



Introducción al desarrollo para Android

Grupo K-Demy

Introduction to development for Android

Resumen

Android, el sistema operativo para dispositivos móviles de Google, es un software que ha dado mucho de qué hablar en el mundo entero, por el lugar que ahora ocupa y en tan corto tiempo para la industria de la tecnología móvil y la sociedad en general.

Este artículo hace una introducción al mundo de Android y en especial al desarrollo de aplicaciones para este sistema operativo. Se enfatiza en las pautas y conocimientos necesarios para ser aplicada por el usuario final, ayudando a leer y entender cómo funciona un teléfono móvil con Android.

Palabras clave: : *Android, SDK, java, móviles, plataforma móvil*

Abstract

Android, the mobile operating system from Google, is software that has given so much to talk about in the world, the place now occupied by so fast and in time for the mobile technology industry and society in general.

This article is an introduction to the world of Android and especially the development of applications for this operating system, but will emphasize patterns and knowledge necessary for developing applications for the end user, reading can help a user understand how operates a mobile phone with Android.

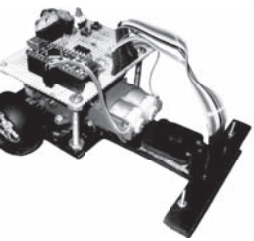
Key words: *Android, SDK, java, mobile, mobile platform.*

* K-demy Grupo de investigación en software y desarrollo e implementación de nuevas tecnologías de la información de la Escuela Tecnológica Instituto Técnico Central.

Carlos Andrés Pérez López: Estudiante de Sistemas Escuela Tecnológica Instituto Técnico Central. Correo electrónico: carlos.perez@k-demy.org

Carlos Eduardo Velásquez Villada: Ingeniero Electrónico Universidad Nacional, Magister en ingeniería Universidad de los Andes. Docente programa de sistemas Escuela Tecnológica Instituto Técnico Central. Correo electrónico: carlos.velasquez@k-demy.org

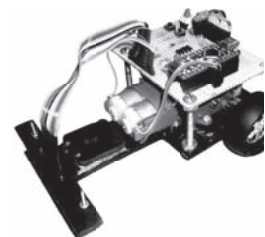
Sócrates Rojas Amador: Ingeniero de sistemas Universidad Distrital, especialista en gestión de proyectos de ingeniería Universidad Distrital. Docente programa de sistemas Escuela Tecnológica Instituto Técnico Central. Correo electrónico: socrates.rojas@k-demy.org



1. Introducción

Android, el sistema operativo para dispositivos móviles que está transformando la industria a nivel mundial, una apuesta por varias empresas para presentar un mejor producto y servicio a los usuarios finales de teléfonos inteligentes y tablets, con una característica fundamental: ser un producto de software libre.

A continuación, se presenta una reflexión sobre el sistema Operativo Android, cual ha sido su evolución histórica para convertirse en el sistema más utilizado en países como Estados Unidos, sus ventajas, arquitectura, describiendo las capas que lo componen, su anatomía y se finaliza con algunas sugerencias para su programación.



2. Sistema Operativo Android y su desarrollo histórico

Cuando se habla de Android o de otro software, es preciso diferenciar los conceptos de software y sistema operativo. El Software o “Programa” es lo intangible o capa lógica que está sobre otra capa física que puede ser un computador, teléfono inteligente, tablets o televisor y que permite obtener funcionalidades sobre ella.

Por su parte el “Sistema Operativo” es el conjunto de programas que efectúan la gestión de los procesos básicos de un sistema informático, y permiten la normal ejecución de las operaciones básicas del componente físico donde está instalado. A este grupo pertenece Android, siendo un sistema operativo para dispositivos móviles, aunque también existe para televisores, relojes o computadores y que se caracteriza básicamente por tener software libre y multiplataforma.

Evolución histórica

Respecto al desarrollo histórico de Android, vale la pena destacar, que se inició en una empresa que trabajaba software para dispositivos móviles llamada Android IN, pero se dio a conocer en el año 2005 cuando fue adquirida por Google.

Entre el 2005 y 2008, Google lidera un proceso de comunicación con empresas fabricantes de hardware y software para dispositivos móviles, que en noviembre de 2007 se conoció como Open Handset Alliance y que para el 2011 la conformaban compañías de tecnología para dispositivos móviles, las cuales colaboran con el desarrollo del sistema operativo Android, trabajan en sociedad para implementar tecnologías y servicios basados en este sistema Operativo, afianzando su característica de código abierto bajo la licencia Apache

V2. (Open, 2012) y durante este mismo lapso de tiempo el equipo de Android en Google, desarrolló la plataforma para dispositivos móviles tomando como base el kernel de Linux (Engadget, 2007).

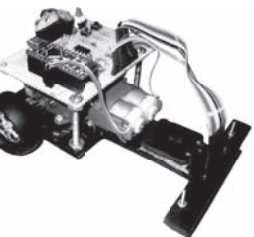
Android abarca otros ámbitos como el desarrollo, consumo y costo de aplicaciones, aceptación del sistema, entre otras (Mobile, 2010). Por ejemplo, los usuarios descargan un promedio de nueve aplicaciones al mes de la tienda virtual Android Market, algunas de estas descargas son gratis y otras son de pago asociado al número telefónico vinculado la cuenta de Google para descargar las aplicaciones.

El lanzamiento inicial del Android Software Development Kit fue en noviembre de 2007, luego en agosto 2008 se publicó la versión de Android 0.9 SDK en fase beta, en septiembre del mismo año, Google lanzó Android 1.0 SDK versión para el sistema operativo del teléfono HTC Dream (G1) de la empresa HTC.

Durante el 2009, se presentaron las versiones 1.1 de Android en marzo, luego Google lanzó la versión 1.5 de Android OS (llamada Cupcake) con su respectivo SDK y basada en el kernel de Linux versión 2.6.27, en septiembre Android 1.6 “Donut” versión basada en el kernel 2.6.29, al finalizar este año se publican las versiones 2.0 y 2.1 conocidas como Eclair también basadas en el kernel 2.6.29.

En el 2010 se publicaron la versión 2.2 llamada Froyo basado en el kernel 2.6.32 de Linux y la versión 2.3 denominada Gingerbread.

En el 2011 surgió la versión 3.0 de Android, destinada para dispositivos Tablets, bajo el nombre de Honeycomb y se publicaron actualizaciones a las versiones 3.1 y 3.2.



La versión 4.0 conocida como Ice Cream Sandwich cuya principal característica, es unificar interfaz para todos los dispositivos móviles sean teléfonos inteligentes o tablets, incluyendo otras mejoras, esta versión se libero en octubre del 2011 (Android, 2012)

Ventajas de Android

Android avanza a pasos agigantados, toda una comunidad de desarrolladores a nivel mundial respaldan la plataforma, lo cual facilita la creación de aplicaciones, ya que se encuentra documentación oficial tanto por Google como de otras comunidades independientes, que influenciadas por la cultura del software libre, publican sus códigos, experiencias y facilitan el trabajo de los programadores y/o usuarios que se inician en esta plataforma.

Una de las ventajas de Android es su plataforma abierta, lo que le permite al sistema operativo estar en dispositivos de diferentes gamas, prestaciones, precios y fabricantes, de manera que puede asegurar tanto la permanencia como una buena cuota del mercado al no estar segmentado a unas partes. Como se observa en la figura 1, el 43% del mercado es dominado por Android en relación con la venta de dispositivos móviles en Estados Unidos en agosto de 2011. (Brownlow, 2012)

Algunas de las razones por las cuales desarrollar para Android, según el desarrollador Tim Bray son:

- *“La experiencia de usuario de Android es muy satisfactoria y lo más importante, mejora rápidamente.*
- *Es amistoso con el desarrollador, las reglas o condiciones de entrada son muy bajas para los*

varios millones de personas en el planeta que están cómodos con el lenguaje de programación java.

- *Las API son muy buenas y completas. No hay nada interesante que los teléfonos puedan hacer que no está expuesto a través de alguna API.*
- *Cualquiera persona puede vender cualquier programa que escriben a través del Android Market, no hay que esperar aprobación a tu aplicación.*
- *Es de código abierto.*
- *El espacio de los móviles ha tenido un enorme impacto en las economías emergentes del mundo menos desarrollado y Android parece ser la plataforma de software adecuado para ello”. (Bray, 2010).*

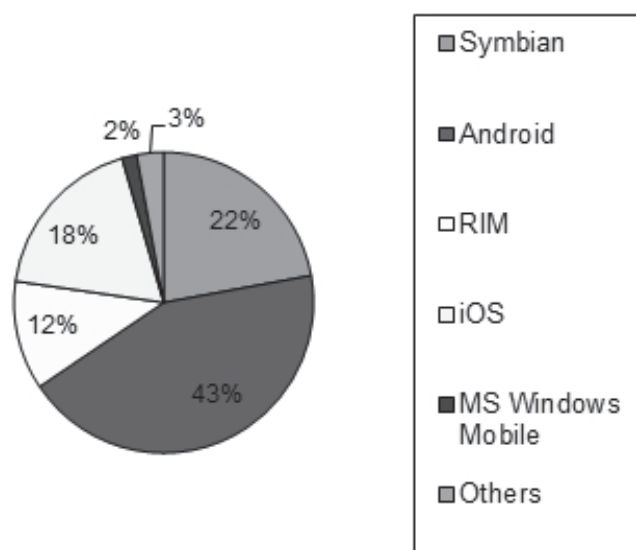
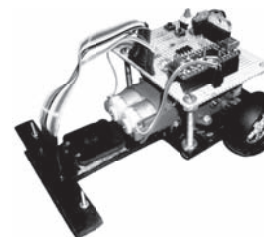


Figura 1. Estadística ventas dispositivos con sistema operativo Android.

Fuente: <http://www.email-marketing-reports.com/wireless-mobile/mobile-email-statistics.htm>



3. Arquitectura de android

En software, “la arquitectura” utiliza para definir cómo está construido un programa, sus elementos y donde se ubican; su conocimiento es necesario para desarrollar aplicaciones, saber que se

puede reutilizar y que operaciones del sistema realizar. La arquitectura de Android está dividida en cinco capas desde la superior a la inferior que son: aplicaciones, framework, librerías, ejecución o Runtime y Kernel de Linux. (Ver figura 2).

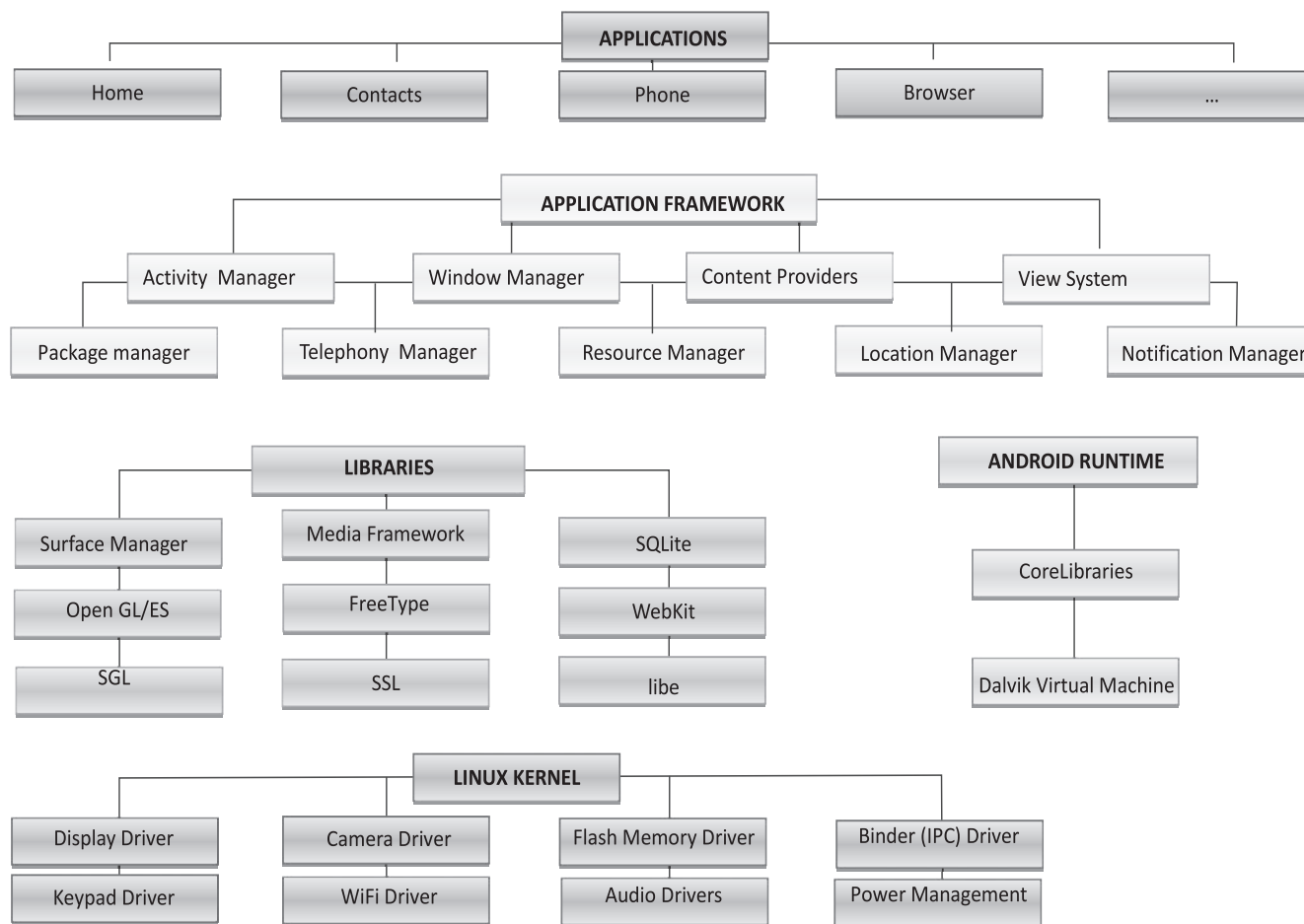


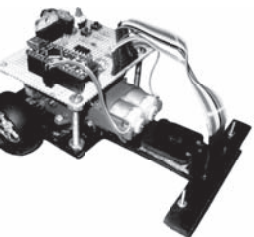
Figura 2. Arquitectura de Android

3.1 Capa de aplicaciones (applications)

Es aquella que ve el usuario final, para Android se usa el lenguaje de programación Java y tiene un conjunto de aplicaciones básicas incluidas como: un cliente de correo electrónico, programa de SMS, calendario, mapas, navegador y contactos.

3.2 Capa del framework (applications Framework)

Un framework es un conjunto de reglas definidas que permiten resolver un problema común u ofrecer herramientas para resolverlo. En el desarrollo de software, un framework se refiere a un soft-



ware o un conjunto de APIs que ayuda a la creación de nuevo software.

Android ofrece a los desarrolladores, la capacidad de crear aplicaciones ricas e innovadoras, quienes son libres para usar el hardware del dispositivo, información de acceso a la ubicación geográfica GPS, ejecutar servicios en segundo plano, configurar alarmas, añadir las notificaciones de la barra de estado, entre otras.

La arquitectura de Android está diseñada para simplificar la reutilización de componentes, y cualquier aplicación puede publicar o exponer sus capacidades y otra podrá usar los botones, cajas de texto, listas, notificaciones, según las restricciones de seguridad impuestas por el framework).

3.3 Capa de librerías (Libraries)

Android incluye un conjunto de librerías escritas en c/c++ usadas por varios componentes del sistema operativo. Las capacidades o funciones que realizan estas librerías están expuestas a los desarrolladores a través del framework de aplicaciones de Android. Algunas de las bibliotecas del núcleo son las siguientes:

System c library

Es una implementación del estándar C para las funciones básicas de manejo de dispositivos por el Kernel de Linux.

Media libraries

Está basado en el paquete de librerías OpenCore, que son capaces de reproducir y grabar audio y vídeo en varios formatos, lo mismo que en archivos de imágenes y fotografías.

Surface manager

Librería encargada de la gestión y el acceso a la pantalla del dispositivo.

Libwebcore

Un motor de navegación que alimenta el navegador de Android y permite ver páginas web embebidas dentro de cualquier otra aplicación.

Sgl

Las librerías necesarias para crear gráficos en 2D

3d libraries

Son librerías basadas en OpenGL ES 1.0; utilizan hardware de aceleración 3D (si están disponible en el dispositivo) o alternativamente renderizado 3D por software.

Free type

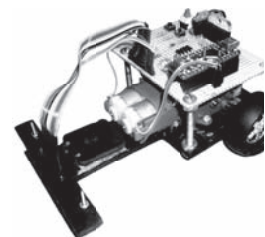
Librerías para la interpretación de mapa de bits y tipografía de textos.

SQLITE

Es un sistema de base de datos relacional reducido pero potente.

3.4 Capa Android Runtime

Cada aplicación Android se ejecuta con su propia instancia de la máquina virtual de Dalvik, la cual ha sido escrita para que un dispositivo pueda ejecutar varias máquinas virtuales de manera eficiente usando el formato Dalvik (.Dex), este ha sido optimizado para un consumo de memoria



mínimo usando el kernel de Linux para la funcionalidad subyacente, como la gestión de memoria e hilos a bajo nivel.

3.5 capa linux kernel

Es la capa de abstracción entre el hardware y el software, Android se basa en versión 2.6.x y superiores del kernel Linux, que es utilizado para los servicios del núcleo del sistema como la seguridad, la gestión de memoria, gestión de procesos, pila de red, modelo de controlador y gestión de usuarios.

4. Android Software Development Kit (SDK)

SDK es un conjunto de archivos con código, que contienen funciones comunes del sistema operativo que facilitan el proceso de desarrollo, como manejo de la cámara y del GPS por el sistema operativo se asocian bajo el nombre de APIs, y todas ellas constituyen una plataforma de software.

El SDK de Android provee un conjunto de herramientas y APIs necesarias para desarrollar aplicaciones usando en su mayoría el lenguaje de programación JAVA y se puede descargar, según el sistema operativo como Windows usando el instalador “archivo.exe” y para Linux se descarga y descomprime el SDK, (Developer, 2012).

Es necesario para instalar y configurar el SDK tener instalada previamente el JDK (Java Development Kit) (Oracle 2012) en la versión 5 o 6 y para Linux se debe instalar además las librerías C glibc.

5. Programación para Android

El lenguaje de programación para aplicaciones de Android es en su mayoría Java, como no existe el

paquete de Java AWT para los componentes gráficos, la plataforma Dalvik de Android interpreta el código hecho en Java.

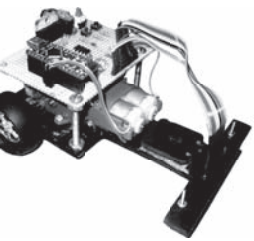
En Android también se puede programar en ++, ya que la capa de librerías están construidas en este lenguaje, pero Google no da soporte, por lo tanto, su uso no es muy extendido y existen interfaces de comunicación de estas librerías en Java, para que los desarrolladores las puedan utilizar.

Ahora bien, cuando se instala una aplicación en Android, se tiene una instancia propia de la JVM y se ejecuta en espacio llamado “sandbox”, que se caracteriza porque cada aplicación es un usuario diferente asignado un único ID para pueda acceder a la aplicación y cada proceso cuenta con su máquina virtual (VM) ejecutando el código de manera aislada, utiliza el principio de “privilegios mínimos” es decir, cada aplicación tiene acceso sólo a aquellos componentes que necesita, haciendo el ambiente muy seguro.

Sin embargo, una aplicación puede solicitar permiso para acceder a los datos del dispositivo, tales como: contactos del usuario, mensajes SMS, el almacenamiento (tarjeta SD), cámara, Bluetooth, estos permisos son autorizados por el usuario final.

6. Anatomía de una aplicación Android

Una aplicación para Android tiene cuatro componentes que son: actividades, servicios, Content Provider, intent e intent filters, los cuales pueden estar todos contenidos, algunos de ellos o inclusive solo uno y el manifiesto.



6.1. Actividad

Una actividad es una única pantalla o ventana en la aplicación, que se implementan en una sola "clase" y muestran la interfaz del usuario compuesta por varias vistas (Views), controles, widgets o elementos gráficos con los que se interactúa.

Algunas aplicaciones tienen varias pantallas, por ejemplo, los mensajes de texto pueden tener ventanas de contactos, escribir mensaje, contacto seleccionado, historial de mensajes; estas ventanas son actividades de la aplicación, que cambian a medida que se escogen, pasando de estado de activas, pausadas, paradas y destruidas a pausa, es decir es colocada en una pila de historial de actividad para el caso de Android. (Ver figura3)

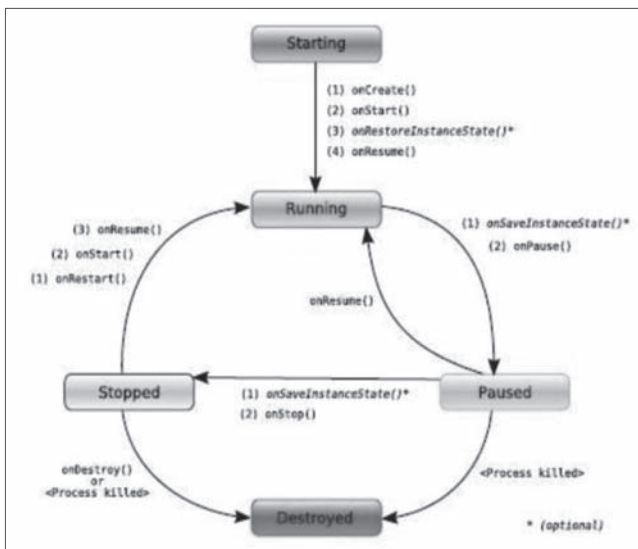


Figura 3. Ciclo de vida de una actividad

6.2. Services (servicios)

Un servicio es un componente de aplicación que puede realizar operaciones de larga duración sin estar activado o en primer plano y que no propor-

ciona una interfaz de usuario, lo que permite que funcione en un segundo plano o unirse a otro para hacer comunicación entre procesos, más conocida como IPC.

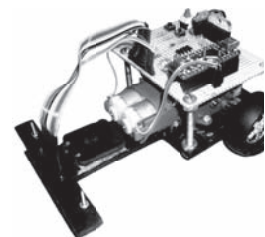
Ejemplos de servicios son: transacciones en la red, reproducir música, realizar operaciones con archivos de E/S (Entrada y Salida), o interactuar con un proveedor de contenido.

Un servicio puede adoptar dos formas que son "Started", cuando un componente de aplicación, inicia el servicio llamando al método `startService`. Una vez iniciado, puede ejecutarse en segundo plano de forma indefinida, incluso si el componente que se inició se destruye, por ejemplo descargar o cargar un archivo a través de la red.

Un servicio está "bound", cuando un componente de aplicación se une a él llamado al método `bindService` y consiste en que un servicio ofrece una interfaz de cliente-servidor que permite a los componentes interactuar, enviar solicitudes, obtener resultados, e incluso hacerlo a través de procesos de comunicación (IPC) y sólo se ejecuta estando ligado con el componente de otra aplicación.

6.3 Content provider

Los proveedores de contenido son la manera de compartir datos entre aplicaciones, Android tiene una serie de proveedores de contenido para los tipos de datos (audio, video, imágenes, información de contacto personal) con dos opciones: crear un proveedor de contenido propio (una subclase `ContentProvider`) o se puede agregar los datos a un proveedor existente.



6.4 Intent e intent filter

Intent es una clase especial que utiliza Android para moverse entre las pantallas, describiendo lo que una aplicación quiere hacer. Sus partes más importantes son “la acción” como ver y editar y “los datos” a los que se les aplica la acción, los cuales se expresan en formato URI (Uniform Resource Indicator).

Un Intent es una solicitud de hacer algo, por ejemplo mostrar la información de contacto y un IntentFilter es una descripción de que el Intent(s) es capaz de gestionar esa actividad es decir Wiews a los datos de un contacto.

6.5 Android Manifest

Cada aplicación debe tener un archivo AndroidManifest.xml en su directorio raíz, en el cual se presenta información esencial de la aplicación al sistema operativo como: nombres de los paquetes de Java que sirve como identificador único para la aplicación, actividades, servicios, intents y los proveedores de contenido que componen la aplicación. Este manifiesto declara los permisos que la aplicación debe tener, para acceder a las partes protegidas de la API, la forma en que interactúa con otras aplicaciones y los permisos que los demás están obligados a tener a fin de interactuar con los componentes de la aplicación.

7. Sugerencias para una aplicación básica para Android

A continuación, se presentan algunas recomendaciones a seguir, para realizar una aplicación básica de Android.

Luego de instalar el SDK para el sistema operativo que se esté utilizando, para Android se debe ins-

talalar Integrated Development Environment (IDE) “Eclipse”, ya que provee un plugin llamado “ADT” para hacer la programación y el depurado del código que permite simular el comportamiento de la aplicación.

Luego, se crea Android Virtual Device (AVD) dentro de Eclipse, este es un emulador de un dispositivo con el sistema operativo que permite ver el comportamiento que tiene la aplicación en desarrollo sin necesidad de disponer un teléfono físico,

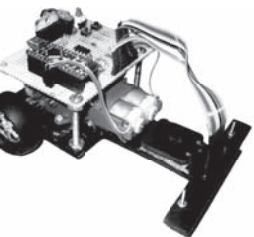
Después se crea un “nuevo proyecto” en el IDE Eclipse de manera similar como se hace en Java, completando la información básica de: nombre del proyecto, Build Target, nombre de la aplicación, nombre del paquete y Create activity.

Se continúa con la construcción de una Interfaz de Usuario (UI) que muestra los cambios en el código en la clase HelloAndroid.java, esta interfaz para Android está compuesta por vistas (views) o elementos gráficos, y se envía al método setContentView() el elemento tv para que lo proyecte en la UI.

Finalmente se ejecuta la aplicación seleccionando la opción “Run” del menú proyecto en Eclipse, éste automáticamente configurará y ejecutará el emulador de Android AVD que se creó con antelación, mostrando el nombre de la aplicación.

8. Conclusiones

Android es un sistema Operativo que abarca varios ámbitos como el de desarrollo, consumo, costos de aplicación que ha evolucionado vertiginosamente desde el 2005, cuando se dio a conocer al mundo al adquirirlo Google y sus descargas tanto pagas como gratuitas son representativas en el mercado mundial



Además de su desarrollo rápido en beneficio de miles de usuarios, Android es una plataforma abierta que permite el trabajo colaborativo de muchas personas y empresas y también facilita su programación, gracias a su desarrollo libre, las ayudas que proporciona, lo que permite estar en dispositivos de distintas gamas presentaciones y precios.

La filosofía de programación libre de Android, permite además de buenas experiencias a los

usuarios, facilidades por la rapidez en la ejecución de actividades, facilidad de entrada por las mínimas normas de restricción, con API completas y de código abierto, que no requiere aprobación previa para la aplicación, factores que inciden no sólo en beneficio del usuario final por la comodidad sino para la comunidad de programadores a nivel mundial, ya que se les permite contribuir la construcción de conocimiento en desarrollo de software.

9. Bibliografía

Bray, (2010) www.tbray.org/ongoing/When/201x/2010/03/15/Joining-Google

Brownlow, M. (2012) Smartphone statistics and market share <http://www.email-marketing-reports.com/wireless-mobile/smartphone-statistics.htm> recuperado 15/04/2012

Developer,(2012). <http://developer.android.com/sdk/index.html> recuperado 17/04/2012

Eclipse,(2010) <http://www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/indigosr1> recuperado 18/04/2012

Engadget, (2007) www.engadget.com/2007/08/28/google-is-working-on-a-mobile-os-and-its-due-out-shortly recuperado 02/05/2012

Mobile entertainment, (2010). (www.slideshare.net/stuardredge/mobile-entertainments-guide-to-android) recuperado 18/04/2012

Open Handset Alliance, (2012). www.openhandsetalliance.com/android_faq.html recuperado 25/03/2012

Oracle, (2012) <http://www.oracle.com/technetwork/java/javase/downloads/jdk-7u1-download-513651.html> recuperado 19/04/2012