UNIVERSIDAD CATÓLICA DE COSTA RICA

ESCUELA DE INGENIERÍA DE SISTEMAS

IS064 SISTEMAS OPERATIVOS II

MINI AGENDA ELECTRÓNICA EN ANDROID STUDIO

ANDRÉS J. JIMÉNEZ LEANDRO

EISNER LÓPEZ ACEVEDO

SEDE SAN CARLOS

DICIEMBRE 2016

Contenido

[CAPÍTULO I INTRODUCCION 4](#_Toc466036869)

[**Resumen ejecutivo** 4](#_Toc466036870)

[**Abstract** 5](#_Toc466036871)

[**Descripción** 7](#_Toc466036872)

[**Objetivos** 8](#_Toc466036873)

[**Objetivo general** 8](#_Toc466036874)

[**Objetivos específicos** 8](#_Toc466036875)

[**Justificación** 9](#_Toc466036876)

[**Introducción** 10](#_Toc466036877)

[CAPÍTULO II MARCO TEORICO 11](#_Toc466036878)

[**Marco teórico** 11](#_Toc466036879)

[**iOS.** 11](#_Toc466036880)

[Características. 12](#_Toc466036881)

[Tecnologías no admitidas. 13](#_Toc466036882)

[Ventajas. 14](#_Toc466036883)

[Desventajas. 15](#_Toc466036884)

[**Desarrollo** 16](#_Toc466036885)

[Programas. 16](#_Toc466036886)

[Swift. 16](#_Toc466036887)

[Xcode. 16](#_Toc466036888)

[Apple LLVM Compiler 17](#_Toc466036889)

[iOS Simulator 18](#_Toc466036890)

[Seguridad. 19](#_Toc466036891)

[Privacidad. 20](#_Toc466036892)

[Actualizaciones 21](#_Toc466036893)

[CAPÍTULO III PROCEDIMIENTO METODOLÓGICO 22](#_Toc466036894)

[**Cronograma**. 22](#_Toc466036895)

[**Bitácora**. 23](#_Toc466036896)

[CAPÍTULO IV INTERACCIÓN Y DISCUSIÓN DE RESULTADOS 24](#_Toc466036897)

[**Análisis del problema.** 24](#_Toc466036898)

[**Solución.** 24](#_Toc466036899)

[**Resultados obtenidos.** 25](#_Toc466036900)

[CAPÍTULO V CONCLUCION Y RECOMENDACIONES 26](#_Toc466036901)

[**Conclusiones.** 26](#_Toc466036902)

[**Recomendaciones.** 26](#_Toc466036903)

[Bibliografía 27](#_Toc466036904)

[Anexos 28](#_Toc466036905)

# CAPÍTULO I INTRODUCCION

## Resumen ejecutivo

La presente investigación de menciona que es ANDROID es un sistema operativo móvil de basado en Linux, que esta exclusivamente creado para dispositivos móviles con pantalla táctil, como teléfonos inteligentes, tablets o tabléfonos; y también para relojes inteligentes, televisores y automóviles. Inicialmente fue desarrollado por Android Inc. y entre otros. para avanzar en los estándares abiertos de los dispositivos móviles.

ANDROID es un dispositivo móvil que presenta ventajas y desventajas como el iOs, BlackBerry OS, Windows *Phone* y *Symbian*. Una de las principales ventajas es que es tiene muy buen diseño y funcionalidad, pero una de las desventaja es que al ser de Open Source, tiene diferentes variaciones del mismo sistema y existen fallas en la soporte para algún hardware, esta sería una de sus mayores desventajas.

*Android Studio* además de un editor de código, Incluye un emulador de dispositivos, herramientas para depuración de memoria y análisis del rendimiento del software. Inicialmente el entorno de desarrollo integrado (IDE) utilizado era Eclipse con el plugin de Herramientas de Desarrollo de Android (ADT). Ahora se considera como entorno oficial Android Studio, descargable desde la página oficial de desarrolladores de Android.

.

## Abstract

The present research mentions that ANDROID is a mobile operating system based on Linux, which is exclusively created for mobile devices with touch screen, such as smartphones, tablets or tablets; And also for smart watches, televisions and cars. Initially developed by Android Inc. and among others. To advance the open standards of mobile devices.

ANDROID is a mobile device that has advantages and disadvantages such as iOS, BlackBerry OS, Windows Phone and Symbian. One of the main advantages is that it is very good design and functionality, but one of the disadvantages is that being Open Source, has different variations of the same system and there are hardware support failures, this would be one of its biggest Disadvantages

Android Studio in addition to a code editor, includes a device emulator, tools for debugging memory and analysis of software performance. Initially the integrated development environment (IDE) used was Eclipse with the Android Development Tools (ADT) plugin. It is now considered as the official Android Studio environment, downloadable from the official Android developers page.

## Descripción

La finalidad de esta investigación se enfoca en saber cómo surge la creación, las características que presentan, las ventajas y desventajas que tiene y crear una aplicación básica para Android.

En esta investigación se realiza con el fin de dar a conocer más sobre los dispositivos de Android, el cual se expondrá una aplicación realizada en *Android Studio*, que constara de una Mini Agenda, a los estudiantes de la carrera de ingeniería en sistemas de la Universidad Católica de Costa Rica.

## Objetivos

### Objetivo general

Desarrollar una investigación sobre la creación y las aplicaciones móviles de Android.

### Objetivos específicos

* Analizar la creación de Android.
* Mencionar el programa en el que se desarrolló la aplicación.
* Mostrar las ventajas y desventajas del tema.

## Justificación

El presente investigación se realiza con el fin de obtener conocimientos acerca de la Android, ya que muy importante para los ingenieros en sistemas y también se realizara una aplicación para dispositivos móviles de Android, la cual se implementara el *Android Studio.*

Asimismo la investigación puede dar a conocer que Android la interfaz de multiusuario que está apoyada en el conocimiento de administración directa, utilizando gestos multitáctiles, los elementos de control son de deslizadores, interruptores y botones, la contestación los deseos de los usuario es inmediata y provee de una interfaz fluida, la interacción con el sistema operativo contiene gestos como deslices, toques, pellizcos, los cuales tienen definiciones diferentes dependiendo del contexto de la interfaz.

Teniendo en cuenta el valor teórico cabe resaltar la evidente eficaz teoría se puede mencionar que es Android es el más atractivo, los desarrolladores poseen vigilancia sobre el tráfico de sus aplicaciones en las tiendas de Google privativas o abiertas ya al ser open Source, los desarrolladores pueden subir sus aplicaciones ya sea al Play Store de Google, tomando en cuenta los requisitos para que sus software sea aprobado y cumpla con los estándares Google, controla y aprueba su trabajo antes que sea publicado.

## Introducción

La presente investigación se refiere a la creación y los usos de las aplicaciones móviles en Android. El sistema operativo para dispositivos móviles Inicialmente fue desarrollado por Android Inc., empresa que Google respaldó económicamente y más tarde, en 2005, la compró.

Android está fundamentado en el sistema operativo de Linux y las herramientas de programación son similares, con la incompatibilidad de que este se adapta a una plataforma móvil, siendo la prioridad el ambiente gráfico para ser más fácil uso e interacción para los usuarios, produciendo al máximo el hardware táctil convirtiéndose en un sistema manejable, intuitivo y divertido.

La versión básica de Android es conocida como Android Open Source Project (AOSP).

Esta exhaustiva investigación se enfoca en la creación y los usos de las aplicaciones móviles en Android se realizó por el interés de entrar en un tema, que tecnológicamente hablando está en desarrollo notables y progresivos.

En este documento inicialmente se describen la creación del dispositivo móvil Android y en el que se basa un sistema operativo. También se hace uso del Linux Ubuntu, donde se instala el aplicación de desarrollo *Android Studio* y haciendo uso del lenguaje , para dar un ejemplo de su programación y de esta forma se despliega completamente todo lo que forma parte de este significativo sistema operativo, llevando a los estudiantes de la carrera de ingeniería en sistemas, de la Universidad Católica de Costa Rica, un conocimiento sobre la composición del sistema operativo móvil Android y su creación.

# CAPÍTULO II MARCO TEORICO

## Marco teórico

### Android.

Según (García & Gutiérrez , 2013)

Android es un sistema operativo basado en el núcleo Linux. Fue diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes, tablets o tabléfonos; y también para relojes inteligentes, televisores y automóviles. Inicialmente fue desarrollado por Android Inc., empresa que Google respaldó económicamente y más tarde, en 2005, la compró. Android fue presentado en 2007 junto la fundación del Open Handset Alliance (un consorcio de compañías de hardware, software y telecomunicaciones) para avanzar en los estándares abiertos de los dispositivos móviles. El primer móvil con el sistema operativo Android fue el HTC Dream y se vendió en octubre de 2008.11 Los dispositivos de Android venden más que las ventas combinadas de Windows Phone e IOS.

El éxito del sistema operativo se ha convertido en objeto de litigios sobre patentes en el marco de las llamadas «Guerras por patentes de teléfonos inteligentes» (en inglés, Smartphone patent wars) entre las empresas de tecnología. Según documentos secretos filtrados en 2013 y 2014, el sistema operativo es uno de los objetivos de las agencias de inteligencia internacionales.

La versión básica de Android es conocida como Android Open Source Project (AOSP).

El 25 de junio de 2014 en la Conferencia de Desarrolladores Google I/O, Google mostró una evolución de la marca Android, con el fin de unificar tanto el hardware como el software y ampliar mercados.

Un sistema operativo Android tiene tres tipos de denominar a las versiones de Android, aunque las tres hacen referencia a la misma versión:

La comercial con el nombre de postre. Por ejemplo: KitKat

La de los fabricantes (y también comercial) con la versión y subversión. Por ejemplo: 4.4

La de desarrollador con el nivel del API (ésta nos interesa mucho para desarrollar en Android): Por ejemplo: 19 Pagiina (10)

#### Arquitectura.

Según el autor Higinio Ramírez García (2014):

Los componentes principales del sistema operativo de Android (cada sección se describe en detalle):

**Aplicaciones**: las aplicaciones base incluyen un cliente de correo electrónico, programa de SMS, calendario, mapas, navegador, contactos y otros. Todas las aplicaciones están escritas en lenguaje de programación Java.

**Marco de trabajo de aplicaciones**: los desarrolladores tienen acceso completo a los mismos APIs del framework usados por las aplicaciones base. La arquitectura está diseñada para simplificar la reutilización de componentes; cualquier aplicación puede publicar sus capacidades y cualquier otra aplicación puede luego hacer uso de esas capacidades (sujeto a reglas de seguridad del framework). Este mismo mecanismo permite que los componentes sean reemplazados por el usuario.

**Bibliotecas**: Android incluye un conjunto de bibliotecas de C/C++ usadas por varios componentes del sistema. Estas características se exponen a los desarrolladores a través del marco de trabajo de aplicaciones de Android; algunas son: System C library (implementación biblioteca C estándar), bibliotecas de medios, bibliotecas de gráficos, 3D y SQLite, entre otras.

**Runtime de Android**: Android incluye un set de bibliotecas base que proporcionan la mayor parte de las funciones disponibles en las bibliotecas base del lenguaje Java. Cada aplicación Android corre su propio proceso, con su propia instancia de la máquina virtual Dalvik. Dalvik ha sido escrito de forma que un dispositivo puede correr múltiples máquinas virtuales de forma eficiente. Dalvik ejecutaba hasta la versión 5.0 archivos en el formato Dalvik Executable (.dex), el cual está optimizado para memoria mínima. La Máquina Virtual está basada en registros y corre clases compiladas por el compilador de Java que han sido transformadas al formato.dex por la herramienta incluida "dx". Desde la versión 5.0 utiliza el ART, que compila totalmente al momento de instalación de la aplicación.

**Núcleo Linux**: Android depende de Linux para los servicios base del sistema como seguridad, gestión de memoria, gestión de procesos, pila de red y modelo de controladores. El núcleo también actúa como una capa de abstracción entre el hardware y el resto de la pila de software.

#### Seguridad, privacidad y vigilancia.

Según un estudio de Symantec de 2013, demuestra que en comparación con iOS, Android es un sistema explícitamente menos vulnerable. El estudio en cuestión habla de 13 vulnerabilidades graves para Android y 387 vulnerabilidades graves para iOS. El estudio también habla de los ataques en ambas plataformas, en este caso Android se queda con 113 ataques nuevos en 2012 a diferencia de iOS que se queda en 1 solo ataque. Incluso así Google y Apple se empeñan cada vez más en hacer sus sistemas operativos más seguros incorporando más seguridad tanto en sus sistemas operativos como en sus mercados oficiales.

Se han descubierto ciertos comportamientos en algunos dispositivos que limitan la privacidad de los usuarios, de modo similar a iPhone, pero ocurre al activar la opción «Usar redes inalámbricas» en el menú «Ubicación y seguridad», avisando que se guardarán estos datos, y borrándose al desactivar esta opción, pues se usan como caché y no como log tal como hace iPhone.

Como parte de las amplias revelaciones sobre vigilancia masiva filtradas en 2013 y 2014, se descubrió que las agencias de inteligencia estadounidenses y británicas, la Agencia de Seguridad Nacional (NSA) y el Cuartel General de Comunicaciones del Gobierno (GCHQ), respectivamente, tienen acceso a los datos de los usuarios de dispositivos Android. Estas agencias son capaces de leer casi toda la información del teléfono como SMS, geolocalización, correos, notas o mensajes.61 Documentos filtrados en enero de 2014, revelaron que las agencias interceptan información personal a través de Internet, redes sociales y aplicaciones populares, como Angry Birds, que recopilan información para temas comerciales y de publicidad. Además, según *The Guardian*, el GCHQ tiene una wiki con guías de las diferentes aplicaciones y redes de publicidad para saber los diferentes datos que pueden ser interceptados. Una semana después de salir esta información a la luz, el desarrollador finlandés Rovio, anunció que estaba reconsiderando sus relaciones con las distintas plataformas publicitarias y exhortó a la industria en general a hacer lo mismo.

Las informaciones revelaron que las agencias realizan un esfuerzo adicional para interceptar búsquedas en Google Maps desde Android y otros smartphones para recopilar ubicaciones de forma masiva. La NSA y el GCHQ insistieron en que estas actividades cumplen con las leyes nacionales e internacionales, aunque The Guardian afirmó que *las últimas revelaciones podrían sumarse a la creciente preocupación pública acerca de cómo se acumula y utiliza la información, especialmente para aquellos fuera de los EE.UU. que gozan de menos protección en temas de privacidad que los estadounidenses*. (James, 2014)

#### Diseño y desarrollo.

Android es considerado como uno de los modelos de negocio más exitosos, pues su desarrollo estratégico contempla los factores que más se tienen en cuenta dentro de las herramientas y metodologías desarrollados por expertos en negocios. Este sistema operativo se ha convertido en un modelo a seguir por desarrolladores de tendencias y negocios de alto impacto. «Actualidad Android».

Android, al contrario que otros sistemas operativos para dispositivos móviles como iOS o Windows Phone, se desarrolla de forma abierta y se puede acceder tanto al código fuente como a la lista de incidencias donde se pueden ver problemas todavía no resueltos y reportar problemas nuevos. (Android Open Source Project, 2016)

El que se tenga acceso al código fuente no significa que se pueda tener siempre la última versión de Android en un determinado móvil, ya que el código para soportar el hardware (controladores) de cada fabricante normalmente no es público, así que faltaría un *trozo* básico del firmware para poder hacerlo funcionar en dicho terminal, y porque las nuevas versiones de Android suelen requerir más recursos, por lo que los modelos más antiguos quedan descartados por razones de memoria (RAM), velocidad de procesador, etc.

En un principio, Android era eminentemente un sistema operativo pensado para usar con teclado, y gracias a un cursor poder navegar entre las aplicaciones. Desde su comienzo, Android ha sido altamente personalizable. Poco después, antes del lanzamiento del primer teléfono Android, esta filosofía cambió para convertirse en eminentemente táctil, y poder competir contra el iPhone, presentado 1 año y 9 meses antes.

¿Porque Android?

La plataforma de software para teléfonos móviles Android de Google puede ser la próxima gran oportunidad para los desarrolladores de aplicaciones de software. Google anunció la Open Handset Alliance y la plataforma Android en noviembre de 2007, lanzando la primera versión beta del Kit de desarrollo de software para Android (SDK) al mismo tiempo. En cuestión de unos meses, más de un millón de personas descargaron versiones del SDK desde el sitio web de Google. En los Estados Unidos, T-Mobile anunció el teléfono móvil G1 Android en octubre de 2008, y se estima que varios cientos de miles de G1 se vendieron antes de finales de ese año. Ya hay varias pilas de software de telefonía móvil en competencia en el mercado, así que ¿por qué existe tal interés en Android? Android tiene el potencial para eliminar las barreras al éxito en el desarrollo y la venta de una nueva generación de software de aplicación de telefonía móvil. Al igual que las plataformas estándar de PC y Macintosh crearon mercados para software de escritorio y servidor, Android, al proporcionar un entorno estándar de aplicaciones para teléfonos móviles, creará un mercado para aplicaciones móviles y la oportunidad para los desarrolladores de aplicaciones de beneficiarse de esas aplicaciones. (Roger, Lombardo, Mednieks, & Meike, 2009)

#### Ventajas y Desventajas.

### Ventajas del sistema Android

El código de Android es abierto: Google liberó Android bajo licencia Apache. Cualquier persona puede realizar una aplicación para Android.

Hoy día hay más de 650.000 aplicaciones disponibles para teléfonos Android, aproximadamente 2/3 son gratis. Además, la libertad de código permite adaptar Android a bastantes otros dispositivos además de teléfonos celulares. Está implantado en Tablets, GPS, relojes, microondas… incluso hay por internet una versión de Android para PC.

El sistema Android es capaz de hacer funcionar a la vez varias aplicaciones y además se encarga de gestionarlas, dejarlas en modo suspensión si no se utilizan e incluso cerrarlas si llevan un periodo determinado de inactividad. De esta manera se evita un consumo excesivo de batería. Esta es una de sus mayores ventajas por la rapidez con la que carga una aplicación abierta previamente. Por ejemplo, abro Google Mapas y en un momento la aplicación localiza mi posición en el mapa.

### Desventajas del sistema Android

A pesar de ser una ventaja el ser un sistema multitarea: El hecho de tener varias aplicaciones abiertas hacen que el consumo de la batería aumente y como no todas las aplicaciones Android las cierra hay que instalar una aplicación para que las cierre. En la Market de Android hay una buena cantidad de aplicaciones para este fin, así que el problema es solucionable pero debería venir pre instalado de fábrica.

Duración de la batería: la batería en un celular Android se agota muy rápido. Utilizando las aplicaciones de manera moderada la batería puede llegar a durar más, pero para un usuario que usa mucho sus aplicaciones la batería no tiende a durar, lo que se puede solucionar con algunas aplicaciones, pero volvemos a lo mismo no viene pre instalada de fábrica, hace falta una aplicación externa al sistema Android para optimizar mejor la batería.

Poco intuitivo: Para la mayoría el sistema operativo es muy complicado. Por ejemplo, se vuelve complicado configurar el teléfono, esto te puede llevar mucho tiempo, y esto es generado por la interfaz de Android. Hay aplicaciones que ayudan en tareas que deberían ser sencillas como desinstalar otras aplicaciones, pero, otra vez, volvemos a lo mismo se hace necesario instalar aplicaciones para solucionar el problema.

**Desarrollo**

Las aplicaciones de Android, como la mayoría de las aplicaciones de telefonía móvil, se desarrollan en un host target entorno de desarrollo. En otras palabras, desarrolla su aplicación en un Host (donde los recursos son abundantes) y descargarlo a un móvil objetivo Teléfono para las pruebas y el uso final. Las aplicaciones pueden ser probadas y depuradas Un dispositivo Android real o en un emulador. Para la mayoría de los desarrolladores, usar un emulador es Más fácil para el desarrollo inicial y depuración, seguido de pruebas finales en dispositivos reales. Para escribir sus propias aplicaciones de teléfono móvil con Android, primero tendrá que recopilar Herramientas necesarias y configurar un entorno de desarrollo adecuado en su PC o Mac.

En este las herramientas que necesite, las descargamos e instalamos en Su computadora, y escriba una aplicación de ejemplo que le permitirá obtener la sensación de escribir Y ejecutar aplicaciones de Android en un emulador. Linux, Windows y OS X son todos Los entornos de desarrollo compatibles y le mostraremos cómo instalar el último conjunto De herramientas en cada uno. A continuación, le mostraremos cualquier configuración que necesite hacer después de instalar Las herramientas (configuración de variables de entorno PATH y similares), de nuevo para cada uno de los tres sistemas operativos. Finalmente, escribiremos una pequeña aplicación "Hello, Android" que Demuestra lo que hay que hacer para poder ejecutar una aplicación genérica. El SDK de Android admite varios entornos de desarrollo integrados diferentes (IDE). Para este libro nos centraremos en Eclipse porque es el IDE que mejor se integra Con el SDK, y, hey, es gratis. Independientemente del sistema operativo que utilice, Necesitará esencialmente el mismo conjunto de herramientas:

• El IDE de Eclipse

• El kit de desarrollo Java de Sun (JDK)

• El kit del desarrollador de software para Android (SDK)

• La herramienta para desarrolladores de Android (ADT), un complemento especial de Eclipse (Roger, Lombardo, Mednieks, & Meike, 2009)

## Componentes de una aplicación de Android

Sus aplicaciones de Android se construirán a partir de cuatro tipos de componentes básicos que se definen Por la arquitectura de Android:

### Actividades

Estas son comparables a las utilidades independientes en los sistemas de escritorio, como Aplicaciones.

Las actividades son piezas de código ejecutable que van y vienen a tiempo, Instanciado por el usuario o el sistema operativo y funcionando siempre y cuando Se necesitan. Pueden interactuar con el usuario y solicitar datos o servicios de otros Actividades o servicios a través de consultas o Intenciones (discutidas en un momento).

La mayor parte del código ejecutable que escribes para Android se ejecutará en el contexto de una actividad. Las actividades suelen corresponder a pantallas: cada actividad muestra Una pantalla para el usuario. Cuando no se está ejecutando activamente, una actividad puede ser El sistema operativo para conservar la memoria.

### Servicios

Estos son análogos a los servicios o daemons en sistemas operativos de escritorio y servidores.

Son piezas ejecutables de código que normalmente se ejecutan en el fondo de El tiempo de su creación de instancias hasta que el teléfono móvil se apaga. Generalmente

Puede tener que implementar Servicios para realizar tareas de fondo que persisten sin Una interfaz de usuario.

### Receptores de difusión y de intenciones

Éstos responden a solicitudes de servicio de otra aplicación. Una transmisión Receptor responde a un anuncio en todo el sistema de un evento. Estos anuncios Puede provenir de Android (p. Ej., Batería baja) o de cualquier programa en ejecución En el sistema. Una actividad o servicio proporciona acceso a otras aplicaciones A su funcionalidad mediante la ejecución de un receptor de intención, una pequeña pieza de código ejecutable Que responde a solicitudes de datos o servicios de otras actividades. El solicitante (Cliente) emite un intento, dejándolo al marco de Android para calcular Qué aplicación debe recibir y actuar sobre ella.

Los intents son uno de los elementos arquitectónicos clave en Android que facilitan la creación Utilizar Intents en su aplicación para interactuar con otras aplicaciones y servicios que Proporcionar la información necesaria para su solicitud. Los Intents y los receptores de intención son Proveedores de contenido Éstos se crean para compartir datos con otras actividades o servicios. Un proveedor de contenido Utiliza una interfaz estándar en forma de URI para satisfacer solicitudes de datos de otros Aplicaciones que incluso no saben qué proveedor de contenido están utilizando. Por Ejemplo, cuando una aplicación emite una consulta para los datos de Consulta a un URI del formulario:

Content: // contactos / personas

El sistema operativo busca ver qué aplicaciones se han registrado Como proveedores de contenido para el URI dado, y envía la solicitud a la Aplicación (iniciando la aplicación si aún no se está ejecutando). Si hay más De un proveedor de contenido registrado para el URI solicitado, el sistema operativo Pregunta al usuario que quiere usar.

Una aplicación no tiene que usar todos los componentes de Android, sino un bien escrito Aplicación hará uso de los mecanismos proporcionados, en lugar de reinventar la funcionalidad O hardcoding referencias a otras aplicaciones. URIs y Intents permiten Android para proporcionar un entorno de usuario muy flexible. Las aplicaciones pueden ser Añadido, eliminado y sustituido, y el acoplamiento suelto de intentos y URIs mantiene Todo trabajando juntos.

## Ciclo de vida del servicio Android

El ciclo de vida de un servicio es similar al de una actividad, pero diferente en unos cuantos detalles

## OnCreate y diferencias de onStart

Los servicios se pueden iniciar cuando un cliente llama al Context.startService (Intent) Método. Si el servicio aún no se está ejecutando, Android lo inicia y llama a OnCreate seguido del método onStart. Si el servicio ya se está ejecutando, Su método onStart se invoca de nuevo con la nueva intención. Así que es muy posible y Normal para que el método onStart de un servicio sea llamado repetidamente en una sola

### Servicio.

OnResume, onPause y onStop no son necesarios

Recuerde que un servicio generalmente no tiene interfaz de usuario, por lo que no es necesario

OnPause, onResume o onStop. Siempre que un servicio se está ejecutando, siempre es en el fondo.

En el carril Si un cliente necesita una conexión persistente a un servicio, puede llamar al Context.bind Método de servicio. Esto crea el servicio si no se está ejecutando y llama a Create pero No enStart. En su lugar, el método onBind se llama con la intención del cliente, y Devuelve un objeto IBind que el cliente puede utilizar para realizar más llamadas al servicio.

Es bastante normal que un servicio tenga clientes que comienzan y los clientes se limitan a él en al mismo tiempo.

# CAPÍTULO III PROCEDIMIENTO METODOLÓGICO

## Cronograma.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | PERIODO 2016- III CUATRIMESTRE | | | | | | | | | | | | | | | | | | | |
| ACTIVIDADES | Septiembre | | | | | | | | | | | | | | | | | | | |
| MES 2 | | | | | | | MES 3 | | | | | | | | MES 4 | | | | |
| Investigar del tema |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| CAPITULO I INTRODUCCION |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Introducción |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Descripción |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Resumen Ejecutivo |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| *Abstract* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Objetivos |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Justificación |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| CAPITULO II MARCO TEORICO |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Marco teórico |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Desarrollo |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| CAPITULO III PROCEDIMIENTO METODOLOGICO |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| CAPITULO IV INTERPRETACION Y DISCUSION DE RESULTADOS |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Análisis del problema |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Solución |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| CAPITULO V CONCLUSION Y RECOMENDACIONES |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Conclusión |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Recomendaciones |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Aplicación |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Reproductor de video |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |

## Bitácora.

|  |  |  |
| --- | --- | --- |
| Fecha | Horas | Actividad |
| 20/09/2016 | **8 horas** | Investigar sobre el tema, realizar la justificación, descripción y marco teórico. |
| 23/10/2016 | **5 horas** | Continuar con la investigar del tema y el marco teórico y también finalizar con la justificación y descripción. |
| 24/10/2016 | **6 horas** | Permanecer con la investigar del tema y el marco teórico. |
| 25/10/2016 | **4 horas** | Extender con la investigar del tema, el marco teórico y también realizar los objetivos. |
| 26/10/2016 | **5 horas** | Continuar con la investigar del tema, elaborar la introducción |
| 17/11/2016 | **2 horas** | Finalizar el marco teórico. |
| 18/11/2016 | **7 horas** | Finalizar con la investigar del tema, la introducción e iniciar con el desarrollo. |
| 19/11/2016 | **3 horas** | Avanzar con el desarrollo. |
| 20/11/2016 | **2 horas** | Continuar con el desarrollo. |
| 21/11/2016 | **3 horas** | Extender con el desarrollo. |
| 22/11/2016 | **2 horas** | Finalizar con el desarrollo, elaborar el análisis del problema. |
| 23/11/2016 | **2 horas** | Elaborar la solución del problema y la conclusión de Android. |
| 1/12/2016 | **2 horas** | Iniciar con el *abstract.* |
| 5/12/2016 | **2 horas** | Continuar con el *abstract* e iniciar con el resumen ejecutivo. |
| 6/12/2016 | **3 horas** | Finalizar el *abstract* y el resumen ejecutivo. |
| 7/12/2016 | **2 horas** | Formalizar las recomendaciones. |
| 8/09/2016 | **3 horas** | Aprendiendo a usar *Android Studio.* |
| 20/09/2016 | **4 horas** | Implementar la aplicación. |
| 3/12/2016 | **8 horas** | Finalizar la aplicación. |

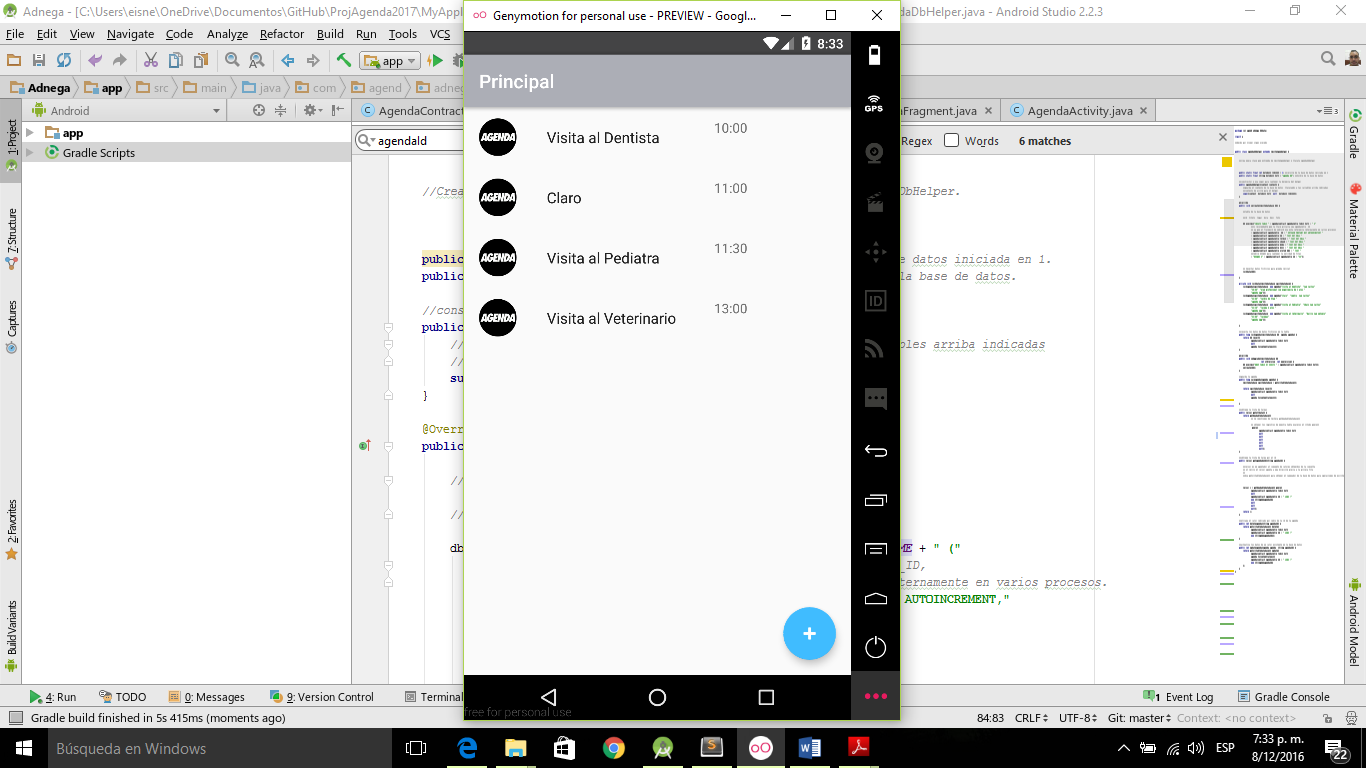
# CAPÍTULO IV INTERACCIÓN Y DISCUSIÓN DE RESULTADOS

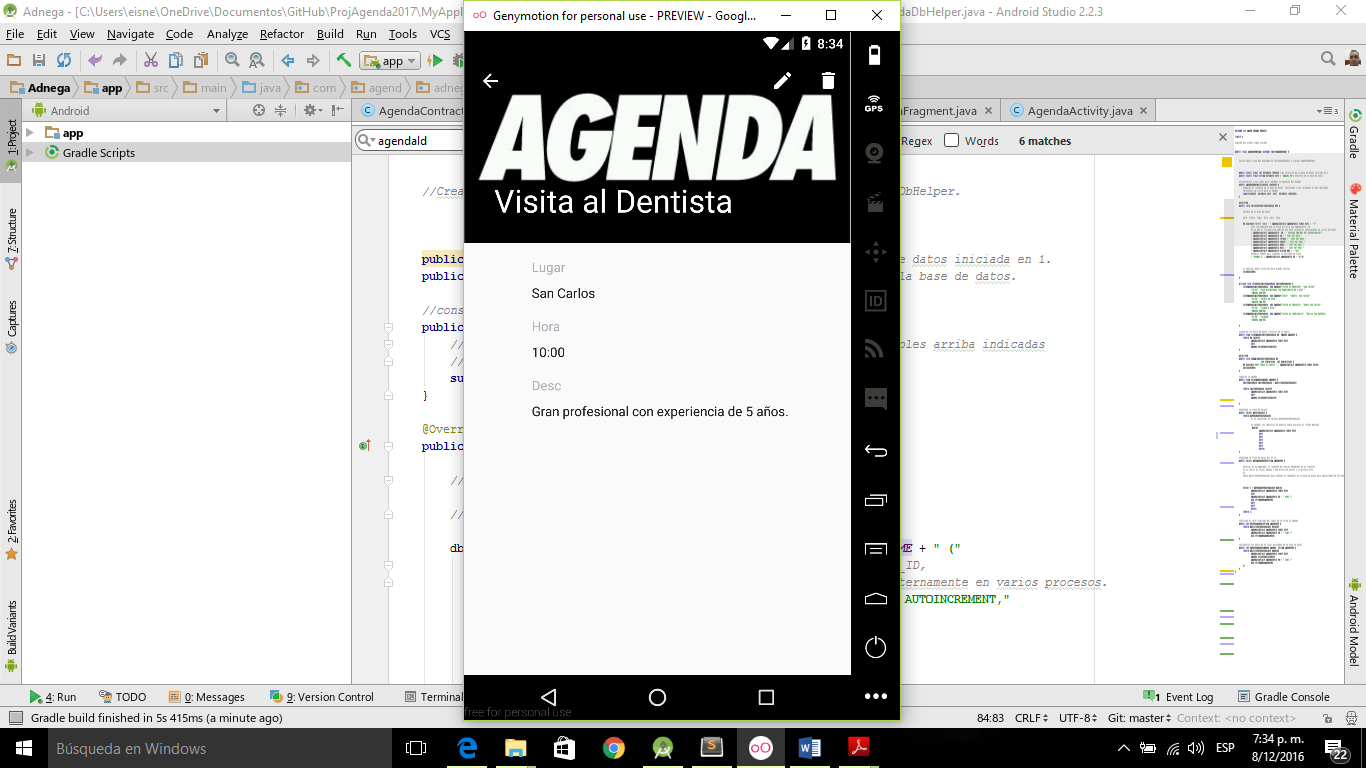
## Análisis del problema.

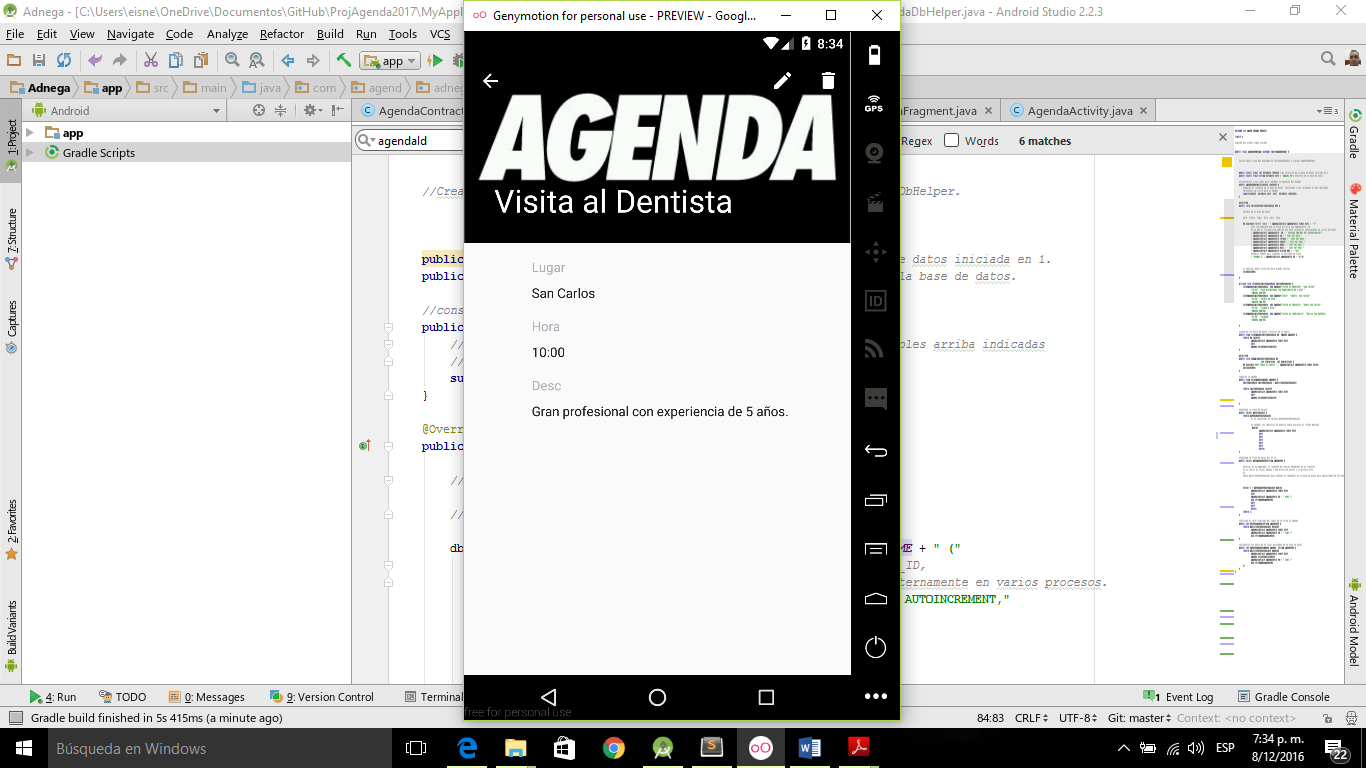
El problema que se presenta, es investigar muy profundo sobre cómo funciona Android después de dicha investigación se debe de realizar una aplicación sencilla, esta debe de puede ser en consola o modo gráfico y luego ser expuestas a los estudiantes de la carrera de Ingeniería en Sistemas, de la Universidad Católica de Costa Rica.

## Solución.

La aplicación que se debe de desarrollar de la investigación se solventó con una Mini Agenda.







## Resultados obtenidos.

Los resultados obtenidos en la investigación fueron excelentes, ya que:

* La búsqueda de la creación Andrid sin ningún problema.
* Se logró realizar la aplicación de una Agenda en Android Studio.

# CAPÍTULO V CONCLUCION Y RECOMENDACIONES

## Conclusiones.

Como conclusiones que se pueden derivar de la investigación son muy positivas.

* Los requisitos iniciales que se presentaban tras la elección del proyecto se han cumplido en su mayoría, lo que se puede considerar un éxito.
* Se reitera que el aprendizaje del conocimiento conseguido la investigación en Android ha sido enriquecedores.
* El ambiente de desarrollo consumido por Android Studio es verdaderamente muy de la mano con el programador, permitiendo personalizar sencillamente las interfaces a la hora de crearla y corregir errores de código.

Android requiere a unos servicios básicos de seguridad integrada y coherente en todas las plataformas Android y en su entorno de las apps. La encriptación de acumulación basada en el hardware brinda en caso de pérdida, la posibilidad de borrar el contenido de un dispositivo y consiente a los usuarios quitar toda la información personal y colectiva si lo venden o lo transfieren a otro usuario.

## Recomendaciones.

Para futuros trabajos de investigación y aplicaciones de Android, en se realiza las siguientes recomendaciones:

* Tener en claro cómo funciona Android.
* Una PC con las siguientes características:

# Requisitos mínimos de Android Studio

### Windows

* Windows 7/8/10 (32 o 64 bits).
* 2 GB de RAM (8 GB de RAM recomendado).
* 2 GB de espacio libre mínimo (4 GB recomendado).
* Resolución mínima de 1.280 x 800.
* Java 8.
* 64 bits y procesador Intel (emulador).

### Mac

* Mac OS X 10.8.5 o superior.
* 2 GB de RAM (8 GB de RAM recomendado).
* 2 GB de espacio libre mínimo (4 GB recomendado).
* Resolución mínima de 1.280 x 800.
* Java 6.

### Linux

* GNOME o KDE Desktop.
* Ubuntu…
* 64 bits / 32 bits.
* GNU C (glibc) 2.1 o superior.
* 2 GB de RAM (8 GB de RAM recomendado).
* 2 GB de espacio libre mínimo (4 GB recomendado).
* Resolución mínima de 1.280 x 800.
* Java 8.
* 64 bits y procesador Intel (emulador).

# Bibliografía.

Ancona Reyes, J. n saA. (2015). *Desarrollo de una aplicación nativa de enseñanza para IOS con SWIFT.*

Recuperado de <http://dspace.biblioteca.um.edu.mx/jspui/bitstream/123456789/431/1/TESIS%20Jos%C3%A9%20%20Ancona.pdf>

Carlo, R. (2012). *Dispositivos moviles.* Europa. Recuperado de <http://api.eoi.es/api_v1_dev.php/fedora/asset/eoi:75585/componente75583.pdf>

Chaves, V. (2014). *La Evolucion del iPhone .* Paraguay. Recuperado de <http://jeuazarru.com/wp-content/uploads/2014/10/iPhone1.pdf>

García, H. R. (2014). *Análisis de Sistemas Operativos Smartphone.* México. Recuperado de <https://core.ac.uk/download/pdf/33659847.pdf>

Moises, A. C. (2010). *Uso basico de Xcode & Interface Builder en MAC OS X usando la libreria Cocoa de Objective-C.* Mexico. Recuperado de <http://delta.cs.cinvestav.mx/~mcintosh/cellularautomata/Summer_Research_files/manualOSX.pdf>

Petrazzini., G. O. (2012). *Sistemas Operativos en Dispositivos.* Argentina. Recuperado de <http://exa.unne.edu.ar/informatica/SO/Sistemas_Operativos_en_Dispositivos_Moviles.pdf>

Vittone, J. C. (2013). *Diseñando apps para móviles.* Recuperado de <https://books.google.co.cr/books?id=ATiqsjH1rvwC&pg=PA8&dq=dise%C3%B1ando+apps+para+m%C3%B3viles+ios+libro&hl=es&sa=X&ved=0ahUKEwjijNW284fQAhVDSiYKHSKSByUQ6AEIJTAA#v=onepage&q&f=false>

Ancona Reyes, J. A. (2015). *Desarrollo de una aplicación nativa de enseñanza para IOS con SWIFT.*

Android Open Source Project. (08 de 12 de 2016). *https://code.google.com/p/android/*. Obtenido de https://code.google.com/p/android/: https://code.google.com/p/android/

Carlo, R. (2012). *Dispositivos moviles.* Europa.

Chaves, V. (2014). *La Evolucion del iPhone .* Paraguay.

García, H. R. (2014). *Análisis de Sistemas Operativos Smartphone.* México.

García, R. M., & Gutiérrez , J. C. (2013). *Android 100%.* jarroba.

J. B. (28 de 01 de 2014). *The Guardian*. Obtenido de https://www.theguardian.com/international: https://www.theguardian.com/world/2014/jan/27/nsa-gchq-smartphone-app-angry-birds-personal-data

Moises, A. C. (2010). *Uso basico de Xcode & Interface Builder en MAC OS X usando la libreria Cocoa de Objective-C.* Mexico.

Morata, J. Q. (2011). *Desarrollo de una aplicación distribuida para dispositivos iOS.* España.

Petrazzini., G. O. (2012). *Sistemas Operativos en Dispositivos.* Argentina.

Roger, R., Lombardo, J., Mednieks, Z., & Meike, B. (2009). Getting to Know Android. En R. Roger, J. Lombardo, Z. Mednieks, & B. Meike, *Android Application Development* (pág. 19). California: O´Reilly.

# Anexos.

