



REPÚBLICA DEL ECUADOR

Escuela Politécnica Nacional

" E S C I E N T I A H O M I N I S S A L U S "

La versión digital de esta tesis está protegida por la Ley de Derechos de Autor del Ecuador.

Los derechos de autor han sido entregados a la "ESCUELA POLITÉCNICA NACIONAL" bajo el libre consentimiento del (los) autor(es).

Al consultar esta tesis deberá acatar con las disposiciones de la Ley y las siguientes condiciones de uso:

- Cualquier uso que haga de estos documentos o imágenes deben ser sólo para efectos de investigación o estudio académico, y usted no puede ponerlos a disposición de otra persona.
- Usted deberá reconocer el derecho del autor a ser identificado y citado como el autor de esta tesis.
- No se podrá obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.

El Libre Acceso a la información, promueve el reconocimiento de la originalidad de las ideas de los demás, respetando las normas de presentación y de citación de autores con el fin de no incurrir en actos ilegítimos de copiar y hacer pasar como propias las creaciones de terceras personas.

Respeto hacia sí mismo y hacia los demás.

ESCUELA POLITECNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

**PROTOTIPO DE SISTEMA DE GENERACIÓN AUTOMÁTICA DE
LLAMADAS TELEFÓNICAS Y ENVÍO MASIVO DE MENSAJES DE TEXTO**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
ELECTRÓNICA Y REDES DE INFORMACIÓN**

EVELYN YESSSENIA ACHIG RUANO

DIRECTOR: ING. ANA RODRÍGUEZ MSc.

CO DIRECTOR: ING. JOSÉ ESTRADA MSc.

Quito, julio 2018

DECLARACIÓN

Yo Evelyn Yessenia Achig Ruano, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Evelyn Yessenia Achig Ruano

CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por Evelyn Yessenia Achig Ruano, bajo nuestra supervisión.

Ing. Ana Rodríguez M.Sc.
DIRECTOR DE PROYECTO

Ing. José Estrada M.Sc.
CODIRECTOR DE PROYECTO

AGRADECIMIENTO

Mi más sincero agradecimiento a la M.Sc. Ana Rodríguez y al M.Sc. José Estrada, quienes fueron una gran guía y apoyo para la posible realización de este proyecto.

A mis padres Luis Eduardo Achig y Laura Cecilia Ruano por ser mi gran ejemplo de vida, mi fuerza y mi soporte siempre.

A mis jefes y queridos amigos Luis Machado y Luis Aguirre quienes confiaron en mi trabajo y me dirigieron siempre con sus consejos y enseñanzas.

Y finalmente a mi familia, mis abuelitos, tíos, tías, primos, primas y hermanas por siempre apoyarme y a mi Hildita que siempre quiso verme como profesional y desde el cielo siempre me cuida.

DEDICATORIA

Este proyecto va dedicado a mis padres, hermanas y familiares, que son el pilar fundamental de mi vida por su ejemplo de unidad y fortaleza ante cualquier adversidad.

CONTENIDO

ÍNDICE DE TABLAS	VII
ÍNDICE DE FIGURAS	VIII
ÍNDICE DE COMANDOS	XII
ÍNDICE DE CÓDIGOS	XIII
CAPÍTULO 1	1
1 MARCO TEÓRICO	1
1.1 PRINCIPALES COMPONENTES TEÓRICOS RELACIONADOS CON LA TELEFONÍA IP	1
1.1.1 TELEFONÍA IP	1
1.1.2 VENTAJAS DE LA VOZ SOBRE IP	1
1.1.3 COMPONENTES DE UN SISTEMA DE TELEFONÍA IP	3
1.1.4 PROTOCOLOS DE SEÑALIZACIÓN	4
1.1.5 PROTOCOLOS DE TRANSPORTE EN TIEMPO REAL	12
1.1.6 CODIFICADORES/DECODIFICADORES (CODECS).....	13
1.1.7 ASTERISK	14
1.2 MENSAJES DE TEXTO SMS.....	22
1.2.1 CARACTERÍSTICAS	23
1.2.2 VENTAJAS DEL SMS	23
1.3 DESARROLLO DE SOFTWARE	24
1.3.1 ANÁLISIS DE REQUERIMIENTOS	24
1.3.2 DISEÑO DE SOFTWARE.....	27
1.3.3 METODOLOGÍAS DE DESARROLLO DE SOFTWARE	29
1.3.4 PATRÓN DE DISEÑO MODELO VISTA CONTROLADOR	31
CAPÍTULO 2	34
2 ANÁLISIS DE REQUERIMIENTOS Y DISEÑO DEL PROTOTIPO AUTOMATIC SEND&DIALING (ASD)	34
2.1 ANÁLISIS DE REQUERIMIENTOS Y NECESIDADES DEL PROTOTIPO	34
2.1.1 ENCUESTA DE REQUERIMIENTOS PARA LLAMADAS Y MENSAJES AUTOMÁTICOS	34
2.1.2 HISTORIAS DE USUARIO	37
2.1.3 ITERACIONES DEL PROYECTO	37

2.1.4	SEGUNDA ITERACIÓN	40
2.1.5	TERCERA ITERACIÓN.....	43
2.1.6	RESUMEN DE REQUERIMIENTOS DEL PROTOTIPO.....	44
2.2	DISEÑO DEL SISTEMA AUTOMATIC SEND&DIALING (ASD).....	45
2.2.1	NIVEL DE DATOS.....	45
2.2.2	MODELO ENTIDAD-RELACIÓN	45
2.2.3	DIAGRAMA DE CLASES	48
2.3	NIVEL DE LÓGICA DE NEGOCIO	49
2.3.1	DIAGRAMA DE DESPLIEGUE.....	49
2.3.2	DIAGRAMA DE ACTIVIDADES.....	50
2.4	NIVEL PRESENTACIÓN.....	56
2.5	INTERFACES DEL SISTEMA	56
CAPÍTULO 3.....		75
3	IMPLEMENTACIÓN Y PRUEBAS DEL SISTEMA.....	75
3.1	IMPLEMENTACIÓN	75
3.1.1	ESQUEMA DEL PROTOTIPO.....	75
3.1.2	IMPLEMENTACIÓN DE LA CENTRAL TELEFÓNICA	76
3.1.3	CONFIGURACIÓN DEL GATEWAY GSM	81
3.1.4	IMPLEMENTACIÓN DE NIVEL DE DATOS	84
3.1.5	IMPLEMENTACIÓN DE NIVEL PRESENTACIÓN	110
3.2	PRUEBAS DE SISTEMA.....	112
3.2.1	PRUEBAS DE FUNCIONALIDAD	112
3.2.2	PRUEBAS UNITARIAS DE LA APLICACIÓN.....	115
3.2.3	PRUEBAS DE ACEPTACIÓN	121
CAPÍTULO 4.....		123
4	CONCLUSIONES Y RECOMENDACIONES.....	123
4.1	CONCLUSIONES.....	123
4.2	RECOMENDACIONES	126
BIBLIOGRAFÍA		129
ANEXO A.....		A-1
ANEXO B.....		B-1
ANEXO C.....		C-1
ANEXO D.....		D-1

ÍNDICE DE TABLAS

Tabla 1.1	CODECS para la telefonía IP.....	14
Tabla 2.1	Encuesta de Requerimientos.....	35
Tabla 2.2	Historia de usuario HU1-01_Crear Campaña Estática.....	38
Tabla 2.3	Historia de usuario HU1-02_Crear Campaña Personalizada.....	38
Tabla 2.4	Historia de usuario HU1-03_Crear Campaña Mensajes de texto.....	39
Tabla 2.5	Historia de usuario HU1-04_Lista de Campañas General.....	39
Tabla 2.6	Historia de usuario HU1-05_Borrar Campañas.....	39
Tabla 2.7	Historia de usuario HU1-06_Reporte de Campañas Estáticas y Personalizadas.....	40
Tabla 2.8	Historia de usuario HU1-07_Reporte de Campañas de Mensajes de texto.....	40
Tabla 2.9	Historia de usuario HU2-01_Crear Campaña Estática número de intentos.....	41
Tabla 2.10	Historia de usuario HU2-02_Crear Campaña Personalizada número de intentos.....	41
Tabla 2.11	Historia de usuario HU2-03_Eventos Compactos en Campañas.....	41
Tabla 2.12	Historia de usuario HU2-04_Subir Archivos en Campaña Estática.....	42
Tabla 2.13	Historia de usuario HU2-05_Subir Archivos en Campaña Personalizada.....	42
Tabla 2.14	Historia de usuario HU2-06_Subir Archivos en Campaña Mensaje de texto.....	42
Tabla 2.15	Historia de usuario HU3-01_Reporte Gráfico de Campañas Estáticas.....	43
Tabla 2.16	Historia de usuario HU3-02_Reporte Gráfico de Campañas Personalizadas.....	43
Tabla 2.17	Historia de usuario HU3-03_Reporte Gráfico de Campañas de Mensajes de texto.....	44
Tabla 2.18	Resumen de los Requerimientos del Prototipo.....	44
Tabla 3.1	Módulos de compilación.....	76
Tabla 3.2	Eventos permitidos para cada tabla.....	86
Tabla 3.3	Descripción de funciones creadas en el prototipo.....	87
Tabla 3.4	Ejemplo de Campaña personalizada.....	112
Tabla 3.5	Registro final de la campaña.....	113
Tabla 3.6	Campaña de mensajes de texto.....	114
Tabla 3.7	Registro final de la campaña de mensajes de texto.....	115
Tabla 3.8	Requerimientos Funcionales para pruebas de aceptación.....	121
Tabla 3.9	Detalle de las Pruebas de Aceptación.....	122

ÍNDICE DE FIGURAS

Figura 1.1	Ejemplo de la estructura del mensaje SIP de solicitud.....	7
Figura 1.2	Ejemplo de la estructura del mensaje SIP de respuesta.....	8
Figura 1.3	Mensajes de respuestas SIP.....	9
Figura 1.4	Transacciones en una llamada SIP.....	11
Figura 1.5	Sintaxis de una extensión.....	18
Figura 1.6	Ejemplo de uso de prioridades y comando same.....	19
Figura 1.7	Configuración archivo sip.conf.....	20
Figura 1.8	Caso de Uso.....	26
Figura 1.9	Diagrama de Casos de Uso.....	26
Figura 1.10	Vista-Modelo-Controlador.....	32
Figura 2.1	Modelo Entidad-Relación.....	46
Figura 2.2	Modelo Diagrama Relacional.....	47
Figura 2.3	Diagrama de Despliegue.....	49
Figura 2.4	Diagrama de clases.....	48
Figura 2.5	Diagrama de Actividades_Ingreso.....	50
Figura 2.6	Diagrama de Actividades_Menú principal.....	50
Figura 2.7	Diagrama de Actividades_Creacion de una campaña	51
Figura 2.8	Diagrama de Actividades_Crear campaña Mensajes de texto	52
Figura 2.9	Diagrama de Actividades_ Llamadas Automáticas	53
Figura 2.10	Diagrama de Actividades_Campaña de Mensajes de Texto.....	54
Figura 2.11	Diagrama de Actividades_Estadísticas_Reportes de campaña Estática.....	55
Figura 2.12	Diagrama de Actividades_Estadísticas_Reportes Campaña Personalizada.....	55
Figura 2.13	Diagrama de Actividades_Estadísticas_Reportes Campaña Mensajes de texto.....	55
Figura 2.14	Intefaz_Ingreso.....	56
Figura 2.15	Interfaz_Menú_Principal.....	57
Figura 2.16	Interfaz_Llamadas Automáticas.....	57
Figura 2.17	Interfaz_Lista Campañas LLamadas Estáticas.....	58
Figura 2.18	Interfaz_Lista Campañas Llamadas Personalizadas.....	58
Figura 2.19	Interfaz_Creación de Campañas de Llamadas Automáticas.....	59
Figura 2.20	Interfaz_Creación de Campañas de Llamadas Automáticas Tipo.....	59
Figura 2.21	Interfaz_ Elementos para subir archivos a las campañas de audio estático.....	60
Figura 2.22	Interfaz_ Elementos para subir archivos a las campañas de audio personalizado.....	60
Figura 2.23	Interfaz_ Seleccionar un archivo de audio	

	para campañas de audio estático.	61
Figura 2.24	Interfaz_Seleccionar un archivo de texto para campañas de audio estático.	61
Figura 2.25	Interfaz_Seleccionar un archivo de texto para campañas de audio personalizado.	62
Figura 2.26	Interfaz_Detalle de la campaña de llamadas de audio estático.	62
Figura 2.27	Interfaz_Detalle de la campaña de llamadas de audio personalizado.	62
Figura 2.28	Interfaz_Eventos de las Campañas estáticas.	63
Figura 2.29	Interfaz_Eventos de las Campañas Personalizadas.	63
Figura 2.30	Interfaz_Mensajes de información de las campañas de llamadas automáticas_Iniciar campaña.	64
Figura 2.31	Interfaz_Mensajes de información de las campañas de llamadas automáticas_Error Iniciar campaña.	64
Figura 2.32	Interfaz_Mensajes de información de las campañas de llamadas automáticas_Detener campaña.	64
Figura 2.33	Interfaz_Mensajes de información de las campañas de llamadas automáticas_Eliminar campaña.	64
Figura 2.34	Interfaz_Reporte gráfico de las campañas estáticas diagrama de barras.	65
Figura 2.35	Interfaz_Reporte gráfico de las campañas estáticas diagrama de pastel.	65
Figura 2.36	Interfaz_Reporte gráfico de las campañas estáticas reporte estadístico.	66
Figura 2.37	Interfaz_Reporte de las campañas estáticas registro estadístico.	66
Figura 2.38	Interfaz_Reporte gráfico de las campañas personalizadas diagrama de barras.	67
Figura 2.39	Interfaz_Reporte gráfico de las campañas personalizadas diagrama pastel.	67
Figura 2.40	Interfaz_Reporte gráfico de las campañas personalizadas reporte estadístico.	68
Figura 2.41	Interfaz_Reporte gráfico de las campañas personalizadas registro estadístico.	68
Figura 2.42	Interfaz_Mensajes de texto.	68
Figura 2.43	Interfaz_Lista Campañas de Mensajes de texto.	69
Figura 2.44	Interfaz_Creación de Campañas de Mensajes de texto.	70
Figura 2.45	Interfaz_Lista de campañas de mensajes de texto.	70
Figura 2.46	Interfaz_Seleccionar un archivo de texto para campañas de mensajes de texto.	71
Figura 2.47	Interfaz_Detalle de la campaña de mensajes de texto.	71
Figura 2.48	Interfaz_Eventos de las campañas de mensajes de texto.	72

Figura 2.49	Interfaz_Mensajes de información de las campañas de mensajes de texto_Iniciar campaña.....	72
Figura 2.50	Interfaz_Mensajes de información de las campañas mensajes de texto_Error_Iniciar campaña.....	72
Figura 2.51	Interfaz_Mensajes de información de las campañas mensajes de texto_Detener campaña.	72
Figura 2.52	Interfaz_Mensajes de información de las campañas de mensajes de texto_Eliminar campaña.	73
Figura 2.53	Interfaz_Reporte gráfico de las campañas de mensajes de texto diagrama de barras.....	73
Figura 2.54	Interfaz_Reporte gráfico de las campañas de mensajes de texto diagrama de pastel.	73
Figura 2.55	Interfaz_Reporte gráfico de las campañas de mensajes de texto detalle.	74
Figura 3.1	Diagrama del Prototipo.....	75
Figura 3.2	Configuración archivo sip.conf.....	77
Figura 3.3	Archivo extensions.conf contexto callmatic.....	78
Figura 3.4	Archivo extensions.conf contexto msg.....	78
Figura 3.5	Ejemplo del cambio a voz en español del archivo voices.scm.	79
Figura 3.6	Ejemplo de URL.	80
Figura 3.7	Ejemplo de una parte del <i>script</i> para envío de sms en perl.....	80
Figura 3.8	Pantalla de inicio del <i>gateway</i> GSM OpenVox.....	81
Figura 3.9	Configuración de red <i>Gateway</i> GSM.	82
Figura 3.10	Configuración de la Troncal SIP.	82
Figura 3.11	Creación de Grupos en el <i>gateway</i> OpenVox GSM.....	83
Figura 3.12	Creación de rutas en el <i>gateway</i> OpenVox GSM.....	83
Figura 3.13	Configuración de parámetros para el envío de SMS.....	84
Figura 3.14	Archivo pg_hba.conf.....	85
Figura 3.15	Ejemplo de creación de tablas en la base de datos.....	85
Figura 3.16	Ejemplo de una función creada en la base de datos.	87
Figura 3.17	Configuración del archivo httpd.conf.....	91
Figura 3.18	Interfaz Creación Campaña de Llamadas Personalizadas.....	110
Figura 3.19	Captura de mensajes de texto en el celular.....	114
Figura 3.20	Pruebas_unitarias Ejecución de funciones crear campañas.	116
Figura 3.21	Pruebas_unitarias Ejecución de funciones para reportes tiempo real progreso de campañas.	116
Figura 3.22	Pruebas_unitarias Ejecución de funciones para contactos en campañas.....	117
Figura 3.23	Pruebas_unitarias Ejecución de funciones audios en campañas.	117
Figura 3.24	Pruebas_unitarias Ejecución de funciones registro de campañas llamadas automáticas.....	118
Figura 3.25	Pruebas_unitarias Ejecución de funciones	

	reportes y estadísticas campaña sms.....	118
Figura 3.26	Pruebas_unitarias Ejecución de funciones reportes estadísticos.	119
Figura 3.27	Pruebas_unitarias Ejecución de funciones inserción de contactos y audios.....	119
Figura 3.28	Pruebas_unitarias Ejecución de funciones eventos de campañas llamadas automáticas.	120
Figura 3.29	Pruebas_unitarias Ejecución de funciones eventos de campaña de mensajes de texto.....	120

ÍNDICE DE COMANDOS

Comando 3.1	Instalación de voces en español para Festival.	79
Comando 3.2	Comandos para instalación de paquetes Postgresql.....	84
Comando 3.3	Comandos para reiniciar y acceder el servicio postgresql.....	85
Comando 3.4	Comandos para instalación de paquetes Apache.	90
Comando 3.5	Ejemplo de ejecución de un archivo en la base de datos.	115

ÍNDICE DE CÓDIGOS

Código 3.1	Archivo para conexión con la base de datos (DataBase.php).	92
Código 3.2	Archivo para formulario de entrada (Index.php).	92
Código 3.3	Archivo para ingreso al sistema (Login.php).....	93
Código 3.4	Menú de funcionalidades del sistema (Menu.php).	94
Código 3.5	Lista de las llamadas estáticas (Llamadas_pregrabadas.php).....	95
Código 3.6	Lista de las llamadas personalizadas (Llamadas_construidas.php).	97
Código 3.7	Lista de mensajes de texto (Mensajes texto.php).....	98
Código 3.8	Formulario para la creación de campañas(Vistacreacióncampaignllamadas.php).	99
Código 3.9	Creación campañas automáticas (Crearcampaignllamadas.php).....	99
Código 3.10	Formulario para la creación de campañas de mensajes de texto (Vistacreacióncampaignsms.php).	100
Código 3.11	Creación campañas de mensajes de texto (CreacióncampaignSms.php).	101
Código 3.12	Formulario para cargar archivos de audio (VistaUploadaudiollamadas.php).	101
Código 3.13	Formulario para cargar archivos de texto (VistaUploadcsvContactos.php).....	102
Código 3.14	Cargar archivo de audio (UploadAudioPregrabadas.php)	102
Código 3.15	Cargar archivo de texto (UploadCsvcontactos.php)	103
Código 3.16	Inicio de campañas de llamadas automáticas (Iniciocampaignllamadas.php).	104
Código 3.17	Inicio de campañas de mensajes de texto (IniciarcampaignSms.php).	105
Código 3.18	Detener las campañas de llamadas automáticas (Stopcampaignllamadas.php).	105
Código 3.19	Detener las campañas de mensajes de texto (StopcampaignSms.php).	106
Código 3.20	Borrar las campañas de mensajes de texto (DeletCampaignSms.php).	106
Código 3.21	Borrar las campañas de llamadas automáticas (DeletCampaign.php).	108
Código 3.22	Formulario de estadísticas de campañas de llamadas estáticas	108
Código 3.23	Formulario de estadísticas de campañas de llamadas	

	personalizadas (Progreso construidas.php).....	109
Código 3.24	Formulario de estadísticas de campañas de mensajes de texto	109
Código 3.25	Formulario para la creación de campañas de llamadas automáticas (VistacrearcampaignLlamadas.php)	111

RESUMEN

El presente proyecto de titulación está conformado por cuatro capítulos. En el primer capítulo se realiza una descripción de los conceptos básicos relacionados con la Telefonía IP. Dicha descripción contiene los componentes típicos de Telefonía IP, servicios y funcionalidades que se permiten desarrollar y crear nuevas herramientas, utilizando softwares muy conocidos *Open Source* para la implementación.

También se describe los conceptos para mecanismos de generación automática de llamadas, conversores de texto a voz y mensajes de texto corto denominado SMS por sus siglas en inglés.

Finalmente se describe algunos conceptos sobre el diseño de Software, detallando conceptos del modelo incremental, la metodología de desarrollo XP (*eXtreme Programming*) y el patrón de diseño Modelo Vista Controlados (MVC), utilizados como referencia para el desarrollo de la aplicación web para la administración del prototipo implementado.

El segundo capítulo está compuesto por dos partes, la primera parte se basa en comprender el análisis de requerimientos de los servicios para la implementación de la telefonía IP, así como los requisitos para la aplicación web que se necesitaron para el desarrollo del prototipo y finalmente la segunda parte del capítulo describe el diseño del prototipo.

EL tercer capítulo describe el proceso de implementación de los distintos componentes del prototipo, entre los cuales se tienen: configuración de la central telefónica y hardware telefónico, instalación de la base de datos y la implementación de la aplicación web, También se presentan las pruebas realizadas en el prototipo entre las cuales se tienen: pruebas unitarias, pruebas de aceptación y pruebas de funcionalidad del prototipo.

El cuarto capítulo corresponde a las conclusiones y recomendaciones que se han obtenido al finalizar el proyecto, basadas en los conocimientos y experiencias adquiridos a lo largo del diseño e implementación del prototipo.

PRESENTACIÓN

El contacto con las personas ha sido siempre una necesidad estratégica de las organizaciones, con distintos fines comunicativos como: recordatorios de pago, difusión de alertas, o incluso mercadeo. Así, la red telefónica ha sido históricamente el medio tecnológico preferido para realizar ese contacto con el cliente o el usuario.

En el Ecuador, tal como en muchos otros países, la forma más rápida y directa de comunicarse con alguien es a través de la telefonía, especialmente móvil, mediante una llamada o un mensaje de texto. La razón es conocida: 86,4% de hogares ecuatorianos tiene teléfono celular activado y existen más líneas activas que habitantes [1], [2].

Debido al gran alcance que brindan las redes telefónicas para el contacto con el cliente o el usuario, muchas organizaciones utilizan este medio para actividades de mercadeo, encuestas de satisfacción, difusión de alertas, etc [3] [4]. Debido al alto costo (en términos de tiempo y dinero) de realizar este proceso manualmente, han aparecido mecanismos automáticos propietarios cuyo costo no está al alcance de pequeñas y medianas organizaciones.

El prototipo para la generación automática de llamadas y mensajes de texto, es administrable a través de una aplicación web. Este proceso automático de comunicación comprende tanto el establecimiento de varias llamadas telefónicas con varios destinatarios para la reproducción de un mensaje de voz como el envío de mensajes de texto (SMS). La administración del prototipo contempla la configuración de varios parámetros relacionados con la generación automática de comunicación.

CAPÍTULO 1

1 MARCO TEÓRICO

En este capítulo se describe los principales componentes teóricos relacionados con la telefonía IP, VoIP (*Voice over IP*), Asterisk, mensajería de texto (SMS), desarrollo de aplicaciones web y metodología de desarrollo XP (*Extreme Programming*).

1.1 PRINCIPALES COMPONENTES TEÓRICOS RELACIONADOS CON LA TELEFONÍA IP

1.1.1 TELEFONÍA IP [1] – [4]

La telefonía IP se define como el servicio telefónico de la tecnología de VoIP (*Voice over IP*), la cual permite la transmisión datos y voz convertida en paquetes por el Internet o redes IP privadas mediante el protocolo IP. Frecuentemente se diferencia a la VoIP de la telefonía IP, porque la voz sobre IP consiste en transmitir únicamente paquetes de voz a través de una red IP, en cambio la telefonía IP asocia una serie de protocolos, servicios y componentes de telefonía.

1.1.2 VENTAJAS DE LA VOZ SOBRE IP [5], [6]

- **Reducción de costos de infraestructura:** La VoIP simplifica la infraestructura de telecomunicaciones en una empresa, por que utiliza la misma red cableada o inalámbrica, permitiendo el ahorro en la implementación de una nueva infraestructura única para el sistema de telefonía IP.
- **Toma de decisiones futuras:** La implementación de sistemas de telefonía IP permite tener estadísticas y reportes de las llamadas recibidas y realizadas por los usuarios del sistema, esto conlleva a la toma de decisiones futuras para el ahorro de costos y mejor elección de

proveedores de la PSTN (*Public Switched Telephone Network*) [6] dentro de la empresa.

- **Optimización de recursos:** En la telefonía IP es posible utilizar tanto recursos de hardware como de software, optimizando los recursos de una empresa. Se pueden optimizar las líneas de comunicación así como también integrar la telefonía de varias sedes de una misma organización en un sistema unificado permitiendo tener llamadas internas totalmente gratuitas.
- **Softphones:** Actualmente las aplicaciones de software, *softphones*, tanto para PC como para dispositivos móviles son muy utilizadas debido a que éstas tienen las mismas funcionalidades de los teléfonos IP y permiten la movilidad de los usuarios de telefonía IP desde lugares que tengan acceso a la comunicación de la red empresarial o al Internet.
- **Administración de red simplificada:** Al tener una sola red de comunicaciones, la administración es más sencilla y efectiva, ya que el mismo administrador de la red puede verificar el servicio de telefonía y los equipos asociados. Adicionalmente los cambios de configuraciones del sistema de telefonía se los puede realizar de forma remota.
- **Estandarización del sistema de telefonía IP (Interoperabilidad):** El sistema de telefonía IP permite tener varios equipos de diferentes proveedores funcionando correctamente a la par, esto hace que la telefonía sea ajustable a las necesidades y requerimientos de la red independientemente de la marca de los dispositivos a utilizarse.
- **Servicios de la telefonía IP:** El sistema de telefonía IP permite tener servicios que la telefonía tradicional no los tiene como: desvío de llamadas, envío de llamada directamente al correo de voz, mensajes de fuera de servicio, conferencias virtuales, etc.

1.1.3 COMPONENTES DE UN SISTEMA DE TELEFONÍA IP

En la telefonía IP, los componentes que intervienen en la comunicación son variados, estos dependen del escenario que se plantee el servicio, los más comunes se especifican a continuación.

1.1.3.1 Equipos Terminales IP

Los equipos terminales IP son dispositivos que permiten al usuario realizar llamadas a través del sistema de telefonía. En la actualidad estos equipos pueden ser implementados tanto en hardware (teléfonos IP) como en software (*softphones*).

1.1.3.2 Teléfonos IP

Los Teléfonos IP son dispositivos que permiten al usuario conectarse al sistema de telefonía para poder realizar y recibir llamadas. Estos equipos pueden tener una interfaz Ethernet o una interfaz inalámbrica la cual les permite conectarse a la red local para formar parte del sistema de telefonía.

1.1.3.3 Softphone [5]

Es un software que emula un teléfono IP dentro de un computador, toma muchas funcionalidades de un teléfono físico permitiendo recibir y realizar llamadas dentro del sistema de telefonía IP. Se puede instalar en el computador como en un dispositivo móvil.

1.1.3.4 Gateway

Un *gateway* es un dispositivo que permite interconectar diferentes redes y arquitecturas de comunicaciones. Un *gateway* puede tener varias interfaces para la intercomunicación como: FXO¹, FXS², GSM³, etc.: las cuales se

¹ FXO: *Foreing eXchange Office*, interfaz que envía tonos de colgado /descolgado.

² FXS: *Foreing eXchange Subscriber*, interfaz envía tono de marcado, niveles de voltaje en la llamada y corriente para la batería.

³ GSM: *Global System for Mobile communications* es un estándar de comunicación para la telefonía Móvil, una tecnología para dispositivos móviles 2G o de segunda generación.

implementan de acuerdo a la solución que se proponga para el servicio de telefonía IP.

1.1.3.5 FXS y FXO

La interfaz FXS o *Foreign eXchange Station* genera tono de marcado y suministra voltaje y corriente al dispositivo final. Integra teléfonos analógicos o dispositivos que sean capaces de recibir o detectar tonos de colgado o descolgado a una central telefónica IP.

La interfaz FXO o *Foreign eXchange Office* detecta tonos de colgado o descolgado. Permite la comunicación de una central telefónica IP con la red telefónica pública conmutada.

1.1.3.6 GSM

La interfaz GSM o *Global System for Mobile communications* permite la integración de manera más sencilla entre una central de telefonía IP, la PSTN⁴ y la telefonía celular móvil, mediante líneas celulares (*GSM SIM Cards*). Con estos dispositivos las empresas pueden reducir significativamente los costos en la telefonía [7].

1.1.3.7 Servidor o Central Telefónica IP [8], [9]

Una central telefónica IP es el componente primordial de la telefonía IP. Es un sistema encargado de procesar el tráfico de llamadas desde y hacia la PSTN utilizando el protocolo IP. Debe registrar y autenticar usuarios, autorizar llamadas mediante la asignación de perfiles a las cuentas de usuario.

1.1.4 PROTOCOLOS DE SEÑALIZACIÓN

Los protocolos de señalización en la telefonía IP se encargan de localizar a los terminales de los usuarios; establecer y liberar sesión en el proceso de una llamada. Los protocolos de señalización más utilizados son:

⁴ PSTN: *Public Switching Telephone Network* o red telefónica pública conmutada.

1.1.4.1 Protocolo SIP

El protocolo SIP (*Session Initiation Protocol*) se encuentra estandarizado por la IETF⁵ especificado en el RFC⁶ 3261, desarrollado específicamente para aplicaciones de VoIP⁷. Utiliza el puerto 5060 UDP/TCP; su sintaxis es similar al protocolo HTTP⁸ o SMTP⁹. Es un protocolo que adopta el modelo cliente-servidor mediante el establecimiento de sesiones en una red IP [10].

El modelo cliente-servidor separa la lógica de transacción y base de datos de la lógica de aplicación e interfaz de usuario, es decir separar la interfaz de usuario de la aplicación [11].

Una sesión puede ser una simple llamada telefónica de dos vías o podría ser la comunicación entre dispositivos terminales mediante el envío o recepción de mensajes.

1.1.4.2 Características

SIP es un protocolo que permite para crear, modificar y terminar sesiones con uno o más integrantes; fue diseñado para incorporar la telefonía como un servicio del Internet.

SIP posee encabezados muy extensos pero al ser un protocolo de señalización esto no afecta a los datos del usuario ya que no es un protocolo de intercambio de datos e información relevante del usuario donde sí tendría consecuencias de latencia.

⁵ IETF: *Internet Engineering Task Force*, es el organismo encargado de administrar y desarrollar los mecanismos de todo lo relacionado con el internet

⁶ RFC: *Request For Comments* o solicitudes de comentarios, es un documento que puede ser escrito por cualquier persona y que contiene una propuesta para una nueva tecnología.

⁷ VoIP: Voz sobre IP transmisión del tráfico de voz sobre el protocolo IP.

⁸ HTTP: *Hypertext Transfer Protocol*, protocolo de transferencia de hipertextos que se utiliza en las direcciones de internet

⁹ SMTP: *Simple Mail Transfer Protocol*, protocolo que permite el transporte del email en el Internet.

1.1.4.3 Mensajes SIP

SIP utiliza una serie de mensajes para la comunicación entre el cliente y servidor, basados en peticiones y correspondientes respuestas.

El formato genérico de los mensajes SIP se compone de: una línea inicial (*Request-line*), uno o más campos de cabecera (*Message Header*), una línea vacía que indica el final de las cabeceras y finalmente el cuerpo del mensaje (*Message Body*), el cual es opcional. A continuación, se describen los mensajes SIP.

- **LÍNEA INICIAL:** Contiene la versión del protocolo, en el caso de mensajes de solicitudes contiene el método y direcciones involucradas en la sesión; en el caso de mensajes de respuestas contiene el estado de la sesión.
- **CABECERA:** Contiene información relacionada con la llamada en formato de texto; por ejemplo el origen y el destino de la petición, el identificador de la llamada, etc.
- **CUERPO DEL MENSAJE O CARGA ÚTIL:** Lleva la información comúnmente mensajes SDP¹⁰ o ISUP (*ISDN User Part*) en caso de interfuncionamiento con RTCP.

1.1.4.4 Mensajes de solicitud SIP

Los mensajes de solicitudes SIP se emplean para iniciar una acción o para la petición de información. La línea inicial del mensaje de solicitud SIP (*Request-line*) contiene el nombre del mensaje, el identificador del destinatario de la petición (*Request-URI*) y la versión del protocolo SIP. Definición de algunos mensajes básicos SIP.

¹⁰ SDP: *Session Description Protocol* o protocolo de descripción de sesión.

- **INVITE**: Es un mensaje de solicitud enviado desde el cliente hacia el servidor indicando que se desea establecer el inicio de una sesión.
- **BYE**: Es un mensaje que alguno de los usuarios envía al otro extremo y que indica la finalización de una sesión existente.
- **REGISTER**: Es un mensaje que implementa el servicio de ubicación de los usuarios cuando están ejecutando una llamada, transmite la dirección IP del cliente hacia el servidor SIP.
- **ACK**: Mensaje de confirmación del cliente al mensaje de solicitud INVITE.
- **OPTIONS**: Es un mensaje de solicitud usado para consultar información sobre las capacidades y estados de clientes y servidores.
- **CANCEL**: Mensaje de cancelación a una solicitud pendiente, no se puede ejecutar sobre una llamada establecida.

La **Figura 1.1** muestra el ejemplo de la estructura del mensaje SIP de solicitud.

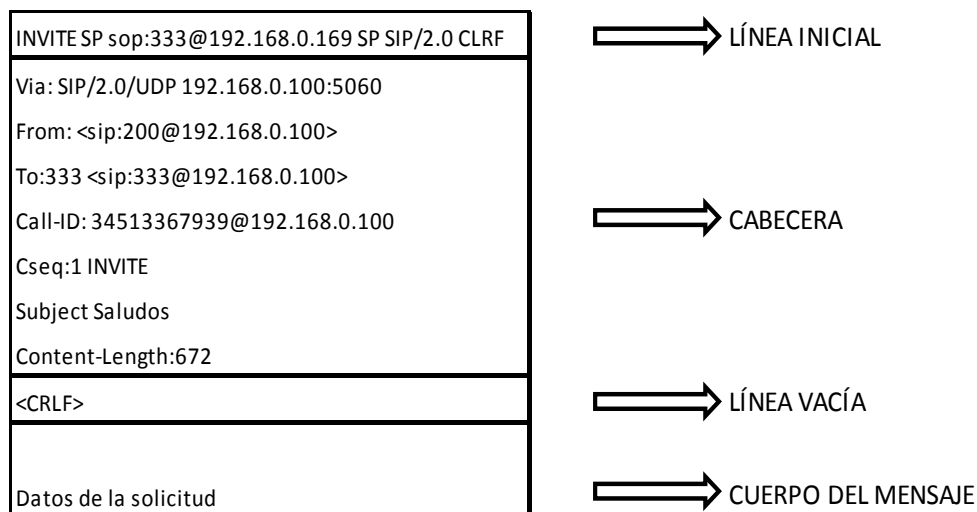


Figura 1.1 Ejemplo de la estructura del mensaje SIP de solicitud.

1.1.4.5 Mensajes de Respuesta SIP

Estos mensajes, a excepción del mensaje *ACK*, se generan como respuesta a cada solicitud SIP. Cada mensaje de respuesta está etiquetado mediante un código numérico cuyo primer dígito sirve para clasificar a cada mensaje.

La **Figura 1.2** muestra el ejemplo de la estructura del mensaje SIP de respuesta.

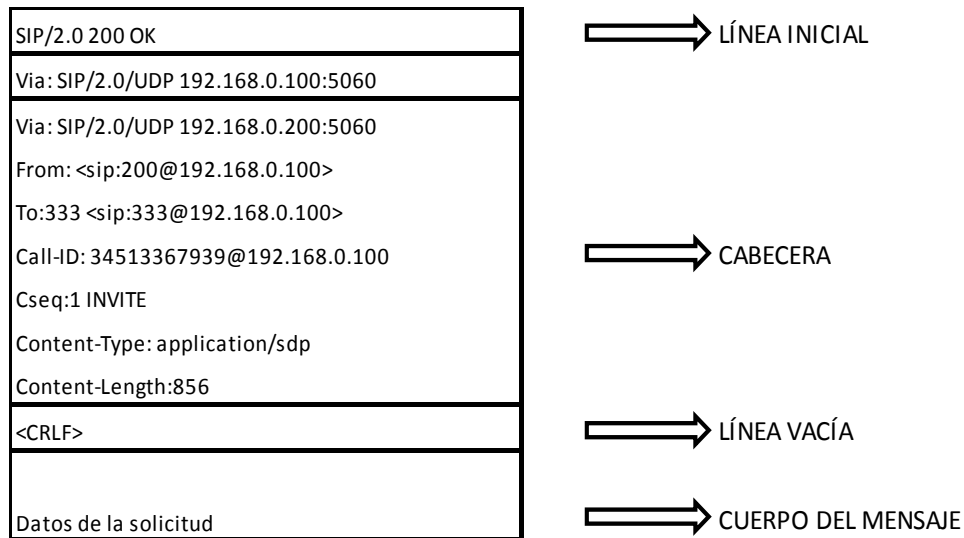


Figura 1.2 Ejemplo de la estructura del mensaje SIP de respuesta [12].

Existen 6 tipos de mensajes de respuesta SIP que a continuación se definen:

- **1XX – Respuestas informativas o Mensajes provisionales:** Indican que la solicitud fue válida y se está procesando.
- **2XX - Respuesta final de éxito:** Indican que la solicitud fue procesada con éxito. Es la última respuesta que recibe el cliente.
- **3XX - Respuesta de redirección:** Informa sobre la ubicación actual del usuario o servicio para re direccionar la petición satisfactoriamente.
- **4XX - Respuesta de fallo de solicitud:** Indica errores existentes en la solicitud del cliente SIP.

- **5XX - Respuesta de fallos de servidor:** Indican que el servidor no pudo procesar ninguna solicitud requerida.
- **6XX - Respuesta de fallos globales:** Indican errores generales, es decir que ninguna solicitud no puede ser procesada por ningún servidor.

La **Figura 1.3** muestra posibles ejemplos de mensajes de respuestas a una solicitud SIP.

Description	Code	Examples	
Informational or provisional response	1XX	100 Trying 180 Ringing 181 Call Is Being Forwarded	182 Queued
Success	2XX	200 OK 202 Accepted	
Redirect	3XX	300 Multiple Choices 301 Moved Permanently 302 Moved Temporarily	303 See Other 305 Use Proxy 380 Alternative Service
Client Errors	4XX	400 Bad Request 401 Unauthorized 402 Payment Required 403 Forbidden 404 Not Found 405 Method Not Allowed 406 Not Acceptable 407 Proxy Authentication Required 408 Request Timeout 409 Conflict 410 Gone	411 Length Required 413 Request Entity Too Large 414 Request-URI Too Large 415 Unsupported Media Type 420 Bad Extension 480 Temporarily not available 481 Call Leg/Transaction Does Not Exist 482 Loop Detected 483 Too Many Hops 484 Address Incomplete 485 Ambiguous 486 Busy Here
Server Errors	5XX	500 Internal Server Error 501 Not Implemented 502 Bad Gateway	503 Service Unavailable 504 Gateway Time-out 505 SIP Version not supported
Global Errors	6XX	600 Busy Everywhere 603 Decline	604 Does not exist anywhere 606 Not Acceptable

Figura 1.3 Mensajes de respuestas SIP [13].

1.1.4.6 Funcionamiento del protocolo SIP

El protocolo SIP basa su funcionamiento en varias transacciones para llevar a cabo una llamada. Estas transacciones gobiernan la localización de terminales, así como el inicio, establecimiento y la terminación de una llamada. Una transacción SIP es una secuencia de mensajes de solicitud y respuesta entre un cliente y un servidor [14].

Adicionalmente se define al servidor SIP como *proxy* SIP es decir un servidor intermedio encargado de comunicar los mensajes de señalización generados por los dispositivos de usuarios que va a realizar la comunicación [15].

Existen 4 tipos de transacciones:

- **Registro de Usuarios:** Los terminales de usuarios envían mensajes de petición **REGISTER** al *proxy* SIP. El *proxy* SIP verifica si el usuario puede o no ser autenticado y envía mensajes de respuesta como: 200 (**OK**), si las credenciales del usuario son correctas, caso contrario se enviará un mensaje 407 indicando que se solicitó registro y no se realizó el registro del usuario.
- **Establecimiento de sesión SIP:** El establecimiento de la sesión se da desde la necesidad de querer comunicar a dos usuarios, por ejemplo, el usuario 1 y el usuario2, a través de un servidor SIP. El establecimiento inicia cuando el usuario 1 envía un mensaje de solicitud **INVITE** hacia el servidor *proxy* SIP. El servidor *proxy* SIP responde con un mensaje de respuesta temporal 100 **TRYING** (mensaje que indica que está procesando la petición) y adicionalmente reenvía el mensaje de solicitud **INVITE** al usuario 2.

Luego el usuario 2 responde al *proxy* con un mensaje **TRYING** y seguidamente con un mensaje 180 **RINGING** (tono de llamada). Estos mensajes son reenviados por el SIP *proxy* al usuario 1. Cuando el terminal destino contesta el *proxy* SIP envía al usuario 1 un mensaje 200 **OK** indicando que el terminal destino ha contestado. Finalmente el usuario 1 confirma el inicio de la sesión con un mensaje n.

- **Llamada establecida:** Cuando la llamada es establecida se intercambia y transmite paquetes de voz a través del protocolo RTP¹¹.

¹¹ RTP: *Real time Transport Protocol* es un protocolo de transmisión de información en tiempo real.

- **Cierre de sesión:** Implica el fin de la comunicación entre dos terminales de voz. Lo puede desencadenar cualquier terminal, mediante un mensaje de solicitud **BYE** que se entrega al otro terminal mediante el *proxy*. Este último terminal confirma el cierre de sesión con un mensaje **ACK** que se entrega al otro también mediante el *proxy*.

La **Figura 1.4** muestra a detalle las transacciones en una llamada SIP.

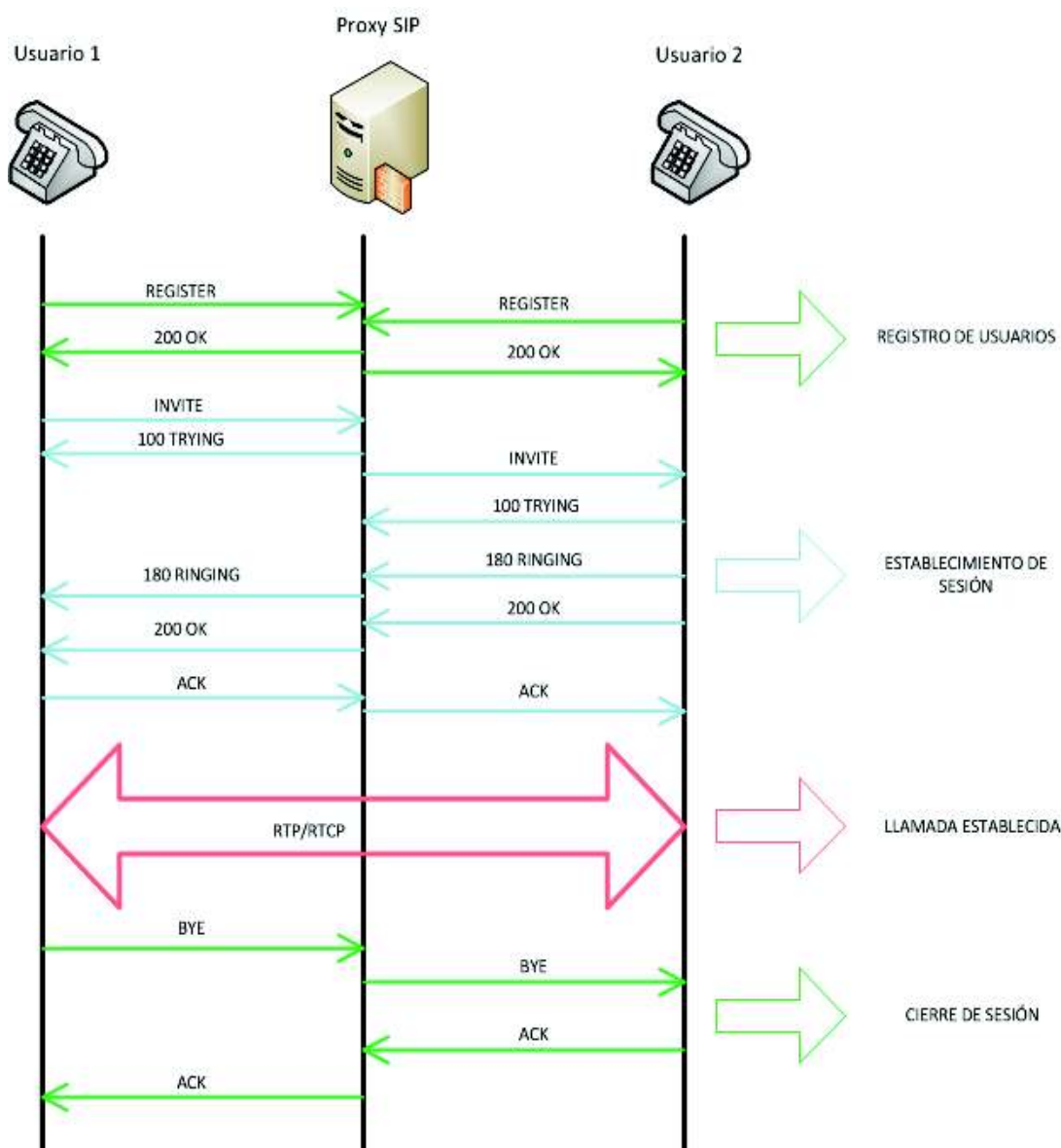


Figura 1.4 Transacciones en una llamada SIP [14].

1.1.5 PROTOCOLOS DE TRANSPORTE EN TIEMPO REAL

1.1.5.1 Protocolo RTP

El protocolo RTP fue desarrollado por el IETF¹² y estandarizado en el RFC 3550, tiene como objetivo transmitir contenido multimedia *unicast* (unidifusión) como *multicast* (multidifusión) en la red en tiempo real.

1.1.5.2 Características

RTP es un protocolo no confiable, es decir que no envía mensajes de confirmación, se ejecuta sobre UDP¹³. El protocolo UDP es un protocolo del nivel de transporte basado en el intercambio de datagramas; proporciona una forma para que las aplicaciones envíen datagramas IP¹⁴ sin tener que establecer una conexión [16]. La función principal de RTP es multiplexar en tiempo real varios flujos de datos en un solo flujo de paquetes UDP.

1.1.5.3 Protocolo RTCP (REAL TIME TRANSPORT CONTROL PROTOCOL)

El Protocolo RTCP es el protocolo de control RTP, transmite periódicamente paquetes de control a todos los integrantes de una sesión, utilizando el mismo mecanismo de distribución de paquetes de datos. Este protocolo funciona en conjunto con el protocolo RTP.

La funcionalidad primordial del protocolo RTCP es la de monitorear la entrega de datos en redes *unicast* o *multicast*, normalmente para administrar el flujo de datos, de telefonía y de videoconferencia. También sirve para monitorear si el RTP está entregando la calidad de servicio necesaria, y compensar las latencias si fuese necesario.

¹² IETF: *Internet Engineering Task Force*, es el organismo encargado de administrar y desarrollar los mecanismos de todo lo relacionado con el internet.

¹³ UDP: *User Datagram Protocol* o protocolo de nivel de transporte orientado a mensajes RFC 768 de la IETF.

¹⁴ IP: *Internet Protocol* o protocolo de Internet es un protocolo de red.

1.1.5.4 Características

El protocolo RTCP describe el intercambio de mensajes de control relacionados con la información de la calidad de servicio (retardo, *jitter*, tasa de pérdidas, etc.) entre los distintos participantes de la sesión [17]. Está definido en el RFC 3550. El protocolo RTCP se encarga de informar al remitente de la calidad de recepción y de la identidad de los interlocutores.

1.1.5.5 Mensajes RTCP

El protocolo RTCP tiene cinco tipos de mensajes de control:

- **Informe de emisor (SR, *Sender Report*)** es un mensaje enviado por el emisor. Contiene información sobre paquetes enviados y tiempos de sincronización.
- **Informe de receptor (RR, *Receiver Report*)** es un mensaje enviado por el receptor. Contienen información sobre paquetes recibidos, perdidos, *jitter* y retardos medidos.
- **Descripción de fuente (SDES, *Source Description*):** Descripciones de un emisor nombre, email, teléfono, etc.
- **Fin de la participación en la sesión (BYE)** es un mensaje enviado tanto por el emisor como el receptor cuando un usuario termina la sesión.
- **Funciones específicas de la aplicación (APP)** es un mensaje de señalización que indica funciones específicas de acuerdo a la aplicación utilizada.

1.1.6 CODIFICADORES/DECODIFICADORES (CODECS)

Una de las características del protocolo RTP es la identificación del sistema en el que está codificada la información de audio o video (sistema codec utilizado).

CODEC es la abreviación de CODIFICADOR/DECODIFICADOR convierte una señal de audio analógico en un formato de audio digital para transmitirlo sobre una red de datos. El codec con frecuencia realiza la función de compresión con el fin de ahorro de ancho de banda [18]. En el destino el decodificador descomprime la señal digital a un formato de señal de audio para poder reproducirlo.

Los codecs operan usando algoritmos avanzados que les permiten tomar las muestras, ordenar, comprimir y empaquetar los datos. Los codecs se pueden clasificar en base a diferentes factores como: tasa de bits, calidad de audio codificado, complejidad, tipo de tecnología o retardo que introducen, existen codecs de audio y video [19].

La **Tabla 1.1** muestra los codecs más utilizados actualmente.

Tabla 1.1 CODECS para la telefonía IP.

Nombre	Estándar	Velocidad de Transmisión (Kbps)	Frecuencia de Muestreo (MHz)	Tamaño de la trama (ms)	Licenciamiento
G.711	ITU-T	64	8	Muestreada	NO
G.723	ITU-T	5.6/6.3	8	30	SI
G.726	ITU-T	16/24/32/40	8	Muestreada	NO
G.729	ITU-T	8	8	10	SI
GSM	ETSI	13	8	22.5	NO
ILBC	-	15.3/12.3	8	20/30	NO

1.1.7 ASTERISK

Asterisk es una plataforma integral de comunicaciones, para la creación de aplicaciones de comunicaciones. Asterisk es un software *open-source*¹⁵ que convierte a un ordenador normal en un sofisticado servidor de comunicaciones VoIP.

¹⁵ *open source*: significa código abierto es decir que se puede utilizar el código y desarrollarlo para el beneficio de la comunidad en general.

Asterisk se adapta a las necesidades de cualquier empresa, desde una pyme que necesite una centralita sencilla para coordinar las llamadas entrantes y salientes de una pequeña oficina, hasta una gran compañía que necesite un *callcenter*¹⁶ de atención al cliente con cientos de operadores [20].

Asterisk puede funcionar con varios sistemas operativos como: Linux, BSD, OS X y Windows, y trabajará con la mayoría de los teléfonos compatibles SIP y *softphones* [8].

Asterisk brinda todas las funcionalidades básicas de una central telefónica convencional y muchas funcionalidades avanzadas que podrían incrementar el costo de la solución si fuera propietario.

A continuación, se mencionan algunas funcionalidades básicas de Asterisk [21]:

- ***Transferencia directa:*** Permite transferir una llamada a otra extensión.
- ***Transferencia atendida:*** Se consulta primero al nuevo destino si quiere que se le pase la llamada, para finalmente realizar la transferencia de la llamada.
- ***Desvíos:*** Es una transferencia automática de una llamada entrante hacia un terminal determinado (interno o externo) cuando el terminal destino está ocupado o no se contesta.
- ***Capturas (de grupo o de extensión):*** La captura permite atender una llamada que se está recibiendo en una extensión desde otra distinta.
- ***Conferencia múltiple:*** La conferencia múltiple permite establecer una comunicación entre múltiples usuarios de la centralita.

¹⁶ *Callcenter:* es un centro donde agentes, asesores y supervisores realizan llamadas según un objetivo empresarial.

- **Ring Groups o grupos de extensiones** se define como una extensión “virtual” que llama a un grupo de teléfonos dependiendo de la estrategia de timbrado previamente establecida, deteniéndose cuando uno de ellos es contestado. Las estrategias de timbrado pueden ser: **Todos** (llama a todos los teléfonos simultáneamente hasta que uno conteste), **Ascendente** (llamada a los teléfonos según prioridades en orden alfanumérico), **Descendente** (llama a los teléfonos según ordenamiento alfanumérico descendente), **Aleatorio** (Llama a los teléfonos aleatoriamente hasta que uno conteste, esta estrategia tiene memoria, recuerda que dispositivo ya sonó).
- **DND (Do Not Disturb)** o No molestar esta función desconecta al teléfono impidiéndole recibir llamadas. Puede ser configurado en Asterisk mediante un código o directamente en el terminal (Teléfono IP).

1.1.7.1 FUNCIONALIDADES AVANZADAS DE ASTERISK

Entre algunas de las funcionalidades avanzadas de Asterisk, se pueden encontrar las siguientes:

- **Correo Vocal (Voicemail) integrado con correo electrónico.** En caso de que el usuario no pueda atender una llamada, se puede programar que esta se transfiera a un sistema de buzón de voz. En caso de que se deje un mensaje, la central podría enviar un correo electrónico avisando del mismo al usuario destinatario.
- **Operadora Automática (IVR):** Un IVR es una aplicación de telefonía que permite interactuar con el usuario que realiza la llamada donde el cliente pulsa opciones previamente anunciadas y accede de forma automática a los destinos programados.
- **Música en espera con archivos WAV:** Asterisk nos permite introducir música en espera basadas en archivos de tipo wav y mp3.
- **Colas de Llamadas (ACD):** Un sistema de colas es una aplicación que distribuye las llamadas entrantes a un grupo específico de agentes de acuerdo a una determinada estrategia.

- **Salas de Audio-Conferencias:** Una sala de audio conferencias es un sistema que permite conectar a múltiples usuarios en una misma conversación telefónica.
- **Gestión de llamadas entrantes según horario o fecha):** Asterisk permite definir horarios y calendarios laborales que permita hacer un tratamiento diferenciado de las llamadas de acuerdo a las restricciones de la fecha y hora.
- **Callback:** consiste en la programación de una llamada automática de respuesta a una llamada perdida.
- **Integración CTI:** Integración de la telefonía con sistemas informáticos de gestión comercial o de atención al cliente (CRM). Estos sistemas permiten por ejemplo ejecutar una llamada desde el PC o bien recibir información sobre una llamada entrante en la pantalla.
- **Informes detallados de llamadas (CDR):** Detalle de llamadas realizadas/recibidas por extensión, para análisis de costos departamentales, por cliente o incluso para facturación.
- **Línea Troncal:** Una línea troncal tiene como propósito proporcionar la conexión compartida de múltiples líneas telefónicas hacia la PSTN. Es el camino que conecta un recurso interno con un recurso externo [22]. Las conexiones pueden ser: una o varias troncales analógicas, uno o varios enlaces digitales de alta capacidad.
- **Grabación de llamada** [23]: Asterisk permite entre una de sus características grabar las llamadas que se colocan a través de la red telefónica. Por ejemplo las llamadas que se realizan a opciones de servicio al cliente generalmente son grabadas por temas de auditoría o ventas.
- **Plan de Marcado:** El plan de marcado es la parte más importante de Asterisk porque aquí se controla cómo se direccionan las llamadas entrantes y salientes además de que módulos y aplicaciones deben ejecutarse según lo configurado. En plan de marcado en Asterisk está

configurado dentro del archivo **extensions.conf**, el cual está conformado por cuatro conceptos primordiales: Contextos, Extensiones, Prioridades y Aplicaciones.

- **Contextos:** Los contextos son las partes o secciones que puede tener un plan de marcado, se los puede definir como el conjunto o agrupación de extensiones [23]. En cada contexto las extensiones utilizan aplicaciones para realizar las gestiones que se necesite. Las extensiones de un contexto pueden interactuar con las extensiones de otro contexto mediante aplicaciones como `Goto`¹⁷ o `GotoIf`¹⁸, o se puede incluir todas las extensiones de otro contexto mediante la directiva `include`, la cual incluye todas las extensiones del contexto indicado. La sintaxis de un contexto consiste en definir un nombre, éste nombre puede contener números, letras de la A –Z, mayúsculas o minúsculas, el cual se lo debe encerrar en corchetes. Ejemplo: `[callmatic]`.
- **Extensiones:** En Asterisk una extensión se puede representar con un número o letras a la que se le asignan una secuencia de pasos a seguir dentro del contexto. La sintaxis de una extensión tiene cuatro partes, primero escribir el comando **exten** seguido de los símbolos, (`=>`), igual y mayor, luego escribir la palabra o número que tendrá la extensión, la prioridad para ser ejecutada y finalmente la aplicación. Estas partes deben estar separadas por comas como se observa en la **Figura 1.5**.

Comando exten seguido de =>	Palabra o número	Prioridad	Aplicacion
<code>exten =></code>	<code>audiofile,</code>	<code>1,</code>	<code>Wait(1)</code>

Figura 1.5 Sintaxis de una extensión.

¹⁷ GoTo: Aplicación de Asterisk que permite realizar saltos a contexto y extensiones especificadas dentro del plan de marcación.

¹⁸ GotoIf: Aplicación de Asterisk que permite realizar saltos a contexto y extensiones especificadas si se cumple con la condición especificada dentro del plan de marcación.

- **Prioridades:** La prioridad es el orden como se ejecutan las instrucciones de la extensión, se enumeran en orden desde el número 1. Cuando las extensiones tienen muchos pasos a ejecutar puede existir conflictos a definir un número en la prioridad; por lo que se recomienda escribir la letra n en el campo de la prioridad. Adicionalmente se utiliza el comando `same`, significa lo mismo que antes [24], en lugar de `exten` para evitar escribir el nombre de la extensión reiteradas veces, como se muestra en la **Figura 1.6** desde el segundo paso.

```
exten => audiofile,1,Wait(1)
same => n,Answer()
same => n,Playback(${CV_FILENAME})
same => n,Hangup()
```

Figura 1.6 Ejemplo de uso de prioridades y comando same.

- **Aplicaciones:** Las aplicaciones son como programas dentro de Asterisk que permiten realizar determinadas acciones. Por ejemplo, reproducir un sonido, colgar una llamada, etc. Las aplicaciones requieren de información extra (argumentos) para poder ejecutar la acción encargada. La sintaxis de una aplicación es el nombre de la aplicación existente en Asterisk seguida de los argumentos encerrados en paréntesis. Ejemplo: `wait(2)`¹⁹.
- ***Canales de comunicación:*** Para permitir la comunicación entre varios dispositivos Asterisk utiliza muchos protocolos, uno de los más comunes y que utilizaremos en el prototipo es el protocolo SIP por lo que se debe configurar el archivo `sip.conf`. Dentro del archivo está la configuración del dispositivo terminal (*Gateway* GSM) que va a permitir comunicarnos con la red telefónica.

Este archivo está dividido en secciones, estas secciones inician con un nombre encerrado entre corchetes. Cada sección contiene información específica y

¹⁹ *Wait*: aplicación de Asterisk que permite esperar el tiempo que se configure en su argumento para continuar con la siguiente orden.

características para cada canal. La sección [general] es estándar y se encuentra al inicio del archivo, define parámetros por defecto y para el prototipo se habilitó el envío de mensajes de texto instantáneos y el contexto al que pertenece (cvmsg).

La **Figura 1.7** presenta un ejemplo de la configuración de un archivo **sip.conf**.

```
[general]
context=default           ; contexto por defecto para llamadas entrantes
allowguest=no            ; rechazar llamadas no autenticadas
udpbindaddr=0.0.0.0      ; escucha peticiones UDP en todas las interfaces
tcpenable=no            ; deshabilitar soporte TCP.
accept_outofcall_message=yes ; permite el envío de mensajes de texto instantaneos
outofcall_message_context=msg ; especifica el contexto en el plan de marcado
                           ; donde se procesaran los mensajes enviados

[gw-gsm]                 ; seccion del terminal gw-gsm
type=friend              ; indica al controlados del canal asociar el nombre del terminal y la IP
host=192.168.0.12        ; se establece la direccion IP que el gateway gsm tendra en su configuracion
context=callmatic        ; contexto con el cual se asocia las llamadas realizadas desde la aplicacion
disallow=all             ; deshabilita todos los codecs de voz
allow=alaw               ; habilita al codec seleccionado
allow=ulaw               ; habilita al segundo codec seleccionado
```

Figura 1.7 Configuración archivo sip.conf.

1.1.7.2 MECANISMOS DE GENERACIÓN AUTOMÁTICA DE LLAMADAS

Los mecanismos de generación automática de llamadas son algoritmos de telefonía que procesan varias instrucciones para realizar una llamada. Los mecanismos de generación automática de llamadas permiten al sistema de telefonía IP agilizar la consulta de una lista de contactos, efectuar la llamada y dejar o no un mensaje. Estos mecanismos pueden ser automáticos o manuales y están orientados al campo del *marketing*.

Las empresas como *callcenters* o de telemarketing que han implementado sistemas de telefonía IP utilizan comúnmente estos mecanismos para generar sus llamadas con o sin la intervención de un asesor o agente. Realizan llamadas de forma masiva automática o manual a través de campañas para realizar ventas, encuestas, cobros, etc.

Un agente o asesor es la persona encargada de realizar o recibir llamadas dentro de un *callcenter* o centro de llamadas. Una campaña en telefonía IP es un listado de números telefónicos con información adicional de los clientes a ser contactados según el criterio y giro de negocio de la empresa [25].

Existen diferentes formas para la generación automática de llamadas. A continuación, se detallan algunas de ellas:

- **Aplicación y Recursos Originate:** Originate genera una llamada desde la consola de Asterisk (CLI²⁰) y la asocia a un contexto o aplicación determinada.
- **Scripts de Generación de Llamadas:** Se puede crear un *script*²¹ para que ejecute varios comandos personalizados en el sistema. Este *script* puede tener cualquier nombre pero es requisito que tenga la extensión **.call**. Para ejecutarlo hay que moverlos al directorio por defecto de Asterisk: ***/var/spool/asterisk/outgoing/***.
- **Marcadores:** Los marcadores son específicamente creados para los sistemas de telefonía IP y muy utilizados en los *callcenters*. El objetivo principal de los marcadores es la automatización del proceso de marcación a números telefónicos para incrementar la productividad.
- Existen varios tipos de marcadores como:
 - **Marcadores manuales:** El agente o asesor realiza la marcación manualmente del número telefónico.
 - **Marcadores predictivos:** Se genera mediante un algoritmo basado en estrategias estadísticas que toman varias variables y factores del sistema de telefonía para asignar a cada agente potenciales clientes de manera más efectiva.

²⁰ CLI: *Command Line Interface* o consola de comandos de Asterisk que permite supervisión y corrección de posibles fallos que pueden surgir en el sistema.

²¹ *Script:* es un conjunto de comandos u órdenes en serie que se ejecutarán de manera secuencial [41].

- **Marcadores progresivos:** Los marcadores progresivos basan su algoritmo en la disponibilidad de los agentes. Son capaces de detectar los diferentes tipos de eventos durante las llamadas y conseguir un cliente óptimo para asignarlo a un agente.

1.1.7.3 CONVERSION DE TEXTO A VOZ (TTS - *TEXT TO SPEECH*)

Es la generación automática de texto escrito a una palabra hablada de una voz artificial utilizando síntesis de voz. Es decir que es un sistema que convierte textos escritos en voz sintética.

La voz sintetizada es una voz artificial generada automáticamente mediante un proceso de sintetización del habla y almacenada como un archivo o fichero. Asterisk ve a este archivo como un programa externo el cual genera un archivo de sonido como fuente. El archivo de sonido obtenido se reproduce en Asterisk como cualquier otro archivo de sonido finalmente se podrá escuchar el audio personalizado [26].

Para el prototipo se va a utilizar la aplicación de Festival para Asterisk. Festival es una aplicación gratuita de Linux para la conversión de texto a voz. Permite reproducir el texto en idioma inglés que viene configurado por defecto pero se puede instalar los paquetes del idioma en español y configurarlos.

1.2 MENSAJES DE TEXTO SMS

El SMS (*Short Message Service*) fue diseñado originalmente como parte del estándar de telefonía móvil digital GSM y en la actualidad está disponible en varias redes móviles incluidas las redes 4G²². El mensaje de texto corto es un servicio que permite el intercambio de mensajes de texto entre dispositivos móviles, pueden ser enviados y recibidos entre los distintos operadores de las diferentes redes y países. SMS es un mecanismo de entrega de mensajes cortos usando la red de telefonía móvil [28].

²² 4G: La cuarta generación de tecnología en telecomunicaciones móviles [27].

1.2.1 CARACTERÍSTICAS

Se define como mensaje “corto” a la cadena alfanumérica de hasta 140 caracteres o 160 caracteres de 7 bits y cuyo encapsulado incluye una serie de parámetros. En el alfabeto latino pueden ser letras, números, símbolos y espacios en blanco y llegar a 160 caracteres, en alfabetos como el chino los SMS tiene un máximo de 70 caracteres.

Los mensajes de texto cuentan con características como su brevedad y su fácil envío. En estos mensajes se comunican cosas más inmediatas, urgentes o breves ya que a veces la persona no puede hablar en todas las situaciones o quizás no es fácil contactarse con el otro. Los mensajes de texto son procesados por un SMSC²³ o centro de servicios de mensajes cortos el cual se encarga de almacenarlos hasta que son enviados.

Además, SMS proporciona una marca de tiempo que informa el envío del mensaje al SMSC. Envía una indicación al teléfono que emite el SMS, de si hay o no más mensajes para ser enviados o la cantidad de mensajes adicionales que aún se van a enviar.

1.2.2 VENTAJAS DEL SMS

Hemos visto que la forma más rápida y directa de comunicarse con alguna persona es a través de un terminal móvil, mediante una llamada o un mensaje de texto SMS. Algunas ventajas de los mensajes de texto SMS son:

- Un mensaje de texto proporciona discreción en la conversa ya que no interviene la comunicación sonora facilitando la privacidad.
- Enviar un mensaje de texto lleva menos tiempo que realizar una llamada o enviar un email.

²³ SMSC: *Short Message Service Center* o Central de Servicio de Mensajes Cortos, su función es enviar o recibir mensajes de texto SMS.

- Son un gran sistema de comunicación para gente con problemas de audición.
- Se utilizan los SMS para actividades de mercadeo, encuestas de satisfacción, difusión de alertas, etc.

1.3 DESARROLLO DE SOFTWARE

Es llamado también “Ciclo de vida del desarrollo de software”. Es una estructura de construcción de software mediante descripciones y requerimientos. Está definido como una serie de acciones u operaciones que conducen a un fin.

Para realizar el desarrollo de software se tienen varias etapas del ciclo de vida del sistema que se describen a continuación.

1.3.1 ANÁLISIS DE REQUERIMIENTOS

Es la primera etapa del desarrollo de software. Es una parte fundamental del proceso de desarrollo de una aplicación. Es fundamental que a través de una recopilación de requerimientos el desarrollador o desarrolladores del software comprendan completamente la naturaleza de la aplicación.

Los requerimientos son la base para la actividad de la estimación de software, puede ser una visión abstracta de alto nivel de una función que el sistema debe suministrar o una restricción del sistema. Los requerimientos pueden clasificarse en funcionales y no funcionales.

1.3.1.1 REQUERIMIENTOS FUNCIONALES

Los requerimientos funcionales son los encargados de definir lo que la aplicación de software realizará según las necesidades del cliente. Son los principales componentes de las estimaciones. Un ejemplo de requerimiento funcional puede ser el valor total de cuantas llamadas se realizaron en una fecha establecida.

1.3.1.2 REQUERIMIENTOS NO FUNCIONALES

Los requerimientos no funcionales son los encargados de definir lo que la herramienta de software debe tener en cuanto a apariencia, características y propiedades para cumplir con la tarea para lo cual fue creada. Un ejemplo de requerimiento no funcional puede ser que el software sea intuitivo para el usuario, de fácil manejo, accesibilidad y manipulación.

Los requisitos de software no se pueden definir de forma estática durante el proyecto. La definición de un requisito es en un instante, pero sufre cambios con el tiempo.

1.3.1.3 CASOS DE USO

Un caso de uso es un conjunto de pasos de las iteraciones entre el usuario y el sistema. Una iteración representa la o las intervenciones que se tuvo con el cliente sobre los requisitos y modificaciones de los mismos para el desarrollo del software.

Los casos de uso describen condiciones previas necesarias para ejecución correcta y condiciones posteriores que son resultados de las medidas tomadas en el transcurso del proyecto. Los casos de uso son representaciones visuales simples de las interacciones del sistema con el mundo exterior [29].

Se documentan con texto informal para describir una tarea que servirá como meta del caso; con un nivel apropiado de detalle y bastante sencillo.

Contiene los siguientes campos:

- Autor: Es la persona que registra el requerimiento.
- Actor: Usuario que interviene en el sistema a desarrollarse.
- Fecha: Fecha en la que se elaboró el caso de uso.
- Código: Identificador del caso de uso.
- Título: Nombre que se asignó al caso de uso.

- Descripción: Conjunto de pasos que describe un requerimiento con éxito.
- Observación: Acciones a tomar frente a un fallo del requerimiento principal.

La **Figura 1.8** muestra un ejemplo de Caso de Uso.

Autor: Evelyn Yessenia Achig Ruano	Usuario: Administrador sistema
Código:HU1-01	Fecha: 2016-12-10
Título: Crear Campaña con audios pregrabados	
Descripción: El sistema debe dejar ingresar los datos de la campaña como nombre de la campaña, canales (número de llamadas simultaneas limitante líneas de salida que se tenga), troncal, descripción.	
Observaciones: Cada campaña tiene asociada un audio en común que se reproducirá a los contactos asociados.	

Figura 1.8 Caso de Uso.

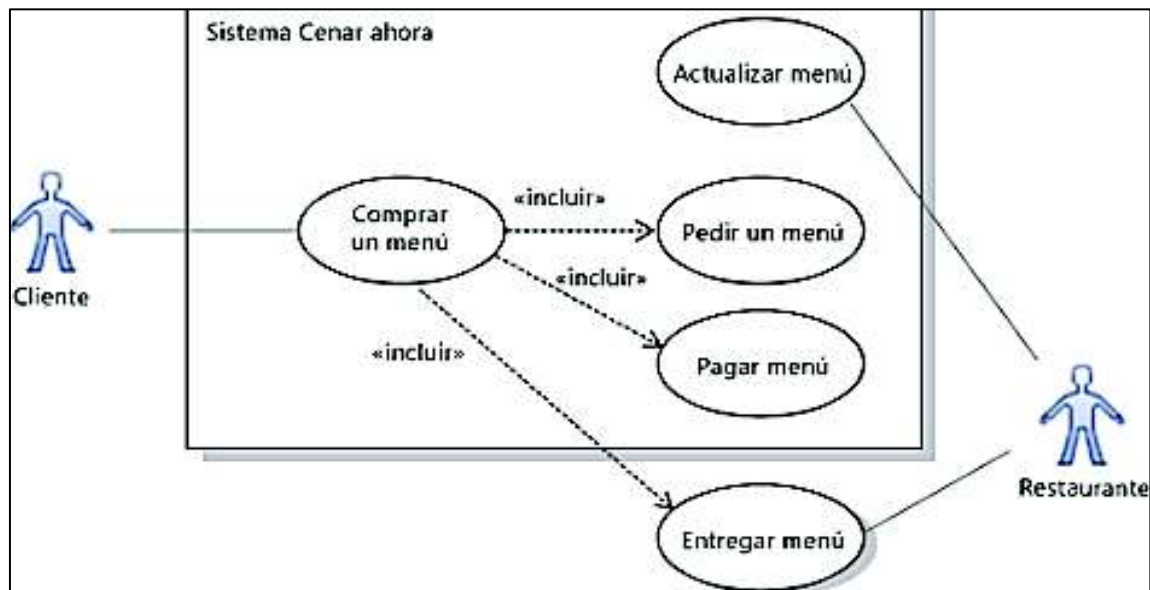


Figura 1.9 Diagrama de Casos de Uso.

Los casos de uso pueden ser representados mediante un diagrama en el que intervienen los componentes de casos de usos y relaciones.

Un diagrama de caso de uso es el contenido de los propios casos de uso, sólo muestran el nombre de los casos de usos, los actores que participan y algunas relaciones entre actor y sistema.

La **Figura 1.9** muestra un ejemplo de Diagrama de Casos de Uso.

1.3.1.4 HISTORIAS DE USUARIO

Una historia de usuario es una descripción de una acción que el usuario efectúa en el sistema. Son escritas en primera persona con un lenguaje simple con el fin de establecer el entendimiento de los requerimientos entre el cliente y el desarrollador o desarrolladores del sistema. A la hora de escribir una historia de usuario debe ser en pocas palabras y esta debe evitar cubrir más de una funcionalidad.

Las historias de usuario se utilizan como una técnica de las metodologías ágiles, utilizadas para especificar los requisitos funcionales y operativos de un producto de software a ser desarrollado. Todas las historias de usuario deben tener un criterio de aceptación bien definido por el cliente, y permiten al desarrollador o desarrolladores identificar si la historia de usuario esta implementada por completo y con éxito [30].

1.3.2 DISEÑO DE SOFTWARE

Es la etapa en la cual se convierte una especificación del sistema en un sistema ejecutable. Es un proceso para planificar y dar solución a un software requerido. Para el diseño de software existe una gran variedad de modelos o paradigmas para su desarrollo.

1.3.2.1 MODELOS DE DESARROLLO DE SOFTWARE

Los modelos de desarrollo son abstracciones de las relaciones entre las actividades y los diferentes enfoques del proceso de software. El comprender los principios, beneficios, ventajas y desventajas de los distintos modelos ayudan al correcto desarrollo del software.

Existen varios tipos de modelos como son:

- **El modelo en cascada o modelo lineal:** es el modelo más antiguo que establece un enfoque secuencial para el desarrollo. Este modelo define etapas para todo el proceso que son: pre análisis, análisis, diseño, desarrollo, pruebas, implantación y mantenimiento. Cada etapa tiene un conjunto de varias actividades y procesos que son primordiales para la siguiente etapa.

Una gran desventaja de este modelo es que no se puede volver atrás; si la fase de diseño empezó mal y no se corrigió a tiempo las siguientes fases tendrán el error y en la fase final se verán complicaciones [31]. No se puede corregir o implementar requerimientos del cliente en el transcurso del proceso del software.

- **El modelo incremental** que es la evolución del modelo en cascada, al ser no secuencial corrige el inconveniente mostrado por el modelo en cascada.
- **El modelo espiral** es un modelo no secuencial, más complejo que los anteriores. Las actividades de este modelo forman un espiral con segmentos, cada repetición o iteración representa un conjunto de actividades. Las actividades a ser implementadas incluyen un análisis de riesgos para ser elegidas. Es el mejor modelo para el desarrollo de grandes sistemas ya que actualmente tiene un enfoque más realista para el desarrollo del software.

El modelo Incremental concuerda con los requisitos del proyecto por lo que se explicara con más detalle.

1.3.2.2 MODELO INCREMENTAL

El modelo incremental basa su estructura en iteraciones. Permite crear versiones del sistema y mostrárselas al cliente para validar sus requisitos. Si no

se cumple con los objetivos se debe analizar los posibles cambios. Se debe planificar una segunda, tercera o n-versiones del software, hasta alcanzar una versión final del sistema con todo lo solicitado inicialmente o en el proceso.

Pueden realizarse tantas iteraciones (también llamadas incrementos o versiones) como sean necesarias; el sistema es una integración de resultados sucesivos obtenidos después de cada iteración. Una desventaja de este modelo es que presenta una estructura deficiente debido a los frecuentes cambios lo que lo hace más complejo y puede incrementar el presupuesto y costos finales.

1.3.3 METODOLOGÍAS DE DESARROLLO DE SOFTWARE

Las metodologías de desarrollo de software son un conjunto de pasos y procedimientos que deben seguirse para el desarrollo de software. Son un conjunto de filosofías, fases, procedimientos, reglas, técnicas, herramientas, documentación y aspectos de formación para los desarrolladores. Una metodología es el camino a seguir para desarrollar software de manera ordenada.

1.3.3.1 METODOLOGÍAS DE DESARROLLO ÁGILES

Las metodologías de desarrollo ágiles se aplican en proyectos en donde el sistema es muy cambiante y con tiempos de desarrollo corto pero manteniendo una alta calidad en el proyecto.

La Agilidad es un aspecto que se puede incorporar a las metodologías estructuradas. Entre las metodologías más conocidas se encuentran: XP (*eXtreme Programming*), SCRUMP, RAD (*Rapid Application Development*).

Los métodos ágiles son ideales para desarrollos de sistemas de negocios pequeños y de tamaño medio así como productos para ordenadores personales.

1.3.3.2 PRINCIPIOS DEL MODELADO ÁGIL

Los principios del modelado ágil son características que diferencian un proceso ágil de un proceso tradicional.

A continuación, se muestran los doce principios del modelado ágil. Los dos primeros son generales y engloban todo el concepto de la metodología de desarrollo ágil y los siguientes involucran a los actores y tareas a realizarse [32], [33].

- La mayor prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor.
- Dar la bienvenida a los cambios. Se capturan los cambios para que el cliente tenga una ventaja competitiva.
- Entregar frecuentemente software que funcione desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas.
- La gente del negocio y los desarrolladores deben trabajar juntos a lo largo del proyecto.
- Construir el proyecto en torno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir finalizar el trabajo.
- El diálogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo.
- El software que funciona es la medida principal de progreso.
- Los procesos ágiles promueven un desarrollo sostenible. Los promotores, desarrolladores y usuarios deberían ser capaces de mantener una paz constante.
- La atención continua a la calidad técnica y al buen diseño mejora la agilidad.

- La simplicidad es esencial.
- Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos.
- En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivo, y según esto sintoniza y ajusta su comportamiento.

1.3.3.3 METODOLOGÍA XP [34]

La metodología XP pone más énfasis en la adaptabilidad que en la previsibilidad. Es una metodología utilizada para desarrollar software de la manera más rápida posible, de alta calidad y satisfaciendo las necesidades del cliente. La metodología XP se caracteriza por tener ciclos de desarrollo extremadamente rápidos, continua retroalimentación por parte del cliente, integración constante entre sus desarrolladores y pruebas automatizadas.

La filosofía de XP promueve el trabajo en equipo, preocupándose del aprendizaje de los desarrolladores y de un buen clima de trabajo. La metodología de programación XP proporciona una serie de pasos fundamentales para crear proyectos de software. Estos pasos no son fijos ni obligatorios, lo que permite discutir sobre las necesidades del uso de algunos pasos para el desarrollo del proyecto

XP utiliza historias de usuario como técnica para la obtención de los requerimientos, las mismas que son implementadas como conjunto de tareas. La metodología XP no se puede aplicar a proyectos de gran escala, que requiera mucho personal o que divida a los desarrolladores en proyectos más pequeños [35].

1.3.4 PATRÓN DE DISEÑO MODELO VISTA CONTROLADOR

Actualmente el patrón Modelo Vista Controlador (MVC) el más utilizado en la programación orientada a objetos. Clasifica a los objetos en tres componentes: modelo, vista, controlador. Un patrón se denomina al conjunto del problema y solución en el diseño del software.

La **Figura 1.10** muestra un ejemplo del Patrón MVC.

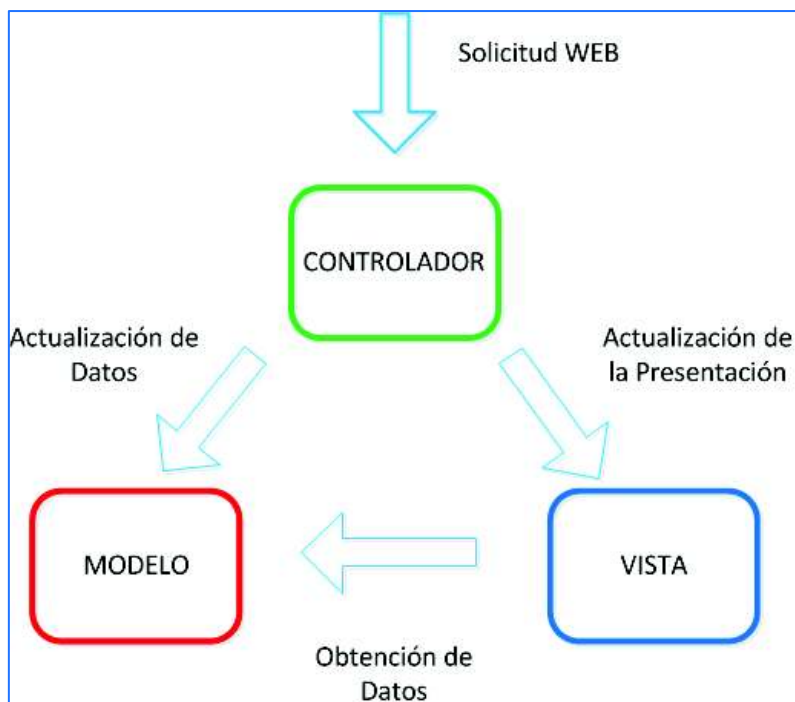


Figura 1.10 Vista-Modelo-Controlador

1.3.4.1 MODELO

El modelo (proceso) contiene solamente los datos puros de aplicación y la lógica y procesos computacionales. Por ejemplo, objetos que tengan *arrays*²⁴ con los datos de la aplicación o datos del usuario, u objetos que interactúan directamente con la base de datos.

1.3.4.2 VISTA

La vista (salida) dibuja la interfaz gráfica, muestra los resultados de la información al usuario final. Conoce y accede a los datos de las peticiones del usuario y entrega al controlador esta información. Por ejemplo, muestra la información de un usuario como nombre y apellido según lo se le indique en la interfaz gráfica.

²⁴ *Arrays*: permite almacenar un conjunto de datos homogéneos.

1.3.4.3 CONTROLADOR

El controlador (entrada) está entre la vista y el modelo, recoge los datos según la petición recibida y comunica el resultado a la vista para que se muestre en la interfaz gráfica.

CAPÍTULO 2

2 ANÁLISIS DE REQUERIMIENTOS Y DISEÑO DEL PROTOTIPO AUTOMATIC SEND&DIALING (ASD)

2.1 ANÁLISIS DE REQUERIMIENTOS Y NECESIDADES DEL PROTOTIPO

Las necesidades que el prototipo debe cumplir fueron obtenidas a partir de una encuesta realizada a miembros de *callcenters*²⁵ (supervisores y agentes) de tres empresas diferentes. También se elaboraron historias de usuario (detalladas en el **capítulo 1 sección 1.3.1.4**) para comprender: los requisitos, características y funcionalidades a considerar en la aplicación web. La aplicación web toma como directriz para el desarrollo del software la metodología XP.

Se describe la encuesta realizada para la determinación de algunas necesidades que tienen las organizaciones con respecto a la generación automática de llamadas y mensajes de texto.

2.1.1 ENCUESTA DE REQUERIMIENTOS PARA LLAMADAS Y MENSAJES AUTOMÁTICOS

Esta encuesta se realizó a tres empresas cuyos *callcenters* son de diferentes tipos (pequeños, medianos y grandes). Las tres empresas a las cuales se realizó la encuesta realizan gestiones tanto de cobranza, ventas e información al cliente.

La **Tabla 2.1** muestra los resultados tabulados de las respuestas obtenidas después de realizar las encuestas a las diferentes empresas.

²⁵*callcenter*: es un centro donde agentes, asesores y supervisores realizan o reciben llamadas según un objetivo empresarial

Tabla 2.1 Encuesta de Requerimientos.

PREGUNTA	Posibles Respuestas	Número de llamadas o mensajes	% Equivalentes
¿Con cuántos clientes en promedio tiene comunicación diaria?	0 a 500	200	7%
	500 a 1500	500	19%
	Más de 1500	2000	74%
¿Qué destinos son los más frecuentes al contactar a sus clientes diariamente?	Números Celulares	225	60%
	Números Convencionales	150	40%
¿Cuál es el horario más común en el que su cliente contesta una llamada?	10:00-11:00	35	10%
	13:00-14:00	98	27%
	16:30-18:30	227	63%
¿Cuál es el horario menos asertivo en el cual su cliente contesta una llamada?	06:00-09:00	290	81%
	10:00-13:00	50	14%
	14:00-16:30	20	5%
¿Cuál es el porcentaje de agentes que asigna la empresa para realizar campañas informativas?	10 %		10%
	30%		30%
	60%		60%
¿Le parece que el contacto con el cliente mediante mensajes SMS es adecuado?	SI	225	60%
	NO	150	40%
¿Cuánto tiempo el operador tarda en brindar la información al cliente?	5 minutos	45	14%
	7 minutos	170	53%
	12 minutos	105	33%
¿Cuál es la frecuencia establece comunicación con sus clientes?	Periódicamente	6000	20%
	Bajo demanda	24000	80%
¿Qué medios adicionales utiliza para comunicar o informar a sus clientes aviso o promociones diariamente?	Correo electrónico	10	67%
	Distribuidor autorizado de llamadas mediante número corto ²⁶	2	13%
	Distribuido de sms autorizado mediante un número corto	3	20%
¿Qué porcentaje de los mensajes en las llamadas generados a sus clientes son genéricos?	0% a 35%		15%
	35% a 70%		15%
	70% a 100%		70%

²⁶ Número corto: es un servicio de comunicación mediante mensajes de texto de forma ágil y eficaz [36].

Estos resultados servirán como referencia para los módulos de la aplicación web.

La empresa 1 es BIODIMED, encargada de medicina integral y ocupacional, la actividad principal del *callcenter* es la toma de citas médicas, venta de planes de salud, entre otras. Una de las principales necesidades del *callcenter* de la empresa 1 es emitir recordatorios de las citas médicas de los pacientes, mediante un sistema automático de llamadas y mensajes de texto. Ya que realizar este trabajo de manera manual, es decir con personal del *callcenter*, es muy demoroso e ineficiente muchas de las veces debido al volumen de citas por confirmar y el número excedente de clientes diarios. El *callcenter* de la empresa 1 se lo denomina como un *callcenter* pequeño.

La empresa 2 es MEMORIAL INTERNATIONAL encargada de asistencia exequial, emocional y funeraria. La actividad principal del *callcenter* es vender planes de servicios funerarios individuales, familiares, corporativos, pymes y empresas públicas. La necesidad de este *callcenter* es promocionar de manera masiva los planes y promociones que la empresa 2 ofrece a sus clientes, informar de manera personalizada el monto de las cuotas adeudadas o simplemente enviar notificaciones de texto con dichos costos. El *callcenter* de la empresa 2 se lo denomina como un *callcenter* mediano.

La empresa 3 es la COOPERATIVA DE AHORRO Y CRÉDITO 29 DE OCTUBRE, encargada de intermediación financiera a nivel nacional. Entre las actividades del *callcenter* se tiene el proceso de cobranzas y recordatorios de pagos de cuotas o créditos obtenidos a la cooperativa. Una gran necesidad del *callcenter* de la empresa 3 es generar llamadas personalizadas a sus clientes para informarles que sus créditos o cuotas están vencidas, están por vencerse o ya su préstamo entran a la central de riesgo. El *callcenter* de la empresa 3 se lo denomina como un *callcenter* grande.

El *callcenter* denominado “pequeño” está conformado por 3 agentes y un supervisor. El *callcenter* denominado “mediano” está conformado por 8 agentes

y 1 supervisor. El *callcenter* denominado “grande” está conformado por 16 agentes y 2 supervisores. En un *callcenter* los agentes son las personas encargadas de realizar llamadas telefónicas mediante campañas (detalladas en el **capítulo 1 sección 1.1.7.1**) dependiendo de la actividad de la empresa.

Una campaña se basa en un conjunto de contactos con un grupo de personas, generalmente clientes o futuros clientes, con el fin de promover un producto, ofrecer promociones, dar seguimiento, etc. Se enfoca en el contacto mediante llamadas telefónicas o mensajes SMS. Las llamadas pueden ser tanto celulares como convencionales (números locales y nacionales).

Las encuestas individuales de los requerimientos del prototipo realizadas a las empresas mencionadas anteriormente se presentan en el Anexo A.

2.1.2 HISTORIAS DE USUARIO

Las historias de usuario son una lista de las tareas a considerar en el desarrollo del prototipo. Las historias de usuario se realizaron a los Administradores de los *callcenters* mencionados anteriormente. Un administrador de *callcenter* es la persona encargada de supervisar las labores de los agentes o asesores y gestionar la información de los clientes finales. A continuación, se presenta el formato de una historia de usuario, a manera de tabla, la cual contiene los campos: autor, usuario, título, descripción breve de la tarea a realizar, fecha en la cual se creó la historia de usuario, observación de posibles cambios o integraciones futuras y un campo código que es el identificador único de la historia de usuario.

2.1.3 ITERACIONES DEL PROYECTO

2.1.3.1 PRIMERA ITERACIÓN

Se realiza una entrevista al administrador del *callcenter* como primera iteración, en la cual se logra recopilar información sobre las necesidades y requisitos que debería presentar la aplicación. Estos requerimientos permiten la elaboración de las historias de usuario presentadas desde la **Tabla 2.2** hasta la **Tabla 2.8**.

Las historias de usuario se representan en las tablas siguientes. Estas historias de usuario son las que describen las distintas funcionalidades que el administrador del prototipo desea del sistema.

Las **Tablas 2.2, 2.3 y 2.4** permiten la creación de las campañas estáticas (campañas con audios previamente grabados), campañas personalizadas (campañas con audios construidos con un tts²⁷) y campañas de mensajes de texto (campaña de mensajes de texto SMS).

Tabla 2.2 Historia de usuario HU1-01_Crear Campaña Estática.

Autor: Evelyn Yessenia Achig Ruano	Usuario: Administrador sistema
Código: HU1-01	Fecha: 2016-12-10
Título: Crear Campaña Estática	
Descripción: El sistema debe permitir crear una campaña dirigida a un conjunto de personas a las que se va a comunicar un mensaje pregrabado. Para ello debe permitir ingresar los datos de la campaña como nombre de la campaña, canales (número de llamadas simultáneas), troncal, descripción.	
Observaciones: Cada campaña tiene asociado un mensaje de audio en común que se reproducirá a los contactos asociados.	

Tabla 2.3 Historia de usuario HU1-02_Crear Campaña Personalizada.

Autor: Evelyn Yessenia Achig Ruano	Usuario: Administrador sistema
Código: HU1-02	Fecha: 2016-12-10
Título: Crear Campaña Personalizada	
Descripción: El sistema debe permitir crear una campaña dirigida a un conjunto de personas a las que se va a comunicar un mensaje construido en tiempo real. Para ello debe permitir ingresar los datos de la campaña como nombre de la campaña, canales (número de llamadas simultáneas), troncal, descripción.	
Observaciones: Cada campaña personalizada tiene asociados sus contactos a llamar y el mensaje personalizado a reproducir.	

La **Tabla 2.5** representa la actividad que permite listar todas las campañas estáticas, personalizadas y de mensajes de texto. La **Tabla 2.6** muestra la actividad borrar una campaña seleccionada.

²⁷ TTS: *Text to speech* o conversor de texto a voz.

Las **Tablas 2.7 y 2.8** describen actividades para obtener un reporte estadístico de las campañas.

Tabla 2.4 Historia de usuario HU1-03_Crear Campaña Mensajes de texto.

Autor: Evelyn Yessenia Achig Ruano	Usuario: Administrador sistema
Código: HU1-03	Fecha: 2016-12-10
Título: Crear Campaña Mensajes de texto	
Descripción: El sistema debe permitir crear una campaña dirigida a un conjunto de personas a las que se va a comunicar un mensaje construido en tiempo real. Para ello debe permitir ingresar los datos de la campaña como nombre de la campaña, canales (número de llamadas simultáneas), troncal, usuario, contraseña, descripción.	
Observaciones: Cada campaña tiene asociados sus contactos a los cuales se les enviará el SMS.	

Tabla 2.5 Historia de usuario HU1-04_Lista de Campañas General.

Autor: Evelyn Yessenia Achig Ruano	Usuario: Administrador sistema
Código: HU1-04	Fecha: 2016-12-10
Título: Lista de Campañas Estáticas, Personalizadas y Mensajes de Texto	
Descripción: El detalle de la campaña debe permitir listar todas las campañas existentes en una lista según el tipo de campaña (estática, dinámica o de mensajes de texto), así como varias características como su fecha de creación, estado de la campaña, troncal, descripción y estadísticas.	
Observaciones: Verificar que permita seleccionar el archivo de texto.	

Tabla 2.6 Historia de usuario HU1-05_Borrar Campañas.

Autor: Evelyn Yessenia Achig Ruano	Usuario: Administrador sistema
Código: HU1-05	Fecha: 2016-12-10
Título: Borrar Campaña Estática, Personalizada y Mensajes de Texto	
Descripción: El sistema debe permitir borrar las campañas (estática, personalizada y mensajes de texto) creadas anteriormente por ejemplo, si se creó mal la campaña.	
Observaciones: Validar que la campaña se ha borrado correctamente	

Tabla 2.7 Historia de usuario HU1-06_Reporte de Campañas Estáticas y Personalizadas.

Autor: Evelyn Yessenia Achig Ruano	Usuario: Administrador sistema
Código: HU1-06	Fecha: 2016-12-10
Título: Reporte de Campañas Estáticas y Personalizadas	
Descripción: El sistema debe permitir consultar cuántas llamadas totales se realizaron, estados de la llamada y cantidad, como un resumen de que ocurrió en las campañas estáticas y personalizadas.	
Observaciones: Se verificará que el reporte tenga el número llamadas totales que se realizaron, el número de llamadas contestadas, no contestadas, fallidas, y ocupadas.	

Tabla 2.8 Historia de usuario HU1-07_Reporte de Campañas de Mensajes de texto.

Autor: Evelyn Yessenia Achig Ruano	Usuario: Administrador sistema
Código: HU1-07	Fecha: 2016-12-10
Título: Reporte de Campañas de Mensajes de Texto	
Descripción: El sistema debe permitir consultar cuantos contactos con sus respectivos mensajes se ingresaron al sistema, el estado del mensaje, así como la cantidad de mensajes enviados en forma de un resumen del estado de las campañas de mensajes de texto.	
Observaciones: Se verificará que el reporte tenga lo solicitado en la descripción y que se puede mejorar.	

2.1.4 SEGUNDA ITERACIÓN

Se presenta una primera versión de la aplicación a 3 administradores de *callcenters* mencionados en la **sección 2.1.1**, para una evaluación general del aplicativo. Los mismos muestran varios criterios que se deberían implementar en el prototipo, así como cambios en algunos aspectos gráficos. En base a estos criterios se generan las siguientes historias de usuario:

Tabla 2.9 Historia de usuario HU2-01_Crear Campaña Estática número de intentos.

Autor: Evelyn Yessenia Achig Ruano	Usuario: Administrador1
Código: HU2-01	Fecha: 2017-08-16
Título: Crear Campaña Estática con número de intentos	
Descripción: El sistema debe dejar ingresar los datos de la campaña como nombre de la campaña, canales (número de llamadas simultáneas), troncal, descripción, número de intentos (número máximo de reintentos incluyendo el intento inicial que por defecto el valor es 0) y tiempo de espera (cuántos segundos esperar antes de volver a intentar. El valor predeterminado es de 300 segundos) entre estos intentos.	
Observaciones: Cada campaña volverá a intentar llamar dependiendo del estado de contestación de la llamada.	

Tabla 2.10 Historia de usuario HU2-02_Crear Campaña Personalizada número de intentos.

Autor: Evelyn Yessenia Achig Ruano	Usuario: Administrador 2
Código: HU2-02	Fecha: 2017-08-16
Título: Crear Campaña Personalizada número de intentos.	
Descripción: El sistema debe permitir el ingreso de los datos de la campaña como nombre de la campaña, canales (número de llamadas simultáneas), troncal, descripción, número de intentos (número máximo de reintentos incluyendo el intento inicial que por defecto el valor es 0) y tiempo de espera (cuántos segundos esperar antes de volver a intentar. El valor predeterminado es 300 segundos) entre estos intentos.	
Observaciones: Cada campaña volverá a intentar llamar dependiendo del estado de contestación de la llamada.	

Tabla 2.11 Historia de usuario HU2-03_Eventos Compactos en Campañas.

Autor: Evelyn Yessenia Achig Ruano	Usuario: Administrador 3
Código: HU2-03	Fecha: 2017-09-10
Título: Eventos Compactos en Campañas	
Descripción: El sistema debe permitir iniciar, detener y continuar la campaña. Se muestra cada evento en un botón. Se requiere consolidar estos eventos para que sea más estético.	
Observaciones: Se agrupará los eventos de la campaña para un mejor manejo de los mismos.	

Tabla 2.12 Historia de usuario HU2-04_Subir Archivos en Campaña Estática.

Autor: Evelyn Yessenia Achig Ruano	Usuario : Administrador 1
Código: HU2-04	Fecha: 2017-09-10
Título: Subir Archivos en Campaña Estática.	
Descripción: El sistema debe permitir subir tanto archivos de audio (formato .wav) como archivos de texto (formato .csv delimitado por comas) para reproducir durante las campañas estáticas. Los archivos de texto deben tener solo una columna con los números de teléfono a llamar.	
Observaciones: Se agrupará los archivos a ingresar a la campaña para una mejor presentación.	

Tabla 2.13 Historia de usuario HU2-05_Subir Archivos en Campaña Personalizada.

Autor: Evelyn Yessenia Achig Ruano	Usuario : Administrador 3
Código: HU2-05	Fecha: 2017-09-10
Título: Subir Archivos en Campaña Personalizada	
Descripción: El sistema debe permitir subir archivos de texto (formato .csv delimitado por comas) para las campañas personalizadas. Los archivos deben tener dos columnas, la primera debe contener el número de teléfono a llamar y la segunda el mensaje personalizado a reproducir.	
Observaciones: Se subirá el archivo con los mensajes personalizados a ser reproducidos mediante llamadas telefónicas.	

Tabla 2.14 Historia de usuario HU2-06_Subir Archivos en Campaña Mensaje de texto.

Autor: Evelyn Yessenia Achig Ruano	Usuario : Administrador 3
Código: HU2-06	Fecha: 2017-09-10
Título: Subir Archivos en Campaña de Mensaje de texto	
Descripción: El sistema debe permitir subir archivos de texto (formato .csv delimitado por comas) para las campañas personalizadas. Los archivos deben tener dos columnas, la primera debe contener el número de teléfono celular al cual se enviará el mensaje y la segunda el mensaje personalizado.	
Observaciones: Los mensajes de texto corto SMS tienen un límite de caracteres. Pueden tener 160 caracteres contando con los espacios, sin tildes, sin signos de puntuación, sin caracteres especiales. O pueden tener 140 caracteres contando con espacios signos de puntuación, tildes, y caracteres especiales.	

2.1.5 TERCERA ITERACIÓN

Después de presentar una segunda versión con los cambios considerados en la segunda iteración, los administradores sugieren realizar cambios en el módulo de reportería como;

- Presentar un diagrama gráfico en porcentajes para presentar las estadísticas de las campañas por estado de contestación.
- Corregir el formulario de ingreso para la autenticación.
- Estandarizar los encabezados de los módulos del sistema en todas las posibles ventanas que se acceda.

De estas sugerencias se crean las siguientes historias de usuario:

Tabla 2.15 Historia de usuario HU3-01_Reporte Gráfico de Campañas Estáticas.

Autor: Evelyn Yessenia Achig Ruano	Usuario: Administrador sistema
Código: HU3-01	Fecha: 2017-06-10
Título: Reporte Gráfico de Campañas Estáticas.	
Descripción: El sistema debe permitir visualizar un gráfico del estado de contestación de las llamadas por campaña estática.	
Observaciones: El reporte gráfico presentará un diagrama de barras que describa el estado de las llamadas realizadas para cada campaña, indicando el porcentaje de llamadas que han alcanzado los distintos estados posibles: contestada, no contestada, fallida, ocupada.	

Tabla 2.16 Historia de usuario HU3-02_Reporte Gráfico de Campañas Personalizadas.

Autor: Evelyn Yessenia Achig Ruano	Usuario: Administrador sistema
Código: HU3-02	Fecha: 2017-06-10
Título: Reporte Gráfico de Campañas Personalizadas	
Descripción: El sistema debe permitir visualizar un gráfico del estado de contestación de las llamadas por campaña personalizada.	
Observaciones: El reporte gráfico presentará un diagrama de barras que describa el estado de las llamadas realizadas para cada campaña, indicando el porcentaje de llamadas que han alcanzado los distintos estados posibles: contestada, no contestada, fallida, ocupada.	

Tabla 2.17 Historia de usuario HU3-03_Reporte Gráfico de Campañas de Mensajes de texto.

Autor: Evelyn Yessenia Achig Ruano	Usuario: Administrador sistema
Código: HU3-03	Fecha: 2017-06-10
Título: Reporte Gráfico de Campañas de Mensajes de Texto	
Descripción: El sistema debe permitir visualizar un gráfico del estado de envío de los mensajes de texto por campaña.	
Observaciones: El reporte gráfico presentará un diagrama de barras que describa el estado de los mensajes de texto enviados para cada campaña, indicando el porcentaje de mensajes de texto que han alcanzado los distintos estados posibles: exitoso, no exitoso, enviando, fallido.	

2.1.6 RESUMEN DE REQUERIMIENTOS DEL PROTOTIPO

Mediante la información recopilada en las **secciones 2.1.3, 2.1.4 y 2.1.5** se pueden determinar los requerimientos del prototipo, los mismos que se muestran en la **Tabla 2.18**.

Tabla 2.18 Resumen de los Requerimientos del Prototipo.

REQUERIMIENTOS	TIPO DE REQUERIMIENTO	HISTORIA DE USUARIO	ITERACIÓN
Creación de campañas estáticas.	Funcional	HU1-01	PRIMERA
Creación de campañas personalizadas.	Funcional	HU1-02	
Creación de campañas de mensajes de texto.	Funcional	HU1-03	
Listar todas las campañas existentes según el tipo de campaña (estática, dinámica o de mensajes de texto.	Funcional	HU1-04	
Eliminación de campañas tanto estáticas, personalizadas y mensajes de texto SMS.	Funcional	HU1-05	
Reporte del número de llamadas totales que se realizaron.	Funcional	HU1-06	
Reporte de Campañas de Mensajes de Texto	Funcional	HU1-07	
Crear Campaña Estática con número de intentos	Funcional	HU2-01	SEGUNDA
Crear Campaña Personalizada número de intentos.	Funcional	HU2-02	
Eventos Compactos en Campañas	No Funcional	HU2-03	
Subir Archivos en Campaña Estática.	Funcional	HU2-04	
Borrar Campaña Estática, Personalizada y Mensajes de Texto	Funcional	HU2-05	
Subir Archivos en Campaña de Mensaje de texto	Funcional	HU2-06	
Reporte Gráfico de Campañas Estáticas	No Funcional	HU3.01	TERCERA
Reporte Gráfico de Campañas Personalizadas	No Funcional	HU3-02	
Reporte Gráfico de Campañas de Mensajes de Texto	No Funcional	HU3-03	

2.2 DISEÑO DEL SISTEMA AUTOMATIC SEND&DIALING (ASD)

En esta sección se presentan algunos diagramas que forman parte de la etapa de diseño del prototipo. A continuación, se presentan los diagramas entidad-relación y de actividades utilizados para el prototipo.

2.2.1 NIVEL DE DATOS

2.2.2 MODELO ENTIDAD-RELACIÓN

Las **Figuras 2.1** y **2.2** muestran el modelo Entidad-Relación de la base de datos del prototipo. Este modelo consta de ocho entidades principales las cuales son: Campañas_automáticas, Campañas_sms, Contactos_pregrabados, Contactos_construidos, Contactos_sms, Audio_pregrabado, Ast_matic_cdr y Sms_mdr, cada uno muestra sus respectivos atributos y relaciones.

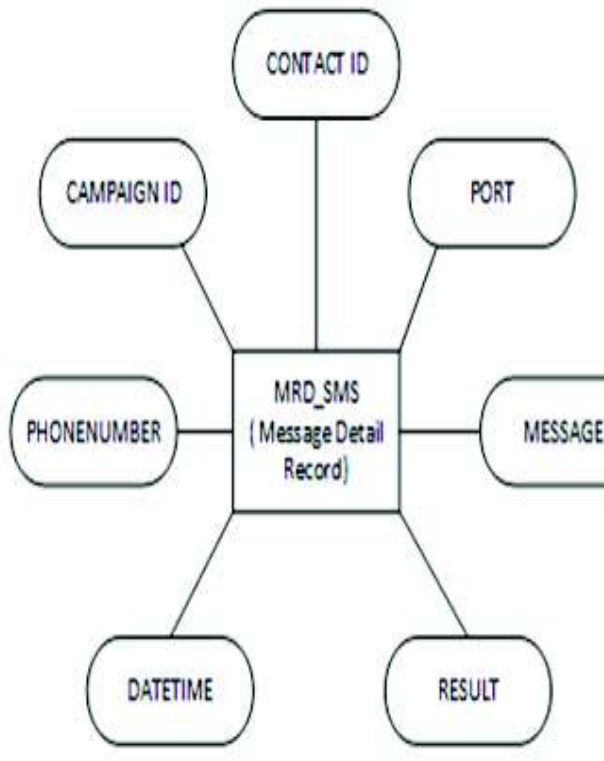
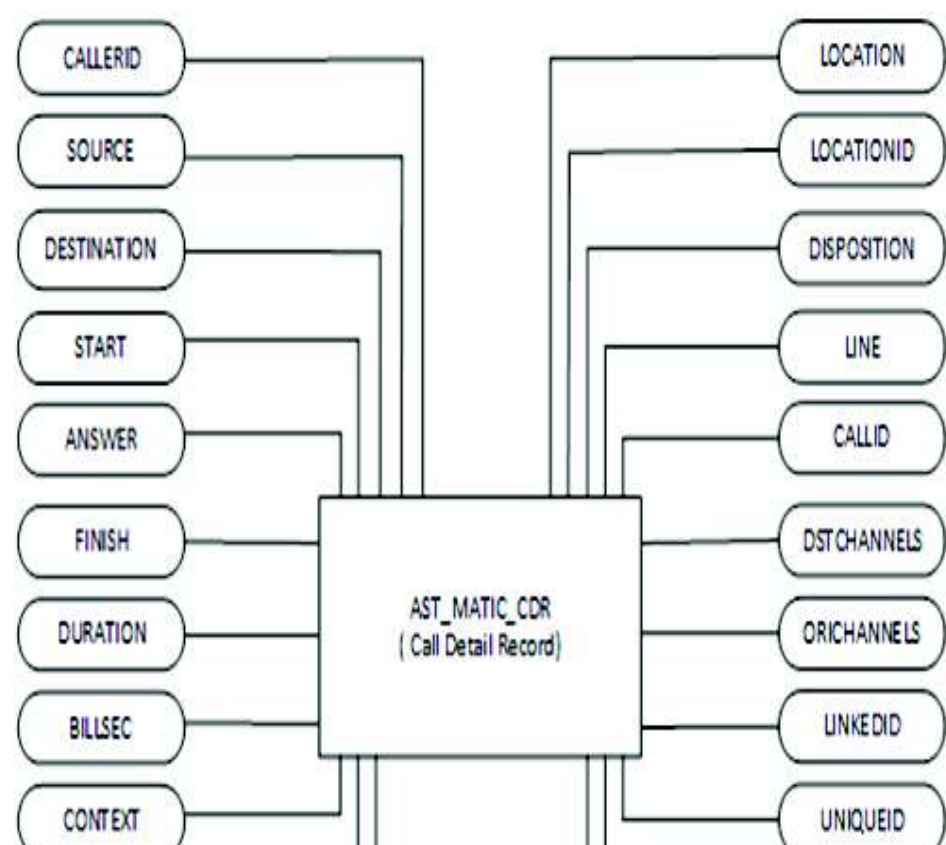
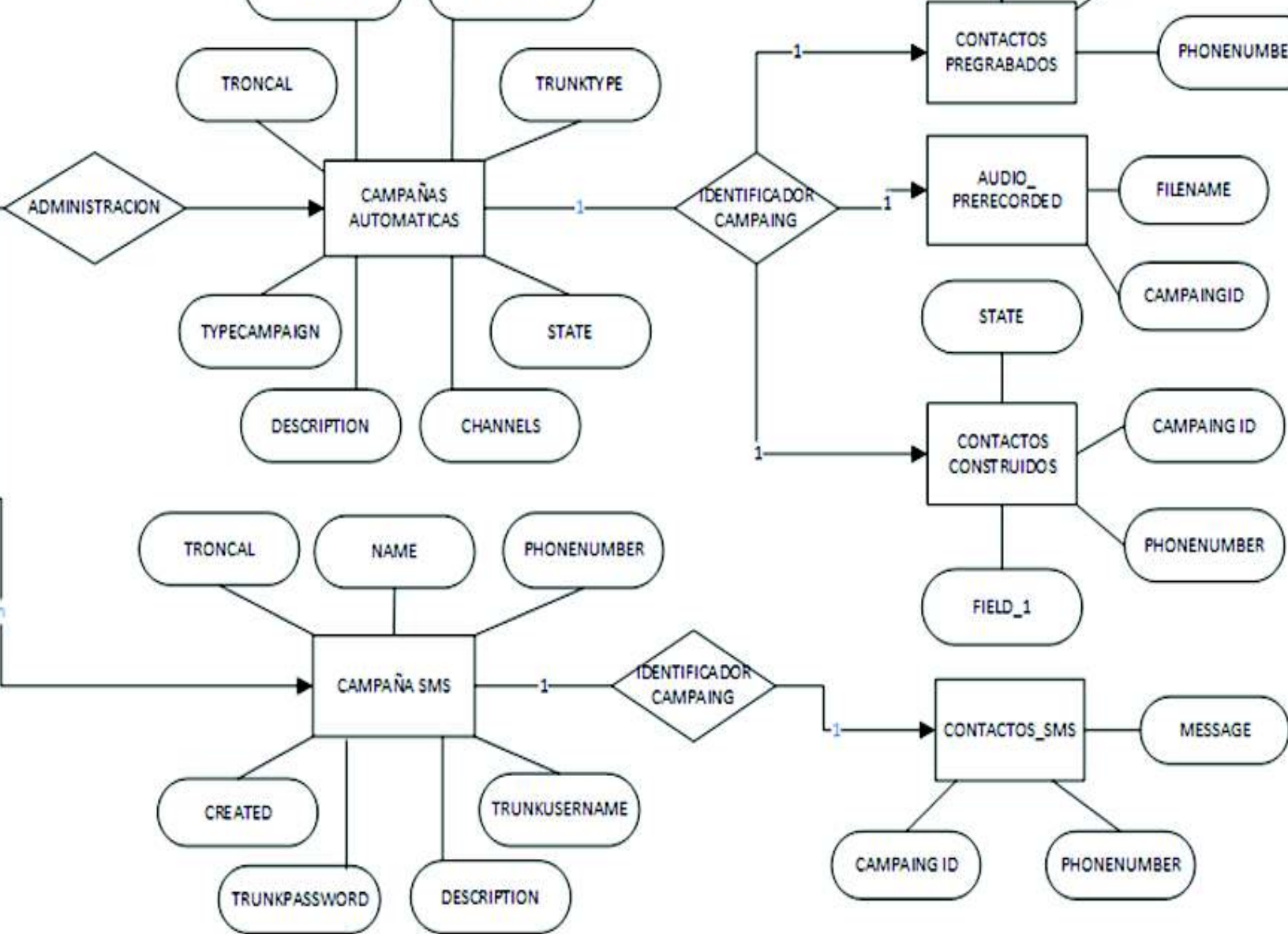
2.2.2.1 Diagrama Relacional

El diagrama relacional fue generado en base al modelo Entidad-Relación presentado en la **Figura 2.1**. Mediante el diagrama relacional se obtiene ocho tablas con sus respectivos atributos. En estas tablas se almacena información de las diferentes campañas y contactos, así como también el detalle de los registros de las llamadas (CDR²⁸) y de los mensajes de texto (MDR²⁹).

Las tablas del diagrama relacional que se utilizó en el prototipo son: automatic_campaigns, sms_campaigns, contacts_prerecorded, contacts_buit, sms_contacts, audio_prerecord, ast_matic_cdr y sms_mdr.

²⁸ CDR: *Call Detail Records* o detalle de los registros de las llamadas.

²⁹ MDR: *Message Detail Records* o detalle de los registros de los mensajes de texto.



La **Figura 2.2** presenta el diagrama relacional con las respectivas tablas, atributos, relaciones, claves primarias³⁰ y claves foráneas³¹. Las claves primarias se representan con la imagen de una llave de color amarillo y las claves foráneas se representan con el nombre de la tabla que hace referencia precedido de un guión bajo. (Ejemplo: _campaign).

2.2.3 DIAGRAMA DE CLASES

La **Figura 2.3** muestra el diagrama de clases correspondiente al prototipo con sus respectivos atributos y métodos.

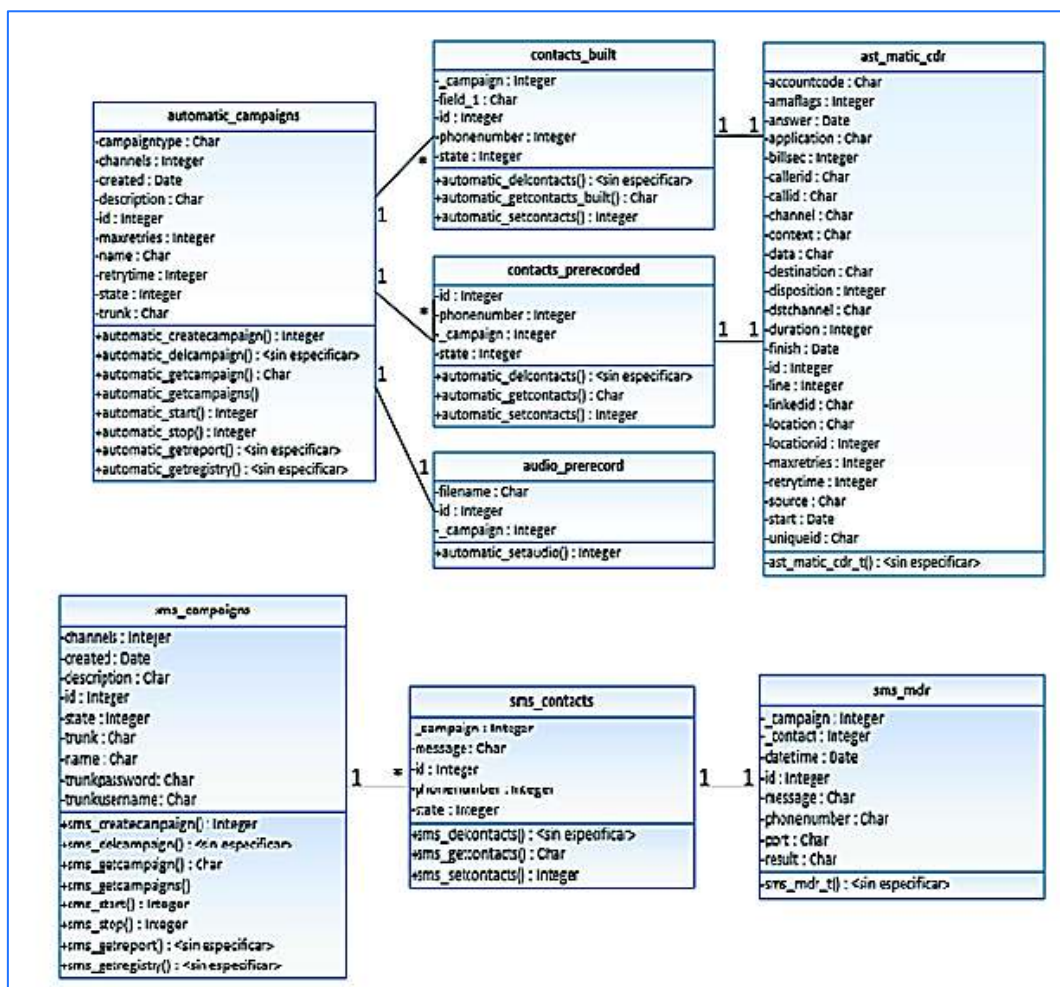


Figura 2.3 Diagrama de clases.

³⁰ Clave primaria: Es un campo único de tipo enteró que identifica a un registro de la tabla

³¹ Clave foránea: Se utilizan para unir o relacionar tablas. Es una columna que contiene la clave primaria de otra tabla.

2.3 NIVEL DE LÓGICA DE NEGOCIO

Se encuentra en medio del nivel de datos y del nivel de presentación, es donde se encuentran los programas que serán ejecutados. En este nivel se recibe las peticiones del usuario y se envían las respuestas después de los procesos solicitados.

2.3.1 DIAGRAMA DE DESPLIEGUE

El prototipo está formado por una aplicación web que permite al usuario gestionar los servicios de la telefonía. En la aplicación web del prototipo se utiliza el patrón de diseño Modelo Vista Controlador debido que es muy utilizado en el desarrollo de este tipo de aplicaciones. La **Figura 2.4** presenta la interacción del usuario con el sistema mediante los respectivos bloques Modelo Vista Controlador.

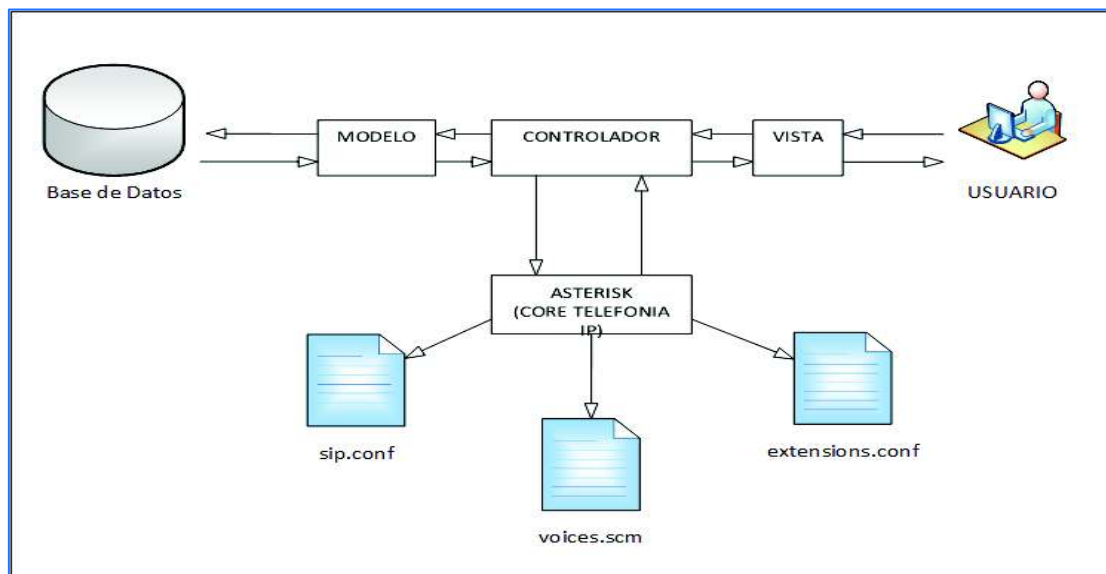


Figura 2.4 Diagrama de Despliegue.

El bloque que interactúa directamente con el usuario es el bloque Vista. Este bloque se lo representa como una página web la cual se puede abrir desde cualquier navegador. Es un archivo de tipo html.

El bloque Controlador es el que interactúa tanto con la base de datos, la Vista y sobre todo con la Central Telefónica Asterisk mediante sus archivos de configuración, específicamente interactúa con los archivos **sip.conf**, **extensions.conf**, y **voices.scm**.

El bloque Modelo es el que gestiona las solicitudes y respuestas a la base de datos de acuerdo a lo que el bloque Controlador le indique. El Modelo permite el acceso a la base de datos y a la información requerida.

2.3.2 DIAGRAMA DE ACTIVIDADES

Las **Figuras 2.5, 2.6, 2.7, 2.8, 2.9, 2.10, 2.11, 2.12 y 2.13** muestran los diagramas que describen el flujo de actividades derivadas de las interacciones del usuario con el prototipo en diferentes escenarios.



Figura 2.5 Diagrama de Actividades_Ingreso.

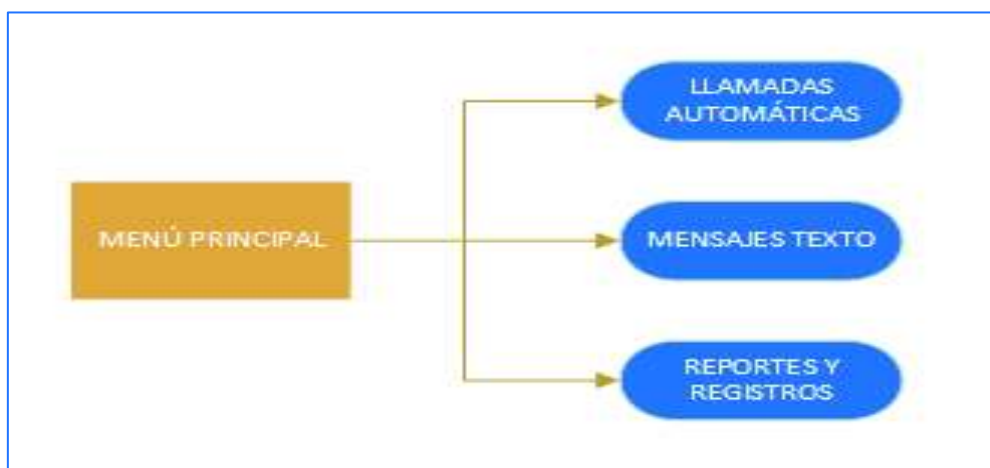


Figura 2.6 Diagrama de Actividades_Menú principal.

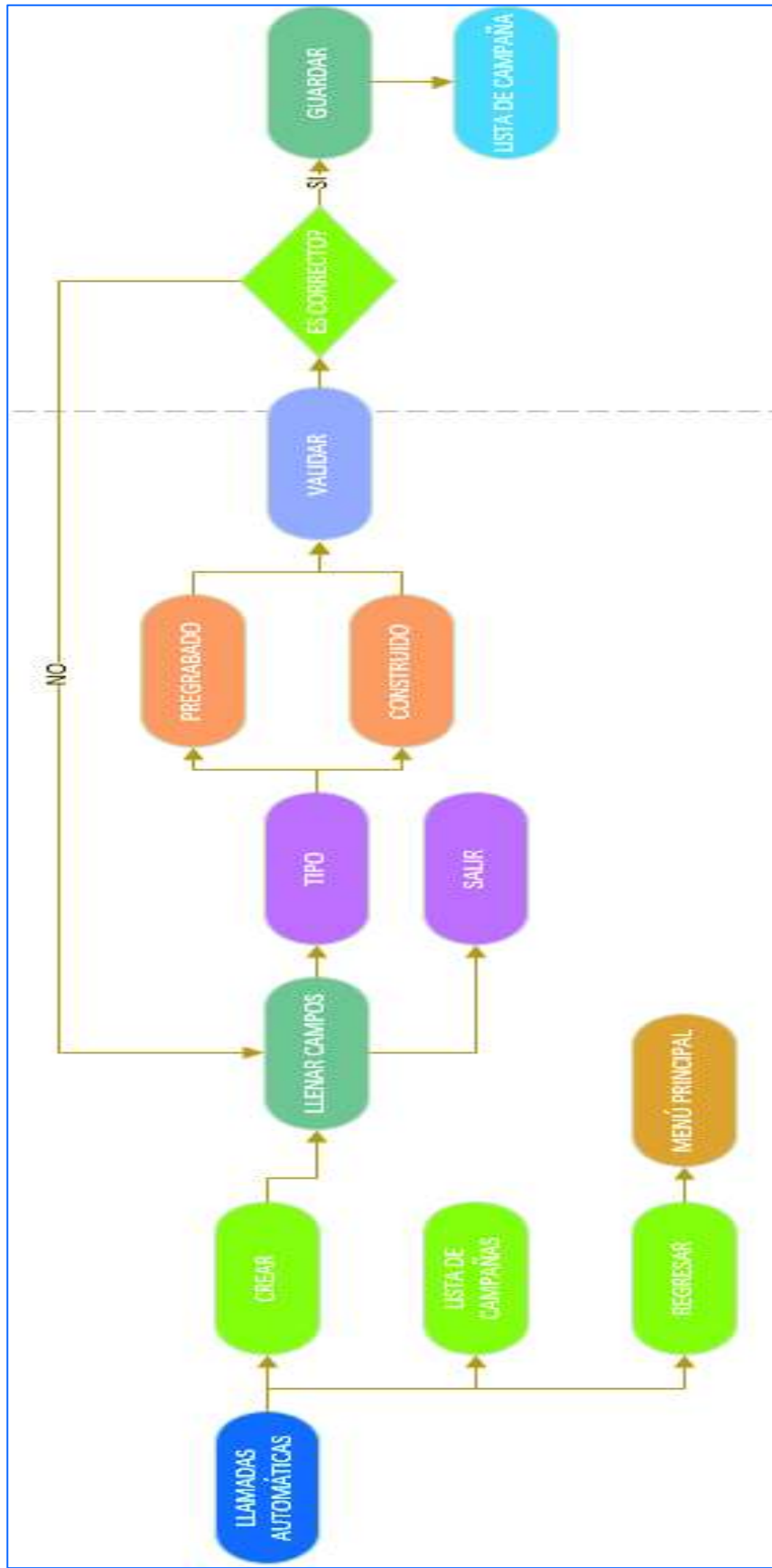


Figura 2.7 Diagrama de Actividades_ Creación de una campaña.

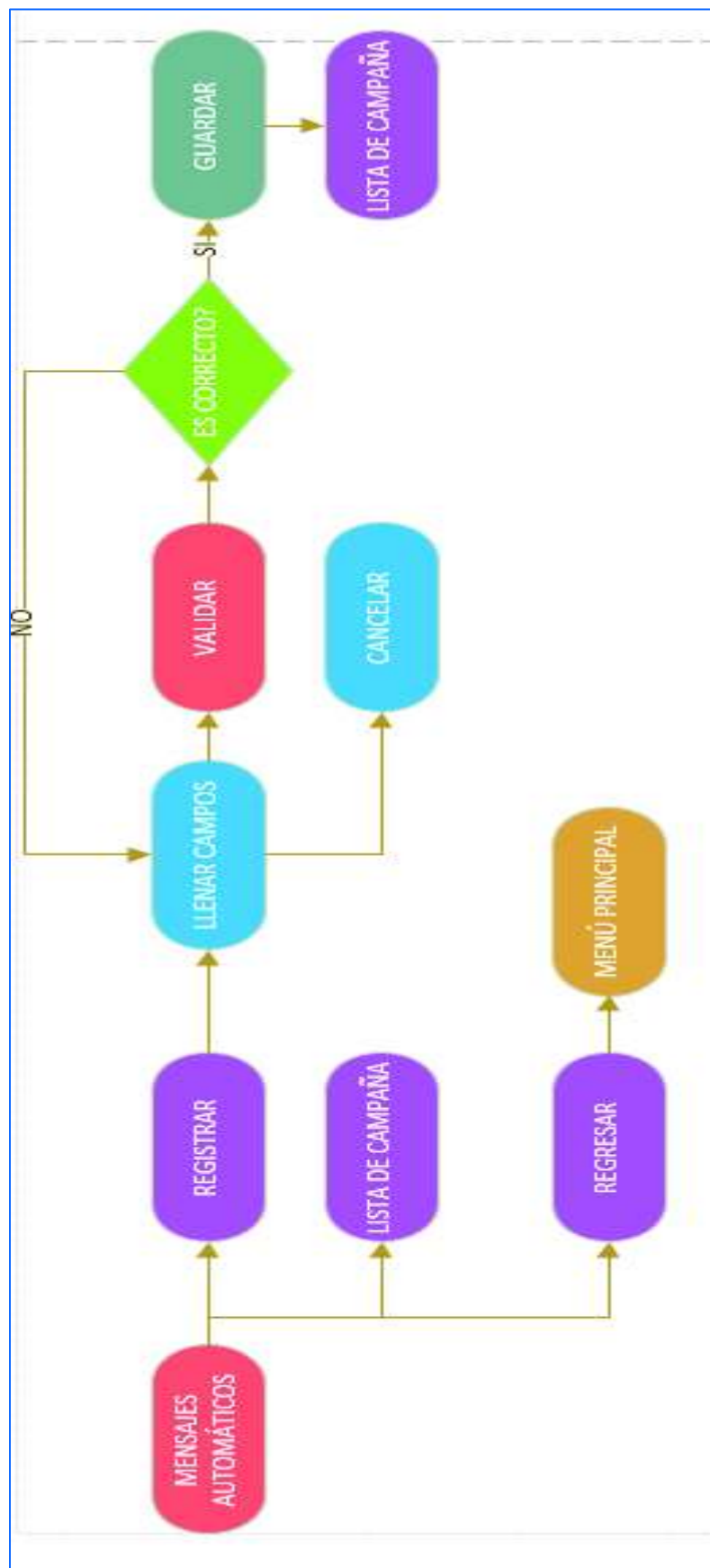


Figura 2.8 Diagrama de Actividades_Crear campaña Mensajes de texto.

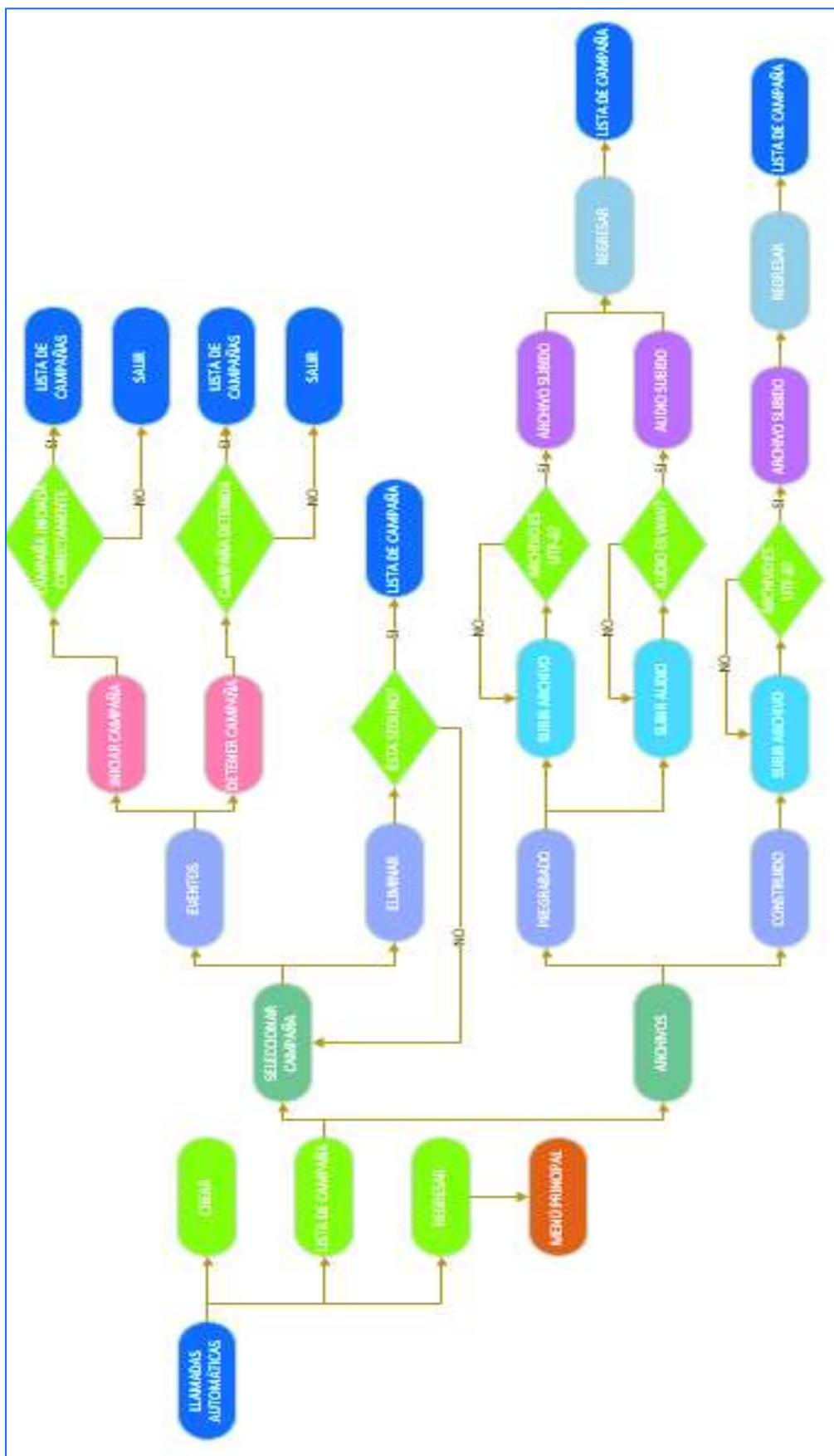


Figura 2.9 Diagrama de Actividades_ Llamadas Automáticas.

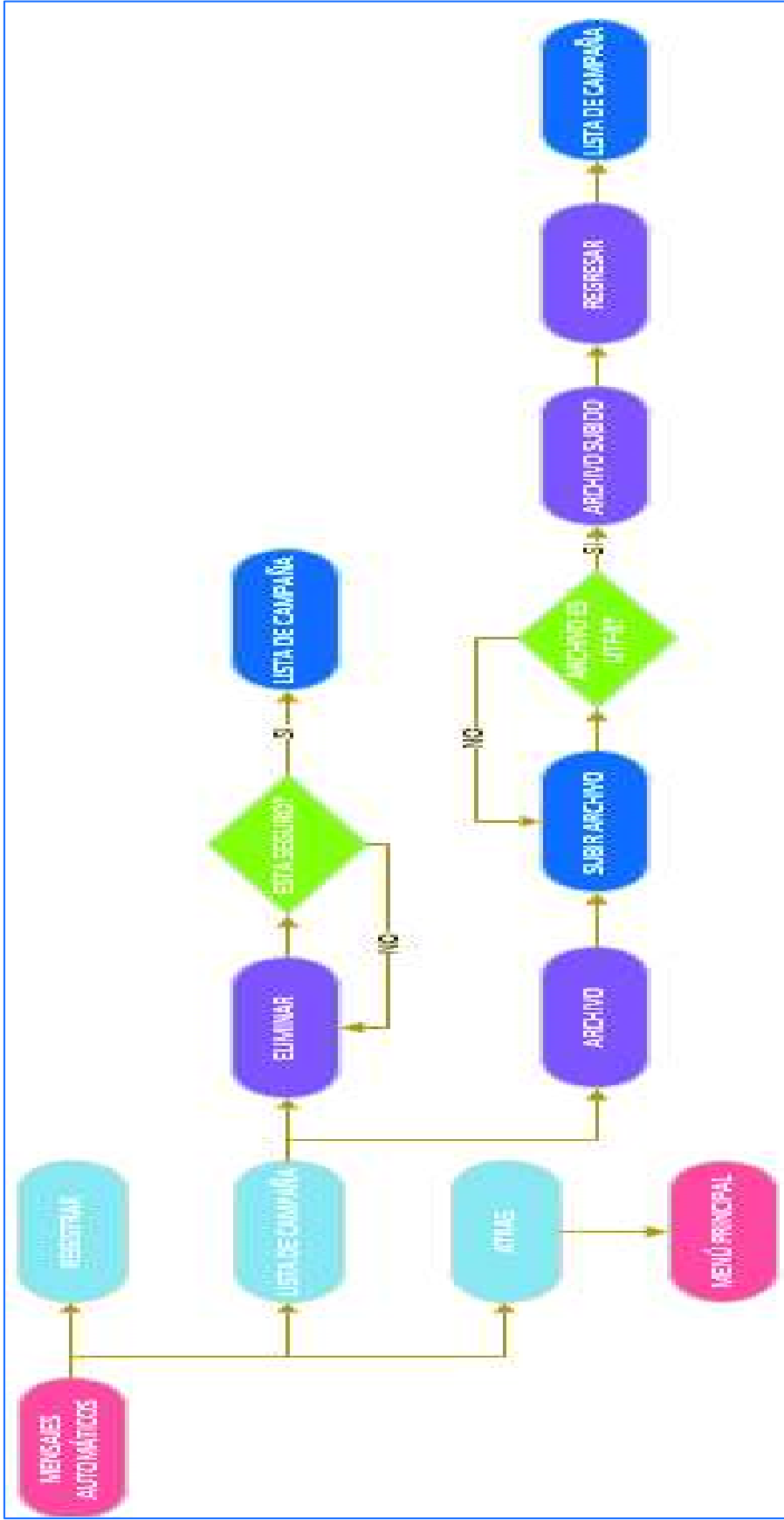


Figura 2.10 Diagrama de Actividades_Campaña de Mensajes de Texto.

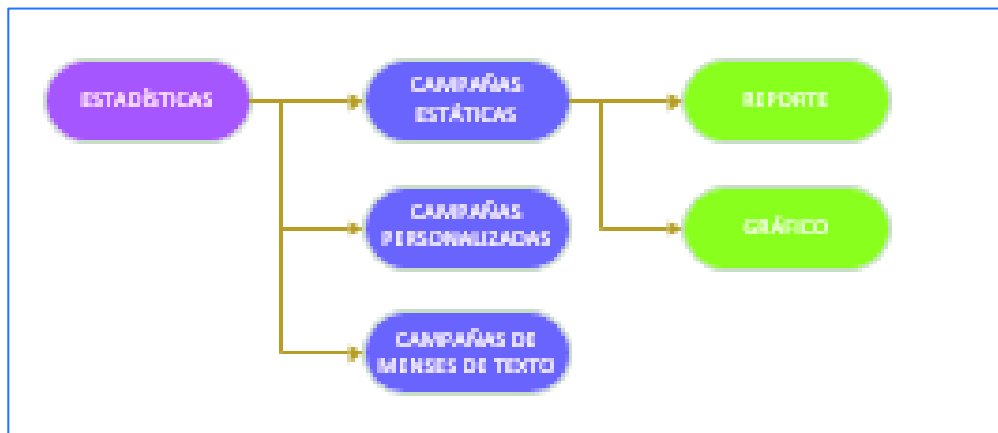


Figura 2.11 Diagrama de Actividades_Estadísticas_Reportes de campaña Estática.



Figura 2.12 Diagrama de Actividades_Estadísticas_Reportes Campaña Personalizada.



Figura 2.13 Diagrama de Actividades_Estadísticas_Reportes Campaña Mensajes de texto.

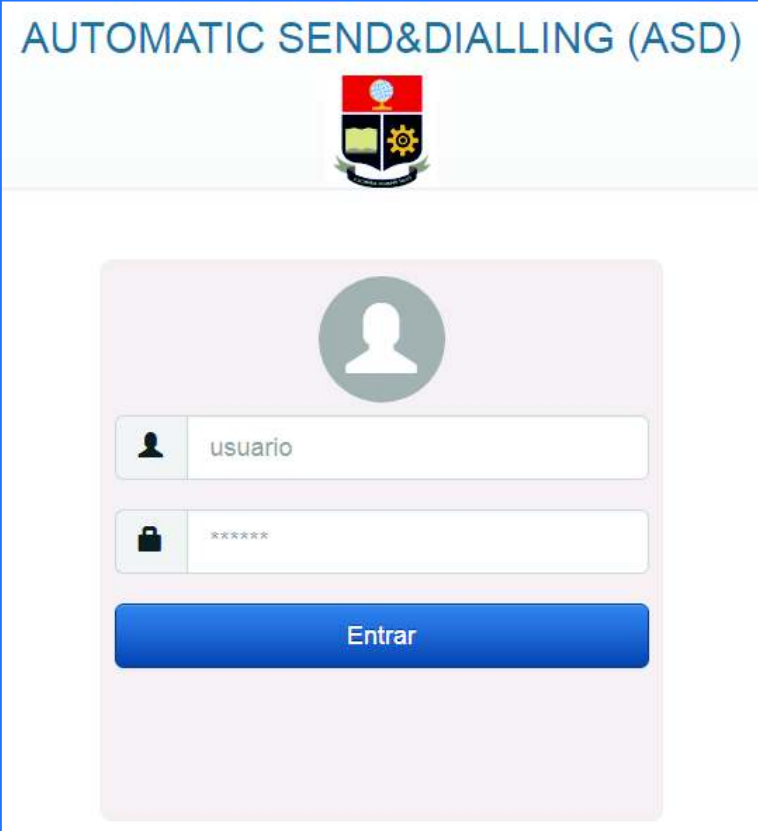
2.4 NIVEL PRESENTACIÓN

El nivel de presentación permite la interacción dinámica del usuario con el sistema. Antes de llegar a la presentación del diseño final se realizaron varias iteraciones de historias de usuario que se presentan desde la **sección 2.1.2** hasta la **sección 2.1.5**.

El diseño del prototipo es intuitivo para el usuario en base en los requisitos analizados previamente en el proyecto.

2.5 INTERFACES DEL SISTEMA

La **Figura 2.14** presenta la interfaz de ingreso al sistema, en este proyecto no se propone la creación, edición, eliminación y administración de usuarios por lo que el proyecto solo consta de un usuario creado directamente en la base de datos. El formulario de ingreso, valida el nombre del usuario y la contraseña creados previamente para direccionar hacia el menú del proyecto.



The image shows a login interface for a system titled "AUTOMATIC SEND&DIALLING (ASD)". At the top, there is a logo featuring a shield with a gear and a book. Below the logo is a light purple rounded rectangle containing a user profile icon. Underneath the icon are two input fields: the first is labeled "usuario" and the second is masked with "*****". A blue button labeled "Entrar" is positioned below the password field.

Figura 2.14 Intefaz_Ingreso.

La **Figura 2.15** presenta la interfaz del Menú Principal del proyecto. Aquí se presenta las diferentes funcionalidades del sistema las cuales son: llamadas automáticas y envío masivo de mensajes de texto.



Figura 2.15 Interfaz_Menú_Principal.

La **Figura 2.16** presenta la interfaz de las llamadas automáticas con sus diferentes tipos de campañas de llamadas (estáticas y personalizadas) y una pestaña para creación de campañas de llamadas automáticas.



Figura 2.16 Interfaz_Llamadas Automáticas.

La **Figura 2.17** presenta la interfaz de la lista las campañas de audio estático existentes en el sistema. Las llamadas automáticas con audio estático

reproducen un audio común para todos los números de teléfonos ingresados al sistema mediante un archivo de texto de tipo csv y codificación utf8.



The screenshot shows the 'AUTOMATIC SEND&DIALLING (ASD)' interface. At the top, there is a logo and navigation links: 'WebSiteEvelyn', 'LLAMADAS AUTOMÁTICAS', 'MENSAJES DE TEXTO (SMS)', and 'REPORTES Y REGISTROS'. Below this is the title 'LISTA DE CAMPAÑAS DE AUDIO ESTÁTICO'. The main content is a table with the following data:

NOMBRE DE LA CAMPAÑA	CANAL	FECHA DE CREACIÓN	DESCRIPCIÓN	ESTADO	SUBIR ARCHIVOS	EVENTOS	ESTADÍSTICAS
lunes prueba 2	2	2018-03-12 12:12:30	12346678	NO INICIADA	ARCHIVOS -	EVENTOS -	ESTADÍSTICAS
campana estatica	1	2018-03-12 05:44:51	qwewe	FINALIZADA	ARCHIVOS -	EVENTOS -	ESTADÍSTICAS
prueba noche lunes	1	2018-03-11 22:15:27	qwerty	NO INICIADA	ARCHIVOS -	EVENTOS -	ESTADÍSTICAS
prueba noche jueves	2	2018-03-09 23:10:45	prueba de noche	NO INICIADA	ARCHIVOS -	EVENTOS -	ESTADÍSTICAS

Figura 2.17 Interfaz_Lista Campañas LLlamadas Estáticas.

La **Figura 2.18** presenta la interfaz de la lista las campañas de audio personalizado existentes en el sistema. Las llamadas automáticas con audio personalizado reproducen el texto ingresado al sistema como un audio mediante la aplicación tts (*text to speech*) de asterisk al número de teléfono asociado inicialmente en el archivo de texto, el cual debe ser de tipo csv y codificación utf8.



The screenshot shows the 'AUTOMATIC SEND&DIALLING (ASD)' interface. At the top, there is a logo and navigation links: 'WebSiteEvelyn', 'LLAMADAS AUTOMÁTICAS', 'MENSAJES DE TEXTO (SMS)', and 'REPORTES Y REGISTROS'. Below this is the title 'LISTA DE CAMPAÑAS DE AUDIO PERSONALIZADO'. The main content is a table with the following data:

NOMBRE DE LA CAMPAÑA	CANAL	FECHA DE CREACIÓN	DESCRIPCIÓN	ESTADO	SUBIR ARCHIVOS	EVENTOS	ESTADÍSTICAS
lunas prueba personalizat	2	2018-03-12 12:21:07	12212	FINALIZADA	ARCHIVOS -	EVENTOS -	ESTADÍSTICAS
campana audio construido lunes	1	2018-03-12 04:43:19	construido	FINALIZADA	ARCHIVOS -	EVENTOS -	ESTADÍSTICAS
campana de texto	2	2018-03-07 10:35:17	wwwqqq	NO INICIADA	ARCHIVOS -	EVENTOS -	ESTADÍSTICAS

Figura 2.18 Interfaz_Lista Campañas Llamadas Personalizadas.

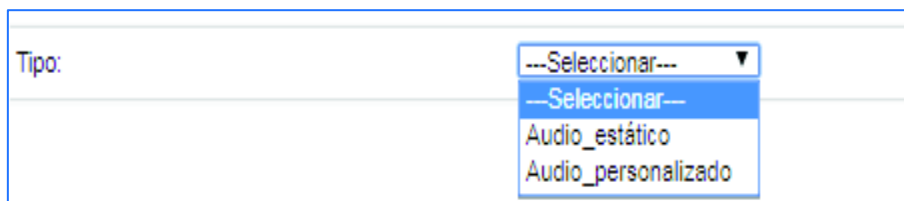
La **Figura 2.19** presenta la interfaz de creación de campañas de llamadas automáticas. Para crear cualquier tipo de campaña de llamadas automáticas se utiliza el mismo formulario en el cual se ingresa el nombre de la campaña, cuantos canales o llamadas simultáneas se quiere enviar, el número de intentos que el sistema ejecutará si la respuesta a nivel de telefonía es no contestada, el tiempo de espera entre cada intento, una descripción de la campaña y el tipo de campaña que se necesita.

La **Figura 2.20** muestra que el tipo de la campaña es un combobox en el cual se selecciona el tipo de campaña a crear.



The screenshot shows a web interface titled "AUTOMATIC SEND&DIALLING (ASD)" with a logo featuring a shield, a gear, and a globe. Below the title is the subtitle "Creación de Campañas Llamadas Automáticas". The form contains several input fields: "Nombre:" (text), "Tipo:" (dropdown menu with "--Seleccionar--"), "Canales:" (text), "Número de intentos:" (text), "Tiempo de espera entre intentos:" (text), and "Descripción:" (text). At the bottom, there are two buttons: "Salir" and "Crear".

Figura 2.19 Interfaz_Creación de Campañas de Llamadas Automáticas.



This close-up shows the "Tipo:" dropdown menu. The menu is open, displaying the following options: "--Seleccionar--", "--Seleccionar--", "Audio_estático", and "Audio_personalizado".

Figura 2.20 Interfaz_Creación de Campañas de Llamadas Automáticas Tipo.

Las **Figuras 2.21 y 2.22** presentan la interfaz de los elementos que permiten subir los diferentes tipos de archivos a las campañas de audio estático como de audio personalizado.

Las campañas de audio estático tienen tres elementos los cuales son: audio, contactos y detalle, las campañas de audio personalizado tienen dos elementos los cuales son contactos y detalle.



NOMBRE DE LA CAMPAÑA	CANAL	FECHA DE CREACIÓN	DESCRIPCIÓN	ESTADO	SUBIR ARCHIVOS	EVENTOS	ESTADÍSTICAS
lunes prueba 2	2	2018-03-12 12:12:30	12345678	NO INICIADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS
campana estatica	1	2018-03-12 05:44:51	qwewqe	FINALIZADA	AUDIO CONTACTOS DETALLE	EVENTOS +	ESTADÍSTICAS
prueba noche lunes	1	2018-03-11 22:15:27	qwerty	NO INICIADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS
prueba noche jueves	2	2018-03-08 23:10:45	prueba de noche	NO INICIADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS

Figura 2.21 Interfaz_ Elementos para subir archivos a las campañas de audio estático.



NOMBRE DE LA CAMPAÑA	CANAL	FECHA DE CREACIÓN	DESCRIPCIÓN	ESTADO	SUBIR ARCHIVOS	EVENTOS	ESTADÍSTICAS
lunes_prueba_personalizad	2	2018-03-12 12:21:07	12212	FINALIZADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS
campana audios contruidos lunes	1	2018-03-12 04:43:19	contruidos	FINALIZADA	CONTACTOS DETALLE	EVENTOS +	ESTADÍSTICAS
campana de texto	2	2018-03-07 10:36:17	wewqeeq	NO INICIADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS

Figura 2.22 Interfaz_ Elementos para subir archivos a las campañas de audio personalizado.

La **Figura 2.23** presenta la interfaz del elemento al seleccionar un archivo de audio para la campaña de audio estático. El archivo de audio debe ser de formato **.wav**.



Figura 2.23 Interfaz_ Seleccionar un archivo de audio para campañas de audio estático.

La **Figura 2.24** presenta la interfaz del elemento la seleccionar un archivo de texto para la campaña de audio estático. El archivo de texto debe ser de formato .csv y codificación UTF8, y una sola columna con los números de teléfono a los cuales se reproducirá el audio seleccionado previamente.



Figura 2.24 Interfaz_ Seleccionar un archivo de texto para campañas de audio estático.

La **Figura 2.25** presenta la interfaz del elemento la seleccionar un archivo de texto para la campaña de audio personalizado. El archivo de texto debe ser de formato .csv separado por comas y codificación UTF8. El archivo debe tener dos columnas en la primera debe estar los números de teléfonos y en la segunda columna el mensaje personalizado que se reproducirá a los números de teléfono asociados.



Figura 2.25 Interfaz_Seleccionar un archivo de texto para campañas de audio personalizado.

Las **Figuras 2.26 y 2.27** presentan la interfaz del elemento para mostrar el detalle de las campañas de llamadas automáticas. El detalle muestra el nombre de la campaña con su respectivo audio utilizado y los números telefónicos ingresados.

DETALLE CAMPAÑA DE LLAMADAS DE AUDIO ESTÁTICO	
NOMBRE DE LA CAMPAÑA ESTÁTICA	
CAMPAÑA ESTÁTICA	
AUDIO UTILIZADO	
0200014.wb	
TOTAL CONTACTOS INGRESADOS	
0	
DETALLE TELEFONOS INGRESADOS	
0995218349	
0996326045	
0995218349	
0996326045	
0995218349	

Figura 2.26 Intefaz_Detalle de la campaña de llamadas de audio estático.

DETALLE CAMPAÑA DE AUDIOS PERSONALIZADOS	
NOMBRE DE LA CAMPAÑA PERSONALIZADA	
LUNES PRUEBA PERSONALIZADO	
TOTAL CONTACTOS INGRESADOS	
11	
DETALLE DE CONTACTOS INGRESADOS	
TELÉFONO	MENSAJE
0995027960	HOLA BUENOS DIAS ES LUNES
0996326045	HOLA BUENOS DIAS 1
0995218349	HOLA BUENOS DIAS 2
0996326045	HOLA BUENOS DIAS 3

Figura 2.27 Intefaz_Detalle de la campaña de llamadas de audio personalizado.

Las **Figuras 2.28 y 2.29** presentan las interfaces de los eventos que puede tener una campaña ya sea de audio estático o audio personalizado. Los eventos son: Iniciar, detener, continuar y eliminar campaña.

NOMBRE DE LA CAMPAÑA	CANAL	FECHA DE CREACIÓN	DESCRIPCIÓN	ESTADO	SUBIR ARCHIVOS	EVENTOS	ESTADÍSTICAS
lunes prueba 2	2	2018-03-12 12:12:30	12345678	NO INICIADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS
campaña estática	1	2018-03-12 05:44:51	qwerty	FINALIZADA	ARCHIVOS +	INICIAR DETENER CONTINUAR ELIMINAR	ESTADÍSTICAS
prueba noche lunes	1	2018-03-11 22:15:27	qwerty	NO INICIADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS
prueba noche jueves	2	2018-03-08 23:10:45	prueba de noche	NO INICIADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS

Figura 2.28 Interfaz_Eventos de las Campañas estáticas.

NOMBRE DE LA CAMPAÑA	CANAL	FECHA DE CREACIÓN	DESCRIPCIÓN	ESTADO	SUBIR ARCHIVOS	EVENTOS	ESTADÍSTICAS
lunes prueba personalizada	2	2018-03-12 12:21:07	12312	FINALIZADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS
campaña audio construido lunes	1	2018-03-12 04:43:18	construido	FINALIZADA	ARCHIVOS +	INICIAR DETENER CONTINUAR ELIMINAR	ESTADÍSTICAS
campaña de texto	2	2018-03-07 15:36:17	wwwqqq	NO INICIADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS
campaña de text to speech	2	2018-03-07 05:24:03	campaña de textos	FINALIZADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS
campaña personalizada	1	2018-03-07 08:07:24	www	NO INICIADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS
contenido prueba 5	1	2018-03-05 10:45:57	123120	FINALIZADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS

Figura 2.29 Interfaz_Eventos de las Campañas Personalizadas.

Las **Figuras 2.30, 2.31, 2.32 y 2.33** presentan las interfaces de los mensajes de información sobre los eventos generados en las campañas de llamadas automáticas (con audio estático y personalizado). Los eventos son los mismos en los dos tipos de llamadas automáticas por lo que generan los mismos mensajes de información.

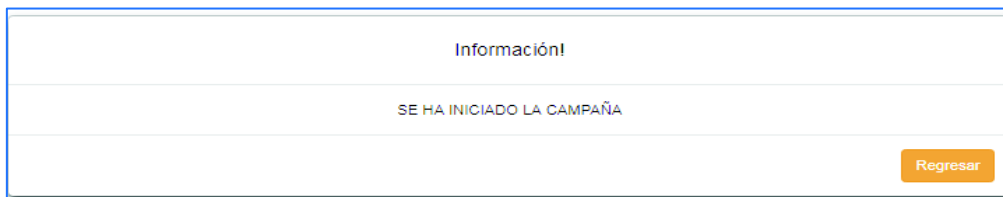


Figura 2.30 Interfaz_Mensajes de información de las campañas de llamadas automáticas_Iniciar campaña.

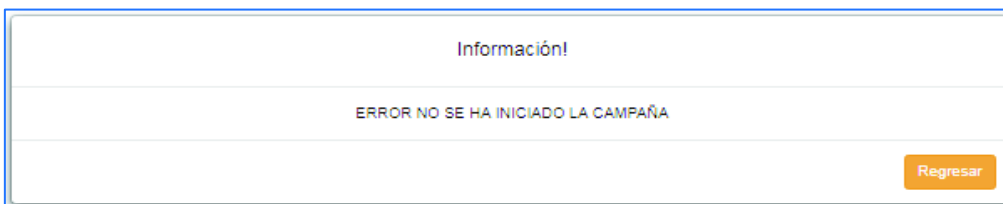


Figura 2.31 Interfaz_Mensajes de información de las campañas de llamadas automáticas_Error Iniciar campaña.

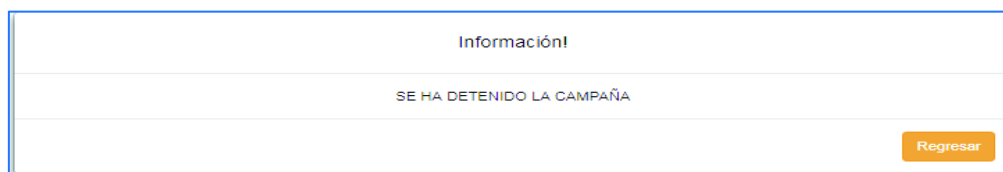


Figura 2.32 Interfaz_Mensajes de información de las campañas de llamadas automáticas_Detener campaña.

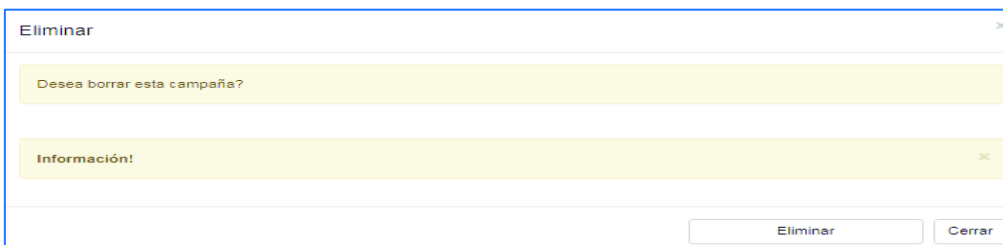


Figura 2.33 Interfaz_Mensajes de información de las campañas de llamadas automáticas_Eliminar campaña.

Las **Figuras 2.34 y 2.35** muestran la interfaz de estadísticas y resultados de la campaña de audio estática seleccionada. Este módulo normalmente se denomina reportería y sirve como una guía para la toma de mejores decisiones futuras para lanzar una campaña de audios pregrabados o personalizados. Se presentan gráficos tanto en barras como en pastel.

Las **Figuras 2.36 y 2.37** muestran reportes del progreso de las llamadas de audio estático o personalizado. Estos reportes muestran información en tiempo

real el porcentaje del progreso de la campaña así como cuantos canales simultáneos se están ejecutando y el estado de la campaña.



Figura 2.34 Interfaz_Reporte gráfico de las campañas estáticas diagrama de barras.

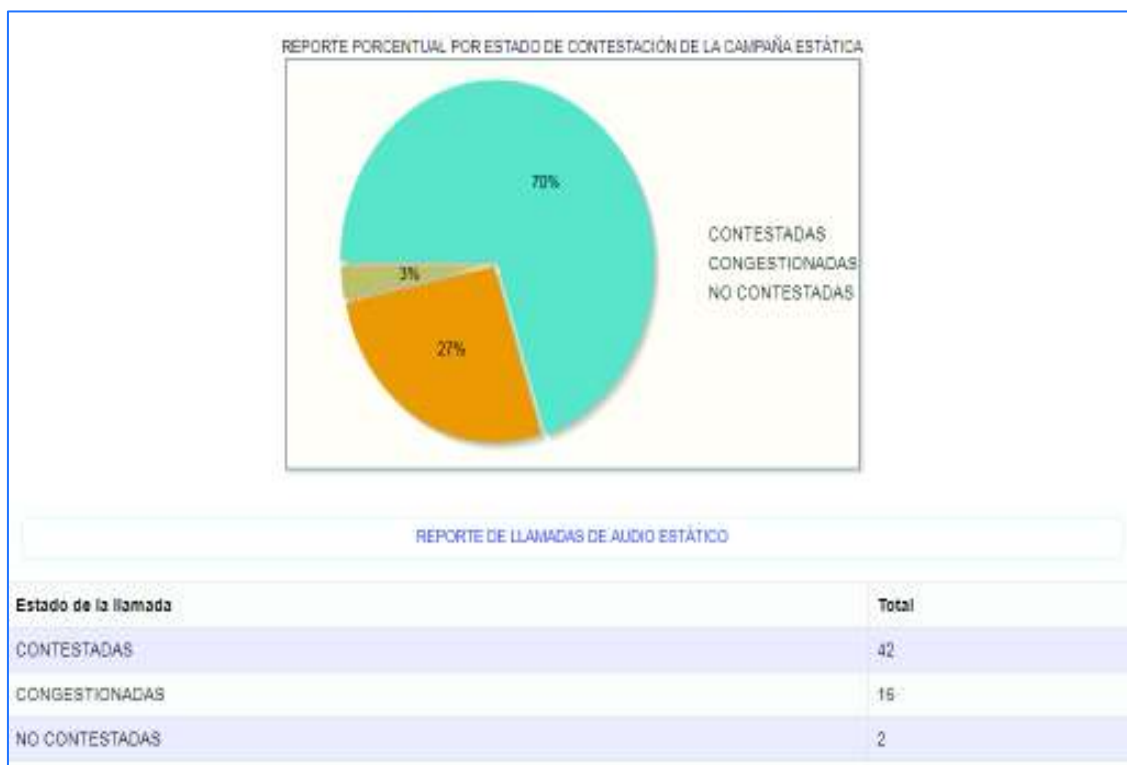


Figura 2.35 Interfaz_Reporte gráfico de las campañas estáticas diagrama de pastel.

REPORTE PROGRESO LLAMADAS AUDIO PREGRABADO				
Campaña	Canal	Estado	Progreso	Descripcion
2017ene24cumplenova	8	6	Finalizada	100%
TIEMPO DE LLAMADAS AUDIO PREGRABADO				
Tiempo Total de llamada	Tiempo Promedio de llamada	Tiempo Máximo de llamada	Tiempo Mínimo de llamada	
00:13:15	00:00:13	00:00:22	00:00:00	

Figura 2.36 Interfaz_Reporte gráfico de las campañas estáticas reporte estadístico.

La **Figura 2.37** muestra un reporte detallado de cada número de teléfono ingresado en la campaña de audio estático, su estado de contestación, el tiempo que duró la llamada y la fecha en la que se realizó la misma.

REGISTRO LLAMADAS AUDIO PREGRABADO			
Número telefónico	Fecha	Tiempo de llamada	Estado de llamada
0961656874	2017-01-24	00:00:00	CONGESTIONADA
0961943205	2017-01-24	00:00:00	NO CONTESTADA
0962978925	2017-01-24	00:00:00	CONGESTIONADA
0964258686	2017-01-24	00:00:21	CONTESTADA
0964576316	2017-01-24	00:00:22	CONTESTADA
0964925129	2017-01-24	00:00:22	CONTESTADA
0965864453	2017-01-24	00:00:22	CONTESTADA
0966975106	2017-01-24	00:00:22	CONTESTADA
0967387660	2017-01-24	00:00:00	CONGESTIONADA

Figura 2.37 Interfaz_Reporte de las campañas estáticas registro estadístico.

Las **Figuras 2.38 y 2.39** muestran la interfaz de estadísticas de la campaña de audio personalizado seleccionada. Los audios personalizados pueden ser utilizados para remitir saldos de deudas o confirmar las citas médicas.

Los reportes muestran gráficamente el estado de las llamadas por estado de contestación. Los gráficos son tanto diagramas de barras como en pastel.

La **Figura 2.40** muestra un reporte del progreso de las llamadas de audio personalizado. Este reporte muestra información en tiempo real el porcentaje del progreso de la campaña así como cuantos canales simultáneos se están ejecutando y el estado de la campaña.



Figura 2.38 Interfaz_Reporte gráfico de las campañas personalizadas diagrama de barras.

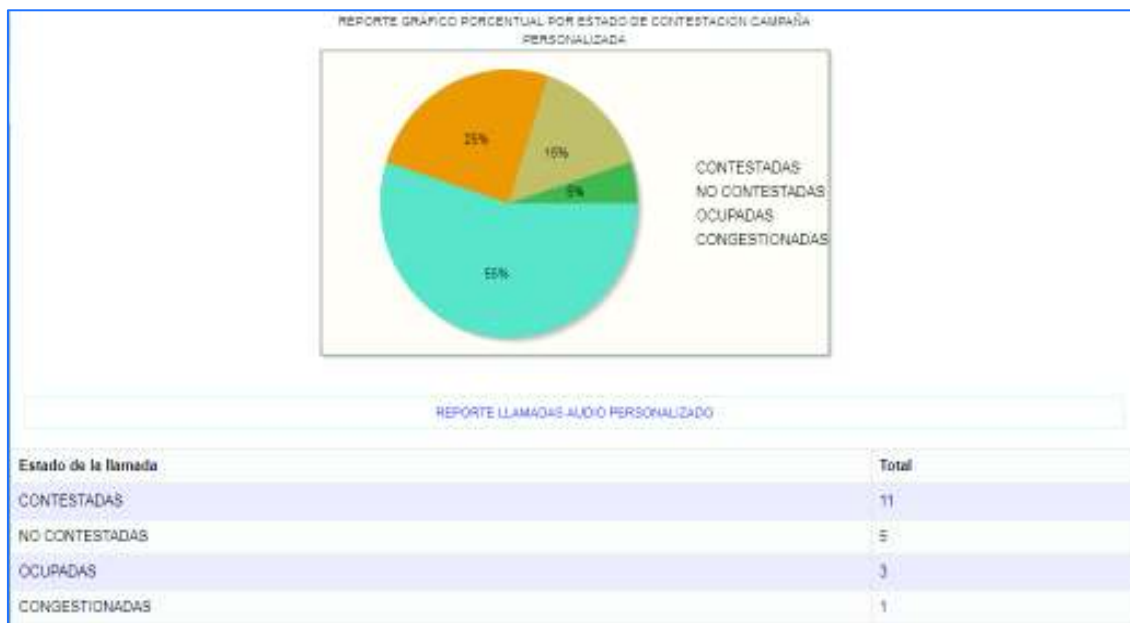


Figura 2.39 Interfaz_Reporte gráfico de las campañas personalizadas diagrama pastel.

REPORTE PROGRESO LLAMADAS AUDIO PERSONALIZADO				
Campaña	Canal	Estado	Progreso	Descripción
construida prueba 6	1	Finalizada	100%	123123

TIEMPO DE LLAMADAS AUDIO PERSONALIZADO			
Tiempo Total de llamada	Tiempo Promedio de llamada	Tiempo Máximo de llamada	Tiempo Mínimo de llamada
00 00 34	00 00 02	00 00 04	00 00 00

Figura 2.40 Interfaz_Reporte gráfico de las campañas personalizadas reporte estadístico.

La **Figura 2.41** muestra un reporte detallado de cada número de teléfono ingresado en la campaña de audio personalizado, su estado de contestación, el tiempo que duró la llamada y la fecha en la que se realizó la misma.

REGISTRO DE LLAMADAS AUDIO PERSONALIZADO			
Número telefónico	Fecha	Tiempo de llamada	Estado de llamada
0995216349	2018-03-07	00 00 03	CONTESTADA
0995216349	2018-03-06	00 00 03	CONTESTADA
0995216349	2017-03-06	00 00 00	OCUPADA
0995216349	2018-03-05	00 00 00	NO CONTESTADA
0995216349	2018-03-05	00 00 03	CONTESTADA
0995216349	2018-03-05	00 00 00	NO CONTESTADA
0995216349	2018-03-05	00 00 00	NO CONTESTADA
0995216349	2018-03-05	00 00 00	NO CONTESTADA
0995216349	2018-03-05	00 00 00	NO CONTESTADA
0995216349	2018-03-06	00 00 04	CONTESTADA
0996326045	2018-03-05	00 00 00	CONGESTIONADA

Figura 2.41 Interfaz_Reporte gráfico de las campañas personalizadas registro estadístico.

La **Figura 2.42** presenta la interfaz de los Mensajes de texto, sus campañas de mensajes de texto y una pestaña para creación de campañas de mensajes de texto (sms).



Figura 2.42 Interfaz_Mensajes de texto.

La **Figura 2.43** presenta la interfaz de la lista las campañas de mensajes de texto existentes en el sistema. Las campañas de mensajes de texto envían masivamente mensajes de texto a los números de teléfono ingresados inicialmente a la campaña de sms.



NOMBRE DE LA CAMPAÑA	CANAL	TRONCAL	FECHA DE CREACIÓN	DESCRIPCIÓN	ESTADO	SUBIR ARCHIVOS	EVENTOS	ESTADÍSTICAS
mensajes_lunes	2	192.168.0.9	2018-03-12 12:30:18	lunes mensajes	INICIADA	ARCHIVOS +	EVENTOS +	Estadísticas
CAMPAÑA MENSAJES 1	4	192.168.0.9	2018-03-12 03:30:43	asdfgh	NO INICIADA	ARCHIVOS +	EVENTOS +	Estadísticas
Campaña mensajes de texto	2	192.168.0.9	2018-03-07 08:22:38	1213	FINALIZADA	ARCHIVOS +	EVENTOS +	Estadísticas
ivályn uno	1	192.168.0.9	2017-03-06 12:33:24	232	FINALIZADA	ARCHIVOS +	EVENTOS +	Estadísticas

Figura 2.43 Interfaz_Lista Campañas de Mensajes de texto.

La **Figura 2.44** presenta la interfaz de creación de campañas de mensajes de texto. Para crear una campaña de mensajes de texto sms se ingresa el nombre de la campaña, la troncal por la cual se va enviar los mensajes de texto. Esta troncal es la dirección IP del *gateway* GSM.

Se ingresa un usuario y contraseña previamente configurados en el módulo de SMS del *gateway* GSM. Se ingresa el número de canales que se va a utilizar, en las campañas de mensajes de texto el número de canales hace referencia a cuantos mensajes de texto simultáneos se van a enviar en la campaña y una breve descripción de la campaña.

Las **Figura 2.45** presenta la interfaz de los elementos que permiten subir el archivo de texto a las campañas de mensajes de texto. Estas campañas tienen dos elementos los cuales son contactos y detalle.

La **Figura 2.46** presenta la interfaz del elemento la seleccionar un archivo de texto para la campaña de mensajes de texto (sms). El archivo de texto debe ser de formato .csv separado por comas y codificación UTF8. El archivo debe tener dos columnas en la primera debe estar los números de teléfonos y en la segunda columna el mensaje personalizado que se enviará a los números de teléfono asociados.

AUTOMATIC SEND&DIALLING (ASD)



Creación de Campañas Mensajes de texto

Nombre:

Troncal:

Usuario:

Contraseña:

Canales:

Descripción:

Figura 2.44 Interfaz_Creación de Campañas de Mensajes de texto.

AUTOMATIC SEND&DIALLING (ASD)



WebSiteEvelyn | LLAMADAS AUTOMÁTICAS + MENSAJES DE TEXTO (SMS) + REPORTES Y REGISTROS +

LISTA DE CAMPAÑAS DE MENSAJES DE TEXTO

NOMBRE DE LA CAMPAÑA	CANAL	TRONCAL	FECHA DE CREACIÓN	DESCRIPCIÓN	ESTADO	SUBIR ARCHIVOS	EVENTOS	ESTADÍSTICAS
mensajes_lunes	2	192.168.0.9	2018-03-12 12:30:19	lunes mensajes	INICIADA	ARCHIVOS +	EVENTOS +	Estadísticas
CAMPAÑA MENSAJES 1	4	192.168.0.9	2018-03-12 03:30:43	asdfgh	NO INICIADA	CONTACTOS DETALLE	EVENTOS +	Estadísticas
Campaña mensajes de texto	2	192.168.0.9	2018-03-07 09:22:36	1213	FINALIZADA	ARCHIVOS +	EVENTOS +	Estadísticas
evelyn uno	1	192.168.0.9	2017-03-06 12:33:24	232	FINALIZADA	ARCHIVOS +	EVENTOS +	Estadísticas
campaña smas	2	192.168.0.9	2017-03-05 12:31:43	qwq	NO INICIADA	ARCHIVOS +	EVENTOS +	Estadísticas
CAMPAÑA DE PRUEBA 2211	2	192.168.0.9	2018-03-05 11:42:06	prueba22	NO INICIADA	ARCHIVOS +	EVENTOS +	Estadísticas
CAMPAÑA DE PRUEBA	2	192.168.0.9	2018-03-05 11:39:19	prueba	FINALIZADA	ARCHIVOS +	EVENTOS +	Estadísticas
lunes	8	192.168.1.91	2017-01-16 10:36:26	16 de enero	FINALIZADA	ARCHIVOS +	EVENTOS +	Estadísticas

Figura 2.45 Interfaz_Lista de campañas de mensajes de texto.



Figura 2.46 Interfaz_Seleccionar un archivo de texto para campañas de mensajes de texto.

La **Figura 2.47** presenta la interfaz del elemento para mostrar el detalle de la campaña de mensajes de texto. El detalle muestra el nombre de la campaña con sus respectivos números telefónicos ingresados y el mensaje de texto que se enviará.

La **Figura 2.48** presenta la interfaz de los eventos que puede tener una campaña de mensajes de texto. Los eventos son: Iniciar, detener, continuar y eliminar campaña.

DETALLE DE LA CAMPAÑA MENSAJES DE TEXTO	
NOMBRE DE LA CAMPAÑA DE MENSAJES DE TEXTO	
CAMPAÑA MENSAJES DE TEXTO	
TOTAL CONTACTOS INGRESADOS	
18	
DETALLE CONTACTOS INGRESADOS	
NÚMERO TELEFÓNICO	MENSAJE INGRESADO
099218349	hola bueno dia 4
099218349	hola bueno dia 5
099218349	hola bueno dia
099218349	hola bueno dia 2
099949506	hola bueno dia 3
099949506	hola bueno dia 5
099218349	hola bueno dia 8

Figura 2.47 Interfaz_Detalle de la campaña de mensajes de texto.

AUTOMATIC SEND&DIALLING (ASD)



[Inicio](#) | [LLAMADAS AUTOMÁTICAS](#) | [MENSAJES DE TEXTO \(SMS\)](#) | [REPORTES Y REGISTROS](#)

LISTA DE CAMPAÑAS DE MENSAJES DE TEXTO

NOMBRE DE LA CAMPAÑA	CANAL	TRONCAL	FECHA DE CREACIÓN	DESCRIPCIÓN	ESTADO	SUBIR ARCHIVOS	EVENTOS	ESTADÍSTICAS
mensajes_lunas	2	100.100.0.0	2018-05-12 12:30:19	lunas mensajes	PENDIENTE	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS
CAMPAÑA MENSAJES 1	4	100.100.0.0	2018-08-12 08:30:43	mensajes	NÓ INICIADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS
Campaña mensajes de texto	2	100.100.0.0	2018-03-07 08:22:38	mensajes	FINALIZADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS
mensajes sms	1	100.100.0.0	2017-03-06 12:33:24	mensajes	FINALIZADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS
campaña sms	2	100.100.0.0	2017-03-06 12:31:43	mensajes	NÓ INICIADA	ARCHIVOS +	EVENTOS +	ESTADÍSTICAS

Figura 2.48 Interfaz_Eventos de las campañas de mensajes de texto.

Las **Figuras 2.49, 2.50, 2.51 y 2.52** presentan las interfaces de los mensajes de información sobre los eventos generados en las campañas de llamadas automáticas (con audio estático y personalizado). Los eventos son los mismos en los dos tipos de llamadas automáticas por lo que generan los mismos mensajes de información.

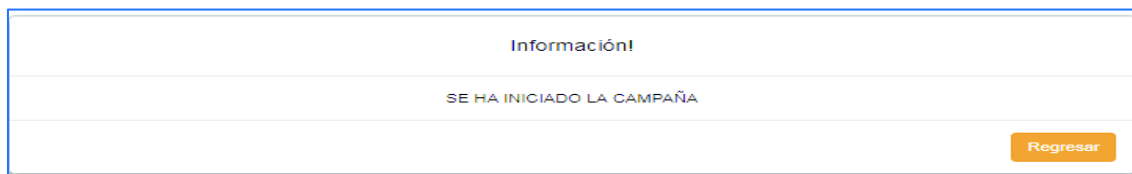


Figura 2.49 Interfaz_Mensajes de información de las campañas de mensajes de texto_Iniciar campaña.

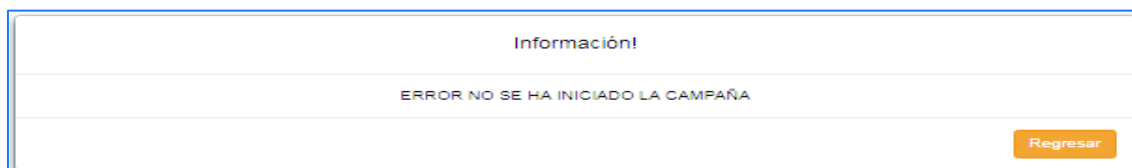


Figura 2.50 Interfaz_Mensajes de información de las campañas mensajes de texto_Error_Iniciar campaña.

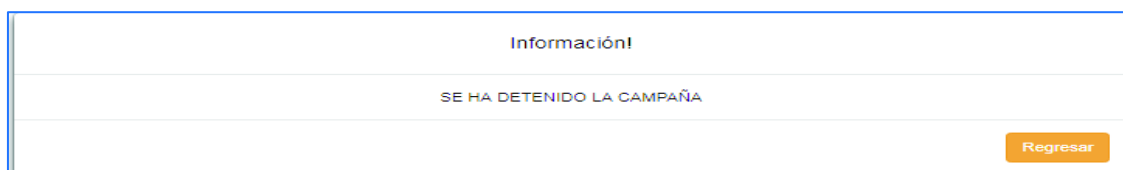


Figura 2.51 Interfaz_Mensajes de información de las campañas mensajes de texto_Detener campaña.

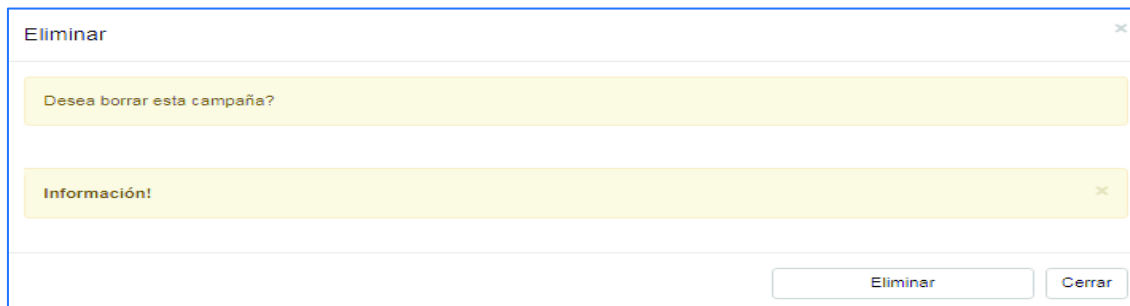


Figura 2.52 Interfaz_Mensajes de información de las campañas de mensajes de texto_Eliminar campaña.

Las **Figuras 2.53 y 2.54** muestran un reporte gráfico de los estados de contestación del *gateway* gsm de los mensajes de texto. Se tiene gráficos tanto en barras como en pastel.



Figura 2.53 Interfaz_Reporte gráfico de las campañas de mensajes de texto diagrama de barras.



Figura 2.54 Interfaz_Reporte gráfico de las campañas de mensajes de texto diagrama de pastel.

La **Figura 2.55** muestra un reporte del progreso de campaña de mensajes de texto. Este reporte muestra información en tiempo real el porcentaje del progreso de la campaña así como cuantos canales simultáneos se están ejecutando y el estado de la campaña.

También se muestra un registro de la campaña con el detalle del número de teléfono al que se envió el mensaje, la fecha y hora cuando se envió, el mensaje enviado el puerto del *gateway* gsm utilizado y el resultado que es la respuesta del mensaje de texto.

2. REPORTE PROGRESO ENVÍO DE MENSAJES DE TEXTO				
CAMPAÑA	CANAL	ESTADO	PROGRESO	DESCRIPCIÓN
CAMPAÑA DE PRUEBA	2	FINALIZADA	100%	prueba
3. REGISTRO DE MENSAJES DE TEXTO				
NÚMERO TELEFÓNICO	FECHA Y HORA DE ENVÍO	MENSAJE ENVIADO	PUERTO UTILIZADO	RESULTADO
0995216348	2019-03-26 12:42:25	hola buenos días prueba intento	GSM-1.2	EXITOSO
099526045	2019-03-26 12:42:30	hola buenos días prueba intento 2	GSM-1.2	EXITOSO
0995216348	2019-03-26 12:45:32	hola buenos días prueba intento 33	GSM-1.2	EXITOSO
099526045	2019-03-26 12:45:47	hola buenos días prueba intento 555	GSM-1.2	EXITOSO
099526045	2019-03-26 12:48:38	hola buenos días prueba intento	GSM-1.2	EXITOSO
DETALLE CONTACTOS INGRESADOS				
NÚMERO TELEFÓNICO	MENSAJE INGRESADO			
0995216348	hola buenos días prueba intento			
099526045	hola buenos días prueba intento 2			
0995216348	hola buenos días prueba intento 33			
099526045	hola buenos días prueba intento 555			
099526045	hola buenos días prueba intento			

Figura 2.55 Interfaz_Reporte gráfico de las campañas de mensajes de texto detalle.

CAPÍTULO 3

3 IMPLEMENTACIÓN Y PRUEBAS DEL SISTEMA

En este capítulo se describe la implementación de la central telefónica, el hardware y software telefónico, la base de datos y la aplicación web. Además se documenta los resultados obtenidos de las pruebas realizadas para evaluar la funcionalidad del prototipo.

3.1 IMPLEMENTACIÓN

El prototipo se implementa sobre un computador virtualizado y usa el sistema operativo Linux (CentOS 6). Se utiliza VirtualBox como software de virtualización para poder instalar el sistema operativo CentOS 6 y que se emule una máquina física. Para la comunicación telefónica se emplea la plataforma de telefonía Asterisk. La aplicación web para la gestión del prototipo es desarrollada usando PHP, HTML, CSS y JavaScript. La base de datos se implementa usando PostgreSQL.

3.1.1 ESQUEMA DEL PROTOTIPO

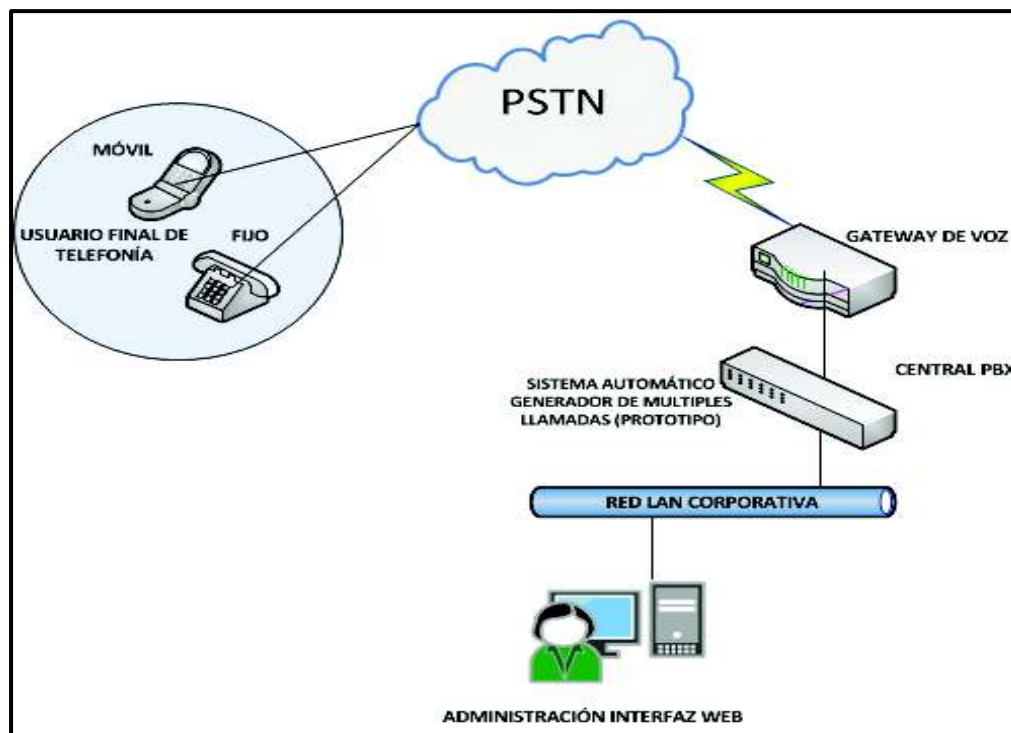


Figura 3.1 Diagrama del Prototipo.

Tal como se muestra en la **Figura 3.1**, para generar las llamadas hacia la PSTN, el prototipo se conecta (mediante una red de datos) a un *gateway* de voz (GSM) provisto de 2 líneas telefónicas celulares habilitadas. No se emplean líneas telefónicas fijas debido a la dificultad de disponer simultáneamente de más de una para realizar las pruebas.

3.1.2 IMPLEMENTACIÓN DE LA CENTRAL TELEFÓNICA

3.1.2.1 Implementación de Asterisk

Para implementar la Central Telefónica Asterisk en el sistema se elige descargar y compilar los repositorios de la versión 13.17.0 de Asterisk, que tienen todo lo necesario en el tema de telefonía para el prototipo.

Se elige la compilación de Asterisk ya que de este modo se pueden escoger que módulos se quiere instalar ya que muchos de los módulos que se instalan por defecto no son necesarios. Haciendo una instalación más ligera para el prototipo.

Para la compilación se debe descargar los archivos y descomprimirlos siguiendo una serie de pasos que se explican en el **Anexo A**.

La **Tabla 3.1** muestra los módulos que se eliminan de la compilación.

Tabla 3.1 Módulos de compilación

CATEGORÍA	MÓDULOS ELIMINADOS
Call Detail Recording	cdr_pgsq, cdr_sqlite3, cdr_odbc.
Channel Event Logging	cel_psql, cel_sqlite
Channel Drivers	chan_skinny y chan_unistim
PBX Modules	pbx_ael y pbx_dundi

El resto de los módulos se instalan como vienen configurados por defecto.

3.1.2.2 Configuración de Asterisk

Para utilizar Asterisk se deben configurar los archivos: **sip.conf**, **extensions.conf** y **voices.scm**, los cuales vienen con la configuración por defecto en la instalación. Antes de configurar los archivos mencionados se realiza una copia de los archivos originales de Asterisk como respaldo para poder utilizarlos si existen errores de configuración o edición en los archivos configurados.

El archivo **sip.conf** se configura como muestra la **Figura 3.2**, sus parámetros fueron explicados en la **sección 1.1.7.1**.

```
[general]
context=default           ; Contexto por defecto para llamadas entrantes.
allowguest=no             ; Rechazar llamadas no autenticadas.
udpbindaddr=0.0.0.0      ; Escucha peticiones UDP en todas las interfaces.
tcpenable=no             ; Deshabilita soporte TCP.
accept_outofcall_message=yes ; Permite el envío de mensajes de texto instantáneos.
outofcall_message_context=cvmsg ; Especifica el contexto en el plan de marcado.
                           ; Donde se procesarán los mensajes enviados.

[gw-gsm]                  ; Sección del terminal gw-gsm.
type=friend               ; Indica al controlador del canal asociar el nombre del terminal y la IP.
host=192.168.0.12         ; Se establece la dirección IP que el gateway gsm tendrá en su configuración.
context=callmatic         ; Contexto con el cuál se asocia las llamadas realizadas desde la aplicación.
disallow=all              ; Deshabilita todos los codecs de voz.
allow=alaw                ; Habilita al códec seleccionado.
allow=ulaw                ; Habilita al segundo códec seleccionado.
nat=no                   ; Cuando el valor es Yes, Asterisk ignora la información de dirección.
                           ; En los encabezados SIP y SDP, respondiendo a la dirección IP y puerto remitente.
                           ; (Ip pública y puerto).
```

Figura 3.2 Configuración archivo sip.conf.

El archivo **extensions.conf** contiene el plan de marcado, en su interior se configura dos contextos; estos contextos son **[callmatic]** y **[msg]**. Los parámetros de este archivo fueron explicados en la **sección 1.1.7.1**.

El contexto **callmatic** se utiliza para generar las llamadas automáticas tanto estáticas como personalizadas. Este contexto contiene dos extensiones **audiofile** y **audiobuilt**. La extensión **audiofile** realiza la llamada a los contactos ingresados y reproduce el audio previamente grabado. La extensión **audiobuilt** realiza la llamada, mediante la aplicación *text2wave* construye el audio del texto ingresado y reproduce al contacto final como se mencionó en la

sección 1.1.7.3. Text2wave es una aplicación de Festival que permite tomar el texto de un archivo de texto, convertirlo en sonido y almacenarlo en formato wav [37], [38].

En la **Figura 3.3** se muestra la configuración del contexto **callmatic** del archivo **extensions.conf**.

```
[callmatic]
exten => audiofile,1,Wait|1)          ; Extensión para llamadas estáticas.
same => n,Answer{}                   ; Aplicación para contestar el canal.
same => n,Playback|${CV_FILENAME}}   ; Reproducir el audio.
same => n,Hangup{}                   ; Terminar la llamada.

exten => audiobuilt,1,Wait|2)         ; Extensión para llamadas personalizadas,
same => n,NoOp|FESTIVAL)              ;
same => n,Answer{}                   ; Aplicación para contestar el canal.
same => n,System|/bin/echo "${CV_FILENAME}" | /usr/bin/text2wave -scale 1.5 -F 8000 -o /tmp/tts-${CDR|uniqueid}.wav} ; Aplicación que envía comandos al sistema.
same => n,Verbose|Creando Archivo ${SYSTEMSTATUS})
                                     ; Imprimir en consola opciones del nivel de detalle.
same => n,System|/usr/bin/mox /tmp/tts-${CDR|uniqueid}.wav /tmp/tts-${CDR|uniqueid}.wav tempo 1.0)
same => n,Verbose|Modificando Archivo ${SYSTEMSTATUS})
same => n,Playback|/tmp/tts-${CDR|uniqueid}.wav}
                                     ; Reproduce el audio en la ruta indicada.
same => n,Verbose|Guardando Archivo ${SYSTEMSTATUS})
```

Figura 3.3 Archivo extensions.conf contexto callmatic.

El contexto **msg** se utiliza para el envío instantáneo de mensajes de texto. En la **Figura 3.4** se muestra la configuración del contexto **msg** del archivo **extensions.conf**.

```
;;;;;;;;SIP MESSAGES;;;;;;;;
[msg]
exten => _!,1,NoOp(SMS receiving dialplan invoked)
same => n,NoOp(To ${MESSAGE(to)})
same => n,NoOp(From ${MESSAGE(from)})
same => n,NoOp(Body ${MESSAGE(body)})
same => n,Set(ACTUALTO=${CUT(MESSAGE(to),@,1)}) ;almacena en una variable el mensaje recibido
same => n,MessageSend(${ACTUALTO},${MESSAGE(from)})
same => n,NoOp(Send status is ${MESSAGE_SEND_STATUS})
same => n,GotoIf("${MESSAGE_SEND_STATUS}" != "SUCCESS"?sendfailedmsg)
same => n,Hangup()
```

Figura 3.4 Archivo extensions.conf contexto msg.

El archivo **voices.scm** se configura para cambiar la voz de Festival que por defecto viene en inglés. Se recomienda realizar este cambio por que la voz en

inglés no pronuncia clara y correctamente muchas palabras del idioma español. Por ejemplo, palabras que contienen letras especiales como la Ñ o tildes.

El **Comando 3.1** presenta como agregar los paquetes para el idioma en español con voces masculina y femenina.

```
# wget www.4shared.com/file/EdA3prfs/festvox-palpc16k-10-2noarch.html
# wget www.4shared.com/file/EdA3prfs/festvox-palpc16k-10-2noarch.html
```

Comando 3.1 Instalación de voces en español para Festival [39], [40].

Después de descargarse los paquetes y descomprimirlos en el sistema operativo, se crean dos carpetas las cuales contienen las voces en español tanto femenina como masculina. La carpeta (JuntaDeAndalucia_es_sf_diphone) contiene la voz femenina en español y la carpeta (JuntaDeAndalucia_es_pa_diphone) contiene la voz masculina en español.

Para establecer la voz que se va a utilizar, se debe agregar una línea con el nombre de la carpeta seleccionada en el archivo voces.scm como se muestra en la **Figura 3.5**.

```
; This list has been reordered for Fedora, since we want to have the option
; to only install a smallish voice by default. And the slt voice is nice.
(defvar default-voice-priority-list
  '(JuntaDeAndalucia es pa diphone
    nitech_us_slt_arctic_hts
    nitech_us_awb_arctic_hts
    nitech_us_bdl_arctic_hts
    nitech_us_clb_arctic_hts
    nitech_us_jmk_arctic_hts
    nitech_us_rms_arctic_hts
    kal_diphone
    ked_diphone
    cstr_us_awb_arctic_multisyn
    cstr_us_jmk_arctic_multisyn
  )
  "default-voice-priority-list
  List of voice names. The first of them available becomes the default voice.")
```

Figura 3.5 Ejemplo del cambio a voz en español del archivo voces.scm.

Para el envío masivo de mensajes de texto se desarrolla un *script* en Perl que permite enviar los datos requeridos por el *gateway* GSM para realizar el envío masivo de mensajes de texto.

El *script* se lo nombra como **sms.pl** y está ubicado en el directorio **/opt/** que es el directorio donde se puede instalar aplicaciones opcionales del sistema operativo Linux. El *script* se encarga del envío de mensajes de texto mediante URL³² hacia el *gateway* gsm.

```
http://192.168.0.12:80/sendsms?username=admin&password=admin&phonenumber=0995216349&message=prueba3
```

Figura 3.6 Ejemplo de URL.

```
#!/usr/bin/perl
use LWP::UserAgent;
use JSON;
use strict;
use DBI;

#Conexión con la base datos

my $dbh = DBI->connect
("DBI:Pg:dbname=automatic;host=localhost;port=5432",
"postgres", "paradise", {'RaiseError' => 1});

#Variables para el URL
my ($campid, $contid, $trunk, $trunkusername, $trunkpassword,
$phone, $msg) = ($ARGV[0], $ARGV[1], $ARGV[2], $ARGV[3], $ARGV
[4], $ARGV[5], $ARGV[6]);

my $ua = LWP::UserAgent->new;

# Construcción del URL para enviarlo al gateway GSM
my $server_endpoint = "http://$trunk:80/sendsms?username=
$trunkusername&password=$trunkpassword&phonenumber=
$phone&message=$msg";
```

Figura 3.7 Ejemplo de una parte del *script* para envío de sms en perl.

³² URL *Uniform Resource Locator* o Localizador Uniforme de Recursos es un identificador de recursos uniforme, se trata de una secuencia de caracteres que sigue un estándar y permite denominar recursos dentro del entorno de Internet para que puedan ser localizados.

El *script* es ejecutado mediante la función **public.sms_start** creada en la base de datos **automatic**, Primero el *script* establece una conexión con la base de datos, luego construye el URL que se enviará al *gateway* GSM, en la **Figura 3.6** se muestra un ejemplo como se envía el URL al *gateway* GSM.

Luego toma la respuesta JSON que responde el *gateway*, almacena los estados obtenidos como resultados en variables e inserta estos valores en la tabla **public.ast_matic_cdr**. Para construir el URL se necesita la troncal, el usuario y contraseña, configurados en el módulo de SMS del *gateway* GSM, el número telefónico y finalmente el mensaje a enviar.

La **Figura 3.6** muestra una parte del *script* para enviar los mensajes de texto, en el **Anexo B** se adjunta los *scripts* completos.

3.1.3 CONFIGURACIÓN DEL GATEWAY GSM

El equipo OpenVox VS-GW1202-4G es configurable y administrable desde la parte web. Es decir que desde cualquier navegador web se puede acceder a la administración del equipo. El *gateway* GSM tiene un puerto ethernet que viene configurado con una dirección IP estática de fábrica (**172.16.99.1**), un usuario y una contraseña (admin, admin).



Figura 3.8 Pantalla de inicio del *gateway* GSM OpenVox.

Para la configuración del equipo se realizaron las siguientes configuraciones:

1. Se asigna una dirección IP estática para el dispositivo en el módulo **NETWORK -> LAN Settings**. La **Figura 3.9** muestra la configuración de red del *gateway* OpenVox GSM.

The screenshot shows the 'IP Settings' configuration page. The 'Address' field is set to 192.168.0.12, the 'Netmask' is 255.255.255.0, and the 'Default Gateway' is 192.168.0.1. A red box highlights these three fields, and a red arrow points to the 'Configuración de red' label.

Figura 3.9 Configuración de red *Gateway* GSM.

2. Configuración de la dirección IP de la Central Telefónica en el módulo **VoIP-> VoIP Endpoints-> Add new SIP Endpoint**. La nueva troncal debe tener un nombre para identificarlo, escogemos el parámetro sin registro y colocamos en el parámetro **Hostname or IP Address** la dirección IP de la central telefónica.

La **Figura 3.10** muestra la configuración de una troncal SIP.

The screenshot shows the 'Add New SIP Endpoint' configuration page. The 'Name' field is set to Test-1, the 'User Name' is Anonymous, the 'Registration' is set to a dropdown menu, the 'Hostname or IP Address' is 192.168.0.212, the 'Transport' is UDP, and the 'NAT Traversal' is Yes. A red box highlights the 'Add New SIP Endpoint' title.

Figura 3.10 Configuración de la Troncal SIP.

3. Creación de grupos de los puertos de los chips, en el módulo **ROUTING-> Groups->New Group** se crea un nuevo grupo de chips y se escoge: el tipo (GSM), la política (*Roundrobin*³³) y los miembros que son los puertos del *gateway* GSM. La **Figura 3.11** muestra la creación de grupos en el *gateway* OpenVox GSM.

³³ *Roundrobin*: es una estrategia de marcación en la cual las llamadas salientes se van a enviar a los puertos en secuencia uno tras otro, dando prioridad al siguiente hasta que inicie de nuevo.

The screenshot shows the 'New Group' configuration interface. At the top, there is a 'New Group' button. Below it, the 'Routing Groups' section contains the following fields:

- Group Name:** An empty text input field.
- Type:** A dropdown menu set to 'GSM'.
- Policy:** A dropdown menu set to 'Roundrobin'.
- Members:** A list of members with checkboxes:

NO.	Member Name	Selected
1	gsm-1.1	<input type="checkbox"/>
2	gsm-1.2	<input checked="" type="checkbox"/>
3	gsm-1.3	<input type="checkbox"/>
4	gsm-1.4	<input type="checkbox"/>

Figura 3.11 Creación de Grupos en el gateway OpenVox GSM.

- Configuración de rutas, en el módulo **ROUTING-> CallRouting Rules->New Call Routing Rule** se crea las rutas para las llamadas entrantes y salientes. Una ruta es el camino por donde se va a recibir o enviar la llamada, desde el origen que puede ser desde una troncal SIP hacia el destino que puede ser el grupo de chips.

La **Figura 3.12** muestra la creación de rutas en el gateway OpenVox GSM.

The screenshot shows the 'Call Routing Rule' configuration interface. The 'Routing Name' is set to 'support'. The 'Call Comes in From' dropdown is set to 'support', and the 'Send Call Through' dropdown is set to 'gsm-1'. Red arrows point to the 'Origen de la llamada' and 'Destino de la llamada' labels.

Figura 3.12 Creación de rutas en el gateway OpenVox GSM.

- Configuración para el envío masivo de mensajes de texto en módulo **SMS-> SMS Settings-> HTTP a SMS** se habilita la función **HTTP a SMS** a **ON**, el valor predeterminado es **OFF**. La **Figura 3.13** muestra los parámetros que se deben configurar para el envío de mensajes de texto mediante el URL. Los campos más críticos de este módulo son:

- **User Name:** es el nombre de usuario que se va a ingresar para el envío de mensajes mediante un URL. Este parámetro debe ser el

mismo que el campo usuario en la creación de campañas de mensajes de texto desde la aplicación web.

- **Password:** es la contraseña que se va a ingresar para el envío de mensajes mediante un URL. Este parámetro debe ser el mismo que el campo contraseña en la creación de campañas de mensajes de texto desde la aplicación web.
- **Port:** son los puertos de los chips del *gateway* GSM por los cuales se va enviar los mensajes de texto.
- **Report:** es el tipo de informe de noticias que el *gateway* genera. Si se elige “No report” quiere decir que no se necesita un informe de las notificaciones del mensaje. Para el prototipo se utiliza JSON.

The screenshot shows a configuration form for SMS sending. The 'Enable' toggle is turned 'ON'. The 'URL' field contains a complex HTTP endpoint. The 'User Name' is set to 'admin' and the 'Use web server's user and password' checkbox is checked. The 'Password' field is masked with dots. The 'Port' section has a grid of checkboxes for various GSM ports (gsm-1.1 to gsm-5.3), all of which are checked. The 'Report' dropdown is set to 'String'. The 'Advanced' toggle is turned 'OFF'.

Figura 3.13 Configuración de parámetros para el envío de SMS.

3.1.4 IMPLEMENTACIÓN DE NIVEL DE DATOS

3.1.4.1 CONFIGURACIÓN DE LA BASE DE DATOS

Para la implementación de la base de datos es necesario instalar los paquetes de postgresql. El **Comando 3.2** muestra los paquetes a instalar.

```
# yum -y install postgresql-server postgresql-plperl postgresql-contrib
postgresql-devel kernel kernel-devel
```

Comando 3.2 Comandos para instalación de paquetes Postgresql.

Para acceder a la base de datos se debe primero ingresar al archivo de configuración `pg_hba.conf` y editar la columna **METHOD** con el valor “*trust*”. El archivo `pg_hba.conf` controla la autenticación del cliente al ingresar a postgresql database, se encuentra en la ruta `/var/lib/pgsql/data/pg_hba.conf`, las siglas **HBA** significan autenticación basada en host [42].

La **Figura 3.14** muestra la configuración del archivo `pg_hba.conf`.

```
# TYPE      DATABASE     USER        CIDR-ADDRESS          METHOD
# "local" is for Unix domain socket connections only
local      all         all                               trust
# IPv4 local connections:
host       all         all         127.0.0.1/32         trust
```

Figura 3.14 Archivo `pg_hba.conf`.

Se reinicia el servicio postgresql y se accede a la base de datos como se muestra en el **Comando 3.3**.

```
# service postgresql restart
# pgsq -U postgres
```

Comando 3.3 Comandos para reiniciar y acceder el servicio postgresql.

```
CREATE TABLE private.automatic_campaigns (
  id integer NOT NULL,
  name character varying(32) NOT NULL,
  channels integer DEFAULT 5 NOT NULL,
  trunk text NOT NULL,
  state integer DEFAULT 0 NOT NULL,
  created timestamp(0) without time zone DEFAULT now() NOT NULL,
  description character varying(128) DEFAULT NULL::character varying,
  campaigntype coc_callmatic_campaigns_type,
  maxretries integer DEFAULT 1,
  retrytime integer DEFAULT 30,
  CONSTRAINT automatic_campaigns_channels_check CHECK ((channels > 0))
);

ALTER TABLE private.automatic_campaigns OWNER TO postgres;
```

Figura 3.15 Ejemplo de creación de tablas en la base de datos.

Se crea la base de datos “**automatic**” con sus respectivas tablas. La **Figura 3.15** presenta un ejemplo de la creación de las tablas que se trató en la **sección 2.2.1.2**. En la **Tabla 3.2** se describe cada una de las tablas creadas para el prototipo con las acciones permitidas.

En el **Anexo B** se presenta el código de la creación de todas las tablas utilizadas en el prototipo.

Tabla 3.2 Eventos permitidos para cada tabla

Nombre de la Tabla	Eventos permitidos
private.automatic_campaign	<ul style="list-style-type: none"> • Inserción (<i>Insert</i>) • Modificación (<i>Update</i>) • Eliminación (<i>Delete</i>) • Lectura (<i>Read</i>)
private.contacts_prerecorded	<ul style="list-style-type: none"> • Inserción (<i>Insert</i>) • Modificación (<i>Update</i>) • Eliminación (<i>Delete</i>) • Lectura (<i>Read</i>)
private.contacts_built	<ul style="list-style-type: none"> • Inserción (<i>Insert</i>) • Modificación (<i>Update</i>) • Eliminación (<i>Delete</i>) • Lectura (<i>Read</i>)
private.audio_prerecord	<ul style="list-style-type: none"> • Inserción (<i>Insert</i>) • Modificación (<i>Update</i>) • Eliminación (<i>Delete</i>) • Lectura (<i>Read</i>)
public.ast_matic_cdr	<ul style="list-style-type: none"> • Inserción (<i>Insert</i>) • Lectura (<i>Read</i>)
private.users	<ul style="list-style-type: none"> • Inserción (<i>Insert</i>) • Modificación (<i>Update</i>) • Eliminación (<i>Delete</i>) • Lectura (<i>Read</i>)
private.sms_campaigns	<ul style="list-style-type: none"> • Inserción (<i>Insert</i>) • Modificación (<i>Update</i>)

	<ul style="list-style-type: none"> • Eliminación (<i>Delete</i>) • Lectura (<i>Read</i>)
private.sms_contacts	<ul style="list-style-type: none"> • Inserción (<i>Insert</i>) • Modificación (<i>Update</i>) • Eliminación (<i>Delete</i>) • Lectura (<i>Read</i>)
private.sms_mdr	<ul style="list-style-type: none"> • Inserción (<i>Insert</i>) • Lectura (<i>Read</i>)

La base de datos es una parte fundamental del proyecto porque todas las funciones que rigen al prototipo son creadas aquí. La **Figura 3.16** presenta la creación de una función de la base de datos. La **Tabla 3.3** describe cada una de las funciones creadas para el prototipo.

En el **Anexo B** se presenta el *script* completo de la base de datos.

```
CREATE FUNCTION public.sms_stop(integer) RETURNS integer
  LANGUAGE plperl SECURITY DEFINER
  AS $_$
    my ($campaignid) = @_ ;
    spi_exec_query ("UPDATE private.sms_campaigns SET state = 0 WHERE id = $campaignid");
    return 1;

$_$;

ALTER FUNCTION public.sms_stop(integer) OWNER TO postgres;
```

Figura 3.16 Ejemplo de una función creada en la base de datos.

Tabla 3.3 Descripción de funciones creadas en el prototipo.

Nombre de la función	Argumentos	Descripción
public.automatic_createcampaign	varchar, integer, varchar, varchar, integer, integer, varchar	Crea las campañas de llamadas automáticas (estáticas y personalizadas).

public.automatic_delcampaign	integer	Borra las campañas de llamadas automáticas (estáticas y personalizadas.)
public.automatic_getcampaign	Integer	Muestra el progreso, en tiempo real, de las campañas en porcentaje, el estado, los canales configurados y la descripción de la campaña.
public.automatic_getcampaigns	vacío	Lista las campañas existentes. Muestra nombre, canales, troncal, estado, fecha de creación y descripción.
public.automatic_getcontacts	integer	Lista los contactos asociados a una campaña estática.
public.automatic_getcontacts_built	integer	Lista los contactos asociados a una campaña personalizada.
public.automatic_getaudio	integer	Lista los audios asociados a una campaña estática.
public.automatic_getreport	integer	Presenta un reporte de la campaña por estado de contestación.
public.automatic_setcontacts	Integer, varying	Inserta los contactos asociados a la campaña.
public.automatic_setaudio	Integer, varying	Inserta los audios asociados a la campaña estática.
public.automatic_start	Integer, varying	Inicializa las campañas de llamadas automáticas (estáticas y personalizadas).
public.automatic_stop	Integer	Detiene las campañas de llamadas automáticas (estáticas y personalizadas).
public.sms_createcampaign	varchar, integer, varchar, varchar, varchar,	Crea campañas de mensajes de Texto.

	varchar	
public.sms_delcampaign	integer	Borra campañas de mensajes de Texto.
public.sms_getcampaign	integer	Muestra el progreso, en tiempo real, de las campañas en porcentaje, el estado, los canales configurados, el usuario, la contraseña y la descripción de la campaña.
public.sms_getcampaigns	vacío	Lista las campañas existentes. Muestra nombre, canales, troncal, el usuario, la contraseña, estado, fecha de creación y descripción.
public.sms_getcontacts	integer	Lista los contactos asociados a la campaña de mensajes de texto.
public.sms_getreport	integer	Presenta un reporte de la campaña por estado de contestación.
public.sms_setcontacts	Integer, varying.	Inserta los contactos asociados a la campaña de mensajes de texto.
public.sms_start	Integer	Inicializa las campañas de mensajes de texto.
public.sms_stop	Integer	Detiene las campañas de mensajes de texto.
private.sms_mdr_t_a	vacío	Disparador, inserta los registros en la tabla después de ejecutar las campañas de mensajes de texto.
private.ast_matic_cdr_t_a	vacío	Disparador, inserta los registros en la tabla después de ejecutar las campañas automáticas (estáticas y personalizadas.)

3.1.4.2 IMPLEMENTACIÓN DE LA APLICACIÓN WEB

La aplicación web es la encargada de la lógica de negocio y de la interacción del usuario con el modelo y el controlador. Para la implementación de la aplicación Web se instala el servidor web Apache. Apache es un servidor HTTP³⁴ de código libre que funciona sobre varios sistemas operativos tanto Linux, Windows, entre otras plataformas. Apache soporta lenguajes de programación como Perl, Php, etc., así como varios códigos como XML³⁵ entre otros; permite autenticación con contraseña y puede manejar tantos sitios web como el sistema operativo lo permita. Se debe instalar los paquetes de apache y php para CentOS. El **Comando 3.4** muestra los paquetes que se van a instalar para el servidor apache.

```
yum -y install httpd dhclient dhcp gcc gcc-c++ php php-pdo php-pgsql
```

Comando 3.4 Comandos para instalación de paquetes Apache.

Una vez instalado Apache se debe configurar el parámetro *DocumentRoot* del archivo **httpd.conf** para que al acceder a la dirección IP de la central telefónica se dirija a la aplicación Web creada. *DocumentRoot* es el directorio en el cual se colocan la mayoría de los archivos HTML; está predefinido con la ruta */var/www/html* pero se puede añadir el directorio en el cual se encuentra la aplicación para tener acceso inmediato mediante el navegador. La **Figura 3.17** muestra la configuración del archivo **httpd.conf**.

Las funcionalidades para la aplicación web fueron desarrolladas en NetBeans IDE 8.2 para CentOS 6. NetBeans es un entorno de desarrollo gratuito y de código abierto para lenguajes de programación como Java, PHP, etc. que ayudan a desarrollar aplicaciones Web de manera más fácil [43].

³⁴ HTTP: *Hypertext Transfer Protocol* o protocolo de transferencia de Hipertexto, es decir es utilizado para la transferencia de información por medio del internet.

³⁵ XML: *eXtensible Markup Language* es un Lenguaje de Etiquetado Extensible, es un lenguaje similar a HTML (*HyperText Markup Language* o lengua de marcas de hipertexto). Su función principal es permitir la lectura de datos a través de diferentes aplicaciones.

```

# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "/var/www/html/tesis"

#
# Each directory to which Apache has access can be configured with respect
# to which services and features are allowed and/or disabled in that
# directory (and its subdirectories).
#
# First, we configure the "default" to be a very restrictive set of
# features.
#
<Directory />
    Options FollowSymLinks
    AllowOverride All
</Directory>

```

Figura 3.17 Configuración del archivo httpd.conf.

A continuación, se presenta segmentos de código de las distintas funcionalidades que se ha implementado en la aplicación web para integrar la base de datos con Asterisk y presentar una interfaz al usuario final.

- **Establecimiento de la conexión de la base de datos:** Para establecer la comunicación con la base de datos se crea el archivo DataBase.php, en el que se crea la clase Database, con la función connectDB. La función connectDB permite tener comunicación y conexión con la base de datos mediante un adaptador. Se necesita declarar el usuario, contraseña, nombre de la base de datos con la cual se conectará, el puerto por el cual escuchará postgresql y mediante el adaptador se verificará la conexión de la base datos.

En el **Código 3.1** se presenta el archivo DataBase.php con su respectivo código encargado de la conexión con la base de datos.

- **Formulario Entrada (index.php):** El formulario de entrada se implementa en el archivo index.php y que mediante el **Código 3.2** se toman los datos que ingresa el usuario al entrar a la aplicación.

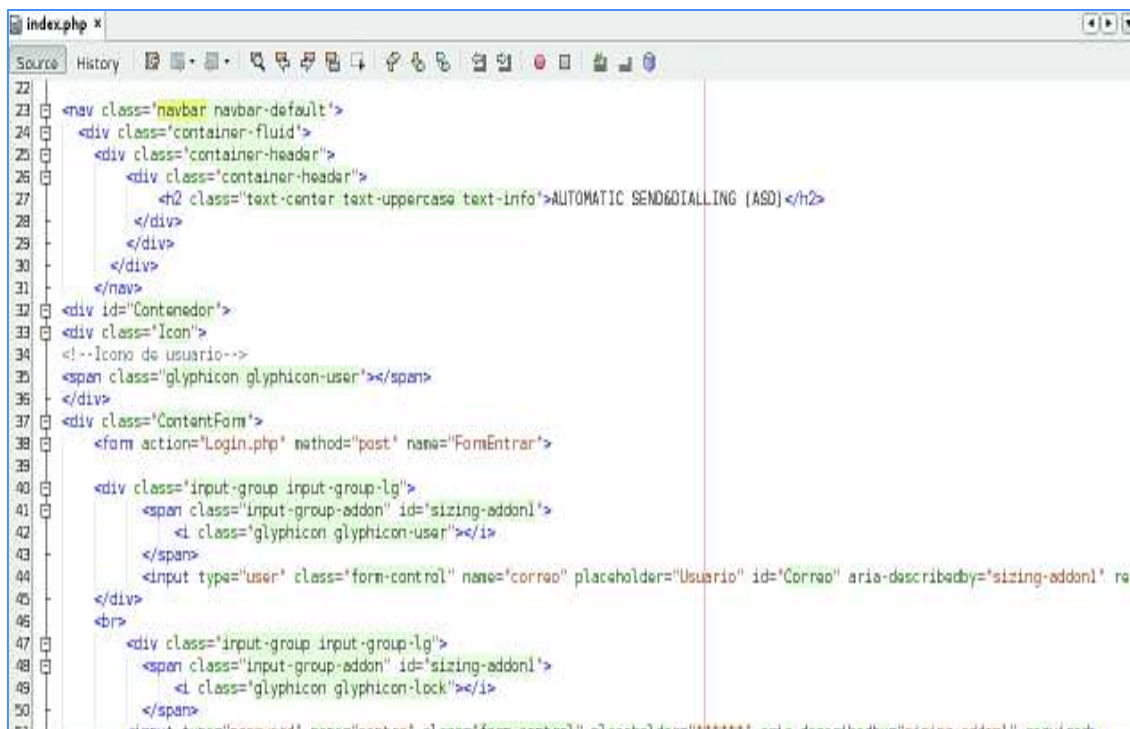
- **Ingreso al sistema (Login.php):** Para permitir el ingreso al sistema se crea el archivo Login.php, aquí se valida los datos obtenidos del formulario de entrada realizando la consulta a la base de datos. En el **Código 3.3** permite o no el ingreso al menú del sistema.

```

<?php
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
class DataBase
{
    function connectDB()
    {
        $user="user=postgres";
        $password="password=T3s1s";
        $dbname="dbname=automatic";
        //$dbname="dbname=automatic";
        $port="port=5432";
        $host="host=192.168.0.218";
        //$host="host=192.168.0.215";
        $adapter = pg_connect("$host $port $dbname $user $password");
        if(!$adapter){
            echo "Error en la Conexion:". pg_last_error;
        }
        return $adapter;
    }
}
?>

```

Código 3.1 Archivo para conexión con la base de datos (DataBase.php).



```

22
23 <nav class="navbar navbar-default">
24   <div class="container-fluid">
25     <div class="container-header">
26       <div class="container-header">
27         <h2 class="text-center text-uppercase text-info">AUTOMATIC SEND&DIALING (ASD) </h2>
28       </div>
29     </div>
30   </div>
31 </nav>
32 <div id="Contenedor">
33   <div class="icon">
34     <!-- Icono de usuario -->
35     <span class="glyphicon glyphicon-user"></span>
36   </div>
37   <div class="ContentForm">
38     <form action="Login.php" method="post" name="FormEntrar">
39
40     <div class="input-group input-group-lg">
41       <span class="input-group-addon" id="sizing-addon1">
42         <i class="glyphicon glyphicon-user"></i>
43       </span>
44       <input type="text" class="form-control" name="correo" placeholder="Usuario" id="Correo" aria-describedby="sizing-addon1" req
45     </div>
46     <br>
47     <div class="input-group input-group-lg">
48       <span class="input-group-addon" id="sizing-addon1">
49         <i class="glyphicon glyphicon-lock"></i>
50       </span>
51       <input type="password" name="contra" class="form-control" placeholder="*****" id="password" aria-describedby="sizing-addon1" require

```

Código 3.2 Archivo para formulario de entrada (Index.php).

```

dex.php x Login.php x
History
<title>Prototipo AUTOMÁTICO</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href=" ../tesis/css/bootstrap.min.css">
<link rel="stylesheet" type="text/css" href="Tesis.css">
<script src="jquery.min.js"></script>
<script src="bootstrap.min.js"></script>
</head>
<?php //logging
    $usermana = $_POST['correo'];
    $password = $_POST['contra'];
    require_once ('DataBase.php');
    $conectar= DataBase::connectDB();
    //consulta a la base de datos verificación que existe los usuarios
    $consulta="SELECT * from private.sys_users where usermana='$usermana' and password='$password'";
    $resultado=pg_query($conectar, $consulta);

    $filas=pg_num_rows($resultado);

    if ($filas>0)
    {
        header ("location:Menu.php");
    }
    else
    {
        echo "Error en la autentificación";
    }
    pg_free_result($resultado);
    pg_close($conectar);

```

Código 3.3 Archivo para ingreso al sistema (Login.php).

- **Menú de la aplicación (Menu.php):** El formulario del menú de la aplicación se implementa en el archivo **Menu.php**. Este presenta un menú que da acceso a las distintas funcionalidades del prototipo es decir permite el acceso a las Llamadas automáticas, Mensajes de texto y Estadísticas, cada uno con sus respectivos submenús como se muestra en el **Código 3.4**.

- **Llamadas Estáticas (Llamadas_pregrabadas.php):** El menú **Llamadas Estáticas** se implementa en el archivo **Llamadas_pregrabadas.php** mediante el **Código 3.5**. En el archivo se realiza una consulta a la base de datos presentando una lista de las campañas de llamadas estáticas existentes en el sistema.

Adicionalmente esta consulta muestra varias características propias de las campañas como: nombre de la campaña, estado, troncal, descripción, fecha de creación. Además también presenta botones que permiten: eliminar, subir archivos que pueden ser de audio y de texto, botones de eventos (iniciar, detener o continuar) y estadísticas para la reportería de la campaña seleccionada.

```

php x Login.php x Menu.php x
History
</head>
<body><!-- Class Menu principal que muestra campañas automáticas, mensajes de texto, reportes -->
<nav class="navbar navbar-default">
<div class="container-fluid">
<div class="container-header">
<div class="container-header">
<h2 class="text-center text-uppercase text-info">AUTOMATIC SENDSDIALING (ASD)</h2>
</div>
<div>
<a class="navbar-brand" href="#">WebSiteEvo</a>
</div>
<div>
<ul class="nav navbar-nav nav-tabs">
<li class="dropdown">
<a class="dropdown-toggle" data-toggle="dropdown">LLAMADAS AUTOMÁTICAS
<span class="caret"></span>
</a>
<ul class="dropdown-menu" role="menu" aria-labelledby="dropdownMenuDivider">
<li><a href="Llamadas_pregrabadas.php" class="text-uppercase">Campañas de Llamadas Estéticas</a></li>
<li role="presentation" class="divider"></li>
<li><a href="Llamadas_construidas.php" class="text-uppercase">Campañas de Llamadas Personalizado</a></li>
<li role="presentation" class="divider"></li>
<li><a href="btn_CrearCampaña" class="text-uppercase" target="VistaCrearCampañaLlamadas.php" onClick="window.opn">
</li>
</ul>
</li>
<li class="dropdown">
<a class="dropdown-toggle" data-toggle="dropdown">MENSAJES DE TEXTO (SMS)
<span class="caret"></span>
</a>
<ul class="dropdown-menu">
<li><a href="Mensajes_texto.php" class="text-uppercase">CAMPAÑAS DE MENSAJES DE TEXTO (SMS)</a></li>
<li role="presentation" class="divider"></li>
<li><a href="btn_CrearCampaña" class="text-uppercase" target="VistaCrearCampañaSms.php" onClick="window.opn">
</li>
</ul>
</li>
<li class="dropdown">
<a class="dropdown-toggle" data-toggle="dropdown">REPORTES Y REGISTROS
<span class="caret"></span>
</a>
<ul class="dropdown-menu">
<li><a href="Llamadas_pregrabadas.php" class="text-uppercase">Campañas de Llamadas Audios Pregrabados</a>
<li role="presentation" class="divider"></li>
<li><a href="Llamadas_construidas.php" class="text-uppercase">Campañas de Llamadas Audios Personalizados</a>
<li role="presentation" class="divider"></li>
<li><a href="Mensajes_texto.php" class="text-uppercase">Campañas de Mensajes de texto Sns</a> </li>
</ul>
</li>
</ul>
</div>
</div>
</div>
</nav>
</body>

```

Código 3.4 Menú de funcionalidades del sistema (Menu.php).


```

Login.php x | Menu.php x | Llamadas_pregrabadas.php x
y
<?php //Class Llamadas pregrabadas o llamadas estáticas.
require_once ('DataBase.php');
$connector= DataBase::connectDB();
$id=$row['id'];

$data = '<div class="panel " >
    <div class="panel-heading text-uppercase" >Lista de Campañas Audio Pregrabados</div>
</div>
<div>
<h3> </h3>
</div>
<table class="table table-bordered table-striped" style="background-color: white">
    <tr>
        <th>Nombre d</th>
        <th>Nombre de la Campaña</th>
        <th>Canal</th>
        <th>troncal</th>
        <th>Fecha de Creación</th>
        <th>Descripción</th>
        <th>Estado</th>
        <th>Borrar</th>
        <th>Subir Archivos</th>
        <th>Eventos</th>
        <th>Estadísticas</th>
    </tr>'; //consulta de las campañas estáticas existentes
$query="SELECT * from private.coc_callmatic_campaigns where campaigntype='prerecorded' order by created desc" ;
if (!$result= pg_query($connector, $query)) {
    exit(psql_error($connector));
}

// if query results contains rows then featch those rows
if(pg_num_rows($result) > 0)
{
    $number = 1;
    while($row = pg_fetch_assoc($result))
    {
        ?>
        <script> var var1 = "<?php echo $row['id']; ?>" </script>

        <?php
        $data .= ' <div class="container-fluid " >
        <tr>
            <td>.$row['id']. ' </td>
            <td>.$row['name']. ' </td>
            <td>.$row['channels']. ' </td>
            <td>.$row['trunk']. ' </td>
            <td>.$row['created']. ' </td>
            <td>.$row['description']. ' </td>
            <td class="text-uppercase">.$row['state']==0?'No Iniciada':($row['state']==1?'Iniciada':Finalizada)). ' </td>
            <td>
                <input type="button" class="dropdown-toggle text-danger text-uppercase" class="btn btn-danger btn-lg" data-toggle

        /td>
        <td>
        <li class="btn btn-default dropdown">
            <a class="dropdown-toggle text-info" data-toggle="dropdown" >ARCHIVOS
                <span class="caret"></span>
            </a>
            <ul class="dropdown-menu">
                <li><a class="btn btn-primary" href="VistaUploadaudiollamadas.php?id=.$row['id'].'">Audio</a></li>
                <li><a class="btn btn-success" href="VistaUploadcsvContactos.php?id=.$row['id'].'">Contactos</a>
            </li>
            </ul>
        </li>
        <td>
        <li class="btn btn-default dropdown">
            <a class="dropdown-toggle text-success" data-toggle="dropdown" >EVENTOS
                <span class="caret"></span>
            </a>
            <ul class="dropdown-menu">
                <li> <a class="btn btn-primary" href="IniciarCampaña.php?id=.$row['id']. '&type=.$row['campaigntype']. '&state=.$row['state']. '" >
                <li> <a class="btn btn-danger" href="StopCampaña.php?id=.$row['id']. '" class="text-uppercase text-info" >Detener</a> </li>
                <li> <a class="btn btn-warning" href="IniciarCampaña.php?id=.$row['id']. '&type=.$row['campaigntype']. '&state=.$row['state']. '" >
            </ul>
        </li>
        </td>
        <td>
        <li id="estadísticas" class="btn btn-default dropdown text-uppercase">
            <a class="dropdown-toggle text-warning" onclick="desactiva()" href="Progreso pregrabadas.php?id=.$row['id']. '">Estadísticas</a>

```

Código 3.5 Lista de las llamadas estáticas (Llamadas_pregrabadas.php).

- **Llamadas Personalizadas (Llamadas_construidas.php):** El menú **Llamadas Personalizadas** se implementa en el archivo **Llamadas_construidas.php** mediante el **Código 3.6**. En el archivo **Llamadas_construidas.php** se realiza una consulta a la base de datos obteniendo una lista de las campañas de llamadas personalizadas existentes en el sistema. También de la consulta realizada a la base de datos se muestra varias características propias de las campañas como: nombre de la campaña, estado, troncal, descripción, fecha de creación. Además también presenta botones que permiten: eliminar, subir archivos de texto, botones de eventos (iniciar, detener o continuar) y estadísticas para la reportería de la campaña seleccionada.

- **Mensajes de Texto (Mensajes_texto.php):** El menú **Mensajes de Texto** se implementa en el archivo **Mensajes_texto.php** mediante el **Código 3.7**. En el archivo **Mensajes_texto.php** se realiza una consulta a la base de datos obteniendo una lista de las campañas de mensajes de texto existentes en el sistema. Al igual que las campañas anteriormente mencionadas también muestra varias características propias de la campaña como botones que permiten eliminar la campaña, subir archivos de texto, botones de eventos (iniciar, detener o continuar) y estadísticas para la reportería de la campaña seleccionada.

- **Creación de campañas de llamadas automáticas estáticas y personalizadas:** El formulario para la creación de campañas de llamadas automáticas se implementa en el archivo **Vistacreacióncampañasllamadas.php** mediante el **Código 3.8**, el usuario ingresa los datos de la campaña que va a crear. Adicionalmente se crea el archivo **Creacióncampañasllamadas.php** el cual toma los datos ingresados por el usuario mediante el **Código 3.9**.

```

Login.php x Menu.php x Llamadas_pregrabadas.php x Llamadas_construidas.php x
ory
</div>
<?php//Class Llamadas construidas o llamadas personalizadas.
require_once ('DataBase.php');
$connectar= DataBase::connectDB();
$data = '<div class="panel " >
    <div class="panel-heading text-uppercase" >Lista de Campañas Audio Personalizados</div>
</div>
<div>
<h3> </h3>
</div>
<table class="table table-bordered table-striped" style="background-color: white">
    <tr>
    <!--<th> </th-->
    <th>Nombre de la Campaña</th>
    <th>Canal</th>
    <th>Troncal</th>
    <th>Descripcion</th>
    <th>Estado</th>
    <th>Borrar</th>
    <th>Subir Contactos</th>
    <th>Eventos</th>
    <th>Estadísticas</th>
    </tr>';//consulta de las campañas personalizadas existentes
$query="SELECT * from private.coc.callmatic_campaigns where campaigntype='built' order by created desc" ;
if (!$result= pg_query($connectar, $query)) {
    exit(mysql_error($connectar));
}
if(pg_num_rows($result) > 0)
{
    $number = 1;
    while($row = pg_fetch_assoc($result))
    {
        $data .= '<div class="container-fluid class="checkbox">
            <tr>
            <!--<td><input type="checkbox" value="red" . $row['id'] .'/></td-->
            <td>.$row['name'].</td>
            <td>.$row['channels'].</td>
            <td>.$row['trunk'].</td>
            <td>.$row['description'].</td>
            <td class="text-uppercase" >'.($row['state']=='0'?'No Iniciada':($row['state']=='1'?'Iniciada':'Finalizada')).</td>
            <td>
            <li class="btn btn-default dropdown">
                <a class="dropdown-toggle text-danger text-uppercase" href="VistaDeletecampaign.php?id=.$row['id'].&type=.$row['can
            </li>
            </td>
            <td><li class="btn btn-default dropdown">
                <a class="dropdown-toggle text-success text-uppercase" href="VistaUploadsvContactostts.php?id=.$row['id'].&type=.$row['
            </li>
            </td>
            <td>
            <!--<a class="btn btn-primary" href="IniciarCampaña.php?id=.$row['id'].&type=.$row['campaigntype'].&state=.$row['

```

Código 3.6 Lista de las llamadas personalizadas (Llamadas_construidas.php).


```

<td>Nombre: </td>
<td><input type="text" name="name"/></td>
</tr>
<tr>
<td>Troncal:</td>
<td><input type="text" name="trunk"/></td>
</tr>
<tr>
<td>Tipo:</td>
<td>
<select name="type">
<option value="vacio">-----</option>
<option value="built">Personalizado</option>
<option value="prerecorded">Pregrabado</option>
</select>
</td>
</tr>
<tr>
<td>Canales: </td>
<td><input type="text" name="channels"/></td>
</tr>
<tr>
<td>Número de intentos: </td>
<td><input type="text" name="maxretries"/></td>
</tr>
<tr>
<td>Tiempo de espera entre intentos: </td>
<td><input type="text" name="retrytime"/></td>
</tr>
<tr>
<td>Descripción: </td>
<td><input type="text" name="description"/></td>

```

Código 3.8 Formulario para la creación de campañas(Vistacreacióncampaignllamadas.php).

```

//Creación de Campañas \llamadas automáticas
require_once ('DataBase.php');
$connectar= DataBase::connectDB();

$name= $_POST['name'];
$channels= $_POST['channels'];
$type= $_POST['type'];
$trunk= $_POST['trunk'];
$maxretries= $_POST['maxretries'];
$retrytime= $_POST['retrytime'];
$description = $_POST['description'];

if($name!=''){
//consulta creación de campañas automáticas
$query="SELECT * FROM public.coc_callmatic_createcampaign_retry('$name','$channels','$type','$trunk','$maxretries','$retrytime','$description')";
$resultado=pg_query($connectar, $query);
echo '
<div class="container-header">
<h3 class="text-center text-uppercase">Creación exitosa</h3>
</div>
<div class="container-fluid">
<td>
<button name="btn_Regresar" onclick="cerrarse()">Salir</button>
</td>';
}

echo '
<div class="container-header">
<h3 class="text-center text-uppercase">Error campos Vacios</h3>
</div>
<div class="container-fluid">
<td>
<button name="btn_Regresar" onclick="cerrarse()">Cerrar</button>
</td>';

```

Código 3.9 Creación campañas automáticas (Crearcampaignllamadas.php).

- **Creación de campañas de mensajes de texto:** El formulario para la creación de campañas de mensajes de texto se implementa en el archivo **Vistacreacióncampañassms.php** mediante el **Código 3.10**, aquí el usuario ingresa los datos de la campaña que va a crear. Esta creación de campañas se diferencia de la creación de campañas de llamadas automáticas debido a que se debe ingresar el usuario y la contraseña configurados previamente en el *gateway* GSM en el módulo **SMS-> SMS Settings-> HTTP a SMS**. Adicionalmente se crea el archivo **CreacióncampañasSms.php** el cual toma los datos ingresados por el usuario mediante el **Código 3.11**.

```

<!-- Class formulario Creación campañas sms-->
<div class="container-header">
<table id="tableUser" class="table table-striped">
<tr>
<td>Nombre: </td>
<td><input type="text" name="name"/></td>
</tr>
<tr>
<td>Troncal: </td>
<td><input type="text" name="trunk"/></td>
</tr>
<tr>
<td>Usuario: </td>
<td><input type="text" name="user"/></td>
</tr>
<tr>
<td>Contraseña: </td>
<td><input type="text" name="password"/></td>
</tr>
<tr>
<td>Canales: </td>
<td><input type="text" name="channels"/></td>
</tr>
<tr>
<td>Descripcion: </td>
<td><input type="text" name="description"/></td>
</tr>
<tr>
<td>

```

Código 3.10 Formulario para la creación de campañas de mensajes de texto (Vistacreacióncampañassms.php).

- **Cargar archivos:** El formulario para la cargar el archivo de audio al sistema se implementa en **VistaUploadaudiollamadas.php** mediante el **Código 3.12**. Para cargar el archivo de texto al sistema se implementa el archivo **VistaUploadcsvContactos.php** mediante el **Código 3.13**. Se crea el archivo **UploadAudioPregrabadas.php** para validar el archivo ingresado en **VistaUploadaudiollamadas.php** esto lo realiza mediante el **Código 3.14**. Adicionalmente se crea el archivo **UploadcsvContactos.php** para validar el

archivo ingresado en **VistaUplocsvContactos.php** esto lo realiza mediante el **Código 3.15**.

```

//Creación de campañas de mensajes de texto
require_once ('DataBase.php');
$connectar= DataBase::connectDB();

$name= $_POST['name'];
$channels= $_POST['channels'];
$trunk= $_POST['trunk'];
$trunkuser= $_POST['user'];
$trunkpasswd= $_POST['passwd'];
$description = $_POST['description'];

if($name!=''){

$query="SELECT * FROM public.coc_smsmatic_createcampaign('$name', '$channels', '$trunk', '$trunkuser', '$trunkpasswd', '$description')";
$resultado=pg_query($connectar, $query);
echo 'exito';
<div class="container-fluid">
<td>
<button name="btn_Regresar" onclick="cerrarse()" >Regresar</button>
</td>
</div>
}
else {
echo 'error campos vacíos de texto';
}
?>

```

Código 3.11 Creación campañas de mensajes de texto (CreacióncampaignSms.php).

```

<div class="container-fluid">
<div>
<button name="btn_Regresar" onclick="location.href = 'llamadas_prograbadas.php'" >Regresar</button>
</div>
</div>

<?php //Clase Subir archivo de audio llamadas estáticas.
require_once ('DataBase.php');
$connectar= DataBase::connectDB();

$a_id=$_GET['id'];

//var_dump($a_id);

?>

<form enctype="multipart/form-data" action="UploadAudioPrograbado.php?id=<?php echo $a_id ?>" method="POST">

<br>
<input type="hidden" name="MAX_FILE_SIZE"/>
<div class="alert alert-info">Seleccionar un archivo de audio: </div>

<input name="file" type="file"/>
<br>

<input type="submit" style="color: blue" value="Subir Archivo" />

</form>
</div>
</body>

```

Código 3.12 Formulario para cargar archivos de audio (VistaUploadaudiollamadas.php).

```

<div>
<button name="btn_Regresar" onclick="location.href='Llamadas_pregrabadas.php'" >Regresar</button>
</div>
</div>
<div class="container-fluid">
<?php//Class Subir archivo de Texto Llamadas estáticas.
require_once ('DataBase.php');
$connectar= DataBase::connectDB();

    $c_id=$_GET['id'];
?>

<form enctype='multipart/form-data' action="UploadCsvContactos.php?id=<?php echo $c_id ?>" method="POST">

<br>
<input type="hidden" name="MAX_FILE_SIZE"/>
Seleccionar un archivo de Texto:
<br>
<br>
<input name="file" type="file"/>
<br>

    <input type="submit" style="color: blue" value="Subir Archivo" />

</form>
</div>
</body>

```

Código 3.13 Formulario para cargar archivos de texto (VistaUploadcsvContactos.php)

```

regrabadas.php x Progreso_construidas.php x Progreso_mensajestexto.php x UploadAudioPregrabado.php x UploadCsvContactos.php x
tory
    echo "Return Code: " . $_FILES["file"]["error"] . "<br />";
}
else
{
    echo '<div class="alert alert-info">Información del archivo subido </div>';
    // echo "Upload: " . $_FILES["file"]["name"] . "<br />";
    echo "Nombre del archivo: " . $audio_name . "<br />";
    echo "Tamaño del archivo: " . ($_FILES["file"]["size"] / 1024) . " Kb<br />";
    echo "Ruta del archivo: " . $_FILES["file"]["tmp_name"] . "<br />";
    //echo "name: '$audio_id'<br />";
    var_dump($audio_id);

    $consulta="select * from public.automatic_setmessage( $audio_id,'/var/www/contactvox-temp/$audio_name')";
    $resultado=pg_query($connectar, $consulta);

    if (file_exists("/var/www/contactvox-temp/" . $_FILES["file"]["name"]))
    {
        echo '<div class="alert alert-info">Información existencia del archivo subido </div>';
        echo " Ya existe el archivo: " . $_FILES["file"]["name"] ;
    }
    else
    {
        move_uploaded_file($_FILES["file"]["tmp_name"], "/var/www/contactvox-temp/" . $_FILES["file"]["name"]);
        echo "Almacenado en: " . "/var/www/contactvox-temp/" . $_FILES["file"]["name"];
    }
}
}
else

```

Código 3.14 Cargar archivo de audio (UploadAudioPregrabadas.php)


```

}
else
{
    echo "Upload: " . $_FILES["file"]["name"] . "<br />";
    echo "Type: " . $_FILES["file"]["type"] . "<br />";
    echo "Size: " . ($_FILES["file"]["size"] / 1024) . " kb<br />";
    echo "Temp file: " . $_FILES["file"]["tmp_name"] . "<br />";
    echo "name: '$audio_id'<br />";

    $consulta='select * from public.automatic_setcontacts('$csv_id', '/var/www/contactvox-temp/$csv_name')';
    $resultado=pq_query($connectar, $consulta);

    echo ".$consulta.";
    if (file_exists('/var/www/contactvox-temp/' . $_FILES["file"]["name"]))
    {
        echo $_FILES["file"]["name"] . " already exists. ";
    }
    else
    {
        move_uploaded_file($_FILES["file"]["tmp_name"], "/var/www/contactvox-temp/" . $_FILES["file"]["name"]);
        echo "Stored in: " . "/var/www/contactvox-temp/" . $_FILES["file"]["name"];
        var_dump($audio_id);
    }
}
}
else
{
    echo "Invalid file";
}
}

```

Código 3.15 Cargar archivo de texto (UploadCsvcontactos.php)

- **Inicio de campañas automáticas (campañas estáticas y pregrabadas):** El inicio de campañas de llamadas automáticas se implementa en el archivo **Iniciocampañasllamadas.php** mediante el **Código 3.16**. En este archivo se envían los parámetros de la campaña seleccionada a la función **public.automatic_start** creada en la base de datos; ésta función enviará la información hacia Asterisk y al sistema operativo para que se inicie las llamadas automáticas.

- **Inicio de campañas de mensajes de texto:** El inicio de campañas de mensajes de texto se implementa el archivo **IniciarcampaignSms.php** mediante el **Código 3.17**. En este archivo se envían los parámetros de la campaña seleccionada a la función **public.sms_start** creada en la base de datos; esta función enviará la información necesaria hacia Asterisk y al sistema operativo para que se inicie el envío instantáneo de mensajes de texto.

- **Detener las campañas automáticas (campañas estáticas y pregrabadas):** Para detener las campañas de llamadas automáticas se implementa el archivo **Stopcampaignllamadas.php** mediante el **Código 3.18**. En este archivo se envían los parámetros de la campaña seleccionada a la función **public.automatic_stop** creada en la base de datos; ésta función enviará la información hacia Asterisk y al sistema operativo para que detenga las llamadas automáticas.

- **Detener las campañas mensajes de texto:** Para detener las campañas automáticas se implementa el archivo **StopcampaignSms.php** mediante el **Código 3.19**. En este archivo se envían los parámetros de la campaña de mensajes de texto seleccionada a la función **public.sms_stop** creada en la base de datos; ésta función enviará la información hacia Asterisk y al sistema operativo para que detenga el envío de mensajes instantáneos.

```

//Class Iniciar Llamadas estáticas y personalizadas.
require_once ('DataBase.php');
$connectar= DataBase::connectDB();

$campc_id=$_REQUEST['id'];
$campc_type=$_REQUEST['type'];
$campc_state=$_REQUEST['state'];

if($campc_state!=2){

$consulta="select * from public.automatic_start('$campc_id','$campc_type')";
$resultado=pg_query($connectar, $consulta);
echo ' se ha iniciado la campaña
<div class="container-fluid">
  <tr>
    <td>
      <a class="btn btn-warning" href="Llamadas_total.php">Regresar</a>
    </td>
  </tr>';
}

else {
  echo 'error no se ha iniciado la campaña';
}
?>

```

Código 3.16 Inicio de campañas de llamadas automáticas (Iniciocampaignllamadas.php).

```

*/
//Class Iniciar campañas de mensajes de Texto
require_once ('DataBase.php');
$connectar= DataBase::connectDB();

$campc_id=$_REQUEST['id'];

$campc_state=$_REQUEST['state'];

if($campc_state!=2){

$consulta="select * from public.sms_start('$campc_id')";
$resultado=pg_query($connectar, $consulta);
echo ' se ha iniciado la de mensajes de texto
<div class="container-fluid">
  <tr>

  <td>
    <a class="btn btn-warning" href="Mensajes_texto.php">Regresar</a>
  </td>
  <tr>';
}

else {
    echo 'error no se ha iniciado la campaña';
}
?>

```

Código 3.17 Inicio de campañas de mensajes de texto (IniciarcampaignSms.php).

```

<?php
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
//Class Detener campañas llamadas personalizadas o estáticas
require_once ('DataBase.php');
$connectar= DataBase::connectDB();

$campc_id=$_REQUEST['id'];

$campc_state=$_REQUEST['state'];

$consulta="select * from public.AUTOMATIC_stop('$campc_id')";
$resultado=pg_query($connectar, $consulta);
echo ' se ha detenido la campaña
  <div class="container-fluid">
    <tr>

    <td>
      <a class="btn btn-primary" href="Llamadas_total.php">Regresar</a>
    </td>
    <tr>';

```

Código 3.18 Detener las campañas de llamadas automáticas (Stopcampaignllamadas.php).

```

</php
/*...5 lines...*/
//Class Detener campañas de mensajes de texto.
require_once ('DataBase.php');
$connectar= DataBase::connectDB();

$campc_id=$_REQUEST['id'];

$campc_state=$_REQUEST['state'];

$consulta='select * from public.sms_stop( $campc_id)';
$resultado=pg_query($connectar, $consulta);
echo ' se ha detenido la campaña
<div class="container-fluid">
<tr>
<td>
<a class="btn btn-primary" href="Llamadas_total.php">Regresar</a>
</td>
</tr>';
?>

```

Código 3.19 Detener las campañas de mensajes de texto (StopcampaignSms.php).

- **Borrar campañas de Mensajes de Texto (DeletCampaignSms.php):**

Para borrar las campañas de mensajes de texto se implementa el archivo **DeletCampaignSms.php** mediante el **Código 3.20**. En este archivo se envían los parámetros de la campaña seleccionada a la función **public.sms_delcampaign** para eliminar la campaña de la base de datos.

```

require_once ('DataBase.php');
$connectar= DataBase::connectDB();

$campc_id=$_REQUEST['id'];
$campc_type=$_REQUEST['type'];

if($campc_id!=''){

$consulta="select * from public.coc_smsmatic_delcampaign ($campc_id)";
$resultado=pg_query($connectar, $consulta);

echo '<div class="alert alert-warning">Esta campaña ha sido borrada</div>

<div class="container-fluid">
<tr>
<td>
<a class="btn btn-default" href="Mensajes_texto.php">Regresar</a>
</td>
</tr>';

}

else {
echo 'error NO existe esa Campaña';
}
?>

```

Código 3.20 Borrar las campañas de mensajes de texto (DeletCampaignSms.php).

- **Borrar campañas de llamadas automáticas (estáticas y personalizadas) (DeletCampaign.php):** Para borrar las campañas de llamadas automáticas se implementa el archivo **DeletCampaign.php** mediante el **Código 3.21**. En este archivo se envían los parámetros de la campaña seleccionada a la función **public.automatic_delcampaign** para eliminar la campaña de la base de datos.

- **Estadísticas campañas estáticas:** El formulario para presentar las estadísticas de la campaña de llamadas estáticas se implementa en el archivo **Progreso_pregradabas.php** mediante el **Código 3.22**. Este archivo presenta un reporte gráfico de barras por estado de contestación de las llamadas, un reporte gráfico tipo pastel en porcentajes del estado de contestación de las llamadas, el audio con el que se reprodujo, los contactos que se han ingresado, y el registro final de la campaña, es decir el detalle de cada llamada realizada.

- **Estadísticas campañas personalizadas:** El formulario para presentar las estadísticas de la campaña de llamadas personalizadas se implementa en el archivo **Progreso_constriudas.php** mediante el **Código 3.23**. Este archivo presenta un reporte gráfico de barras por estado de contestación de las llamadas, un reporte gráfico tipo pastel en porcentajes del estado de contestación de las llamadas, los contactos que se han ingresado, y el registro final de la campaña, es decir el detalle de cada llamada realizada.

- **Estadísticas campañas mensajes de texto:** El formulario para presentar las estadísticas de las campañas de mensajes de texto se implementa en el archivo **Progreso_mensajestexto.php** mediante el **Código 3.24**. Este archivo presenta un reporte gráfico de barras por estado de respuesta de los mensajes de texto, un reporte gráfico tipo pastel en porcentajes de la respuesta de los mensajes de texto, los contactos que se han ingresado a esta campaña, y el registro final de la campaña, es decir el detalle de cada mensaje enviado.

```

<?php
require_once ('DataBase.php');
$connectar= DataBase::connectDB();

$campe_id=$_REQUEST['id'];
$campe_type=$_REQUEST['type'];
var_dump($campe_id);

if($campe_id!=''){
    $consulta="select * from public.coc_callmatic_delcampaign ($campe_id)";
    $resultado=pg_query($connectar, $consulta);

    echo '
        <div class="alert alert-warning">Esta campaña ha sido borrada</div>

        <div class="container-fluid">
            <tr>

                <td>
                    <a class="btn btn-default" href="Llamadas_pregrabadas.php">Regresar</a>
                    <!--<button name="btn_Regresar" onClick="cerrarse()" >Regresar</button>
                </td>
            </tr>';
        }
    else {
        echo 'error No existe esa Campaña';
    }
}

```

Código 3.21 Borrar las campañas de llamadas automáticas (DeletCampaign.php).

```

<?php //Clase Estadísticas de campañas de llamadas estáticas.

require_once ('DataBase.php');
$connectar= DataBase::connectDB();
$campe_id=$_REQUEST['id'];
//Reporte de llamadas personalizadas
$datareport2 = '<div>
    <tr>
        <div class="container-fluid">
            <tr>
                <div class="panel panel-info">
                    <th >h5 class="text-center text-uppercase" style="color: blue">Reporte Llamadas Audio Pregrabado</h5></th>
                </div> </tr>
            </div>
        </tr>

        <table class="table table-bordered table-striped" style="background: lavender">

            <tr>
                <th>Estado de la llamada</th>
                <th>Porcentajes %</th>
            </tr>
        </div>
    </tr>';

$queryB='select * from public.coc_callmatic_getreport_p($campe_id) ORDER BY countp desc;';

if (!$result= pg_query($connectar, $queryB)) {
    exit(mysql_error($connectar));
}

```

Código 3.22 Formulario de estadísticas de campañas de llamadas estáticas (Progreso pregrabadas.php).

```

<?php //Class Estadísticas de campañas de llamadas personalizadas.

require_once ('DataBase.php');
$connector= DataBase::connectDB();

$campe_id=$REQUEST['id'];
//Reporte de llamadas estáticas
$datareport2 = '<div>
    <tr>
    <div class="container-fluid">
        <tr>
            <div class="panel panel-info">
                <th ><h5 class="text-center text-uppercase" style="color: blue">Reporte Llamadas Audio Pregrabado</h5></th>
            </div> </tr>
        </div> </tr>

        <table class="table table-bordered table-striped" style="background: lavender">

            <tr>
                <th>Estado de la llamada</th>
                <th>Porcentajes %</th>
            </tr>
        </div>
    </tr>
';

$query8="select * from public.coc_callmatic_getreport_c($campe_id) ORDER BY countp desc;";

if (!$result= pg_query($connector, $query8)) {

```

Código 3.23 Formulario de estadísticas de campañas de llamadas personalizadas (Progreso construidas.php).

```

</php

require_once ('DataBase.php');
$connector= DataBase::connectDB();

$campe_id=$REQUEST['id'];

$datareport = '<div>
    <tr>
    <div class="container-fluid">
        <tr>
            <div class="panel panel-info">
                <th ><h5 class="text-center text-uppercase" style="color: blue">Reporte de Mensajes de texto</h5></th>
            </div> </tr>
        </div> </tr>

        <table class="table table-bordered table-striped" style="background: lavender">

            <tr>
                <th class="text-center text-uppercase" >Estado del Mensaje</th>
                <th class="text-center text-uppercase" >Total</th>
            </tr>
        </div>
    </tr>
';

$query2="SELECT result, count(*) from public.coc_smsmatic_getregistry($campe_id) group by result";

if (!$result= pg_query($connector, $query2)) {
    exit(html_error($connector));
}

```

Código 3.24 Formulario de estadísticas de campañas de mensajes de texto (Progreso Mensajes de Texto.php)

3.1.5 IMPLEMENTACIÓN DE NIVEL PRESENTACIÓN

En el nivel de presentación de la aplicación para el prototipo se crearon los formularios para que el usuario pueda administrar el prototipo desde la interfaz gráfica. Las interfaces del prototipo pueden verse en la **sección 2.5**.

Para el estilo de las interfaces se utilizó las librerías de *bootstrap*³⁶ de forma gratuita ya que es de código abierto. También se utiliza links de librerías de internet para archivos ssc³⁷. Parte de los *scripts* de algunos archivos se presentan en el **Anexo C**. El código completo de la aplicación web y de la base de datos se adjunta en el respaldo digital.

The screenshot shows a web browser window titled 'Prototipo AUTOMÁTIC - Google Chrome'. The address bar shows 'No es seguro | 192.168.100.23/tesis/VistaCrearcamp...'. The main content area displays the title 'AUTOMATIC SEND&DIALLING (ASD)' and the subtitle 'Creación de Campañas Llamadas Automáticas'. Below this is a form with the following fields and values:

Nombre:	Campaña Diciembre 3
Troncal:	192.168.0.12
Tipo:	Personalizado ▼
Canales:	3
Número de intentos:	1
Tiempo de espera entre intentos:	4
Descripción:	Campaña prueba 3 diciemb

At the bottom of the form are two buttons: 'Salir' and 'Crear'.

Figura 3.18 Interfaz Creación Campaña de Llamadas Personalizadas.

³⁶ *Bootstrap*: es un esquema o un patrón desarrollado por Twitter para facilitar el diseño web, permite crear de forma aceptable y sencilla webs.

³⁷ *SSC Satellite Situation Center* permite a un desarrollador utilice partes software de SSC en su propia aplicación.

En la **Figura 3.18** muestra la interfaz de la creación de campañas de llamadas personalizadas. Para crear una campaña de llamadas personalizadas es necesario ingresar el nombre, troncal, tipo de campaña, canales, número de intentos, tiempo de espera entre intentos y descripción. Todos los campos son obligatorios a excepción del campo descripción.

Este formulario se implementa mediante el **Código 3.25**, el cual consta de un encabezado que muestra el nombre del prototipo, seguido del nombre del formulario y el cuerpo del mismo con los principales datos que se requieren para crear una campaña de llamadas automáticas, Finalmente los botones **Salir** y **Crear** ; el botón **Salir** no guarda ningún dato ingresado por el usuario y cierra el formulario: el botón **Crear** toma los datos ingresados por el usuario y los almacenará para enviarlos al formulario **Crearcampaignllamadas.php**.

```

</tr>
<td>Nombre: </td>
<td><input type="text" name="name"/></td>
</tr></tr>
<td>Troncal:</td>
<td><input type="text" pattern="^([1-9][0-9]{10-9}|2[9-4][0-9]{25(0-5)}|3[1-9][0-9]{10-9}|2[9-4][0-9]{25(0-5)}|4" name="trun
</tr></tr>
<td>Tipo:</td>
<td><select name="type">
<option value="vacío">----</option>
<option value="built">Personalizado</option>
<option value="prerecorded">Pregrabado</option>
</select></td>
</tr>
<tr>
<td>Canales: </td>
<td><input type="number" name="channels"/></td>
</tr>
<tr>
<td>Número de intentos: </td>
<td><input type="number" name="maxretries"/></td>
</tr>
<td>Tiempo de espera entre intentos: </td>
<td><input type="number" name="retrytime"/></td>
</tr>
<tr>
<td>Descripción: </td>
<td><input type="text" name="description"/></td>
</tr>
</tr>

```

Código 3.25 Formulario para la creación de campañas de llamadas automáticas
(VistacrearcampaignLlamadas.php)

Para el formulario de la creación de campañas se creó una tabla de dos columnas, la primera columna llevará los nombres de los campos a ingresar, estos campos son de tipo texto, la segunda columna pueden ser de tipo texto como el nombre y descripción o tipo numérico como el campo canal o troncal.

Como se mencionó anteriormente se utilizó *bootstrap* para el diseño del formulario y enlaces URL de internet para *bootstrap*.

3.2 PRUEBAS DE SISTEMA

El objetivo de las pruebas del sistema que establece la metodología XP es comprobar el correcto funcionamiento de la aplicación mediante una serie de pruebas. A continuación, se presenta las pruebas realizadas al prototipo con sus respectivos resultados.

3.2.1 PRUEBAS DE FUNCIONALIDAD

Para probar el funcionamiento del prototipo a nivel telefónico se realizan pruebas de llamadas automáticas personalizadas y envío de mensajes de texto.

Tabla 3.4 Ejemplo de Campaña personalizada

Teléfono	Mensaje a reproducir
0999703430	Artefacta te invita a usar tu crédito de 1133 dólares este fin de semana. Ven y visítanos ya!
0984252217	Artefacta te invita a usar tu crédito de 1133 dólares este fin de semana. Ven y visítanos ya!
0981686044	Artefacta te invita a usar tu crédito de 1133 dólares este fin de semana. Ven y visítanos ya!
0992831283	Artefacta te invita a usar tu crédito de 1133 dólares este fin de semana. Ven y visítanos ya!
0991774073	Artefacta te invita a usar tu crédito de 1133 dólares este fin de semana. Ven y visítanos ya!
0979341661	Artefacta te invita a usar tu crédito de 1133 dólares este fin de semana. Ven y visítanos ya!
0995216349	Artefacta te invita a usar tu crédito de 1133 dólares este fin de semana. Ven y visítanos ya!
0996326045	Artefacta te invita a usar tu crédito de 513 dólares este fin de semana. Ven y visítanos ya!

La **Tabla 3.4** es un ejemplo del archivo que se carga en una campaña personalizada el cual debe tener dos columnas. La primera columna debe ser

el número telefónico y la segunda columna debe ser el mensaje que se va a reproducir. El archivo que se debe cargar al sistema debe ser de formato csv³⁸ delimitado por comas y con codificación UTF8³⁹.

Tabla 3.5 Registro final de la campaña

Fecha	Hora	Teléfono	Duración	Estado	Mensaje
01/11/2017	15:57:03	999703430	12	ANSWERED	Artefacta te invita a usar tu crédito de 1525 dólares este fin de semana. Ven y visítanos ya!
01/11/2017	15:57:04	984252217	10	ANSWERED	Artefacta te invita a usar tu crédito de 1114 dólares este fin de semana. Ven y visítanos ya!
01/11/2017	15:57:56	981686044	11	ANSWERED	Artefacta te invita a usar tu crédito de 1949 dólares este fin de semana. Ven y visítanos ya!
01/11/2017	15:58:18	992831283	10	ANSWERED	Artefacta te invita a usar tu crédito de 872 dólares este fin de semana. Ven y visítanos ya!
01/11/2017	15:59:02	991774073	9	ANSWERED	Artefacta te invita a usar tu crédito de 932 dólares este fin de semana. Ven y visítanos ya!
01/11/2017	15:59:33	979341661	10	ANSWERED	Artefacta te invita a usar tu crédito de 1443 dólares este fin de semana. Ven y visítanos ya!
01/11/2017	15:59:48	995216345	5	FAILED	Artefacta te invita a usar tu crédito de 1133 dólares este fin de semana. Ven y visítanos ya!
01/11/2017	16:01:25	996326045	6	FAILED	Artefacta te invita a usar tu crédito de 513 dólares este fin de semana. Ven y visítanos ya!

En el **Anexo D**, se añaden los archivos de audio de las campañas con audio personalizado de prueba.

³⁸ CSV del inglés *comma separate values* o valores separados por comas.

³⁹ UTF8 *Unicode Transformation Format 8-bit*, es decir que codifica cada carácter Unicode como un número variable de 1 a 4 octetos[44].

En la **Tabla 3.5** se presenta un registro final de la campaña, el cual consta de fecha, hora, teléfono, duración en segundos, estado de la llamada y mensaje que se ha reproducido.

La **Tabla 3.6** presenta un ejemplo del archivo que se carga en una campaña de mensajes de texto. El archivo debe tener dos columnas. La primera columna debe ser el número telefónico y la segunda columna debe ser el mensaje que se va a enviar.

Tabla 3.6 Campaña de mensajes de texto

0999703430	Prueba de sms 3
0995216349	Prueba de sms 3
0996326045	Prueba de sms 3
0995216349	Prueba de sms 3
0998596067	Prueba de sms 3

La **Figura 3.19** muestra una captura de los mensajes de texto que fueron recibidos en un dispositivo móvil.



Figura 3.19 Captura de mensajes de texto en el celular.

La **Tabla 3.7** presenta un registro del envío de mensajes de texto de la campaña de prueba.

Tabla 3.7 Registro final de la campaña de mensajes de texto.

Fecha	hora	Teléfono	Estado	Mensaje
31/07/2017	11:18:03	0995216349	SUCCESS	Prueba de sms 3
31/07/2017	12:02:40	0995216349	SUCCESS	Prueba de sms 3
31/07/2017	12:19:13	0996326045	SUCCESS	Prueba de sms 3
31/07/2017	12:21:44	0999703430	ERROR	Prueba de sms 3
31/07/2017	12:28:28	0998596067	SENDING	Prueba de sms 3

3.2.2 PRUEBAS UNITARIAS DE LA APLICACIÓN

Para realizar las pruebas unitarias se elabora un *script* denominado **pruebas_unitarias.sql** ubicado en el directorio `/opt`. El *script* contiene las funciones que se utilizan en el prototipo.

El objetivo de este *script* es ejecutarlo directamente en la base de datos mediante el **Comando 3.5** y evaluar los resultados. Cuando una función o consulta no tiene los parámetros correctos o no esta creada correctamente, la base de datos ejecuta el comando y muestra la línea del error para corregirla.

De esta manera podemos verificar que las funciones y tablas creadas en la base de datos están correctas. En el **Anexo C** se muestra todo el contenido del *script* elaborado.

```
automatic=# \i /opt/pruebas_unitarias.sql
```

Comando 3.5 Ejemplo de ejecución de un archivo en la base de datos.

La **Figura 3.20** presenta resultados después de ejecutar el *script* para la creación de campañas mediante las funciones respectivas. Se crean campañas de llamadas automáticas primero personalizada, seguido de estática y finalmente de mensajes de texto.

La **Figura 3.21** presenta resultados después de ejecutar el *script* para la obtención de reportes de las campañas mediante las funciones respectivas.

```

automatic=# \i /opt/pruebas_unitarias.sql
automatic_createcampaign
-----
                                126
(1 row)

automatic_createcampaign
-----
                                127
(1 row)

automatic_createcampaign
-----
                                128
(1 row)

```

Figura 3.20 Pruebas_unitarias Ejecución de funciones crear campañas.

campaign	channels	state	progress	description	
eve_camp_b	2	2	100%	uioo	
(1 row)					
campaign	channels	state	campaigntype	progress	description
Demo PYCCA	1	2		100%	qwert
(1 row)					
campaignid	campaign	channels	state	progress	description
128	campaña prueba unitaria 16	4	0	0%	Prueba unitaria creación campaña 16
127	campaña prueba unitaria 15	3	0	0%	Prueba unitaria creación campaña 15
126	campaña prueba unitaria 14	2	0	0%	Prueba unitaria creación campaña 14
114	ssesse	2	0	0%	sadada
113	qqqqqqq	3	0	0%	adefasdfa
111	campaña 44444444	4	0	0%	456
71	luis campaña	2	0	0%	
69	campaña martes noche	4	0	0%	eve
67	Estacion-01 prueba2	3	0	0%	234456
66	Estaciones Calicenter123	2	0	0%	1233
65	Demo PYCCA	1	2	0%	qwert
63	Campaña cumpleaños prueba	3	0	0%	que
62	crearcampa9	2	0	0%	descrip
61	crearcampa8	2	2	0%	descrip
59	crearcampa6	2	0	0%	descrip
43	eve_camp_b	2	2	100%	uioo
37	prueba creacion	4	2	100%	
34	demo listo	2	0	0%	
33	demo prueba	2	0	0%	
32	Demo PYCCA LISTO	2	2	0%	demo
6	Prueba 666	4	0	0%	pruebas d descripcion 666
4	Prueba programada	1	1	33%	pruebas riiii
132 rows					

Figura 3.21 Pruebas_unitarias Ejecución de funciones para reportes tiempo real progreso de campañas.

Las **Figuras 3.22 y 3.23** presentan resultados después de ejecutar el *script* para la obtención de los contactos y archivos de audio existentes en las campañas mediante las funciones respectivas.

contactid	phonenumber
19211	0995216349
19212	0996326045
19213	0995216349
19214	0996326045
19215	0995216349
19216	0996326045
19217	0995216349
19218	0996326045
19219	0995216349

(9 rows)

contactid	phonenumber
29	0995216349
34	0996326045

(2 rows)

Figura 3.22 Pruebas_unitarias Ejecución de funciones para contactos en campañas.

automatic_getmessage
/var/www/contact-temp/FinalOff2015.wav

(1 row)

1

(1 row)

automatic_setmessage
1

(1 row)

Figura 3.23 Pruebas_unitarias Ejecución de funciones audios en campañas.

Las **Figuras 3.24, 3.25 y 3.26** presentan resultados de los registros y reportes estadísticos de las campañas existentes en el sistema, pueden ser de llamadas automáticas o mensajes de texto mediante las funciones respectivas.

La **Figura 3.27** presenta resultados de ejecución del *script* para insertar y asociar los contactos y el audio en cada una de las campañas creadas.

Las **Figuras 3.28 y 3.29** presentan resultados de ejecución de funciones de los eventos que las campañas poseen, las campañas pueden ser de llamadas automáticas o mensajes de texto.

callid	date	time	phonenumber	duration	state
19211	2017-11-14	16:01:51	0995216349	00:00:07	ANSWERED
19212	2017-11-14	16:06:52	0996326045	00:00:05	BUSY
19213	2017-11-14	16:02:12	0995216349	00:00:07	FAILED
19214	2017-11-14	16:02:33	0996326045	00:00:04	ANSWERED
19215	2017-11-14	16:02:49	0995216349	00:00:06	CONGESTION
19216	2017-11-14	16:03:10	0996326045	00:00:05	NO ANSWER
19217	2017-11-14	16:08:26	0995216349	00:00:09	NO ANSWER
19218	2017-11-14	16:07:11	0996326045	00:00:04	ANSWERED
19219	2017-11-14	16:07:25	0995216349	00:00:03	ANSWERED

(9 rows)

callid	date	time	phonenumber	duration	state
29	2017-11-07	16:45:42	0995216349	00:00:14	ANSWERED
29	2017-11-07	16:28:37	0995216349	00:00:03	ANSWERED
29	2017-11-07	16:49:15	0995216349	00:00:15	ANSWERED
29	2017-11-07	16:44:12	0995216349	00:00:13	ANSWERED
29	2017-11-07	16:54:38	0995216349	00:00:16	ANSWERED
29	2017-11-07	16:56:56	0995216349	00:00:15	ANSWERED
29	2017-11-07	16:21:39	0995216349	00:00:03	ANSWERED
29	2017-11-07	16:38:29	0995216349	00:00:17	ANSWERED
34	2017-11-07	16:58:33	0996326045	00:00:15	ANSWERED
34	2017-11-07	16:21:58	0996326045	00:00:02	ANSWERED
34	2017-11-07	16:32:15	0996326045	00:00:15	ANSWERED
34	2017-11-07	16:40:04	0996326045	00:00:16	ANSWERED
34	2017-11-07	16:44:49	0996326045	00:00:13	ANSWERED
34	2017-11-07	16:47:07	0996326045	00:00:17	ANSWERED
34	2017-11-07	16:50:49	0996326045	00:00:15	ANSWERED

(15 rows)

Figura 3.24 Pruebas_unitarias Ejecución de funciones registro de campañas llamadas automáticas.

contactid	phonenumber	message
372	995216349	hola buenas noches
373	996326045	hola buenas noches
374	995216349	hola buenas noches
375	996326045	hola buenas noches
376	995216349	hola buenas noches
377	996326045	hola buenas noches
378	995216349	hola buenas noches
379	996326045	hola buenas noches
380	995216349	hola buenas noches

(9 rows)

mddrid	contactid	datetime	phonenumber	port	result
62	93	2017-01-16 10:44:27	0995216349	FAMILY RUN ULTIMAS INSCRIPCIONES SABADO 14 DE EN	ERO. CLIENTE PREVER PAGA UNA TARIFA PROMOCIONAL DE USD \$16 gsm-4.1 sending
65	94	2017-01-16 10:44:27	0997886859	FAMILY RUN ULTIMAS INSCRIPCIONES SABADO 14 DE EN	ERO. CLIENTE PREVER PAGA UNA TARIFA PROMOCIONAL DE USD \$17 gsm-4.3 sending
60	95	2017-01-16 10:44:27	0988269348	FAMILY RUN ULTIMAS INSCRIPCIONES SABADO 14 DE EN	ERO. CLIENTE PREVER PAGA UNA TARIFA PROMOCIONAL DE USD \$18 gsm-5.1 sending
69	96	2017-01-16 10:44:51	0995216349	FAMILY RUN ULTIMAS INSCRIPCIONES SABADO 14 DE EN	ERO. CLIENTE PREVER PAGA UNA TARIFA PROMOCIONAL DE USD \$19 gsm-3.2 sending
61	97	2017-01-16 10:44:27	0997886859	FAMILY RUN ULTIMAS INSCRIPCIONES SABADO 14 DE EN	ERO. CLIENTE PREVER PAGA UNA TARIFA PROMOCIONAL DE USD \$20 gsm-5.4 sending
70	98	2017-01-16 10:44:51	0988269348	FAMILY RUN ULTIMAS INSCRIPCIONES SABADO 14 DE EN	ERO. CLIENTE PREVER PAGA UNA TARIFA PROMOCIONAL DE USD \$21 gsm-4.1 sending
64	99	2017-01-16 10:44:27	0995216349	FAMILY RUN ULTIMAS INSCRIPCIONES SABADO 14 DE EN	ERO. CLIENTE PREVER PAGA UNA TARIFA PROMOCIONAL DE USD \$22 gsm-1.1 sending
57	100	2017-01-16 10:44:12	0997886859	FAMILY RUN ULTIMAS INSCRIPCIONES SABADO 14 DE EN	ERO. CLIENTE PREVER PAGA UNA TARIFA PROMOCIONAL DE USD \$23 gsm-2.1 success
58	101	2017-01-16 10:44:16	0988269348	FAMILY RUN ULTIMAS INSCRIPCIONES SABADO 14 DE EN	ERO. CLIENTE PREVER PAGA UNA TARIFA PROMOCIONAL DE USD \$24 gsm-2.3 success
59	102	2017-01-16 10:44:22	0995216349	FAMILY RUN ULTIMAS INSCRIPCIONES SABADO 14 DE EN	

Figura 3.25 Pruebas_unitarias Ejecución de funciones reportes y estadísticas campaña sms.

state	count
ANSWERED	4
BUSY	1
CONGESTION	1
FAILED	1
NO ANSWER	2
(5 rows)	

state	count
ANSWERED	15
(1 row)	

state	countp
ANSWERED	44
BUSY	11
CONGESTION	11
FAILED	11
NO ANSWER	22
(5 rows)	

state	countp
ANSWERED	750
(1 row)	

Figura 3.26 Pruebas_unitarias Ejecución de funciones reportes estadísticos.

```

psql:/opt/pruebas_unitarias.sql:19: INFO: /usr/bin/sudo /bin/awk '{ print "124," $0 }' < /var/www/contact
t-temp/contactos_sin_audio.csv > /var/www/contact-temp/contactos_sin_audio.csv2
/bin/sed '1s/^\xEF\xBB\xBF//' /var/www/contact-temp/contactos_sin_audio.csv > /var/www/contact-temp
/contactos_sin_audio.csv0
/bin/sed 's/\n//' /var/www/contact-temp/contactos_sin_audio.csv > /var/www/contact-temp/contactos_sin_au
dio.csv
psql:/opt/pruebas_unitarias.sql:19: INFO: COPY private.automatic_contacts ( campaign, phonenumber) FR
OM '/var/www/contactvox-temp/contactos_sin_audio.csv2n' WITH DELIMITER ','
automatic setcontacts
-----
1
(1 row)

psql:/opt/pruebas_unitarias.sql:20: INFO: /usr/bin/sudo /bin/awk '{ print "123," $0 }' < /var/www/contact
t-temp/contactos_martes_texto.csv > /var/www/contact-temp/contactos_martes_texto.csv2
/bin/sed '1s/^\xEF\xBB\xBF//' /var/www/contactvox-temp/contactos_martes_texto.csv > /var/www/contact-temp
/contactos_martes_texto.csv0
/bin/sed 's/\n//' /var/www/contact-temp/contactos_martes_texto.csv > /var/www/contact-temp/contact
os_martes_texto.csv
psql:/opt/pruebas_unitarias.sql:20: INFO: COPY private.automatic_contacts(_campaign, phonenumber) FR
OM '/var/www/contact-temp/contactos_martes_texto.csv2n' WITH DELIMITER ','
automatic_setmessage
-----
1
(1 row)

automatic_setmessage
-----
1
(1 row)

```

Figura 3.27 Pruebas_unitarias Ejecución de funciones inserción de contactos y audios.

```

automatic_start
-----
                0
(1 row)

automatic_stop
-----
                1
(1 row)

automatic_delcampaign
-----
                1
(1 row)

```

Figura 3.28 Pruebas_unitarias Ejecución de funciones eventos de campañas llamadas automáticas.

```

state | count
-----+-----
sending |    10
success |     9
(2 rows)

psql:/opt/pruebas_unitarias.sql:36: INFO:  /bin/awk '{ print "43," $0 }' < /var/www/contact-temp/contactos_miercoles_texto_texto.csv > /var/www/contact-temp/contactos_miercoles_texto_texto.csv2
sms_setcontacts
-----
                1
(1 row)

sms_start
-----
                0
(1 row)

sms_stop
-----
                1
(1 row)

```

Figura 3.29 Pruebas_unitarias Ejecución de funciones eventos de campaña de mensajes de texto.

Al analizar los resultados se debe mencionar que la mayoría de las funciones retornan una variable booleana 1 o 0. Si el proceso se ejecutó correctamente el retorno de la función será 1; si el valor es 0 se debe verificar por que no se realizó correctamente la ejecución. Las dos únicas funciones que retornan el valor de 0 son las de inicio de las campañas tanto de llamadas automáticas como de mensajes de texto, esto se debe a que el momento de realizar las pruebas el *gateway* GSM no estuvo activo.

3.2.3 PRUEBAS DE ACEPTACIÓN

Las pruebas de aceptación que la metodología XP sugiere son test que permiten evaluar las tareas que se definen en las historias de para comprobar si cumplen con su cometido [45]. Es decir que las pruebas de aceptación son las tareas como consecuencias de las historias de usuario obtenidas en la **sección 2.1.6**, que se muestran en la **Tabla 3.8**.

Tabla 3.8 Requerimientos Funcionales para pruebas de aceptación.

REQUERIMIENTOS	TIPO DE REQUERIMIENTO	HISTORIA DE USUARIO
Creación de campañas estáticas.	Funcional	HU1-01
Creación de campañas personalizadas.	Funcional	HU1-02
Creación de campañas de mensajes de texto.	Funcional	HU1-03
Listar todas las campañas existentes según el tipo de campaña (estática, dinámica o de mensajes de texto.	Funcional	HU1-04
Eliminación de campañas tanto estáticas, personalizadas y mensajes de texto SMS.	Funcional	HU1-05
Reporte del número de llamadas totales que se realizaron.	Funcional	HU1-06
Reporte de Campañas de Mensajes de Texto	Funcional	HU1-07
Crear Campaña Estática con número de intentos	Funcional	HU2-01
Crear Campaña Personalizada número de intentos.	Funcional	HU2-02
Subir Archivos en Campaña Estática.	Funcional	HU2-04
Borrar Campaña Estática, Personalizada y Mensajes de Texto	Funcional	HU2-05
Subir Archivos en Campaña de Mensaje de texto	Funcional	HU2-06

Las pruebas de aceptación se las realizaron a un administrador de *callcenter*. Esta persona fue realizando las tareas mostradas en la **Tabla 3.9** comprobando la funcionalidad de la aplicación.

La prueba realizada al administrador del *callcenter* que comprobó el funcionamiento de las tareas, se detallan en el **Anexo D**. La Tabla 3.9 detalla las respuestas a la prueba que el administrador lleno para comprobar el cumplimiento de los requisitos.

Tabla 3.9 Detalle de las Pruebas de Aceptación

Nro.	TAREA	Satisfacción	Satisfacción
1	Crear de campañas estáticas.	100%	Muy Satisfecho
2	Crear de campañas personalizadas.	100%	Muy Satisfecho
3	Crear de campañas de mensajes de texto.	100%	Muy Satisfecho
4	Listar todas las campañas existentes según el tipo de campaña (estática, personalizada o de mensajes de texto.	100%	Muy Satisfecho
5	Eliminación de campañas tanto estáticas, personalizadas y mensajes de texto SMS.	90%	Satisfecho
6	Reporte de la campaña de llamadas Automáticas (estáticas y personalizadas)	100%	Muy Satisfecho
7	Reporte de Campañas de Mensajes de Texto	100%	Muy Satisfecho
8	Subir Archivos de audio y texto en Campaña Estática.	90%	Satisfecho
9	Subir Archivos en Campaña Personalizada	90%	Satisfecho
10	Borrar campañas (Estática, Personalizada y Mensajes de Texto)	100%	Muy Satisfecho
11	Subir Archivos en Campaña de Mensaje de texto	100%	Muy Satisfecho

CAPÍTULO 4

4 CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES

- El presente prototipo basa la implementación de telefonía IP mediante la plataforma Asterisk *Open Source*, logrando desarrollar un sistema muy versátil con una variedad de funcionalidades propias de Asterisk como: grabación de llamadas, conversión de texto a voz, envío automático de mensajes, entre otras; marcando así una diferencia notable con los sistemas tradicionales de telefonía analógica.
- Mediante la encuesta presente en el Anexo A, realizada a varios administradores de *callcenter* para determinar los requerimientos necesarios para la automatización de llamadas y mensajes de texto, se determina que, los beneficios de la implementación del prototipo desarrollado en el presente proyecto de Titulación solventan muchas de las necesidades de los *callcenters*.
- Las pruebas realizadas en la **sección 3.2.1** muestran el correcto funcionamiento del prototipo implementado. El prototipo permite crear campañas de llamadas de tipo estática o personalizada; generar automáticamente llamadas y reproducir el audio deseado. También permite la creación de campañas de mensajes de texto generando automáticamente el envío instantáneo de sms, satisfaciendo los requerimientos obtenidos en la encuesta realizada presente en el Anexo A.
- Mediante los resultados obtenidos en la encuesta mostrada en el Anexo D, se concluye que la aplicación web implementada tiene un alto grado de cumplimiento y satisfacción por parte del usuario y se cubre con los requerimientos propuestos en la **sección 2.1.6**. Lo que causó un poco de dificultad es el crear un archivo de texto con la información requerida

ya que el prototipo solo acepta archivos de texto csv con codificación UTF8 y archivos de audio .wav.

- El prototipo implementado satisface los requerimientos obtenidos mediante la encuesta presente en el Anexo A, ya que brinda información de retro alimentación al administrador del *callcenter* para tomar correctas decisiones futuras como enviar información personalizada para determinados contactos dependiendo del estado de contestación de la llamada (Contestadas) o generar un plan de posibles actividades, dependiendo de los reportes, que estén relacionadas con el giro del negocio de la empresa a la que pertenece.
- Actualmente ya existen prototipos que realizan llamadas automáticas múltiples y envíos de texto masivos pero no son accesibles para muchas empresas debido a sus altos costos y planes de contratación limitantes El prototipo creado permite tener varias funcionalidades juntas en una sola aplicación web, no independientes como los servicios que ofrecen las grandes empresas de mensajería en las cuales se deben acceder a cada servicio.
- En el Ecuador existen empresas dedicadas a realizar actividades similares a las que proporciona el prototipo implementado pero la diferencia es que dichas empresas facturan cada servicio solicitado mediante un plan establecido por ellos, limitando al cliente con un número de envíos o llamadas mínimo para la contratación de servicios a costos elevados. Esto discrimina a empresas que necesitan implementar comunicaciones masivas pero con un número menor de envío de mensajes de texto y llamadas automáticas y de recursos limitados.
- Mediante las pruebas realizadas en la **sección 3.2.1** se comprueba el correcto funcionamiento del envío masivo de mensajes de texto y se obtiene un reporte por el estado de envío del mensaje. Este reporte no puede garantizar que el sms sea leído por el cliente ya que la operadora

no maneja un protocolo de respuesta que permita saber si el cliente leyó o no el mensaje.

- Solo se obtiene resultados del mensaje de texto, desde el *gateway* GSM, como los siguientes: exitoso, enviando o fallido. No se puede determinar si el cliente leyó el contenido del mensaje de texto en su dispositivo móvil antes de borrarlo o solamente lo dejó almacenado sin leerlo, esto depende de cada uno de los usuarios.
- El prototipo, al generar las llamadas automáticas, no detecta la diferencia entre una contestadora automática o una persona; ya que no existe un estándar o protocolo en las operadoras de telefonía móvil a nivel de contestadora automática que permitan identificar a la contestadora. Por ello, los resultados obtenidos en las estadísticas de los reportes se deben estimar como un aproximado para el estado de Contestada.
- En el Ecuador, las operadoras de telefonía móvil establecen “POLITICAS DE USO RAZONABLE DE LOS SERVICIOS MOVILES”. Estas políticas determinan las prácticas que NO podrían considerarse como uso razonable y podría ser considerado uso comercial. Entre ellas están: “Uso compartido de planes entre usuarios a través de un sistema PBX, centro de llamadas, equipo informático o cualquier otro medio y envío de mensajes masivos a través de una máquina/modem/PC u otro dispositivo de envío automático de SMS” [46].

Para no violar estas políticas, se recomienda realizar campañas con un límite de aproximadamente 50 mensajes de texto por chip o tarjeta SIM⁴⁰ que permite el acceso a la línea celular, lo cual es permitido por las operadoras, y de igual manera para las llamadas. Como conclusión, al

⁴⁰ SIM: *Subscriber Identity Module* o módulo de identificación de suscripción es una tarjeta inteligente diseñada para equipos móviles.

sobrepasar estas políticas, las operadoras bloquean por 24 horas los chips.

- La aplicación Web puede ser mejorada agregando funcionalidades como un control de horarios para programar automáticamente el inicio y fin de la campaña. Esta mejora resulta de una prueba realizada al generar una campaña de llamadas estáticas obteniendo resultados de contestación por debajo del 10% del total de contactos. Lo que no se definió inicialmente es el horario en el cual las personas tienen mayor índice de contestación, el cual está entre las 13:00-14:00 y con sumo éxito de 18:00-19:00.
- Se apuesta por el uso de mensajería de texto debido a que no se necesita tener un plan de datos que les permita verificar sus notificaciones y es menos invasivo que realizar una llamada. Los SMS utilizan la misma infraestructura que el de una llamada GSM.
- En el presente proyecto se toma como análisis de resultados todo lo mencionado como estadísticas y reportería de cada una de las campañas. Esto permiten obtener respuestas del desarrollo de las campañas tanto en contactabilidad como en efectividad. La contactabilidad hace referencia al estado de la llamada a nivel telefónico o envío del mensaje de texto (contestado, no contestado, fallido, ocupado para llamadas y para mensajes exitoso, fallido o enviando.) y la efectividad se la puede verificar con la reportería dependiendo de la hora que se inicialice la campaña; tomando en cuenta cuantos contactos recibieron las llamadas y escucharon el audio reproducido.

4.2 RECOMENDACIONES

- En el futuro se recomienda la implementación de mecanismos de reconocimiento de voz automática ASR para Asterisk los cuales pueden detectar si la respuesta, al contestarse una llamada, corresponde a una máquina o una persona.

- Al ingresar el mensaje de texto para las campañas de llamadas automáticas personalizadas, se debe utilizar correctamente los signos de puntuación, pausas, caracteres especiales y espacios en blanco, para que el audio creado sea claro y tenga coherencia al ejecutarse.
- Se recomienda adquirir planes de telefonía móvil de solo llamadas y mensajes de texto ya que no se utiliza Internet para la utilización del prototipo y sería un desperdicio de recursos.
- Si se va a utilizar *Gateways* GSM de una cantidad alta de puertos de tarjetas SIM para conectar las llamadas con la PSTN y existen pérdidas de paquetes en la llamada o las llamadas no se están generando, se debe tratar de buscar un sector físico de gran cobertura de la operadora móvil contratada o se utilice antenas coaxiales para exteriores.
- Es necesario revisar periódicamente el espacio en disco de la máquina que se utilice para no tener problemas de lectura y escritura en el sistema operativo. Ya que, al generar los archivos de audio para las llamadas personalizadas, ocupa espacio en el disco y se puede saturar. Dependiendo de la campaña generada se pueden estimar algunos parámetros para tener datos del espacio utilizado en el disco. Los parámetros que se pueden estimar son: el volumen de contactos ingresados, la duración del audio creado y el peso del archivo. Por ejemplo, tiempo de duración promedio del audio personalizado 6 segundos y promedio de peso del archivo aproximado 110KB.
- Se recomienda que al compilar asterisk se siga el proceso que sugiere la página oficial. Primero instalar una serie de dependencias que requiere el sistema operativo para la compilación e instalación de Asterisk. Para el proyecto se crea un *script* con el nombre ***addons.sh*** con las dependencias a instalar debido a que después de falla y error se logró condensar las dependencias necesarias para la instalación de Asterisk y la base de datos postgresql. En el anexo D se muestra el *script* creado.

- Realizar respaldos de la base de datos, las configuraciones de asterisk y como el proyecto se encuentra virtualizado, también se podría realizar un clon de la máquina virtual debido a que cualquier cambio en la configuración puede eliminar lo configurado y ejecutado previamente. De esta manera se tiene un punto de partida ya avanzado y no se tiene que empezar todo de nuevo.

BIBLIOGRAFÍA

- [1] «Ecuador tiene 16,9 millones de líneas celulares, cifra que supera a su población. (2016).», 2016. [En línea]. Disponible en: <http://www.eluniverso.com/2013/01/22/1/1356/ecuador-tiene-169-millones-lineas-celulares-cifra-supera-poblacion.html>.
- [2] «B., C. (2014). Cobertura Digital – Penetración telefonía móvil en Ecuador: Smartphones superan a conexiones fijas. Coberturadigital.com». [En línea]. Disponible en: <http://www.coberturadigital.com/2014/10/17/penetracion-telefonía-movil-en-ecuador-smartphones-superan-a-conexiones-fijas/>. [Accedido: 27-mar-2016].
- [3] «Clientes Xopik Marketing móvil». [En línea]. Disponible en: <http://www.xopik.com/es/>.
- [4] «Cooperativas y bancos son los que más utilizan los SMS para llegar a sus clientes. Las mismas telefónicas notifican de sus promociones y días de pago por medio del SMS. (2016). El COMERCIO.» [En línea]. Disponible en: <http://www.mediatelecom.com.mx/index.php/agencia-informativa/noticias/itemlist/tag/SMS?start=25>.
- [5] «Memoria_Mayo_Diana.pdf». .
- [6] D. Mallory, K. Salhoff, y D. Donohue, *Cisco Voice Gateways and Gatekeepers (paperback)*. Pearson Education, 2006.
- [7] ferramnav, « ¿Qué es un Gateway GSM o GoIP?», 29-abr-2014. [En línea]. Disponible en: <https://www.taringa.net/posts/hazlo-tu-mismo/17772505/Que-es-un-Gateway-GSM-o-GoIP.html>. [Accedido: 21-oct-2017].
- [8] Q. I. M. and Consulting, « ¿Que es una Centralita IP? - Central Telefónica VoIP - IP-PBX». [En línea]. Disponible en: <http://www.quarea.com/es/que-es-una-centralita-ip-central-telefonica-voip-ip-pbx>. [Accedido: 22-oct-2017].
- [9] «IPTELEPHONYCOOKBOOK.pdf». .
- [10] «Protocolo de iniciación de sesión», *Wikipedia, la enciclopedia libre*. 28-ago-2017.
- [11] «informe23.pdf». .

- [12] Eng. Fernando Mendioroz, M.Sc., «Telefonía IP (SIP, Diameter, RTP/RTPC)», 12:47:05 UTC.
- [13] «SIP - 3Cu Electrónica». [En línea]. Disponible en: <https://sites.google.com/site/3cuelelectronica/home/voip/sip>. [Accedido: 23-oct-2017].
- [14] «VoIP Foro - Ejemplo comunicacion y flujo SIP». [En línea]. Disponible en: <http://www.voipforo.com/SIP/SIPejemplo.php>. [Accedido: 23-oct-2017].
- [15] «Qué es un proxy», CCM. [En línea]. Disponible en: <http://es.ccm.net/faq/2755-que-es-un-proxy>. [Accedido: 30-oct-2017].
- [16] A. B. Solsona, J. M. H. Moya, y J. J. Calero, *Redes de área local: administración de sistemas informáticos*. Editorial Paraninfo, 2006.
- [17] J. A. C. Falcón, *VoIP: la telefonía de Internet*. Editorial Paraninfo, 2007.
- [18] J. M. H. Moya y R. C. Pastor, *Sistemas de telefonía*. Editorial Paraninfo, 2006.
- [19] «Codecs en la Telefonía IP, Codecs VoIP». [En línea]. Disponible en: <http://www.telefoniavoz.com/voip/codecs-voip.htm>. [Accedido: 24-oct-2017].
- [20] «Qué es Asterisk y cómo puede ayudar a su empresa», *Mas IP | Telefonía IP, Centralitas VOIP, Call Center*, 18-mar-2014. .
- [21] Q. I. M. and Consulting, «Asterisk: Funcionalidades básicas y avanzadas». [En línea]. Disponible en: <http://www.quarea.com/es/asterisk-funcionalidades-basicas-avanzadas>. [Accedido: 24-oct-2017].
- [22] L. Madsen, J. V. Meggelen, y R. Bryant, *asterisk-the-definitive-guide-definitive-guides.*, 3rd Edition. O'REILLY, 2011.
- [23] D. G. Barrie Dempster, *Building Telephony Systems With Asterisk*. 2005.
- [24] «Introducción Dialplan - Asterisk Wiki». [En línea]. Disponible en: https://www.wikiasterisk.com/index.php/Introducci%C3%B3n_Dialplan#Contextos. [Accedido: 01-dic-2017].

- [25] «Campaña de telemarketing», *Luxor Technologies*, 03-jul-2011. [En línea]. Disponible en: <http://www.luxortec.com/preguntas-frecuentes/campana-de-telemarketing/>. [Accedido: 08-nov-2017].
- [26] «5.3. Text-to-Speech (TTS)». [En línea]. Disponible en: <http://the-asterisk-book.com/1.6/text2speech.html>. [Accedido: 31-oct-2017].
- [27] Dairox, « ¿Qué es y cómo funciona el 4G?», *Xataka México*, 19-mar-2014. [En línea]. Disponible en: <https://www.xataka.com.mx/celulares-y-smartphones/que-es-y-como-funciona-el-4g>. [Accedido: 31-oct-2017].
- [28] *Servicio de Mensajes Cortos (SMS) el Mercado Telefónico de España*. The Mint.
- [29] D. R. Cardozzo, *Desarrollo de Software: Requisitos, Estimaciones y Análisis. 2 Edición*. IT Campus Academy, 2016.
- [30] P. Lledó, *GESTIÓN LEAN y GIL de proyectos*. Trafford Publishing, 2014.
- [31] «Ventajas y Desventajas Modelo Cascada | Que Computadora Comprar».
- [32] joselynvaleria93, «Principios de las metodologías ágiles», 23:03:24 UTC.
- [33] P. Letelier y M. C. Penadés, «Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP)», www.cyta.com.ar/ta0502/v5n2a1.htm, 15-abr-2006. [En línea]. Disponible en: <http://www.cyta.com.ar/ta0502/v5n2a1.htm>. [Accedido: 08-nov-2017].
- [34] «is1-t02-trans.pdf».
- [35] «2244_555_COD_18_290814203015.pdf».
- [36] «SMS número corto - Antel». [En línea]. Disponible en: <http://www.antel.com.uy/empresas/soluciones/sms-numero-corto>. [Accedido: 11-dic-2017].
- [37] «Del blog al podcast con síntesis de voz - Sopa de bits». [En línea]. Disponible en: <https://www.sopadebits.com/blog-podcast-sintesis-voz>. [Accedido: 03-dic-2017].
- [38] «Text2wave - LQWiki». [En línea]. Disponible en: <http://wiki.linuxquestions.org/wiki/Text2wave>. [Accedido: 03-dic-2017].

- [39] «Instalar y configurar Festival en Asterisk 1.6.X | VozToVoice». [En línea]. Disponible en: <https://www.voztovoice.org/?q=node/97>. [Accedido: 03-dic-2017].
- [40] «Usando Festival programa de texto a voz desde consola | BlogDRAKE». [En línea]. Disponible en: <http://blogdrake.net/blog/javierreta/usando-festival-programa-de-texto-voz-desde-consola>. [Accedido: 03-dic-2017].
- [41] « ¿Qué es un script?» [En línea]. Disponible en: https://enablement.acl.com/helpdocs/analytics/125/scripting-guide/es/Content/how_to_script/complete_beginners/what_is_a_script.htm. [Accedido: 06-dic-2017].
- [42] «PostgreSQL: Documentación: 9.1: El archivo pg_hba.conf». [En línea]. Disponible en: <https://www.postgresql.org/docs/9.1/static/auth-pg-hba-conf.html>. [Accedido: 04-dic-2017].
- [43] «Información NetBeans IDE 6.1». [En línea]. Disponible en: https://netbeans.org/community/releases/61/index_es.html. [Accedido: 05-dic-2017].
- [44] «UTF-8 and Unicode Standards». [En línea]. Disponible en: <http://www.utf-8.com/>. [Accedido: 11-dic-2017].
- [45] «Fases». [En línea]. Disponible en: <http://programacionextrema.tripod.com/fases.htm>. [Accedido: 11-dic-2017].
- [46] «Política de Uso Razonable de Servicios de OTECEL S A Febrero 2016.pdf» . .

ANEXO A.

ENCUESTA

ENCUESTA EMPRESA 1

PREGUNTA	Opciones	Respuestas (aproximadas)
¿Con cuántos clientes en promedio tiene comunicación diaria?	0 a 500	200
	500 a 1500	0
	Más de 1500	0
¿Qué destinos son los más frecuentes al contactar a sus clientes diariamente?	Números Celulares	35
	Números Convencionales	20
¿Cuál es el horario más común en el que su cliente contesta una llamada?	10:00-11:00	5
	13:00-14:00	10
	16:30-18:30	40
¿Cuál es el horario menos asertivo en el cual su cliente contesta una llamada?	06:00-09:00	35
	10:00-13:00	20
	14:00-16:30	10
¿Cuál es el porcentaje de agentes que asigna la empresa para realizar campañas informativas? (número de agentes que representa el porcentaje) 3 agentes	10 %	0
	30%	1 agente de 3
	60%	0
¿Le parece que el contacto con el cliente mediante mensajes SMS es adecuado?	SI	75
	NO	40
¿Cuánto tiempo el operador tarda en brindar la información al cliente? (cantidad de contactos en una hora)	5 minutos	5
	7 minutos	20
	12 minutos	5
¿Cuál es la frecuencia que establece comunicación con sus clientes? (número de clientes)	Periódicamente	800
	Bajo demanda	1000
¿Qué medios adicionales utiliza para comunicar o informar a sus clientes aviso o promociones diariamente? (número total de notificaciones)	Correo electrónico	10
	Distribuidor autorizado de llamadas mediante número corto ⁴¹	2
	Distribuido de sms autorizado mediante un número corto	3
¿Qué porcentaje de los mensajes en las llamadas generados a sus clientes son genéricos?	0% a 35%	15%
	35% a 70%	0%
	70% a 100%	0%

⁴¹ Número corto: número con el cuál las grandes empresas cobran para enviar mensajes o llamadas múltiples

ENCUESTA EMPRESA 2

PREGUNTA	Opciones	Respuestas
¿Con cuántos clientes en promedio tiene comunicación diaria?	0 a 500	0
	500 a 1500	500
	Más de 1500	0
¿Qué destinos son los más frecuentes al contactar a sus clientes diariamente?	Números Celulares	45
	Números Convencionales	30
¿Cuál es el horario más común en el que su cliente contesta una llamada?	10:00-11:00	10
	13:00-14:00	30
	16:30-18:30	77
¿Cuál es el horario menos asertivo en el cual su cliente contesta una llamada?	06:00-09:00	55
	10:00-13:00	30
	14:00-16:30	10
¿Cuál es el porcentaje de agentes que asigna la empresa para realizar campañas informativas? (número de agentes que representa el porcentaje) 8 agentes	10 %	1 agente de 8
	30%	0
	60%	0
¿Le parece que el contacto con el cliente mediante mensajes SMS es adecuado?	SI	30
	NO	30
¿Cuánto tiempo el operador tarda en brindar la información al cliente? (cantidad de contactos en una hora)	5 minutos	25
	7 minutos	60
	12 minutos	20
¿Cuál es la frecuencia que establece comunicación con sus clientes? (número de clientes)	Periódicamente	700
	Bajo demanda	1000
¿Qué medios adicionales utiliza para comunicar o informar a sus clientes aviso o promociones diariamente? (número total de notificaciones)	Correo electrónico	10
	Distribuidor autorizado de llamadas mediante número corto	2
	Distribuido de sms autorizado mediante un número corto	3
¿Qué porcentaje de los mensajes en las llamadas generados a sus clientes son genéricos?	0% a 35%	15%
	35% a 70%	0%
	70% a 100%	0%

ENCUESTA EMPRESA 3

PREGUNTA	Opciones	Respuestas
¿Con cuántos clientes en promedio tiene comunicación diaria?	0 a 500	0
	500 a 1500	0
	Más de 1500	2000
¿Qué destinos son los más frecuentes al contactar a sus clientes diariamente?	Números Celulares	150
	Números Convencionales	100
¿Cuál es el horario más común en el que su cliente contesta una llamada?	10:00-11:00	38
	13:00-14:00	80
	16:30-18:30	110
¿Cuál es el horario menos asertivo en el cual su cliente contesta una llamada?	06:00-09:00	200
	10:00-13:00	50
	14:00-16:30	20
¿Cuál es el porcentaje de agentes que asigna la empresa para realizar campañas informativas? (número de agentes que representa el porcentaje) 15 agentes	10 %	0
	30%	0
	60%	9 agentes de 16
¿Le parece que el contacto con el cliente mediante mensajes SMS es adecuado?	SI	120
	NO	80
¿Cuánto tiempo el operador tarda en brindar la información al cliente? (cantidad de contactos)	5 minutos	10
	7 minutos	100
	12 minutos	80
¿Cuál es la frecuencia que establece comunicación con sus clientes? (número de clientes)	Periódicamente	4500
	Bajo demanda	22000
¿Qué medios adicionales utiliza para comunicar o informar a sus clientes aviso o promociones diariamente? (número total de notificaciones)	Correo electrónico	10
	Distribuidor autorizado de llamadas mediante número corto	2
	Distribuido de sms autorizado mediante un número corto	3
¿Qué porcentaje de los mensajes en las llamadas generados a sus clientes son genéricos?	0% a 35%	0%
	35% a 70%	0%
	70% a 100%	70%

ANEXO B.

CREACIÓN DE TABLAS EN LA BASE DE DATOS

```

CREATE TABLE private.automatc_campaigns (
  id integer NOT NULL,
  name character varying(32) NOT NULL,
  channels integer DEFAULT 5 NOT NULL,
  trunktype tel_trunk_type DEFAULT 'sip'::tel_trunk_type NOT NULL,
  trunk text NOT NULL,
  state integer DEFAULT 0 NOT NULL,
  created timestamp(0) without time zone DEFAULT now() NOT NULL,
  description character varying(128) DEFAULT NULL::character varying,
  campaigntype coc_callmatic_campaigns_type,
  maxretries integer DEFAULT 1,
  retrytime integer DEFAULT 30,
  CONSTRAINT coc_callmatic_campaigns_channels_check CHECK ((channels > 0))
);

ALTER TABLE private.automatc_campaigns OWNER TO postgres;

```

```

CREATE TABLE private.contacts_prerecorded (
  id integer NOT NULL,
  _campaign integer NOT NULL,
  phonenumber character varying(16) NOT NULL,
  state integer DEFAULT 0 NOT NULL
);

ALTER TABLE private.contacts_prerecorded OWNER TO postgres;

```

```

CREATE TABLE private.contacts_prerecorded (
  id integer NOT NULL,
  _campaign integer NOT NULL,
  phonenumber character varying(16) NOT NULL,
  state integer DEFAULT 0 NOT NULL
);

ALTER TABLE private.contacts_prerecorded OWNER TO postgres;

```

```

CREATE TABLE private.contacts_built (
  id integer NOT NULL,
  _campaign integer NOT NULL,
  phonenumber character varying(32) NOT NULL,
  field_1 text NOT NULL,
  state integer DEFAULT 0 NOT NULL
);

ALTER TABLE private.contacts_built OWNER TO postgres;

```

```

CREATE TABLE private.audio_prerecord (
  id integer NOT NULL,
  _campaign integer NOT NULL,
  filename character varying(512) NOT NULL
);

ALTER TABLE private.audio_prerecord OWNER TO postgres;

```

```

CREATE TABLE private.sms_contacts (
  id integer NOT NULL,
  _campaign integer NOT NULL,
  phonenumber character varying(32) NOT NULL,
  message text NOT NULL,
  state integer DEFAULT 0 NOT NULL
);

ALTER TABLE private.sms_contacts OWNER TO postgres;

```

```

CREATE TABLE private.sms_mdr (
  id integer NOT NULL,
  _campaign integer NOT NULL,
  _contact integer NOT NULL,
  port text,
  phonenumber text,
  message text,
  datetime timestamp(0) without time zone,
  result text
);

ALTER TABLE private.sms_mdr OWNER TO postgres;

```

```

CREATE TABLE private.users (
  id integer NOT NULL,
  username text NOT NULL,
  firstname text NOT NULL,
  lastname text,
  "position" text,
  password text NOT NULL,
  created timestamp without time zone DEFAULT now() NOT NULL,
  CONSTRAINT users_username_check CHECK ((username ~* '^[a-zA-z0-9_-]+$'::text))
);

ALTER TABLE private.users OWNER TO postgres;

```

```

CREATE TABLE private.sms_campaigns (
  id integer NOT NULL,
  name text NOT NULL,
  channels integer DEFAULT 1 NOT NULL,
  trunk text NOT NULL,
  trunkusername text,
  trunkpassword text,
  state integer DEFAULT 1 NOT NULL,
  created timestamp(0) without time zone DEFAULT now() NOT NULL,
  description character varying(100) DEFAULT NULL::character varying,
  CONSTRAINT sms_campaigns_channels_check CHECK ((channels > 0))
);

ALTER TABLE private.sms_campaigns OWNER TO postgres;

```

```

CREATE TABLE public.ast_matic_cdr (
  id integer NOT NULL,
  callerid character varying(64),
  source character varying(32),
  destination character varying(64),
  start timestamp(0) without time zone,
  answer timestamp(0) without time zone,
  finish timestamp(0) without time zone,
  duration integer,
  billsec integer,
  context character varying(32),
  channel character varying(64),
  dstchannel character varying(64),
  application character varying(32),
  data character varying(64),
  accountcode character varying(16),
  disposition character varying(32),
  amaflags integer,
  location character varying(32),
  locationid integer,
  line integer,
  callid character varying(32),
  uniqueid character varying(32),
  linkedid character varying(32),
  maxretries integer,
  retrytime integer
);

ALTER TABLE public.ast_matic_cdr OWNER TO postgres;

```

ANEXO C.

CÓDIGO DE LA APLICACIÓN WEB

```

<!-------CREACIÓN DE CAMPAÑAS DE LLAMADAS AUTOMATICAS ----->
<!DOCTYPE html>
<html lang="en">
<head>
<title>Prototipo AUTOMÁTIC</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1">
  <link rel="stylesheet" href="../tesis/css/bootstrap.min.css">
  <script src="jquery.min.js"></script>
  <script src="bootstrap.min.js"></script>
  <script>
//creamos la variable ventana_secundaria que contendrá; una
referencia al popa que vamos a abrir
//la creamos como variable global para poder acceder a ella
desde las distintas funciones
var ventana_secundaria

function abrirVentana(){
//guardo la referencia de la ventana para poder utilizarla luego
ventana_secundaria =
window.open(href="VistaCrearcampaignLlamadas.php","miventana","w
idth=300,height=200,menubar=no")
}

function cerrarVentana(){
//la referencia de la ventana es el objeto window del popup. Lo
utilizo para acceder al método close
ventana_secundaria.close()
}
</script>
<script language="JavaScript">
function pregunta(){
  if (confirm('¿Estas seguro de enviar este formulario?')){
    document.DeleteCampaña.submit()
  }
  else
  {

  }
}
</script>
<script>

  function cerrarse(){
    window.close()
  }

</script>
</head>

```

```

<body>

<nav class="navbar navbar-default">
  <div class="container-fluid">
    <div class="container-header">
      <div class="container-header">
        <h3 class="text-center text-uppercase" >AUTOMATIC
SEND&DIALLING (ASD)</h3>
      </div>
      <a class="navbar-brand" href="#" class="text-center">
Creación de Campañas Llamadas Automáticas </a>
    </div>
  </div>
</nav>
<?php

    //Creación de Campañas llamadas automáticas
    require_once ('DataBase.php');
    $connectar= DataBase::connectDB();

    $name= $_POST['name'];
    $channels= $_POST['channels'];
    $type= $_POST['type'];
    $trunk= $_POST['trunk'];
    $maxretries= $_POST['maxretries'];
    $retrytime= $_POST['retrytime'];
    $description = $_POST['description'];

    if($name!=''){
//funcion creación de campañas automáticas
    $consulta="SELECT * FROM
public.automatic_createcampaign_retry('$name','$channels','$type
','$trunk','$maxretries','$retrytime','$description') ";
    $resultado=pg_query($connectar, $consulta);
    echo '
        <div class="container-header">
<h3 class="text-center text-uppercase" >CreaciÃ³n
exitosa</h3>
        </div>
        <div class="container-fluid">
        <td>
            <button name="btn_Regresar" onclick="cerrarse()" "
>Salir</button>
        </td>';
    }

    else {
        echo '
        <div class="container-header">
<h3 class="text-center text-uppercase" >Error campos
Vacios</h3>
        </div>
        <div class="container-fluid"><td>

```

```

<button name="btn_Regresar" onclick="cerrarse()"
>Cerrar</button>
        </td>';
    }
    ?>
<!-------FIN CREACIÓN DE CAMPAÑAS DE LLAMADAS AUTOMATICAS ----->

<!-------CREACIÓN DE CAMPAÑAS DE MENSAJES DE TEXTO ----->
<!DOCTYPE html>
<html lang="en">
<head>
<title>Prototipo AUTOMÁTIC</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1">
    <link rel="stylesheet" href="../tesis/css/bootstrap.min.css">
    <script src="jquery.min.js"></script>
    <script src="bootstrap.min.js"></script>
    <script>

        function cerrarse(){
            window.close()
        }

    </script>
</head>
<body>

<nav class="navbar navbar-default">
    <div class="container-fluid">
        <div class="container-header">
            <h1 class="text-center">Tesis</h1>
            <a class="navbar-brand" href="#"> Registro Campañas
Mensajes de Texto SMS </a>
        </div>
    </div>
</nav>
<?php

//Creación de campañas de mensajes de texto
    require_once ('DataBase.php');
    $connectar= DataBase::connectDB();

    $name= $_POST['name'];
    $channels= $_POST['channels'];
    $trunk= $_POST['trunk'];
    $trunkuser= $_POST['user'];
    $trunkpasswd= $_POST['passwd'];
    $description = $_POST['description'];

    if($name!=''){
        //funcion creación de campañas de mensajes de texto
        $consulta="SELECT * FROM
public.sms_createcampaign('$name','$channels','$trunk','$trunkus
er', '$trunkpasswd','$description')";
        $resultado=pg_query($connectar, $consulta);

```

```

        echo 'exito
        <div class="container-fluid">
        <td>
                <button name="btn_Regresar"
onclick="cerrarse()" >Regresar</button>

                </td>';

        }

else {
        echo 'error campos vacios de texto';

}

?>

<!-------FIN CREACIÓN DE CAMPAÑAS DE MENSAJES DE TEXTO ----->

<!------- CONEXIÓN CON LA BASE DE DATOS ----->

<?php
/*
 * To change this license header, choose License Headers in
Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
class DataBase
{
    function connectDB()
    {
        $user="user=postgres";
        $password="password=T3s1s";
        $dbname="dbname=automatic";
        //$dbname="dbname=automatic";
        $port="port=5432";
        $host="host=192.168.0.218";
        //$host="host=192.168.0.215";
        $adapter = pg_connect("$host $port $dbname $user
$password");
        if(!$adapter){

            echo "Error en la Conexion:". pg_last_error;

        }
        return $adapter;
    }
}
?>
<!------- FIN CONEXIÓN CON LA BASE DE DATOS ----->

<!------- BORRAR CAMPAÑA DE LLAMADAS AUTOMÁTICAS ----->

```



```

    <?php
require_once ('DataBase.php');
$connectar= DataBase::connectDB();

    $campc_id=$_REQUEST['id'];
    $campc_type=$_REQUEST['type'];
    var_dump($campc_id);

    if($campc_id!=''){
    $consulta="select * from
public.coc_callmatic_delcampaign ($campc_id)";
    $resultado=pg_query($connectar, $consulta);

    echo '
        <div class="alert alert-warning">Esta campaña
ha sido borrada</div>

        <div class="container-fluid">
            <tr>

                <td>
                    <a class="btn btn-default"
href="Llamadas_pregrabadas.php">Regresar</a>
                    <!--<button name="btn_Regresar"
onclick="cerrarse()" >Regresar</button>
                    </td>
                <tr>';

    }

    else {
        echo 'error No existe esa Campaña';
    }
    ?>
<!------- FIN BORRAR CAMPAÑA DE LLAMADAS AUTOMÁTICAS ----->

<!------- BORRAR CAMPAÑA DE MENSAJES DE TEXTO ----->

<!--<html>
    <head>
        <meta charset="UTF-8">
        <title></title>
        <link rel="stylesheet" type="text/css" href="Tesis.css">
    </head>-->
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Prototipo AUTOMÁTIC</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1">
    <link rel="stylesheet" href="../tesis/css/bootstrap.min.css">
    <link rel="stylesheet" type="text/css" href="Tesis.css">
    <script src="../tesis/js/jquery.min.js"></script>
    <script src="../tesis/js/jquery.simpletable"></script>
    <script src="../tesis/js/bootstrap.min.js"></script>
</head>

```

```

<body>
  <div class="container-fluid">
    <?php

/*
 * To change this license header, choose License Headers in
Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

require_once ('DataBase.php');
    $connectar= DataBase::connectDB();

    $campc_id=$_REQUEST['id'];
    $campc_type=$_REQUEST['type'];

    if($campc_id!=''){

        $consulta="select * from
public.coc_smsmatic_delcampaign ($campc_id)";
        $resultado=pg_query($connectar, $consulta);

        echo '<div class="alert alert-warning">Esta
campaña ha sido borrada</div>

        <div class="container-fluid">
            <tr>

                <td>
                    <a class="btn btn-default"
href="Mensajes_texto.php">Regresar</a>
                </td>
            <tr>';

    }

    else {
        echo 'error No existe esa Campaña';
    }
?>
    </div>

    </body>
</html>
<!------- FIN BORRAR CAMPAÑA DE MENSAJES DE TEXTO ----->

<!------- INICIAR CAMPAÑA DE LLAMADAS AUTÓMATICAS ----->
<?php

/*
 * To change this license header, choose License Headers in
Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.

```

```

*/
//Class Iniciar LLamadas estáticas y personalizadas.
require_once ('DataBase.php');
$connectar= DataBase::connectDB();

$campc_id=$_REQUEST['id'];
$campc_type=$_REQUEST['type'];
$campc_state=$_REQUEST['state'];

if($campc_state!=2){

    $consulta="select * from
public.coc_callmatic_start('$campc_id','$campc_type')";
    $resultado=pg_query($connectar, $consulta);
    echo ' se ha iniciado la campaña
    <div class="container-fluid">
        <tr>

            <td>
                <a class="btn btn-warning"
href="Llamadas_total.php">Regresar</a>
            </td>
        <tr>';
    }

    else {
        echo 'error no se ha iniciado la campaña';
    }
?>

<!------- FIN INICIAR CAMPAÑA DE LLAMADAS AUTÓMATICAS ----->

<!------- INICIAR CAMPAÑA DE MENSAJES DE TEXTO ----->
<?php

/*
 * To change this license header, choose License Headers in
Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
*/

//Class Iniciar campañas de mensajes de Texto
require_once ('DataBase.php');
$connectar= DataBase::connectDB();

$campc_id=$_REQUEST['id'];

$campc_state=$_REQUEST['state'];

if($campc_state!=2){

    $consulta="select * from
public.coc_smsmatic_start('$campc_id')";
    $resultado=pg_query($connectar, $consulta);
    echo ' se ha iniciado la campaña
    <div class="container-fluid">
        <tr>

```

```
                <td>
                    <a class="btn btn-warning"
href="Mensajes_texto.php">Regresar</a>
                </td>
            <tr>';
        }
    else {
        echo 'error no se ha iniciado la campaña';
    }
?>

<!------- FIN INICIAR CAMPAÑA DE MENSAJES DE TEXTO ----->
```

El resto de código se presenta en el archivo digital.

ANEXO D.

Encuesta para comprobar las pruebas de aprobación del prototipo.

Califique la realización de las tareas de la aplicación de acuerdo a la tabla de equivalencias siendo 5 Muy Satisfecho y 1 Muy insatisfecho.

Tabla D Equivalencias del evaluador

Muy insatisfecho	1
Insatisfecho	2
Algo satisfecho	3
Satisfecho	4
Muy Satisfecho	5

Nro	TAREA	Satisfacción
1	Crear campañas estáticas.	5
2	Crear campañas personalizadas.	5
3	Crear campañas de mensajes de texto.	5
4	Listar todas las campañas existentes según el tipo de campaña (estática, personalizada o de mensajes de texto.	5
5	Eliminación de campañas tanto estáticas, personalizadas y mensajes de texto SMS.	4
6	Reporte de la campaña de llamadas Automáticas (estáticas y personalizadas)	5
7	Reporte de Campañas de Mensajes de Texto	5
8	Subir Archivos de audio y texto en Campaña Estática.	4
9	Subir Archivos en Campaña Personalizada	4
10	Borrar campañas (Estática, Personalizada y Mensajes de Texto)	5
11	Subir Archivos en Campaña de Mensaje de texto	5

SCRIPT para la instalación de las dependencias necesarias para la compilación e instalación de asterisk y postgresql.

```
#!/bin/bash

rpm -ivh epel-release-6-5.noarch.rpm

rpm -ivh elrepo-release-6-6.el6.elrepo.noarch.rpm

yum -y install sox svn httpd dhclient dhcp gcc gcc-c++ php php-pdo php-pgsql postgresql-
server postgresql-plperl postgresql-contrib postgresql-odbc postgresql-devel kernel kernel-
devel unixODBC unixODBC-devel openldap-devel speex-devel libxml2-devel ncurses-devel
openssl-devel neon-devel libical-devel libtool-ltdl-devel libcurl-devel net-snmp-devel
vorbis-tools libvorbis-devel alsa-lib-devel libtiff-devel glib2-devel spandsp-devel iksemel-
devel libsrtp-devel gmime-devel libresample-devel portaudio-devel erlang erlang-pgsql
sysstat iftop bwm-ng lm_sensors mod_xsendfile ntp telnet tftp-server zip perl-Digest-Perl-
MD5 spandsp-devel iksemel-devel libsrtp-devel gmime-devel libresample-devel portaudio-
devel mutt vsftpd perl-Text-CSV perl-DBI perl-DBD-Pg sqlite-devel perl-DateTime-Format-
DateParse perl-Time-modules drbd83-utils kmod-drbd83
```