

**BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE
PUEBLA**

Facultad de Ciencias de la Computación
Maestría en Ciencias de la Computación



Aplicación Distribuida: VoIP

**TESIS
PARA OBTENER EL GRADO DE
Maestro en Ciencias de la Computación
Ramón Mora Márquez**

**PRESENTA
L.S.C.A. Ramón Mora Márquez
ASESOR
Dr. Ivo Humberto Pineda Torres**

PUEBLA, PUE.

MAYO, 2007

Prefacio

Esta Tesis es presentada como parte de los requisitos para optar al grado académico de Maestro en Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla, y no ha sido presentada previamente para la obtención de otro título en esta Universidad u otras. La misma contiene los resultados obtenidos en investigaciones llevadas a cabo en el la Facultad de Ciencias de la Computación, durante el período comprendido entre el 1 de septiembre de 2005 y el 1 de Mayo de 2007, bajo la dirección del Dr. Ivo Humberto Pineda Torres, Profesor Titular de la Facultad de Ciencias de la Computación.

Ramón Mora Márquez

ultimarmm@hotmail.com

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN
BENEMÉRITA UNIVERSIDAD AUTONOMÍA DE PUEBLA
Puebla, Mayo de 2007.

A mi hija Naomi Abigail por ser la razón de mi vida, a mis padres Gabriel y Yaira por todo el apoyo, a Zuleima porque cambió mi vida, a mis hermanos Lizeth y Gabriel.

Índice general

Índice general	I
I Introducción	1
1 Introducción	3
1.1. Motivación	3
1.2. Antecedentes	3
1.3. Objetivo general	4
1.4. Objetivos específicos	4
1.5. Estructura del documento	5
II Marco teórico	7
2 Telefonía	9
2.1. Señalización	10
2.2. Sistema de Señalización SS7	11
2.2.1. Topología de la red SS7	11
2.2.2. Tipos de enlaces de señalización	12
2.2.3. Pila de protocolos de SS7	12
2.3. SGCP-MGCP	15
2.3.1. Modos de conexión soportados	16
2.3.2. Mapeo de dígitos	17
3 Protocolo de inicio de sesión (SIP)	19
3.1. Componentes del sistema SIP	20
3.2. Localizador de recursos uniforme - URL	21
3.3. Metodología de señalización SIP	21
3.4. Modelo de llamadas SIP	22

3.4.1. Modelo directo	22
3.4.2. Modelo con servidor proxy	23
3.4.3. Modelo con servidor de redirección	24
3.5. Protocolo de inicio de sesión para teléfonos	25
3.5.1. Flujo de llamada PSTN a teléfono IP.	26
3.5.2. Flujo de llamada de teléfono IP a red PSTN.	26
III Metodología	29
4 Modelo de Análisis	31
4.1. Casos de uso	31
4.2. Diagrama de clases	34
5 Modelo de diseño	39
5.1. Diagramas de Secuencia	39
5.2. Diagramas de colaboración	49
5.3. Diagrama de Estados	57
6 Implementación	63
6.1. Plataforma de desarrollo	63
6.2. Implementación en Java	65
6.2.1. Paquetes y bibliotecas	67
IV Pruebas y análisis de resultados	69
7 Pruebas y resultados	71
7.1. Pruebas	71
7.2. Modelo conexión directa	72
7.3. Modelo conexión con servidor proxy	72
7.4. Modelo conexión con servidor de redirección	72
7.5. Conexión múltiple	73
7.6. Identificación de tipos de llamadas	73
7.7. Registro de llamadas a la base de datos	73
7.8. Identificación de puertos	73
7.9. Resultados modelo conexión directa	73
7.9.1. Manejo de señales	74
7.10. Resultados modelo conexión con servidor proxy	78
7.10.1. Manejo de señales	79
7.11. Resultados modelo de conexión servidor de redirección	84
7.11.1. Manejo de señales	84
7.12. Resultado manejo de conexiones múltiples	88

7.13. Resultado prueba identificación de tipos de llamada	90
7.14. Resultados registro de llamadas a la base de datos	93
7.15. Resultado de identificación de puertos	93
8 Conclusiones	95
Terminología	99
Lista de Símbolos y Abreviaciones	101
Índice de figuras	103
Índice de cuadros	106
Bibliografía	107

Parte I

Introducción

Capítulo 1

Introducción

1.1. Motivación

El proyecto aquí planteado consiste en desarrollar una simulación para el control de una llamada VoIP, contemplando los elementos necesarios para el control de una llamada VoIP mediante la tarificación.

Basados en el estándar IETF - SIP(session initiation protocol)[1] que nos brinda las herramientas necesarias para el control de la llamada.

Este proyecto esta formado por dos partes; en la primera se plantea modela el proyecto al utilizar el Proceso Unificado de Desarrollo de Software (PUDS)[2] apoyado sobre UML.[2] y en la segunda parte, es la implementación basada en el modelado previo.

La principal motivación para el desarrollo de este documento es desarrollar una aplicación que sirva como base de conocimiento, y practica para poder establecer controles sobre llamadas telefónicas utilizando los estándares de VoIp y apoyo para futuros trabajos bajo este campo.

1.2. Antecedentes

Dentro de la Facultad de Ciencias de la Computación se esta desarrollando un laboratorio de Voice over Internet Protocol(VoIP) donde se esta instalando

la infraestructura para brindar servicio de Telefonía usando la infraestructura de red actual dentro de la universidad.

VoIP (Voice over Internet Protocol), se basa en la transmisión de la voz humana (con sus características particulares) a través de una red gestionada por el protocolo IP. Este sistema fue concebido para sacar provecho de la infraestructura que Internet ofrece entre otros costos bajos de acceso, comunicación a nivel mundial y poder transmitir voz. Uno de los principales logros de este sistema es el hecho de reducir extraordinariamente los costos de una comunicación por voz[?] .

Por este motivo, los sistemas basados en VoIP, se dirigen a cualquier sector que quiera reducir costos en sus llamadas: desde el usuario doméstico hasta grandes multinacionales.

La red telefónica de nuestros días, no ha mantenido desde los años ochenta, durante todo este tiempo los avances en redes de datos han sido muy importantes, tanto en fiabilidad, capacidad como en costes. Todos estos adelantos se pueden empezar a aplicar a nuestras comunicaciones de voz gracias a los últimos desarrollos presentados sobre la tecnología Voz IP. Como marco para este avance se ha formalizado un estándar que ha permitido aclarar como será el desarrollo de todas las comunicaciones, con la suficiente amplitud como para abarcar todas las posibilidades existentes.

1.3. Objetivo general

Desarrollar una aplicación como base para el control de una llamada VoIP implementando un método de tarificación basado en el estándar de VoIP - SIP. Bajo los métodos de conexión soportados por el estándar SIP.

1.4. Objetivos específicos

- Analizar los elementos involucrados en una llamada VoIP
- Analizar el estándar IETF SIP para la creación, finalización y control de llamadas VoIP.
- Analizar los elementos necesarios para la tarificación
- Desarrollar el Modelo de Análisis, que consiste en:
 1. Diseñar el Diagrama de Casos de Uso.
 2. Diseñar el Diagrama de Clases

- Desarrollar el Modelo de Diseño, que consiste en
 1. Diseñar el Diagrama de Clases Refinado
 2. Diseñar el Diagrama de Secuencia
 3. Diseñar el Diagrama de Estados
 4. Diseñar el Diagrama de Actividades
- Desarrollar Implementación basado en los modelos anteriores

1.5. Estructura del documento

Este documento esta estructurado de la siguiente manera:

En la primera parte se enfoca al marco teórico necesario para entender el problema y su solución. Es decir se referencia a los conceptos necesarios para la señalización en la telefonía, así como una revisión de las características de los protocolos SIP, MGCP, y los elementos necesarios para la creación de una llamada VoIP basado en SIP.

En la segunda parte se enfoca al modelo de análisis y diseño. Tomando en cuenta los elementos necesarios para la creación de una llamada VoIP se modelan y contemplan los pasos para la creación y tarifación. Cada uno de estos casos refleja los pasos para tomar el control de la llamada. Además se expone el diagrama de clases el cual muestra las clases que nos permitirán resolver el problema expuesto.

Los diagramas de colaboración y secuencia indicando como las clases interactúan entre si para poder resolver la tarifación.

La tercera parte se enfoca al software desarrollado y descripción de las pruebas al que él software fue expuesto y sus respectivos resultados.

Parte II

Marco teórico

Capítulo 2

Telefonía

Introducción

Desde la invención de los sistemas de telecomunicación, uno de los primeros tipos de información enviados siempre ha sido la voz; primero con el teléfono, después telefonía inalámbrica ahora conocida como telefonía celular. Con la revolución tecnológica que representaron las redes de datos se buscaron nuevas estrategias con el fin de utilizar estos sistemas de transporte de datos para transmitir voz: Voz sobre (Voice over Internet Protocol- VoIP).

VoIP es la tecnología que se basa en la transmisión de la voz humana (con sus características particulares) a través de una red gestionada por el protocolo IP.

La red telefónica pública conmutada (PSTN) de nuestros días, no ha cambiado desde hace más de 20 años como se muestra en la Figura 2.1, durante todo este tiempo los avances en redes de datos han sido muy importantes, tanto en fiabilidad, como capacidad. Todos estos adelantos se pueden empezar a aplicar a nuestras comunicaciones de voz gracias a los últimos desarrollos presentados sobre la tecnología Voz IP.

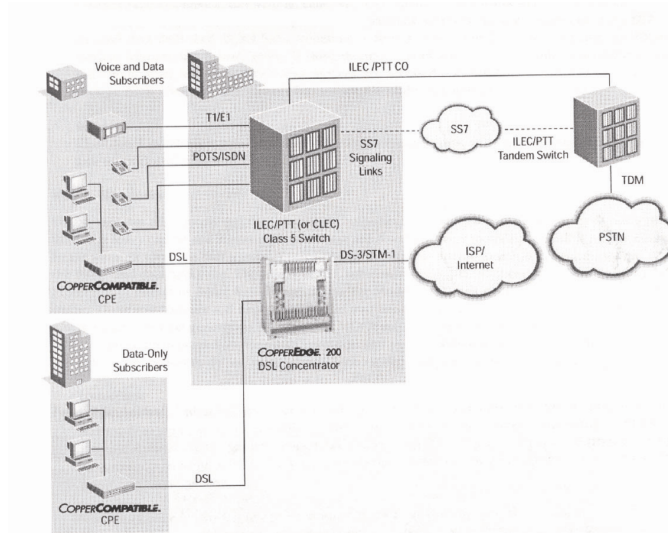


Figura (2.1) – Red telefónica tradicional (PSTN)

2.1. Señalización

El término señalización en VoIP es el método en el cual se crean las llamadas, el método tradicional en la red telefónica tradicional es marcar los dígitos en el teléfono, si la llamada es exitosa, se escuchará un tono de timbre (Ringing tone) hasta que la otra persona responda la llamada.

La señalización es un conjunto independiente de acciones para el flujo de medios, este controla el tipo de medios usados en la llamada

La señalización no se detiene necesariamente cuando la llamada ya está preparada, o hasta que uno o mas participantes cuelguen sino que puede ocurrir mientras la llamada esta activa.

Este trabajo se basa en el protocolo de señalización SIP (protocolo de inicio de sesión - session initiation protocol). Este protocolo a su vez se basado en el sistema de Señalización 7 (SS7), el cual es el mecanismo de Señalización de la red telefónica publica conmutada.

2.2. Sistema de Señalización de Canal Común Numero 7 (SS7)

Sistema de Señalización de Canal Común Numero 7 es un estándar global para comunicaciones definida por la Unión Internacional de Telecomunicaciones (ITU), define los procedimientos y protocolos por los cuales los elementos en la red intercambiaran información en la red telefónica tradicional sobre una red de señalización digital para la configuración de llamadas, ruteo y control tanto inalámbricas (Celulares) como líneas alambradas.[3]

El sistema de señalización 7 es la pila de protocolos actuales usados en la red telefónica pública de los Estados Unidos de América para establecer y terminar llamadas telefónicas. SS7 esta implementado como una red de paquetes conmutados, los cuales usan enlaces dedicados.[3]

La red SS7 tiene la función:

- Configurar una llamada básica, administrarla y finalizarla.
- Portabilidad de números locales (LNP - Local Number Portability).
- Servicios de llamadas libre de cobro (01800) y con cobro (01900).
- Telecomunicación eficiente y segura a nivel mundial.

2.2.1. Topología de la red SS7

La topología de SS7 esta formada por tres componentes distribuidos de tal forma en la red a manera de ofrecer un máximo de confiabilidad, flexibilidad y velocidad a cada uno de estos elementos se les llama punto de Señalización. En la figura 2.2 se muestran los tres puntos de Señalización.

- **Punto de servicio de conmutación - SSP (Service switching point).**
- **Punto de transferencia de señal - STP (Signal transfer point).**
- **Punto de control del servicio - SCP (Service control point).**

SSP son conmutadores que originan, terminan las llamadas. Un SSP envía mensajes de Señalización a otros SSP para configurar, administrar y liberar el circuito requerido para completar una llamada.

Un SCP envía una respuesta a él SSP que origina el número marcado con la ruta de números asociada Una ruta alternativa puede ser usado por el SSP si el número primario esta usado o la llamada no es contestada por un tiempo especificó EL tráfico de red entre puntos de señalización puede ser ruteada vía una

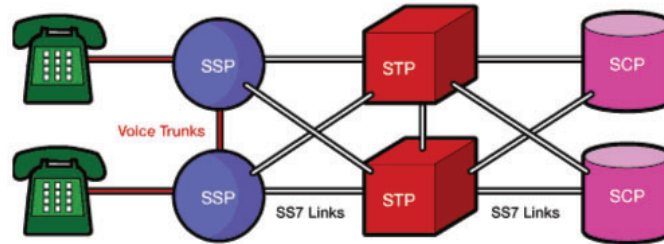


Figura (2.2) – Puntos de señalización

comutación de paquetes llamado un STP. Un STP enruta cada mensaje entrante a un enlace de señalización de salida basado en ruteo de información contenida en mensaje SS7. Porque actúa como un concentrador de red. Un STP provee utilización eficiente de la red SS7 eliminando la necesidad de enlaces directos entre puntos de señalización.[3]

2.2.2. Tipos de enlaces de señalización

Los enlaces de señalización son organizados lógicamente por tipos de enlaces de acuerdo a su uso en la red SS7. En la figura 2.3 se muestra los tipos de enlaces.

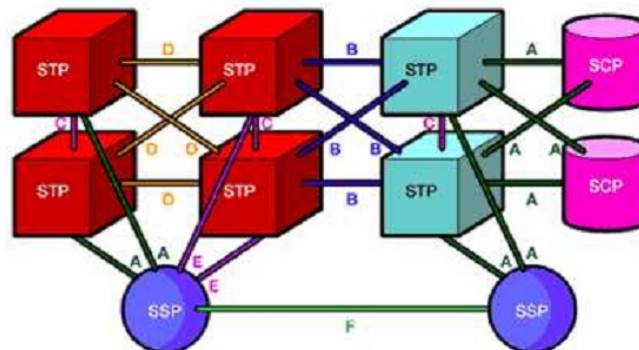


Figura (2.3) – Tipos de enlaces

2.2.3. Pila de protocolos de SS7

Las funciones de hardware y software de los protocolos SS7 son divididos en una abstracción funcional llamada “niveles”. Estos niveles están mapeados al sistema de interconexión de sistemas abiertos (OSI) definidos por la Internacional Standard Organization (ISO) como se muestra en la figura 2.4.

Cuadro (2.1) – *Tipos de enlaces.[3]*

Enlace A	Un enlace A (acceso) conecta a punto final de Señalización (SCP o SSP) a un STP. Solo mensajes destinados u originados por el punto final de Señalización son transmitidos por un enlace A.
Enlace B	Un enlace B (puente) conecta un STP a otro STP. Típicamente, un cuadrante de enlaces B interconecta puntos STPs, ejemplo. Un STP de una red a otro STP de otra red.
Enlace C	Un enlace C (cruz) conecta STPs que desarrollan funciones idénticas en un par ordenado. Un enlace C es usado solo cuando un STP no tiene otra ruta disponible para un punto de Señalización destino debido a falla de red.
Enlace D	Un enlace D (diagonal) conecta un par de STPs secundarios (local o regional) a un par primario de STPs en una configuración de enlace del cuadrante.
Enlace E	Un enlace C (extendido) conecta un SSP a un STP alternativo. enlaces E proveen una Señalización alterna si un SSP o STP no puede ser alcanzado vía un enlace A.
Enlace F	Un enlace F (completamente asociado) conecta 2 puntos finales de Señalización (SSP y SCP). enlaces F no son usualmente usados en redes con STPs. En redes sin STPs enlaces F directamente conecta puntos de Señalización.

Parte de transferencia del mensaje

► MTP nivel 1

La parte de transferencia del mensaje (Message Transfer Part MTP) está dividido en tres niveles. La parte inferior es MTP nivel 1, es equivalente a la capa física del modelo de referencia OSI. MTP nivel 1 define lo físico, eléctrico, y características funcionales de los enlaces lógicos de Señalización. Interfaces físicas definidas incluyen E-1, DS-1, V.35, DS-0, y DS-0A.

► MTP nivel 2

Asegura adecuada transmisión fin a fin de un mensaje a través de un enlace de Señalización. El nivel 2 implemente el control de flujo, validación de secuencia de mensajes y corrección de errores.

Cuando un error ocurre sobre un enlace de señalización, el mensaje (o conjunto de mensajes) son retransmitidos, el nivel 2 de MTP es equivalente

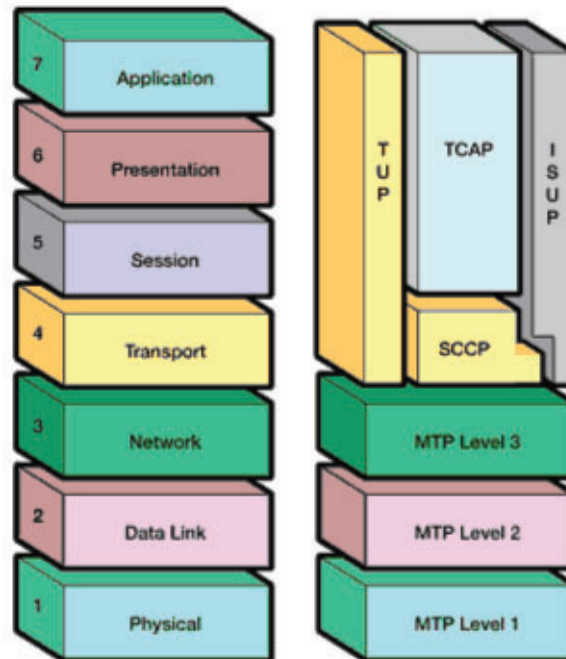


Figura (2.4) – Pila de protocolos SS7

a la capa de enlace de datos de OSI.

► MTP nivel 3

Provee ruteo de mensajes entre puntos de señalización en la red SS7. El nivel 3 provee re-ruteo de tráfico fuera de enlaces fallidos y puntos de señalización y tráfico de control cuando ocurre congestión. MTP nivel 3 es equivalente a la capa de red del OSI.

Parte de Usuario ISDN

La parte de usuario ISDN (ISDN User Part ISUP) define el protocolo usado para configurar, manejar y liberar circuitos que acarrean voz y datos entre líneas de intercambio terminales. ISUP es usado para ambas llamadas, ISDN y no-ISDN. Sin embargo, llamadas que originan y terminan en el mismo conmutador no usan señalizaron ISUP.

Parte de usuario de teléfono

En algunos partes del mundo, ejemplo China, Brasil, la parte de teléfono de usuario (Telephone user part- TUP) es usado para soporte básico de manejo de llamadas y terminación. TUP maneja circuitos analógicos solamente. En muchos países, ISUP ha reemplazado TUP para manejo de llamadas.

Parte del control de conexión de la señalización

La parte de control de conexión de la señalización (Signaling Connection Control Part - SCCP) provee servicios de red orientados a conexión y sin conexión y capacidades de títulos globales de traducción (GTT) sobre MTP nivel 3. Un título global es una dirección el cual es traducido por un SCCP en un código de punto de destino y número de subsistema. Un número de subsistema únicamente identifica una aplicación en el punto de señalización de destino. SCCP esta usado como la capa de transporte para servicios basados en TCP.

Parte de aplicaciones de capacidades de transacciones

La parte de aplicaciones de capacidades de transacciones (Transaction Capabilities Applications Part - TCAP) soporta el intercambio de datos no relacionados a circuitos entre aplicaciones a través de redes SS7 usando un servicio sin conexión SCCP. Consultas y respuestas que se envían entre SSPs y SCPs son acarreadas en mensajes TCAP.

La parte de operaciones, mantenimiento y administración *OMAP* y ASE .OMAP y ASE son áreas para definiciones futuras. Servicios OMAP pueden se usados para verificar bases de datos de ruteo y detectar problemas de enlaces SS7 sobre IP.

La carga de las llamadas de voz de la red publica telefónica *PSTN* las compañías telefónicas la descargan a la red con el protocolo de voz sobre Internet *VoIP* debido a que es mas barato acarrear trafico de voz sobre redes de paquetes que sobre redes de conmutación de circuitos. En el futuro, las redes de telefonía IP están esperando innovación de servicios multimedia mientras trabajan similarmente con las redes telefónicas. Una red VoIP acarrea tráfico de voz mas barato que una red de conmutación de circuitos debido a que la redes de Telefonía IP hacen posible el mejor uso del ancho de banda disponible.

2.3. Protocolo de Control de Pasarela Simple - Protocolo de Control de Pasarela de Medios

Protocolo de control de pasarela de medios (Simple Gateway Control protocol - SGCP) definido por IETF propone definir un protocolo basado en texto para controlar las pasarelas de medios (Media Gateways- MG) para el control de elementos de la llamada externos como los protocolos de control de pasarela de medios (Media Gateway Controller - MGC).

Cuadro (2.2) – *Verbos usados por S/MGCP*

Verbo	Código	Dirección
CreateConnection	CRCX	MGC→MG
ModifyConnection	MDCX	MGC→MG
DeleteConnection	DLCX	MGC→MG o MG→MGC
NotificationResquest	RQNT	MGC→MG
Notify	NTFY	MGC→MG
AuditEndPoint	AUEP	MG→MGC
AuditConnection	AUCX	MGC→MG
ReStartInProgress	RSIP	MG→MGC

SGCP asume una arquitectura de control de llamada en la cual la inteligencia del mismo esta fuera de la pasarela de medios y manejado por elementos de control de llamada externos. En otras palabras las llamadas son controladas y establecidas por Controlador de pasarela de medios (Media Gateway Controller - MGC), es externo y puede estar separado geográficamente de la pasarela de medios. SGCP y MCGP son protocolos sin estados, Indicando que el protocolo no requiere una máquina de estados para describir la secuencia de transiciones entre dos entidades de señalización y no mantiene memoria de la transacción anterior entre el MGC y el MG.

MGCP relacionado a la pila de protocolos IP se localiza en la capa de aplicación encima de UDP.

MGCP desarrolla sus funciones en secuencia de comandos y reconocimientos (ACK) mandatorios. El MGC es responsable de enviar comandos MGCP al Terminal y recibir los reconocimientos (ACK) de cada comando. Los comandos contienen una solicitud "verbo", que es una acción a realizar y parámetros adicionales, son 8 comandos en el vocabulario de MGCP[4] mostrados en la Tabla 2.2[3].

2.3.1. Modos de conexión soportados

El MGCP permite por una variedad de modos de conexión, la Tabla 2.3 muestra los tipos de conexión soportados. Un modo de conexión puede ser cambiado a cualquier otro modo via el comando MDCX y parámetros apropiados, que puede incluir un nuevo descriptor de sesión.

Cuadro (2.3) – Modos de conexión para S/MGCP

Modo	Significado
M:sendonly	La pasarela solo envía paquetes sobre esta conexión
M:reconly	La pasarela solo recibe paquetes sobre esta conexión
M:sendrcv	La pasarela solo envía y recibe paquetes sobre esta conexión
M:inactive	La pasarela no envía ni recibe paquetes sobre esta conexión
M:netwloop	La pasarela se coloca en modo loopback en la red
M:netwtst	La pasarela se coloca en modo de prueba continua en la red

2.3.2. Mapeo de dígitos

Los dígitos son cadenas de texto ASCII, estos son cargados en el media gateway para poder hacer eficiente el uso de mensajes con el MGC durante la recolección de dígitos.

El propósito de un mapa de dígitos es permitir validar cuando una secuencia de dígitos es correcta o no cuando es marcado, antes del intercambio de mensajes es realizado con el Controlador de pasarela de medios

En el servicio telefónico tradicional, el intercambio local analiza la secuencia de dígitos cuando es marcada y si es correcto el proceso de la llamada se inicia. Sin embargo, en señalización basada en IP pueden ocurrir muchos mensajes entre el MGC y el MG requeridos para poder comenzar el procesamiento de la llamada después que los dígitos han sido marcados, analizados y una decisión ha sido tomada por el MG, si ha marcado un número válido.

El sistema de mapeo de dígitos para Norte América se muestra a continuación:

([2-9]xxxxxxx — *xx — [01][2-9]xxxxxxxxxxx — 00 — [044]xxxxxxxxxxx — [00] [1-9]xxxxxxxxxxx)

- [2-9]xxxxxxx - Un número de 7 dígitos para llamadas locales
- xx - Solicitar un servicio
- [01][2-9]xxxxxxxxxxx - Llamada larga distancia nacional
- 00 - Llamada larga distancia por operadora
- [044]xxxxxxxxxxx - Llamadas a celulares locales
- [00] [1-9]xxxxxxxxxxx - Llamadas larga distancia internacional

El MGC tiene que hacer el análisis de los dígitos a este nivel para cada uno de los miles de puntos finales a los cuales sirve.[5]

Capítulo 3

Protocolo de inicio de sesión

Introducción

El protocolo de Iniciación de Sesión (*SESSION INITIATION PROTOCOL - SIP*) es un protocolo de comunicación estándar de multimedia de IETF, basado en texto para crear y controlar las sesiones con dos o más participantes, que incluye la voz, el vídeo, y la comunicación de datos sobre redes de conmutación de paquetes [5].

SIP es descrito en RFC 3621 - SIP: protocolo de inicio de sesión, en la figura 3.1 se observa la pila de protocolos SIP. Se apoya de una serie de protocolos para establecer las comunicaciones, entre estos están LDAP (Lightweight Directory Access Protocol) para la búsqueda de identidades de usuarios o búsquedas de direcciones, DNS(Domain Name Server) para determinar los saltos necesarios para llegar al destino, TRIP(Telephony Routing over IP) permite el intercambio de información de ruteo para evitar la sobrecarga o replicas de puertas de enlace usando protocolos de Internet. SIP puede inicializa los servicios del RSVP (Reservation Protocol) para la reserva de recursos de red y así evitar fluctuaciones en la señales de voz.

SIP es un protocolo de transporte basado en la arquitectura cliente-servidor y este puede trabajar sobre ambos protocolos TCP o UDP, pero para mayor implementación es usado sobre UDP por simplicidad y velocidad.[6]

Las invitaciones de SIP suelen crear las descripciones de las sesiones que permiten a participantes estar de acuerdo sobre un conjunto de tipos de medios de comunicación compatibles tales como RTP(Real-Time Transport protocol)[7], RTCP(Real-Time Transport Control protocol).[7] SIP hace uso de elementos servidores para ayudar a peticiones de ruteo, autenticar y autorizar a usuarios para servicios, poner en práctica las políticas de ruteo de llamadas, y proporcionar características a los usuarios. SIP también proporciona una función de registro que permite cargar posiciones de usuarios como servidores proxy.[6]

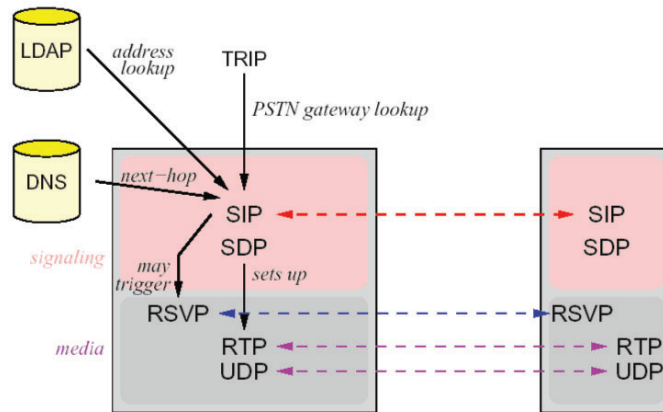


Figura (3.1) – Arquitectura SIP

3.1. Componentes del sistema SIP

Un sistema SIP consiste de agente de usuario (User Agents - UA) y uno o más servidores. El sistema SIP pueden ser segmentos de red dedicados, segmentos de red conectados al Internet públicos, o grupos lógicos de dispositivos en una red empresarial, que soporte otro protocolo de señalización.

Los servidores pueden operar dando servicios proxy o de redirección. Servidores proxy ejecutan señalización de llamadas entre las partes a las cuales sirve. Servidor de redirección determina la localización actual de la persona llamada y instruye inicializar una señalización con la persona llamada directamente. Ambos conceptos de señalización son similares a los modelos H.323 gatekeeper-routed y direct-routed respectivamente. En ambos casos el servidor busca por la localización del usuario y en el caso del modelo de redirección este informa al llamador la localización actual del llamado.

Existen cinco aspectos de SIP para establecer una llamada y mantenerla.

1. **Configurar la Llamada:** SIP es auto contenido en establecer la configuración ya sea punto a punto o conferencia multipunto o una simple llamada.
2. **Servicios de Localización de Usuarios:** Los usuarios tiene la habilidad de moverse a otros lugares y acceder sus teléfonos de lugares remotos.
3. **Capacidades de los Usuarios:** Determinación de los medios y parámetros a ser usados. SIP usa el formato del protocolo SDP (Session Description protocol)[8] para negociar los parámetros de los medios.
4. **Disponibilidad de los usuarios:** Determinación de la voluntad de los usuarios a aceptar una comunicación. SIP define muchos códigos de repuestas para proveer información detallada sobre la disponibilidad de un usuario.
5. **Manejo de la Llamada:** Incluye transferencia de una llamada establecida, aspectos de la llamada telefónica y una terminación simple de la llamada.

3.2. Localizador de recursos uniforme

SIP utiliza un URL (Uniform Resource Locators) para identificar la fuente, el destino actual y el último destino, además de especificar direcciones de redirección

Usando el siguiente formato `recipiente@host.com` que puede ser independiente a el anfitrión o dependiente.

Números de puertos UDP pueden ser incluidos en los URL.

Los puertos UDP para SIP son 5060 y 5061 para SIP - Secure.

La funcionalidad y operación correcta de SIP no depende en otros protocolos tales como RSVP(Reservation protocol)[7], RTCP(Real time control protocol)[7] o RTSP(Real Time Streaming protocol)[7] usados para establecer conferencias multimedia, QoS y Señalización.

3.3. Metodología de señalización SIP

SIP utiliza seis métodos para señalización que se muestran en la tabla 2.4, estos son *INVITE*, *ACK*, *OPTIONS*, *BYE*, *CANCEL*, y *REGISTER*.

Cuadro (3.1) – Métodos del protocolo SIP

INVITE	Este es el primer mensaje enviado por el llamador en el ciclo de procesamiento de una llamada. Este contiene información en la cabecera SIP, la cual identifica al llamador, Call-ID, persona llamada, número de secuencia de llamada, entre otras cosas. Básicamente, indica que una llamada está siendo inicializada. El mensaje INVITE usualmente contiene una descripción en formato SDP[s], aquellos seleccionados devuelven un código de éxito (200) en el correspondiente mensaje.
ACK	El agente de la llamada responde con ACK solo a la solicitud de INVITE que ha sido exitosamente aceptada con un código 200. ACK indica que la persona llamada ha recibido confirmación a la petición de INVITE.
OPTIONS	Este mensaje es enviado para solicitar las capacidades de un agente de usuario. Esto determina que tipos de medios soporta el usuario remoto para establecer la llamada.
BYE	El cliente envía este mensaje al agente de la llamada para liberar la llamada. Los terminales terminan el flujo de medios y se considera la llamada terminada. Un regreso de BYE de la otra parte no es necesaria.
CANCEL	Este método cancela una petición en progreso, este método debe explícitamente identificar la llamada mediante su Call-ID, Call sequence, origen y destino en la cabecera SIP.
REGISTER	Un cliente usa el método REGISTER para registrar la dirección en un servidor SIP.

3.4. Modelo de llamadas SIP

3.4.1. Modelo directo

En la figura 3.2 se muestra el modelo de una llamada básica exitosa, en este caso se toma como ejemplo a `alice@dflx.com` a `bob@remote.com`.

El flujo de la llamada toma lugar cuando el agente de usuario opera directamente y manda una solicitud *INVITE* de Alice a Bob, Esto es que el agente de usuario de Alice ha resuelto el nombre de `bob@remote.com` en una dirección IP vía DNS. La solicitud *INVITE* es enviado al puerto UDP 5060 y contiene la información Call-ID, Cseq, y SDP[s] del formato de los medios, además de información de ruteo como origen-destino.

Cuando la llamada ha alcanzado el Terminal remoto y el teléfono suena, una respuesta es generada y es enviada a Alice indicando un *RINGING(180)*.

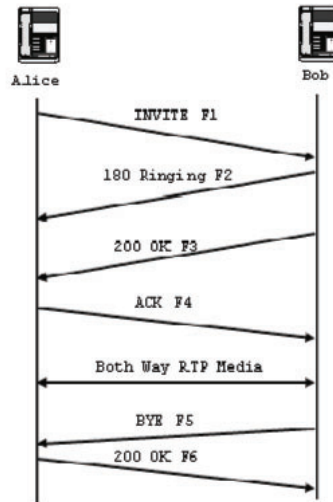


Figura (3.2) – Modelo de llamada básica SIP[8]

Cuando Bob toma el teléfono una respuesta 200 es enviada al agente de usuario de Alice. El agente de usuario (UA) envía una solicitud de reconocimiento *ACK* para la respuesta exitosa a la solicitud de *INVITE*, en este punto empieza el flujo de medios en las direcciones de transporte de los dos terminales.

Esto es muy simple, el transporte de flujo de medios mediante RTP, con RTCP proveyendo la monitorización de la calidad de la conexión y estadísticas.

Una solicitud *BYE* de cualquiera de las partes termina la llamada, debido a que los mensajes son en UDP, no hay más que hacer.

3.4.2. Modelo con servidor proxy

Una llamada SIP con servidor proxy es un poco mas complicada en la Figura 3.3 se muestra el modelo para una llamada con servidor proxy.

En este caso una solicitud *INVITE* es enviada al servidor SIP de Bob, el flujo de la llama en el servidor proxy primero se necesita encontrar si Bob esta registrado en el servidor, y si lo cual es su localización actual. El servicio de localización es parte del diseño de la plataforma de cualquier red que soporte

movilidad.

Una vez que el servidor SIP ha localizado a Bob, propaga la petición sin alterar los campos de la cabecera, excepto solo para añadir los campos vía. Esto puede considerarse que muchos servidores están involucrados en el proceso.

El servidor ahora requiere mantener algún estado de información sobre la llamada esto es posible relacionando el número de secuencia Cseq, Call-ID y otros elementos de la cabecera mientras estos pasan. Este envía una respuesta informativa *TRYING (100)*. Cuando Bob toma el teléfono la respuesta *RINGING* regresa al servidor. Ambos tendrán las entradas en el campo vía en el mensaje de respuesta a la petición. El resto del flujo de la llamada hasta el *ACK* es como se explicó en el esquema anterior.

Cuando la llamada está establecida, la voz se transmitirá (medios) entre Alice y Bob, hasta el punto que alguno desee terminar la llamada. Finalmente la terminación de la llamada es mediante una propagación de la solicitud *BYE*.

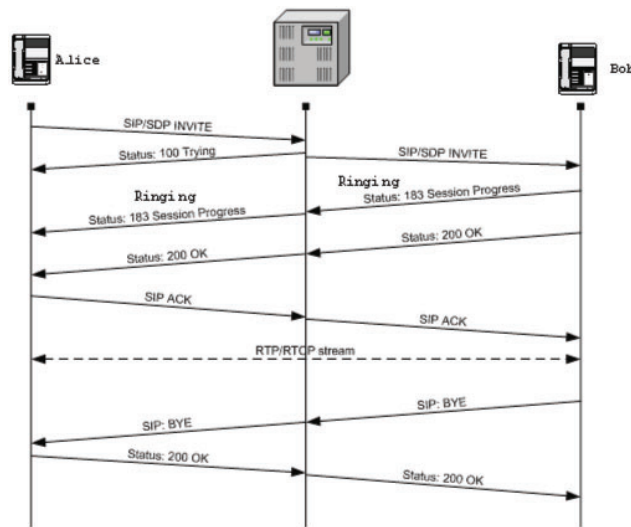


Figura (3.3) – Modelo de llamada básica SIP con servidor proxy[8]

3.4.3. Modelo con servidor de redirección

El modelo de servidor de redirección se reduce a un modelo de conexión directa después de que el mensaje de *INVITE* inicial es enviado al servidor SIP. El servidor regresará una respuesta de redirección al mensaje *INVITE*, con código 301 o 302, indicando que Bob se localiza en la parte indicada por el campo *contact*

del cuerpo del mensaje. MGC de Alice cierra la señalización con el servidor SIP e inicia otra petición INVITE a la localización indicada por el servidor, a partir de ese momento el flujo de la llamada es el modelo directo. En la figura 3.4 se muestra el funcionamiento de este modelo.

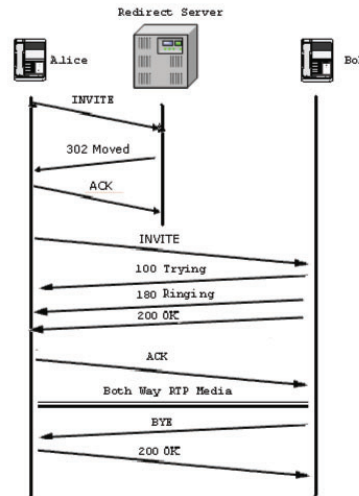


Figura (3.4) – Modelo de llamada básica SIP con servidor de redirección[8]

3.5. Session Initiation Protocol for Telephones (SIP-T)

Realizando la señalización de llamada de telefonía y el transporte de los medios de comunicación asociados de audio sobre IP crea ventajas significativas sobre la telefonía tradicional, una red de VoIP no puede existir en el aislamiento de redes tradicionales telefónicas. Es vital para una red de telefonía de SIP para trabajar con la red telefónica tradicional *PSTN*. [9]

Una característica importante de cualquier red de telefonía de SIP es la transparencia con respecto a PSTN. Servicios de telecomunicación tradicionales como la espera de llamada, números gratuitos, etc., puestos en práctica en PSTN como en SS7 deberían ser ofrecidos a una red de SIP en alguna manera que excluya cualquier diferencia debilitante en la experiencia del usuario mientras no limite la flexibilidad de SIP.

SIP-T (SIP para Teléfonos) el esfuerzo proporciona un marco para la integración de señalización de la telefonía tradicional en mensajes SIP. SIP-T proporciona dos características por técnicas como son la encapsulación y la traducción respectivamente.

- **Encapsulación:** Una pasarela SIP-ISUP, mensajes SS7 ISUP son encapsulados dentro de SIP para que esta información para que esta información necesaria para los servicios no sea descartada en una solicitud SIP. Sin embargo, intermediarios como servidores Proxy que realizan ruteo para las peticiones SIP no pueden entender ISUP, así simultáneamente alguna información crítica es traducida de un mensaje ISUP a su correspondiente en cabecera SIP para poder determinar como la petición SIP será enrutada.
- **Traducción:** La pasarela SIP-ISUP, se encarga de realizar la traducción de la Señalización SIP con su correspondiente señalización ISUP.

Se debe notar que existen muchos modos de señalización que son usados en la telefonía (SS7 ISUP, BTNUP, Q.931, MF). Por el momento sólo se describe el comportamiento a través de las interfaces ISUP-SIP. Debido a que muchas de estas tecnologías son sistemas de señalización propietarias.

Pueden existir muchos flujos de llamadas en los cuales SIP-T puede ser usado debido a que las llamadas se pueden originar en la red telefónica o en la telefonía IP, y ser dirigidas a la red telefónica tradicional o a la red IP.

3.5.1. Flujo de llamada PSTN a teléfono IP.

En la figura 3.5 se muestra el flujo que inicia en la red telefónica tradicional dirigida a un teléfono VoIP. En este el sistema SS7 crea una conexión al MGC la cual su función es traducir y encapsular a SIP, creando la petición *INVITE*, a partir de este momento el flujo de la llamada entre el MGC, el servidor Proxy y el teléfono SIP se realiza como en el esquema anteriormente descrito. Básicamente el MGC traduce las peticiones de SS7-ISUP a SIP y viceversa.

3.5.2. Flujo de llamada de teléfono IP a red PSTN.

Una llamada originada por un teléfono SIP y terminada en la red PSTN a diferencia del anterior no hay encapsulación ISUP en la solicitud, el MGC realiza las traducciones del contenido de la cabecera SIP a valores para los parámetros de ISUP.

En la Figura 3.6 se muestra el flujo de la llamada, Al pasar por el MGC las señales SIP son transformados en valores de ISUP, al regresar al emisor en la red SIP son convertidos nuevamente al atravesar el MGC a su valor correspondiente en SIP a excepción de la señal de reconocimiento.

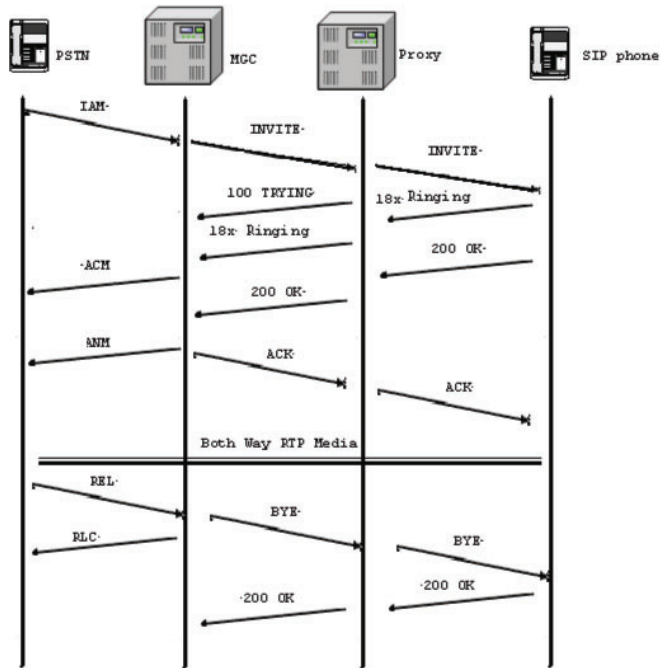


Figura (3.5) – Modelo de llamada PSTN a Teléfono VoIP[9]

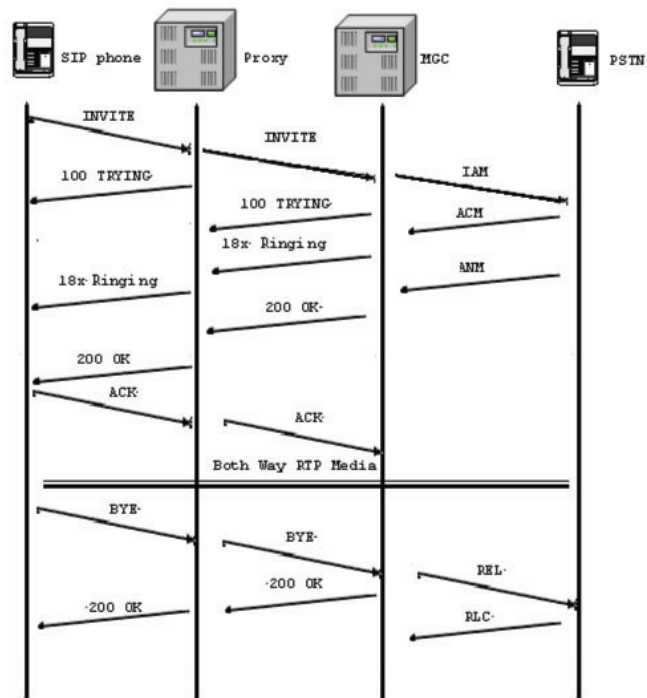


Figura (3.6) – Modelo de llamada Teléfono VoIP a PSTN[9]

Parte III

Metodología

Capítulo 4

Modelo de Análisis

Introducción

Un modelo es una abstracción del sistema especificando, el sistema de modelado desde cierto punto de vista y en determinado nivel de abstracción. A menudo se representa gráficamente mediante uno o mas diagramas, y tienen por objetivo expresar la esencia de algunos aspectos de lo que se está haciendo sin especificar detalles innecesarios. Su propósito es expresar sobre problemas y soluciones sin desviarse de los objetivos

Continuación se muestra el modelo de análisis siguiendo la metodología que propone el Lenguaje Unificado de Modelado expresar los distintos modelos que se producen en el proceso de desarrollo expresando los distintos modelos que se producen en el proceso de desarrollo.

4.1. Casos de uso

En esta parte se encuentra el modelo de análisis y diseño del sistema para la simulación del control de una llamada VoIP basado en los modelos de llamadas anteriormente descritos.

En la figura 4.1 se muestra el caso de uso del sistema, en el podemos observar a los diferentes actores que intervienen para llevar el control de la llamada VoIP en alguno de sus métodos de conexión.

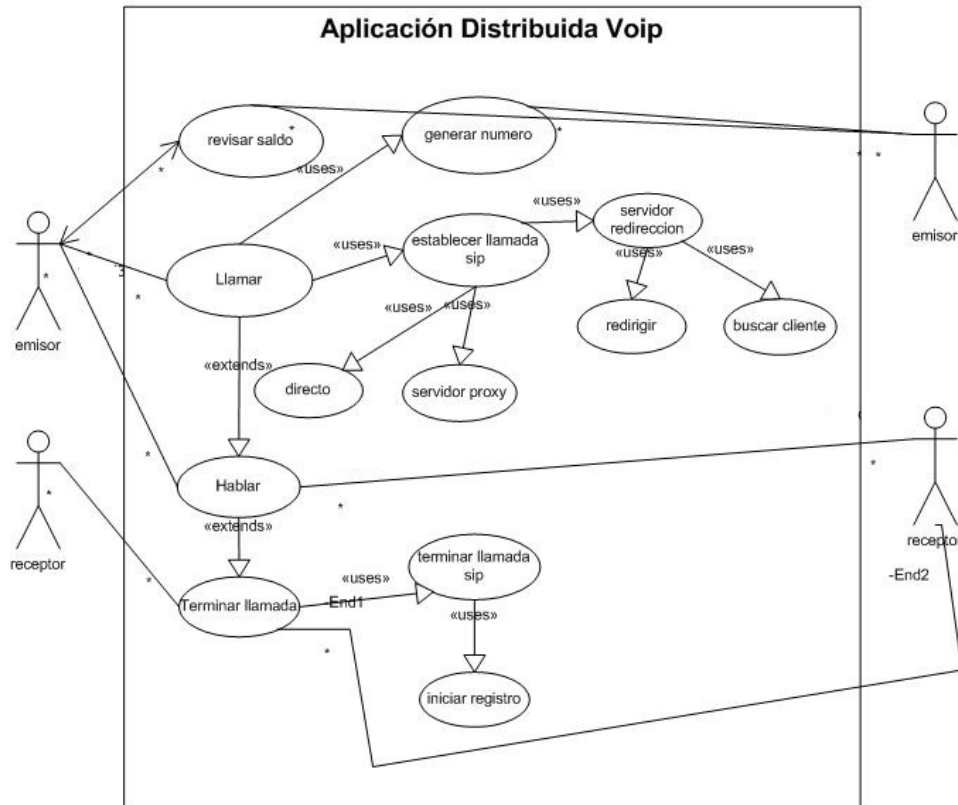


Figura (4.1) – Casos de uso del sistema

El actor emisor es el encargado de inicializar una llamada hacia un receptor, el cual recibe esta.

El emisor tiene las posibilidades de revisar su saldo actual, crear una llamada hacia algún receptor, una vez establecida esta llamada puede hablar por medio del sistema, y al finalizar terminar una llamada.

En condiciones normales el emisor crea una nueva llamada con el caso de uso llamar, esto incluye el caso de uso generar número, el cual proporciona un número telefónico correcto y valido. Esta llamada se establece mediante uno de los 3 métodos de conexión, que puede ser directo, mediante servidor proxy o servidor de redirección.

Una vez establecida la conexión se procede a hablar que en este caso implica la transmisión de medios.

Para terminar el emisor puede finalizar la llamada en el momento que lo desee así como también lo puede realizar el receptor.

Métodos de conexión

Directo

- A)** Permite la conexión directa entre el emisor y el receptor sin intermediarios esta opción la puede elegir el emisor al momento de establecer la llamada sip.

B) Servidor proxy

En primer lugar se encuentra el servidor proxy que establece completamente la conexión entre el emisor y el receptor, tomando todas las señales necesarias para ello, como se describió en la sección 3.4.2(pag. 23). Además el servidor proxy también puede terminar la llamada actual que este administrando.

C) Servidor de redirección

El servidor de redirección su función básica es la de servir como directorio ya que nos permite conocer la identidad de algún número al que se desea llamar y el método de conectarse.

Sus funciones básicamente son buscar cliente, y redirigir.

Buscar cliente permite a un emisor el poder conocer el destino al que desea llamar.

Una vez encontrado el receptor se envía una señal de redirección hacia el emisor.

El caso de uso registro se encarga de mantener la información correspondientes al tiempo de las llamadas y la tarificación enviándolas a un paquete de bases de datos.

4.2. Diagrama de clases

En la figura 4.2 se encuentra el diagrama de clases con las clases necesarias para llevar a cabo las operaciones descritas en los casos de uso.

Las clases están basados en el modelo jerárquico distinguiendo al primer bloque como clases pertenecientes a nivel de interfaz gráfica de usuario estas tienen contacto con la clase control 1 la cuál es el límite entre la interfaz gráfica y el control lógico del sistema que se encuentra en el segundo nivel. La clase control 2 es el límite entre el control lógico del sistema y los paquetes de comunicación y manejo de datos externos al sistema. Entre estos están los manejadores de sockets para la creación de servidores y manejadores de bases de datos y la interfaz con el la red telefónica tradicional.

Las clases principales son emisor y receptor ya que estos son los encargados de inicializar las actividades del sistema, ambas clases descienden de la clase terminal la cual contiene una interfaz para la creación de llamadas, heredando a las clases receptor y emisor, aunque descienden de la misma clase madre tienen comportamiento distintos ya que el emisor es capaz de generar números y crear llamadas.

El receptor tiene la misma interfaz y su función es la de recibir llamadas telefónicas.

Ambas clases tiene relación con otras específicamente con la clase llamada activa y sesión VoIP, como se muestra en la figura 4.3 el emisor inicia una llamada activa para la creación de una nueva llamada, esta a su vez controla el tiempo que esta hablando y crea una sesión VoIP para lograr comunicarse con el receptor en cualquiera de los métodos de conexión de SIP-VoIP.

En la figura 4.4 se observa la relación entre la clase llamada activa y el paquete de base de datos que se encarga del registro de las llamadas.

También se encuentra las clases correspondientes al envío de medios y los servidores.

Los servidores funcionan como el estándar SIP, teniendo relación con las clases sesión VoIP.

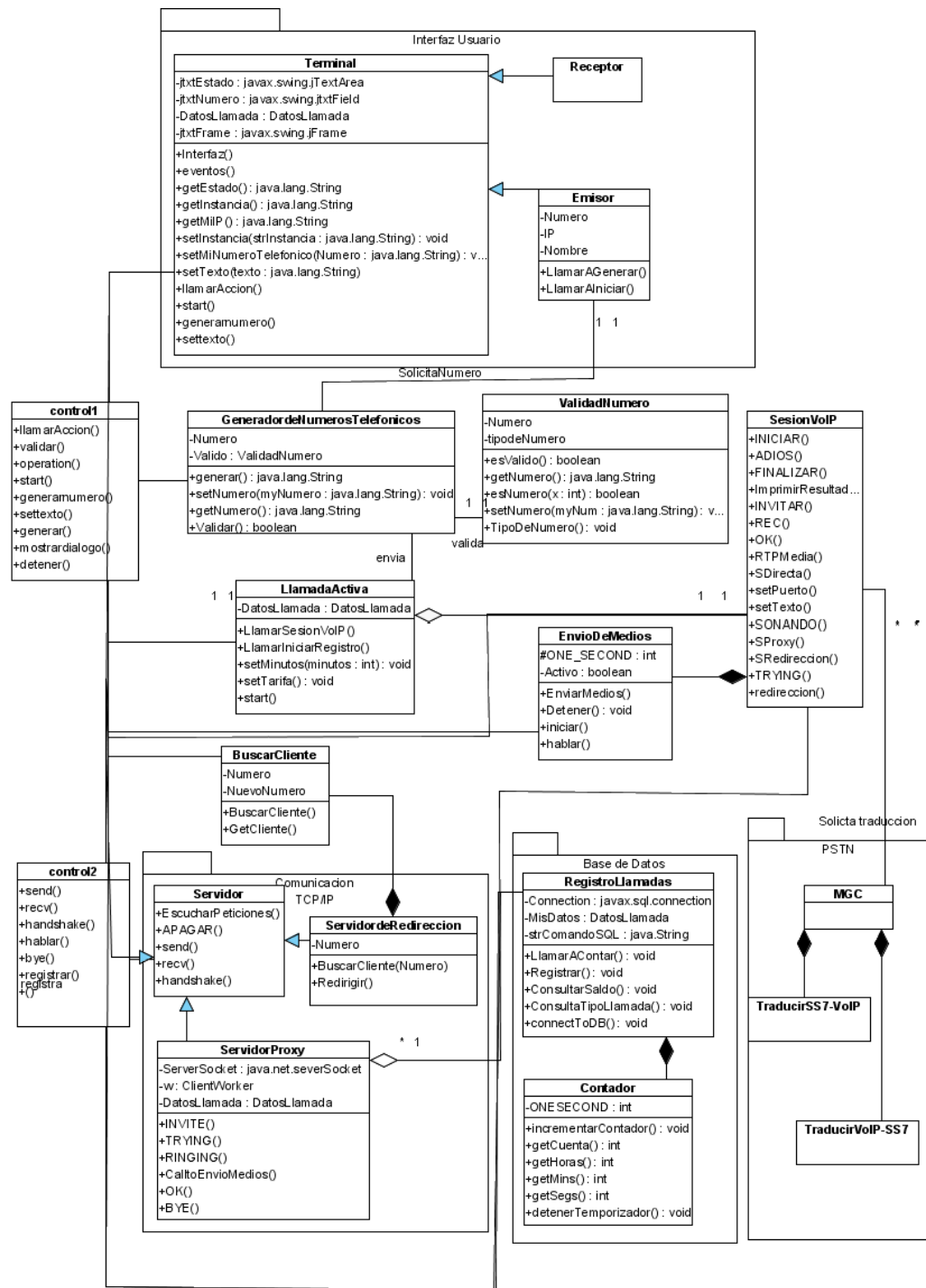


Figura (4.2) – Clases del sistema

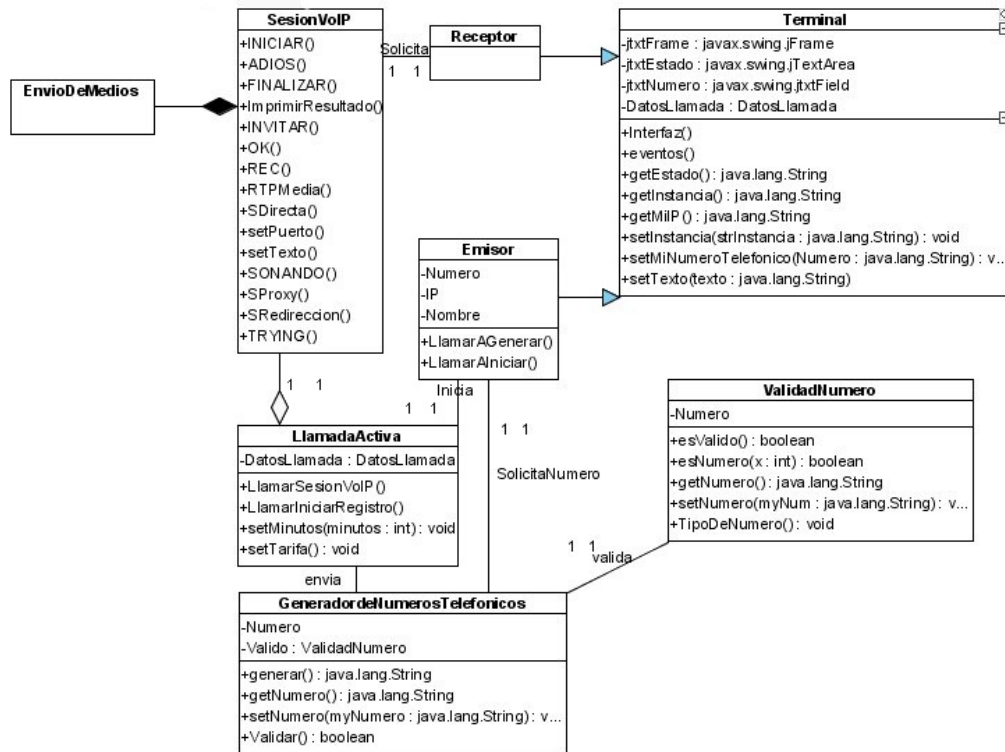


Figura (4.3) – Clases emisor, receptor, llamada activa y sesión VoIP

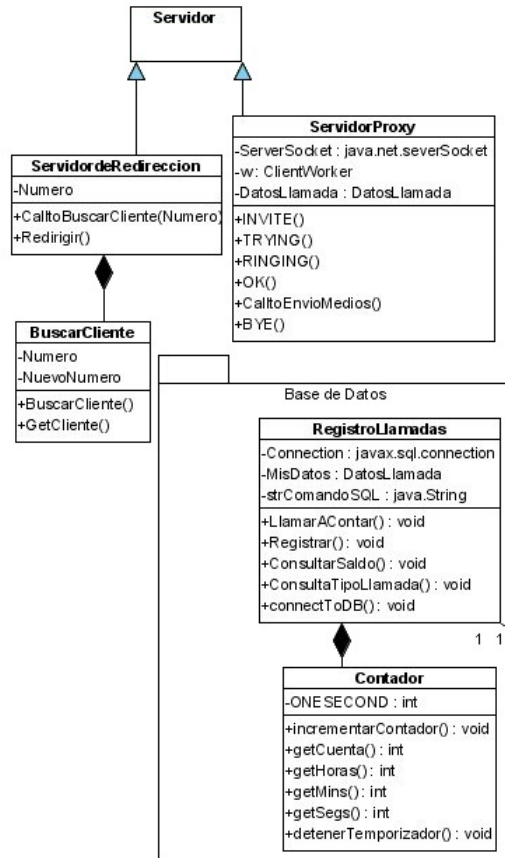


Figura (4.4) – Clases servidor proxy y servidor de redirección

Capítulo 5

Modelo de diseño

Introducción

El modelo de análisis muestra un panorama estático de cómo es el sistema, pero para poder saber como interactúan los diversos elementos es necesario el modelo de diseño.

En esta sección se describirán diferentes diagramas que permiten entender el comportamiento del sistema en general y su relación con los casos de uso y clases expuestos.

El modelo de diseño esta formado por los diagramas de secuencia, diagramas de colaboración y diagramas de estado.

5.1. Diagramas de Secuencia

Los diagramas de secuencia muestra la forma en que los objetos se comunican entre sí al transcurrir el tiempo.[10]

A continuación se muestran los diagramas de secuencia correspondientes a los casos de uso y una breve explicación de su funcionamiento.

I) Operación llamar

Esta operación se especifica en la figura ?? **Casos de uso del sistema** (pag. 32, sección “**Casos de uso**”), el diagrama de secuencia se muestra en la figura 5.1 y se detalla a continuación: El emisor es responsable de inicializar la comunicación, en un inicio genera un número telefónico este es validado y posteriormente devuelto a la clase emisor.

Seguido inicializa una llamada con la clase llamada activa, que se encarga de generar una sesión SIP usando el método de conexión adecuado.

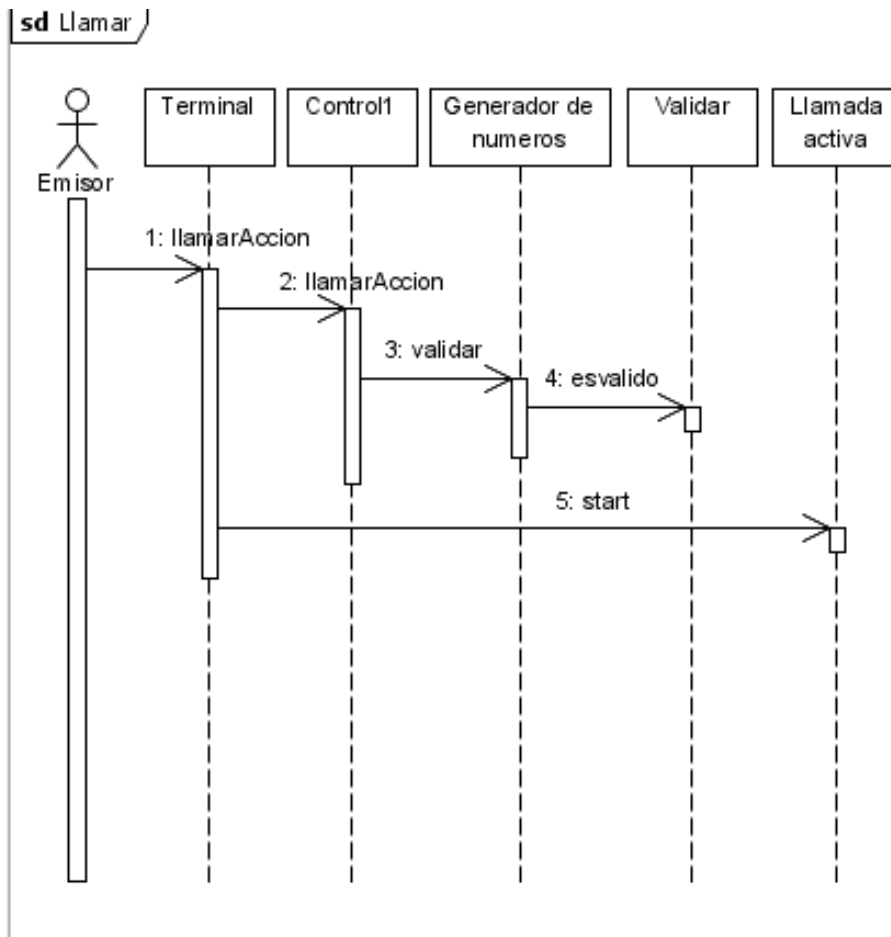


Figura (5.1) – Operación llamar

II) Operación hablar

Esta operación se especifica en la figura 4.1 “**Casos de uso del sistema**” (pag. 32, sec. “**Casos de uso**”), el diagrama de secuencia se muestra en la figura 5.2 y se detalla a continuación: En este diagrama se muestra como se envía

los medios.

Intervienen las clases de emisor y receptor y una tercera clase que dedica exclusivamente a la transmisión de los medios.

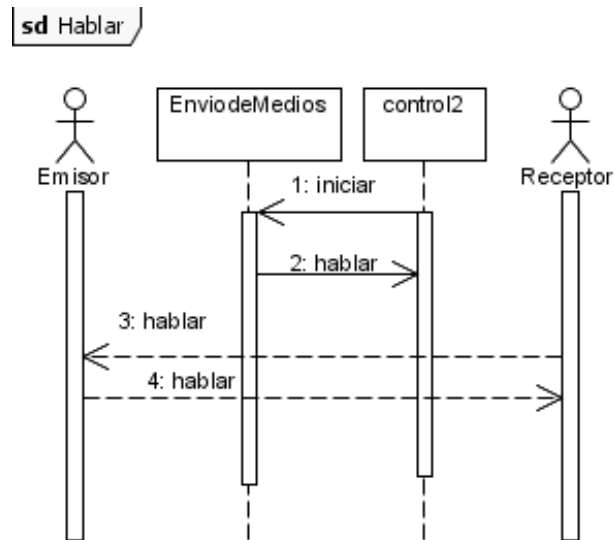


Figura (5.2) – Operación hablar

III) Operación terminar llamada

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.3 y se detalla a continuación: El emisor indica que ya no quiere transmitir más medios entonces envía una señal de despedida y se cierra la comunicación.

IV) Operación generar número

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.4 y se detalla a continuación: El emisor genera un número de acuerdo a la estructura presentada anteriormente

V) Operación establecer llamada SIP

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.5 y se detalla a continuación:

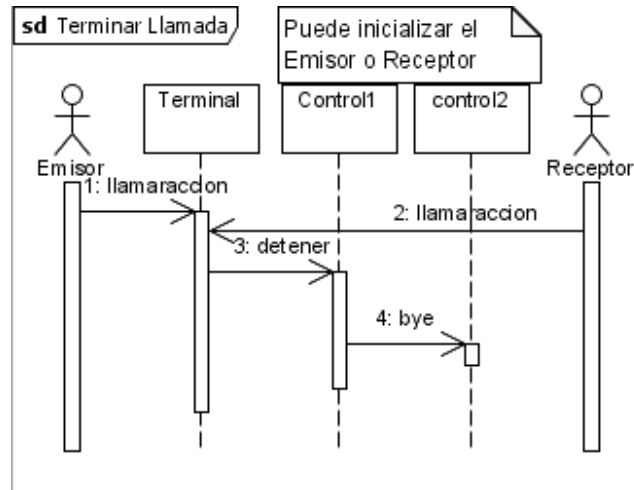


Figura (5.3) – Operación terminar llamada

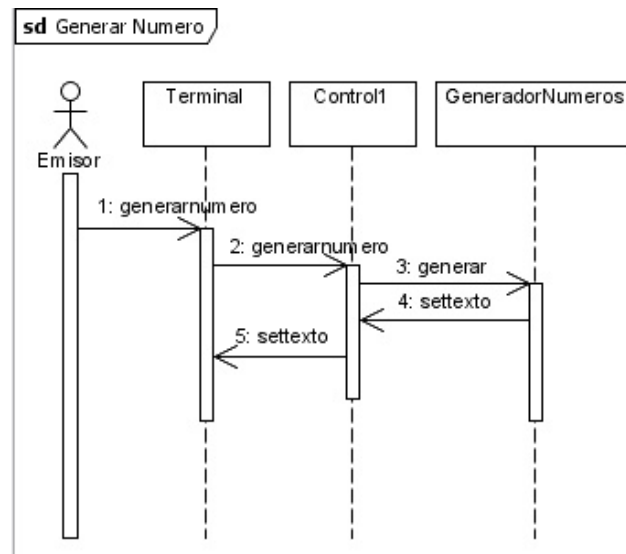


Figura (5.4) – Operación generar número

Estable una conexión mediante el servidor Proxy, utilizando el modelo de conexión mediante servidor Proxy hasta crear la comunicación y pase a la transmisión de medios.

VI) Operación terminar llamada SIP

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.6 y se detalla a continuación:

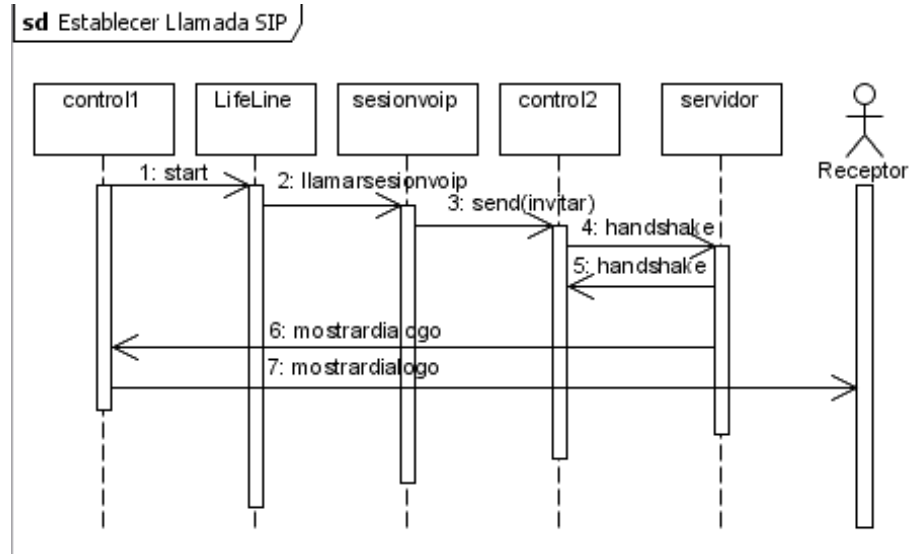


Figura (5.5) – Operación establecer llamada SIP

El emisor desea terminar la llamada y manda una señal de finalizar, el servidor la toma y se encarga de terminar la llamada.

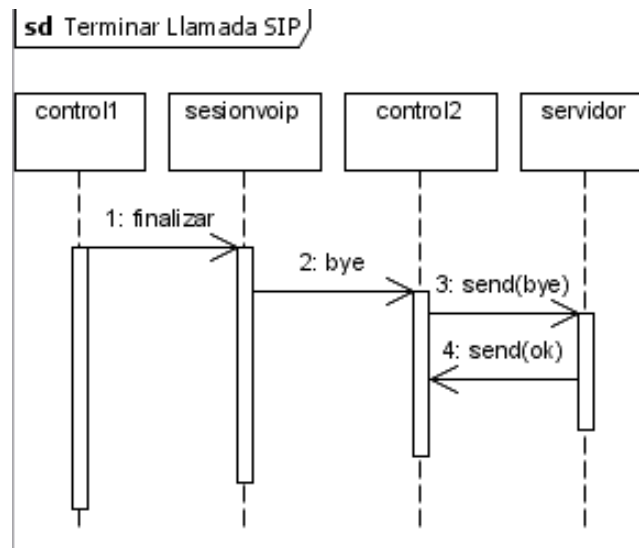


Figura (5.6) – Operación terminar llamada SIP

VII) Operación buscar cliente

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.7 y se detalla a continuación:

El servidor solicita una búsqueda y esta es procesada.

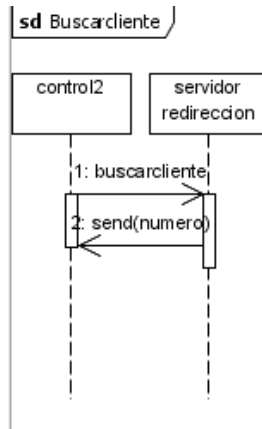


Figura (5.7) – Operación buscar cliente

VIII) Operación redirigir

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.8 y se detalla a continuación: Una vez realizada la búsqueda el servidor manda una señal de redirección con la nueva dirección del receptor.

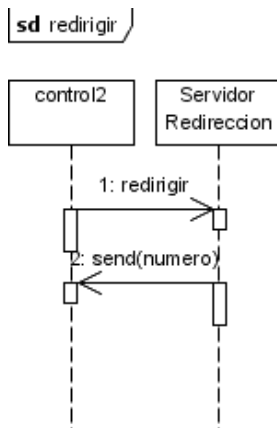


Figura (5.8) – Operación redirigir

IX) Operación registro

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.9 y se detalla a continuación:

Servidor inicia el conteo y la clase contador registra el tiempo que tome la llamada y la lleva a una base de datos.

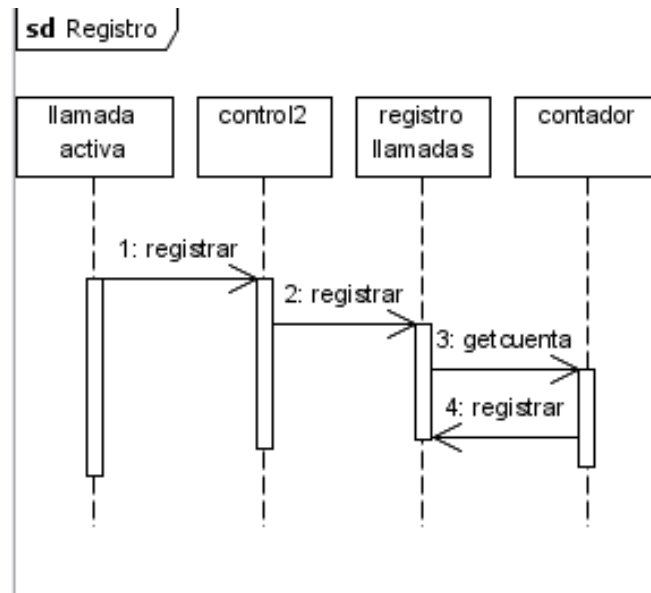


Figura (5.9) – Operación registro

X) Operación servidor proxy

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.10 y se detalla a continuación:

La clase sesión voip establece una conexión con el servidor proxy, a su vez el servidor actúa como intermediario entre el servidor del receptor y mi sistema de comunicación ejecutado por control 2.

XI) Operación directo

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.11 y se detalla a continuación:

La clase sesión voip indica a control 2 iniciar la conexión con el servidor del receptor para establecer la llamada.

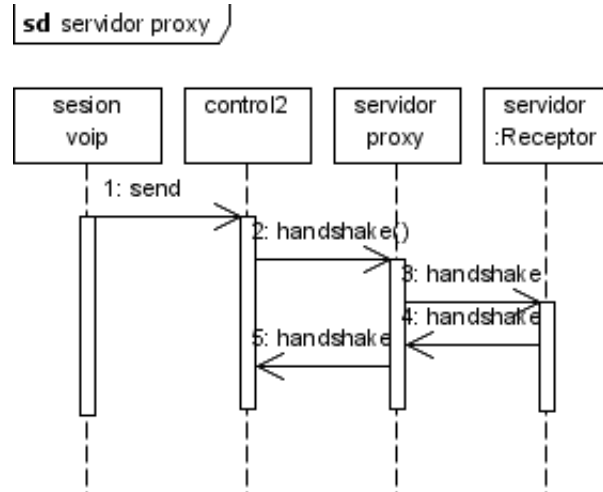


Figura (5.10) – Operación servidor proxy

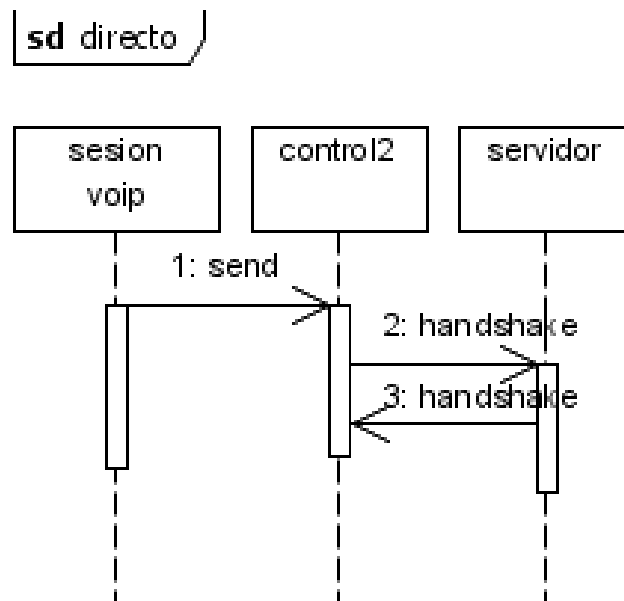


Figura (5.11) – Operación directo

XII) Operación redirección

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.12. La clase *sesion voip* indica a *control 2* establecer la llamada con el servidor. El servidor devuelve una señal indicando donde se encuentra la dirección de receptor a *control*

2, ahora se realiza una nueva conexión con el servidor del receptor.

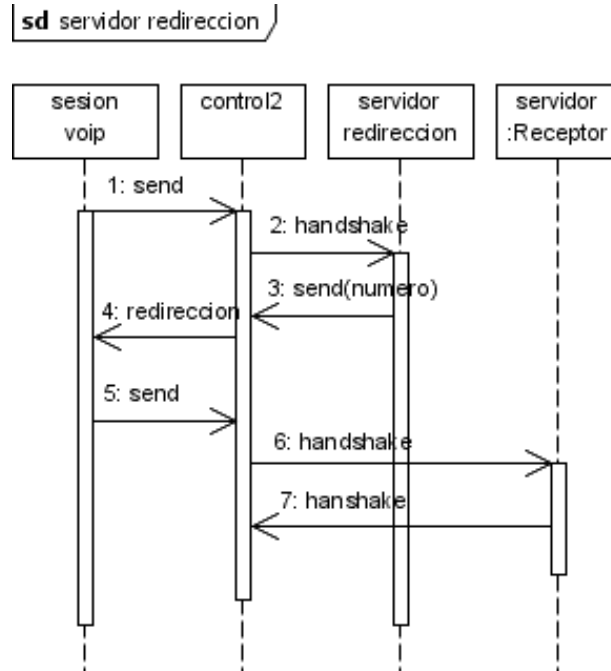


Figura (5.12) – Operación redirección

XIII) Operación revisar saldo

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.13.

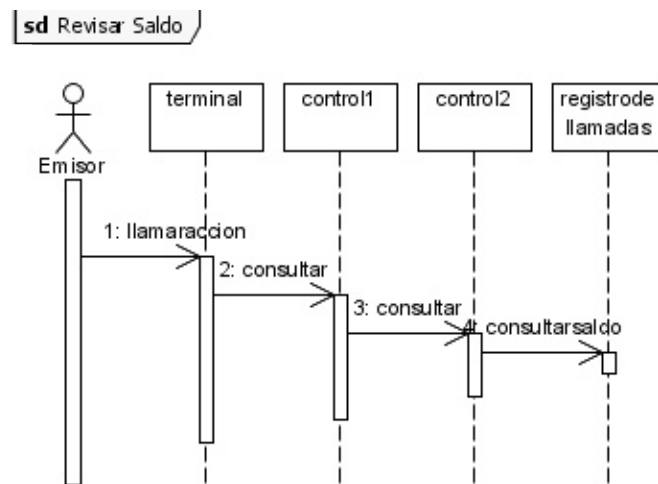


Figura (5.13) – Operación revisar saldo

5.2. Diagramas de colaboración

Un diagrama de objetos muestra a los objetos como tales y sus relaciones entre sí. Un diagrama de colaboraciones es una extensión de uno de objetos. Además de las relaciones entre objetos, el diagrama de colaboraciones muestra los mensajes que se envían los objetos entre sí.[?]

I) Operación llamar

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de colaboración se muestra en la figura 5.14 y se detalla a continuación: El emisor es responsable de inicializar la comunicación, en un inicio genera un número telefónico este es validado y posteriormente devuelto a la clase emisor.

Seguido inicializa una llamada con la clase llamada activa, que se encarga de generar una sesión SIP usando el método de conexión adecuado.

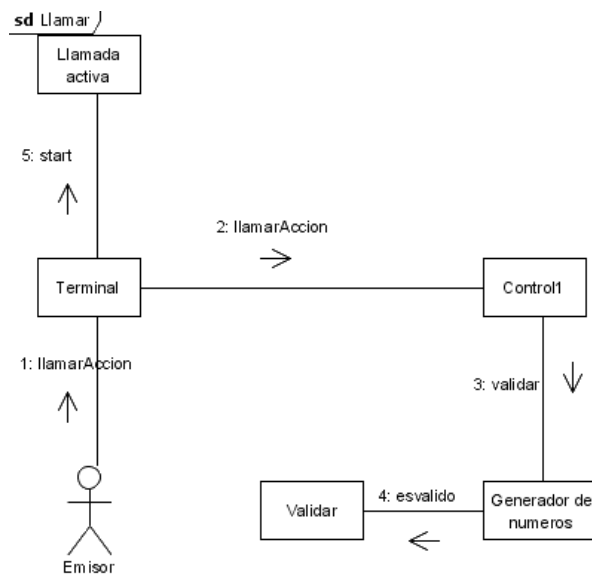


Figura (5.14) – Operación llamar

II) Operación hablar

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de colaboración se muestra en la figura 5.15 y se detalla a continuación: En este diagrama se muestra como se envía los medios.

Intervienen las clases de emisor y receptor y una tercera clase que dedica exclusivamente a la transmisión de los medios.

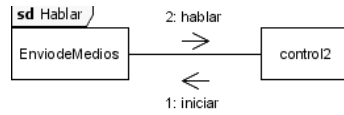


Figura (5.15) – Operación hablar

III) Operación terminar llamada

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de colaboración se muestra en la figura 5.16 y se detalla a continuación: El emisor indica que ya no quiere transmitir mas medios entonces envía una señal de despedida y se cierra la comunicación.

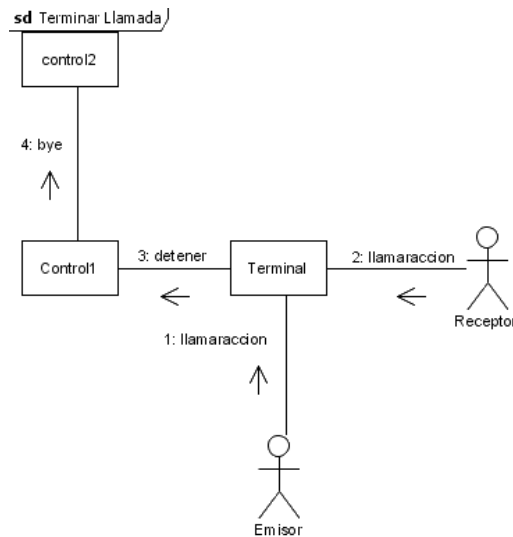


Figura (5.16) – Operación terminar llamada

IV) Operación generar número

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de colaboración se muestra en la figura 5.17 y se detalla a continuación: El emisor genera un número de acuerdo a la estructura presentada anteriormente

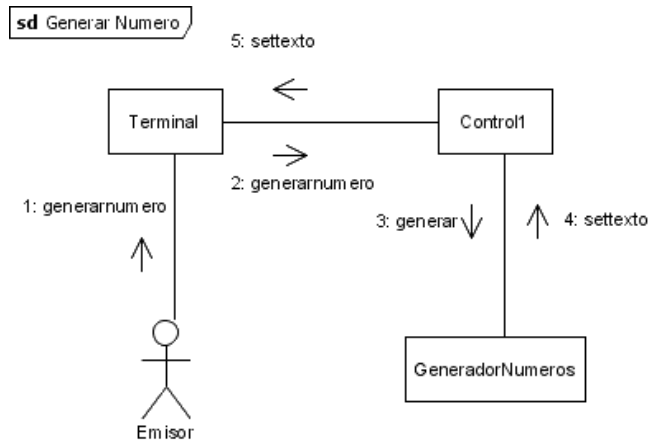


Figura (5.17) – Operación generar número

V) Operación establecer llamada SIP

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de colaboración se muestra en la figura 5.18 y se detalla a continuación:

Estable una conexión mediante el servidor Proxy, utilizando el modelo de conexión mediante servidor Proxy hasta crear la comunicación y pase a la transmisión de medios.

VI) Operación terminar llamada SIP

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de colaboración se muestra en la figura 5.19 y se detalla a continuación:

El emisor desea terminar la llamada y manda una señal de finalizar, el servidor la toma y se encarga de terminar la llamada.

VII) Operación buscar cliente

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de colaboración se muestra en la figura 5.20 y se detalla a continuación:

El servidor solicita una búsqueda y esta es procesada.

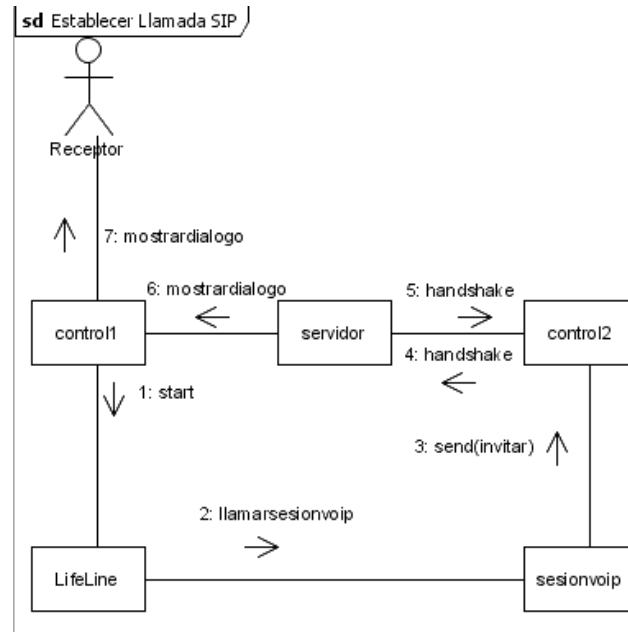


Figura (5.18) – Operación establecer llamada SIP

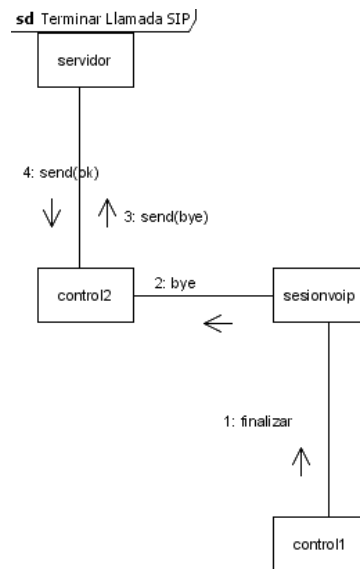


Figura (5.19) – Operación terminar llamada SIP

VIII) Operación redirigir

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de colaboración se muestra en la figura 5.21 y se detalla a continuación: Una vez realizada la búsqueda el servidor manda una señal

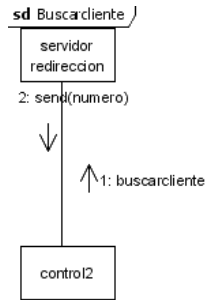


Figura (5.20) – Operación Buscar cliente

de redirección con la nueva dirección del receptor.

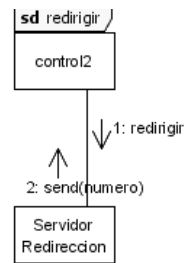


Figura (5.21) – Operación redirigir

IX) Operación registro

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de colaboración se muestra en la figura 5.22 y se detalla a continuación:

Servidor inicia el conteo y la clase contador registra el tiempo que tome la llamada.

X) Operación servidor proxy

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.23 y se detalla a continuación:

La clase sesión voip establece una conexión con el servidor proxy, a su vez el servidor actúa como intermediario entre el servidor del receptor y mi sistema de comunicación ejecutado por control 2.

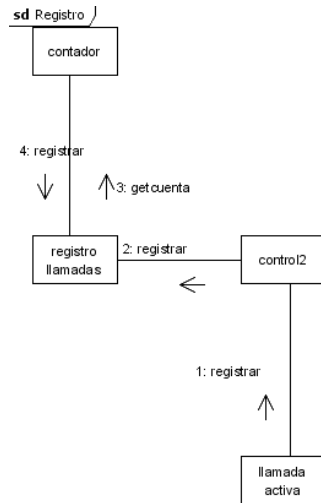


Figura (5.22) – Operación registro

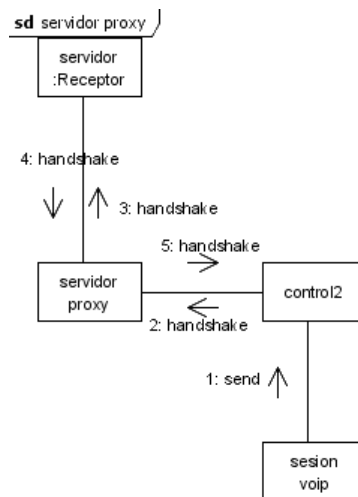


Figura (5.23) – Operación servidor proxy

XI) Operación directo

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.24 y se detalla a continuación:

La clase sesión voip indica a control 2 iniciar la conexión con el servidor del receptor para establecer la llamada.

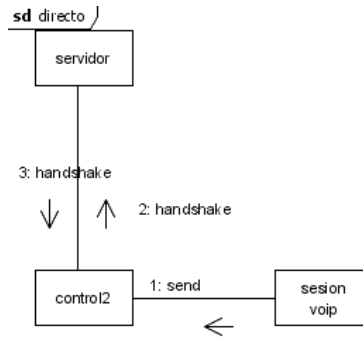


Figura (5.24) – Operación directa

XII) Operación redirección

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.25. La clase sesion voip indica a control 2 establecer la llamada con el servidor. El servidor devuelve una señal indicando donde se encuentra la dirección de receptor a control 2, ahora se realiza una nueva conexión con el servidor del receptor.

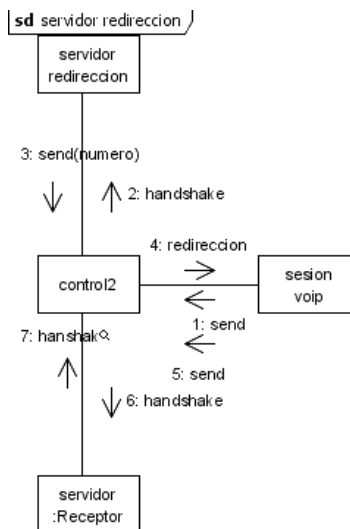


Figura (5.25) – Operación redirección

XIII) Operación revisar saldo

Esta operación se especifica en la figura 4.1 en la sección Casos de uso, el diagrama de secuencia se muestra en la figura 5.26.

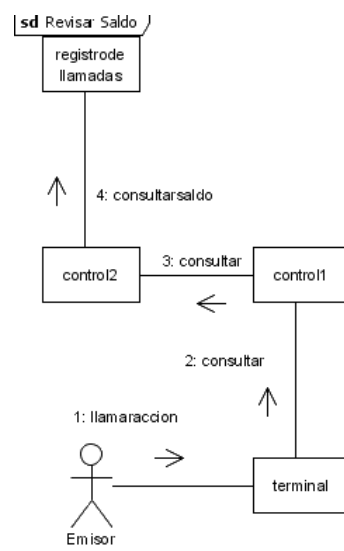


Figura (5.26) – Operación revisar saldo

5.3. Diagrama de Estados

Una manera para caracterizar un cambio en un sistema es que los objetos que lo componen modifican su estado como respuesta a los sucesos y al tiempo.[?]

Se muestran los diagramas de estados de los objetos que componen el sistema.

I) Emisor

En la figura 5.27 se muestra los estados que afectar este objeto.

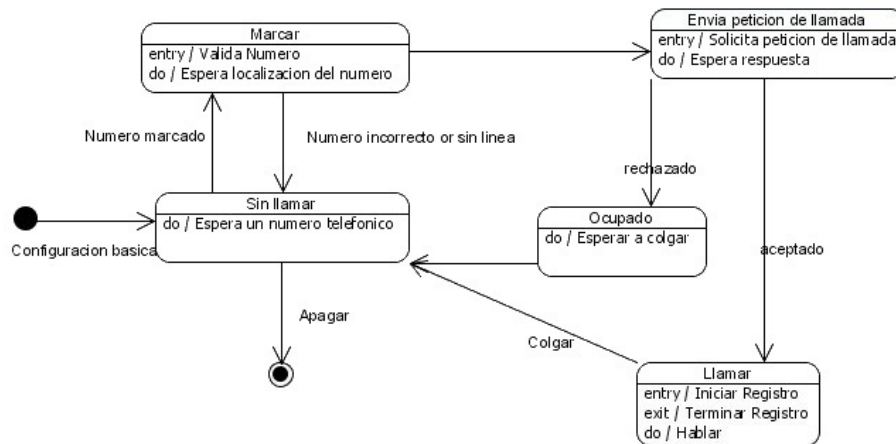


Figura (5.27) – Objeto emisor

II) Receptor

En la figura 5.28 se muestra los estados que afectar este objeto.

III) Generador de números telefónicos

En la figura 5.29 se muestra los estados que afectar este objeto.

IV) Validar número

En la figura 5.30 se muestra los estados que afectar este objeto.

V) Registro llamadas

En la figura 5.31 se muestra los estados que afectar este objeto.

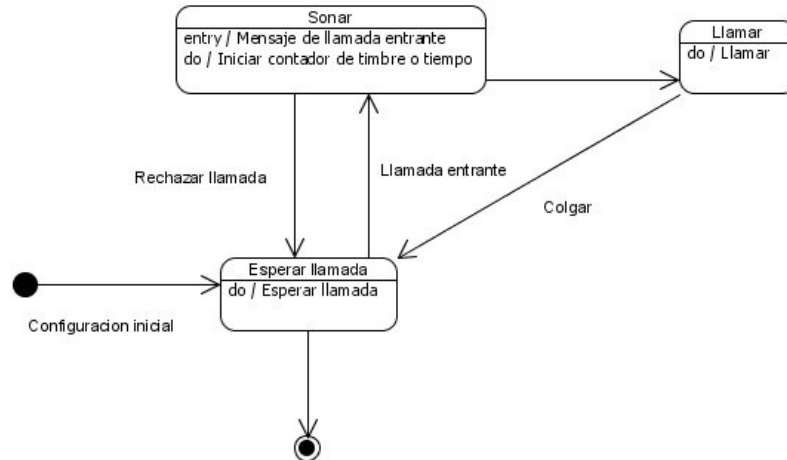


Figura (5.28) – Objeto receptor

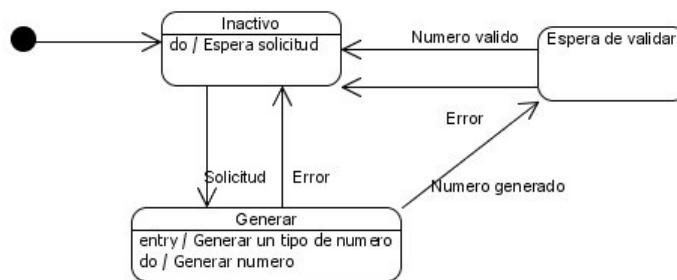


Figura (5.29) – Objeto generador de números telefónicos

VI) Contador

En la figura 5.32 se muestra los estados que afectar este objeto.

VII) Llamada activa

En la figura 5.33 se muestra los estados que afectar este objeto.

VIII) Sesión VoIP

En la figura 5.34 se muestra los estados que afectar este objeto.

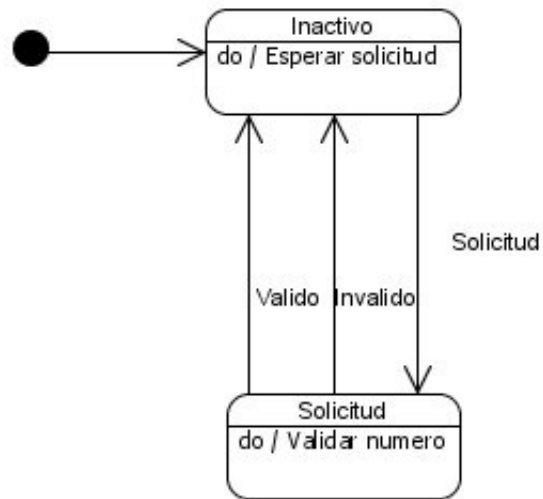


Figura (5.30) – Objeto validar número

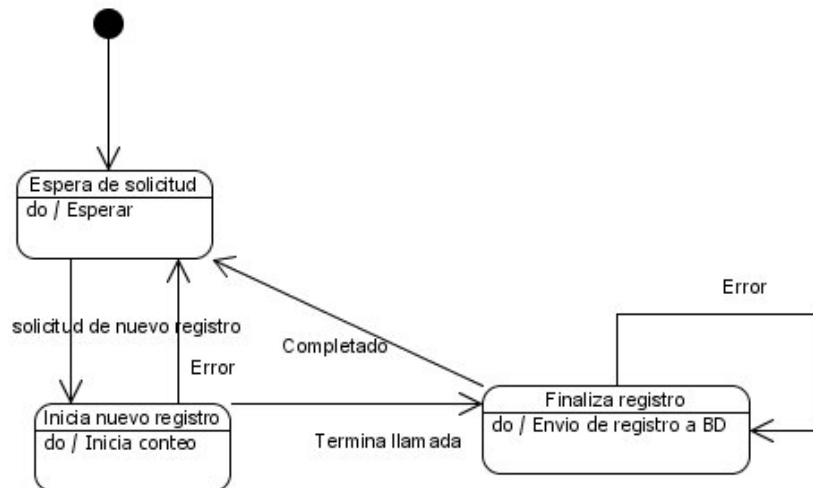


Figura (5.31) – Objeto registro de llamadas

IX) Servidor proxy

En la figura 5.35 se muestra los estados que afectan este objeto.

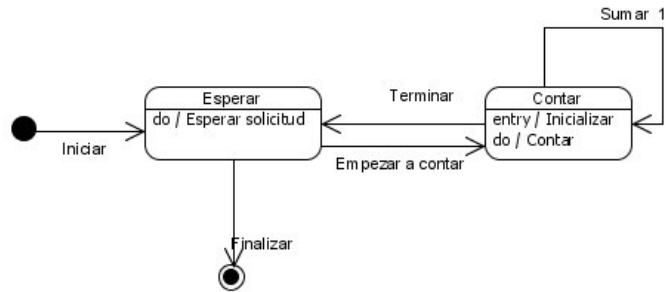


Figura (5.32) – Objeto Contador

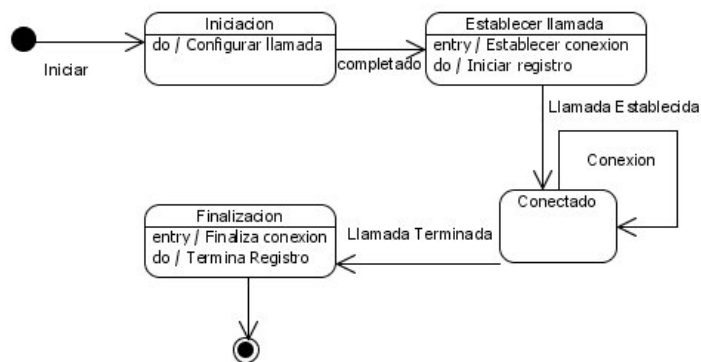


Figura (5.33) – Objeto llamada activa

X) Servidor de redirección

En la figura 5.36 se muestra los estados que afectan este objeto.

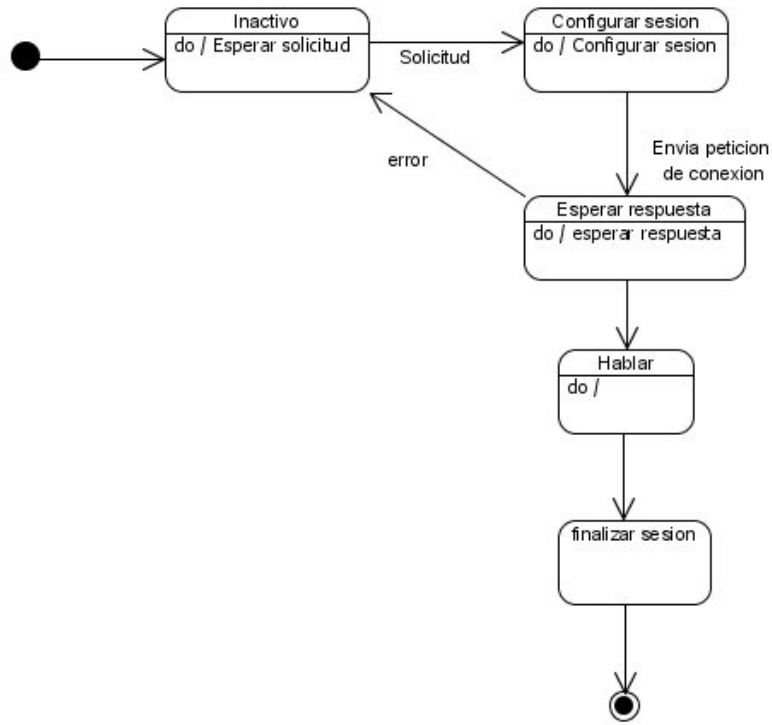


Figura (5.34) – Objeto sesion VoIP

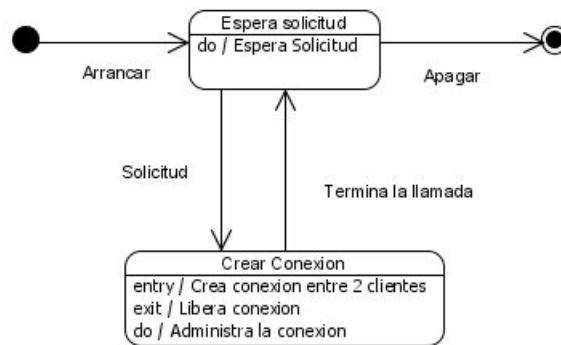


Figura (5.35) – Objeto Servidor Proxy

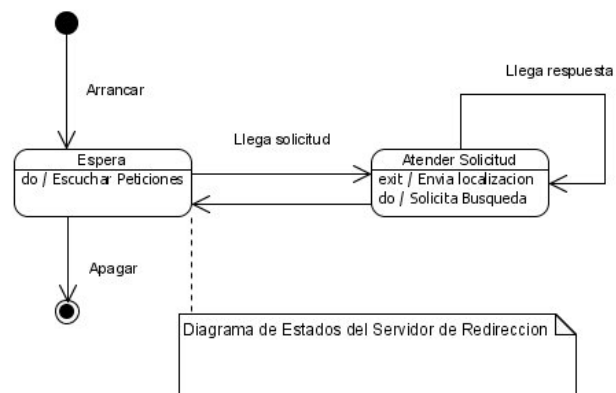


Figura (5.36) – Objeto servidor de redirección

Capítulo 6

Implementación

Introducción

El modelo de implementación es el resultado análisis y diseño, una vez definido los requisitos de sistema y la interacción de objetos se procede a llevar a cabo la implementación respectiva[11].

En esta sección se describirán los elementos utilizados para la implementación del sistema planteado.

6.1. Plataforma de desarrollo

Una plataforma de desarrollo es el entorno común en el cuál se desenvuelve la programación de un grupo definido de aplicaciones. Comúnmente se encuentra relacionada directamente a un sistema operativo, sin embargo, también es posible encontrarlas ligadas a una familia de lenguajes de programación o a una Interfaz de programación de aplicaciones o API.

Se ha elegido la plataforma de desarrollo Java de la compañía SUN, debido a las ventajas que representa.

Lenguaje de Programación Java

La plataforma de programación en Java es tanto lenguaje de programación y plataforma de desarrollo. A continuación se presentan las ventajas de la tecnología

Java.

Es un lenguaje de alto nivel que puede ser caracterizado por las siguientes capacidades:

- Simple
- Orientado a Objetos
- Distribuido
- Multihilo
- Arquitectura neutral
- Portable
- Alto rendimiento
- Robusto
- Dinámico
- Seguro

En el ambiente de programación Java, todo el código fuente es escrito en archivos de texto plano teniendo extensión `.java`. Estas fuentes son compiladas en un archivo `.class` por el compilador de Java.

Un archivo `.class` no contiene código nativo al procesador donde se compilo la aplicación, en vez contiene código binario relativo a la máquina virtual Java como se muestra en la figura 6.1.

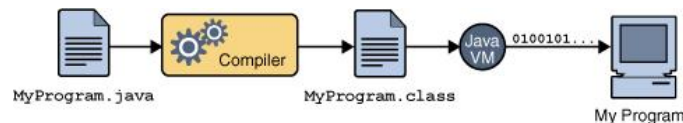


Figura (6.1) – *Proceso de compilación*

Debido a que la máquina virtual de Java esta disponible en diferentes sistemas operativos, el mismo archivo `.class` es capaz de correr sobre Microsoft Windows, Solaris OS, Linux, Mac OS.

Algunas máquinas virtuales desarrollan pasos adicionales para en tiempo de corrida dar empuje al desempeño. Esto incluye varias tareas tales como encontrar cuellos de botella y recompilado de código usado frecuentemente.

La misma aplicación es capaz de correr en múltiples plataformas como se muestra en la figura 6.2 debido a la máquina virtual Java.

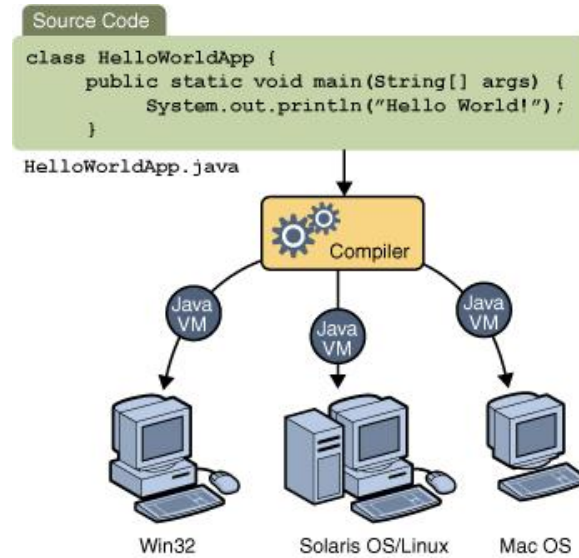


Figura (6.2) – Programa Java sobre múltiples plataformas

Plataforma de desarrollo Java

Una plataforma es el hardware o el ambiente de software en el cual un programa se ejecuta. La mayor parte de las plataformas pueden ser descritas como una combinación del sistema operativo y el hardware que lo soporta. La plataforma Java se diferencia de otras plataformas porque es únicamente software que se ejecuta sobre otras plataformas basadas en hardware. En la figura 6.3 muestra la plataforma Java.

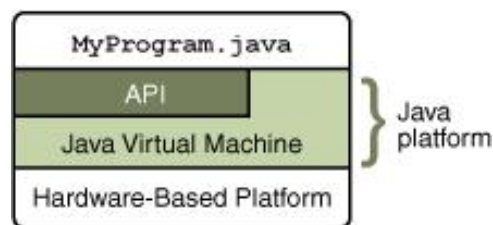


Figura (6.3) – Plataforma Java

6.2. Implementación en Java

Java es una de las plataformas más poderosas para el desarrollo de sistemas distribuidos, por su robustez y capacidad de manejo de hilos.

Debido a las características presentadas anteriormente se desarrolló el presente sistema en dicha plataforma de desarrollo.

A continuación se presenta los aspectos más importantes de la implementación realizada.

Características de la implementación:

Multihilo La implementación permite la generación de múltiples llamadas a diferentes receptores al mismo tiempo.

Orientado a Objetos Los elementos involucrados en las comunicaciones así como los datos son tratados todos como objetos y son transmitidos como tales entre los elementos de la implementación.

Cliente-Servidor La implementación maneja servidores para las comunicaciones y clientes que se comunican con estos para establecer las llamadas.

Clases

Basado en el diagrama de clases de la figura 4.2 se desarrollaron las clases necesarias, en primer lugar las clases de interfaz de usuario.

Clase terminal

Esta clase es la interfaz principal de usuario en la cuál se acceden a las opciones principales como generar números o llamar como se observa en la figura 6.4.

Esta clase se instancia en Emisor y Receptor debido a que ambos objetos tienen los mismos elementos.

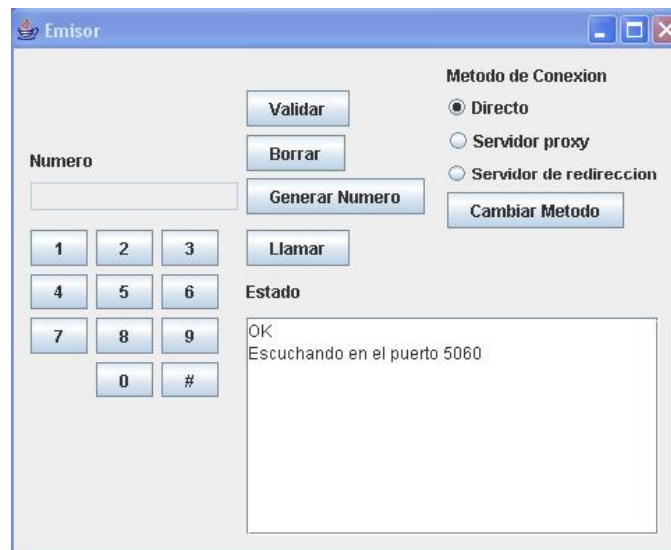


Figura (6.4) – Clase terminal, principal interfaz de usuario

Clase envío de medios

Esta clase muestra información referente a la llamada activa, estos datos incluyen datos del emisor y del receptor, no se incluye información referente al tipo de conexión ya que esto es transparente al usuario.

la figura 6.5 muestra la interfaz de usuario de esta clase.

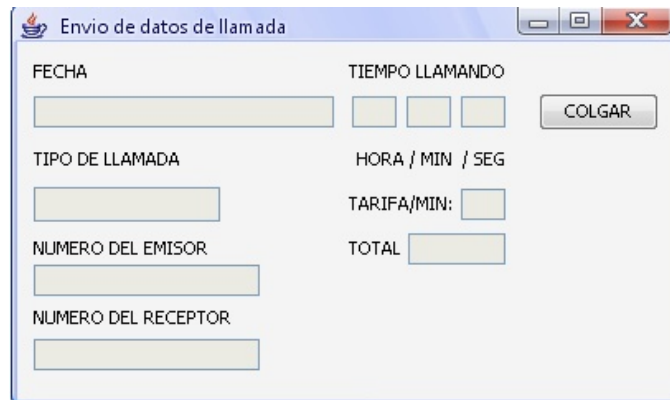


Figura (6.5) – Clase envío de medios

Clase servidor

Esta clase permite las comunicaciones dentro de la implementación de esta clase se heredan 2 tipos de servidores, el servidor de redirección y el servidor proxy los cuales permiten los modelos de comunicación correspondientes.

Además de ello cada terminal posee una implementación de la clases servidor para hacer posible el modelo de conexión directa.

Esta clase hace uso de paquetes descritos en la sección “Paquetes y bibliotecas” en la página 67.

6.2.1. Paquetes y bibliotecas

La implementación hace uso de paquetes para su correcto funcionamiento estos paquetes incluyen manejo de bases de datos mediante JDBC y manejo de servidores mediante las bibliotecas de Java.net mediante el protocolo orientado a conexión TCP de la familia de protocolos TCP/IP.

La implementación no define un sistema manejador de bases de datos, debido a que puede variar de acuerdo a las necesidades del usuario, por ello se utiliza el sistema de conexión abierto para bases de datos - ODBC(Open Data Base Connectivity), específicamente las bibliotecas JDBC dentro de Java.sql y manejo de lenguaje estructurado de consultas - SQL(Structured Query Language).

Parte IV

Pruebas y análisis de resultados

Capítulo 7

Pruebas y resultados

Introducción

En esta sección detallamos las pruebas necesarias para determinar la validez de la implementación.

Se realizaron diferentes tipos de pruebas buscando la aprobación tomando como base los conceptos de sistema de señalización SIP sección [3.3] [Metodología de señalización SIP](#)

7.1. Pruebas

Las pruebas que llevaremos son:

- I) Modelos de conexión que soporta el software.
 - a) Conexión directa
 - b) Conexión con servidor proxy
 - c) Conexión con servidor de redirección
- II) Capacidad de conexión múltiple realizando múltiples llamadas a diferentes números.
- III) Identificación de tipos de llamadas.

IV) Registro de llamadas en la base de datos.

V) Puertos utilizados para la implementación VoIP.

A continuación detallaremos cada una de las pruebas a realizar.

7.2. Modelo conexión directa

Basado en el modelo de conexión descrito en la sección 3.4.1 *Modelo directo*(pág. 22) muestra los pasos para crear la conexión entre el emisor y receptor.

Se evaluará lo siguiente:

- *Mensajes enviados y recibidos del emisor al receptor.*
- *Mensajes enviados y recibidos del receptor al emisor.*
- *Orden de los mensajes*

7.3. Modelo conexión con servidor proxy

Basado en el modelo de conexión descrito en la sección 3.4.2 *Modelo con servidor proxy*(pág. 23) muestra los pasos para crear la conexión entre el emisor y receptor usando un servidor de tipo proxy.

Se evaluará lo siguiente:

- *Mensajes enviados del emisor al receptor.*
- *Mensajes enviados del receptor al emisor.*
- *Mensajes enviados y recibidos del servidor proxy*

7.4. Modelo conexión con servidor de redirección

Basado en el modelo de conexión descrito en la sección 3.4.3 *Modelo con servidor de redirección*(pág. 24) muestra los pasos para crear la conexión entre el emisor y receptor usando un servidor de tipo proxy.

Se evaluará lo siguiente:

- *Mensajes enviados del emisor al receptor.*
- *Mensajes enviados del receptor al emisor.*
- *Mensajes enviados y recibidos del servidor de redirección*

7.5. Conexión múltiple

En esta prueba determinaremos si el software es capaz de soportar conexiones entre un emisor y dos o mas receptores diferentes.

7.6. Identificación de tipos de llamadas

En esta prueba se obtendrá la capacidad del software para identificar el tipo de marcación, el emisor debe identificar el tipo de número al cual va a marcar y su respectiva tarifa.

Para las pruebas se marcaran los siguientes números

- 2865676
- 044 5528693
- 01 226 2548655
- *60
- 047748659

7.7. Registro de llamadas a la base de datos

En esta prueba determinaremos la capacidad del software para registrar la información referente a la llamada realizada, incluyendo tiempo de la llamada, y tarifa aplicada a la misma.

7.8. Identificación de puertos

En esta prueba se verificara los puertos y conexiones necesarios para establecer los diferentes tipos de llamadas y sus registros.

7.9. Resultados modelo conexión directa

El software utiliza los tres modelos, en esta sección analizamos los mensajes enviados entre el emisor y el receptor utilizando el modelo de conexión directo.

El software funciona en los puertos 5060 para el emisor y el 5061 para él receptor como se muestra en la figura 7.1 para el emisor, la figura 7.2 para el receptor y en la figura 7.41 muestra los puertos utilizados.

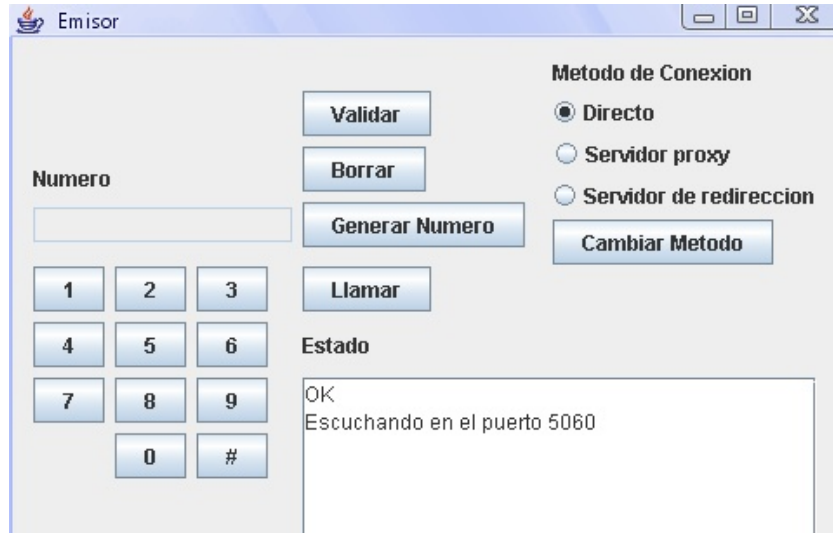


Figura (7.1) – Interfaz emisor

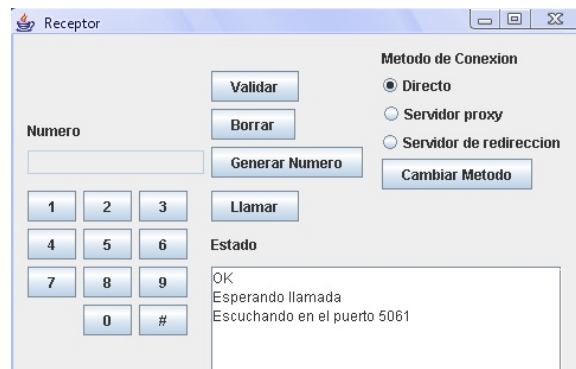


Figura (7.2) – Interfaz receptor

```
TCP ragnarok:5060 0.0.0.0 LISTENING
TCP ragnarok:5061 0.0.0.0 LISTENING
TCP ragnarok:5062 0.0.0.0 LISTENING
TCP ragnarok:5063 0.0.0.0 LISTENING
```

Figura (7.3) – Puertos utilizados al iniciar el sistema

7.9.1. Manejo de señales

Basados en el modelo de conexión directa, el emisor y receptor deben intercambiar las señales necesarias para establecer la comunicación entre ambos.

Las señales intercambiadas por parte de emisor deber ser las indicadas en la figura 7.4 y contrastamos con la figura 7.5.

En esta figura 7.5 se observa que la aplicación verifica el número marcado, determinando su validez así como la categoría al cual corresponde el número

marcado.

Se comprobó el número 2865676, indicando que el número es válido y pertenece a la categoría de número local, asignando la tarifa correspondiente.

Una vez validado el número marcado, la aplicación envía el mensaje invitar (INVITE) como se muestra en la figura 7.6 después el emisor recibe el mensaje sonando (RINGING) representado por la señal 180, indicando que el receptor ha recibido la invitación y está esperando a que este le responda con una aceptación o negación de la llamada. Para enviar las señales el emisor crea una conexión TCP al receptor como se muestra en la figura 7.8, el receptor está escuchando en el puerto 5061, y el emisor utiliza un puerto aleatorio para conectarse al receptor.

Una vez confirmada la llamada el emisor envía un reconocimiento (ACK) y empieza a transmitir mediante mensajes de tipo RTP encapsulados en TCP. El receptor a su vez al recibir el reconocimiento se prepara a recibir y enviar mensajes RTP [7.11.1], y la llamada está activa en este momento y empieza el registro de esta tanto por parte del emisor como del receptor como se muestra en la figura 7.7.

Tanto el emisor como el receptor pueden terminar la llamada en cualquier momento de la llamada. El emisor envía la señal de adiós (BYE), el receptor recibe este mensaje y envía un mensaje 200 OK y la llamada se finaliza como se muestra en las figuras 7.9 y 7.10.

Acto seguido el tiempo de la llamada es registrada en la base de datos como se muestra en la figura 7.40 (pag. page 93).

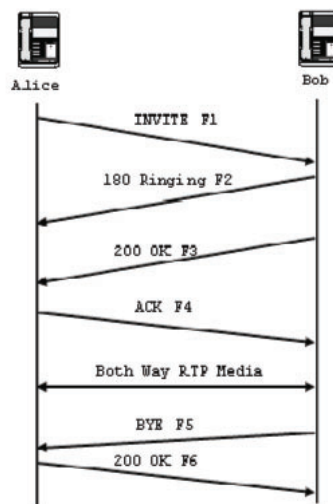


Figura (7.4) – Modelo de llamada básica SIP

Con lo anterior se observa que el manejo de señales está de acuerdo al referido por el estándar VoIP para modelo de conexión directa [1].

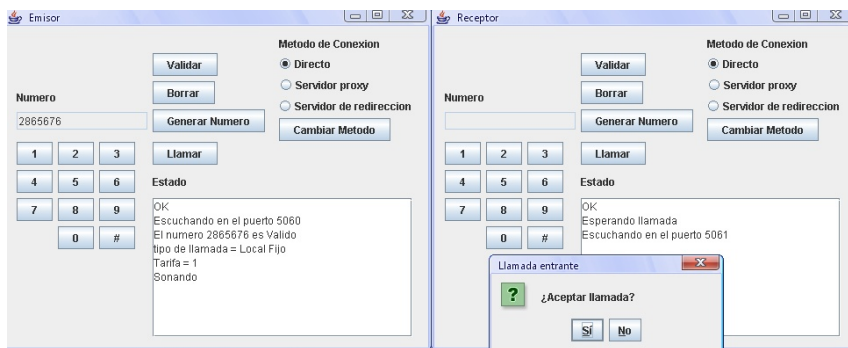


Figura (7.5) – Mensajes invitar y sonando entre el emisor y el receptor

```

Contenido tabla tipo de llamadas
TipoLlamada | Costo
Local Fijo  | 1
mensaje 1 del cliente: INVITE

Texto recibido del servidor: 180.

```

Figura (7.6) – Mensajes enviados a la consola de salida

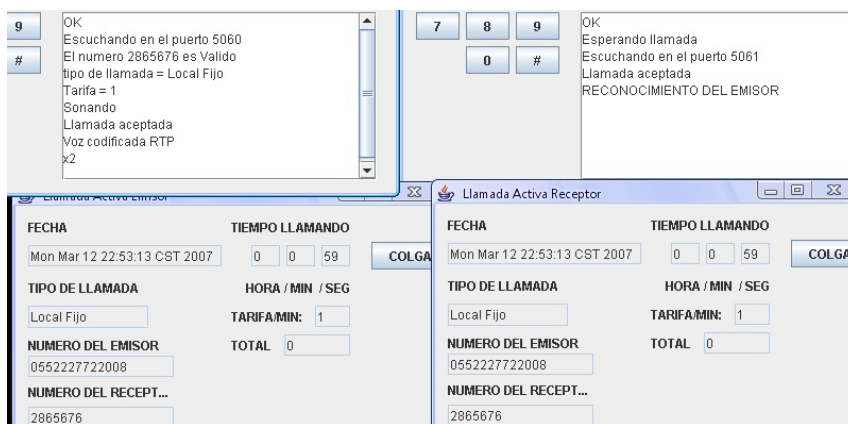


Figura (7.7) – Llamada aceptada entre el emisor y el receptor

```

TCP *agarak:1037          localhost:5061      ESTABLISHED
TCP *agarak:5061         localhost:1037     ESTABLISHED

```

Figura (7.8) – Conexión TCP entre el emisor y el receptor

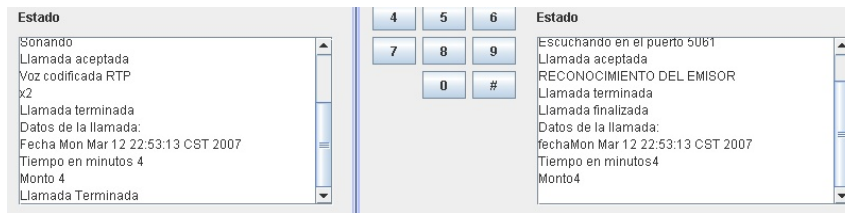


Figura (7.9) – Llamada finalizada entre el emisor y el receptor

```

Receptor: Recibe RTPMEDIARTPMEDIA
Receptor: Recibe RTPMEDIARTPMEDIA
Receptor: Recibe RTPMEDIARTPMEDIA
Receptor: Recibe RTPMEDIARTPMEDIA
Receptor: Recibe RTPMEDIARTPMEDIA
Receptor: Recibe RTPMEDIARTPMEDIA
emisor enviaBYE
Receptor: Recibe RTPMEDIARTPMEDIA
emisor enviaBYE
Receptor: Recibe BYE200 OK
Minutos 4
Minutos 4
Lineas insertadas 1

```

Figura (7.10) – Mensajes enviados a la consola

IpEmisor	IpReceptor	NumeroEmisor	NumeroReceptor	TipoLlamada	Minutos	Monto	Fecha
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	4	4	Mon Mar 12 22:53:13 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	1	1	Mon Mar 12 23:21:39 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	2	2	Mon Mar 12 23:24:41 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	1	1	Tue Mar 13 00:30:09 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	4	4	Mon Mar 12 23:48:26 CST 2007

Figura (7.11) – Registro de llamadas en base de datos

7.10. Resultados modelo conexión con servidor proxy

En esta sección analizamos los mensajes enviados entre el emisor, el receptor y el servidor proxy utilizando el modelo de conexión con servidor proxy.

La aplicación funciona en los puertos 5060 para el emisor y el 5061 para el receptor y 5063 para el servidor proxy como se muestra en la figuras 7.12 para el emisor y receptor, 7.13 para el servidor proxy y en la figura 7.41 (pag. 93) muestra los puertos utilizados.

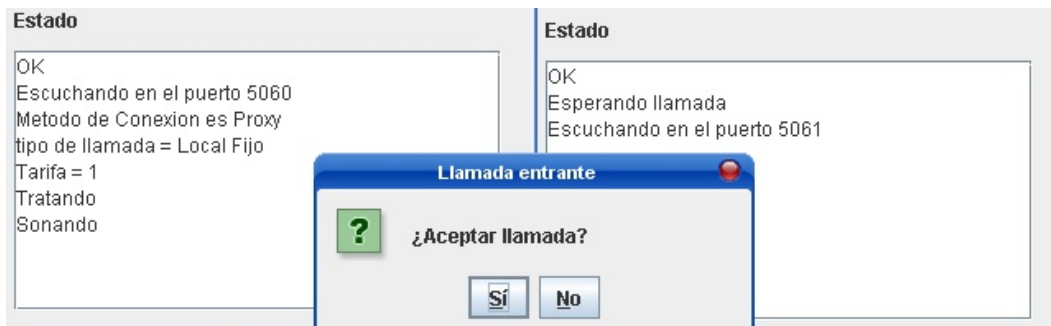


Figura (7.12) – receptor y emisor

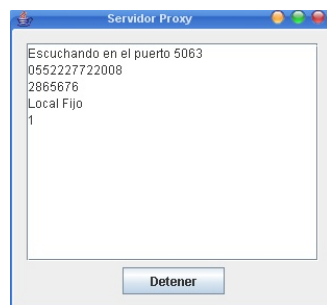


Figura (7.13) – servidor proxy

TCP	ragnarok:5060	0.0.0.0	LISTENING
TCP	ragnarok:5061	0.0.0.0	LISTENING
TCP	ragnarok:5062	0.0.0.0	LISTENING
TCP	ragnarok:5063	0.0.0.0	LISTENING

Figura (7.14) – Puertos

7.10.1. Manejo de señales

Basados en el modelo de conexión con servidor proxy, el emisor y receptor deben intercambiar las señales necesarias para establecer la comunicación utilizando al servidor proxy, ya que este actuará como intermediario para poder establecer la comunicación entre ambas partes.

De este modo el emisor se conectara al servidor proxy, actuando como el receptor, sin embargo el servidor proxy se conectara al receptor actuando como el emisor.

Las señales intercambiadas están indicadas en la figura 7.15 y contrastamos con la figura 7.16. La aplicación verifica el número marcado, determinando su

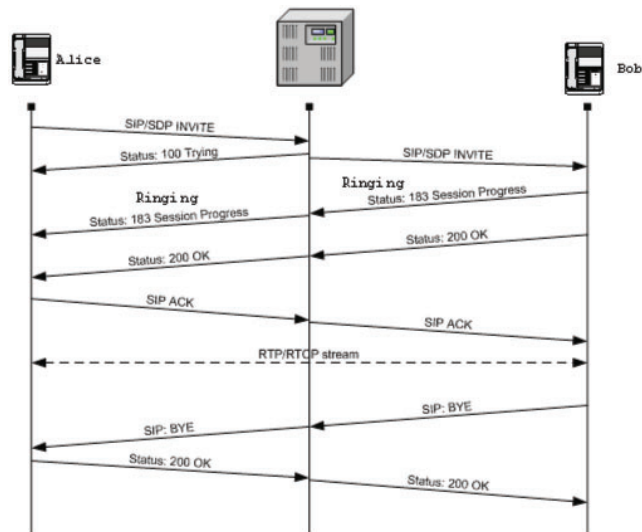


Figura (7.15) – Modelo de llamada básica SIP con servidor proxy

```

Contenido tabla tipo de llamadas
TipoLlamada      | Costo
Local Fijo       | 11
Texto recibido del servidor:100 TRYING.

Emisor:Texto recibido del servidor:.

dormir 1 seg
mensaje 1 del cliente: INVITE

Proxy: Texto recibido del Receptor:180.

Proxy: Texto enviado a Emisor:180

Proxy: Texto recibido de Receptor

Emisor:Texto recibido del servidor:180.

```

Figura (7.16) – *Mensajes entre servidor proxy, emisor y receptor*

validez así como la categoría al cual corresponde el número marcado.

Se comprobó el número 2865676, indicando que el número es valido y pertenece a la categoría de número local, asignando la tarifa correspondiente.

Una vez validado el número marcado, la aplicación envía el mensaje invitación(INVITE) como se muestra en la figura 7.16 después el servidor proxy envía un mensaje de intentando(TRYING) al emisor,a su vez envía un mensaje de invitación(INVITE)al receptor.

El receptor envía el mensaje sonando(RINGING) al servidor proxy el cuál lo reenvía al emisor, indicando que el receptor ha recibido la invitación y responde con una aceptación o negación de la llamada como se muestra en las figuras 7.12, 7.17 y estableciendo una serie de conexiones TCP como se muestra en la figura 7.18. Donde el emisor usando el puerto un puerto aleatorio en este caso el puerto 1059 se conecta al servidor proxy puerto 5063, este a su vez utiliza un puerto aleatorio en este caso el 1060 para conectarse con el receptor puerto 5061.

Una vez confirmada la llamada el emisor envía un reconocimiento(ACK) y empieza a transmitir mediante mensajes de tipo RTP encapsulados en TCP que pasan a través del servidor proxy como se muestra en la figura 7.19. El receptor a su vez al recibir el reconocimiento se prepara a recibir y enviar mensajes RTP, y la llamada esta activa en este momento y empieza el registro de esta tanto por parte del emisor como del receptor.

Tanto el emisor como el receptor pueden terminar la llamada en cualquier momento de la llamada. El emisor envía la señal de adiós(BYE), el receptor recibe este mensaje y envía un mensaje 200 OK y la llamada se finaliza, al igual que con las señales anteriores el servidor proxy sirve de intermediario en el envío y recepción de mensajes como se muestra en las figuras 7.20, 7.21 y 7.19.

Acto seguido el tiempo de la llamada es registrada en la base de datos como se muestra en la figura 7.40.

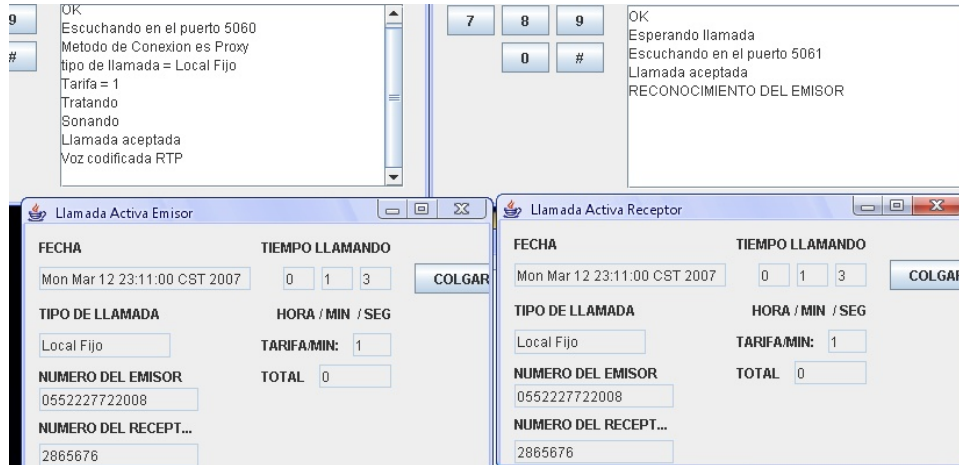


Figura (7.17) – Llamada establecida entre emisor y receptor

```
TCP ragnarok:1059 localhost:5063 ESTABLISHED
TCP ragnarok:1060 localhost:5061 ESTABLISHED
TCP ragnarok:5061 localhost:1060 ESTABLISHED
TCP ragnarok:5063 localhost:1059 ESTABLISHED
```

Figura (7.18) – Puertos utilizados por la aplicación en la llamada

```
Proxy: enviado a receptor RTPMEDIA
Receptor: Recibe RTPMEDIAReceptor: envia BYE
Proxy: Recivido receptor RTPMEDIA

Emisor: lee RTPMEDIA
Proxy: enviado a emisor RTPMEDIA

Proxy: Recivido emisor RTPMEDIA

Proxy: enviado a receptor RTPMEDIA

Proxy: Recivido receptor BYE

Emisor: lee BYE
Minutos 2
Proxy: enviado a emisor BYE

Proxy: Recivido emisor BYE

Receptor: Recibe RTPMEDIALineas insertadas 1
```

Figura (7.19) – Registro de los mensajes enviados durante la llamada

Con lo anterior se observa que el manejo de señales esta de acuerdo al referido por el estándar VoIP para modelo de conexión con servidor proxy[1].

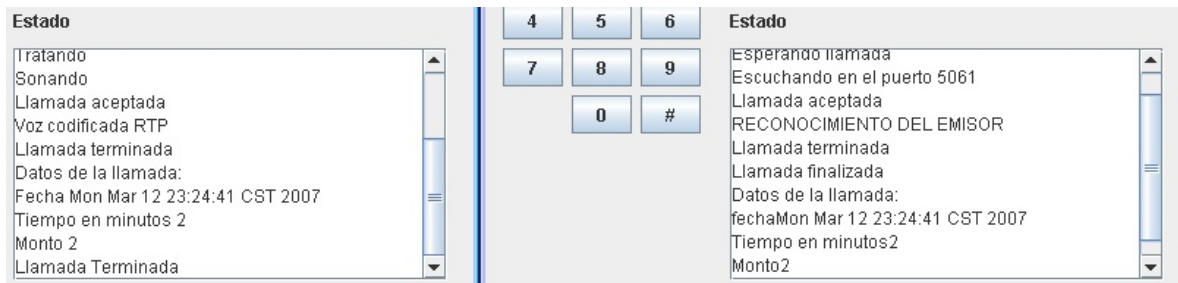


Figura (7.20) – *Llamada terminada emisor (izquierda) y receptor(derecha)*

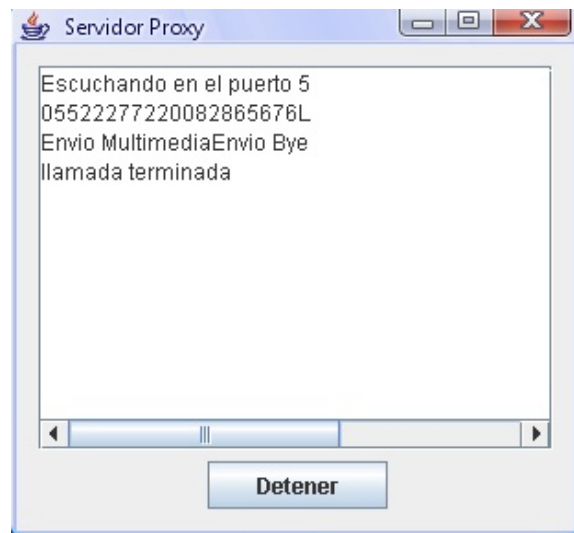


Figura (7.21) – *Resultado del servidor proxy*

IpEmisor	IpReceptor	NumeroEmisor	NumeroReceptor	TipoLlamada	Minutos	Monto	Fecha
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	4	4	Mon Mar 12 22:53:13 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	1	1	Mon Mar 12 23:21:39 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	2	2	Mon Mar 12 23:24:41 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	1	1	Tue Mar 13 00:30:09 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	4	4	Mon Mar 12 23:48:26 CST 2007

Figura (7.22) – *Registro en la base de datos*

7.11. Resultados modelo de conexión servidor de redirección

En esta sección analizamos los mensajes enviados entre el emisor, el receptor y el servidor de redirección utilizando el modelo de conexión con servidor de redirección.

La aplicación funciona en los puertos 5060 para el emisor y el 5061 para el receptor y 5062 para el servidor de redirección como se muestra en la figuras 7.23 para el emisor y receptor, 7.24 para el servidor de redirección y en la figura 7.41 muestra los puertos utilizados.

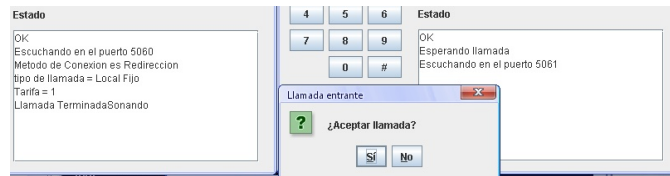


Figura (7.23) – receptor y emisor

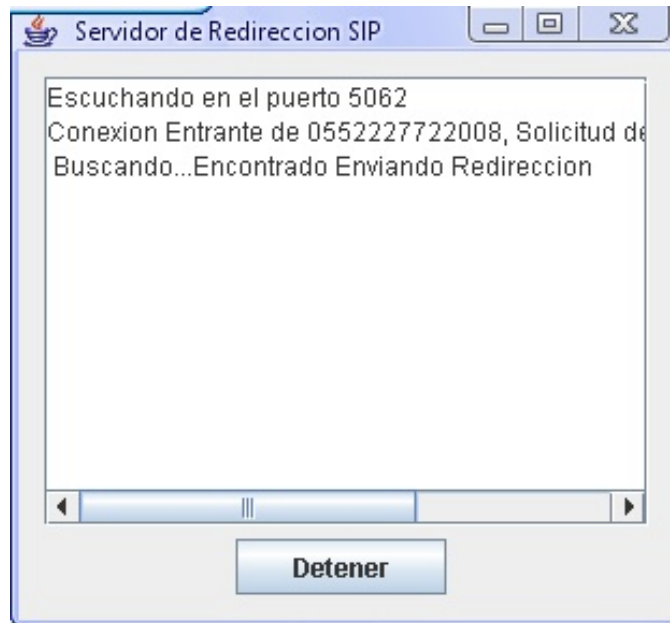


Figura (7.24) – servidor de redirección

7.11.1. Manejo de señales

Basados en el modelo de conexión con servidor de redirección, el emisor no conoce la dirección del receptor, de este modo el emisor crea una conexión al

```

TCP ragnarok:5060 0.0.0.0 LISTENING
TCP ragnarok:5061 0.0.0.0 LISTENING
TCP ragnarok:5062 0.0.0.0 LISTENING
TCP ragnarok:5063 0.0.0.0 LISTENING

```

Figura (7.25) – Puertos

servidor de redirección preguntado la dirección del receptor.

Una vez conociendo la dirección del receptor el emisor crea una conexión usando el método de conexión directa.

Las señales intercambiadas están indicadas en la figura 7.26 y contrastamos con la figura 7.27.

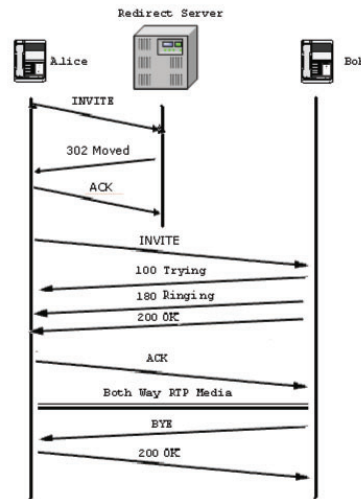


Figura (7.26) – Modelo de llamada básica SIP con servidor de redirección

```

Contenido tabla tipo de llamadas
TipoLlamada | Costo
Local Fijo | 11
Texto recibido del servidor:302 MOVED.

mensaje 1 del cliente: INVITE

Texto recibido del servidor:180.
|
Emisor: Mensaje
dormir 1 seg

```

Figura (7.27) – Mensajes entre servidor proxy, emisor y receptor

La aplicación verifica el número marcado, determinando su validez así como la categoría al cual corresponde el número marcado.

Se comprobó el número 2865676, indicando que el número es válido y pertenece a la categoría de número local, asignando la tarifa correspondiente.

Una vez validado el número marcado, la aplicación desconoce la dirección donde se encuentra el receptor, para conocerlo envía el mensaje invitar (INVITE) como se muestra en la figura 7.27 al servidor de redirección, el servidor busca la dirección del receptor y manda un mensaje de movido(MOVED) al emisor. El emisor termina la conexión con el servidor enviando un reconocimiento(ACK).

A partir de este momento el emisor ya conoce la dirección del receptor de este modo crea una nueva conexión usando el método de conexión directa descrito en la sección 3.4.1 *Modelo directo*(pág. 22).

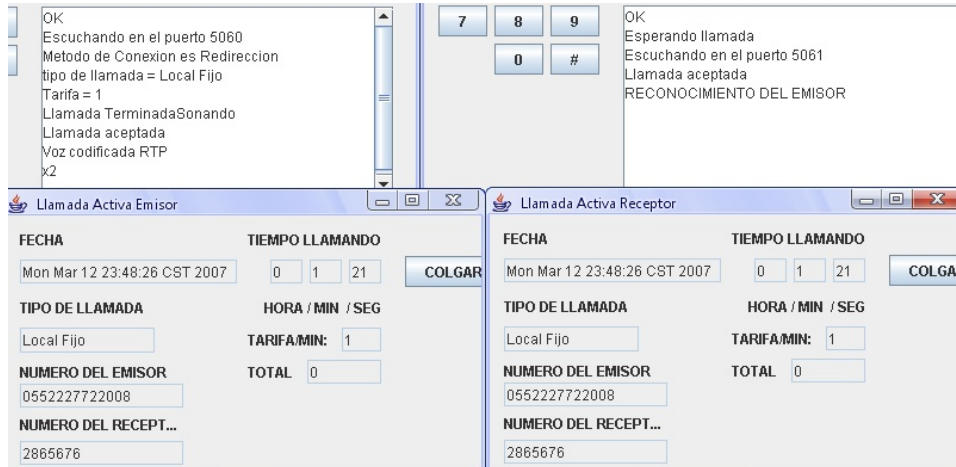


Figura (7.28) – Llamada establecida entre el emisor y receptor

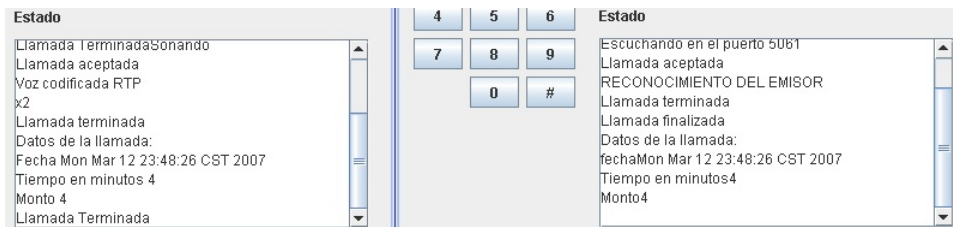


Figura (7.29) – Fin de llamada entre emisor y receptor

La aplicación envía el mensaje invitar (INVITE), después el emisor recibe el mensaje sonando(RINGING) representado por la señal 180, indicando que el receptor ha recibido la invitación y aun no ha respondido con una aceptación o negación de la llamada 7.23.

Una vez confirmada la llamada el emisor envía un reconocimiento(ACK) y empieza a transmitir mediante mensajes de tipo RTP encapsulados en TCP. El receptor a su vez al recibir el reconocimiento se prepara a recibir y enviar mensajes RTP, y la llamada esta activa en este momento y empieza el registro de esta tanto por parte del emisor como del receptor 7.28.

Tanto el emisor como el receptor pueden terminar la llamada en cualquier momento de la llamada. El emisor envía la señal de adiós(BYE), el receptor recibe este mensaje y envía un mensaje 200 OK y la llamada se finaliza 7.29.

Acto seguido el tiempo de la llamada es registrada en la base de datos como se muestra en la figura 7.40.

IpEmisor	IpReceptor	NumeroEmisor	NumeroReceptor	TipoLlamada	Minutos	Monto	Fecha
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	4	4	Mon Mar 12 22:53:13 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	1	1	Mon Mar 12 23:21:39 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	2	2	Mon Mar 12 23:24:41 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	1	1	Tue Mar 13 00:30:09 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	4	4	Mon Mar 12 23:48:26 CST 2007

Figura (7.30) – Registro en la base de datos

Con lo anterior se observa que el manejo de señales esta de acuerdo al referido por el estándar VoIP para modelo de conexión con servidor de redirección[1].

7.12. Resultado manejo de conexiones múltiples

El software es capaz de crear conexiones múltiples de distintos emisores a un receptor o un emisor a distintos receptores.

La figura 7.31 muestra como un receptor soporta varias llamadas de un mismo emisor de este modo también es capaz de múltiples emisores.

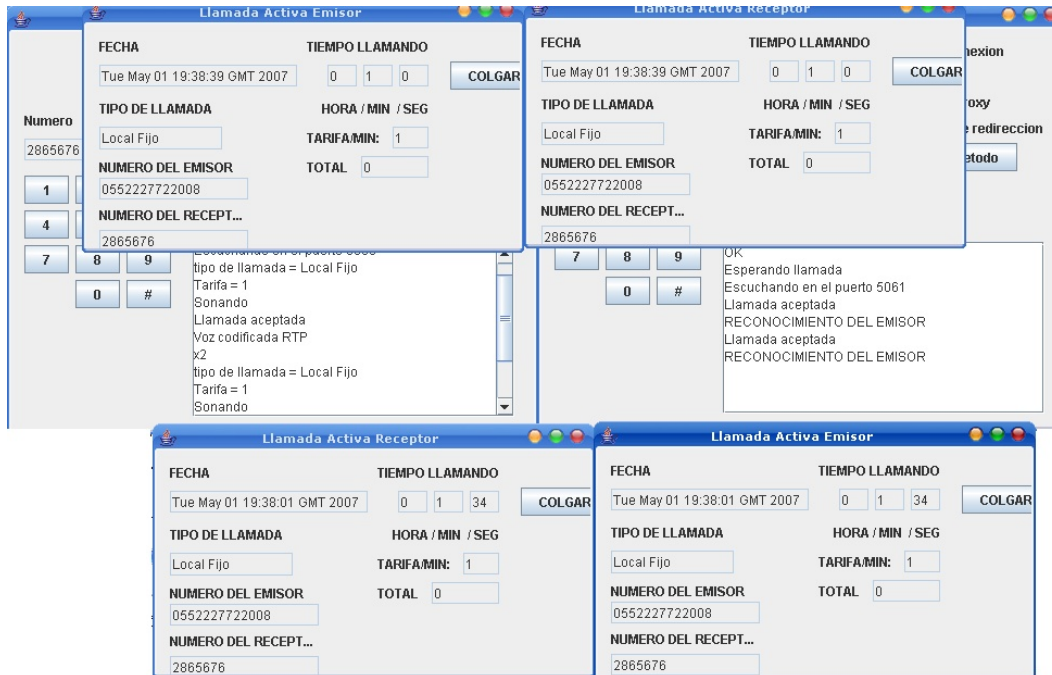


Figura (7.31) – emisor creando dos conexión al receptor

La figura 7.32 muestra como el resultado de ambas llamadas soporta varias llamadas de un mismo emisor de este modo también es capaz de múltiples emisores.

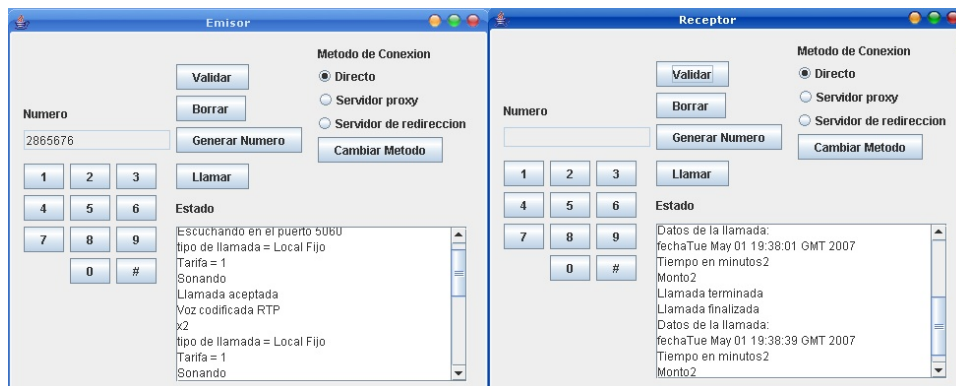


Figura (7.32) – Resultados de conexiones múltiples

La figura 7.33 muestra como un receptor soporta varias llamadas de un mismo emisor de este modo también es capaz de múltiples emisores.

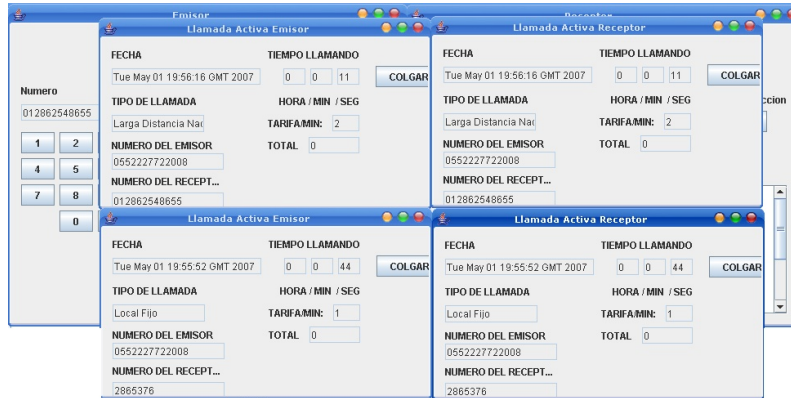


Figura (7.33) – emisor creando conexión a distintos receptores

De este modo se demuestra la capacidad del software de crear conexiones múltiples.

7.13. Resultado prueba identificación de tipos de llamada

El software es capaz de reconocer los tipos de marcado mostrados en la figura 7.39.

Para las pruebas se marcaran los siguientes números

- 2865676
- 044 5528693
- 01 226 2548655
- *60
- 047748659

Mostrando los siguientes resultados:

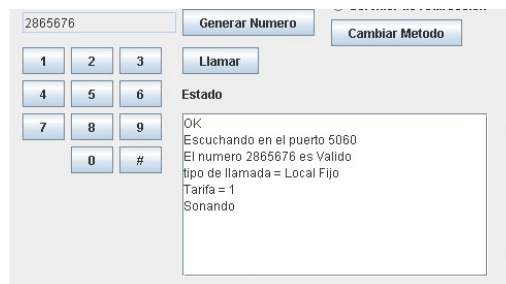


Figura (7.34) – Prueba de marcado para el número 2865676

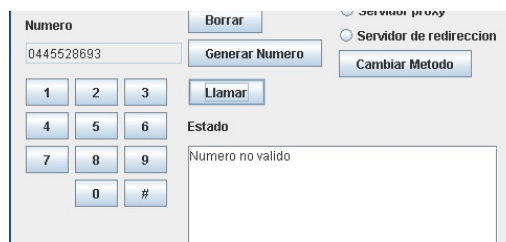


Figura (7.35) – Prueba de marcado para el número 0445528693

Numero: 012262548655

Validar | Borrar | Generar Numero | Llamar

Metodo de Conexion: Servidor de redireccion | Servidor proxy | Servidor de redireccion | Cambiar Metodo

Estado: Numero no validotipo de llamada = Larga Distancia Nacional
Tarifa = 2
Sonando
Llamada cancelada por el Receptor
tipo de llamada = Larga Distancia Nacional
Tarifa = 2
Sonando

Figura (7.36) – Prueba de marcado para el número 012262548655

Numero: *60

Validar | Borrar | Generar Numero | Llamar

Metodo de Conexion: Directo | Servidor proxy | Servidor de redireccion | Cambiar Metodo

Estado: OK
Escuchando en el puerto 5060
tipo de llamada = Solicita Servicio
Tarifa = 0
Sonando

Figura (7.37) – Prueba de marcado para el número *60

Numero: 047748659

Validar | Borrar | Generar Numero | Llamar

Metodo de Conexion: Directo | Servidor proxy | Servidor de redireccion | Cambiar Metodo

Estado: Numero no valido

Figura (7.38) – Prueba de marcado para el número 047748659

TipoLlamada	Costo
Larga Distancia Por Operadora	3
Solicita Servicio	0
Local Fijo	1
Larga Distancia Nacional	2
Local Cellular	2
Larga Distancia Internacional	3

Figura (7.39) – *Tipos de marcado soportado y costo respectivo*

7.14. Resultados registro de llamadas a la base de datos

Como se mostró en la secciones 7.9 *Resultados modelo conexión directa*, 7.10 *Resultados modelo conexión con servidor proxy*, 7.11 *Resultados modelo de conexión servidor de redirección* en la figura 7.40 muestra que cada llamada realiza es registrada en la base de datos.

IpEmisor	IpReceptor	NumeroEmisor	NumeroReceptor	TipoLlamada	Minutos	Monto	Fecha
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	4	4	Mon Mar 12 22:53:13 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	1	1	Mon Mar 12 23:21:39 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	2	2	Mon Mar 12 23:24:41 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	1	1	Tue Mar 13 00:30:09 CST 2007
127.0.0.1	127.0.0.1	0552227722008	2865676	Local Fijo	4	4	Mon Mar 12 23:48:26 CST 2007

Figura (7.40) – Registro en la Base de Datos

7.15. Resultado de identificación de puertos

Como se mostró en la secciones 7.9 *Resultados modelo conexión directa*, 7.10 *Resultados modelo conexión con servidor proxy*, 7.11 *Resultados modelo de conexión servidor de redirección* en la figura 7.40 muestra los puertos utilizados para que cada servidor y cliente escuche peticiones de llamadas.

TCP	ragnarok:5060	0.0.0.0	LISTENING
TCP	ragnarok:5061	0.0.0.0	LISTENING
TCP	ragnarok:5062	0.0.0.0	LISTENING
TCP	ragnarok:5063	0.0.0.0	LISTENING

Figura (7.41) – Registro de puertos

Capítulo 8

Conclusiones

Para terminar, no resta más que presentar las conclusiones obtenidas del desarrollo de esta tesis, cuyo principal objetivo presentando desde el inicio es realizar una aplicación de software que pueda llevar a cabo la tarificación de llamadas telefónicas usando el estándar de VoIP.

En base a lo anterior se concluye lo siguiente:

- Se realizó todo un proceso de desarrollo de software utilizando el método de desarrollo unificado de software. El software permite la conexión y envío de señales a través de la Internet para establecer las llamadas entre los usuarios, incluye los tres mecanismos de conexión establecidos en el estándar que incluyen conexión directa entre emisor y receptor, conexión utilizando un servidor de redirección y conexión con servidor proxy. Cada uno de estos mecanismos tienen sus ventajas y desventajas para la tarificación y control de las llamadas, que detallaremos a continuación.

Conexión directa Este mecanismo tiene la ventaja que no requiere de elementos externos para poder establecer la conexión con otro usuario de la red telefónica, y fue sencillo su implementación dentro del software al establecer las conexiones y el control de los mensajes y señales intercambiados entre los usuarios, sin embargo al ser un mecanismo simple no es tan sencillo integrar la tarificación ya que los paquetes son enviados por la Internet y nadie sabe en que momento

empezaron la llamada y en que momento terminará. Es por ello que para este mecanismo se recomienda integrar dentro del dispositivo de conexión del usuario emisor (Terminal de Usuario), de ese modo el propio dispositivo llevara el registro y gestión de las llamadas realizadas por el usuario, esto implica que el usuario tiene acceso al dispositivo y puede modificarlo para evitar que sus llamadas sean registradas. No se recomienda implementar este mecanismo dentro de una red grande de usuarios ya que seria poco practico y difícil su control.

Conexión con servidor de redirección Este mecanismo de conexión implica utilizar un equipo que sea destinado como servidor, este mecanismo es sencillo de implementar, implica que los usuarios cuando deseen comunicarse con otro usuario, primero deben acceder al servidor de redirección solicitando las direcciones de los demas usuarios para poder conectarse, en este caso el control y registro de llamadas y tarifa no puede aplicarse en el servidor de redirección y debe funcionar desde la terminal de usuario.

Tenerlo en el servidor no es posible en primer lugar porque cuando un usuario quiera conectarse a otro debe preguntar al servidor de redirección en este momento se podría registrar la llamada, pero como solo es una consulta de dirección no hay llamada establecida aún, solo la intención de establecerla. Suponiendo que se registra no hay manera de saber en que momento se finalizará la llamada entre los usuarios que pidieron el servicio de redirección. Regresando a un esquema de mecanismo de conexión directo con sus respectivas particularidades expuestas anteriormente.

Conexión con servidor proxy Este mecanismo implica la utilización de un equipo para el manejo de las llamadas entre los usuarios, este mecanismo es el utilizado por las empresas para llevar el control de las llamadas entre usuarios de la red, su implementación es un poco mas complicada que las anteriores ya que debe existir un intermediario que debe manejar la conexión con el emisor y receptor al mismo tiempo con la perdida mínima de calidad de servicio, ya que de lo contrario el servicio se vería muy degradado dando un bajo rendimiento.

En este caso el servidor proxy es que debe llevar el control y gestión de las llamadas, así que es mucho mas fácil controlarlas desde el servidor proxy, aplicarles tarifas. Los usuarios solo podrán conectarse con otros usuarios a travez del proxy haciendo el mecanismo de conexión mas adecuad para la tarificación de las llamadas y por esa razón se recomienda esté método.

- Durante la elaboración de este proyecto ha habido avances en los desarrollos de telefonía Voip. Entre las cuales se encuentra el ser servidor Asterisk de licencia GNU, proveyendo un sistema de central telefónica (PBX) el cuál tiene las capacidades de conexión Voip específicamente con SIP, sin embargo esta desarrollando sus propios protocolos *IAX (Inter-Asterisk eXchange protocol)* para el manejo de llamadas VoIP se recomienda un análisis de lo planteado en esta tesis para aplicarlo en un sistema Asterisk.

Llegado a este punto este trabajo de tesis, sirve como referencia para la realización de trabajos futuros en el campo de VoIP y específicamente en el protocolo de señalización SIP y métodos de tarificación para el control de llamadas VoIp.

Terminología

Termino	Descripción
Emisor	Persona que transmite algo a los demás, en este contexto aquel que inicia una llamada.
Receptor	Persona que recibe el mensaje a través del canal y lo interpreta, en este contexto aquel que es susceptible a recibir llamadas telefónica
Canal	Elemento físico que establece la conexión entre el emisor y el receptor, en este caso se refiere a la conexión utilizando los protocolos TCP/IP.
Mensaje	Información que el emisor envía al receptor
Envío de Medios	Representa el mensaje codificado y transportado a travez de la conexión TCP/IP.
Puerto	Representa un puerto TCP o UDP, es decir una numeración lógica que se asigna a las conexiones, tanto en el origen como en el destino
Protocolo	Conjunto de reglas para establecer una comunicación

Lista de Símbolos y Abreviaciones

Abreviación	Descripción	Definición
VoIP	Voice over Internet Protocol Voz sobre Protocolo de Internet	page 9
SIP	Session Initiation Protocol Protocolo de inicio de sesión	page 3
LDAP	Lightweight Directory Access Protocol Protocolo Ligero de Acceso a directorios	page 19
DNS	Domain Name Server Servidor de Nombres de Dominio	page 19
TRIP	Telephony Routing over IP Ruteo de Telefonía sobre IP	page 19
RSVP	Reservation Protocol Protocolo de Reservación de recursos	page 19
IETF	Internet Engineering Task Force Fuerza Ingenieril de Internet	page 3
PUDS	Proceso Unificado de Desarrollo de Software	page 3
UML	Unified Modeling Language Lenguaje Unificado de Modelado	page 3
MGCP	Media Gateway Control Protocol Protocolo de Control de pasarela de Medio	page 5
PSTN	Public Switched Telephone Network Red Telefónica Pública Tradicional	page 9
OSI	Open System Interconnectivity Interconectividad de Sistemas Abiertos	page 12
ISO	International Standar Organization Organización Internacional de Estándares	page 12
TCP/IP	Transport Control Protocol/Internet Protocol Protocolo de Control de Transporte/Protocolo de Internet	page 75

Abreviación	Descripción	Definición
TCP	Transport Control Protocol Protocolo de Control de Transporte	page 75
UDP	User Datagram Protocol Protocolo de Datagrama de Usuario	page 19
SSP	Service Switching Point page 11	
STP	Signal Transfer Point	page 11
SCP	Service Control Point	page 11
SS7	Common Channel Signaling System No.7 Sistema de Señalización de Canal Común Numero 7	page 11
MTP	Message Transfer Part	page 13
ISDN	Integrated Services Digital Network Red Digital de Servicios Integrados	page 14
API	Aplication Programming Interface Interfaz de programación de aplicaciones	page 63

Índice de figuras

2.2. Puntos de señalización	12
2.3. Tipos de enlaces	12
2.4. Pila de protocolos SS7	14
3.1. Arquitectura SIP	20
3.2. Modelo de llamada básica SIP[8]	23
3.3. Modelo de llamada básica SIP con servidor proxy[8]	24
3.4. Modelo de llamada básica SIP con servidor de redirección[8]	25
3.5. Modelo de llamada PSTN a Teléfono VoIP[9]	27
3.6. Modelo de llamada Teléfono VoIP a PSTN[9]	27
4.1. Casos de uso del sistema	32
4.2. Clases del sistema	35
4.3. Clases emisor, receptor, llamada activa y sesión VoIP	36
4.4. Clases servidor proxy y servidor de redirección	37
5.1. Operación llamar	40
5.2. Operación hablar	41
5.3. Operación terminar llamada	42
5.4. Operación generar número	42
5.5. Operación establecer llamada SIP	43
5.6. Operación terminar llamada SIP	43
5.7. Operación buscar cliente	44
5.8. Operación redirigir	44
5.9. Operación registro	45
5.10. Operación servidor proxy	46
5.11. Operación directo	46
5.12. Operación redirección	47
5.13. Operación revisar saldo	48
5.14. Operación llamar	49

5.15. Operación hablar	50
5.16. Operación terminar llamada	50
5.17. Operación generar número	51
5.18. Operación establecer llamada SIP	52
5.19. Operación terminar llamada SIP	52
5.20. Operación Buscar cliente	53
5.21. Operación redirigir	53
5.22. Operación registro	54
5.23. Operación servidor proxy	54
5.24. Operación directo	55
5.25. Operación redirección	55
5.26. Operación revisar saldo	56
5.27. Objeto emisor	57
5.28. Objeto receptor	58
5.29. Objeto generador de números telefónicos	58
5.30. Objeto validar número	59
5.31. Objeto registro de llamadas	59
5.32. Objeto Contador	60
5.33. Objeto llamada activa	60
5.34. Objeto sesion VoIP	61
5.35. Objeto Servidor Proxy	61
5.36. Objeto servidor de redirección	62
6.1. Proceso de compilación	64
6.2. Programa Java sobre múltiples plataformas	65
6.3. Plataforma Java	65
6.4. Clase terminal, principal interfaz de usuario	66
6.5. Clase envío de medios	67
7.1. Interfaz emisor	74
7.2. Interfaz receptor	74
7.3. Puertos utilizados al iniciar el sistema	74
7.4. Modelo de llamada básica SIP	75
7.5. Mensajes invitar y sonando entre el emisor y el receptor	76
7.6. Mensajes enviados a la consola de salida	76
7.7. Llamada aceptada entre el emisor y el receptor	76
7.8. Conexión TCP entre el emisor y el receptor	76
7.9. Llamada finalizada entre el emisor y el receptor	77
7.10. Mensajes enviados a la consola	77
7.11. Registro de llamadas en base de datos	77
7.12. receptor y emisor	78
7.13. servidor proxy	78

7.14. Puertos	79
7.15. Modelo de llamada básica SIP con servidor proxy	79
7.16. Mensajes entre servidor proxy, emisor y receptor	80
7.17. Llamada establecida entre emisor y receptor	81
7.18. Puertos utilizados por la aplicación en la llamada	81
7.19. Registro de los mensajes enviados durante la llamada	81
7.20. Llamada terminada emisor (izquierda) y receptor(derecha)	82
7.21. Resultado del servidor proxy	82
7.22. Registro en la base de datos	83
7.23. receptor y emisor	84
7.24. servidor de redirección	84
7.25. Puertos	85
7.26. Modelo de llamada básica SIP con servidor de redirección	85
7.27. Mensajes entre servidor proxy, emisor y receptor	85
7.28. Llamada establecida entre el emisor y receptor	86
7.29. Fin de llamada entre emisor y receptor	86
7.30. Registro en la base de datos	87
7.31. emisor creando dos conexión al receptor	88
7.32. Resultados de conexiones múltiples	88
7.33. emisor creando conexión a distintos receptores	89
7.34. Prueba de marcado para el número 2865676	90
7.35. Prueba de marcado para el número 0445528693	90
7.36. Prueba de marcado para el número 012262548655	91
7.37. Prueba de marcado para el número *60	91
7.38. Prueba de marcado para el número 047748659	91
7.39. Tipos de marcado soportado y costo respectivo	92
7.40. Registro en la Base de Datos	93
7.41. Registro de puertos	93

Índice de cuadros

2.1. Tipos de enlaces.[3]	13
2.2. Verbos usados por S/MGCP	16
2.3. Modos de conexión para S/MGCP	17
3.1. Métodos del protocolo SIP	22

Bibliografía

- [1] I. R. 2543, “Sip: Session initial protocol,” Internet, IETF, Tech. Rep., Marzo 1999. [Online]. Available: <http://www.ietf.org/rfc/rfc2543.txt> [citado en p. 3, 75, 81, 87]
- [2] C. R. C. C. Laborja J., López M., “Redes fijas: Voz sobre ip (voip),” 2005. [Online]. Available: <http://salaam.cs.buap.mx/VoIP> [citado en p. 3]
- [3] P. Technologies, “Ss7 tutorial,” Internet, Performance Technologie, Tech. Rep., 2006. [Online]. Available: <http://www.pt.com/tutorials/ss7/> [citado en p. 11, 12, 13, 16, 106]
- [4] I. R. 2705, “Media gateway control protocol (mgcp),” Internet, IETF, Tech. Rep., Octubre 1999. [Online]. Available: <http://www.ietf.org/rfc/rfc2705.txt> [citado en p. 16]
- [5] D. Bill., *IP Telephony, The Integration of Robust VoIP Services*. Ed. Prentice Hall PTR, 2000. [citado en p. 17, 19]
- [6] T. Faq, “What is voip,” Tech Faq, Tech. Rep., 2005. [Online]. Available: <http://www.tech-faq.com/voip.shtml> [citado en p. 19, 20]
- [7] I. R. 1889, “Rtp: A transport protocol for real-time applications,” Internet, IETF, Tech. Rep., 1996 Enero. [Online]. Available: <http://www.ietf.org/rfc/rfc1889.txt> [citado en p. 20, 21]
- [8] I. R. 2327, “Sdp: Session description protocol,” Internet, IETF, Tech. Rep., Abril 1998. [Online]. Available: <http://www.ietf.org/rfc/rfc2327.txt> [citado en p. 21, 22, 23, 24, 25, 103]
- [9] I. R. 3372, “Session initiation protocol for telephones (sip-t),” Internet, IETF, Tech. Rep., Septiembre 2002. [Online]. Available: <http://www.ietf.org/rfc/rfc3372.txt> [citado en p. 25, 27, 103]
- [10] S. j, *Aprendiendo UML en 24 horas*. Prentice Hall, 1999. [citado en p. 39]
- [11] J. S. Tutorial, “The java programming language,” Internet, Java sun, Tech. Rep., 2006. [Online]. Available: <http://java.sun.com/docs/books/tutorial/> [citado en p. 63]