



UNIVERSIDAD NACIONAL DE LOJA

**ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y
LOS RECURSOS NATURALES NO RENOVABLES**

CARRERA DE INGENIERÍA EN SISTEMAS

TÍTULO:

**IMPLANTACIÓN DE TELEFONÍA MÓVIL VOZ SOBRE IP,
UTILIZANDO LA HERRAMIENTA μELASTIX EN UN
RASPBERRY PI, PARA LA UNIVERSIDAD NACIONAL DE
LOJA EN EL AEIRNNR.**

*Tesis previa a la obtención del
título de Ingeniero en Sistemas*

AUTORES:

**Benjamín Alejandro Pinza Tapia
Ximena Marisol Herrera Iñiguez**

DIRECTOR:

Ing. Mario Andrés Palma Jaramillo, Mg. Sc.

**LOJA - ECUADOR
2015**

CERTIFICADO

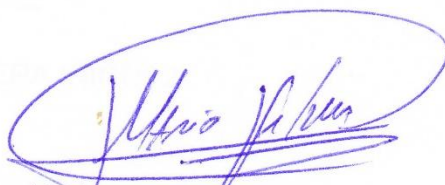
Ingeniero Mario Andrés Palma Jaramillo

**DOCENTE DEL ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS
NATURALES NO RENOVABLES Y DIRECTOR DE TESIS**

CERTIFICA:

Que los egresados Benjamín Alejandro Pinza Tapia y Ximena Marisol Herrera realizaron el presente trabajo de titulación denominado "Implantación de telefonía móvil voz sobre ip, utilizando la herramienta pelastix en un raspberry pi, para la Universidad Nacional de Loja en el AEIRNNR, bajo mi dirección y asesoramiento cumpliendo con los requerimientos establecidos por las normas generales para la graduación en la Universidad Nacional de Loja, tanto de forma como de contenido.

Loja, 20 de julio del 2015.



Ing. Mario Andrés Palma Jaramillo

DIRECTOR DE TESIS

AUTORÍA

Nosotros: **BENJAMIN ALEJANDRO PINZA TAPIA Y XIMENA MARISOL HERRERA IÑIGUEZ**, declaramos ser autores del presente trabajo de tesis y eximimos expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos y acciones legales por el contenido de la misma.

Adicionalmente aceptamos y autorizamos a la Universidad Nacional de Loja, la publicación de nuestra tesis en el Repositorio Institucional – Biblioteca Virtual.

Firma:


BENJAMIN ALEJANDRO PINZA TAPIA

Cédula: 1104252489

Firma:


XIMENA MARISOL HERRERA IÑIGUEZ

Cédula: 1104341696

Fecha: 04 de agosto del 2015

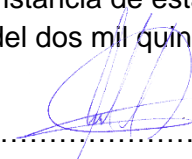
CARTA DE AUTORIZACIÓN DE TESIS POR PARTE DE LOS AUTORES, PARA LA CONSULTA, REPRODUCCIÓN PARCIAL O TOTAL Y PUBLICACIÓN ELECTRÓNICA DEL TEXTO COMPLETO

Nosotros, **BENJAMIN ALEJANDRO PINZA TAPIA Y XIMENA MARISOL HERRERA IÑIGUEZ**, declaramos ser los autores de la tesis titulada: **“IMPLANTACIÓN DE TELEFONÍA MÓVIL VOZ SOBRE IP, UTILIZANDO LA HERRAMIENTA μELASTIX EN UN RASPBERRY PI, PARA LA UNIVERSIDAD NACIONAL DE LOJA EN EL AEIRNNR”**, autorizamos al Sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos, muestre al mundo la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el repositorio digital institucional:

Los usuarios pueden consultar el contenido de este trabajo en el RDI, en las redes de información del país y del exterior, con los cuales tengan convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia que realice un tercero.

Para constancia de esta autorización en la ciudad de Loja, a los cinco días del mes de agosto del dos mil quince.

Firma: .....

Autor: Benjamín Pinza Tapia

Cédula: 1104252489

Dirección: Loja. (Cdla. Zamora Huayco)

Correo electrónico: pinzabenzamin22@hotmail.com

Teléfono: 2560525

Celular: 0997427327

Firma: .....

Autor: Ximena Marisol Herrera Iñiguez

Cédula: 1104341696

Dirección: Loja. (Cdla. Zamora Huayco)

Correo electrónico: ximari.hi@hotmail.com

Teléfono: 2560525

Celular: 0989415592

DATOS COMPLEMENTARIOS

Director de tesis: Ing. Mario Andrés Palma Jaramillo, Mg.Sc

Tribunal de grado: Ing. Walter Rodrigo Tene Ríos, Mg.Sc

Ing. Franco Hernán Salcedo López, Mg. Adm

Dra. María Esperanza Gaona, Mg. Sc

AGRADECIMIENTO

A las Autoridades de la Universidad Nacional de Loja, de manera especial a las Autoridades de la Carrera de Ingeniería en Sistemas del Área de Energía, las Industrias y los Recursos Naturales No Renovables, quienes con solvencia académica, profesional y calidad humana contribuyeron para que se lleve a feliz término nuestra formación profesional en este nivel.

Al Ing. Mario Andrés Palma Jaramillo, Mg.Sc. Director de Tesis, por su aporte, sugerencias y experiencias en la Dirección de la presente tesis.

Los Autores

DEDICATORIA

A mis hijos Dayana, Israel y Amelia, quienes son la fuente de mi inspiración diaria. A mi padre Benjamín, a mi madre Rita quienes me inculcaron por la senda de la sabiduría y del bienestar común, a mis hermanos y hermanas Fabián, Eliana, Rita y Katherine, quienes han estado junto a mí en los momentos difíciles de mi vida.

Benjamín Alejandro.

A mis hijos Dayana, Israel y Amelia, quienes son la fuente de mi inspiración diaria. A mi padre Francel, a mi madre Graciela quienes me apoyaron durante mi trayectoria académica, a mis hermanos Edy, Richard y Sandro, quienes frecuentamos desde la niñez los senderos de la vida.

Ximena Marisol

a. Título

“Implantación de telefonía móvil voz sobre ip, utilizando la herramienta µelastix en un raspberry pi, para la Universidad Nacional de Loja en el AEIRNNR.

b. Resumen

El presente proyecto se lo ejecutó con el propósito de implementar telefonía voz sobre IP con teléfonos móviles, utilizando la herramienta µelastix en un raspberry pi B, para el “AEIRNNR de la Universidad Nacional de Loja”, el mismo que se lo configuró conforme a las exigencias y requerimientos del AEIRNNR de la Universidad Nacional de Loja.

Para el análisis, interpretación y recolección de los requerimientos del presente proyecto se utilizó métodos científicos, así como técnicas (entrevista y observación) e instrumentos (cuestionario y hoja de observación) que permitieron la recopilación y análisis de la información, aplicados al personal de cada departamento que utiliza la red telefónica.

Para tener un buen funcionamiento de nuestro sistema de telefonía ip, se utilizó un hardware Raspberry Pi B y software µelastix, estos se los consideran libres de uso tanto en hardware como en software respectivamente, además se los tomó con el propósito de cumplir con las necesidades que afronta el AEIRNNR al no disponer cada docente con un teléfono ip fijo. Por otra parte, se utilizó equipos que funcionan con el anterior software de telefonía ip. El Software µelastix cuenta con amigables interfaces, para poder comprender más acerca de las configuraciones básicas y necesarias para un sistema de telefonía, este incluye desde la creación de extensiones, ivr's, troncales, etc. Esta aplicación fue diseñada de una forma amigable al administrador o delegado desde el cual puede realizar varias configuraciones y soluciones inmediatas.

En los resultados se detalla el proceso aplicado para el mejoramiento de la red telefónica, este contexto describe las configuraciones de µelastix y de los voip (voz sobre IP), además las diferentes pruebas realizadas en la red telefónica y utilización de herramientas para los resultados de calidad de servicio.

Se pudo obtener como resultado y evidenciar, que en el sistema de red telefónica fija y la telefonía voip (voz sobre IP) móvil, entre administrativos y docentes del AEIRNNR de UNL, fue exitosa y ayudó a la comunicación entre los mencionados.



Summary

The present research study was carried out with the purpose of implementing voice over IP with mobile phones by using µelastix tools in a raspberry pi B. This telephony was designed to be used by the AEIRNNR area of the Universidad Nacional de Loja; that is the reason why it was set up according to its needs and requirements.

For the data collection, and analysis and interpretation of results, scientific methods as well as techniques and instruments were used. The techniques used were the interview and the observation and the instruments were a questionnaire and an observation sheet. The questionnaire was applied to the academic and administrative staff that uses telephone network.

For the proper functioning of our IP telephony system, a Raspberry Pi B hardware and a µelastix software were used. They are free to use and were considered with the purpose of meeting the needs of the AEIRNNR area which does not have landline IP telephony for each professor. Furthermore, the equipment that works with the older IP telephony software was used. The µelastix software has friendly interfaces to understand more about the basic settings which are necessary for a telephony system. It involves the creation of extensions, ivr's, etc. This application was designed in an administrator-friendly manner that can provide immediate solutions.

In the results section the applied process for the improvement of the telephony network is explained, also the µelastix and voip settings and the different tests done to the telephony network as well as the use of tools for the results of the quality service are described.

Based on the obtained results, it was concluded that the voice over IP used by the academic and administrative staff of the AEIRNNR of the UNL was successful and helped to the communication among them.



Índice de Contenidos

PORTADA	I
CERTIFICADO.....	II
AUTORÍA.....	III
CARTA DE AUTORIZACIÓN	IV
AGRADECIMIENTO	V
DEDICATORIA	VI
a. Título	1
b. Resumen	VIII
Summary	IX
Índice de Contenidos	X
Índice de tablas.....	XII
c. Introducción	1
d. Revisión de Literatura.....	2
1.TELEFONÍA VOZ SOBRE IP.....	2
1.1. Introducción.....	2
1.2. Definición de Voz sobre Ip.....	2
1.2.1. Características de Voz sobre IP	3
1.2.2. Ventajas VoIP (Voz sobre IP).....	4
1.2.3. Elementos Implicados.....	4
1.3. Tecnologías Voz sobre IP	5
1.3.1. Códec.....	5
1.3.2. Protocolos.....	5
1.3.3. Funciones de SIP	6
1.4. Arquitectura de SIP	7
2.ASTERISK.....	9
2.1. PBX.....	9
2.2. Introducción Asterisk	9
2.4. ¿Qué es Asterisk?.....	11
2.5. Funcionalidad	12



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
μelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*

2.6.	Interfaces	13
2.7.	Arquitectura de Asterisk	14
2.8.	Módulos	16
2.9.	Asterisk para linux	17
3.	RASPBERRY PI	19
3.1.	Historia	19
3.2.	¿Qué es raspberry pi?	19
3.3.	Especificaciones de hardware	19
3.4.	Sistemas Operativos Soportados	20
3.5.	¿Qué podemos hacer con raspberry pi?	21
4.	TELEFONÍA MÓVIL	22
4.1.	Introducción	22
4.2.	Funcionamiento de la telefonía móvil	22
4.2.1.	Redes inalámbricas	22
4.2.2.	Protocolos	22
4.2.3.	Codificación	23
4.3.	¿Qué es un dispositivo móvil?	23
4.4.	Características	23
4.5.	¿Cuáles son los dispositivos móviles?	24
4.6.	Tipos de dispositivos móviles	24
4.7.	Aplicaciones para voip en móviles	25
4.7.1.	<i>Acrobats Groundwire</i>	25
4.7.2.	<i>3CX</i>	26
4.7.3.	<i>Bria iPhone edition</i>	26
4.7.4.	<i>Zoiper</i>	27
4.7.5.	<i>Netelip Phone</i>	28
4.7.6.	<i>Csip simple</i>	28
e.	Materiales y Métodos	30
f.	Resultados	32
g.	Discusión	84
h.	Conclusiones	88



i. Recomendaciones	89
j. Referencias	90
k. Anexos.....	91
Anexo 1.....	91
Anexo 2.....	93
Anexo 3.....	101
Anexo 4.....	102
Anexo 5.....	103

Índice de tablas

Tabla N°1 Recursos Humanos	85
Tabla N°2 Bienes	86
Tabla N°3 Materiales de oficina	86
Tabla N°4 Servicios	86
Tabla N°5 Total Recursos	87
Tabla N°6 Relación Laboral	93
Tabla N°7 Uso aplicación telefonía móvil.....	94
Tabla N°8 Comunicación Docente y Administrativo.....	95
Tabla N°9 Calidad de voz	96

Índice de Figuras

Figura 1. Encapsulamiento trama voz ip	8
Figura 2. Ejemplo de Escabilidad de Asterisk	14
Figura 3. Arquitectura de Asterisk.....	15
Figura 4. Raspberry modelo B	20
Figura 5. Aplicación Acrobist.....	25
Figura 6. Aplicación 3CX	26
Figura 7. Aplicación Brian	27
Figura 8. Aplicación Zoiper	27



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
μelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*

Figura 9. Aplicación Netelip	28
Figura 10. Aplicación Csip	29
Figura 11. Diseño de telefonía voip actual.....	32
Figura 12. Teléfono ip	33
Figura 13. Teléfono ip modelo Grandstream GXP 1400	33
Figura 14. Memoria SD.....	35
Figura 15. Adaptador de corriente	36
Figura 16. Case raspberry pi	36
Figura 17. Disipadores.....	36
Figura 18. Raspberry Pi B.....	37
Figura 19. Raspberry Pi Armado.....	37
Figura 20. Partición de la memoria sd en terminal ubuntu.....	38
Figura 21. Página oficial μelastix	39
Figura 22. Transferir archivos en partición BOOT y Roofs	40
Figura 23. Raspberry pi en funcionamiento mediante consola.	40
Figura 24. Página de ingreso μelastix.....	41
Figura 25. Menú principal μelastix	42
Figura 26. PBX y su menú	42
Figura 27. Creación de extensiones	43
Figura 29. Creación de extensiones de Sistemas.....	44
Figura 30. Creación de extensiones de Electromecánica	45
Figura 31. Creación de extensiones de Electrónica y Telecomunicaciones	45
Figura 32. Creación de extensiones de Geología	46
Figura 33. Gráfica de extensiones creadas	46
Figura 34. Creación de Audios para operadores.	47
Figura 35. Convertir Audio mp3 canal mono.....	48
Figura 36. Cargar audio a nuestro μelastix.....	48
Figura 37. Lista de IVR creados.....	49
Figura 38. Configuración IVR anuncio Sistemas.....	50
Figura 39. Configuración IVR anuncio Electromecánica.....	51
Figura 40. Configuración IVR anuncio Electrónica y Telecomunicaciones.	51
Figura 41. Configuración IVR anuncio Geología Ambiental.....	52



Figura 42. Smartphone diferentes modelos	52
Figura 43. Entrar al Play Store de nuestro Smartphone	53
Figura 44. Aplicaciones de Play store.....	53
Figura 45. Top gratis de nuestro Play store	54
Figura 46. Aplicación Zoiper IAX SIP VOIP	54
Figura 47. Instalar la Aplicación Zoiper.....	55
Figura 48. Ingresar a la aplicación Zoiper.....	55
Figura 49. Ajustes Zoiper	56
Figura 50. Opción Cuentas	56
Figura 51. Agregar cuenta SIP.....	57
Figura 52. Configuración de nuestra extensión dentro de la Aplicación Zoiper.	57
Figura 53. Guardar Cambios en nuestra configuración.	58
Figura 54. Conectando en PBX	58
Figura 55. Acceder a Putty	59
Figura 56. Datos requeridos para acceder consola raspberry pi.	59
Figura 57. Consola raspberry pi.....	60
Figura 58. Ingreso a la consola de asterisk.	60
Figura 59. Extensiones conectadas a central telefónica móvil.....	61
Figura 60. Código generado por consola en asterisk para extensión 101.	61
Figura 61. Código generado por consola en asterisk para extensión 113	62
Figura 62. Código generado al llamar a nuestro CALL CENTER.	62
Figura 63. Código generado al llamar de la extensión 101 a la extensión 113.	63
Figura 64. Código generado al terminar llamada.....	64
Figura 65. Diseño a implementar la telefonía móvil.	65
Figura 66. Equipos necesarios para la simulación.....	66
Figura 67. Crear Troncal IAX2 llamada Valparaiso.....	67
Figura 68. Código para enlazar al elastix del raspberry pi.....	68
Figura 69. Configuración Ruta de salida ValparaisoExten.....	69
Figura 70. Crear Troncal IAX2 llamada Vinadelmar.....	69
Figura 71. Código para enlazar al μelastix del servidor telefonía fija.....	70
Figura 72. Configuración Ruta de salida VinadelmarExten.....	71
Figura 73. Troncal Valparaiso y Vinadelmar enlazadas.....	72



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
μelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*

Figura 74. Ingreso a la interface del teléfono Grandstream GXP 1400.....	72
Figura 75. Estado de nuestro teléfono Grandstream GXP 1400.....	73
Figura 76. Entrar Account 1.	73
Figura 77. Configuración de nuestro Account 1.....	74
Figura 78. Account 1 habilitado.....	74
Figura 79. Teléfono Grandstream habilitado en nuestro μelastix de telefonía fija.	75
Figura 80. Llamada entre extensión 113 a 101.....	76
Figura 81. Llamada entre extensión 101 a 113.....	76
Figura 82. Prueba llamada entre dispositivos móviles.	77
Figura 83. Llamada entre el teléfono grandstream gxp 1400 y dispositivo móvil.	77
Figura 84. Código generado troncal Valparaíso y extensión 113	78
Figura 85. Llamada terminada entre ambos servidores.....	78
Figura 86. Llamada entre dispositivo móvil y teléfono grandstream gxp 1400.	79
Figura 87. Código generado troncal Vinadelmar y extensión 101.....	79
Figura 88. Llamada terminada entre dispositivo móvil y teléfono grandstream gxp 1400.	80
Figura 89. Llamadas realizadas extensión 101.....	81
Figura 90. Gráfica de llamadas extensión 101.....	81
Figura 91. Tiempo de duración de cada llamada realizada de la extensión 101.	82
Figura 92. Gráfica pastel de cada llamada a la extensión 101.	82
Figura 93. Registro de llamadas no contestadas cuando ha llamado la extensión 101.....	83

Introducción

En la actualidad la Universidad Nacional de Loja brinda un servicio de calidad en telefonía voz sobre ip fija, por lo cual requieren adecuar sistemas acordes a sus necesidades. Un punto muy importante en el AERNNR de la UNL, es el uso de la telefonía, ya que ayuda a interactuar entre sus administrativos y los docentes, o viceversa.

En el AEIRNNR de la Universidad Nacional de Loja no utilizan telefonía móvil voz sobre ip, por lo cual encontramos problemas en la comunicación de administrativos y docentes.

El objetivo principal de este proyecto es Implantar telefonía móvil Voz sobre IP, utilizando la herramienta µelastix en un raspberry pi en la Universidad Nacional de Loja en el AEIRNNR, realizando un análisis para encontrar falencias y necesidades.

La telefonía móvil es la comunicación a través de dispositivos, que no están conectados mediante cables, la mayoría de las personas en la actualidad cuentan con un Smartphone, el cual es accesible por sus bajos costos, este dispositivo móvil nos permitirá tener nuestro propio teléfono ip personal, con sólo bajar la aplicación Zoiper.

El raspberry pi es un hardware libre que cuenta con la capacidad suficiente para la instalación de nuestro sistema µelastix basado en arm, que tiene los módulos necesarios como creación de extensiones sip, ivr, entre otros.

Para resolver estos problemas de comunicación se aplicó encuestas a docentes y administrativos del AEIRNNR de la UNL, por otra parte también, se analizó, diseñó, configuró e implantó una herramienta µelastix en un raspberry pi B, conectándose con otro servidor µelastix ya existente en el AEIRNNR de la UNL que cuenta con voz sobre ip fija y en los dispositivos móviles se utilizó software gratuito zoiper, que nos ayudó a resolver, nuestro objetivo principal.



c. Revisión de Literatura

1. TELEFONÍA VOZ SOBRE IP

1.1. Introducción

La historia de la telefonía IP está en sus primeras etapas. Todo empezó solo unos años atrás, en 1995, cuando VocalTec lanzó su primer teléfono Internet. Antes de eso, la telefonía IP fue un campo que atraía el interés principalmente de investigadores; pero ya que la comunicación de voz sobre Internet ha probado ser no solo posible sino viable comercialmente, muchas son las empresas que han ingresado al mercado de telefonía VoIP (voice over Internet protocol), tratando de ser los líderes.

Las principales ventajas de la telefonía IP son la simplificación de la infraestructura de comunicaciones en la empresa, la integración de las diferentes sedes y trabajadores móviles de la organización en un sistema unificado de telefonía con gestión centralizada, llamadas internas gratuitas, plan de numeración integrado y optimización de las líneas de comunicación.

El crecimiento y fuerte implantación de las redes IP, tanto en local como en remoto, el desarrollo de técnicas avanzadas de digitalización de voz, mecanismos de control y priorización de tráfico, protocolos de transmisión en tiempo real, así como el estudio de nuevos estándares que permitan la calidad de servicio en redes IP, han creado un entorno donde es posible transmitir telefonía sobre IP lo que no significará en modo alguno la desaparición de las redes telefónicas modo circuito, sino que habrá, al menos temporalmente, una fase de coexistencia entre ambas.[1]

1.2. Definición de Voz sobre Ip

VoIP proviene del inglés Voice Over Internet Protocol, que significa "voz sobre un protocolo de internet".

Voz sobre Protocolo de Internet, también llamado Voz sobre IP, Voz IP, VozIP, VoIP, es un grupo de recursos que hacen posible que la señal de voz viaje a través de Internet empleando un protocolo IP (Protocolo de Internet). Esto significa que se envía la señal de voz en forma digital, en paquetes de datos, en lugar de enviarla en forma analógica



a través de circuitos utilizables sólo por telefonía convencional como la Red Telefónica Pública Conmutada.

Los Protocolos que se usan para enviar las señales de voz sobre la red IP se conocen como protocolos de Voz sobre IP o protocolos IP. Estos pueden verse como aplicaciones comerciales de la “Red experimental de Protocolo de Voz”. [1]

1.2.1. Características de Voz sobre IP

- Configuración automática de terminales.
- Salas de conferencias.
- Llamadas a grupos.
- Videoconferencia.
- Buzón de voz con integración con correo electrónico.
- Soporte para voz y vídeo a través de protocolo SIP.
- Menús IVR (Interactive Voice Response).
- Estadísticas de uso en web.
- Panel de control de administración a través de web.
- Panel de control para usuarios a través de web.
- Calidad de voz de alta definición.
- Grabación de conversaciones.
- Monitorización de llamadas.
- Desvío de llamadas programable.
- Función follow-me.
- Call parking.
- Cifrado de las comunicaciones.
- Alta disponibilidad.
- SIP trunking.
- Soporte de carriers SIP.
- Algoritmo de Least Cost Routing.
- Fax electrónico por email.
- Síntesis de voz multiidioma.
- Importación automática de tarifas de compra.



- Gestión de tarifas de venta.
- Cálculo de costes y exportación por cliente. [2]
- Gráficas de rendimiento y estadísticas.
- Multiidioma

1.2.2. Ventajas VoIP (Voz sobre IP)

- Reducir los gastos de desplazamiento y formación, mediante el uso de videoconferencias y conferencias en línea.
- Actualizar su sistema telefónico de acuerdo a sus necesidades.
- Tener un número de teléfono que suena a la vez en varios dispositivos, para ayudar a sus empleados a estar conectados entre sí y con sus clientes.
- Reducir sus gastos telefónicos.
- Utilizar una sola red para voz y datos, simplificando la gestión y reduciendo costes.
- Acceder a las funciones de su sistema telefónico en casa o bien en las oficinas de sus clientes, en aeropuertos, hoteles o en cualquier parte donde haya una conexión de banda ancha. [3]

1.2.3. Elementos Implicados

La telefonía IP engloba todo el conjunto de servicios VoIP, incluidos:

- La interconexión de teléfonos VoIP para las comunicaciones.
- Servicios relacionados como facturación y planes de acceso telefónico.
- Funciones básicas, como conferencia, transferencia, reenvío de llamadas y llamada en espera.

Las comunicaciones IP incluyen aplicaciones comerciales que mejoran las comunicaciones en un teléfono VoIP, ofreciendo funciones tales como:

- Mensajería unificada.
- Centros de atención y manejo de contactos integrados.
- Conferencia multimedia con voz, datos y video.

Las Comunicaciones Unificadas elevan a las comunicaciones IP a un nivel superior al utilizar tecnologías SIP (Protocolo de inicio de sesión) y de presencia junto con soluciones de movilidad, con el fin de unificar y simplificar todas las formas de



comunicación en un teléfono VoIP, con independencia del lugar, tiempo o dispositivo.
[3]

1.3. Tecnologías Voz sobre IP

Protocolo: Es el 'lenguaje' que se utiliza para negociar y establecer las comunicaciones de voz sobre IP.

Códec: Es la forma de digitalizar la voz humana para ser enviada por las redes de datos. Algunos ejemplos: G.711, G729A, GSM, iLBC, Speex, G.723. [4]

1.3.1. Códec

Los códec's se utilizan para transformarla la señal de voz analógica en una versión digital. Los softphones, hardphones, centralitas IP, soportan una serie de códec's cada uno. Cuando hablan entre si negocian un códec común.

Aspectos a tener en cuenta por códec:

- Calidad de sonido
- Ancho de banda requerido
- Requisitos de computación.

1.3.2. Protocolos

Los protocolos que se utilizan en las redes de voz sobre IP son: SIP, H323, IAX2, MGCP, entre otros. Estos protocolos tienen interfaz abierta y estándares definidos y cuentan con una buena infraestructura de paquetes.

1. Protocolo IAX2: Inter Asterisk exchange. Creado y estandarizado por la centralita Asterisk, es robusto, lleno de novedades y muy simple en comparación con otros protocolos. Permite manejar una gran cantidad de *códec's* y un gran número de *streams*, lo que significa que puede ser utilizado para transportar virtualmente cualquier tipo de dato. Utiliza el puerto 4569 UDP. [5]

Características Principales:

- Media y señalización por el mismo flujo de datos.
- Trunking
- Cifrado



2. **Protocolo H.323:** Es un estándar del ITU (International Telecommunications Union) que provee especificaciones para ordenadores, sistemas y servicios multimedia por redes que no proveen calidad de servicio. Es un protocolo muy complejo que fue originalmente pensado para videoconferencias. Este provee especificaciones para conferencias interactivas en tiempo real, para compartir data y audio como aplicaciones VoIP. Actualmente H323 incorpora muchos protocolos individuales que fueron desarrollados para aplicaciones específicas. [1]
3. **Protocolo SIP:** Session Initiation Protocol. Es un protocolo mucho más lineal, desarrollado específicamente para aplicaciones de VoIP. Más chicas y más eficientes que H.323. SIP toma ventaja de los protocolos existentes para manejar ciertas partes del proceso.

Algunas de las características claves que SIP ofrece son:

- Resolución de direcciones, mapeo de nombres y redirección de llamadas.
- Descubrimiento dinámico de las capacidades media del endpoint, por uso del Protocolo de Descripción de Sesión (SDP).
- Descubrimiento dinámico de la disponibilidad del endpoint.
- Originación y administración de la sesión entre el host y los endpoints. [6]

1.3.3. Funciones de SIP

El protocolo SIP actúa de forma transparente, permitiendo el mapeo de nombres y la redirección de servicios ofreciendo así la implementación de la IN (Intelligent Network) de la PSTN o RTC.

Para conseguir los servicios de la IN el protocolo SIP dispone de distintas funciones. A continuación, se enumeran las más importantes:

- *Localización de usuarios* (SIP proporciona soporte para la movilidad).
- *Capacidades de usuario* (SIP permite la negociación de parámetros).
- *Disponibilidad del usuario*
- *Establecimiento y mantenimiento* de una sesión.

En definitiva, el protocolo SIP permite la interacción entre dispositivos, cosa que se consigue con distintos tipos de mensajes propios del protocolo que abarca esta sección. Dichos mensajes proporcionan capacidades para registrar y/o invitar un



usuario a una sesión, negociar los parámetros de una sesión, establecer una comunicación entre dos a más dispositivos y, por último, finalizar sesiones. [6]

1.4. Arquitectura de SIP

El estándar define varios componentes SIP y hay varias formas de implementarlos en un sistema de control de llamadas.

- Agentes de Usuario: Son los puntos extremos del protocolo, es decir son los que emiten y consumen los mensajes del protocolo SIP. Un videoteléfono, un teléfono, un cliente de software (softphone) y cualquier otro dispositivo similar es para el protocolo SIP un agente de usuario. Todos los agentes de usuario se comportan como clientes (UAC: User Agent Clients) y como servidores (UAS: User Agent Servers).
- Servidores de Registro (Registrar): Al iniciarse el agente de usuario SIP envía una petición con el método REGISTER a un Servidor de Registro, informando a qué dirección física debe asociarse la dirección lógica del usuario (binding). Esta asociación tiene un período de vigencia y si no es renovada, caduca. También puede terminarse mediante el método Deregister. El protocolo SIP no determina la forma en que se debe gestionar los registros.
- Servidores Intermediarios (Proxy) y de Redirección (Redirectors): Para encaminar un mensaje entre un UAC y un UAS normalmente se recurre a los servidores (aunque puede utilizarse una estrategia tipo p2p). Estos servidores a su vez se sirven del sistema DNS para localizar los dominios y pueden actuar de dos maneras:
 - Como intermediario, encaminando el mensaje hacia destino
 - Como redirector, generando una respuesta que indica al remitente la dirección del destino o de otro servidor que lo acerque al destino.
 - La principal diferencia es que el servidor intermediario forma parte de la comunicación, mientras que el servidor de redirección una vez que indica al UAC cómo encaminar el mensaje ya no interviene más.
 - Un mismo servidor puede actuar como redirector o como intermediario dependiendo de la situación.

1.5. Encapsulamiento de una trama VoIP

Una vez que la llamada ha sido establecida, la voz será digitalizada y entonces transmitida a través de la red en tramas IP. Las muestras de voz son primero encapsuladas en RTP (protocolo de transporte en tiempo real) y luego en UDP (protocolo de datagrama de usuario) antes de ser transmitidas en una trama IP. En la siguiente (figura 1. Encapsulamiento trama voz ip) se muestra un ejemplo de una trama VoIP sobre una red LAN y WAN

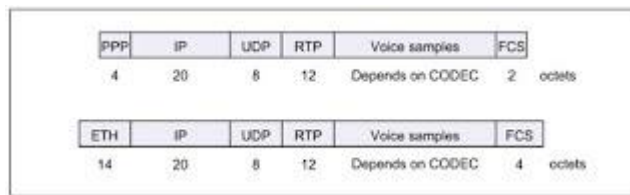


Figura 1. Encapsulamiento trama voz ip

Por ejemplo, si el códec usado es G.711 y el período de paquetización es 20 ms, la carga útil será de 160 bytes. Esto resultará en una trama total de 206 bytes en una red WAN y en 218 bytes en una red LAN. [7]



2. ASTERISK

2.1. PBX

Antes de adentrarnos en la definición o descripción de in IP PBX, conviene entender qué significa cada sigla por separado y luego sabremos comprender mejor sus funciones y demás características.

Por un lado, tenemos que PBX es la abreviatura de Private Branch Exchange, que no es más que la red privada de telefonía que se utiliza dentro de una empresa. Los usuarios de esta red privada comparten una cierta cantidad de líneas externas para hacer llamadas fuera de la PBX. La mayoría de las medianas a grandes compañías usan esta clase de sistemas porque sencillamente es mucho más económico que conectar la línea externa a cada uno de los teléfonos internos de la empresa. Además, es muy fácil comunicarse entre los teléfonos de la misma empresa que estén conectados a la misma PBX, ya que comúnmente se marca un número de 3 o 4 cifras, y su uso, como es interno y dentro de la plataforma, es totalmente gratuito.

Con respecto a IP, es la abreviatura de Internet Protocol, que sumado a PBX nos queda Internet Protocol Private Branch Exchange, también podemos encontrarlo como IPPBX ó IP/PBX. La función de este aparato es la misma, sólo que, en lugar de utilizar telefonía convencional, es un sistema que soporta VoIP. La ventaja del VoIP es que es una tecnología que utiliza Internet (redes LAN o WAN) como vía para llevar a cabo la llamada, por lo que la información se la digitaliza y se transmite en forma de paquetes de data digital. Con estas plataformas se pueden usar teléfonos IP, softphones o teléfonos convencionales (depende del fabricante) y funciona igual que una PBX convencional. [8]

2.2. Introducción Asterisk

Asterisk es el mayor proyecto de software libre diseñado para la integración y unificación de los sistemas de comunicaciones conocidos.

Originalmente fue concebido como una plataforma para la generación de un sistema PBX, pero con el tiempo ha ido evolucionando a otro tipo de usos, como Pasarelas VoIP,



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
µelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*

sistemas integrales para call-centers, salas de conferencias, buzones de voz, y todo tipo de aplicaciones que tengan relación con las comunicaciones en tiempo real.

Comparativamente Asterisk es para el mundo de las comunicaciones lo mismo que sería Apache para el mundo de las aplicaciones web. Apache es un servidor web, y Asterisk es un servidor de comunicaciones.

Asterisk es una plataforma de comunicaciones basada en la filosofía Open Source, es capaz de convertir una computadora común en un completo servidor de comunicaciones. [9]

2.3. Evolución de Asterisk

El desarrollo de Asterisk fue iniciado en el año 1999 por Mark Spencer. En aquel momento ya existían otros proyectos de telefonía basados en la ideología Open Source. Pero esos proyectos estaban realmente enfocados a la telefonía a nivel software, basados en la búsqueda de establecer comunicaciones entre ordenadores a través de los sistemas de audio. La diferencia de Asterisk con estos proyectos, era la idea preconcebida, de entrelazar por primera vez la red de telefonía común (PSTN) con un ordenador.

Y para ello se formó una alianza con un proyecto de telefonía llamado Zaptel iniciado por Jim Dixon. La idea del proyecto Zaptel, era la posibilidad de diseñar tarjetas específicas para convertir la señal analógica que provenía de la PSTN a una señal digital, y ahorrar costes en la construcción de dispositivos de telefonía y audio avanzados, gracias a la tremenda reducción de costes que sufrían año tras año los procesadores. A través de estos, cada vez más potentes, y baratos, se podrían procesar una o varias señales de audio digital (DSP) sin gran dificultad y poder paralelamente construir teléfonos con ese coste reducido.

Hoy en día el proyecto Zaptel, fue integrado totalmente en Asterisk, y Asterisk patrocinado por una empresa que construye dispositivos de telefonía digital, llamada Digium, y se ha acomodado como una parte más de Asterisk con un nuevo nombre: DAHDI (son las siglas de Digium Asterisk Hardware Device Interface).



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
μelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*

La primera versión estable surgió casi 5 años después, Asterisk 1.0. A partir de aquí el sistema de versiones ha evolucionado de la siguiente forma:

- Asterisk 1.0 – 2004
- Asterisk 1.2 – 2005
- Asterisk 1.4 – 2006
- Asterisk 1.6.0 – 2008
- Asterisk 1.6.1 – 2009
- Asterisk 1.6.2 – 2009
- Asterisk 1.8 – 2010
- Asterisk 1.10 – 2011 (Nuevo Nombre Asterisk 10) [9]

2.4. ¿Qué es Asterisk?

Realmente esta pregunta puede ir mucho más allá de la definición común que se establece para el sistema.

Existen ciertas consideraciones que hacen a Asterisk lo que realmente es en la actualidad, y una de las principales surge de la propia naturaleza del sistema: La filosofía Open Source.

Considerando Asterisk como una plataforma integral de comunicaciones, podría considerarse la más importante, y ha resultado como única por muchos años en un entorno, donde todos los sistemas de comunicación eran totalmente privativos. Aunque con el tiempo, fueron sacando interfaces comúnmente conocidos como CTI para la integración de sistemas de terceros para cumplir funciones muy específicas, la potencia de estas interfaces era bastante limitada dado que el núcleo de los sistemas privados, permanencia cerrado al público.

Con Asterisk se sienta un precedente desde el momento que el código es de libre acceso, modificación y ampliación, dado que múltiples colectivos pudieron ampliar sus necesidades y adaptarlas a lo que venían buscando en un momento concreto, que un sistema de comunicaciones pudiera ofrecerles, dentro de sus extensivas bondades. [9]



2.5. Funcionalidad

Asterisk dispone de todas las funcionalidades de las grandes centralitas propietarias (Cisco, Avaya, Alcatel, Siemens, etc.). Desde las más básicas (desvíos, capturas, transferencias, multi-conferencias,) hasta las más avanzadas (Buzones de voz, IVR, CTI, ACD...).

Escalabilidad

El sistema puede dar servicio desde 10 usuarios en una sede de una pequeña empresa, hasta 10.000 de una multinacional repartidos en múltiples sedes.

Competitividad en coste

No solo por ser un sistema de código abierto (Open Source) sino gracias a su arquitectura hardware: utiliza plataforma servidor estándar (de propósito no específico) y tarjetas PCI para los interfaces de telefonía, que por la competencia del mercado se han ido abaratando progresivamente.

Interoperabilidad y Flexibilidad

Asterisk ha incorporado la mayoría de estándares de telefonía del mercado, tanto los tradicionales (TDM) con el soporte de puertos de interfaz analógicos (FXS y FXO) y RDSI (básicos y primarios), como los de telefonía IP (SIP, H.323, MGCP, SCCP/Skinny). Eso le permite conectarse a las redes públicas de telefonía tradicional e integrarse fácilmente con centralitas tradicionales (no IP) y otras centralitas IP. Asterisk puede funcionar como cualquier centralita tradicional, e incorpora todas sus funcionalidades. Enumeramos las más importantes:

- Conexión **con líneas de telefonía tradicional**, mediante interfaces tipo analógico (FXO) para líneas de teléfono fijo o bien móvil y RDSI (BRI o PRI).
- Soporte de **extensiones analógicas**, bien para terminales telefónicos analógicos, terminales DECT o bien equipos de fax.
- Soporte de **líneas (trunks) IP**: SIP, H323 o IAX.
- Soporte de **extensiones IP**: SIP, SCCP, MGCP, H323 o IAX.
- Música en Espera basada en archivos MP3 y similar.
- Funciones básicas de usuario:
 - Transferencias (directa o consultiva)



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
μelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*

- Desvíos
- Capturas (de grupo o de extensión)
- Conferencia múltiple
- Aparcamiento de llamadas (*Call parking*)
- Llamada directa a extensión
- Retrollamada - *Callback* (llamada automática cuando disponible)
- *Paging* - Megafonía a través del altavoz del teléfono
- DND

El sistema incorpora así mismo muchísimas funcionalidades avanzadas que tendrían un elevado coste en sistemas tradicionales propietarios. Enumeramos sólo los más importantes:

- **Buzón de Voz:** sistema de contestador automático personalizado por usuario. Se integra con el sistema de directorio (LDAP) y con el email.
- **Sistema de Audioconferencias:** Sistema que permite la conexión remota de diferentes usuarios que quieren mantener una reunión virtual y suministra la correcta gestión y control de los usuarios que se incorporan a ella.
- **IVR:** Operadora Automática. Sistema automatizado de respuesta que permite redirigir las llamadas entrantes en función de las opciones seleccionadas por el llamante.
- **Informes detallados de llamadas (CDR):** Detalle de llamadas realizadas/recibidas por extensión, para imputación de costes departamentales, por cliente o incluso para facturación.
- **ACD:** Sistema Automático de Distribución de Llamadas entrantes. Pensado para Centros de Llamadas para atención comercial o soporte técnico.
- **CTI:** Integración con sistemas de gestión comercial o de atención a cliente (CRM).
- **IPCC (*IP Contact Center*):** Integración con sistemas avanzados de gestión de centros de llamadas, vía soluciones abiertas o propietarias.

2.6. Interfaces

Hasta ahora se ha mencionado a Asterisk como una PBX por software que puede correr sobre cualquier ordenador estándar, sin embargo, parte importante de las PBX es la posible interactuar con troncales analógicas y está claro que un ordenador estándar no

cuenta con puertos físicos que nos permitan administrar estos canales. Según Bryant et al. (2013), cuando el proyecto Asterisk nació, habían otros posibles proyectos de telefonía de código abierto, uno de esos proyectos era la asociación de Asterisk con Zaptel Telephony Project para el desarrollo de interfaces PSTN (public switched telephone network).

Hoy en día se ha incrementado la cantidad de fabricantes como también la variedad de tipos de interfaces, entre los más destacados podemos mencionar a Digium, OpenVox, Pika, Sangoma, Dialogic, etc.

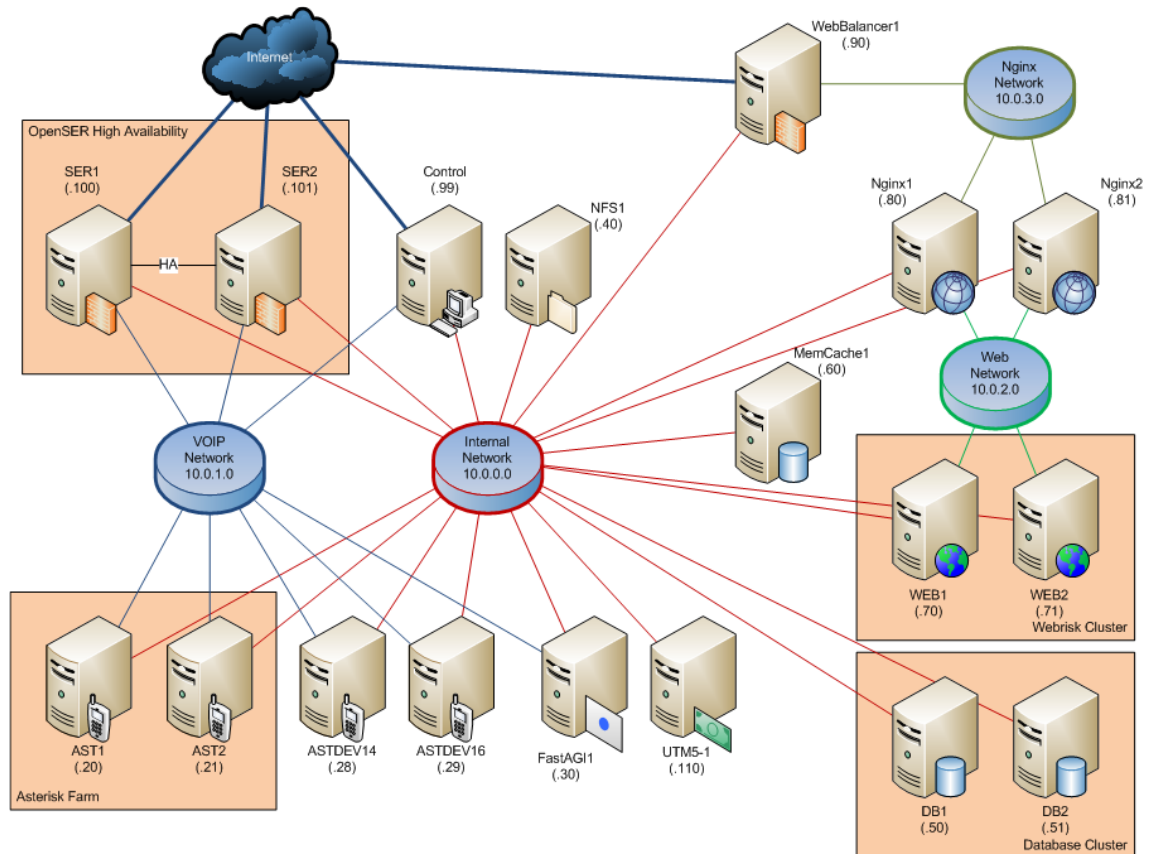


Figura 2. Ejemplo de Escalabilidad de Asterisk

2.7. Arquitectura de Asterisk

La arquitectura de Asterisk está basada en un sistema modular, que depende del núcleo principal del sistema.

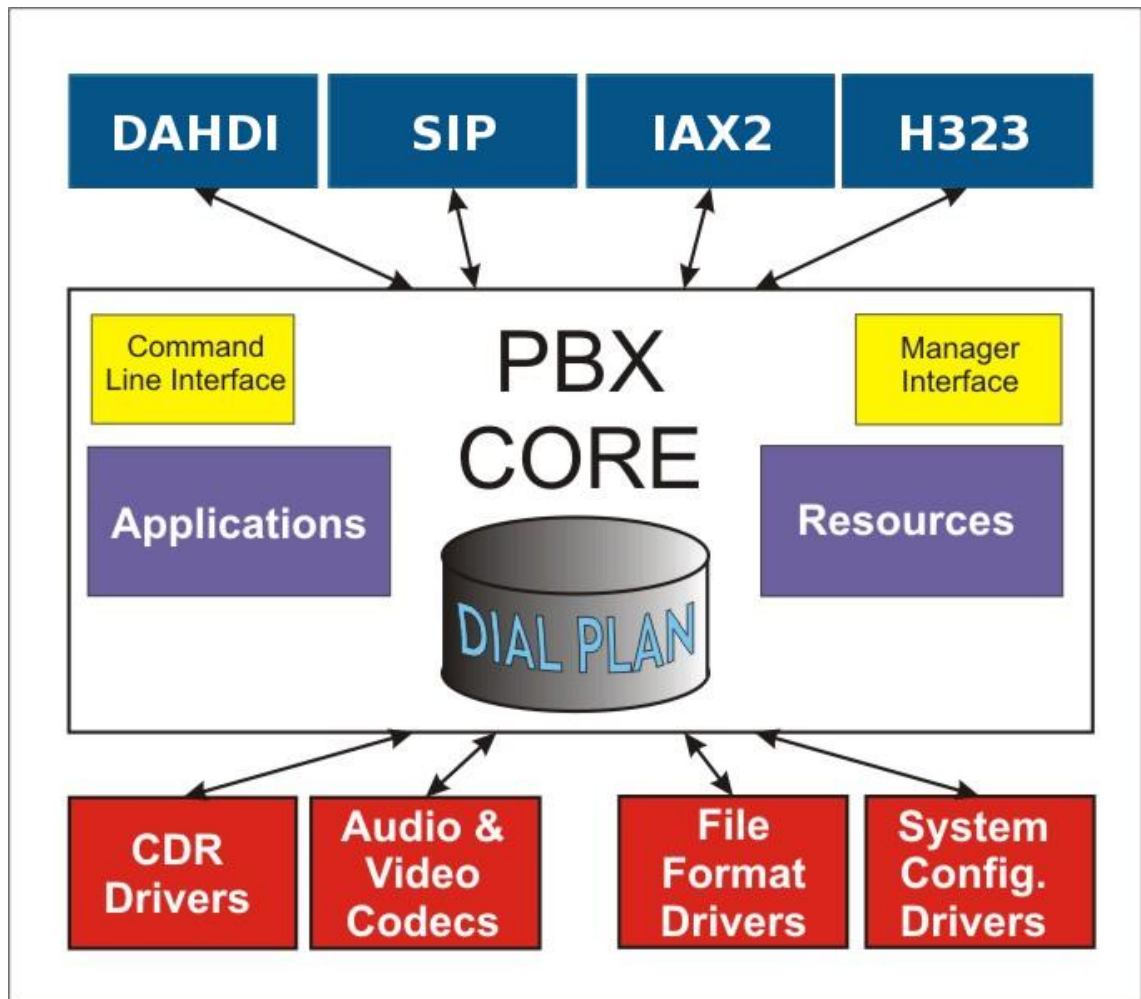


Figura 3. Arquitectura de Asterisk

El núcleo del sistema principalmente se basa en cuatro componentes:

- Gestión de Módulos
- Temporizador de Sistema
- Gestión de Canales
- Interfaces de Sistema

La estructura de Asterisk es totalmente independiente de los dispositivos, troncales, y demás útiles externos al mismo. En la gran parte de las PBX tradicionales, es muy común observar el hecho que existen estructuras lógicas para diferenciar entre los teléfonos y otros dispositivos de telefonía como faxes, y por otro lado líneas entrantes como troncales primarios, líneas RDSI, etc. En Asterisk este concepto no existe, todo entra al sistema como un canal genérico y luego es gestionado de forma integral.



Aunque sean diferenciados, todos se gestionan de la misma forma por eso, incluso recursos externos pueden llegar a ser manejados dentro de las posibilidades, de la misma forma, que una extensión SIP interna.

A priori esto puede parecer bastante obvio, pero el verdadero potencial se demuestra cuando empezamos a trabajar con estos canales dentro del plan de marcación.

2.8. Módulos

Como se ha mencionado anteriormente la estructura de Asterisk es modular. Es como una estructura de legos en donde cada lego (modulo) cumple una función en específico y al unirlos todos obtenemos un sistema con una amplia diversidad de servicios, desde los más simples hasta los más complejos (Bryant et al., 2013). [10]

Según García (2008) [11], dependiendo de su función los módulos pueden agruparse en los siguientes grupos:

- Núcleo de PBX: Aquí se agrupan todos los módulos relacionados con un sistema de conmutación de intercambio de rama privada o bien las capacidades esenciales de una PBX.
- Lanzador de aplicaciones: Este grupo contiene aplicaciones que mejoran servicios y van un poco más allá de los servicios primitivos de una PBX.
Ejemplo de esto puede ser voicemail, lista de directorios, mensajería SMS, grabación de llamadas, etc.
- Códecs y traductor de códecs: Contiene todos los codificadores y decodificadores con los que dispone la PBX para comunicarse con el resto de dispositivos. Estos son indispensables para realizar un eficiente uso del ancho de banda (G723, G729, iLBC, GSM, LPC10).
- Gestor de entra y salida: Contiene organización de tareas de bajo nivel y sistemas de operación. Su función primordial es mantener un buen rendimiento ante cambios en la demanda de servicios.
- Cargador de módulos API (Application Programming Interface): Aquí se agrupan los módulos que permiten el uso flexible de las aplicaciones para ejecutar las acciones requeridas. Además, permite el desarrollo de aplicaciones de acuerdo a requerimientos específicos. Se clasifican en 4 tipos:
 - Canal API
 - Aplicación API



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
µelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*

- Traductor de códec API
- Formato de archivo API

Canales: Contiene todos los módulos necesarios para la utilización de diversos canales de comunicación (FXS, ISDN, Bluetooth, ITU, PSTN, etc.).

Protocolos: Contiene información de señalización al momento de establecer comunicación y además determinar punto de destino (SIP, H323, IAX, MGCP, etc.). En ocasiones maneja aspectos relacionados con tono, tiempo de repique, identificación de llamadas, etc. [11]

2.9. Asterisk para linux

La página de referencia es <http://www.asterisk.org/>

Nos descargamos la versión 1.6.2.20 y lo descomprimos

1) Instalar Asterisk

```
# tar -zxvf asterisk-1.6.2.20.tar.gz  
# rm -f asterisk-1.6.2.20.tar.gz  
# cd asterisk-1.6.2.20
```

2) Ejecutar “make”

Suponiendo que todo ha ido correctamente.

3) Ejecutar “make install”

Si es la primera vez que instalas la centralita Asterisk es recomendable instalar los ejemplos con el comando.

4) “make samples”

Pero recuerda que este comando sobrescribirá todos los archivos de configuración que ya tengas.

Finalmente puedes arrancar el Asterisk con el comando:

```
# asterisk -vvvc
```

Verás un montón de mensajes en la pantalla cuando Asterisk se inicializa. (las vvvc pertenecen al modo “very very verbose” y la c a que nos mostrará al final una línea de comandos en forma consola).

```
*CLI>
```

A partir de este momento ya está Asterisk instalado y funcionando. Se puede utilizar el comando “help” para ayuda.



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
µelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*

También puedes utilizar el comando “man asterisk” en la línea de comandos de linux para obtener detalles de cómo arrancar y parar el servidor Asterisk.

Los ficheros de configuración de Asterisk se habrán instalado en el directorio /etc/asterisk donde podrás encontrar un montón de información.

Vamos a comprobar que funciona:

Configuramos un softphone como el SJPhone o X-Lite, para poder acceder a nuestro propio Asterisk. La configuración que hemos hecho trae dos usuarios por defecto que podemos utilizar:

A: usuario: 3000 password=cualquiera vale

B: usuario: 3001 password=cualquiera vale

Una vez que lo tenemos configurado y el usuario se ha registrado correctamente en nuestro servidor podemos llamar a algunos números de prueba que vienen por defecto en el plan de numeración:

1000 – Menú principal

1234 – Pasar llamada a la consola (veréis en la consola la llamada)

1235 – Contestador automático de la consola

1236 – LLamar a la consola

3000 – LLamar al usuario SIP 3000

3001 – LLamar al usuario SIP 3001

500 – Llamar a Digium

600 – Prueba de eco

8500 – Menú del contestador

99990 Test AGI

99991 Test EAGI

99992 Dice la hora

99999 Suena música de manera infinita

700 Deja aparcada la llamada

701-720 Llamadas aparcadas

Una buena prueba en este momento es configurar 2 softphones en dos computadoras diferentes; uno con el usuario 3000 y otro con el usuario 3001 e intentar hacer una llamada entre ambos. Si funciona podemos pasar a aprender a configurar Asterisk y crear nuevos usuarios y planes de numeración.



3. RASPBERRY PI

3.1. Historia

En 2006 ya existían diseños de esta pequeña placa y portaba un microcontrolador Atmel, concretamente el ATmega644 (familia de microcontroladores que posee Arduino). Estos primeros esquemas están totalmente abiertos y se pueden acceder a ellos sin ningún tipo de problemas.

En mayo de 2009, la Fundación Raspberry Pi fue fundada en Caldecote, South Cambridgeshire, Reino Unido como una asociación caritativa que es regulada por la Comisión de Caridad de Inglaterra y Gales. Hay que destacar que esta fundación es totalmente sin ánimo de lucro, es sólo con fin académico desinteresado para que pueda la mayoría de estudiantes aprendan mediante el software libre. [13]

3.2. ¿Qué es raspberry pi?

Raspberry Pi, es un ordenador de placa reducida (SBC) de bajo coste desarrollado en Reino Unido por la Fundación Raspberry Pi, con el objetivo de estimular la enseñanza de ciencias de la computación en las escuelas.

3.3. Especificaciones de hardware

CPU: ARM1176JZF-S a 700 MHz (familia ARM11) perteneciente a la versión 6, por lo que puede soportar RISC OS

GPU: Broadcom VideoCore IV, es capaz de reproducir videos en 1080p a 30fps. Hay reviews sobre esta placa que dicen que hasta puede asimilarse a unos gráficos de la primera XBOX

RAM: Tiene una versión de 256MB (modelo A) y una de 512MB (modelo B). Comparte una parte de la memoria con la GPU

Salidas de vídeo, HDMI y audio

1/2 Puertos USB dependiendo el modelo

Puerto Ethernet (sólo el modelo B)

Consumo energético 500 mA y 2,5W en su modelo A y 700mA y 3,5W en el modelo B

Alimentación de 5V por el puerto micro USB [13]

RASPBERRY PI MODEL B

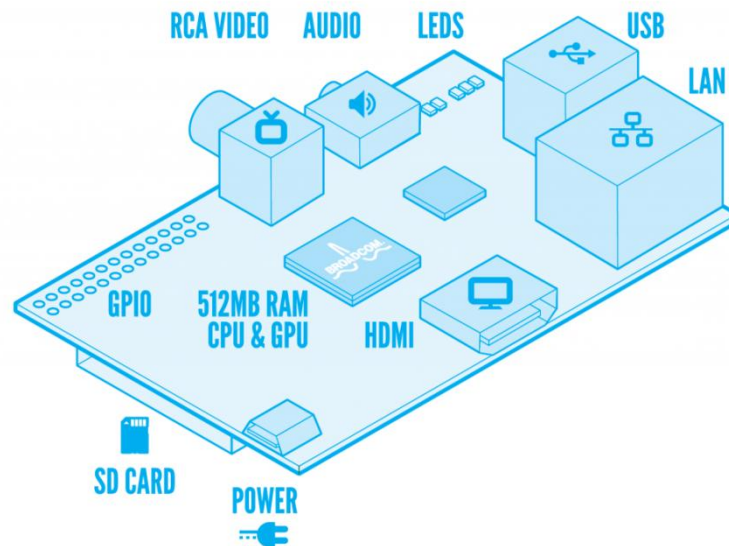


Figura 4. Raspberry modelo B

3.4. Sistemas Operativos Soportados

- Android, testeada hasta la versión 4.0
- Arch Linux ARM
- Debian Whezzy Soft-Float, versión de Debian sin soporte para coma flotante por hardware
- Firefox OS
- Gentoo Linux
- Google Chromium OS
- Kali Linux
- Open webOS
- PiBang Linux , distribución Linux derivada de Raspbian con diferente escritorio y aplicaciones
- Pidora, versión Fedora Remix optimizada
- QtonPi, distribución linux con un framework de aplicaciones multiplataforma basado en Qt framework



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
μelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*

- Raspbian, versión de Debian Wheezy para ARMv6 con soporte para coma flotante por hardware
- Slackware ARM, también conocida como ARMedslack
- OpenELEC (Media Center)
- Raspbmc (Media Center)

3.5. ¿Qué podemos hacer con raspberry pi?

3.5.1. Android y XBMC

Lo más típico que podemos hacer con nuestro pequeño ordenador es crear un Media Center con OpenELEC o con Raspbmc con el fin de poder ver nuestras películas desde cualquier sitio y sin la necesidad de tener conectado toda una torre a nuestro TV.

3.5.2. Raspberry Pi y Ambilight

Otro de los usos que está tomando fuerza, es la posibilidad de crear un sistema de luces Ambilight de bajo coste y en sencillos pasos, no tenemos más que crear una malla de LED's conectados a Raspberry Pi, ponerlos detrás de nuestro TV y ya ella se puede encargar del resto. Si queremos realizar este montaje podemos seguir los pasos proporcionados por los proyectos de PiBob o AmbiPi.

3.5.3. Domótica y Raspberry Pi

Además de todo esto también podemos crear nuestro pequeño servidor para gestionar por ejemplo las luces de casa. Nosotros recomendamos utilizar una Raspberry Pi y un Arduino y tratar siempre a Arduino como un "Actuador", ya que será el encargado de la parte más electrónica y utilizar a Raspberry como "Servidor" o cabeza pensante.



4. TELEFONÍA MÓVIL

4.1. Introducción

La telefonía celular es un sistema de comunicación telefónica totalmente inalámbrica, en este caso los sonidos se convierten en señales electromagnéticas, que viajan a través del aire, siendo recibidas y transformadas nuevamente en mensaje a través de antenas repetidoras o vía satélite. [14]

4.2. Funcionamiento de la telefonía móvil

4.2.1. Redes inalámbricas

Para poder hablar mediante VoIP a través de un terminal móvil, éste debe estar dotado de un procesador suficientemente potente que permita una conexión inalámbrica a Internet. Ya sea por Wi-Fi, HSDPA, WiMAX o EV-DO rev A, esta última es más específica para telefonía móvil y la variante rev A de EVDO permite una creación más rápida de paquetes tanto para el enlace de subida como por el de bajada.

También existe un compromiso entre precio y fiabilidad según el tipo de red inalámbrica que utiliza la conexión de VoIP para su funcionamiento. Una conexión Wifi es más económica, pero para que sea posible hay que estar bajo la cobertura de este, esto quiere decir que hay que estar en el interior de edificios. Una conexión EVDO es más cara pero la operadora garantiza una mejor cobertura y mayor velocidad de transferencia de datos por lo que la calidad será mejor.

4.2.2. Protocolos

Para establecer, modificar y finalizar conexiones entre dos usuarios se utiliza el protocolo de señalización SIP. Además, éste es capaz de determinar la ubicación de los usuarios permitiendo de esta manera más movilidad. El protocolo SIP se complementa con el protocolo RTP, éste es el que transporta los paquetes de datos de voz, es decir la conversación que enviamos a través de nuestro terminal móvil digitalizada y empaquetada mediante el protocolo IP.



La VoIP móvil también utiliza el sistema de telecomunicaciones GAN que permite al teléfono móvil operar mediante Wifi bajo cobertura de este y cuando se pasa en un dominio de cobertura de teléfono móvil operar mediante GSM.

4.2.3. Codificación

La voz se debe digitalizar y codificar para que se pueda enviar a través del protocolo IP. Se utilizan códecs como los G.711, G.722, G.723, G.728 o G.729. Estos comprimen los datos de audio para usar menos ancho de banda en la transmisión. De esta manera se permiten muchas más conexiones en un mismo canal.

Hay que tener en cuenta que la calidad de los datos transmitidos es inversamente proporcional a la compresión de los datos, si la compresión de datos es muy grande podemos perder inteligibilidad. Existe pues un compromiso entre el ancho de banda que se quiere utilizar y una calidad suficientemente buena para que haya inteligibilidad.

4.3. ¿Qué es un dispositivo móvil?

Un dispositivo móvil se puede definir como un aparato de pequeño tamaño, con algunas capacidades de procesamiento, con conexión permanente o intermitente a una red, con memoria limitada, que ha sido diseñado específicamente para una función, pero que puede llevar a cabo otras funciones más generales. [15]

4.4. Características

- Funcionalidad limitada.
- No necesariamente extensible y actualizable.
- En pocos años el usuario deberá cambiarlo.
- Más barato.
- Menos complicado en su manejo.
- Fácil de aprender su operación.
- No se requieren usuarios expertos.



4.5. ¿Cuáles son los dispositivos móviles?

Algunos de los ejemplos de estos dispositivos son los siguientes:

- Paginadores.
- Comunicadores de bolsillo.
- Internet Screen Phones.
- Sistemas de navegación de automóviles.
- Sistemas de entretenimiento.
- Sistemas de televisión e Internet (WebTV).
- Teléfonos móviles.
- Organizadores y asistentes personales digitales (Personal Digital Assistant).

4.6. Tipos de dispositivos móviles

- *Dispositivo Móvil de Datos Limitados*: (Limited Data Mobile Device): teléfonos móviles clásicos. Se caracterizan por tener una pantalla pequeña de tipo texto. Ofrecen servicios de datos generalmente limitados a SMS y acceso WAP.
- *Dispositivo Móvil de Datos Básicos*: (Basic Data Mobile Device): se caracterizan por tener una pantalla de mediano tamaño, menú o navegación basada en iconos, y ofrecer acceso a emails, lista de direcciones, SMS, y, en algunos casos, un navegador web básico. Un típico ejemplo de este tipo de dispositivos son los teléfonos inteligentes (“smartphones”).
- *Dispositivo Móvil de Datos Mejorados*: (Enhanced Data Mobile Device): se caracterizan por tener pantallas de medianas a grandes (por encima de los 240 x 120 pixels), navegación de tipo stylus, y que ofrecen las mismas características que el “Dispositivo Móvil de Datos Básicos” (Basic Data Mobile Devices) más aplicaciones nativas como aplicaciones de Microsoft Office Mobile (Word, Excel, PowerPoint) y aplicaciones corporativas usuales, en versión móvil, como Sap, portales intranet, etc. Este tipo de dispositivos incluyen los S.O. como Windows Mobile [14]

4.7. Aplicaciones para voip en móviles

Existen numerosas aplicaciones gratuitas y de pago para poder usar el servicio de telefonía IP (voip) en tu smartphone y tu tablet. El protocolo voip más popular es conocido como SIP, session initiation protocol, que es el que usamos en netelip.

4.7.1. Acrobats Groundwire

De todas las aplicaciones que hemos probado en iPhone es sin lugar a dudas la mejor de todas. La única pega es que es de pago, pero merece la pena si no quieres perder llamadas.

Permite configurar multitud de opciones, y la opción más interesante de todas es las notificaciones push. Tiene una calidad de voz excelente y no hace falta tener la aplicación corriendo en segundo plano para que las llamadas lleguen sin perderlas. Aunque tengas el móvil recién encendido sin haber abierto la APP te entrarán las llamadas de tu fijo o centralita perfectamente. Tiene soporte para auriculares bluetooth y permite grabar las llamadas.



Figura 5. Aplicación Acrobats

4.7.2. 3CX

Está disponible en Android y en iPhone. Aplicación gratuita que ofrece multitud de opciones de configuración. Una de las ventajas es que permite el uso de auriculares bluetooth. Lo malo es que a veces pierde el registro y no entran las llamadas, hay que volver a abrir la aplicación para que se registre de nuevo.

De las aplicaciones gratuitas es la mejor y la que más opciones ofrece.



Figura 6. Aplicación 3CX

4.7.3. Bria iPhone edition

Esta aplicación dispone también de una versión para ipad, que analizaremos en otra entrada del blog. Esta aplicación es de pago al igual que Groundwire. Tiene soporte para tener configuradas hasta 12 cuentas en el mismo terminal y presenta una interfaz y un diseño muy intuitivos.

También dispone soporte para auriculares bluetooth y también permite la grabación de llamadas. Podrás personalizarlo con una imagen de fondo que tú elijas.



Figura 7. Aplicación Brian

4.7.4. Zoiper

Otra aplicación gratuita disponible en el PLAY STORE, se ejecutan en una multitud de plataformas diferentes. No importa si usted está usando Mac, Linux o Windows, iPhone, Android o un navegador. Es uno de los mejores en su tipo, por su fácil manejo, poco peso y excelente calidad técnica. [16]



Figura 8. Aplicación Zoiper

4.7.5. Netelip Phone

Es nuestra aplicación propia desarrollada para Android (la de iPhone está en desarrollo) es la que mayor integración ofrece con el servicio de Netelip, ya que está pensada para nuestros clientes. Permite ver antes de cada llamada el coste por minuto del destino. También permite cambiar el identificador de llamada justo antes de llamar y ver el saldo a tiempo real. Puede grabar conversaciones, permite el envío de SMSs usando el servicio Netelip, y desde ella podrás recargar saldo Netelip de forma rápida y segura.

Otra funcionalidad interesante de esta APP es la opción de agenda virtual, que permite hacer una copia online de tus contactos y que luego podrás acceder desde otro dispositivo como un Tablet, otro móvil, etc.

No necesita configurar servidor ni cuenta sip, simplemente con las credenciales web (con los que accedes a tu panel privado) ya estará lista para emitir y recibir llamadas.



Figura 9. Aplicación Netelip

4.7.6. Csip simple

Una de las aplicaciones gratuitas para android más descargadas. Hay que tenerla en segundo plano para que las llamadas entren correctamente con el consumo de batería que supone, pero no suele perder registro. Permite grabar llamadas y configurar varios temas de interfaz. También podrás elegir el códec de audio si alguno no da la calidad adecuada.

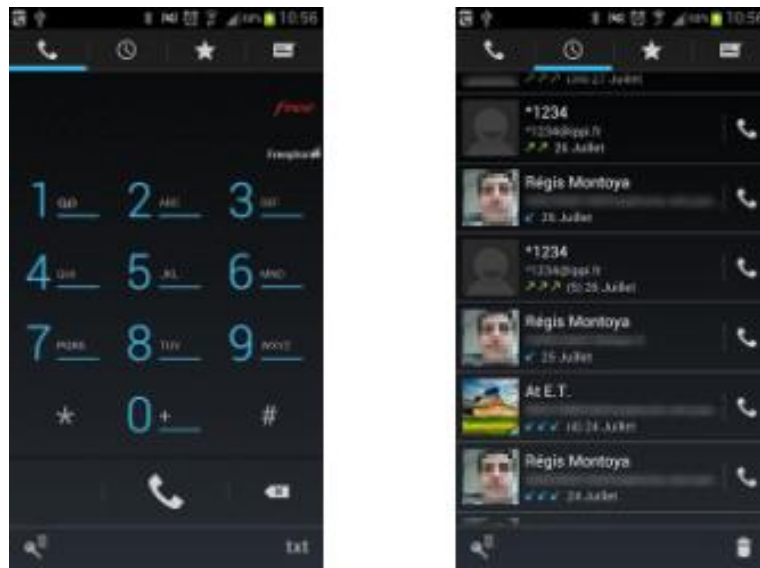


Figura 10. Aplicación Csip



d. Materiales y Métodos

1. Método Científico

El desarrollo del proyecto de investigación requiere seguir los lineamientos de ciertas técnicas e instrumentos que permitan la recopilación, análisis y valoración de la información necesaria para la presentación del proyecto de tesis.

2. Técnicas

- **La Entrevista.** - Esta técnica la utilizamos para mantener conversaciones directas con los administrativos y docentes del AEIRNNR de la UNL, permitiendo tratar diferentes puntos relacionados con el desarrollo del proyecto, así mismo para obtener los problemas y requerimientos que necesita el AEIRNNR y conocer más detalladamente el funcionamiento del sistema voip actual.
- **La Observación.** - Mediante esta técnica evidenciamos que no aplican una telefonía móvil, y se sustentan problemas de comunicación entre Administrativos y Docentes, además se evidencia que una vez aplicada la telefonía móvil voz sobre ip, se emprende en una comunicación eficaz e inmediata lo cual mejora las condiciones de comunicación interna dentro del AEIRNNR.
- **La Encuesta.** - Esta técnica nos permitió sacar el resultado de nuestra implantación de la telefonía móvil voz sobre ip, y establecer su representación gráfica mediante la aplicación de técnicas de estadísticas.

3. Metodología de desarrollo

Para el desarrollo del proyecto se utilizarán las siguientes fases:

- **Fase 1:** Analizar la situación actual de la red telefónica Voip del AEIRNNR.

En esta fase obtendremos toda la información necesaria para determinar la situación real de la red de comunicación telefónica y sus necesidades en la institución para lo cual vamos a detallar las actividades a realizarse:

- Analizar cómo funciona su red telefónica actual.
- Entrevistar a los docentes y administrativos que estén utilizando la red telefónica.
- Sacar un borrador de los resultados que se obtenga en las entrevistas (falencias y necesidades).



Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta μ elastix en un Raspberry Pi para la Universidad Nacional de Loja en el AEIRNNR

Fase2: Diseñar y configurar μ elastix en el raspberry pi para telefonía móvil según las necesidades del AEIRNNR.

Lo que vamos a realizar en esta fase es el diseño de cómo implantar μ elastix en la red telefónica móvil y sus diferentes configuraciones con las que podemos hacer las pruebas preliminares de la red telefónica VoIP sobre aplicaciones móviles por lo cual vamos a realizar las siguientes actividades:

- Armar el HW raspberry pi con sus periféricos que nos va ayudar a funcionar nuestra red.
- Instalar el Software μ elastix en el raspberry pi.
- Configuración y pruebas de μ elastix sobre aplicaciones móviles según las conclusiones o necesidades de la Fase 1:
En esta actividad vamos hacer cada configuración y a su vez ir haciendo las pruebas de configuración para así poder llegar al objetivo de cada necesidad del sistema Voip sobre aplicaciones móviles.
- Pruebas preliminares del μ elastix sobre aplicaciones móviles de todas las configuraciones que se realice:
En esta última actividad vamos a realizar y ultimar los detalles antes de implantar en la red telefónica móvil.

Fase 3: Implantación de la Solución Planteada.

En esta fase se realiza la implementación y ejecución del μ elastix en el raspberry pi sobre aplicaciones móviles para su funcionamiento en su red telefónica.

- Armar el equipo raspberry pi en un lugar donde va a funcionar el sistema de telefonía.
- Configuración de la red telefónica móvil para adaptación del μ elastix.
- Realizar pruebas de control de registros de la red telefónica móvil.
- Pruebas de validación del sistema μ elastix en la red telefónica móvil.
- Conclusiones de validación del sistema μ elastix en la red telefónica móvil.

e. Resultados

1. Fase 1: Analizar la situación actual de la red telefónica Voip del AEIRNRR.

DISEÑO DE TELEFONÍA VOIP ACTUAL

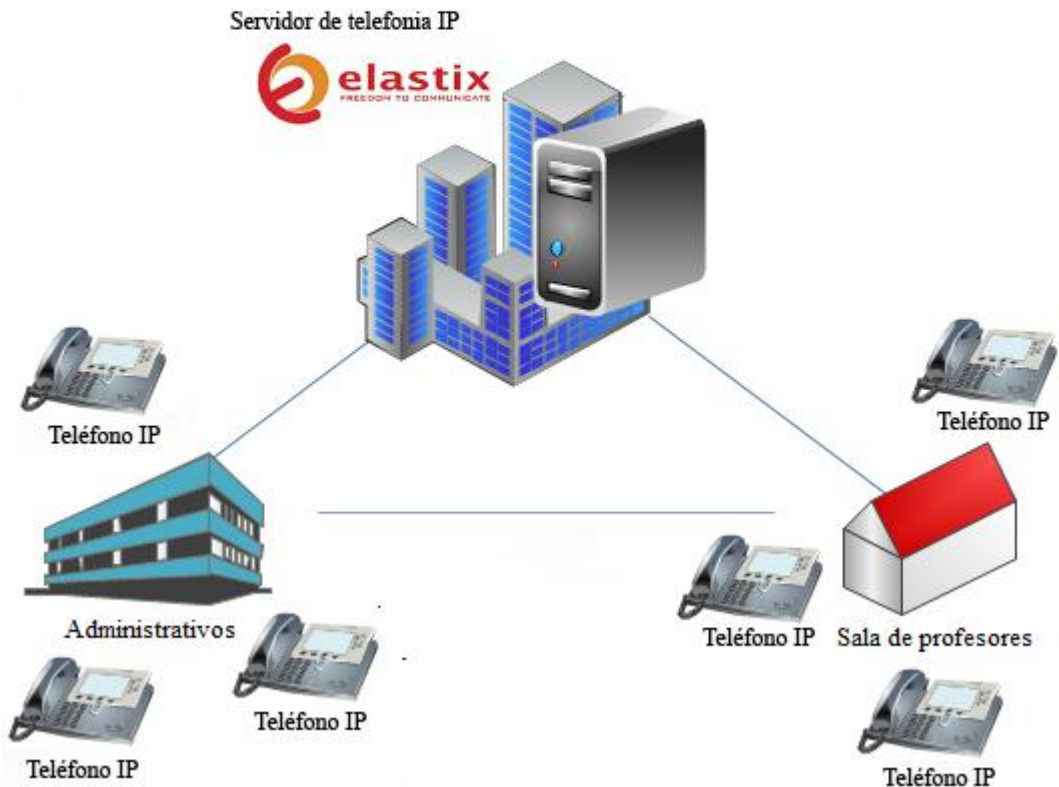


Figura 11. Diseño de telefonía voip actual

1.1. Analizar cómo funciona su red telefónica actual

Para poder verificar el estado actual de la red telefónica, utilizamos la técnica de la observación, teniendo como resultado que la telefonía móvil no está incluida ya que solo mantienen telefonía voip con teléfonos fijos (véase las figuras 12 y 13).



Figura 12. Teléfono ip



Figura 13. Teléfono ip modelo Grandstream GXP 1400

Las características del Hardware donde se encuentra instalado el sistema de telefonía voip, con teléfono fijo y elastix.

- Cpu, Intel Dual Core, Disco Duro 500Gb, Memoria 2Gb.
- Equipos de hardware voip que utiliza los docentes y administrativos.
 - Teléfono Ip Grandstream GXP1400.



1.2. Entrevistar a los docentes y al personal administrativo del AEIRNNR de la UNL, que está utilizando la red telefónica.

Analizado el funcionamiento de la telefonía desde su hardware y software podemos concluir que se encuentra en perfecto estado, obteniendo como resultado que el problema radica, cuando el administrativo trata de comunicarse con el docente, y este está fuera del sala de profesores donde hay un teléfono ip fijo grandstrem gxp1400; es imposible la comunicación entre ellos, teniendo como solución llamarlo por celular lo que implica gastar saldo o enviar a la persona encargada a buscarlo y entregar el mensaje.

En la cual se aplicó una entrevista a las siguientes personas y recomendaron (ver anexo 1). Estos análisis se los puede verificar en el siguiente proceso de entrevista, a todos los involucrados del sistema de telefonía que son administrativos y docentes, en lo cual procedemos a formular las siguientes preguntas para facilitarnos la identificación de falencias y necesidades.

A continuación, citaremos las preguntas que se formularon en dicha entrevista.

- ¿Qué ventajas ofrece la telefonía móvil voz sobre ip entre Docentes y Administrativos?
- ¿Actualmente la Carrera de Ingeniería en Sistemas, para establecer la comunicación entre docentes y personal administrativo mantiene un sistema de comunicación interna móvil voz sobre ip?
- ¿Apoyaría a la implementación de un sistema de telefonía móvil de voz sobre ip para el AEIRNNR?

1.3. Sacar un borrador de los resultados que se obtenga en las entrevistas (falencias y necesidades).

La identificación de falencias constituye una parte fundamental en la configuración de software, a través del cual se consigue un mejor entendimiento del problema y se pueden identificar con mayor exactitud las necesidades de telefonía en el AEIRNNR.

Al revisar los resultados es latente el problema de no tener una telefonía móvil voip, ya que la sugerencia de la mayoría de los entrevistados es que esta sea implementada.

Su hardware cumple con los requisitos para la instalación del sistema de telefonía móvil, este, ocupa un espacio físico considerable, aunque este no es un problema nosotros propusimos utilizar un hardware como raspberry pi, una alternativa que nos da funcionalidades básicas para un sistema voip, por eso es nuestro reto que este dispositivo funcione de manera adecuada y sobretodo cumpliendo los requerimientos y necesidades del AEIRNNR de la UNL.

Para que un usuario interactúe de manera más amigable estamos presentando el sistema de voip μ elastix, este software es basado en asterisk, así que es un mejoramiento de la herramienta voip, pero con la inclusión de una interfaz más amigable con el usuario, una aplicación que trabaja con nuestro hardware a utilizar y sobretodo nos va ayudar a cumplir con las necesidades y requerimientos del AEIRNNR de la UNL.

2. Fase 2: Diseñar y configurar μ elastix en el raspberry pi para telefonía móvil según las necesidades del AEIRNNR.

1.1. Partes que necesitamos para armar del raspberry pi B

Se necesita una Memoria SD de 8GB o de 4GB



Figura 14. Memoria SD

Adaptador de corriente de 5 voltios de 1.5 Amperios.



Figura 15. Adaptador de corriente

Case raspberry pi



Figura 16. Case raspberry pi

Disipadores de calor para raspberry pi



Figura 17. Disipadores

Raspberry pi B

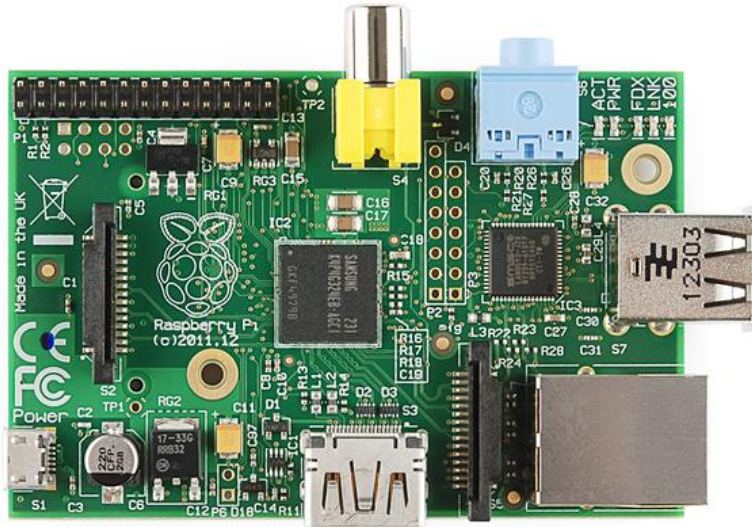


Figura 18. Raspberry Pi B

1.2. Armar el raspberry pi B



Figura 19. Raspberry Pi Armado

1.3. Instalar el Software *µelastix* en el raspberry pi.

En este proceso se realiza la instalación del sistema *µelastix* desde cero paso a paso con sus respectivas imágenes.

1. Lo primero que se hace es preparar nuestra memoria sd de 4 GB por lo cual se inserta en nuestro lector sd de nuestro computador. Se formatea y particiona nuestra memoria sd. La primera partición va ser de tipo FAT16 y su tamaño de 256 MB. La segunda partición va ser de tipo EXT3 y su tamaño será el resto de nuestra memoria sd, aproximadamente 3,75 GB. Esta partición de la sd es para la instalación de nuestro *µelastix*. Ya creada las particiones se a crea los sistemas de archivos en la primera partición es / dev / sdb1 y la segunda partición / dev / sdb2. Como se detalla a continuación en la figura 20.

```
Examine y ejecute aplicaciones instaladas alBox: ~
Archivo Editar Ver Buscar Terminal Ayuda
/dev/sdb1      1      67      255440    83  Linux
/dev/sdb2      68     1019    3629976   83  Linux

Orden (m para obtener ayuda): t
Número de partición (1-4): 1
Código hexadecimal (escriba L para ver los códigos): e
Se ha cambiado el tipo de sistema de la partición 1 por e (W95 FAT16 (LBA))

Orden (m para obtener ayuda): p

Disco /dev/sdb: 3980 MB, 3980394496 bytes
123 cabezas, 62 sectores/pista, 1019 cilindros
Unidades = cilindros de 7626 * 512 = 3904512 bytes
Tamaño de sector (lógico / físico): 512 bytes / 512 bytes
Tamaño E/S (mínimo/óptimo): 512 bytes / 512 bytes
Identificador de disco: 0x0009cae3

Disposit. Inicio Comienzo Fin Bloques Id Sistema
/dev/sdb1      1      67      255440    e  W95 FAT16 (LBA)
/dev/sdb2      68     1019    3629976   83  Linux

Orden (m para obtener ayuda):
```

Figura 20. Partición de la memoria sd en terminal ubuntu

2. Ya una vez preparada se procede a la instalación de nuestro *µelastix*, esta instalación se la realiza por el terminal de Ubuntu.

Antes de este proceso es necesario indicar cómo y dónde descargar μ elastix a continuación se procede a los detalles.

- Primeramente μ elastix es creada por la empresa ecuatoriana palo santo así que ellos tienen su página oficial de descargas en el siguiente link <http://www.uelastix.com/>, una vez ingresado a esta página aprecia que hay algunos componentes que nos indican de algunos arm pero al dirigimos a la descarga de nuestro μ elastix para raspberry pi B como se muestra en la siguiente figura 21.



Figura 21. Página oficial μ elastix

3. Ya descargado nuestro archivo μ elastix, “**elastix-arm-2014-01-30.tar**” se procede a descomprimirlo, aquí nos asoman dos archivos los cuales nos sirven para el montaje en las dos particiones de la sd:
 - BOOT.tar.gz estará en la primera partición /dev/sdb1 el cual este nos permite el arranque de μ elastix en nuestro raspberry pi.
 - root.tar.gz está en la segunda partición /dev/sdb2 este es nuestro módulo μ elastix de instalación en nuestro raspberry pi.

En la Figura 22 se aprecia cómo están ya montados nuestros archivos en la micro sd.

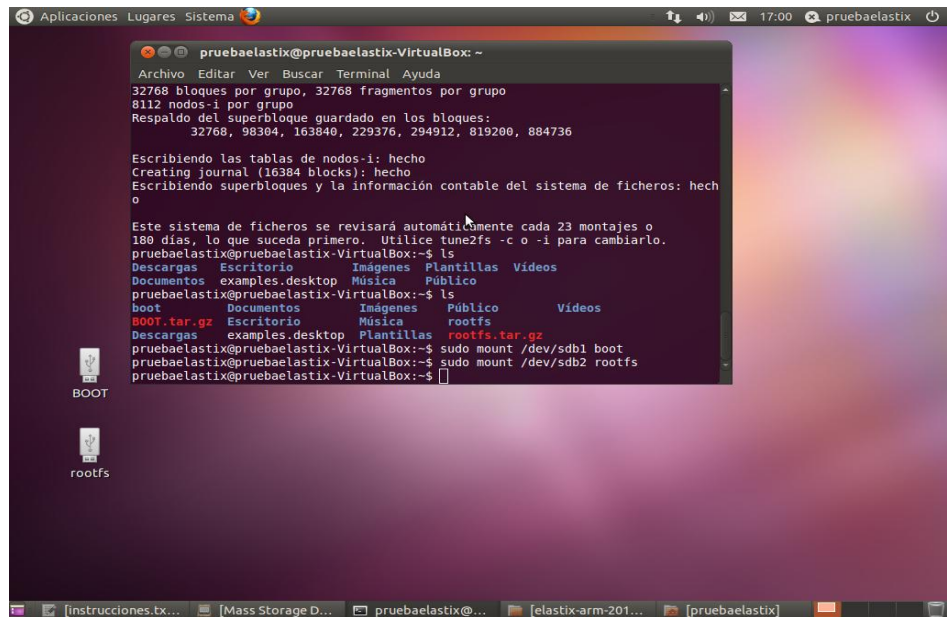


Figura 22. Transferir archivos en partición BOOT y Rootfs

- Una vez ejecutado y seguido los pasos anteriores se procede a desmontar nuestro micro sd para la colocación del raspberry pi.

Ya instalado en la micro sd se procede a darle energía al raspberry pi y automáticamente se ejecutará nuestro *µelastix* ya lista para las configuraciones respectivas como se ve en la figura 23, para poder ingresar a nuestro raspberry pi por consola se utiliza el programa putty, pudiendo acceder con nuestra ip, que por defecto será 192.168.1.251.

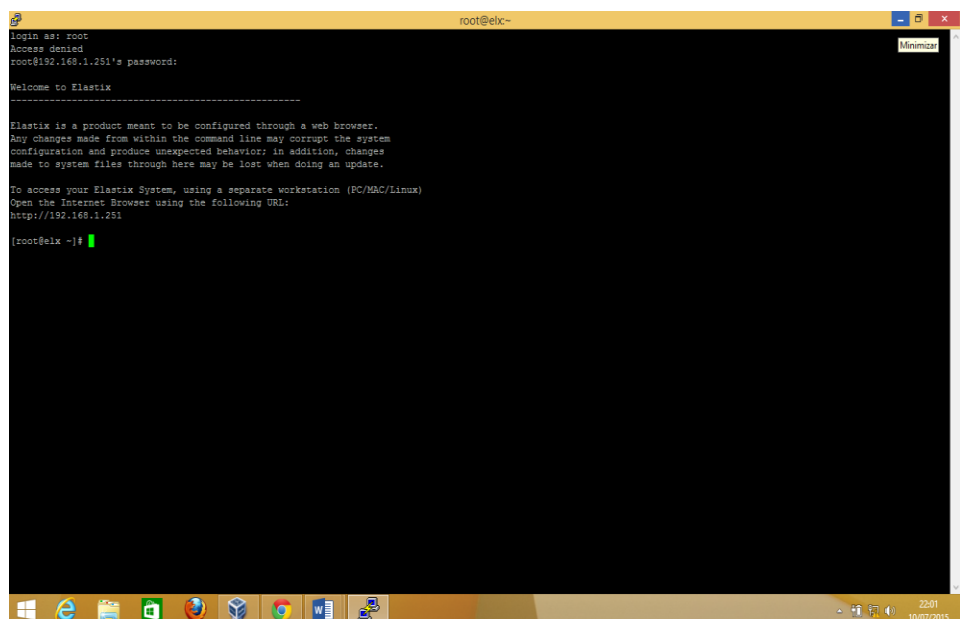


Figura 23. Raspberry pi en funcionamiento mediante consola.

1.4. Configuración de μ elastix según las conclusiones o necesidades de la Fase 1.

El sistema VOIP μ elastix es una herramienta que nos va ayudar a configurar según las necesidades que necesita la UNL en el AEIRNNR, obviamente raspberry pi tienen sus limitaciones, pero vamos demostrar que es suficiente para ejecutar nuestro proyecto; en los siguientes puntos se a configura extensiones, troncales, ivr y la unión de dos troncales con centralillas μ elastix diferentes, como es la del raspberry pi para móviles y la actual μ elastix que funciona en el AEIRNNR para telefonía ip fija. Cumpliendo las necesidades de la UNL del AEIRNNR que a continuación se detalla:

1. Para acceder a μ elastix una vez instalado en la memoria sd, debemos abrir cualquier explorador en este caso se utiliza el Google Chrome, ingresando a la Ip de μ elastix, que por defecto la dirección ip es 192.168.1.251. Al llenar los campos usuario y contraseña, que por defecto de nuestra imagen que descomprimimos en nuestra sd, es la siguiente “admin” y “palosanto” respectivamente (Ver figura 24)

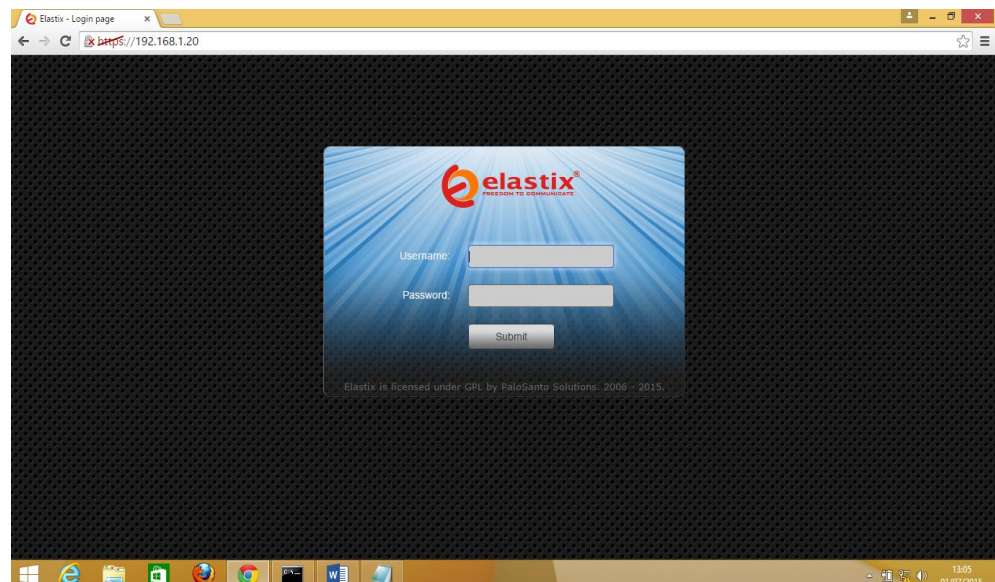


Figura 24. Página de ingreso μ elastix

2. La figura 25 nos presenta como se muestra la página principal de nuestro melastix.

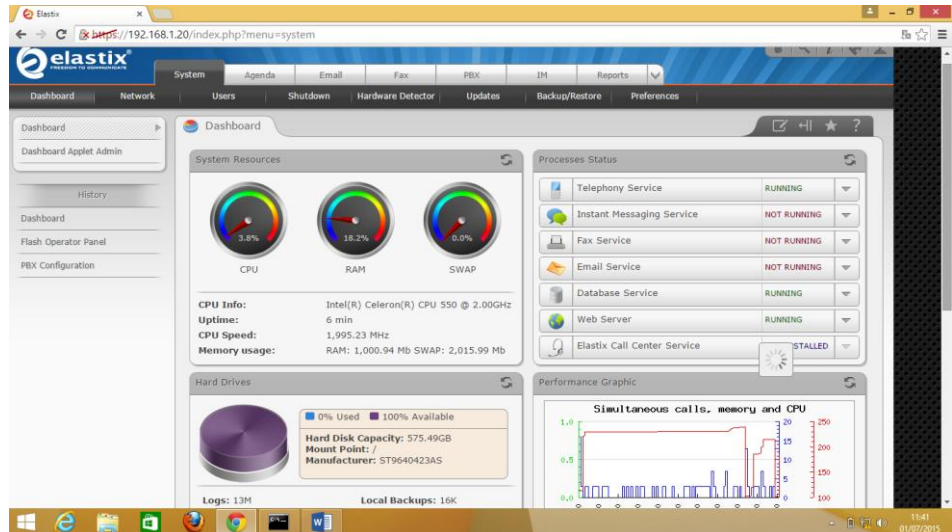


Figura 25. Menú principal melastix

3. Al ubicarnos en la pestaña PBX como lo muestra la figura 26 en la cual se despliega en un menú todas nuestras actividades a configurar.

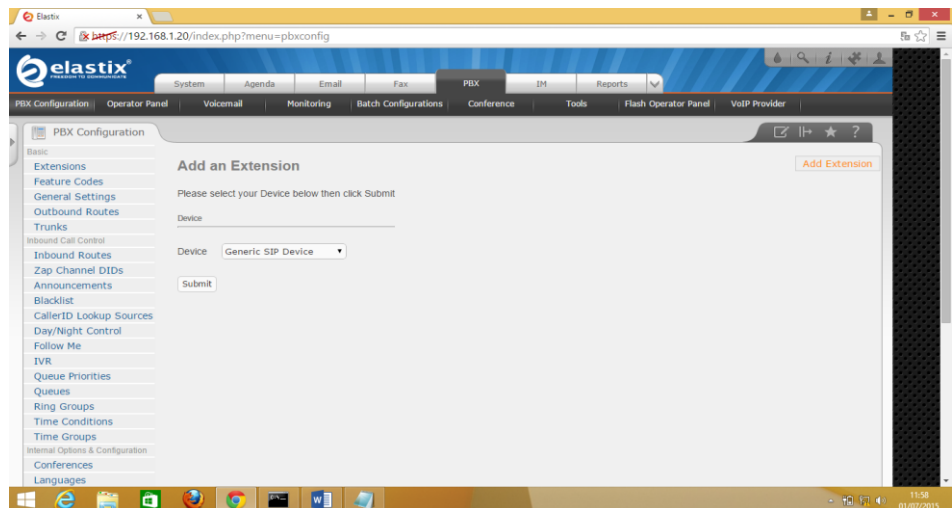


Figura 26. PBX y su menú

4. Luego se procede a la creación de extensiones con sus respectivos nombres de profesores, de acuerdo a la carrera que pertenezcan. El total

que se va a crear son 57 extensiones, los cuales son el número de profesores actuales en el AEIRNNR.

Las extensiones que generadas son de tipo SIP (Service internet Provider). En la figura 27, llenamos los campos necesarios de la ventana User Extension, Display Name, CID Num Alias y secret. En User Extension se llena el campo con el número de extensión en el caso nuestro será 101. En Display Name se llena el nombre del profesor el cual le asignamos la extensión en el caso nuestro ANA TROYA. En CID Num será un número alternativo. Y en secret se llenará una clave que se sugiere que sea la misma en las demás extensiones para luego no tener el conflicto de no acordarse.

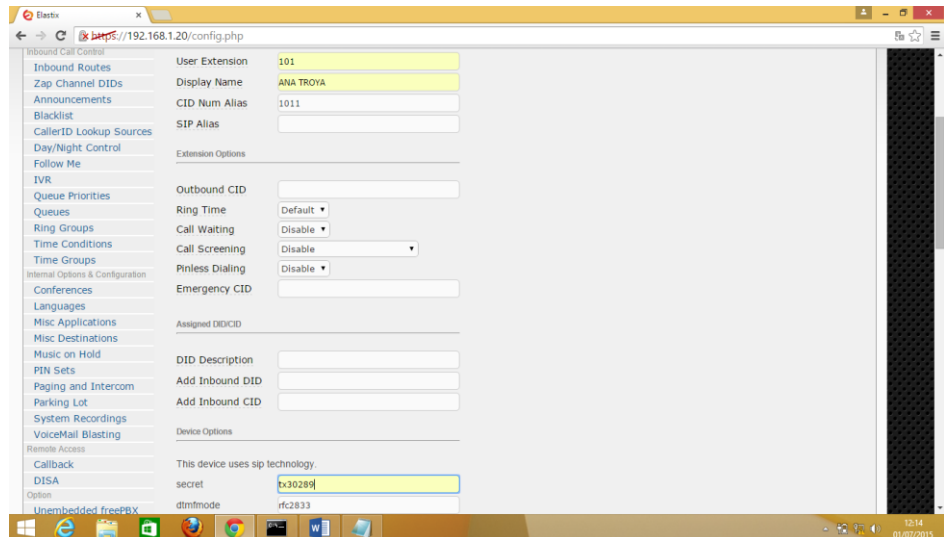


Figura 27. Creación de extensiones

5. Es necesario aplicar los cambios en cada acción que se a configuran en nuestro µelastix como muestra la figura 28.

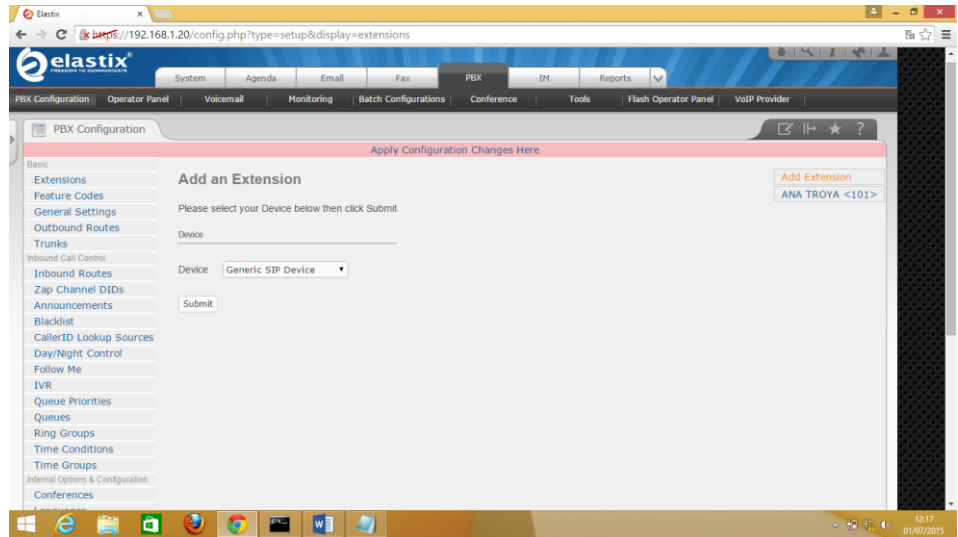


Figura 28. Aplicar cambios a las extensiones

6. Ya teniendo, una como patrón, procedemos a crear las 57 extensiones tipo SIP y el resultado de esta nos da la figura 29, 30, 31 y 32.

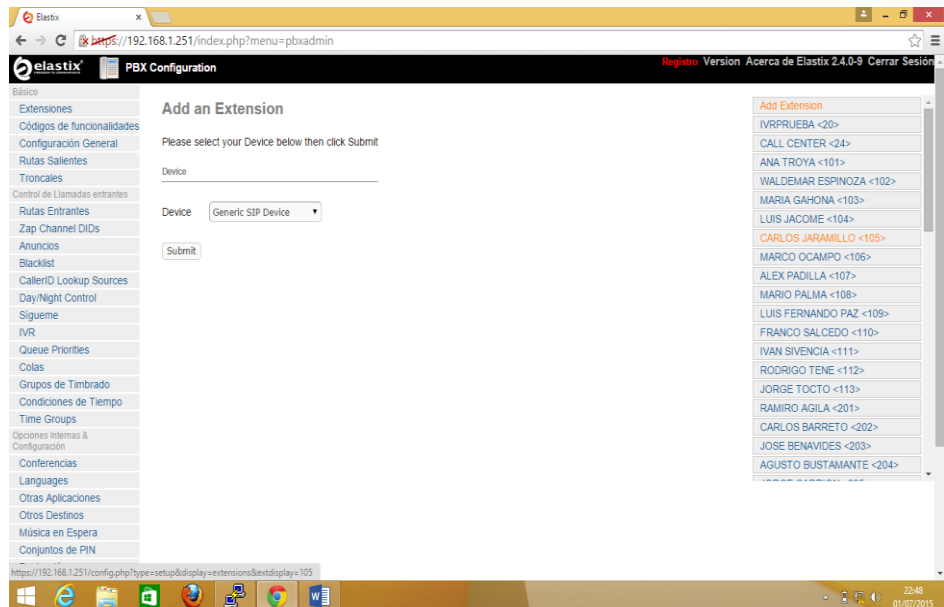


Figura 29. Creación de extensiones de Sistemas

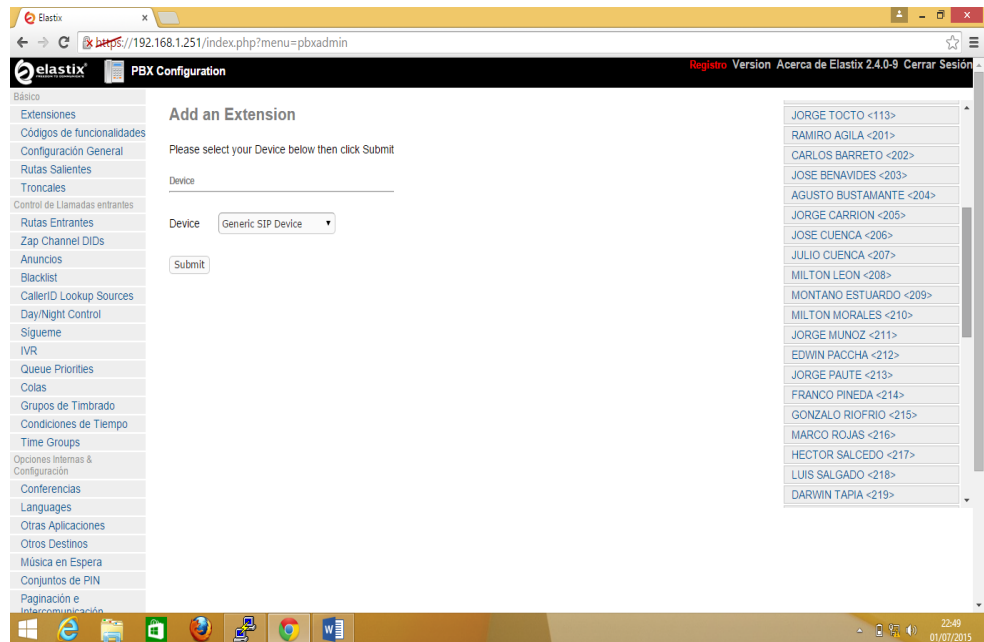


Figura 30. Creación de extensiones de Electromecánica

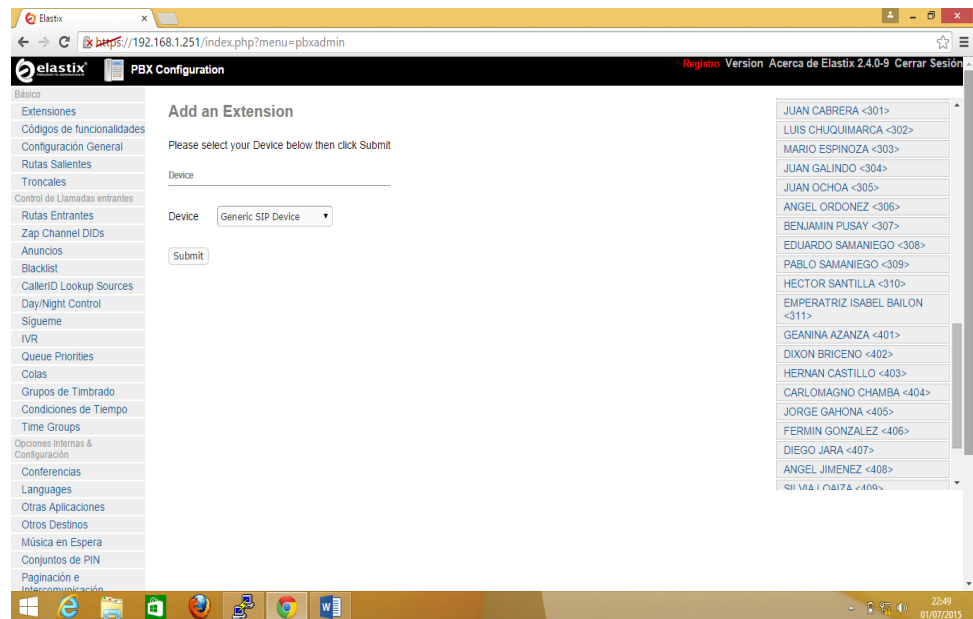


Figura 31. Creación de extensiones de Electrónica y Telecomunicaciones

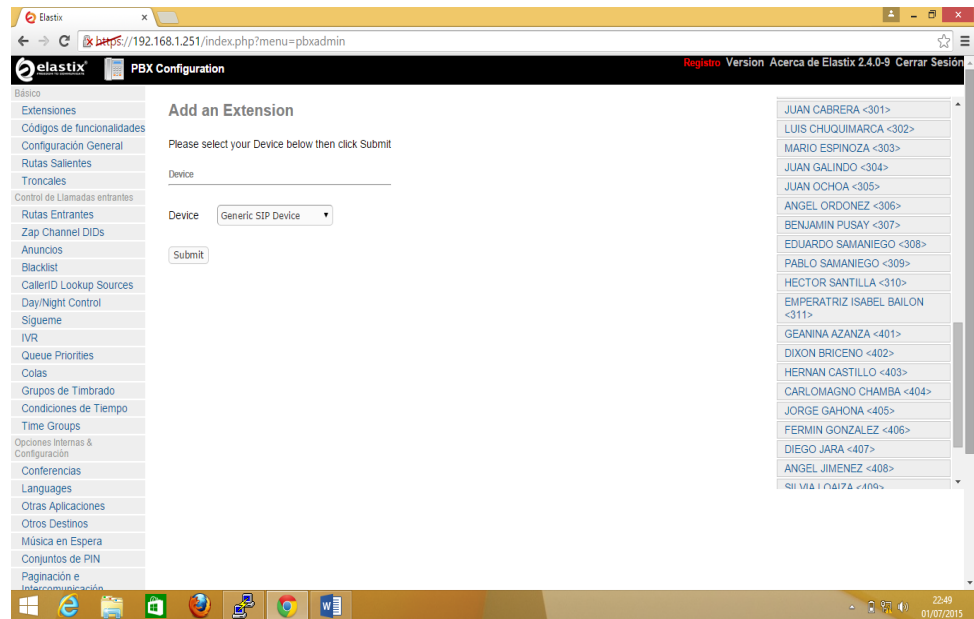


Figura 32. Creación de extensiones de Geología

7. En la parte gráfica de nuestro **μelastix** nos mostrará la figura 33 dónde nos da el resultado visual de nuestras extensiones SIP creadas:

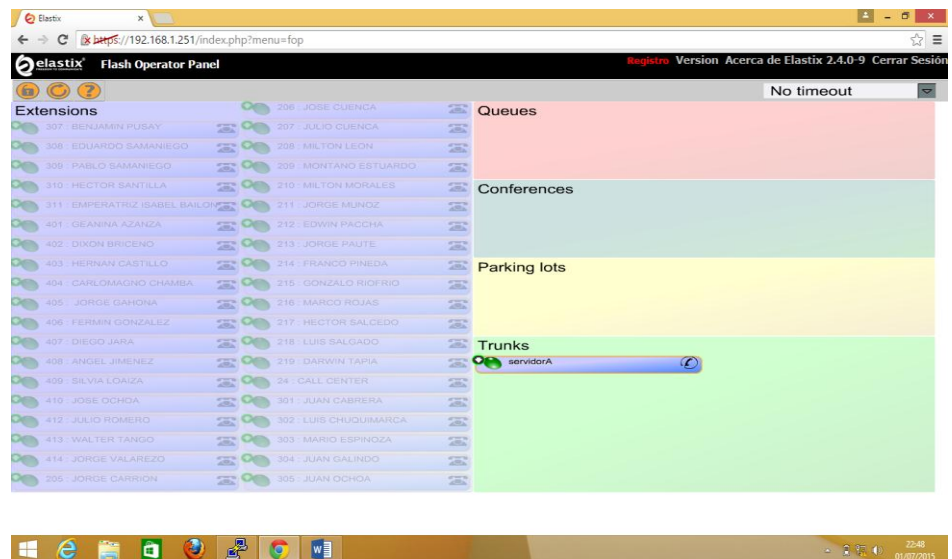


Figura 33. Gráfica de extensiones creadas

8. Nos dirigimos a la siguiente dirección <http://www.nuance.com/for-business/by-solution/customer-service-solutions/solutions-services/inbound-solutions/loquendo-small-business-bundle/interactive-tts-demo/index.htm> y se llena el campo de texto con la información que

deseemos que escuche nuestro usuario como lo muestra en la figura 34 con el siguiente texto: “Bienvenido al directorio de docentes, presione 1 Ingeniería en Sistemas, presione 2 Ingeniería Electromecánica presione 3 Ingeniería Electrónica y Telecomunicaciones, presiones 4 Ingeniería Geología Ambiental”.

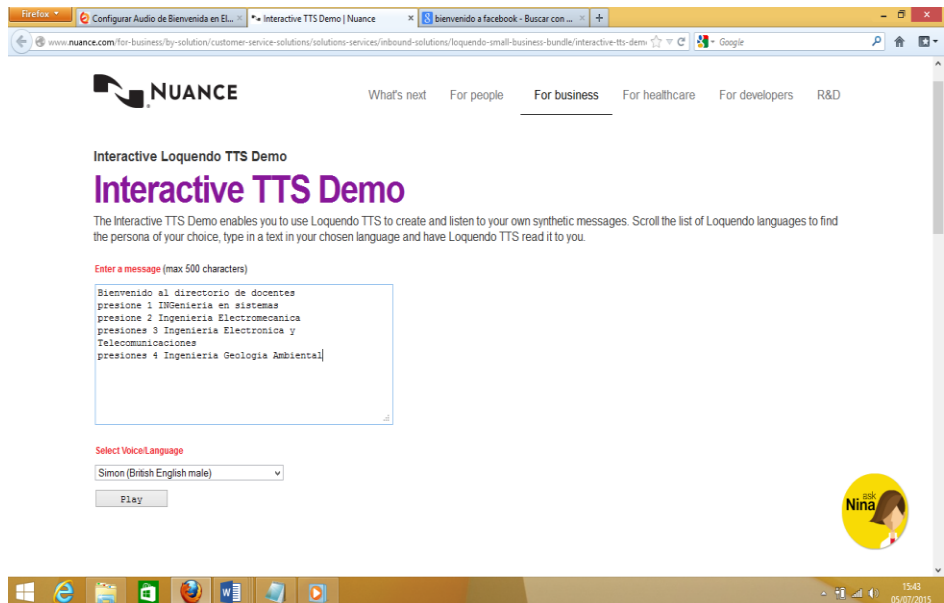


Figura 34. Creación de Audios para operadores.

9. Presionando play en la figura 34 a continuación se creará un archivo en este caso le nombramos MENU PRINCIPAL que nos dará formato wav. Pero en nuestro μ elastix nos permite ejecutar archivos mp3, por lo cual se procede a cambiar de formato ingresado a este enlace <http://audio.online-convert.com/es/convertir-a-mp3> . Teniendo en cuenta que en el parámetro modificar canales de audio seleccionamos en tipo mono, ya que nuestro μ elastix lo reproduce así. Y tendremos la figura 35.

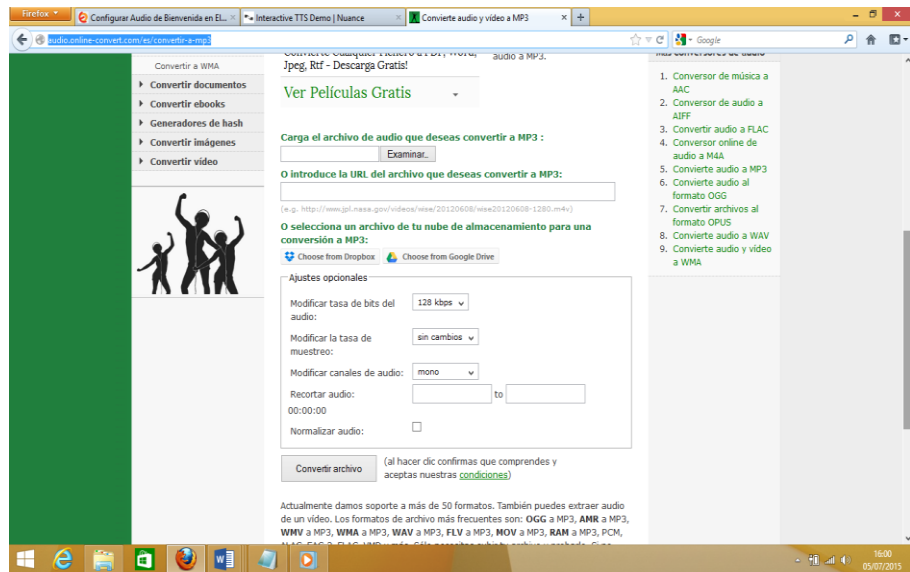


Figura 35. Convertir Audio mp3 canal mono.

10. A continuación, se sitúa en grabaciones del sistema, el uso de este será grabar la voz de nuestro operador, el cual guiará en nuestra centralilla para acceder a comunicarnos con el docente, en la parte seleccionar archivo de la Figura 36 escogemos todos los archivos que grabamos anteriormente quedándonos subidos al sistema μ elastix nuestros operadores.

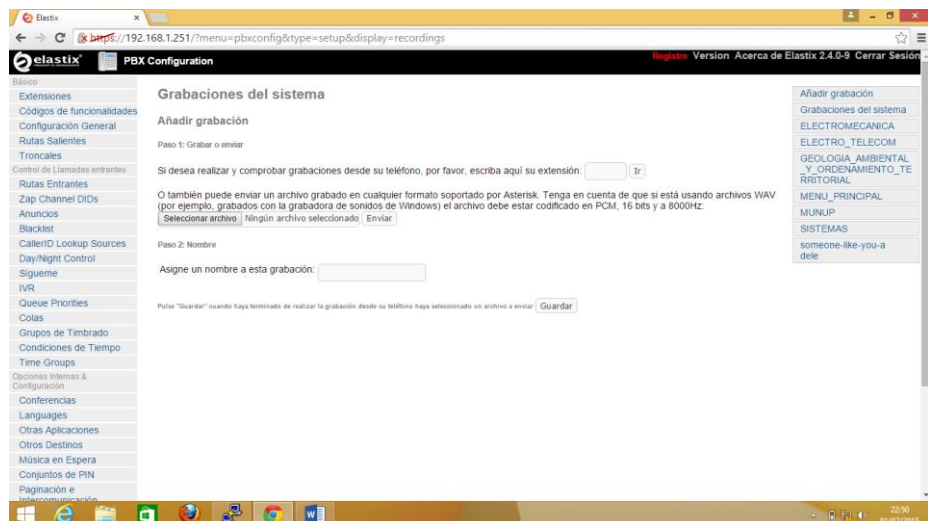


Figura 36. Cargar audio a nuestro μ elastix.

11. Se procede a crear nuestros IVR, estos ayudan a la contestación de nuestros operados, para poder guiar en nuestra comunicación de la carrera y extensión al cual se quiera llamar:

- IVR-MENUPRINCIPAL este operador ofrece un menú el cual informa a que carrera queremos comunicar.
- IVR-SISTEMAS este operador ayuda con la información correspondiente a la extensión y nombre de cada uno de los 13 docentes que están actualmente en la Carrera de Sistemas.
- IVR-ELECTROMECHANICA este operador ayuda con la información correspondiente a la extensión y nombre de los 19 docentes que están actualmente en la carrera de electromecánica.
- IVR-ELECTRONICA-Y-TELECOMUNICACIONES este operador ayuda con la información correspondiente a la extensión y nombre de los 11 docentes que están actualmente en la Carrera de Electrónica y Telecomunicaciones.
- IVR-GEOLOGIA este operador ayuda con la información correspondiente a la extensión y nombre de los 14 docentes que están actualmente en la Carrera de Geología.

A continuación, la figura nos presenta la configuración correspondiente de nuestros seis IVR principales.

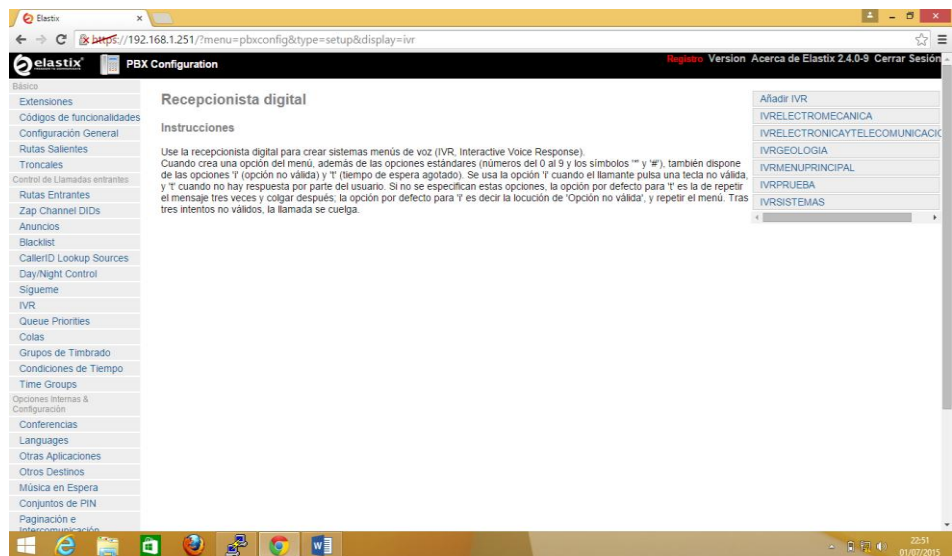


Figura 37. Lista de IVR creados

12. Ahora al configurar nuestros anuncios, esto ayuda a nuestros IVR, ya que, en cada salto de contestación de operador, nos dá una voz, sin hacer el salto brusco a nuestras extensiones.

- Comunicando con Ingeniería en Sistemas, Este anuncio va ser entre IVR_SISTEMAS y las extensiones desde la 101 hasta la 113. (Ver figura 38).

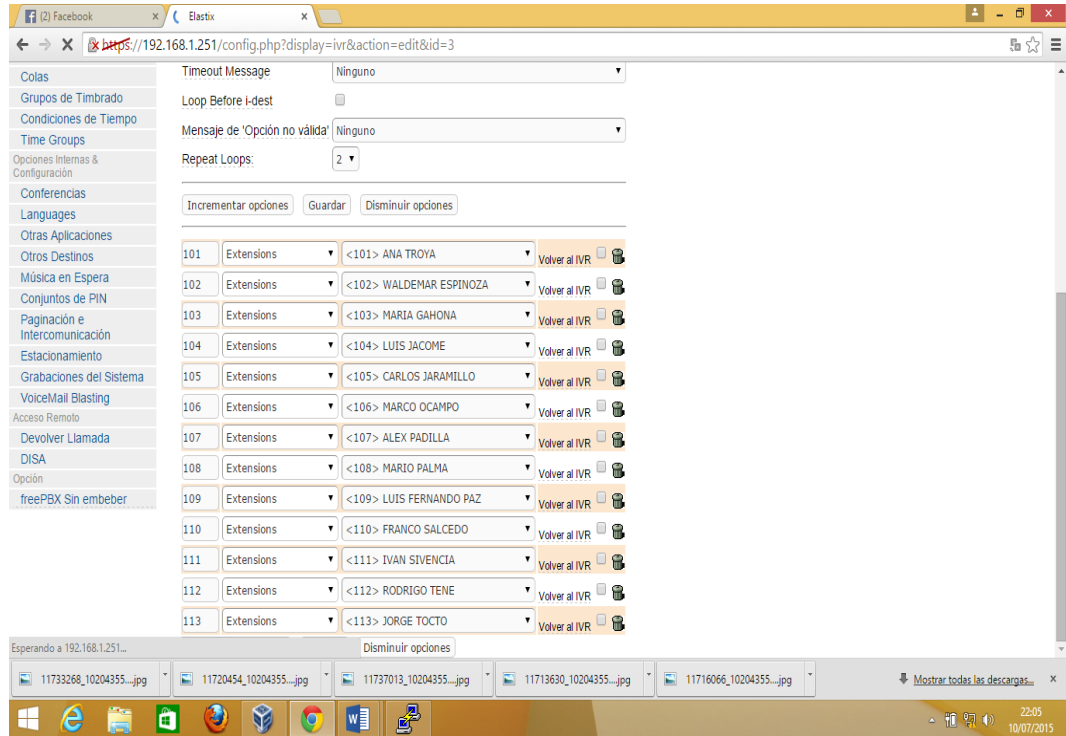


Figura 38. Configuración IVR anuncio Sistemas.

- Comunicando con Ingeniería en Electromecánica, Este anuncio va ser entre IVR_ELECTROMECHANICA y las extensiones desde la 201 hasta la 219. (Ver figura 39)

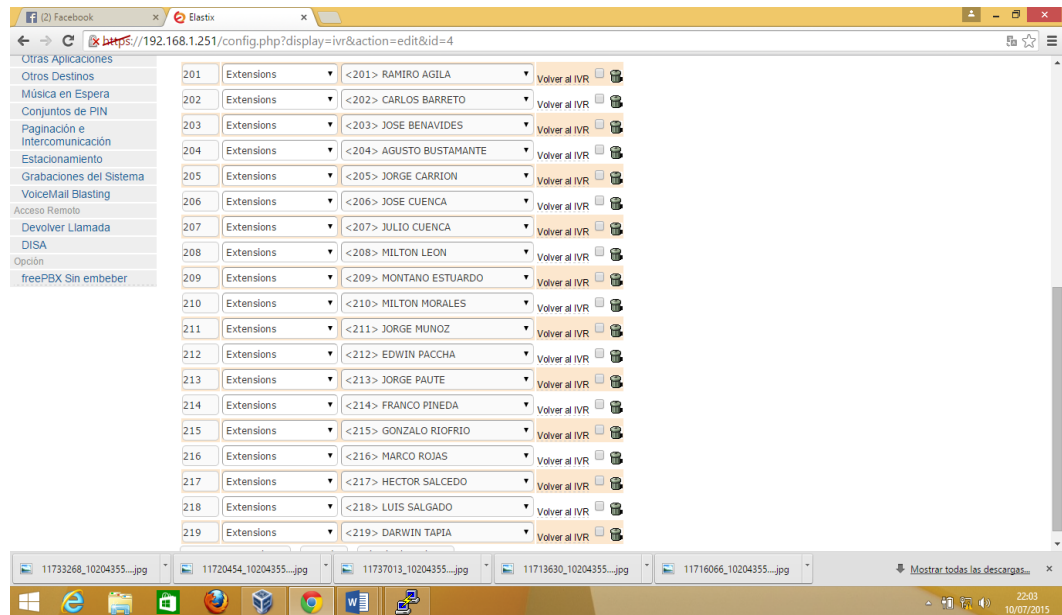


Figura 39. Configuración IVR anuncio Electromecánica.

- Comunicando con Ingeniería en Electrónica y Telecomunicaciones, Este anuncio va ser entre IVR_ELECTRONICAYTELECOMUNICACIONES y las extensiones desde la 301 hasta la 311. (Ver figura 40).

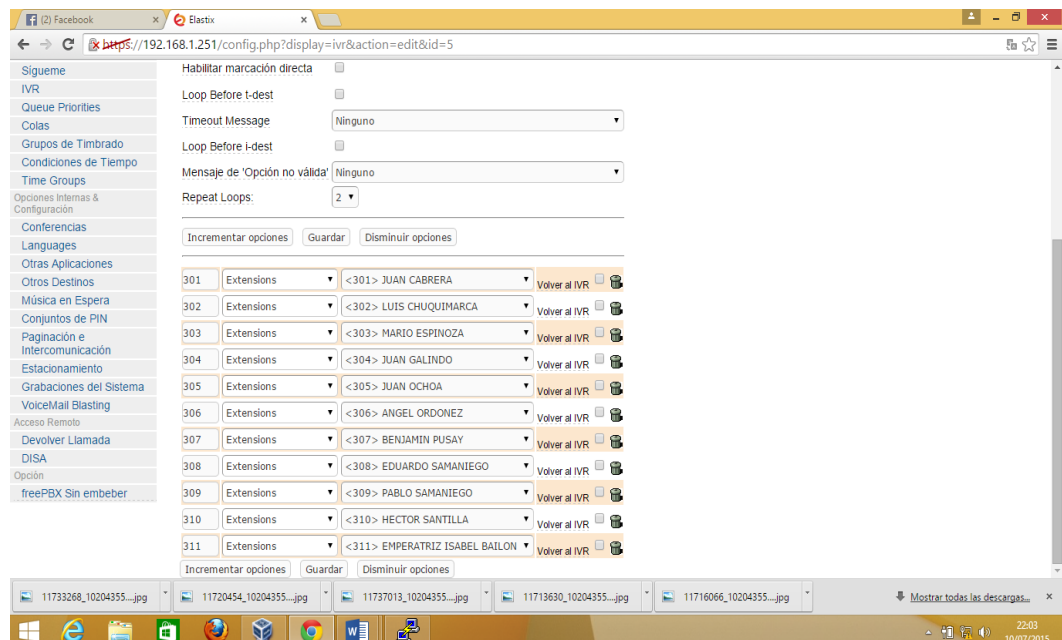


Figura 40. Configuración IVR anuncio Electrónica y Telecomunicaciones.

- Comunicando con Ingeniería en Geología, Este anuncio va ser entre IVR_GEOLOGIA y extensiones desde la 401 hasta la 414. (Ver figura 41).

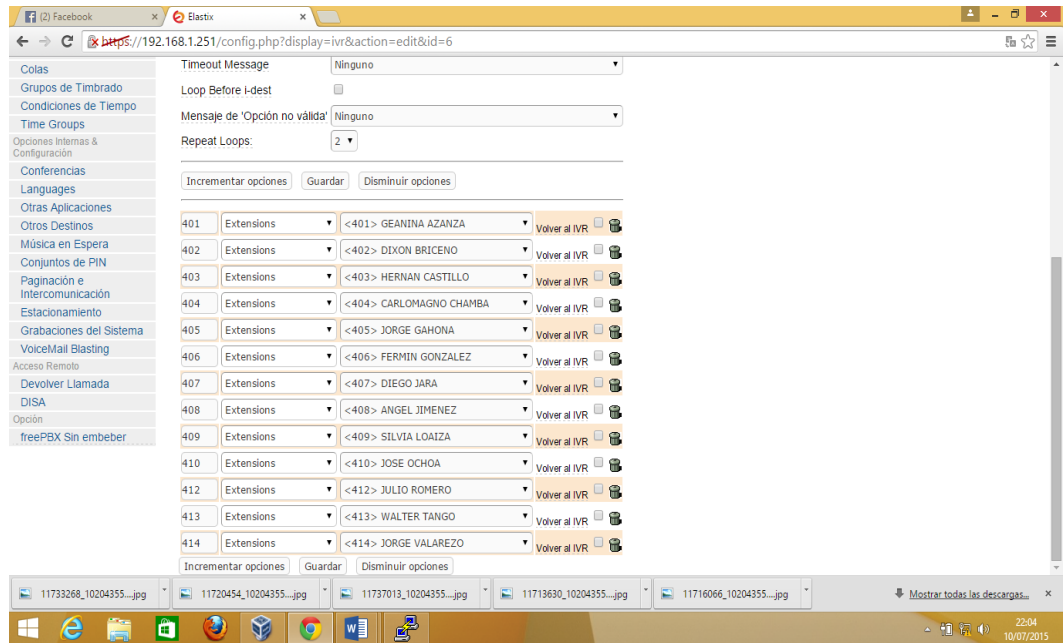


Figura 41. Configuración IVR anuncio Geología Ambiental.

1.5. Configuración en telefonía móvil.

- Necesitamos un teléfono Smartphone.



Figura 42. Smartphone diferentes modelos

- Entrar al play store de nuestro dispositivo móvil (Ver figura 43).

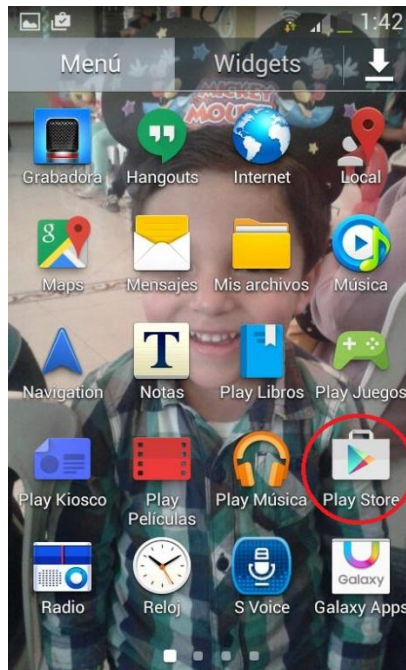


Figura 43. Entrar al Play Store de nuestro Smartphone

- Seleccionamos aplicaciones y nos ubicamos en Top Gratis (Ver figura 44 y 45).

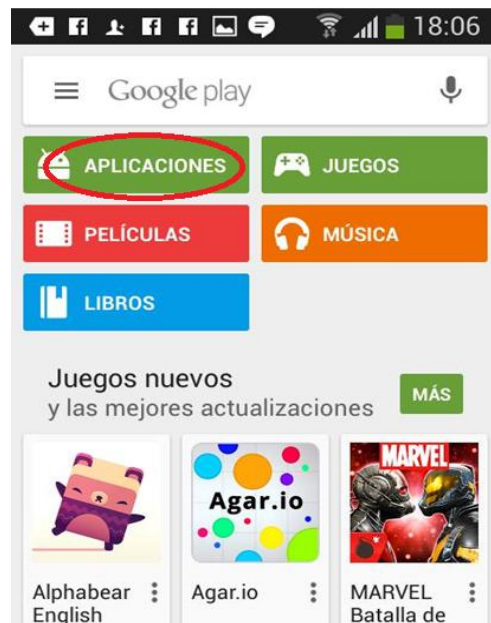


Figura 44. Aplicaciones de Play store



Figura 45. Top gratis de nuestro Play store

- Buscar la aplicación gratuita Zoiper (Ver figura 46).



Figura 46. Aplicación Zoiper IAX SIP VOIP

- Instalar la Aplicación Zoiper IAX SIP VOIP (Ver figura 47).



Figura 47. Instalar la Aplicación Zoiper

- Ingresamos en el icono de la aplicación Zoiper.

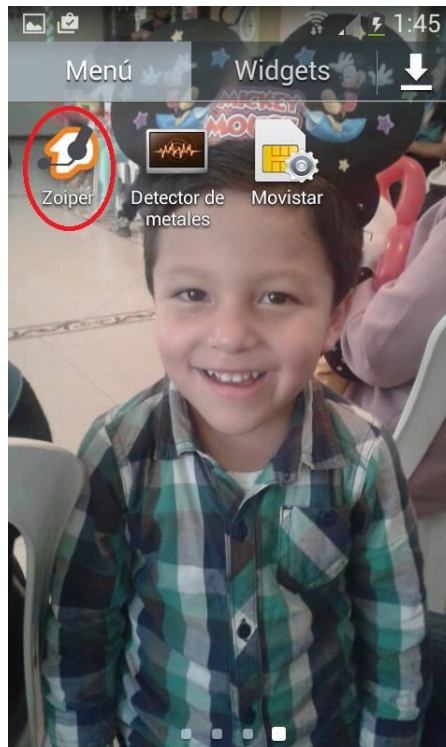


Figura 48. Ingresar a la aplicación Zoiper.

- Abrimos la pestaña **Ajustes** en nuestra aplicación (Ver figura 49).

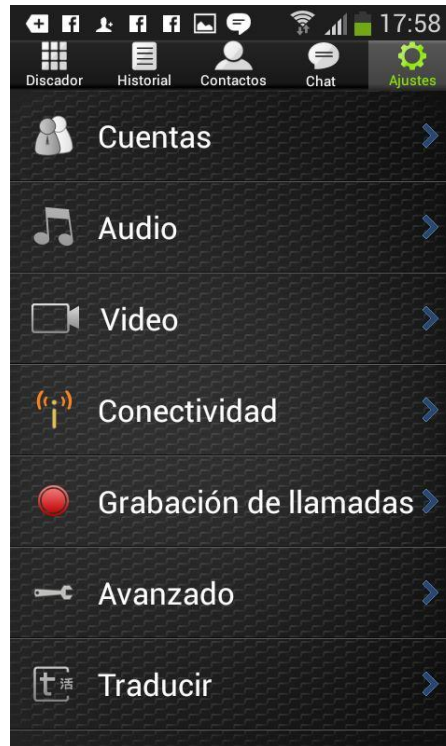


Figura 49. Ajustes Zoiper

- Seleccionamos la opción **Cuentas** (Ver figura 50).

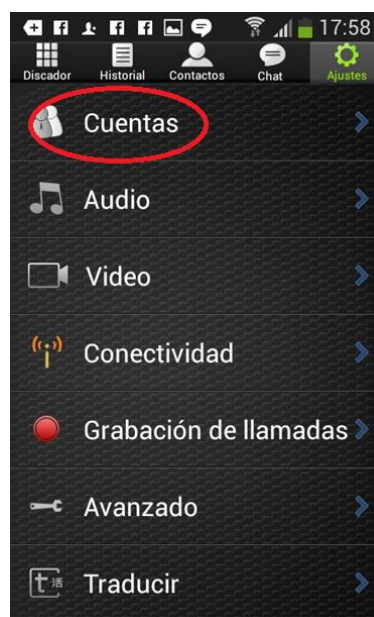


Figura 50. Opción Cuentas

- Agregamos una cuenta SIP dentro de nuestra aplicación Zoiper (Ver figura 51).

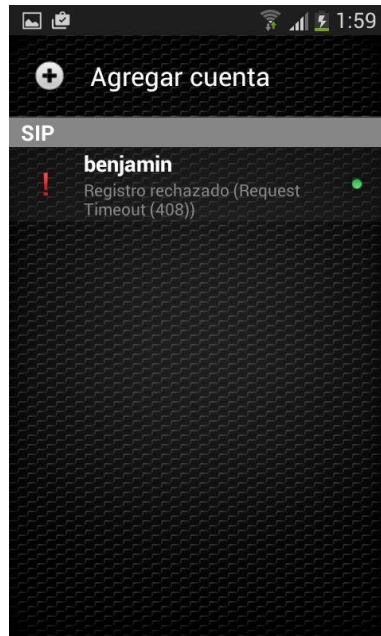


Figura 51. Agregar cuenta SIP

- Llenamos los datos solicitados: nombre de cuenta, host, nombre de usuario y clave de acuerdo a nuestros datos que hemos creado en las extensiones ya creadas (Véase figura 52).

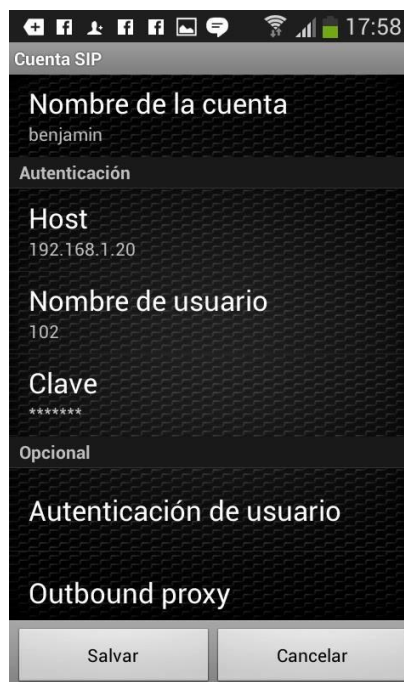


Figura 52. Configuración de nuestra extensión dentro de la Aplicación Zoiper.

- Seleccionamos salvar para guardar nuestros cambios (Ver figura 53).

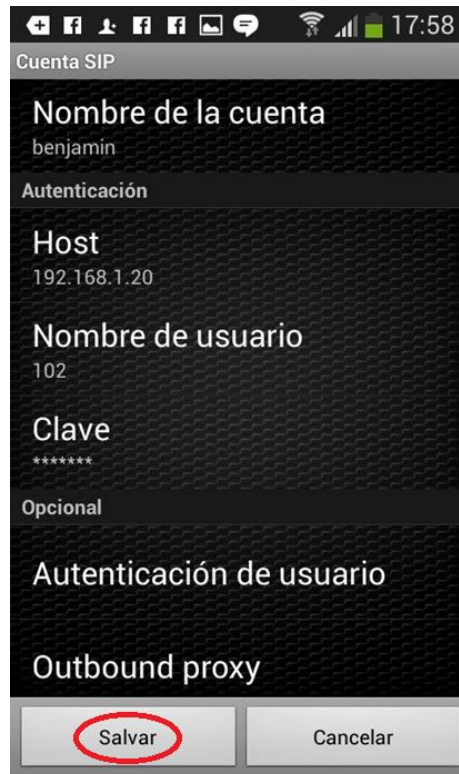


Figura 53. Guardar Cambios en nuestra configuración.

- Veremos en la figura 54 que estamos conectados a nuestro PBX.



Figura 54. Conectando en PBX

- 1.6. Pruebas de *μ*elastix sobre aplicaciones móviles según las conclusiones y necesidades de la Fase 1: Pruebas preliminares del *μ*elastix sobre aplicaciones móviles de todas las configuraciones que se realice:

Para poder ver cómo actúa asterisk en este caso, nosotros hemos descargado la aplicación putty, en la cual nos ayuda a entrar en la consola de elastix (Ver figura 55).

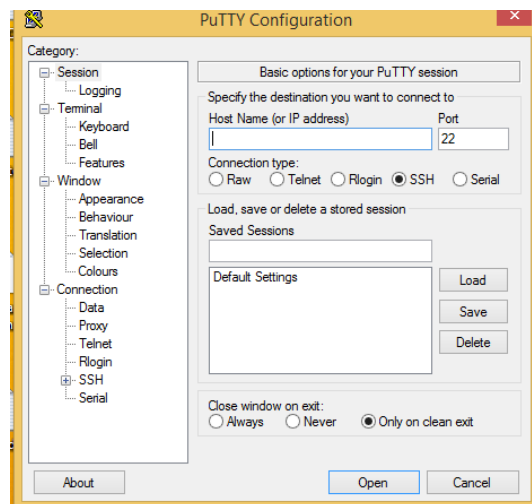


Figura 55. Acceder a Putty.

A continuación se llena el campo “host name o Ip address” con la dirección ip de nuestro raspberry pi, en nuestro caso la ip que esta por defecto es 192.168.1.251(Ver figura 56).

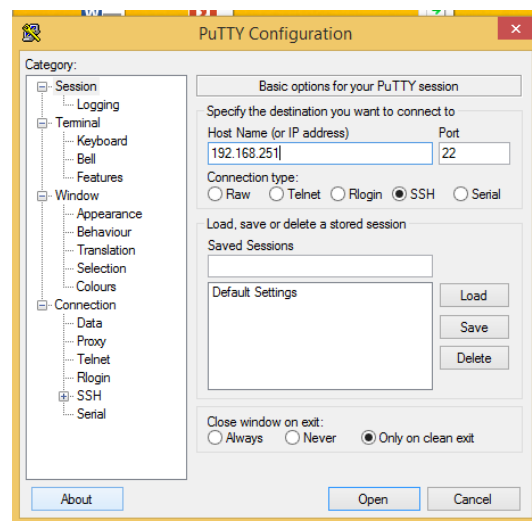


Figura 56. Datos requeridos para acceder consola raspberry pi.

Luego aparece la consola como lo muestra la figura 57 en la cual tendremos que llenar **login as:** con “root” y **password:** con “palosanto”.

```
root@elx~
login as: root
Access denied
root@192.168.1.251's password:
Welcome to Elastix
-----
Elastix is a product meant to be configured through a web browser.
Any changes made from within the command line may corrupt the system
configuration and produce unexpected behavior; in addition, changes
made to system files through here may be lost when doing an update.

To access your Elastix System, using a separate workstation (PC/MAC/Linux)
Open the Internet Browser using the following URL:
http://192.168.1.251

[root@elx ~]#
```

Figura 57. Consola raspberry pi.

Seguido escribimos el siguiente comando “asterisk –rvvddd” , con el fin de ingresar al código que genera asterisk, quedandonos la figura 58.

```
root@elx~
login as: root
Access denied
root@192.168.1.251's password:
Welcome to Elastix
-----
Elastix is a product meant to be configured through a web browser.
Any changes made from within the command line may corrupt the system
configuration and produce unexpected behavior; in addition, changes
made to system files through here may be lost when doing an update.

To access your Elastix System, using a separate workstation (PC/MAC/Linux)
Open the Internet Browser using the following URL:
http://192.168.1.251

[root@elx ~]# asterisk -rvvddd
-Bash: asterisk: command not found
[root@elx ~]# asterisk -rvvddd
Parsing /etc/asterisk/asterisk.conf
Seeding global EID 'b8:27:ebffd:ab:64' from 'eth0' using 'dloogifhwaddr'
Asterisk 11.4.0, Copyright (C) 1999 - 2012 Digium, Inc. and others.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for detail
s.
This is free software, with components licensed under the GNU General Public
License version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
-----
Connected to Asterisk 11.4.0 currently running on elx (pid = 1368)
elx*CLI>
```

Figura 58. Ingreso a la consola de asterisk.

Ahora hacemos la conexión en dos teléfonos móviles, enlazándonos en este caso con la extensión 101 perteneciente a: ANA TROYA y con la extensión 113 de JORGE TOCTO (Véase figura 59).



Figura 59. Extensiones conectadas a central telefónica móvil.

En el momento en que nos enlazamos con la extensión 101 nos aparece la figura 60.

```
root@elx~  
login as: root  
Access denied  
root@192.168.1.251's password:  
Access denied  
root@192.168.1.251's password:  
Welcome to Elastix  
-----  
Elastix is a product meant to be configured through a web browser.  
Any changes made from within the command line may corrupt the system  
configuration and produce unexpected behavior; in addition, changes  
made to system files through here may be lost when doing an update.  
  
To access your Elastix System, using a separate workstation (PC/MAC/Linux)  
Open the Internet Browser using the following URL:  
http://192.168.1.251  
  
[root@elx ~]# asterisk -rvvddd  
Parsing /etc/asterisk/asterisk.conf  
Seeding global EID '88:27:ebfd:ab:64' from 'eth0' using 'siocgifwaddr'  
Asterisk 11.4.0, Copyright (C) 1999 - 2012 Digium, Inc. and others.  
Created by Mark Spencer <markster@digium.com>  
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for details.  
This is free software, with components licensed under the GNU General Public  
License version 2 and other licenses; you are welcome to redistribute it under  
certain conditions. Type 'core show license' for details.  
-----  
Connected to Asterisk 11.4.0 currently running on elx (pid = 1368)  
-- Registered SIP '113' at 192.168.1.103:62523  
-- Remote UNIX connection  
-- Remote UNIX connection disconnected  
elx*CLI> █
```

Figura 60. Código generado por consola en asterisk para extensión 101.

Al conectarnos con la extensión 113 nos da la figura 61.

```
login as: root
Access denied
root@192.168.1.251's password:
Access denied
root@192.168.1.251's password:

Welcome to Elastix
-----
Elastix is a product meant to be configured through a web browser.
Any changes made from within the command line may corrupt the system
configuration and produce unexpected behavior; in addition, changes
made to system files through here may be lost when doing an update.

To access your Elastix System, using a separate workstation (PC/MAC/Linux)
Open the Internet Browser using the following URL:
http://192.168.1.251

[root@elx ~]# asterisk -rvvvddd
Parsing /etc/asterisk/asterisk.conf
Seeding global SIP '88377eb5fcb464' from 'echo' using 'sioconfigwaddr'
Asterisk 11.4.0, Copyright (C) 1999 - 2012 Digium, Inc. and others.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for details.
This is free software, with components licensed under the GNU General Public
License version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.

=====
Connected to Asterisk 11.4.0 currently running on elx (pid = 1368)
-- Registered SIP '113' at 192.168.1.103:62523
-- Remote UNIX connection
-- Remote UNIX connection disconnected
-- Registered SIP '101' at 192.168.1.104:55659
elx*CLI>
```

Figura 61. Código generado por consola en asterisk para extensión 113

Al marcar a nuestro CALL CENTER que es la extensión 100 nos da como resultado la figura 62, aquí podrás observar cómo se conectan con los IVR que se crearon anteriormente.

```
-- Goto (macro-dial,s,3)
-- Executing [s@macro-dial:3] AGI("SIP/113-00000000", "dialparties.agi") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/dialparties.agi
dialparties.agi: Starting New Dialparties.agi
dialparties.agi: Caller ID name is 'JORGE TOCTO' number is '1023'
dialparties.agi: Methodology of ring is 'ringallv2'
-- dialparties.agi: Added extension 100 to extension map
-- dialparties.agi: Extension 100 CF is disabled
-- dialparties.agi: Extension 100 do not disturb is disabled
dialparties.agi: EXTENSION_STATE: 4 (UNAVAILABLE)
dialparties.agi: Extension 100 has ExtensionState: 4
-- dialparties.agi: Checking CW and CFB status for extension 100
-- dialparties.agi: dset CALLTRACE/100 to 1023
-- dialparties.agi: Filtered AMGS: 100
<SIP/113-00000000>AGI Script dialparties.agi completed, returning 0
-- Executing [s@macro-dial:7] Dial("SIP/113-00000000", "SIP/100,22,tcM(auto-blkvm)") in new stack
Everyone is busy/congested at this time (110/0/1)
-- Executing [s@macro-dial:8] Set("SIP/113-00000000", "DIALSTATUS=CHANUNAVAIL") in new stack
-- Executing [s@macro-dial:9] GosubIf("SIP/113-00000000", "07CHANUNAVAIL,1") in new stack
-- Executing [1008from-internal:19] Goto("SIP/113-00000000", "nextstep") in new stack
-- Goto (from-internal:100,22)
-- Executing [1008from-internal:21] Set("SIP/113-00000000", "RingGroupMethods") in new stack
-- Executing [1008from-internal:22] GotoIf("SIP/113-00000000", "07nodest") in new stack
-- Executing [1008from-internal:23] Set("SIP/113-00000000", "NODEST=") in new stack
-- Executing [1008from-internal:24] NoOp("SIP/113-00000000", "Deleting: BLKVM/100/SIP/113-00000000 TRUE") in new stack
-- Executing [1008from-internal:25] Goto("SIP/113-00000000", "lvz-7,s,1") in new stack
-- Goto (lvz-7,s,1)
-- Executing [s@ivr-7:1] Set("SIP/113-00000000", "MSG=custom/MENU_PRINCIPAL") in new stack
-- Executing [s@ivr-7:2] Set("SIP/113-00000000", "LOOPCOUNT=0") in new stack
-- Executing [s@ivr-7:3] Set("SIP/113-00000000", "DIR-CONTEXT=") in new stack
-- Executing [s@ivr-7:4] Set("SIP/113-00000000", "IVR_CONTEXT=ivr-7") in new stack
-- Executing [s@ivr-7:5] Set("SIP/113-00000000", "IVR_CONTEXT=ivr-7") in new stack
-- Executing [s@ivr-7:6] GotoIf("SIP/113-00000000", "0begin") in new stack
-- Executing [s@ivr-7:7] Answer("SIP/113-00000000", "") in new stack
-- Executing [s@ivr-7:8] Wait("SIP/113-00000000", "1") in new stack
-- Executing [s@ivr-7:9] Set("SIP/113-00000000", "TIMEOUT(digit)=3") in new stack
-- Digit timeout set to 3.000
-- Executing [s@ivr-7:10] Set("SIP/113-00000000", "TIMEOUT(response)=10") in new stack
-- Response timeout set to 10.000
-- Executing [s@ivr-7:11] Set("SIP/113-00000000", " IVR RETVM=") in new stack
-- Executing [s@ivr-7:12] ExecIf("SIP/113-00000000", "12Background(custom/MENU_PRINCIPAL)") in new stack
-- <SIP/113-00000000> Playing 'custom/MENU_PRINCIPAL.slin' (language 'en')
elx*CLI>
```

Figura 62. Código generado al llamar a nuestro CALL CENTER.

En la figura 63 se hace la llamada de nuestra extensión 113 de JORGE TOCTO a nuestra extensión 101 ANA TROYA donde podemos ver cómo funciona nuestro Asterisk.

```
root@elc~
-- Executing [dstring@macro-dial-one:3] ExecIf("SIP/113-00000001", "0?Return()") in new stack
-- Executing [dstring@macro-dial-one:4] ExecIf("SIP/113-00000001", "0?Set(DEVICES=01)") in new stack
-- Executing [dstring@macro-dial-one:5] Set("SIP/113-00000001", "LOOPCNT=1") in new stack
-- Executing [dstring@macro-dial-one:6] Set("SIP/113-00000001", "ITER=1") in new stack
-- Executing [dstring@macro-dial-one:7] Set("SIP/113-00000001", "THISDIAL=SIP/101") in new stack
-- Executing [dstring@macro-dial-one:8] GosubIf("SIP/113-00000001", "1?zap2dahdi,1") in new stack
-- Executing [zap2dahdi@macro-dial-one:1] ExecIf("SIP/113-00000001", "0?Return()") in new stack
-- Executing [zap2dahdi@macro-dial-one:2] Set("SIP/113-00000001", "NEWDIAL=") in new stack
-- Executing [zap2dahdi@macro-dial-one:3] Set("SIP/113-00000001", "LOOPCNT=1") in new stack
-- Executing [zap2dahdi@macro-dial-one:4] Set("SIP/113-00000001", "ITER=1") in new stack
-- Executing [zap2dahdi@macro-dial-one:5] Set("SIP/113-00000001", "THISPART=SIP/101") in new stack
-- Executing [zap2dahdi@macro-dial-one:6] ExecIf("SIP/113-00000001", "0?Set(THISPART=DAHDI/101)") in new stack
-- Executing [zap2dahdi@macro-dial-one:7] Set("SIP/113-00000001", "NEWDIAL=SIP/101") in new stack
-- Executing [zap2dahdi@macro-dial-one:8] Set("SIP/113-00000001", "ITER=2") in new stack
-- Executing [zap2dahdi@macro-dial-one:9] GosubIf("SIP/113-00000001", "1?zap2dahdi,1") in new stack
-- Executing [zap2dahdi@macro-dial-one:10] Set("SIP/113-00000001", "THISDIAL=SIP/101") in new stack
-- Executing [zap2dahdi@macro-dial-one:11] Return("SIP/113-00000001", "") in new stack
-- Executing [dstring@macro-dial-one:9] Set("SIP/113-00000001", "DSTRING=SIP/101") in new stack
-- Executing [dstring@macro-dial-one:10] Set("SIP/113-00000001", "ITER=2") in new stack
-- Executing [dstring@macro-dial-one:11] GotoIf("SIP/113-00000001", "0?begin") in new stack
-- Executing [dstring@macro-dial-one:12] Set("SIP/113-00000001", "DSTRING=SIP/101") in new stack
-- Executing [dstring@macro-dial-one:13] Return("SIP/113-00000001", "") in new stack
-- Executing [s@macro-dial-one:27] GotoIf("SIP/113-00000001", "0?nodial") in new stack
-- Executing [s@macro-dial-one:28] GotoIf("SIP/113-00000001", "0?skiptrace") in new stack
-- Executing [s@macro-dial-one:29] GosubIf("SIP/113-00000001", "1?cset,1?ctclear,1") in new stack
-- Executing [cset@macro-dial-one:1] Set("SIP/113-00000001", "DB(CALLTRACE/101)=1023") in new stack
-- Executing [cset@macro-dial-one:2] Return("SIP/113-00000001", "") in new stack
-- Executing [s@macro-dial-one:30] Set("SIP/113-00000001", "D_OPTIONS=tr") in new stack
-- Executing [s@macro-dial-one:31] ExecIf("SIP/113-00000001", "0?SIPAddHeader(Alert-Info: )") in new stack
-- Executing [s@macro-dial-one:32] ExecIf("SIP/113-00000001", "0?SIPAddHeader()") in new stack
-- Executing [s@macro-dial-one:33] ExecIf("SIP/113-00000001", "0?Set(CHANNEL=musicclass)") in new stack
-- Executing [s@macro-dial-one:34] GosubIf("SIP/113-00000001", "0?wait,1") in new stack
-- Executing [s@macro-dial-one:35] Set("SIP/113-00000001", "CHANNELS=") in new stack
-- Executing [s@macro-dial-one:36] Set("SIP/113-00000001", "RECEIVED=TRUE") in new stack
-- Executing [s@macro-dial-one:37] Dial("SIP/113-00000001", "SIP/101,,,,ct") in new stack
== Using SIP RTP TOS bits 184
== Using SIP RTP CoS mark 5
== Called SIP/101
-- SIP/101-00000002 is ringing
-- SIP/101-00000002 answered SIP/113-00000001
-- Remote UNIX connection
-- Remote UNIX connection disconnected
elc*CLI>
```

Figura 63. Código generado al llamar de la extensión 101 a la extensión 113.

Y por último descolgamos la llamada quedándonos como resultado la figura 64.

```

root@elc-
-- Executing [c:set@macro-dial-one:1] Set("SIP/113-00000001", "DB(CALLTRACE/101)=1023") in new stack
-- Executing [c:set@macro-dial-one:2] Return("SIP/113-00000001", "") in new stack
-- Executing [s@macro-dial-one:30] Set("SIP/113-00000001", "D_OPTIONS=tr") in new stack
-- Executing [s@macro-dial-one:31] ExecIf("SIP/113-00000001", "0?SIPAddHeader(Aster-Info: )") in new stack
-- Executing [s@macro-dial-one:32] ExecIf("SIP/113-00000001", "0?SIPAddHeader()") in new stack
-- Executing [s@macro-dial-one:33] ExecIf("SIP/113-00000001", "0?Set(CHANNEL(musicclass)=)") in new stack
-- Executing [s@macro-dial-one:34] GosubIf("SIP/113-00000001", "0?qwait,1") in new stack
-- Executing [s@macro-dial-one:35] Set("SIP/113-00000001", "CWIGNORE=") in new stack
-- Executing [s@macro-dial-one:36] Set("SIP/113-00000001", "REPECID=TRUE") in new stack
-- Executing [s@macro-dial-one:37] Dial("SIP/113-00000001", "SIP/101","",tz) in new stack
== Using SIP RTP TOS bits 184
== Using SIP RTP CoS mark 5
== Called SIP/101
-- SIP/101-00000002 is ringing
-- SIP/101-00000002 answered SIP/113-00000001
-- Remote UNIX connection
-- Remote UNIX connection disconnected
-- Executing [h@macro-dial-one:1] Macro("SIP/113-00000001", "hangupcall",) in new stack
-- Executing [s@macro-hangupcall:1] GotoIf("SIP/113-00000001", "1?endmixmoncheck") in new stack
-- Goto (macro-hangupcall,s,9)
-- Executing [s@macro-hangupcall:9] NoOp("SIP/113-00000001", "End of MIXMON check") in new stack
-- Executing [s@macro-hangupcall:10] GotoIf("SIP/113-00000001", "1?hometemnon") in new stack
-- Goto (macro-hangupcall,s,28)
-- Executing [s@macro-hangupcall:28] NoOp("SIP/113-00000001", "End of MEETME check") in new stack
-- Executing [s@macro-hangupcall:29] GotoIf("SIP/113-00000001", "1?noautomon") in new stack
-- Goto (macro-hangupcall,s,39)
-- Executing [s@macro-hangupcall:34] NoOp("SIP/113-00000001", "TOUCH MONITOR OUTPUT=") in new stack
-- Executing [s@macro-hangupcall:35] GotoIf("SIP/113-00000001", "1?noautomon2") in new stack
-- Goto (macro-hangupcall,s,41)
-- Executing [s@macro-hangupcall:41] NoOp("SIP/113-00000001", "MONITOR FILENAME=") in new stack
-- Executing [s@macro-hangupcall:42] GotoIf("SIP/113-00000001", "1?skiprg") in new stack
-- Goto (macro-hangupcall,s,45)
-- Executing [s@macro-hangupcall:45] GotoIf("SIP/113-00000001", "1?skipblkvm") in new stack
-- Goto (macro-hangupcall,s,48)
-- Executing [s@macro-hangupcall:48] GotoIf("SIP/113-00000001", "1?cheend") in new stack
-- Goto (macro-hangupcall,s,50)
-- Executing [s@macro-hangupcall:50] AGI("SIP/113-00000001", "hangup.agi") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/hangup.agi
-- <SIP/113-00000001>AGI Script hangup.agi completed, returning 0
-- Executing [s@macro-hangupcall:51] Hangup("SIP/113-00000001", "") in new stack
== Spawn extension (macro-hangupcall, s, 51) exited non-zero on 'SIP/113-00000001' in macro 'hangupcall'
== Spawn extension (macro-dial-one, h, 1) exited non-zero on 'SIP/113-00000001'
cli*cli>

```

Figura 64. Código generado al terminar llamada.

3. Fase 3: Implantación de la Solución Planteada

DISEÑO A IMPLEMENTAR DE TELEFONÍA MÓVIL



Figura 65. Diseño a implementar la telefonía móvil.

3.1. Se armó y se ubicó el equipo raspberry pi en el lugar donde funciona el sistema de telefonía.

1. En este caso para instalar el raspberry pi, se realiza el proceso para enlazarlo a nuestro raspberry pi a la red mediante un cable de red a un switch interno donde va también el servidor existente en el AEIRNNR, tenemos también un teléfono ip grandstream gxp 1400 el mismo que utilizan los docentes, administrativos y dos dispositivos móviles.



Figura 66. Equipos necesarios para la simulación.

2. Configuración de enlace entre el servidor de telefonía ip fija y de la red telefónica móvil utilizando μ elastix.

Para la configuración de enlace de nuestra red telefónica móvil y de la telefonía ip fija, se necesita crear una troncal y una ruta de salida tanto el servidor de telefonía de red fija como para el raspberry pi, esto con el fin de poder comunicarnos con la telefonía ip fija y móvil.

Al crear una troncal IAX2 de nombre “valparaiso” en el servidor de telefonía ip fija (véase figura 67).

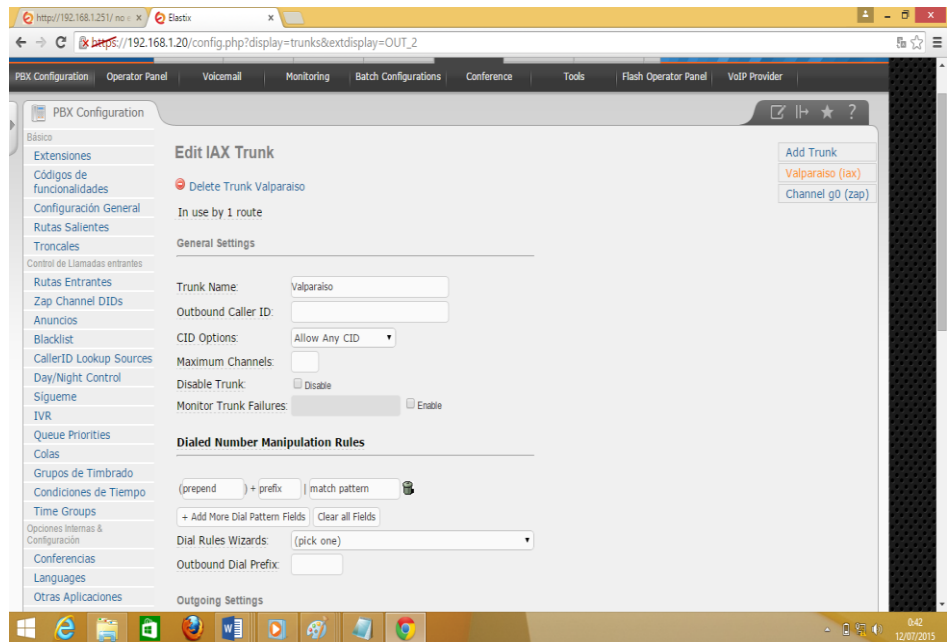


Figura 67. Crear Troncal IAX2 llamada Valparaiso.

En la figura 68 se llenan los campos con el siguiente código en:

ServidorA:

TRUNK NAME: Valparaiso

username=Valparaiso

type=peer

secret=123456

qualify=yes

host=192.168.1.251 (ip de mi raspberry pi)

context=from-internal

trunk=yes

disallow=all

allow=gsm&ilbc

User Context : Vinadelmar

USER DETAILS

type=user

secret=123456

host=192.168.1.251 (ip de mi raspberry pi)

context=from-internal

disallow=all

allow=gsm&ilbc

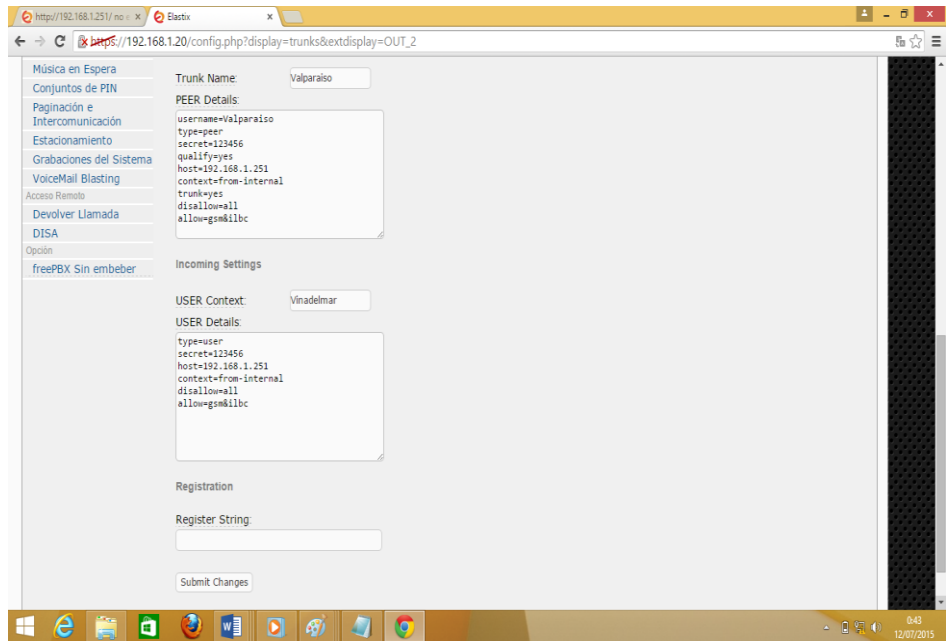


Figura 68. Código para enlazar al elastix del raspberry pi.

Luego creamos una ruta de salida que se muestra en la figura 69, teniendo en cuenta que en la parte del Dial Patterns that will use this Route, en este caso en el segundo espacio se ha llenado con 2, eso significa que para hacer la llamada a nuestro raspberry pi tendremos que anteponer el número 2 y seguido el número de la extensión a comunicarnos.

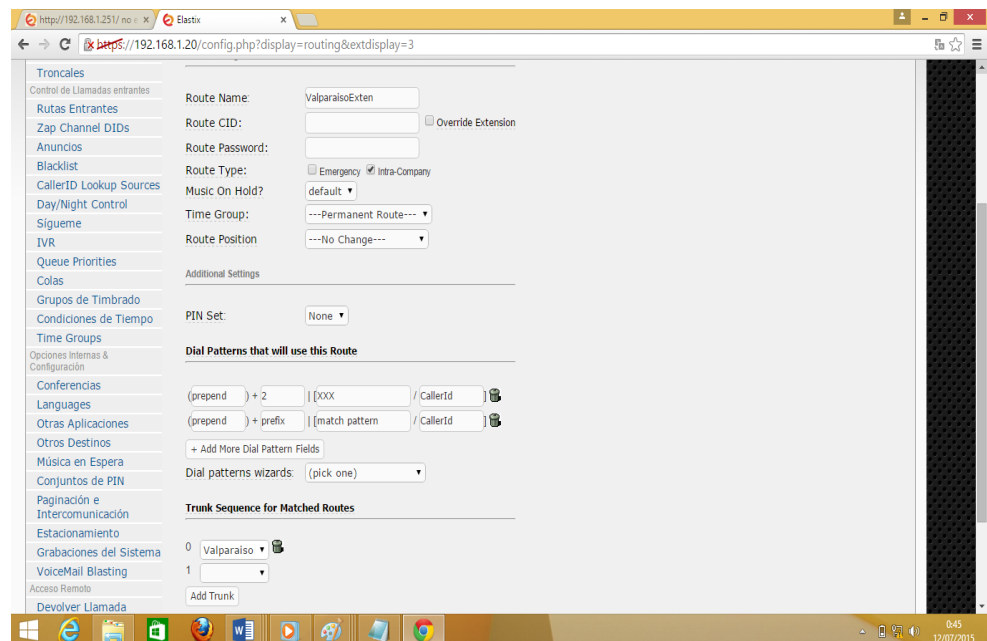


Figura 69. Configuración Ruta de salida ValparaisoExten.

Ahora vamos a nuestro segundo servidor que en este caso sería nuestro raspberry pi, se crea una troncal IAX2 de nombre vinadelmar en el servidor de telefonía ip móvil (véase figura 70).

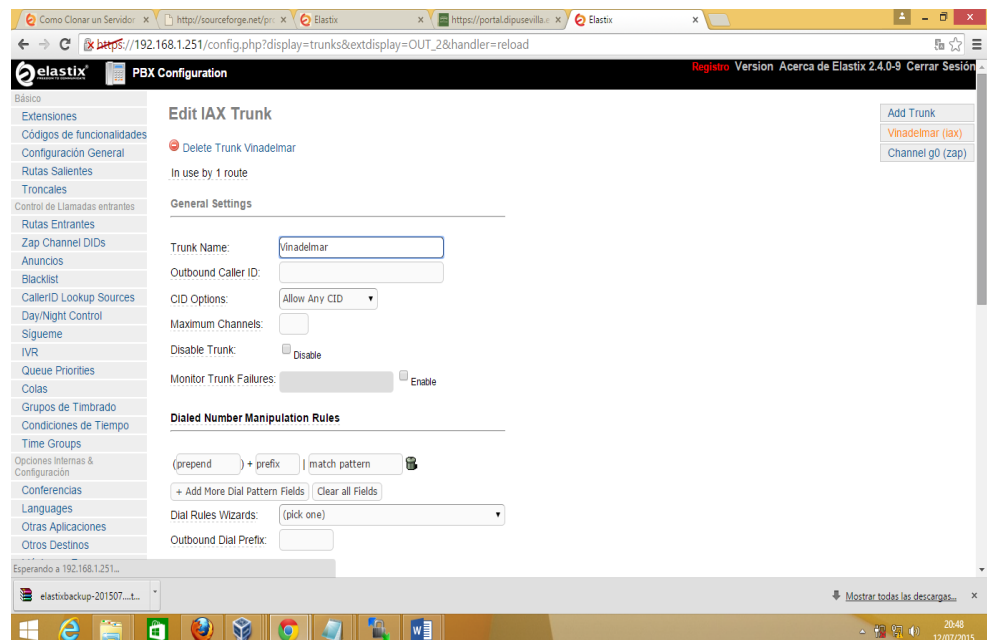


Figura 70. Crear Troncal IAX2 llamada Vinadelmar.

En la figura 71 se llenan los campos con el siguiente código en:
ServidorB:

TRUNK NAME: Vinadelmar
username=Vinadelmar
type=peer
secret=123456
qualify=yes
host=192.168.1.20 (ip de mi servidor telefonía móvil fija)
context=from-internal
trunk=yes
disallow=all
allow=gsm&ilbc

User Context : Vinadelmar

USER DETAILS

type=user
secret=123456
host=192.168.1. 20 (ip de mi servidor telefonía móvil fija)
context=from-internal
disallow=all
allow=gsm&ilbc

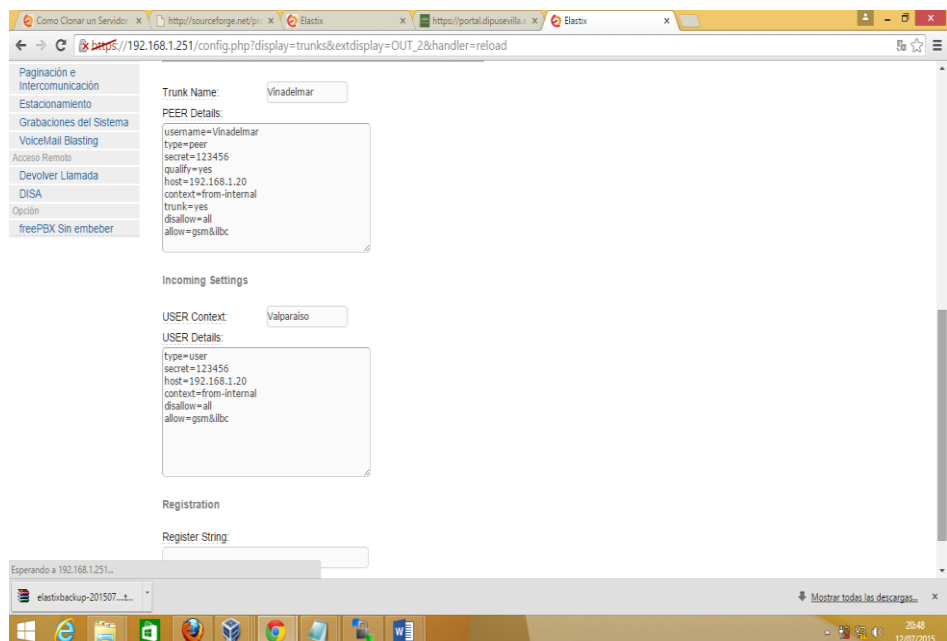


Figura 71. Código para enlazar al μelastix del servidor telefonía fija.

Luego se crean una ruta de salida que se muestra en la figura 72 teniendo en cuenta que en la parte del Dial Patterns that will use this Route, en este caso en el segundo espacio lo hemos llenado con 3, eso significa que para hacer la llamada a nuestro servidor de telefonía ip fija, se debe de anteponer el número 3 y seguido el número de la extensión a comunicarnos.

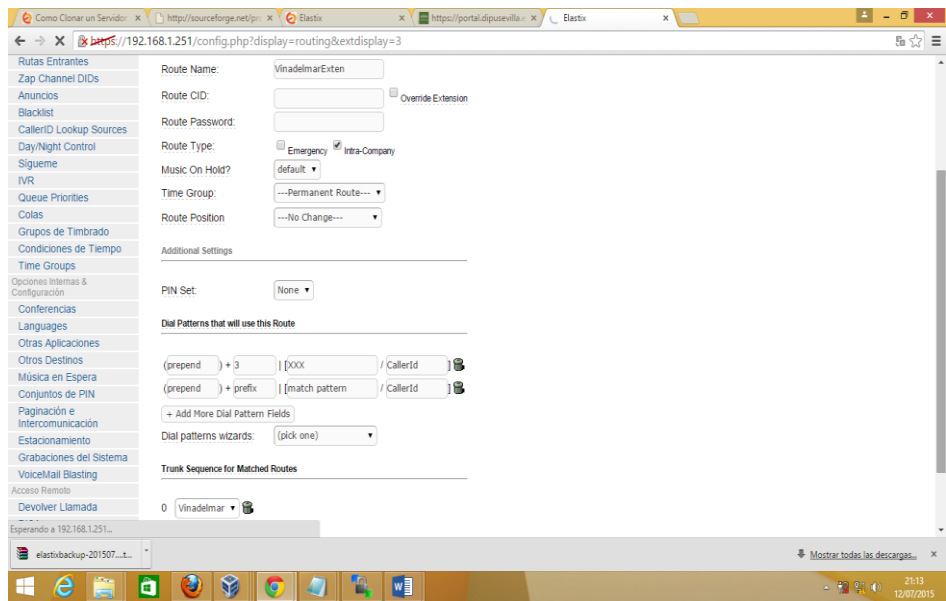


Figura 72. Configuración Ruta de salida VinadelmarExten

Como resultado de esta configuración se puede observar la figura 73, en la que se muestran nuestras dos troncales habilitadas, teniendo como resultado el enlace entre el servidor de telefonía ip fija y el raspberry pi.

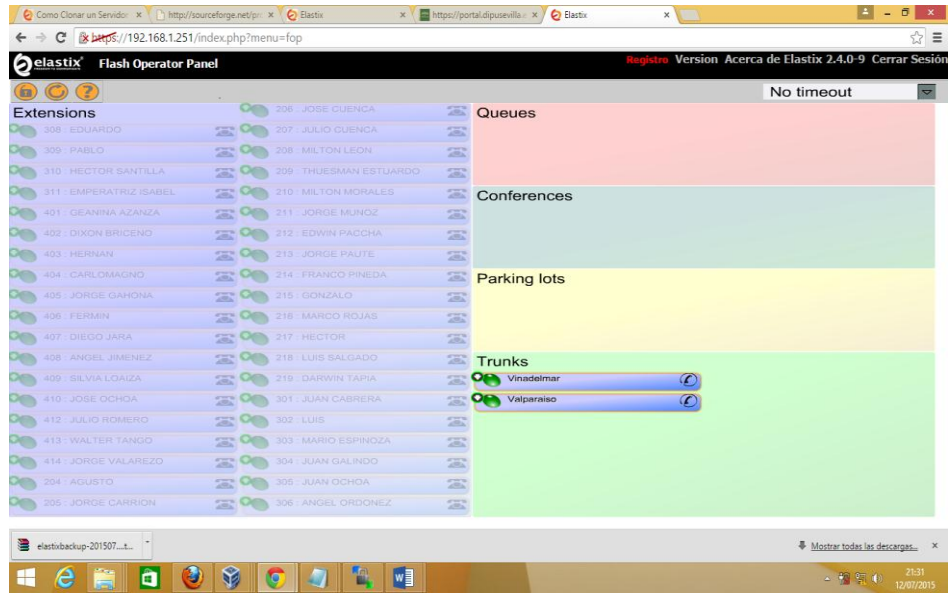


Figura 73. Troncal Valparaiso y Vinadelmar enlazadas.

3. Configuración del teléfono grandstream GXP 1400

Para poder configurar se abre cualquier navegador web, ingresamos la ip del teléfono grandstream GXP 1400, en nuestro caso la ip es 192.168.1.100 (Véase figura 74).

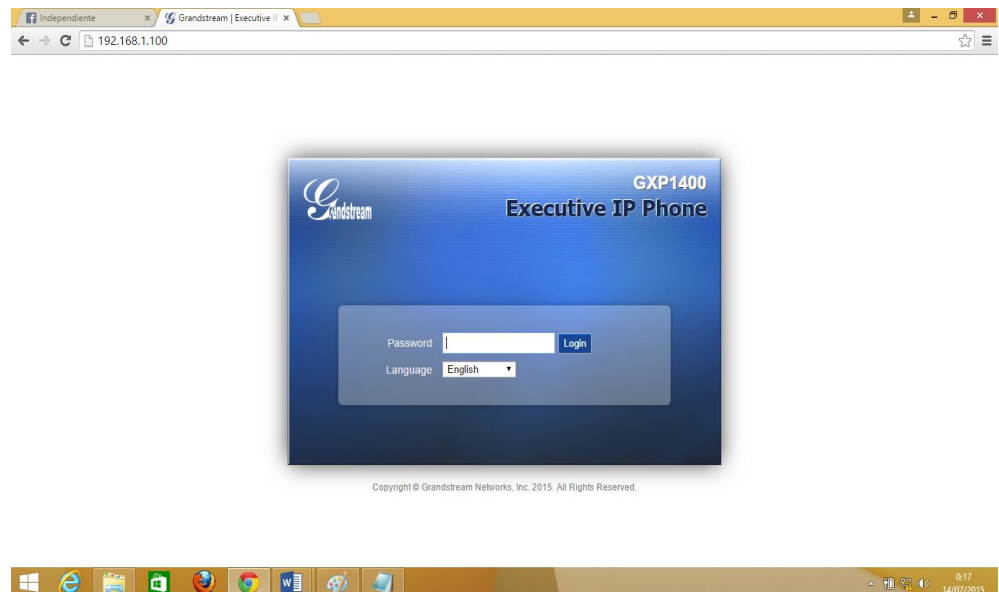


Figura 74. Ingreso a la interface del teléfono Grandstream GXP 1400.

Como muestra la figura 75 no se habilitado ninguno de los dos Account's.

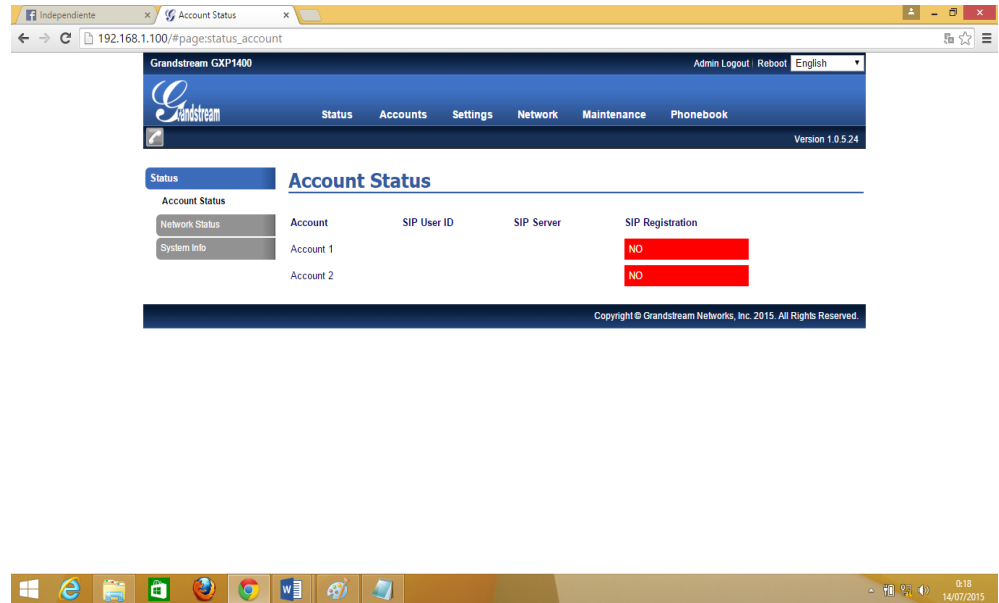


Figura 75. Estado de nuestro teléfono Grandstream GXP 1400.

Para habilitar el Account 1, se debe dirigir a la pestaña Accounts y seleccionamos Account 1 y luego en la opción General Settings, quedándonos la figura 76.

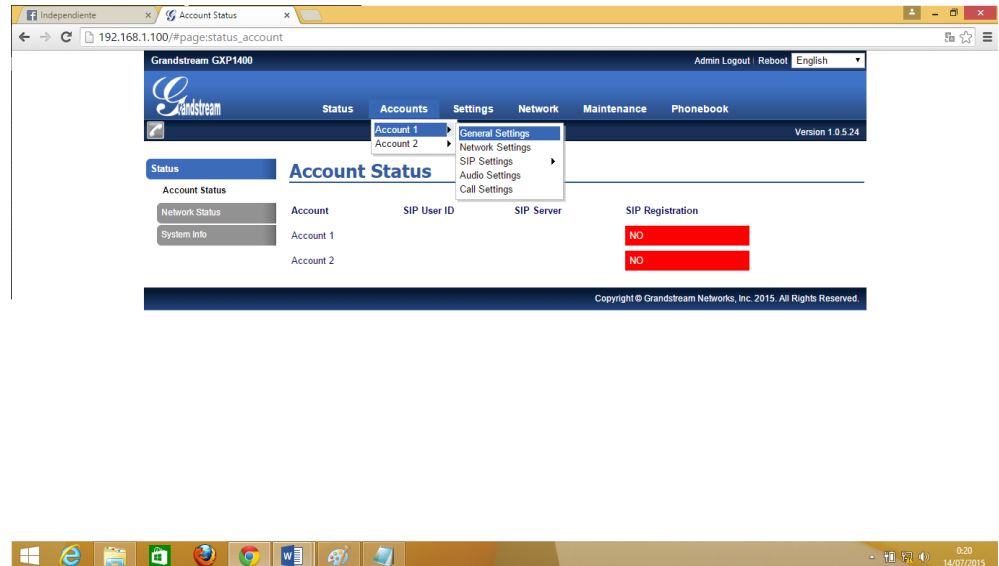


Figura 76. Entrar Account 1.

A continuación, en la figura 77, se llenan con los datos de nuestra extensión, en este caso hemos simulado una extensión SECRETARIA, donde nuestro

SIP SERVER lleva la ip 192.168.1.20 y su SIP USER ID será el número de extensión 101.

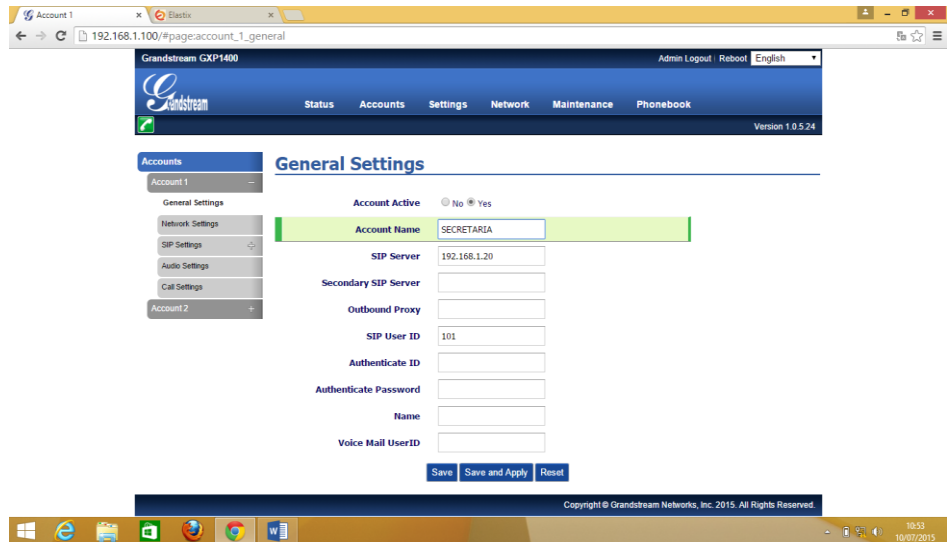


Figura 77. Configuración de nuestro Account 1.

Nuestro teléfono grandstream queda habilitado en su Account 1 (véase la figura 78).

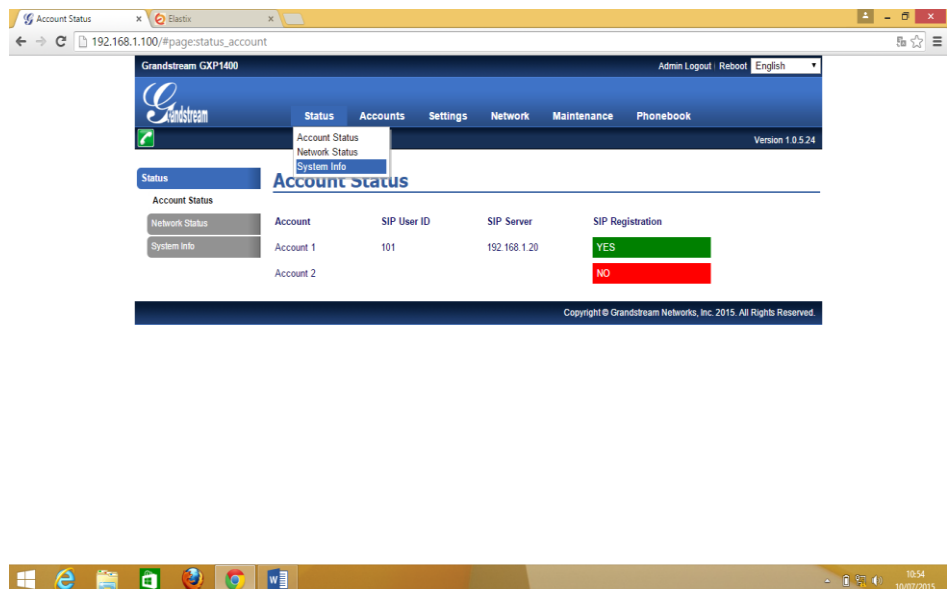


Figura 78. Account 1 habilitado.

Entrado a nuestro servidor se puede observar en la figura 79, que SECRETARIA esta activa y lista para poder comunicarse.

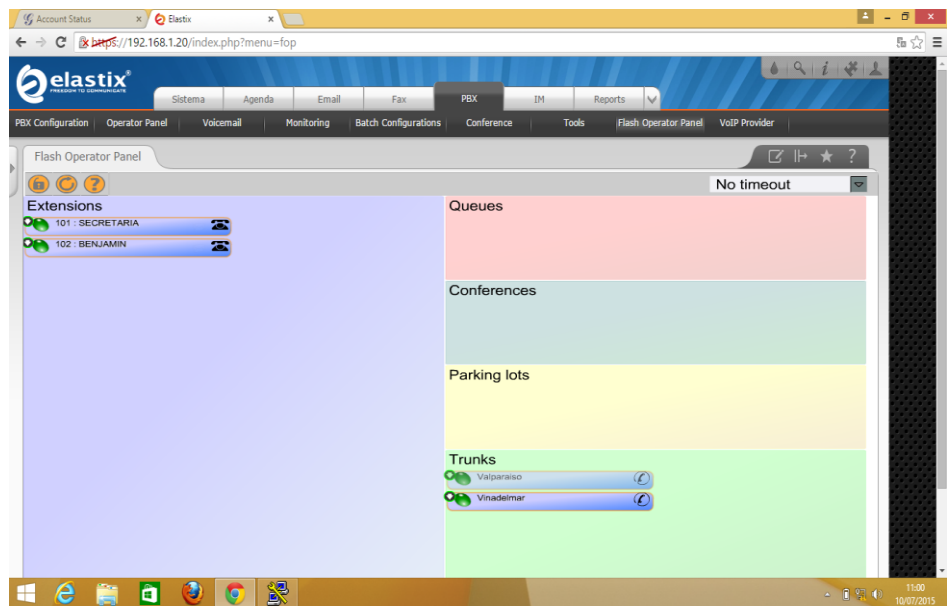


Figura 79. Teléfono Grandstream habilitado en nuestro μ elastix de telefonía fija.

4. Pruebas de validación del sistema μ elastix sobre la red telefónica fija y móvil.

En las pruebas de validación del sistema μ elastix se lo realiza de las dos formas: en consola e interfaz gráfica, desde la realización de una llamada interna o externa.

Para las diferentes pruebas se va a ingresar a la consola del μ elastix desde cualquier máquina en la red interna, por el programa putty que permite el ingreso externo a los servidores, una vez abierto con la ip μ elastix y con puerto 22 se procede a ingresar el usuario y contraseña en modo root.

4.1. Prueba de Llamadas en el raspberry pi usando elastix.

En esta prueba se realiza una llamada de la extensión 113 a la extensión 101, donde podemos observar que nuestro μ elastix hace la llamada a la extensión 101 (Véase la figura 80 y 81).

```

root@elc-
-- Executing [macro-dial-one:26] GosubIf("SIP/113-00000001", "1?dstring,1?dlocal,1") in new stack
-- Executing [dstring@macro-dial-one:1] Set("SIP/113-00000001", "DSTRING") in new stack
-- Executing [dstring@macro-dial-one:2] Set("SIP/113-00000001", "DEVICES=101") in new stack
-- Executing [dstring@macro-dial-one:3] ExecIf("SIP/113-00000001", "0?Return()") in new stack
-- Executing [dstring@macro-dial-one:4] ExecIf("SIP/113-00000001", "0?Set(DEVICES=01)") in new stack
-- Executing [dstring@macro-dial-one:5] Set("SIP/113-00000001", "LOOPCNT=1") in new stack
-- Executing [dstring@macro-dial-one:6] Set("SIP/113-00000001", "ITER=1") in new stack
-- Executing [dstring@macro-dial-one:7] Set("SIP/113-00000001", "THISDIAL=SIP/101") in new stack
-- Executing [dstring@macro-dial-one:8] GosubIf("SIP/113-00000001", "1?zap2dahdi,1") in new stack
-- Executing [zap2dahdi@macro-dial-one:1] ExecIf("SIP/113-00000001", "0?Return()") in new stack
-- Executing [zap2dahdi@macro-dial-one:2] Set("SIP/113-00000001", "NEWDIAL=") in new stack
-- Executing [zap2dahdi@macro-dial-one:3] Set("SIP/113-00000001", "LOOPCNT=1") in new stack
-- Executing [zap2dahdi@macro-dial-one:4] Set("SIP/113-00000001", "ITER=1") in new stack
-- Executing [zap2dahdi@macro-dial-one:5] Set("SIP/113-00000001", "THISPART=SIP/101") in new stack
-- Executing [zap2dahdi@macro-dial-one:6] ExecIf("SIP/113-00000001", "0?Set(THISPART=NEWDI(101))") in new stack
-- Executing [zap2dahdi@macro-dial-one:7] Set("SIP/113-00000001", "NEWDIAL=SIP/101") in new stack
-- Executing [zap2dahdi@macro-dial-one:8] Set("SIP/113-00000001", "ITER=2") in new stack
-- Executing [zap2dahdi@macro-dial-one:9] Set("SIP/113-00000001", "0?begin()") in new stack
-- Executing [zap2dahdi@macro-dial-one:10] Set("SIP/113-00000001", "0?Begin()") in new stack
-- Executing [zap2dahdi@macro-dial-one:11] Return("SIP/113-00000001", "") in new stack
-- Executing [dstring@macro-dial-one:9] Set("SIP/113-00000001", "DSTRING=SIP/101") in new stack
-- Executing [dstring@macro-dial-one:10] Set("SIP/113-00000001", "ITER=2") in new stack
-- Executing [dstring@macro-dial-one:11] Set("SIP/113-00000001", "0?begin()") in new stack
-- Executing [dstring@macro-dial-one:12] Set("SIP/113-00000001", "DSTRING=SIP/101") in new stack
-- Executing [dstring@macro-dial-one:13] Return("SIP/113-00000001", "") in new stack
-- Executing [macro-dial-one:27] GotoIf("SIP/113-00000001", "0?nodial") in new stack
-- Executing [macro-dial-one:28] GotoIf("SIP/113-00000001", "0?starttrace") in new stack
-- Executing [macro-dial-one:29] GosubIf("SIP/113-00000001", "1?cset,1?ctclear,1") in new stack
-- Executing [cset@macro-dial-one:1] Set("SIP/113-00000001", "DB(CALLTRACE/101)=1023") in new stack
-- Executing [cset@macro-dial-one:2] Return("SIP/113-00000001", "") in new stack
-- Executing [macro-dial-one:30] Set("SIP/113-00000001", "0?OPTION=cr") in new stack
-- Executing [macro-dial-one:31] ExecIf("SIP/113-00000001", "0?SIPAddHeader(Alert-Info: )") in new stack
-- Executing [macro-dial-one:32] ExecIf("SIP/113-00000001", "0?SIPAddHeader()") in new stack
-- Executing [macro-dial-one:33] ExecIf("SIP/113-00000001", "0?Set(CHANID(musicclass)=)") in new stack
-- Executing [macro-dial-one:34] GosubIf("SIP/113-00000001", "0?wait,1") in new stack
-- Executing [macro-dial-one:35] Set("SIP/113-00000001", "CWIGNORE=1") in new stack
-- Executing [macro-dial-one:36] Set("SIP/113-00000001", "KEEPID=TRUE") in new stack
-- Executing [macro-dial-one:37] Dial("SIP/113-00000001", "SIP/101,,cr") in new stack
== Using SIP RTP TOS bits 184
== Using SIP RTP CoS mark 5
-- Called SIP/101
-- SIP/101-00000002 is ringing
elc*CLI>

```

Figura 80. Llamada entre extensión 113 a 101

```

root@elc-
-- Executing [cset@macro-dial-one:1] Set("SIP/113-00000001", "DB(CALLTRACE/101)=1023") in new stack
-- Executing [cset@macro-dial-one:2] Return("SIP/113-00000001", "") in new stack
-- Executing [macro-dial-one:30] Set("SIP/113-00000001", "0?OPTION=cr") in new stack
-- Executing [macro-dial-one:31] ExecIf("SIP/113-00000001", "0?SIPAddHeader(Alert-Info: )") in new stack
-- Executing [macro-dial-one:32] ExecIf("SIP/113-00000001", "0?SIPAddHeader()") in new stack
-- Executing [macro-dial-one:33] ExecIf("SIP/113-00000001", "0?Set(CHANID(musicclass)=)") in new stack
-- Executing [macro-dial-one:34] GosubIf("SIP/113-00000001", "0?wait,1") in new stack
-- Executing [macro-dial-one:35] Set("SIP/113-00000001", "CWIGNORE=1") in new stack
-- Executing [macro-dial-one:36] Set("SIP/113-00000001", "KEEPID=TRUE") in new stack
-- Executing [macro-dial-one:37] Dial("SIP/113-00000001", "SIP/101,,cr") in new stack
== Using SIP RTP TOS bits 184
== Using SIP RTP CoS mark 5
-- Called SIP/101
-- SIP/101-00000002 is ringing
-- SIP/101-00000002 answered SIP/113-00000001
-- Remote UNIX connection
-- Remote UNIX connection disconnected
-- Executing [macro-dial-one:1] Macro("SIP/113-00000001", "hangupcall,") in new stack
-- Executing [macro-hangupcall:1] GotoIf("SIP/113-00000001", "1?endmixmoncheck") in new stack
-- Goto (macro-hangupcall,s,9)
-- Executing [macro-hangupcall:5] NoOp("SIP/113-00000001", "End of MIXMON check") in new stack
-- Executing [macro-hangupcall:10] GotoIf("SIP/113-00000001", "1?noetmeetmon") in new stack
-- Goto (macro-hangupcall,s,28)
-- Executing [macro-hangupcall:28] NoOp("SIP/113-00000001", "End of MEETME check") in new stack
-- Executing [macro-hangupcall:29] GotoIf("SIP/113-00000001", "1?noautomon") in new stack
-- Goto (macro-hangupcall,s,34)
-- Executing [macro-hangupcall:34] NoOp("SIP/113-00000001", "TOUCH_MONITOR_OUTPUT=") in new stack
-- Executing [macro-hangupcall:35] GotoIf("SIP/113-00000001", "1?noautomon2") in new stack
-- Goto (macro-hangupcall,s,41)
-- Executing [macro-hangupcall:41] NoOp("SIP/113-00000001", "MONITOR_FILENAME=") in new stack
-- Executing [macro-hangupcall:42] GotoIf("SIP/113-00000001", "1?skipprg") in new stack
-- Goto (macro-hangupcall,s,45)
-- Executing [macro-hangupcall:45] GotoIf("SIP/113-00000001", "1?skipbkvm") in new stack
-- Goto (macro-hangupcall,s,48)
-- Executing [macro-hangupcall:48] GotoIf("SIP/113-00000001", "1?cheend") in new stack
-- Goto (macro-hangupcall,s,50)
-- Executing [macro-hangupcall:50] AGI("SIP/113-00000001", "hangup.agi") in new stack
-- Launched AGI Script /var/lib/asterisk/agi-bin/hangup.agi
-- SIP/113-00000001>AGI Script hangup.agi completed, returning 0
-- Executing [macro-hangupcall:51] Hangup("SIP/113-00000001", "") in new stack
== Spawn extension (macro-hangupcall, s, 51) exited non-zero on 'SIP/113-00000001' in macro 'hangupcall'
== Spawn extension (macro-dial-one, h, 1) exited non-zero on 'SIP/113-00000001'
elc*CLI>

```

Figura 81. Llamada entre extensión 101 a 113.

En la figura 82, podemos observar que los dos teléfonos móviles están comunicándose.

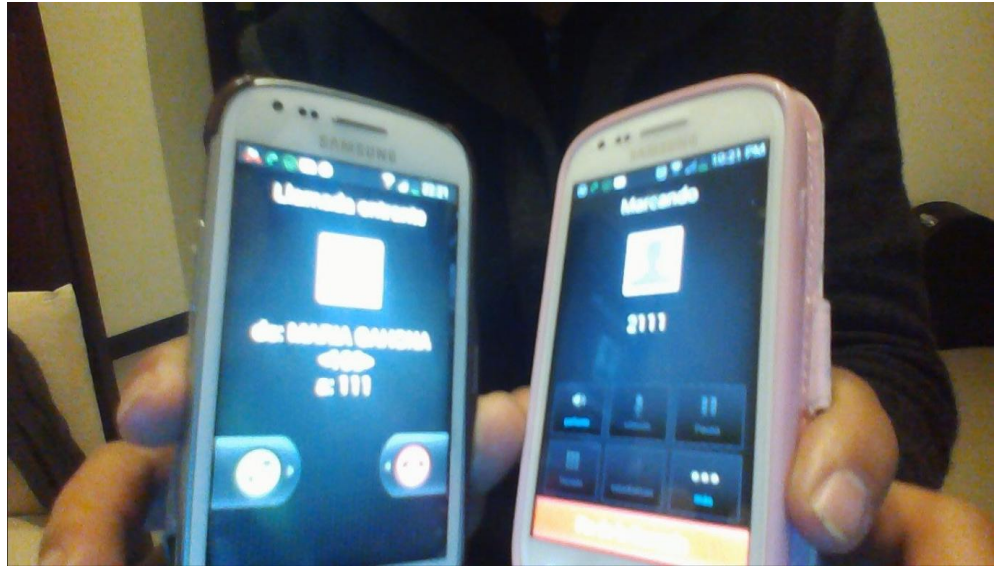


Figura 82. Prueba llamada entre dispositivos móviles.

4.2. Pruebas de llamadas de teléfono grandstream GXP 1400 a dispositivo móvil.

En la figura 83, tendremos que el teléfono grandstream GXP 1400, marca la extensión 113, anteponiendo en número “2”, para poder comunicarse con nuestro segundo servidor, que en nuestro caso es el del raspberry pi.



Figura 83. Llamada entre el teléfono grandstream gxp 1400 y dispositivo móvil.

En la figura 84, se genera el código tanto en el terminal del servidor de telefonía fija y del raspberry pi. Se puede observar que para entrar la llamada a la extensión 113 primero se comunica con nuestra troncal IAX2 llamada Valparaiso y luego hace la llamada a la extensión 113.

```

root@elk-
-- Executing [dstring@macro-dial-one:9] Set("IAX2/Valparaiso-2821", "DSTRING=SIP/111") in new stack
-- Executing [dstring@macro-dial-one:10] Set("IAX2/Valparaiso-2821", "ITER=2") in new stack
-- Executing [dstring@macro-dial-one:11] GotoIf("IAX2/Valparaiso-2821", "0?begin") in new stack
-- Executing [dstring@macro-dial-one:12] Set("IAX2/Valparaiso-2821", "DSTRING=SIP/111") in new stack
-- Executing [dstring@macro-dial-one:13] Return("IAX2/Valparaiso-2821", "") in new stack
-- Executing [s@macro-dial-one:27] GotoIf("IAX2/Valparaiso-2821", "0?nodial") in new stack
-- Executing [s@macro-dial-one:28] GotoIf("IAX2/Valparaiso-2821", "0?skiptrace") in new stack
-- Executing [s@macro-dial-one:29] GosubIf("IAX2/Valparaiso-2821", "1?set,1?clear,1") in new stack
-- Executing [cset@macro-dial-one:1] Set("IAX2/Valparaiso-2821", "DB(CALLTRACE/111)=103") in new stack
-- Executing [cset@macro-dial-one:2] Return("IAX2/Valparaiso-2821", "") in new stack
-- Executing [s@macro-dial-one:30] Set("IAX2/Valparaiso-2821", "D_OPTIONS=tr") in new stack
-- Executing [s@macro-dial-one:31] ExecIf("IAX2/Valparaiso-2821", "0?SIPAddHeader(Alert-Info: 1)") in new stack
-- Executing [s@macro-dial-one:32] ExecIf("IAX2/Valparaiso-2821", "0?SIPAddHeader()") in new stack
-- Executing [s@macro-dial-one:33] ExecIf("IAX2/Valparaiso-2821", "0?Set(CHANNEL(musicclass)=)") in new stack
-- Executing [s@macro-dial-one:34] GosubIf("IAX2/Valparaiso-2821", "0?wait,1") in new stack
-- Executing [s@macro-dial-one:35] Set("IAX2/Valparaiso-2821", "CWIGNORE=") in new stack
-- Executing [s@macro-dial-one:36] Set("IAX2/Valparaiso-2821", "KEEPID=TRUE") in new stack
-- Executing [s@macro-dial-one:37] Dial("IAX2/Valparaiso-2821", "SIP/111,,,,tr") in new stack
== Using SIP RTP CoS mark 5
== Using SIP RTP ToS bits 184
-- Called SIP/111
elk*CLI>

root@root-
-- Executing [s@macro-dialout-trunk:5] Set("SIP/103-000002d", "DIAL_TRUNK_OPTIONS=tr") in new stack
-- Executing [s@macro-dialout-trunk:6] Set("SIP/103-000002d", "OUTBOUND_GROUP_OUT_2") in new stack
-- Executing [s@macro-dialout-trunk:7] GotoIf("SIP/103-000002d", "1?nomax") in new stack
-- Goto (macro-dialout-trunk,s,9)
-- Executing [s@macro-dialout-trunk:9] GotoIf("SIP/103-000002d", "1?skipoutcid") in new stack
-- Goto (macro-dialout-trunk,s,12)
-- Executing [s@macro-dialout-trunk:12] GosubIf("SIP/103-000002d", "0?sub-fip-2,s,1") in new stack
-- Executing [s@macro-dialout-trunk:13] Set("SIP/103-000002d", "OUNUM=111") in new stack
-- Executing [s@macro-dialout-trunk:14] Set("SIP/103-000002d", "custom=IAX2/Valparaiso") in new stack
-- Executing [s@macro-dialout-trunk:15] ExecIf("SIP/103-000002d", "0?Set(DIAL_TRUNK_OPTIONS=M(setmusic"default):tr)") in new stack
-- Executing [s@macro-dialout-trunk:16] Macro("SIP/103-000002d", "dialout-trunk-predial-hook,") in new stack
-- Executing [s@macro-dialout-trunk-predial-hook:1] MacroExit("SIP/103-000002d", "") in new stack
-- Executing [s@macro-dialout-trunk:17] GotoIf("SIP/103-000002d", "0?bypass,1") in new stack
-- Executing [s@macro-dialout-trunk:18] GotoIf("SIP/103-000002d", "0?nomstrunk") in new stack
-- Executing [s@macro-dialout-trunk:19] Dial("SIP/103-000002d", "IAX2/Valparaiso/111,300,tr") in new stack
-- Called IAX2/Valparaiso/111
-- Call accepted by 192.168.1.251 (format gsm)
-- Format for call is gsm
-- IAX2/Valparaiso-20177 is ringing
  
```

Figura 84. Código generado troncal Valparaiso y extensión 113.

La llamada finaliza entre extensión 101 de la telefonía fija y extensión 113 de telefonía móvil (Vease figura 85).

```

root@elk-
-- Executing [dstring@macro-dial-one:11] GotoIf("IAX2/Valparaiso-2638", "0?begin") in new stack
-- Executing [dstring@macro-dial-one:13] Return("IAX2/Valparaiso-2638", "") in new stack
-- Executing [s@macro-dial-one:27] GotoIf("IAX2/Valparaiso-2638", "0?nodial") in new stack
-- Executing [s@macro-dial-one:28] GotoIf("IAX2/Valparaiso-2638", "0?skiptrace") in new stack
-- Executing [s@macro-dial-one:29] GosubIf("IAX2/Valparaiso-2638", "1?set,1?clear,1") in new stack
-- Executing [cset@macro-dial-one:1] Set("IAX2/Valparaiso-2638", "DB(CALLTRACE/111)=103") in new stack
-- Executing [cset@macro-dial-one:2] Return("IAX2/Valparaiso-2638", "") in new stack
-- Executing [s@macro-dial-one:30] Set("IAX2/Valparaiso-2638", "D_OPTIONS=tr") in new stack
-- Executing [s@macro-dial-one:31] ExecIf("IAX2/Valparaiso-2638", "0?SIPAddHeader(Alert-Info: 1)") in new stack
-- Executing [s@macro-dial-one:32] ExecIf("IAX2/Valparaiso-2638", "0?SIPAddHeader()") in new stack
-- Executing [s@macro-dial-one:33] ExecIf("IAX2/Valparaiso-2638", "0?Set(CHANNEL(musicclass)=)") in new stack
-- Executing [s@macro-dial-one:34] GosubIf("IAX2/Valparaiso-2638", "0?wait,1") in new stack
-- Executing [s@macro-dial-one:35] Set("IAX2/Valparaiso-2638", "CWIGNORE=") in new stack
-- Executing [s@macro-dial-one:36] Set("IAX2/Valparaiso-2638", "KEEPID=TRUE") in new stack
-- Executing [s@macro-dial-one:37] Dial("IAX2/Valparaiso-2638", "SIP/111,,,,tr") in new stack
== Using SIP RTP CoS mark 5
== Using SIP RTP ToS bits 184
-- Called SIP/111
-- SIP/111-000001d is ringing
-- SIP/111-000001d answered IAX2/Valparaiso-2638
elk*CLI>

root@root-
-- Executing [s@macro-dialout-trunk:7] GotoIf("SIP/103-000002e", "1?nomax") in new stack
-- Goto (macro-dialout-trunk,s,9)
-- Executing [s@macro-dialout-trunk:9] GotoIf("SIP/103-000002e", "1?skipoutcid") in new stack
-- Goto (macro-dialout-trunk,s,12)
-- Executing [s@macro-dialout-trunk:12] GosubIf("SIP/103-000002e", "0?sub-fip-2,s,1") in new stack
-- Executing [s@macro-dialout-trunk:13] Set("SIP/103-000002e", "OUNUM=111") in new stack
-- Executing [s@macro-dialout-trunk:14] Set("SIP/103-000002e", "custom=IAX2/Valparaiso") in new stack
-- Executing [s@macro-dialout-trunk:15] ExecIf("SIP/103-000002e", "0?Set(DIAL_TRUNK_OPTIONS=M(setmusic"default):tr)") in new stack
-- Executing [s@macro-dialout-trunk-predial-hook:1] MacroExit("SIP/103-000002e", "") in new stack
-- Executing [s@macro-dialout-trunk:17] GotoIf("SIP/103-000002e", "0?bypass,1") in new stack
-- Executing [s@macro-dialout-trunk:18] GotoIf("SIP/103-000002e", "0?nomstrunk") in new stack
-- Executing [s@macro-dialout-trunk:19] Dial("SIP/103-000002e", "IAX2/Valparaiso/111,300,tr") in new stack
-- Called IAX2/Valparaiso/111
-- Call accepted by 192.168.1.251 (format gsm)
-- Format for call is gsm
-- IAX2/Valparaiso-22366 is ringing
-- IAX2/Valparaiso-22366 answered SIP/103-000002e
-- IAX2/Valparaiso-22366 answered SIP/103-000002e
  
```

Figura 85. Llamada terminada entre ambos servidores.

En la figura 86, tendremos que el teléfono móvil, marca la extensión 101, anteponiendo en número “3”, para poder comunicarse con nuestro primer servidor, que en nuestro caso es el de telefonía móvil.



Figura 86. Llamada entre dispositivo móvil y teléfono grandstream gxp 1400

En la figura 87, se genera el código tanto en el terminal del servidor de telefonía fija y del raspberry pi. Se puede observar que para entrar la llamada a la extensión 101 primero se comunica con nuestra troncal IAX2 llamada Vinadelmar y luego hace la llamada a la extensión 101.

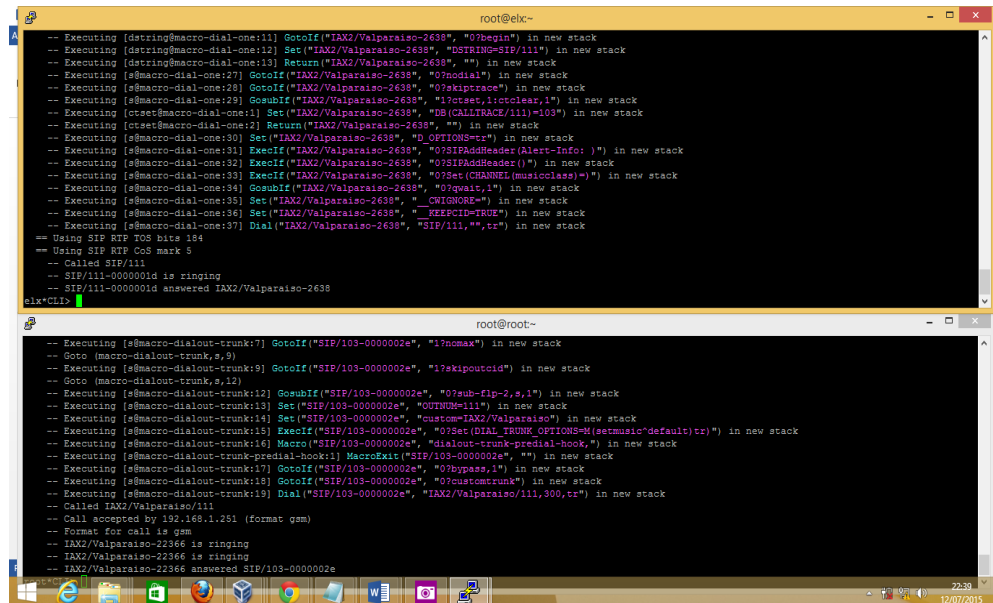
```

root@elk:~#
-- Executing [dsting@macro-dial-one:9] Set("IAX2/Valparaiso-2821", "DSTRING=SIP/111") in new stack
-- Executing [dsting@macro-dial-one:10] Set("IAX2/Valparaiso-2821", "ITER=2") in new stack
-- Executing [dsting@macro-dial-one:11] GotoIf("IAX2/Valparaiso-2821", "0?begin") in new stack
-- Executing [dsting@macro-dial-one:12] Set("IAX2/Valparaiso-2821", "DSTRING=SIP/111") in new stack
-- Executing [dsting@macro-dial-one:13] Return("IAX2/Valparaiso-2821", "") in new stack
-- Executing [s@macro-dial-one:27] GotoIf("IAX2/Valparaiso-2821", "0?modal") in new stack
-- Executing [s@macro-dial-one:28] GotoIf("IAX2/Valparaiso-2821", "0?skiptrace") in new stack
-- Executing [s@macro-dial-one:29] GosubIf("IAX2/Valparaiso-2821", "1?set,1?clear,1") in new stack
-- Executing [otset@macro-dial-one:1] Set("IAX2/Valparaiso-2821", "DB(CALLTRACE/111)=103") in new stack
-- Executing [otset@macro-dial-one:2] Return("IAX2/Valparaiso-2821", "") in new stack
-- Executing [s@macro-dial-one:30] Set("IAX2/Valparaiso-2821", "D_OPTIONS=tr") in new stack
-- Executing [s@macro-dial-one:31] ExecIf("IAX2/Valparaiso-2821", "0?SIPAddHeader(Alert-Info: )") in new stack
-- Executing [s@macro-dial-one:32] ExecIf("IAX2/Valparaiso-2821", "0?SIPAddHeader()") in new stack
-- Executing [s@macro-dial-one:33] ExecIf("IAX2/Valparaiso-2821", "0?Set(CHANNEL(musicclass)=)") in new stack
-- Executing [s@macro-dial-one:34] GosubIf("IAX2/Valparaiso-2821", "0?play1") in new stack
-- Executing [s@macro-dial-one:35] Set("IAX2/Valparaiso-2821", "CUSTOM=tr") in new stack
-- Executing [s@macro-dial-one:36] Set("IAX2/Valparaiso-2821", "KEEPID=TRUE") in new stack
-- Executing [s@macro-dial-one:37] Dial("IAX2/Valparaiso-2821", "SIP/111,*,tr") in new stack
== Using SIP RTP TOS bits 184
== Using SIP RTP COS mark 5
-- Called SIP/111
elk*CLI>

root@root:~#
-- Executing [s@macro-dialout-trunk:5] Set("SIP/103-000002d", "DIAL_TRUNK_OPTIONS=tr") in new stack
-- Executing [s@macro-dialout-trunk:6] Set("SIP/103-000002d", "OUTBOUND_GROUP=OUT_2") in new stack
-- Executing [s@macro-dialout-trunk:7] GotoIf("SIP/103-000002d", "1?nomax") in new stack
-- Goto (macro-dialout-trunk,s,9)
-- Executing [s@macro-dialout-trunk:9] GotoIf("SIP/103-000002d", "1?skipoutcid") in new stack
-- Goto (macro-dialout-trunk,s,12)
-- Executing [s@macro-dialout-trunk:12] GosubIf("SIP/103-000002d", "0?sub-ftp=2,s,1") in new stack
-- Executing [s@macro-dialout-trunk:13] Set("SIP/103-000002d", "OUTNUM=11") in new stack
-- Executing [s@macro-dialout-trunk:14] Set("SIP/103-000002d", "CUSTOM=IAX2/Valparaiso") in new stack
-- Executing [s@macro-dialout-trunk:15] ExecIf("SIP/103-000002d", "0?Set(DIAL_TRUNK_OPTIONS=M(setmusic=default)tr)") in new stack
-- Executing [s@macro-dialout-trunk:16] Macro("SIP/103-000002d", "dialout-trunk-pre-dial-hook") in new stack
-- Executing [s@macro-dialout-trunk-pre-dial-hook:1] MacroExit("SIP/103-000002d", "") in new stack
-- Executing [s@macro-dialout-trunk:17] GotoIf("SIP/103-000002d", "0?ypass,1") in new stack
-- Executing [s@macro-dialout-trunk:18] GotoIf("SIP/103-000002d", "0?customtrunk") in new stack
-- Executing [s@macro-dialout-trunk:19] Dial("SIP/103-000002d", "IAX2/Valparaiso/111,300,tr") in new stack
-- Called IAX2/Valparaiso/111
-- Call accepted by 192.168.1.251 (format gsm)
-- Format for call is gsm
-- IAX2/Valparaiso-20177 is ringing
  
```

Figura 87. Código generado troncal Vinadelmar y extensión 101.

La llamada finaliza entre extensión 113 de la telefonía móvil y extensión 101 de telefonía fija (Véase la figura 88).



```

root@eic-
-- Executing [dsting@macro-dial-one:11] GotoIf("IAX2/Valparaiso-2638", "0?begin") in new stack
-- Executing [dsting@macro-dial-one:12] Set("IAX2/Valparaiso-2638", "DSTRING=SIP/111") in new stack
-- Executing [dsting@macro-dial-one:13] Return("IAX2/Valparaiso-2638", "") in new stack
-- Executing [s@macro-dial-one:27] GotoIf("IAX2/Valparaiso-2638", "0?nodial") in new stack
-- Executing [s@macro-dial-one:28] GotoIf("IAX2/Valparaiso-2638", "0?skiptrace") in new stack
-- Executing [s@macro-dial-one:29] GosubIf("IAX2/Valparaiso-2638", "1?rset,1?otclear,1") in new stack
-- Executing [cuser@macro-dial-one:1] Set("IAX2/Valparaiso-2638", "DB(CALLTRACE/111)=103") in new stack
-- Executing [cuser@macro-dial-one:2] Return("IAX2/Valparaiso-2638", "") in new stack
-- Executing [s@macro-dial-one:30] Set("IAX2/Valparaiso-2638", "D_OPTIONS=r") in new stack
-- Executing [s@macro-dial-one:31] ExecIf("IAX2/Valparaiso-2638", "0?SIPAddHeader(Alert-Info: )") in new stack
-- Executing [s@macro-dial-one:32] ExecIf("IAX2/Valparaiso-2638", "0?SIPAddHeader( )") in new stack
-- Executing [s@macro-dial-one:33] ExecIf("IAX2/Valparaiso-2638", "0?Set(CHANNEL(misc:cas))") in new stack
-- Executing [s@macro-dial-one:34] GosubIf("IAX2/Valparaiso-2638", "0?wait,1") in new stack
-- Executing [s@macro-dial-one:35] Set("IAX2/Valparaiso-2638", "CWIGNORE=") in new stack
-- Executing [s@macro-dial-one:36] Set("IAX2/Valparaiso-2638", "XFERCID=TRUE") in new stack
-- Executing [s@macro-dial-one:37] Dial("IAX2/Valparaiso-2638", "SIP/111,,tr") in new stack
= Using SIP RTP TOS bits 18
= Using SIP RTP CoS mark 5
-- Called SIP/111
-- SIP/111-000000id is ringing
-- SIP/111-000000id answered IAX2/Valparaiso-2638
el*CLI>

root@root-
-- Executing [s@macro-dialout-trunk:7] GotoIf("SIP/103-0000002e", "1?nmax") in new stack
-- Goto (macro-dialout-trunk,s,9)
-- Executing [s@macro-dialout-trunk:9] GotoIf("SIP/103-0000002e", "1?skipoutcid") in new stack
-- Goto (macro-dialout-trunk,s,12)
-- Executing [s@macro-dialout-trunk:12] GosubIf("SIP/103-0000002e", "0?sub-fip-2,s,1") in new stack
-- Executing [s@macro-dialout-trunk:13] Set("SIP/103-0000002e", "OUIID=111") in new stack
-- Executing [s@macro-dialout-trunk:14] Set("SIP/103-0000002e", "Custom=IAX2/Valparaiso") in new stack
-- Executing [s@macro-dialout-trunk:15] ExecIf("SIP/103-0000002e", "0?Set(DIAL_TRUNK_OPTIONS={setmusic:default|tr})") in new stack
-- Executing [s@macro-dialout-trunk:16] Macro("SIP/103-0000002e", "dialout-trunk-predial-hook,") in new stack
-- Executing [s@macro-dialout-trunk-predial-hook:1] MacroExit("SIP/103-0000002e", "") in new stack
-- Executing [s@macro-dialout-trunk:17] GotoIf("SIP/103-0000002e", "0?bypass,1") in new stack
-- Executing [s@macro-dialout-trunk:18] GotoIf("SIP/103-0000002e", "0?outstrunk") in new stack
-- Executing [s@macro-dialout-trunk:19] Dial("SIP/103-0000002e", "IAX2/Valparaiso/111,300,tr") in new stack
-- Call accepted by 192.168.1.251 (format gsm)
-- Format for call is gsm
-- IAX2/Valparaiso-22866 is ringing
-- IAX2/Valparaiso-22866 is ringing
-- IAX2/Valparaiso-22866 answered SIP/103-0000002e
  
```

Figura 88.Llamada terminada entre dispositivo móvil y teléfono grandstream gxp 1400.

5. Pruebas de reportes de nuestro *melastix*

El *melastix* nos ayuda a tener nuestros reportes, ayudándonos a poseer una visualización de llamadas salientes, entrantes, gráficas por cada extensión, duración de llamadas, llamadas contestadas, llamadas no contestadas (Véase figura 89, 90, 91,92 y 93).

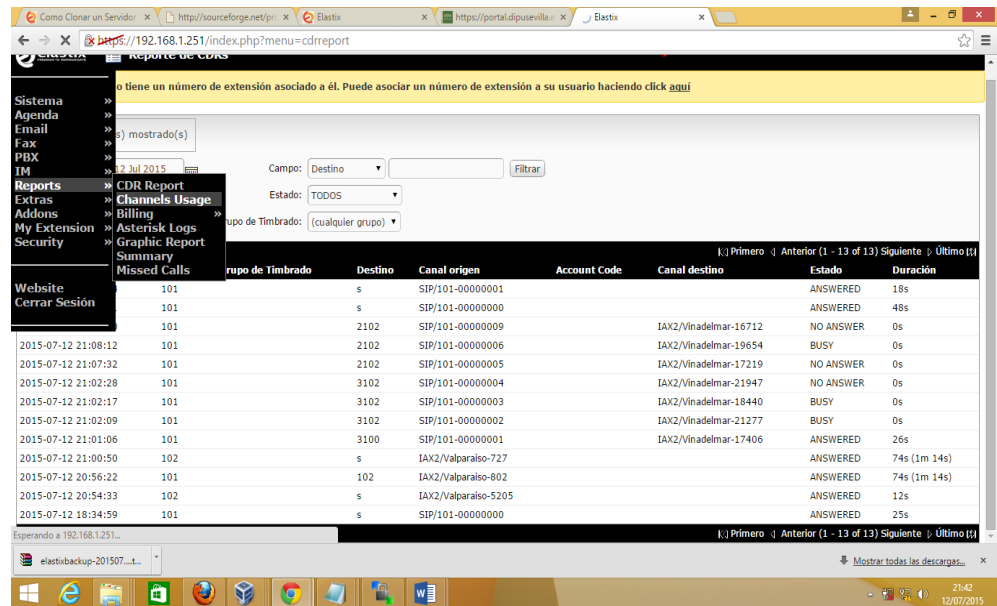


Figura 89. Llamadas realizadas extensión 101.

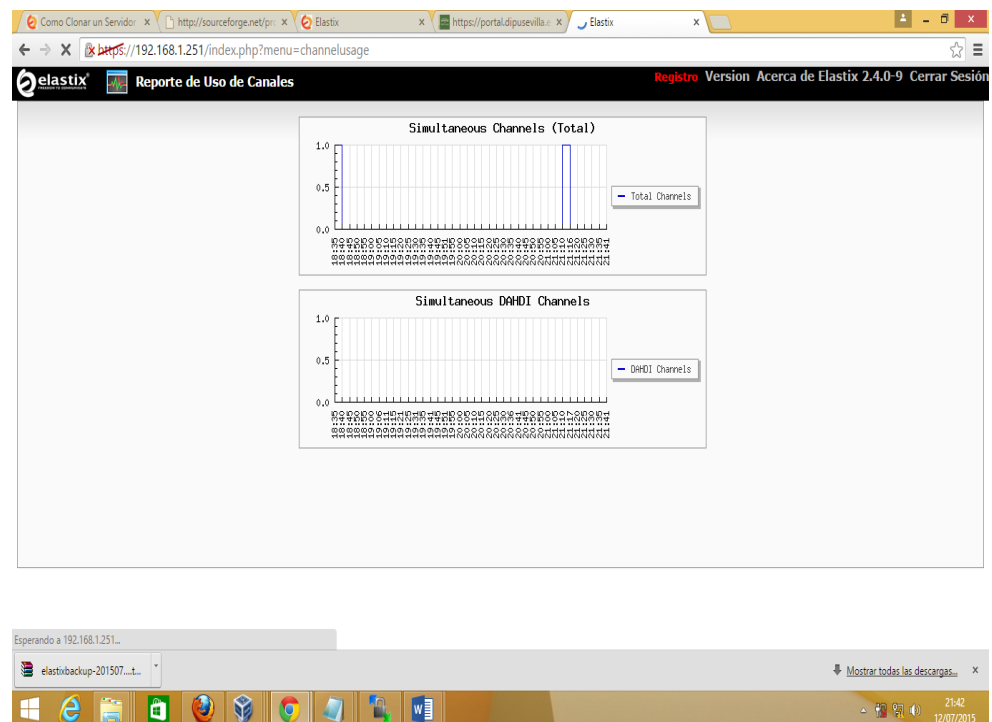


Figura 90. Gráfica de llamadas extensión 101.

Extensión	nombre usuario	# llamadas entrantes	# llamadas salientes	tiempo total (Llamadas Entrantes)	tiempo total (Llamadas Salientes)	Detalles
100	MENU PRINCIPAL	0	0	00h.00m.00s	00h.00m.00s	Ver
101	ANA TROYA	0	11	00h.00m.00s	00h.03m.11s	Ver
102	WALDEMAR ESPINOZA	1	2	00h.01m.14s	00h.01m.26s	Ver
103	MARIA GAHONA	0	0	00h.00m.00s	00h.00m.00s	Ver
104	LUIS JACOME	0	0	00h.00m.00s	00h.00m.00s	Ver
105	CARLOS JARAMILLO	0	0	00h.00m.00s	00h.00m.00s	Ver
106	MARCO OCAMPO	0	0	00h.00m.00s	00h.00m.00s	Ver
107	ALEX PADILLA	0	0	00h.00m.00s	00h.00m.00s	Ver
108	MARIO PALMA	0	0	00h.00m.00s	00h.00m.00s	Ver
109	LUIS FERNANDO PAZ	0	0	00h.00m.00s	00h.00m.00s	Ver
110	FRANCO SALCEDO	0	0	00h.00m.00s	00h.00m.00s	Ver
111	IVAN SIVENCIA	0	0	00h.00m.00s	00h.00m.00s	Ver
112	RODRIGO TENE	0	0	00h.00m.00s	00h.00m.00s	Ver
113	JORGE TOCTO	0	0	00h.00m.00s	00h.00m.00s	Ver
201	RAMIRO AGILA	0	0	00h.00m.00s	00h.00m.00s	Ver
202	CARLOS BARRETO	0	0	00h.00m.00s	00h.00m.00s	Ver
203	JOSE BENAVIDES	0	0	00h.00m.00s	00h.00m.00s	Ver
204	AGUSTO BUSTAMANTE	0	0	00h.00m.00s	00h.00m.00s	Ver
205	JORGE CARRION	0	0	00h.00m.00s	00h.00m.00s	Ver
206	JOSE CUENCA	0	0	00h.00m.00s	00h.00m.00s	Ver
207	JULIO CUENCA	0	0	00h.00m.00s	00h.00m.00s	Ver
208	MILTON LEON	0	0	00h.00m.00s	00h.00m.00s	Ver
209	THUESMAN ESTUARDO MONTANO	0	0	00h.00m.00s	00h.00m.00s	Ver
210	MILTON MORALES	0	0	00h.00m.00s	00h.00m.00s	Ver
211	JORGE MUNOZ	0	0	00h.00m.00s	00h.00m.00s	Ver

Figura 91. Tiempo de duración de cada llamada realizada de la extensión 101.

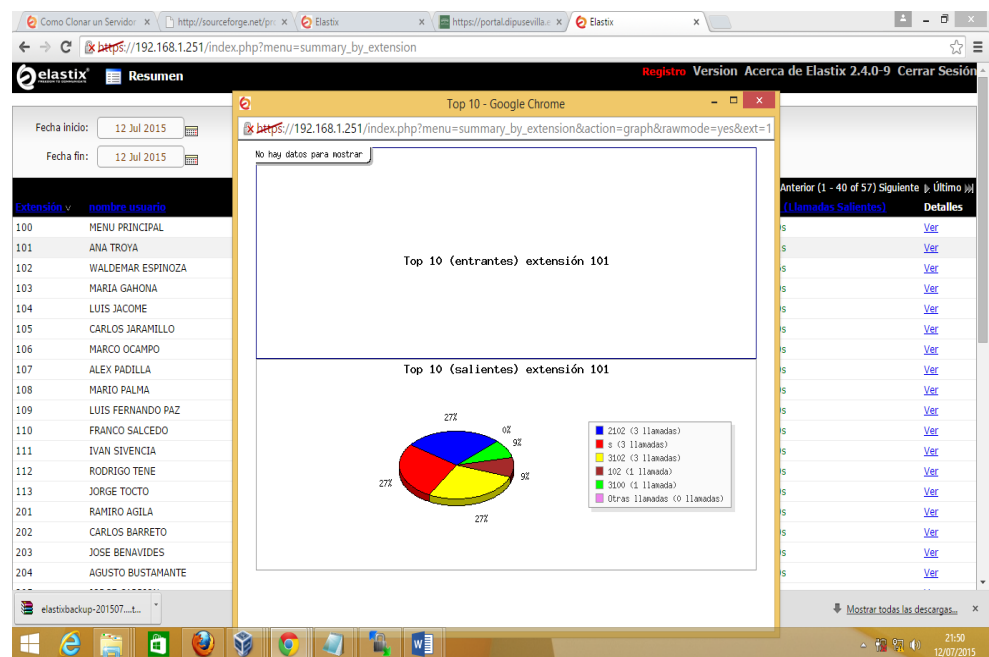


Figura 92. Gráfica pastel de cada llamada a la extensión 101.



Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta melastix en un Raspberry Pi para la Universidad Nacional de Loja en el AEIRNNR

Reporte de llamada Perdidas

Fecha Inicio: 12 Jul 2015 00:00
Fecha Fin: 12 Jul 2015 23:59

Fecha	Origen	Destino	Tiempo desde la última llamada	Número de intentos	Estado
12-Jul-2015 21:10:20	101	2102	40 minuto(s) 46 segundo(s)	2	NO CONTESTADA
12-Jul-2015 21:02:28	101	3102	48 minuto(s) 38 segundo(s)	1	NO CONTESTADA

Figura 93. Registro de llamadas no contestadas cuando ha llamado la extensión 101.



f. Discusión

Concluido el desarrollo del presente proyecto es importante realizar una evaluación de los objetivos planteados y determinar si logramos cumplir cada uno de ellos, lo cual se detalla a continuación:

- **Objetivo Específico 1:** Analizar la situación actual de la red telefónica Voip del AEIRNNR para encontrar las diferentes falencias y necesidades.

Para el cumplimiento de este objetivo se utilizó las técnicas de la observación y entrevista (ver anexo 1), lo que nos permitió conocer información necesaria para identificar los diferentes problemas (ver sección Resultados Fase1 de la situación actual y Figura 11), en los cuales se detalla las necesidades que se le complementó al sistema para un mejor rendimiento.

Objetivo Específico 2: Diseñar y configurar µelastix en el raspberry pi para telefonía móvil según las necesidades del AEIRNNR.

Se diseñó un esquema previo que permitió visualizar cómo quedaría el sistema de telefonía (Ver Figura. 65), seguidamente se armó el raspberry pi y se configuró µelastix en el raspberry pi de manera que cumpla con los requerimientos necesitados y de acuerdo al análisis planteado (Ver sección Resultados en Fase 2, apartado 1.2, 1.3 y 1.4).

Objetivo Específico 3: Implantar µelastix para telefonía móvil en la red telefónica actual en el AERNNR para tener una mejor comunicación entre todo el personal de la institución.

En este objetivo se procedió armar y ubicar los equipos donde funcionara sistema de telefonía (Ver sección Resultados, en Fase 3 apartados 3.1 literal 1, Figuras. 66), y posterior se realizó la configuración de enlace entre el servidor y telefonía ip fija y la red de telefonía móvil (Ver sección Resultados Fase3, apartado 3.1, literal 2, Figuras. 67 al 79), estos puntos ayudaron a cumplir con el objetivo y a su vez a la comunicación entre docentes y administrativos.



Valoración técnica económica ambiental

El desarrollo del presente trabajo requirió de la utilización de diversos recursos: Recursos humanos en los cuales participan el director de tesis, guía en el desarrollo del proyecto fin de carrera y los tesisistas, quienes desarrollaron el proyecto.

Recursos económicos que involucran los costos de servicios como internet y transportación.

Recursos materiales en el cual forman parte todo el material de oficina como papel y cartuchos de tinta utilizados para la impresión de los avances del proyecto, informe final y manuales del sistema.

Recursos técnicos que son las herramientas hardware utilizadas en el desarrollo del proyecto como el raspberry pi y sus periféricos, computador e impresora.

Recursos tecnológicos que son las herramientas software empleadas en el desarrollo del proyecto fin de carrera.

TALENTO HUMANO

Descripción	Cantidad	Horas	Valor/unit	Subtotal
Desarrollador del proyecto	2	500	\$3.00	\$3000.00
Asesorías del Proyecto	3	-	\$150.00	\$450.00
Total				\$3450.00

Tabla N°1 Recursos Humanos

BIENES (HARDWARE y SOFTWARE)

Descripción	Cantidad	Valor/unit	Subtotal
Hardware			
DELL Core I7 Led 4t Gen.+8gb+1tb+video Nvidia 2gb	1	\$ 1390.00	\$ 1390.00
Canakit Raspberry Pi (Raspberry Pi 512 MB + Clear Case + Micro USB Power	1	\$ 75	\$ 75



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
μelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*

Supply + Original Preloaded SD Card + HDMI Cable)			
Disco Duro Samsung Externo 1 TERA byte	1	\$ 120	\$ 120
Software			
Herramienta μelastix	-	-	\$ 00.00
Aplicaciones móviles	-	-	\$ 00.00
Total	-	-	\$1585.00

Tabla N°2 Bienes

MATERIALES DE OFICINA

Descripción	Cantidad	Valor/unit	Valor/Total
Resmas de hojas A4	5	\$ 5	\$ 25.00
Impresiones	1500	\$0.20	\$ 300.00
Suministros de oficina (esferos, perfiles, cds, etc.)	varios	Varios	\$ 45.00
Total			\$ 370.00

Tabla N°3 Materiales de oficina

SERVICIOS

Descripción	Cantidad	Unidad	Valor/unit	Subtotal
Internet	180	Horas	\$ 0.75	\$ 135.00
Transporte				\$ 50.00
Luz	700	Kw/h	\$ 0.11	\$ 77.00
Total				\$ 262.00

Tabla N°4 Servicios



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
μelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*

TOTAL

RECURSO	SUBTOTAL
TALENTO HUMANOS	\$0.00
BIENES	\$ 1585.00
MATERIALES DE OFICINA	\$ 370.00
SERVICIOS	\$ 262.00
IMPREVISTOS	\$ 240.00
TOTAL	\$ 2457.00

Tabla N°5 Total Recursos

Valoración técnica económica

El presente proyecto fin de carrera desde el punto de vista técnico, es factible puesto que es una necesidad que presentaba el AEIRNNR con el propósito de mejorar la comunicación de docentes con administrativos. Económicamente el proyecto también se considera factible gracias a las diversas herramientas de software que a más de ser de libre distribución, permiten programar de manera cómoda y sin restricciones, además es un hardware económico que permite realizar múltiples tareas.



g. Conclusiones

- El presente proyecto establece el crecimiento profesional ya que la RASBERRY PI, es usado en muchos países desarrollados por su bajo costo en comparación a una portátil u ordenador de escritorio. Esta placa posee múltiples aplicaciones que incentivan al estudiante a aprender ciencias computacionales mediante el uso de software libre.
- Debido a la diversidad de puertos con los que cuenta *RaspBerry Pi*, es posible implementar servicios adicionales acorde a otras necesidades que se requieran.
- La implantación de una PBX IP basada en un raspberry pi con software µelastix, incide en el uso básico del sistema de telefonía móvil voip de acuerdo a las necesidades del AEIRNNR.
- Entre las aplicaciones de telefonía móvil voip como Zoiper, implantada en los Smartphone, permite una funcionalidad adecuada para la comunicación entre los docentes y administrativos.
- La implementación del sistema µelastix, a través de su interfaz web ofrece una alternativa de configuración básica e inmediata ante los problemas que se presenten, como la interconexión entre dos o más telefonías basadas en µelastix.
- Con la implantación del sistema µelastix mediante el raspberry pi resulta gratuita la comunicación por medio de sus dispositivos móviles.



h. Recomendaciones

- De acuerdo a los resultados experimentales obtenidos en el capítulo referente a las pruebas, se recomienda el uso de PBX IP basados en RaspBerry Pi & μelastix para situaciones en las cuales no se requiera un tráfico superior a 6 o 7 llamadas simultaneas en una red local.
- Para la instalación de μelastix dentro de un raspberry pi se necesita un micro sd mínima de 4 GB con dos particiones, una de tipo FAT 16 y otra partición de tipo EXT3.
- Para la utilización de raspberry pi como hardware de μelastix se debe considerar que solo se puede ejecutar el proyecto en las pequeñas empresas, ya que cumple con limitaciones básicas dicho hardware.
- Se debe considerar que para la instalación de μelastix en su versión elastix-arm-2014-01-30.tar u otro, solo puede ser instalada en el raspberry pi tipo B, debido a que el nuevo hardware raspberry pi plus no permiten la ejecución del software.
- Se recomienda la utilización μelastix ya que es una herramienta potente que puede cumplir varias actividades, desde la creación de correos de voz, sala de conferencias virtuales, entre otros.
- Se recomienda la Universidad Nacional de Loja la implementación de esta aplicación ya que el costo del hardware es accesible para cualquier persona y el software μelastix es gratuito.



i. Referencias

- [1] «TELEFONIA VOZ IP,» [En línea]. Available: <http://www.telefoniavozip.com/voip/que-es-la-telefonía-ip.htm>.
- [2] «ECONSEPT,» [En línea]. Available: <http://econcept.es/econtel-características-centralita>.
- [3] «CISCO,» [En línea]. Available: Reducir los gastos de desplazamiento y formación, mediante el uso de videoconferencias y conferencias en línea..
- [4] «TECNOLOGIA DE VOIP,» GABRIEL ASTUDILLO, 14 MARZO 2011. [En línea]. Available: <http://es.slideshare.net/gastudillo/tecnologías-de-voip>.
- [5] «ELASTICTECH,» [En línea]. Available: <http://elastixtech.com/protocolo-iax/>.
- [6] «QUAREA,» [En línea]. Available: http://www.quarea.com/es/sip_session_initiation_protocol.
- [7] «TELEFONIA VoIP,» 12 ENERO 2011. [En línea]. Available: <http://telefoniasandovoip.blogspot.com/2011/01/encapsulamiento-de-una-trama-voip.html>.
- [8] «ASIMTELEC,» [En línea]. Available: <http://serviciosdetelecomunicaciones.com/pbx/>.
- [9] «SOINGLO,» [En línea]. Available: http://web.soinglo.com/index.php?option=com_content&view=category&id=39:voz-sobre-ip&layout=blog&Itemid=27&layout=default.
- [10] M. Y. M. BRYANT R, «ASTERISK THE DEFINITIVE GUIDE,» 2013. [En línea].
- [11] G. R., «INTEGRACION PBX,» 2008. [En línea].
- [12] «SERVICOM,» [En línea]. Available: <https://www.servicomecuador.com/telefonía.html>.
- [13] «MUSEPO INFORMATICO,» 18 DICIEMBRE 2013. [En línea]. Available: <http://histinf.blogs.upv.es/2013/12/18/raspberry-pi/>.
- [14] L. A. ARANGO, «BIBLIOTECA VIRTUAL,» [En línea]. Available: http://www.banrepcultural.org/blaavirtual/ayudadetareas/comunicacion/medios_telefonicos.
- [15] «admsaludv,» [En línea]. Available: <https://admsaludv.wordpress.com/59-2/>.
- [16] «ZOIPER,» [En línea]. Available: <http://www.zoiper.com/en>.



j. Anexos

Anexo 1.

Análisis de la entrevista antes de la Implantación de la telefonía móvil voz sobre ip.

1. ¿Qué ventajas ofrece la telefonía móvil voz sobre ip entre Docentes y Administrativos?

Los entrevistados señalaron que las ventajas más relevantes son comunicación ininterrumpida; enlace de difusión de información. Gratuidad del servicio; localización; diseño de agendas virtuales para el sector de secretaría.

Análisis: con estos datos nos damos cuenta que la implementación de la telefonía móvil voz sobre ip mantiene una comunicación directa entre el personal administrativo y docente para la fluidez de datos relativos del desempeño académico y cumplimiento de jornadas académicas y extracurriculares, eliminando los inconvenientes de localización de los docentes que retardan el cumplimiento de actividades pre planificadas.

2. ¿Actualmente la carrera de ingeniería en sistemas, para establecer la comunicación entre docentes y personal administrativo mantiene un sistema de comunicación interna móvil voz sobre ip?

Los entrevistados señalaron que en la actualidad la carrera no cuenta con un programa de comunicación interna, y esta solo se limitan en las llamadas telefónicas fijas hacia la sala de profesores las cuales no pueden ser contestadas en horario de clases o cuando nos trasladamos dentro de las instalaciones, siendo dificultosa la localización inmediata tanto de docentes como de personal administrativo.

Análisis: la innovación de la tecnología digital nos permite implementar un sistema de comunicación ininterrumpido de bajo costo para ser desarrollado como un mecanismo de localización efectivo de la planta docente y administrativa de Área de



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
μelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*

Energía las Industrias y los recursos no renovables de la Universidad Nacional de Loja.

3. ¿Apoyaría a la implementación de un sistema de telefonía móvil de voz sobre ip para el AEIRNNR?

Los entrevistados señalaron que este es un programa con iniciativa que ofrece muchas ventajas de comunicación inmediata donde solo se incluyen para acciones laborales y desempeño académico a docentes y administrativos, elevando el nivel comunicacional mediante la transmisión de datos relevantes y oportunos para mejorar la atención y el cumplimiento adecuado de nuestras responsabilidades académicas.

Análisis: podemos apreciar que el sistema de comunicación de telefonía móvil voz sobre ip, mantiene la aceptación de su implementación toda vez que este sistema se relaciona con la innovación tecnológica que permite una comunicación oportuna y versátil tanto para docente y administrativos e incluso puede ser ampliado a empresas.

Anexo 2.

RESULTADO DE APLICACIÓN DE ENCUESTAS.

Luego de haber realizado la recopilación de la información y datos ya implantada la telefonía móvil voz sobre ip, procedemos a presentar los resultados obtenidos a través de la recopilación de información de campo, que se desarrolló por medio de la siguiente encuesta que está compuesta por las siguientes interrogantes y que fueron planteadas a 13 docentes y 7 administrativos.

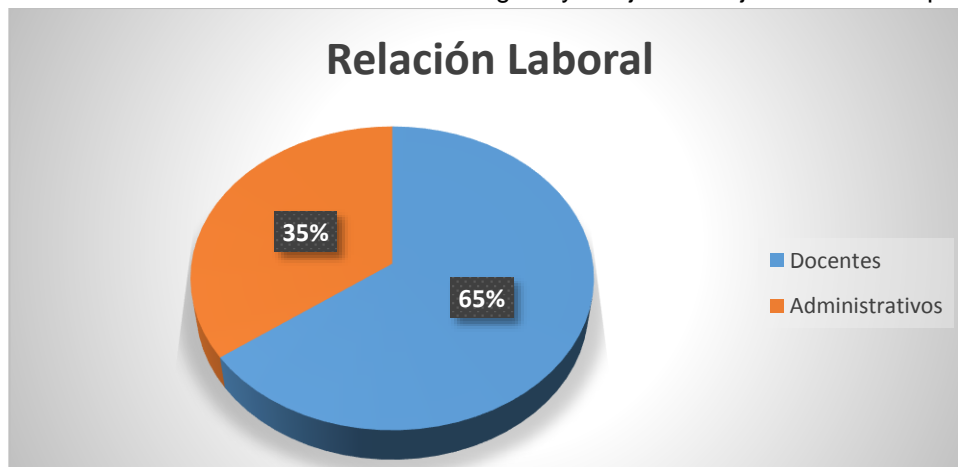
1. ¿Marque con una X el cargo que desempeña usted dentro del Área de la Energía, las Industrias y los Recursos Naturales No Renovables?

Tabla N°6 Relación Laboral

x	f	%
Docentes	13	65
Administrativos	7	35
TOTAL	20	100

Fuente: Encuesta aplicada a los Docentes y Administrativos de la Carrera de Ingeniería en Sistemas

Elaboración: Ximena Marisol Herrera Iñiguez y Benjamín Alejandro Pinza Tapia



Análisis e Interpretación:

De los encuestados 13 personas que representan el 65% son Docentes y 7 encuestados que suman el 35% son Administrativos.

La mayoría de la población encuestada son docentes de la Carrera de Ingeniería en Sistemas, quienes mantienen más relación en empleo, uso, manejo de sistemas informáticos.

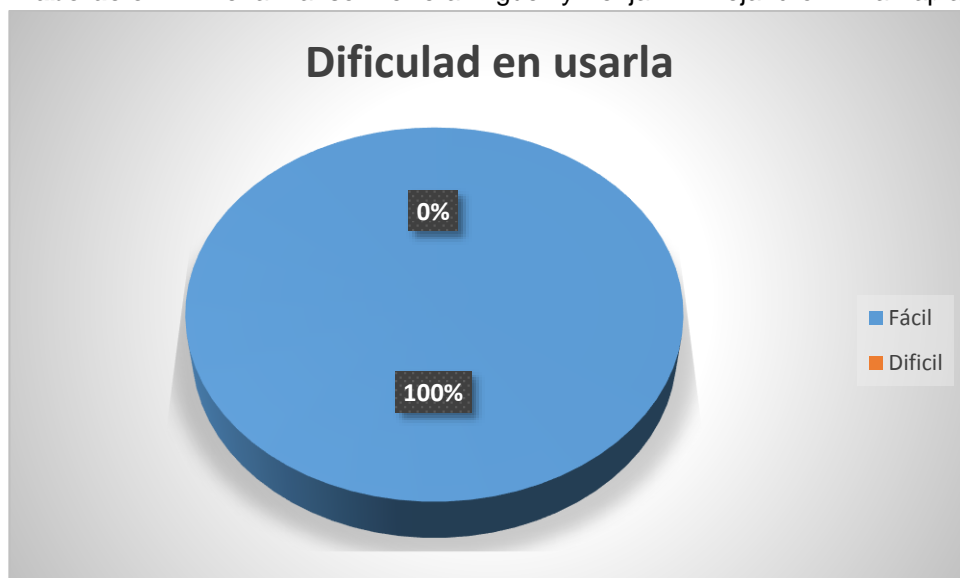
2. ¿El uso de la aplicación móvil Zoiper le parece?

Tabla N°7 Uso aplicación telefonía móvil

x	f	%
Fácil	20	100
Difícil	0	0
TOTAL	20	100

Fuente: Encuesta aplicada a los Docentes y Administrativos de la Carrera de Ingeniería en Sistemas

Elaboración: Ximena Marisol Herrera Iñiguez y Benjamín Alejandro Pinza Tapia



Análisis e interpretación:

En esta pregunta tenemos que los 20 encuestado que representan el 100% señalan que el uso es fácil.

El manejo y empleo de la aplicación móvil Zoiper es adecuada, versátil, y no presenta dificultades en su uso y empleo, lo cual se perfila como un medio adecuado de comunicación interna.

3. ¿La telefonía móvil ha resuelto la no comunicación entre Docente y Administrativo?

Tabla N°8 Comunicación Docente y Administrativo

x	f	%
Sí	20	100
No	0	0
TOTAL	20	100

Fuente: Encuesta aplicada a los Docentes y Administrativos de la Carrera de Ingeniería en Sistemas

Elaboración: Ximena Marisol Herrera Iñiguez y Benjamín Alejandro Pinza Tapia



Análisis e interpretación:

Los 20 encuestados que representan el 100% de la población señala la opción sí.

El grado de satisfacción en la solución de la comunicación interna entre Docentes y Administrativos, es segura confiable y responde a las necesidades de comunicación interna.

4. ¿Cómo califica La calidad de voz en la comunicación entre dispositivos móviles?

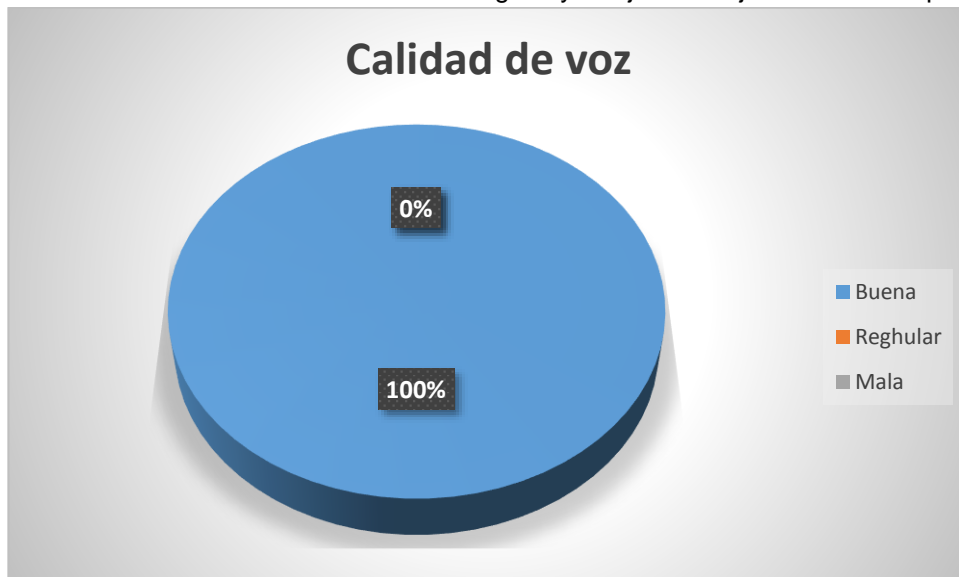
Tabla N°9 Calidad de voz

X	f	%
Buena	20	100
Regular	0	0
Mala	0	0
TOTAL	20	100

Fuente: Encuesta aplicada a los Docentes y

Administrativos de la Carrera de Ingeniería en Sistemas

Elaboración: Ximena Marisol Herrera Iñiguez y Benjamín Alejandro Pinza Tapia



Análisis e interpretación:

Los 20 encuestados que representan el 100% de la población señala la opción sí.

La calidad de la voz en este dispositivo es excelente toda vez que es entendible y manipulable al momento de regularlo en base a la capacidad auditiva de cada usuario.



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
μelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*



UNIVERSIDAD NACIONAL DE LOJA
**Área de la Energía, las Industrias y los Recursos
Naturales No Renovables**

CARRERA DE INGENIERÍA EN SISTEMAS

ENCUESTA

Encuesta aplicada con fin de comprobar la implantación de la telefonía móvil Voz sobre Ip dentro del Área de la Energía, las Industrias y los Recursos Naturales No Renovables entre los Docentes y Administrativos.

1. ¿Marque con una X el cargo que desempeña usted dentro del Área de la Energía, las Industrias y los Recursos Naturales No Renovables?

- a. ADMINISTRATIVO ()
b. DOCENTE ()

2. ¿El uso de la aplicación móvil Zoiper le parece?

- c. FÁCIL ()
d. DIFÍCIL ()

3. ¿La telefonía móvil, ha resuelto la no comunicación entre Docente y Administrativo?

- SI ()
NO ()

4. ¿Cómo califica la calidad de voz en la comunicación entre dispositivos móviles?

- BUENA ()
REGULAR ()
MALA ()



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
μelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*



UNIVERSIDAD NACIONAL DE LOJA
Área de la Energía, las Industrias y los Recursos
Naturales No Renovables
CARRERA DE INGENIERÍA EN SISTEMAS

ENCUESTA

Encuesta aplicada con fin de comprobar la implantación de la telefonía móvil Voz sobre Ip dentro del Área de la Energía, las Industrias y los Recursos Naturales No Renovables entre los Docentes y Administrativos.

1. ¿Marque con una X el cargo que desempeña usted dentro del Área de la Energía, las Industrias y los Recursos Naturales No Renovables?
 - a. ADMINISTRATIVO ()
 - b. DOCENTE (X)

2. ¿El uso de la aplicación móvil Zoiper le parece?
 - c. FÁCIL (X)
 - d. DIFÍCIL ()

3. ¿La telefonía móvil, ha resuelto la no comunicación entre Docente y Administrativo?
 - SI (X)
 - NO ()

4. ¿Cómo califica la calidad de voz en la comunicación entre dispositivos móviles?
 - BUENA (X)
 - REGULAR ()
 - MALA ()



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
μelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*



UNIVERSIDAD NACIONAL DE LOJA
Área de la Energía, las Industrias y los Recursos
Naturales No Renovables

CARRERA DE INGENIERÍA EN SISTEMAS

ENCUESTA

Encuesta aplicada con fin de comprobar la implantación de la telefonía móvil Voz sobre Ip dentro del Área de la Energía, las Industrias y los Recursos Naturales No Renovables entre los Docentes y Administrativos.

1. ¿Marque con una X el cargo que desempeña usted dentro del Área de la Energía, las Industrias y los Recursos Naturales No Renovables?
 - a. ADMINISTRATIVO ()
 - b. DOCENTE ()

2. ¿El uso de la aplicación móvil Zoiper le parece?
 - c. FÁCIL ()
 - d. DIFÍCIL ()

3. ¿La telefonía móvil, ha resuelto la no comunicación entre Docente y Administrativo?
 - SI ()
 - NO ()

4. ¿Cómo califica la calidad de voz en la comunicación entre dispositivos móviles?
 - BUENA ()
 - REGULAR ()
 - MALA ()



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
μelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*



UNIVERSIDAD NACIONAL DE LOJA
Área de la Energía, las Industrias y los Recursos
Naturales No Renovables

CARRERA DE INGENIERÍA EN SISTEMAS

Encuesta aplicada con fin de comprobar la implantación de la telefonía móvil Voz sobre Ip dentro del Área de la Energía, las Industrias y los Recursos Naturales No Renovables entre los Docentes y Administrativos.

1. ¿Marque con una X el cargo que desempeña usted dentro del Área de la Energía, las Industrias y los Recursos Naturales No Renovables?
 - a. ADMINISTRATIVO (X)
 - b. DOCENTE ()

2. ¿El uso de la aplicación móvil Zoiper le parece?
 - c. FÁCIL (X)
 - d. DIFÍCIL ()

3. ¿La telefonía móvil, ha resuelto la no comunicación entre Docente y Administrativo?
 - SI (X)
 - NO ()

4. ¿Cómo califica la calidad de voz en la comunicación entre dispositivos móviles?
 - BUENA (X)
 - REGULAR ()
 - MALA ()

Anexo 3.

Evidencias





Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta μelastix en un Raspberry Pi para la Universidad Nacional de Loja en el AEIRNNR

Anexo 4.

Licencia Creative Commons del Informe Final del Proyecto Fin de Carrera.



IMPLANTACIÓN DE TELEFONÍA MÓVIL VOZ SOBRE IP, UTILIZANDO LA HERRAMIENTA μELASTIX EN UN RASPBERRY PI, PARA LA UNIVERSIDAD NACIONAL DE LOJA EN EL AEIRNNR. is licensed under a [Creative Commons Reconocimiento-NoComercial 4.0 Internacional License](#).



*Implantación de Telefonía móvil Voz Sobre Ip utilizando la Herramienta
µelastix en un Raspberry Pi para la Universidad Nacional de Loja en el
AEIRNNR*

Anexo 5.

Abstract

The present research study was carried out with the purpose of implementing voice over IP with mobile phones by using µelastix tools in a raspberry pi B. This telephony was designed to be used by the AEIRNNR area of the Universidad Nacional de Loja; that is the reason why it was set up according to its needs and requirements.

For the data collection, and analysis and interpretation of results, scientific methods as well as techniques and instruments were used. The techniques used were the interview and the observation and the instruments were a questionnaire and an observation sheet. The questionnaire was applied to the academic and administrative staff that uses telephone network.

For the proper functioning of our IP telephony system, a Raspberry Pi B hardware and a µelastix software were used. They are free to use and were considered with the purpose of meeting the needs of the AEIRNNR area which does not have landline IP telephony for each professor. Furthermore, the equipment that works with the older IP telephony software was used.

The µelastix software has friendly interfaces to understand more about the basic settings which are necessary for a telephony system. It involves the creation of extensions, ivr's, etc. This application was designed in an administrator-friendly manner that can provide immediate solutions.

In the results section the applied process for the improvement of the telephony network is explained, also the µelastix and voip settings and the different tests done to the telephony network as well as the use of tools for the results of the quality service are described.

Based on the obtained results, it was concluded that the voice over IP used by the academic and administrative staff of the AEIRNNR of the UNL was successful and helped to the communication among them.

Eliana Ivanova Pinza Tapia
DOCENTE INVESTIGADORA UTPL