:1] AGI("SIP/2442268-00000012", "citamedica.agi") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/citamedica.agi
-- <SIP/2442268-00000012> Playing 'cedula0' (language 'es')
-- <SIP/2442268-00000012> Playing 'especialidad0' (language 'es')
-- AGI Script Executing Application: (background) Options: (fechal)
-- <SIP/2442268-00000012> Playing 'fechal' (language 'es')
-- <SIP/2442268-00000012> Playing 'ano0' (language 'es')
-- <SIP/2442268-00000012> Playing 'mes0' (language 'es')
-- <SIP/2442268-00000012> Playing 'dia0' (language 'es')
-- <SIP/2442268-00000012> Playing 'otrafecha0' (language 'es')
-- AGI Script Executing Application: (Playback) Options: (despedida0)
-- <SIP/2442268-00000012> Playing 'despedida0' (language 'es')
== Spawn extension (cita, 701, 1) exited non-zero on 'SIP/2442268-00000012'
```

Figura 7: Fecha no Valida

Si la fecha es anterior a la de hoy o no se encuentra disponible se le volverá a solicitar que ingrese una nueva fecha, pero ingresando nuevamente su cédula y especialidad, caso contrario digitará cero para poder salir del sistema. Como se muestra en la Figura 7

### 4.3.5. Fin de Llamada

```
-- AGI Script Executing Application: (Playback) Options: (despedida0)
-- <SIP/2442268-0000000b> Playing 'despedida0' (language 'es')
== Spawn extension (cita, 701, 1) exited non-zero on 'SIP/2442268-0000000b'
```

Figura 8: Confirmación del Usuario

La llamada Finaliza cuando el paciente digita la opción, salir del Sistema, o al confirmar sus datos ingresados en ambos caso se reproduce el audio de despedida Vease Figura 8.

## 5. Conclusiones

- Con el uso de la interfaz de desarrollo de Asterisk AGI, las librerías phpagi facilita en gran parte el desarrollo de aplicaciones complejas, permitiendo de esta manera trabajar en conjunto con el motor de base de datos tan utilizado como lo es MySQL.
- La implementación de un sistema telefónico con el fin de solicitar citas médicas de una manera sencilla y que pueda solventar a las necesidades del paciente, es fácil de realizar con las funciones para crear IVR's que nos ofrece Asterisk.
- La instalación de software libre provee diferentes herramientas y librerías, con un tiempo menor de proceso lo que permite una implementación de sistemas de gran utilidad, eficacia, eficiencia, adaptables, escalables, confiables y a un bajo costo.
- Para la administración de la base de datos que maneja el sistema de citas médicas se necesita desarrollar una aplicación que permita poder ejecutar todas las operaciones sobre ella.

## 6. Agradecimientos

Agradecemos a Dios por la fortaleza y las bendiciones que ha derramado en nuestra vida universitaria, para poder culminar con éxito el presente trabajo, como también a nuestros padres quienes fueron, son y serán un pilar importante en

nuestro camino, y finalmente a los profesores de quienes aprendimos a ser perseverante y responsable.

## 7. Referencias

- [1] Phpmyadmin.com, [http://www.phpmyadmin.net/home\\_page/index.php](http://www.phpmyadmin.net/home_page/index.php), Septiembre 2010.
- [2] asteriskguide.com, Integración de Asterisk usando AGI y AMI, [http://www.asteriskguide.com/mediawiki/index.php/Integraci%C3%B3n\\_de\\_Asterisk\\_usando\\_AGI\\_y\\_AMI](http://www.asteriskguide.com/mediawiki/index.php/Integraci%C3%B3n_de_Asterisk_usando_AGI_y_AMI), Septiembre 2010.
- [3] asteriskguide.com, Introducción al Asterisk, [http://www.asteriskguide.com/mediawiki/index.php/Introducci%C3%ADn\\_al\\_Asterisk](http://www.asteriskguide.com/mediawiki/index.php/Introducci%C3%ADn_al_Asterisk), octubre 2010.
- [4] VOIP-Info.org, Asterisk AGI, <http://www.voip-info.org/wiki/view/Asterisk+AGI>, Noviembre 2010
- [5] Packtpub.com, Asterisk Gateway Interface Scripting with PHP <http://www.packtpub.com/article/asterisk-gateway-interface-scripting-with-php>, Noviembre 2010
- [6] Van Meggelen J., Smith J., Madsen L., Asterisk The Future of Telephony, Editorial O' Relly Media, 2005, publicado en <http://www.asterix.org>, Julio 2010
- [7] VOIP-Info.org, Asterisk simple php lookup MySQL database, <http://www.voip-info.org/wiki/view/Asterisk+simple+php+lookup+mysql+database+to+set+callerid+name>, Enero 2011
- [8] sourceforge.net, PHPAGI the connection de php and asterisk, <http://phpagi.sourceforge.net/>, Enero 2011
- [9] blog.oaxrom.com, Como implementar un agi con php, [http://blog.oaxrom.com/index.php/blog/show/Como\\_implementar\\_un\\_AGI\\_con\\_PHP\\_en\\_un\\_conmutador\\_de\\_VoIP\\_sobre\\_Asterisk.html](http://blog.oaxrom.com/index.php/blog/show/Como_implementar_un_AGI_con_PHP_en_un_conmutador_de_VoIP_sobre_Asterisk.html), Enero 2011
- [10] Wikipedia, Asterisk, <http://es.wikipedia.org/wiki/Asterisk>, Noviembre 2010.