



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**IMPLEMENTACIÓN DE TECNOLOGÍAS WEB Y MÓVILES PARA EL USO DE SERVICIOS
EN LA NUBE Y EL ALMACENAMIENTO DE ARCHIVOS EN GOOGLE DRIVE Y DROPBOX**

Guillermo Estuardo Urías Teque

Giancarlo Guilá Reina

Asesorado por el Ing. Ricardo Mazariegos e Ing. Edgar Santos

Guatemala, noviembre de 2014

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**IMPLEMENTACIÓN DE TECNOLOGÍAS WEB Y MÓVILES PARA EL USO DE SERVICIOS
EN LA NUBE Y EL ALMACENAMIENTO DE ARCHIVOS EN GOOGLE DRIVE Y DROPBOX**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR

GUILLERMO ESTUARDO URÍAS TEQUE

GIANCARLO GUILÁ REINA

ASESORADO POR EL ING. RICARDO MAZARIEGOS E ING. EDGAR SANTOS

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, NOVIEMBRE DE 2014

UNIVERSIAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympo Paiz Recinos
VOCAL I	Ing. Alfredo Enrique Beber Aceituno
VOCAL II	Ing. Pedro Antonio Aguilar Polanco
VOCAL III	Inga. Elvia Miriam Ruballos Samayoa
VOCAL IV	Br. Narda Lucía Pacay Barrientos
VOCAL V	Br. Walter Rafael Véliz Muñoz
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Murphy Olympo Paiz Recinos
EXAMINADOR	Ing. Oscar Alejandro Paz Campos
EXAMINADOR	Ing. José Ricardo Morales Prado
EXAMINADOR	Ing. Pedro Pablo Hernández Ramírez
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

Giancarlo Guilá Reina

UNIVERSIAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympo Paiz Recinos
VOCAL I	Ing. Alfredo Enrique Beber Aceituno
VOCAL II	Ing. Pedro Antonio Aguilar Polanco
VOCAL III	Inga. Elvia Miriam Ruballos Samayoa
VOCAL IV	Br. Narda Lucía Pacay Barrientos
VOCAL V	Br. Walter Rafael Véliz Muñoz
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Murphy Olympo Paiz Recinos
EXAMINADOR	Ing. Ludwing Federico Altan Sac
EXAMINADOR	Ing. José Ricardo Morales Prado
EXAMINADOR	Ing. Oscar Alejandro Paz Campos
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

Guillermo Estuardo Urías Teque

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

IMPLEMENTACIÓN DE TECNOLOGÍAS WEB Y MÓVILES PARA EL USO DE SERVICIOS EN LA NUBE Y EL ALMACENAMIENTO DE ARCHIVOS EN GOOGLE DRIVE Y DROPBOX

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha febrero de 2013



Guillermo Estuardo Urías Teque



Giancarlo Guilá Reina

Guatemala 3 de Marzo de 2014

Ingeniero
Carlos Azurdia
Revisor de Trabajo de Graduación
Escuela de Ciencias y Sistemas
Facultad de Ingeniería

Respetable Ingeniero Azurdia:

Por este medio hago de su conocimiento que he revisado el Trabajo de Graduación de los estudiantes Guillermo Estuardo Urías Teque identificado con número de carnet 200819111 y Giancarlo Guilá Reina identificado con número de carnet 200819105, titulado: **"Implementación de tecnologías web y móviles para el uso de servicios en la nube y el almacenamiento de archivos en Google Drive y Dropbox"**, y a mi criterio el mismo cumple con los objetivos propuestos, según el protocolo.

Sin otro particular, me suscribo de usted,

Atentamente,



Edgar Santos
Ingeniero en Ciencias y Sistemas
Colegiado No. 5266
Asesor de Trabajo de Graduación

Edgar Santos
INGENIERO EN CIENCIAS Y SISTEMAS
Colegiado 5266

Guatemala 3 de Marzo de 2014

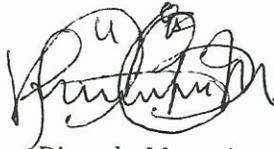
Ingeniero
Carlos Azurdia
Revisor de Trabajo de Graduación
Escuela de Ciencias y Sistemas
Facultad de Ingeniería

Respetable Ingeniero Azurdia:

Por este medio hago de su conocimiento que he revisado el Trabajo de Graduación de los estudiantes Guillermo Estuardo Urías Teque identificado con número de carnet 200819111 y Giancarlo Guilá Reina identificado con número de carnet 200819105, titulado: **"Implementación de tecnologías web y móviles para el uso de servicios en la nube y el almacenamiento de archivos en Google Drive y Dropbox"**, y a mi criterio el mismo cumple con los objetivos propuestos, según el protocolo.

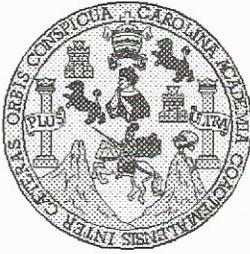
Sin otro particular, me suscribo de usted,

Atentamente,



Ricardo Israel Mazariegos Castillo
Ingeniero en Ciencias y Sistemas
Colegiado No. 12451

Ricardo Mazariegos
Ingeniero en Ciencias y Sistemas
Colegiado No. 12451
Asesor de Trabajo de Graduación



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 30 de Abril de 2014

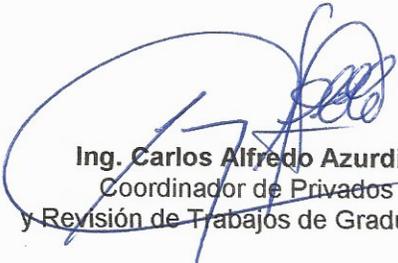
Ingeniero
Marlon Antonio Pérez Turk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación de los estudiantes **GUILLERMO ESTUARDO URÍAS TEQUE** con carné 2008-19111, y **GIANCARLO GUILÁ REINA** con carné 2008-19105, titulado: “IMPLEMENTACIÓN DE TECNOLOGÍAS WEB Y MÓVILES PARA EL USO DE SERVICIOS EN LA NUBE Y EL ALMACENAMIENTO DE ARCHIVOS EN GOOGLE DRIVE Y DROPBOX”, y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



E
S
C
U
E
L
A

D
E

C
I
E
N
C
I
A
S

Y

S
I
S
T
E
M
A
S

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS
TEL: 24767644

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, del trabajo de graduación **“IMPLEMENTACIÓN DE TECNOLOGÍAS WEB Y MÓVILES PARA EL USO DE SERVICIOS EN LA NUBE Y EL ALMACENAMIENTO DE ARCHIVOS EN GOOGLE DRIVE Y DROPBOX”**, realizado por los estudiantes GIANCARLO GUILÁ REINA y GUILLERMO ESTUARDO URÍAS TEQUE , aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAD A TODOS”

Ing. Marlon Antonio Pérez Türk

Director, Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 14 de noviembre 2014





El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **IMPLEMENTACIÓN DE TECNOLOGÍAS WEB Y MÓVILES PARA EL USO DE SERVICIOS EN LA NUBE Y EL ALMACENAMIENTO DE ARCHIVOS EN GOOGLE DRIVE Y DROPBOX**, presentado por los estudiantes universitarios: **Guillermo Estuardo Urías Teque y Giancarlo Guilá Reina**, y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, se autoriza la impresión del mismo.

IMPRÍMASE.

Ing. Murphy Olympo Paiz Recinos
Decano



Guatemala, noviembre de 2014

ACTO QUE DEDICO A:

Dios	Por darme la vida y guiarme por el camino correcto.
Mi madre	Porque me dio la vida y porque sin su amor, apoyo y esfuerzo, nunca lo hubiera logrado.
Mi abuela	Por su apoyo y amor incondicional, que me ayudó a alcanzar cada una de mis metas.
Mi hermana	Yolanda Guilá, por su apoyo incondicional y cariño durante el ciclo universitario y en cada uno de mis días.
Mi familia	A mis tíos Roberto Reina, Ruth de Reina, y primos Luis Fernando Reina, Jeffrey Reina y Jenifer Reina, gracias por su apoyo y consejos.
Mi novia	Cecilia García, por su apoyo y comprensión en cada situación de mi vida.
Mis amigos	Randy Ruano, Guillermo Urias, y todos mis demás amigos de la universidad, por su apoyo incondicional durante el ciclo universitario.

Giancarlo Guilá Reina

ACTO QUE DEDICO A:

Dios	Por permitirme vivir una vida llena de oportunidades.
Mis padres	Juan Guillermo Urias y Norma de Urías, por ser el motivo que me inspira a luchar día a día.
Mis hermanos	Christian y Helber Urías, por ser quienes me animan a continuar creciendo profesionalmente.
Mis tíos	Por sus sabios consejos y su incondicional apoyo en los buenos y malos momentos.
Mis primos	Por ser un gran ejemplo de vida y apoyo incondicional.
Mi abuelo	Pedro Teque, quien ha sido mi ejemplo a seguir por muchas razones.

Guillermo Urías Teque

AGRADECIMIENTOS A:

Dios	Porque continúa dándome vida, me llena cada día de bendiciones y ha permitido que alcance un peldaño más en mi vida profesional.
Universidad de San Carlos de Guatemala	Por ser mi casa de estudios y lugar de tantas experiencias vividas en el transcurso de mi carrera universitaria.
Facultad de Ingeniería	Por brindar las bases de profesionalismo que acompañan mi carrera.
Mis catedráticos	Que con empeño y dedicación me brindaron sus conocimientos y su valioso tiempo en las aulas; especialmente a los Ingenieros Edgar Santos y Ricardo Mazariegos, por su asesoría en este trabajo.

Giancarlo Guilá Reina

AGRADECIMIENTOS A:

**Universidad de San
Carlos de Guatemala**

Por ser el pilar que fundamentó mi carrera profesional.

Facultad de Ingeniería

Por brindar las bases del profesionalismo que acompañan mi carrera.

**Mis amigos de la
Facultad**

Yonatan Grajeda, Iris Us, Juan José Grajeda, Giancarlo Guilá, Leonel Rojas, María Lucero, Álvaro Cruz, Enrique Rezzio, Miguel Zacarías, Cecilia García, Ana Valesca Samayoa, Emmanuel Santiago, Axel Mazariegos, Roger Jiménez, Estuardo Álvarez, Alejandro Mérida y Helen Pineda, por la compañía y apoyo durante la carrera.

Guillermo Urías Teque

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	V
LISTA DE SÍMBOLOS	VII
GLOSARIO	IX
RESUMEN.....	XIII
OBJETIVOS.....	XV
INTRODUCCIÓN.....	XVII
1. ESTUDIO DE LA TECNOLOGÍA Y SU IMPACTO EN GUATEMALA.....	1
1.1. Almacenamiento en la nube	2
1.2. Sistema operativo Android.....	3
1.2.1. Historia de Android	3
1.2.2. Arquitectura de Android	4
1.2.2.1. Núcleo Linux.....	5
1.2.2.2. Librerías nativas	5
1.2.2.3. Entorno de ejecución	6
1.2.2.4. Entorno de aplicación	6
1.2.2.5. Aplicaciones.....	7
1.3. La teoría de la virtualización	7
2. IDENTIFICACIÓN DEL PROBLEMA Y SOLUCIÓN PLANTEADA	9
2.1. Antecedentes.....	10
2.1.1. Historia del almacenamiento.....	10
2.1.2. Historia del almacenamiento en la nube	13
2.2. Identificación del sistema operativo a utilizar.....	16
2.3. Descripción del problema	20

2.3.1.	Principales servicios de almacenamiento.....	20
2.3.1.1.	Dropbox.....	20
2.3.1.2.	Google Drive	21
2.4.	Comparativa entre plataformas	22
2.4.1.	Relación capacidad/precio	22
2.4.1.1.	Google Drive	23
2.4.1.2.	Dropbox.....	23
2.4.2.	Clientes	24
2.4.3.	Aplicación web	25
2.4.4.	Límite de tamaño.....	27
2.4.5.	Servicios adicionales.....	28
2.5.	Aspectos legales del almacenamiento en la nube	29
2.5.1.	Dropbox.....	29
2.5.2.	Google Drive	32
3.	DOCUMENTACIÓN BASE PARA EL DESARROLLO DE LA APLICACIÓN	37
3.1.	Componentes de una aplicación Android.....	37
3.1.1.	Actividades.....	37
3.1.2.	Intents.....	37
3.1.3.	Servicios.....	38
3.1.4.	Tarea asíncrona	38
3.1.5.	<i>Content providers</i>	38
3.1.6.	<i>Broadcast receivers</i>	38
3.1.7.	<i>Manifest</i>	39
3.2.	Conceptos involucrados en la comunicación con sistemas externos	40
3.2.1.	¿Qué es un sistema externo?	40
3.3.	API de desarrollo.....	41

3.3.1.	Kit de desarrollo de software o SDK (Software development kit)	41
3.3.2.	Google Drive SDK	42
3.3.2.1.	Principales usos.....	42
3.4.	Métodos de autorización.....	43
3.4.1.	OAuth (<i>open authorization</i>).....	43
3.4.1.1.	Antecedentes.....	44
3.5.	Documentación base para el desarrollo de la aplicación móvil	45
3.5.1.	Requerimientos para el desarrollo	46
3.5.1.1.	Android SDK.....	46
3.5.1.2.	Android Development Tools (ADT)	47
3.5.1.3.	Instalación y configuración.....	47
3.5.1.3.1.	Descarga e Instalación del Android SDK.....	47
3.5.1.3.2.	Android SDK Manager..	47
3.5.1.3.3.	Crear un nuevo proyecto en Eclipse	48
3.5.1.4.	Ambiente de realización de pruebas....	49
4.	MANUAL DE USO DE LA APLICACIÓN.....	51
4.1.	Componentes de una aplicación móvil en Android	51
4.1.1.	Vista.....	51
4.1.2.	<i>Layout</i>	51
4.1.3.	Actividad	52
4.1.4.	Servicio	52
4.1.5.	Intención	52
4.1.6.	Receptor de anuncios.....	53

4.1.7.	Proveedores de contenido.....	53
4.2.	Instalación del entorno de desarrollo.....	54
4.2.1.	Instalación de la máquina virtual Java.....	54
4.2.2.	Instalación basada en Eclipse con ADT Bundle	54
4.2.3.	Instalación basada en Eclipse de cada componente.....	55
4.2.3.1.	Instalación de Eclipse.....	55
4.2.3.2.	Instalación de Android SDK de Google.....	56
4.2.3.3.	Instalación del <i>plug-in</i> Android para Eclipse (ADT)	57
4.2.3.4.	Creación de un dispositivo virtual Android (AVD)	59
4.3.	Manual de uso de la aplicación	62
4.3.1.	Sitio web.....	62
4.3.2.	Descarga e instalación de la aplicación	65
4.3.3.	Uso de la aplicación	67
4.3.3.1.	Pantalla de ingreso.....	68
4.3.3.2.	Pantalla de registro	69
4.3.3.3.	Panel de opciones.....	70
4.3.3.4.	Pantalla de configuración	71
4.3.3.5.	Pantalla de fotografías	72
4.3.3.6.	Pantalla de favoritos.....	73
4.3.3.7.	Pantalla de archivos	74
4.3.3.7.1.	Menú de opciones.....	75
	CONCLUSIONES.....	77
	RECOMENDACIONES	79
	BIBLIOGRAFÍA.....	81

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Arquitectura del sistema operativo Android.....	4
2.	Plataformas móviles.....	17
3.	Porcentaje de teléfonos inteligentes vendidos según su sistema operativo	19
4.	Google Drive vs. Dropbox	25
5.	Aplicación web de Dropbox.....	26
6.	Tipos de archivos soportados por Google Drive	26
7.	Aplicación web de Google Drive	27
8.	Comparativa de tamaño, Google Drive vs. Dropbox	27
9.	Dropbox wiki.....	28
10.	Aplicaciones de Google Play para Google Drive.....	29
11.	Android SDK manager	48
12.	Creación de aplicación Android.....	49
13.	Android device chooser.....	50
14.	Espacio de trabajo en Eclipse	56
15.	Instalación de <i>plug-in</i> Android.....	58
16.	Preferencias de Eclipse.....	59
17.	Creación de un dispositivo virtual Android	60
18.	Android virtual device manager	61
19.	Emulación de dispositivo Android	61
20.	Página principal del sitio web.....	62
21.	Características de GCloud I	63
22.	Características de GCloud II	63

23.	Características de GCloud III	64
24.	Información y descarga de GCloud.....	64
25.	Beneficios de utilizar GCloud	65
26.	Descarga de la aplicación.....	66
27.	Autorización de permisos de la aplicación	67
28.	Pantalla de ingreso	68
29.	Registro en GCloud	69
30.	Panel de opciones	70
31.	Pantalla de configuración.....	71
32.	Pantalla de fotografías	72
33.	Pantalla de favoritos	73
34.	Pantalla de archivos.....	74
35.	Menú de opciones.....	75

TABLAS

I.	Comparativa de las principales plataformas móviles	18
II.	Comparativa de Google Drive vs. Dropbox.....	22
III.	Planes de almacenamiento de Google Drive	23
IV.	Planes de almacenamiento de dropbox.....	24
V.	Operaciones permitidas por el SDK de Google Drive	42

LISTA DE SÍMBOLOS

Símbolo	Significado
Mb	Cantidad de datos informáticos equivalente a 1024 bytes.
Gb	Cantidad de datos informáticos equivalente a 1024 Mb.
Tb	Es una unidad de almacenamiento de información equivalente a 1024 Gb.

GLOSARIO

<i>Add-on</i>	Es un paquete de actualización de software que agrega una funcionalidad específica.
Android	Sistema operativo basado en el kernel de Linux utilizado principalmente para dispositivos móviles con pantalla táctil y teléfonos inteligentes.
API	Se refiere por sus siglas en inglés al término <i>Application programming interface</i> . Es el conjunto de funciones y procedimientos de un lenguaje de programación específico.
APK	Se refiere por sus siglas en inglés al término <i>Application package file</i> . Es el tipo de archivo que contiene el código de una aplicación específica.
AVD	Se refiere por sus siglas en inglés al término <i>Android virtual device</i> . Es un software que realiza la emulación de un dispositivo android como un celular o una tableta electrónica.
<i>Cookies</i>	Es una pequeña información enviada por un sitio web y almacenada en el navegador del usuario.

CPU	Se refiere por sus siglas en inglés al término <i>Central processing unit</i> . Es el hardware dentro de un computador que se encarga de realizar las operaciones básicas aritméticas, lógicas y de entrada y salida del sistema.
Driver	Es el programa o conjunto de programas que brindan el soporte a nivel de <i>software</i> para el funcionamiento de algún dispositivo físico.
Framework	Es un conjunto estandarizado de conceptos y prácticas para realizar un software.
HTTP	Se refiere por sus siglas en inglés al término <i>Hypertext Transfer Protocol</i> . Es el protocolo utilizado por los navegadores web para poder enviar y recibir una página solicitada a través de internet.
IDE	Se refiere por sus siglas en inglés al término <i>Integrated development environment</i> . Es un programa informático por medio del cual se pueden crear aplicaciones de software.
IT	Agrupación de los elementos y las técnicas utilizadas en el tratamiento y la transmisión de información, internet y telecomunicaciones.
JRE	Se refiere por sus siglas en inglés al término <i>Java runtime environment</i> . Es un conjunto de programas

que permiten utilizar y desarrollar aplicaciones que utilicen el lenguaje de programación Java.

JVM	Se refiere por sus siglas en inglés al término <i>Java virtual machine</i> . Es el software encargado de ejecutar las aplicaciones desarrolladas en el lenguaje de programación Java.
Máquina virtual	Es un software que emula el funcionamiento real de un equipo de cómputo.
Oauth	Es un protocolo que permite la autorización segura de información a una aplicación.
SDK	Se refiere por sus siglas en inglés al término <i>Software development kit</i> . Es el conjunto de aplicaciones que permiten la creación de nuevas aplicaciones o de un software en específico.
SSL	Se refiere por sus siglas en inglés al término <i>Secure sockets layer</i> . Es un protocolo utilizado por las aplicaciones para poder enviar y recibir información por internet de manera segura.

RESUMEN

El acceso a la información se ha visto afectado por el auge que ha tenido internet en los últimos años. Internet ha revolucionado el acceso a la información, llevándola hacia el almacenamiento en la nube, lo cual permite guardar cualquier objeto digital en internet y que este sea accesible prácticamente desde cualquier dispositivo y en cualquier lugar del mundo, impulsando la creación de servicios que permitan gestionar los contenidos digitales que se almacenan en internet, de donde nacen servicios como Dropbox, Google Drive, entre muchos otros.

Estos servicios brindan la gestión de la información por medio de distintas soluciones innovadoras. La variedad de servicios para el almacenamiento en línea que existen en el mercado ha llevado a pensar en una aplicación móvil que unifique estos servicios, cuyo desarrollo se sustenta en la teoría de la virtualización, como lo es GCloud.

GCloud es una aplicación móvil que puede ser utilizada en cualquier dispositivo electrónico que utilice el sistema operativo Android. Por medio de esta aplicación se pueden unificar las cuentas y la información que se tiene almacenada en los servicios de Dropbox y Google Drive. La aplicación cuenta con una interfaz donde almacenar los documentos favoritos, filtros para imágenes digitales y gestión de todos los documentos almacenados, mostrándolos de manera transparente en relación con el servicio al cual pertenecen, así como la gestión de nuevas cuentas o la modificación de las existentes, proveyéndole al usuario una experiencia distinta y simplificada para el acceso de su información.

OBJETIVOS

General

Desarrollar una aplicación móvil para el sistema operativo Android que permita el acceso de forma genérica a la información almacenada en distintos servicios de almacenamiento en la nube, utilizando las interfaces de programación que brindan estos servicios.

Específicos

1. Implementar la gestión de cuentas de usuario y de la información almacenada en los servicios de almacenamiento en la nube de Dropbox y Google Drive.
2. Estudiar de forma legal las restricciones, el alcance y la funcionalidad que brindan las interfaces de programación de los servicios de almacenamiento en la nube de Dropbox y Google Drive.
3. Documentar el proceso de análisis, desarrollo e implementación de una aplicación móvil para el sistema operativo Android, que permita la gestión unificada de los servicios de almacenamiento en la nube Dropbox y Google Drive.

INTRODUCCIÓN

Las formas de almacenamiento tradicionales han estado vigentes desde el inicio de las computadoras y probablemente seguirán siendo válidas por mucho tiempo. Sin embargo, las tendencias tecnológicas actuales, el estilo de vida de las personas y los diferentes puestos de trabajo requieren que el almacenamiento de información evolucione, por lo que el acceso a los archivos desde diferentes dispositivos se ha vuelto una necesidad de primer nivel.

Los servicios de almacenamiento en la nube han emergido como la solución a esta necesidad. Estos nacen con el objetivo de brindar la mejor experiencia para el almacenamiento en línea, ofreciendo seguridad de la información, integración con diferentes plataformas y sistemas operativos, espacio de almacenamiento para cada necesidad de usuario, procesamiento *online* y *offline* de la información, sincronización de archivos y carpetas, entre muchas otras funcionalidades innovadoras.

Muchos usuarios de este tipo de servicios utilizan varias cuentas de usuario para almacenar su información, debido a la diversidad de ventajas y funcionalidades que presenta cada uno de los productos disponibles en el mercado, cuyas características generan ventajas significativas entre cada uno de ellos.

Actualmente cada proveedor de almacenamiento en la nube proporciona sus medios de acceso a través de aplicaciones móviles dedicadas, lo cual conlleva al uso de varias aplicaciones por parte del usuario final, ocasionando que la información se encuentre distribuida.

Es por esto que nace la necesidad de desarrollar una aplicación que pueda unificar cada una de las características que proporcionan los servicios en la nube y gestionar toda la información desde una sola aplicación móvil.

1. ESTUDIO DE LA TECNOLOGÍA Y SU IMPACTO EN GUATEMALA

En los últimos años la tecnología se desarrolla tan rápido que ha llegado a ser un factor decisivo en la globalización. Esto ha hecho que muchos países se valgan de sus posibilidades económicas para hacer uso de la tecnología y alcanzar así un nivel de vida más adecuado para sus habitantes.

Las tecnologías de la información se refieren a todas aquellas tecnologías que permitan el desarrollo y operación de los sistemas de información. En inglés, el término es representado por las siglas IT. Estas tecnologías, como su nombre lo indica, se basan básicamente en el uso extensivo de la tecnología como un medio para manejar datos, que relacionados, se convierten propiamente en información. En este manejo de datos se emplean las telecomunicaciones, las cuales se han desarrollado de la mano de la tecnología, debido a que sus creaciones, usos, implementaciones y aplicaciones, influyen directamente en los avances tecnológicos.

En Guatemala, un país en vías de desarrollo, prácticamente no se ha participado en el desarrollo tecnológico que se está llevando alrededor del mundo. Esto es ocasionado por ser un país pobre que cuenta con poca infraestructura en comunicaciones, poco acceso a la educación, entre otros factores como: económicos, sociales, culturales y muchos más. Estos problemas se ven reflejados, ya que a pesar de ser un país pequeño, existen comunidades que se encuentran muy aisladas del resto del país y por ende del resto del mundo.

Otro factor muy importante a considerar es el analfabetismo que existe en Guatemala, afectando directamente a que la información no se encuentra disponible para toda la población y en consecuencia, existe un analfabetismo informático.

1.1. Almacenamiento en la nube

Las formas de almacenamiento de la información han significado años de estudio y evolución en la historia de la computación. Las empresas siempre han buscado formas de mantener su información centralizada, especialmente cuando necesitan mantener información sensible bajo control.

La computación en la nube como tal, es un paradigma emergente que afecta diversos aspectos de la computación, como el almacenamiento de datos y la ejecución de aplicaciones. Se basa en la virtualización de los servicios que le permiten proporcionar ciertas ventajas frente al sistema tradicional físico; algunas de ellas son: la percepción para el usuario de contar con recursos infinitos, y mejora en la gestión de recursos, haciendo ver la eficiencia de consumir únicamente aquellos que son necesarios.

Un concepto importante que va de la mano con el almacenamiento en la nube es que el usuario únicamente pagará por el espacio que él considere necesario, lo cual es adecuado, ya que este paga únicamente lo que utiliza.

Los datos como servicio, o bien mencionados por sus siglas en inglés Daas, surgen como un nuevo concepto de la computación en la nube, ofreciendo una solución de virtualización del almacenamiento.

La característica más importante de este tipo de servicio es que provee al usuario de numerosas interfaces que le permiten tener el control de sus datos, sin que él tenga que preocuparse por la forma en la que estos son manejados internamente, ya que muchas veces estos datos pueden estar almacenados en distintas ubicaciones físicas, lo cual interesa muy poco al usuario final.

1.2. Sistema operativo Android

Android es un sistema operativo utilizado en cientos de millones de dispositivos alrededor del mundo. Es una de las plataformas de más rápido crecimiento en los últimos años. Este sistema operativo está basado en Linux, del cual depende para los servicios básicos del sistema. Junto con el núcleo Linux está Dalvik, que es una máquina virtual de Java, la cual se ejecuta sobre el núcleo del sistema. Este sistema operativo está basado principalmente para dispositivos con pantalla táctil y provee todas las herramientas para desarrollar y crear aplicaciones, así como una tienda en línea para distribuir las mismas.

1.2.1. Historia de Android

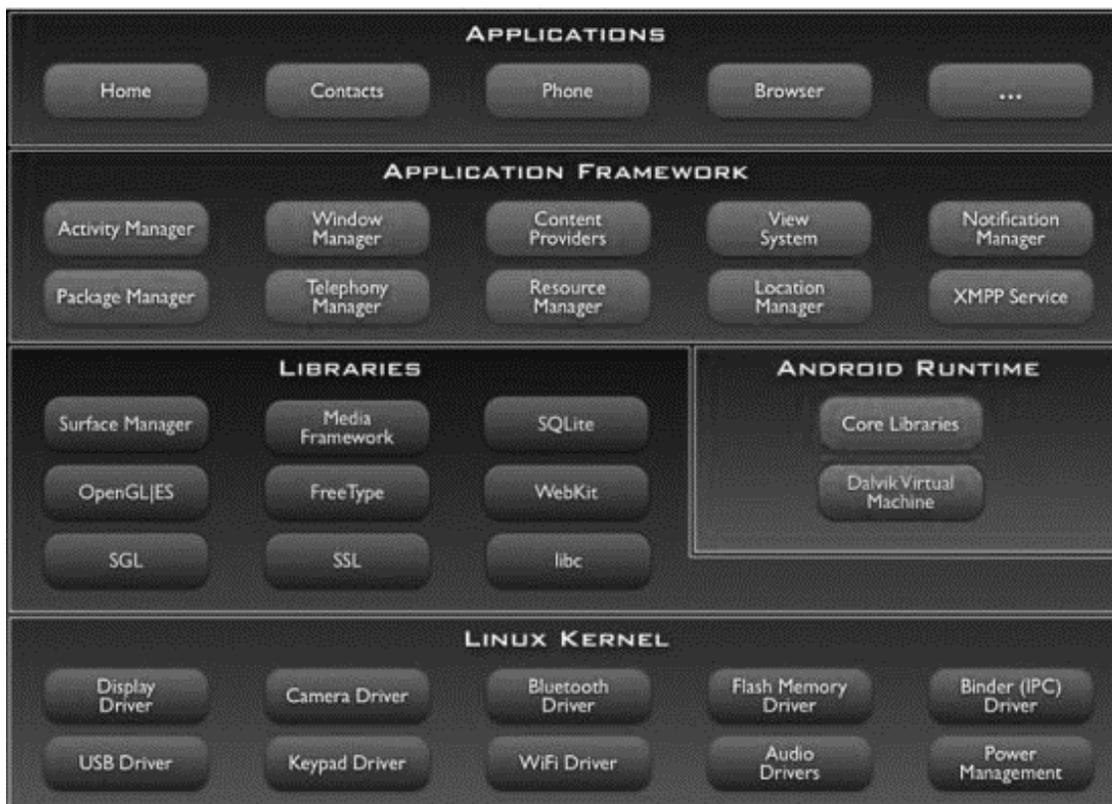
La compañía Android Inc. fue fundada en Palo Alto, California en octubre de 2004 por Andy Rubin, Rich Miner, Nick Sears y Chris White. La compañía desarrolló el sistema operativo Android hasta que en junio de 2005 fue adquirida por Google.

El 5 de noviembre de 2007, un consorcio de varias compañías llamado Open Handset Alliance, presentó Android, un sistema operativo para móviles. Su modelo de desarrollo es de software libre y estaba construido sobre la versión 2.6 de Linux.

1.2.2. Arquitectura de Android

La arquitectura del sistema operativo Android está formada por varias capas, lo cual genera una gran ventaja para el desarrollador, ya que por medio del uso de librerías ya no se necesita programar a bajo nivel las funcionalidades de la aplicación.

Figura 1. **Arquitectura del sistema operativo Android**



Fuente: <http://picandocodigo.net/2007/android-la-nueva-plataforma-de-desarrollos-moviles/>.

Consulta: julio de 2013.

1.2.2.1. Núcleo Linux

Es una capa de atracción entre el hardware y el resto de las capas; por lo tanto es la única capa que depende del hardware.

Esta capa proporciona el manejo de la memoria y el soporte de *drivers* para dispositivos. De esta forma el desarrollador no debe acceder directamente a esta capa, ya que puede utilizar librerías que están disponibles en otras capas.

1.2.2.2. Librerías nativas

Las librerías están escritas en C/C++ y están compiladas específicamente para la arquitectura de hardware del dispositivo.

Las principales librerías que pueden ser utilizadas por el programador para el desarrollo de sus aplicaciones son:

- System C Library
- Media framework
- Surface manager
- SGL
- SQLite
- FreeType

1.2.2.3. Entorno de ejecución

Los ingenieros de Google diseñaron la máquina virtual Dalvik para los dispositivos Android. Esta máquina virtual se adapta a la memoria y procesador del dispositivo, ya que estos no soportan la máquina virtual estándar de Java.

Dalvik es una variación de la máquina virtual de Java. La principal diferencia con la máquina virtual de Java es que el *bytecode* no es compatible con el de Java. El *bytecode* es el conjunto de instrucciones que la máquina virtual ejecuta, es el resultado de usar el compilador del lenguaje Java.

La ventaja está en que las aplicaciones se compilan una sola vez y se generan los ejecutables con el SDK; estas pueden ejecutarse en cualquier dispositivo Android.

1.2.2.4. Entorno de aplicación

Esta capa permite la reutilización de componentes, sujetos a las reglas de seguridad del entorno de aplicación. El SDK de Android provee las herramientas y los API necesarios para el desarrollo de la aplicación, utilizando el lenguaje Java. Los servicios que incluye son:

- Vistas: son usadas para crear la parte gráfica de una aplicación; incluye listas, botones, cuadros de texto, etc.
- Controlador de recursos: proporciona acceso a recursos que no son de código, como por ejemplo: archivos, cadenas, gráficas y plantillas.

- Controlador de actividades: se utiliza para administrar el ciclo de vida de las actividades ejecutadas por la aplicación.
- Controlador de notificaciones: se utiliza para enviar notificaciones al usuario, mostrando alertas en la barra de estado.

1.2.2.5. Aplicaciones

En esta capa están todas las aplicaciones instaladas en el dispositivo; estas aplicaciones son las que se ejecutan en la máquina virtual Dalvik, las cuales están escritas en Java, aunque también están las aplicaciones nativas escritas en C o C++.

1.3. La teoría de la virtualización

La rápida comercialización y adopción de las tecnologías de internet han provocado cambios en los procesos convencionales de las organizaciones y el comportamiento individual de las personas. Los procesos convencionales que han sido manejados por medio de mecanismos físicos están siendo sustituidos por procesos virtualizados.

Este fenómeno es conocido como “Proceso de virtualización” y está sucediendo en diferentes contextos de la sociedad. Los cursos virtuales, el trabajo a distancia, las ventas y compras por medio de portales de internet, las redes sociales, y la telefonía inteligente forman parte de los ejemplos claros del proceso de virtualización.

La facilidad con la que un proceso se virtualiza depende enteramente de su naturaleza y de las necesidades que existen para provocar que la

virtualización de dicho proceso se ponga en marcha. La teoría de la virtualización se basa sobre cuatro principios fundamentales, los cuales son:

- Requisitos sensoriales
- Requisitos de relación
- Requerimientos de identificación y control
- Requisitos de sincronización

La virtualización es el resultado de separar el software de los sistemas físicos sobre los que se encuentra instalado. La razón por la que el proceso de virtualización suena atractivo para un usuario es el bajo costo de mantenimiento y el principio de la “no instalación” de sistemas, por lo que al contratar un servicio virtualizado, este estará listo para su uso en cuanto se cumpla con las normas impuestas por la empresa que ofrece el servicio.

Otra de las características de este tipo de servicios es que el usuario no tiene que preocuparse por la implementación concreta de los servicios de la nube, ni tener en cuenta el hardware asociado a ellos o las modificaciones en el hardware que impliquen grandes cambios, dependiendo de las aplicaciones instaladas.

2. IDENTIFICACIÓN DEL PROBLEMA Y SOLUCIÓN PLANTEADA

En la actualidad, la generación de información ha sobrepasado los límites de la imaginación; más del 70 % de la información digital que se genera proviene de los usuarios finales y son las empresas proveedoras de servicios quienes se responsabilizan por el almacenamiento, protección y gestión de más del 85 % de dicha información.

Se estima que entre el 2009 y 2020, la información generada por año crecerá 44 veces a lo que se genera actualmente. También se estima que en el 2020 más de 1/3 de la información generada al año residirá o atravesará la nube.

Se calcula que un incremento de la inversión en innovación podría generar un aumento de los ingresos de las organizaciones en más de 1 billón de dólares de aquí a finales del 2014; esto basándose en el uso que hacen las empresas de los servicios en la nube para reducir la parte del presupuesto de TI que se destina al mantenimiento de los sistemas.

La virtualización de archivos ha venido a ayudar al manejo de la información, tanto para las personas individuales como para las empresas; actualmente existen muchas opciones que permiten la virtualización de archivos. Estos presentan planes de virtualización tanto para usuarios finales, como para empresas e instituciones a precios considerables.

2.1. Antecedentes

El término de virtualización y computación en la nube es algo que hace algunos años no se conocía ni se llegaba a pensar en el potencial que esto podría ofrecer a los sistemas informáticos. Sin embargo es necesario conocer la historia y la evolución de la computación hasta llegar al punto de lo que se conoce hoy en día como computación en la nube.

2.1.1. Historia del almacenamiento

Las formas de almacenamiento de la información, han significado años de estudio y evolución en la historia de la computación. Las empresas siempre han buscado formas de mantener su información centralizada, especialmente cuando necesitan mantener información sensible bajo control.

El almacenamiento de la información fue necesario desde el inicio de la computación, ya que era un requisito estrictamente necesario para el procesamiento de datos.

En 1725, un trabajador textil con el nombre de Bouchon Basile utilizó tarjetas perforadas para controlar los telares. Con esto, el telar puede leer, ejecutar y automatizar la configuración de patrones. Por primera vez, surgió una máquina que utilizara un dispositivo para leer el código físico y con ello ejecutar un proceso. La mayoría de la gente piensa en las tarjetas perforadas como una forma de reloj para controlar la entrada y salida del proceso. También fueron utilizados para cargar el sistema operativo. Como la mayoría de las cosas, tuvo un comienzo humilde que hizo el día a día los procesos más fáciles. No fue sino hasta 1890, cuando Herman Hollerith usó una tarjeta perforada para grabar y leer datos de un ordenador.

Se utilizó para completar el censo de 1890 en solo un año. Antes de esto, el censo de 1880 tardó ocho años.

En 1947, por primera vez se utilizó el acceso aleatorio a una unidad de almacenamiento digital. Su velocidad era de 1,2 milisegundos por instrucción; poseía una capacidad de almacenamiento de 0.0525 Kb y era llamada “Williams Tube”.

Drum Memory fue la forma de almacenamiento de la época de 1950, con un almacenamiento de 10 Kb; fue el precursor de la bandeja de disco duro moderno.

En 1951, Uniservo fue presentado comercialmente, utilizando una unidad de cinta para el almacenamiento; su capacidad era de 128 bits por pulgada.

En 1956, IBM presentó su modelo IBM 350 con una capacidad de almacenamiento de 4.4 Mb. Considerado el primer disco duro “moderno” con cincuenta discos de 24” que se hacen girar a 1.200 rpm.

El cassette con una capacidad de almacenamiento de 660 Kb por lado, hizo su aparición en 1972; fue utilizado como una forma barata de guardar los datos; este fue sustituido poco a poco por los disquetes, siendo esta una tecnología aún más barata.

Una de las unidades más famosas ha sido el disco *Floppy* de 5.25”, con una capacidad de almacenamiento de 1.2 Mb; fue uno de los primeros dispositivos de almacenamiento producidos en masa. Se utilizó para guardar documentos e incluso cargar.

Uno de los dispositivos de almacenamiento que mayor uso ha tenido y que aún sigue vigente es el CD-R, presentado en 1990 con una capacidad de almacenamiento de 700 Mb; este formato es todavía ampliamente utilizado. Su gran capacidad de almacenamiento y precios más bajos ayudaron a superar el disquete y sobrevivir a la subida de los discos Zip y Jaz. En 1993 hizo su aparición el MiniDisc MD Data, disco magneto-óptico digital desarrollado en los 90, por la multinacional de origen japonés Sony, de menor tamaño que los CD convencionales y mayor capacidad en comparación.

Los discos Zip son un formato todavía ampliamente utilizado. Su gran capacidad de almacenamiento y precios más bajos ayudaron a superar el disco, ya que contaban con una capacidad de 100 Mb.

El disco Seagate Barracuda se considera como la primera unidad de 7200 rpm; velocidad que se convirtió en el estándar actual de las unidades de almacenamiento; poseía una capacidad de 2.5 Gb.

En 1999 aparece el IBM 170 MicroDrive, unidad famosa entre los primeros iPods con un almacenamiento de 170 Mb.

Un dispositivo ampliamente empleado en cámaras digitales y dispositivos móviles fue la SD Card, con un almacenamiento de 32 Mb. Delgado, con una parte más pequeña, este dispositivo ya no está restringido por el tamaño físico de la unidad. Este se dio a conocer en el año 2000.

Solid-State Drive, dispositivo sin partes móviles, más rápido en tiempos de arranque y menos espacio físico, es ahora la nueva forma de almacenar medios digitales. Está disponible desde el año 2008 y su capacidad de almacenamiento va desde los 64 Gb.

Actualmente, se tiene la tendencia de utilizar un nuevo “Servicio” de almacenamiento, el cual es conocido como “*Cloud Storage*” o almacenamiento en la nube, servicio que se brinda en línea; los consumidores pueden acceder al almacenamiento casi ilimitado desde cualquier dispositivo conectado a internet.

2.1.2. Historia del almacenamiento en la nube

El concepto de computación en la nube surgió a finales del siglo XX como una nueva perspectiva de un grupo de investigadores. Pero fue J.C.R. Licklider y John McCarthy quienes introdujeron el concepto de "Computación en la nube"; durante el discurso que presentó John McCarthy en 1961 para celebrar el centenario del Instituto Tecnológico de Massachusetts (MIT por sus siglas en inglés).

La computación en la nube forma parte de la evolución natural de la adopción de la virtualización, la arquitectura orientada a servicios y utilidad del cómputo. Lo que se busca con la computación en la nube es que los usuarios finales ya no necesiten tener conocimientos o control sobre la infraestructura de la tecnología que están utilizando, sino que, únicamente hagan uso de los servicios, convirtiéndose en personas más productivas al no tener que preparar el ambiente que necesiten para trabajar; al contrario, cuando hagan uso del servicio únicamente deben preocuparse por la esencia y las capacidades que el servicio contratado les permite tener.

La idea expuesta por John McCarthy en la ya mencionada conferencia del centenario del MIT, era que el tiempo compartido (*Time-sharing*) de las computadoras podría venderse como un servicio, tal es el caso del agua o la electricidad.

Esta idea fue muy popular en la época de 1960, tanto que las empresas comenzaron a proporcionar recursos para ofrecerlos como oficinas de servicios.

Esta popularidad provocó que los sistemas de tiempo compartido proporcionaran a los usuarios un ambiente operacional muy completo, el cual incluía desde editores de texto, entornos de desarrollo integrado para lenguajes de programación, paquetes de programas informáticos, almacenamiento de archivos hasta impresión masiva. Para hacer uso de estos servicios los usuarios debían pagar el costo de alquiler por la terminal, las horas de tiempo de conexión, tiempo del CPU y kilobytes mensuales de almacenamiento en disco. Sin embargo, a mediados de los años 70s se pensó que el hardware, software y las tecnologías de comunicación aún no estaban preparados para brindar el servicio que debían, por lo que la popularidad de estos centros se desvaneció.

En 1962, Joseph Carl Robnett Licklider, físico, matemático, psicólogo e informático estadounidense, planteó su idea sobre computación interactiva; esto lo llevó a desarrollar satisfactoriamente lo que sería el precursor del internet, la red ARPANET, mientras era director de Ciencias del Comportamiento de Mando y Control de Investigación en ARPA. Su visión llevó a un nuevo concepto de computación en el que se podían compartir recursos y mejorar las capacidades de la computación; su principal idea era aprovechar las capacidades de la computación y dar a conocer que dichas capacidades podían ser utilizadas para desarrollar tareas que no fueran únicamente cálculos.

En 1996, Douglas Parkhill mejor conocido por ser el pionero en lo que ahora se conoce como "computación en la nube", en su libro llamado "El desafío de la utilidad computacional" realizó un estudio profundo sobre muchas de las características actuales de la computación en la nube.

En 1998, Herb Grosch afirmaba que "las economías eficientes y adaptables podrían alcanzar su objetivo si confían en centros de datos centralizados en lugar de confiar en el almacenamiento de unidades"; con esta afirmación ya daba pie a confiar en la computación en la nube, sin embargo una de sus afirmaciones también decía que "la potencia de una computadora es proporcional al cuadrado de su precio"; esta afirmación es conocida como "Ley Grosch" la cual, según un estudio posterior, no es correcta sobre el modelo del costo de la computación en la nube.

Uno de los pioneros en la computación en nube fue Salesforce.com, que en 1999 introdujo el concepto de entrega de aplicaciones empresariales a través de una sencilla página web. Amazon fue el siguiente en lanzar Amazon Web Service en el 2002. Posteriormente Google introdujo su nuevo servicio Google Docs en el 2006, que realmente trajo la computación en la nube a la vanguardia de la conciencia del público. En el 2006 también se introdujo el Elastic Compute Cloud de Amazon (EC2) como un servicio web comercial que permitió a las empresas pequeñas y particulares alquilar equipos en los que pudieran ejecutar sus propias aplicaciones informáticas, aludiendo a la forma de computación en la nube como un servicio.

La evolución de los servicios en la nube fue seguida por una colaboración de toda la industria informática en 2007; entidades como Google, IBM y una serie de universidades de los Estados Unidos se vieron involucradas de lleno en esta evolución.

Luego vino Eucalyptus en el 2008, como la primera plataforma de código abierto compatible con el API-AWS para el despliegue de nubes privadas, seguido por OpenNebula, el primer software de código abierto para la implementación de nubes privadas e híbridas.

Microsoft se integró a esta tecnología hasta el 2009 con el lanzamiento de Windows Azure. Luego en el 2010 proliferaron servicios en distintas capas de servicio: cliente, aplicación, plataforma, infraestructura y servidor. En el 2011, Apple lanzó su servicio iCloud, un sistema de almacenamiento en la nube para documentos, música, videos, fotografías, aplicaciones y calendarios, con la cual prometió cambiar la forma en que se usa la computadora.

2.2. Identificación del sistema operativo a utilizar

Debido a la variedad de sistemas operativos disponibles en el mercado, es necesario realizar una comparación de las características principales que cada plataforma móvil ofrece en la actualidad. De esta manera se puede obtener la solución más viable para la realización de una aplicación móvil.

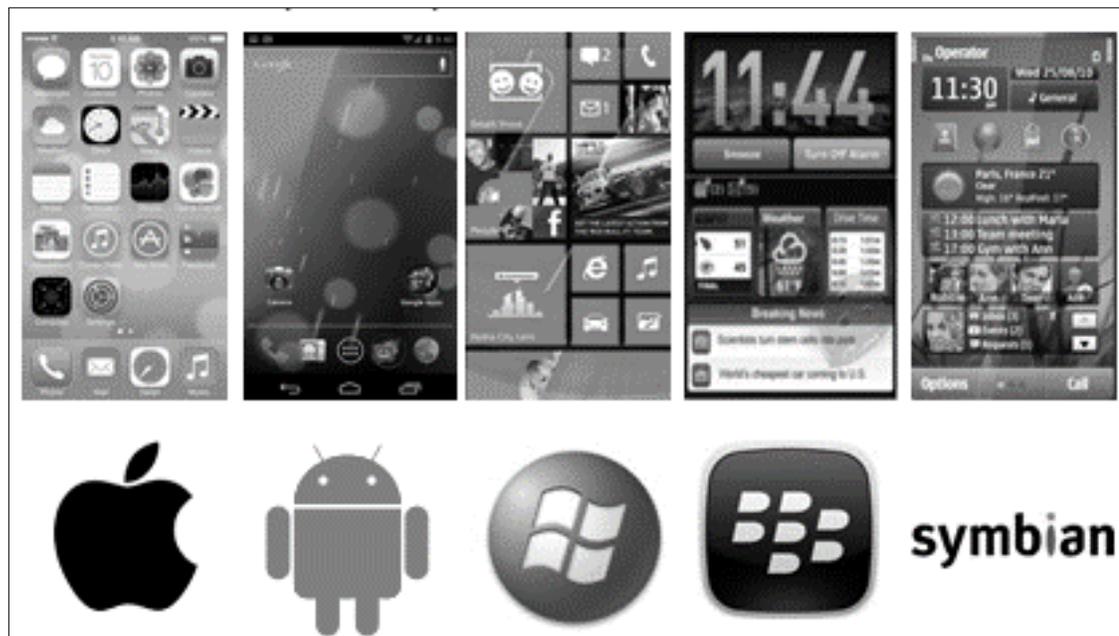
La variedad de sistemas operativos que funcionan para dispositivos móviles varía desde aspecto, características, funcionalidad, personalización, usabilidad y muchos otros factores. Es por ello que es necesario elegir el que más se adapte a las necesidades de la aplicación móvil.

Entre las principales plataformas móviles disponibles actualmente en el mercado se encuentran:

- Apple
- Android
- Windows Mobile
- Blackberry
- Symbian
- Firefox OS

Las plataformas anteriores se ilustran a continuación:

Figura 2. **Plataformas móviles**



Fuente: elaboración propia.

Cada plataforma móvil provee al usuario final y al programador de aplicaciones ciertas funcionalidades especiales que las hacen destacar una sobre otra; sin embargo, es necesario elegir entre la que más funcionalidad ofrece para poder desarrollar una aplicación óptima y que brinde la mejor experiencia al usuario final.

En la siguiente tabla se hace una comparación entre los sistemas operativos mencionados anteriormente, donde se evalúan las principales características en que se basan los programadores para elegir una plataforma.

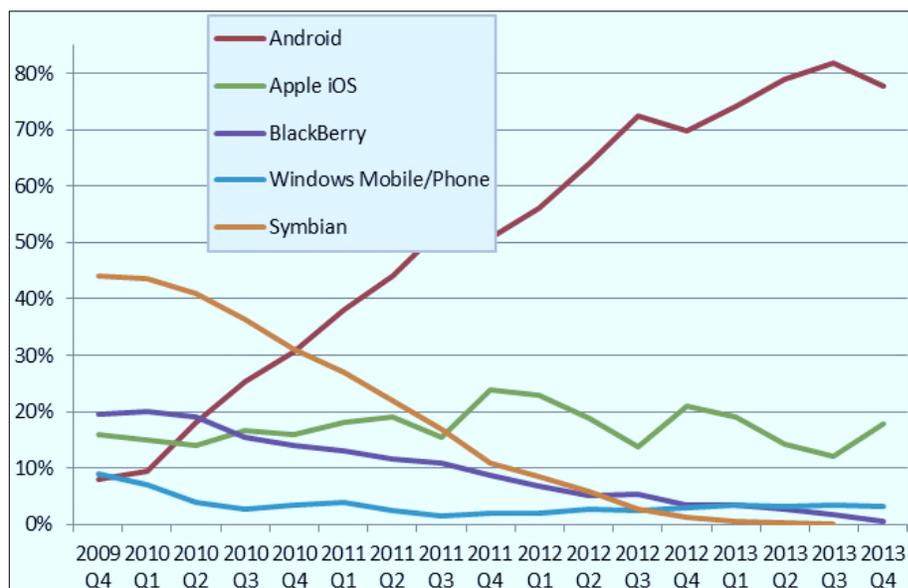
Tabla I. **Comparativa de las principales plataformas móviles**

	Apple iOS 7	Android 4.3	Windows Phone 8	BlackBerry OS 7	Symbian 9.5
Compañía	Apple	Handset	Microsoft	RIM	Symbian Foundation
Núcleo del SO	Mac OS X	Linux	Windows NT	Mobile OS	Mobile OS
Licencia de software	Propietaria	Software libre	Propietaria	Propietaria	Software libre
Variedad de dispositivos	Modelo único	Muy alta	Media	Baja	Muy alta
Soporte memoria externa	No	Si	Si	Si	Si
Tienda de aplicaciones	App Store	Google Play	Windows Marketplace	BlackBerry App World	Ovi Store
Número de aplicaciones	825,000	850,000	160,000	100,000	70,000
Coste publicar	\$ 99 / año	\$ 25 una vez	\$ 99 / año	Sin coste	\$ 1 una vez
Actualización automática del S.O.	Si	Depende del fabricante	Depende del fabricante	Si	Si
Máquina virtual	No	Dalvik	.net	Java	No
Aplicaciones nativas	Siempre	Si	Si	No	Siempre
Lenguaje de programación	Objective C, C++	Java, C++	C#, muchos	Java	C++
Plataforma de desarrollo	Mac	Windows, Mac, Linux	Windows	Windows, Mac	Windows, Mac, Linux

Fuente: <https://cienciaedu.wordpress.com/2012/09/20/comparacion-de-sistemas-operativos-moviles/>. Consulta: julio de 2013.

Otro aspecto fundamental a la hora de comparar las plataformas móviles es su cuota en el mercado. En la siguiente gráfica se puede ver un estudio realizado por la empresa Gartner Group, donde se muestra la evolución del mercado de los sistemas operativos para móviles, según el número de terminales vendidos.

Figura 3. **Porcentaje de teléfonos inteligentes vendidos según su sistema operativo**



Fuente: <http://www.gartner.com/technology/home.jsp>.

Consulta: julio de 2013.

De la gráfica anterior se puede destacar: el importante descenso de ventas de la plataforma Symbian de Nokia y el declive continuo de BlackBerry; cómo la plataforma Windows parece no aumentar mayor porcentaje, cómo Apple tiene asegurada una cuota en el mercado y finalmente el ascenso de la plataforma Android, que le ha permitido alcanzar en dos años una cuota de mercado superior al 75 %.

Por las razones listadas anteriormente es que se ha escogido la plataforma Android para el desarrollo de la aplicación, la cual brindará al usuario final la mejor experiencia en relación con las otras plataformas móviles y el impacto que tendrá en el mercado de aplicaciones para cada plataforma.

2.3. Descripción del problema

Existen muchas aplicaciones que permiten utilizar servicios de computación en la nube, sin embargo no todos brindan la capacidad requerida para cierto volumen de información, así como también existen unos más completos que otros.

2.3.1. Principales servicios de almacenamiento

A continuación se establece una comparación entre los dos principales servicios que se encuentran en el mercado: Dropbox y Google Drive.

2.3.1.1. Dropbox

Es uno de los sistemas de almacenamiento en la nube más completos que existen hasta el momento; entre sus propiedades se encuentra la sincronización automática de carpetas, la cual, hasta hace un tiempo debía utilizar una carpeta por defecto, cosa que ha cambiado y ahora permite especificar qué carpeta se quiere sincronizar con el servicio asociado.

Entre los sistemas operativos para los cuales se tiene soporte de Dropbox se encuentran sistemas operativos de escritorio como: Linux, Windows y Mac; así como sistemas operativos móviles tales como Android, BlackBerry y iOS.

Dropbox provee de un espacio de almacenamiento de 2 Gb para clientes que desean utilizar el espacio de almacenamiento de forma gratuita, permitiendo expandirlo hasta un total de 18 Gb por medio de técnicas de referencia del servicio a usuarios conocidos que acepten hacer uso del servicio por primera vez; esta modalidad irá expandiendo el servicio 500 Mb por cada referencia aceptada. Del mismo modo, existen tarifas de pago para los usuarios que deseen expandir su espacio de almacenamiento, variando las cuotas, dependiendo del espacio solicitado, el cual, puede variar de 100, 200 o 500 Gb por una cuota de US \$9.99 mensuales.

Existen paquetes específicos para empresas, las cuales incluyen planes de 1 Tb para 5 usuarios, herramientas administrativas y facturación centralizada con un costo de US\$ 795.00; este precio incluye posibilidades de recuperación de información, asistencia en vivo por teléfono o correo electrónico, mayor control sobre las opciones de compartición de archivo y enlaces, así como, integración con *active directory*, informes de actividad del equipo, gestión de equipos y desvinculación remota de dispositivos.

2.3.1.2. Google Drive

Es, definitivamente, el servicio de almacenamiento en la nube más completo, ya que por ser parte de las tecnologías Google, permite establecer integración con múltiples servicios, además de contar con mejores propiedades, características y operaciones sobre archivos; estas son: la edición de archivos de forma concurrente y en línea, visualización en línea, compartir archivos entre usuarios, creación de archivos en línea sobre la misma plataforma, chat grupal entre los usuarios que editan un archivo en un momento dado y compatibilidad con múltiples plataformas para la instalación de clientes que manejen el sistema de archivos relacionado.

2.4. Comparativa entre plataformas

La siguiente tabla muestra la comparativa entre los servicios de Google Drive y dropbox.

Tabla II. **Comparativa de Google Drive vs. Dropbox**

Google Drive	Dropbox
Es compatible con sistemas Windows, OS X, Linux, iOS, Android y Blackberry.	Es compatible con sistemas Windows, OS X, Linux, Android.
Se sincroniza a través de una carpeta fija donde se pueden colocar todos los datos.	Posee un navegador de archivos, visor de imágenes y capacidad de edición, con la interfaz de Google Docs.
Se puede navegar entre los archivos en una interfaz minimalista.	Integración con Gmail y Google+.
Ofrece 500 MB adicionales por cada persona que use el enlace para crear una cuenta.	Los desarrolladores podrán usar Google Drive para integrar servicios y aplicaciones web.
Se puede adquirir 50 GB por 99 dólares al año; 100 GB están valorizados en 200 al año.	Se puede adquirir hasta 25 GB más de alojamiento por 2.49 dólares al mes. Es renovable.

Fuente: <http://www.adslzone.net/>. Consulta: julio de 2013.

2.4.1. Relación capacidad/precio

Es sin duda lo más importante para los usuarios. Establecer la mejor relación entre la capacidad de la nube, y lo que se pagará por ella se convierte en el factor determinante a la hora de elegir entre un servicio u otro. Tanto Dropbox como Google Drive ofrecen numerosos planes de almacenamiento que van desde las cuentas gratuitas, con una capacidad mínima, a otras que alcanzan varios terabytes.

2.4.1.1. Google Drive

Google Drive ofrece en total 10 planes diferentes. En la figura que se muestra a continuación se puede ver el espacio de almacenamiento ofrecido y su coste en dólares. A diferencia de Dropbox, en Google Drive, la capacidad de la cuenta gratuita es inamovible.

Tabla III. Planes de almacenamiento de Google Drive

Cantidad de espacio	Costo
5 GB	Gratis
25 GB	2,49 \$ al mes
100 GB	4,99 \$ al mes
200 GB	9,99 \$ al mes
400 GB	19,99 \$ al mes
1 TB	49,99 \$ al mes
2 TB	99,99 \$ al mes
4 TB	199,99 \$ al mes
8 TB	399,99 \$ al mes
16 TB	799,99 \$ al mes

Fuente: <https://support.google.com/drive/answer/2375123?hl=es>.

Consulta: julio de 2013.

2.4.1.2. Dropbox

La oferta de planes de Dropbox tiene otro enfoque, y permite elegir entre el pago mensual o anual, con la ventaja de que esta última proporciona ese 17 % de descuento en el precio final del que se ha hablado. La cuenta gratuita es de 2 GB, pero se puede llegar hasta los 18 GB por diferentes vías, como recomendar el programa a otros usuarios o participando en encuestas.

Tabla IV. **Planes de almacenamiento de dropbox**

Cantidad de espacio	Costo
2 Gb a 18 GB	Gratis
100 GB	9,99 \$ al mes
200 GB	19,99 \$ al mes
500 GB	49,99 \$ al mes
1 TB en adelante	795 \$ al mes

Fuente: <https://www.dropbox.com/>.

Consulta: julio de 2013.

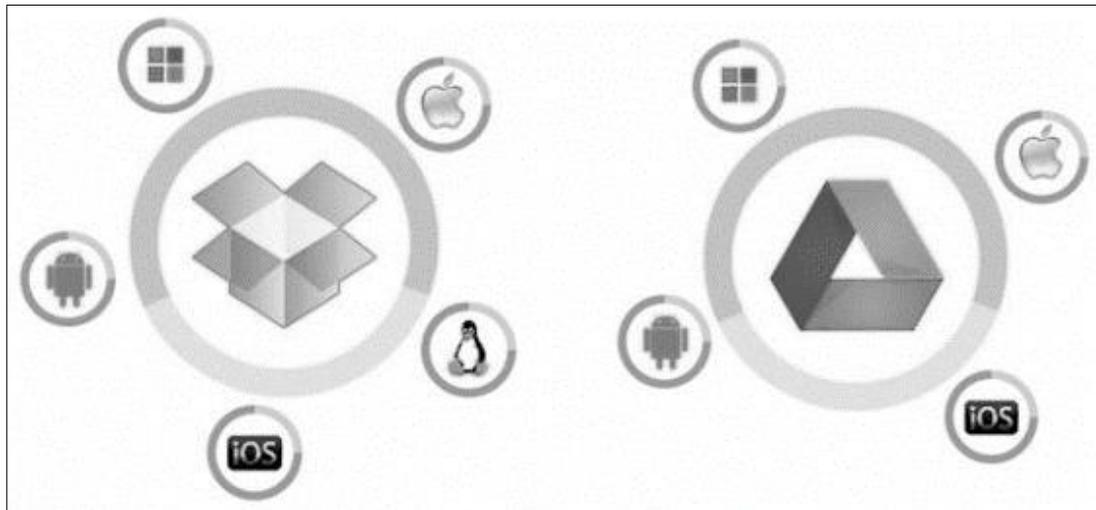
La otra diferencia respecto de Google Drive es que a partir de 1 TB la suscripción pasa a ser obligatoriamente anual para un mínimo de 5 ordenadores. Eso sí, no hay límite de tamaño, se podrá tener tanto como se necesite.

2.4.2. Clientes

Ambos programas cuentan con sus propios clientes de escritorio y aplicaciones móviles para gestionar los archivos de la nube y utilizan el mismo sistema de sincronización. Están presentes en Windows, Mac, Android e iOS.

Sin embargo, Dropbox ofrece también cliente para Linux. No dejan de ser carpetas o aplicaciones que sincronizan los archivos con la nube, pero son indispensables para lograr la mejor experiencia de uso.

Figura 4. **Google Drive vs. Dropbox**



Fuente: elaboración propia.

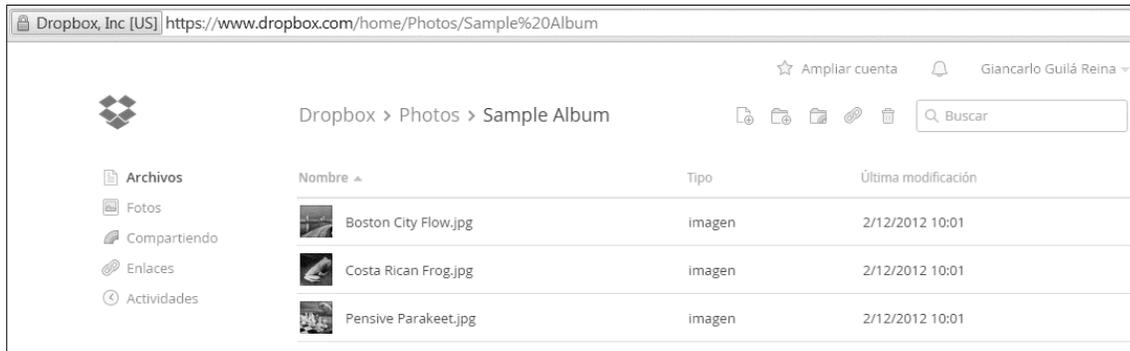
2.4.3. **Aplicación web**

Los directorios *online* de Dropbox y Google Drive también son importantes, pues desde ellos se tendrá que gestionar y administrar todo el contenido de la nube.

La simplicidad de Dropbox en este sentido no tiene comparación. La interfaz es extremadamente sencilla y clara; todo el contenido que hay guardado aparece en primer plano. En la parte superior unos iconos permiten subir archivos y crear carpeta. El menú de la izquierda sirve para crear carpetas compartidas y vínculos, así como para acceder al historial de cambios o a la papelera.

La interfaz de la aplicación web de Dropbox se muestra en la siguiente figura.

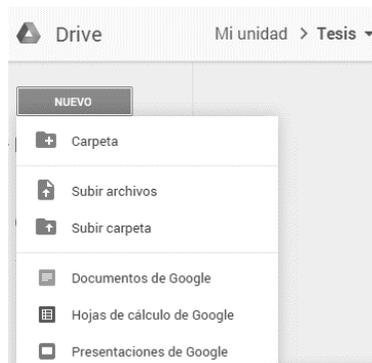
Figura 5. **Aplicación web de Dropbox**



Fuente: elaboración personal.

La interfaz de Google Drive es muy similar en todo. La distribución de elementos es calcada y también la mayoría de opciones disponibles. Sin embargo, al estar integrado entre los servicios de Google, hay otras opciones para manipular los archivos almacenados en esta plataforma, integrándolos con los otros productos ofrecidos. Google Drive también ofrece la manipulación de distintos tipos de archivos como se muestra en la siguiente imagen:

Figura 6. **Tipos de archivos soportados por Google Drive**



Fuente: elaboración propia.

Figura 7. **Aplicación web de Google Drive**

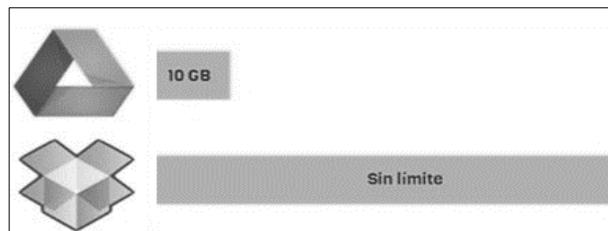


Fuente: elaboración propia

2.4.4. Límite de tamaño

El tamaño límite de los archivos que se puede subir es otro factor determinante. En la siguiente figura se ilustra la capacidad de cada uno de los servicios.

Figura 8. **Comparativa de tamaño, Google Drive vs. Dropbox**

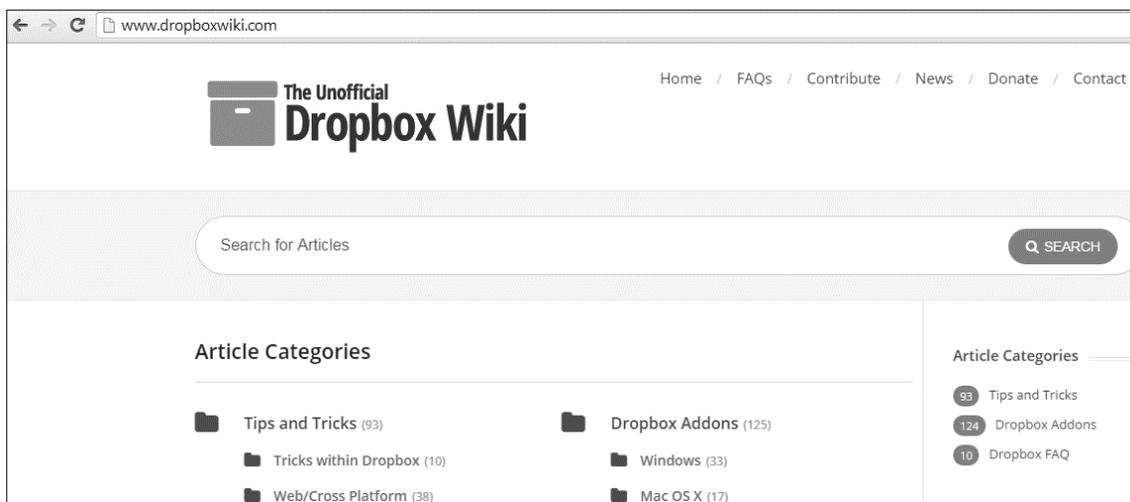


Fuente: elaboración propia.

2.4.5. Servicios adicionales

Dropbox es exclusivamente un servicio de almacenamiento, pero cuenta con *addons* y aplicaciones de terceros que aumentan las posibilidades. Entre estas aplicaciones se encuentran algunas que brindan la funcionalidad de descarga de videos desde sitios web, almacenamiento automático de capturas de pantalla, entre otros. También cuenta con un sitio informativo y de ayuda, el cual es manejado por los usuarios.

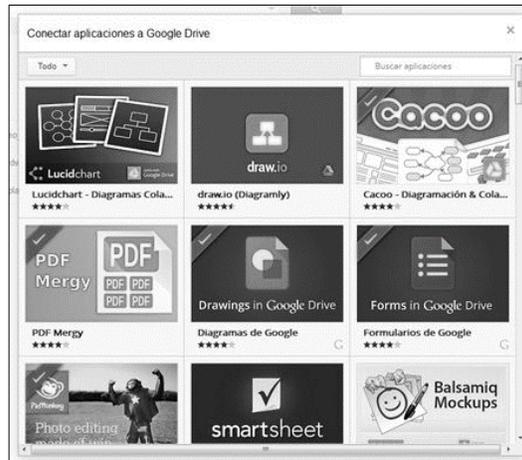
Figura 9. **Dropbox wiki**



Fuente: <http://www.dropboxwiki.com/>. Consulta: abril de 2013.

Google Drive se integra en la compleja plataforma de servicios de Google. La integración de nuevas aplicaciones *online* de terceras compañías amplía de forma infinita las opciones.

Figura 10. **Aplicaciones de Google Play para Google Drive**



Fuente: https://play.google.com/store?hl=es_419. Consulta: junio de 2013.

2.5. Aspectos legales del almacenamiento en la nube

A continuación se describen los diferentes sistemas de almacenamiento en la nube y sus aspectos legales.

2.5.1. Dropbox

El servicio de Dropbox solicita permisos para recopilar información y almacenarlos; estos permisos, son otorgados por los usuarios al momento de aceptar los términos y condiciones de uso del servicio Dropbox. Entre la información personal más común que el servicio Dropbox recopila de los usuarios están: nombre del usuario, número de teléfono, dirección de correo electrónico, direcciones postales de casa y trabajo, datos para facturación como tarjeta de crédito e importación de cuentas de correo, siempre y cuando el usuario otorgue el permiso para compartir las cuentas de correo electrónico, historial del uso del servicio, entre otros.

Dropbox hará uso de esta información para conocer mejor las necesidades e intereses de los usuarios, personalizar la experiencia y administrar el uso que el usuario hace del servicio. Entre los datos que por el momento Dropbox no almacena ni hace uso está la ubicación geográfica del dispositivo desde el cual se está accediendo al servicio, pero se aclara que podría ser solicitada en políticas futuras, por lo que queda bajo la responsabilidad del usuario permitir el uso de dicha información.

Al aceptar el uso de Dropbox también se acepta que Dropbox recopile información acerca de los archivos que se almacenan en la plataforma o que se comparten con otros usuarios; esto le es útil al servicio para conocer si el archivo compartido ya fue almacenado por otro usuario y asociar completa o parcialmente el archivo anterior para no almacenar duplicados.

Dropbox también hace uso de datos de registro para almacenar las sesiones realizadas, así como la información del dispositivo desde el que se realiza la conexión; esto permite conocer datos de navegación y preferencias que le sirven a los desarrolladores a mejorar la usabilidad de la plataforma para los usuarios.

Dropbox, hace uso de *Cookies* para almacenar preferencias del usuario y permitir ciertos aspectos del servicio; por lo que la usabilidad del servicio se ve afectada si no se acepta el uso de estos archivos en el explorador. Los encargados del servicio Dropbox hacen uso de servicios de terceros como Google analytics, para observar y analizar información relevante sobre el uso del servicio, ayudando a mejorar la administración técnica del servicio, su facilidad de uso y funcionalidad, así como a comprobar que los usuarios tengan la autorización necesaria para que el servicio procese sus solicitudes.

También se vale de proveedores de servicios, socios comerciales y otros que le permiten mejorar la funcionalidad de la plataforma; utiliza Amazon S3 para el almacenamiento de información parcial de la cuenta, tomando en cuenta que el proveedor debe cumplir con políticas de privacidad y obligaciones similares a Dropbox para hacer uso de ellos.

El usuario es responsable de la información que se muestra en su perfil, dadas las configuraciones realizadas en las preferencias de su cuenta. Cuando un usuario permite a un software de terceros acceder a su cuenta, debe saber que Dropbox no se responsabiliza del uso que esta aplicación haga con su información, por lo que el usuario debe conocer las políticas de la aplicación a la cual le concede permisos además de confiar en el proveedor de dicha aplicación. Dropbox también tiene políticas en cuanto a la divulgación de la información, en dado caso esta sea solicitada por algún servicio policial o se cumpla con la ejecución de alguna ley que Dropbox considere de buena fe que la divulgación sea razonablemente necesaria.

Si en algún momento los ejecutivos de Dropbox deciden participar en una fusión, adquisición o venta total o parcial de sus activos, la información de los usuarios podrá ser transferida como parte de esa transacción, con la condición de que todo usuario activo en el servicio será notificado previamente para que sea él quien decida continuar con el servicio y sus nuevas políticas (si estas cambiaran) o suspender el servicio.

Al momento de que un usuario decida suspender o eliminar su cuenta del servicio Dropbox, se eliminarán todos los registros referentes a dicha cuenta; pero Dropbox podrá conservar y usar la información del usuario según sea necesario para cumplir con las obligaciones legales y resolución de conflictos, a manera de poder cumplir con los convenios de la empresa.

Al hacer uso del servicio de forma empresarial, se debe tomar en cuenta que el administrador de dicha cuenta será quien tendrá el acceso a modificar todas las políticas anteriormente mencionadas, tomando como "usuario" y su información a la empresa participante de la cuenta como tal.

Dropbox garantiza la seguridad de la información a través del uso de certificados de seguridad conocidos como la Capa de Socket Seguros (SSL), haciendo énfasis en que la seguridad de medios de internet no se puede garantizar al 100 %; por lo que se aclara que no se puede garantizar la plena seguridad de la información.

2.5.2. Google Drive

Entre las políticas de Google Drive, se hace especial énfasis en la forma de acceso a los servicios de Google, a manera de evitar las intrusiones a su sistema, por lo que se estipula en que uno de los métodos para acceder a dicho servicio debe ser por medio de la interfaz gráfica y de las instrucciones dadas por los administradores de Google Drive; esto incluye el uso de API's para conexión del servicio.

Una variante de las políticas de Google en relación con las estipuladas para Dropbox, es que Google no otorga derecho de propiedad intelectual a alguno de los servicios proporcionados por esta empresa y tampoco al contenido que acceden.

Como en la mayoría de servicios, la información que se recopila de los usuarios es utilizada para mejorar la usabilidad del software desarrollado, así como las funcionalidades que este proporciona. Google hace uso de la información pública y básica, siempre y cuando el usuario permita hacer uso de

ella, se toma como referencia los perfiles públicos y la información personal que el usuario publique; entre la información más relevante que un servicio de Google puede recabar está: nombre del usuario, dirección de correo electrónico, número de teléfono, tarjeta de crédito y fotografía, en algunos casos.

Google también puede utilizar información propia del uso de sus servicios, realizando análisis sobre información de conexión, transacciones y modo de uso. La información que se recopila del dispositivo de conexión puede incluir, modelo de hardware, versión del sistema operativo, identificadores únicos de dispositivo e incluso el número de teléfono del usuario. Al mismo tiempo, al hacer uso de alguno de los servicios de Google se está permitiendo almacenar información en los registros de los servidores de Google, en los cuales, se pueden incluir detalles sobre el modo de uso del servicio, información de registro de telefonía, fecha y hora de las llamadas, duración de las mismas, etc.

Al contrario del servicio de almacenamiento Dropbox, Google Drive, es capaz y hace uso del servicio de posicionamiento global (GPS) para conocer la ubicación del dispositivo; al mismo tiempo puede utilizar diversas tecnologías para determinar la ubicación de puntos de acceso a *Wi-Fi* y torres de telefonía cercana, utilizando datos del sensor del dispositivo.

El almacenamiento local permite hacer uso de técnicas de almacenamiento caché y de parámetros específicos que pueden mejorar el rendimiento de los servicios utilizados; las *cookies* o identificadores anónimos son utilizados para recopilar y almacenar información cuando un usuario visita un servicio de Google, así como cuando se hacen uso de servicios de terceros que utilizan también los servicios de Google.

Toda la información que se recopila por Google, a través de los diferentes medios es utilizada de forma conjunta en todos los servicios que Google proporciona; esto permite que la experiencia de los usuarios sea percibida como usuario de un único servicio, ya que Google está en la capacidad de hacer uso de la información proporcionada como correo electrónico para asociar varias cuentas de diferentes servicios; la información que se almacena no incluye información de carácter sensible, como por ejemplo, preferencias sexuales, raza, religión o salud.

Al momento de hacer uso de información combinada de diferentes servicios, Google solicitará el consentimiento del usuario para poder hacer uso de dicha información; de lo contrario, accederá únicamente a la información marcada como pública.

Google, como parte de sus políticas de transparencia, permite configurar todas y cada una de las opciones que se mencionan en el acuerdo de privacidad y configuraciones propias; por lo que se puede deshabilitar la publicidad y ciertas preferencias de determinado servicio; esto permite que el usuario tenga la libertad de elegir la configuración que mejor se apegue a sus necesidades; al mismo tiempo, cumple con el principio de Google de permitir personificar todo aquello que pertenece al usuario.

La información que se provee a los servicios de Google, no se comparte con ninguna entidad externa a menos que el usuario lo permita, por lo que siempre se solicitará su consentimiento para cualquier acción que se vaya a realizar con su información; también se hará uso de dicha información cuando asuntos legales requieran que Google comparta información con empresas, organizaciones e individuos, siempre y cuando Google de buena fe crea que el acceso a dicha información es necesario para aclarar el caso.

La seguridad de la información se garantiza, no de forma total, en los servicios de Google, por medio de la encriptación de la información, usando SSL, estándar de autenticación OAuth y una función de navegación segura por medio de su navegador particular. También se hace énfasis en garantizar la seguridad física de los medios de almacenamiento, tanto que no se permite el acceso no autorizado a dicha localidad. La información que utilizan empleados, contratistas y agentes de Google están sujetos bajo estos términos de confidencialidad, por lo que de no cumplirlos serán sancionados o despedidos.

3. DOCUMENTACIÓN BASE PARA EL DESARROLLO DE LA APLICACIÓN

Antes de iniciar con el desarrollo de aplicaciones Android es necesario conocer cómo está construida una aplicación y en especial el ciclo de vida de sus componentes.

3.1. Componentes de una aplicación Android

Existen distintos componentes que interactúan en el funcionamiento de una aplicación móvil, los cuales se describen a continuación.

3.1.1. Actividades

Forman la capa de presentación de la aplicación. En una aplicación de Android cada pantalla se ve representada por una clase que extiende o hereda las propiedades de la clase base *Activity*. Las actividades utilizan las vistas para formar interfaces de usuario gráficas que permiten mostrar información y responder a las acciones por medio de la captura de eventos. Una actividad en Android es equivalente a un formulario en las aplicaciones de escritorio.

3.1.2. Intents

Estos objetos sirven como intermediarios y transportadores de datos entre actividades y servicios; son lanzados como mensajes asíncronos y pueden ser enviados hacia múltiples procesos o hacia un proceso o actividad específica; el sistema determinará quién realizará la acción que se demanda.

3.1.3. Servicios

Estos son procesos que se ejecutan en segundo plano en una aplicación Android. Son capaces de actualizar información por medio de la ejecución de procesos, así como de lanzar notificaciones a las actividades visibles en un momento dado. Son utilizados para ejecutar procesos cuando las actividades de la aplicación no se encuentren visibles.

3.1.4. Tarea asíncrona

Estas son tareas que se ejecutan en segundo plano, con la diferencia de que pueden manejar actualizaciones de estados directas hacia la interfaz de usuario que las ejecutó; esto permite conocer el progreso de dicha tarea, ya que son capaces de actualizar la vista de la aplicación al finalizar el proceso.

3.1.5. *Content providers*

Los proveedores de contenido de Android permiten recibir y transmitir información entre aplicaciones y base de datos. Esto significa que Android permite configurar acceso a cualquier tipo de datos, los cuales pueden ir desde datos multimedia hasta detalles de contacto.

3.1.6. *Broadcast receivers*

Este tipo de componentes reaccionan a eventos generales del tipo *broadcast* los cuales pueden ser producidos por aplicaciones o por el propio sistema. Ejemplos de eventos que pueden ser atendidos por este tipo de componentes son: los de batería baja, llamadas, SMS, etc.

3.1.7. *Manifest*

Todo proyecto de Android consta de un archivo *Manifest* llamado “AndroidManifest.xml”; este se crea con el proyecto y se encuentra en la raíz de la jerarquía de archivos del mismo.

Este fichero permite definir la estructura y metadatos de la aplicación, sus componentes y requerimientos. Incluye nodos para cada componente que forman la aplicación; en este archivo se deben declarar todas las actividades, servicios, proveedores de contenido, receptores de transmisión, intentos y los permisos que la aplicación solicitará al usuario para poder hacer uso de ella. Un nodo superior permite configurar las propiedades de seguridad de la aplicación, unidades de prueba y definir el hardware y los requerimientos de la plataforma a utilizar.

Está compuesto por una etiqueta raíz “<manifest>”, la cual posee el atributo *package*, el cual utiliza para identificar el paquete principal del proyecto. Esta etiqueta usualmente incluye el atributo “xmlns: android” que suministra varios atributos del sistema a utilizar dentro del archivo.

Hace uso del atributo “versionCode” para definir la versión actual que va a utilizarse en la aplicación como un atributo tipo entero (*integer*), el cual sirve internamente para comparar la versión de Android sobre la cual se va a ejecutar la aplicación.

3.2. Conceptos involucrados en la comunicación con sistemas externos

Existen muchos factores que se ven involucrados en el desarrollo de aplicaciones móviles y que interactúan entre ellas para englobar una única funcionalidad. Entre estos factores se encuentran: API's de desarrollo, librerías, componentes externos, etc.

3.2.1. ¿Qué es un sistema externo?

Es todo aquel sistema con el que interactúa la aplicación, pero que únicamente la utiliza como recurso para mejorar su funcionalidad.

La administración de este tipo de sistemas queda fuera de las manos del programador, quien debe restringir su desarrollo y adecuarse a lo que el sistema externo le permita realizar. La utilización de un sistema externo procura expandir las características y alcances de la aplicación, permitiendo mejorar las propiedades de usabilidad del sistema en cuestión.

Una sesión de un sistema externo incluye información de conexión y autenticación para una sesión específica de un sistema externo. La configuración de varias sesiones de un sistema externo permite al diseñador de soluciones establecer distintos parámetros de seguridad en conexiones de datos externos, para admitir varias formas de conexión al mismo. Sin embargo, en muchas aplicaciones todo lo que se necesita es una única sesión de sistema.

3.3. API de desarrollo

API, acrónimo de *Application Programming Interface*, es la encargada de proveer los recursos y definiciones necesarias para establecer comunicación entre aplicaciones, permitiendo que estos interactúen entre sí, como si fueran una única aplicación. A los sistemas externos que proveen APIs para su integración muchas veces se les conoce como “servicios”. Un API de un sistema externo provee recursos de comunicación con diferentes aplicaciones; en el caso de ser un sistema externo alojado en la red, provee de comunicación con el cliente por medio de mensajes HTTP, los cuales, viajan a través de la red, estableciendo conexiones entre cliente y servidor llamadas “sesiones”.

Cuando un API provee comunicación con un componente de software, este se hace por medio de las referencias necesarias que le permiten interactuar entre aplicaciones locales.

Una aplicación local es toda aquella que se aloja dentro del mismo dispositivo sobre el que se ejecuta la aplicación principal que está haciendo uso de ella para mejorar su funcionalidad.

3.3.1. Kit de desarrollo de software o SDK (Software development kit)

Generalmente se encuentra formado por un conjunto de herramientas de desarrollo de software que le permiten al programador crear aplicaciones para un sistema concreto, pudiendo ser estos: paquetes de software, *frameworks*, plataformas de hardware, computadoras, videoconsolas, sistemas operativos, etc.

3.3.2. Google Drive SDK

Es el recurso que Google pone a disposición de los desarrolladores, brindándoles un grupo de APIs junto con bibliotecas cliente, ejemplos en lenguajes de programación específicos y documentación necesaria para ayudar a desarrollar aplicaciones capaces de interactuar con Google Drive.

La principal funcionalidad que permite realizar el Google Drive SDK incluye las operaciones de subir y descargar archivos desde Google Drive. Sin embargo, el Google Drive SDK provee mucho más que solo almacenamiento.

3.3.2.1. Principales usos

El Google Drive SDK permite con el consentimiento del usuario, realizar las siguientes operaciones sobre los archivos almacenados en el Google Drive; estas operaciones son:

Tabla V. Operaciones permitidas por el SDK de Google Drive

Método	Descripción
<i>Get</i>	Obtiene los metadatos de un archivo.
<i>Insert</i>	Inserta un nuevo archivo en Google Drive, en el directorio especificado.
<i>Patch</i>	Actualiza los metadatos de un archivo.
<i>Update</i>	Actualiza los metadatos de un archivo y/o su contenido.
<i>Copy</i>	Crea una copia del archivo especificado.
<i>Delete</i>	Elimina de forma permanente un archivo especificado por su identificador.
<i>List</i>	Lista los archivos de un usuario.
<i>Touch</i>	Actualiza el tiempo de actualización del archivo hacia el tiempo actual en el servidor.
<i>Trash</i>	Mueve un archivo al basurero.

Fuente: <https://developers.google.com/drive/>. Consulta: octubre de 2013.

3.4. Métodos de autorización

Actualmente existen diferentes aplicaciones que utilizan información alojada en internet sobre un usuario en específico, para un servicio específico; muchas veces esta información puede ser considerada como sensible, es decir, de mucha importancia y privacidad para la persona que posee la información, por lo que cada aplicación cliente ya sea web, móvil o instalada, debe solicitar autorización del usuario, quien debe permitir a la aplicación hacer uso de su información.

3.4.1. OAuth (*open authorization*)

Protocolo abierto para autorización segura de un API de modo estándar; permite interactuar con datos protegidos y utilizarlos en aplicaciones para clientes de escritorio, móviles y web.

La ventaja principal de este protocolo es que permite compartir información entre un proveedor de servicio y un consumidor, sin ser necesario hacerlo con toda la identidad del propietario de la información.

OAuth se conoce como la forma más segura de compartir información con el consentimiento de los usuarios dueños de dicha información.

Algo interesante de esta forma de autenticación es que las credenciales del usuario no necesitan ser transmitidas por web, sino que OAuth se encarga de dar acceso a la aplicación para hacer uso de los datos de usuario, mientras se protegen sus datos de cuenta como contraseñas.

3.4.1.1. Antecedentes

OAuth comenzó su desarrollo por Blaine Cook y Chris Messina, en noviembre de 2006; mientras Blaine Cook desarrollaba OpenID para twitter, el cual tenía como objetivo delegar el método de autenticación a OpenID al utilizar el API de twitter. Al mismo tiempo surgió la necesidad por parte de los desarrolladores del conocido CMS en Java de código abierto Magnolia, de utilizar una solución en OpenID para la autorización de *widgets* del *dashboard* para acceder a su servicio; esta investigación estaba siendo realizada por Larry Halff.

Blaine Cook y Chris Messina se reunieron con David Recordon, Larry Halff y otros, en OpenID CitizenSpace para discutir las soluciones existentes. En donde, después de revisar la funcionalidad de OpenID y otras prácticas de autenticación en la industria, llegaron a la conclusión de que no existía una norma abierta para la delegación de acceso a un API.

En abril de 2007, Google creó un grupo con un propósito similar, pero posteriormente DeWitt Clinton de Google se enteró del proyecto, por lo que de forma interesada expresó su interés en apoyar el esfuerzo. En julio de 2007 el equipo elaboró una especificación inicial; asimismo, fue abierto un grupo para toda aquella persona interesada en contribuir.

OAuth identifica dos partes en el juego de la autenticación; la primera se trata del proveedor de servicios, el cual es el sitio al que se le permite el acceso a los recursos privados por parte del usuario. La segunda se trata del consumidor, quien es la aplicación cliente que se encarga de consumir literalmente los datos a los que el usuario le ha otorgado sus permisos.

OAuth, a diferencia de OpenID, trata de dar acceso únicamente a la información que el usuario permita sin compartir toda su identidad o sus partes secretas; mientras que en OpenID se trata todo acerca de la misma identidad y utiliza los mismos medios de autenticación para iniciar sesión en muchos sitios. Otra diferencia significativa es que OAuth tiene que ver con los accesos que un usuario concede a un servicio, mientras que en OpenID se trata de conocer si el usuario es realmente quien dice ser.

OAuth no es una idea nueva, más bien es la utilización de las mejores prácticas en autenticación que pretende crear un mejor medio de acceso a la información; se basa en las convenciones ya establecidas en los protocolos como: Google AuthSub, AOL's Open Auth, Yahoo BBAuth, FlickrAuth y FacebookAuth.

El objetivo era diseñar el mejor protocolo de autenticación, pero también que fuera compatible con los métodos de autenticación existentes, así como reutilizar los estándares web, siempre que sea posible para garantizar la compatibilidad con los métodos ya existentes, que permitieran su fácil migración entre protocolos; el protocolo resultó ser fácil de leer y ejecutar sin necesidad de un gran cantidad de esfuerzo, por lo que se pone a la disposición de los desarrolladores un número de bibliotecas que pretenden reducir la cantidad de trabajo necesario para hacer lo correcto y facilitar la migración de protocolos existentes.

3.5. Documentación base para el desarrollo de la aplicación móvil

Es necesario conocer los procesos de instalación y configuración de las herramientas para la creación de la aplicación móvil, así como el desarrollo y pruebas de la aplicación móvil.

El lenguaje utilizado para desarrollar aplicaciones en Android es Java, un lenguaje orientado a objetos. Es necesario tener conocimientos de la arquitectura de Android y conocimientos de XML para la parte visual y para los textos que son desplegados en los botones, etiquetas y mensajes.

3.5.1. Requerimientos para el desarrollo

En esta sección se especifican los requerimientos para el desarrollo de la solución en el sistema operativo Android. Entre estos requerimientos se encuentran: el kit de desarrollo para dispositivos android, las herramientas de desarrollo y el dispositivo virtual.

3.5.1.1. Android SDK

El Android SDK (Android Software Development Kit) provee un API con las herramientas para compilar y probar aplicaciones para Android.

Una de las grandes ventajas que el Android SDK provee al desarrollador son los dispositivos virtuales Android (AVD), ya que permite utilizarlos obteniendo los mismos resultados que se tendrían en un dispositivo real; el Android AVD puede emular un dispositivo real, definiendo las especificaciones de hardware y software del dispositivo virtual.

El SDK cuenta con la herramienta Android Debug Bridge (ADB) que permite realizar una depuración de la aplicación en un dispositivo real o virtual; esta herramienta funciona como un cliente-servidor que ejecuta en segundo plano un proceso en el emulador o el dispositivo.

3.5.1.2. Android Development Tools (ADT)

Es una extensión del IDE Eclipse que provee todas las herramientas necesarias para el desarrollo, depuración y ejecución de aplicaciones para Android, provee editores especializados de XML, que permiten al desarrollador una vista previa del XML que representa una interfaz de usuario.

3.5.1.3. Instalación y configuración

Para utilizar la plataforma de desarrollo para dispositivos Android se debe de instalar una serie de librerías y componentes, las cuales se detallan en la siguiente sección.

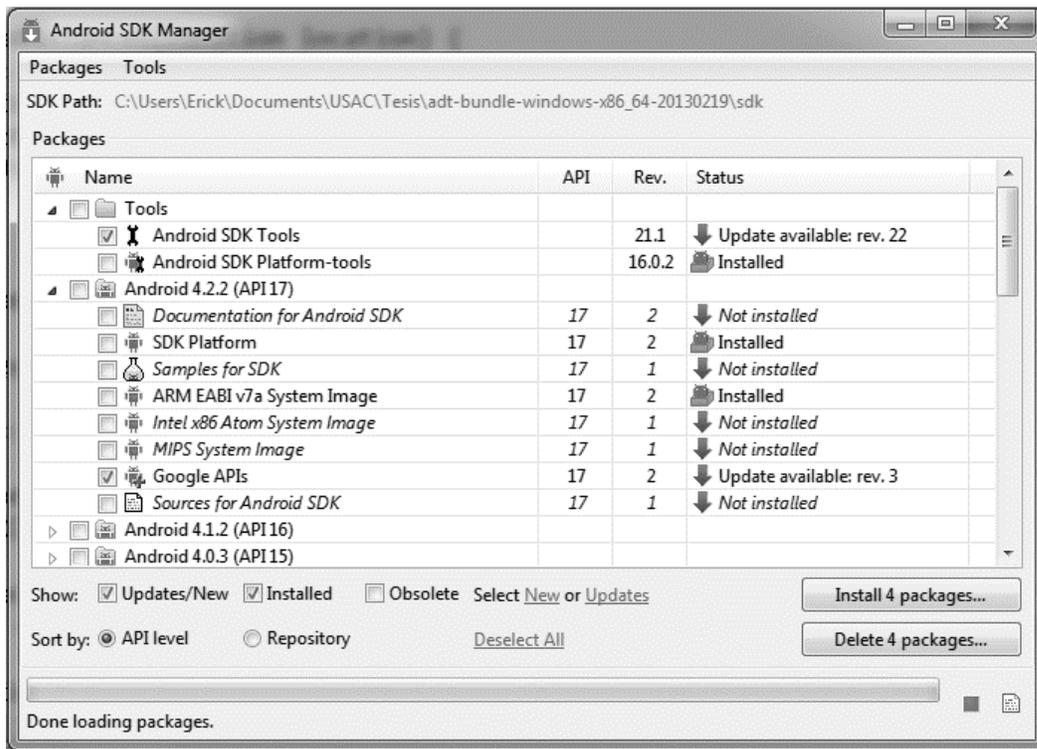
3.5.1.3.1. Descarga e Instalación del Android SDK

Desde la página <http://developer.android.com/sdk/index.html> se puede descargar Eclipse y el ADT *plugin* con el Android SDK *tools* y todas las herramientas necesarias para el desarrollo. Al finalizar la descarga hay que descomprimir el archivo descargado, en la ruta donde sea deseado instalar el SDK. En dicha ruta de instalación existen dos directorios: uno contiene el SDK de Android y en el otro se encuentra Eclipse.

3.5.1.3.2. Android SDK Manager

El Android SDK manager es una herramienta aparte que permite al usuario obtener componentes para el desarrollo. Para entrar al android SDK manager desde Eclipse se puede ir al menú Window>Android SDK Manager. También se encuentra el ejecutable en el directorio raíz del Android SDK.

Figura 11. Android SDK manager

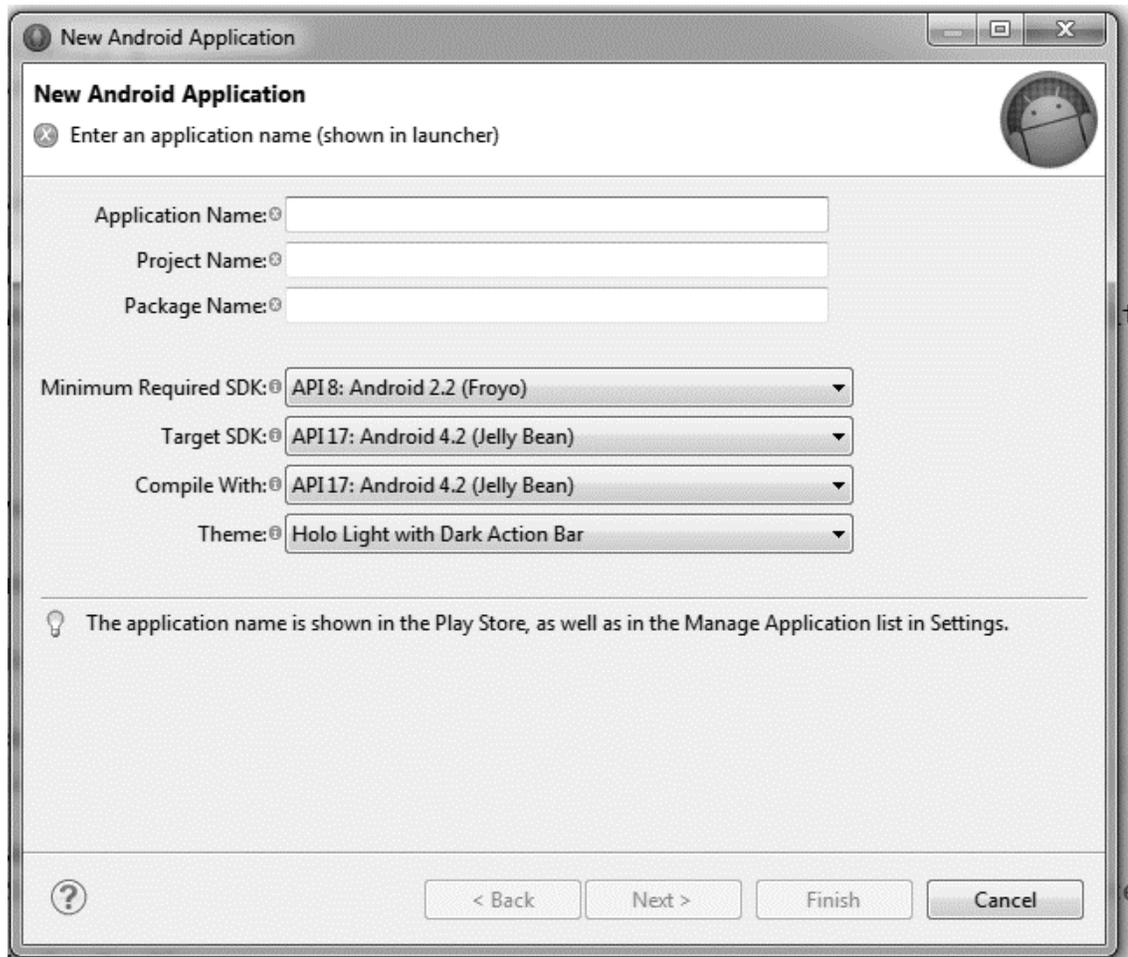


Fuente: elaboración propia.

3.5.1.3.3. Crear un nuevo proyecto en Eclipse

Para crear un nuevo proyecto en Eclipse se entra al menú File > New > Android Project; se debe ingresar el nombre de la aplicación y Eclipse autocompletará el nombre del proyecto y el del paquete; como recomendación es preferible que el nombre de la aplicación inicie con una letra mayúscula.

Figura 12. Creación de aplicación Android



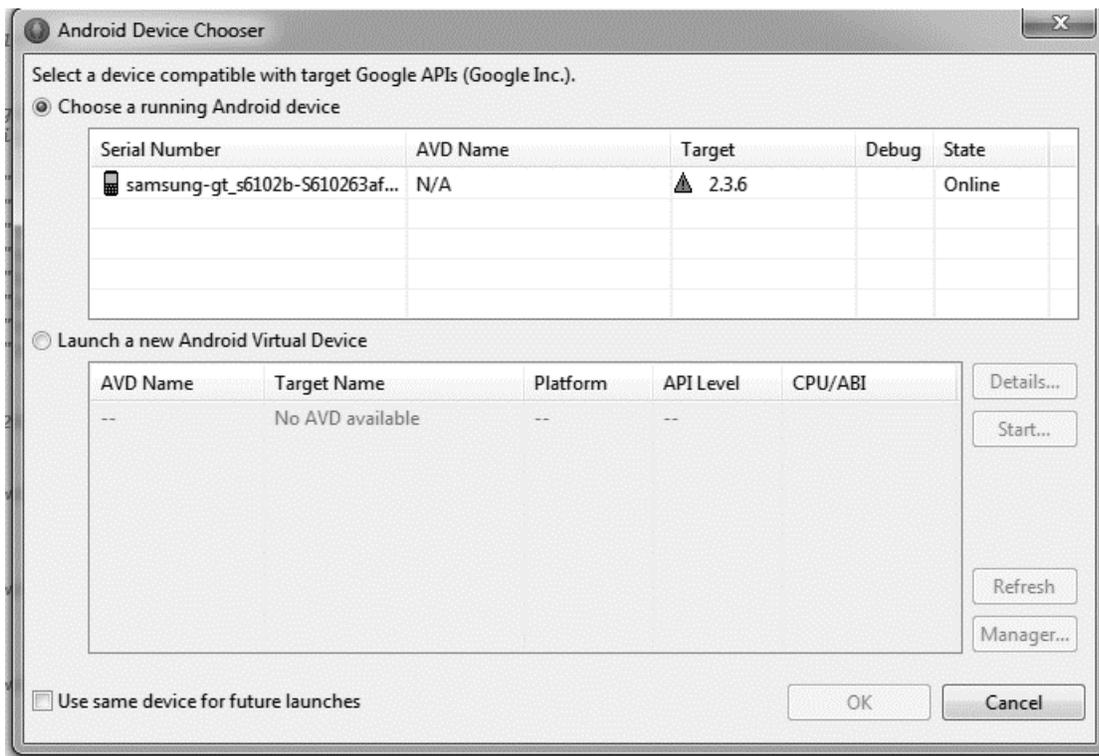
Fuente: elaboración propia.

3.5.1.4. Ambiente de realización de pruebas

Para realizar pruebas de una aplicación para Android en un dispositivo real, es necesario tener instalado el controlador del dispositivo en la computadora; también es necesario que el dispositivo permita la instalación de aplicaciones que sean de la tienda *online*.

Al ejecutar la aplicación se debe tener conectado el dispositivo vía USB; la siguiente imagen muestra la ventana que despliega Eclipse, donde se puede seleccionar si se quiere ejecutar la aplicación en el dispositivo o en un AVD.

Figura 13. **Android device chooser**



Fuente: elaboración propia.

4. MANUAL DE USO DE LA APLICACIÓN

Antes de iniciar con el desarrollo de aplicaciones en Android es necesario conocer cómo está construida una aplicación y en especial el ciclo de vida de sus componentes.

4.1. Componentes de una aplicación móvil en Android

Existe una serie de elementos clave que resultan imprescindibles para desarrollar aplicaciones en Android. De todos los elementos existentes, hay ciertos elementos que son los más importantes que deben destacar en cualquier aplicación móvil.

4.1.1. Vista

Son elementos que componen la interfaz de usuario de una aplicación. Entre estos elementos se encuentran, por ejemplo: un botón, una entrada de texto, etc. Todas las vistas van a ser objetos descendientes de la clase *View*, y por tanto, pueden ser definidos utilizando código Java. Sin embargo, lo habitual va a ser definir las vistas utilizando un fichero XML y dejar que el sistema cree los objetos a partir de este fichero.

4.1.2. *Layout*

Es un conjunto de vistas agrupadas de una determinada forma. Se debe de disponer diferentes tipos de *layouts* para organizar las vistas de forma línea, en cuadrícula o indicando la posición absoluta de cada vista.

Los *layouts* también son objetos descendientes de la clase View. Los *layouts*, al igual que las vistas, también pueden ser definidos en código, aunque la forma habitual de definirlos es utilizando código XML.

4.1.3. Actividad

Una aplicación en Android va a estar formada por un conjunto de elementos básicos de visualización, coloquialmente conocidos como pantallas de la aplicación. A cada uno de estos elementos o pantallas, se le conoce como actividad. Su función principal es la creación de la interfaz de usuario. En una aplicación se suelen necesitar varias actividades para crear la interfaz de usuario. Las diferentes actividades creadas serán independientes entre sí, aunque todas trabajarán para un objetivo común. Toda actividad ha de pertenecer a una clase descendiente de Activity.

4.1.4. Servicio

Es un proceso que se ejecuta en segundo plano, sin la necesidad de una interacción con el usuario. Es algo parecido a un demonio en Unix o a un servicio de Windows. En Android se dispone de dos tipos de servicios: servicios locales, que son ejecutados en el mismo proceso y servicios remotos que son ejecutados en procesos separados.

4.1.5. Intención

Representa la voluntad de realizar alguna acción; como realizar una llamada de teléfono o visualizar una página web.

Se utiliza cada vez que se quiera:

- Lanzar una actividad
- Lanzar un servicio
- Enviar un anuncio de tipo *broadcast*
- Establecer comunicación con otro servicio

Los componentes lanzados pueden ser internos o externos a la aplicación. También se pueden utilizar intenciones para el intercambio de información entre los componentes.

4.1.6. Receptor de anuncios

Recibe y reacciona ante anuncios de tipo *broadcast*. Los anuncios *broadcast* pueden ser originados por el sistema o por las aplicaciones. Algunos tipos de anuncios originados por el sistema son: batería baja, llamada entrante, entre otros. Las aplicaciones también pueden crear y lanzar nuevos tipos de anuncios *broadcast*. Los receptores de anuncios no disponen de interfaz de usuario, aunque pueden iniciar una actividad si lo estiman oportuno.

4.1.7. Proveedores de contenido

En muchas ocasiones las aplicaciones instaladas en una terminal Android necesitan compartir información. Android define un mecanismo estándar para que las aplicaciones puedan compartir datos sin necesidad de comprometer la seguridad del sistema de ficheros. Con este mecanismo es posible acceder a datos de otras aplicaciones como una lista de contactos, o proporcionar datos a otras aplicaciones.

4.2. Instalación del entorno de desarrollo

Google ha preparado el paquete de software Android SDK, que incorpora todas las herramientas necesarias para el desarrollo de aplicaciones en Android. En él se incluye conversor de código, *debugger*, librerías, emulador, documentación, ejemplos de código, etc.

No obstante, el programador habitual prefiere utilizar un IDE, o entorno que integre un editor de texto con todas las herramientas de desarrollo. Aunque no son únicas, las alternativas más recomendables son Eclipse e IntelliJ Idea.

Para el desarrollo de la aplicación GCloud, se utilizará el entorno de desarrollo integrado Eclipse, cuya instalación se detalla paso a paso.

4.2.1. Instalación de la máquina virtual Java

Este software va a permitir ejecutar código Java en el equipo. A la máquina virtual Java también se le conoce como entorno de ejecución Java, Java Runtime Environment (JRE) o Java Virtual Machine (JVM).

Para instalar la máquina virtual Java se debe acceder a <http://java.com/es/download/>; descargar e instalar el fichero correspondiente al sistema operativo que se esté utilizando.

4.2.2. Instalación basada en Eclipse con ADT Bundle

Google ha sacado recientemente ADT Bundle, un paquete que incluye los elementos para instalar de forma sencilla el IDE basado en Eclipse. El ADT Bundle se debe descargar desde <http://developer.android.com/sdk>.

4.2.3. Instalación basada en Eclipse de cada componente

Una instalación con IDE Eclipse requiere la instalación de los siguientes elementos:

- Java Runtime Environment 5.0 o superior
- Eclipse IDE
- Android SDK
- Eclipse Plug-in para Android

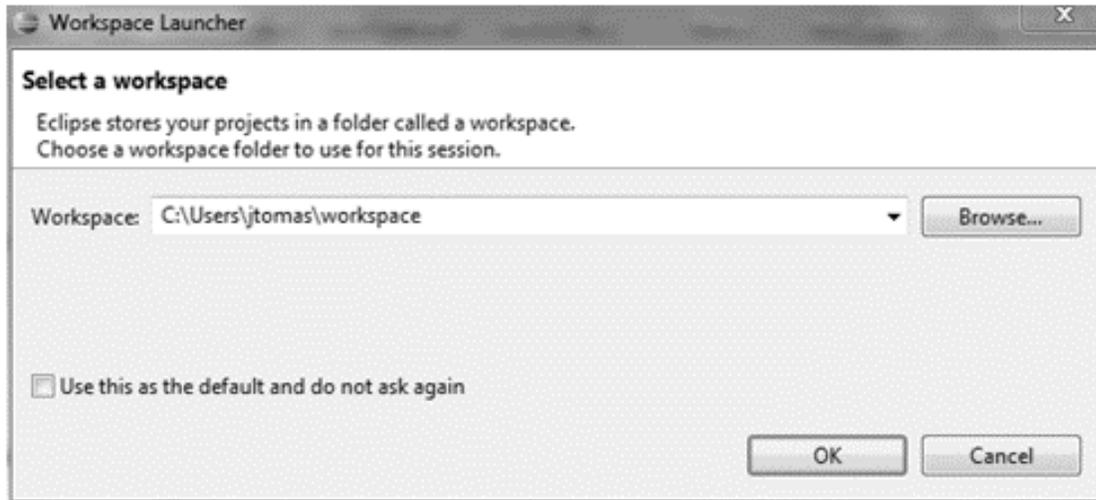
4.2.3.1. Instalación de Eclipse

Eclipse resulta el entorno de desarrollo más recomendable para Android; es libre y además es soportado por Google, ya que ha sido utilizado por los desarrolladores de Google para crear Android. Se puede utilizar cualquier versión de Eclipse a partir de la versión 3.3.1 del software.

Para instalar Eclipse hay que seguir los siguientes pasos:

- Acceder a la página <http://www.eclipse.org/downloads/> y descargar la última versión de “Eclipse IDE for Java Developers”;
- Este software no requiere una instalación específica, simplemente deben descomprimirse los ficheros en la carpeta que se prefiera;
- Al arrancar, Eclipse lanzará la pregunta acerca de qué carpeta se quiere utilizar como espacio de trabajo. En esta carpeta serán almacenados los proyectos que se creen en Eclipse.

Figura 14. **Espacio de trabajo en Eclipse**



Fuente: elaboración propia.

4.2.3.2. **Instalación de Android SDK de Google**

El siguiente paso consiste en instalar el Android SDK de Google. Los pasos se listan a continuación.

- Acceder a la página <http://developer.android.com/sdk> y descargar el fichero correspondiente al sistema operativo que se utilice.
- Este software no requiere una instalación específica; simplemente deben descomprimirse los ficheros en la carpeta que se prefiera.
- Ejecutar el programa SDK Manager.
- Seleccionar los paquetes a instalar. Aparecerá una ventana donde se pueden seleccionar los paquetes a instalar.

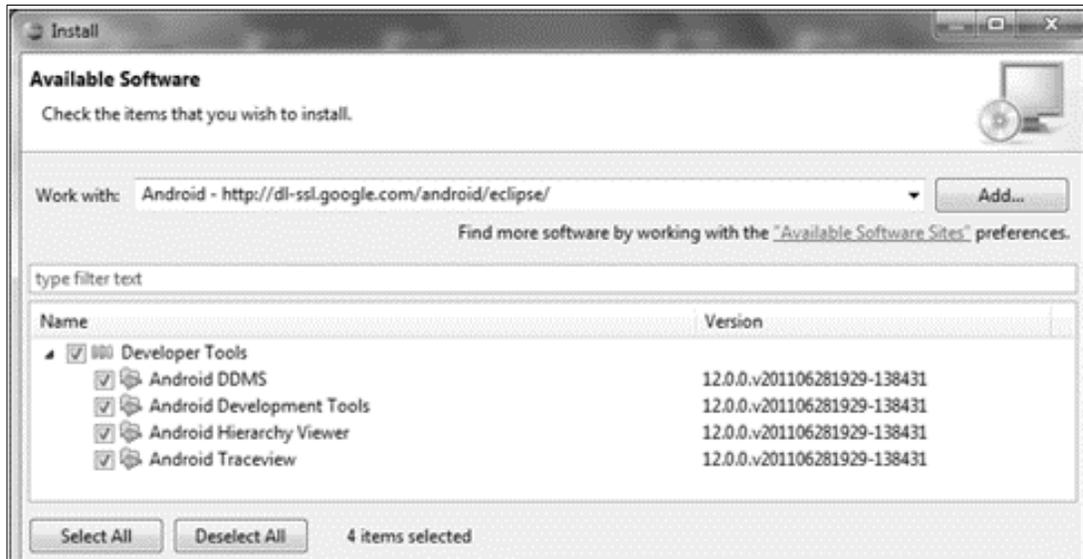
4.2.3.3. Instalación del *plug-in* Android para Eclipse (ADT)

El último paso consiste en instalar el *plug-in* Android para Eclipse, también conocido como ADT. Este software desarrollado por Google, instala una serie de complementos en Eclipse, de forma que el entorno de desarrollo se adapte al desarrollo de aplicaciones para Android. Se crearán nuevos botones, tipos de aplicación y vistas para integrar Eclipse con el Android SDK, que se acaba de instalar.

Para instalar el *plug-in* Android se deben de seguir los siguientes pasos:

- Arrancar “Eclipse” y seleccionar “Ayuda”, seguido de “Instalar nuevo software”.
- En el cuadro de diálogo “*Available Software*” que aparece se debe de hacer clic en “Agregar”.
- En el cuadro de diálogo “Agregar sitio” se debe de introducir un nombre para el sitio remoto. En el campo “Ubicación” se debe introducir la siguiente URL: <http://dl-ssl.google.com/android/eclipse/>.
- Ahora en el cuadro de “Software disponible” debe aparecer como se muestra en la siguiente imagen.

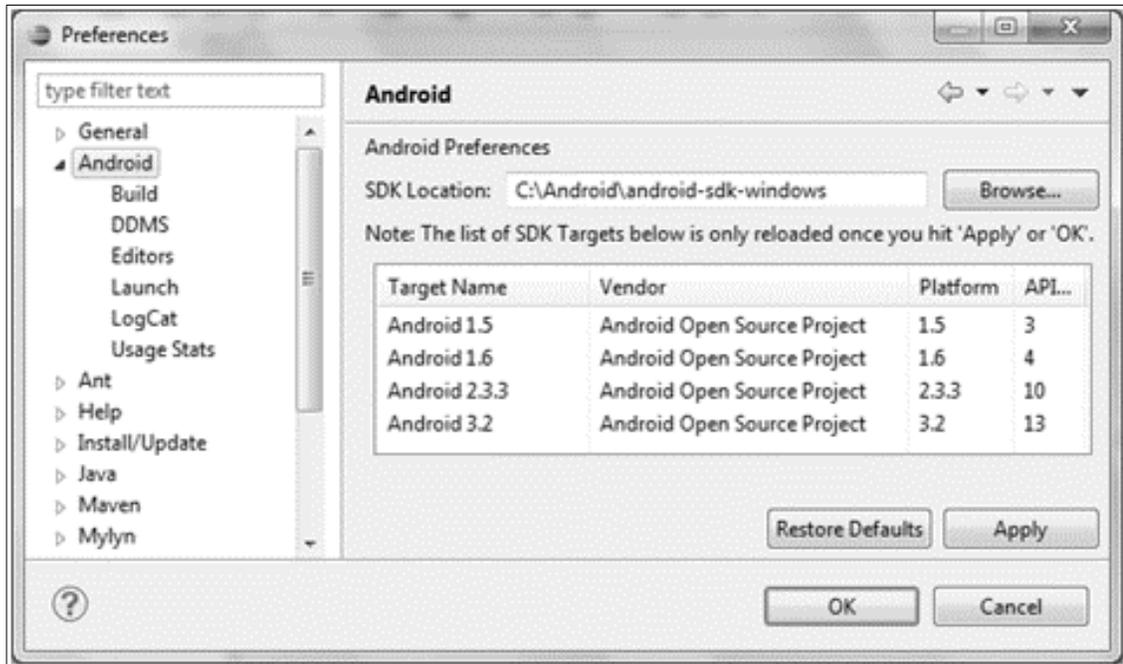
Figura 15. Instalación de *plug-in* Android



Fuente: elaboración propia.

- Se deben seleccionar los paquetes a instalar y luego pulsar “Siguiente”. Aparecerán listadas las características de Android DDMS y Android Development Tools.
- Se debe pulsar “Siguiente” para leer y aceptar la licencia e instalar cualquier dependencia.
- Debe reiniciarse Eclipse.
- Se recomienda configurar Eclipse para que se sepa dónde se ha instalado Android SDK. Para ello se debe entrar en las preferencias y seleccionar Android del panel de la izquierda. Ahora es necesario pulsar “Seleccionar” para ingresar la ubicación del SDK, donde se debe buscar la ruta donde se ha descomprimido Android SDK.

Figura 16. Preferencias de Eclipse



Fuente: elaboración propia.

4.2.3.4. Creación de un dispositivo virtual Android (AVD)

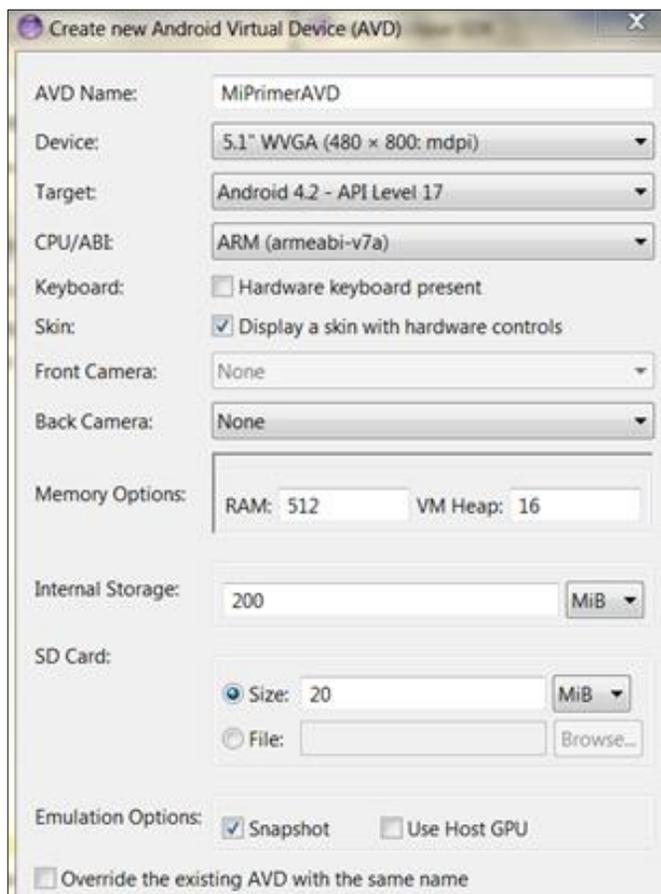
Un dispositivo virtual Android (AVD) permite emular en el ordenador cualquier tipo de dispositivo con Android. De esta forma se podrá probar la aplicación en gran variedad de teléfonos y tabletas con cualquier versión, pantalla o tipo de entrada.

Los pasos para crear un dispositivo virtual Android (AVD) son los siguientes:

- Abrir eclipse y pulsar en el botón Android Virtual Device Manager.

- Pulsar en el botón “Nuevo” para crear un nuevo AVD. Aparecerá una ventana como la que se muestra a continuación:

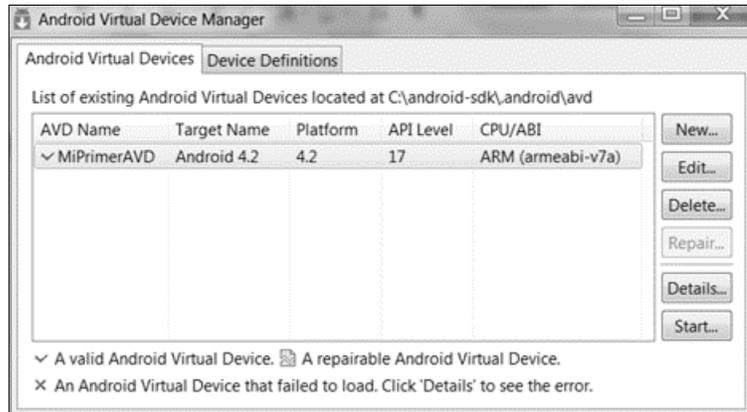
Figura 17. **Creación de un dispositivo virtual Android**



Fuente: elaboración propia.

- Una vez introducida la configuración se debe de pulsar la opción “Crear AVD”).
- Aparecerá el dispositivo creado en la ventana que aparece. Para iniciarlo se debe de seleccionar y pulsar el botón “Iniciar”.

Figura 18. **Android virtual device manager**



Fuente: elaboración propia.

- Una vez iniciado se puede ver una emulación de un dispositivo Android como se muestra a continuación.

Figura 19. **Emulación de dispositivo Android**



Fuente: elaboración propia.

4.3. Manual de uso de la aplicación

Se redactó una guía de uso para el sitio web y para la aplicación móvil desarrollada. Cada componente se describe a continuación.

4.3.1. Sitio web

La aplicación desarrollada cuenta con un sitio web donde puede encontrarse información, características de contacto, y otros temas relacionados con la aplicación. Dicho sitio web se puede ubicar en la URL: <http://seminariofinal.webege.com/>.

La página principal es la que se muestra cuando se ingresa al sitio web de la aplicación; esta contiene únicamente el logo de la aplicación y la opción para navegar por el sitio web.

Figura 20. **Página principal del sitio web**



Fuente: elaboración propia.

En el sitio web se pueden encontrar las características ilustradas propias de la aplicación desarrollada, tal y como se muestra en las siguientes ilustraciones.

Figura 21. **Características de GCloud I**



Fuente: elaboración propia.

Figura 22. **Características de GCloud II**



Fuente: elaboración propia.

Figura 23. **Características de GCloud III**



Fuente: elaboración propia.

En el sitio web se puede encontrar una sección donde se permite al usuario descargar la última actualización de la aplicación desarrollada. El link que se provee ahí descargará un archivo con extensión “apk”, el cual podrá ser instalado en el dispositivo donde se quiere utilizar dicha aplicación.

Figura 24. **Información y descarga de GCloud**



Fuente: elaboración propia.

El sitio web también cuenta con una sección donde se muestran las propiedades y beneficios de utilizar GCloud.

Figura 25. **Beneficios de utilizar GCloud**



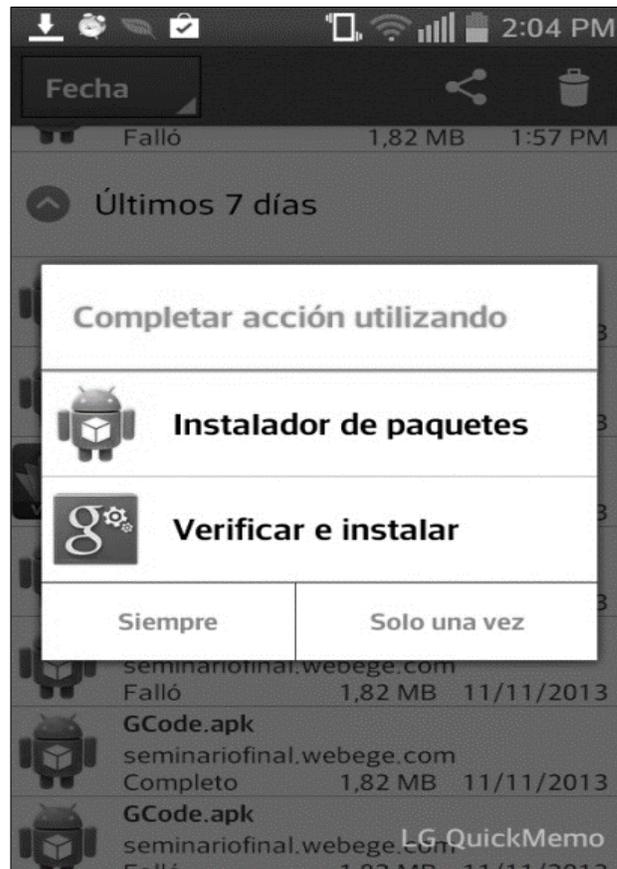
Fuente: elaboración propia.

4.3.2. Descarga e instalación de la aplicación

Desde el sitio web <http://seminariofinal.webege.com/> es posible descargar la última versión de la aplicación desarrollada. Una vez que el archivo con extensión “apk” ha sido descargado, se puede proceder a su instalación.

Para instalar la aplicación se debe seleccionar el archivo descargado y aparecerá el siguiente cuadro de diálogo. En ella se debe de seleccionar la opción “Instalador de paquetes”.

Figura 26. Descarga de la aplicación



Fuente: elaboración propia.

En el siguiente paso el proceso preguntará si se desean aceptar los permisos que la aplicación requiere. En este paso se debe de pulsar el botón “Instalar”.

Figura 27. **Autorización de permisos de la aplicación**



Fuente: elaboración propia.

En el último paso el dispositivo procederá a instalar la aplicación, como se muestra en las siguientes imágenes. Una vez finalizada la instalación, la aplicación está lista para ser utilizada.

4.3.3. **Uso de la aplicación**

Una vez que la aplicación ha sido instalada, ya es posible hacer uso de ella. A continuación se detalla cada una de las pantallas y la forma en que deben de ser utilizadas.

4.3.3.1. Pantalla de ingreso

Esta es la primera pantalla que se visualiza al ingresar a la aplicación. Esta permitirá ingresar con un usuario si es que ya se tiene un usuario creado. Para ello se debe ingresar el correo electrónico registrado y la contraseña asociada, seguido de pulsar el botón “Ingresar”. Si ocurre el caso contrario en el que el usuario no tiene una cuenta creada, es necesario crear una; para ello se debe de ingresar al vínculo “Regístrate aquí”.

Figura 28. Pantalla de ingreso

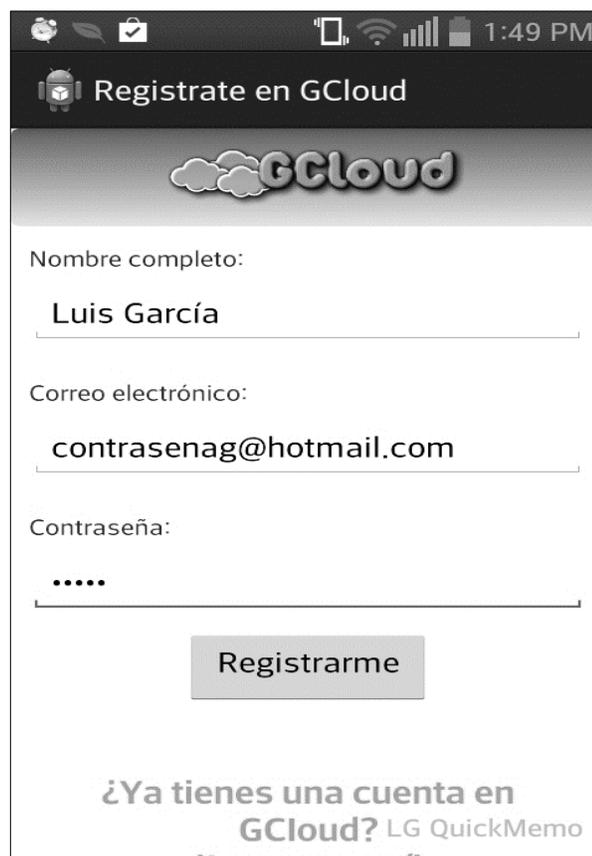


Fuente: elaboración propia.

4.3.3.2. Pantalla de registro

Para el registro de una nueva cuenta en GCloud se presenta al usuario la pantalla que se muestra a continuación. En ella se debe ingresar el nombre completo del usuario que está registrándose, el correo electrónico y una contraseña. Posteriormente se debe de seleccionar el botón “Registrarme” para concluir con el proceso.

Figura 29. Registro en GCloud



Registrate en GCloud

GCloud

Nombre completo:
Luis García

Correo electrónico:
contrasenag@hotmail.com

Contraseña:
.....

Registrarme

¿Ya tienes una cuenta en
GCloud? LG QuickMemo

Fuente: elaboración propia.

4.3.3.3. Panel de opciones

GCloud presenta al usuario un panel de opciones donde es posible navegar hacia cualquier parte de la aplicación y donde se muestran sus principales funcionalidades. Estas opciones son:

- Archivos
- Fotos
- Configuración
- Favoritos

El panel de opciones se muestra a continuación.

Figura 30. Panel de opciones



Fuente: elaboración propia.

4.3.3.4. Pantalla de configuración

Permite agregar cuentas de los diferentes servicios que GCloud soporta, como también la gestión y mantenimiento de las cuentas existentes.

Para agregar una cuenta, el usuario debe ingresar el correo que tiene registrado en el servicio y la contraseña. Luego seleccionar qué servicio es el que tiene asociado a esa cuenta y por último pulsar la opción “Agregar cuenta”.

Figura 31. Pantalla de configuración



Fuente: elaboración propia.

4.3.3.5. Pantalla de fotografías

La aplicación GCloud permite realizar filtros respecto de los tipos de archivos que se tienen almacenados en las cuentas de los usuarios. Mediante la pantalla de fotografías, los usuarios podrán ver todos los archivos que sean de formato imagen unificados en un solo lugar.

Figura 32. Pantalla de fotografías

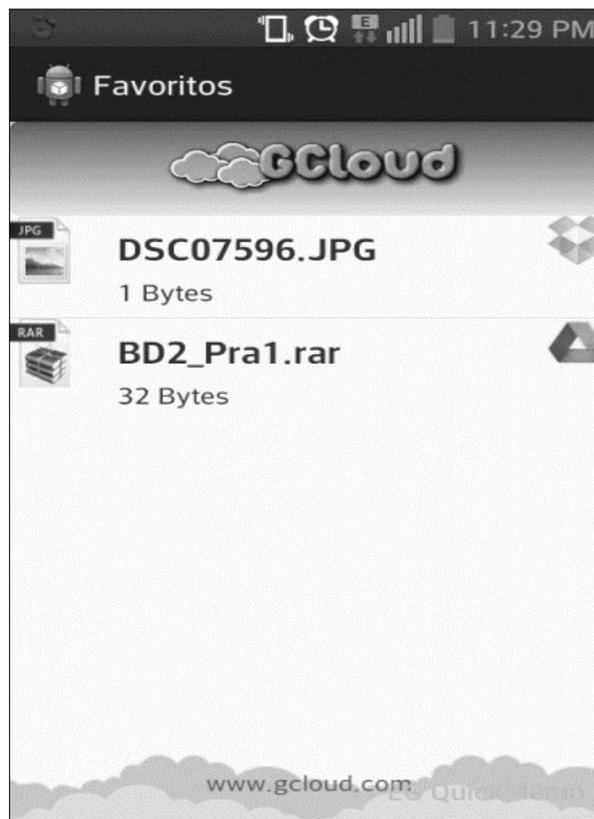


Fuente: elaboración propia.

4.3.3.6. Pantalla de favoritos

Al igual que en la pantalla de fotografías, la aplicación GCloud también permite realizar filtros de los archivos que se han marcado previamente como favoritos. En esta pantalla aparecerán dichos archivos, independientemente del servicio donde estos se encuentren almacenados.

Figura 33. Pantalla de favoritos



Fuente: elaboración propia.

4.3.3.7. Pantalla de archivos

La pantalla de archivos es la principal y de mayor uso en la aplicación GCloud. Mediante esta pantalla el usuario podrá visualizar mediante una lista todos los archivos que se encuentren en las cuentas en línea que se tengan registradas en la aplicación.

Mediante esta vista el usuario también podrá filtrar los archivos que se le presenten y marcar aquellos archivos importantes como favoritos. Dichos archivos son presentados en una vista unificada, sin importar en qué servicio se encuentren, y únicamente se podrá identificar por el icono que aparece del lado derecho, el cual identifica si pertenece al servicio de Dropbox o al de Google Drive. La vista también mostrará las carpetas que se tengan creadas en los servicios, siendo estas identificadas por el icono de fólдер que caracteriza a una carpeta. Los archivos igualmente están identificados por un icono característico del programa al que están asociados.

Figura 34. Pantalla de archivos



Fuente: elaboración propia.

4.3.3.7.1. Menú de opciones

La aplicación le presenta al usuario un menú en donde se podrá seleccionar una acción a realizar. El menú cuenta con las siguientes opciones:

- Favorito: permite marcar un archivo como favorito para que aparezca en la pantalla correspondiente de archivos frecuentes.
- Descargar: esta opción le permite al usuario realizar una descarga del archivo que se tenga seleccionado para el almacenamiento local en el dispositivo.
- Eliminar: esta opción le permite al usuario eliminar el archivo tanto de la aplicación GCloud, como el del servicio que lo tenga alojado.
- Subir archivo: esta opción le permite al usuario almacenar un archivo que se tiene localmente en el dispositivo, en el servicio que se tenga en ese momento en uso. El menú está disponible manteniendo presionada la pantalla encima de un nombre de archivo.

Figura 35. Menú de opciones



Fuente: elaboración propia.

CONCLUSIONES

1. Los servicios en la nube permiten ofrecer servicios vía internet en conjunto con sus modelos de servicios como infraestructura, plataforma y software, los cuales presentan características de gran utilidad para las empresas y los usuarios individuales, y genera oportunidades de desarrollo para ellos, ayudados por la calidad y rapidez de los servicios que se ofrecen en el mercado.
2. Se comprueba que la evolución acelerada de la computación en la nube depende de las necesidades de los usuarios o clientes y los factores determinantes como la capacidad, seguridad, servicios y otros que los clientes requieran para sus propósitos.
3. La computación en la nube es un modelo innovador que implica compartir los recursos, creando retos nuevos al tema de seguridad y privacidad, tanto para las empresas como para los usuarios individuales.
4. Los servicios para la gestión de archivos en la nube permiten a las empresas y usuarios ahorrar dinero y gastos en la capacidad de almacenamiento a través de internet.

RECOMENDACIONES

1. Es necesario conocer el servicio al que se le delegará la confianza de almacenar y gestionar la información desde el aspecto legal y contractual que garantiza la seguridad de la información, hasta las características que brinda para sacar el mayor provecho a la herramienta.
2. Se recomienda que las compañías que proveen los servicios en la nube, así como los usuarios, deberán prestar especial atención a los riesgos relacionados con la seguridad de la información asociados a este modelo.
3. Se propone hacer necesarios los procesos de análisis, diseño, desarrollo, pruebas y mantenimiento sobre computación en la nube, orientándose para su corrección, mejoramiento y puesta en producción final para los sistemas que utilizan APIs o manejan archivos a través de servicios como Google Drive o Dropbox.
4. Se recomienda incluir otros servicios de almacenamiento en la nube, como GBox, Ubuntu One, Amazon Cloud, iCloud y SugarSync a la funcionalidad de la aplicación desarrollada, para que tenga un valor agregado sobre las otras aplicaciones que existen en el mercado y esta pueda competir contra ellas.

BIBLIOGRAFÍA

1. *Acontecimientos de SaaS* [en línea]. <<http://www.saasmania.com/2010/01/20/saas-de-la-semana-quota-micro-erp/>>. [Consulta: 10 de octubre de 2013].
2. BENNETH, John. *Service based Software: the future for flexible Software* [en línea]. <<http://bds.ie/Pdf/ServiceOriented1.pdf>>. [Consulta: 15 de agosto de 2013].
3. BOTELHO, Bridget. *Hardware standardization benefits cloud computing virtualization* [en línea]. <<http://searchcloudcomputing.techtarget.com/news/1355038/Hardware-standardization-benefits-cloud-computing-virtualization>>. [Consulta: 08 de agosto de 2013].
4. CANO, Jeimy. *Cloud computing: evolución en los servicios de tecnología y grandes retos en seguridad* [en línea]. <<http://www.channelplanet.com/?idcategoria=23693>>. [Consulta: 20 de julio de 2013].
5. FALLA, Stephanie. *Cloud Computing: nueva era de desarrollo* [en línea]. <<http://www.maestrosdelweb.com/editorial/cloud-computing-nueva-era-de-desarrollo/>>. [Consulta: 15 de julio de 2013].

6. LEON, Ernest. *The five layers within Cloud Computing*. [en línea]. <<http://cloudcomputing.sys-con.com/node/1200642>>. [Consulta: 05 de septiembre de 2013].
7. MALCOLM, Dave. *The five defining characteristics of cloud computing* [en línea]. <<http://www.zdnet.com/news/the-five-defining-characteristics-of-cloud-computing/287001>>. [Consulta: 02 de octubre de 2013].
8. SAAVEDRA, Esteban. *Cloud computing*. [en línea]. <<http://www.slideshare.net/estebansaavedra/introduccion-al-cloud-computing>>. [Consulta: 02 de septiembre de 2013].
9. YORK, Joel. *Competitive advantage in software as a service*. [en línea]. <<http://www.chaoticflow.com/media/SaaS%20Competitive%20Advantage.pdf>>. [Consulta: 16 de septiembre de 2013].