



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
Facultad de Ingeniería en Electricidad y Computación

“Análisis e Implementación de un sistema telefónico de respuesta automática para consultar las calificaciones en el sistema académico de la ESPOL”

INFORME DE MATERIA DE GRADUACIÓN

Previa a la obtención del Título de:

INGENIERO EN TELEMÁTICA

Presentada por:

DENNISSE ANDREA CEDEÑO SUÁREZ

ROCÍO ELIZABETH MERA SUÁREZ

Guayaquil – Ecuador
2012

AGRADECIMIENTO

A Dios, que nos ha conservado con vida y salud.

*A nuestros padres, quienes han sido y son un
pilar fundamental en nuestras vidas.*

*A nuestros profesores, por todos los conocimientos
y consejos transmitidos*

DEDICATORIA

*A Dios
A nuestros padres
A nuestros familiares
A nuestros amigos*

TRIBUNAL DE SUSTENTACIÓN

Ing. Gabriel Astudillo

PROFESOR DEL SEMINARIO

Ing. Patricia Chávez

PROFESOR DELEGADO POR EL DECANO

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de este Proyecto de Graduación, nos corresponde exclusivamente; y el patrimonio intelectual de la misma, a la Escuela Superior Politécnica del Litoral.”

(Reglamento de exámenes y títulos profesionales de la ESPOL)

Dennisse Andrea Cedeño Suárez

Rocío Elizabeth Mera Suárez

RESUMEN

El proyecto consistió en la implementación de un Sistema Telefónico de Respuesta Automática para estudiantes de la Escuela Superior Politécnica del Litoral (ESPOL) para consultar su información académica, como alternativa al sitio web Académico en Línea de la ESPOL.

Cuando una llamada es generada, el sistema reproduce una grabación solicitando la autenticación del usuario a través de una clave única, que en este caso es el número de matrícula del estudiante. Si el usuario se encuentra registrado en la universidad se le reproduce un menú con las opciones que del sistema.

La implementación de este sistema fue de bajo costo de hardware y alto grado de accesibilidad. Tan sólo se necesitó acceso a una línea telefónica para emplear el sistema, el cual fue fácil de usar y de administrar.

INDICE GENERAL

| | |
|------------------------------------|-------|
| AGRADECIMIENTO..... | i |
| DEDICATORIA..... | ii |
| TRIBUNAL DE SUSTENTACIÓN..... | iii |
| DECLARACIÓN EXPRESA..... | iv |
| RESUMEN | v |
| INDICE GENERAL | vi |
| ABREVIATURAS..... | x |
| INDICE DE FIGURAS | xii |
| INDICE DE TABLAS | xv |
| INTRODUCCIÓN | xviii |
| CAPÍTULO 1 | 1 |
| 1. PLANTEAMIENTO..... | 1 |
| 1.1 ANTECEDENTES | 1 |
| 1.2 JUSTIFICACIÓN | 2 |
| 1.3 DESCRIPCIÓN DEL PROYECTO | 3 |
| 1.3.1 OBJETIVO GENERAL | 3 |
| 1.3.2 OBJETIVOS ESPECÍFICOS..... | 3 |

| | | |
|------------------|--|----|
| 1.4 | METODOLOGÍA..... | 7 |
| 1.5 | LIMITACIONES..... | 7 |
| CAPÍTULO 2 | | 8 |
| 2. | MARCO TEÓRICO | 8 |
| 2.1 | ASTERISK | 8 |
| 2.1.1 | DEFINICIÓN DE ASTERISK | 8 |
| 2.1.2 | FUNCIONAMIENTO DE ASTERISK | 9 |
| 2.2 | INTERFAZ DE PUERTA DE ENLACE DE ASTERISK | 11 |
| 2.3 | PHP- AGI | 13 |
| 2.4 | ASTERISK Y LAS INTERCONEXIONES..... | 13 |
| 2.5 | FESTIVAL | 16 |
| CAPÍTULO 3 | | 17 |
| 3. | ANÁLISIS Y DISEÑO..... | 17 |
| 3.1 | ANÁLISIS | 17 |
| 3.2 | DISEÑO DE LA SOLUCIÓN | 19 |
| 3.2.1 | DISEÑO DE LA BASE DE DATOS..... | 19 |
| 3.2.2 | DISEÑO DEL PLAN DE MARCADO | 21 |
| CAPÍTULO 4 | | 23 |
| 4. | IMPLEMENTACIÓN PILOTO | 23 |
| 4.1 | SELECCIÓN DE HARDWARE..... | 23 |
| 4.2 | INSTALACIONES..... | 26 |
| 4.2.1 | INSTALACIÓN DE PHP | 26 |
| 4.2.2 | INSTALACIÓN DE MYSQL | 27 |
| 4.2.3 | INSTALACIÓN DE FESTIVAL..... | 28 |
| 4.2.4 | INSTALACIÓN DE ASTERISK | 30 |
| 4.3 | CREACIÓN DE LA BASE DE DATOS | 32 |
| 4.4 | CONFIGURACIONES DE ARCHIVOS DE ASTERISK..... | 33 |

| | | |
|--|--|----|
| 4.4.1 | PLAN DE MARCADO | 33 |
| 4.4.2 | SCRIPTS AGI | 38 |
| 4.4.3 | DESCRIPCIÓN DE LOS ARCHIVOS DE AUDIO | 43 |
| CAPÍTULO 5 | | 47 |
| 5. | PRUEBAS Y ANÁLISIS DE LOS RESULTADOS | 47 |
| 5.1 | PRUEBAS DE FUNCIONALIDAD | 48 |
| 5.2 | PRUEBAS DE RENDIMIENTO | 83 |
| 5.2.1 | ARQUITECTURA DE LA PRUEBA DE RENDIMIENTO | 84 |
| 5.2.2 | INSTALACIÓN SIPP | 85 |
| 5.2.3 | CONFIGURACIÓN DEL SERVIDOR AUDIO ESPOL IP | 85 |
| 5.2.4 | REALIZACIÓN DE PRUEBA DE RENDIMIENTO | 87 |
| 5.2.5 | INICIO DE LA PRUEBA DE CLIENTE SIPP | 87 |
| 5.2.6 | REALIZACIÓN DE PRUEBA DE RENDIMIENTO CPU Y MEMORIA | 88 |
| 5.2.7 | RESULTADOS DE LA PRUEBA DE RENDIMIENTO | 89 |
| CONCLUSIONES Y RECOMENDACIONES | | |
| CONCLUSIONES | | |
| RECOMENDACIONES | | |
| ANEXOS | | |
| ANEXO A | | |
| MANUAL DE USUARIO | | |
| ANEXO B | | |
| MODELO LÓGICO DE LA BASE DE DATOS | | |
| SCRIPT CREACIÓN DE TABLAS PARA LA BASE DE DATOS ASTERISK | | |
| ANEXO C | | |
| PLAN DE MARCADO | | |
| ANEXO D | | |

SCRIPT PARA INGRESO AL SISTEMA

SCRIPT PARA CONSULTAR MATERIA

SCRIPT PARA CONSULTAR HORARIO DE EXAMENES

SCRIPT PARA CONSULTAR HORARIO DE CLASES

ANEXO E

SCRIPT USADO EN LA PRUEBA DE RENDIMIENTO

ANEXO F

PABX

IP PBX PROPIETARIAS

BIBLIOGRAFÍA

ABREVIATURAS

ACD.- Distribuidor Automático de Llamadas.

AGI.- Interfaz de Puerta de Enlace de Asterisk.

BRI.- Interfaz de Acceso Básico.

CO.- Oficina Central.

DISA.- Acceso Directo al Sistema Interno.

DTM.- Multifrecuencia de Doble Tono.

FXO.- Oficina de Intercambio Exterior.

IAX2.- Protocolo de Intercambio Entre-Asterisk Versión 2.

IP.- Protocolo de Internet.

IVR.- Respuesta de Voz Interactiva.

PBX.- Central Privada de Intercambio.

PCM.- Modulación de Códigos de Pulsos.

PHP.- Pre-Procesador de Hipertexto.

PRI.- Interfaz de Acceso Primario.

PSTN.- Red Telefónica Conmutada.

QoS.- Calidad de Servicio.

RDSI o ISDN.- Red Digital de Servicios Integrados.

SIP.- Protocolo de Iniciación de Sesión.

TDM.- Multiplexación por División de Tiempo.

UCD.- Distribución Uniforme de Llamadas.

VoIP.- Voz sobre Protocolo de Internet.

XDP.- Función de Doblar Puertos.

INDICE DE FIGURAS

| | |
|--|----|
| FIGURA 1.1.- ESQUEMÁTICO PROCESO DE INGRESO DE MATRÍCULA Y SELECCIÓN DEL MENÚ EN EL SISTEMA | 6 |
| FIGURA 2.1.- ARQUITECTURA DE ASTERISK..... | 10 |
| FIGURA 2.2.- DESCRIPCIÓN DE AGI..... | 12 |
| FIGURA 2.3.- DESCRIPCIÓN DE UNA PETICIÓN DE CONEXIÓN | 14 |
| FIGURA 2.4.- EJEMPLO DE INTERACCIÓN FXS / FXO TÍPICA EN UN HOGAR | 15 |
| FIGURA 2.5.- EJEMPLO DE INTERACCIÓN FXS / FXO CON CENTRALITA.. | 15 |
| FIGURA 4.1 DISEÑO DE LA INFRAESTRUCTURA DEL SISTEMA AUDIO ESPOL IP INTERCONECTADO CON EL SERVIDOR VOIP DE LA ESPOL..... | 24 |
| FIGURA 4.2 DISEÑO DE LA INFRAESTRUCTURA DEL SISTEMA AUDIO ESPOL IP CONECTADO A LA PSTN USANDO TARJETA DE TELEFONÍA ANALÓGICA DIGIUM..... | 24 |
| FIGURA 5.1.- RESPUESTA DEL SISTEMA AL ESTABLECERSE UNA LLAMADA..... | 49 |
| FIGURA 5.2.- RESPUESTA DEL SISTEMA AL ESTABLECERSE UN INGRESO FALLIDO AL SISTEMA | 51 |
| FIGURA 5.3.- RESPUESTA DEL SISTEMA AL ESTABLECERSE UN INGRESO EXITOSO AL SISTEMA | 53 |
| FIGURA 5.4.- RESPUESTA DEL SISTEMA AL INGRESAR LOS DATOS DE CONSULTA PARA CALIFICACIONES | 55 |

| | |
|---|----|
| FIGURA 5.5.- RESPUESTA DEL SISTEMA LUEGO DE LA CONFIRMACIÓN DE AÑO Y TÉRMINO..... | 56 |
| FIGURA 5.6.- RESPUESTA DEL SISTEMA: SALIDA DE CONSULTA DE CALIFICACIONES POR AÑO Y TÉRMINO | 57 |
| FIGURA 5.7.- RESPUESTA DEL SISTEMA: OPCIONES DE CONSULTA DE FECHA DE REGISTRO..... | 58 |
| FIGURA 5.8.- RESPUESTA DEL SISTEMA: CONSULTA DE FECHA DE REGISTRO DADO UNA MATRÍCULA | 59 |
| FIGURA 5.9.- RESPUESTA DEL SISTEMA: SALIDA DE CONSULTA DE FECHA DE REGISTRO POR NÚMERO DE MATRÍCULA | 60 |
| FIGURA 5.10.- RESPUESTA DEL SISTEMA: CONSULTA TODAS LAS FECHAS DE REGISTRO DEL SIGUIENTE TÉRMINO..... | 61 |
| FIGURA 5.11.- RESPUESTA DEL SISTEMA: SALIDA DE CONSULTA TODAS LAS FECHAS DE REGISTRO DEL SIGUIENTE TÉRMINO..... | 62 |
| FIGURA 5.12.- RESPUESTA DEL SISTEMA: INGRESO A CONSULTA DE FECHAS DE EXÁMENES | 63 |
| FIGURA 5.13.- RESPUESTA DEL SISTEMA: CONSULTA DE FECHAS DE EXÁMENES PARCIALES | 64 |
| FIGURA 5.14.- RESPUESTA DEL SISTEMA: SALIDA DE CONSULTA DE FECHAS DE EXÁMENES PARCIALES..... | 65 |
| FIGURA 5.15.- RESPUESTA DEL SISTEMA: CONSULTA DE FECHAS DE EXÁMENES FINALES..... | 66 |
| FIGURA 5.16.- RESPUESTA DEL SISTEMA: SALIDA DE CONSULTA DE FECHAS DE EXÁMENES FINALES | 67 |
| FIGURA 5.17.- RESPUESTA DEL SISTEMA: CONSULTA DE FECHAS DE EXÁMENES MEJORAMIENTO..... | 68 |
| FIGURA 5.18.- RESPUESTA DEL SISTEMA: SALIDA DE CONSULTA DE FECHAS DE EXÁMENES MEJORAMIENTO | 69 |
| FIGURA 5.19.- RESPUESTA DEL SISTEMA: INGRESO A CONSULTA DE HORARIO POR MATERIA..... | 70 |

| | |
|--|----|
| FIGURA 5.20.- RESPUESTA DEL SISTEMA: CONSULTA DE FECHAS DE HORARIO POR MATERIA DEL DÍA LUNES | 72 |
| FIGURA 5.21.- RESPUESTA DEL SISTEMA: SALIDA DE CONSULTA DE HORARIO POR MATERIA DEL DÍA LUNES | 73 |
| FIGURA 5.22.- RESPUESTA DEL SISTEMA: CONSULTA DE FECHAS DE HORARIO POR MATERIA DEL DÍA MARTES | 74 |
| FIGURA 5.23.- RESPUESTA DEL SISTEMA: SALIDA DE CONSULTA DE HORARIO POR MATERIA DEL DÍA MARTES | 75 |
| FIGURA 5.24.- RESPUESTA DEL SISTEMA: CONSULTA DE FECHAS DE HORARIO POR MATERIA DEL DÍA MIÉRCOLES | 76 |
| FIGURA 5.25.- RESPUESTA DEL SISTEMA: SALIDA DE CONSULTA DE HORARIO POR MATERIA DEL DÍA MIÉRCOLES | 77 |
| FIGURA 5.26.- RESPUESTA DEL SISTEMA: CONSULTA DE FECHAS DE HORARIO POR MATERIA DEL DÍA JUEVES | 78 |
| FIGURA 5.27.- RESPUESTA DEL SISTEMA: SALIDA DE CONSULTA DE HORARIO POR MATERIA DEL DÍA JUEVES | 79 |
| FIGURA 5.28.- RESPUESTA DEL SISTEMA: CONSULTA DE FECHAS DE HORARIO POR MATERIA DEL DÍA VIERNES | 80 |
| FIGURA 5.29.- RESPUESTA DEL SISTEMA: SALIDA DE CONSULTA DE HORARIO POR MATERIA DEL DÍA VIERNES | 81 |
| FIGURA 5.30.- RESPUESTA DEL SISTEMA: CONSULTA DE FECHAS DE HORARIO POR MATERIA DEL DÍA SÁBADO | 82 |
| FIGURA 5.31.- RESPUESTA DEL SISTEMA: SALIDA DE CONSULTA DE HORARIO POR MATERIA DEL DÍA SÁBADO | 83 |
| FIGURA 5.32.- DISEÑO DE PRUEBA RENDIMIENTO | 84 |
| FIGURA 5.33.- ESTADÍSTICAS DE LAS PRUEBAS REALIZADAS CON SIPP | 90 |
| FIGURA 5.34.- GRÁFICA DE CONSUMO DE CPU DURANTE LA PRUEBA DE RENDIMIENTO | 91 |
| FIGURA 5.35.- GRÁFICA DE CONSUMO DE MEMORIA DURANTE LA PRUEBA DE RENDIMIENTO..... | 92 |

INDICE DE TABLAS

| | |
|---|----|
| TABLA I ESPECIFICACIONES DE LA TARJETA DE TELEFONÍA ANALÓGICA 1AEX2406EF..... | 25 |
| TABLA II.- CARACTERÍSTICAS DE HARDWARE DEL SERVIDOR..... | 26 |
| TABLA III TABLA DE CONSULTAS QUE SE PUEDE REALIZAR CON SU RESPECTIVO CONTEXTO | 38 |
| TABLA IV.- LISTA DE SCRIPTS USADOS EN CADA CONSULTA..... | 39 |
| TABLA V.- LISTA DE GRABACIONES CREADAS CON FESTIVAL | 43 |
| TABLA VI.- ESPECIFICACIONES DEL SERVIDOR ASTERISK | 47 |
| TABLA VII.- PRUEBA DE FUNCIONALIDAD: INGRESO AL SISTEMA | 49 |
| TABLA VIII.- PRUEBA DE FUNCIONALIDAD: INGRESO FALLIDO AL SISTEMA | 50 |
| TABLA IX.- PRUEBA DE FUNCIONALIDAD: INGRESO EXITOSO AL SISTEMA | 51 |
| TABLA X.- CONSULTA DE CALIFICACIONES: INGRESO DE AÑO Y TÉRMINO | 54 |
| TABLA XI.- CONSULTA DE CALIFICACIONES: CONFIRMACIÓN DE AÑO Y TÉRMINO..... | 55 |
| TABLA XII.- CONSULTA DE CALIFICACIONES: SALIDA..... | 56 |
| TABLA XIII.- CONSULTA DE FECHA DE REGISTRO: MENÚ PRINCIPAL | 57 |

| | |
|--|----|
| TABLA XIV.- CONSULTA DE FECHA DE REGISTRO DADO UNA MATRÍCULA | 58 |
| TABLA XV.- CONSULTA DE FECHA DE REGISTRO: SALIDA | 59 |
| TABLA XVI.- CONSULTA DE FECHAS DE REGISTRO: MENÚ PRINCIPAL ... | 60 |
| TABLA XVII.- CONSULTA DE FECHAS DE REGISTRO: SALIDA..... | 61 |
| TABLA XVIII.- CONSULTA FECHAS DE EXÁMENES: MENÚ PRINCIPAL | 63 |
| TABLA XIX.- CONSULTAR DE FECHAS DE EXÁMENES PARCIALES..... | 64 |
| TABLA XX.- CONSULTA DE FECHAS DE EXÁMENES PARCIALES: SALIDA | 64 |
| TABLA XXI.- CONSULTAR DE FECHAS DE EXÁMENES FINALES..... | 65 |
| TABLA XXII.- CONSULTA DE FECHAS DE EXÁMENES FINALES: SALIDA... | 66 |
| TABLA XXIII.- CONSULTAR DE FECHAS DE EXÁMENES MEJORAMIENTO | 67 |
| TABLA XXIV.- CONSULTA DE FECHAS DE EXÁMENES MEJORAMIENTO: SALIDA..... | 68 |
| TABLA XXV.- CONSULTA DE HORARIO POR MATERIA: MENÚ PRINCIPAL | 70 |
| TABLA XXVI.- CONSULTA DE HORARIO POR MATERIA DEL DÍA LUNES ... | 71 |
| TABLA XXVII.- CONSULTA DE HORARIO POR MATERIA DEL DÍA LUNES: SALIDA..... | 72 |
| TABLA XXVIII.- CONSULTA DE HORARIO POR MATERIA DEL DÍA MARTES | 73 |
| TABLA XXIX.- CONSULTA DE HORARIO POR MATERIA DEL DÍA MARTES: SALIDA..... | 74 |
| TABLA XXX.- CONSULTA DE HORARIO POR MATERIA DEL DÍA MIÉRCOLES | 75 |
| TABLA XXXI.- CONSULTA DE HORARIO POR MATERIA DEL DÍA MIÉRCOLES: SALIDA..... | 76 |
| TABLA XXXII.- CONSULTA DE HORARIO POR MATERIA DEL DÍA JUEVES | 77 |
| TABLA XXXIII.- CONSULTA DE HORARIO POR MATERIA DEL DÍA JUEVES: SALIDA..... | 78 |

| | |
|---|----|
| TABLA XXXIV.- CONSULTA DE HORARIO POR MATERIA DEL DÍA VIERNES | 79 |
| TABLA XXXV.- CONSULTA DE HORARIO POR MATERIA DEL DÍA VIERNES: SALIDA..... | 80 |
| TABLA XXXVI.- CONSULTA DE HORARIO POR MATERIA DEL DÍA SÁBADO | 81 |
| TABLA XXXVII.- CONSULTA DE HORARIO POR MATERIA DEL DÍA SÁBADO: SALIDA..... | 82 |
| TABLA XXXVIII.- ESPECIFICACIONES DEL SERVIDOR SIPP..... | 85 |
| TABLA XXXIX.- PARÁMETROS CONFIGURADOS EN SIPP AL REALIZAR LAS PRUEBAS DE RENDIMIENTOS..... | 89 |

INTRODUCCIÓN

En la comunidad politécnica se realiza de manera cotidiana las consultas de información académica, el medio más común que utiliza el alumnado para satisfacer esta necesidad es a través del sitio web “Académico en Línea”. Debido al creciente número de estudiantes que realizan consultas de sus datos simultáneamente, se presenta saturación en el sitio web; adicionalmente, este sistema presenta ciertas limitaciones para estudiantes que no tienen acceso a internet, puesto que es el único medio no presencial para obtener datos académicos del alumnado

El Académico en Línea contiene información específica de los estudiantes de la ESPOL, entre los datos que abarca están el historial de calificaciones, las calificaciones de las materias tomadas o registradas en el semestre, las deudas, las fechas de registro y exámenes. El resto de información acerca de la institución se encuentra en diversos sitios web dedicados, debido a la gran cantidad de datos que se genera a diario. Una alternativa es implementar un sistema de Voz sobre IP (VoIP) que permita a los usuarios acceder a la información de manera centralizada en la comodidad de su casa sin la necesidad de emplear internet.

CAPÍTULO 1

PLANTEAMIENTO

1.1 ANTECEDENTES

La telefonía tradicional es una red basada en la conmutación por circuitos, la cual presenta limitaciones. La implementación de una Central Privada de Intercambio (PBX) de forma tradicional presenta un sistema costoso y complejo que necesita gran variedad de equipos para su implementación.

En la actualidad la mayoría de las empresas cuentan con acceso a internet de banda ancha y buscan tener redes convergentes, la cual integra multimedia, voz y datos; aprovechando los recursos de la red. Una central telefónica compatible con el protocolo IP, que realiza conmutación por paquetes a través de la red de internet, es beneficioso

porque elimina la gran infraestructura de una central telefónica tradicional y permite la conexión con redes de telefonía digital e IP. Para la implementación de una central IP se utilizan softwares especializados para ello, como por ejemplo Asterisk.

El uso de centrales IP nos ayuda a abaratar costos de mantenimiento, implementación y llamadas entre sucursales; y, nos brinda beneficios adicionales como: correo de voz, música de espera, distribuidor de llamadas (ACD), faxes virtuales, etc.

1.2 JUSTIFICACIÓN

Los sistemas de respuestas automáticas de Voz son una manera rápida y fácil de acceder a información, ahorrando dinero por no necesitar personas que contesten los teléfonos y dando un buen servicio a los clientes.

Asterisk es una buena alternativa para montar estos sistemas, debido a la gran cantidad de opciones de configuración que proporciona, su fácil administración y bajo costo de implementación. Si se requiere que la

central se comunique con la Red Telefónica Conmutada (PSTN) simplemente se necesita tarjetas de telefonía.

1.3 DESCRIPCIÓN DEL PROYECTO

La implementación del proyecto pretende alcanzar los siguientes objetivos:

1.3.1 OBJETIVO GENERAL

Implementar un sistema telefónico de respuesta automática que permita a los estudiantes de la ESPOL consultar su información académica: calificaciones, fechas de registro, horario de clases y exámenes.

1.3.2 OBJETIVOS ESPECÍFICOS

- Implementar una central telefónica desarrollada en Asterisk y otros softwares de código abierto.
- Brindar una alternativa a los estudiantes para realizar consultas académicas.
- Permitir 24 llamadas concurrentes sin disminuir la calidad del servicio.
- Permitir llamadas desde las líneas de telefonía convencional e IP al sistema.

El proyecto consiste en la implementación de un sistema telefónico de respuesta automática para consultas académicas usando un servidor de Voz sobre IP.

El nombre del sistema implementado es **Audio ESPOL IP**

El usuario llama a la central telefónica y luego de establecer la conexión escucha una grabación solicitando que digite el número matrícula seguido. Después el sistema reproduce el número de matrícula marcado pidiendo confirmar si el número ingresado es correcto; se presiona 1 para confirmar y 0 para volver a digitar la matrícula. Luego de confirmar la matrícula se realiza la consulta en la base de datos para verificar si existe un estudiante registrado con ese número de matrícula. Si el número de matrícula no es válido, se reproduce una grabación diciendo que el usuario no existe y se pide ingresar nuevamente un número de matrícula. Si el usuario existe se guarda el número de matrícula para futuras consultas y se reproduce el menú con las opciones que brinda el sistema.

Las opciones que brinda el sistema son cinco:

1. Consulta de las calificaciones de las materias tomadas en un término y año específico.

2. Consulta de fechas de registro para el próximo término.
3. Consulta de fecha de Exámenes en el presente término.
4. Consulta de horario de clases dado un día a la semana.
5. Salir del sistema.

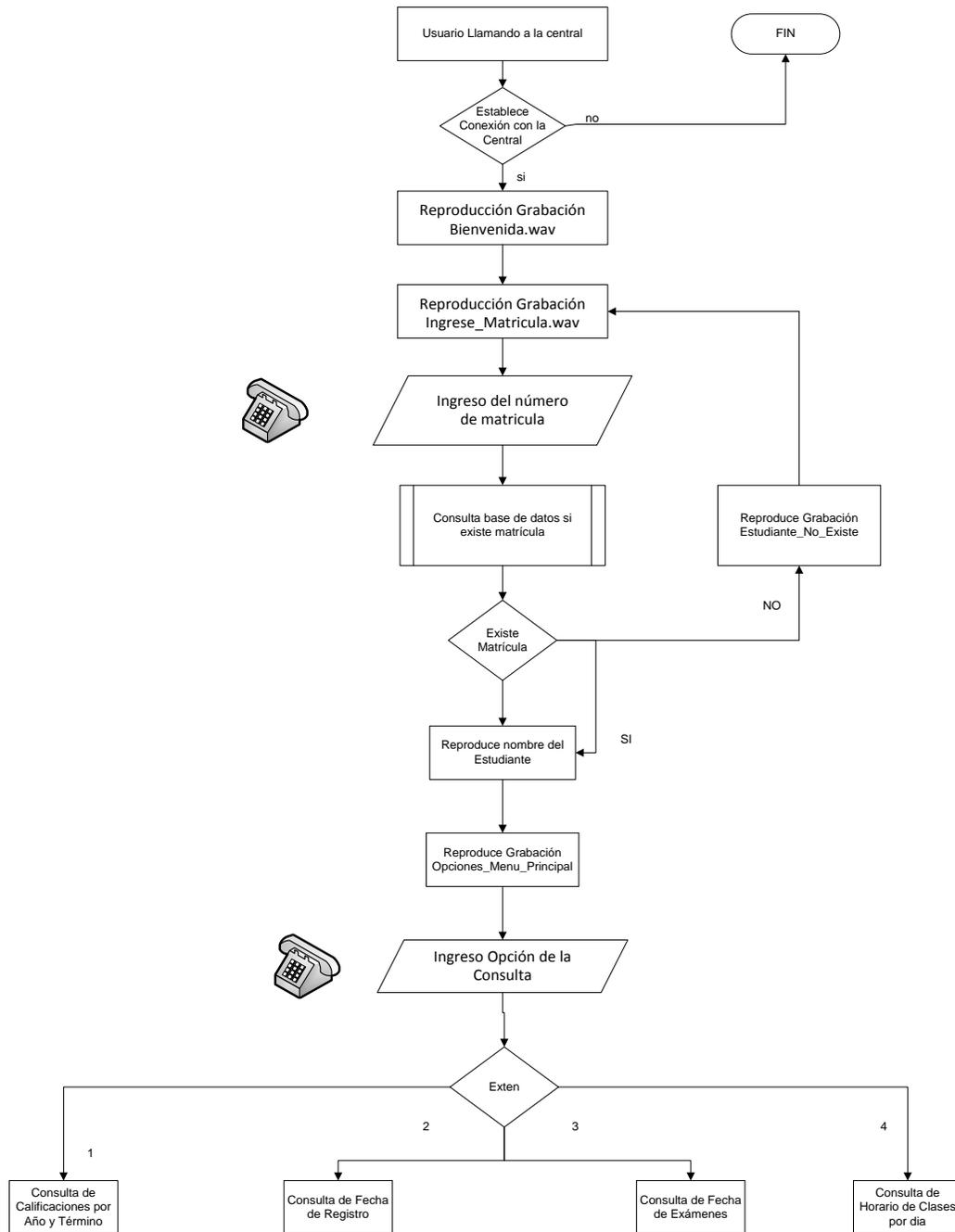


Figura 1.1- Esquemático Proceso de ingreso de matrícula y selección del menú en el sistema

1.4 METODOLOGÍA

Instalamos el software Asterisk versión 1.8.3 como servidor de telefonía IP en un equipo con sistema operativo Centos 5.5. El servidor tiene tarjetas de telefonía analógica FXO para establecer comunicación con la red PSTN.

También instalamos el motor de bases de datos MySQL para la creación de la base de datos del proyecto con sus respectivas tablas y establecer MySQL como servicio. Empleamos el lenguaje PHP para crear los script AGI los cuales realizan la conexión con la base de datos, los procesos de lógica avanzada y obtención de recursos externos. Utilizamos el sintetizador de voz Festival ya que es de libre distribución y compatible con Asterisk.

1.5 LIMITACIONES

En el presente proyecto no se realizara la implementación física del mismo solo una implementación piloto debido a las siguientes razones:

- Alto costos de las tarjetas analógicas con las características requeridas para la implementación que están alrededor de \$1.900.
- Falta de acceso a los servidores de la ESPOL por motivos de seguridad de la institución.

CAPÍTULO 2

MARCO TEÓRICO

2.1 ASTERISK

2.1.1 DEFINICIÓN DE ASTERISK

Asterisk es un software libre multiplataforma que opera sobre la aplicación VoIP, este brinda funcionalidades de una central telefónica PBX permitiendo beneficios como capturas, desvíos y transferencias de llamadas, buzón de voz, audio conferencias, grabación de llamadas, extensiones remotas, entre otras características.

Asterisk es compatible con la mayoría de estándares de telefonía, presenta soporte para la telefonía tradicional analógica mediante el uso de tarjetas electrónicas telefónicas FXS y FXO. Para la

integración con la Red Digital de Servicios Integrados (RDSI) utiliza Interfaz de Acceso Básico (BRI) o Interfaz de Acceso Primario (PRI). Para la telefonía IP utilizan protocolos que negocian la comunicación de VOIP como SIP, H323, IAX2, MGCP.

Asterisk es escalable, puede ser implementado para una pequeña empresa o para una multinacional; es menos costoso, debido a que se implementa en un ordenador o un servidor, independiente del fabricante, y tarjetas PCI; es programable, se puede añadir nuevas funciones dependiendo del requerimiento o necesidades que tenga el sistema.

2.1.2 FUNCIONAMIENTO DE ASTERISK

Asterisk cuenta con una arquitectura modular la cual está basada en aplicaciones, formatos de archivos, codecs (Codificadores y decodificadores) y canales. En la Figura 2.1 se muestran los codecs soportados por Asterisk.

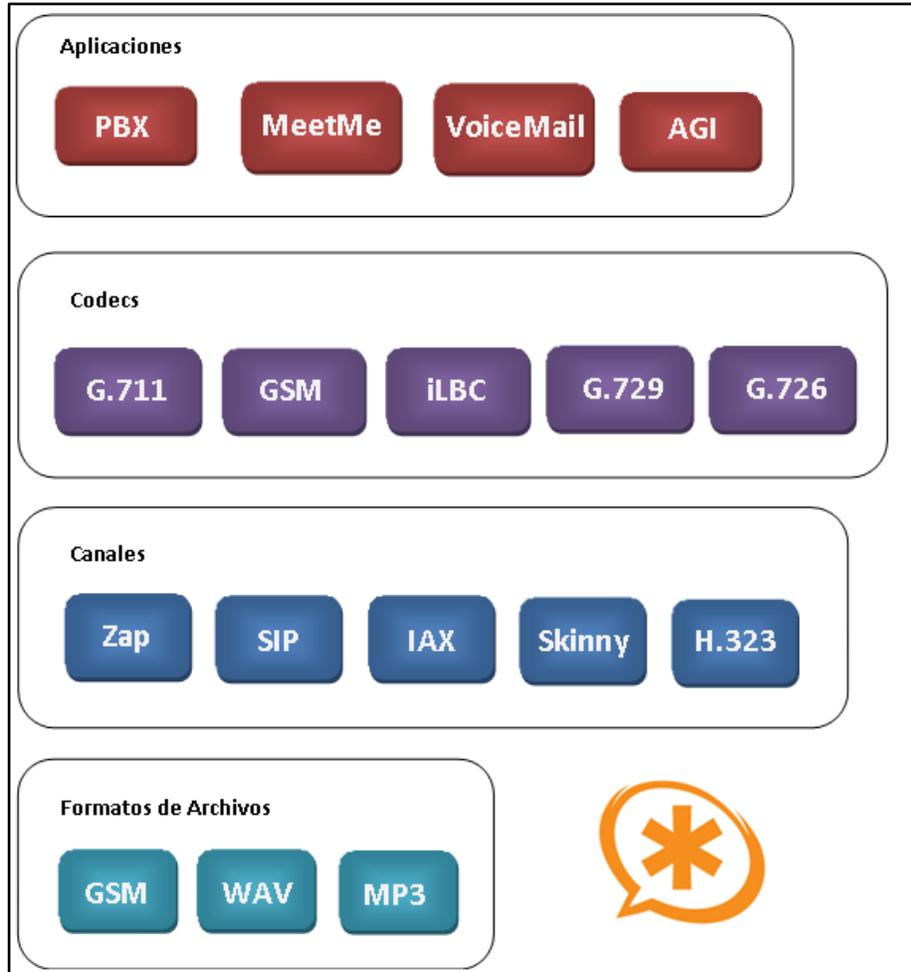


Figura 2.1.- Arquitectura de Asterisk

2.2 INTERFAZ DE PUERTA DE ENLACE DE ASTERISK

La interfaz de puerta de enlace de Asterisk (AGI), proporciona una interfaz estándar para que programas externos pueden controlar el plan de marcado, siendo un enlace entre las aplicaciones externas y el núcleo de Asterisk. Por lo general, los scripts AGI se utilizan para hacer operaciones de lógica avanzada, comunicarse con bases de datos relacionales (como PostgreSQL o MySQL), y acceder a recursos externos, esto nos permite hacer sistemas más complejos y de mayor utilidad.

Estos scripts pueden ser escritos en casi cualquier lenguaje de programación moderno, se escogió PHP.

Los scripts de AGI se comunican con Asterisk sobre los canales de comunicación conocidos como STDIN, STDOUT, y STDERR. Estos son canales por los cuales los sistemas basados en ambientes Unix envían y reciben información a los programas externos:

- Un script AGI lee **STDIN** para obtener información de Asterisk. [1]
- Un script AGI escribe datos en **STDOUT** para enviar información a Asterisk. [1]

- Un script AGI puede escribir en **STDERR** para enviar información de depuración a la consola de Asterisk. [1]

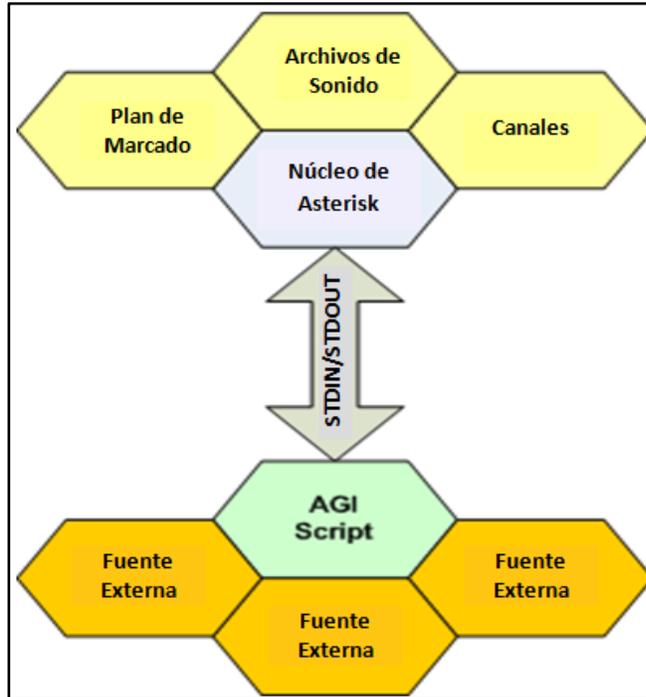


Figura 2.2.- Descripción de AGI

Fuente: Artículo “A Primer to AGI: Asterisk Gateway Interface”,
<http://www.packtpub.com/article/primer-to-agi-asterisk-gateway-interface>

Los Script AGI deben ser ejecutables, generalmente se encuentran en el directorio AGI (/ var / lib / asterisk / agi-bin) y para utilizarse dentro del plan de marcado, se debe llamar a la aplicación AGI (), con el nombre del script AGI como argumento, así:

```
exten => 123,1, Answer ()
```

```
exten => 123,2, AGI (AGI-test.agi)
```

2.3 PHP- AGI

En PHP-AGI existe una clase con varias funciones previamente definidas que hacen más fácil la tarea de elaborar códigos PHP para Asterisk, que permiten construir aplicaciones más complejas de manera sencilla. Se la debe descargar, descomprimir y copiar dentro del directorio `/var/lib/asterisk/agibin/`; luego, para poder utilizar las funciones, se debe incluir la clase en el código mediante la sentencia: `require "/var/lib/asterisk/agi-bin/phpagi-2.20/phpagi.php"`.

2.4 ASTERISK Y LAS INTERCONEXIONES

Asterisk además de prestar servicios de VoIP es capaz de interconectarse con otras redes de telefonía como son la PSTN, ISDN, redes de telefonía celular, lo cual requiere de hardware adicional. Digium es uno de los principales fabricantes de este tipo de tarjetas con placas T1 y E1 de varios puertos para la interconexión a líneas PRI y bancos de canal. Además, están disponibles tarjetas analógicas FXO y/o FXS de varios

puertos para telefonía analógica que son muy populares para instalaciones pequeñas.

La interfaz de dispositivos Digium/Asterisk (DAHDI), es la interfaz de Asterisk encargada de controlar el hardware de DIGIUM y otras tarjetas de telefonía tradicional.

Asterisk utiliza los canales SIP e IAX para las comunicaciones de VoIP, y los canales ZAP y DAHDI para comunicaciones con las tarjetas PCI del teléfono. A diferencia de los canales SIP e IAX, el canal ZAP y DAHDI, no hace el proceso de registro, se comunica directamente con los drivers de las tarjetas.

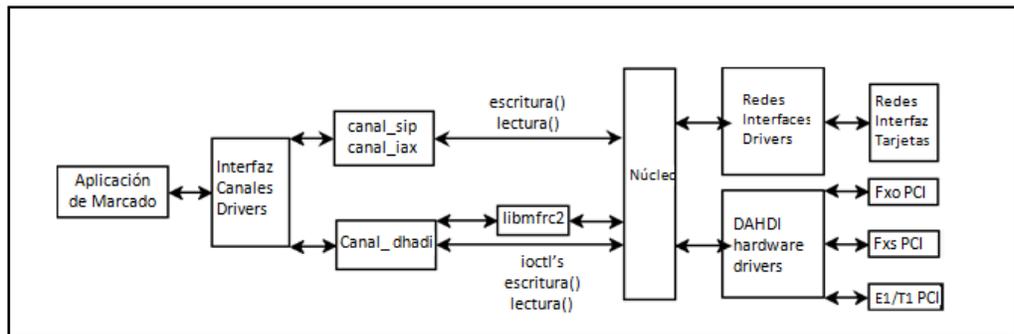


Figura 2.3.- Descripción de una petición de conexión

Fuente: Moisés Silva, Instalación y Configuración de Asterisk,
http://files.pablasso.com/docs/manual_asterisk.pdf

Existen los puertos Oficina de Intercambio Exterior (FXO) y Subscriptor de Intercambio Exterior (FXS) que son usados en la telefonía analógica. FXO es el módulo encargado de aceptar tono, enviar dígitos y recibir llamadas. Usualmente destinados a ser nodos finales. Los teléfonos analógicos usan señalización FXO. FXS es el módulo capaz de generar tono de marcado proporcionando un nivel de voltaje adecuado. Deben estar conectados en su otro extremo hacia un puerto FXO. [2]

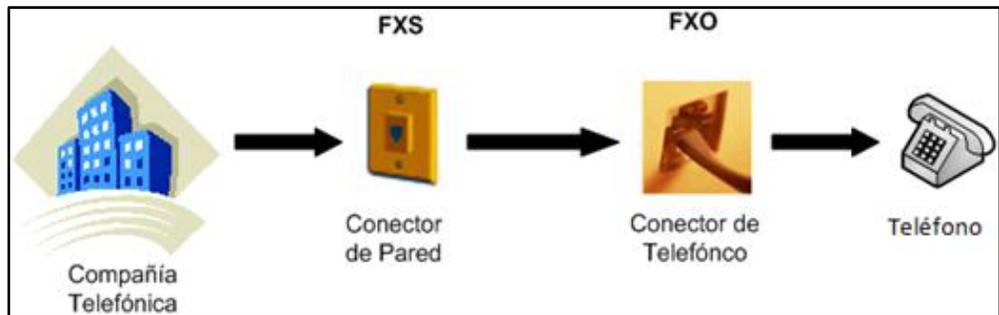


Figura 2.4.- Ejemplo de interacción FXS / FXO típica en un hogar

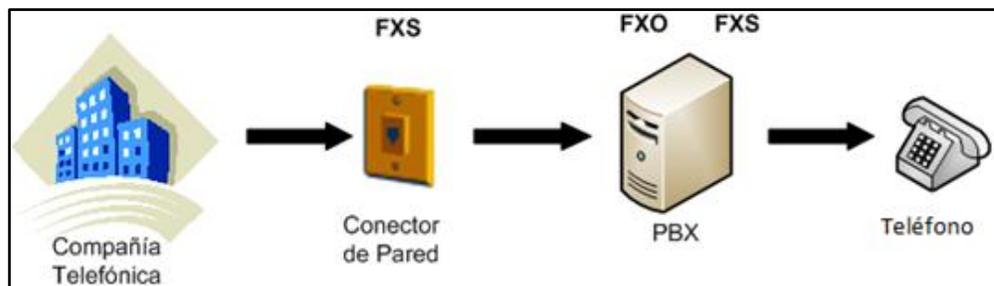


Figura 2.5.- Ejemplo de interacción FXS / FXO con centralita

2.5 FESTIVAL

Es un software de libre distribución para sistemas Unix que genera a través de un texto plano una salida de voz audible y viceversa. Festival es multilinguaje, inglés (británico y americano) y español, siendo más avanzado el primero.

CAPÍTULO 3

ANÁLISIS Y DISEÑO

3.1 ANÁLISIS

Cuando se analizaron las debilidades que presenta el sistema telefónico Audio ESPOL, se encontraron oportunidades para mejorar dicha herramienta, aprovechando las ventajas que ofrece la telefonía IP y los programas de código abierto. Por este motivo centramos el proyecto en los siguientes puntos:

1. Una sola red. No se desea implementar una nueva red para prestar el servicio, sino emplear la red de datos actual. Una PBX IP envía paquetes a través de la red de Internet lo cual es beneficioso porque elimina la gran infraestructura de una central telefónica. Para esto se requiere de un ancho de banda significativo, que el

servidor del sistema posea un procesador y una memoria robustos para que soporte un alto número de llamadas. De esta manera se tendrá la seguridad de brindar un servicio de calidad a cada usuario.

2. Flexibilidad. Se requiere de un sistema que se adapte de manera flexible a los requerimientos que surjan. Realizar el mantenimiento y actualización de un sistema VoIP es más flexible que un sistema analógico debido a su interoperabilidad. Para lo cual se requiere del correcto funcionamiento de todos los programas utilizados y las dependencias de las mismas, para contar con un sistema confiable y solvente.
3. Interactivo. El sistema debe ser sencillo de manejar para los estudiantes, permitiendo una rápida adaptación hacia el sistema, por lo que se realizó Audio ESPOL IP en base a las opciones que presenta el sitio web Académico en Línea. Se investigaron cuales consultas prestaban mayor utilidad para el alumnado y se prepararon grabaciones inspiradas por las opciones de menú del sitio web Académico en Línea.
4. Mantenimiento. El mantenimiento del sistema IP a nivel lógico permite realizar cambios específicos sin intervenir en otras áreas

del sistema. El mantenimiento a nivel físico es menos complejo que uno analógico debido a que en una central IP no se hace uso del banco de conmutación. Para un mantenimiento sencillo se proporciona una documentación detallada en cada segmento del sistema, se guardan los archivos de voz con nombres que identifican lo que se encuentra grabado y en la base de datos los campos se nombran de manera sencilla de identificar.

3.2 DISEÑO DE LA SOLUCIÓN

Al analizar el problema se determinó que existen dos etapas de diseño contenidas en el diseño del presente proyecto: el diseño de la base de datos y el diseño del plan de marcado.

3.2.1 DISEÑO DE LA BASE DE DATOS

Se escogió MySQL como motor de búsqueda debido a que es un gestor de código abierto e integración perfecta con PHP, lo cual nos facilita la elección del lenguaje de programación, además no tiene límite en el tamaño de los registros y posee un bajo costo en requerimientos para la elaboración de bases de datos. En base al análisis del sistema se decidió crear siete tablas

1. Cdr, proporciona información del ingreso al sistema como: hora, fecha, fuente y consulta que se realizó. Esta información es útil en temas de seguridad, operabilidad y funcionamiento del sistema.
2. Materia, almacena información de las materias como: código, nombre, paralelo, y créditos.
3. Detalle_materia, almacena información más específica sobre la materia como: paralelo, aula, día y hora en la cual se dicta la misma.
4. Estudiante, almacena información sobre el estudiante como: número de matrícula, nombres, apellidos, promedio general y la carrera.
5. Materia_estudiante, almacena información de cada materia registrada en un año y término específico por estudiante. Estos datos incluyen: código de la materia, paralelo, calificaciones (parcial, final y de mejoramiento), promedio, si aprobó o reprobó la materia y si el estudiante se encuentra registrado en la materia en el término actual.

6. Examen, almacena el código de la materia, tipo de examen (parcial, final o mejoramiento), paralelo, aula, fecha y hora del examen.
7. F_registro, almacena información sobre las fechas de registros de un año y término específico, estas fechas se basan en un intervalo de promedio.

En el Anexo B en la Figura B.1 se muestra el modelo lógico de la base de datos utilizado en el sistema.

3.2.2 DISEÑO DEL PLAN DE MARCADO

Para poder suplir las necesidades de los estudiantes de la ESPOL, se decidió que el sistema brinde las siguientes opciones de consulta

- Consulta de las calificaciones de las materias tomadas en un término y año específico dado el número de matrícula del estudiante.
- Consulta de fechas de registro para el próximo término.
- Consulta de fecha de Exámenes en el presente término.
- Consulta de horario de clases por día.

Como estrategia al momento de implementar el plan marcado se lo dividió en diferentes contextos. Dentro de cada contexto se procesa una opción de consulta diferente, lo cual hace más sencilla la administración del plan de marcado y fácil de extender si en un futuro se decide implementar más opciones de consultas.

Para realizar la conexión con la base de datos y el procesamiento de la información de las consultas se decidió usar scripts AGI escritos en PHP.

Para realizar las grabaciones se decidió usar la aplicación System de Asterix, utilizando la aplicación text2wave de festival que crea un archivo de audio a partir de un texto dado.

CAPÍTULO 4

IMPLEMENTACIÓN PILOTO

4.1 SELECCIÓN DE HARDWARE

La selección del hardware necesario para implementación del sistema Audio ESPOL IP, depende del modelo de infraestructura escogido para su funcionamiento. De acuerdo al análisis realizado se cuenta con los siguientes dos escenarios:

1. La ESPOL cuenta con un servidor VoIP compatible con Asterisk, el cual es el encargado de gestionar todas las llamadas con la PSTN, por lo que es necesario la interconexión del servidor VoIP de la ESPOL al servidor Audio ESPOL IP por medio de un enlace troncal, usando el protocolo IAX.

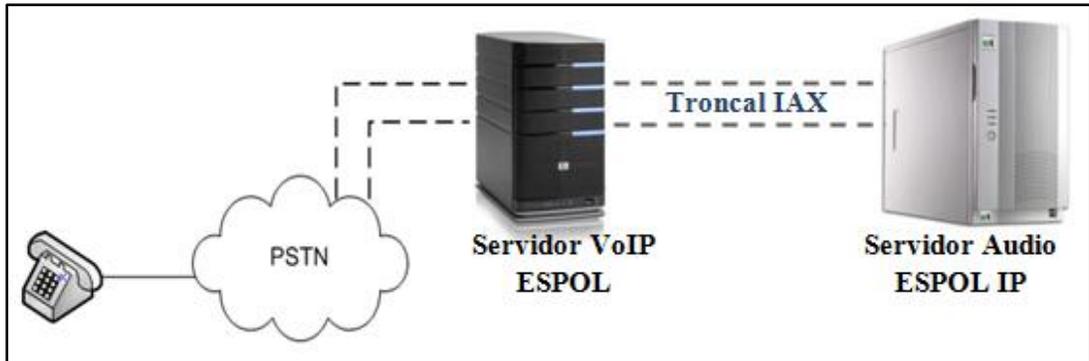


Figura 4.1 Diseño de la infraestructura del Sistema Audio ESPOL IP interconectado con el servidor VoIP de la ESPOL

2. El servidor Audio ESPOL IP es responsable de gestionar las llamadas entrantes al sistema desde la PSTN, por lo tanto es necesario su comunicación con la PSTN. Para realizar esta interconexión se utilizan tarjetas de telefonía analógica

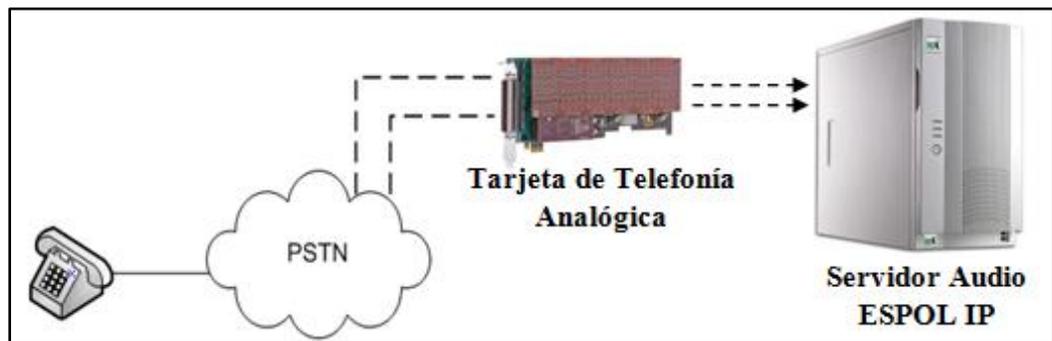


Figura 4.2 Diseño de la infraestructura del Sistema Audio ESPOL IP conectado a la PSTN usando Tarjeta de telefonía analógica Digium

Para la selección de la tarjeta analógica se toma en cuenta los siguientes tres aspectos:

- No se necesitan puertos FXS, debido a que no es necesario conectar teléfonos analógicos al sistema.
- La tarjeta debe de prestar funciones de cancelación de eco, para garantizar QoS del sistema.
- Tener un número considerable de puertos FXO, para permitir un alto número de llamadas concurrentes.

En base a lo expuesto anteriormente se recomienda usar la tarjeta analógica de Digium modelo 1AEX2406EF, que tiene las especificaciones mostradas en la Tabla I.

Tabla I Especificaciones de la Tarjeta de telefonía analógica 1AEX2406EF

| Especificaciones de la Tarjeta 1AEX2406EF | |
|--|----|
| Puertos analógicos FXS | 0 |
| Puertos analógicos FXO | 24 |
| Cancelación de Eco por Hardware | Si |
| Tarjeta modular full length PCI- Express | |

Para los requerimientos del sistema Audio ESPOL IP es necesario que el servidor cuente, como mínimo, con las características de hardware mostradas en la Tabla II.

Tabla II.- Características de hardware del servidor

| Características de hardware del servidor | |
|--|----------------------|
| Procesador | Intel I3 o Dual Core |
| Memoria RAM | 2 GB |
| Tarjeta de Red (Escenario 1) | 100/1000 Mbps |
| Soporte para tarjeta PCIe Full Length (Escenario 2) | |

De acuerdo al análisis expuesto con anterioridad **se seleccionó el escenario 2** para la implementación piloto del sistema.

4.2 INSTALACIONES

4.2.1 INSTALACIÓN DE PHP

Se instala el compilador de PHP5, necesario para ejecutar los scripts AGI, que van a estar escritos en dicho lenguaje. Para instalar PHP se ejecuta en consola el comando:

```
yum install -y php*
```

Se descarga la librería PHPAGI versión 2.20, y se la copia dentro del directorio agi-bin (/var/lib/asterisk/agi-bin), para usar las funciones implementadas en esta librería.

4.2.2 INSTALACIÓN DE MYSQL

Se instala el motor de base de datos MySQL ejecutando en consola el comando:

```
yum install -y mysql* unixODBC-devel* libtool-ltdl-devel
```

Se instala las herramientas: Mysql Administrator y Mysql Query Browser, que proporcionan una interfaz gráfica, lo que hace más sencillo la administración de la base de datos. Estas herramientas se descargan de la página oficial de MySQL, <http://dev.mysql.com>.

Una vez concluido el proceso de instalación de MySQL se procede a levantar el servicio MySQL, ejecutando en consola el comando:

```
service mysqld start
```

Como medida de seguridad se procede a asignar al usuario root de MySQL una contraseña, ejecutando en consola el comando:

```
mysqladmin -u root password root
```

4.2.3 INSTALACIÓN DE FESTIVAL

Para instalar festival, se ejecuta en consola el comando:

```
yum install festival festival-devel
```

Se puede configurar el programa festival para que sintetice voz en español, para ello es necesario descargar paquetes de voz en español y escribir en los archivos de configuración de festival.

El paquete de voz en español usado en el presente proyecto es **festvox-palpc16k_1.0-1_all.deb**, que pertenece a la Junta de Andalucía y se lo descargó del siguiente enlace: http://forja.guadalinux.org/repositorio/frs/download.php/153/festvox-palpc16k_1.0-1_all.deb.

El paquete de voz que se descarga no es compatible con la distribución de Linux, Centos, por lo que antes instalarlo es necesario convertirlo a un paquete compatible con Centos. Para convertir el paquete se usa el programa **Alien**.

Las fuentes del programa Alien se las descargó del enlace http://ftp.de.debian.org/debian/pool/main/a/alien/alien_8.78.tar.gz.

Una vez conseguidas las fuentes del programa y dentro del

directorio Alien se instala el programa, ejecutando en consola los comandos:

```
perl Makefile.PL
```

```
make
```

```
make install
```

Se transforma el paquete de voz festvox-palpc16k_1.0-1_all.deb descargado anteriormente, ejecutando el comando:

```
alien -rv festvox-palpc16k_1.0-1_all.deb
```

Del proceso anterior se obtiene el paquete festvox-palpc16k-1.0-2.noarch.rpm el cual se instala en el sistema, ejecutando el comando:

```
rpm -ivh festvox-palpc16k-1.0-2.noarch.rpm
```

Se edita el archivo **festival.scm**, localizado en el directorio **/usr/share/festival/**, para agregarle las líneas a continuación:

```
;(language__spanish)
(set! voice_default
'voice_JuntaDeAndalucia_es_pa_diphone)
```

```
(define (tts_textasterisk string mode)
"(tts_textasterisk STRING MODE)
Apply tts to STRING. This function is specifically
designed for
use in server mode so a single function call may
synthesize the string.
```

```

This function name may be added to the server safe
functions."
(let ((wholeutt (utt.synth (eval (list 'Utterance 'Text
string))))))
(utt.wave.resample wholeutt 8000)
(utt.wave.rescale wholeutt 5)
(utt.send.wave.client wholeutt))

```

Una vez concluido todo el proceso anterior se ejecuta Festival como servicio, ejecutando los comandos:

```

festival --server

festival_server 2>&1 >/dev/null &

```

4.2.4 INSTALACIÓN DE ASTERISK

Para la instalación de Asterisk se siguen los siguientes pasos:

1. Comprobar que el S.O. tenga la misma versión de kernel y de las fuentes del Kernel, esto es necesario para poder compilar el código fuente. Para comprobar esto se ejecuta los comandos:

```

rpm -q kernel

rpm -q kernel-devel.

```

Si se da el caso de tener versiones diferentes, es necesario ejecutar el comando **yum update**, con esto se actualiza todo el sistema.

2. Configurar el sistema para que utilice un servidor de tiempo externo, ejecutando el siguiente comando:

```
yum install -y ntp && ntpdate mx.pool.ntp.org &&  
chkconfig ntpd on && service ntpd start
```

3. Descargar y descomprimir las fuentes de Asterisk, Dhadi y libpri de la página oficial de Asterisk:
<http://downloads.asterisk.org/pub/telephony/>

4. A continuación instalar Libpri, ejecutando los siguientes comandos:

```
cd /usr/src/libpri-1.4.11.5  
make  
make install
```

5. Luego instalar DADHI, ejecutando los siguientes comandos:

```
cd /usr/src/ dahdi-linux-complete-2.4.1.2+2.4.1  
make  
make install  
make config
```

6. Instalar Asterisk, esto se hace ejecutando los siguientes comandos:

```
cd /usr/src/asterisk-1.8.3.2
```

```
./configure
```

```
make menuselect
```

```
make install
```

```
make samples
```

```
make config
```

4.3 CREACIÓN DE LA BASE DE DATOS

El sistema utiliza una base de datos llamada **Asterisk**. Para crear la base de datos **Asterisk** y sus tablas se ingresa a la terminal de líneas de comando de MySQL y desde ahí se ejecutan los siguientes comandos:

```
create database asterisk;
```

```
use Asterisk;
```

El script utilizado para la creación de las tablas se muestra en el Anexo B, Script Creación De Tablas Para La Base De Datos Asterisk.

4.4 CONFIGURACIONES DE ARCHIVOS DE ASTERISK

La forma de interactuar con la PBX Asterisk es a través de sus Archivos de configuración, estos archivos se encuentran dentro del directorio `/etc/asterisk`.

4.4.1 PLAN DE MARCADO

El archivo de configuración `extensions.conf` contiene el plan de marcado de la central telefónica. El plan de marcado en Asterisk define el flujo y comportamiento de todas las llamadas entrantes y salientes del sistema, en él se definen también los números de extensiones o conexiones de la central.

El plan de marcado se encuentra dividido en secciones. Existen dos secciones especiales que son: **general** y **globals**; las que contienen configuraciones y definiciones de variables respectivamente.

Después de estas secciones se definen secciones especiales, llamados **contextos**, que son una colección de extensiones; estos contextos además pueden ser definidos en otros archivos que son referenciados dentro del archivo `extensions.conf`.

Las **extensiones** son patrones que pueden estar compuestos por números, letras o ambos, que la central reconoce y que tienen asociado una secuencia de instrucciones que se desarrollan cuando los dispositivos que tienen acceso a la central las marcan.

Contexto general

Se define el parámetro **autofallthrough=yes**, para que Asterisk, una vez que la lógica de programación de una extensión termine, se encargue de terminar la llamada con BUSY, CONGESTION o HANGUP, que es la conducta deseada.

[general]

autofallthrough=yes

clearglobalvars=no

Contexto Globals

No hizo falta definir ninguna variable dentro de globals

[globals]

Contextos entrantes-PSTN

Este es el contexto o lugar de llegada de todas las llamadas entrantes al sistema. Las llamadas son recibidas por la extensión **s**, que en Asterisk es una extensión especial que toma el control de

todas las llamadas que son dirigidas a un contexto sin una extensión específica. Después que la llamada es contestada con la aplicación Answer(), se ejecuta la siguiente sentencia y la llamada es redirigida al contexto internos, extensión 888, prioridad 1, haciendo uso de la aplicación Goto()

```
[entrantes-PSTN]
```

```
exten => s,1,Answer()
```

```
same=>n,Goto(internos,888,1)
```

Contexto internos

Dentro de este contexto se definen las extensiones **124** y **888**.

La extensión **124** se utiliza para realizar las grabaciones que se usan en el sistema, haciendo uso de la aplicación System y el sintetizador de voz Festival.

```
exten => 124,1,Answer()
```

```
exten =>124,2,System(echo "Anio" | /usr/bin/text2wave -  
scale 1.5 -F 8000 -o
```

```
/var/lib/asterisk/sounds/tmp/anio.wav)
```

```
exten=>124,3,Playback(/var/lib/asterisk/sounds/tmp/anio)
```

```
exten => 124,4,Hangup()
```

A la extensión **888**, llegan todas las llamadas externas que fueron redirigidas desde el contexto **from-pstn**. Aquí se hace uso de la aplicación AGI, para llamar al script ingreso_sistema.php que se encarga de reproducir una grabación solicitando al usuario que digite su número de matrícula. Si existe un estudiante registrado en la base de datos del sistema con el número de matrícula ingresado se redirige la llamada al contexto **Menú Principal**.

`exten => 888,1,AGI(ingreso_sistema.php)`

Contexto Menú_Principal

Este contexto reproduce la grabación Opciones_Menu_Principal, que contienen las opciones de consulta del sistema. Los usuarios pueden realizar sus consultas, presionando la extensión correspondiente a la consulta que quieren hacer dentro de las opciones de las grabaciones del menú.

Si el estudiante presiona una extensión que no está definida en el contexto la entrada será recibida por la extensión **i**, extensión especial de Asterisk usada para esos casos. Se reproduce una grabación de opción inválida y se vuelve a reproducir la grabación que tiene las opciones de consulta, esperando por una opción válida. Si el tiempo de espera definido en la aplicación WaitExten()

es superado y el usuario no ha presionado ningún dígito, el control pasa a la extensión **t**, llamada **time out**, y se procede a colgar el canal.

[MenuPrincipal]

exten => start,1,Background(Opciones_Menu_Principal)

same => n,WaitExten(10)

; presione 1 consultar calificaciones.

exten => 1,1,Goto(Calificaciones,start,1)

; presione 2 consultar fecha de registros

exten => 2,1,Goto(FechaRegistros,start,1)

; presione 3 consultar horario de exámenes

exten => 3,1,Goto(HorarioExámenes,start,1)

; presione 4 consultar horario de clases

exten => 4,1,Goto(HorarioClases,start,1)

; presione 5 para volver a escuchar el menú

exten => 5,1,Goto(MenuPrincipal,start,1)

; presione 6 para salir de sistema

exten => 6,1,Hangup

exten => i,1,Playback(opcion_invalida)

same => n,Background(Opciones_Menu_Principal)

same => n,WaitExten(5)

exten => t,1,Hangup

Cada consulta en el sistema se hace dentro de un contexto diferente. Esto se aprecia en la Tabla III.

Tabla III Tabla de consultas que se puede realizar con su respectivo contexto

| N° de Opción de Consulta | Nombre de la Consulta | Contexto |
|--------------------------|--|-----------------|
| 1 | Consulta de Calificaciones por Año y Término | Calificaciones |
| 2 | Consulta de Fecha de Registro | FechaRegistros |
| 3 | Consulta de Horario de Exámenes | HorarioExámenes |
| 4 | Consulta de Horario de Clases | HorarioClases |

Los contextos mencionados en la **Tabla III**, se configuraron dentro del archivo extensions.conf. Los cuales se pueden observar en el plan de marcado adjunto en el Anexo C.

4.4.2 SCRIPTS AGI

La parte lógica así como la comunicación a la base de datos **Asterisk**, necesarias en cada opción de consulta, se implementó usando diferentes scripts AGI. Los scripts AGI que se usaron en el sistema se encuentran en la Tabla IV.

Tabla IV.- Lista de scripts usados en cada consulta

| Nombre de la Consulta | Script |
|--|-----------------------|
| Consultar existencia de matrícula en la base | ingreso_sistema.php |
| Consulta de Calificaciones por Año y Término | consultarMateria.php |
| Consulta de Fecha de Registro | consultarRegistro.php |
| Consulta de Horario de Exámenes | HorExamenes.php |
| Consulta de Horario de Clases | HorClases.php |

El script **ingreso_sistema.php**, se encarga del proceso de ingreso de un usuario al sistema.

El script reproduce una grabación pidiendo el usuario ingrese un número de matrícula, se captura el valor ingresado por el usuario en la variable **\$mat**, luego se verifica el contenido de la variable. Si el contenido es vacío el usuario no ingreso nada en el tiempo límite y se procede a colgar el canal, caso contrario se reproduce el valor ingresado por el usuario y a continuación se reproduce una grabación en la que se pide al usuario que confirme si está de acuerdo o no con el número de matrícula ingresado. Si el usuario confirme el número de matrícula se procede a consultar la

existencia del estudiante en la base de datos **Asterisk** usando el número de matrícula ingresado. Si existe un estudiante con ese número matrícula se almacena dicho valor en la variable **__matricula**, la cual va a seguir existiendo una vez que el script deje de ejecutarse y pase el control nuevamente al plan de marcado. Como se muestra en el Anexo D, Script Para Ingreso Al Sistema.

El script **consultarMateria.php**, se encarga de la consulta de las calificaciones dado un año y término específicos

Dicho script recibe como parámetros las variables **matrícula, año y término**, valores ingresados por el usuario dentro del plan de marcado. Estos parámetros son leídos dentro del script y almacenado en las variables \$mat, \$anio y \$ter, en base a estas variables se consulta en la base de datos las calificaciones del estudiante en dicho año y término. Los resultados de la consulta son reproducidos al usuario, si no existen datos de la consulta se reproduce una grabación con ese mensaje. Una vez hecha la consulta, el usuario puede volver a escuchar la consulta o volver al

menú principal, en cuyo caso se termina de ejecutar el script. Como se muestra en el Anexo D, Script Para Consultar Materia.

El script **consultarRegistro.php** recibe como parámetros las variables **matrícula y exten**. Se usa la variable **exten** para definir el tipo de consulta que quiere realizar el usuario: 1 para consultar solo su fecha de registro y 2 para consultar todas las fechas de registros. Estos parámetros son leídos dentro del script y almacenado en las variables \$mat y \$exten, en base a estos parámetros se realiza la consulta en la base de datos. Los resultados de la consulta son reproducidos al usuario, si no existen datos de la consulta se reproduce una grabación con ese mensaje. Una vez hecha la consulta, el usuario puede volver a escuchar la consulta o volver al menú principal, en cuyo caso se termina de ejecutar el script. Como se muestra en el Anexo D, Script Para Consultar Registro.

El script **HorExámenes.php** recibe como parámetros las variables **matrícula y exten**. Se usa la variable **exten** para definir el tipo de consulta que quiere realizar el estudiante: 1 para consultar su

horario de exámenes primera evaluación, 2 para consultar su horario de exámenes segunda evaluación y 3 para consultar su horario de exámenes tercera evaluación. Estos parámetros son leídos dentro del script y almacenado en las variables \$mat y \$opcion, con ellas se procede a ser la consulta en la base de datos. Los resultados de la consulta son reproducidos al usuario, si no existen datos de la consulta se reproduce una grabación con ese mensaje. Una vez hecha la consulta, el usuario puede volver a escuchar la consulta o volver al menú principal, en cuyo caso se termina de ejecutar el script. Como se muestra en el Anexo D, Script Para Consultar Horario de Exámenes.

El script **HorClases.php** recibe como parámetros las variables **matrícula, año y término**. Estos parámetros son leídos dentro del script y almacenado en las variables \$mat, \$anio y \$ter, con ellas se procede a ser la consulta en la base de datos de las calificaciones del estudiante en dicho año y término. Los resultados de la consulta son reproducidos al usuario, si no existen datos de la consulta se reproduce una grabación con ese mensaje. Una vez hecha la consulta, el usuario puede volver a escuchar la consulta o

volver al menú principal, en cuyo caso se termina de ejecutar el script. Como se muestra en el Anexo D, Script Para Consultar Horario de Clases.

4.4.3 DESCRIPCIÓN DE LOS ARCHIVOS DE AUDIO

Dentro del sistema se usan algunas grabaciones para interactuar con el usuario. Estas grabaciones fueron creadas usando el sintetizador de voz Festival, desde Asterisk con la extensión 124.

Las grabaciones que se crearon se listan en la Tabla V:

Tabla V.- Lista de Grabaciones Creadas con Festival

| | Nombre de la Grabación | Texto |
|-----------|-------------------------------|--|
| 1. | bienvenida.wav | Bienvenido al sistema de consultas académicas, Audio Espol Ip, porfavor ingrese su numero de matricula |
| 2. | ingrese_matricula.wav | Por favor ingrese su numero de matricula |
| 3. | opcion_invalida.wav | Opcion Invalida |
| 4. | Opciones_Menu_Principal.wav | Presione uno, para consulta de calificaciones, presione 2, para consulta de fecha de registros, presione 3, para |

| | | |
|------------|------------------------|--|
| | | consulta de horario de exámenes, presione 4, para consulta de horario de clases, presione 5, para volver escuchar las opciones del menu, presione 6, o cuelgue para terminar |
| 5. | ingrese_anio.wav | Ingrese Anio |
| 6. | ingrese_termino.wav | Ingrese Termino |
| 7. | anio.wav | Anio |
| 8. | termino.wav | Termino |
| 9. | Confirma_Ing_Calif.wav | Presione uno, para realizar la consulta, presione 2, para volver a ingresar los datos, presione 0, para volver al menu principal |
| 10. | Escuchar_Con_again.wav | Presione uno, para volver a escuchar la consulta, presione 0, para volver al menu principal, o cuelgue para terminar |
| 11. | FechaRegistros.wav | No existen datos para presentar, el estudiante no ha tomado materias en dicho año y termino ;No_existen_Calificaciones.wav 12.- Presione uno, para consultar su |

| | | |
|------------|-----------------------------|--|
| | | fecha de registro, presione dos, para consultar el calendario de fechas de registros, presione 0, para volver al menu principal |
| 12. | Menu_Con_Fecha_Examenes.wav | Presione uno, para consultar su horario de examenes Parciales, presione dos, para consultar su horario de examenes Finales, presione tres, para consultar su horario de examenes de Mejoramiento, presione 0, para volver al menu principal |
| 13. | No_tiene_materias.wav | No existen datos para presentar, el estudiante no se encuentra tomando materias |
| 14. | HorarioClases.wav | Presione uno, para consultar su horario del lunes, presione dos, para consultar su horario del martes, presione tres, para consultar su horario del miercoles, presione cuatro, para consultar su horario del jueves, presione cinco, para consultar su horario del viernes, presione 0, para volver al menu principal |

| | | |
|------------|-------------------------------|--|
| 15. | eleccion_matricula.wav | Presione uno para confirmar la matricula ingresada, presione cero, para volver a ingresar el numero de matricula |
| 16. | No_existen_Calificaciones.wav | No existen datos para presentar, el estudiante no ha tomado materias en dicho año y termino |

CAPÍTULO 5

PRUEBAS Y ANÁLISIS DE LOS RESULTADOS

Para verificar la capacidad del sistema Audio ESPOL IP se realizan pruebas de funcionalidad y rendimiento. Estas pruebas se realizan ingresando dentro de la base de datos Asterisk, información académica de dos estudiantes con los números de matrícula 200703072 y 200612745. Las llamadas al sistema se las realiza utilizando el softphone Zoiper. La Tabla VI muestra las especificaciones del servidor Asterisk que contiene el sistema Audio ESPOL IP utilizado en ambas pruebas.

Tabla VI.- Especificaciones del servidor Asterisk

| SERVIDOR ASTERISK | |
|----------------------------|---|
| Modelo | HP Pavilion DV4-2165DX |
| Procesador | Intel ® core [™] i3 CPU M330 @ 2.13GHz |
| Memoria | 2GB |
| Sistema Operativo | CentOS release 5.6 (Final) |
| Version de Asterisk | Asterisk 1.8.3.2 |

5.1 PRUEBAS DE FUNCIONALIDAD

El objetivo de estas pruebas es mostrar cómo funciona el sistema Audio ESPOL IP, al momento de que un usuario llama al sistema y realiza una consulta.

Durante las pruebas de funcionalidad las llamadas se realizan desde un softphone, el cual se configura como un agente SIP dentro del servidor Audio ESPOL IP. Para acceder al sistema se marca la extensión 888 desde el softphone.

A continuación se muestran los diferentes escenarios de las consultas con su respectivo reporte desde una terminal de Asterisk.

1. Ingreso al sistema, desde el softphone al marcar el número 888 se establece una conexión con el sistema Audio ESPOL IP, el cual responde la llamada reproduciendo la grabación bienvenida, la cual brinda la bienvenida al sistema al usuario y le solicita que ingrese un número de matrícula. El sistema espera por cuatro segundos a que el usuario ingrese su número de matrícula, si durante este tiempo el usuario no ingresa nada se cuelga la llamada. En la Figura 5.1 se muestra la salida del terminal.

Tabla VII.- Prueba de Funcionalidad: Ingreso al Sistema

| Ingreso | Salida |
|--|--|
| Se marca 888 desde el softphone | Se crea el canal seminario-00000004 |
| | Toma el control de la llamada el AGI ingreso_sistema.php |
| | Responde la llamada la aplicación Answer() y toma control la aplicación GetData(bienvenida,4000,9) |

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
[root@localhost ~]# asterisk -r
Verbosity is at least 9
== Using SIP RTP CoS mark 5
-- Executing [888@internos:1] AGI("SIP/seminario-00000004", "ingreso_sistema.php") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/ingreso_sistema.php
<SIP/seminario-00000004>AGI Tx >> agi_request: ingreso_sistema.php
<SIP/seminario-00000004>AGI Tx >> agi_channel: SIP/seminario-00000004
<SIP/seminario-00000004>AGI Tx >> agi_language: es
<SIP/seminario-00000004>AGI Tx >> agi_type: SIP
<SIP/seminario-00000004>AGI Tx >> agi_uniqueid: 1321827203.4
<SIP/seminario-00000004>AGI Tx >> agi_version: 1.8.3.2
<SIP/seminario-00000004>AGI Tx >> agi_callerid: 301
<SIP/seminario-00000004>AGI Tx >> agi_calleridname: VoIP
<SIP/seminario-00000004>AGI Tx >> agi_callingpres: 0
<SIP/seminario-00000004>AGI Tx >> agi_callingani2: 0
<SIP/seminario-00000004>AGI Tx >> agi_callington: 0
<SIP/seminario-00000004>AGI Tx >> agi_callingtns: 0
<SIP/seminario-00000004>AGI Tx >> agi_dnid: 888
<SIP/seminario-00000004>AGI Tx >> agi_rdnis: unknown
<SIP/seminario-00000004>AGI Tx >> agi_context: internos
<SIP/seminario-00000004>AGI Tx >> agi_extension: 888
<SIP/seminario-00000004>AGI Tx >> agi_priority: 1
<SIP/seminario-00000004>AGI Tx >> agi_enhanced: 0.0
<SIP/seminario-00000004>AGI Tx >> agi_accountcode:
<SIP/seminario-00000004>AGI Tx >> agi_threadid: -1219855472
<SIP/seminario-00000004>AGI Tx >>
<SIP/seminario-00000004>AGI Rx << ANSWER
<SIP/seminario-00000004>AGI Tx >> 200 result=0
<SIP/seminario-00000004>AGI Rx << GET DATA bienvenida 4000 9

```

Figura 5.1.- Respuesta del Sistema al establecerse una llamada

2. Ingreso fallido, el usuario ingresa un número de matrícula no válido, entonces se reproduce la grabación `estudiante_no_existe`, en la que se informa al usuario que dicho número de matrícula no existe en el sistema.

Tabla VIII.- Prueba de Funcionalidad: Ingreso Fallido al Sistema

| Ingreso | Salida |
|----------------------------------|--|
| Usuario ingresó 200703072 | Se ejecuta la aplicación SayDigits() |
| | Se ejecuta aplicación GetData(elección_matricula,3000,1) , pidiendo confirmación de la matricula |
| Usuario ingreso 1 | Se consulta en la base de datos la matricula 200703075 , si esta no existe y se reproduce grabación estudiante_no_existe |

```

root@localhost:~
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
<SIP/seminario-00000004>AGI Tx >> 200 result=200703075
<SIP/seminario-00000004>AGI Rx << EXEC SayDigits 200703075
-- AGI Script Executing Application: (SayDigits) Options: (200703075)
-- <SIP/seminario-00000004> Playing 'digits/2.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/0.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/0.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/7.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/0.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/3.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/0.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/7.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/5.ulaw' (language 'es')
<SIP/seminario-00000004>AGI Tx >> 200 result=0
<SIP/seminario-00000004>AGI Rx << GET DATA eleccion_matricula 3000 1
-- <SIP/seminario-00000004> Playing 'eleccion_matricula.slin' (language 'es')
<SIP/seminario-00000004>AGI Tx >> 200 result=1
<SIP/seminario-00000004>AGI Rx << EXEC Playback estudiante_no_existe
-- AGI Script Executing Application: (Playback) Options: (estudiante_no_existe)
e)
-- <SIP/seminario-00000004> Playing 'estudiante_no_existe.slin' (language 'es')
')
    
```

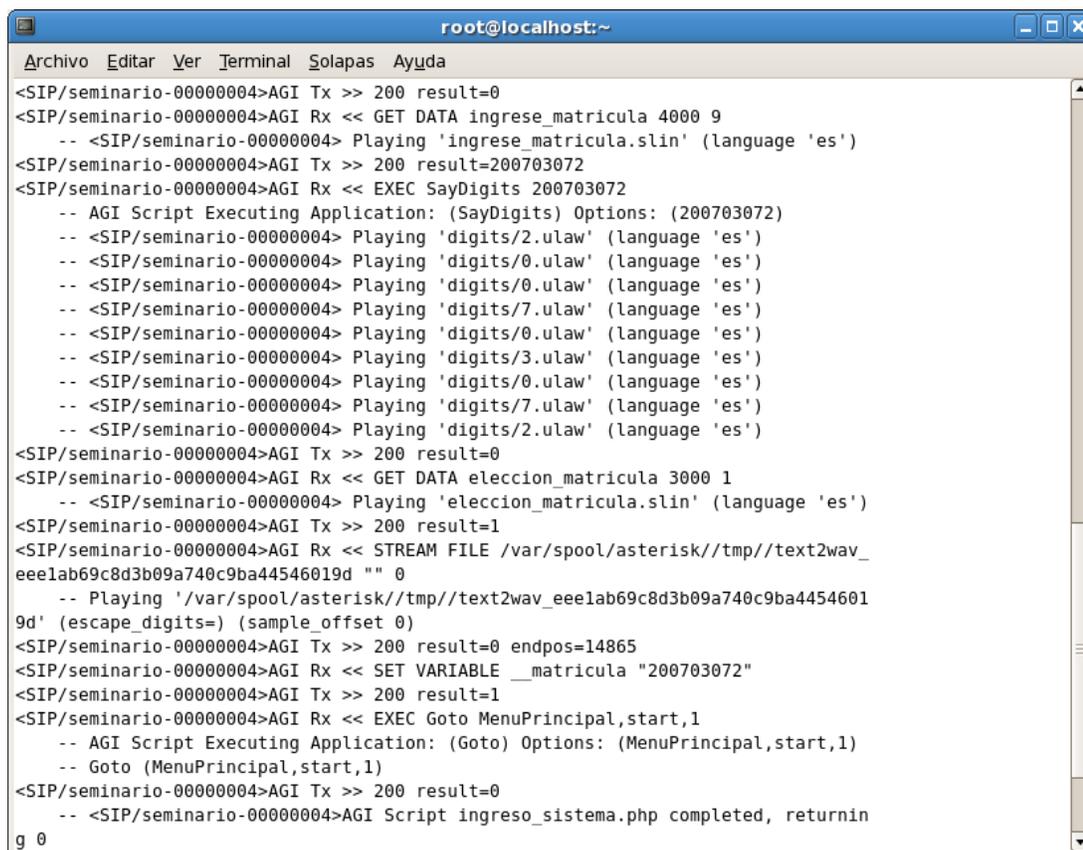
Figura 5.2.- Respuesta del Sistema al establecerse un ingreso fallido al sistema

3. Ingreso exitoso, después del ingreso fallido al sistema, se reproduce la grabación ingrese_matricula, el usuario ingresa un número de matrícula válido, se reproduce el nombre y apellido del estudiante usando la aplicación txt2wave, se almacena el número de matrícula ingresado en la variable matricula y se redirige la llamada a el contexto MenuPrincipal. Se cierra el canal AGI que había abierto.

Tabla IX.- Prueba de Funcionalidad: Ingreso Exitoso al Sistema

| Ingreso | Salida |
|---------|---|
| | Se ejecuta aplicación GetData(ingrese_matricula,4000,9), pidiendo |

| | |
|-------------------------------------|--|
| | ingreso de un número de matrícula |
| Usuario ingreso 200703072 | Se ejecuta la aplicación SayDigits() |
| | Se ejecuta aplicación GetData(elección_matricula,3000,1) , pidiendo al usuario confirme el número de ingresado. |
| Usuario ingresó 1 | Se consulta en la base de datos si existe un estudiante con número de matrícula 200703072 . |
| | Se ejecuta la aplicación text2wav(nombre y apellido), con el nombre y apellido del estudiante consultado desde la base de datos |
| | Se ejecuta la aplicación Set(__matricula=200703072) , que define la variable matrícula para este canal y todos los demás canales que se originen a partir de este |
| | Se ejecuta la aplicación Goto(MenuPrincipal,start,1) |



```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
<SIP/seminario-00000004>AGI Tx >> 200 result=0
<SIP/seminario-00000004>AGI Rx << GET DATA ingreso_matricula 4000 9
-- <SIP/seminario-00000004> Playing 'ingrese_matricula.slin' (language 'es')
<SIP/seminario-00000004>AGI Tx >> 200 result=200703072
<SIP/seminario-00000004>AGI Rx << EXEC SayDigits 200703072
-- AGI Script Executing Application: (SayDigits) Options: (200703072)
-- <SIP/seminario-00000004> Playing 'digits/2.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/0.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/0.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/7.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/0.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/3.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/0.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/7.ulaw' (language 'es')
-- <SIP/seminario-00000004> Playing 'digits/2.ulaw' (language 'es')
<SIP/seminario-00000004>AGI Tx >> 200 result=0
<SIP/seminario-00000004>AGI Rx << GET DATA eleccion_matricula 3000 1
-- <SIP/seminario-00000004> Playing 'eleccion_matricula.slin' (language 'es')
<SIP/seminario-00000004>AGI Tx >> 200 result=1
<SIP/seminario-00000004>AGI Rx << STREAM FILE /var/spool/asterisk//tmp//text2wav_
eeelab69c8d3b09a740c9ba44546019d "" 0
-- Playing '/var/spool/asterisk//tmp//text2wav_eeelab69c8d3b09a740c9ba4454601
9d' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000004>AGI Tx >> 200 result=0 endpos=14865
<SIP/seminario-00000004>AGI Rx << SET VARIABLE __matricula "200703072"
<SIP/seminario-00000004>AGI Tx >> 200 result=1
<SIP/seminario-00000004>AGI Rx << EXEC Goto MenuPrincipal,start,1
-- AGI Script Executing Application: (Goto) Options: (MenuPrincipal,start,1)
-- Goto (MenuPrincipal,start,1)
<SIP/seminario-00000004>AGI Tx >> 200 result=0
-- <SIP/seminario-00000004>AGI Script ingreso_sistema.php completed, returnin
g 0

```

Figura 5.3.- Respuesta del Sistema al establecerse un ingreso exitoso al sistema

4. Consultar calificaciones por año y término, la llamada se encuentra dentro del contexto MenuPrincipal, se reproduce la grabación Opciones_Menu_Principal, con las opciones de consulta del menú. El usuario ingresa 1 con lo que consulta las calificaciones por año y término específicos.

Tabla X.- Consulta de Calificaciones: Ingreso de año y término

| Ingreso | Salida |
|-------------------------|--|
| | Se ejecuta la aplicación BackGround(Opciones_Menu_Principal) , con las opciones de la consulta |
| Usuario ingresa 1 | Se ejecuta aplicación Goto(Calificaciones,start,1) . La llamada es redirigida al contexto Calificaciones |
| Usuario ingresa 2011 | Se ejecuta la aplicación Read(anio,ingreso_anio,4) , con lo que se ingresa el año de la consulta |
| | Se ejecuta aplicación System que crea una grabación del año ingresado |
| Usuario Ingresa 1 | Se ejecuta la aplicación Read(termino,ingreso_termino,1) , con lo que se ingresa el termino de la consulta |
| | Se ejecuta aplicación System que crea una grabación del término ingresado |
| | Se reproduce las grabaciones anio y termino antes creadas con la aplicación System |

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
-- Executing [start@MenuPrincipal:1] Background("SIP/seminario-00000006", "Opciones_Menu_Princip
al") in new stack
-- <SIP/seminario-00000006> Playing 'Opciones_Menu_Principal.slin' (language 'es')
-- Executing [l@MenuPrincipal:1] Goto("SIP/seminario-00000006", "Calificaciones,start,1") in new
stack
-- Goto (Calificaciones,start,1)
-- Executing [start@Calificaciones:1] Read("SIP/seminario-00000006", "anio,ingrese_anio,4") in n
ew stack
-- Accepting a maximum of 4 digits.
-- <SIP/seminario-00000006> Playing 'ingrese_anio.slin' (language 'es')
-- User entered '2011'
-- Executing [start@Calificaciones:2] System("SIP/seminario-00000006", "echo "2011" | /usr/bin/t
ext2wave -scale 1.5 -F 8000 -o /var/lib/asterisk/sounds/es/ing_anio.wav") in new stack
-- Executing [start@Calificaciones:3] Read("SIP/seminario-00000006", "termino,ingrese_termino,1"
) in new stack
-- Accepting a maximum of 1 digits.
-- <SIP/seminario-00000006> Playing 'ingrese_termino.slin' (language 'es')
-- User entered '1'
-- Executing [start@Calificaciones:4] System("SIP/seminario-00000006", "echo "1" | /usr/bin/text
2wave -scale 1.5 -F 8000 -o /var/lib/asterisk/sounds/es/ing_termino.wav") in new stack
-- Executing [start@Calificaciones:5] Playback("SIP/seminario-00000006", "anio&ing_anio&termino&
ing_termino") in new stack
-- <SIP/seminario-00000006> Playing 'anio.slin' (language 'es')
-- <SIP/seminario-00000006> Playing 'ing_anio.slin' (language 'es')
-- <SIP/seminario-00000006> Playing 'termino.slin' (language 'es')
-- <SIP/seminario-00000006> Playing 'ing_termino.slin' (language 'es')
-- Executing [start@Calificaciones:6] System("SIP/seminario-00000006", "rm /var/lib/asterisk/sou
nds/es/ing_termino.wav") in new stack
-- Executing [start@Calificaciones:7] System("SIP/seminario-00000006", "rm /var/lib/asterisk/sou
nds/es/ing_anio.wav") in new stack

```

Figura 5.4.- Respuesta del Sistema al ingresar los datos de consulta para calificaciones

Tabla XI.- Consulta de Calificaciones: Confirmación de año y término

| Ingreso | Salida |
|-------------------|---|
| | Se reproduce la grabación Confirma_Ing_Calif, la que se pide al usuario que ingrese 1 para realizar la consulta o 0 para volver ingresar año y término |
| Usuario ingresa 1 | Se ejecuta la aplicación AGL(consultarMateria.php,200703072,2011,1) . Con lo que se pasa el control de la llamada a consultarMateria.php y se consulta las calificaciones del 1 Término año 2011 del estudiante . |

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
-- <SIP/seminario-00000006> Playing 'Confirma_Ing_Calif.slin' (language 'es')
-- Executing [start@Calificaciones:9] WaitExten("SIP/seminario-00000006", "5") in new stack
== CDR updated on SIP/seminario-00000006
-- Executing [1@Calificaciones:1] AGI("SIP/seminario-00000006", "consultarMateria.php,200703072,
2011,1") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/consultarMateria.php

```

Figura 5.5.- Respuesta del Sistema luego de la confirmación de año y término

Tabla XII.- Consulta de Calificaciones: Salida

| Ingreso | Salida |
|---------|--|
| | Se consulta en la base de datos con los parámetros pasados como argumentos |
| | Se reproducen las notas de las calificaciones en la que se encuentra el estudiante registrado en el año y termino antes especificados. Hacemos esto con la aplicación Text2wav. El estudiante se encontraba registrado en 5 materias |

```
root@localhost:~
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
<SIP/seminario-00000006>AGI Rx << ANSWER
<SIP/seminario-00000006>AGI Tx >> 200 result=0
<SIP/seminario-00000006>AGI Rx << STREAM FILE /var/spool/asterisk//tmp//text2wav_d433da290f2530df80e
acbf52a22aa35 "" 0
-- Playing '/var/spool/asterisk//tmp//text2wav_d433da290f2530df80eacbf52a22aa35' (escape_digits=
) (sample_offset 0)
<SIP/seminario-00000006>AGI Tx >> 200 result=0 endpos=73362
<SIP/seminario-00000006>AGI Rx << STREAM FILE /var/spool/asterisk//tmp//text2wav_d5f860a6f1e19db8d37
7b40abe67e4a1 "" 0
-- Playing '/var/spool/asterisk//tmp//text2wav_d5f860a6f1e19db8d377b40abe67e4a1' (escape_digits=
) (sample_offset 0)
<SIP/seminario-00000006>AGI Tx >> 200 result=0 endpos=64642
<SIP/seminario-00000006>AGI Rx << STREAM FILE /var/spool/asterisk//tmp//text2wav_7d8d36f78eacc780a24
25635de2789bf "" 0
-- Playing '/var/spool/asterisk//tmp//text2wav_7d8d36f78eacc780a2425635de2789bf' (escape_digits=
) (sample_offset 0)
<SIP/seminario-00000006>AGI Tx >> 200 result=0 endpos=68402
<SIP/seminario-00000006>AGI Rx << STREAM FILE /var/spool/asterisk//tmp//text2wav_007e31542a3af3d0cef
6b9b59d4518cc "" 0
-- Playing '/var/spool/asterisk//tmp//text2wav_007e31542a3af3d0cef6b9b59d4518cc' (escape_digits=
) (sample_offset 0)
<SIP/seminario-00000006>AGI Tx >> 200 result=0 endpos=69522
<SIP/seminario-00000006>AGI Rx << STREAM FILE /var/spool/asterisk//tmp//text2wav_b0d13c21a4b38f4ef6f
d5c48db11c483 "" 0
-- Playing '/var/spool/asterisk//tmp//text2wav_b0d13c21a4b38f4ef6fd5c48db11c483' (escape_digits=
) (sample_offset 0)
```

Figura 5.6.- Respuesta del sistema: salida de consulta de calificaciones por año y término

- 5. Consultar fecha de registro, la llamada se encuentra dentro del contexto MenuPrincipal, se reproduce la grabación Opciones_Menu_Principal, con las opciones de consulta del menú. El usuario ingresa 2, se realiza la consulta de fecha de registros.

Tabla XIII.- Consulta de Fecha de Registro: Menú Principal

| Ingreso | Salida |
|---------|---|
| | Se ejecuta la aplicación BackGround(Opciones_Menu_Principal) , con las opciones de la consulta |

| | |
|----------------------|--|
| Usuario ingresa 2 | Se ejecuta aplicación Goto(FechaRegistros,start,1) . La llamada es redirigida al contexto FechaRegistros |
| | Se ejecuta la aplicación Background(FechaRegistros) . Se reproduce la grabación FechaRegistros (1 consulta de su fecha de registro, 2 consulta de todas la fechas de registro) |

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
-- <SIP/seminario-0000000b> Playing 'Opciones_Menu_Principal.slin' (language 'es')
-- Executing [2@MenuPrincipal:1] Goto("SIP/seminario-0000000b", "FechaRegistros,start,1") in new stack
-- Goto (FechaRegistros,start,1)
-- Executing [start@FechaRegistros:1] Background("SIP/seminario-0000000b", "FechaRegistros") in new stack
-- <SIP/seminario-0000000b> Playing 'FechaRegistros.stin' (language 'es')
-- Executing [start@FechaRegistros:2] WaitExten("SIP/seminario-0000000b", "5") in new stack
== CDR updated on SIP/seminario-0000000b
    
```

Figura 5.7.- Respuesta del sistema: Opciones de Consulta de Fecha de Registro

Dentro de la consulta de fecha de registro brinda dos opciones: opción 1 para la consulta de fecha de registro dada una matrícula, como se muestra en la Figura 5.8 y 5.9; opción 2 para la consulta de todas las fechas de registro como se muestra en la Figura 5.10 y 5.11.

Tabla XIV.- Consulta de Fecha de Registro dado una matrícula

| Ingreso | Salida |
|-------------------------|--|
| Usuario ingresa 1 | Se ejecuta la aplicación AGI(consultarRegistro.php,\${matricula},\${exten}) . Con lo que se pasa el control de la llamada al script consultarRegistro.php , consulta la fecha de registro del estudiante en base de su promedio general y matrícula . |

```

-- Executing [1@FechaRegistros:1] AGI("SIP/seminario-000000b", "consultarRegistro.php,200703072,1") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/consultarRegistro.php
<SIP/seminario-000000b>AGI Tx >> agi_request: consultarRegistro.php
<SIP/seminario-000000b>AGI Tx >> agi_channel: SIP/seminario-000000b
<SIP/seminario-000000b>AGI Tx >> agi_language: es
<SIP/seminario-000000b>AGI Tx >> agi_type: SIP
<SIP/seminario-000000b>AGI Tx >> agi_uniqueid: 1321829687.11
<SIP/seminario-000000b>AGI Tx >> agi_version: 1.8.3.2
<SIP/seminario-000000b>AGI Tx >> agi_callerid: 301
<SIP/seminario-000000b>AGI Tx >> agi_calleridname: VoIP
<SIP/seminario-000000b>AGI Tx >> agi_callingpres: 0
<SIP/seminario-000000b>AGI Tx >> agi_callingani2: 0
<SIP/seminario-000000b>AGI Tx >> agi_callington: 0
<SIP/seminario-000000b>AGI Tx >> agi_callingtns: 0
<SIP/seminario-000000b>AGI Tx >> agi_dnid: 888
<SIP/seminario-000000b>AGI Tx >> agi_rdnis: unknown
<SIP/seminario-000000b>AGI Tx >> agi_context: FechaRegistros
<SIP/seminario-000000b>AGI Tx >> agi_extension: 1
<SIP/seminario-000000b>AGI Tx >> agi_priority: 1
<SIP/seminario-000000b>AGI Tx >> agi_enhanced: 0.0
<SIP/seminario-000000b>AGI Tx >> agi_accountcode:
<SIP/seminario-000000b>AGI Tx >> agi_threadid: -1219855472
<SIP/seminario-000000b>AGI Tx >> agi_arg_1: 200703072
<SIP/seminario-000000b>AGI Tx >> agi_arg_2: 1
<SIP/seminario-000000b>AGI Tx >>

```

Figura 5.8.- Respuesta del sistema: consulta de fecha de registro dado una matrícula

Tabla XV.- Consulta de Fecha de Registro: Salida

| Ingreso | Salida |
|--------------------------|--|
| | Se realiza una consulta a la base asterisk por la fecha de registro del estudiante. Se produce una salida de audio que dice la fecha a partir de la cual es estudiante está habilitado a registrarse usando la aplicación text2wav |
| | Se ejecuta la aplicación getData(Escuchar_con_again,5000,1), con lo que se pide al usuario que ingrese 1 para volver a escuchar la consulta y 0 para regresar al menú principal |
| Usuario Ingresa 0 | Se ejecuta la aplicación Goto(MenuPrincipal,start,1). Se termina el script y se regresa al plan de marcado |

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
<SIP/seminario-0000000b>AGI Rx << ANSWER
<SIP/seminario-0000000b>AGI Tx >> 200 result=0
<SIP/seminario-0000000b>AGI Rx << STREAM FILE /var/spool/asterisk//tmp//text2wav_4edbe5522beffb2aa6355ee6d82b314 "" 0
-- Playing '/var/spool/asterisk//tmp//text2wav_4edbe5522beffb2aa6355ee6d82b314' (escape_digits=) (sample_offset 0)
<SIP/seminario-0000000b>AGI Tx >> 200 result=0 endpos=38091
<SIP/seminario-0000000b>AGI Rx << GET DATA Escuchar_Con_again 5000 1
-- <SIP/seminario-0000000b> Playing 'Escuchar_Con_again.stin' (language es)
<SIP/seminario-0000000b>AGI Tx >> 200 result=0
<SIP/seminario-0000000b>AGI Rx << SET CONTEXT MenuPrincipal
<SIP/seminario-0000000b>AGI Tx >> 200 result=0
<SIP/seminario-0000000b>AGI Rx << EXEC Goto MenuPrincipal,start,1
-- AGI Script Executing Application: (Goto) Options: (MenuPrincipal,start,1)
-- Goto (MenuPrincipal,start,1)
<SIP/seminario-0000000b>AGI Tx >> 200 result=0
-- <SIP/seminario-0000000b>AGI Script consultarRegistro.php completed, returning 0
    
```

Figura 5.9.- Respuesta del sistema: salida de consulta de fecha de registro por número de matrícula

Tabla XVI.- Consulta de Fechas de Registro: Menú Principal

| Ingreso | Salida |
|----------------------|---|
| Usuario ingresa 2 | Se ejecuta la aplicación AGI(consultarRegistro.php,\${matricula},\${exten}) . Con lo que se pasa el control de la llamada al script consultarRegistro.php y se consulta todas las fechas de registro del siguiente termino |
| | Se realiza una consulta a la base asterisk por la fecha de registro del estudiante. Se produce una salida de audio que dice la fecha a partir de la cual es estudiante está habilitado a registrarse usando la aplicación text2wav |

```

-- Executing [start@FechaRegistros:1] Background("SIP/seminario-0000000b", "FechaRegistros") in new stack
-- <SIP/seminario-0000000b> Playing 'FechaRegistros.slin' (language 'es')
-- Executing [2@FechaRegistros:1] AGI("SIP/seminario-0000000b", "consultarRegistro.php,200703072,2") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/consultarRegistro.php
<SIP/seminario-0000000b>AGI Tx >> agi_request: consultarRegistro.php
<SIP/seminario-0000000b>AGI Tx >> agi_channel: SIP/seminario-0000000b
<SIP/seminario-0000000b>AGI Tx >> agi_language: es
<SIP/seminario-0000000b>AGI Tx >> agi_type: SIP
<SIP/seminario-0000000b>AGI Tx >> agi_uniqueid: 1321829687.11
<SIP/seminario-0000000b>AGI Tx >> agi_version: 1.8.3.2
<SIP/seminario-0000000b>AGI Tx >> agi_callerid: 301
<SIP/seminario-0000000b>AGI Tx >> agi_calleridname: VoIP
<SIP/seminario-0000000b>AGI Tx >> agi_callingpres: 0
<SIP/seminario-0000000b>AGI Tx >> agi_callingani2: 0
<SIP/seminario-0000000b>AGI Tx >> agi_callington: 0
<SIP/seminario-0000000b>AGI Tx >> agi_callingtns: 0
<SIP/seminario-0000000b>AGI Tx >> agi_dnid: 888
<SIP/seminario-0000000b>AGI Tx >> agi_rdnis: unknown
<SIP/seminario-0000000b>AGI Tx >> agi_context: FechaRegistros
<SIP/seminario-0000000b>AGI Tx >> agi_extension: 2
<SIP/seminario-0000000b>AGI Tx >> agi_priority: 1
<SIP/seminario-0000000b>AGI Tx >> agi_enhanced: 0.0
<SIP/seminario-0000000b>AGI Tx >> agi_accountcode:
<SIP/seminario-0000000b>AGI Tx >> agi_threadid: -1219855472
<SIP/seminario-0000000b>AGI Tx >> agi_arg_1: 200703072
<SIP/seminario-0000000b>AGI Tx >> agi_arg_2: 2
<SIP/seminario-0000000b>AGI Tx >>

```

Figura 5.10.- Respuesta del sistema: consulta todas las fechas de registro del siguiente término

Tabla XVII.- Consulta de Fechas de Registro: Salida

| Ingreso | Salida |
|--------------------------|---|
| | Se ejecuta la aplicación getData(Escuchar_con_again,5000,1), con lo que se pide al usuario que ingrese 1 para volver a escuchar la consulta y 0 para regresar al menú principal |
| Usuario Ingresa 0 | Se ejecuta la aplicación Goto(MenuPrincipal,start,1). Se termina el script y se regresa al plan de marcado |

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
<SIP/seminario-0000000b>AGI Rx << ANSWER
<SIP/seminario-0000000b>AGI Tx >> 200 result=0
<SIP/seminario-0000000b>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_b28a9ebac3aa681d39bb43ab8dc2b207 "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_b28a9ebac3aa681d39bb43ab8dc2b207' (escape_digits=) (sample_offset 0)
<SIP/seminario-0000000b>AGI Tx >> 200 result=0 endpos=23453
<SIP/seminario-0000000b>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_b26ea207dcf6d391b4f7ccb353b080d6 "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_b26ea207dcf6d391b4f7ccb353b080d6' (escape_digits=) (sample_offset 0)
<SIP/seminario-0000000b>AGI Tx >> 200 result=0 endpos=32162
<SIP/seminario-0000000b>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_545ea628bfe9b2e52bc55d7e8f43b895 "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_545ea628bfe9b2e52bc55d7e8f43b895' (escape_digits=) (sample_offset 0)
<SIP/seminario-0000000b>AGI Tx >> 200 result=0 endpos=32642
<SIP/seminario-0000000b>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_748ee7be78f65db8d79c709e716316cc "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_748ee7be78f65db8d79c709e716316cc' (escape_digits=) (sample_offset 0)
<SIP/seminario-0000000b>AGI Tx >> 200 result=0 endpos=32882
<SIP/seminario-0000000b>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_fc13ecaaf69f0dad3e507c5404c3b040 "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_fc13ecaaf69f0dad3e507c5404c3b040' (escape_digits=) (sample_offset 0)
<SIP/seminario-0000000b>AGI Tx >> 200 result=0 endpos=33611
<SIP/seminario-0000000b>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_292188978ece416204264627e0f8dfb4 "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_292188978ece416204264627e0f8dfb4' (escape_digits=) (sample_offset 0)
<SIP/seminario-0000000b>AGI Tx >> 200 result=0 endpos=32642
<SIP/seminario-0000000b>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_e3f1c33e2ec82c6e1bf1ea03c8894647 "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_e3f1c33e2ec82c6e1bf1ea03c8894647' (escape_digits=) (sample_offset 0)
<SIP/seminario-0000000b>AGI Tx >> 200 result=0 endpos=33518
<SIP/seminario-0000000b>AGI Rx << GET DATA Escuchar_Con_again 5000 1
-- <SIP/seminario-0000000b> Playing 'Escuchar_Con_again.slin' (language 'es')
<SIP/seminario-0000000b>AGI Tx >> 200 result= (timeout)
<SIP/seminario-0000000b>AGI Rx << HANGUP
<SIP/seminario-0000000b>AGI Tx >> 200 result=1
-- <SIP/seminario-0000000b>AGI Script consultarRegistro.php completed, returning -1
localhost*CLI>

```

Figura 5.11.- Respuesta del sistema: salida de consulta todas las fechas de registro del siguiente término

6. Consultar fechas de exámenes, la llamada está dentro del contexto MenuPrincipal, aquí se reproduce la grabación Opciones_Menu_Principal, con las opciones de consulta del menú. El usuario ingresa la opción 3 con lo que se realiza la consulta del horario de exámenes del estudiante.

Tabla XVIII.- Consulta Fechas de Exámenes: Menú Principal

| Ingreso | Salida |
|--------------------------|--|
| | Se ejecuta la aplicación BackGround(Opciones_Menu_Principal) , con las opciones de la consulta |
| Usuario ingresa 3 | Se ejecuta aplicación Goto(HorarioExámenes,start,1) . La llamada es redirigida al contexto HorarioExámenes |
| | Se ejecuta la aplicación Background(Menu_Con_Fecha_Exámenes) . Se reproduce la grabación Menu_Con_Fecha_Exámenes (1 exámenes parciales, 2 exámenes finales, 3 mejoramiento) |

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
-- Executing [start@MenuPrincipal:1] Background("SIP/seminario-00000010", "Opciones_Menu_Principal") in new stack
-- <SIP/seminario-00000010> Playing 'Opciones_Menu_Principal.slin' (language 'es')
-- Executing [3@MenuPrincipal:1] Goto("SIP/seminario-00000010", "HorarioExámenes,start,1") in new stack
-- Goto (HorarioExámenes,start,1)
-- Executing [start@HorarioExámenes:1] Background("SIP/seminario-00000010", "Menu_Con_Fecha_Exámenes") in new stack
-- <SIP/seminario-00000010> Playing 'Menu_Con_Fecha_Exámenes.slin' (language 'es')

```

Figura 5.12.- Respuesta del sistema: Ingreso a Consulta de Fechas de Exámenes

Dentro de la consulta de fechas de exámenes brinda tres opciones: opción 1 para consulta de las fechas de exámenes parciales, como se muestra en la Figura 5.13 y 5.14; opción 2 para la consulta de las fechas de exámenes finales, como se muestra en la Figura 5.15 y 5.16; y, opción 3 para consulta de las fechas de exámenes de mejoramiento, como se muestra en la Figura 5.17 y 5.18.

Tabla XIX.- Consultar de Fechas de Exámenes Parciales

| Ingreso | Salida |
|----------------------|--|
| Usuario ingresa 1 | Se ejecuta la aplicación AGI(HorExámenes.php,\${matricula},\${exten}) . Con lo que se pasa el control de la llamada al script HorExámenes.php y se consulta la fecha de los exámenes parciales del estudiante |

```

-- Executing [start@HorarioExámenes:1] Background("SIP/seminario-00000010", "Menu_Con_Fecha_Exámenes") in new stack
-- <SIP/seminario-00000010> Playing 'Menu_Con_Fecha_Exámenes.slin' (language 'es')
-- Executing [1@HorarioExámenes:1] AGI("SIP/seminario-00000010", "HorExámenes.php,200612745,1") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/HorExámenes.php
<SIP/seminario-00000010>AGI Tx >> agi_request: HorExámenes.php
<SIP/seminario-00000010>AGI Tx >> agi_channel: SIP/seminario-00000010
<SIP/seminario-00000010>AGI Tx >> agi_language: es
<SIP/seminario-00000010>AGI Tx >> agi_type: SIP
<SIP/seminario-00000010>AGI Tx >> agi_uniqueid: 1321831957.16
<SIP/seminario-00000010>AGI Tx >> agi_version: 1.8.3.2
<SIP/seminario-00000010>AGI Tx >> agi_callerid: 301
<SIP/seminario-00000010>AGI Tx >> agi_calleridname: VoIP
<SIP/seminario-00000010>AGI Tx >> agi_callingpres: 0
<SIP/seminario-00000010>AGI Tx >> agi_callingani2: 0
<SIP/seminario-00000010>AGI Tx >> agi_callington: 0
<SIP/seminario-00000010>AGI Tx >> agi_callingtms: 0
<SIP/seminario-00000010>AGI Tx >> agi_dnid: 888
<SIP/seminario-00000010>AGI Tx >> agi_rdnis: unknown
<SIP/seminario-00000010>AGI Tx >> agi_context: HorarioExámenes
<SIP/seminario-00000010>AGI Tx >> agi_extension: 1
<SIP/seminario-00000010>AGI Tx >> agi_priority: 1
<SIP/seminario-00000010>AGI Tx >> agi_enhanced: 0.0
<SIP/seminario-00000010>AGI Tx >> agi_accountcode:
<SIP/seminario-00000010>AGI Tx >> agi_threadid: -1219855472
<SIP/seminario-00000010>AGI Tx >> agi_arg_1: 200612745
<SIP/seminario-00000010>AGI Tx >> agi_arg_2: 1
<SIP/seminario-00000010>AGI Tx >>
    
```

Figura 5.13.- Respuesta del sistema: consulta de Fechas de Exámenes Parciales

Tabla XX.- Consulta de Fechas de Exámenes Parciales: Salida

| Ingreso | Salida |
|---------|--|
| | Se realiza una consulta a la base asterisk por la fecha de registro del estudiante. Se produce una salida de audio que dice los exámenes |

| | |
|-------------------|---|
| | parciales que tiene el estudiante indicando la fecha aula y hora de los mismos con la aplicación text2wav |
| | Se ejecuta la aplicación getData(Escuchar_con_again,5000,1), con lo que se pide al usuario que ingrese 1 para volver a escuchar la consulta y 0 para regresar al menú principal |
| Usuario Ingresa 0 | Se ejecuta la aplicación Goto(MenuPrincipal,start,1). Se termina el script y se regresa al plan de marcado |

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
<SIP/seminario-00000010>AGI Rx << ANSWER
<SIP/seminario-00000010>AGI Tx >> 200 result=0
<SIP/seminario-00000010>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_09f8352b3d81e00cda5549740bcd0c88 "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_09f8352b3d81e00cda5549740bcd0c88' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000010>AGI Tx >> 200 result=0 endpos=79438
<SIP/seminario-00000010>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_5874d19af376bf22e66c6d8f46f16355 "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_5874d19af376bf22e66c6d8f46f16355' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000010>AGI Tx >> 200 result=0 endpos=74238
<SIP/seminario-00000010>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_20c020b54f5dd05e58f48d58e7d5bd3f "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_20c020b54f5dd05e58f48d58e7d5bd3f' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000010>AGI Tx >> 200 result=0 endpos=71438
<SIP/seminario-00000010>AGI Rx << GET DATA Escuchar_Con_again 3000 1
-- <SIP/seminario-00000010> Playing 'Escuchar_Con_again.slin' (language 'es')
<SIP/seminario-00000010>AGI Tx >> 200 result=0
<SIP/seminario-00000010>AGI Rx << EXEC Goto MenuPrincipal,start,1
-- AGI Script Executing Application: (Goto) Options: (MenuPrincipal,start,1)
-- Goto (MenuPrincipal,start,1)
<SIP/seminario-00000010>AGI Tx >> 200 result=0
-- <SIP/seminario-00000010>AGI Script HorExamenes.php completed, returning 0
    
```

Figura 5.14.- Respuesta del sistema: salida de consulta de Fechas de Exámenes Parciales

Tabla XXI.- Consultar de Fechas de Exámenes Finales

| Ingreso | Salida |
|-------------------|--|
| Usuario ingresa 2 | Se ejecuta la aplicación AGI(HorExamenes.php,\${matricula},\${exten}) . Con lo que se pasa el control de la llamada al script HorExamenes.php y se consulta la fecha de los exámenes parciales del estudiante |

```

-- Executing [start@HorarioExámenes:1] Background("SIP/seminario-00000010", "Menu_Con_Fecha_Exámenes") in new stack
-- <SIP/seminario-00000010> Playing 'Menu_Con_Fecha_Exámenes.slin' (language 'es')
-- Executing [2@HorarioExámenes:1] AGI("SIP/seminario-00000010", "HorExámenes.php,200612745,2") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/HorExámenes.php
<SIP/seminario-00000010>AGI Tx >> agi_request: HorExámenes.php
<SIP/seminario-00000010>AGI Tx >> agi_channel: SIP/seminario-00000010
<SIP/seminario-00000010>AGI Tx >> agi_language: es
<SIP/seminario-00000010>AGI Tx >> agi_type: SIP
<SIP/seminario-00000010>AGI Tx >> agi_uniqueid: 1321831957.16
<SIP/seminario-00000010>AGI Tx >> agi_version: 1.8.3.2
<SIP/seminario-00000010>AGI Tx >> agi_callerid: 301
<SIP/seminario-00000010>AGI Tx >> agi_calleridname: VoIP
<SIP/seminario-00000010>AGI Tx >> agi_callingpres: 0
<SIP/seminario-00000010>AGI Tx >> agi_callingani2: 0
<SIP/seminario-00000010>AGI Tx >> agi_callington: 0
<SIP/seminario-00000010>AGI Tx >> agi_callingtms: 0
<SIP/seminario-00000010>AGI Tx >> agi_dnid: 888
<SIP/seminario-00000010>AGI Tx >> agi_rdnis: unknown
<SIP/seminario-00000010>AGI Tx >> agi_context: HorarioExámenes
<SIP/seminario-00000010>AGI Tx >> agi_extension: 2
<SIP/seminario-00000010>AGI Tx >> agi_priority: 1
<SIP/seminario-00000010>AGI Tx >> agi_enhanced: 0.0
<SIP/seminario-00000010>AGI Tx >> agi_accountcode:
<SIP/seminario-00000010>AGI Tx >> agi_threadid: -1219855472
<SIP/seminario-00000010>AGI Tx >> agi_arg_1: 200612745
<SIP/seminario-00000010>AGI Tx >> agi_arg_2: 2
<SIP/seminario-00000010>AGI Tx >>

```

Figura 5.15.- Respuesta del sistema: consulta de Fechas de Exámenes Finales

Tabla XXII.- Consulta de Fechas de Exámenes Finales: Salida

| Ingreso | Salida |
|-----------------------------|---|
| | Se realiza una consulta a la base asterisk por los exámenes finales del estudiante. Se produce una salida de audio que dice los exámenes finales que tiene el estudiante indicando la fecha, aula y hora de los mismos con la aplicación text2wav |
| | Se ejecuta la aplicación getData(Escuchar_con_again,5000,1), con lo que se pide al usuario que ingrese 1 para volver a escuchar la consulta y 0 para regresar al menú principal |
| Usuario Ingresa 0 | Se ejecuta la aplicación Goto(MenuPrincipal,start,1). Se termina el script y se regresa al plan de marcado |

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
<SIP/seminario-00000010>AGI Tx >> 200 result=0
<SIP/seminario-00000010>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_19178af0f63618397c38a2b5c83fa8fa "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_19178af0f63618397c38a2b5c83fa8fa' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000010>AGI Tx >> 200 result=0 endpos=80318
<SIP/seminario-00000010>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_19175e5a82c249d8e7194c7603e1b798 "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_19175e5a82c249d8e7194c7603e1b798' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000010>AGI Tx >> 200 result=0 endpos=73678
<SIP/seminario-00000010>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_4622f42cc8edb991c3ccf0b80f91250b "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_4622f42cc8edb991c3ccf0b80f91250b' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000010>AGI Tx >> 200 result=0 endpos=73918
<SIP/seminario-00000010>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_7b61b9107e8fa1469316b3efd6fb8568 "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_7b61b9107e8fa1469316b3efd6fb8568' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000010>AGI Tx >> 200 result=0 endpos=68238
<SIP/seminario-00000010>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav_1b2bad2f26626dc58acf6a3dc2db941a "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_1b2bad2f26626dc58acf6a3dc2db941a' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000010>AGI Tx >> 200 result=0 endpos=71118
<SIP/seminario-00000010>AGI Rx << GET DATA Escuchar_Con_again 3000 1
-- <SIP/seminario-00000010> Playing 'Escuchar_Con_again.slin' (language 'es')
<SIP/seminario-00000010>AGI Tx >> 200 result=2
<SIP/seminario-00000010>AGI Rx << EXEC Playback opcion_invalida
-- AGI Script Executing Application: (Playback) Options: (opcion_invalida)
-- <SIP/seminario-00000010> Playing 'opcion_invalida.slin' (language 'es')
<SIP/seminario-00000010>AGI Tx >> 200 result=0
    
```

Figura 5.16.- Respuesta del sistema: salida de consulta de Fechas de Exámenes Finales

Tabla XXIII.- Consultar de Fechas de Exámenes Mejoramiento

| Ingreso | Salida |
|----------------------|---|
| Usuario ingresa 3 | Se ejecuta la aplicación AGI(HorExámenes.php,\${matricula},\${exten}) . Con lo que se pasa el control de la llamada al script HorExámenes.php y se consulta la fecha de los exámenes de mejoramiento del estudiante |

```

-- Executing [start@HorarioExamenes:1] Background("SIP/seminario-00000011", "Menu_Con_Fecha_Examenes") in new stack
-- <SIP/seminario-00000011> Playing 'Menu_Con_Fecha_Examenes.slin' (language 'es')
-- Executing [3@HorarioExamenes:1] AGI("SIP/seminario-00000011", "HorExamenes.php,200612745,3") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/HorExamenes.php
<SIP/seminario-00000011>AGI Tx >> agi_request: HorExamenes.php
<SIP/seminario-00000011>AGI Tx >> agi_channel: SIP/seminario-00000011
<SIP/seminario-00000011>AGI Tx >> agi_language: es
<SIP/seminario-00000011>AGI Tx >> agi_type: SIP
<SIP/seminario-00000011>AGI Tx >> agi_uniqueid: 1321832896.17
<SIP/seminario-00000011>AGI Tx >> agi_version: 1.8.3.2
<SIP/seminario-00000011>AGI Tx >> agi_callerid: 301
<SIP/seminario-00000011>AGI Tx >> agi_calleridname: VoIP
<SIP/seminario-00000011>AGI Tx >> agi_callingpres: 0
<SIP/seminario-00000011>AGI Tx >> agi_callingani2: 0
<SIP/seminario-00000011>AGI Tx >> agi_callington: 0
<SIP/seminario-00000011>AGI Tx >> agi_callingtms: 0
<SIP/seminario-00000011>AGI Tx >> agi_dnid: 888
<SIP/seminario-00000011>AGI Tx >> agi_rdnis: unknown
<SIP/seminario-00000011>AGI Tx >> agi_context: HorarioExamenes
<SIP/seminario-00000011>AGI Tx >> agi_extension: 3
<SIP/seminario-00000011>AGI Tx >> agi_priority: 1
<SIP/seminario-00000011>AGI Tx >> agi_enhanced: 0.0
<SIP/seminario-00000011>AGI Tx >> agi_accountcode:
<SIP/seminario-00000011>AGI Tx >> agi_threadid: -1219855472
<SIP/seminario-00000011>AGI Tx >> agi_arg_1: 200612745
<SIP/seminario-00000011>AGI Tx >> agi_arg_2: 3
<SIP/seminario-00000011>AGI Tx >>

```

Figura 5.17.- Respuesta del sistema: consulta de Fechas de Exámenes Mejoramiento

Tabla XXIV.- Consulta de Fechas de Exámenes Mejoramiento: Salida

| Ingreso | Salida |
|----------------------|---|
| | Se realiza una consulta a la base asterisk por los exámenes de mejoramiento del estudiante. Se produce una salida de audio que dice los exámenes de mejoramiento que tiene el estudiante indicando la fecha, aula y hora de los mismos con la aplicación text2wav |
| | Se ejecuta la aplicación getData(Escuchar_con_again,5000,1), con lo que se pide al usuario que ingrese 1 para volver a escuchar la consulta y 0 para regresar al menú principal |
| Usuario Ingresa 1 | Se vuelve a escuchar la información de la consulta anterior |
| | Se ejecuta la aplicación getData(Escuchar_con_again,5000,1), con lo |

| | |
|-------------------|---|
| | que se pide al usuario que ingrese 1 para volver a escuchar la consulta y 0 para regresar al menú principal |
| Usuario Ingresa 0 | Se ejecuta la aplicación Goto(MenuPrincipal,start,1). Se termina el script y se regresa al plan de marcado |

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
<SIP/seminario-00000011>AGI Rx << ANSWER
<SIP/seminario-00000011>AGI Tx >> 200 result=0
<SIP/seminario-00000011>AGI Rx << STREAM FILE /var/spool/asterisk//tmp/text2wav_9bcca6472cccdcc0f35039ac65828e0 "" 0
-- Playing '/var/spool/asterisk//tmp/text2wav_9bcca6472cccdcc0f35039ac65828e0' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000011>AGI Tx >> 200 result=0 endpos=70398
<SIP/seminario-00000011>AGI Rx << STREAM FILE /var/spool/asterisk//tmp/text2wav_4c89b0927ee812ee57bc220ee41a93fd "" 0
-- Playing '/var/spool/asterisk//tmp/text2wav_4c89b0927ee812ee57bc220ee41a93fd' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000011>AGI Tx >> 200 result=0 endpos=73198
<SIP/seminario-00000011>AGI Rx << STREAM FILE /var/spool/asterisk//tmp/text2wav_76888172ada5b6cbe358ea35822179c3 "" 0
-- Playing '/var/spool/asterisk//tmp/text2wav_76888172ada5b6cbe358ea35822179c3' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000011>AGI Tx >> 200 result=0 endpos=81438
<SIP/seminario-00000011>AGI Rx << GET DATA Escuchar_Con_again 3000 1
-- <SIP/seminario-00000011> Playing 'Escuchar_Con_again.slin' (language 'es')
<SIP/seminario-00000011>AGI Tx >> 200 result=1
<SIP/seminario-00000011>AGI Rx << STREAM FILE /var/spool/asterisk//tmp/text2wav_9bcca6472cccdcc0f35039ac65828e0 "" 0
-- Playing '/var/spool/asterisk//tmp/text2wav_9bcca6472cccdcc0f35039ac65828e0' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000011>AGI Tx >> 200 result=0 endpos=70398
<SIP/seminario-00000011>AGI Rx << STREAM FILE /var/spool/asterisk//tmp/text2wav_4c89b0927ee812ee57bc220ee41a93fd "" 0
-- Playing '/var/spool/asterisk//tmp/text2wav_4c89b0927ee812ee57bc220ee41a93fd' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000011>AGI Tx >> 200 result=0 endpos=73198
<SIP/seminario-00000011>AGI Rx << STREAM FILE /var/spool/asterisk//tmp/text2wav_76888172ada5b6cbe358ea35822179c3 "" 0
-- Playing '/var/spool/asterisk//tmp/text2wav_76888172ada5b6cbe358ea35822179c3' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000011>AGI Tx >> 200 result=0 endpos=81438
<SIP/seminario-00000011>AGI Rx << GET DATA Escuchar_Con_again 3000 1
-- <SIP/seminario-00000011> Playing 'Escuchar_Con_again.slin' (language 'es')
<SIP/seminario-00000011>AGI Tx >> 200 result=0
<SIP/seminario-00000011>AGI Rx << EXEC Goto MenuPrincipal,start,1
-- AGI Script Executing Application: (Goto) Options: (MenuPrincipal,start,1)
-- Goto (MenuPrincipal,start,1)
<SIP/seminario-00000011>AGI Tx >> 200 result=0
-- <SIP/seminario-00000011>AGI Script HorExamenes.php completed, returning 0

```

Figura 5.18.- Respuesta del sistema: salida de consulta de Fechas de Exámenes Mejoramiento

- Consultar horario de clases por día, la llamada está dentro del contexto MenuPrincipal, se reproduce la grabación Opciones_Menu_Principal, con las opciones de consulta del menú. El usuario ingresa la opción 4 y se realiza la consulta del horario de clases del estudiante.

Tabla XXV.- Consulta de Horario por Materia: Menú Principal

| Ingreso | Salida |
|--------------------------------|---|
| | Se ejecuta la aplicación BackGround(Opciones_Menu_Principal) , con las opciones de la consulta |
| Usuario ingresa 4 | Se ejecuta aplicación Goto(HorarioClases,start,1) . La llamada es redirigida al contexto HorarioClases |
| | Se ejecuta la aplicación Background(HorarioClases) . Se reproduce la grabación HorarioClases (1 horario de clases día lunes, 2 horario de clases día martes, 3 horario de clases día miércoles, 4 horario de clases día jueves, 5 horario de clases día viernes, 6 horario de clases día sábado). |

```

root@localhost:~
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
-- Executing [start@MenuPrincipal:1] Background("SIP/seminario-00000015", "Opciones_Menu_Principal") in new stack
k
-- <SIP/seminario-00000015> Playing 'Opciones_Menu_Principal.slin' (language 'es')
-- Executing [start@MenuPrincipal:2] WaitExten("SIP/seminario-00000015", "10") in new stack
== CDR updated on SIP/seminario-00000015
-- Executing [4@MenuPrincipal:1] Goto("SIP/seminario-00000015", "HorarioClases,start,1") in new stack
-- Goto (HorarioClases,start,1)
-- Executing [start@HorarioClases:1] Background("SIP/seminario-00000015", "HorarioClases") in new stack
-- <SIP/seminario-00000015> Playing 'HorarioClases.slin' (language 'es')

```

Figura 5.19.- Respuesta del sistema: Ingreso a Consulta de Horario por Materia

Dentro de la consulta de horario de clases por día existen seis opciones: opción 1 para consultar el horario de clases del día lunes, como se muestra en la Figura 5.20 y 5.21; opción 2 para

consultar el horario de clases del día martes, como se muestra en la Figura 5.22 y 5.23; opción 3 para consultar del horario de clases del día miércoles , como se muestra en la Figura 5.24 y 5.25; opción 4 para consultar el horario de clases del día jueves , como se muestra en la Figura 5.26 y 5.27; opción 5 para consultar el horario de clases del día viernes , como se muestra en la Figura 5.28 y 5.29; y, opción 6 para consulta del horario de clases del día sábado, como se muestra en la Figura 5.30 y 5.31.

Tabla XXVI.- Consulta de Horario por Materia del Día Lunes

| Ingreso | Salida |
|----------------------|--|
| Usuario ingresa 1 | Se ejecuta la aplicación AGI(HorClases.php,{matricula},{exten}) . Con lo que se pasa el control de la llamada al script HorClases.php y se consulta las clases que el estudiante tiene para el día lunes |

```

-- Executing [start@HorarioClases:1] Background("SIP/seminario-00000015", "HorarioClases") in new stack
-- <SIP/seminario-00000015> Playing 'HorarioClases.clin' (language 'es')
-- Executing [1@HorarioClases:1] AGI("SIP/seminario-00000015", "HorClases.php,200703072,1") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/HorClases.php
<SIP/seminario-00000015>AGI Tx >> agi_request: HorClases.php
<SIP/seminario-00000015>AGI Tx >> agi_channel: SIP/seminario-00000015
<SIP/seminario-00000015>AGI Tx >> agi_language: es
<SIP/seminario-00000015>AGI Tx >> agi_type: SIP
<SIP/seminario-00000015>AGI Tx >> agi_uniqueid: 1321835380.21
<SIP/seminario-00000015>AGI Tx >> agi_version: 1.8.3.2
<SIP/seminario-00000015>AGI Tx >> agi_callerid: 301
<SIP/seminario-00000015>AGI Tx >> agi_calleridname: VoIP
<SIP/seminario-00000015>AGI Tx >> agi_callingpres: 0
<SIP/seminario-00000015>AGI Tx >> agi_callingani2: 0
<SIP/seminario-00000015>AGI Tx >> agi_callington: 0
<SIP/seminario-00000015>AGI Tx >> agi_callingtns: 0
<SIP/seminario-00000015>AGI Tx >> agi_dnid: 888
<SIP/seminario-00000015>AGI Tx >> agi_rdnis: unknown
<SIP/seminario-00000015>AGI Tx >> agi_context: HorarioClases
<SIP/seminario-00000015>AGI Tx >> agi_extension: 1
<SIP/seminario-00000015>AGI Tx >> agi_priority: 1
<SIP/seminario-00000015>AGI Tx >> agi_enhanced: 0.0
<SIP/seminario-00000015>AGI Tx >> agi_accountcode:
<SIP/seminario-00000015>AGI Tx >> agi_threadid: -1219855472
<SIP/seminario-00000015>AGI Tx >> agi_arg_1: 200703072

```

Figura 5.20.- Respuesta del sistema: consulta de Fechas de Horario por Materia del Día Lunes

Tabla XXVII.- Consulta de Horario por Materia del Día Lunes: Salida

| Ingreso | Salida |
|---------|--|
| | Se realiza una consulta a la base Asterisk para obtener las materias que el estudiante tiene el día lunes. Se produce una salida de audio que dice, ordenada por hora , una a una las materias que el estudiante tiene el día lunes con su respectiva aula, con la aplicación text2wav |
| | Se ejecuta la aplicación getData(Escuchar_con_again,5000,1), con lo que se pide al usuario que ingrese 1 para volver a escuchar la consulta y 0 para regresar al menú principal |
| Usuario | Se ejecuta la aplicación Goto(MenuPrincipal,start,1). Se termina el |

Ingresas **0** script y se regresa al plan de marcado

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
<SIP/seminario-00000015>AGI Tx >> agi_arg_2: 1
<SIP/seminario-00000015>AGI Tx >>
<SIP/seminario-00000015>AGI Rx << ANSWER
<SIP/seminario-00000015>AGI Tx >> 200 result=0
<SIP/seminario-00000015>AGI Rx << STREAM FILE /var/spool/asterisk//tmp//text2wav_fe33d0a6f399210d221cef674bf61ddc "
0
-- Playing '/var/spool/asterisk//tmp//text2wav_fe33d0a6f399210d221cef674bf61ddc' (escape digits=) (sample offset
0)
<SIP/seminario-00000015>AGI Tx >> 200 result=0 endpos=65278
<SIP/seminario-00000015>AGI Rx << STREAM FILE /var/spool/asterisk//tmp//text2wav_ae6cd8b2f5eab1bc14c9c482f6b01232 "
0
-- Playing '/var/spool/asterisk//tmp//text2wav_ae6cd8b2f5eab1bc14c9c482f6b01232' (escape digits=) (sample offset
0)
<SIP/seminario-00000015>AGI Tx >> 200 result=0 endpos=60958
<SIP/seminario-00000015>AGI Rx << STREAM FILE /var/spool/asterisk//tmp//text2wav_653e213055e0176e0959aefabc74bf4b "
0
-- Playing '/var/spool/asterisk//tmp//text2wav_653e213055e0176e0959aefabc74bf4b' (escape digits=) (sample offset
0)
<SIP/seminario-00000015>AGI Tx >> 200 result=0 endpos=65438
<SIP/seminario-00000015>AGI Rx << GET DATA Escuchar_Con_again 3000 1
-- <SIP/seminario-00000015> Playing 'Escuchar_Con_again.slin' (language 'es')
<SIP/seminario-00000015>AGI Tx >> 200 result=0
<SIP/seminario-00000015>AGI Rx << EXEC Goto MenuPrincipal,start,1
-- AGI Script Executing Application: (Goto) Options: (MenuPrincipal,start,1)
-- Goto (MenuPrincipal,start,1)
<SIP/seminario-00000015>AGI Tx >> 200 result=0
-- <SIP/seminario-00000015>AGI Script HorClases.php completed, returning 0
    
```

Figura 5.21.- Respuesta del sistema: salida de Consulta de Horario por Materia del Día Lunes

Tabla XXVIII.- Consulta de Horario por Materia del Día Martes

| Ingreso | Salida |
|-----------------------------|--|
| Usuario ingresa 2 | Se ejecuta la aplicación AGI(HorClases.php,\${matricula},\${exten}) . Con lo que se pasa el control de la llamada al script HorClases.php y se consulta las clases que el estudiante tiene para el día martes |

```

-- <SIP/seminario-00000014> Playing 'HorarioClases.slin' (language 'es')
-- Executing [2@HorarioClases:1] AGI("SIP/seminario-00000014", "HorClases.ph
p,200703072,2") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/HorarioClases.php
<SIP/seminario-00000014>AGI Tx >> agi_request: HorClases.php
<SIP/seminario-00000014>AGI Tx >> agi_channel: SIP/seminario-00000014
<SIP/seminario-00000014>AGI Tx >> agi_language: es
<SIP/seminario-00000014>AGI Tx >> agi_type: SIP
<SIP/seminario-00000014>AGI Tx >> agi_uniqueid: 1321834932.20
<SIP/seminario-00000014>AGI Tx >> agi_version: 1.8.3.2
<SIP/seminario-00000014>AGI Tx >> agi_callerid: 301
<SIP/seminario-00000014>AGI Tx >> agi_calleridname: VoIP
<SIP/seminario-00000014>AGI Tx >> agi_callingpres: 0
<SIP/seminario-00000014>AGI Tx >> agi_callingani2: 0
<SIP/seminario-00000014>AGI Tx >> agi_callington: 0
<SIP/seminario-00000014>AGI Tx >> agi_callingtns: 0
<SIP/seminario-00000014>AGI Tx >> agi_dnid: 888
<SIP/seminario-00000014>AGI Tx >> agi_rdnis: unknown
<SIP/seminario-00000014>AGI Tx >> agi_context: HorarioClases
<SIP/seminario-00000014>AGI Tx >> agi_extension: 2
<SIP/seminario-00000014>AGI Tx >> agi_priority: 1
<SIP/seminario-00000014>AGI Tx >> agi_enhanced: 0.0
<SIP/seminario-00000014>AGI Tx >> agi_accountcode:
<SIP/seminario-00000014>AGI Tx >> agi_threadid: -1219855472
<SIP/seminario-00000014>AGI Tx >> agi_arg_1: 200703072
<SIP/seminario-00000014>AGI Tx >> agi_arg_2: 2
<SIP/seminario-00000014>AGI Tx >>

```

Figura 5.22.- Respuesta del sistema: consulta de Fechas de Horario por Materia del Día Martes

Tabla XXIX.- Consulta de Horario por Materia del Día Martes: Salida

| Ingreso | Salida |
|-------------------|--|
| | Se realiza una consulta a la base Asterisk para obtener las materias que el estudiante tiene el día martes. Se produce una salida de audio que dice, ordenada por hora , una a una las materias que el estudiante tiene el día martes con su respectiva aula, con la aplicación text2wav |
| | Se ejecuta la aplicación getData(Escuchar_con_again,5000,1), con lo que se pide al usuario que ingrese 1 para volver a escuchar la consulta y 0 para regresar al menú principal |
| Usuario Ingresa 0 | Se ejecuta la aplicación Goto(MenuPrincipal,start,1). Se termina el script y se regresa al plan de marcado |

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
<SIP/seminario-00000014>AGI Rx << ANSWER
<SIP/seminario-00000014>AGI Tx >> 200 result=0
<SIP/seminario-00000014>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav
_bfd71305e843d59df565a5a1acdcbc52 "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_bfd71305e843d59df565a5a1acdcbc
c52' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000014>AGI Tx >> 200 result=0 endpos=56718
<SIP/seminario-00000014>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav
_aef1a70cf6982bcb10107f2178227b38 "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_aef1a70cf6982bcb10107f2178227
b38' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000014>AGI Tx >> 200 result=0 endpos=63918
<SIP/seminario-00000014>AGI Rx << GET DATA Escuchar_Con_again 3000 1
-- <SIP/seminario-00000014> Playing 'Escuchar_Con_again.slin' (language 'es'
)
<SIP/seminario-00000014>AGI Tx >> 200 result=0
<SIP/seminario-00000014>AGI Rx << EXEC Goto MenuPrincipal,start,1
-- AGI Script Executing Application: (Goto) Options: (MenuPrincipal,start,1)
-- Goto (MenuPrincipal,start,1)
<SIP/seminario-00000014>AGI Tx >> 200 result=0
-- <SIP/seminario-00000014>AGI Script HorClases.php completed, returning 0
    
```

Figura 5.23.- Respuesta del sistema: salida de Consulta de Horario por Materia del Día Martes

Tabla XXX.- Consulta de Horario por Materia del Día Miércoles

| Ingreso | Salida |
|----------------------|--|
| Usuario ingresa 3 | Se ejecuta la aplicación AGI(HorClases.php,{matricula},{exten}) . Con lo que se pasa el control de la llamada al script HorClases.php y se consulta las clases que el estudiante tiene para el día miércoles |

```

-- <SIP/seminario-0000014> Playing 'HorarioClases_slin' (language 'es')
-- Executing [3@HorarioClases:1] AGI("SIP/seminario-0000014", "HorClases.ph
,200703072,3") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/HorClases.php
<SIP/seminario-0000014>AGI Tx >> agi_request: HorClases.php
<SIP/seminario-0000014>AGI Tx >> agi_channel: SIP/seminario-0000014
<SIP/seminario-0000014>AGI Tx >> agi_language: es
<SIP/seminario-0000014>AGI Tx >> agi_type: SIP
<SIP/seminario-0000014>AGI Tx >> agi_uniqueid: 1321834932.20
<SIP/seminario-0000014>AGI Tx >> agi_version: 1.8.3.2
<SIP/seminario-0000014>AGI Tx >> agi_callerid: 301
<SIP/seminario-0000014>AGI Tx >> agi_calleridname: VoIP
<SIP/seminario-0000014>AGI Tx >> agi_callingpres: 0
<SIP/seminario-0000014>AGI Tx >> agi_callingani2: 0
<SIP/seminario-0000014>AGI Tx >> agi_callington: 0
<SIP/seminario-0000014>AGI Tx >> agi_callingtms: 0
<SIP/seminario-0000014>AGI Tx >> agi_dnid: 888
<SIP/seminario-0000014>AGI Tx >> agi_rdnis: unknown
<SIP/seminario-0000014>AGI Tx >> agi_context: HorarioClases
<SIP/seminario-0000014>AGI Tx >> agi_extension: 3
<SIP/seminario-0000014>AGI Tx >> agi_priority: 1
<SIP/seminario-0000014>AGI Tx >> agi_enhanced: 0.0
<SIP/seminario-0000014>AGI Tx >> agi_accountcode:
<SIP/seminario-0000014>AGI Tx >> agi_threadid: -1219855472
<SIP/seminario-0000014>AGI Tx >> agi_arg_1: 200703072
<SIP/seminario-0000014>AGI Tx >> agi_arg_2: 3
<SIP/seminario-0000014>AGI Tx >>

```

Figura 5.24.- Respuesta del sistema: consulta de Fechas de Horario por Materia del Día Miércoles

Tabla XXXI.- Consulta de Horario por Materia del Día Miércoles: Salida

| Ingreso | Salida |
|------------------------------|--|
| | Se realiza una consulta a la base Asterisk para obtener las materias que el estudiante tiene el día miércoles. Se produce una salida de audio que dice, ordenada por hora , una a una las materias que el estudiante tiene el día miércoles con su respectiva aula, con la aplicación text2wav |
| | Se ejecuta la aplicación getData(Escuchar_con_again,5000,1), con lo que se pide al usuario que ingrese 1 para volver a escuchar la consulta y 0 para regresar al menú principal |
| Usuario Ingresas 0 | Se ejecuta la aplicación Goto(MenuPrincipal,start,1). Se termina el script y se regresa al plan de marcado |

```

Archivo Editar Ver Terminal Solapas Ayuda
<SIP/seminario-00000014>AGI Rx << ANSWER
<SIP/seminario-00000014>AGI Tx >> 200 result=0
<SIP/seminario-00000014>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav
_fe33d0a6f399210d221cef674bf61ddc "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_fe33d0a6f399210d221cef674bf61
ddc' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000014>AGI Tx >> 200 result=0 endpos=65278
<SIP/seminario-00000014>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav
_b32b9390b96c0912e895d613297c89e6 "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_b32b9390b96c0912e895d613297c8
9e6' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000014>AGI Tx >> 200 result=0 endpos=62878
<SIP/seminario-00000014>AGI Rx << STREAM FILE /var/spool/asterisk/tmp/text2wav
_653e213055e0176e0959aefabc74bf4b "" 0
-- Playing '/var/spool/asterisk/tmp/text2wav_653e213055e0176e0959aefabc74b
t4b' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000014>AGI Tx >> 200 result=0 endpos=65438
<SIP/seminario-00000014>AGI Rx << GET DATA Escuchar_Con_again 3000 1
-- <SIP/seminario-00000014> Playing 'Escuchar_Con_again.slin' (language 'es'
)
<SIP/seminario-00000014>AGI Tx >> 200 result=0
<SIP/seminario-00000014>AGI Rx << EXEC Goto MenuPrincipal,start,1
-- AGI Script Executing Application: (Goto) Options: (MenuPrincipal,start,1)
-- Goto (MenuPrincipal,start,1)
<SIP/seminario-00000014>AGI Tx >> 200 result=0
-- <SIP/seminario-00000014>AGI Script HorClases.php completed, returning 0
    
```

Figura 5.25.- Respuesta del sistema: salida de Consulta de Horario por Materia del Día Miércoles

Tabla XXXII.- Consulta de Horario por Materia del Día Jueves

| Ingreso | Salida |
|----------------------|---|
| Usuario ingresa 4 | Se ejecuta la aplicación AGI(HorClases.php,{matricula},{exten}) . Con lo que se pasa el control de la llamada al script HorClases.php y se consulta las clases que el estudiante tiene para el día jueves |

```

-- <SIP/seminario-00000014> Playing 'HorarioClases.slin' (language 'es')
-- Executing [4@HorarioClases:1] AGI("SIP/seminario-00000014", "HorClases.ph
p,200703072,4") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/HorClases.php
<SIP/seminario-00000014>AGI Tx >> agi_request: HorClases.php
<SIP/seminario-00000014>AGI Tx >> agi_channel: SIP/seminario-00000014
<SIP/seminario-00000014>AGI Tx >> agi_language: es
<SIP/seminario-00000014>AGI Tx >> agi_type: SIP
<SIP/seminario-00000014>AGI Tx >> agi_uniqueid: 1321834932.20
<SIP/seminario-00000014>AGI Tx >> agi_version: 1.8.3.2
<SIP/seminario-00000014>AGI Tx >> agi_callerid: 301
<SIP/seminario-00000014>AGI Tx >> agi_calleridname: VoIP
<SIP/seminario-00000014>AGI Tx >> agi_callingpres: 0
<SIP/seminario-00000014>AGI Tx >> agi_callingani2: 0
<SIP/seminario-00000014>AGI Tx >> agi_callington: 0
<SIP/seminario-00000014>AGI Tx >> agi_callingtms: 0
<SIP/seminario-00000014>AGI Tx >> agi_dnid: 888
<SIP/seminario-00000014>AGI Tx >> agi_rdnis: unknown
<SIP/seminario-00000014>AGI Tx >> agi_context: HorarioClases
<SIP/seminario-00000014>AGI Tx >> agi_extension: 4
<SIP/seminario-00000014>AGI Tx >> agi_priority: 1
<SIP/seminario-00000014>AGI Tx >> agi_enhanced: 0.0
<SIP/seminario-00000014>AGI Tx >> agi_accountcode:
<SIP/seminario-00000014>AGI Tx >> agi_threadid: -1219855472
<SIP/seminario-00000014>AGI Tx >> agi_arg_1: 200703072
<SIP/seminario-00000014>AGI Tx >> agi_arg_2: 4
<SIP/seminario-00000014>AGI Tx >>

```

Figura 5.26.- Respuesta del sistema: consulta de Fechas de Horario por Materia del Día Jueves

Tabla XXXIII.- Consulta de Horario por Materia del Día Jueves: Salida

| Ingreso | Salida |
|---------|--|
| | Se realiza una consulta a la base Asterisk para obtener las materias que el estudiante tiene el día jueves. Se produce una salida de audio que dice, ordenada por hora , una a una las materias que el estudiante tiene el día jueves con su respectiva aula, con la aplicación text2wav |
| | Se ejecuta la aplicación getData(Escuchar_con_again,5000,1), con lo que se pide al usuario que ingrese 1 para volver a escuchar la consulta y 0 para regresar al menú principal |
| | El usuario no ingresa nada dentro de los 5 segundos de espera que se le da para que ingrese una de las opciones anteriores, por lo que se |

cumple el **timeout** lo que provoca que se cierre el canal y se **cuelgue la llamada**

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
<SIP/seminario-00000014>AGI Rx << ANSWER
<SIP/seminario-00000014>AGI Tx >> 200 result=0
<SIP/seminario-00000014>AGI Rx << STREAM FILE /var/spool/asterisk//tmp//text2wav
aef1a70cf6982bcb10107f2178227b38 "" 0
-- Playing '/var/spool/asterisk//tmp//text2wav aef1a70cf6982bcb10107f2178227
b38' (escape_digits=) (sample_offset 0)
<SIP/seminario-00000014>AGI Tx >> 200 result=0 endpos=63918
<SIP/seminario-00000014>AGI Rx << GET DATA Escuchar_Con_again 3000 1
-- <SIP/seminario-00000014> Playing 'Escuchar_Con_again.slin' (language 'es'
)
<SIP/seminario-00000014>AGI Tx >> 200 result= (timeout)
<SIP/seminario-00000014>AGI Rx << HANGUP
<SIP/seminario-00000014>AGI Tx >> 200 result=1
-- <SIP/seminario-00000014>AGI Script HorClases.php completed, returning -1
localhost*CLI>
    
```

Figura 5.27.- Respuesta del sistema: salida de Consulta de Horario por Materia del Día Jueves

Tabla XXXIV.- Consulta de Horario por Materia del Día Viernes

| Ingreso | Salida |
|----------------------|--|
| Usuario ingresa 5 | Se ejecuta la aplicación AGI(HorClases.php,{matricula},{exten}) . Con lo que se pasa el control de la llamada al script HorClases.php y se consulta las clases que el estudiante tiene para el día viernes |

```

-- <SIP/seminario-00000015> Playing 'HorarioClases.slin' (language 'es')
-- Executing [5@HorarioClases:1] AGI("SIP/seminario-00000015", "HorClases.php,200703072,5") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/HorClases.php
<SIP/seminario-00000015>AGI Tx >> agi_request: HorClases.php
<SIP/seminario-00000015>AGI Tx >> agi_channel: SIP/seminario-00000015
<SIP/seminario-00000015>AGI Tx >> agi_language: es
<SIP/seminario-00000015>AGI Tx >> agi_type: SIP
<SIP/seminario-00000015>AGI Tx >> agi_uniqueid: 1321835380.21
<SIP/seminario-00000015>AGI Tx >> agi_version: 1.8.3.2
<SIP/seminario-00000015>AGI Tx >> agi_callerid: 301
<SIP/seminario-00000015>AGI Tx >> agi_calleridname: VoIP
<SIP/seminario-00000015>AGI Tx >> agi_callingpres: 0
<SIP/seminario-00000015>AGI Tx >> agi_callingani2: 0
<SIP/seminario-00000015>AGI Tx >> agi_callington: 0
<SIP/seminario-00000015>AGI Tx >> agi_callingtns: 0
<SIP/seminario-00000015>AGI Tx >> agi_dnid: 888
<SIP/seminario-00000015>AGI Tx >> agi_rdnis: unknown
<SIP/seminario-00000015>AGI Tx >> agi_context: HorarioClases
<SIP/seminario-00000015>AGI Tx >> agi_extension: 5
<SIP/seminario-00000015>AGI Tx >> agi_priority: 1
<SIP/seminario-00000015>AGI Tx >> agi_enhanced: 0.0
<SIP/seminario-00000015>AGI Tx >> agi_accountcode:
<SIP/seminario-00000015>AGI Tx >> agi_threadid: -1219855472
<SIP/seminario-00000015>AGI Tx >> agi_arg_1: 200703072

```

Figura 5.28.- Respuesta del sistema: consulta de Fechas de Horario por Materia del Día Viernes

Tabla XXXV.- Consulta de Horario por Materia del Día Viernes: Salida

| Ingreso | Salida |
|--------------------|--|
| | Se realiza una consulta a la base Asterisk para obtener las materias que el estudiante tiene el día viernes. Se produce una salida de audio que dice, ordenada por hora , una a una las materias que el estudiante tiene el día viernes con su respectiva aula, con la aplicación text2wav |
| | Se ejecuta la aplicación getData(Escuchar_con_again,5000,1), con lo que se pide al usuario que ingrese 1 para volver a escuchar la consulta y 0 para regresar al menú principal |
| Usuario Ingresas 0 | Se ejecuta la aplicación Goto(MenuPrincipal,start,1). Se termina el script y se regresa al plan de marcado |

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
<SIP/seminario-00000015>AGI Tx >> agi_arg_2: 5
<SIP/seminario-00000015>AGI Tx >>
<SIP/seminario-00000015>AGI Rx << ANSWER
<SIP/seminario-00000015>AGI Tx >> 200 result=0
<SIP/seminario-00000015>AGI Rx << STREAM FILE /var/spool/asterisk//tmp/text2wav_bfd71305e843d59df565a5a1acdcbc52 ""
0
-- Playing '/var/spool/asterisk//tmp/text2wav_bfd71305e843d59df565a5a1acdcbc52' (escape_digits=) (sample_offset
0)
<SIP/seminario-00000015>AGI Tx >> 200 result=0 endpos=56718
<SIP/seminario-00000015>AGI Rx << GET DATA Escuchar_Con_again 3000 1
-- <SIP/seminario-00000015> Playing 'Escuchar_Con_again.slin' (language 'es')
<SIP/seminario-00000015>AGI Tx >> 200 result=0
<SIP/seminario-00000015>AGI Rx << EXEC Goto MenuPrincipal,start,1
-- AGI Script Executing Application: (Goto) Options: (MenuPrincipal,start,1)
-- Goto (MenuPrincipal,start,1)
<SIP/seminario-00000015>AGI Tx >> 200 result=0
-- <SIP/seminario-00000015> AGI Script HorClases.php completed, returning 0
    
```

Figura 5.29.- Respuesta del sistema: salida de Consulta de Horario por Materia del Día Viernes

Tabla XXXVI.- Consulta de Horario por Materia del Día Sábado

| Ingreso | Salida |
|----------------------|--|
| Usuario ingresa 6 | Se ejecuta la aplicación AGI(HorClases.php,\${matricula},\${exten}) . Con lo que se pasa el control de la llamada al script HorClases.php y se consulta las clases que el estudiante tiene para el día sábado |

```

-- <SIP/seminario-0000001b> Playing 'HorarioClases.slin' (language 'es')
-- Executing [6@HorarioClases:1] AGI("SIP/seminario-0000001b", "HorClases.php,206703072,6") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/HorClases.php
<SIP/seminario-0000001b>AGI Tx >> agi_request: HorClases.php
<SIP/seminario-0000001b>AGI Tx >> agi_channel: SIP/seminario-0000001b
<SIP/seminario-0000001b>AGI Tx >> agi_language: es
<SIP/seminario-0000001b>AGI Tx >> agi_type: SIP
<SIP/seminario-0000001b>AGI Tx >> agi_uniqueid: 1321837126.27
<SIP/seminario-0000001b>AGI Tx >> agi_version: 1.8.3.2
<SIP/seminario-0000001b>AGI Tx >> agi_callerid: 301
<SIP/seminario-0000001b>AGI Tx >> agi_calleridname: VoIP
<SIP/seminario-0000001b>AGI Tx >> agi_callingpres: 0
<SIP/seminario-0000001b>AGI Tx >> agi_callingani2: 0
<SIP/seminario-0000001b>AGI Tx >> agi_callington: 0
<SIP/seminario-0000001b>AGI Tx >> agi_callingtns: 0
<SIP/seminario-0000001b>AGI Tx >> agi_dnid: 888
<SIP/seminario-0000001b>AGI Tx >> agi_rdnis: unknown
<SIP/seminario-0000001b>AGI Tx >> agi_context: HorarioClases
<SIP/seminario-0000001b>AGI Tx >> agi_extension: 6
<SIP/seminario-0000001b>AGI Tx >> agi_priority: 1
<SIP/seminario-0000001b>AGI Tx >> agi_enhanced: 0.0
<SIP/seminario-0000001b>AGI Tx >> agi_accountcode:
<SIP/seminario-0000001b>AGI Tx >> agi_threadid: -1219855472
<SIP/seminario-0000001b>AGI Tx >> agi_arg_1: 206703072
<SIP/seminario-0000001b>AGI Tx >> agi_arg_2: 6
<SIP/seminario-0000001b>AGI Tx >>

```

Figura 5.30.- Respuesta del sistema: consulta de Fechas de Horario por Materia del Día Sábado

Tabla XXXVII.- Consulta de Horario por Materia del Día Sábado: Salida

| Ingreso | Salida |
|---------|--|
| | Se realiza una consulta a la base Asterisk para obtener las materias que el estudiante tiene el día lunes. A realizar la consulta se observa que el estudiante no tiene clases el día sábado, por lo que se reproduce una con la aplicación background la grabación No_tiene_Materias |
| | Se ejecuta la aplicación Goto(MenuPrincipal,start,1). Se termina el script y se regresa al plan de marcado |

```

root@localhost:~
Archivo Editar Ver Terminal Solapas Ayuda
<SIP/seminario-00000015>AGI Rx << ANSWER
<SIP/seminario-00000015>AGI Tx >> 200 result=0
<SIP/seminario-00000015>AGI Rx << EXEC Background No_tiene_materias
-- AGI Script Executing Application: (Background) Options: (No_tiene_materias)
-- <SIP/seminario-00000015> Playing 'No_tiene_materias.slin' (language 'es')
<SIP/seminario-00000015>AGI Tx >> 200 result=0
<SIP/seminario-00000015>AGI Rx << EXEC Goto MenuPrincipal,start,1
-- AGI Script Executing Application: (Goto) Options: (MenuPrincipal,start,1)
-- Goto (MenuPrincipal,start,1)
<SIP/seminario-00000015>AGI Tx >> 200 result=0
-- <SIP/seminario-00000015>AGI Script HorClases.php completed, returning 0
-- Executing [start@MenuPrincipal:1] Background("SIP/seminario-00000015", "Opciones_Menu_Principal") in new stack

```

Figura 5.31.- Respuesta del sistema: salida de Consulta de Horario por Materia del Día Sábado

5.2 PRUEBAS DE RENDIMIENTO

Antes de poner un servidor de telefonía en producción, es necesaria la realización de pruebas de rendimiento de dicho sistema para conocer cuál es el número de llamadas concurrentes que el servidor puede manejar, sin disminuir la QoS.

Para el Sistema Audio ESPOL IP, se tiene que el máximo número de llamadas concurrentes que el servidor va a recibir es 24, ya que ese es el número de canales que proporciona la tarjeta 1AEX2406EF del fabricante Digium, la cual se instalara en el servidor.

Al momento de realizar las pruebas de rendimiento no se tuvo a disposición la tarjeta 1AEX2406EF por lo cual se decidió realizar las pruebas con canales SIP en lugar de analógicos. Para la generación de llamadas usando canales SIP se utiliza el software SIPp que es un

generador de tráfico SIP de libre distribución y muy utilizado para realizar pruebas de rendimiento con servidores Asterisk.

La prueba consistió en verificar si el servidor responde correctamente a las 24 llamadas simultáneas.

5.2.1 ARQUITECTURA DE LA PRUEBA DE RENDIMIENTO

El escenario de prueba consiste de un servidor SIPp el cual genera el tráfico de llamadas SIP y el servidor del sistema Audio ESPOL IP el cual recibe el tráfico de llamadas SIP, como se muestra la Figura 5.32.

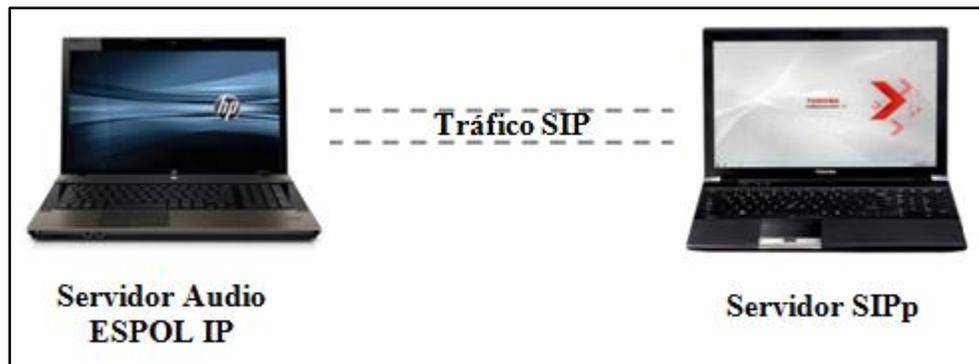


Figura 5.32.- Diseño de Prueba Rendimiento

En las Tabla XXXVIII se muestra las características del servidor SIPp usado en las pruebas.

Tabla XXXVIII.- Especificaciones del Servidor SIPp

| SERVIDOR SIPp | |
|--------------------------|---------------------------------------|
| Procesador | Intel ® Core ™ i7 CPU Q 720 @ 1.60GHz |
| Modelo | Toshiba Satellite A505 |
| Memoria RAM | 2.00 GB |
| Sistema Operativo | CentOS release 5.6 (Final) |
| Version SIPp | sipp-standard-3.2-linux-2.6-intel |

5.2.2 INSTALACIÓN SIPP

Para la instalación de SIPp, se descarga el archivo sipp-standard-3.2-linux-2.6-intel de la página oficial del proyecto SIPp (<http://sipp.sourceforge.net>), el cual es un archivo rpm, se da doble clic al archivo rpm y se procede la instalación, considerando instalar previamente las dependencias: el compilador C++ y las librerías de *curses* o *ncurses*, *openssl*, *libpcap*, *libnet* y *gsl*.

5.2.3 CONFIGURACIÓN DEL SERVIDOR AUDIO ESPOL IP

Se configura un cliente SIP en el archivo de configuración sip.conf

```
[sipp]
type=friend
context=prueba_rendimiento
host=dynamic
nat=yes
dtmfmode=auto
disallow=all
allow=ulaw
allow=gsm
```

Se crea el contexto prueba_rendimiento en el archivo extensions.conf. Este es el contexto de llegada de las llamadas de prueba. Una vez la llamada llega al contexto, esta es redirigida al script prueba.php

```
[prueba_rendimiento]
exten => 888,1,AGI(prueba.php,200703072,2)
```

Se crea el script prueba.php el cual consulta la fecha de registro de un estudiante dado el número de matrícula. Como se muestra en el Anexo E, Script usado en la prueba de rendimiento.

La duración de una llamada dentro del escenario de prueba es en promedio de 28 segundos.

Se modifica el número de archivos que GNU/Linux permite abrir por defecto en el servidor Audio ESPOL IP, ejecutando el comando:

```
ulimit -s 4096
```

5.2.4 REALIZACIÓN DE PRUEBA DE RENDIMIENTO

El propósito de las pruebas de rendimiento es conocer el número de llamadas simultáneas que el servidor puede gestionar, teniendo en cuenta que el máximo número de llamadas que el servidor recibe es 24, ya que este es el número de puertos físicos FXO de la tarjeta usada en el servidor Audio ESPOL IP. Se analiza el consumo de CPU y memoria. El programa SIPp realiza la generación de tráfico SIP, pero no nos brinda información sobre el requerimiento de CPU y memoria por lo cual se utiliza el programa SAR (Reporte de Actividades del sistema) con lo que se obtiene un reporte sobre el rendimiento del sistema.

5.2.5 INICIO DE LA PRUEBA DE CLIENTE SIPP

Para iniciar el generador de tráfico SIP se ejecuta el siguiente comando

```
sipp -s 888 -sn uac direccionIP -d duracion -m  
total_de_llamadas -r llamadas/seg -l llamadas_simultaneas -  
trace_screen -trace_err -trace_stat
```

- sn uac: se ingresa la dirección IP que se va a generar el tráfico SIP, en este caso la dirección IP del servidor Asterisk
- d: tiempo máximo de duración de cada llamada
- m: número total de llamadas
- r: número de llamada por segundo
- l: número de llamadas simultáneas
- trace_screen: almacena la información mostrada en la pantalla durante la generación de llamadas SIP
- trace_err: almacena información sobre los errores ocurridos durante la generación de llamadas SIP
- trace_stat: almacena estadísticas de las llamadas realizadas

5.2.6 REALIZACIÓN DE PRUEBA DE RENDIMIENTO CPU Y MEMORIA

Para visualizar el consumo de recursos del CPU se ejecuta el siguiente comando:

```
sar -o intervalo muestra
```

Y para visualizar el consumo de recursos de memoria se ejecuta el siguiente comando:

```
sar -m intervalo muestra
```

Donde el parámetro *intervalo* nos indica el tiempo que toma entre muestras y el parámetro *muestra* nos indica el número de total de toma de datos.

5.2.7 RESULTADOS DE LA PRUEBA DE RENDIMIENTO

Es importante considerar si el servidor Asterisk se puede saturar por el parámetro `-l` que establece el número máximo de llamadas simultáneas del cliente SIPp, brindara un resultado en procesamiento y/o de memoria del 100% del servidor Asterisk.

El comando ejecutado en el servidor SIPp para la realización de las pruebas de rendimiento

```
sipp -s 888 -sn uac 192.168.0.102 -d 28000 -m 100 -r 2 -l 100 -
trace_screen -trace_err -trace_stat
```

Tabla XXXIX.- Parámetros configurados en SIPp al realizar las Pruebas de Rendimientos

| Tiempo de duración máximo de cada llamada | Número de llamadas máximas | Llamadas por segundo | Llamadas simultaneas |
|---|----------------------------|----------------------|----------------------|
| 28000 | 100 | 2 | 100 |

El resultado de las pruebas de rendimiento, nos indica que todas las llamadas se realizaron con éxito, que la tasa de llamadas por

segundo fue de 1.224, que la duración total de la prueba fue de un minuto con 21 segundos y que el máximo número de llamadas simultáneas que se alcanzó fue 58, como se muestra en la Figura 5.33.

```

|----- Scenario Screen ----- [1-9]: Change Screen --
Call-rate(length)  Port  Total-time  Total-calls  Remote-host
2.0(28000 ms)/1.000s  5061      81.67 s      100  192.168.0.102:5060(UDP)

Call limit reached (-m 100), 0.000 s period  0 ms scheduler resolution
0 calls (limit 100)                          Peak was 58 calls, after 43 s
0 Running, 62 Paused, 0 Woken up
0 dead call msg (discarded)                  0 out-of-call msg (discarded)
1 open sockets

Messages  Retrans  Timeout  Unexpected-Msg
INVITE  ----->      100      0         0           0
100 <-----      100      0         0           0
180 <-----         0         0         0           0
183 <-----         0         0         0           0
200 <-----      E-RTD1 100      0         0           0
ACK  ----->      100      0         0           0
Pause [ 28.0s]      100      0         0           0
BYE  ----->      100      0         0           0
200 <-----      100      0         0           0

----- Test Terminated -----

----- Statistics Screen ----- [1-9]: Change Screen --
Start Time      | 2012-06-04  07:54:05:173  1338814445.173021
Last Reset Time | 2012-06-04  07:55:26:852  1338814526.852125
Current Time    | 2012-06-04  07:55:26:855  1338814526.855810
-----+-----+-----
Counter Name    | Periodic value          | Cumulative value
-----+-----+-----
Elapsed Time    | 00:00:00:003           | 00:01:21:682
Call Rate       | 0.000 cps               | 1.224 cps
-----+-----+-----
Incoming call created | 0                       | 0
OutGoing call created | 0                       | 100
Total Call created  | 0                       | 100
Current Call       | 0                       |
-----+-----+-----
Successful call    | 0                       | 100
Failed call        | 0                       | 0
-----+-----+-----
Response Time 1   | 00:00:00:000           | 00:00:03:314
Call Length      | 00:00:00:000           | 00:00:31:319
-----+-----+-----
Test Terminated

```

Figura 5.33.- Estadísticas de las pruebas realizadas con SIPp

En las Figuras 5.34 y 5.35, se muestran los resultados de consumo de CPU y memoria bajo la prueba de rendimiento, en la cual se realizaron máximo 58 llamadas concurrentes, las cuales se visualizan cuando se satura el servidor Asterisk con el consumo de CPU al 100%.

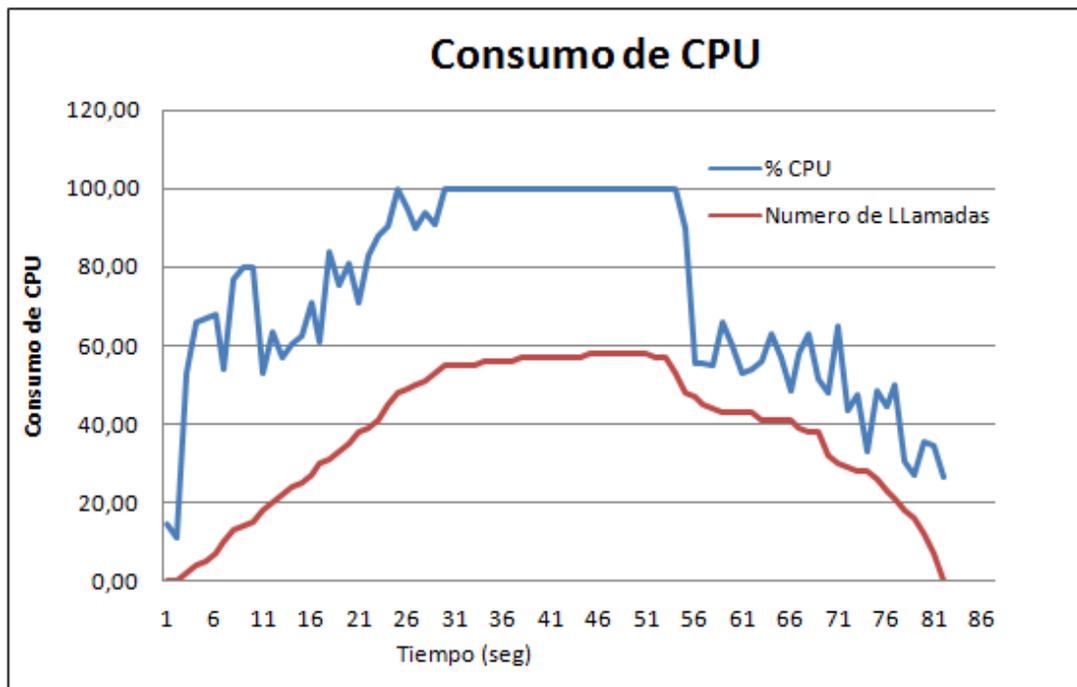


Figura 5.34.- Gráfica de Consumo de CPU durante la Prueba de Rendimiento

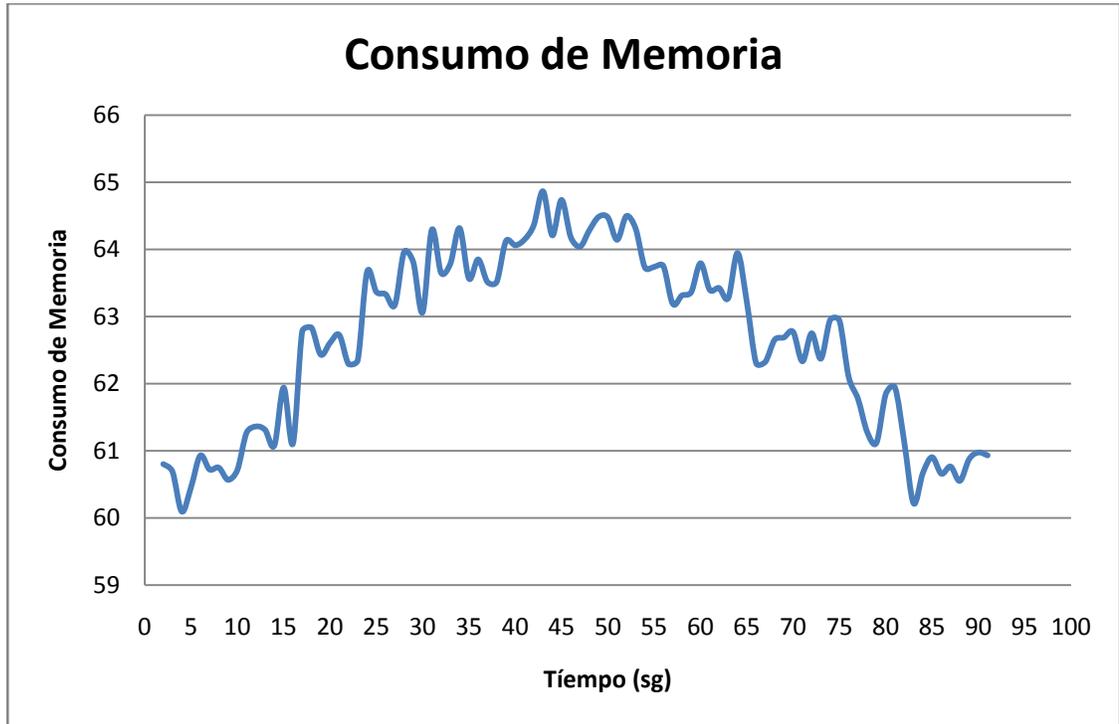


Figura 5.35.- Gráfica de Consumo de Memoria durante la Prueba de Rendimiento

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

1. Una central IP basado en Asterisk es de bajo costo de implementar, alrededor de \$2.500 comparada con las centrales conmutadas y propietarias IP con las mismas características de cuyo costo aproximado es \$3.384 y \$3.982 respectivamente.

En el Anexo F se puede observar información sobre los costos expuestos previamente.

2. Durante las Pruebas de Rendimiento se observo que el factor limitante para el número máximo de llamadas concurrentes que el servidor puede atender con éxito es el procesador. El sistema atendió con éxito 58 llamadas simultáneas lo cual sobrepasa el límite de, máximo de 24 llamadas concurrentes, planteada en el proyecto.

3. El consumo de memoria no es un factor crítico en el sistema, se observó en promedio un aumento del 4% del consumo de memoria durante las pruebas.
4. El uso de la librería phpAGI nos permite implementar scripts de forma más sencilla haciendo uso de funciones implementada por esta librería, lo cual nos permite realizar acciones que requieren de una lógica más avanza.
5. Los sistemas telefónicos VoIP son escalables, ya que permiten un fácil crecimiento a medida que aumentan los requerimientos. Además permiten la interconexión de distintas centrales IP implementadas con Asterisk mediante el uso de troncales lo que disminuye los costos de hardware.

RECOMENDACIONES

1. Las características físicas de servidor depende en gran manera del número de llamadas simultáneas que el sistema vaya a recibir, por lo tanto se debe realizar un análisis para obtener un estimado de cuanto es el número de llamadas simultáneas que el servidor de telefonía va a soportar. Para ello es importante la realización de pruebas de rendimiento antes de la puesta en producción del sistema para garantizar la disponibilidad y la Qos del sistema.
2. Si el sistema va a soportar un número considerables de llamadas simultáneas entonces es preferible contratar un proveedor de servicios de telefonía IP (sip o iax) o digital (E1, T1, PRI), según las requerimientos, en lugar de tarjetas analógicas lo cual se ve reflejado en un reducción de costos en la adquisición de hardware y ahorro de espacio.
3. Al usar tarjetas analógicas es importante tomar acciones para evitar que el eco disminuya la calidad de la comunicación. En lo posible comprar tarjetas que tengan mecanismo de cancelación de eco por hardware y si no se cuenta con ello se recomiendo el uso de programas ayudan a disminuir el eco, pero que producen una carga extra al procesador.

4. Siempre seleccionar servers que traigan tarjetas de red gigabit ethernet en lo posible, si traen 2 tarjetas es mucho más adecuado.

5. Al momento de comprar tarjetas de telefonía, revisar bien las especificaciones de la misma para evitar problemas de incompatibilidad de hardware y driver, así como la alimentación eléctrica que estas necesiten.

ANEXOS

ANEXO A

MANUAL DE USUARIO

Este manual le permitirá utilizar todas las opciones brindadas por el sistema Audio ESPOL IP, el cual cuenta con un menú principal que desglosa 4 opciones de consulta:

Opción 1: Consultar las calificaciones de las materias tomadas en un término y año específico dado el número de matrícula.

Se reproduce una grabación pidiendo que ingrese el año electivo que se desea consultar. Después de haber ingresado el año se reproduce una grabación pidiendo que ingrese el término que se desea consultar. Después de haber ingresado el término se reproducirá una grabación del año y termino ingresado pidiéndole confirmar si los datos ingresados son correctos. Se presiona 1 para confirmar y 0 para volver a digitar año y término. Sí se presionó la opción 1 se reproduce una grabación de las materias registradas en ese año y terminó con sus respectivas calificaciones, si no hay materias que presentar se reproducirá una grabación indicando que el estudiante no tomó materias en ese término. Al final de la consulta brinda la opción de volver a escuchar la información de la consulta o volver al menú principal.

Opción 2: Consultar fechas de registro para el próximo término

Se reproduce una grabación con las siguientes opciones:

- Presione 1 si desea escuchar su fecha de registro.
- Presione 2 si desea escuchar todas las fechas de registro.

Si se presiono 1 solo se escuchará la fecha de registro del usuario dependiendo de su promedio. Si se presiono 2 se escuchará una grabación con todas las fechas de registro. Al final de cada consulta se brinda la opción de volver a escuchar la información de la consulta o volver al menú principal.

Opción 3: Consultar fecha de exámenes en el presente término

Se reproduce una grabación con las siguientes opciones:

- Presione 1 si desea escuchar sus exámenes parciales.
- Presione 2 si desea escuchar sus exámenes finales.
- Presione 3 si desea escuchar sus exámenes mejoramiento.

Sí se presiona 1 se escucha el nombre de la materia, paralelo, aula, fecha y hora de los examen parciales. Sí se presiona 2 se escucha el nombre de la materia, paralelo, aula, fecha y hora de los examen finales. Sí se presiona 3 se

escucha el nombre de la materia, paralelo, aula, fecha y hora de los exámenes de mejoramiento. Al final de cada consulta se brinda la opción de volver a escuchar la información de la consulta o volver al menú principal.

Opción 4: Consultar horario por día de las materias registradas

Se reproduce una grabación con las siguientes opciones:

- Presione 1 si desea escuchar su horario del día lunes.
- Presione 2 si desea escuchar su horario del día martes.
- Presione 3 si desea escuchar su horario del día miércoles.
- Presione 4 si desea escuchar su horario del día jueves.
- Presione 5 si desea escuchar su horario del día viernes.
- Presione 6 si desea escuchar su horario del día sábado.

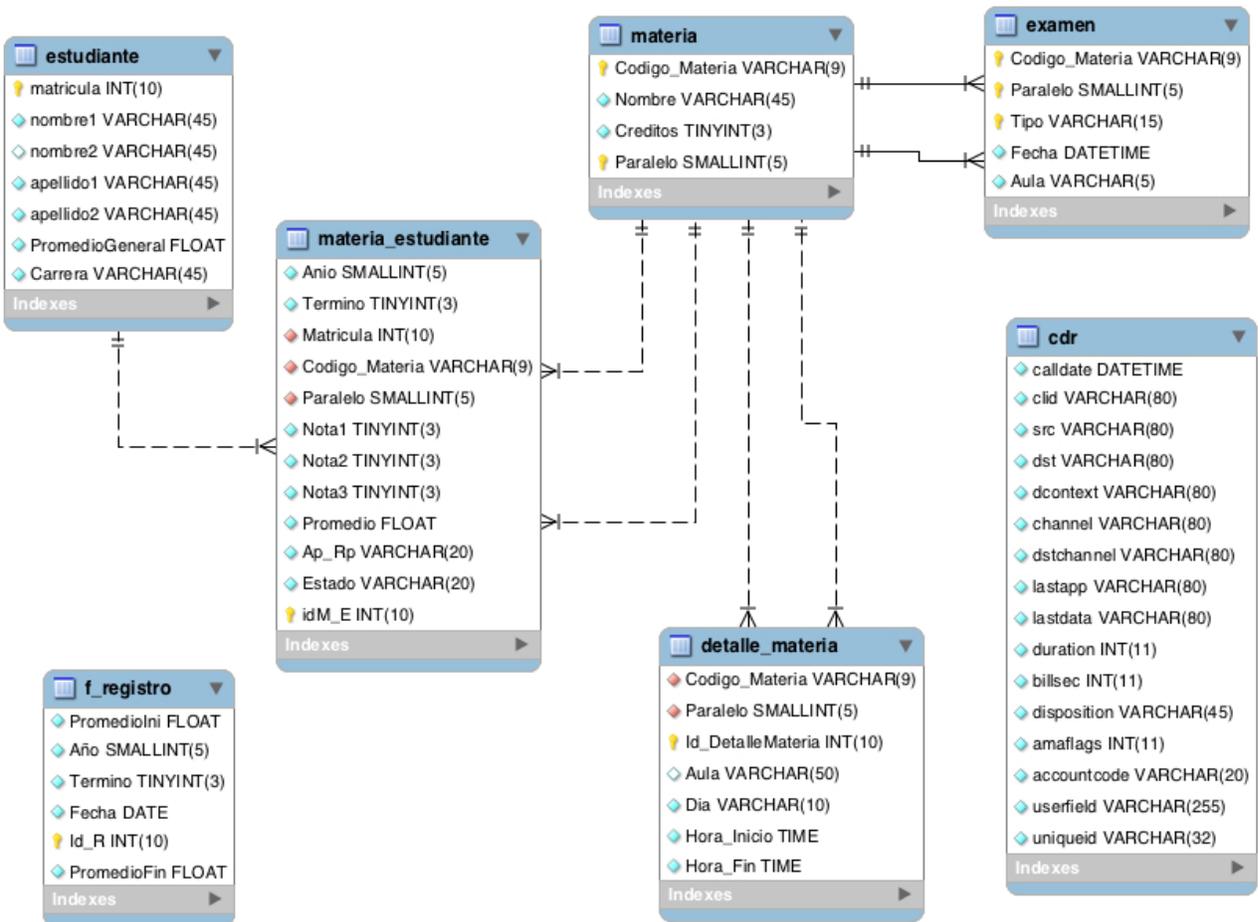
Cualquiera de estas opciones brinda información del nombre de las materias con la hora y aula. Al final de cada consulta se brinda la opción para volver a escuchar la información o volver al menú principal.

Importante: En todos los puntos del proceso donde se le solicita al usuario el ingreso de algún dato, se establece un tiempo de espera después del cual sino se ha detectado el ingreso de ningún de dato por parte del usuario se cierra la conexión de Asterisk con el usuario.

ANEXO B

MODELO LÓGICO DE LA BASE DE DATOS

Figura B.1 Modelo Lógico de la Base de Datos Asterisk



SCRIPT CREACIÓN DE TABLAS PARA LA BASE DE DATOS ASTERISK

```
DROP TABLE IF EXISTS `asterisk`.`cdr`;
CREATE TABLE `asterisk`.`cdr` (
  `calldate` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  `clid` varchar(80) NOT NULL DEFAULT "",
  `src` varchar(80) NOT NULL DEFAULT "",
  `dst` varchar(80) NOT NULL DEFAULT "",
  `dcontext` varchar(80) NOT NULL DEFAULT "",
  `channel` varchar(80) NOT NULL DEFAULT "",
  `dstchannel` varchar(80) NOT NULL DEFAULT "",
  `lastapp` varchar(80) NOT NULL DEFAULT "",
  `lastdata` varchar(80) NOT NULL DEFAULT "",
  `duration` int(11) NOT NULL DEFAULT '0',
  `billsec` int(11) NOT NULL DEFAULT '0',
  `disposition` varchar(45) NOT NULL DEFAULT "",
  `amaflags` int(11) NOT NULL DEFAULT '0',
  `accountcode` varchar(20) NOT NULL DEFAULT "",
  `userfield` varchar(255) NOT NULL DEFAULT "",
  `uniqueid` varchar(32) NOT NULL DEFAULT "",
  KEY `calldate` (`calldate`),
  KEY `dst` (`dst`),
  KEY `accountcode` (`accountcode`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```
DROP TABLE IF EXISTS `asterisk`.`estudiante`;
CREATE TABLE `asterisk`.`estudiante` (
  `matricula` int(10) unsigned NOT NULL,
  `nombre1` varchar(45) NOT NULL,
  `nombre2` varchar(45) DEFAULT NULL,
  `apellido1` varchar(45) NOT NULL,
  `apellido2` varchar(45) NOT NULL,
  `PromedioGeneral` float unsigned NOT NULL,
  `Carrera` varchar(45) NOT NULL,
  PRIMARY KEY (`matricula`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
DROP TABLE IF EXISTS `asterisk`.`materia`;
```

```

CREATE TABLE `asterisk`.`materia` (
  `Codigo_Materia` varchar(9) NOT NULL,
  `Nombre` varchar(45) NOT NULL,
  `Creditos` tinyint(3) unsigned NOT NULL,
  `Paralelo` smallint(5) unsigned NOT NULL,
  PRIMARY KEY (`Codigo_Materia`,`Paralelo`) USING BTREE
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

DROP TABLE IF EXISTS `asterisk`.`detalle_materia`;
CREATE TABLE `asterisk`.`detalle_materia` (
  `Codigo_Materia` varchar(9) NOT NULL,
  `Paralelo` smallint(5) unsigned NOT NULL,
  `Id_DetalleMateria` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `Aula` varchar(50) DEFAULT NULL,
  `Dia` varchar(10) NOT NULL,
  `Hora_Inicio` time NOT NULL,
  `Hora_Fin` time NOT NULL,
  PRIMARY KEY (`Id_DetalleMateria`),
  KEY `Paralelo` (`Codigo_Materia`,`Paralelo`),
  CONSTRAINT `Codigo_Materia` FOREIGN KEY (`Codigo_Materia`)
REFERENCES `materia` (`Codigo_Materia`),
  CONSTRAINT `Paralelo` FOREIGN KEY (`Codigo_Materia`,`Paralelo`)
REFERENCES `materia` (`Codigo_Materia`,`Paralelo`)
) ENGINE=InnoDB AUTO_INCREMENT=20 DEFAULT CHARSET=latin1;

```

```

DROP TABLE IF EXISTS `asterisk`.`examen`;
CREATE TABLE `asterisk`.`examen` (
  `Codigo_Materia` varchar(9) NOT NULL,
  `Paralelo` smallint(5) unsigned NOT NULL,
  `Tipo` varchar(15) NOT NULL,
  `Fecha` datetime NOT NULL,
  `Aula` varchar(5) NOT NULL,
  PRIMARY KEY (`Codigo_Materia`,`Paralelo`,`Tipo`) USING BTREE,
  CONSTRAINT `Cod_Materia` FOREIGN KEY (`Codigo_Materia`,`Paralelo`)
REFERENCES `materia` (`Codigo_Materia`,`Paralelo`),
  CONSTRAINT `Par` FOREIGN KEY (`Codigo_Materia`,`Paralelo`)
REFERENCES `materia` (`Codigo_Materia`,`Paralelo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='InnoDB free: 4096
kB; (`Codigo_Materia`) REFER `asterisk/mat';

```

```

DROP TABLE IF EXISTS `asterisk`.`materia_estudiante`;
CREATE TABLE `asterisk`.`materia_estudiante` (
  `Anio` smallint(5) unsigned NOT NULL,
  `Termino` tinyint(3) unsigned NOT NULL,
  `Matricula` int(10) unsigned NOT NULL,
  `Codigo_Materia` varchar(9) NOT NULL,
  `Paralelo` smallint(5) unsigned NOT NULL,
  `Nota1` tinyint(3) unsigned NOT NULL,
  `Nota2` tinyint(3) unsigned NOT NULL,
  `Nota3` tinyint(3) unsigned NOT NULL,
  `Promedio` float NOT NULL,
  `Ap_Rp` varchar(20) NOT NULL,
  `Estado` varchar(20) NOT NULL,
  `idM_E` int(10) unsigned NOT NULL AUTO_INCREMENT,
  PRIMARY KEY (`idM_E`),
  KEY `Paral` (`Codigo_Materia`,`Paralelo`),
  KEY `matricula` (`Matricula`),
  CONSTRAINT `Codigo` FOREIGN KEY (`Codigo_Materia`,`Paralelo`)
REFERENCES `materia` (`Codigo_Materia`,`Paralelo`),
  CONSTRAINT `matricula` FOREIGN KEY (`Matricula`) REFERENCES
`estudiante` (`matricula`),
  CONSTRAINT `Paral` FOREIGN KEY (`Codigo_Materia`,`Paralelo`)
REFERENCES `materia` (`Codigo_Materia`,`Paralelo`)
) ENGINE=InnoDB AUTO_INCREMENT=11 DEFAULT CHARSET=latin1;

```

```

DROP TABLE IF EXISTS `asterisk`.`f_registro`;
CREATE TABLE `asterisk`.`f_registro` (
  `PromedioIni` float unsigned NOT NULL,
  `Año` smallint(5) unsigned NOT NULL,
  `Termino` tinyint(3) unsigned NOT NULL,
  `Fecha` date NOT NULL,
  `Id_R` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `PromedioFin` float unsigned NOT NULL,
  PRIMARY KEY (`Id_R`)
) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=latin1;

```

ANEXO C

PLAN DE MARCADO

[general]

autofallthrough=yes

clearglobalvars=no

[globals]

USER1=SIP/seminario ;case sensitive

;llamadas que son recibidas desde la PSTN

[entrantes-PSTN]

exten => s,1,Answer()

 same=>n,Goto(internos,888,1)

[internos]

;extension donde se responden las llamadas recibidas al ivr Audio Espol IP

exten => 888,1,AGI(ingreso_sistema.php)

;extension usada para grabar mensaje usando el sentitizador de voz Festival

exten => 124,1,Answer()

exten => 124,2,System(echo "No existen datos para presentar, el estudiante no
tiene clases" | /usr/bin/text2wave -scale 1.5 -F 8000 -o

/var/lib/asterisk/sounds/tmp/No_clases.wav)

exten => 124,3,Playback(/var/lib/asterisk/sounds/tmp/No_clases.wav)

exten => 124,4,Hangup()

.....

[prueba_rendimiento]

exten => 888,1,AGI(consultarRegistro.php,200703072,2)

.....

[MenuPrincipal]

exten => start,1,Background(Opciones_Menu_Principal)

 same => n,WaitExten(10)

; presione 1 consultar calificaciones


```
.....  
[FechaRegistros]  
exten => start,1,Background(FechaRegistros)  
    same => n,WaitExten(5)
```

```
exten => 1,1,AGI(consultarRegistro.php,{matricula},1)  
exten => 2,1,AGI(consultarRegistro.php,{matricula},2)  
exten => 0,1,Goto(MenuPrincipal,start,1)
```

```
exten => i,1,Playback(opcion_invalida)  
    same => n,Background(FechaRegistros)  
    same => n,WaitExten(5)
```

```
exten => t,1,Hangup
```

```
.....  
[HorarioExamenes]  
exten => start,1,Background(Menu_Con_Fecha_Examenes)  
    same => n,WaitExten(5)
```

```
exten => 1,1,AGI(HorExamenes.php,{matricula},1)  
exten => 2,1,AGI(HorExamenes.php,{matricula},2)  
exten => 3,1,AGI(HorExamenes.php,{matricula},3)  
exten => 0,1,Goto(MenuPrincipal,start,1)
```

```
exten => i,1,Playback(opcion_invalida)  
    same => n,Background(Menu_Con_Fecha_Examenes)  
    same => n,WaitExten(5)
```

```
exten => t,1,Hangup
```

```
.....  
[HorarioClases]
```

```
exten => start,1,Background(HorarioClases)  
    same => n,WaitExten(5)
```

```
exten => 1,1,AGI(HorClases.php,{matricula},1)  
exten => 2,1,AGI(HorClases.php,{matricula},2)
```

```
exten => 3,1,AGI(HorClases.php,${matricula},3)
exten => 4,1,AGI(HorClases.php,${matricula},4)
exten => 5,1,AGI(HorClases.php,${matricula},5)
exten => 6,1,AGI(HorClases.php,${matricula},6)
exten => 0,1,Goto(MenuPrincipal,start,1)
```

```
exten => i,1,Playback(opcion_invalida)
    same => n,Background(HorarioClases)
    same => n,WaitExten(5)
```

```
exten => t,1,Hangup
```

ANEXO D

SCRIPT PARA INGRESO AL SISTEMA

ingreso_sistema.php

```
#!/usr/bin/php -q
<?php
ob_implicit_flush(true);
set_time_limit(6);
error_reporting(E_ALL);
require ('phpagi-2.20/phpagi.php');
    function connect_db()
    {
    $db_connection = mysql_connect ('localhost', 'root', 'root') or die
(mysql_error());
    $db_select = mysql_select_db('asterisk',$db_connection) or die
(mysql_error());
    }
//programa inicial
$agi = new AGI();
$agi->answer();
$banderapedir=0;
do{
if($banderapedir==0)
$mensaje='bienvenida';
else
$mensaje='ingrese_matricula';
//pedimos ingresar la matricula, seteamos un máximo tiempo de espera en
segundos por una respuesta y el ingreso máximo de 9 dígitos
$opcion = $agi->get_data($mensaje, 4000, 9);
//obtenemos la información introducida desde el telefono
$mat = $opcion['result'];
//verificamos que se haya introducido un numero de matricula caso
contrario se cuelga
if($mat==NULL)
$agi->hangup();
//se dice el número de matrícula y se pide la confirmación del mismo
else{
```

```

$agi->exec(SayDigits,$mat);
do{
    $opcion = $agi->get_data('eleccion_matricula', 3000, 1);
    //obtenemos la información introducida desde el teléfono
    $clave = $opcion['result'];
    if($clave==NULL)
    $agi->hangup();
    if($clave!='0' && $clave!='1')
    $agi->exec(Playback,"opcion_invalida");
    //verificamos que estudiante exista en la base de datos
    if($clave=='1'){
        //se crea la conexión a la base de datos
        connect_db();
        //se realiza el query de consulta para verificar si el estudiante
        existe
        $query1 = "SELECT nombre1,nombre2,apellido1,apellido2
FROM estudiante WHERE matricula = '$mat' ";
        $result_estudiante = mysql_query($query1);
        $row_count = mysql_num_rows($result_estudiante);
        $row1 = mysql_fetch_array ($result_estudiante);
        if($row_count !=0 ) { //estudiante existe
            $nombre1=$row1[0];
            $apellido1=$row1[2];
            $agi->text2wav("bienvenido $nombre1 $apellido1");
            $banderapedir=0;
            $agi->set_variable("__matricula",$mat);
            $agi->exec_goto("MenuPrincipal","start","1");
        }else { // estudiante no existe
            $agi->exec(Playback,"estudiante_no_existe");
            $banderapedir=1;
        }
    }
    if($clave=='0')
        $banderapedir=1;
}while($clave!='0' && $clave!='1');
}
}while($banderapedir==1);
exit;
?>

```

SCRIPT PARA CONSULTAR MATERIA

consultarMateria.php

```
#!/usr/bin/php -q
<?php
ob_implicit_flush(true);
set_time_limit(6);
error_reporting(E_ALL);
require ('phpagi-2.20/phpagi.php');
//función usada para crear la conexión a la base de datos
function connect_db()
{
    $db_connection = mysql_connect ('localhost', 'root', 'root') or die
(mysql_error());
    $db_select = mysql_select_db('asterisk',$db_connection) or die
(mysql_error());
}
//se usa esta función para que al momento de que el sintetizador de
//voz leer los números los lea correctamente
function camb_dig_letras($dato)
{
    switch($dato)
    {
        case "00": $cad="cero"; break;
        case "01": $cad="uno"; break;
        case "02": $cad="dos"; break;
        case "03": $cad="tres"; break;
        case "04": $cad="cuatro"; break;
        case "05": $cad="cinco"; break;
        case "06": $cad="seis"; break;
        case "07": $cad="siete"; break;
        case "08": $cad="ocho"; break;
        case "09": $cad="nueve"; break;
        default: $cad=$dato;
    }
    return $cad;
}
```

```

}
// programa inicial
$mat=$argv[1];
$anio=$argv[2];
$ter=$argv[3];
$sagi = new AGI();
$sagi->answer();
//se crea la conexión a la base de datos
connect_db();
//se consulta a la base por las notas correspondientes a la primera,
segunda y tercera evaluación usando como parámetros el número de
matrícula del estudiante año y termino
$query1 = "SELECT m.Nombre, me.Paralelo, me.Nota1, me.Nota2,
me.Nota3, me.Ap_Rp FROM estudiante e, materia_estudiante me, materia
m WHERE e.matricula = '$mat' and e.matricula = me.Matricula and
me.Codigo_Materia= m.Codigo_Materia and me.anio='$anio' and
me.termino='$ter' and me.Paralelo=m.Paralelo";
do{
    $result_materiaReg = mysql_query($query1);
    $row_count = mysql_num_rows($result_materiaReg);
    if($row_count!=0){
        while ($row1=mysql_fetch_array ($result_materiaReg)) {
//registro ya existe
            $nombre=$row1[0];
            $paralelo=camb_dig_letras($row1[1]);
            $nota1=camb_dig_letras($row1[2]);
            $nota2=camb_dig_letras($row1[3]);
            $nota3=camb_dig_letras($row1[4]);
            $ApRp=$row1[5];
            if($ApRp=="Rp")
                $ApRp="Reprobado";
            else
                $ApRp="Aprobado";
            $sagi->text2wav("materia $nombre, paralelo $paralelo, primer parcial
            $nota1, segundo parcial $nota2, tercer parcial $nota3, $ApRp ");
        }
//lazo para pedir que ingrese un 1 para volver a escuchar la consulta o cero
para regresar al menú principal, si presiona
//otra cosa manda un mensaje de opción inválida y vuelve a pedir el
ingreso

```

```

do{
    $opcion = $agi->get_data('Escuchar_Con_again', 3000,
1);
    //obtenemos la información introducida desde el
telefono
    $clave = $opcion['result'];
    if($clave==NULL)
        $agi->hangup();
    //si el usuario no presiono 0 ni 1, entonces no es una
opción valida
    if($clave!='0' && $clave!='1')
        $agi->exec(Playback,"opcion_invalida");
    //si el usuario presiono 0 regresa al menú principal
    if($clave=='0'){
        $agi->set_context("MenuPrincipal");
        $agi->exec_goto("MenuPrincipal","start","1");}
    }while($clave!='0' && $clave!='1');
}
else { // estudiante no ha tomado materias en dicho termino
    $agi->exec(Background,"No_existen_Calificaciones");
    $agi->set_context("MenuPrincipal");
    $agi->exec_goto("MenuPrincipal","start","1");
}
}while($clave=='1');
exit;
?>

```

SCRIPT PARA CONSULTAR REGISTRO

consultarRegistro.php

```

#!/usr/bin/php -q
<?php
ob_implicit_flush(true);
set_time_limit(6);
error_reporting(E_ALL);
require ('phpagi-2.20/phpagi.php');
//función para crear una conexión con la base de datos
function connect_db()

```

```

{
$db_connection = mysql_connect ('localhost', 'root', 'root') or die
(mysql_error());
$db_select = mysql_select_db('asterisk',$db_connection) or die
(mysql_error());
}
function rangoMes($mes){
    switch($mes)
    {
        case 1: $cad="Enero"; break;
        case 2: $cad="Febrero"; break;
        case 3: $cad="Marzo"; break;
        case 4: $cad="Abril"; break;
        case 5: $cad="Mayo"; break;
        case 6: $cad="Junio"; break;
        case 7: $cad="Julio"; break;
        case 8: $cad="Agosto"; break;
        case 9: $cad="Septiembre"; break;
        case 10: $cad="Octubre"; break;
        case 11: $cad="Noviembre"; break;
        case 12: $cad="Diciembre"; break;
    }
    return $cad;
}
function dia($diaR){
    switch($diaR)
    {
        case "00": $cad="cero"; break;
        case "01": $cad="uno"; break;
        case "02": $cad="dos"; break;
        case "03": $cad="tres"; break;
        case "04": $cad="cuatro"; break;
        case "05": $cad="cinco"; break;
        case "06": $cad="seis"; break;
        case "07": $cad="siete"; break;
        case "08": $cad="ocho"; break;
        case "09": $cad="nueve"; break;
        default: $cad=$diaR;
    }
    return $cad;
}

```

```

}
// programa inicial
//se lee el argumento 1 que se pasa al script y se lo almacena en la variable
$mat
$mat=$argv[1];
//se lee el argumento 2 que se pasa al script y se lo almacena en la variable
$exten que nos indica que tipo de consulta el usuario quiere realizar
$exten=$argv[2];
//se crea la conexión a la base de datos
connect_db();
//se crea una nueva instancia de la clase AGI
$agi = new AGI();
$agi->answer();
do{
//se consulta la fecha de registro del usuario dado el numero de matricula
ingresado
    if($exten==1){
        $query1 = "SELECT f.Fecha FROM estudiante e, f_registro f
WHERE e.matricula = '$mat' and e.PromedioGeneral <= f.PromedioIni and
e.PromedioGeneral > f.promedioFin ";
        $result_fecha = mysql_query($query1);
        $row_count = mysql_num_rows($result_fecha);
        $row1 = mysql_fetch_array ($result_fecha);
        if($row_count!=0){
            $fecha=$row1[0];
            $fechaR = explode("-", $fecha);
            $mes=rangoMes($fechaR[1]);
            $fechaR[2]=dia($fechaR[2]);
            $agi->text2wav("Usted está habilitado para registrarse,
desde el dia $fechaR[2] de $mes del $fechaR[0]");
        }
    }
//se consulta todas las fechas de registro
    if($exten==2){
        $query1 = "SELECT * FROM f_registro f ORDER BY Fecha
ASC";
        $result_fecha = mysql_query($query1);
        $row_count = mysql_num_rows($result_fecha);
        $fila=0;

```

```

while ($row1=mysql_fetch_array ($result_fecha)) { //registro
existe
    $fila=$fila+1;
    $promIni=$row1[0]; //valor obtenido de la consulta a la
base
    $anio=$row1[1]; //valor obtenido de la consulta a la base
    $ter=$row1[2]; //valor obtenido de la consulta a la base
    $fecha=$row1[3]; //valor obtenido de la consulta a la
base
    $promFin=$row1[5]; //valor obtenido de la consulta a la
base
$fechaR = explode("-", $fecha);
    $mes=rangoMes($fechaR[1]);
    $fechaR[2]=dia($fechaR[2]);
    if($fila==1)
        $agi->text2wav("Fechas de registros del $anio
termino $ter");
        $agi->text2wav("$fechaR[2] de $mes del $fechaR[0] ,
para promedios desde $promIni hasta $promFin");
    }
}
//lazo para pedir que ingrese un 1 para volver a escuchar la consulta o cero
para regresar al menú principal, si realiza otra entrada manda un mensaje
de opción inválida y vuelve a pedir el ingreso
do{
    $opcion = $agi->get_data('Escuchar_Con_again', 5000, 1);
    //obtenemos la información introducida desde el teléfono
    $clave = $opcion['result'];
    if($clave==NULL)
        $agi->hangup();

        if($clave!='0' && $clave!='1')
            $agi->exec(Playback,"opcion_invalida");
    if($clave=='0'){
        $agi->set_context("MenuPrincipal");
        $agi->exec_goto("MenuPrincipal","start","1");}
    }while($clave!='0' && $clave!='1');
}while($clave=='1');
exit;
?>

```

SCRIPT PARA CONSULTAR HORARIO DE EXAMENES

HorExamenes.php

```
#!/usr/bin/php -q
<?php
ob_implicit_flush(true);
set_time_limit(6);
error_reporting(E_ALL);
require ('phpagi-2.20/phpagi.php');
function connect_db()
{
$db_connection = mysql_connect ('localhost', 'root', 'root') or die
(mysql_error());
$db_select = mysql_select_db('asterisk',$db_connection) or die
(mysql_error());
}
function mes($FechaEx)
{
switch($FechaEx)
{
case 1: $cad="Enero"; break;
case 2: $cad="Febrero"; break;
case 3: $cad="Marzo"; break;
case 4: $cad="Abril"; break;
case 5: $cad="Mayo"; break;
case 6: $cad="Junio"; break;
case 7: $cad="Julio"; break;
case 8: $cad="Agosto"; break;
case 9: $cad="Septiembre"; break;
case 10: $cad="Octubre"; break;
case 11: $cad="Noviembre"; break;
case 12: $cad="Diciembre"; break;
}
return $cad;
}
function camb_dig_letras($horaEx)
```

```

{
  switch($horaEx)
  {
    case "00": $cad="cero"; break;
    case "01": $cad="uno"; break;
    case "02": $cad="dos"; break;
    case "03": $cad="tres"; break;
    case "04": $cad="cuatro"; break;
    case "05": $cad="cinco"; break;
    case "06": $cad="seis"; break;
    case "07": $cad="siete"; break;
    case "08": $cad="ocho"; break;
    case "09": $cad="nueve"; break;
    default: $cad=$horaEx;
  }
  return $cad;
}
// programa inicial
//se lee el argumento 1 correspondiente a la matrícula y se lo almacena en
la variable $mat
$mat=$argv[1];
//se lee el argumento 2 correspondiente al tipo de examen que se quiere
consultar y se lo almacena en la variable $opcion
$opcion=$argv[2];
if($opcion==1)
    $tipo_examen="PARCIAL";
elseif($opcion==2)
    $tipo_examen="FINAL";
else
    $tipo_examen="MEJORAMIENTO";
$agi = new AGI();
$agi->answer();
//se realiza la conexión a la base de datos
connect_db();
$query1 = "SELECT m.Nombre, me.Paralelo, ex.Aula, ex.Fecha FROM
materia_estudiante me, examen ex, materia m WHERE me.Matricula= '$mat'
and me.Codigo_Materia=m.Codigo_Materia and me.paralelo=m.paralelo
and m.Codigo_Materia=ex.Codigo_Materia and m.paralelo=ex.paralelo and
ex.tipo='$tipo_examen' and me.estado='T' ORDER BY ex.Fecha DESC ";
do{

```

```

$result_examenes = mysql_query($query1);
$row_count = mysql_num_rows($result_examenes);
if($row_count!=0){
while ($row1=mysql_fetch_array ($result_examenes)) { //aquí hace el lazo
para ir leyendo el resulset de los datos recibidos
    $nombre=$row1[0];
    $paralelo=$row1[1];
    $aula=$row1[2];
    $fecha=$row1[3];
    $cadenaFecha=explode(" ",$fecha);
    $fecha_ex=explode("-", $cadenaFecha[0]);
    $hora_ex=explode(":", $cadenaFecha[1]);
    $mes=mes($fecha_ex[1]);
    $paralelo=camb_dig_letras($paralelo);
    $hora_ex[1]=camb_dig_letras($hora_ex[1]);
    $hora_ex[0]=camb_dig_letras($hora_ex[0]);

    $agi->text2wav("materia $nombre, paralelo $paralelo,
aula $aula, fecha $fecha_ex[2] de $mes del $fecha_ex[0], hora $hora_ex[0]
horas con $hora_ex[1] minutos");
    }
//lazo para pedir que ingrese un 1 para volver a escuchar la consulta o cero
para regresar al menú principal, si presiona otra cosa manda un mensaje
de opción invalida y vuelve a pedir el ingreso
do{
    $opcion = $agi->get_data('Escuchar_Con_again', 3000,
1);
    //obtenemos la información introducida desde el
telefono

    $clave = $opcion['result'];
    if($clave==NULL)
        $agi->hangup();
    if($clave!='0' && $clave!='1')
        $agi->exec(Playback,"opcion_invalida");
    if($clave=='0'){
        $agi->set_context("MenuPrincipal");
        $agi->exec_goto("MenuPrincipal", "start", "1");}
    }while($clave!='0' && $clave!='1');
}
else { // estudiante no se encuentra tomando materias

```

```

        $agi->exec(Background,"No_tiene_materias");
        $agi->set_context("MenuPrincipal");
        $agi->exec_goto("MenuPrincipal","start","1");
    }
}while($clave=='1');
exit;
?>

```

SCRIPT PARA CONSULTAR HORARIO DE CLASES

HorClases.php

```

#!/usr/bin/php -q
<?php
ob_implicit_flush(true);
set_time_limit(6);
error_reporting(E_ALL);
require ('phpagi-2.20/phpagi.php');
//función que se utiliza para conectar con la base de datos
function connect_db()
{
    $db_connection = mysql_connect ('localhost', 'root', 'root') or die
(mysql_error());
    $db_select = mysql_select_db('asterisk',$db_connection) or die
(mysql_error());
}
//se pasa como parámetro un digito y se lo pasa a un dia de la semana
function dia($diaClases)
{
    switch($diaClases)
    {
        case 1: $cad="lunes"; break;
        case 2: $cad="martes"; break;
        case 3: $cad="miercoles"; break;
        case 4: $cad="jueves"; break;
        case 5: $cad="viernes"; break;
        case 6: $cad="sabado"; break;
    }
}

```

```

        return $cad;
    }
function camb_dig_letras($horaEx)
{
    switch($horaEx)
    {
        case "00": $cad="cero"; break;
        case "01": $cad="uno"; break;
        case "02": $cad="dos"; break;
        case "03": $cad="tres"; break;
        case "04": $cad="cuatro"; break;
        case "05": $cad="cinco"; break;
        case "06": $cad="seis"; break;
        case "07": $cad="siete"; break;
        case "08": $cad="ocho"; break;
        case "09": $cad="nueve"; break;
        default: $cad=$horaEx;
    }
    return $cad;
}
// programa inicial
$mat=$argv[1];
$opcion=$argv[2];
$dia=dia($opcion);
$agi = new AGI();
$agi->answer();
//se hace una conexión con la base de datos de asterisk
connect_db();
//se hace la consulta a la base del horario de clases dado una matricula
$query1 = "SELECT m.nombre,d.aula,d.hora_inicio,d.hora_fin From materia
m, materia_estudiante me, detalle_materia d WHERE me.matricula='$mat'
and me.codigo_materia= m.codigo_materia and me.paralelo=m.paralelo
and me.codigo_materia= d.codigo_materia and me.paralelo=d.paralelo and
d.dia='$dia' ORDER BY d.hora_inicio";
do{
    $result_dia = mysql_query($query1);
    $row_count = mysql_num_rows($result_dia);
    if($row_count!=0){
while ($row1=mysql_fetch_array ($result_dia)) { //aqui hace el lazo parair
leyendo el resulset de los datos recibidos

```

```

        $nombre=$row1[0];
        $aula=$row1[1];
        $hInicio=$row1[2];
        $hFin=$row1[3];
        $hora_Inicio=explode(":",$hInicio);
        $hora_Fin=explode(":",$hFin);
        $hora_Inicio[1]=camb_dig_letras($hora_Inicio[1]);
        $hora_Inicio[0]=camb_dig_letras($hora_Inicio[0]);
        $hora_Fin[1]=camb_dig_letras($hora_Fin[1]);
        $hora_Fin[0]=camb_dig_letras($hora_Fin[0]);
$agi->text2wav("materia $nombre, aula $aula, desde $hora_Inicio[0] horas
con $hora_Inicio[1] minutos, a $hora_Fin[0] horas con $hora_Fin[1]
minutos");
    }
//lazo para pedir que ingrese un 1 para volver a escuchar la consulta o cero
para regresar al menú principal, si presiona otra cosa manda un mensaje
de opción inválida y vuelve a pedir el ingreso
    do{
        $opcion = $agi->get_data('Escuchar_Con_again', 3000,
1);
        //obtenemos la información introducida desde el
teléfono
        $clave = $opcion['result'];
        if($clave==NULL)
            $agi->hangup();
        if($clave!='0' && $clave!='1')
            $agi->exec(Playback,"opcion_invalida");
        if($clave=='0'){
            $agi->set_context("MenuPrincipal");
            $agi->exec_goto("MenuPrincipal","start","1");}
        }while($clave!='0' && $clave!='1');
    }
else { // estudiante no se encuentra tomando materias
    $agi->exec(Background,"No_tiene_materias");
    $agi->set_context("MenuPrincipal");
    $agi->exec_goto("MenuPrincipal","start","1");
}
}while($clave=='1');
exit;
?>

```

ANEXO E

SCRIPT USADO EN LA PRUEBA DE RENDIMIENTO

```
#!/usr/bin/php -q
<?php
ob_implicit_flush(true);
set_time_limit(6);
error_reporting(E_ALL);
require ('phpagi-2.20/phpagi.php');
function connect_db(){
    $db_connection = mysql_connect ('localhost', 'root', 'root') or die
(mysql_error());
    $db_select = mysql_select_db('asterisk',$db_connection) or die
(mysql_error());
}
function rangoMes($mes){
switch($mes){
    case 1: $cad="Enero"; break;
    case 2: $cad="Febrero"; break;
    case 3: $cad="Marzo"; break;
    case 4: $cad="Abril"; break;
    case 5: $cad="Mayo"; break;
    case 6: $cad="Junio"; break;
    case 7: $cad="Julio"; break;
    case 8: $cad="Agosto"; break;
    case 9: $cad="Septiembre"; break;
    case 10: $cad="Octubre"; break;
    case 11: $cad="Noviembre"; break;
    case 12: $cad="Diciembre"; break;
    }    return $cad;
}
function dia($diaR){
switch($diaR)
{
case "00": $cad="cero"; break;
case "01": $cad="uno"; break;
case "02": $cad="dos"; break;
case "03": $cad="tres"; break;
case "04": $cad="cuatro"; break;
```

```

case "05": $cad="cinco"; break;
case "06": $cad="seis"; break;
case "07": $cad="siete"; break;
case "08": $cad="ocho"; break;
case "09": $cad="nueve"; break;
default: $cad=$diaR;
}
return $cad;
}
// programa inicial
//se lee el argumento 1 que se pasa al script y se lo almacena en la variable
$mat
$mat=$argv[1];

//se lee el argumento 2 que se pasa al script y se lo almacena en la variable
$exten que nos indica que tipo de consulta el usuario quiere realizar
$exten=$argv[2];

connect_db();

$agi = new AGI();
$agi->answer();
//se consulta la fecha de registro del usuario dado el numero de matricula
ingresado
if($exten==1){
$query1 = "SELECT f.Fecha FROM estudiante e, f_registro f WHERE
e.matricula = '$mat' and e.PromedioGeneral <= f.PromedioIni and
e.PromedioGeneral > f.promedioFin ";

$result_fecha = mysql_query($query1);
$row_count = mysql_num_rows($result_fecha);
$row1 = mysql_fetch_array ($result_fecha);

if($row_count!=0){
    $fecha=$row1[0];
    $fechaR = explode("-", $fecha);
    $mes=rangoMes($fechaR[1]);
    $fechaR[2]=dia($fechaR[2]);
    $agi->text2wav("Usted esta habilitado para registrarse, desde el dia
    $fechaR[2] de $mes del $fechaR[0]");
}
}

```

```

}
}
//se consulta todas las fechas de registro
if($exten==2){
    $query1 = "SELECT * FROM f_registro f ORDER BY Fecha ASC";
    $result_fecha = mysql_query($query1);
    $row_count = mysql_num_rows($result_fecha);
    $fila=0;
    while ($row1=mysql_fetch_array ($result_fecha)) { //registro ya existe
        $fila=$fila+1;
        $promIni=$row1[0];
        $anio=$row1[1];
        $ter=$row1[2];
        $fecha=$row1[3];
        $promFin=$row1[5];
        $fechaR = explode("-", $fecha);
        $mes=rangoMes($fechaR[1]);
        $fechaR[2]=dia($fechaR[2]);

        if($fila==1)
            $sagi->text2wav("Fechas de registros del $anio termino $ter");
        $sagi->text2wav("$fechaR[2] de $mes del $fechaR[0] , para promedios desde
        $promIni hasta $promFin");
    }
}exit;
?>

```

ANEXO F

PABX

La central de telefonía analógica Panasonic modelo KD-TX1232 brinda 12 llamadas concurrentes con 64 extensiones usando XDP, tiene la capacidad de interconexión de un armario esclavo para duplicar la capacidad hasta 24 llamadas externas (CO) y hasta 128 extensiones.



Figura F.1 Central analógica fabricante Panasonic modelo KD-TX1232 con 24 llamadas externas

Fuente: Guía de especificaciones de la central analógica Panasonic KD-TX1232:

<http://pdf.textfiles.com/manuals/TELECOM-F-R/Panasonic%20KX-TD1232%20Installation.pdf>

La central analógica Panasonic posee las siguientes características:

- Restricción de llamadas programable
- Capacidad para ISDN

- Capacidad para correo de voz
- Opción para programación y mantenimiento remoto
- Manejo inteligente de llamadas
- Opción DISA
- UCD (Distribución uniforme de llamadas)
- Desvío de llamadas
- Modo diurno, nocturno y almuerzo
- Opción Portero eléctrico y abridor
- Llamadas de emergencia
- Conferencia de 5 participantes
- Monitoreo de cuarto
- Grupo de extensiones
- Selección de patrón de timbrado
- Mensaje en espera
- Reportes de actividad de llamadas
- Introducción de código de cuentas
- Restricción de llamadas
- Bloqueo electrónico de extensiones
- Duración de llamada limitada
- Introducción de código de cuentas

- Bloqueo electrónico de extensiones
- Extensión fantasma

En la Tabla F.1 muestra la información técnica de la central de telefonía analógica Panasonic modelo KD-TX1232:

Tabla F.1 Características técnicas de la central de telefonía analógica fabricante Panasonic modelo KD-TX1232

Fuente: Guía de especificaciones de la central analógica Panasonic KD-TX1232:

<http://pdf.textfiles.com/manuals/TELECOM-F-R/Panasonic%20KX-TD1232%20Installation.pdf>

| Características técnicas de la central de telefonía analógica Panasonic modelo KD-TX1232 | |
|---|---|
| Fuente de Alimentación | 220 VAC – 230 VAC, 50 Hz |
| Peso | 7.8 Kg |
| Consumo Máximo de Potencia | 140 W |
| Minima fuga de resistencia | 15 000 ohmios |
| Voltage de timbrado | 70 Vrms a 25 Hz dependiendo de la carga de la llamada |
| Conmutado | No bloqueo de interruptor de tiempo PCM |

| | |
|-----------------------------|---|
| Voltaje del Circuito | 5 V, 15 V |
| Batería | Litio |
| Marcado | Marcado por pulsos (DP) 10 pps, 20 pps |
| Tono de marcado | DTMF |
| Modular | Jack |
| Conector | KX-TD1232: Amphenol |
| Temporizador | 84 ms a 1000 ms |
| Temperatura | 0°C - 40°C |
| Humedad | 10% - 90% |

El costo de la PBX Panasonic se basa en los siguientes parámetros para que opere:

- Se requiere el uso de teléfonos Panasonic como KX-T7520, KX-T7531, KX-T7533, KX-T7536, KX-T7550, entre otros compatibles con la central, lo cual permite la utilización de todas las funciones que brinda la misma.
- La necesidad de operadores para brindar la atención al usuario, como un máximo de 24 operadores para que realice la misma función de atender un máximo de 24 llamadas simultaneas exitosas.

- La instalación de un armario secundario o modulo de expansión el cual brinda 12 puertos adicionales para permitir un máximo de 24 llamadas concurrentes.

Se presenta en la Tabla F.1 un resumen de costo aproximado tan solo de la implementacion de central analogica Panasonic:

Tabla F.2 Costo de Central Analógica Panasonic KX-TD1232 que permita 24 llamadas concurrentes

Fuente: Página web de la empresa Panafonic:

http://www.panafonic.com/b2c/index.php?cPath=21_32_36&osCsid=9c6bafb29c223e96

0f031a6762813fa0

| Costo de central analógica Panasonic KX-TD1232 que permita 24 llamadas concurrentes | | |
|--|-----------------|------------------|
| | Cantidad | Precio |
| Central analógica Panasonic modelo KX- TD1232 | 1 | \$1,692.12 |
| Modulo de expansión | 1 | \$1,692.12 |
| | TOTAL | \$3384.24 |

IP PBX PROPIETARIAS

La central IP de fabricante Xorcom IP PBX modelo XR2021 con 24 módulos FXO, soporta 300 usuarios y 24 llamadas concurrentes.

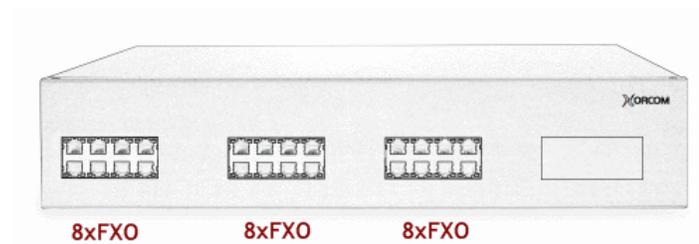


Figura F.2 Central IP de fabricante Xorcom IP PBX modelo XR2021 con 24 módulos FXO

Fuente: Pagina Web Oficial de Xorcom: <http://www.xorcom.com/catalog/xr2021.html>

La central posee las siguientes características al sistema:

- Interfaz web para el usuario, fácil de emplear.
- Interfaz en diferentes lenguajes.
- CDR (Historial de detalles de llamadas) vía acceso Web.
- Troncales TDM/SIP/IAX.
- Extensiones remotas.
- Correo de voz.
- Soporta Faxes.

- Envía correo de voz al e-mail.
- Sistema de Menú para el IVR.
- Grupos de llamadas.
- Colas de llamada.
- Sala de Conferencia.
- Tiempo basado en enrutamiento.
- Reglas de marcado avanzada.
- Música de espera.
- Acceso vía web a los correos de voz.
- Paginación e intercomunicación.
- Pantalla de Estado de Administración.
- Administrador de Paquetes (para realizar actualizaciones fácilmente).
- Herramienta de configuración de red.
- Herramienta de aprovisionamiento del teléfono (el proceso de preparación y equipamiento del teléfono IP).
- Cancelación de eco.

En la Tabla F.3 se describe información técnica de hardware, red, software, módulo FXO de la central IP.

Tabla F.3 Características de hardware de la central IP Xorcom IP PBX modelo XR2021 con 24 módulos FXO

Fuente: Pagina Web Oficial de Xorcom: <http://www.xorcom.com/catalog/xr2021.html>

| Características de hardware de la central IP Xorcom modelo XR2021 | |
|--|----------------------------------|
| Procesador | Intel Atom D525 Dual Core 1.8GHz |
| Disco Duro | 250 GB 2.5" |
| RAM | 1GB |
| USB | 2 puertos externos USB 2.0 |
| Dimensiones | 48.26 cm |
| Costo | \$3,982.95 |

Tabla F.4 Características de red de la central IP Xorcom IP PBX modelo XR2021 con 24 módulos FXO

Fuente: Pagina Web Oficial de Xorcom: <http://www.xorcom.com/catalog/xr2021.html>

| Características de red de la central IP Xorcom modelo XR2021 | |
|---|-------------|
| Puerto Ethernet | 10/100 Mb/s |

Tabla F.5 Características de modulo FXO de la central IP Xorcom IP PBX modelo XR2021 con 24 módulos FXOs

Fuente: Pagina Web Oficial de Xorcom: <http://www.xorcom.com/catalog/xr2021.html>

| Características de modulo FXO de la central IP Xorcom modelo XR2021 |
|--|
| Compatibilidad con la impedancia |
| Flexibilidad con parámetros DC, Se ajusta automáticamente 110/120 V y 50 a 60 Hz |
| Luces indicadoras de configuraciones activas y status de las líneas, por canales |

Tabla F.6 Características de Software de la central IP Xorcom IP PBX modelo XR2021 con 24 módulos FXO

Fuente: Pagina Web Oficial de Xorcom: <http://www.xorcom.com/catalog/xr2021.html>

| Características de Software de la central IP Xorcom modelo XR2021 | |
|--|---|
| Distribución de Asterisk | 1.8.x (Elastix 2.2x) |
| Version Linux | CentOS 5.x |
| Interfaz Gráfica de Usuario | FreePBX |
| Tunneling Rapido | Provee acceso remoto seguro del producto para propósitos de soporte |

Se puede aumentar las capacidades de respaldo, confiabilidad, capacidad, cancelación de eco y puertos de entrada USB de la central IP Xorcom IP PBX modelo XR2021 con 24 módulos FXO añadiendo los siguientes módulos.

Tabla F.7 Características adicionales de la central IP Xorcom IP PBX modelo XR2021
con 24 modulos FXO

Fuente: Pagina Web Oficial de Xorcom: <http://www.xorcom.com/catalog/xr2021.html>

| Características adicionales de la central IP Xorcom modelo XR2021 | |
|--|---|
| Recuperación Rápida | P/N XR0068 Provee respaldo al disco |
| USB Externo Hub | P/N XR0107 Provee 4 bancos de canales Astribank USB |
| Modulo de cancelación de ECO | P/N XR0124 Modulo de cancelación de ECO a través del hardware |

BIBLIOGRAFÍA

- [1] **Van Meggelen J., Smith J. y Madsen L.**, “Conceptos de Asterisk”, Asterisk The Future of Telephony, Editorial O’Reilly Media, 2005.
- [2] **Edgar Landívar**, Comunicaciones Unificadas con Elastix Volumen 1, Primera Edición, versión Beta, 2008.
- [3] **VOIP-Info.org**, “Asterisk AGI”, info.org/wiki/view/Asterisk+AGI, 2011, Online, 2011.
- [4] **Jesus_Camacho_Rodriguez**, “Proyecto de Fin De Carrera”, http://www.adminso.es/images/b/b2/PFC_Jesus_Camacho_Rodriguez_Capitulo_5.pdf , Online, 2007.
- [5] **Julio Gómez López, Francisco Gil Montoya**, “Proyecto de Fin de Carrera”, http://www.adminso.es/images/d/dc/PFC_eugenio.pdf, Online, 2010.
- [6] **Gonzalo Marcote**, “Como instalar e integrar festival con Asterisk” <http://www.gonzalomarcote.com/2010/como-instalar-e-integrar-festival-con-asterisk/>, Online, 2010.