

Capítulo I

Introducción General

1. Motivación y objetivos

El imparable avance de las tecnologías de sistemas de información y las comunicaciones está haciendo posible cada vez más que éstas se apliquen a campos donde hasta hace unos años era una idea simplemente impensable. Las posibilidades que se nos presentan hoy día, por ejemplo con la evolución de los servicios móviles, son innumerables, permitiendo el desarrollo de aplicaciones con múltiples funcionalidades y en muy diversos ámbitos. Uno de estos ámbitos en los que las tecnologías de la información y las comunicaciones han experimentado un crecimiento considerable es la medicina, permitiendo el desarrollo de nuevos y más completos servicios de telemedicina. Básicamente, la telemedicina consiste en aplicar la medicina a distancia, empleando para ello las tecnologías anteriormente mencionadas. Dentro de esta disciplina podremos encuadrar también a la teleasistencia, entendiendo como tal la prestación de servicios sanitarios, de auxilio, etc. con el objetivo de mejorar la calidad de vida de las personas a quien va dirigido, generalmente personas vulnerables, ya sea por enfermedad, discapacidad, edad avanzada, etc. Esta conjunción, la de aplicar tecnologías en claro crecimiento y con un futuro prometedor, junto a la de ofrecer un servicio muy útil para un conjunto de personas muy amplio y que les puede suponer una mejora considerable en su estilo de vida, es lo que me hizo decantarme por la realización de este tipo de proyecto.

El objetivo primordial, por tanto, que se persigue con este trabajo, es el intento de aunar las tecnologías más modernas de telefonía móvil con la idea de ofrecer un servicio de asistencia a personas cuyas capacidades físicas y/o mentales se encuentran notablemente mermadas. Es por ello que se pensó en realizar un servicio de teleasistencia destinado a un amplio abanico de personas, desde personas mayores en general, hasta por ejemplo, personas enfermas de alzheimer, en las que la pérdida de memoria y las dificultades de orientación son unos de los principales problemas.

El servicio que se propone en este proyecto difiere en su forma respecto a los servicios tradicionales de teleasistencia. Normalmente, este tipo de ofertas se han centrado en la atención a las personas permitiendo la permanencia en sus hogares sin la necesidad de que estuvieran permanente vigilados, ofreciéndoles una vía de comunicación fácil con sus cuidadores. Estos servicios normalmente son también de compañía para estas personas, estableciendo con ellas comunicaciones periódicas para comprobar su estado y por ejemplo, para recordarles las medicinas que deben tomar. La implementación más normal de este tipo de sistemas consiste en un terminal conectado a la red telefónica que a su vez establece comunicación con otro dispositivo, mucho más pequeño, que es el que tiene que llevar consigo el paciente. Con este pequeño sistema el paciente tiene atención inmediata en caso de emergencia, pero sólo en el ámbito de su casa. Esto está muy bien si nuestro paciente tiene movilidad reducida, pero si no es así, si es capaz de moverse por sí mismo fuera de su casa, es necesario algo más, pues se hace necesario conocer la posición del mismo en todo momento.

Es en este punto donde el presente proyecto busca aportar ideas, soluciones, modernas desde el punto de vista tecnológico y asequibles desde el punto de vista económico. Por ambas razones, se optó por realizar un servicio de teleasistencia sobre redes móviles de tercera generación utilizando el estándar de OSA/Parlay. Se eligió la tecnología UMTS porque es la más moderna y porque ofrece una serie de funcionalidades y capacidades que la red GSM no contempla, si bien es verdad, que al emplear la interfaz OSA/Parlay estamos independizando lo que es nuestra aplicación de la red que haya por debajo, es decir, para nosotros la red será un conjunto de ‘cajas’ y cada una de ellas nos ofrecerá una serie de funcionalidades. Nuestra aplicación tendrá que usar aquellas que le sean útiles para el fin perseguido, obtener de ellas la

información que creamos conveniente, procesarla y tomar decisiones cuando sea necesario.

La opción de crear una aplicación utilizando telefonía móvil se basa también en que nos interesaba la idea de que la implementación del sistema fuera lo más sencilla posible y además, como se ha comentado antes, no sólo abarcara la residencia de nuestros pacientes. La forma más sencilla, a nuestro juicio, es aprovechar las redes móviles existentes, lo que implica que los pacientes sólo necesitarían llevar con ellos su teléfono móvil, para por ejemplo, saber en cada momento cuál es su posición, y en caso de emergencia, contactar con ellos.

Otro aspecto que interesaba mucho era la opción de poder testear la aplicación, por lo que, una vez decidido el tema a realizar y antes de delimitar más el alcance y profundidad del mismo, se decidió buscar algún tipo de software o simulador que lo permitiera. La verdad es que apenas se encontraron un par de herramientas que permitieran probar una aplicación de este tipo, y finalmente se escogió el software *MiLife™ ISG SDK 7.1* de Lucent Technologies, por ser el más completo y el que más se adecuaba a lo que se buscaba. Esta herramienta, como se explicará en el punto 3 de la Introducción Teórica, nos ofrece, por un lado, el conjunto de funcionalidades antes mencionadas, es decir, un conjunto de capacidades de servicio abstraídas de la red que se comunican a través de una interfaz OSA/Parlay, y por otro, un entorno de desarrollo de aplicaciones más simple junto a un simulador para poder testear dichas aplicaciones. En conclusión, esta herramienta nos ofrecía un entorno de simulación completo para probar la aplicación.

Una consecuencia directa de querer probar la aplicación es que es indispensable el uso del simulador, y por lo tanto, adecuarse a las características del mismo. Eso implica que en el código creado haya sido inevitable la mezcla de código destinado a la aplicación en sí junto a código cuya única funcionalidad es la de interactuar con el simulador para permitir de esa manera la simulación, y poder crear así una situación de interactividad que muestre en la mayor medida de lo posible la funcionalidad real de la aplicación. Esa mezcla de código provoca que la aplicación creada no sea una versión real, pues gran parte del mismo se ha tenido que crear para recrear situaciones y

actividades que en la realidad no las realizaría la aplicación, sino los pacientes o demás elementos reales que compongan el conjunto del servicio.

Otra consecuencia de utilizar el software antes mencionado es que se presentaba la posibilidad de programar utilizando una API en Java (las *Convenience Classes*) suministrada por él mismo, en vez de usar la API de OSA/Parlay directamente. Por lo pronto, dicha API había que utilizarla en la comunicación del programa con el simulador. Se estudiaron las dos posibilidades, y se eligió la suministrada por *MiLife™ ISG SDK*, pues esta API suponía utilizar un nivel de abstracción superior y ocultaba muchas de las dificultades en cuanto programación de la API de OSA/Parlay y en cuanto a la complejidad de la estructura de los datos a utilizar.

Una vez definido el entorno de desarrollo a utilizar se empezó a definir de una manera más concreta el servicio de teleasistencia. La funcionalidad principal que se buscaba, como se ha comentado anteriormente, era la localización permanente de todos los pacientes o abonados al servicio, independientemente del lugar donde se encontraran. Partiendo de esa idea, se fueron añadiendo funcionalidades, intentando utilizar la mayor cantidad de capacidades de servicio. Evidentemente, el uso del simulador ha limitado también muchas de estas ideas iniciales, pues no soportaba algunas de esas características. A continuación se van a enumerar una serie de puntos que describen el funcionamiento del servicio creado:

✚ En la configuración inicial del servicio y creación del entorno se han definido los siguientes actores:

- Los **pacientes**: elementos principales de la aplicación, pues entorno a ellos gira todo el servicio.
- Los **cuidadores**: cada paciente va a tener su cuidador designado. Un mismo cuidador puede estar asignado a varios pacientes. La misión del cuidador no es estar siempre con el paciente, si no una persona que conoce los hábitos de sus pacientes, que los visita periódicamente y es, digamos, la cara visible del servicio de teleasistencia de cara al paciente. Cada cuidador va a disponer también de su terminal móvil.

- La **central**: es el lugar donde se coordina todo el servicio. Puede ser el lugar donde se ejecute la aplicación. Dispone de varios terminales (números de abonado) para realizar las distintas tareas. Está el número principal (denominado central) y que es el que usa la aplicación para realizar las llamadas y envío de mensajes de manera automática y un número asignado a una operadora para dar soporte de atención al cliente. Todas las llamadas entrantes a la central que no correspondan a situaciones de emergencia, se desvían a la operadora.
 - Las **ambulancias**: cada una dispondrá de su terminal móvil para ser requerida en caso de que un paciente la necesite en caso de emergencia.
- ✚ El servicio principal, como ya se ha comentado, es el de localización continua de los pacientes. La aplicación conoce cuál es la posición de referencia de cada uno de ellos, y periódicamente comprueba cuál es su posición (en un intervalo que puede ser configurable, dependiendo de la capacidad de proceso de que se disponga), de tal manera que si en cualquier momento la distancia de dicho paciente con respecto a la de referencia supera el kilómetro, se generará una alarma.
- ✚ En paralelo a la acción anterior, la aplicación es capaz de tratar las alarmas generadas, de tal manera que cuando detecta la creación de una, se procede a su identificación y posterior tratamiento.
- ✚ Como vemos, el servicio depende totalmente de que los terminales de los pacientes tengan un correcto funcionamiento, es decir, estén siempre encendidos y tengan cobertura. Por ello, la aplicación es capaz de detectar cuando algún terminal pasa a un estado no alcanzable. Cuando esto sucede con algún terminal de los pacientes, inmediatamente se avisa mediante un mensaje multimedia al cuidador correspondiente, para que sea éste el que se encargue de revisar el estado del paciente y su terminal. En caso de que no pueda o haya algún tipo de emergencia, será él el encargado de dar la alarma. De la misma manera, cuando un terminal vuelve a un estado alcanzable, se avisa al cuidador.
- ✚ Cuando se genera una alarma por una posición no adecuada del paciente, el protocolo de actuación que se sigue es el siguiente:

1. Se comprueba cuál es la posición del cuidador asociado al paciente, pues si está muy cerca del paciente, es muy probable que se encuentre con él. Se ha decidido que si la distancia entre ambos es menor de 20 metros, suponemos que están juntos. Se procede entonces a enviar un mensaje multimedia al cuidador informando de que la posición de su paciente ha excedido el radio de seguridad y que se considera que él se encuentra acompañándole debido a su proximidad. Si esto no fuera así, el cuidador debe informar de ello. Por último se anotará el estado del paciente y no se comprobará más su posición.
2. Si la distancia entre ambos supera los 20 metros, la central se pone en contacto con el cuidador. Como es un proceso automático, se le reproduce un mensaje en el que se le pregunta si él está con el paciente (o sabe si está acompañado) o si desconoce cuál es su situación.
3. El cuidador responderá (mediante pulsación de teclas). Si el paciente se encuentra acompañado, se anotará que el paciente no está sólo y no se comprobará su posición más. Por el contrario, si el paciente está sólo, se procederá a llamar a una ambulancia que esté disponible y su posición sea la más cercana a él.
4. Cuando el cuidador deje de nuevo sólo al paciente, él, mediante un mensaje multimedia dirigido un terminal específico de la central, activa de nuevo la comprobación de la posición de su paciente.

✚ Se ha contemplado también otro tipo de alarmas, por ejemplo de caídas. Se supone que el terminal del paciente estaría conectado a algún tipo de sensor de caídas, y que, cuando éste le avisara, pues inmediatamente mandaría un mensaje de alarma a un número específico de la central. El tratamiento que se ha hecho de este tipo de alarma es simplemente avisar a la ambulancia más cercana (y disponible) para ir en su auxilio.

Por último comentar que en cuanto al tratamiento de los mensajes y de las alarmas enviadas a través de este método, se pretendía especificar un formato concreto

en el contenido de los mismos, y que fuera la aplicación la que capturase el mensaje y analizara el contenido. Capturar el mensaje sí se ha conseguido, pero ha sido imposible recuperar el contenido debido a que el simulador no soporta esta funcionalidad. Por este motivo, ha sido necesario distinguir los mensajes mediante su dirección de origen y su dirección destino, y tomando mediante esas direcciones las medidas oportunas. Es por eso por lo que existen números específicos en la central para enviar los mensajes por alarmas de caída y para cambiar el estado de un paciente.

Como conclusión a este punto me gustaría añadir que, normalmente, el proceso de creación de este tipo de servicios se ciñe a las peticiones del cliente que lo haya encargado. Evidentemente, aquí no ha existido ningún cliente, por eso los objetivos no han constituido un conjunto de requisitos inamovible. Se ha partido de una idea, la adaptación de un servicio útil a la sociedad a las tecnologías de telefonía móvil de nueva generación, y según avanzaba el tiempo y se comprobaban los medios y la complejidad de la misma, se iba modelando hasta obtener la aplicación que conforma este proyecto. El objetivo principal era pues el de introducirnos en el mundo de la creación de aplicaciones para redes móviles usando una interfaz estándar y abierta, independiente de la red subyacente y del propietario de la misma, es decir, crear una aplicación basada en OSA/Parlay, pues el utilizar esta tecnología supone un hecho diferenciador con respecto a otros servicios del mismo estilo.

2. Ejecución temporal

En la realización del presente proyecto, la fase de documentación ha ocupado un lugar importante. El desconocimiento acerca de la tecnología OSA/Parlay era total, por lo que ha sido necesario tomarse un tiempo para recopilar información sobre el tema. Establecer un límite de tiempo para cada etapa de implementación del proyecto es bastante complicado, pues muchas de ellas se han realizado en diferentes momentos, alternándose unas a otras, pero las principales, y en el orden en que se indican, se pueden resumir en las siguientes:

- ✚ Documentación acerca de los servicios móviles de tercera generación y la tecnología OSA/Parlay.
- ✚ Búsqueda de un software que permitiera probar una aplicación que usara OSA/Parlay. El software elegido fue, como ya se ha comentado, *MiLife™ ISG SDK 7.1* de Lucent Technologies.
- ✚ Familiarización con el software anterior, comprobando las posibilidades que ofrecía y cuáles no, y refrescando el lenguaje XML para la creación del entorno de simulación que necesitaba el software.
- ✚ Desarrollo de la base de datos y de la aplicación Java que usa la API suministrada por el software *MiLife*.
- ✚ Redacción de la memoria del proyecto.

En la representación de la página siguiente (figura 1) se aprecia un diagrama de Gantt en el que quedan reflejadas todas las etapas antes mencionadas junto a su duración aproximada.

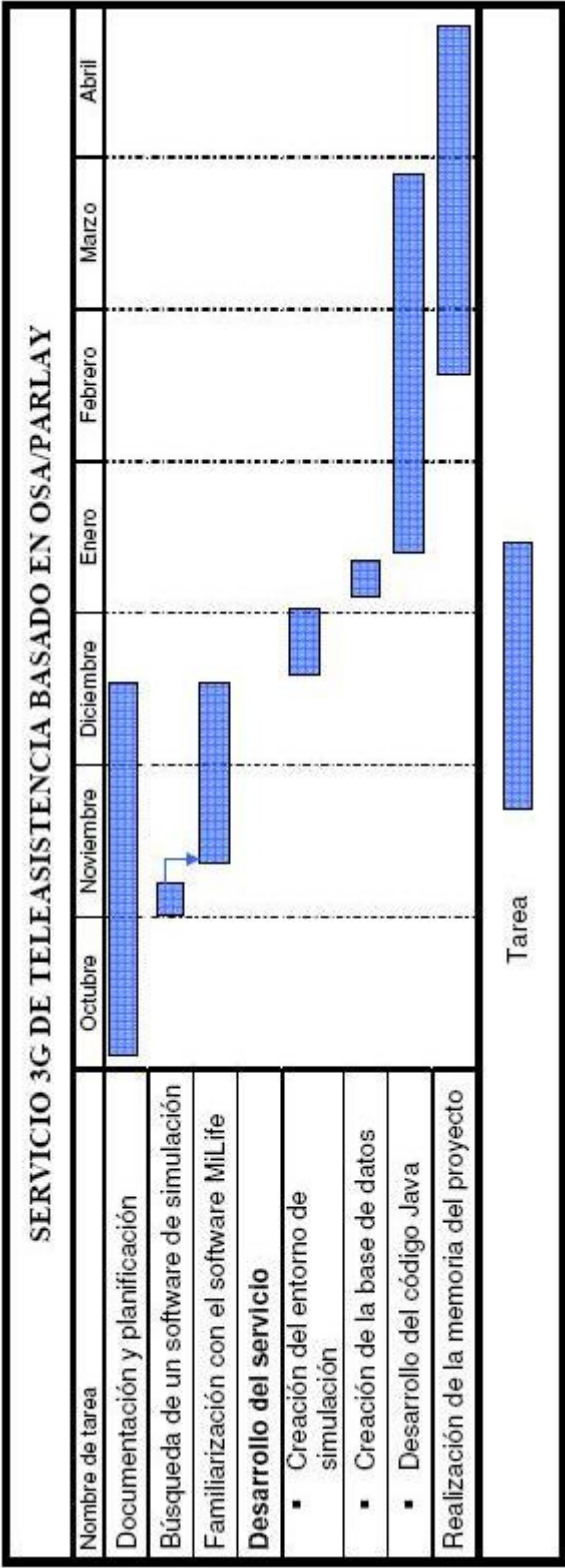


Figura 1. Diagrama de Gantt

3. Material empleado

En este punto se va a enumerar el conjunto de programas utilizado en la realización de este proyecto. Se ha intentado en todo momento utilizar software gratuito, de libre distribución, y en algunos casos, software libre.

Como ya se ha comentado anteriormente, para crear un entorno de simulación apropiado para testear la aplicación se ha empleado el software *MiLife™ ISG SDK 7.1* de Lucent Technologies. No se va a describir nada más sobre este producto en este punto porque se le dedicará otro en exclusiva más adelante. Para configurar el simulador del software anterior y cargarlo de información para la simulación ha sido necesario trabajar con archivos XML. Para su tratamiento se ha utilizado un simple editor de texto muy completo, que soporta multitud de lenguajes: el **Notepad++ v. 3.9**.

Además de la información suministrada al simulador mediante los archivos XML, se ha creado una base de datos que almacena toda la información que necesita el servicio. Las características de la misma se describirán con más detalle en el punto III de la memoria, que describe todo el trabajo realizado. Para la creación de la base de datos y su posterior tratamiento se ha utilizado **MySQL**. Para ello se instaló **MySQL Server 5.0** en el sistema. Además, se prefería utilizar algún tipo de herramienta gráfica para la manipulación de la base de datos, por entenderse que es mucho más cómodo y rápido,

así que también se instaló **MySQL Tools for 5.0**. Todo este software es suministrado de forma gratuita por *MySQL AB*.

En cuanto a la programación del código, como al final se utilizó la API Java suministrada por el software MiLife, se utilizó el **SDK Eclipse 3.2.1**, por facilitar enormemente este proceso. Para ello se importaron una serie de librerías necesarias para utilizar dicha API y que también suministra MiLife. Por último, para obtener las clases y los distintos diagramas en UML se utilizó el plugin gratuito para eclipse **OMONDO**.

4. Contenido de la memoria

Toda la documentación que se presenta en este documento está dividida en los capítulos que se enumeran y explican a continuación:

- ✚ **Capítulo I: Introducción general.** Este capítulo servirá para exponer tanto la motivación que ha llevado a la realización del presente proyecto como los objetivos que se pretendían alcanzar. Se explicará además el material empleado así como la planificación temporal.

- ✚ **Capítulo II: Introducción teórica.** En este apartado se va a desarrollar toda la base teórica que soporta la implementación de este proyecto. Se comenzará con un pequeño resumen sobre los servicios móviles de tercera generación, exponiendo las posibilidades de los mismos y las aportaciones respecto a tecnologías anteriores. Posteriormente se explicará ampliamente en qué consiste la tecnología OSA/Parlay y qué ventajas se obtienen de su uso. Y para finalizar se analizará con profundidad el software *MiLife™ ISG SDK 7.1*, describiendo todas sus partes y cómo se han utilizado en el presente proyecto.

- ✚ **Capítulo III: Desarrollo del proyecto.** En este punto se describirá todo el trabajo realizado. En primer lugar la base de datos implementada; posteriormente se explicará la configuración realizada en el simulador y la

creación del entorno de simulación, y por último se expondrán todas las clases Java creadas a partir de sus representaciones en UML, comentando sus características principales.

✚ **Capítulo IV: Conclusiones y líneas futuras.**

✚ **Capítulo V: Referencias bibliográficas.**

✚ **Capítulos VI: Anexos.** Se incluye en este punto todo el código realizado: el archivo XML necesario para la simulación, todo el código Java y el Javadoc.

