

Desarrollo de una Aplicacion de Gestion de Comunicaciones para dispositivos moviles en la plataforma Android

Jorge Eloy Alan Atala

Universidad Tecnológica Nacional – Facultad Regional Córdoba, Córdoba, Argentina
alanatala@gmail.com

Abstract. En el marco de la carrera Ingenieria en Sistemas de Informacion, la materia electiva Tecnología de Desarrollo de Aplicaciones Moviles brinda el conocimiento necesario para llevar adelante un proyecto en la plataforma Android durante todo el semestre. Aprovechando las distintas características generales que proporciona este sistema gestor de comunicaciones, se avanza a pasos firmes en el desarrollo clase a clase en la cátedra, presentando las soluciones óptimas para cada nuevo detalle de implementación que se va presentando. De esta manera se van adquiriendo paulatinamente los conocimientos que son el objetivo último del curso. El gestor de comunicaciones consiste en una aplicación capaz de administrar los contactos del usuario móvil, de una manera en la cual este acceda de manera intuitiva a efectuar acciones con estos, ya sean llamadas, envío de mensajes de texto, emails y una cuarta alternativa de comunicación denominada Web Mail, la cual consiste en comunicarse a un servidor desarrollado por los responsables de la cátedra con el fin de enviar y recibir mensajes por ese canal, entre aquellos usuarios que estén registrados. Es importante destacar, que la capacidad de la aplicación consiste en agrupar todas estas funcionalidades que están presentes en el la plataforma Android y reunirlos en una sola aplicación en la que converjan de manera simple e intuitiva.

Keywords. Android, plataformas móviles, administración de contactos comunicaciones

1 Introducción

En el mundo actual, estar comunicado es casi una necesidad primaria en los seres humanos, y si a esto le sumamos el hecho de que mantenemos ocupado casi todo nuestro tiempo disponible en múltiples actividades, se hace evidente la importancia de contar con una manera eficaz de interactuar con nuestro entorno y con nuestros pares. A partir de esta premisa es que comenzó el desarrollo de los dispositivos móviles, hace alrededor de tres décadas, y en este lapso de tiempo se fue ampliando la cantidad de usuarios hasta llegar a la actualidad, donde es normal ver personas con varios dispositivos móviles en el día a día.

Pero este aumento en la cantidad de usuarios vino acompañado también por el desarrollo tecnológico que permitieron los grandes avances en la electrónica, el cual multiplicó exponencialmente las capacidades de procesamiento de información de estos dispositivos, además de permitió darles mayor versatilidad y funcionalidad. Los dispositivos móviles actuales tienen la capacidad de procesamiento que poseían las computadoras de escritorio de alta gama de hace solo entre 5 y 10 años, con lo cual se han convertido en verdaderos equipos móviles capaces de reemplazar a la PC tradicional en muchas de sus funciones. Este avance a determinado que muchas grandes empresas dedicadas a la producción de software se volcaran al desarrollo de aplicaciones móviles, lo cual se ha visto beneficiado además por los avances en los sistemas operativos los cuales brindan mayores facilidades de diseño y programación, lo que ha permitido que miles de desarrolladores de software más pequeños tengan acceso a este nuevo mundo de aplicaciones que conviven en un espacio móvil, con todo lo que esto implica. Tomando este contexto como base, es que se plantea una solución que facilite y agilice la confluencia de información a la que una persona necesita acceder diariamente, desarrollando una aplicación para la plataforma móvil Android, la cual se ha convertido en una de las más utilizadas mundialmente. De esta forma podemos decir entonces que a través de su utilización tendremos acceso a un porcentaje importante de usuarios de aplicaciones móviles.

2 Plataforma Android

2.1 Breve historia

La plataforma Android surgió como una propuesta de Google Inc., A mediados de 2005 surgió el rumor[1] de que Google planeaba desarrollar un dispositivo móvil de distribución gratuita con el cual esperaba beneficiarse mediante publicidades en las búsquedas personales. Sin embargo, este rumor se convirtió en algo mucho más revolucionario, un sistema operativo móvil open source, que además, estaba respaldado por este gigante de Internet. La primera versión beta del SDK apareció en noviembre de 2007 y para septiembre de 2008 se liberó finalmente la versión 1.0 del SDK lo cual fue el puntapié inicial para la popularización de los Smartphone que corrían la plataforma Android.

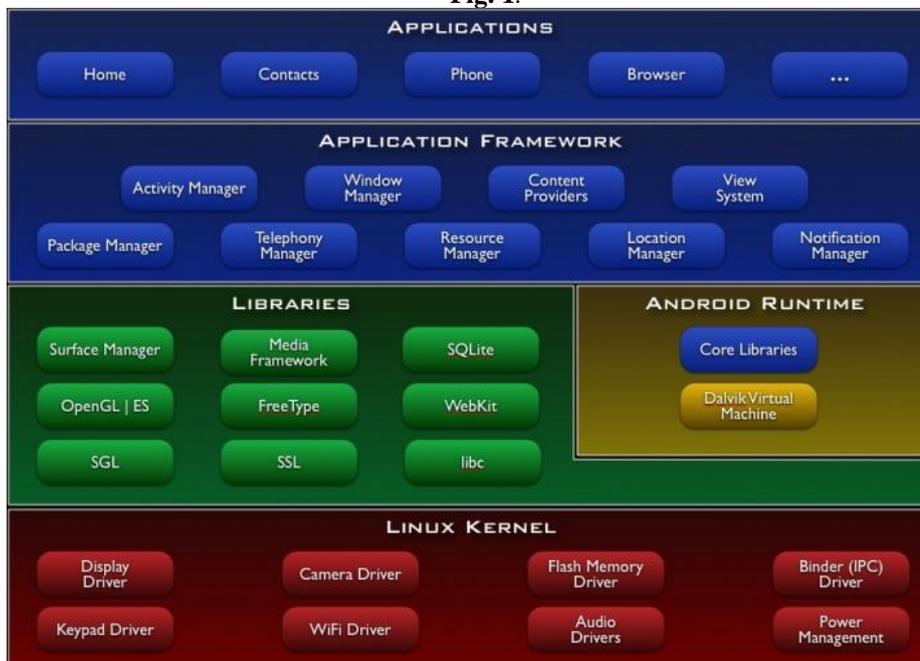
Sin embargo el hecho de ser un sistema open source, le abrió las puertas a los desarrolladores para crear versiones capaces de correr en cualquier tipo de dispositivos, pudiéndose encontrar por internet hasta cafeteras a las que se les ha instalado Android, aunque esta versatilidad también tiene un problema, y es que hay tantas versiones distintas de Android que el usuario común no es capaz de reconocer cual es la que debe utilizar para actualizar su dispositivo, pues cada fabricante crea una versión especial para el suyo. Ante esta situación, Google ha enfocado sus esfuerzos en crear una línea de versionado "oficial" de su sistema. Actualmente el desarrollo se encuentra en la versión 2.3 conocida como "GingerBread" para Smartphones y la 3.0 también conocida como "HoneyComb" para tabletas, puesto que esta empresa nombra con algún tipo de postre a cada una de sus versiones de Android[2]. "HoneyComb"

fue la respuesta de Google a la gran demanda de estos dispositivos conocidos como tabletas, los cuales estaban utilizando versiones anteriores adaptadas.

Pero lo que realmente estableció una posición de liderazgo para Android no fueron sus capacidades técnicas, sino que fue una idea que ya estaba siendo explotada por el sistema operativo IOS de Apple en las diferentes versiones de su Smartphone Iphone. Crear un mercado de aplicaciones oficial, al cual todos los usuarios pueden acceder y adquirirlas para responder a sus necesidades, marcó un nuevo paradigma, ya que anteriormente, para encontrar una aplicación, era necesario utilizar buscadores, invirtiendo una buena cantidad de tiempo, y una vez encontrada, no se estaba seguro de que funcionase correctamente. Ahora uno accede al Android Market y puede ver las aplicaciones en un entorno amigable, separadas por categorías y leer opiniones de usuarios sobre las mismas, además de tener información sobre la versión necesaria para poder correrla. Esto no fue solo una ventaja para los usuarios, sino también para los desarrolladores, quienes ahora se aseguraban una vidriera para mostrar y vender sus productos, es decir, pasaron de depender de la suerte de que les cayera un potencial cliente a sus páginas web, a asegurarse un flujo mucho mayor en una gran tienda on-line.

Este beneficio mutuo de publicadores de aplicaciones y usuarios, ha resultado en que el Android Market disponga a la fecha de Mayo de 2011 de 206680 aplicaciones[3].

Fig. 1.



2.2 Características técnicas

Como vemos en la Fig. 1, Android consiste en una maquina virtual de Java diseñada por Google, la cual han llamado Dalvik y corre sobre un kernel de Linux. Estrictamente hablando Dalvik no es una maquina virtual de Java, porque el bytecode generado no es Java bytecode, lo cual no significa un gran problema, ya que es posible portar archivos .Class a Dalvik.

Como se puede observar en la Fig. 1 obtenemos una abstracción del hardware a través del kernel de Linux, el cual nos provee drivers para acceder a los distintos periféricos del dispositivo, luego hay una capa de librerías, en el mismo nivel donde está la VM Dalvik. Estas librerías son utilizadas por varios componentes del sistema Android, y les permiten a los desarrolladores acceder a ciertas características que de otro modo serian difíciles de implementar, como por ejemplo, codificación de videos, manejo de bases de datos o conexiones de red. Dalvik esta optimizada para manejar múltiples instancias de la VM, ya que cada aplicación corre en un proceso con su propia maquina virtual. Dalvik ejecuta archivos del tipo Dalvik Ejecutable (.dex) por lo que los archivos .class de Java que necesitemos utilizar deben ser transformados. Este formato ocupa mucho menos espacio en memoria, puesto que está pensado para aplicaciones móviles, donde tenemos menos capacidad de almacenamiento.

Por encima de esto existe un framework de aplicaciones, el cual es la API de Android, estos componentes abstraen al desarrollador de aquellas características comunes entre aplicaciones, logrando la reutilización y la disminución de tiempos de desarrollo. Finalmente llegamos a la capa de aplicaciones, en la cual conviven todas las aplicaciones de desarrolladores que se nutren de las capas inferiores para su funcionamiento[4].

3 El Gestor de comunicaciones (Communication manager)

3.1 Definición:

La definición formal de la aplicación, que representa el trabajo práctico integrador de la cátedra Tecnología de Desarrollo de Aplicaciones Móviles, consiste en un conjunto de funcionalidades necesarias para cumplimentar el objetivo, el cual se define a continuación:

Objetivo: *Desarrollar una aplicación en la plataforma Android que le permita al usuario comunicarse con sus contactos de teléfono a través de: llamadas telefónicas, envío de mensajes de texto, envío de mails y envío de mensajes a un servicio disponible vía web.*

El conjunto de requerimientos solicitados es el siguiente:

Funcionales:

- La aplicación debe cargar y mostrar los contactos del teléfono
- La aplicación debe permitir realizar llamadas telefónicas y enviar mensajes de texto a un contacto. Considerar que un contacto puede tener un número indefinido de números de teléfono, incluso ninguno.
- La aplicación debe permitir enviar emails a un contacto. Considerar que un contacto puede tener un número indefinido de cuentas de email, incluso ninguna.
- La aplicación debe permitir enviar mensajes al contacto utilizando un servicio disponible vía web. (Este punto utiliza un servicio web de mensajería desarrollado por la cátedra, es la base del modulo Web Mail detallada más adelante)
- La aplicación debe notificar a través del Notification Manager el resultado de envío de los mensajes utilizando el servicio web.
- La aplicación debe registrar y poder mostrar cada una de las acciones realizadas con un contacto: llamadas telefónicas, mensajes de texto enviados, mails enviados y mensajes enviados a través del servicio web; especificando fecha y hora, nombre del contacto y tipo de acción realizada.
- La aplicación debe permitir ver el detalle de los mensajes enviados a través del servicio web a un contacto.
- La aplicación debe ser capaz de registrar los cambios en el estado de conectividad de datos del equipo móvil, inclusive cuando no esté el foco del usuario en ella.
- La aplicación debe permitir enviar mensajes utilizando el servicio web sólo cuando exista conectividad de datos.
- La aplicación debe proporcionar al usuario la posibilidad de definir filtros al momento de presentar los contactos, los registros de acciones y los mensajes recibidos por el servicio web, los cuales se deben mantener una vez que se ha cerrado la aplicación.
- La aplicación debe proporcionar al usuario la posibilidad de definir criterios de ordenamiento al momento de presentar los contactos, los registros de acciones y los mensajes recibidos, los cuales se deben mantener una vez que se ha cerrado la aplicación.
- La aplicación debe solicitar que el usuario registre un nombre de usuario en el servicio de envío de mensajes para que pueda utilizarlo y permitirle cambiarlo en cualquier momento.
- El servicio de la aplicación debe iniciarse cuando el usuario abre la aplicación y ésta no se encontraba previamente abierta
- La aplicación debe permitir eliminar registros de acciones realizadas por el usuario
- La aplicación debe ejecutar polling sobre el servicio web para consultar si hay mensajes disponibles y guardarlos.
- La aplicación debe mostrar los mensajes recibidos a través del servicio web especificando: fecha y hora, usuario que lo envió y el detalle del mensaje.
- La aplicación debe permitir eliminar registros de mensaje recibidos por el usuario
- Se debe incorporar alguna funcionalidad proporcionada por la plataforma Android que se considere apropiada para la aplicación y que incorpore valor agregado.

No funcionales:

- El acceso a las preferencias debe realizarse a través de un menú de opciones
- Los contactos y los registros de acciones realizadas y mensajes recibidos deben ser presentados en formato de lista. Adicionalmente pueden incorporarse otros datos que se consideren pertinentes además de los planteados en los requerimientos funcionales.
- La aplicación debe mantener el estado de las ventanas a través de las rotaciones
- La aplicación debe proporcionar soporte para internacionalización
- Durante la ejecución de la aplicación no se deben generar mensajes enviados por el sistema del tipo “Waiting”
- La aplicación debe ser usable y “User friendly”.
- Las operaciones realizadas contra el servidor no deben impedir que el usuario siga utilizando la aplicación y el equipo móvil.
- La aplicación debe implementarse como “Single task”

3.2 Descomposición modular:

Partiendo del requerimiento no funcional de presentar la información en listas, se realizó una interfaz que permitiera una interacción fluida con las mismas. También se buscó una interacción dinámica entre las distintas pantallas, reduciendo la utilización de los botones físicos al máximo, de esta manera el usuario rápidamente se adapta a

Fig. 2.

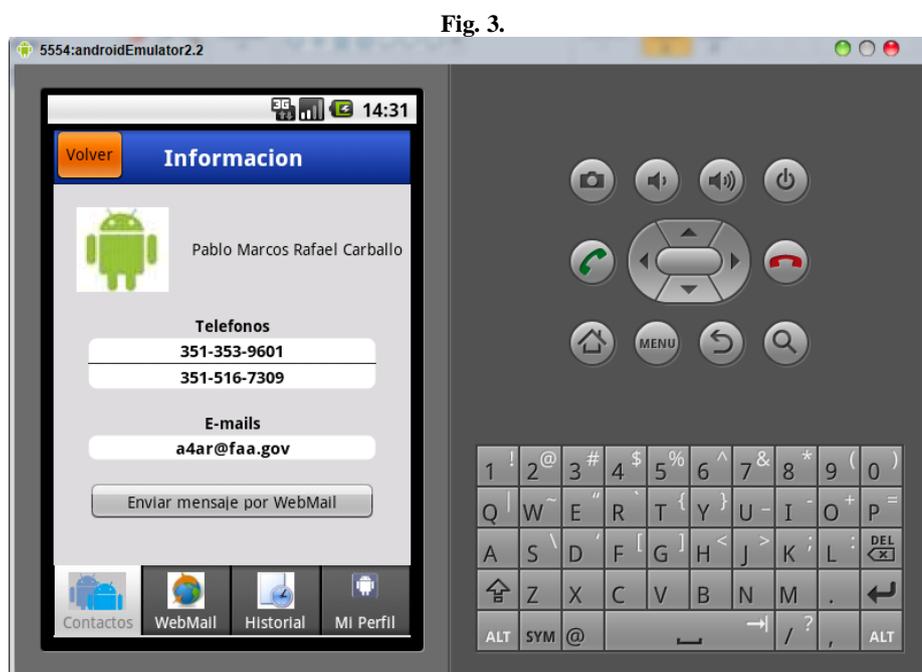


las características de la aplicación, y debido a la distribución lógica de las funciones, le resulta intuitivo encontrarlas. Esta distribución está definida por 4 módulos que son: Contactos, Historial, Web Mail e Información Personal. Estos grupos de funciones están definidos como pestañas y son accesibles en todo momento, reduciendo la cantidad de pasos que el usuario debe realizar en caso de querer acceder a alguno de ellos. En la siguiente captura del emulador de Android, puede verse el módulo de Contactos y en el extremo inferior se observan las cuatro pestañas mencionadas anteriormente.

En lo que respecta al diseño del software, la necesidad de cumplir con tiempos muy acotados y de ir conociendo como desarrollar en la plataforma Android en forma paralela al diseño y la implementación, determinó la necesidad de un desarrollo modular, basado en la implementación de la interfaz completa en primera instancia, para luego agregarle funcionalidades paulatinamente, siguiendo el orden de aprendizaje de temas de la currícula de la materia. Una vez que se afianzaron los primeros conceptos, fue posible definir ciertas características importantes de la arquitectura, para luego poder comenzar a implementar los distintos módulos que componen la aplicación.

3.3 Descripción de la interacción y funcionamiento de interfaces

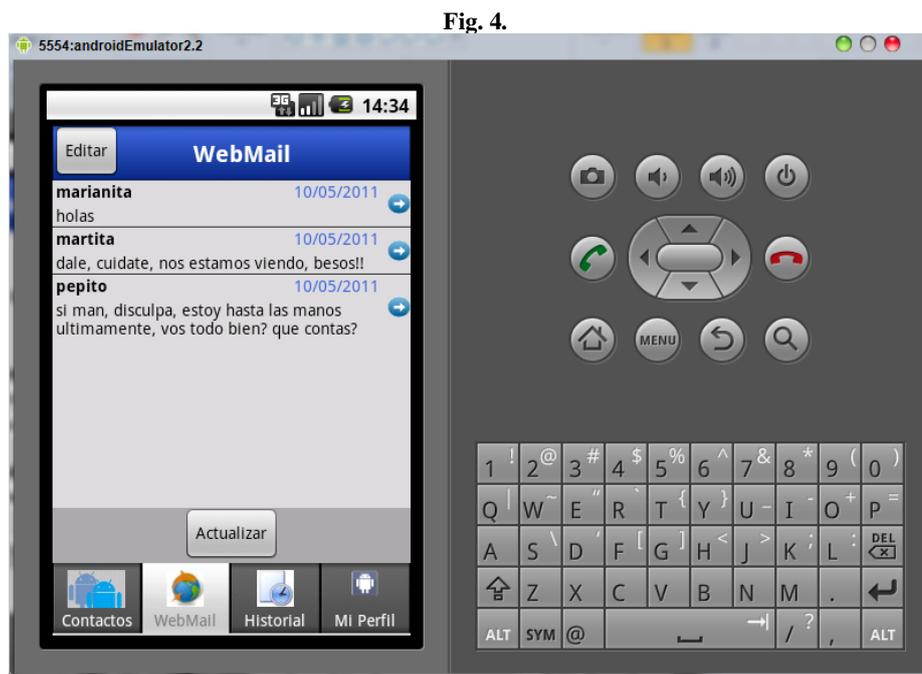
En la Fig. 2, se presenta la interfaz del módulo de contactos, en la cual se muestra



la lista de contactos, la cual puede ser filtrada utilizando el cuadro de ingreso de texto superior, y al seleccionar un contacto se accede a la información particular del mismo, como se puede ver en la Fig. 3.

Desde esta interfaz se puede llamar, enviar un SMS, enviar un mail o enviar un mensaje a través del servicio Web al contacto seleccionado. El cambio entre la lista de contactos y los datos de uno en particular, esta manejado por el módulo de Contactos, el cual tiene definidas estas dos interfaces y hereda este comportamiento y todos los relacionados a listas de una clase que los define. Esto se ha planteado de esta forma para que todos los módulos que fueron creados mantengan la consistencia. Particularmente para el manejo de listas se utilizó programación genérica (Generics), puesto que los contenidos en las listas de los distintos módulos son de diferente naturaleza.

En la Fig. 4 podemos ver el módulo de Web Mail,



con el cual tenemos acceso a los mensajes que ya han sido transferidos desde el servidor Web al dispositivo, en la interfaz principal se listan los hilos de conversación, mostrando parte del último mensaje realizado en cada uno de ellos. Al seleccionar uno en particular se accede a la conversación, como vemos en la Fig. 5,

Fig. 5.

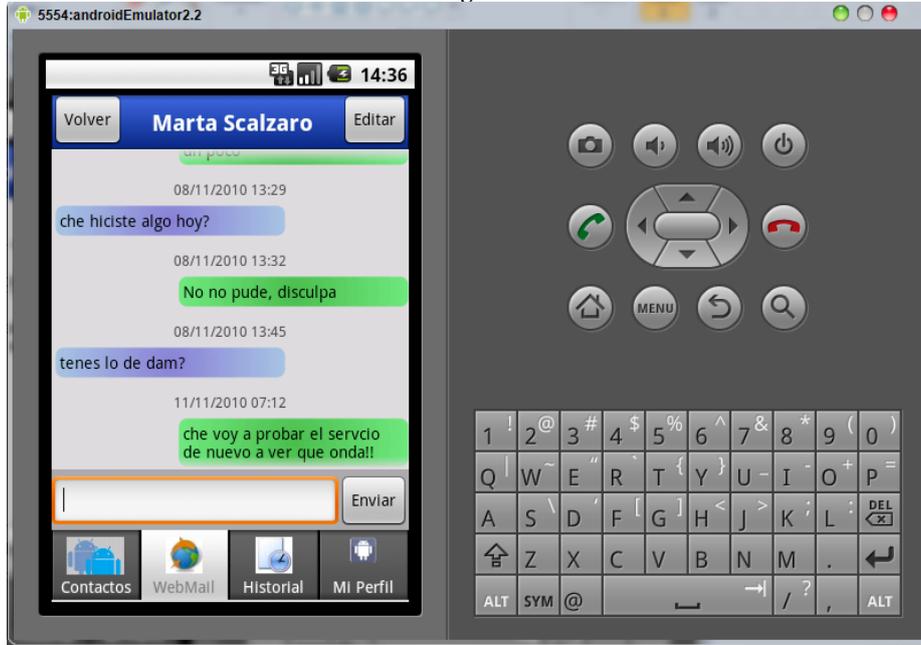


Fig. 6.



en la cual tenemos acceso a todo el historial de mensajes intercambiados con un contacto en particular, pudiendo editar esta lista y la anterior simplemente con el botón correspondiente en la esquina superior derecha. Cabe destacar que la asociación de un contacto con un usuario del servicio Web Mail no se hizo en una base de datos de la aplicación, sino que se utilizó la ventaja que ofrece la plataforma Android de crear *customs fields* a la definición de usuario de la base de datos del mismo dispositivo, por lo tanto la relación es instantánea, ahora el nombre de usuario de Web mail del contacto es un campo más, al igual que su número de teléfono o su dirección de mail.

El siguiente es el módulo de Historial, donde accedemos a las distintas listas de acciones realizadas con los contactos, las cuales brindan información del sentido de la acción (básicamente si fue saliente o entrante al dispositivo) y nos permite eliminar las acciones. Esta eliminación no se realiza en la aplicación, sino que se accede a la base de datos del teléfono y se elimina el registro de una determinada llamada, SMS o mail.

Por último, el módulo de Información personal, está planteado como la vista de cada contacto individual, sólo que desde aquí podemos ver nuestros datos personales y configurar algunos ajustes globales de la aplicación, como por ejemplo cuál va a ser el criterio de ordenamiento de los contactos, o el filtro del historial. Estos ajustes están definidos en las preferencias de la aplicación, por lo tanto persisten a distintas instancias de la misma.

Además de las interfaces, la aplicación carga servicios independientes en el sistema Android. El más importante de ellos es el que se encarga de consultar periódicamente si hay nuevos mensajes en el servidor para ser descargados, de esta manera se evita una gran carga de la red, porque se van bajando los mensajes en pequeños lotes y no todos de una sola vez cuando el usuario decide hacerlo manualmente. Una vez que adquiere estos mensajes el servicio no depende de que la aplicación este siendo utilizada para poder avisarle de este evento al usuario, sino que aprovecha el sistema de notificaciones que provee Android para tal fin.

4 Conclusiones

Es posible destacar que el desarrollo de esta aplicación ha significado un conocimiento acabado de los planes de estudio de la cátedra Tecnología de Desarrollo de Aplicaciones Móviles, dejando abierta la posibilidad de continuar el proyecto para ponerlo en producción y comercializarlo, ya que sus características técnicas y de calidad son comparables a la gran parte de las aplicaciones disponibles en el Android Market. Teniendo en cuenta que el cursado de la cátedra es semestral, resulta relevante destacar el nivel de conocimiento adquirido desarrollando en la plataforma An-

droid al finalizar el curso, siendo esta aplicación el punto cúlmine y la puesta en práctica de los mismos.

Si bien la aplicación cumple con los requerimientos solicitados, se puede observar que es posible agregarle muchas más funcionalidades con el fin de que se convierta en un administrador completo, relegando la necesidad del usuario de interactuar con otras aplicaciones. Sin embargo, la cantidad de opciones y funciones que ofrece es superior a otras de similares características. Otro punto pendiente es el de mejorar el servicio Web, para que este alcance ciertas características de seguridad y calidad, que son características necesarias para asegurar el óptimo funcionamiento de la aplicación.

Finalmente el tiempo dedicado al gestor de comunicaciones ha sido una inversión en conocimiento y me abre la puerta al desarrollo de aplicaciones futuras en la plataforma Android con el plus de la experiencia adquirida.

Referencias.

1. Wikipedia Android <http://es.wikipedia.org/wiki/Android>
2. Android <http://www.android.com/media/>
3. Android Market Place <https://market.android.com/>
4. Comunidad de desarrolladores Android <http://developer.android.com/guide/basics/what-is-android.html>