



TABLA DE CONTENIDO

C1	MODELO DE CASOS DE USO DEL DISEÑO	2
C1.1	DIAGRAMAS DE CASOS DE USO DEL DISEÑO	2
C1.2	DESCRIPCIÓN DE CASOS DE USO DEL DISEÑO	3
C2	MODELO ESTÁTICO DEL DISEÑO	14
C2.1	DIAGRAMAS DE PAQUETES DEL DISEÑO	14
C2.1.1	VISTA DEL SERVIDOR	14
C2.1.1.1	PRIMER NIVEL	15
C2.1.1.2	SEGUNDO NIVEL.....	17
C2.1.1.3	TERCER NIVEL.....	17
C2.1.2	VISTA DEL CLIENTE MÓVIL	18
C2.2	DIAGRAMAS DE CLASES DEL DISEÑO.....	19
C2.2.1	VISTA DEL CLIENTE MÓVIL MIDP 2.0	19
C2.2.2	VISTA DEL CLIENTE MÓVIL MIDP 1.0	20
C2.2.3	VISTA DEL SERVIDOR	21
C3	MODELO DINÁMICO DEL DISEÑO.....	22
C3.1	ADICIONAR CONTACTO	22
C3.2	CAMBIAR PRESENCIA	22
C3.3	CERRAR SESIÓN	23
C3.4	ELIMINAR CONTACTO.....	24
C3.5	INICIAR CHAT	25
C3.6	INICIAR SESIÓN	26
C4	DIAGRAMA ENTIDAD RELACIÓN.....	27
C5	DIAGRAMA DE IMPLANTACIÓN	27



ANEXO C

DISEÑO DE PAUMIM.

C1 Modelo de Casos de Uso del Diseño

C1.1 Diagramas de Casos de Uso del Diseño

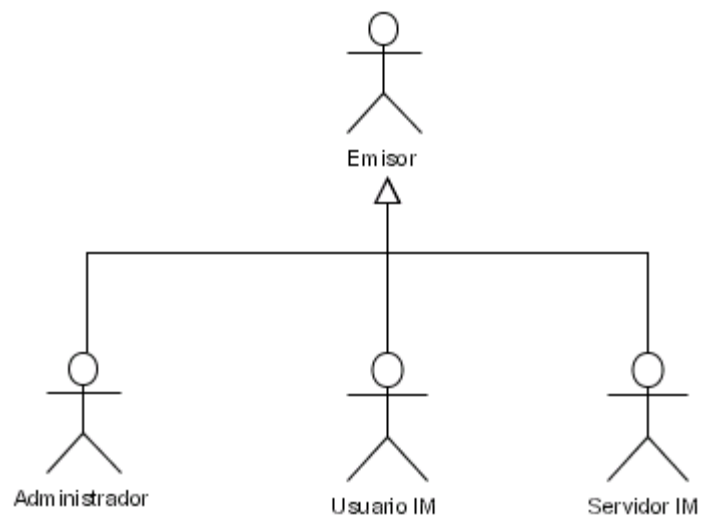


Figura C1. Diagrama de Casos de Uso, vista de los actores

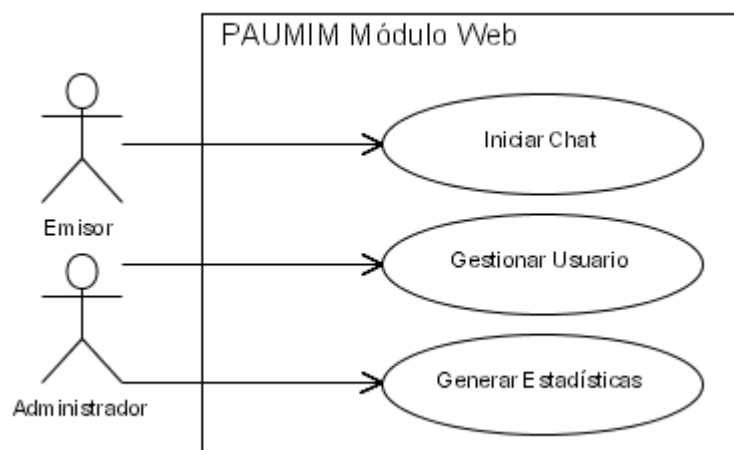


Figura C2. Diagrama de Casos de Uso, vista del administrador

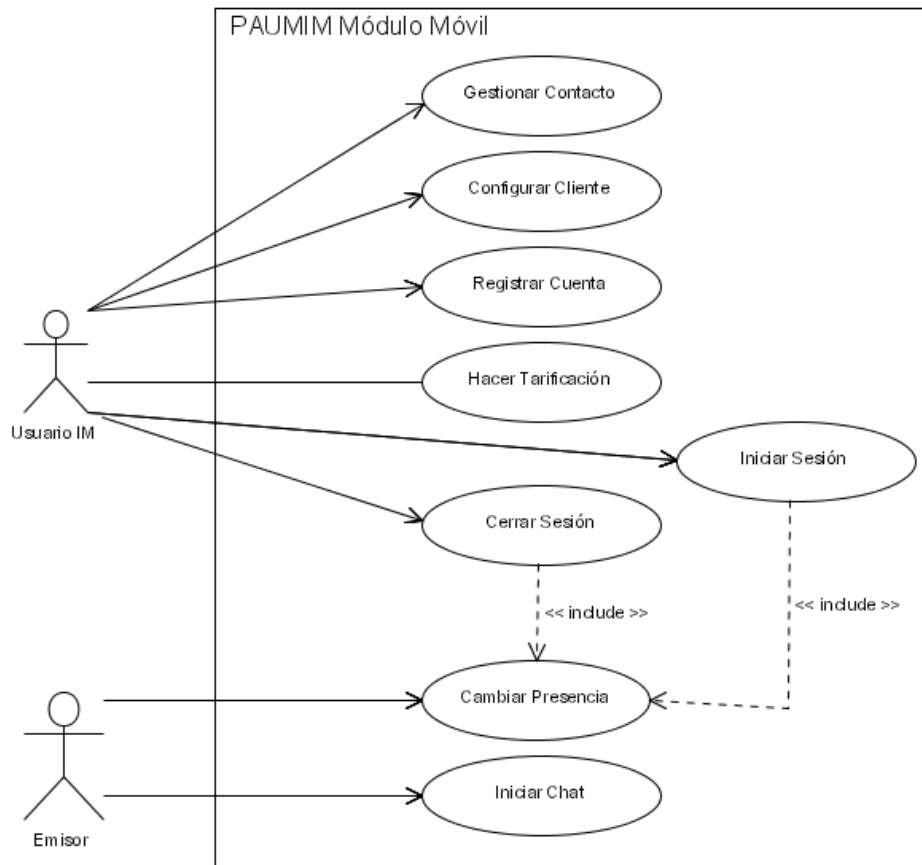


Figura C3. Diagrama de Casos de Uso, vista del usuario

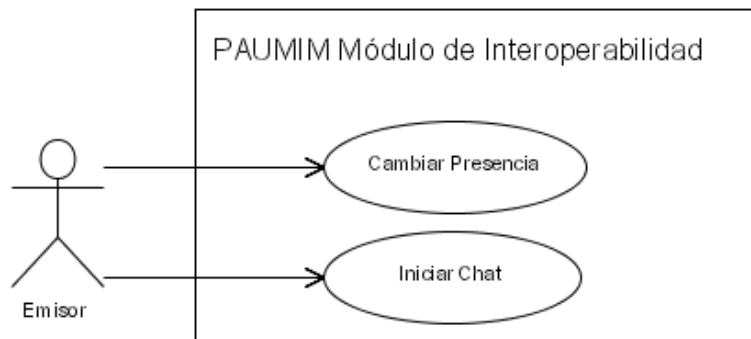


Figura C4. Diagrama de Casos de Uso, vista del servidor

C1.2 Descripción de Casos de Uso del Diseño

Caso de Uso	Generar Estadísticas
--------------------	----------------------

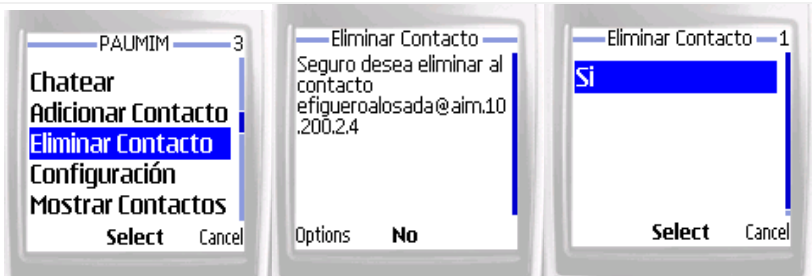


Actores	Administrador								
Tipo	Automático								
Resumen	El sistema genera información estadística referente al tiempo de uso de la plataforma, tráfico cursado y horas pico. Estos datos son producidos de manera automática y luego consultados por el administrador.								
Precondiciones									
Poscondiciones	Las tablas de tiempo de uso, tráfico cursado y horas pico son modificadas.								
Flujo Normal	<table border="1"><thead><tr><th>Actores</th><th>Sistema</th></tr></thead><tbody><tr><td>1. Cualquier evento de comunicación dentro de la plataforma.</td><td>2. Cuenta el tiempo de uso, el tráfico cursado, evalúa las horas pico.</td></tr><tr><td>4. El administrador elige la opción de ver estadísticas.</td><td>3. Actualiza los registros de cada una de estas variables medidas.</td></tr><tr><td></td><td>5. Muestra las tablas y gráficas con la información estadística de la plataforma.</td></tr></tbody></table>	Actores	Sistema	1. Cualquier evento de comunicación dentro de la plataforma.	2. Cuenta el tiempo de uso, el tráfico cursado, evalúa las horas pico.	4. El administrador elige la opción de ver estadísticas.	3. Actualiza los registros de cada una de estas variables medidas.		5. Muestra las tablas y gráficas con la información estadística de la plataforma.
Actores	Sistema								
1. Cualquier evento de comunicación dentro de la plataforma.	2. Cuenta el tiempo de uso, el tráfico cursado, evalúa las horas pico.								
4. El administrador elige la opción de ver estadísticas.	3. Actualiza los registros de cada una de estas variables medidas.								
	5. Muestra las tablas y gráficas con la información estadística de la plataforma.								
Cursos Alternos									

Caso de Uso	Hacer Tarificación				
Actores	Usuario IM				
Tipo	Automático				
Resumen	El caso de uso comienza cuando se envía cualquier tipo de mensaje a través de la red, el sistema cuenta el número de bytes transmitidos para generar un acumulado y una tabla de consumo que el usuario y puede ver posteriormente.				
Precondiciones	Iniciar Sesión, transmitir mensajes a través de la red.				
Poscondiciones	Se ha modificado la tabla de consumo del usuario MI.				
Flujo Normal	<table border="1"><thead><tr><th>Actores</th><th>Sistema</th></tr></thead><tbody><tr><td>1. El usuario ha enviado un</td><td>2. Cuenta el número de bytes</td></tr></tbody></table>	Actores	Sistema	1. El usuario ha enviado un	2. Cuenta el número de bytes
Actores	Sistema				
1. El usuario ha enviado un	2. Cuenta el número de bytes				



	mensaje hacia la red. 4. El usuario elige ver la tarificación de su consumo.	transmitidos. 3. Actualiza el acumulado y la tabla de consumo del usuario. 5. Muestra la información del consumo del usuario.
Cursos Alternos		

Caso de Uso	Eliminar Contacto	
Actores	Usuario IM (iniciador)	
Tipo	Primario	
Resumen	El actor inicia el caso de uso, eligiendo de su lista de contactos, al usuario que va a eliminar. El sistema elimina dentro de las cuentas de usuario al contacto eliminado.	
Precondiciones	Iniciar Sesión	
Poscondiciones	Se ha eliminado de cada lista de contactos el usuario eliminado.	
Flujo Normal	Actores 1. El usuario escoge de la lista el contacto que va a ser eliminado.	Sistema 2. Elimina al contacto especificado, de la lista perteneciente al actor iniciador. 3. Elimina al usuario que ha iniciado el caso de uso, de la lista de contactos del usuario eliminado.
Cursos Alternos		
GUIs Relacionadas		



Caso de Uso	Adicionar Contacto	
Actores	Usuario IM (iniciador)	
Tipo	Primario	
Resumen	El caso de uso comienza cuando un usuario IM desea agregar un nuevo contacto a su lista, especifica su cuenta y envía la petición. Si el contacto acepta la inclusión dentro de la lista, el sistema modifica la lista de contactos de cada uno de los usuarios involucrados.	
Precondiciones	Iniciar Sesión	
Poscondiciones	Las cuentas de usuario de los dos actores involucrados, son cambiadas; específicamente la lista contactos.	
Flujo Normal	Actores 1. El usuario escoge la acción de adicionar contacto. 3. Escribe el nombre de la cuenta y confirma la adición. 5. El otro usuario aprueba	Sistema 2. Muestra un campo para que escriba la cuenta del nuevo contacto. 4. Envía una petición al usuario solicitado, para aprobar la agregación a la lista 6. Cambia la cuenta de cada uno de los usuarios involucrados, agregando a cada quien el nuevo elemento en su lista de contactos.
Cursos Alternos	L2. El usuario ingresa el nombre de una cuenta que no existe, entonces el sistema luego de verificar informa de esto al actor para que escriba correctamente. L5. El otro usuario no aprueba la adición, entonces el caso de uso termina y no se hace ninguna modificación en la lista de contactos.	



Caso de Uso	Configurar Cliente	
Actores	Usuario MI (iniciador)	
Tipo	Primario	
Resumen	En este caso de uso, el usuario MI define las características de comunicación y de uso para el cliente móvil, el sistema por su parte guarda esta nueva configuración y la usa mientras no sea cambiada nuevamente.	
Precondiciones	Ninguna	
Poscondiciones	Es cambiada la configuración de la aplicación móvil	
Flujo Normal	<p style="text-align: center;">Actores</p> <ol style="list-style-type: none"> 1. El usuario elige la opción de cambiar configuración. 3. Escoge y cambia lo ítems que desea editar. 4. Confirma los cambios realizados previamente 	<p style="text-align: center;">Sistema</p> <ol style="list-style-type: none"> 2. Muestra un formulario para cambiar las diferentes opciones de la aplicación. 5. El sistema guarda la nueva configuración de la aplicación.
Cursos Alternos	L4. El usuario no confirma los cambios en la configuración, entonces el caso de uso termina y no se concreta ningún cambio.	





Caso de Uso	Eliminar Usuario	
Actores	Administrador (iniciador)	
Tipo	Primario	
Resumen	El caso de uso comienza cuando el administrador elige un usuario que desea eliminar del sistema, a continuación pregunta si realmente se va al eliminar al usuario, una vez se haga la confirmación, la cuenta del usuario elegido es borrada del sistema.	
Precondiciones	La cuenta de usuario debe haber sido creada anteriormente	
Poscondiciones	Cuenta de usuario eliminada	
Flujo Normal	Actores 1. El administrador elige de una lista, la cuenta al usuario que va a eliminar. 3. El administrador confirma la eliminación del usuario.	Sistema 2. El sistema pregunta si realmente se va a eliminar al usuario escogido. 4. El sistema borra del sistema la cuenta del usuario.
Cursos Alternos	L3. El administrador no confirma la eliminación, entonces el caso de uso termina y no ocurre nada referente a la cuenta de usuario.	

Caso de Uso	Editar Usuario	
Actores	Administrador (iniciador)	
Tipo	Primario	
Resumen	El caso de uso comienza con la elección del usuario involucrado dentro de este caso de uso, luego de que el sistema muestre al administrador, los datos actuales del usuario, entonces este realiza los cambios y finalmente confirma el cambio.	
Precondiciones	La cuenta de usuario debe haber sido creada anteriormente.	
Poscondiciones	Son cambiados los datos de una cuenta de usuario.	
Flujo Normal	Actores 1. El administrador elige de una	Sistema 2. El sistema muestra la

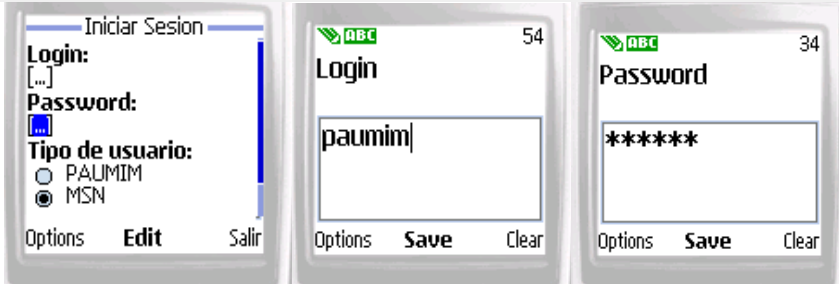


	lista, la cuenta del usuario que va a modificar. 3. El administrador cambia los datos de la cuenta. 4. Confirma la modificación de la cuenta.	información personal de los usuarios. 5. Modifica la cuenta de usuario.
Cursos Alternos	L4. Si alguno de los datos de no es válido o no corresponde al formato requerido por el sistema, entonces el sistema muestra un mensaje informando de este suceso.	

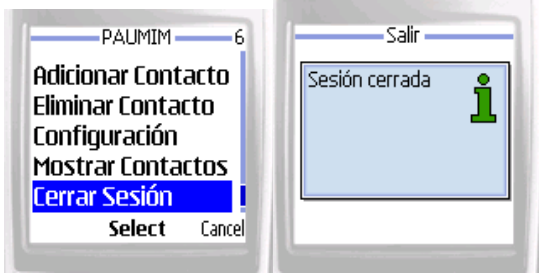
Caso de Uso	Adicionar Usuario	
Actores	Administrador (iniciador)	
Tipo	Primario	
Resumen	Este caso de uso permite agregar nuevos usuarios a la plataforma, luego de llenar una plantilla con los datos requeridos, el administrador inicia el evento de adición. Entonces el sistema crea una nueva cuenta para el usuario concerniente.	
Precondiciones	Ninguna	
Poscondiciones	Es creada una nueva cuenta de usuario	
Flujo Normal	Actores 1. El administrador ingresa el login, la contraseña y otros datos personales del usuario. 2. Envía la notificación de adición de nuevo usuario.	Sistema 3. Hace el nuevo registro del usuario. 4. Muestra un mensaje de confirmación.
Cursos Alternos		

Caso de Uso	Iniciar Sesión
Actores	Usuario MI(iniciador), Servidor MI



Tipo	Primario														
Resumen	El caso de uso iniciar sesión comienza cuando un Usuario MI, después de haber escrito un nombre de usuario y una contraseña envía un mensaje para indicar el inicio de una sesión de mensajería instantánea. Entonces el sistema realiza los procedimientos correspondientes para establecer los caminos de comunicación y además informa de una actualización de presencia por parte del usuario iniciador.														
Precondiciones	Ninguna														
Poscondiciones	Es creada una nueva sesión, es cambiada la presencia del usuario														
Flujo Normal	<table border="1"><thead><tr><th>Actores</th><th>Sistema</th></tr></thead><tbody><tr><td>1. El usuario ingresa el login y la contraseña.</td><td></td></tr><tr><td>2. Envía la notificación de inicio de sesión.</td><td>3. Hace la validación del login y de la contraseña del usuario.</td></tr><tr><td></td><td>4. Envía un mensaje de validación al servidor MI</td></tr><tr><td>5. El servidor MI hace la validación del login y de la contraseña del usuario.</td><td>6. Crea una nueva sesión.</td></tr><tr><td></td><td>7. Muestra al usuario la lista de contactos con la presencia de cada uno.</td></tr><tr><td></td><td>8. Cambiar la presencia del usuario.</td></tr></tbody></table>	Actores	Sistema	1. El usuario ingresa el login y la contraseña.		2. Envía la notificación de inicio de sesión.	3. Hace la validación del login y de la contraseña del usuario.		4. Envía un mensaje de validación al servidor MI	5. El servidor MI hace la validación del login y de la contraseña del usuario.	6. Crea una nueva sesión.		7. Muestra al usuario la lista de contactos con la presencia de cada uno.		8. Cambiar la presencia del usuario.
Actores	Sistema														
1. El usuario ingresa el login y la contraseña.															
2. Envía la notificación de inicio de sesión.	3. Hace la validación del login y de la contraseña del usuario.														
	4. Envía un mensaje de validación al servidor MI														
5. El servidor MI hace la validación del login y de la contraseña del usuario.	6. Crea una nueva sesión.														
	7. Muestra al usuario la lista de contactos con la presencia de cada uno.														
	8. Cambiar la presencia del usuario.														
Cursos Alternos	L3, L5. La validación no es exitosa, entonces el sistema informa al usuario, para que haga nuevamente ingrese el login y la contraseña.														
GUIs Relacionadas															



Caso de Uso	Cerrar Sesión				
Actores	Usuario IM(iniciador), Servidor IM				
Tipo	Primario				
Resumen	El caso de uso iniciar sesión comienza cuando un Usuario IM desea terminar la sesión de mensajería y entonces elige la opción del menú terminar sesión. El sistema realiza los procedimientos correspondientes para terminar la comunicación y además informa de una actualización de presencia a los contactos del usuario iniciador.				
Precondiciones	Iniciar Sesión				
Poscondiciones	Es eliminada una nueva sesión, es cambiada la presencia del usuario				
Flujo Normal	<table border="1"><thead><tr><th>Actores</th><th>Sistema</th></tr></thead><tbody><tr><td>1. El usuario elige del menú la opción cerrar sesión.</td><td>2. Termina la sesión. 3. Cambia la presencia del usuario. 4. Envía esta nueva presencia a los contactos del usuario.</td></tr></tbody></table>	Actores	Sistema	1. El usuario elige del menú la opción cerrar sesión.	2. Termina la sesión. 3. Cambia la presencia del usuario. 4. Envía esta nueva presencia a los contactos del usuario.
Actores	Sistema				
1. El usuario elige del menú la opción cerrar sesión.	2. Termina la sesión. 3. Cambia la presencia del usuario. 4. Envía esta nueva presencia a los contactos del usuario.				
Cursos Alternos					
GUIs Relacionadas					


Caso de Uso	Cambiar Presencia
Actores	Usuario MI(iniciador), Servidor MI
Tipo	Primario
Resumen	El actor comienza el caso de uso, en el momento en que realiza el evento de cambiar su presencia, es entonces cuando el sistema realiza la acción de notificar a todos los contactos conectados del usuario



	iniciador, acerca del cambio en la presencia del usuario MI.	
Precondiciones	Iniciar Sesión	
Poscondiciones	Es cambiada la presencia del usuario	
Flujo Normal	Actores 1. El usuario cambia el estado en su presencia.	Sistema 2. Envía un mensaje de cambio de presencia al servidor MI. 3. Cambia el estado de presencia del usuario. 4. Actualiza la interfaz gráfica con el nuevo estado de presencia del usuario
Cursos Alternos		

Caso de Uso	Iniciar Chat	
Actores	Usuario IM(iniciador), Servidor IM	
Tipo	Primario	
Resumen	<p>El caso de uso comienza cuando un usuario IM, desde un dispositivo móvil, desea iniciar una conversación con uno de sus contactos. Entonces el actor iniciador primero elige de su lista, al contacto elegido para tal propósito, seguidamente escribe el mensaje de texto y da la orden de enviar.</p> <p>El sistema por su parte, se encarga de manejar el direccionamiento del mensaje de acuerdo a la red a la cual pertenece el usuario de destino, y hacerlo llegar hasta su ventana de chat.</p>	
Precondiciones	Iniciar Sesión, disponibilidad en la presencia del usuario IM receptor	
Poscondiciones	Un nuevo mensaje ha llegado al usuario IM receptor.	
Flujo Normal	Actores 1. El usuario IM elige el destinatario de su lista de contactos. 3. El usuario IM escribe el	Sistema 2. Muestra una pantalla de chat. 4. Agrega el mensaje a la



	<p>mensaje y da la orden de enviar.</p> <p>8. El servidor IM reenvía el mensaje hacia el usuario IM receptor.</p>	<p>ventana de chat del usuario IM emisor.</p> <p>5. Envía el mensaje desde el dispositivo móvil hasta el servidor PAUMIM.</p> <p>6. El servidor PAUMIM verifica la dirección de destino para encaminar el mensaje hacia la pasarela adecuada.</p> <p>7. Una vez el mensaje llegue a la pasarela, se hace la conversión de protocolos para que el mensaje se adecue a la red de mensajería propietaria y llegue al servidor IM.</p> <p>9. El mensaje llega al usuario IM receptor y se adiciona a la ventana de chat.</p>
Cursos Alternos		
GUIs Relacionadas		



C2 Modelo Estático del Diseño

C2.1 Diagramas de Paquetes del Diseño

La arquitectura del sistema con base a los paquetes se muestra en las Figuras C5 y C6 correspondientes a los paquetes del móvil y del servidor. En estos diagramas de paquetes no se tuvo en cuenta los paquetes propios del núcleo de Java. Para el diagrama de paquetes del servidor se hizo una división de la arquitectura de tres niveles. En el primer nivel se encuentran los paquetes que corresponden a módulos funcionales de la plataforma externos al ambiente de desarrollo Java. El segundo nivel corresponde a APIs de terceras partes adicionadas a manera de bibliotecas, y en el tercer nivel están los paquetes que contienen las clases que fueron implementadas por completo. Para el diagrama de paquetes del cliente se identificó un solo nivel, el correspondiente a las clases implementadas por completo.

C2.1.1 Vista del Servidor

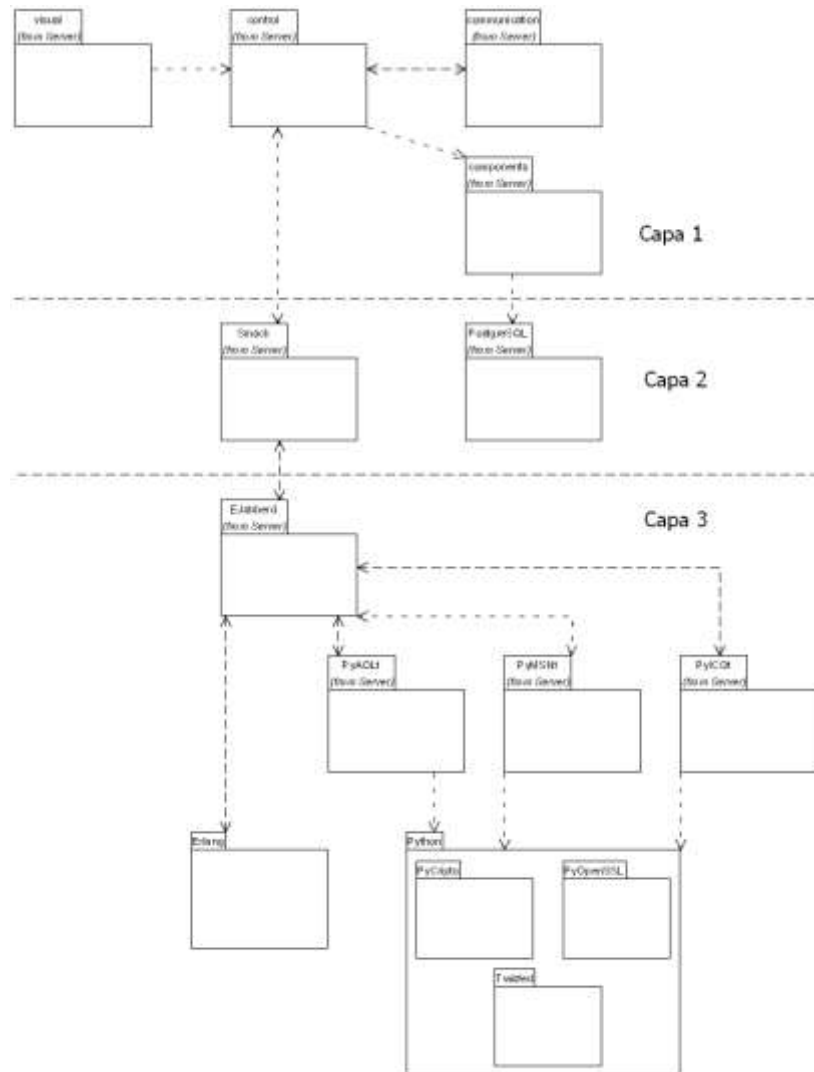


Figura C5. Diagrama de paquetes del servidor

C2.1.1.1 Primer Nivel

python: Es un lenguaje interpretado que en el momento se está desarrollando como proyecto de código abierto. Python permite dividir las aplicaciones en módulos reutilizables desde otros programas Python. Viene con una gran colección de módulos estándar que se pueden utilizar como base de las aplicaciones. [<http://es.wikipedia.org/wiki/Python>]

pyOpenSSL: Envoltura de alto nivel alrededor de una subconfiguración de la librería OpenSSL, incluye Objetos de conexión SSL, reciclaje los métodos de los sockets



portables de Python, Callbacks escritas en Python, Mecanismo extensivo de gestión de errores, reflejo de los códigos de error OpenSSL entre otros.

[http://penguin.triumf.ca/comps/9/es_frames/rpms/pyOpenSSL.php]

pyCripto: Paquete de rutinas criptográficas para Python. Contiene recursos como el manejo de funciones hash: MD2, MD4, RIPEMD, SHA256, algoritmos de encriptación de bloques: AES, ARC2, Blowfish, CAST, DES, Triple-DES, IDEA, RC5, algoritmos de encriptación de 'streams': ARC4, simple XOR, algoritmos de clave pública: RSA, DSA, ElGamal, qNEW y protocolos de transformaciones 'todo-o-nada', 'chaffing/winnowing'.

[<http://www.pythonhispano.org/node/140>]

Twisted: Es un Framework de código abierto implementado en Python especializado para el desarrollo de aplicaciones basadas en red. Está diseñado para correr en múltiples sistemas operativos y plataformas. Twisted es un intento de construcción de un Framework capaz de soportar las nuevas y modernas aplicaciones basadas en redes, desde un simple protocolo hasta un sistema integrado, multiescala y multiprotocolo. Provee múltiples niveles de abstracción, comenzando desde eventos de red de bajo nivel, pasando por código de redes genérico, implementaciones de protocolos de red hasta Frameworks de alto nivel.

[http://www.onlamp.com/pub/a/python/2004/01/15/twisted_intro.html]

Ejabberd: Es un servidor Jabber multiplataforma. Cuenta con el soporte total de las características del protocolo Jabber.

Erlang: Lenguaje funcional utilizado especialmente para desarrollo de aplicaciones distribuidas. Tiene soporte para concurrencia, distribución y tolerancia de fallos.

PyMSNt: Es un transporte desarrollado en Python. Provee una pasarela con la cual el servidor puede comunicarse con la red de MSN Messenger. El transporte debe estar instalado en el servidor Jabber, y su operación es transparente para el usuario. Este podrá interactuar con sus contactos MSN como si fueran contactos Jabber.



PyAOL - t: Es un transporte desarrollado en Python. Provee una pasarela con la cual el servidor puede comunicarse con la red de AOL. El transporte debe estar instalado en el servidor Jabber, y su operación es transparente para el usuario. Este podrá interactuar con sus contactos AOL como si fueran contactos Jabber.

PyICQ – t: Es un transporte desarrollado en Python. Provee una pasarela con la cual el servidor puede comunicarse con la red de ICQ Messenger. El transporte debe estar instalado en el servidor Jabber, y su operación es transparente para el usuario. Este podrá interactuar con sus contactos ICQ como si fueran contactos Jabber.

C2.1.1.2 Segundo Nivel

Smack API: Es una librería de código abierto para construir clientes de mensajería instantánea en lenguaje de programación Java. Puede ser embebida en aplicaciones para crear componentes de presencia y mensajería instantánea basados en el protocolo Jabber.

PostgresLibs: Este conjunto de librerías permite a los programas Java conectarse al motor de base de datos PostgreSQL. Contiene un driver tipo IV construido en su totalidad en código Java que provee en casi su totalidad la implementación de la especificación JDBC 3 en conjunto con otras extensiones específicas para PostgreSQL.

C2.1.1.3 Tercer Nivel

vista: En este paquete se encuentran las páginas JSP por medio de las cuales se lleva a cabo la comunicación directa entre el sistema y el administrador. Por medio de estas páginas el administrador puede observar las variables de estado de la plataforma y así realizar funciones de administración. Todas las páginas JSP se comunican con el paquete de control.

control: En este paquete se ubican las clases que sirven para hacer el control de la lógica



de la aplicación y la coordinación de la comunicación entre módulos y entre la plataforma y el administrador.

componentes: En este paquete se encuentran las clases Java propietarias que implementan todas las funcionalidades de la plataforma, entre las que se tienen la tarificación, la gestión de usuarios, funcionalidades de administración y manejo de presencia de usuarios.

C2.1.2 Vista del Cliente Móvil

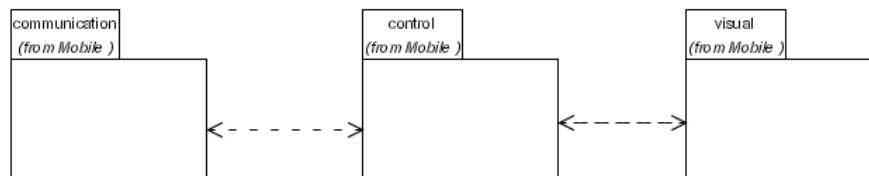


Figura C6 Diagrama de paquetes del cliente móvil

vista: En este paquete se encuentran las clases que sirven para llevar a cabo la comunicación directa entre el sistema y el usuario. Estas clases son las que se muestran en la pantalla del dispositivo móvil y por medio de las cuales el usuario podrá observar el estado de la aplicación y así mismo comunicarse con ella. Todas las clases se comunican de forma directa con la clase de control Client.

control: En este paquete se ubican las clases que sirven para hacer el control de la lógica de la aplicación y coordinar la comunicación entre esta, el usuario y el servidor. En el paquete control están las clases que permiten manejar el ciclo de vida de la aplicación y el manejo de contactos de usuario.

comunicación: En este paquete se ubican las clases involucradas en la comunicación entre el cliente móvil y el servidor PAUMIM. Para establecer la comunicación se utilizan las clases definidas en JXTA para J2ME en la parte de manejo de mensajes y una clase propietaria encargada de establecer y mantener la conexión con el servidor.



C2.2 Diagramas de Clases del Diseño

C2.2.1 Vista del Cliente Móvil MIDP 2.0

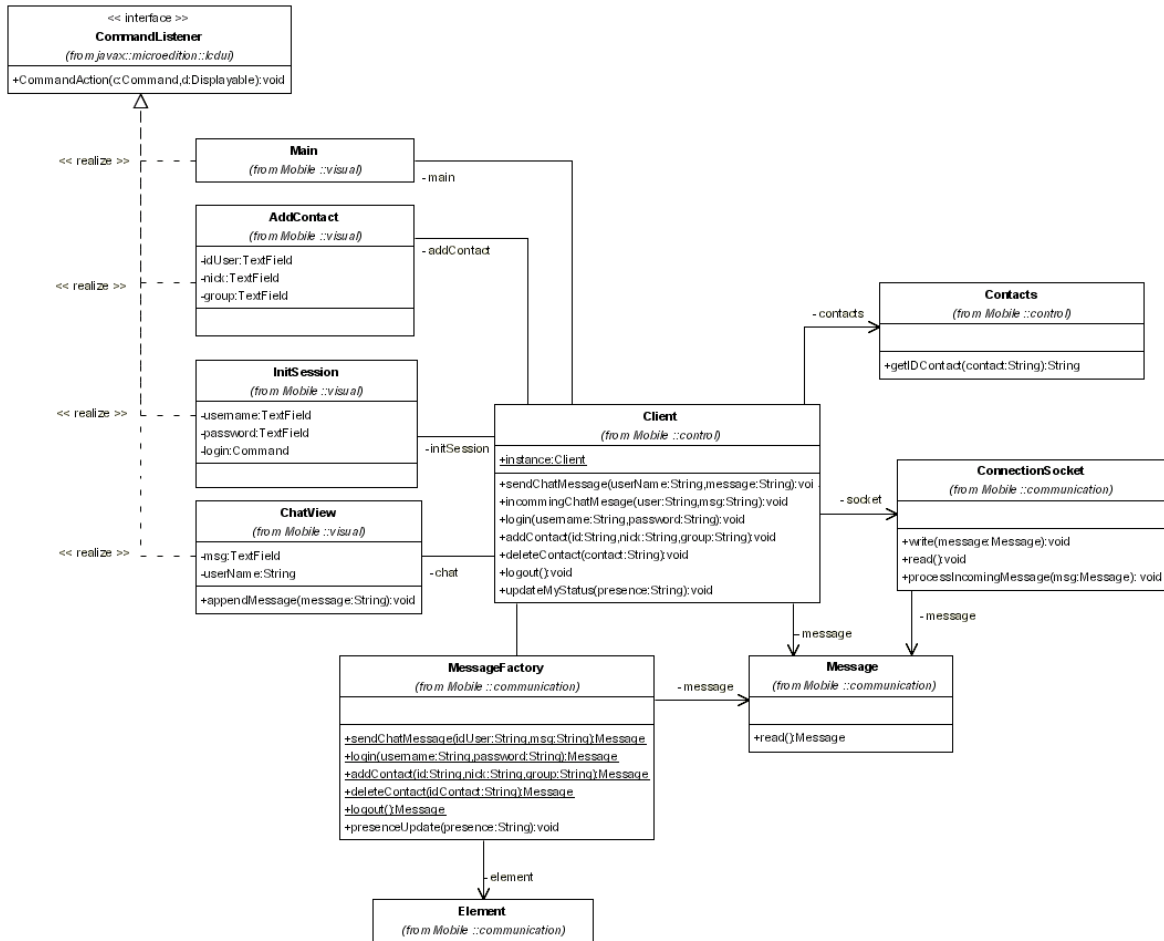


Figura C7. Diagrama de Clases del Cliente Móvil MIDP 2.0



C2.2.2 Vista del Cliente Móvil MIDP 1.0

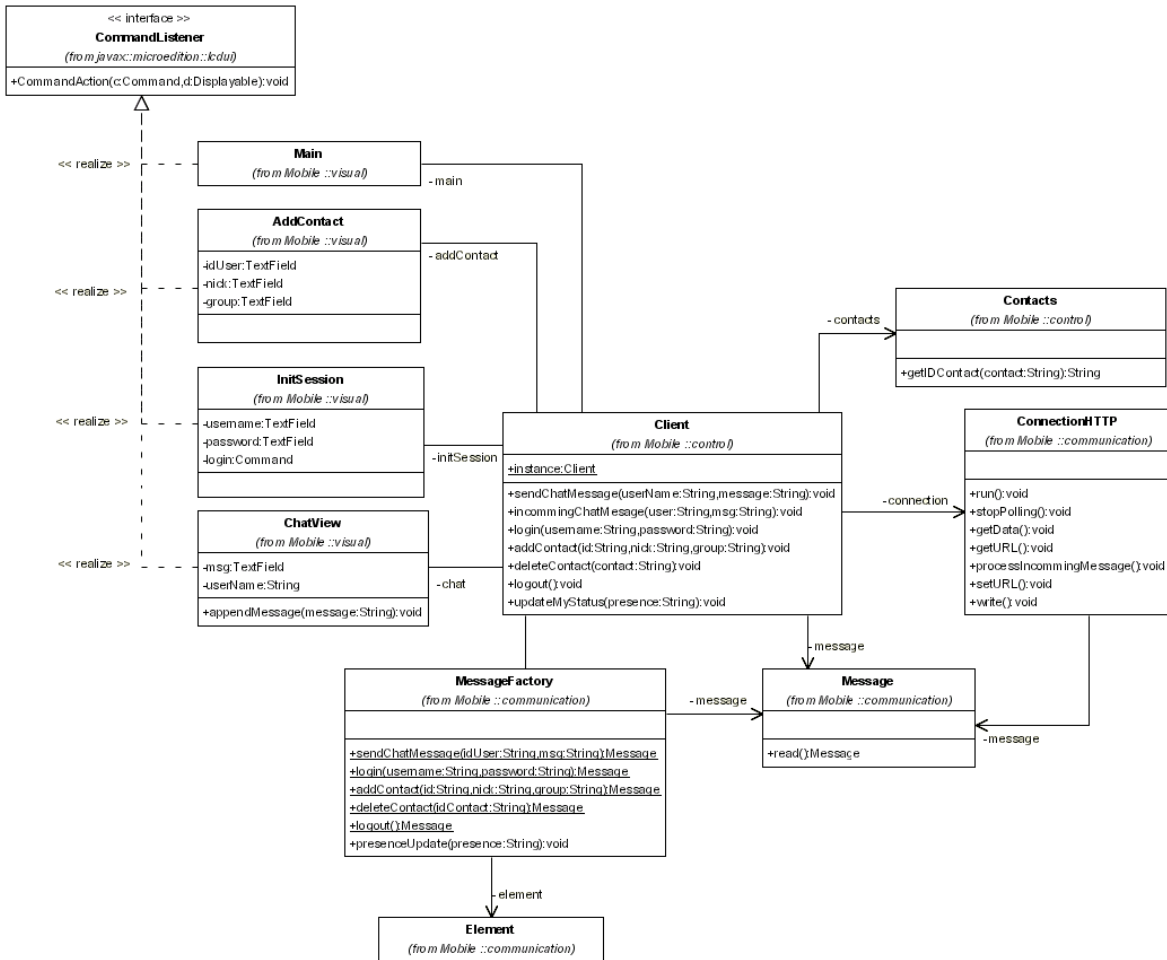


Figura C8. Diagrama de Clases del Cliente Móvil MIDP 1.0



C2.2.3 Vista del Servidor

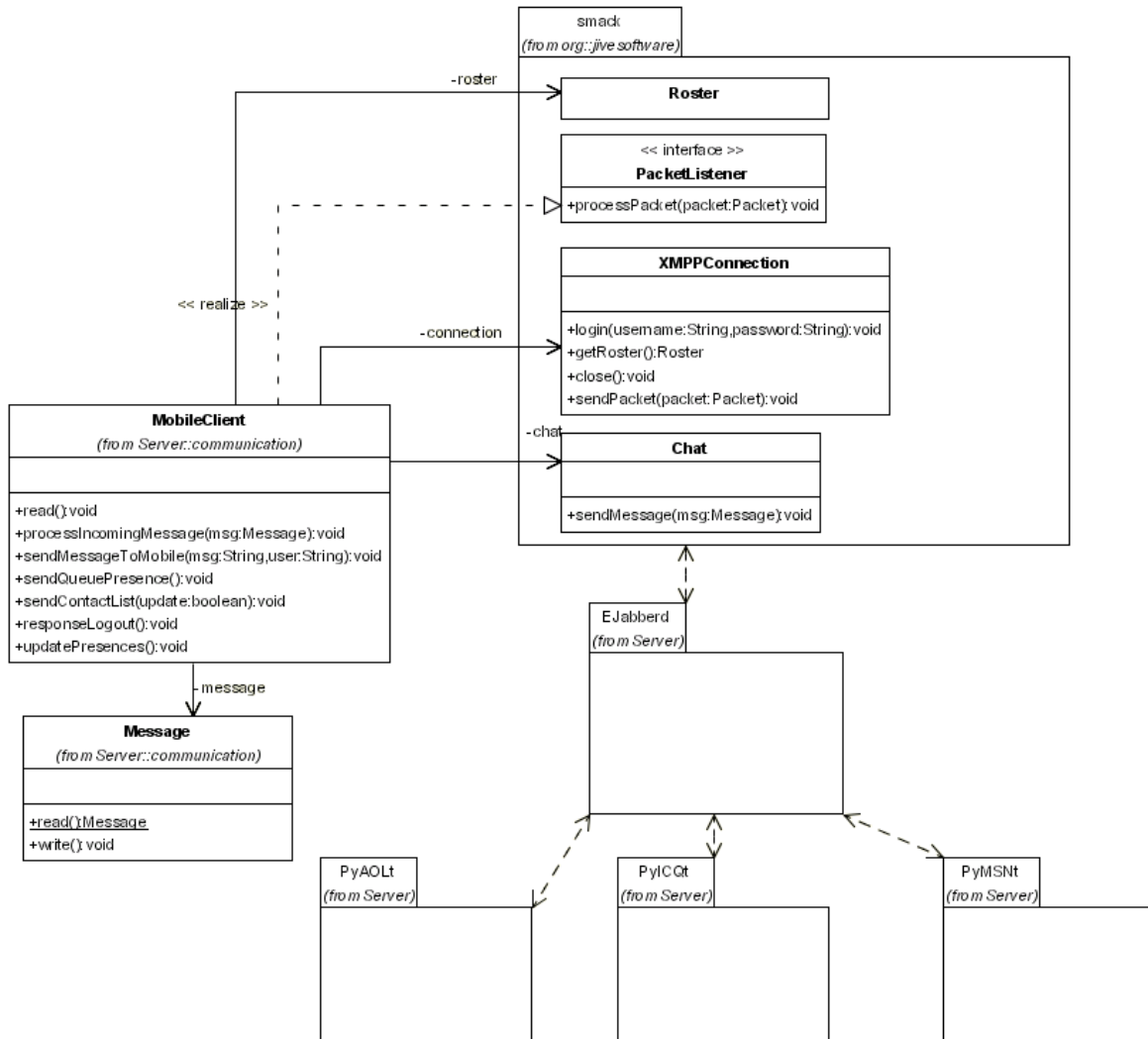


Figura C9. Diagrama de Clases del Servidor

C3 Modelo Dinámico del Diseño

C3.1 Adicionar Contacto

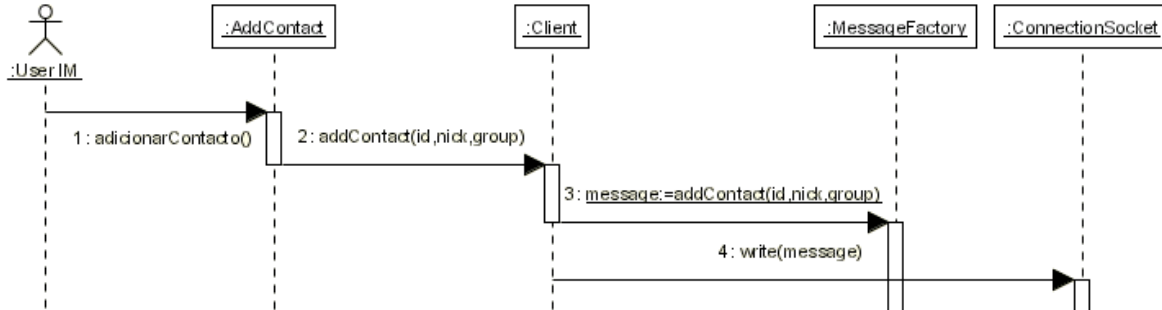


Figura C10. Diagrama de Secuencia, Adicionar Contacto, Cliente Móvil

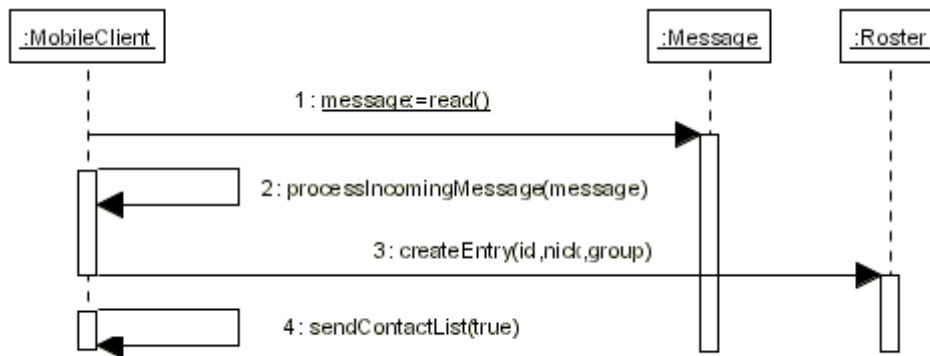


Figura C11. Diagrama de Secuencia, Adicionar Contacto, Servidor

C3.2 Cambiar Presencia

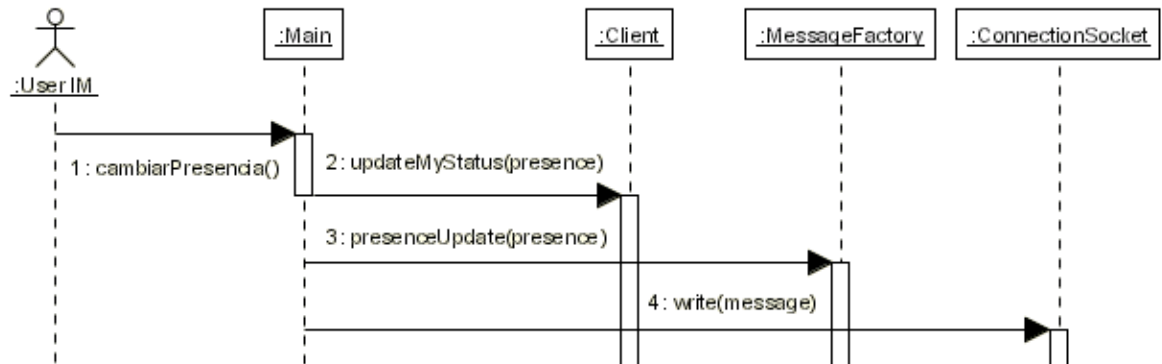


Figura C12. Diagrama de Secuencia, Cambiar Presencia, Cliente Móvil

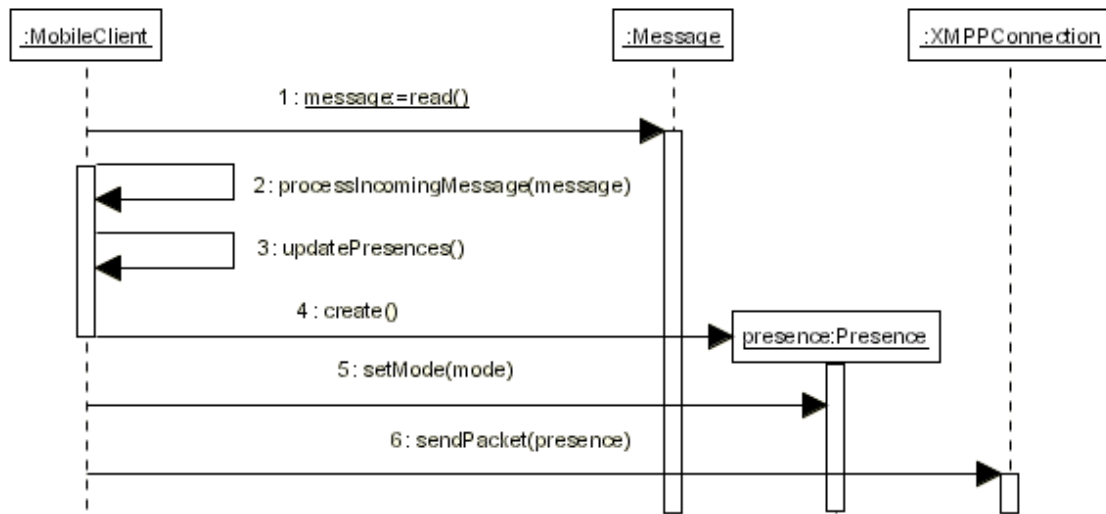


Figura C13. Diagrama de Secuencia, Cambiar Presencia, Servidor

C3.3 Cerrar Sesión

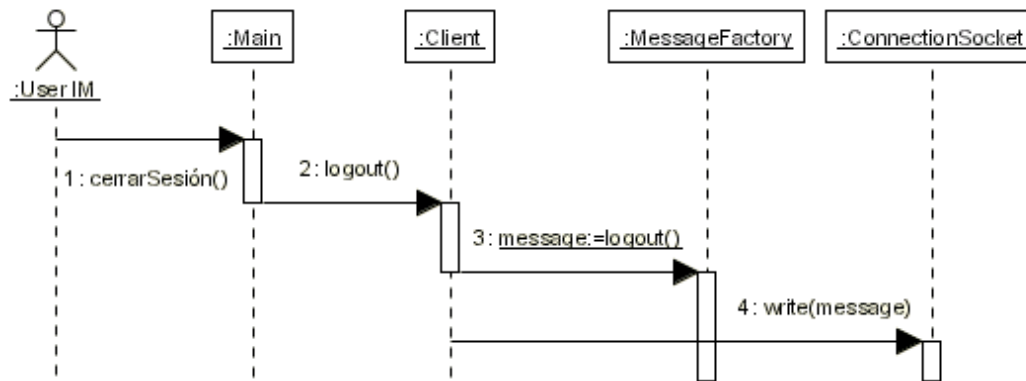


Figura C14. Diagrama de Secuencia, Cerrar Sesión, Cliente Móvil

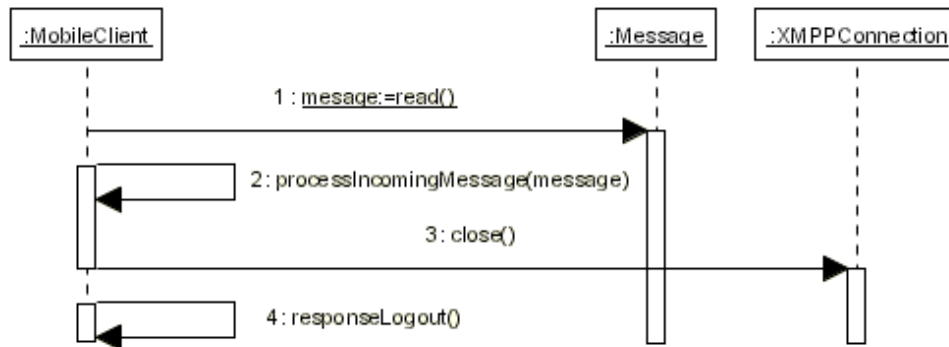


Figura C15. Diagrama de Secuencia, Cerrar Sesión, Servidor

C3.4 Eliminar Contacto

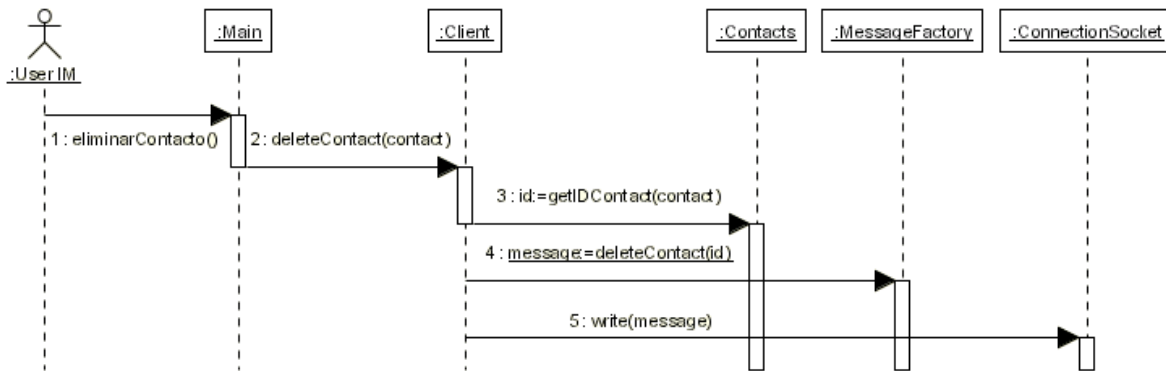


Figura C16. Diagrama de Secuencia, Eliminar Contacto, Cliente Móvil

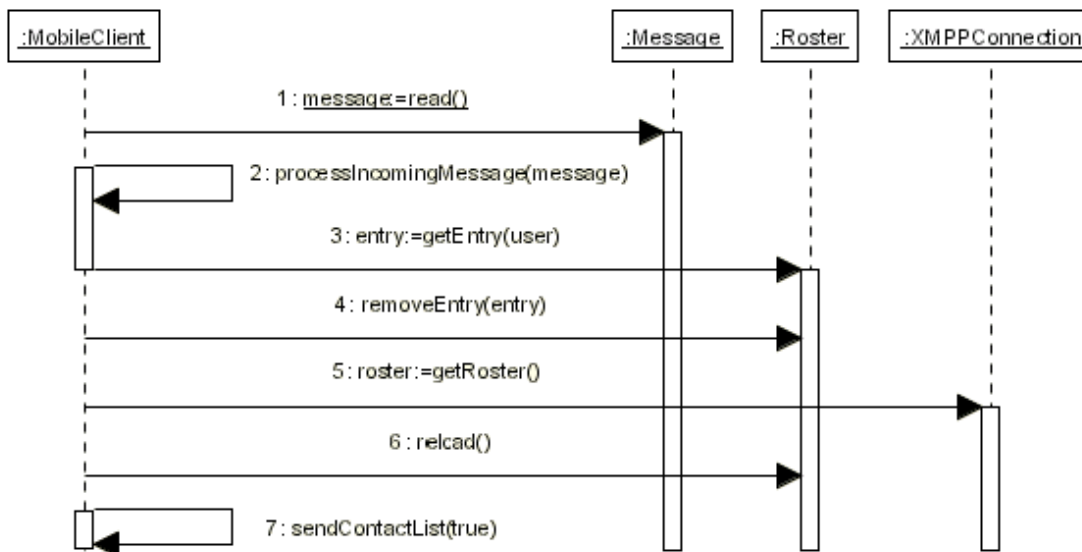


Figura C17. Diagrama de Secuencia, Eliminar Contacto, Servidor

C3.5 Iniciar Chat

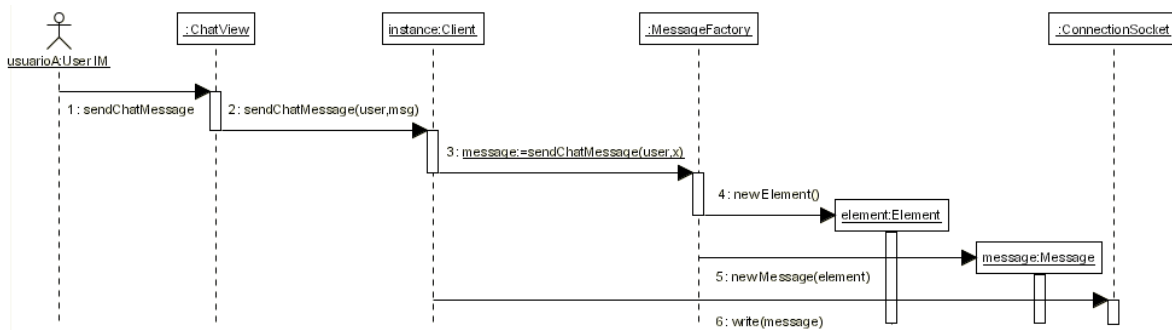


Figura C18. Diagrama de Secuencia, Iniciar Chat, Cliente Móvil Emisor

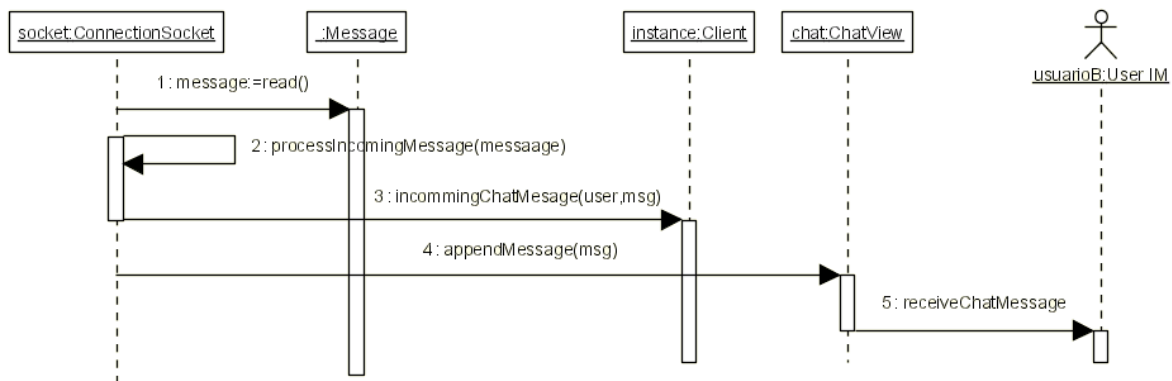


Figura C19. Diagrama de Secuencia, Iniciar Chat, Cliente Móvil Receptor

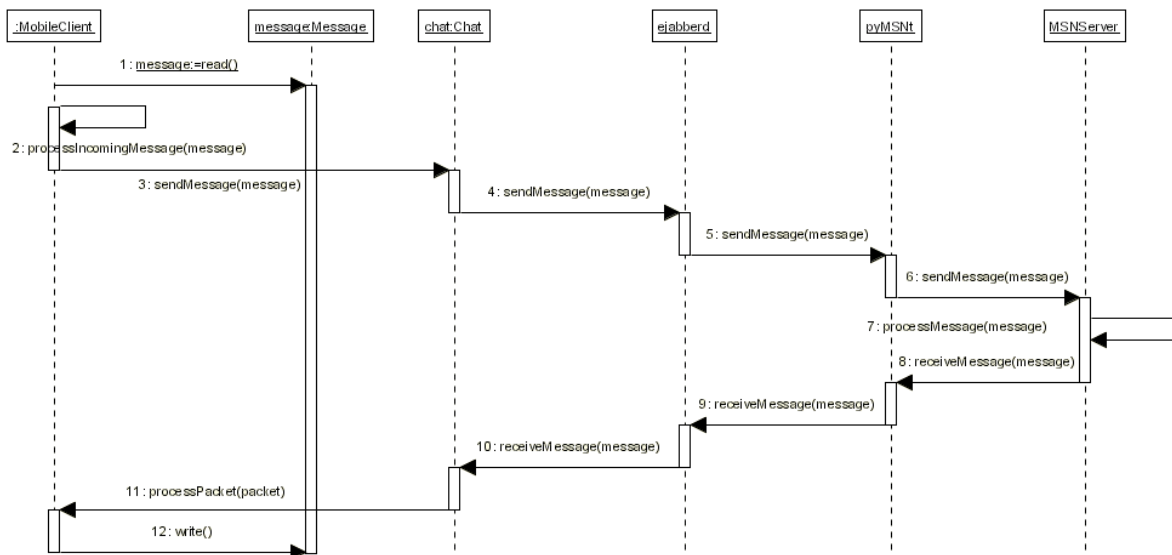


Figura C20. Diagrama de Secuencia, Iniciar Chat, Servidor



C3.6 Iniciar Sesión

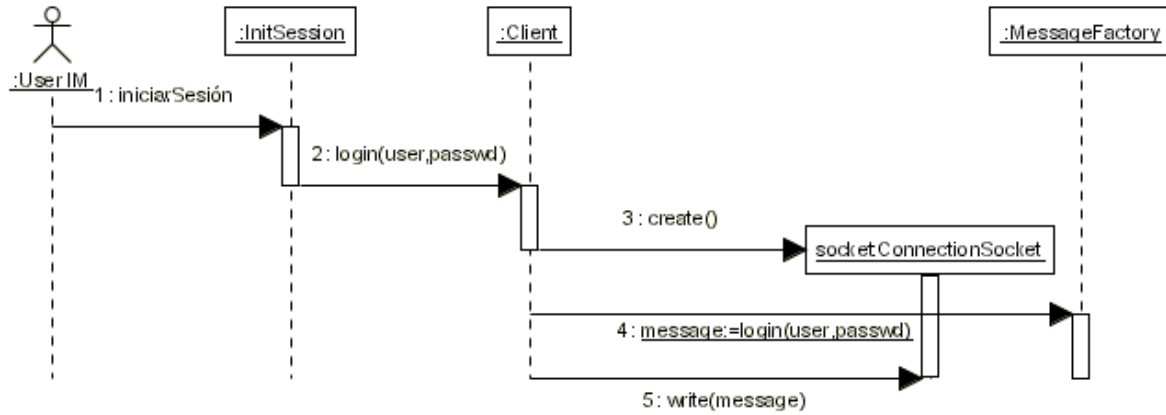


Figura C21. Diagrama de Secuencia, Iniciar Sesión, Cliente Móvil

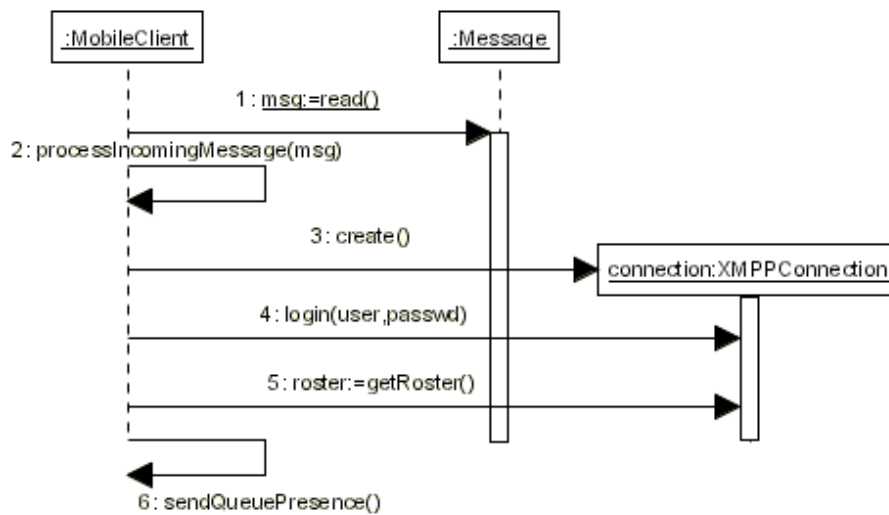


Figura C22. Diagrama de Secuencia, Iniciar Sesión, Servidor



C4 Diagrama Entidad Relación

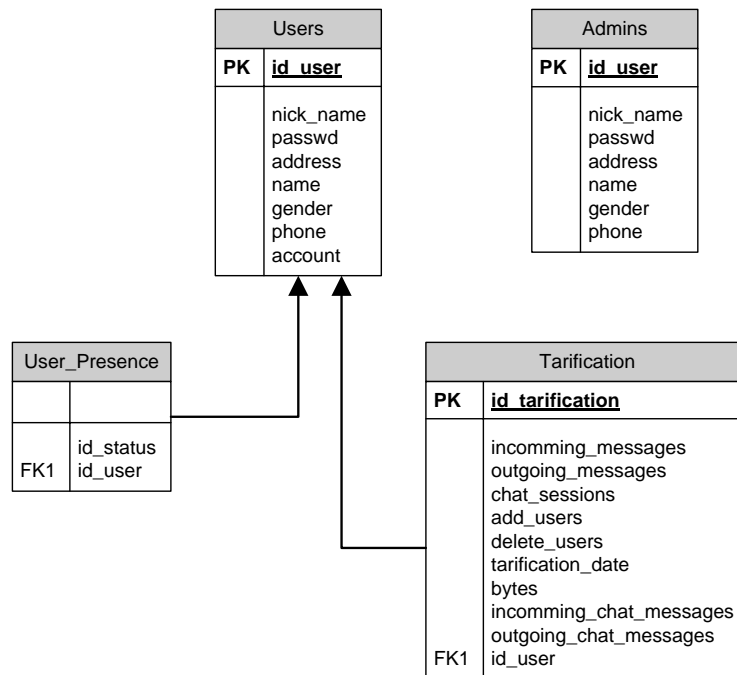


Figura C23. Diagrama de Entidad Relación

C5 Diagrama de Implantación

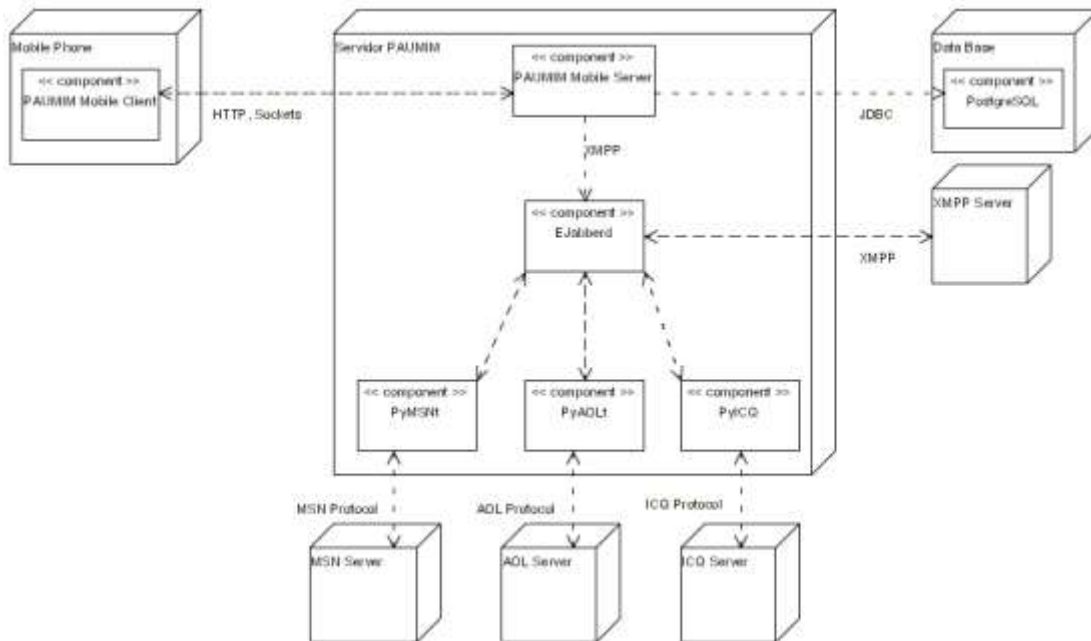


Figura C24. Diagrama de implantación del sistema

C6 Internacionalización de la Aplicación

De acuerdo con uno de los múltiples estudios realizados en revistas especializadas en telefonía móvil, se espera que los abonados a operadores móviles en periodo de dos a tres años sea de un billón de personas, con el 70% de ellos en Asia y Europa [1]. Es por eso que la internacionalización es una parte esencial del desarrollo de las aplicaciones para dispositivos móviles. Esta se define como el proceso de software que permite que una aplicación pueda ser adaptada fácilmente a múltiples lenguajes y regiones sin realizar cambios en la ingeniería del sistema [2].

Para la implementación dentro del cliente PAUMIM se utilizan clases Java que contienen los recursos adecuados para cada lenguaje, las clases se compilan y se adicionan al contenido del archivo JAR de la Suite de MIDlets; y en tiempo de ejecución, los recursos de cada lenguaje se recuperan instanciando la clase correspondiente. Este diseño sigue el esquema de internacionalización que desarrolla J2SE [3] mediante las clases ResourceBunndle [4] y ListResourceBundle [5]. Pero por cuestiones de limitación de memoria y economía de recursos del dispositivo móvil, se ha hecho una simplificación en



sólo una clase abstracta `ResourceBundle` de la cual heredan clases especializadas en cada lenguaje, como se ve en la figura C25.

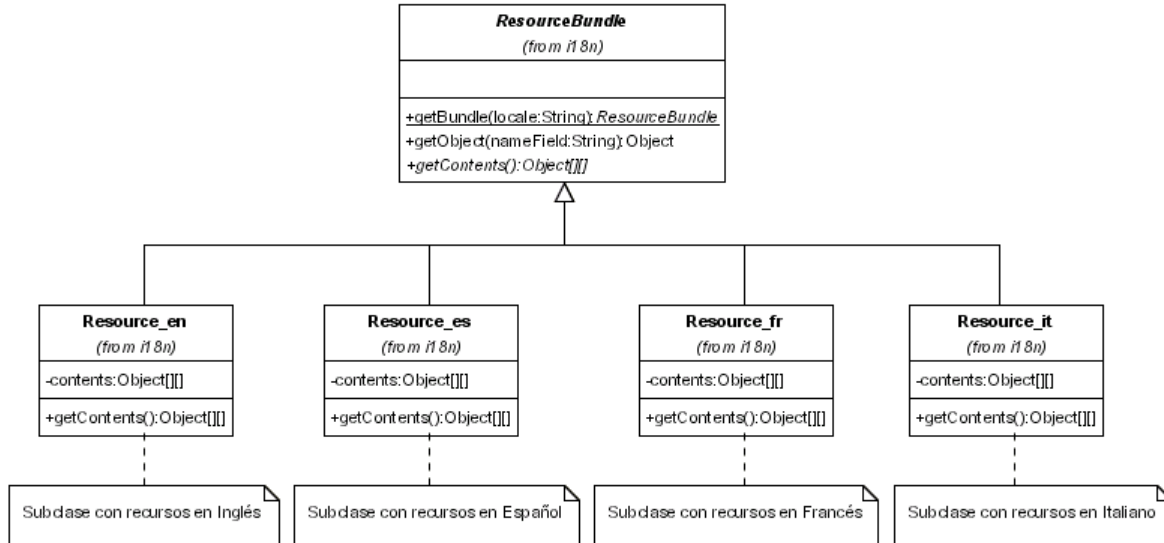


Figura C25. Diagrama de clases para la implementación de la internacionalización

[1] **Java 2 Microedition Manual de Usuario y Tutorial.** Agustín Froufe y Patricia Jorge. Alfa Omega Grupo Editor. 2004

[2] **Core Java Internationalization.** Sun Developer Network. Sun Microsystems Inc. <http://java.sun.com/j2se/corejava/intl/index.jsp>

[3] **Java 2 SDK Internationalization Overview.** Sun Microsystems Inc. <http://java.sun.com/j2se/1.3/docs/guide/intl/intl.doc.html>

[4] **Class ResourceBundle.** Java 2 Platform Developer Documentation. Sun Microsystems Inc. <http://java.sun.com/j2se/1.3/docs/api/java/util/ResourceBundle.html>

[5] **Class ListResourceBundle.** Java 2 Platform Developer Documentation. Sun Microsystems Inc. <http://java.sun.com/j2se/1.3/docs/api/java/util/ListResourceBundle.html>