

Pontificia Universidad Católica de Valparaíso
Facultad de Ingeniería
Escuela de Ingeniería Informática

**DESARROLLO DE APLICACIÓN MÓVIL DE
TELEASISTENCIA PARA ADULTOS MAYORES**

DANIEL ELIAS TOLEDO MARTINEZ

INFORME FINAL DEL PROYECTO
PARA OPTAR AL TÍTULO PROFESIONAL DE
INGENIERO DE EJECUCIÓN EN INFORMÁTICA

Julio 2013

Pontificia Universidad Católica de Valparaíso
Facultad de Ingeniería
Escuela de Ingeniería Informática

DESARROLLO DE APLICACIÓN MÓVIL DE TELEASISTENCIA PARA ADULTOS MAYORES

Daniel Elías Toledo Martínez

Profesor Guía: **José Miguel Rubio León.**

Profesor Co-referente: **Aldo Migliaro Osorio.**

Carrera: **Ingeniería Ejecución Informática**

Julio 2013

*Dedico este trabajo a mi familia especialmente a mis padres
por ser el pilar fundamental en todo lo que soy y
por todo su apoyo durante este largo proceso*

*Agradezco profundamente todo el apoyo brindado por mi familia,
mis padres, amigos, compañeros, a Yarelyn por estar siempre
en los momentos difíciles y también a mi profesor guía José Miguel,
sin su apoyo este trabajo no hubiese sido posible.
Gracias a todos*

Resumen.

Android en la actualidad es una de las tecnologías más revolucionarias e innovadoras del último tiempo. Este sistema operativo nació de las necesidades de comodidad y portabilidad que la sociedad ha solicitado y seguirá haciéndolo. Inicialmente su funcionalidad abarcaba solamente algunas características básicas como llamar, grabar, guardar contactos, y estaba disponible sólo en algunos dispositivos móviles de gama baja. Actualmente, abarca diversos tipos de dispositivos móviles, e incluso desde lavadoras hasta notebooks

Considerando lo anterior, este trabajo de título consiste en el desarrollo de una aplicación de tele asistencia que permite monitorear y cuidar a un adulto mayor a través de los distintos componentes de hardware del smartphone, en este caso se utilizó el acelerómetro, el cual, al detectar ciertos patrones de movimientos notificará al usuario final la localización geográfica del adulto mayor.

También permite generar ciertos recordatorios, que estimulan la memoria cognitiva del adulto mayor, esto permite reorganizar las tareas diarias como por ejemplo, tomar medicamentos, comer, caminar, etc.

Abstract.

Android is currently one of the most revolutionary and innovative technologies of the last time. This operating system was born out of the needs of portability and convenience that the company has sought and will continue its functionality initially covered only some basic features like calling, recording, storing contacts, and was available only in some low-end mobile devices. Currently, encompassing various types of mobile devices, and even from washing machines to notebooks

Considering the above, this is the reason why telecare application was created. This application allows us to monitor and look after an old person through the different smart phone hardware components. In this case the accelerometer was used, which when it detects certain patterns of movement will inform to the final user about the geographical localization of the old person.

It allows too generating certain reminders that stimulate the cognitive memory of the elderly; this allows them to reorganize daily tasks such as taking medication, eating, walking, etc.

Índice.

Resumen.	i
Abstract.....	i
Índice.	ii
Lista de Figuras.	iv
Lista de Tablas.....	v
Capítulo I: Introducción.	1
1.1 Motivación.....	1
1.2 Objetivo General.....	1
1.3 Objetivos Específicos.	2
1.4 Metodología.....	2
1.5 Plan de Trabajo.....	2
1.6 Plan de trabajo tentativo.	4
Capítulo II: Dispositivos móviles.....	5
2.1 ¿Qué es un Dispositivo Móvil?	5
2.2 Características de un Dispositivo Móvil.....	5
2.3 Sistemas Operativos para Dispositivos Móviles.	6
2.4 Aprendizaje y dispositivos móviles.....	7
2.5 ¿Por qué Android?.....	7
Capítulo III Especificación de Requerimientos.....	9
3.1 Descripción del problema.....	9
3.1.1 Solución propuesta.	9
3.1.2 C2DM.....	12
3.1.3 Análisis de requerimientos.	12
3.1.4 Requerimientos Funcionales.	12
3.1.5 Requerimientos No Funcionales.....	14
Capítulo IV Diseño de la Solución.....	15
4.1 Arquitectura.....	15
4.2 Diagrama de Clases.....	17
4.3 Modelo Relacional.....	18
4.4 Casos de uso.	18
4.5 Diagrama de Secuencia.	22
4.6 Diseño de Interfaz (Prototipo 1).....	24
4.7 Diseño de Interfaz Móvil (Prototipo 2).....	27

4.8 Diseño de Interfaz Web.....	31
Capítulo V Plan de Pruebas.....	32
5.1 Documento de Casos de Pruebas.....	32
5.2 Plan de Pruebas.....	34
5.3 Calendario de Pruebas.....	35
Capítulo VI Conclusiones y Trabajo Futuro.....	36
Anexos.....	38
Anexo A.....	38
A.1 Introducción a Dispositivos Móviles.....	38
A.2 Telefonía móvil.....	38
A.2.1 CelularesConceptoGeneral.....	38
A.2.2 Historia.....	39
A.3 Sistemas Operativos Móviles.....	40
A.3.1Capas.....	40
A.3.2Mercado.....	41
Anexo B.....	41
B.1 Factibilidad técnica.....	41
B.1.2 Hardware.....	41
B.1.2.1 Requerimientos de hardware para Eclipse.....	42
B.1.3 Software.....	42
B.1.3.3Java.....	43
B.2 Factibilidad técnica.....	44
B.3 Factibilidad legal.....	44
B.4 Análisis de Riesgo.....	44
B.4.1 Orden de prioridad riesgos más importantes.....	44
Referencias.....	47

Lista de Figuras.

Figura 2. 1 Grafico Evolución Android.....	8
Figura 3. 2Formula de Aceleración	10
Figura 3. 3 Grafico de Aceleración v/s Tiempo	11
Figura 3. 4 C2DM	12
Figura 4. 1 Arquitectura Lógica del sistema.....	16
Figura 4. 2 Diagrama de clases.....	17
Figura 4. 3 Modelo Relacional	18
Figura 4. 4 Caso de uso general.....	19
Figura 4. 5 Caso de uso Gestionar Datos.....	20
Figura 4. 6 Caso de uso Gestionar Tiempo	21
Figura 4. 7 Diagrama de secuencia Gestionar Cuidado.....	22
Figura 4. 8 Diagrama de secuencia Listar características.....	23
Figura 4. 9 Diseño de Login y Menú.....	24
Figura 4. 10 Configurar Notificación y Administrar Monitoreo	25
Figura 4. 11 Interfaz Perfil de Usuario y Interfaz Historial Monitoreo.....	26
Figura 4. 12 Figura de Interfaz de Notificación y Alarma de medicamento	26
Figura 4. 13 Login y Menú.....	27
Figura 4. 14 Ingresar datos de cuidado.....	28
Figura 4. 15 Ver Historial.....	29
Figura 4. 16 Notificaciones e Inicio de Monitoreo.....	30
Figura 4. 17 Login y Listar/Agregar Abuelo.....	31
Figura 4. 18 Listar y Agregar enfermedades abuelo	31

Lista de Tablas.

Tabla 1. 1 Plan Tentativo.....	4
Tabla 3. 1 Requerimientos Funcionales Aplicación Web	13
Tabla 3. 2 Requerimientos Funcionales Aplicación Móvil	13
Tabla 4. 4Narrativo Extendido Gestionar Tiempo	21
Tabla 4. 1 Narrativo Extendido de Gestionar Datos.....	20
Tabla 5. 1 Caso de Prueba 001	33
Tabla 5. 2 Caso de Prueba 002	33
Tabla 5. 3 Caso de Prueba 003	34
Tabla 5. 4 - Calendario de Pruebas.....	35
Anexo B. 1 Tabla de análisis de riegos y grado de exposición	45
Anexo B. 2 Plan de mitigación de riesgos.....	46

Capítulo I: Introducción.

1.1 Motivación.

En el año 2009 se libera el SDK de Android Cupcake [5] esto permitió la libre creación de aplicaciones para los usuarios de Android, y por ende se abre la gran posibilidad de suplir todas las necesidades que se tengan, también de empezar a echar a volar la imaginación y crear cualquier tipo de aplicación. “Android tiene la capacidad de estar en cientos de millones de dispositivos móviles en más de 190 países de todo el mundo. Es la mayor base instalada de cualquier plataforma móvil y de rápido crecimiento, todos los días otro millón de usuarios encienden sus dispositivos Android por primera vez y empiezan a buscar aplicaciones, juegos y otros contenidos digitales.”[5].

Pero el uso de Android va más allá de crear aplicaciones de juegos y/o para suplir necesidades de ocio, sino es un gran aporte para el desarrollo de nuevas funciones para los dispositivos móviles.

Tanto el nombre *Android* (androide en español) como Nexus One hacen alusión a la novela de Philip K. Dick *¿Sueñan los androides con ovejas eléctricas?*, que posteriormente fue adaptada al cine como *Blade Runner*. Tanto el libro como la película se centran en un grupo de androides llamados *replicantes* del modelo Nexus-6. [11]

Este proyecto principalmente está enfocado en la creación de una aplicación móvil la cuál permita monitorear a un adulto mayor en su quehacer diario, la aplicación ocupará un hardware del Smartphone llamado acelerómetro, este dispositivo es un microchip que permite detectar la orientación del teléfono. Gracias a este componente no solo se puede saber la orientación del teléfono, sino también determinar golpes, caídas, momentos de inactividad e incluso gracias a este hardware se podrían crear modelos estadísticos para la rehabilitación de niños con discapacidad motriz, etc.

1.2 Objetivo General.

El principal objetivo de este proyecto es realizar una aplicación para Smartphone con sistema operativo Android, enfocado para la asistencia del adulto mayor en el quehacer de sus tareas cotidianas.

1.3 Objetivos Específicos.

Una vez definido el objetivo general del estudio, éste se descompone de los siguientes objetivos específicos:

- ❖ Aplicación Android que sea compatible con versión 2.2 froyo.
- ❖ Establecer un módulo web que envíe los datos desde el PC al dispositivo a través de webservice.
- ❖ Preparar la aplicación a través de un módulo para la recepción de datos que envía el emisor (webservice).
- ❖ Unión de ambas aplicaciones.
- ❖ Pruebas y validaciones de la aplicación realizada.
- ❖ Integración a App de Google.
- ❖ Generar un Módulo que permita la mensajería instantánea.
- ❖ Generar un Módulo que permita la relocalización.

1.4 Metodología.

Una de las determinaciones más importantes para el desarrollo de este proyecto es la elección de la metodología de modelos del proceso de software, mediante este modelo se establecerá las actividades necesarias para llevar a cabo el cumplimiento de los requerimientos que se necesitarán. Mediante esta elección es que se determinan los sucesos a seguir para realizar un software que cumpla ciertos parámetros de calidad para que finalmente cumpla con las expectativas.

No existe la forma de escoger una metodología ideal para un proyecto, simplemente se busca la metodología que sea más conveniente y que cumpla de mejor manera con las necesidades del proyecto o en su defecto adaptar un modelo a las necesidades personales.

En efecto de los apartados 1.2 y 1.3 además de no tener un cliente real, el modelo iterativo incremental es el que más se adapta a las necesidades de desarrollo del proyecto.

1.5 Plan de Trabajo.

En consecuencia a los objetivos planteados en los apartados 1.2 y 1.3 la planificación del estudio consta de las siguientes etapas:

1.-Etapa de recolección de información y análisis: En esta primera etapa en un principio se recolecta la información de las fuentes seleccionadas, luego se realiza un análisis de esta información viendo cada una de las referencias investigadas, luego se

selecciona cuál de estas referencias son las de más relevancia y las que se utilizarán en este estudio. De estas referencias se pretende llegar a obtener el marco teórico y comprender la problemática del tema a estudiar.

Los hitos de esta etapa serán la obtención del marco teórico y estado de arte, tanto como de Android como de webservice, una vez obtenido este se podrá determinar finalmente cual es el prototipo de solución del estudio para luego pasar a la siguiente etapa.

2.-Análisis, desarrollo y construcción de modelos: Con la información obtenida y analizada en la etapa anterior, se procede al análisis, desarrollo y construcción de modelos. Para lograr un buen modelo es necesaria en una primera instancia una buena captura de requerimientos del problema planteado, una vez realizado esto, se procede a la construcción del modelo de la solución, haciendo los respectivos análisis u observaciones que surjan en el proceso.

3.- Implementación formal de los modelos y etapas de pruebas finales: En base a los modelos y prototipos obtenidos se desarrolla un software (Firmware en este caso) para obtener la solución al problema planteado. Una vez desarrollado el código fuente se realizarán sus posteriores pruebas en base a las distintas ejecuciones del middleware que se realicen y se compararán los resultados de este hasta ver cuál es la solución más óptima.

4.- Retroceso a las etapas de recolección de información y análisis: Como estamos trabajando con el modelo iterativo incremental es necesario volver a la etapa de recolección de información y análisis ya que debemos realimentar y aplicar repetidamente, con una filosofía iterativa.

1.6 Plan de trabajo tentativo.

Nombre de la Tarea	Comienzo	Fin	Duración
Proyecto 1			
Definir ámbito, alcance y objetivos del proyecto	31.07.2012	04.08.2012	5 días
Desarrollo del marco teórico sobre Portafolio Electrónico.	05.08.2012	09.08.2012	5 días
Desarrollo del marco teórico sobre dispositivos móviles.	10.08.2012	13.08.2012	4 días
Investigación sobre la plataforma Android.	14.08.2012	18.08.2012	5 días
Investigación de Tecnologías Servicios Web y su arquitectura en Dispositivos Móviles.	19.08.2012	24.08.2012	6 días
Estudio de Factibilidad (Ver Anexo B)	25.08.2012	29.08.2012	5 días
Estudio de los requerimientos del sistema ya propuesto.	30.08.2012	06.09.2012	8 días
Análisis de los módulos a los que se les incorporará movilidad.	07.09.2012	10.09.2012	4 días
Modelado	10.09.2012	17.09.2012	8 días
Desarrollo de prototipo 1	01.10.2012	15.11.2012	34 días
Desarrollo Informe Proyecto 1	09.11.2012	15.11.2012	5 días
Proyecto 2			
Desarrollo de prototipo 2	04.03.2013	15.04.2013	43 días
Desarrollo de Servicios Web	04.03.2013	15.04.2013	43 días
Desarrollo sistema final	22.04.2013	15.06.2013	55 días
Plan de pruebas	15.06.2013	19.06.2013	4 días

Tabla 1. 1 Plan Tentativo

Capítulo II: Dispositivos móviles.

2.1 ¿Qué es un Dispositivo Móvil?

Los dispositivos móviles son aparatos pequeños con algunas capacidades de procesamientos, con conexión permanente o intermitente a una red, con memoria limitada, diseñados específicamente para una función, pero que pueden llevar a cabo otras más generales.

Normalmente se asocian al uso individual, tanto en posesión como en operación, los cuales pueden adaptarse al gusto de la persona.

Una característica importante es el concepto de movilidad: los dispositivos móviles son aquellos suficientemente pequeños para ser transportados y empleados durante su transporte. Normalmente se sincronizan con un sistema de sobremesa para actualizar aplicaciones y datos por medio de una red inalámbrica, dando la impresión al usuario de que los datos están almacenados en el propio dispositivo [2].

2.2 Características de un Dispositivo Móvil.

Las principales características de un dispositivo móvil son [2]:

- ❖ Portabilidad
- ❖ Aparatos pequeños.
- ❖ Tienen una limitada capacidad de procesamiento.
- ❖ Pueden o no tener una conexión a una red.
- ❖ Su memoria es limitada.
- ❖ Diseñados específicamente para una función, aunque se pueden añadir algunas otras.
- ❖ Normalmente son para el uso individual.

Otra característica importante es el término en inglés “Wireless” (inalámbrico), un dispositivo inalámbrico es aquel que es capaz de comunicarse o acceder a una red sin cables, por ejemplo un teléfono móvil o un PDA. Este tipo de dispositivos se comportan como si estuviera conectado a una red mediante un cable, pero su tecnología es inalámbrica.

Más allá de estas características comunes, los dispositivos móviles forman en la actualidad un grupo sumamente heterogéneo y pueden incorporar casi cualquier componente de hardware y software que amplía y diversifica su función inicial. El más frecuente sin duda es la conexión telefónica (incluyendo servicios como el envío de SMS (mensajes de texto), MMS (mensajes multimedia) y acceso WAP) o la conexión a internet.

Alguna de las características que hacen que estos dispositivos sean diferentes de los computadores de escritorio son las siguientes:

- ❖ Tienen una funcionalidad limitada.
- ❖ No es necesariamente extensible y actualizable.
- ❖ En pocos años, el usuario deberá cambiarlo.
- ❖ Son más baratos.
- ❖ Son menos complicados en su manejo.
- ❖ Fácil de aprender su operación.
- ❖ No se requiere un usuario experto.

2.3 Sistemas Operativos para Dispositivos Móviles.

El sistema operativo destinado a correr en un dispositivo móvil necesita ser fiable y tener una gran estabilidad, ya que incidencias habituales y toleradas en computadores personales como reinicios o caídas no tienen cabida en un dispositivo de estas características. Además, ha de adaptarse adecuadamente a las consabidas limitaciones de memoria y procesamiento de datos, proporcionando una ejecución exacta y excepcionalmente rápida al usuario.

Estos sistemas han de estar perfectamente testeados y libres de errores antes de incorporarse definitivamente a la línea de producción. Las posibilidades que existen en un computador estándar de realizar actualizaciones e incluso reinstalar mejores versiones del sistema para cubrir fallos o deficiencias son más limitadas en un dispositivo móvil.

Es posible incluso que un aparato de esta naturaleza deba estar funcionando ininterrumpidamente durante semanas e incluso meses antes de ser apagado y reiniciado a diferencia de lo que ocurre con un computador personal. El consumo de energía es otro tema muy delicado, es importante que el sistema operativo haga un uso lo más racional y provechoso posible de la batería, ya que esta es limitada y el usuario siempre exige una mayor autonomía.

Todos estos aspectos de los dispositivos móviles, entre otros, han de ser tenido en cuenta a la hora de desarrollar un sistema operativo competente en el mercado, atractivo para los fabricantes y que permita al usuario sacar máximo provecho de su terminal.

En la actualidad, existen varios sistemas operativos para toda la gama de dispositivos móviles como por ejemplo Symbian y Android que son descritos por Baz en el año 2009 [2], Windows Mobile el cual lo describe Aranaz también en el año 2009 [2], BlackBerry OS, iOS y Palm OS. En el año 2009 Andrade [2] realizó una comparación de todos estos sistemas operativos, comparando distintos aspectos como el kernel, la conectividad y la adaptabilidad.

En el Anexo A se encuentra un detalle más técnico de la Plataforma Android.

2.4 Aprendizaje y dispositivos móviles.

Los dispositivos computacionales móviles, como el teléfono celular, los PDA o Tablet-PC aplicados a entornos educativos, son una alternativa innovadora que potencialmente puede apoyar una mejora en los procesos de enseñanza-aprendizaje.

A la intersección de la educación en línea y los dispositivos móviles se le conoce como “aprendizaje móvil”. Las ventajas que ofrece es que promete el acceso frecuente e integral a las aplicaciones software que apoyan el aprendizaje “en cualquier momento y en cualquier lugar”.

Dicho de otra forma, el aprendizaje móvil puede ser visto como la utilización de dispositivos móviles en el proceso de aprendizaje.

El aprendizaje móvil se ha utilizado en cursos de educación formal y también en educación informal, como puede ser la visita a un museo o jardín botánico. Lo interesante de estas tecnologías emergentes es que permite aprovechar sus características únicas y las nuevas oportunidades que brindan para un uso innovador de la interacción, comunicación y conexión entre los estudiantes, el tutor y los sistemas de gestión del aprendizaje [5].

2.5 ¿Por qué Android?.

Android, iOS, WindowsPhone y BlackBerry han sido comparados entre sí una y otra vez con cada versión y actualización que liberan, e indiscutiblemente cada uno ha hecho que el otro mejore, tomando características entre si y evolucionando cada vez más buenas y otras no tanto y aunque algunos disfruten mucho la experiencia de otras plataformas se plantea la pregunta ¿Por qué elegir a Android sobre otras plataformas?

En este proyecto se ha escogido el sistema operativo Android para el desarrollo de la aplicación por los siguientes motivos:

- ❖ **Extenso mercado:** Cubre desde teléfonos de gama baja, los cuales cumplen con éxito sus funcionalidades básicas, hasta terminales de gama alta pioneros en hardware.
- ❖ **Código abierto:** El sistema operativo Android está liberado (bajo la licencia Apache) y su código es totalmente abierto, por lo que cualquier desarrollador puede modificarlo y mejorarlo, construyendo entre todos un sistema operativo mejor.
- ❖ **Libertad:** El usuario puede instalar cualquier tipo de aplicación en el dispositivo desde el mercado de Android (Google Play) o descargando aplicaciones .apk (extensión de las aplicaciones de Android) de terceros sin ningún problema de restricción.
- ❖ **Sin fronteras:** Android no se reserva nunca el derecho a escoger una determinada operadora para imponer al usuario el hecho de contratarla para poder disfrutar de él.
- ❖ **Comunidad:** Android cuenta con la comunidad de desarrolladores más grande del mundo, siendo así fácil encontrar información de éste. Además de eventos, foros, etc.

- ❖ **Multitarea:** Android es un dispositivo que permite realizar diferentes tareas a la vez, ya que el propio sistema operativo se encarga de gestionar sus recursos pausando tareas, suspendiéndolas, cerrándolas en periodos de inactividad, etc.
- ❖ **Personalizable:** El usuario tiene la opción de personalizarlo todo a su gusto. No solo cambiar colores e iconos, si no todo el contenido del sistema.
- ❖ **Liderazgo:** La plataforma Android de Google mantiene un liderazgo indiscutible en el mercado global de los teléfonos inteligentes, correspondiéndole una cuota de casi el 40% a finales de 2011.

A continuación se puede ver una comparativa sobre los sistemas operativos más utilizados:

En el año 2007, dentro de los Smartphone, Symbian se lleva la mayor cuota de mercado con un 66%, seguido de Windows Mobile con un 10%.

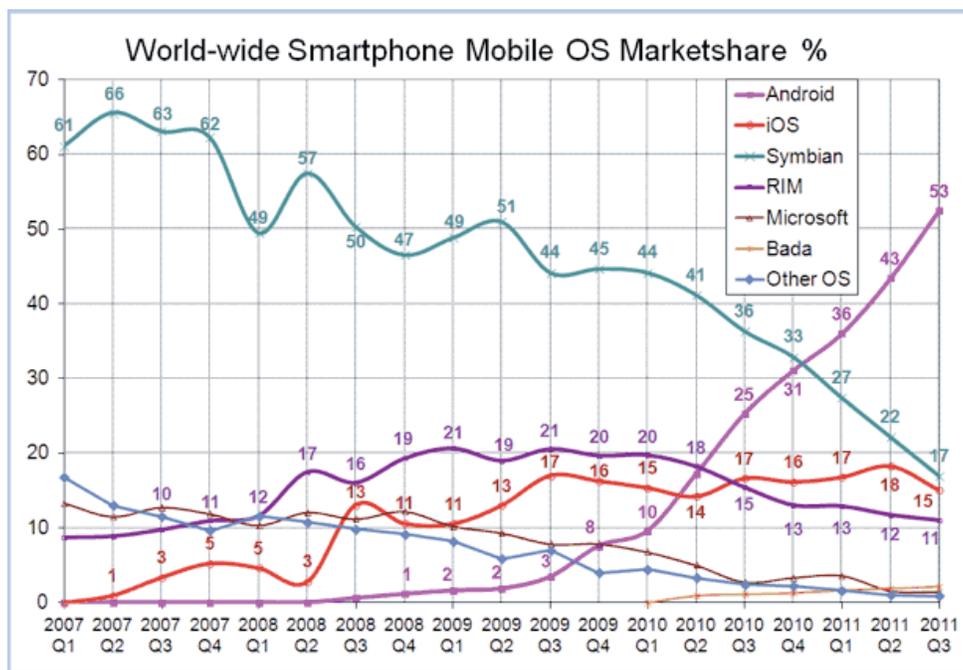


Figura 2. 1 Grafico Evolución Android[1]

Valorando todas estas opciones, se decide apostar por el desarrollo de una aplicación el sistema operativo Android.

Capítulo III Especificación de Requerimientos.

3.1 Descripción del problema.

La vejez es un proceso por el que todos los seres humanos pasarán, si privilegiadamente llegamos siquiera a los 60 años de edad, no es una enfermedad es una etapa mediante la cual en forma natural el ser humano experimenta en forma personal y con diferente intensidad, su llegada a la senectud.

Las personas que llegan a cierta edad afrontan problemas físicos, psíquicos y sociales que derivan de cambios biológicos propios de la edad, y que a veces establecen de forma muy rígida, la capacidad para llevar a cabo un acto tan natural como, el de comer o dormir diariamente.

Más de un 80% del cuidado es suministrado por familiar que convive con él, esa persona proporciona supervisión, cuidado directo y coordinación de estos servicios, pero la mayoría de las personas que cuidan de un adulto mayor sufren de falta de conocimiento del tiempo que se requiere para el cuidado, agotamiento de los recursos económicos, conflicto en la unidad familiar, etc. [7]

Hoy en día no existe ningún programa, ni aplicación que permita el monitoreo de actividades de un adulto mayor, el que pueda enviar notificaciones periódicas sobre sus quehaceres diarios, ni tampoco sobre su seguridad al momento de sufrir caídas o momentos críticos de inactividad.

3.1.1 Solución propuesta.

Crear una aplicación para Smartphone que sea compatible con Android (2.2 froyo hacia adelante) complementado con webservice.

Todos los Smartphone contienen un acelerómetro de tres ejes, su principal función es detectar la orientación del teléfono.

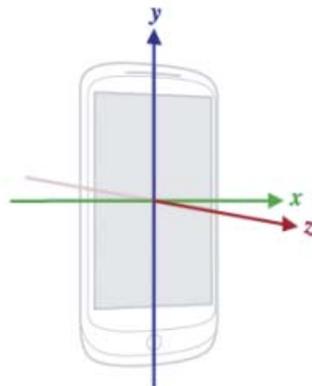


Figura 3. 1 Acelerómetro [4]

Gracias al acelerómetro podremos determinar movimientos. La aplicación utilizará el acelerómetro para determinar ciertos patrones de movimiento los cuales interpretará como posibles golpes y/o caídas bruscas, como también momentos de inactividad. Al momento de ocurrir esto la aplicación se comunicará a través de webservice enviando una notificación, las cuales se conocen como notificaciones push, que ocupan el servidor de google (C). Para poder determinar estos patrones de movimiento, se optó por tomar la aceleración en el instante que el acelerómetro capte algún movimiento, utilizando herramientas de “*ANDROID DEVELOPER*” que permiten trabajar con las formulas de la aceleración.

```
double a = Math.round(Math.sqrt(Math.pow(x, 2) + Math.pow(y, 2) + Math.pow(z, 2)));  
currentAcceleration = Math.abs((float)(a-9.8));
```

Figura 3. 2Formula de Aceleración [6]

El acelerómetro capta los movimientos en milisegundos, es decir, si el usuario lleva el móvil en el bolsillo mientras camina, la cantidad de movimientos a registrar serian miles e incluso millones, esto se traduciría en la misma cantidad de notificaciones lo cual saturaría el servidor, por este motivo es fundamental definir un criterio para identificar un movimiento “brusco”.

El criterio de identificación de movimiento brusco, estará dado por un límite de aceleración permitida, esto quiere decir que el acelerómetro estará siempre escuchando y cuando capte un movimiento con una aceleración mayor a cierto rango especifico (11mts/seg²), esté reconocerá como uno, todos los movimientos captados hasta que la aceleración vuelva a su estado inicial (0 mts/seg²) para posteriormente enviar una notificación (ver Figura 3.3) hay que tener en cuenta que esto se realizara en milisegundos.

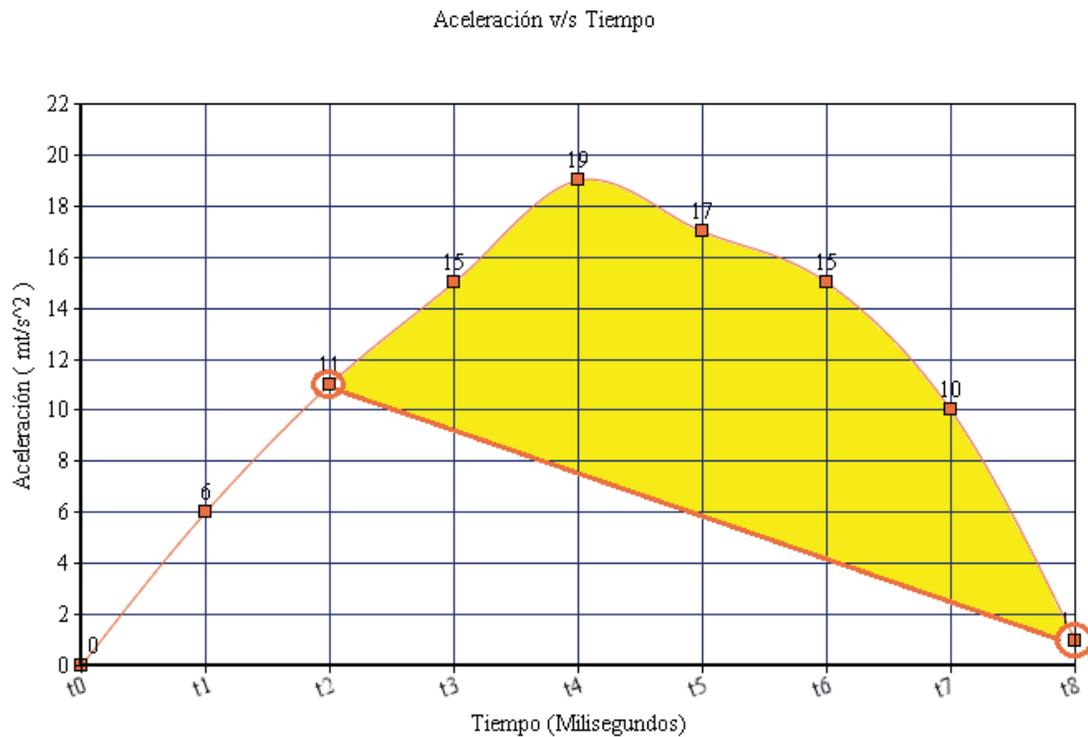


Figura 3. 3 Grafico de Aceleración v/s Tiempo

Las notificaciones se verán en la barra de estado del Smartphone del Cuidador para así poder ahorrar recursos de batería, ya que la aplicación estará conectada en todo momento a internet.

El Cuidador tendrá dos aplicaciones, una APP y otra web, las cuales le permitirán poder acceder tanto de su PC como de su Smartphone.

La aplicación web estará montada en un servidor el cual se podrá comunicar con la plataforma de Android gracias a webservice y C2DM.

Cabe mencionar que para la parte de cuidado, se ha entrevistado a una enfermera con más 30 años en cuidado de adulto mayor, esta proporcionó los siguientes requerimientos.

3.1.2 C2DM.

C2DM es un servicio que ayuda a los desarrolladores a enviar datos desde sus servidores a sus aplicaciones móviles instaladas en terminales móviles Android. Se trata de un servicio gratuito, siempre y cuando no sobrepasemos los volúmenes de tráfico de datos establecidos por Google. En la mayoría de los casos no será necesario contratar una ampliación de estas cuotas de tráfico, ya que son bastante altas (Aproximadamente 200.000 notificaciones/día). [10]

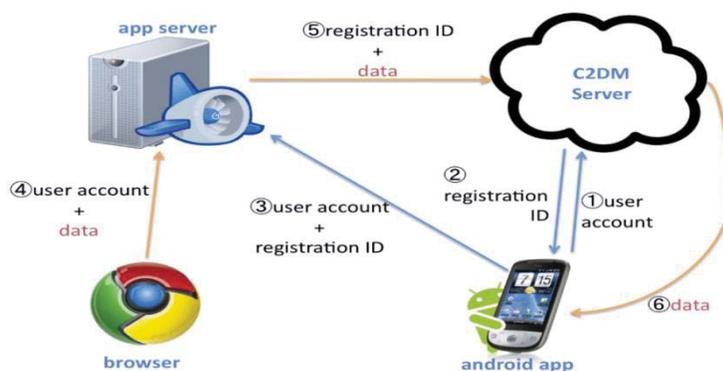


Figura 3. 4 C2DM [10]

3.1.3 Análisis de requerimientos.

Cuando el Cliente solicita que se desarrolle un sistema tiene algunas nociones de lo que debe hacer. Por esta razón cada sistema basado en software tiene un propósito, usualmente expresado con algo que el sistema debe hacer.

Un Requerimiento *“es una característica del sistema o una descripción de algo que el sistema es capaz de hacer con el objeto de satisfacer el propósito del sistema”*. [6]

Es decir, los requerimientos son lo que los clientes/Cuidadores esperan que haga el sistema. En sí el objetivo del análisis de requerimientos es resolver el problema.

3.1.4 Requerimientos Funcionales.

Un requerimiento es una condición o capacidad que el sistema (siendo construido) debe cumplir, en software es una capacidad de este, requerida por el Cuidador para resolver un problema o alcanzar un objetivo.

Cuando se expresan como requerimiento del sistema describen con detalle la función de éste, sus entradas y salidas, excepciones, etc.

En este proyecto se dividirán en dos partes las cuales serán los requerimientos de la aplicación web y aplicación móvil.

A continuación se identificarán el(los) Cuidador(s) y/o participante(s) del sistema:

Cuidador: Encargado de administrar e ingresar los datos y los tiempos de monitoreo del sistema.

Servidor C2DM: Servidor externo al sistema, estará encargado de registrar al Smartphone y también de enviar las notificaciones.

3.1.4.1 Requerimientos Funcionales de la aplicación web.

Luego de identificar a los Cuidador y/o participantes se listarán los requerimientos funcionales del sistema web:

1)	Permitir el registro de los Cuidadores.
2)	Permitir el ingreso del tiempo de las notificaciones.
3)	Permitir listar todos los datos del Cuidador.
4)	Permitir eliminar los datos del Cuidador.
5)	Permitir modificar los datos del Cuidador.
6)	Verificar los datos del Cuidador al momento de ingresar sus datos.
7)	Buscar Cuidadores.

Tabla 3. 1 Requerimientos Funcionales Aplicación Web

En la tabla 3.1 se pueden apreciar los requerimientos funcionales los cuales son esenciales para el desarrollo del proyecto ya que de estos saldrán los casos de uso.

3.1.4.2 Requerimientos Funcionales de la aplicación móvil.

Esta es la parte más fundamental de este proyecto, debido a que los Cuidadores finales llámense abuelos, tienen que ser “*Adultos mayores Independientes*” los cuales tienen cierto tipo de independencia, los cuales no se encuentran postrados y tienen un cierto nivel de demencia aceptable.

1)	Permitir Loguearse
2)	Acceder al menú de monitoreo
3)	Listar el Historial médico del abuelo
4)	Recibir notificaciones Push
5)	Ingresar los datos del cuidado
6)	Ingresar Grabación
7)	Listar Características del abuelo
8)	Administrar tiempos de monitoreo
9)	Saber la localización del abuelo al momento de la notificación
10)	Saber la hora de la notificación
11)	Saber si el abuelo recibió el cuidado

Tabla 3. 2 Requerimientos Funcionales Aplicación Móvil

3.1.5 Requerimientos No Funcionales.

Los requerimientos no funcionales son todas aquellas características o funcionalidades externas al sistema en sí, pero que influyen o limitan al sistema, como por ejemplo, rendimiento(el tiempo y espacio), interfaces de Cuidador, fiabilidad(robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, servidores, portabilidad, estándares, velocidades, etc.

3.1.5.1 Requisitos tecnológicos.

Sistema computacional: El Cuidador debe contar con al menos con dos Smartphone y un computador para poder hacer funcionar la aplicación.

Sistema de información: El Cuidador debe de contar con cualquier navegador web en el cualquier sistema operativo, además de contar con una conexión a internet ya sea Wi-Fi, Lan o plan de datos(Smartphone), para poder acceder al sistema de monitoreo y las funcionalidades de la aplicación

3.1.5.2 Interfaz.

La interfaz del sistema es súper importante para que el sistema tenga una buena adaptación para el Cuidador, para ellos se desarrollaran prototipos para que el Cuidador acepte las propuestas

3.1.5.3 Disponibilidad.

El sistema es una aplicación web y una App que estará disponible durante todo el tiempo a toda hora en cualquier lugar donde se tenga acceso a internet, es importante para el Cuidador acceder a toda la información del sistema independiente del lugar físico en donde se encuentre

3.1.5.4 Seguridad.

El sistema cuenta con una función de autenticación del sistema, en la cual se realiza mediante la petición de un Cuidador y contraseña, también se ocupara el servidor de google C2DM el cual enviara n ID o token único a cada Smartphone, que es para el caso de que un Cuidador tenga más de uno y quiera saber cuál de todos las adultos mayores fue el que se encuentra en mal estado.

Capítulo IV Diseño de la Solución.

4.1 Arquitectura.

Una vez ya definida que metodología utilizar, el grupo de trabajo también ha determinado cómo será la arquitectura a implementar para el desarrollo del software. Esta será desarrollada en la plataforma J2EE, la cual está basada en estándares para el desarrollo, despliegue y gestión de aplicaciones compuestas de múltiples capas.

J2EE nos es útil por el motivo de que está basada en la idea componente-contenedor y por su portabilidad de código, permitiendo que la aplicación pueda funcionar en cualquier servidor, y plataforma.

Con respecto a la arquitectura lógica de la solución, se implementará un sistema de 3 capas la cual estará formada por la capa de Presentación, Lógica de Negocios y Datos.

A modo de una mejor comprensión se detallarán cada una de las capas de la siguiente manera:

- **Capa de Presentación:** Esta capa consta de lo que será presentado visualmente al Cuidador, por medio de esta es en donde hará interacción con el software y proporcionará los datos de entrada. Estos datos serán enviados a la capa Lógica de Negocios a través de protocolos SOAP.
- **Capa Lógica de Negocios:** En esta capa está situado el servicio web, la cual contendrá los métodos del sistema y estará situada en GlassFish. Aquí los datos entregados por el Cuidador son procesados, para así automatizar los métodos, brindándole al Cuidador el servicio correspondiente.
- **Capa de Datos:** Es la cual en donde está la Base de Datos. Esta se relaciona con la capa de Lógica de Negocios con JDBC (Java Database Connectivity), el cual permitirá la conexión entre ambas capas, permitiendo hacer operaciones sobre la Base de Datos en PostgeSql, desde el lenguaje Java.

La separación de nuestro sistema en 3 capas, permitirá la optimización de tiempo en su desarrollo, como la protección de los datos. Además ante cualquier falla se podrá acceder directamente a la capa, evitando tener que revisar el sistema completamente para así hacer el arreglo correspondiente.

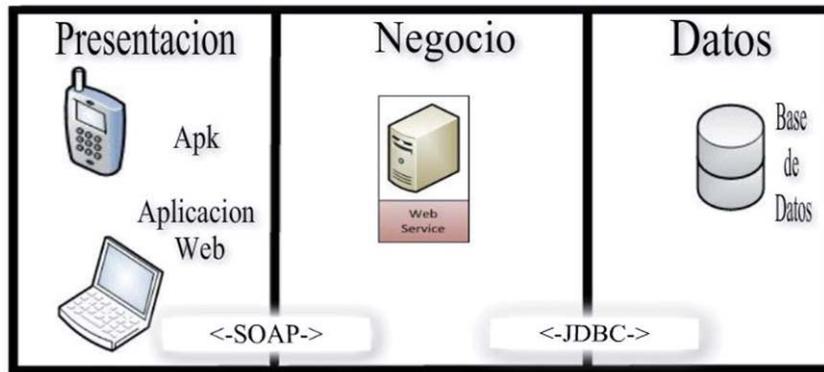


Figura 4. 1 Arquitectura Lógica del sistema.

Además como arquitectura física nuestro sistema está desarrollado por cliente-servidor. En la parte de cliente el Cuidador tendrá la forma de acceso y comunicación con nuestro sistema, por otro lado en el servidor se encontrarán almacenado todos los métodos y programas para el adecuado funcionamiento de la aplicación, teniendo ahí mismo alojado la base de dato correspondiente.

4.2 Diagrama de Clases.

Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargarán del funcionamiento y la relación entre uno y otro.

Para este proyecto es esencial debido a que el diagrama de clases será la estructura del modelo entidad relación y el modelo relacional, debido a que Android trae sus propias librerías con sus respectivas clases llamadas activity. Las cuales no se deben agregar al diagrama de clases.

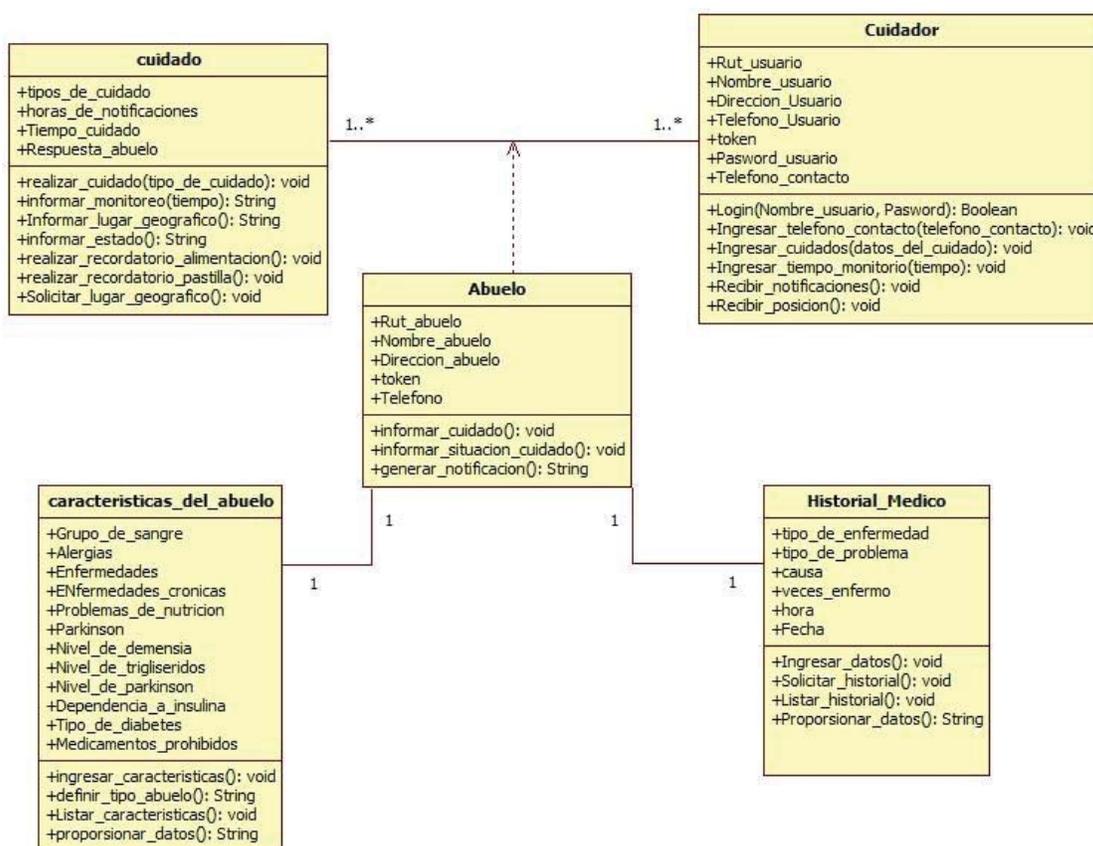


Figura 4. 2 Diagrama de clases

4.3 Modelo Relacional.

Después de realizar en modelo entidad relación (revisar anexo), en el cual nos basamos en el modelo de clases podemos llegar al modelo relacional.

Acá se puede apreciar el modelo relacional, que es similar al modelo entidad de relación, sólo que este se encuentra más refinado.

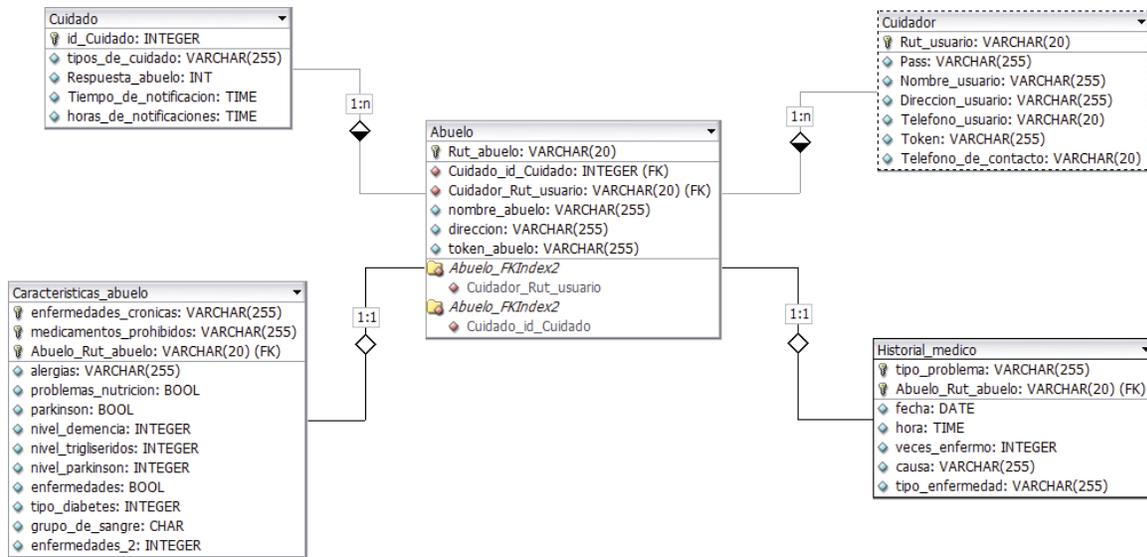


Figura 4. 3 Modelo Relacional

4.4 Casos de uso.

Con lo mencionado anteriormente y basándose en toda la información entregada, se muestran las distintas funcionalidades que realiza el sistema, con sus respectivas relaciones con el Cuidador, demostrando que las actividades que realiza están dentro de una arquitectura compleja que interactúan entre sí, para terminar por conformar un sistema completo.

Cabe mencionar que las actividades de caso de uso que se encuentren de color blanco hacen referencia a funcionalidades correspondiente a la parte móvil.

4.4.1 Caso de uso General.

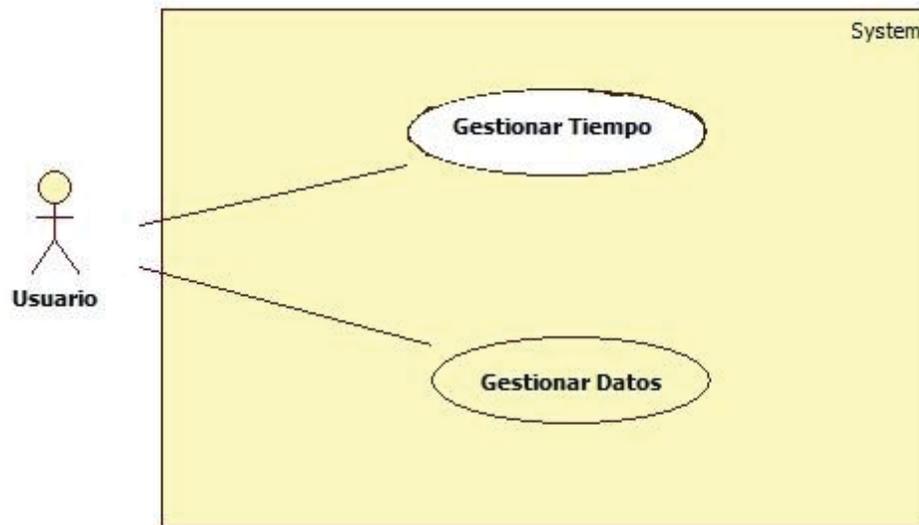


Figura 4. 4 Caso de uso general

En la figura 4.4 se puede apreciar el caso de uso general, el cual consta de tres grandes funcionalidades las cuales corresponden a Gestionar tiempo, Gestionar datos y Logout. La acción de Gestionar Tiempo que se encuentra de color blanco, esta corresponde a la actividad realizada desde el Smartphone mientras que las otras corresponden a la aplicación web. Posteriormente cada una será explicada con su caso de uso correspondiente.

4.4.2 Casos de uso Gestionar Datos.

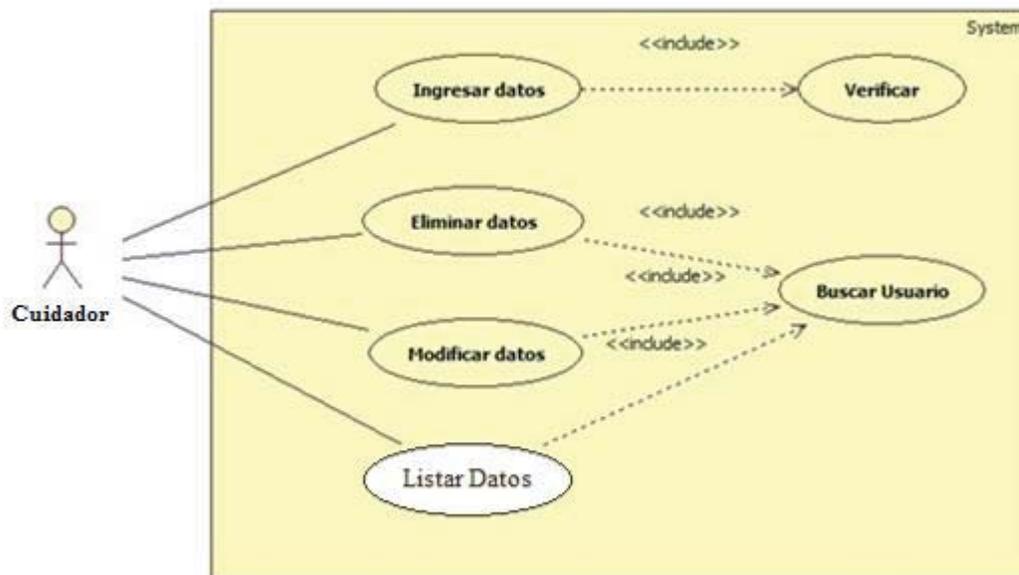


Figura 4. 5 Caso de uso Gestionar Datos

En la figura 4.5 se puede apreciar el caso de uso Gestionar Datos el cual muestra como el Cuidador tiene acceso a poder ingresar datos, eliminar datos, modificar sus datos y listar sus datos, obviamente se observa que para poder ingresar los datos hay que verificar que no se encuentre repetido, también para poder eliminar modificar y listar se necesita buscar al Cuidador. Cabe mencionar que listar datos lo hace de manera móvil.

Caso de uso	Gestionar Datos.
Objetivo	Este caso de uso tiene como objetivo permitir que el Cuidador poder acceder a todos las opciones que otorga el sistema para poder gestionar sus datos.
Actor principal	Cuidador
Precondición	EL cuidador ingreso al campo de Gestionar datos
Inicio	Cuidador

Tabla 4. 1 Narrativo Extendido de Gestionar Datos

4.4.3 Caso de uso Gestionar Tiempo.

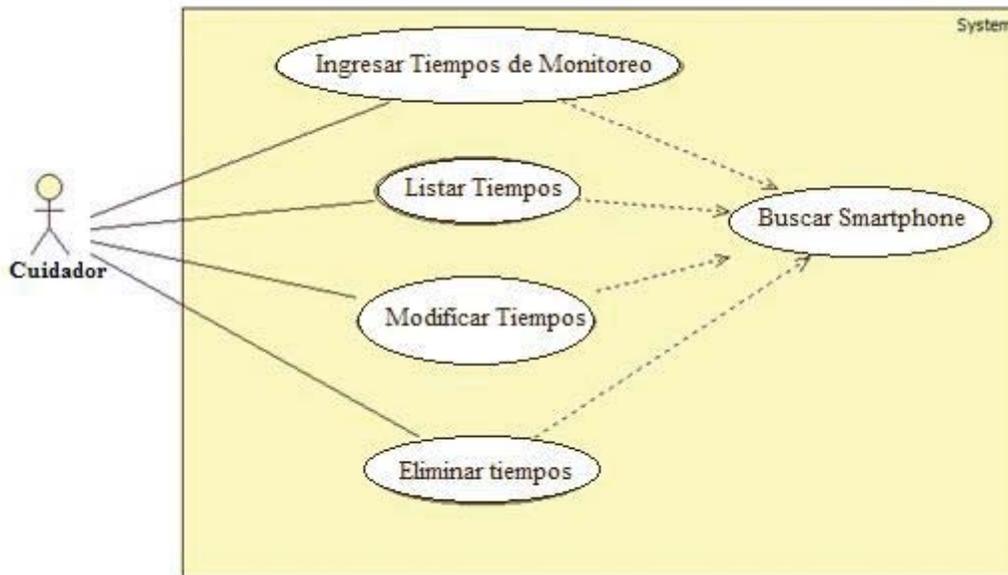


Figura 4. 6 Caso de uso Gestionar Tiempo

En la figura 4.6 se puede observar el caso de uso Gestionar tiempo el cual muestra las opciones de tiempo para el monitoreo del adulto mayor, los cuales consta de ingresar tiempos de monitoreo, listar, eliminar y modificar tiempos obviamente todos estos atributos se encuentran en la aplicación del Smartphone.

Caso de uso	Gestionar Tiempo
Objetivo	Este caso de uso tiene como objetivo permitir que el Cuidador poder acceder a todos las opciones que otorga el sistema para poder gestionar el tiempo correspondiente al cuidado del adulto mayor.
Actor principal	Cuidador
Precondición	EL cuidador ingreso al campo de Gestionar datos
Inicio	Cuidador

Tabla 4. 2 Narrativo Extendido Gestionar Tiempo

4.5 Diagrama de Secuencia.

Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada caso de uso. Mientras que el diagrama de casos de uso permite el modelado de una vista de negocio del escenario, el diagrama de secuencia contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario y mensajes intercambiados entre los objetos.

A continuación se describirán algunos diagramas de clases empleados para este proyecto.

4.5.1 Diagrama de secuencia Gestionar Cuidado.

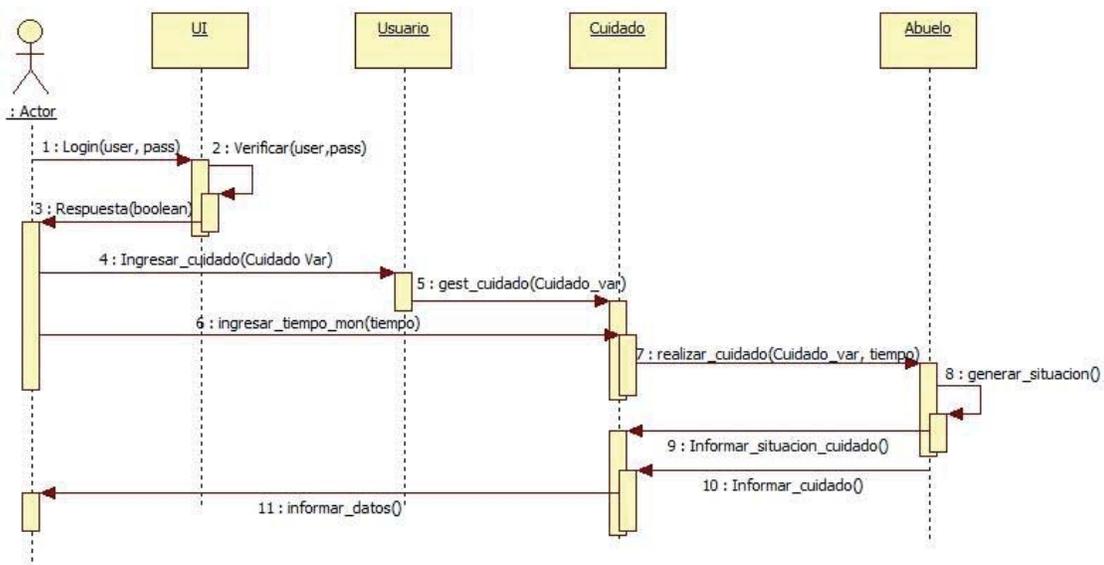


Figura 4. 7 Diagrama de secuencia Gestionar Cuidado

En la figura 4.7 se puede apreciar el diagrama de secuencia de Gestionar Cuidado, en este se observa como el Cuidador se debe de logear para así poder acceder a Gestionar datos del cuidado, cabe mencionar que en el método N°5 llamado `get_cuidado()` este alude a que se están tomando todos los métodos correspondientes para poder gestionar de manera correcta el cuidado, después de haber ingresado todos los datos para poder gestionar los cuidados del abuelo, esta entidad generará su situación actual y enviaría la situación del cuidado y el resultado del cuidado, para posteriormente informar al Cuidador.

4.5.2 Diagrama de secuencia Listar características.

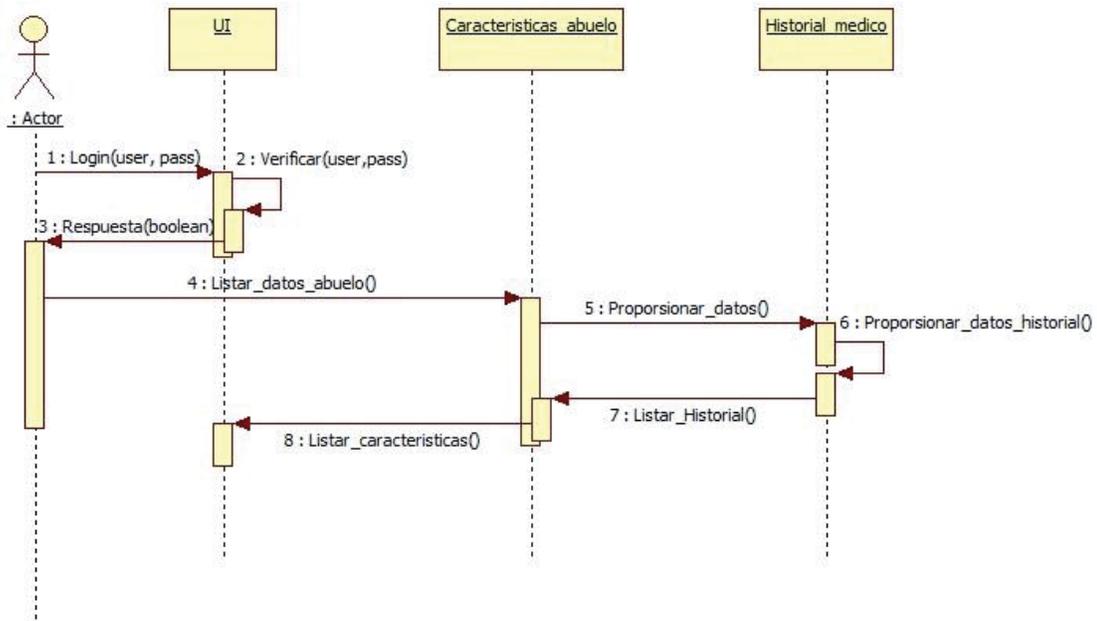


Figura 4. 8 Diagrama de secuencia Listar características

En la figura 3.12 se puede apreciar el diagrama de secuencia Listar características e Historial Abuelo en las que se puede observar como el Cuidador se debe de logear y directamente se comunica con la entidad `caracteristicas_abuelo` la cual proporcionara los datos para posteriormente solicitar el historial médico del abuelo y para finalizar se retornara el método `Listar_caracteristicas`.

4.6 Diseño de Interfaz (Prototipo 1).

Una interfaz a la conexión física y funcional entre dos sistemas o dispositivo, esta es por lo tanto, una conexión entre dos máquinas de cualquier tipo, a las cuales les brinda un soporte para la comunicación entre distintos niveles. Es posible entender la interfaz como un espacio (el lugar donde se desarrolla la interacción y el intercambio), instrumento (a modo de extensión del cuerpo humano, como el mouse que permite interactuar con una computadora) o superficie (el objeto que aporta información a través de su textura, forma o color). Se conoce como interfaz de Cuidador al medio que permite a una persona comunicarse con una máquina. La interfaz, en este caso, está compuesta por los puntos de contacto entre un Cuidador y el equipo. Además del mencionado ejemplo del mouse, otra interfaz de este tipo es la pantalla del monitor o el teclado.

A continuación se mostraran los prototipos de interfaces:

4.6.1 Login y Menú.



Figura 4. 9 Diseño de Login y Menú

En la Figura 4.9 se muestra la interfaz correspondiente al login, y menú las cuales mediante esta se podrá acceder al sistema tele asistencia (iniciar sesión), utilizando el email asociado a la cuenta y la contraseña correspondiente (ingresadas previamente a través de la aplicación web). Mientras la interfaz del menú principal el cual se desplegará luego del inicio de sesión. Existen 4 opciones: ingresar datos de cuidado, administración de monitoreo, ver historial e iniciar.

La opción “Ingresar datos de cuidado” es para establecer recordatorios respecto a los medicamentos que el adulto mayor deba tomar.

La opción “Administración de monitoreo” es para ver el perfil de Cuidador (datos del adulto mayor) y también para ver el historial de los registros de movimientos captados por el acelerómetro.

La opción “Iniciar” es para comenzar a registrar los datos y activar las notificaciones previamente configuradas en las opciones anteriormente descritas.

4.6.2 Ingresar datos de cuidado.



Figura 4. 10 Configurar Notificación y Administrar Monitoreo

En la Figura 4.10 se muestra la interfaz que permite configurar la notificación para un medicamento, para esto será necesario ingresar el nombre de este junto con una pequeña descripción. Adicionalmente se debe seleccionar una hora para las notificaciones, estas pueden tener asociada una alarma de voz (grabación) u otro sonido (mp3), también se puede establecer la opción de vibrar. Una vez ingresados todos los datos necesarios existen dos opciones: guardar e ingresar otro medicamento (se guarda y se actualiza la pantalla) o en su defecto guardar y salir (retornar al menú principal). Por otra parte Administrar Monitoreo permite al Cuidador configurar el intervalo de tiempo en que desea que lleguen las notificaciones respecto al monitoreo del adulto mayor.

4.6.3 Ver Historial.



Figura 4. 11 Interfaz Perfil de Usuario y Interfaz Historial Monitoreo

En la Figura 4.11 se encuentra la interfaz que mostrará los datos personales del adulto mayor (perfil de Cuidador), esta posee una opción llamada “ver historial de monitoreo” la cuál direccionara a la interfaz historial médico la cual mostrará todos los datos significativos captados mediante el acelerómetro, luego está la opción “salir” la cual retorna al menú principal.

4.6.4 Notificaciones de Cuidador.



Figura 4. 12 Figura de Interfaz de Notificación y Alarma de medicamento

En la Figura 3.16 se muestra la interfaz que vera el Cuidador al momento que le llegue una notificación (previamente establecida). Esta notificación consta de un mapa en el cual se detalla la ubicación actual del adulto mayor, la hora y un mensaje que detalla los movimientos significativos. Mientras que la interfaz de Alarma de medicamento a esta se le asocia un color debido a que los adultos mayores no recuerdan el nombre del medicamento sino que lo asocian a un color, también corresponde a la alarma de comer.

4.7 Diseño de Interfaz Móvil (Prototipo 2).

En el prototipo número 2 cumple las mismas funcionalidades que el prototipo anterior, el cambio principalmente se le realizo a la interfaz.

4.7.1 Login y Menú.



Figura 4. 13 Login y Menú

En la Figura 4.12 se muestra la interfaz correspondiente al login, y menú las cuales mediante esta se podrá acceder al sistema tele asistencia (iniciar sesión), utilizando el rut asociado a la cuenta y la contraseña correspondiente (ingresadas previamente a través de la aplicación web). Mientras la interfaz del menú principal el cual se desplegará luego del inicio de sesión. Existen 5 opciones: ingresar datos de cuidado, administración de monitoreo, ver historial, seleccionar abuelo (en el caso que sean más de dos) iniciar.

4.7.2 Ingresar datos de cuidado.

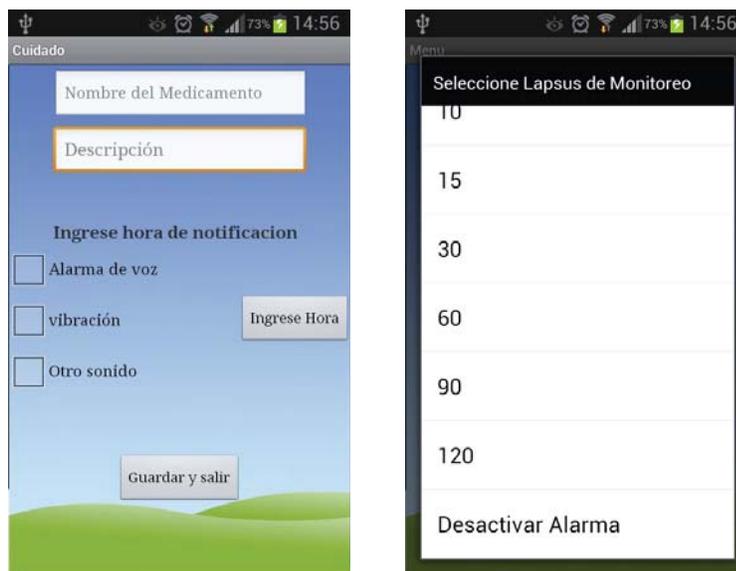


Figura 4. 14 Ingresar datos de cuidado

En la Figura 4.13 se muestra la interfaz que permite configurar la notificación para un medicamento, para esto será necesario ingresar el nombre de este junto con una pequeña descripción. Adicionalmente se debe seleccionar una hora para las notificaciones, estas pueden tener asociada una alarma de voz (grabación) u otro sonido (mp3), también se puede establecer la opción de vibrar. Una vez ingresados todos los datos necesarios existen dos opciones: guardar e ingresar otro medicamento (se guarda y se actualiza la pantalla) o en su defecto guardar y salir (retornar al menú principal). Por otra parte Administrar Monitoreo permite al Cuidador configurar el intervalo de tiempo en que desea que le lleguen las notificaciones respecto al monitoreo del adulto mayor

4.7.3 Ver Historial.



Figura 4. 15 Ver Historial

En la Figura 4.14 se encuentra la interfaz que mostrará los datos personales del adulto mayor (perfil de Cuidador), esta posee una opción llamada “ver historial de monitoreo” la cuál direccionara a la interfaz historial médico la cual mostrará todos los datos significativos captados mediante el acelerómetro, luego está la opción “salir” la cual retorna al menú principal.

4.7.4 Notificaciones de Cuidador e Inicio de Cuidado

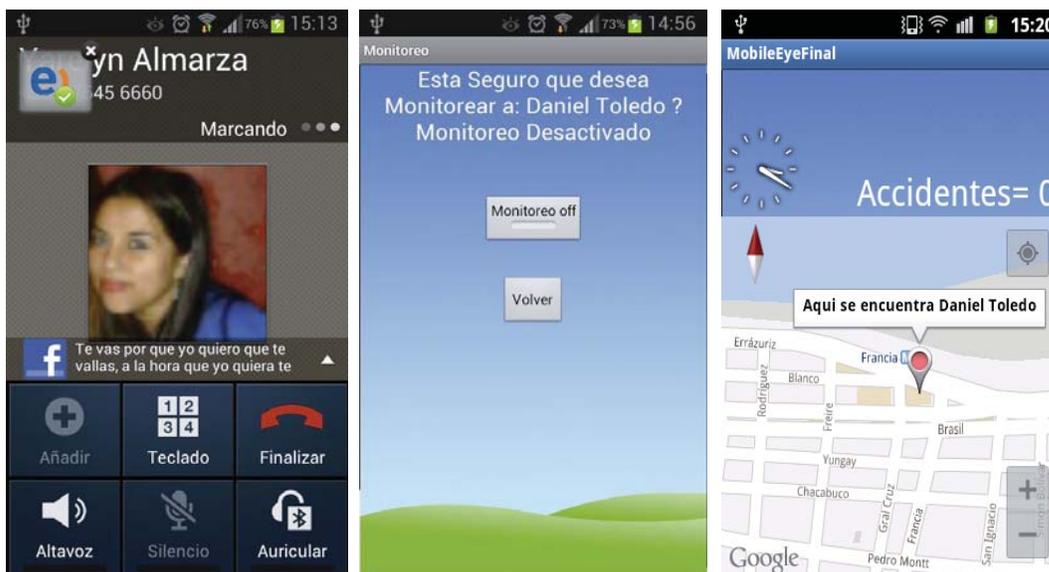


Figura 4. 16 Notificaciones e Inicio de Monitoreo

En la Figura 4.15 se muestra la interfaz que vera el Cuidador al momento que le llegue una notificación (previamente establecida). Esta notificación consta de un mapa en el cual se detalla la ubicación actual del adulto mayor, la hora y un mensaje que detalla los movimientos significativos. Por otra parte la interfaz de inicio de monitoreo que consta de dos botones uno de inicio de monitoreo y otro de volver, mientras que la otra interfaz corresponde al llamado automático al momento de ocurrir un accidente.

4.8 Diseño de Interfaz Web.

Se diseñó un módulo web el cual permite gestionar la cantidad de personas a cuidar con sus respectivas

4.8.1 Login y Listar/Agregar Abuelo.

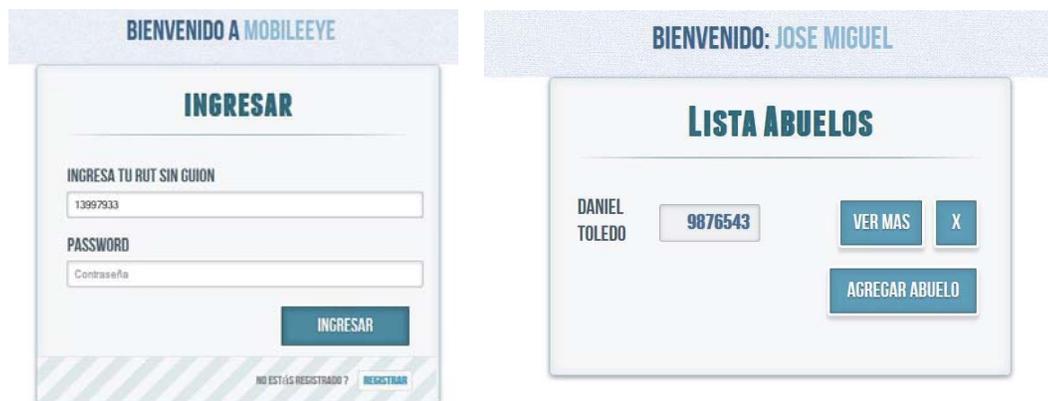


Figura 4. 17 Login y Listar/Agregar Abuelo

En la figura 4.16 se puede apreciar la interfaz del login que utiliza los mismos parámetros que la aplicación móvil, mientras que la otra interfaz corresponde a la lista de abuelos o personas a cuidar, se pueden observar 2 botones los cuales corresponden al detalle y agregar abuelo.

4.8.2 Listar y Agregar Enfermedades Abuelo.



Figura 4. 18 Listar y Agregar enfermedades abuelo

En la figura 4.17 se puede apreciar los datos del abuelo los cuales se pueden modificar mientras que la otra interfaz se puede apreciar el historial médico, que también tiene la cualidad de modificar.

Capítulo V Plan de Pruebas.

5.1 Documento de Casos de Pruebas.

Dentro de la realización de un proyecto, los documentos de casos de pruebas son los que nos permite tener una vista general de cómo reaccionará la aplicación en los distintos escenarios de pruebas. La finalidad es tener las actividades de los casos de pruebas, especialmente la salida esperada y obtenida, como las acciones que tendrá la aplicación.

A continuación presentaremos los documentos de casos de pruebas:

Introducción/visión general
Identificador: 001
Caso de prueba dueño/creador: Daniel Toledo.
Versión: Primaria
Nombre: Informar al usuario el registro del teléfono y el guardar su ID en la BD.
Propósito: Dar a cada Smartphone un id único el cual puede variar en el tiempo, para saber a quién realmente se le entrego el Smartphone
Dependencias: No incluye otros subsistemas.
Actividades de los casos de prueba
Ambiente de prueba/configuración: La prueba realizara en un Smartphone.
Inicialización: Necesita de la base de datos.
Finalización: Se desprenderá una opción para “Ver ID”
Acciones: 1.- El usuario ingresa a la aplicación. 2.- El usuario ingresa a la opción “Login”. 3.- El usuario ingresa selecciona datos en el menú. 4.- El sistema muestra el resultado de la petición requerida.
Descripción de los datos de entrada: Rut y Password
Salida esperada: la salida esperada es el ID del Smartphone
Salida obtenida: ID completo prueba.
Resultado: Correcto.

Severidad: Menor.
Seguimiento: contiene la o las líneas de código.
Fecha: 15/06/2013

Tabla 5. 1 Caso de Prueba 001

Introducción/visión general
Identificador: 002
Caso de prueba dueño/creador: Daniel Toledo.
Versión: Primaria
Nombre: Informar al usuario una notificación.
Propósito: Comprobar el envío de mensajería instantánea.
Dependencias: No incluye otros subsistemas.
Actividades de los casos de prueba
Ambiente de prueba/configuración: La prueba realizara en un Smartphone.
Inicialización: Necesita de la base de datos.
Finalización: Muestra la notificación con la posición geográfica.
Acciones: 1.- El usuario ingresa a la aplicación. 2.- El usuario deja la aplicación corriendo. 3.- el “acelerómetro” detecta movimiento y/o no movimientos dentro de un lapsus de tiempo. 4.- El sistema muestra el resultado de la petición requerida.
Descripción de los datos de entrada: No necesita datos de entrada.
Salida esperada: la salida esperada son las coordenadas para su posterior muestra en el mapa.
Salida obtenida: Coordenadas que son mostradas en el mapa.
Resultado: Correcto.
Severidad: Grave, Mayor.
Seguimiento: contiene la o las líneas de código, en donde la prueba fallo.
Fecha: 16/06/2013

Tabla 5. 2 Caso de Prueba 002

Introducción/visión general
Identificador: 003
Caso de prueba dueño/creador: Daniel Toledo.
Versión: Primaria
Nombre: Permitirle al usuario guardar todos sus preferencias en la aplicación.
Propósito: Tener en cuenta todas las preferencias del usuario.
Dependencias: No incluye otros subsistemas.
Actividades de los casos de prueba
Ambiente de prueba/configuración: La prueba realizara en un Smartphone.
Inicialización: Se necesita de base de datos.
Finalización: Guardar datos preferenciales del cuidado del adulto mayor.
Acciones: 1.- El usuario ingresa a la aplicación. 2.- El usuario ingresa a la opción “Ingresar Datos del cuidado”. 3.- El usuario ingresa lo “hora”, “nombre del medicamento”, “alarma de voz”. 4.- El usuario ingresa Guardar y Salir. 5.- El sistema muestra el resultado de la petición requerida.
Descripción de los datos de entrada: No necesita datos de entrada.
Salida esperada: Guardar preferencias del usuario a través de una función que retornaba un boolean.
Salida obtenida: True.
Resultado: Correcto.
Severidad: Grave, Mayor.
Seguimiento: contiene la o las líneas de código, en donde la prueba fallo.
Fecha: 17/06/2013

Tabla 5. 3 Caso de Prueba 003

5.2 Plan de Pruebas.

El plan de pruebas es una fase muy importante dentro de un proyecto, ya que permite comprobar la aplicación. Además se debe encontrar la mayor cantidad de errores y

fallos posibles al momento de la ejecución, por lo cual se cree que la mayor parte del tiempo y esfuerzo del proyecto está dedicado a la creación y realización de esta fase. Esta fase forma parte de una de las últimas etapas del proyecto, antes que la aplicación quede completamente funcional y lista para ser usada.

El plan de pruebas correspondiente a este proyecto, tendrá pruebas unitarias por cada uno de los requerimientos principales:

- ❖ Informar al usuario cualquier evento que conlleve la asistencia del adulto mayor.
- ❖ Informar al usuario el lugar donde se encuentra el adulto mayor.
- ❖ Permitirle al usuario planificar un cuidado que sea detallando y permita el grabado de voz.
- ❖ Permitirle al usuario tener un registro sobre todas las actividades del adulto mayor por otra parte tener un recuento sobre sus enfermedades y problemas.

El uso de los requerimientos funcionales como plan de pruebas, se debe a que corresponden a las necesidades propuestas por el cliente y aspectos básicos de cualquier aplicación

5.3 Calendario de Pruebas.

Nombre de la tarea	Fecha de inicio	Fecha de finalizaci	% Compl	Asignado a	Jun 9							Jun 16							Jun 23							Jun 30						
					M	M	J	V	S	D	L	M	M	J	V	S	D	L	M	M	J	V	S	D	L	M	M	J	V	S	D	
Casos de Prueba	15/06/13	17/06/13	100%	Daniel Toledo																												
Prueba 01: Informar al usuario el registro del teléfono y el guardar su ID en la BD	15/06/13	15/06/13	100%	Daniel Toledo																												
Prueba 02: Informar al usuario una notificación	16/06/13	16/06/13	100%	Daniel Toledo																												
Prueba 03: Permitirle al usuario guardar todos sus preferencias en la aplicación	17/06/13	17/06/13	100%	Daniel Toledo																												

Tabla 5. 4 - Calendario de Pruebas

Capítulo VI Conclusiones y Trabajo Futuro.

Durante este estudio se ha observado la importancia que tiene en la actualidad Android, creado en un principio para cumplir las necesidades básicas de un teléfono celular pero que con el tiempo y la intervención de código fuente en las aplicaciones ha ido evolucionando en útiles e innovadoras ideas. Es por esto que se ha decidido llevar a cabo este proyecto, donde se busca la creación de una aplicación intuitiva, innovadora e ingeniosa, para ampliar aún más el campo de esta tecnología que es Android y comprobar que tan poderosa y eficiente es esta herramienta que día a día abarca más seguidores en el mundo.

Para experimentar en dicha tecnología, fue necesario implementar todos los componentes que se necesitan, y por ende se instaló el SDK de Android para poder conectar directamente el Smartphone con la aplicación. También a través de la obtención del marco teórico se observó como hoy en día es tan esencial la plataforma de Android por casi su estandarización y en especial por su infinidad de aplicaciones ya sean gratis o pagadas las cuales tienen la funcionalidad desde abrir un automóvil como hablar por radio, ocupar mensajería de texto gratis hasta conectar el Smartphone al PC a través de Wi-Fi. Cabe resaltar que Android crece día a día a pasos agigantados, gracias a esto podemos decir que en un futuro no muy lejano se podría interactuar de manera natural con sus dispositivos y realizar cualquier tarea diaria de manera completamente transparente con respecto a sus dispositivos móviles, algo impensado hace una década atrás debido a que no existía la tecnología necesaria y principalmente porque casi nada era código abierto, hoy cualquiera puede hacer una aplicación y todo por el SDK de Android que presta sus librerías para poder crear todo tipo aplicación de cualquier índole y subirla a las App de Google, cabe mencionar que para que todo esto sea posible es necesario que la telefonía móvil siga creciendo como lo ha hecho hasta ahora, hay que tener en cuenta que hace una década ningún celular contaba ni siquiera con procesador y para qué hablar de memoria RAM o ROM, es importante visualizar que cada dos años las capacidades de todos los dispositivos móviles se duplican.

Una vez estudiado el marco teórico de ambas tecnologías salto, surgió dos grandes interrogantes, una de ella, era como conectar la aplicación con una base de datos externa, ya que la base de datos nativa de Android es muy pequeña y poco óptima, para solucionar este problema se optó por Servicios web o Webservice la cual presta la solución directa de dicho problema, ya que esta tecnología permite comunicar plataformas distintas con aplicaciones programadas en lenguajes distintos. La otra interrogante era si un Cuidador tiene más de un Smartphone donde instalar la aplicación, como se podría saber cuál de estos enviaría una notificación y poder distinguir uno del otro, por ende se optó en el servidor gratuito de Google, este se comunica con el Smartphone para otorgarle un ID o token para posteriormente guardarlo en la base de datos.

Sin lugar a dudas Android seguirá evolucionando con el tiempo, es probable que esta tecnología sea una de las que perdure más en el tiempo, como lo ha hecho Microsoft, pero depende de cada desarrollador en dejar un legado en aplicaciones novedosas y que sean de real aporte a la sociedad. El campo donde puede ser aplicada esta tecnología parece no tener fin y en un futuro no muy lejano Android empezará a ser parte de nuestras vidas,

siendo indispensable el uso de este debido a su alto potencial , bajo costo , y la facilidad de poder intervenir su código fuente.

Una de las cosas más importantes realizadas en este proyecto fue como se abordó el tema, debido a que no se contaba con un cliente real es por ello que la captura de requerimientos fue una tarea demasiado complicada, se logró juntar partes totalmente distantes entre sí, lo que es el cuidado y monitoreo constante de un adulto mayor con la tecnología de información, es por esto que estas últimas están llenando completamente las necesidades del ser humano, ya no son simplemente para jugar, oír, y buscar.

Por otro lado se espera a futuro la inserción a GooglePlay para así ser una aplicación gratis para todos los usuarios que tengan un android, en el caso que si se masificara se podría comprar un servidor propio para no depender de hosting por otro lado se podría habilitar a otra plataforma como por ejemplo IOS. También será de gran utilidad la valoración de los usuarios para poder realizar las actualizaciones correspondientes, debido a que el mejor test de calidad son los usuarios.

Anexos.

Anexo A.

A.1 Introducción a Dispositivos Móviles.

Los dispositivos móviles son aparatos de pequeño tamaño en comparación a computadores y notebooks, con algunas capacidades de procesamiento, con conexión permanente o intermitente a una red, con memoria limitada, diseñados especialmente para una función, pero que pueden llevar a cabo otras funciones generales.

A.2 Telefonía móvil.

A.2.1 Celulares Concepto General.

Definimos teléfono móvil o celular como un dispositivo electrónico de comunicación, normalmente de diseño reducido y sugerente y basado en la tecnología de ondas de radio (es decir, transmite por radiofrecuencia), que tiene la misma funcionalidad que cualquier teléfono de línea fija. Su rasgo característico principal es que se trata de un dispositivo portable e inalámbrico, esto es, que la realización de llamadas no es dependiente de ningún terminal fijo y que no requiere de ningún tipo de cableado para llevar a cabo la conexión a la red telefónica.

Además de ser capaz de realizar llamadas como cualquier otro teléfono convencional, un celular más o menos moderno suele incorporar un conjunto de funciones adicionales, tales como mensajería instantánea (SMS), agenda, juegos, etc., que aumentan la potencialidad de utilización de estos dispositivos.

Es más, su desarrollo y exigencia ha llegado a tal punto, que ya se puede hablar incluso de términos tales como memoria RAM y ligarlos al uso de móviles, dentro información de todo tipo (audio, video, texto, etc.), lo que hace de ellas un complemento perfecto tanto para el hombre de a pie como para el de negocios.

Su manejo suele ser sencillo, ya sea a través de la escritura sobre una pantalla (sensible el electrónicamente) mediante un "stick" específico, o de un pequeño teclado (su integración o no en el propio dispositivo es opcional).

A.2.2 Historia.

La telefonía móvil usa ondas de radio para poder ejecutar todas y cada una de las operaciones, ya sea llamar, mandar un mensaje de texto, etc., y esto es producto de lo que sucedió hace algunas décadas.

La comunicación inalámbrica tiene sus raíces en la invención del radio por Nikola Tesla en los años 1880, aunque formalmente presentado en 1894 por un joven italiano llamado Guglielmo Marconi.

El teléfono móvil se remonta a los inicios de la Segunda Guerra Mundial, donde ya se veía que era necesaria la comunicación a distancia, es por eso que la compañía Motorola creó un equipo llamado Handie TalkieH12-16, que es un equipo que permite el contacto con las tropas vía ondas de radio que en ese tiempo no superaban más de 600 kHz.

Fue sólo cuestión de tiempo para que las dos tecnologías de Tesla y Marconi se unieran y dieran a la luz la comunicación mediante radio-teléfonos: Martin Cooper, pionero y considerado como el padre de la telefonía celular, fabricó el primer radio teléfono entre 1970 y 1973, en Estados Unidos.

En 1979 aparecieron los primeros sistemas a la venta en Tokio (Japón), fabricados por la Compañía NTT. Los países europeos no se quedaron atrás y en 1981 se introdujo en Escandinavia un sistema similar a AMPS (Advanced Mobile Phone System). Y si bien Europa y Asia dieron los primeros pasos, en Estados Unidos, gracias a que la entidad reguladora de ese país adoptó reglas para la creación de un servicio comercial de telefonía celular, en 1983 se puso en operación el primer sistema comercial en la ciudad de Chicago. Este fue el inicio de una de las tecnologías que más avances tiene, aunque continúa en la búsqueda de novedades y mejoras.

Resumiendo, hace una década aproximadamente los teléfonos celulares se caracterizaban sólo por llamar, pero ha sido tanta la evolución que ya podemos hablar de equipos Multimedia que puede llamar y ejecutar aplicaciones, jugar juegos3D, ver videos, ver televisión y muchas cosas más. Obviamente muchas marcas de placas madres para PC o fabricantes de hardware en general se hacen presentes en los teléfonos móviles como por ejemplo: ASUS e INTEL que construyen las placas matrices de los celulares o ayudan con el acelerador gráfico o el sistema de video. En fin, debemos tener conciencia y prepararnos para lo que se viene más adelante y pensar que el teléfono celular ya no está sólo para hablar.

A.3 Sistemas Operativos Móviles.

Un sistema operativo móvil o SO móvil es un sistema operativo que controla un dispositivo móvil al igual que los PC utilizan Windows o Linux entre otros. Sin embargo, los sistemas operativos móviles son mucho más simples y están más orientados a la conectividad inalámbrica, los formatos multimedia para móviles y las diferentes maneras de introducir información en ellos. [2]

A.3.1 Capas.

A.3.1.1 Kernel.

El núcleo o kernel proporciona el acceso a los distintos elementos del hardware del dispositivo. Ofrece distintos servicios a las superiores como son los controladores o drivers para el hardware, la gestión de procesos, el sistema de los archivos y el acceso y gestión de la memoria.

A.3.1.2 Middleware.

Es el conjunto de módulos que hacen posible la propia existencia de aplicaciones para móviles. Esto talmente transparente para el Cuidador y ofrece servicios claves como el motor de mensajería y comunicaciones, códec multimedia, intérpretes de páginas web, gestión del dispositivo y seguridad.

A.3.1.3 Entorno de ejecución de aplicaciones.

El entorno de ejecución de aplicaciones consiste en un gestor de aplicaciones y un conjunto de interfaces programables abiertas y programables por parte de los desarrolladores para facilitar la creación de software.

A.3.1.4 Interfaz de Cuidador.

Las interfaces de Cuidador facilitan la interacción con el Cuidador y el diseño de la presentación visual de la aplicación. Los servicios que incluye son el de componentes gráficos (botones, pantallas, listas, etc.) y el del marco de interacción.

Aparte de estas capas también existe una familia de aplicaciones nativas del teléfono que suelen incluir los menús, el marcador de números de teléfono etc.

A.3.2 Mercado.

A medida que los teléfonos móviles crecen en popularidad, los sistemas operativos con los que funcionan adquieren mayor importancia. La cuota de mercado de sistemas operativos móviles a mediados de 2012 era el siguiente:

- ❖ Android 68,1% (En países como España las diferencias son más significativas, donde Android tiene el 87% de la cuota de mercado)
- ❖ iOS 16,9%
- ❖ BlackBerryOS 4,8%
- ❖ Symbian OS 4,4%
- ❖ Windows Phone y Windows Mobile 3,5 %
- ❖ Linux u otros 2,3% [3]

Anexo B.

B.1 Factibilidad técnica.

La factibilidad técnica, consistió en realizar una evaluación de la tecnología requerida para la implementación del sistema propuesto y de ser necesarios los requerimientos tecnológicos que deben ser adquiridos para el correcto funcionamiento de la misma.

Principalmente en lo que respecta a tecnología se han dividido en 2 tipos para el desarrollo del sistema, estas son las siguientes:

B.1.2 Hardware.

En lo que implica al Hardware del proyecto en sí, corresponde al computador a utilizar, este debe ser lo suficientemente potente para poder compilar los códigos que se desarrollen en los SDK de Android que utiliza el software de eclipse y webservice que utiliza netbeans.

En este proyecto no es necesario nombrar al servidor puesto que se arrendara un hosting el cual nos brindara con ilimitada capacidad y transferencias de archivos.

B.1.2.1 Requerimientos de hardware para Eclipse.

Los requerimientos mínimos de hardware para poder ejecutar Eclipse son los mismos que requiere Windows 7:

- ❖ Procesador de 32 bits (x86) o 64 bits(x64) a 1 gigahercio (GHz) o más.
- ❖ Memoria RAM de 1 gigabyte (GB) (32 bits) o memoria RAM de 2 GB (64 bits).
- ❖ Espacio disponible en disco rígido de 16 GB (32 bits) o 20 GB (64 bits).
- ❖ Dispositivo gráfico DirectX 9 con controlador WDDM 1.0 o superior. [9]

B.1.2.2 Requerimientos de hardware para Netbeans.

Los requerimientos mínimos de hardware para poder ejecutar Netbeans son los mismos que requiere Windows 7:

- ❖ Procesador de 32 bits (x86) o 64 bits(x64) a 1 gigahercio (GHz) o más.
- ❖ Memoria RAM de 1 gigabyte (GB) (32 bits) o memoria RAM de 2 GB (64 bits).
- ❖ Espacio disponible en disco rígido de 16 GB (32 bits) o 20 GB (64 bits).
- ❖ Dispositivo gráfico DirectX 9 con controlador WDDM 1.0 o superior. [9]

B.1.3 Software.

En el apartado anterior se ha descrito los requerimientos que debe tener el computador para poder compilar los códigos fuentes tanto para Android, WebService. Ahora a continuación se describe todo lo necesario con respecto al software utilizados para el desarrollo de la aplicación.

B.1.3.1 SDK de Android.

El software es muy intuitivo, ofrece una estructura de datos orientada a objeto. Con esta estructura jerárquica se pueden crear una amplia gama de bibliotecas para el desarrollo de aplicaciones. El SDK de Android soporta una gran gama de lenguajes, principalmente se programa en java, pero no implica que no sea compatible C, C++ entre otros.

B.1.3.2 WebService.

Web service o Servicios web es simplemente un estándar para intercambiar datos entre aplicaciones. Al ser un estándar, no importa el lenguaje en el que se implemente ni en qué sistema operativo se ejecute ya que se respeta el estándar es como si las aplicaciones hablaran el mismo idioma.

Un servicio web se parece en muchos aspectos a una **página web** tradicional y aquí menciono sus similitudes:

- ❖ Ambas trabajan en el modelo cliente-servidor dónde el cliente es el Cuidador que consume (página web o servicio web) y el servidor es quien provee (ofrece el contenido).
- ❖ Ambas por lo general corren sobre el puerto 80.
- ❖ Ambas utilizan el protocolo de comunicación HTTP.
- ❖ Ambos servidores reciben una petición y de acuerdo a eso producen un resultado.
- ❖ Ambos clientes envían peticiones a los servidores.

Para este proyecto se desarrollara un Web service en Aplicación web java.

B.1.3.3Java.

Java es un lenguaje de programación de alto nivel orientado a objetos, desarrollado por James Gosling en 1995. El lenguaje en sí mismo toma mucha de su sintaxis de C, Cobol y Visual Basic, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria. La memoria es gestionada mediante un recolector de basura.

Las aplicaciones Java están típicamente compiladas en un bytecode, aunque la compilación en código máquina nativo también es posible. En el tiempo de ejecución, el bytecode es normalmente interpretado o compilado a código nativo para la ejecución, aunque la ejecución directa por hardware del bytecode por un procesador Java también es posible.

Entre diciembre de 2006 y mayo de 2007, Sun Microsystems liberó la mayor parte de sus tecnologías Java bajo la licencia GNU GPL, de acuerdo con las especificaciones del Java Community Process, de tal forma que prácticamente todo el Java de Sun es ahora software libre aunque la biblioteca de clases de páginas web comprendidas en las librerías de obtención de objetos para ser compilados como aplicaciones comprimidas no están totalmente acopladas de acuerdo con Sun que dice que se requiere un intérprete para ejecutar los programas de Java. [8]

B.2 Factibilidad técnica.

Para que la implementación de nuestra App y aplicación web se realice de la mejor forma, se contara con un gran servidor arrendado en justhost.com, el cual cuenta con ilimitadas bases de datos e ilimitada capacidad de transferencia de archivos. Dicho servidor soporta todas las características de nuestra App y sistema web a implementar.

Antes de contratar el servicio se probara localmente la App y el Sistema web y serán mostrados los prototipos no funcionales, y funcionales terminado parcial al profesor guía, para posteriormente montar el sistema funcional completo en el servidor de justhost.com.

B.3 Factibilidad legal.

El sistema es desarrollado bajo las licencias de software libres como Netbeans, Eclipse, Postgresql, Licencia Apacha 2.0, Licencia de software libre como Linux Distribución Ubuntu.

B.4 Análisis de Riesgo.

Para conseguir un buen análisis de riesgo es necesario identificar los posibles riesgos que puede tener el desarrollo del sistema. Una vez identificado los posibles riesgos se seleccionan los 5 principales o más importantes y se ordenan en orden de prioridad para luego calcular el factor de que tanto impacto podría generar ese riesgo en el proyecto.

B.4.1 Orden de prioridad riesgos más importantes.

- ❖ Que el sistema no cumpla con los objetivos planteados.
- ❖ No existe un lugar físico adecuado para ocupar en el desarrollo de las fases.
- ❖ El ordenador y el Smartphone o cualquier parte del hardware del desarrollador se estropean gravemente
- ❖ Obtención del marco teórico y estado del arte incorrecto.
- ❖ Mala Programación de las actividades.

Riesgo	Probabilidad	Impacto	Exposición	Prioridad
El sistema no cumpla con los objetivos	3	4	12	1
No existe un lugar físico adecuado para ocupar en el desarrollo de las fases.	2	3	6	2
El ordenador y el Smartphone o cualquier parte del hardware del desarrollador se estropean gravemente	1	4	4	3
Obtención del marco teórico y estado del arte incorrectos	1	2	2	4
Mala programación de las actividades.	1	2	2	5

Anexo B. 1 Tabla de análisis de riesgos y grado de exposición

En la figura B.1 se puede observar que el riesgo con mayor a exposición y que puede afectar el normal desarrollo del proyecto corresponde al riesgo número 1, donde con un factor de exposición de 12 es el mayor indicador de la tabla. Para calcular el factor de exposición solo basta con multiplicar la probabilidad de que ese riesgo ocurra por el impacto que este pueda tener. Importante mencionar que el rango de los factores es de 1 hasta 5.

Posteriormente se crea un plan de mitigación de riesgos donde se crean las estrategias necesarias y la forma de actuar del equipo de trabajo para que estos no ocurran y tengan un impacto sobre el proyecto. Por ejemplo verificar constantemente los requerimientos y objetivos del sistema. En caso de no ser así, aún se puede estar a tiempo de adaptarse a los cambios y cambiar la planificación. Esto causaría menos daño en comparación en pleno desarrollo del software. Para esto es esencial obtener el marco teórico de manera correcta para el éxito del proyecto.

Riesgo	Exposición	Medidas de Mitigación
El sistema no cumpla con los objetivos	12	Seguir la planificación, estar preparado a los cambios en los requerimientos según como se vaya desarrollando el proyecto.
No existe un lugar físico adecuado para ocupar en el desarrollo de las fases.	6	Solicitar al profesor las dependencias de la universidad para tener un mejor entorno para el desarrollo del software.
El ordenador y el Smartphone o cualquier parte del hardware del desarrollador se estropean gravemente.	4	Respaldar periódicamente en pendrives y/o páginas tales como Dropbox.
Obtención del marco teórico y estado del arte incorrectos	2	Seguir la planificación del proyecto, consultar a fuentes confiables y oficiales, Búsqueda de información en la literatura disponible.
Mala programación de las actividades.	2	Construcción de una carta Gantt, Registrar los hitos del proyecto.

Anexo B. 2 Plan de mitigación de riesgos

En la Tabla B.2 se muestra los riesgos principales del proyecto, además del factor de exposición calculado en la tabla anterior ordenado de mayor a menor y posteriormente una especificación de que medidas de mitigación se pueden realizar para evitar que ocurra los riesgos.

Referencias.

- [1] Dykinson, L.-E. (2006). *La informática, presente y futuro en la sociedad*. s.l
- [2] Editores, M. (2010). N.º 47. *Revista Gadgets* . s.l
- [3] Gartner,G. (2012). *Sala de Prensa*. <http://www.gartner.com/it/page.jsp?id=1924314>
- [4] Herrera, P. C. (2011). *Diseño e implementación del juego Super Pang*. s.l
- [5] Huddleston, R. (03-08-2012). *Android Fully Loaded*.John Wiley & Sons. s.l
- [6] Kenneth E. Kendall, J. E. (2005). *Analisis Y Diseño de Sistemas*. Pearson Educación. s.l
- [7] Lewis, S. M. s.f *Enfermeria Medicoquirurgica*. Elsevier España.
- [8] Llinás, L. F. (2011). *Programación orienta a objetos en Java*. Universidad del Norte.
- [9] Microsoft. (2012). *Miicrosoft Corporation*. <http://www.microsoft.com>
- [10] Montero, R. (2012). *Java Hispano*. <http://www.javahispano.org/android/2012/6/18/mecanismos-para-el-envio-de-notificaciones-a-los-usuarios-de.html>
- [11] WORTHAM, J. (2009). *Bits*. <http://bits.blogs.nytimes.com/2009/12/15/is-the-google-phone-an-unauthorized-replicant/>