



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

**IMPLEMENTACIÓN DE NUEVAS TECNOLOGÍAS MÓVILES ANDROID  
PARA DIFUSIÓN DE NOTICIAS DE CURSOS MEDIANTE LA RED SOCIAL**

**Alan Gonzalo Ucelo Morán**

**Jacob Israel Noj López**

Asesorados por el Ing. Ricardo Israel Mazariegos Castillo

Guatemala, octubre de 2013

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**IMPLEMENTACIÓN DE NUEVAS TECNOLOGÍAS MÓVILES ANDROID  
PARA DIFUSIÓN DE NOTICIAS DE CURSOS MEDIANTE LA RED SOCIAL**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA  
FACULTAD DE INGENIERÍA

POR

**ALAN GONZALO UCELO MORÁN**

**JACOB ISRAEL NOJ LÓPEZ**

ASESORADOS POR EL ING. RICARDO ISRAEL MAZARIEGOS CASTILLO

AL CONFERÍRSELES EL TÍTULO DE

**INGENIERO EN CIENCIAS Y SISTEMAS**

GUATEMALA, OCTUBRE DE 2013

UNIVERSIDAD 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. Walter Rafael Véliz Muñoz
VOCAL V	Br. Sergio Alejandro Donis Soto
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

**TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO**

DECANO	Ing. Murphy Olympo Paiz Recinos
EXAMINADOR	Ing. Pedro Pablo Hernández Ramírez
EXAMINADOR	Ing. Marlon Francisco Orellana López
EXAMINADOR	Ing. José Ricardo Morales Prado
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

**ALAN GONZALO UCELO MORÁN**

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA  
FACULTAD DE INGENIERÍA



**NÓMINA DE JUNTA DIRECTIVA**

DECANO	Ing. Murphy Olympto 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. Walter Rafael Véliz Muñoz
VOCAL V	Br. Sergio Alejandro Donis Soto
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

**TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO**

DECANO	Ing. Murphy Olympto Paiz Recinos
EXAMINADOR	Ing. Oscar Alejandro Paz Campos
EXAMINADOR	Ing. José Alfredo González Díaz
EXAMINADOR	Ing. Edgar Estuardo Santos Sutuj
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

**JACOB ISRAEL NOJ LÓPEZ**

## HONORABLE TRIBUNAL EXAMINADOR

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

### **IMPLEMENTACIÓN DE NUEVAS TECNOLOGÍAS MÓVILES ANDROID PARA DIFUSIÓN DE NOTICIAS DE CURSOS MEDIANTE LA RED SOCIAL**

Tema que nos fue asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha 16 de septiembre de 2012.

  
Jacob Israel Noj López

  
Alan Gonzalo Ucelo Morán

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



Guatemala, jueves 30 de mayo de 2013

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

Respetable Ingeniero Turk

Por este medio de la presente hago de su conocimiento que he revisado el trabajo de graduación titulado *"Implementación de nuevas tecnologías móviles Android para difusión de noticias de cursos mediante la red social"* elaborado por los estudiantes Alan Gonzalo Ucelo Morán y Jacob Israel Noj López, que se identifican con los carné 200714768 y 200715223 respectivamente.

En mi calidad de Asesor, he analizado el contenido, así como las conclusiones y recomendaciones expuestas y, a mi criterio, el mismo cumple con los objetivos propuestos para su desarrollo.

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

Atentamente,

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

Ing. Ricardo Mazariegos Castillo  
Asesor de Trabajo de Tesis  
ricardo.mazariegos@ingeniodp.com



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

Guatemala, 5 de Junio de 2013

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 **ALAN GONZALO UCELO MORÁN** carné 200714768 y **JACOB ISRAEL NOJ LÓPEZ** carné 200715223, titulado: **"IMPLEMENTACIÓN DE NUEVAS TECNOLOGÍAS MÓVILES ANDROID PARA DIFUSIÓN DE NOTICIAS DE CURSOS MEDIANTE LA RED SOCIAL"**, 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  
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 NUEVAS TECNOLOGÍAS MÓVILES ANDROID PARA DIFUSIÓN DE NOTICIAS DE CURSOS MEDIANTE LA RED SOCIAL”**, realizado por los estudiantes ALAN GONZALO UCELO MORÁN y JACOB ISRAEL NOJ LÓPEZ, aprueba el presente trabajo y solicita la autorización del mismo.*

**“ID Y ENSEÑAD A TODOS”**



Ing. Marlon Antonio Pérez Turk  
Director, Escuela de Ingeniería en Ciencias y Sistemas

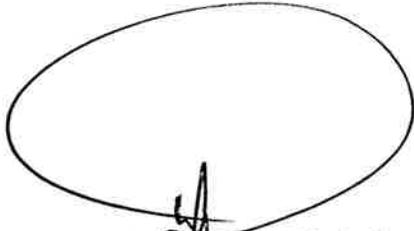
Guatemala, 01 de Octubre 2013



DTG. 677 .2013

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 NUEVAS TECNOLOGÍAS MÓVILES ANDROID PARA DIFUSIÓN DE NOTICIAS DE CURSOS MEDIANTE LA RED SOCIAL**, presentado por los estudiantes universitarios: **Alan Gonzalo Ucelo Morán y Jacob Israel Noj López**, autoriza la impresión del mismo.

IMPRÍMASE:

  
Ing. Murphy Olimpo Paiz Recinos  
Decano

Guatemala, 1 de octubre de 2013

/gdech



## **ACTO QUE DEDICO A:**

- Dios** Por ser una gran influencia en mi vida.
- Mis padres** José Gonzalo Ucelo Portillo y Egda Maribel Morán Pérez, por su amor sincero, ayuda incondicional, consejos y por ser siempre mi inspiración. Qué Dios los bendiga siempre.
- Mis hermanas** Luvia Carolina y Amarilis Ucelo Morán, por su apoyo, ayuda y comprensión durante la etapa de mi vida universitaria.
- Mis abuelos** Por su ayuda y sabios consejos que me ayudaron a perseverar durante la carrera.
- Mis sobrinos** Porque este triunfo sea un ejemplo para su futuro y los motive a alcanzar sus metas.
- Mis tíos y demás familia** Por su apoyo, consejos, entre otras cosas. Qué Dios los bendiga.

**Alan Gonzalo Ucelo Morán**

## **ACTO QUE DEDICO A:**

<b>Dios</b>	Por ser el creador.
<b>Mi madre</b>	Norma López, por su apoyo incondicional y amor a lo largo de toda mi vida, siendo la primera persona en creer en mí para este logro.
<b>Mis hermanos</b>	Isaac, Tabita, Jessé y Zoey Noj López por las alegrías y dificultades que hemos llevado juntos a través de nuestras vidas.
<b>Mis abuelos</b>	Por su ejemplo, para que sea una buena persona y un buen ciudadano. Gracias por sus consejos.

**Jacob Israel Noj López**

## **AGRADECIMIENTOS A:**

<b>La Universidad de San Carlos de Guatemala</b>	Por ser parte importante en la formación de mi carrera universitaria.
<b>Facultad de Ingeniería</b>	Por ser una importante influencia en mi formación profesional.
<b>Mi asesor y tutor</b>	Ing. Ricardo Mazariegos e Ing. Edgar por su asesoría y tutoría profesional brindada en la ejecución del presente trabajo de graduación desinteresadamente.
<b>Mis amigos en general</b>	Por el apoyo en tareas, prácticas, proyectos, etc., de todos los cursos durante la etapa de la carrera, así como las alegrías y tristezas compartidas. Qué Dios los bendiga siempre.

**Alan Gonzalo Ucelo Morán**

## **AGRADECIMIENTOS A:**

**La Universidad de San  
Carlos de Guatemala**

Por ser la casa de estudios que me formó profesionalmente.

**Facultad de Ingeniería**

Por ser el lugar de aprendizaje, tanto académico como humano, en mi vida enseñando la perseverancia y el coraje que se debe tener para seguir adelante ante cualquier situación.

**Mi asesor y tutor**

Ing. Ricardo Mazariegos e Ing. Edgar Santos por su asesoría y tutoría profesional brindada en la ejecución del presente trabajo de graduación desinteresadamente.

**Mis amigos de la  
Universidad**

Por apoyo y soporte otorgado durante todo el transcurso de la carrera, académicamente y humanamente, así como cada uno de las alegrías y dificultades que compartimos, otorgándome siempre su amistad.

**Jacob Israel Noj López**

## ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	VII
GLOSARIO .....	IX
RESUMEN.....	XV
OBJETIVOS.....	XVII
INTRODUCCIÓN .....	XIX
1. ESTUDIO DE LA TECNOLOGÍA Y SU IMPACTO EN GUATEMALA.....	1
1.1. Aplicaciones móviles en sistemas operativos Android .....	1
1.2. Datos históricos del sistema .....	2
1.3. Ventajas de Android .....	3
1.3.1. Android Marketplace.....	3
1.3.2. Es simple y poderoso .....	4
1.3.3. Usos y dispositivos .....	6
1.3.4. Diseño y desarrollo .....	6
1.3.5. Mejoras en las aplicaciones.....	7
1.3.6. Google Play (Android Market).....	8
1.3.7. Potente privacidad .....	8
1.3.8. Confiable seguridad.....	9
1.4. Redes sociales .....	9
1.4.1. Mercadotecnia en las redes sociales.....	10
1.4.2. Riesgos en las redes sociales .....	11
1.4.2.1. Menores de edad.....	11
1.4.2.2. Pishing.....	11
1.4.2.3. Privacidad .....	11
1.4.2.4. Perfiles falsos .....	12

	1.4.2.5. Trabajo .....	12
	1.4.3. Redes sociales más conocidas .....	12
1.5.	Red social Twitter.....	13
	1.5.1. Funcionamiento de Twitter .....	14
	1.5.2. ¿Qué son los <i>hashtags</i> ?.....	14
	1.5.3. ¿Por qué utilizar <i>hashtag</i> ?.....	15
	1.5.4. Otras aplicaciones que ofrece Twitter .....	15
	1.5.5. Resumen del desarrollo del proyecto de Twitter .....	15
1.6.	Identificando la teoría que fundamenta esta investigación.....	16
	1.6.1. Utilidad percibida.....	17
	1.6.2. Facilidad de uso percibida.....	17
1.7.	Teoría y la relación con la tecnología escogida .....	18
1.8.	Validación del modelo .....	18
	1.8.1. Variables externas.....	19
	1.8.2. Facilidad de uso percibida.....	20
	1.8.3. Utilidad percibida.....	20
	1.8.4. Actitud a través del uso .....	20
	1.8.5. Intención de uso .....	20
	1.8.6. Uso actual del sistema .....	21
2.	IDENTIFICACIÓN DEL PROBLEMA Y SOLUCIÓN QUE LA APLICACIÓN BRINDA.....	23
	2.1. Antecedentes .....	23
	2.2. Descripción del problema.....	26
	2.3. Desarrollo del software.....	28
	2.4. Mercado objetivo.....	29
	2.5. BenchMark de la aplicación .....	29
	2.5.1. Blackboard Mobile <sup>TM</sup> .....	30
	2.5.2. Schedule Timetable Deluxe.....	31

3.	DOCUMENTACIÓN BASE PARA LA PROGRAMACIÓN DE LA APLICACIÓN MÓVIL .....	33
3.1.	Requerimientos preliminares para el desarrollo de la aplicación.....	33
3.1.1.	Android SDK.....	34
3.1.2.	Dalvik Virtual Machine .....	34
3.2.	¿Cómo desarrollar aplicaciones Android? .....	35
3.3.	Recursos necesarios .....	36
3.3.1.	Ambiente de desarrollo .....	36
3.3.2.	Ambiente real.....	37
3.4.	Componentes de ambos ambientes .....	37
3.5.	Conceptos generales.....	37
3.5.1.	Actividad .....	38
3.5.2.	Fragmentos.....	38
3.5.3.	Puntos de vista y grupos de vistas .....	38
3.5.4.	Diferencias entre actividades, fragmentos y vistas .....	39
3.5.5.	Otros componentes de Android .....	40
3.5.5.1.	Intenciones .....	40
3.5.5.2.	Servicios .....	40
3.5.5.3.	Proveedor de contenidos.....	41
3.5.5.4.	Receptor de transmisión.....	41
3.5.5.5.	Pantalla de inicio.....	41
3.5.5.6.	Más componentes .....	42
3.5.6.	Aspectos específicos en aplicaciones Android .....	42
3.5.6.1.	AndroidManifest.xml .....	42
3.5.6.2.	Actividades y ciclo de vida .....	44
3.5.6.3.	Contexto .....	45
3.6.	Proceso de Instalación .....	45

3.6.1.	Pasos para descargar e instalar SDK para Android.....	46
3.6.2.	Pasos para descargar e instalar Eclipse .....	47
3.6.3.	Pasos para instalar el ADT.....	48
3.6.4.	Ejecutar una aplicación Android de prueba .....	50
4.	DESCRIPCIÓN DEL DESARROLLO DE LA APLICACIÓN COMO SOLUCIÓN AL PROBLEMA DE ESTUDIO .....	53
4.1.	Arquitectura diseñada .....	53
4.2.	Aplicación de servicio web .....	55
4.2.1.	Interfaz de comunicación.....	56
4.2.2.	Lógica de negocio .....	56
4.2.3.	Acceso a la base de datos .....	56
4.3.	Aplicación Android.....	58
4.4.	Otros aspectos a tomar en cuenta en la arquitectura.....	58
4.4.1.	Modelo.....	59
4.4.2.	Vista .....	59
4.4.3.	Controlador.....	59
4.5.	Componentes utilizados en la aplicación .....	60
4.5.1.	Actividades .....	60
4.5.1.1.	Relative Layout.....	61
4.5.1.2.	Absolute Layout.....	61
4.5.1.3.	Linear Layout.....	61
4.5.2.	Tab Activity.....	61
4.5.3.	List Activity .....	62
4.5.4.	List View .....	62
4.5.5.	Image View.....	63
4.5.6.	Text View.....	63
4.5.7.	Button .....	63

4.5.8.	Edit Text .....	63
4.5.9.	Alert Dialog .....	64
4.5.10.	JSON Parser.....	64
4.6.	Integración con API Twitter.....	64
4.6.1.	Registrar la aplicación Twitter.....	65
4.6.2.	Configurar la aplicación Twitter.....	66
4.6.3.	Configurar parámetros iniciales en la aplicación.....	68
4.7.	Aplicación de registro .....	72
5.	DOCUMENTACIÓN DE USUARIO PARA USO DE LA APLICACIÓN DESARROLLADA .....	75
5.1.	Resumen de la aplicación.....	75
5.2.	Proceso básico de la aplicación .....	76
5.3.	Pantalla principal .....	77
5.4.	Proceso de registro o inicio de sesión .....	78
5.4.1.	Proceso de registro.....	78
5.4.2.	Proceso de inicio de sesión .....	87
5.5.	Pantalla principal de usuario.....	89
5.5.1.	Menú para la pantalla principal de usuario .....	89
5.6.	Perfil de usuario.....	91
5.6.1.	Menú para el perfil de usuario .....	92
5.7.	Pantalla para cursos.....	93
5.8.	Consultar y crear mensajes ( <i>tweets</i> ) .....	95
5.8.1.	Consultar <i>tweets</i> .....	95
5.8.2.	Crear <i>tweets</i> .....	96
	CONCLUSIONES .....	101
	RECOMENDACIONES.....	103
	BIBLIOGRAFÍA.....	105



## ÍNDICE DE ILUSTRACIONES

### FIGURAS

1.	Modelo de Aceptación Tecnológica (TAM).....	19
2.	Crecimiento de la telefonía móvil y fija en Guatemala.....	25
3.	Fragmentos en pequeñas y grandes pantallas .....	39
4.	Ejemplo de archivo AndroidManifest.xml .....	43
5.	Modificación de variables del sistema operativo .....	46
6.	Ruta para el área de trabajo de las aplicaciones .....	47
7.	Actualización del software.....	49
8.	Definiendo la ruta del SDK.....	50
9.	Crear un nuevo proyecto en Android.....	52
10.	Diagrama de la arquitectura .....	54
11.	Diagrama de Modelo Vista Controlador .....	58
12.	Crear aplicación en la página de desarrolladores Twitter.....	65
13.	Cambiando el ícono de la aplicación Twitter .....	66
14.	Seleccionando el tipo de aplicación Twitter.....	67
15.	Configurando la autenticación en Twitter .....	67
16.	Ejemplo de archivo Manifest de la aplicación desarrollada .....	68
17.	Variables para validación de autenticación Twitter .....	69
18.	Variables para Twitter, RequestToken y SharedPreferences.....	70
19.	Proceso para solicitar el inicio de sesión en Twitter .....	71
20.	Validación de autenticación en la página de Twitter .....	72
21.	Pantalla inicial de la aplicación.....	77
22.	Opción para registrar nuevo usuario .....	78
23.	Pantalla de registro de usuario nuevo .....	79

24.	Pantalla de registro de usuario nuevo.....	80
25.	Usuario ingresado erróneamente.....	81
26.	Credenciales de acceso incorrectas .....	82
27.	Usuario registrado anteriormente.....	83
28.	Opción de ingreso a la aplicación en pantalla de registro .....	84
29.	Iniciar sesión en Twitter .....	85
30.	Cargando cursos en interfaz principal.....	86
31.	Pantalla principal del usuario en la aplicación .....	86
32.	Opción para iniciar sesión.....	87
33.	Iniciar sesión con un usuario registrado en la aplicación .....	88
34.	Menú para la pantalla principal de usuario .....	90
35.	Perfil de usuario .....	91
36.	Menú para el perfil de usuario .....	92
37.	Pantalla para cursos .....	93
38.	Seleccionar año para gestionar cursos.....	94
39.	Seleccionar el semestre para gestionar cursos .....	95
40.	Crear <i>tweet</i> para un curso específico .....	96
41.	Pantalla para crear nuevo <i>tweet</i> .....	97
42.	Actualizar en Twitter al crear un nuevo <i>tweet</i> .....	98
43.	<i>Tweet</i> creado de forma correcta .....	99
44.	<i>Tweet</i> con mensaje de estatus vacío.....	100
45.	<i>Tweet</i> con estatus que excede el tamaño máximo .....	100

## GLOSARIO

<b>ADB</b>	Android Debug Bridge (ADB) es una herramienta de línea de comandos versátil que le permite comunicarse con una instancia de emulador o dispositivo conectado con tecnología Android.
<b>ADT</b>	Android Development Tools (ADT) es un plugin para el IDE de Eclipse que está diseñado para darle un ambiente potente, integrado en el que la construcción de aplicaciones de Android.
<b>Android</b>	Es una plataforma móvil, que incluye un sistema operativo móvil basado en Linux, junto con un conjunto de aplicaciones Middleware y aplicaciones nativas. Está enfocado para ser utilizado en dispositivos móviles como teléfonos inteligentes, tabletas, Google TV y otros dispositivos.
<b>Android Market Place</b>	Es una tienda de software en línea desarrollada por Google para los dispositivos Android. Es una aplicación que está preinstalada en la mayoría de los dispositivos Android y que permite a los usuarios buscar y descargar aplicaciones publicadas por desarrolladores terceros.

**Aplicación Middleware**

Es un software que sirve como asistente para una aplicación; permite interactuar o comunicarse con otras aplicaciones, redes, hardware o sistemas operativos.

**Arquitectura RAM**

Es una arquitectura RISC (Reduced Instruction Set Computer, o bien dicho, Ordenador con Conjunto de Instrucciones Reducidas) de 32 bits desarrollada por ARM Holdings. Se llamó Advanced RISC Machine y anteriormente Acorn RISC Machine.

**Base de Datos (BD)**

Es una estructura de datos estática donde se puede almacenar información de forma persistente, no importando si el dispositivo electrónico es removido de corriente eléctrica.

**Benchmark**

Es una técnica utilizada para medir el rendimiento de un sistema o componente del mismo, frecuentemente en comparación con el que se refiere específicamente a la acción de ejecutar un benchmark. La palabra benchmark es un anglicismo traducible al español como comparativa.

**Eclipse**

Es un entorno de desarrollo integrado de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores.

**Hashtag**

Es una cadena de caracteres formada por una o varias palabras concatenadas y precedidas por una almohadilla o gato (#). Se usa en servicios web tales como Twitter, FriendFeed, Facebook o en mensajería para señalar un tema sobre el que gira cierta conversación.

**Interfaz**

Es un término que procede del vocablo inglés interface ("superficie de contacto"). En informática, esta noción se utiliza para nombrar a la conexión física y funcional entre dos sistemas o dispositivos de cualquier tipo.

**J2ME**

Es una especificación de un subconjunto de la plataforma Java orientada a proveer una colección certificada de API (aplicaciones) de desarrollo de software para dispositivos con recursos restringidos. Está orientado a productos de consumo como PDA, teléfonos móviles o electrodomésticos.

**JDK**

Java Development Kit por sus siglas en inglés, es un software que provee herramientas de desarrollo para la creación de programas en Java. Puede instalarse en una computadora local o en una unidad de red.

**JSON**

Acrónimo de JavaScript Object Notation, es un formato ligero para el intercambio de datos. JSON es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML.

**Microblogging**

Es un servicio que permite a sus usuarios enviar y publicar mensajes breves (alrededor de 140 caracteres), generalmente solo de texto. Las opciones para el envío de los mensajes varían desde sitios web, a través de SMS, mensajería, etc.

**RAM**

Una Memoria de Acceso Aleatorio o RAM (siglas en inglés de Random Access Memory) es un componente electrónico donde se almacena la información en tiempo de ejecución de un programa. Esta memoria deja de almacenar la información cuando ya no se le provee de alimentación eléctrica.

**SDK**

Un *kit* de desarrollo de software o SDK (siglas en inglés de Software Development Kit) es generalmente un conjunto de herramientas de desarrollo de software que le permite al programador crear aplicaciones para un sistema concreto, por ejemplo ciertos paquetes de software, plataformas de hardware, computadoras, videoconsolas, etc.

**Sistema operativo**

Un sistema operativo o SO (frecuentemente OS, siglas en Inglés de Operating System) es un programa o conjunto de programas que en un sistema informático gestiona los recursos de hardware y provee servicios a los programas de aplicación, ejecutándose en modo privilegiado respecto de los restantes.

***Smartphone***

Nombre que se utiliza comúnmente para identificar a un teléfono inteligente, el cual es un teléfono móvil construido sobre una plataforma informática móvil, con una mayor capacidad de computación y conectividad que un teléfono móvil convencional.

**TIC**

Las Tecnologías de la Información y la Comunicación (TIC o bien NTIC para Nuevas Tecnologías de la Información y de la Comunicación) agrupan los elementos y las técnicas usadas en el tratamiento y la transmisión de la información, principalmente informática, Internet y las telecomunicaciones.

***Tweet***

Es una palabra creada a partir del inicio de la red social Twitter y hace referencia a escribir un mensaje dentro de la red social.

***Updates***

En informática está relacionado con las actualizaciones que se realizan sobre algún programa informático desarrollado. Se utiliza el vocablo en inglés para su uso en el lenguaje informático español.

**URL**

Un localizador uniforme de recursos o URL (siglas en inglés de Uniform Resource Locator, es una secuencia de caracteres, de acuerdo a un formato modélico y estándar, que se usa para nombrar recursos en Internet para su localización o identificación, como por ejemplo documentos textuales, imágenes, vídeos, presentaciones digitales, etc.

## RESUMEN

Actualmente el uso de redes sociales por parte de los usuarios de todo el mundo se ha convertido en una actividad cada vez más frecuente. La mayoría de las personas que tienen conocimiento en computación o dispositivos móviles, tienen el deseo de crear una cuenta en particular para poder compartir su información con otras personas y asimismo informarse de los servicios que las redes sociales ofrecen. Debido a la creciente popularización que han tenido las redes sociales y con la variedad de servicios existentes, ha surgido la necesidad por parte de los usuarios de vincularlos todos de alguna forma.

Por tal razón y debido a las necesidades que la Facultad de Ingeniería tiene para difundir la información y mantener avisados a todos los alumnos, surge la motivación de realizar una aplicación difusora de noticias como trabajo de graduación, con el propósito de aportarla a la facultad y adicionalmente un documento, tipo manual, que describe el proceso de creación de la aplicación base, para que en un futuro, los interesados puedan implementarla y/o ampliarla. La aplicación permite que los alumnos y catedráticos puedan compartir información a través de ella y el uso de la red social Twitter; relacionada a los cursos que tienen asignados los alumnos en cierta temporada (semestre o vacaciones).

Este documento contiene información técnica de cada uno de los sistemas clave que en la aplicación interactúan, como también se ofrece una guía tutorial que describe la creación de un proceso básico de flujo de comunicación que facilita su implementación.



## **OBJETIVOS**

### **General**

Proporcionar una aplicación para dispositivos móviles que ejecuten el sistema operativo Android, utilizado por cualquier estudiante para el control de sus cursos y las noticias en la red social de Twitter correspondiente, con el fin de tener una mejor difusión de las noticias utilizando tecnologías de la actualidad.

### **Específicos**

1. Crear un manual de creación de aplicaciones para dispositivos móviles Android, dirigidos a estudiantes de los primeros años de la carrera de Ingeniería en Ciencias y Sistemas de la Universidad de San Carlos de Guatemala.
2. Modernizar al estudiante de la Facultad de Ingeniería utilizando Android y Twitter como tecnologías para la mejor difusión de las noticias asociado a cada estudiante dentro de dicha facultad.
3. Mostrar la importancia de la creación de nuevas aplicaciones que le brinden valor agregado al estudiante con el fin de fomentar la innovación dentro de la Facultad de Ingeniería.



## INTRODUCCIÓN

La Universidad de San Carlos de Guatemala es la universidad estatal, es por eso que su Facultad de Ingeniería ha sido considerada como la cuna del conocimiento de Guatemala, razón por la que siempre ha estado a la vanguardia de los cambios y actualizaciones que la época moderna impone. De esa cuenta, es importante implementar nuevas tecnologías que faciliten la comunicación e interacción del proceso de enseñanza aprendizaje y que a la vez, provean valor agregado a los estudiantes de la facultad y en un futuro inmediato sea aplicado en toda la universidad.

Dentro de los cursos de la Escuela de Ciencias y Sistemas, las nuevas tecnologías están siendo introducidas con muy poca penetración al pensum y sin un manual sólido basado en investigación para la enseñanza de los nuevos profesionales, lo cual es de vital importancia en el perfil que debe de obtener el estudiante para egresar como profesional, no únicamente con una base sólida de conocimientos, sino también como conocedor de las nuevas tecnologías que están tomando auge actualmente en el mercado de la informática.

Este trabajo de graduación está compuesto por cinco capítulos, de los cuales los primeros dos capítulos brindan información relevante de la tecnología actual en Guatemala y asimismo, se identifica el problema y solución actual que presenta la situación por la cual se realiza este trabajo de graduación. Los siguientes tres capítulos brindan información básica del proceso de desarrollo e implementación (llamado manual técnico), así como la explicación de la utilización de la herramienta (llamado manual de usuario).



# **1. ESTUDIO DE LA TECNOLOGÍA Y SU IMPACTO EN GUATEMALA**

En este capítulo se hace una breve referencia sobre el estudio de las tecnologías, su avance y desarrollo, así como el impacto que están alcanzando en el contexto guatemalteco.

También incluye parte de la teoría que permitió fundamentar la investigación, tomando en cuenta elementos históricos que permitieron conocer los momentos por los cuales ha atravesado el desarrollo de la tecnología, como también, se hizo acopio de aquellos estudios realizados que aportan soluciones a las distintas disciplinas y que se relacionan con la internacionalización y enfoque competitivo mediante las innovaciones y cambios.

Este marco teórico permitió identificar los aspectos fundamentales que permitieron analizar y explicar durante el proceso investigativo el problema objeto de estudio, delimitar y organizar con precisión los temas y subtemas, como también, contar con elementos para argumentar la propuesta que se ofrece al final de este trabajo de graduación.

## **1.1. Aplicaciones móviles en sistemas operativos Android**

Actualmente los dispositivos móviles se están desarrollando sobre varias plataformas. La aplicación se desarrolla en el sistema operativo Android. Este sistema operativo móvil basado en Linux, junto con aplicaciones Middleware, está enfocado para ser utilizado en dispositivos móviles como teléfonos

inteligentes, tabletas, Google TV y otros dispositivos novedosos. Es desarrollado por la Open Handset Alliance, la cual es liderada por Google.

Este sistema por lo general maneja aplicaciones como Google Play. El sistema permite programar aplicaciones en una variación de Java llamada Dalvik.

El sistema operativo proporciona todas las interfaces necesarias para desarrollar aplicaciones que accedan a las funciones del teléfono (como el GPS, llamadas, agenda, etc.) de una forma muy sencilla en un lenguaje de programación muy conocido como lo es Java.

Una de las mejores características de este sistema operativo es que es completamente libre. Es decir, ni para programar en este sistema, ni para incluirlo en un teléfono hay que realizar un pago al respecto. Y esto lo hace muy popular entre fabricantes y desarrolladores, ya que los costes para lanzar un teléfono o una aplicación son muy bajos.

Cualquiera puede descargar el código fuente, inspeccionarlo, compilarlo e incluso cambiarlo; esto da seguridad a los usuarios ya que permite detectar fallos más rápidamente, ayudando incluso a los fabricantes, pues estos pueden adaptar mejor el sistema operativo a los terminales.

## **1.2. Datos históricos del sistema**

Android era un sistema operativo para móviles prácticamente desconocido hasta que Google lo compró en 2005. Hasta noviembre de 2007 solo hubo rumores, pero en esa fecha se lanzó la Open Handset Alliance, que agrupaba a muchos fabricantes de teléfonos móviles, conjunto de circuitos integrados

diseñados con base a la arquitectura de un procesador donde Google proporcionó la primera versión de Android, junto con el SDK para que los programadores empezaran a crear las aplicaciones para este sistema.

Aunque los inicios fueron un poco lentos, debido a que se lanzó antes el sistema operativo que el primer móvil, rápidamente se ha colocado como el sistema operativo de móviles más vendido del mundo, situación que se alcanzó en el último trimestre de 2010. En febrero de 2011 se anunció la versión 3,0 de Android, llamada con nombre en clave: Honeycomb.

Honeycomb, es la versión 3,2 del sistema operativo Android, liberado el 15 de julio de 2011; está optimizado para tabletas en lugar de teléfonos móviles. En este sistema se trabajó sobre el apartado 3D para tener una mejor visualización en todas las aplicaciones; también tiene mejora en la manipulación multitareas para el usuario final. Por tanto Android ha trascendido a partir de este sistema operativo en los teléfonos móviles para trascender a dispositivos más grandes.

### **1.3. Ventajas de Android**

A continuación se dan a conocer algunas de las ventajas principales del sistema operativo Android respecto a otros dispositivos en el mercado.

#### **1.3.1. Android Marketplace**

Al analizar los avances de lo que se puede realizar con una computadora, se establece que uno de los aspectos más importantes de un sistema operativo tan avanzado, es la cantidad de aplicaciones disponibles. El Android Market todavía no cuenta con la misma cantidad de aplicaciones que el

iPhone, pues el número de aplicaciones disponibles ha aumentado considerablemente en el transcurso del tiempo.

El impacto de los avances se refleja en que la mayoría de redes sociales importantes (Facebook, Foursquare, LinkedIn y Twitter), cuentan con clientes disponibles para la aplicación. Todos se integran de manera transparente a Android, por lo que se puede recibir notificaciones instantáneas apenas alguien comenta en el perfil, por ejemplo.

Y eso no es todo; con cada aplicación se puede sincronizar diferentes detalles de diferentes redes sociales, con los contactos existentes. ¿Se tiene un amigo que tiene cuentas en Facebook, Twitter, LinkedIn y además está en la libreta de direcciones? Todo esto quedará sincronizado automáticamente, anexando la información proporcionada en cada una de las redes sociales y unificándola bajo un solo contacto, todo lo anterior descrito, está disponible bajo Android Marketplace. Atrás quedaron los días de ir obteniendo aplicaciones en diferentes páginas web, porque ahora se tiene a todas disponibles bajo un mismo techo.

### **1.3.2. Es simple y poderoso**

En los primeros días de los teléfonos inteligentes, estos ofrecían muchas funciones pero, a la vez, traían algo más: complicaciones innecesarias. No era raro, por ejemplo, saltar menú tras menú para realizar una función tan simple, como leer los mensajes. Seguramente se escucha a personas decir que prefieren un teléfono Nokia por lo simple que resultaba navegar entre menús. Pues bien, uno de los puntos centrales de Android, es mantener esta simplicidad en todas las opciones.

No importa qué modelo de Android se adquiera, todos tendrán menús idénticos para llegar a la mayoría de opciones. Todos los teléfonos con Android tienen, por defecto, cuatro botones: el botón de retroceder, el botón de menú, el botón “Home” (que los lleva a la pantalla principal) y el botón de “Search”. Este último botón es un sistema operativo de Google, que no solo busca entre las aplicaciones y archivos, sino que además, puede buscar en la web directamente.

Y esta simplicidad en el manejo, se refleja también en todas las opciones de la interfaz. La mayoría de tareas está a un par de “*tap*” de distancia (ahora ya no se le puede llamar “clic”). Es un sistema simple, pero gracias a las aplicaciones, tan poderoso como una computadora de escritorio.

Es el más elegante sistema de multitarea dado que otros teléfonos limitan las aplicaciones que se pueden correr simultáneamente. Es decir, si el usuario está jugando (por ejemplo) y luego desea revisar el correo, en otros sistemas operativos se tiene que cerrar una aplicación para así poder ir a la aplicación de correo electrónico. Terminando de revisar el correo electrónico, el usuario debe cerrar la aplicación de correo, para finalmente poner a cargar nuevamente el juego. En Android, es posible mandar el juego “a dormir”, poniéndolo en pausa, mientras el usuario visualiza el correo.

Con el sistema operativo Android, el usuario no debe estar abriendo la aplicación de correo dado que el teléfono automáticamente le avisará, por ejemplo, cuando lleguen nuevos mensajes, o cuando alguien deje un mensaje instantáneo. Simplemente debe deslizar el dedo por la parte superior de la pantalla hacia abajo y se visualiza todo tipo de notificaciones.

### **1.3.3. Usos y dispositivos**

El sistema operativo Android se usa en teléfonos inteligentes, ordenadores portátiles, *netbooks*, tabletas, Google TV, relojes de pulsera, auriculares y otros dispositivos, siendo este sistema operativo accesible desde terminales de menos de Q.1 000,00 hasta terminales que superen los Q.7 000,00 obviando, evidentemente las diferencias técnicas.

La plataforma de hardware principal de Android es la arquitectura ARM (Advanced RISC Machine). Hay soporte para x86 en el proyecto Android-x86 y Google TV utiliza una versión especial de Android x86.

El primer teléfono inteligente disponible en el mercado para ejecutar Android fue el HTC Dream, dado a conocer al público el 22 de octubre de 2008. A principios de 2010, Google ha colaborado con HTC para lanzar su producto estrella en dispositivos Android, el famoso: Nexus One. A esto siguió en 2010 el Samsung Nexus S y en 2011 el Galaxy Nexus. En la actualidad existen más de 650 000 aplicaciones para Android y se estima que un millón de teléfonos móviles se activan diariamente.

Los teléfonos inteligentes iOS y Android Gingerbread 2.3.3 pueden ser configurados para un arranque dual en un iPhone o iPod Touch liberados con la ayuda de iDroid y OpeniBoot.

### **1.3.4. Diseño y desarrollo**

Android, al contrario que otros sistemas operativos para dispositivos móviles como iOS o Windows Phone, se desarrolla de forma abierta y se puede

acceder tanto al código fuente como al listado de incidencias donde se pueden ver problemas aún no resueltos y reportar problemas nuevos.

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

En los comienzos, Android era eminentemente un sistema operativo pensado para usar con teclado y que con la ayuda de un cursor, tener la capacidad de poder navegar entre las aplicaciones. Desde su comienzo, Android ha sido altamente personalizable. Poco después, antes del lanzamiento del primer teléfono Android, esta filosofía cambió para convertirse en eminentemente táctil y poder competir contra el famoso iPhone.

### **1.3.5. Mejoras en las aplicaciones**

Las aplicaciones se desarrollan habitualmente en el lenguaje Java con Android Software Development Kit (Android SDK), pero están disponibles otras herramientas de desarrollo, incluyendo un *kit* de desarrollo nativo para aplicaciones o extensiones en lenguaje C o C++, Google App Inventor, un entorno visual para programadores novatos y varias aplicaciones de la plataforma web móvil, teniendo la posibilidad de usar Librerías Qt.

El desarrollo de aplicaciones para Android no requiere aprender lenguajes complejos de programación. Todo lo que se necesita es un conocimiento

aceptable de Java y estar en posesión del *kit* de desarrollo de software, es decir SDK, el cual es provisto por Google y se puede descargar gratuitamente.

Todas las aplicaciones están comprimidas en formato APK, que se pueden instalar sin dificultad desde cualquier explorador de archivos en la mayoría de dispositivos.

### **1.3.6. Google Play (Android Market)**

Es la tienda en línea de software desarrollado por Google para dispositivos Android. Una aplicación llamada "Play store" que se encuentra instalada en la mayoría de los dispositivos Android, permite a los usuarios navegar y descargar aplicaciones publicadas por los desarrolladores. Google retribuye a los desarrolladores el 70% del precio de las aplicaciones.

Por otra parte, los usuarios pueden instalar aplicaciones desde otras tiendas virtuales, tales como Amazon, Appstore, SlideME, o incluso directamente en el dispositivo si en dado caso se dispone del archivo APK de la aplicación.

### **1.3.7. Potente privacidad**

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

### **1.3.8. Confiable seguridad**

Según un estudio de Symantec a mediados de 2011, comparado con el iOS, Android es un sistema más vulnerable, debido principalmente a que el proceso de certificación de aplicaciones es menos riguroso que el de Apple y el esquema de gestión de permisos que pone muchas decisiones de seguridad importantes en manos del usuario, lo expone a mayores riesgos y le hace más atractivo a los ataques de ingeniería social. Sin embargo, con base en números absolutos contabilizados de los boletines de seguridad de Microsoft, Adobe, Oracle y Apple, el 85% de los agujeros de seguridad en 2011 sucedieron en sistemas iOS, a pesar que tiene una cuota de mercado inferior a la de Android.

### **1.4. Redes sociales**

Las redes sociales son comunidades virtuales donde los propios usuarios interactúan con personas de todo el mundo, con quienes encuentran gustos o intereses en común. Funcionan como una plataforma de comunicación que permite conectar gente que se conoce o que desea conocerse y que les permite centralizar recursos, como fotos y vídeos, en un lugar fácil de acceder y administrado por los mismos usuarios. Internet es un medio propicio para hacer este tipo de conexiones y contrario a la creencia popular que la primera red social disponible en Internet no es Facebook (del 2004), sino una página llamada: Classmates.com (que todavía existe), creada en 1995.

Las redes sociales se basan en una gran medida en la teoría conocida como “Seis grados de separación” (publicada en 1967), que básicamente dice que todos las habitantes del mundo están conectados entre sí, con un máximo de seis personas como intermediarios en la cadena.

Por ejemplo: si se conoce a alguien que a su vez conoce a alguien más y esa persona a otra, quien a su vez conoce a otra más, se forma una cadena de seis o menos personas de por medio. Esto hace que el número de personas conocidas en la cadena crezca exponencialmente conforme el número de conexiones crece, dando por resultado que solo se necesita un número pequeño de estas conexiones para formar una red que conecta a todos, haciendo teóricamente posible hacer llegar un mensaje a cualquier persona en el mundo (se necesitan precisamente seis niveles de conexiones, o bien dicho: grados de separación, para conectar un billón de personas si se asume que cada persona conoce a cien personas en promedio).

#### **1.4.1. Mercadotecnia en las redes sociales**

Dado que las redes sociales son de fácil acceso a quienes cuentan con una conexión a Internet, surgen los programas de mercadotecnia en las redes sociales. Estos programas básicamente tienen como objetivo crear el contenido que además de atraer la atención de la gente, también la motive a compartir dicho contenido. Este contenido, en teoría, debe tener un mayor impacto en la gente que lo recibe porque proviene de una fuente confiable (como amigos y familiares).

La creación de páginas de productos y compañías, por ejemplo en Twitter y Facebook, permiten que la gente interactúe con dichos productos y compañías, creando una relación que se percibe como personalizada. Estas páginas permiten además que los productos y compañías envíen mensajes e incluso tengan conversaciones, creando mayor lealtad hacia el producto que se está adquiriendo.

## **1.4.2. Riesgos en las redes sociales**

Existen ciertos riesgos que vienen de la mano con el uso de redes sociales. Algunos de estos se listan a continuación.

### **1.4.2.1. Menores de edad**

A pesar de que varias redes sociales tienen una edad mínima para usuarios, es relativamente fácil falsificar información. Es importante que se haga control paterno para evitar que los menores de edad sean víctimas de depredadores que están al asecho en redes sociales.

### **1.4.2.2. Pishing**

Es un término informático que denomina un tipo de abuso informático y que se comete mediante el uso de un tipo de ingeniería social caracterizado por intentar adquirir información confidencial de forma fraudulenta (como puede ser una contraseña o información detallada sobre tarjetas de crédito u otra información bancaria). Existe un gran número de estafas, la mayoría de ellas enfocadas a obtener la información personal. Por ejemplo, en Facebook está el engaño de "¿estás en este vídeo?".

### **1.4.2.3. Privacidad**

La información que se publica en una red social debe ser tratada con la importancia de la misma. Esta información privada podría referirse a números de teléfono, dirección, correos electrónicos, los cuales pudieran ser usados para dañar a cierta persona si caen en las manos equivocadas.

Es importante que el usuario haga la configuración de privacidad necesaria para que la información que se publica esté visible solo para las personas en las que realmente se confía.

#### **1.4.2.4. Perfiles falsos**

Perfiles en redes sociales de áter ego son de más comunes de lo que se puede imaginar. Existen diversas motivaciones para esto, siendo las de mala intención el robo de identidad, pedófilos, secuestradores, chantajistas, entre otros. Hay que poner atención a quiénes los usuarios agregan como amigos en las redes sociales, especialmente personas que no se conocen y que tienen una fotografía de perfil demasiado atractiva para ser su verdadera identidad. En todo caso, si al usuario le interesa conocer gente nueva y la foto de perfil es demasiado atractiva para resistirse, puede usar filtros de privacidad (como listas en Facebook o círculos de Google+) para separar gente que se conoce de la que no se conoce bien.

#### **1.4.2.5. Trabajo**

Los usuarios deben ser discretos con lo que publican, sobre todo si las personas con la que se relacionan en el ámbito laboral es parte de los contactos en la red social personal. Las redes sociales se dedican a perseguir a los personajes públicos en busca de un artículo o una fotografía más o menos única.

#### **1.4.3. Redes sociales más conocidas**

Existe un alto número de redes sociales, varias de las cuales cubren intereses muy específicos. En este trabajo de graduación se concentra sobre la

red social Twitter debido a la importancia que está teniendo a nivel mundial. A continuación se detalla una lista de las redes sociales más usadas:

- Twitter – para uso general a través de seguidores
- Badoo – para uso general, muy popular en Latinoamérica
- Facebook – para uso general
- Flickr – para uso de fotografía
- Google+ – uso general
- Hi5 – para uso general, popular en Latinoamérica
- LinkedIn – para redes profesionales
- MySpace – para uso general
- NowPublic – para noticias sociales generales por usuarios
- Reddit – para noticias sociales generales por usuarios
- Sonico.com – para uso general, popular en países de habla hispana
- StumbleUpon – para páginas en Internet con un interés en específico
- Taringa! – para uso general
- YouTube – para uso general de sintonización de vídeos

### **1.5. Red social Twitter**

Twitter es una aplicación web gratuita de *microblogging* (servicio que permite a los usuarios enviar y publicar mensajes breves, alrededor de 140 caracteres, generalmente solo de texto) que reúne las ventajas de los blogs, las redes sociales y la mensajería instantánea. Esta nueva forma de comunicación, permite a los usuarios estar en contacto en tiempo real con personas de su interés a través de mensajes breves de texto, a los que se denominan actualizaciones o *tweets*, por medio de una sencilla pregunta: “¿Qué estás haciendo?” o también: “¿Qué pasa?”.

### 1.5.1. Funcionamiento de Twitter

En Twitter los usuarios envían y reciben actualizaciones de otros usuarios a través de breves mensajes que no deben superar los 140 caracteres, vía web, teléfono móvil, mensajería instantánea o a través del correo electrónico; e incluso desde aplicaciones de terceros, como pueden ser Twitterrific, Facebook, Twitterlicious, Twinkle y muchas otras.

El usuario puede estar al día de todo en la sección de perfil, tanto de los seguimientos (*following\_profile*) como de los seguidores (*followers\_profile*). Además de buscar amigos, familiares, compañeros u otras personas de interés, Twitter también ofrece otras opciones, como buscar en otras redes, invitar a amigos por correo electrónico o seleccionar a usuarios recomendados.

### 1.5.2. ¿Qué son los *hashtags*?

Un *hashtag* es simplemente una palabra o serie de caracteres precedidos por el símbolo #. El *hashtag* ayuda a categorizar los mensajes y puede hacer que sea más fácil para los usuarios de Twitter para buscar *tweets*.

Cuando el usuario busca o hace clic en un *hashtag*, puede observar todos los *tweets* que utilizan el mismo *hashtag*. Solo otros que están interesados en el mismo tema probablemente van a utilizar el *hashtag*.

Por ejemplo, si el usuario busca el *hashtag*: #Apple, es menos probable que los *tweets* incluyan referencias a la fruta y más probabilidades de ver información acerca de la empresa de tecnología.

Sin embargo Twitter es una plataforma en tiempo real y su función de búsqueda solo se remonta una semana. Si el usuario desea sacar mayores conversaciones, tiene que utilizar los servicios de terceros, como la aplicación Topsy, que archiva los mensajes enviados a través de redes sociales públicas.

### **1.5.3. ¿Por qué utilizar *hashtag*?**

Twitter es una red social abierta y cualquiera puede ver los *tweets* públicos siempre y cuando no haya configurado la cuenta para ser completamente privada. Pero pocas personas quieren seguir a todos en el mundo. El *hashtag* puede hacer que sea más fácil descubrir otros usuarios de Twitter que están interesados en las mismas conversaciones que se desean.

### **1.5.4. Otras aplicaciones que ofrece Twitter**

Twitter ofrece diversas aplicaciones que permiten desde buscar noticias o eventos, hasta encontrar trabajo, pero también existen aplicaciones en línea basadas en Twitter que amplían las posibilidades y están disponibles para todos los sistemas operativos.

Algunas de las muchas aplicaciones de Twitter son: Search Twitter, TwitPic, Twitter Job Search o Twitmail. Otras aplicaciones conocidas basadas en Twitter son: The Twitter Updater, Twiterrific, Twitterfox o Twitterlicious.

### **1.5.5. Resumen del desarrollo del proyecto de Twitter**

Cuando Jack Dorsey vio la puesta en marcha de la mensajería instantánea, se preguntó si el rendimiento del software del usuario podría ser compartido entre amigos fácilmente. Entonces se reunió con los fundadores de

Odeo (sitio web de agregación y publicación proporcionada por Podcast Inc.), Noah Glass y Evan Williams, quienes además habían sido previos fundadores de la corporación Audioblog y Blogger y Pyra Labs respectivamente.

Jack Dorsey y Biz Stone construyeron un prototipo de Twitter en marzo de 2006 (en tan solo dos semanas) y en agosto de 2006 lanzaron el producto. En octubre de ese mismo año, Odeo se reformó pasando a llamarse Obvious Corp. El servicio adquirió popularidad rápidamente, ganando en marzo de 2007 el premio “South by Southwest Web Award” en la categoría de blog. Obvious derivó en la compañía Twitter Incorporated en mayo de 2007.

En abril de 2008 Twitter anunció en su blog que había creado una versión de Twitter para los usuarios japoneses. El rápido crecimiento de Twitter para muchos radica en su simplicidad. La generación de servicio de mensajes cortos ha visto en este recurso una vía rápida de comunicación a tiempo real. Pero las estadísticas han demostrado que este tipo de fenómenos (Myspace y Facebook) han sufrido un paulatino declive alrededor del año de su auge.

## **1.6. Identificando la teoría que fundamenta esta investigación**

La teoría se puede definir como un conjunto de hipótesis, un campo a lo cual se le aplica, políticas o disciplinas para obtener consecuencias de las hipótesis planteadas, que servirán para realizar predicciones sobre un sistema real. Es importante tener una teoría que soporte la investigación que se va a realizar, porque ayudan a la supervivencia de las tecnologías escogidas para solucionar cierto problema en el campo al cual se va aplicar.

En este trabajo de graduación se propone la Teoría del Modelo de Aceptación Tecnológica o TAM (según sus siglas en inglés Technology

Acceptance Model) que se basa en la teoría de la “Acción Razonada” desarrollada por Martin Fishbein e Icek Ajzen (durante el periodo comprendido de 1975 a 1980), la cual es un modelo general de las relaciones entre actitudes, convicciones, presión social, intenciones y conducta.

La teoría del TAM fue desarrollada por Fred D. Davis en 1986, la cual se ocupa más específicamente con la predicción de la aceptabilidad de un sistema de información; consiste en visualizar la evolución de la tecnología, ayudando en el desarrollo del ser humano en todos los ámbitos que lo rodean, minimizando el tiempo, esfuerzo y costo.

El propósito del modelo es predecir la aceptación de una herramienta e identificar las modificaciones que se tienen que realizar en el sistema, de manera que sea aceptable por los usuarios en general. El modelo sugiere que la aceptación de un sistema de información se debe basar en dos factores, los cuales se explican a continuación.

#### **1.6.1. Utilidad percibida**

Se refiere al grado en que una persona cree, que usando un sistema en particular, mejorará su desempeño en el trabajo.

#### **1.6.2. Facilidad de uso percibida**

Señala hasta qué grado una persona cree, que usando un sistema en particular, realizará menos esfuerzo para desempeñar tareas.

## **1.7. Teoría y la relación con la tecnología escogida**

Según se demuestra en la teoría de la “Acción Razonada”, el Modelo de Aceptación Tecnológica postula que el uso de un sistema de información está determinado por la intención de comportamiento; pero por otro lado, que la intención de comportamiento se determina por la actitud de la persona hacia el uso del sistema y también por su percepción de su utilidad.

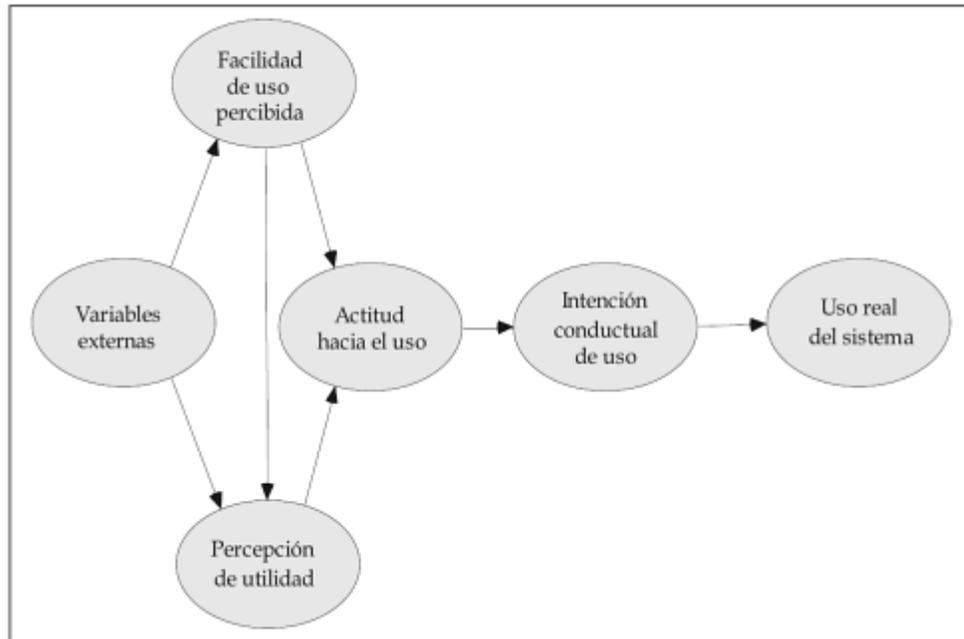
De acuerdo con Fred D. Davis, la actitud de un individuo no es el único factor que determina su uso en el sistema, pero también se basa en el impacto que puedan tener sobre su rendimiento. Además, el Modelo de Aceptación de Tecnología de la hipótesis da un vínculo directo entre la utilidad y facilidad de uso percibida. Con dos sistemas que ofrecen las mismas características, el usuario encontrará más útil un sistema y el más fácil de usar.

Según Fred D. Davis, el propósito del Modelo de Aceptación Tecnológica es explicar las causas de aceptación de las tecnologías por los usuarios, proponiendo así a través de este modelo, que las percepciones de un individuo en la utilidad percibida y la facilidad de uso percibida de un sistema de información, sean concluyentes para determinar su intención para usar un sistema.

## **1.8. Validación del modelo**

El modelo propuesto por este autor se encuentra en la siguiente figura:

Figura 1. **Modelo de Aceptación Tecnológica (TAM)**



Fuente: CARREÓN GUILLÉN, Javier y HERNÁNDEZ VALDÉS, Jorge. Psicología de la globalización neoliberal: análisis de la exclusión social, la brecha digital y el desarrollo sustentable. p. 94.

Según la figura anterior, a continuación se describe cada una de las partes que componen el Modelo de Aceptación Tecnológica.

### **1.8.1. Variables externas**

Son todos los ambientes y circunstancias en la cual se encontrará dicho software, teniendo en cuenta la capacidad de responder a los mismos. La aplicación de software permite que los usuarios la usen como herramienta de trabajo, asimismo garantiza que tengan una alta disponibilidad en las consultas y manejo de datos, así como tener servicios web para tener control del mismo.

### **1.8.2. Facilidad de uso percibida**

Es la fácil interacción que tiene el usuario con la herramienta para realizar las actividades. La aplicación desarrollada posee una interfaz gráfica amigable, a través de la cual el usuario puede interactuar con las opciones de forma fácil, manejando módulos de cada una de las actividades a realizar.

### **1.8.3. Utilidad percibida**

Es la utilidad que se le otorga a dicha aplicación en las diferentes actividades. La aplicación de software está orientada para llevar un control eficiente con los datos, pero también garantiza la difusión de noticias a través de una de las redes sociales más importantes en el mundo.

### **1.8.4. Actitud a través del uso**

Se refiere a la disposición que se tiene al utilizarlo en las tareas diarias. La aplicación de software pretende convencer a cada usuario para que pueda elegir y utilizar dicha herramienta, dándole a conocer las ventajas en la implementación de la misma, pues logrará obtener la información en tiempo real.

### **1.8.5. Intención de uso**

Es el deseo que se tendrá en simplificar el trabajo del usuario con las herramientas indispensables. La aplicación de software tiene como objetivo que el usuario ya no tenga que realizar consultas y operaciones tediosas que le consuman tiempo, sino más bien que realice todo de una forma fácil y práctica.

### **1.8.6. Uso actual del sistema**

Es cuando la aplicación de software es capaz de ejecutar los análisis y tomar una decisión basándose en patrones establecidos, originados por la interacción del usuario y software. La aplicación de software permite realizar un proceso común relacionado a los problemas que se presentan en las actividades diarias y/o por temporadas dentro del campo educativo.

El modelo TAM es dependiente de todos los factores (variables) externos y pueden ser muy diversos; como por ejemplo: aspectos culturales, ya que se pretende demostrar que juegan un papel importante en la aceptación de las TIC. Además, suele utilizarse versiones más simplificadas del TAM, las cuales examinan directamente los efectos de los factores (variables) externos en la utilidad percibida y la facilidad de uso percibida de las TIC.



## **2. IDENTIFICACIÓN DEL PROBLEMA Y SOLUCIÓN QUE LA APLICACIÓN BRINDA**

En el área de dispositivos móviles se tiene un marco histórico de documentos vinculados con el tema, lo cual permite ubicar y describir el desarrollo actual de una aplicación con el fin de la modernización de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala.

### **2.1. Antecedentes**

Existen otros trabajos de graduación desarrollados en la facultad en algunos años de la última década, los cuales se han realizado para la utilización de los móviles como herramientas apropiadas para apoyar a los estudiantes.

De estos trabajos de graduación, se considera importante describir brevemente el objetivo que tiene cada uno de ellos y también una pequeña descripción de los mismos, usando desde el más reciente al más antiguo, encontrados en el catálogo de tesis que cuenta la biblioteca de la Universidad de San Carlos. A continuación se describen los mismos.

“Desarrollo de una herramienta móvil de software para apoyar a estudiantes universitarios en su organización”, trabajo de graduación realizado por Rolando Iván Chali Monroy, marzo de 2012. Dicho trabajo de graduación estuvo desarrollado sobre la plataforma Java 2 Micro Edition, realizada para ejecutarse sobre dispositivos móviles con pocos recursos de hardware.

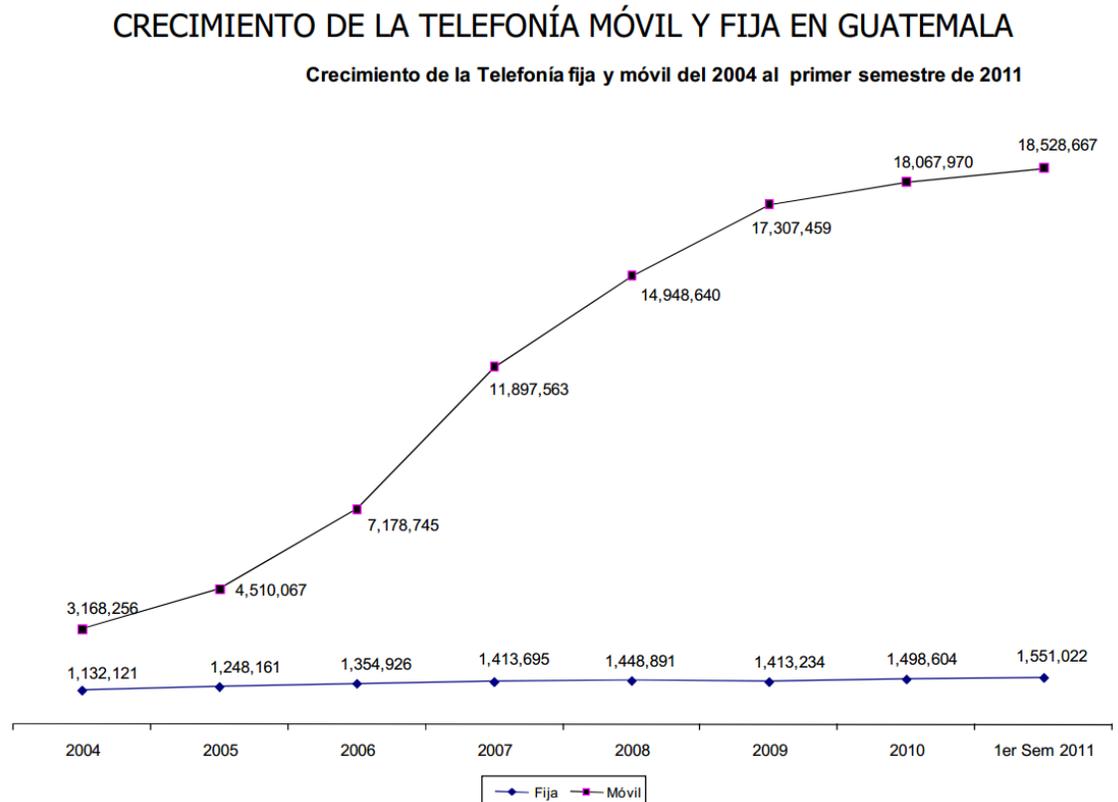
La aplicación se encargaba de llevar el control de los horarios de cursos, registro de tareas, apuntes importantes y como extra, un módulo llamado Time Boxing, el cual se encargaba de establecer los tiempos específicos a dedicar a una actividad; este último suele ser un problema para los estudiantes a la hora de organizar los tiempos de estudio.

“Programación de dispositivos móviles”, trabajo de graduación realizado por Félix Javier Damián Marroquín, septiembre de 2006. Dicho trabajo de graduación tiene como objetivo explicar los diferentes entornos de desarrollo que en ese año se estaban desarrollando, incluyendo tecnologías como Mobile Development Resource Kit, Mobile Internet Toolkit, PalmOS Emulator, J2ME, Satellite Forms, además de establecer los lineamientos que se deben tomar en cuenta en el desarrollo de una aplicación móvil; como por ejemplo: los criterios de seguridad, riesgos, escalabilidad, funcionalidad, administración del software y otros.

Aun cuando estos trabajos de graduación abarcaron la tecnología de los dispositivos móviles y el primero de los investigados introdujo una herramienta de innovación, las tendencias de las tecnologías eran cambiantes y para hoy en día existen nuevos avances de uso de telefonía móvil, especialmente los teléfonos inteligentes Android.

Existe un crecimiento grande en cuanto a los dispositivos móviles según las estadísticas que proporciona la Superintendencia en Telecomunicaciones (SIT) de Guatemala. Estas estadísticas detallan el crecimiento de las redes móviles y líneas fijas, siendo la parte del crecimiento móvil de mayor interés. En la siguiente figura se detalla el crecimiento de los teléfonos móviles y fijos en Guatemala.

Figura 2. **Crecimiento de la telefonía móvil y fija en Guatemala**



Fuente: Superintendencia en Telecomunicaciones. Crecimiento de la telefonía fija y móvil en Guatemala. [http://www.sit.gob.gt/oldSite/uploads/docs/stats/ctfm/CreTelefonia\\_1sem11.pdf](http://www.sit.gob.gt/oldSite/uploads/docs/stats/ctfm/CreTelefonia_1sem11.pdf).

Consulta: noviembre de 2012.

La figura anterior permite establecer el crecimiento de adquisición de teléfonos móviles en el país, pues dicha figura muestra que para el primer semestre del 2011 se ha dado un crecimiento rápido en relación al 2004, lo que evidencia que en Guatemala ha aumentado el uso de teléfonos móviles y asimismo la utilización de aplicaciones para móviles.

Aparte de las estadísticas anteriormente descritas, considerar el crecimiento de los teléfonos inteligentes en las ventas de los teléfonos móviles es muy importante, ya que los móviles están creciendo en Guatemala. Según estudios realizados en 2012 por Anson Alexander, estadounidense graduado de la Universidad de Tampa, realizó estadísticas sobre los teléfonos inteligentes en el mundo, describiéndose un resumen a continuación:

- El 80% de las personas tiene su propio móvil
- Existen alrededor de 5 billones de teléfonos móviles en el mundo, de los cuales 1,08 billones son teléfonos inteligentes.
- Android tiene el mayor mercado con un 46,9%, mucho más que iPhone (con un promedio de 28,7%).
- La actividad más popular en un teléfono inteligente es redactar texto, seguido por navegación por Internet y juegos.
- Existen 9 de 10 personas que tienen un teléfono inteligente para utilizarlo en las actividades diarias.

## **2.2. Descripción del problema**

La población estudiantil de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, reconoce y valora que el contexto actual de cambios y avances tecnológicos demanda que todas las instituciones estén en constante renovación y actualización; sin embargo, durante el periodo comprendido de 2007 a la fecha, se nota cierto problema en la difusión de noticias en los cursos de cada semestre, siendo como posibles causas la falta de acceso al medio informativo, factor tiempo o cualquier otro inconveniente.

La facultad ya cuenta con un medio en el cual los estudiantes pueden visualizar el horario. Está diseñado específicamente para ser visualizado por

computadoras de escritorio o portátiles mediante un navegador web; sin embargo, cuando se trata de acceder a la información mediante un dispositivo móvil, se afrontan varios inconvenientes que no permiten su uso, ya que este tipo de páginas no están diseñadas para ser visualizadas por móviles.

Se considera dos factores en la actualidad, primero es el crecimiento en el uso de los dispositivos móviles en el mundo, en este caso el país de Guatemala, siendo uno de los países a nivel latinoamericano con mejor crecimiento en cuanto al consumo de dispositivos móviles y segundo, la creciente aceptación que están teniendo los dispositivos móviles con sistema operativo Android, que por dicho motivo cada día se está desarrollando más y más aplicaciones; se toma la decisión de crear un sistema aplicativo para dispositivo móvil que proporcione una base funcional para que en un futuro, algún estudiante, ingeniero (o interesado) pueda utilizarlo, ampliarlo, mejorarlo y adaptarlo ante esta y otras necesidades existentes en la facultad.

Los estudiantes formados en la carrera de Ingeniería en Ciencias y Sistemas, tienen la convicción que la modernización de la Facultad de Ingeniería es imprescindible y más que todo por la imagen pública que representa al país de Guatemala.

Por lo tanto, la modernización se debe implementar de manera permanente aprovechando el potencial de cada uno de los estudiantes que se encuentran dentro de la facultad al proponer proyectos de innovación tecnológica.

### **2.3. Desarrollo del software**

En este trabajo de graduación se desarrolla una aplicación de software para dispositivos móviles con sistema operativo Android, por medio de la cual se brinda el manual técnico y manual de usuario, así como una aplicación base que permite la difusión de noticias en los cursos asignados durante un semestre específico (haciendo uso de la red social Twitter), de tal forma que los estudiantes, ingenieros o personas interesadas en el tema, puedan utilizarla y de ser posible mejorarla para fortalecerla y satisfacer las necesidades que la Facultad de Ingeniería presenta en estos momentos.

Esta aplicación se caracteriza ante otras posibles aplicaciones ya que posee un perfil de usuario, así como su interacción con los cursos y noticias correspondientes. Por medio del perfil el estudiante accede a los cursos que se asigna personalmente al principio de cada semestre, de manera que no es una visualización web, sino que la aplicación se encarga de tener el perfil con información del estudiante.

Esta aplicación, aprovechando la innovación de la tecnología y el auge de las redes sociales, permite que el estudiante vea las noticias asociadas a su perfil de cuenta en la red social Twitter, asimismo puede crear cualquier noticia relacionada a los cursos que se encuentra asignado y se entera de los avances de los mismos.

En términos generales, la aplicación está diseñada para que en un futuro se le agreguen funcionalidades, como por ejemplo: asistencia vía móvil, asignación oficial de cursos, utilización de mapas de Google para localización satelital a través de GPS, lector de QR Codes, etc.; incorporando así varias tecnologías en una sola aplicación para modernizar el sistema interno de la

facultad, con un alcance y enfoque a nivel de universidad y que todas las facultades del Campus Central cuenten con este tipo de herramienta para que se puedan beneficiar con una administración personal.

#### **2.4. Mercado objetivo**

El mercado objetivo lo forma cada uno de los estudiantes de la Facultad de Ingeniería de la Universidad de San Carlos, en plazo corto, lo cual representa un total aproximado de doce mil estudiantes, siendo así una de las más numerosas dentro de la Universidad de San Carlos que actualmente cuenta con una población mayor de cien mil estudiantes.

Por lo indicado anteriormente, los estudiantes son los únicos que mediante la popularidad de las herramientas lograrán que la penetración de cualquier aplicación nueva sea utilizada dentro de toda la universidad. El estudiante de la Facultad de Ingeniería será el mayor beneficiado de este proyecto, únicamente necesitará contar con un teléfono inteligente, conocido comúnmente como *smartphone*, el cual incluya un sistema operativo Android para tener uso de la aplicación ya que no tiene ningún costo para todo aquel estudiante de la Facultad de Ingeniería.

#### **2.5. BenchMark de la aplicación**

En este apartado se tratan los productos o servicios, realizando la comparativa de otras aplicaciones móviles respecto a la aplicación desarrollada. Existen actualmente muchas aplicaciones móviles Android, las cuales ayudan de cierta manera a realizar las tareas de las personas. Las aplicaciones comúnmente ayudan a realizar las tareas rutinarias: blog de notas, calendario, calculadoras, etc.; en este caso, la aplicación desarrollada beneficia a los

estudiantes de la Facultad de Ingeniería permitiendo realizar operaciones relacionadas con la organización de los cursos asignados al inicio de cada semestre.

Por lo tanto, a continuación se describe brevemente alguna de las aplicaciones mencionadas anteriormente. Es necesario notar que las aplicaciones no son iguales, dado que la aplicación móvil desarrollada es para un caso de estudio específico, que llega a ser la solución al problema planteado de la Facultad de Ingeniería de la Universidad de San Carlos.

### **2.5.1. Blackboard Mobile™**

Aplicación que permite utilizar la mayor parte del contenido básico disponible a través de los cursos, así como las mismas organizaciones, de una manera atractiva e intuitiva en dispositivos móviles. Permite que los usuarios realicen las siguientes actividades:

- Recibir notificaciones *push* de actividad del curso
- Usar pruebas compatibles con el móvil
- Ver calificaciones en cualquier momento
- Visualizar los documentos en múltiples formatos
- Guardar archivos en otras aplicaciones para verlas sin conexión
- Ver y publicar en foros de discusión
- Comentar en blogs y revistas
- Consultar la lista de cursos
- Listar los cursos a través de filtros o por favoritos
- Cargar los medios de comunicación como archivos adjuntos a los foros de discusión y blogs.
- Publicar anuncios (instructores)

### **2.5.2. Schedule Timetable Deluxe**

La aplicación está diseñada para alumnos que quieren tener su vida bajo control escolar, estudiantes que quieren monitorear actividades o información relacionada con los estudios o simplemente padres que desean tener acceso al programa de estudios de los hijos. Además, desde el propio programa también se permite gestionar las tareas, asignaciones y los grados de manera fácil y sencilla.

A continuación se explica brevemente las cinco razones principales para este calendario:

- Función principal: calendario
  - Los colores para los sujetos pueden ser elegidos libremente
  - Vista de día clásica, semanal o los sujetos próximos
  - Varios planes de lecciones creados
- Entrada muy cómoda (sin limitaciones)
  - Reconocimiento de voz y el selector de fecha
  - Edición, borrado y copiado de entradas
- Tareas con prioridad y fecha de vencimiento
- Administración de los exámenes y las calificaciones
  - Creación de exámenes
  - Aplicación de exámenes incluyendo grados de ponderación
- Función de copia de seguridad



### **3. DOCUMENTACIÓN BASE PARA LA PROGRAMACIÓN DE LA APLICACIÓN MÓVIL**

Este capítulo se organiza con la finalidad de explicar los elementos básicos del proceso de instalación, configuración, desarrollo (programación) y evaluación de la aplicación creada. Así también se explican varios conceptos y temas importantes para poder elaborar aplicaciones móviles Android.

Es necesario aclarar que para programar en Android se requiere tener conocimientos de lenguaje Java, Programación Orientada a Objetos (POO) y arquitectura “Modelo, Vista y Controlador” (MVC). Si se desconoce el lenguaje Java, pero se sabe programar, se reconoce el lenguaje Java como un lenguaje sencillo y rápido de aprender.

Además de Java, se utiliza XML para la parte visual, otro lenguaje descriptivo muy fácil de utilizar y que aporta sencillez a la hora de programar en Android.

#### **3.1. Requerimientos preliminares para el desarrollo de la aplicación**

Es necesario cumplir los requerimientos preliminares para que la aplicación funcione correctamente. A continuación se detallan estos requerimientos.

### **3.1.1. Android SDK**

Android Software Development Kit (SDK) contiene las herramientas necesarias para crear, compilar y empaquetar aplicaciones Android. La mayoría de estas herramientas están basadas en línea de comandos. El SDK de Android también proporciona un emulador de dispositivos Android, por lo cual las aplicaciones se pueden probar sin un verdadero teléfono Android. Se puede crear Dispositivos Virtuales Android a través del SDK de Android, que corren en dicho emulador. El SDK de Android contiene la depuración Android Bridge, herramienta que permite conectarse a un dispositivo Android virtual o real.

Google proporciona las herramientas de desarrollo de Android para desarrollar aplicaciones Android con Eclipse. Las herramientas de desarrollo de Android son un conjunto de componentes que se extienden en el entorno de desarrollo integrado: Eclipse, con capacidades de desarrollo de Android. Estas herramientas contienen todas las funciones necesarias para crear, compilar, depurar y desplegar aplicaciones de Android desde Eclipse.

Otra característica de estas herramientas es que proporciona ayuda a los editores especializados para los archivos de recursos, por ejemplo ficheros de diseño. Estos editores permiten conmutar entre la representación XML del archivo y una interfaz de usuario más rica a través de pestañas en la parte inferior del editor.

### **3.1.2. Dalvik Virtual Machine**

El sistema Android utiliza una máquina virtual especial, es decir, la máquina virtual Dalvik para ejecutar aplicaciones basadas en Java. Dalvik utiliza un formato de código binario que es diferente del código binario del lenguaje

Java. Por lo tanto, no se puede ejecutar directamente archivos de clase Java en Android, necesitan convertirse en el formato “Bytecode” de Dalvik.

### **3.2. ¿Cómo desarrollar aplicaciones Android?**

Las aplicaciones de Android están principalmente escritas en el lenguaje de programación Java. Los archivos fuente de Java se convierten en archivos de clases de Java por el compilador Java. El SDK de Android incluye una herramienta que convierte los archivos de clase Java en un archivo ejecutable Dalvik. Todos los archivos de clase de una aplicación se colocan en un archivo comprimido.

Durante este proceso de conversión de información redundante, los archivos de clase están optimizados en dicho archivo. Por ejemplo, si la misma cadena se encuentra en los archivos de clases diferentes, este archivo contiene solo una referencia, una vez por cada cadena. Estos archivos son por lo tanto mucho más pequeños en tamaño que los archivos de clases correspondientes.

El archivo ejecutable y los recursos de un proyecto Android, por ejemplo, las imágenes y los archivos XML, se colocan en un paquete de Android. El programa AAPT (Asset Android Packaging Tool) realiza esta encapsulación. El resultado del archivo empaquetado contiene todos los datos necesarios para ejecutar la aplicación para Android y se puede implementar en un dispositivo Android.

Las herramientas de desarrollo de Android realizan los pasos de forma transparente para el usuario. Al momento que el usuario pulsa un botón y usa la herramienta, el conjunto de aplicaciones de Android será creado y desplegado.

### **3.3. Recursos necesarios**

Para el cumplimiento del proyecto es necesario que el interesado establezca el ambiente de desarrollo y el ambiente real, así como otros componentes específicos que se necesitaran para llevarlo a cabo, dentro de los cuales es importante mencionar:

#### **3.3.1. Ambiente de desarrollo**

A continuación se describen los recursos para el ambiente de desarrollo:

- Contar con computadoras que tengan instalado el SDK Android para el desarrollo de la aplicación.
- Un entorno de desarrollo integrado; en este caso se puede mencionar Eclipse, siendo el entorno con mayor popularidad para el desarrollo de aplicaciones Android.
- JDK como lenguaje nativo para emular el SDK de Android y ejecutar Eclipse como entorno de desarrollo.
- Requerimientos de hardware: una máquina moderna bien equipada con al menos 1GB de RAM o 2GB RAM, 2 núcleos de procesador, para un mejor funcionamiento.
- Servidor accesible vía Internet; accesible desde los dispositivos móviles para la solicitud de información asociada a los perfiles de los usuarios, cursos asignados e información de la red social.
- Procesador de 1GHz, el cual se puede mejorar si la difusión de la aplicación tiene el uso e impacto deseado.

### **3.3.2. Ambiente real**

A continuación se describen los recursos para el ambiente real:

- Dispositivos móviles con sistema operativo Android para la instalación y ejecución de la aplicación dado que la aplicación funciona correctamente en celulares.
- Servidor accesible vía Internet, donde se aloja la aplicación para su descarga y posterior instalación de las aplicaciones.
- Proveedor de la base de datos de la información que se considere necesaria para la ejecución de la aplicación, que no se pueda guardar en los dispositivos de los usuarios finales.

### **3.4. Componentes de ambos ambientes**

A continuación se describen los recursos necesarios para ambos ambientes (desarrollo y real):

- Cuenta de Twitter personal única para la utilización de la aplicación
- Creación de *hashtag* para ser utilizados; un *hashtag* por cada curso
- Servidor web para autenticación de usuarios

### **3.5. Conceptos generales**

A continuación se presenta una breve descripción de los componentes de la interfaz de usuario más importantes en Android.

### **3.5.1. Actividad**

Una actividad representa la representación visual de una aplicación Android. Las actividades utilizan vistas y fragmentos para crear la interfaz de usuario y la interacción con el usuario. Una aplicación Android puede tener varias actividades.

### **3.5.2. Fragmentos**

Los fragmentos son componentes que se ejecutan en el contexto de una actividad. Estos componentes encapsulan código de la aplicación para que sea más fácil de reutilizar y para soportar diferentes dispositivos de tamaño.

Los fragmentos son opcionales, pueden utilizar vistas y grupos de vistas directamente en una actividad, pero en aplicaciones profesionales siempre los utilizan para permitir la reutilización de los componentes de interfaz de usuario sobre diferentes dispositivos del tamaño.

### **3.5.3. Puntos de vista y grupos de vistas**

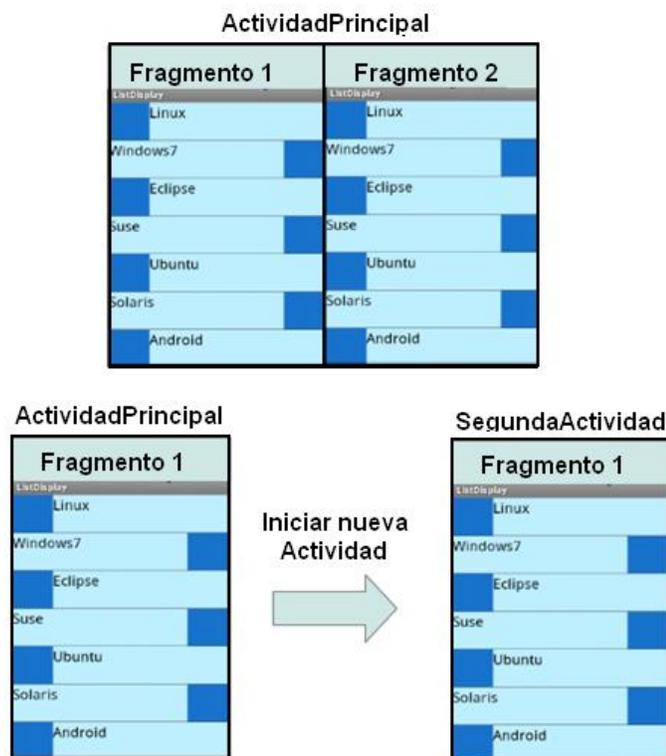
La clase base para todas las reproducciones es la “android.view.View”. Las vistas tienen atributos que se pueden utilizar para configurar su apariencia y comportamiento. Un grupo de vistas es responsable de organizar otras vistas. A los grupos de vistas también se les conoce como controladores de distribución. La clase base para los controladores de distribución es la “android.view.ViewGroup” la cual extiende la vista de clases. Los grupos de vistas se encuentran para crear diseños complejos. No es recomendable tener demasiados grupos de vistas ya que esto tiene un impacto negativo en el rendimiento.

### 3.5.4. Diferencias entre actividades, fragmentos y vistas

El sistema Android utiliza una máquina virtual especial. Las actividades se definen con diferentes diseños. Estos diseños pueden ser recogidos sobre la base de diferentes factores, incluyendo el tamaño del dispositivo real.

La siguiente figura muestra una actividad llamada "ActividadPrincipal". En una pantalla panorámica se muestran dos fragmentos. En una pantalla más pequeña se muestra un fragmento y permite que el usuario navegue a otra actividad llamada "SegundaActividad", la cual muestra el segundo fragmento.

Figura 3. Fragmentos en pequeñas y grandes pantallas



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

### **3.5.5. Otros componentes de Android**

Android tiene varios componentes más que se pueden utilizar en su aplicación. A continuación se describen los componentes.

#### **3.5.5.1. Intenciones**

Las Intenciones son mensajes asíncronos que permiten a la aplicación solicitar la funcionalidad de otros componentes del sistema Android, por ejemplo de servicios o actividades. Una aplicación puede llamar a un componente directamente (intención explícita) o pedir al sistema Android para evaluar componentes registrados en base a las intenciones de datos (intención implícita).

Por ejemplo, la aplicación podría implementar el intercambio de datos a través de una intención y todos los componentes que permiten el intercambio de datos estarían disponibles para que el usuario lo seleccione. Las solicitudes se registran a una intención a través de un filtro de intenciones.

Las intenciones permiten combinar componentes débilmente acoplados para realizar ciertas tareas.

#### **3.5.5.2. Servicios**

Los servicios se encargan de realizar las tareas de fondo sin proporcionar una interfaz de usuario. Se puede notificar al usuario a través del marco de notificación de Android.

### **3.5.5.3. Proveedor de contenidos**

Proporciona una interfaz estructurada a los datos de aplicación. A través de ellos, la aplicación puede compartir datos con otras aplicaciones. Android contiene una base de datos (SQLite) que se usa frecuentemente en combinación con un proveedor de contenidos. En la base de datos SQLite se almacenan los datos, los cuales se acceden a través del proveedor de contenidos.

### **3.5.5.4. Receptor de transmisión**

Permite registrar información para recibir mensajes del sistema. Un receptor de transmisión recibe dicha transmisión por el sistema Android si la situación específica sucede. Por ejemplo, un receptor de transmisión podría ser llamado una vez que el sistema Android haya completado el proceso de arranque o si una llamada ha sido recibida.

### **3.5.5.5. Pantalla de inicio**

Es un componente interactivo que se utiliza principalmente en Android. Estas pantallas de inicio presentan típicamente algún tipo de dato que permite al usuario realizar acciones a través de ellos. Por ejemplo, una pantalla de inicio puede mostrar un breve resumen de los correos electrónicos nuevos y si el usuario selecciona una dirección de correo electrónico, puede iniciar la aplicación con el correo electrónico seleccionado.

### **3.5.5.6. Más componentes**

Android proporciona muchos más componentes, pero los descritos anteriormente son los más importantes. Otros componentes son: Android Live Folders y Android Live Wallpapers. Los Android Live Folders sirven para mostrar datos en la pantalla de inicio sin ejecutar la aplicación correspondiente, mientras que los Android Live Wallpapers permiten crear fondos animados.

### **3.5.6. Aspectos específicos en aplicaciones Android**

Al desarrollar aplicaciones Android se debe considerar varios aspectos específicos para que la configuración en todo dispositivo sea acorde a la necesidad que va solventar la aplicación. A continuación se describen estas especificaciones.

#### **3.5.6.1. AndroidManifest.xml**

Los componentes y configuraciones de una aplicación Android se describen en el archivo: AndroidManifest.xml, el cual es el archivo principal de configuración de la aplicación. Por ejemplo, todas las actividades y servicios de la solicitud deben ser declaradas en este archivo. También debe contener los permisos necesarios para la aplicación. Por ejemplo, si la aplicación requiere acceso a la red se debe especificar en dicho archivo. A continuación se muestra una figura que ejemplifica la configuración dentro del archivo como tal.

Figura 4. **Ejemplo de archivo AndroidManifest.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="de.vogella.android.temperature" android:versionCode="1"
    android:versionName="1.0">

    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".Convert" android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

    <uses-sdk android:minSdkVersion="9" />

</manifest>
```

Fuente: elaboración propia a partir de la aplicación Android desarrollada.

En la figura anterior, el atributo del paquete define la base para los objetos Java. Si un objeto de Java se encuentra dentro de un paquete diferente, debe ser declarado con el nombre del paquete completo calificado.

Google Play requiere que todas las aplicaciones de Android utilicen su paquete único. Por lo tanto, es una buena recomendación usar el nombre de dominio inverso al nombre del paquete. Eso evitará colisiones con otras aplicaciones de Android. El atributo de nombre de versión y versión de código especifican el nombre y la versión de la aplicación respectivamente. El nombre

de versión es lo que el usuario visualiza. Este nombre puede estar compuesto por cualquier cadena de texto; la versión de código debe ser un número entero.

El Android Market determina la base de la versión de código si en dado caso se debe realizar una actualización de las aplicaciones para las instalaciones existentes. Por lo general, comienzan con "1" y aumenta este valor en uno, si en dado caso se despliega una nueva versión de la aplicación.

El parámetro de actividad la define como tal, en este caso apunta a la conversión de clase en el paquete: `de.vogella.android.temperature`. Un filtro de intención está registrado para esta clase, definiendo que la actividad se inicie una vez que la acción de la aplicación arranca.

El uso del archivo de recursos hace que sea fácil de proporcionar diferentes recursos, cadenas, colores, iconos, para los diferentes dispositivos y hace fácil la traducción de las aplicaciones.

La etiqueta `Uses-sdk`, que es parte del archivo `AndroidManifest.xml`, define la mínima versión del SDK para que la aplicación sea válida. Eso evitará que la aplicación se instale en los dispositivos incompatibles.

### **3.5.6.2. Actividades y ciclo de vida**

El sistema Android controla el ciclo de vida de la aplicación. En cualquier momento el sistema Android puede detener o destruir la aplicación; por ejemplo, destruir la aplicación a causa de una llamada entrante.

El sistema Android define un ciclo de vida de las actividades a través de métodos predefinidos. Los métodos más importantes son los descritos a continuación.

- `onSaveInstanceState()` – es dedicado a la actividad cuando se detiene. Se utiliza para guardar datos de forma que la actividad puede restaurar los estados si se reinició.
- `onPause()` – es llamado siempre si la actividad termina, puede ser utilizado para liberar los recursos o guardar los datos
- `onResume()` – si la llamada actividad se reinicia, se puede utilizar para inicializar los campos.

### **3.5.6.3. Contexto**

Esta clase proporciona la conexión con el sistema Android y los recursos del proyecto. Es la interfaz a la información global sobre el entorno de aplicación. El contexto también proporciona acceso a los servicios de Android, por ejemplo, el servicio de localización. Las actividades y servicios de Android amplían el contexto de la clase.

## **3.6. Proceso de Instalación**

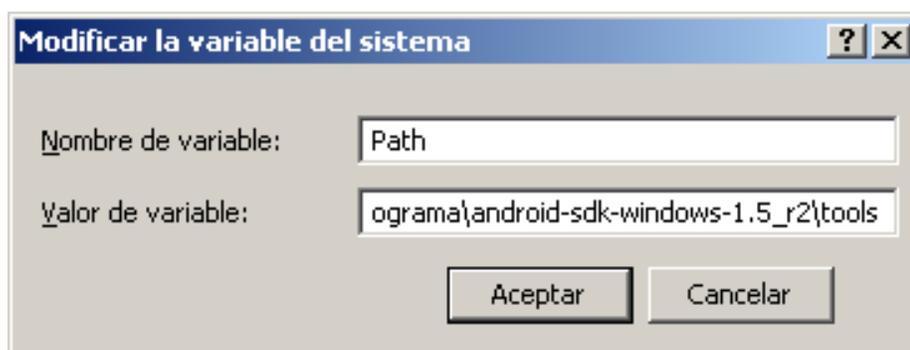
Para asegurarse de una instalación exitosa, el proceso de instalación básicamente se lleva a cabo con cuatro pasos. A continuación se describen brevemente estos pasos, tomando en cuenta que los tres primeros pasos son de configuración y el último paso es de prueba para validar que los primeros tres pasos se hayan realizado satisfactoriamente.

### 3.6.1. Pasos para descargar e instalar SDK para Android

A continuación se describen los pasos para descargar e instalar el archivo SDK para desarrollar aplicaciones Android, utilizando un sistema operativo Windows 7:

- Descargar el archivo de SDK desde la página principal de Android.
- Descomprimir el archivo descargado en el lugar donde se desee tener instalado el SDK, por ejemplo, en los archivos de programa del computador. Este proceso crea un directorio que incluye la versión y liberación que se ha descargado desde la página, como por ejemplo: android-sdk-windows-1.5\_r2.
- Añadir el directorio “/tools” (del archivo SDK descargado) a la variable de entorno de direcciones o rutas de Windows. En la siguiente figura se muestra un ejemplo.

Figura 5. **Modificación de variables del sistema operativo**



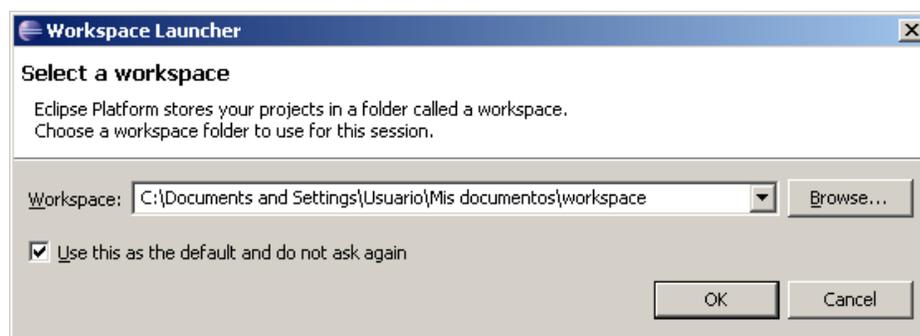
Fuente: elaboración propia a partir de la aplicación desarrollada y sistema operativo Windows 7.

### 3.6.2. Pasos para descargar e instalar Eclipse

A continuación se describe el proceso para descargar e instalar el entorno de desarrollo integrado (Eclipse) que va servir para desarrollar aplicaciones Android, utilizando un sistema operativo Windows 7:

- Visitar la página web de descargas de Eclipse y descargar la última versión. Se recomienda descargar la llamada: Eclipse IDE for Java EE Developers, que tiene un tamaño aproximado de 165 MB.
- Descomprimir el archivo de instalación en el lugar donde se desea instalar.
- Entrar en el directorio “/eclipse” y ejecutar el archivo “eclipse.exe” para lanzar la aplicación. La primera vez que se arranque, pide que se indique un directorio para colocar el espacio de trabajo.
- Seleccionar la ruta para el espacio de trabajo, que es donde se guardan los proyectos.
- Marcar la opción “Use this as the default...” para que no vuelva a preguntar más y presionar el botón OK.

Figura 6. Ruta para el área de trabajo de las aplicaciones



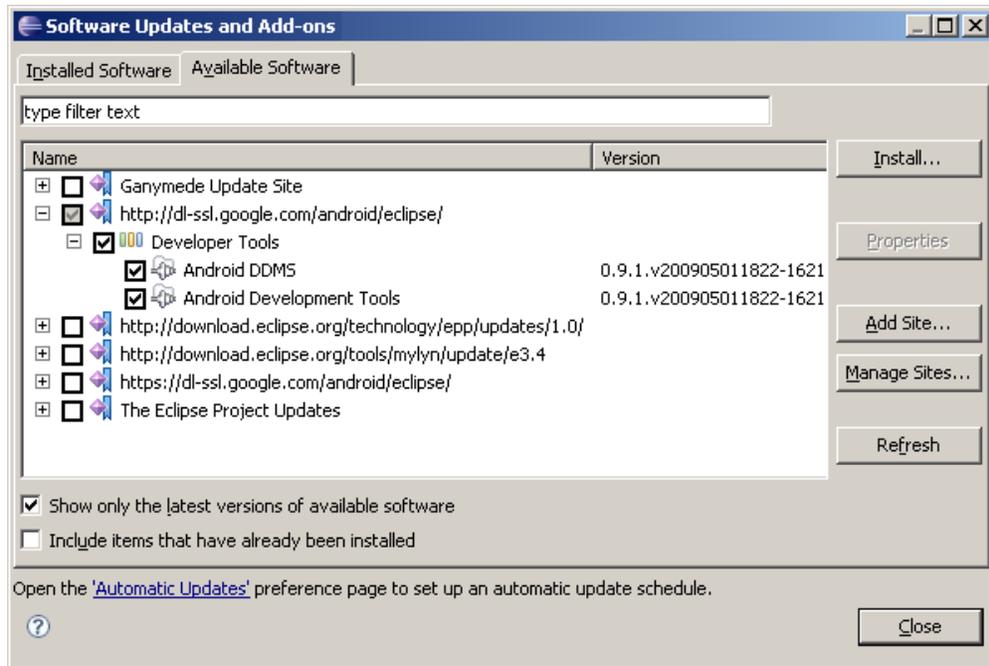
Fuente: elaboración propia a partir de la aplicación desarrollada y programa Eclipse.

### 3.6.3. Pasos para instalar el ADT

A continuación se describe el proceso para descargar e instalar el Android Development Tools (ADT) que va servir para desarrollar aplicaciones Android en Eclipse, utilizando un sistema operativo Windows 7:

- En la carpeta Eclipse, seleccionar actualización de software.
- Hacer clic en la pestaña de software disponible, en la parte superior.
- En la parte derecha, hacer clic en agregar sitio.
- En la ventana que aparece, introducir alguna de las dos siguientes direcciones:
  - <https://dl-ssl.google.com/android/eclipse/>
  - <http://dl-ssl.google.com/android/eclipse/>
- La URL que se introduce en el paso anterior, aparece nuevamente en la lista de software disponible.
- Seleccionar las opciones: “Android DDMS y Android Developers Tools”.
- Presionar el botón “Install...” para instalar las opciones marcadas anteriormente.
- En el proceso de instalación solicita que se acepte la licencia de uso de la herramienta.

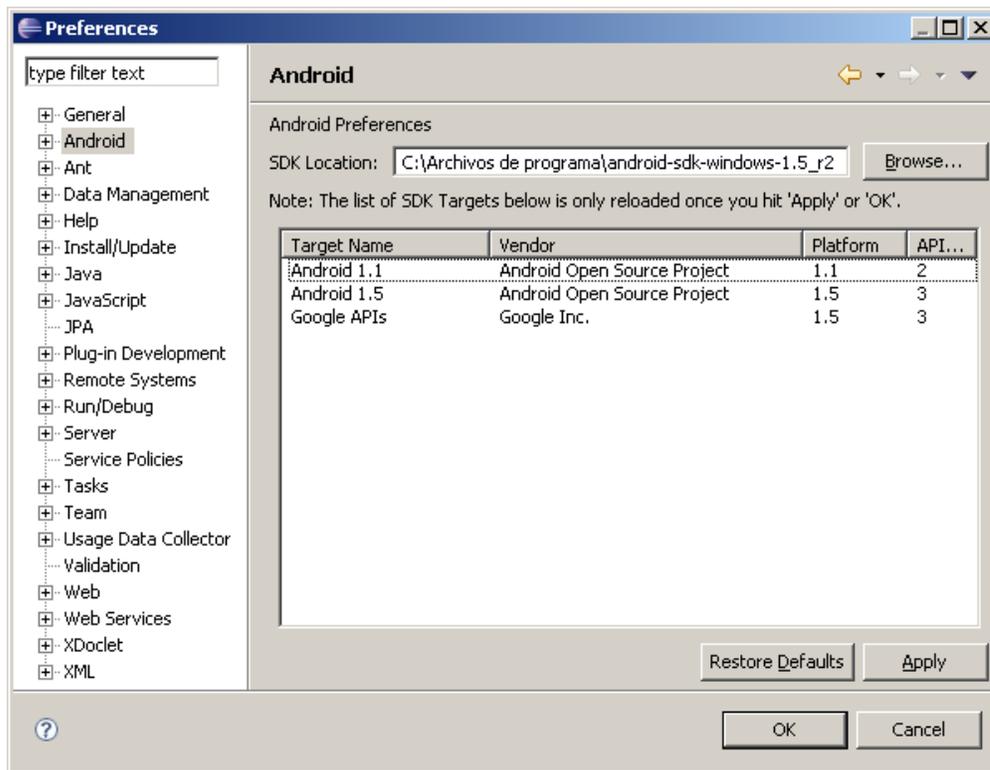
Figura 7. Actualización del software



Fuente: elaboración propia a partir de la aplicación desarrollada y programa Eclipse.

- Una vez finalizada la instalación, se reinicia Eclipse.
- Una vez reiniciado, se debe configurar Eclipse para que acceda al SDK de Android ya instalado. Para eso hacer lo siguiente:
  - Abrir las preferencias de la herramienta.
  - En el apartado Android, en el campo de directorio del SDK se debe introducir la ruta donde se ha descomprimido el SDK de Android.

Figura 8. Definiendo la ruta del SDK



Fuente: elaboración propia a partir de la aplicación desarrollada y programa Eclipse.

- Por último presionar el botón de OK para cerrar la ventana de preferencias.
- Con esto ya se tiene el SDK de Android integrado en Eclipse para empezar a programar.

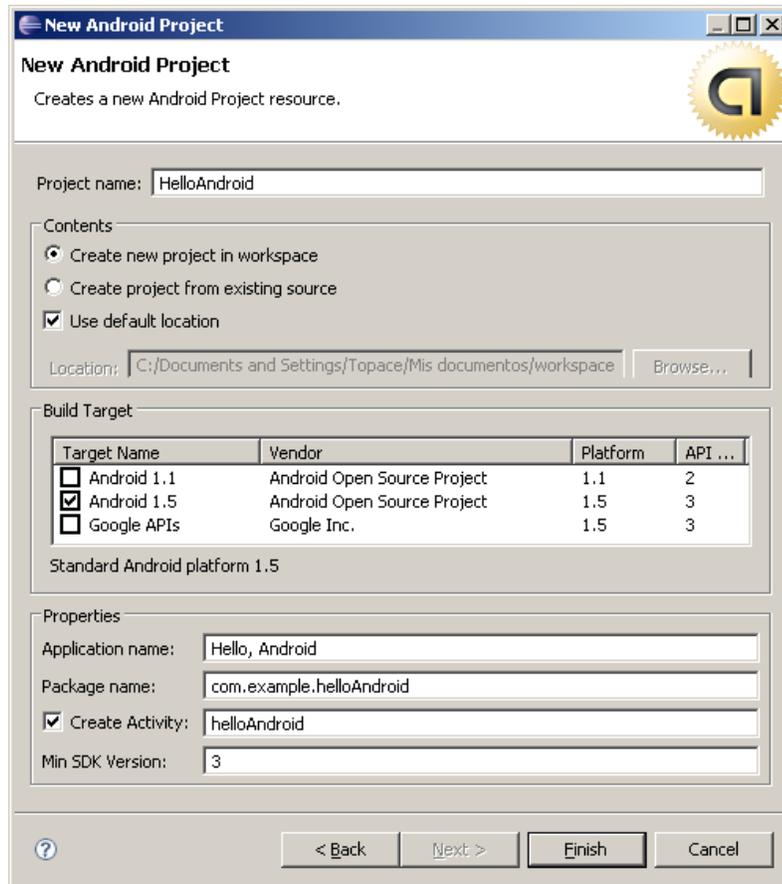
#### 3.6.4. Ejecutar una aplicación Android de prueba

Cuando se tiene el entorno de desarrollo listo, se puede iniciar a crear proyectos; para eso se necesita crear un dispositivo virtual Android o AVD (Android Virtual Device) en el que se realizan las pruebas de las aplicaciones

que se desarrollen. A continuación una serie de pasos para crear una aplicación de ejemplo.

- En Eclipse, abrir el administrador AVD.
- Asignar el nombre que se le da al dispositivo virtual; para eso, se selecciona la versión de Android (en este caso la versión 1.5) y se asigna el tamaño de la tarjeta de memoria SD que tendrá (en este caso la memoria recomendada es de 128 Mb).
- Aceptar la creación y después finalizar la misma.
- Crear un nuevo proyecto en Android a través del menú correspondiente. Para eso se debe ingresar los siguientes datos para el proyecto:
  - Nombre del proyecto
  - Nombre de aplicación
  - Nombre del paquete
  - Nombre de la actividad para crear
  - Memoria mínima de SDK
  - Versión

Figura 9. Crear un nuevo proyecto en Android



Fuente: elaboración propia a partir de la aplicación Android desarrollada y programa Eclipse.

- Después de ingresar los datos del proyecto, se puede finalizar la creación del proyecto. Para eso se selecciona el botón “Finish”.
- A continuación, se ejecuta el proyecto usando la opción de “Ejecución de proyecto” que se encuentra en la parte superior del menú principal de Eclipse.
- Esperar que abra el emulador con la aplicación que se ha creado. En este caso se logra visualizar el famoso “Hola mundo”.

## **4. DESCRIPCIÓN DEL DESARROLLO DE LA APLICACIÓN COMO SOLUCIÓN AL PROBLEMA DE ESTUDIO**

En este capítulo se explica el diseño y desarrollo de la aplicación, permitiendo que se pueda dar continuidad a través de otra persona interesada. Sin embargo, es importante aclarar que no se ofrece una explicación detallada del código, pues la finalidad de este capítulo es explicar su funcionamiento, teniendo en cuenta que se debe tener habilidades de programación para poder iniciar con la lectura de este material.

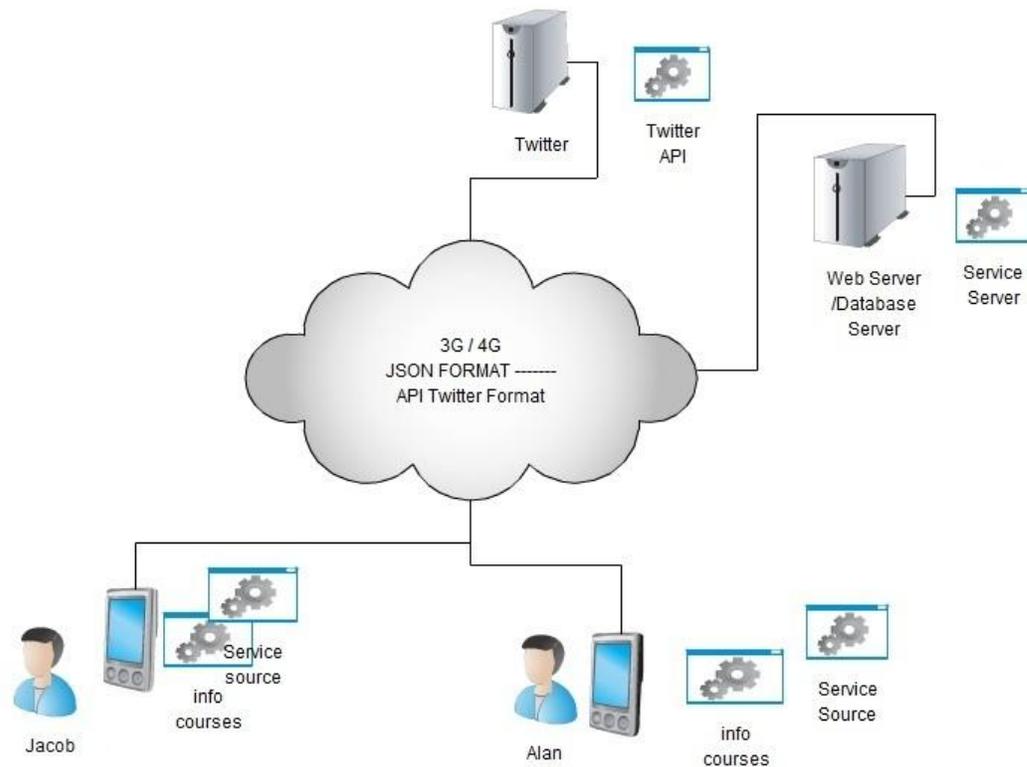
### **4.1. Arquitectura diseñada**

En términos generales, la aplicación se divide en dos áreas importantes que son:

- Aplicación de servicio web
- Aplicación Android

En la siguiente figura se ilustran estas dos áreas.

Figura 10. Diagrama de la arquitectura



Fuente: elaboración propia utilizando el programa Microsoft Visio 2007.

En la figura anterior se puede observar que en cada usuario de Android se ejecuta la aplicación, dentro de la cual hay dos servicios. El primero ejecuta un servicio para la comunicación con el servidor web donde se almacena la información de los periodos de los semestres, cursos, asignaciones y usuarios de la aplicación. El segundo servicio sirve para la comunicación con la aplicación de Twitter. Esta aplicación de Twitter funciona para la autenticación.

Otra funcionalidad obtenida desde la aplicación de Twitter es que permite crear *tweets* y listar *tweets* según el curso seleccionado, agregando un *hashtag* para diferenciar los cursos entre ellos.

La aplicación de Twitter obtiene los *tweets* asociados al *hashtag*, validando de esa forma el curso al que pertenece cada *tweet*; esto está integrado a la vez con el motor de Android para la representación visual de la información a los usuarios y el motor de comunicación hacia la aplicación de servicio web.

La forma de comunicación entre las aplicaciones ejecutándose sobre Android, los servicios web de Twitter y servicios internos, se realiza a través de una conexión de Internet sobre el dispositivo móvil que está ejecutando la aplicación, por ejemplo 3G.

Como se menciona anteriormente, el trabajo de graduación deja la integración de registro de usuarios a la continuidad de los interesados, teniendo en cuenta que el registro se realiza a través de un servidor propio de la aplicación. Sin embargo, el registro puede ser integrado de manera fácil, solo basta modificar el código según el método de autenticación acordado con las partes interesadas, por ejemplo los encargados de la Universidad Virtual de la Escuela de Ciencias y Sistemas o los encargados de la página de la Facultad de Ingeniería.

#### **4.2. Aplicación de servicio web**

Consiste en una aplicación web que se encarga del control de los cursos, así como también la asignación, desasignación y autenticación de los usuarios en la aplicación de Android. La aplicación web está diseñada en un modelo de

tres capas y desarrollada con Programación Orientada a Objetos (POO). Las tres capas del modelo son las descritas a continuación:

#### **4.2.1. Interfaz de comunicación**

Se encarga estrictamente de manejar las solicitudes que se realizan desde la aplicación Android, utilizando métodos HTTP GET para obtener información, métodos HTTP POST para actualizar información y también formato JSON en cada solicitud.

#### **4.2.2. Lógica de negocio**

Se encarga de procesar la información recibida desde el Android y realizar los flujos necesarios para hacer solicitudes a la base de datos, según sea la solicitud de Android, por ejemplo: cursos, usuario, asignación, etc.

#### **4.2.3. Acceso a la base de datos**

Es un componente el cual tiene implementados los lenguajes de manipulación de datos. Se usa Programación Orientada a Objetos (POO) sobre lenguaje PHP para el desarrollo de la misma.

A nivel de implementación de código, las clases importantes desarrolladas tienen como nombres:

- service\_get.php
- service\_set.php

La clase `service_get.php` está implementada para obtener información desde un cliente a través del método HTTP GET. La clase `service_set.php` está implementada para enviar información de actualización hacia la base de datos a través del método HTTP POST.

El orden del flujo de ejecución de una petición consiste en la verificación en las clases de `service_get.php` o `service_set.php`, encargadas de verificar las solicitudes con el método correspondiente: HTTP POST o HTTP GET.

Cuando el método valida a qué parte de la lógica del negocio pertenece (solicitud de información de usuario, solicitud de información de cursos, actualización de información de usuario u otro tipo de información), se invoca la clase correspondiente para procesar la información solicitada, obtenida en los parámetros enviados para la construcción adecuada de proceso de información. Con esto se da a entender que se actualiza la base de datos, agregando u obteniendo información.

Las clases creadas para la lógica del negocio en el lado del servicio web son las siguientes:

- `class_user.php`
- `class_registro.php`
- `class_semetre.php`
- `class_courses.php`

La clase encargada de la conexión hacia la base de datos, así como las operaciones de manipulación de datos sobre ella, tiene el nombre de: `class_connection.php`.

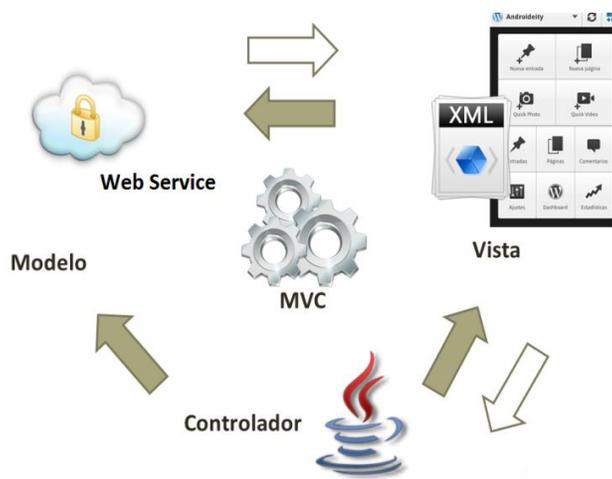
### 4.3. Aplicación Android

Sobre la aplicación Android se enfoca la mayor parte del capítulo, describiendo el funcionamiento a través del material de apoyo, así como una explicación sobre el desarrollo de un proyecto de base para implementar nuevos proyectos en Android.

### 4.4. Otros aspectos a tomar en cuenta en la arquitectura

Para el desarrollo de la aplicación se usa arquitectura híbrida, considerando que el modelo que Android propone es: Modelo, Vista y Controlador (MVC). Se usa un modelo en capas utilizando paradigma de Programación Orientada a Objetos (POO) con el fin de realizar una aplicación altamente cohesiva y débilmente acoplada. El modelo MVC se muestra en la siguiente figura.

Figura 11. Diagrama de Modelo Vista Controlador



Fuente: elaboración propia utilizando Paint como parte de accesorio de Windows 7.

A continuación se describe brevemente la arquitectura de MVC y cómo es que se implementa en la aplicación.

#### **4.4.1. Modelo**

Son las representaciones construidas, basadas en la información con la que opera la aplicación; puede ser a través del SQLite o servicios web. Esta aplicación utiliza un servicio web programado como interfaz de comunicación en contacto con la aplicación desarrollada en lenguaje PHP. La interfaz de comunicación utiliza como componente HttpClient que es nativo de Java y la codificación JSON, al igual que en la aplicación de servicio web; la clase implementada es: `service_source.java`.

#### **4.4.2. Vista**

Es la manera que el usuario percibe la información; entendido de otra manera, es la interfaz de usuario con la que se interactúa directamente. Ayuda a los estilos de los componentes, colores, posición, tamaño, geometría, etc. En el caso de Android se realiza a través de archivos XML, los cuales son configurados en su forma nativa a través de etiquetas o ya sea con la ayuda del entorno de desarrollo integrado que se usa para el desarrollo de la aplicación. En un proyecto de Android, los archivos XML se encuentran dentro de la carpeta “/res” del proyecto.

#### **4.4.3. Controlador**

Es el código y la programación que se encarga de darle flujo a la aplicación. Estos controladores son programados en lenguaje Java y son el núcleo de la aplicación para su funcionamiento.

La aplicación Android implementa Programación Orientada a Objetos (POO) donde existen clases que tienen funciones específicas, siendo usadas por las clases de controlador de cada actividad en toda la aplicación, de manera que se extiende a un modelo en capas para la interacción entre el controlador y el modelo en Android.

La aplicación es desarrollada de esta forma dado que existen componentes de software altamente cohesivos y débilmente acoplados; entonces, con una organización correcta de clases se logra obtener lo que se pretende.

#### **4.5. Componentes utilizados en la aplicación**

A continuación se describe brevemente los componentes más importantes utilizados en el desarrollo de la aplicación.

##### **4.5.1. Actividades**

Son las interfaces que muestran la información de manera formateada. La aplicación está dividida en varias actividades para percibir con mejor precisión su funcionalidad.

Para mostrar los componentes se utilizan Layouts, que son la forma de representación de los componentes. Funcionan a través de una clase Java, donde se utilizan los archivos XML de la vista y ayudan a crear la actividad. Los más utilizados son los siguientes:

#### **4.5.1.1. Relative Layout**

Se utiliza en la actividad LoginActivity.java que tiene como vista: Login.xml, debido a que la actividad maneja el inicio de sesión de la aplicación. Ayuda a que cada componente tome como referencia la posición del padre (al momento de ordenar los componentes), de esta manera se tiene una geometría dependiente de los componentes padre.

#### **4.5.1.2. Absolute Layout**

Se utiliza en la actividad CrearTweet.java que tiene como vista: Tweet.xml, debido a que el Layout permite organizar los objetos con coordenadas específicas para cada uno de los componentes. Se usa para poder tener varios tipos de geometría, además otorga una mayor flexibilidad en cuanto a posición en la actividad.

#### **4.5.1.3. Linear Layout**

Se utiliza en la actividad mainMenu.java que tiene como vista Activity\_mainmenu.xml. Este tipo de geometría sirve para tener listados. Se usa para que el menú principal de la aplicación permita mostrar los cursos en forma de lista.

### **4.5.2. Tab Activity**

Se utiliza para la actividad de la cuenta del usuario. La cuenta de usuario es donde se puede agregar nuevos cursos, eliminar cursos asignados o ver cursos anteriores al periodo actual que ya no son visibles en el menú principal de la aplicación. Además, es usada así dado que necesita un menú accesible,

rápido e intuitivo, a diferencia del menú principal que está oculto. Considerar que el menú principal necesita aprovechar el espacio por lo importancia de los cursos asignados.

En el menú de la cuenta de usuario es importante tener accesible todas las opciones para poder personalizar; funciona como un integrador de las actividades de perfil del usuario y cursos.

#### **4.5.3. List Activity**

Se usa en las actividades que despliegan la información de los cursos. Este tipo de actividad permite adaptar un arreglo de Android, llamado Array Adapter, como fuente de información para cada uno de las filas donde se muestra la información. También se usa en la actividad MainmenuActivity, para intercambiar información de las filas de la actividad; se actualiza un arreglo de la información y a la vez se actualiza el Array Adapter para poder actualizar la información de la actividad.

Adicionalmente se implementan las opciones de Add Header View y Add Footer View que sirven dentro del listado para agregar las funcionalidades de desplazarse en el menú de los cursos, así como para agregar una fila extra para poder agregar dinámicamente más cursos cuando el listado es demasiado extenso.

#### **4.5.4. List View**

Se usa en las actividades que despliegan información de los cursos. Sin embargo, se crea un componente llamado Adapter que sirve para personalizar cada fila del componente List View.

Si no se realiza esta funcionalidad, no se puede dar formato a cada registro del listado ya que por configuración predeterminada solamente posee texto plano y sin formato. Para poder hacer uso de elementos personalizados se debe usar el componente Custom Adapter.

#### **4.5.5. Image View**

Se utiliza para cargar la imagen que se muestra en el perfil del usuario, dentro del menú de cuenta.

#### **4.5.6. Text View**

Se utiliza para mostrar el listado de cursos. En el desarrollo de la aplicación, los Text View que están dentro de cada registro del List View son personalizados a través de XML, que permite agregar cambios en cuanto a color de letra, posición del texto, etc.

#### **4.5.7. Button**

Se utiliza dentro de las actividades de la aplicación y son botones que son usados para ejecutar una acción; por ejemplo los botones: “Nuevo Tweet”, “Regresar”, “Asignar cursos”, etc., los cuales se personalizan desde un XML para asignar atributos; aunque existe la manera de modificar los atributos desde código Java durante la ejecución.

#### **4.5.8. Edit Text**

Se utiliza como un campo para ingresar texto a la aplicación; es utilizado como una línea, permitiendo habilitar la opción de muchas líneas. Algunas

veces se utiliza en las actividades de registro a la aplicación y en la creación de nuevos *tweets*.

#### **4.5.9. Alert Dialog**

Componente de Java que permite crear diálogos dentro de la aplicación. Se utiliza para la confirmación de la asignación y desasignación de cursos en la parte de la cuenta del usuario.

#### **4.5.10. JSON Parser**

La clase JSON Parser dentro de la aplicación Android, es la encargada de mandar parámetros a la aplicación de servicio web, mediante los métodos HTTP POST y HTTP GET. Dentro de esta clase se implementa el objeto Default Http Client, Http Post y Http Client; estos tres objetos sirven para obtener información o bien actualizar información en la base de datos en Internet.

La comunicación de recepción de información del lado del servicio web hacia la aplicación Android está en formato JSON; se utiliza el JSON\_Object, el cual decodifica la URL a un objeto manejable como un arreglo.

### **4.6. Integración con API Twitter**

El motor de noticias de la aplicación utiliza la famosa red social: Twitter, debido a la popularidad y versatilidad que provee a los desarrolladores. Para los desarrolladores, esta red social permite cohesionar la cuenta de Twitter personal de los usuarios con las herramientas de Twitter. A continuación se explican los pasos para crear la aplicación en la página de desarrollo de Twitter.

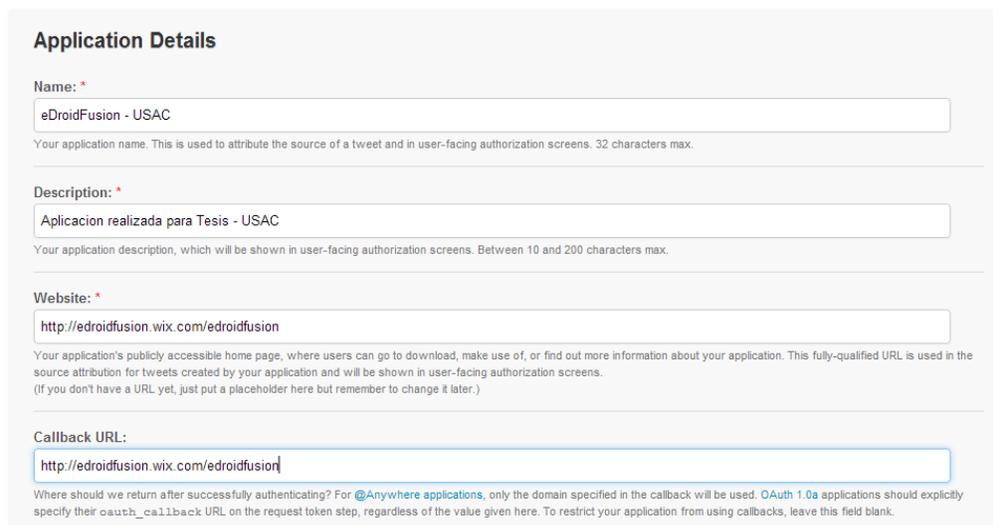
Para la creación de una nueva aplicación que va utilizar la API de Twitter debe de registrarse el desarrollador como usuario de Twitter e iniciar sesión para crear una nueva aplicación de desarrollo según la descripción siguiente:

#### 4.6.1. Registrar la aplicación Twitter

Al iniciar el registro de la aplicación se solicita el nombre que tendrá la aplicación en Twitter, la descripción de la aplicación, el sitio web de la aplicación y CallbackURL (dirección a donde se dirige cuando es autenticado correctamente). Si el usuario no desea especificar este último campo, se debe dejar vacío. La siguiente figura da una muestra al respecto.

Figura 12. Crear aplicación en la página de desarrolladores Twitter

### Create an application



The screenshot shows the 'Create an application' form on the Twitter developer page. It is titled 'Application Details' and contains four input fields with their respective labels and instructions:

- Name:** \* eDroidFusion - USAC  
Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.
- Description:** \* Aplicacion realizada para Tesis - USAC  
Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.
- Website:** \* http://edroidfusion.wix.com/edroidfusion  
Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully-qualified URL is used in the source attribution for tweets created by your application and will be shown in user-facing authorization screens. (If you don't have a URL yet, just put a placeholder here but remember to change it later.)
- Callback URL:** http://edroidfusion.wix.com/edroidfusion  
Where should we return after successfully authenticating? For @Anywhere applications, only the domain specified in the callback will be used. OAuth 1.0a applications should explicitly specify their oauth\_callback URL on the request token step, regardless of the value given here. To restrict your application from using callbacks, leave this field blank.

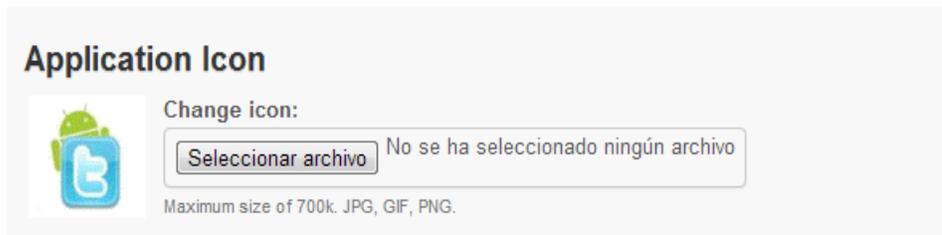
Fuente: elaboración propia sobre la aplicación desarrollada y API Twitter.

En la misma página se muestran las especificaciones a considerar como desarrollador. Aceptando estos términos y apegado a las normas, se comienza con la utilización de los parámetros que ofrece el desarrollo sobre Twitter.

#### 4.6.2. Configurar la aplicación Twitter

Cuando ya se encuentra creada la aplicación, se puede realizar cambios a la misma; por ejemplo la elección de un logo para la aplicación. En la figura siguiente se muestra un ejemplo.

Figura 13. Cambiando el ícono de la aplicación Twitter

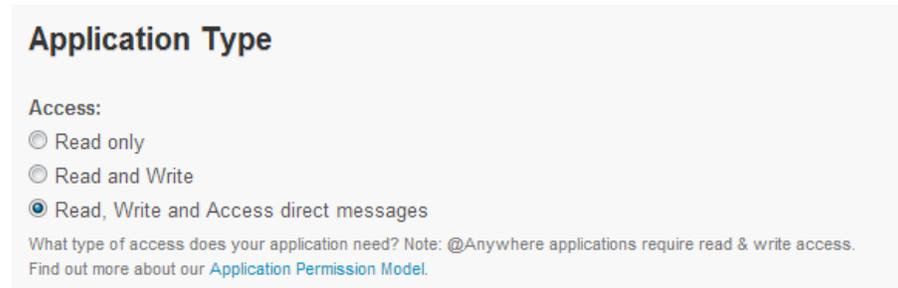


Fuente: elaboración propia sobre la aplicación desarrollada y API Twitter.

Se debe seleccionar una imagen en formato JPG, GIF o PNG. La dimensión recomendada para la utilización del logo es: 73x73 píxel.

También en la sección del tipo de aplicación se selecciona la opción de modo de lectura y escritura, debido a que se usa lectura de *tweets* del usuario y se escriben *tweets* hacia el perfil del usuario. A continuación se muestra una figura de ejemplo.

Figura 14. **Seleccionando el tipo de aplicación Twitter**



**Application Type**

Access:

- Read only
- Read and Write
- Read, Write and Access direct messages

What type of access does your application need? Note: @Anywhere applications require read & write access. Find out more about our [Application Permission Model](#).

Fuente: elaboración propia sobre la aplicación desarrollada y API Twitter.

La aplicación creada brinda los parámetros importantes que se deben utilizar para codificar dentro de la aplicación de Android; estos son los siguientes: Consumer Key, Consumer Secret, Acces Token y Access Token Secret. Los parámetros sirven para autenticación de la aplicación hacia el servidor de Twitter. Ver la siguiente figura para mayor entendimiento.

Figura 15. **Configurando la autenticación en Twitter**



**OAuth Settings**

Consumer key: \*

te4kc3c9BvUJVWnFAGRb6A

Consumer secret: \*

ULBw31WIHvVw3XA39aUV7PNSGtRNyzQ2qOmGRax8

Remember this should not be shared.

Access token: \*

845700794-dmONVvpYlepAjAS6Po3PhZDEklsUKGkXtIVlzjNC

Access token secret: \*

ln3qdrxNC6OQrMxSMKpHBfpSkfscEkOaj4HVSE

Remember this should not be shared.

Fuente: elaboración propia sobre la aplicación desarrollada y API Twitter.

### 4.6.3. Configurar parámetros iniciales en la aplicación

La librería de uso sobre Android tiene por nombre: Twitter4j, que ayuda a integrar con tecnología Java (API de Twitter y Android). La librería se puede descargar desde el link: <<http://twitter4j.org/en/index.html#download>>.

Dentro de la aplicación desarrollada se debe de especificar en el archivo Manifest cuyos permisos que hacen uso de la API de Java. Los permisos son los siguientes:

- Internet
- Access\_network\_state
- Intent\_filter

Figura 16. Ejemplo de archivo Manifest de la aplicación desarrollada

```
<application
  android:icon="@drawable/ic_launcher"
  android:label="@string/app_name"
  android:theme="@style/AppTheme" >
  <activity
    android:name=".MainActivity"
    android:label="@string/title_activity_main" >
    <intent-filter>
      <action android:name="android.intent.action.MAIN" />

      <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
    <intent-filter>
      <action android:name="android.intent.action.VIEW" />
      <category android:name="android.intent.category.DEFAULT" />
      <category android:name="android.intent.category.BROWSABLE" />
      <data android:scheme="oauth" android:host="t4jsample"/>
    </intent-filter>
  </activity>
</application>

<!-- Permission - Internet Connect -->
<uses-permission android:name="android.permission.INTERNET" />

<!-- Network State Permissions -->
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

Fuente: elaboración propia a partir de la aplicación Android desarrollada.

El archivo “JAR” descargado se debe de importar al proyecto donde va ser utilizado. Con la librería importada se puede usar, para eso se requiere declarar los parámetros que se obtienen en la página web de dev.twitter.com que se generan al registrar la aplicación.

Entonces, en la clase Java, donde se usa la aplicación de Twitter, se debe definir varios parámetros que son dados por su propio nombre. La siguiente figura muestra una porción de código donde se declaran esos parámetros.

Figura 17. **Variables para validación de autenticación Twitter**

```
* Register your here app https://dev.twitter.com/apps/new and get your
* consumer key and secret
* */
static String TWITTER_CONSUMER_KEY = "LsCQaPOwd8k7WkyRFRZF4Q";
static String TWITTER_CONSUMER_SECRET = "KJbJU5IQr1wxW7Cwnax3mMzAc4j3n6Wd2dG125srgk";

// Preference Constants
static String PREFERENCE_NAME = "twitter_oauth";
static final String PREF_KEY_OAUTH_TOKEN = "oauth_token";
static final String PREF_KEY_OAUTH_SECRET = "oauth_token_secret";
static final String PREF_KEY_TWITTER_LOGIN = "isTwitterLoggedin";

static final String TWITTER_CALLBACK_URL = "oauth://t4jsample";

// Twitter oauth urls
static final String URL_TWITTER_AUTH = "auth_url";
static final String URL_TWITTER_OAUTH_VERIFIER = "oauth_verifier";
static final String URL_TWITTER_OAUTH_TOKEN = "oauth_token";
```

Fuente: elaboración propia a partir de la aplicación Android desarrollada.

De la figura anterior, los dos primeros parámetros mostrados son los que se obtienen con el nombre de “Consumer Key” y “Consumer Secret”, dentro de la página web de desarrollo de Twitter. Adicionalmente se necesita utilizar los objetos “RequestToken” y “SharedPreferences”, los cuales se definen como variables y se utilizan para poder enviar los parámetros de autenticación y obtener información desde el usuario de Twitter.

Figura 18. **Variables para Twitter, RequestToken y SharedPreferences**

```
// Twitter
private static Twitter twitter;
private static RequestToken requestToken;

// Shared Preferences
private static SharedPreferences mSharedPreferences;
```

Fuente: elaboración propia a partir de la aplicación Android desarrollada.

Para poder realizar el proceso de reinicio de sesión dentro de la página de Twitter a través de la aplicación, se necesita de varias configuraciones y procesos previos. Para eso, se crea un objeto de la clase “Configuration Builder”, que en conjunto con un objeto llamado: ConfigurationObject, configuran y asignan la información de autenticación de las llaves (obtenidas de la aplicación web de Twitter) que se encuentran declaradas en el código de la aplicación Android que se desarrolla.

Posteriormente se utiliza la clase Twitter para crear un objeto que obtiene la instancia, la cual es utilizada por el procedimiento que realiza la solicitud a la aplicación web y hace que el dispositivo arranque la página de Twitter para poder iniciar sesión.

En la siguiente figura se muestra el fragmento de código que sirve para realizar el proceso respectivo de inicio de sesión en Twitter a partir del código de la aplicación.

Figura 19. **Proceso para solicitar el inicio de sesión en Twitter**

```
private void loginToTwitter() {
    // Check if already logged in
    if (!isTwitterLoggedInAlready()) {
        ConfigurationBuilder builder = new ConfigurationBuilder();
        builder.setOAuthConsumerKey(TWITTER_CONSUMER_KEY);
        builder.setOAuthConsumerSecret(TWITTER_CONSUMER_SECRET);
        Configuration configuration = builder.build();

        TwitterFactory factory = new TwitterFactory(configuration);
        twitter = factory.getInstance();

        try {
            requestToken = twitter
                .getOAuthRequestToken(TWITTER_CALLBACK_URL);
            this.startActivity(new Intent(Intent.ACTION_VIEW, Uri
                .parse(requestToken.getAuthenticationURL())));
        } catch (TwitterException e) {
            e.printStackTrace();
        }
    } else {
        // user already logged into twitter
        Toast.makeText(getApplicationContext(),
            "Already Logged into twitter", Toast.LENGTH_LONG).show();
    }
}
```

Fuente: elaboración propia a partir de la aplicación Android desarrollada.

Para la utilización básica de la aplicación y utilizando los parámetros y procesos realizados en las figuras 18 y 19, se debe tomar en cuenta que al momento que se realiza la solicitud a Twitter y el usuario inicia sesión, la aplicación retorna a su proceso, de tal forma que sigue el fragmento de código relacionado y ejecuta ciertas instrucciones de validación.

En la siguiente figura se puede observar el fragmento de código que realiza la verificación que el inicio de sesión retorne a partir de una solicitud web realizada previamente. Si es así, ingresa a la condición encargada de obtener el *token* de acceso para poder obtener la información del usuario que inicia sesión. Asimismo almacena (dentro de las preferencias compartidas) estas variables para utilizarlas en futuras operaciones, de tal forma que no se pierda la validación del usuario que inicia sesión.

Figura 20. Validación de autenticación en la página de Twitter

```
Uri uri = getIntent().getData();
if (uri != null && uri.toString().startsWith(TWITTER_CALLBACK_URL)) {
    // OAuth verifier
    String verifier = uri
        .getQueryParameter(URL_TWITTER_OAUTH_VERIFIER);

    try {
        // Get the access token
        AccessToken accessToken = twitter.getOAuthAccessToken(
            requestToken, verifier);

        // Shared Preferences
        Editor e = mSharedPreferences.edit();

        // After getting access token, access token secret
        // store them in application preferences
        e.putString(PREF_KEY_OAUTH_TOKEN, accessToken.getToken());
        e.putString(PREF_KEY_OAUTH_SECRET,
            accessToken.getTokenSecret());
        // Store login status - true
        e.putBoolean(PREF_KEY_TWITTER_LOGIN, true);
        e.commit(); // save changes

        Log.e("Twitter OAuth Token", "> " + accessToken.getToken());

        // Hide login button
        btnLoginTwitter.setVisibility(View.GONE);

        // Show Update Twitter
        lblUpdate.setVisibility(View.VISIBLE);
        txtUpdate.setVisibility(View.VISIBLE);
        btnUpdateStatus.setVisibility(View.VISIBLE);
        btnLogoutTwitter.setVisibility(View.VISIBLE);

        // Getting user details from twitter
        // For now i am getting his name only
        long userID = accessToken.getUserID();
        User user = twitter.showUser(userID);
        String username = user.getName();
    }
}
```

Fuente: elaboración propia a partir de la aplicación Android desarrollada.

#### 4.7. Aplicación de registro

Es la parte de la aplicación que puede ser cambiada según sea el origen del registro que se quiere; se puede integrar con cualquier plataforma a través de la utilización del servicio web de autenticación; por ejemplo, a través de la Universidad Virtual de la Escuela de Ciencias y Sistemas o el sitio oficial de la Facultad de Ingeniería.

Sin embargo, para los fines prácticos de este trabajo de graduación, se utiliza un registro propio de desarrollo para informar a los interesados la forma de cambio para integraciones próximas con la aplicación.

Dentro del servidor web que posee la accesibilidad hacia la base de datos, donde se encuentra la aplicación de servicio web, existe el servicio que tiene como función recibir solicitudes mediante el método HTTP GET realizado en lenguaje PHP; el archivo que maneja esto es: Wservicesdroid.php.

Este servicio se comunica con otro objeto manejado por el archivo Registro.php, el cual es una clase realizada en lenguaje PHP que recibe los parámetros correspondientes. Posterior a las validaciones correspondientes se procede a realizar el registro sobre la base de datos de la aplicación que se encuentra dentro de la web, para luego ser utilizado en la aplicación Android y poder enlazarse con el registro de Twitter.

Lo descrito con anterioridad es la parte que se puede reemplazar al utilizar otro sitio de registro o autenticación al intercambiar este servicio por algún otro proporcionado. En la aplicación, el proceso de registro se realiza a través de una página HTML pública, accesible desde la aplicación Android; dentro de la aplicación se usa la actividad MyBrowser.java, teniendo como vista: activity\_my\_browser.xml.

La página HTML realiza el proceso básico de recopilación de información a través de componentes HTML, enviándolos finalmente consolidados a las validaciones correspondientes para mantener la integridad de los datos y por último, enviarlos como una solicitud HTTP GET hacia el servicio web para poder guardar los registros. La página es accesible desde la web o bien en el flujo principal de la aplicación Android.

En la aplicación Android, la visualización de la página web se realiza a través del componente: Android WebView, debido a que se necesita tener un control sobre la página. Desde el componente Android WebView se envían

métodos controlados desde Java, los cuales son ejecutados en la página web para poder retornar a la aplicación Android después de la autenticación.

La página web de registro es accesible desde la página web de publicación de la aplicación: <<http://edroidfusion.wix.com/edroidfusion>>.

Con el botón de “Registrarse”, se redirige la aplicación a la página donde se realiza el proceso de registro de un nuevo usuario. Se describe de manera general la forma en que está desarrollada la parte de registro de la aplicación y poder darle continuidad a este proyecto.

## **5. DOCUMENTACIÓN DE USUARIO PARA USO DE LA APLICACIÓN DESARROLLADA**

Es importante aclarar que Twitter es la red social de 140 caracteres, muy diferente al resto y cada día adquiere mayor importancia por ser efectiva en tiempo real.

### **5.1. Resumen de la aplicación**

Twitter es un constante flujo de información, donde los usuarios comentan absolutamente de todo, interactúan entre sí y es un mundo donde las empresas deben saber incorporarse. Como red social da la posibilidad de llegar a un enorme público de manera gratuita; sin embargo, hay que saber cómo hacerlo y crear una imagen favorable.

Por otro lado, la Universidad Virtual (UV) de la Carrera de Ingeniería en Ciencias y Sistemas de la Universidad de San Carlos de Guatemala, es una plataforma virtual propia de la carrera que permite a los estudiantes asignarse a los cursos impartidos en los diferentes semestres y acceder a la información principal de dichos cursos; esto demuestra que la UV es la plataforma ideal de integración con la aplicación o bien un buen ejemplo de integración.

La aplicación está encargada de interactuar entre ambos sistemas: Twitter y la aplicación web, de tal forma que permite a los alumnos tener un perfil específico para poder asignarse los cursos de la carrera de Ingeniería en Ciencias y Sistemas de la Universidad de San Carlos de Guatemala y poder difundir o recibir información a través de los *tweets* realizados por los diferentes

estudiantes. Para poder descargar la aplicación se debe acceder al sitio oficial de la aplicación: <<http://edroidfusion.wix.com/edroidfusion>>.

Dentro del sitio antes mencionado, se presiona el botón correspondiente para descargar. En el sitio web se muestra una descripción breve de la aplicación, teniendo como único fin, descargar la aplicación y promocionar la misma.

Este capítulo no describe una guía completa. No se pretende proporcionar demasiada información, pues el objetivo es brindar información para obtener conocimiento base y esencial para poder utilizar la aplicación. Está especialmente enfocado para principiantes y novatos en la rama de informática, pero también se intenta dar un pequeño enfoque hacia la Escuela de Ciencias y Sistemas.

## **5.2. Proceso básico de la aplicación**

Para facilitar el uso de la aplicación se considera importante conocer el proceso que los estudiantes interesados y docentes deben seguir. Básicamente el proceso para utilizar la aplicación y realizar comentarios (o *tweets*) sobre un curso, consta de los siguientes tres pasos:

- Registrarse o iniciar sesión
- Asignar o seleccionar el curso
- Crear nuevo mensaje (*tweet*)

En los siguientes apartados se describe brevemente la información que se debe saber para poder realizar cada uno de los pasos anteriores,

adicionalmente se muestra otro tipo de opciones que son importantes tomarlas en cuenta para poder administrar correctamente los cursos.

### 5.3. Pantalla principal

La pantalla inicial de la aplicación es similar a la siguiente figura:

Figura 21. **Pantalla inicial de la aplicación**



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

En dicha pantalla se tiene dos opciones disponibles:

- Registrar: es el proceso principal, es necesario para poder utilizar la aplicación a través del usuario correspondiente que se crea.
- Iniciar sesión: se necesita estar registrado en la aplicación para poder dar inicio de sesión con el usuario y contraseña correspondiente.

## 5.4. Proceso de registro o inicio de sesión

En este apartado se explica cómo registrar o iniciar sesión con un usuario ya creado en la aplicación.

### 5.4.1. Proceso de registro

Para registrarse a la aplicación se debe seleccionar la etiqueta “Nuevo en eDroidFusion? Registrar aquí...” que aparece en la parte inferior de la pantalla inicial de la aplicación. A continuación una figura de ejemplo:

Figura 22. Opción para registrar nuevo usuario



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

El proceso anterior hace que se cargue la página web de registro del usuario, el cual se usa como registro. Este proceso puede ser omitido y realizar cambios correspondientes para que se pueda autenticar mediante otro sistema;

por ejemplo: Universidad Virtual de la Escuela de Ciencias y Sistemas, SAE/SAP de la Facultad de Ingeniería u otro, según lo explicado con anterioridad, dando como accesibilidad mediante la web o bien a través de la aplicación Android. En la aplicación Android se tiene la pantalla mostrada en la siguiente figura:

Figura 23. **Pantalla de registro de usuario nuevo**

Ya tengo usuario, regresar a la App y finalizar

Regresar a la

Registro EdroidFusion

Nombre Completo

Usuario (Carnet)

Password

Reingrese el Password

Correo Electronico

Telefono

Seleccionar Carrera

Ing. Civil

Direccion residencia

Fuente: elaboración propia a partir de la aplicación Android desarrollada.

El usuario debe llenar los datos solicitados en la pantalla de registro de tal forma que se pueda crear un correcto registro de autenticación que se asocie a la cuenta de Twitter. Entonces, al finalizar la creación del usuario, el usuario se redirecciona a la aplicación Android para continuar con el registro, el cual consiste en unir ambas cuentas: la cuenta de la aplicación desarrollada y la cuenta de la aplicación Twitter.

En la siguiente figura se observan los campos en los cuales se debe ingresar el usuario y contraseña creada en el paso anterior.

Figura 24. **Pantalla de registro de usuario nuevo**



Registrar nueva cuenta

ecys

Usuario edroid (Carnet)

Contraseña edroid

Registrar Nueva Cuenta

Ya tienes cuenta? Ingresar aquí...

Fuente: elaboración propia a partir de la aplicación Android desarrollada.

Presionar el botón “Registrar Nueva Cuenta” si el usuario desea registrarse; luego se espera que la aplicación se encargue de validar que el usuario y contraseña se encuentren registrados dentro del sistema de usuarios permitidos.

Si ocurre un error en el proceso de autenticación, la aplicación muestra el mensaje dependiendo el tipo de error. A continuación se muestra ejemplos de algunos posibles errores:

- El mensaje: “Asegúrese que el usuario sea un Número de carné. Corrija y vuelva intentarlo.”, significa que el usuario ingresado debe ser un número de carné y no debe contener letras o signos. Solamente se acepta el número como tal.

Figura 25. **Usuario ingresado erróneamente**



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

- El mensaje “No existe el usuario (carné) registrado en la UV. Favor de ingresar Usuario/Contraseña correcto para poder registrarse.”, indica claramente que la contraseña o el usuario ingresado no coinciden (son incorrectas las credenciales de acceso). Puede darse el caso que el usuario no está registrado en la aplicación, para lo cual, se debe solicitar al administrador local de la aplicación que haga el favor de configurar un usuario para el estudiante. El administrador puede ser contactado a través del correo electrónico: <edroidfusion@gmail.com>.

Figura 26. **Credenciales de acceso incorrectas**



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

- El mensaje “Ya existe registrado este usuario (carné) en la aplicación. Favor de iniciar sesión con el Usuario/Contraseña correcto para la aplicación.”, indica que el usuario ya se ha registrado anteriormente y por lo tanto no se puede volver a registrar. En este caso ya se puede iniciar sesión a la aplicación con el usuario y contraseña correspondiente.

Figura 27. **Usuario registrado anteriormente**

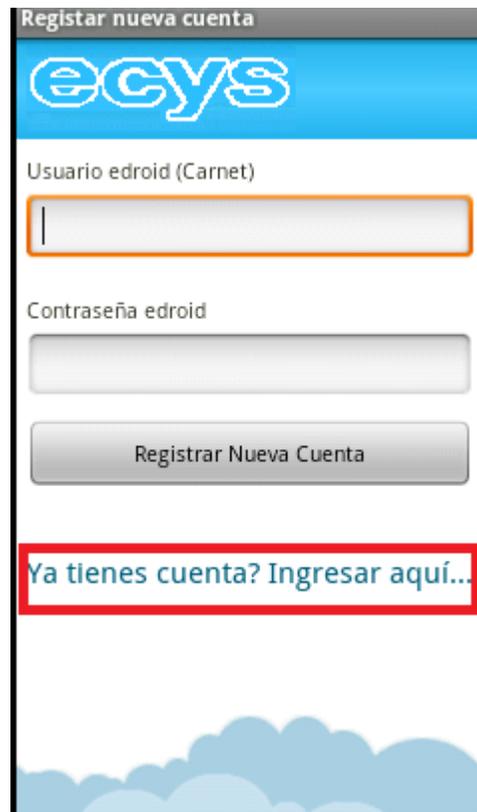


Fuente: elaboración propia a partir de la aplicación Android desarrollada.

NOTA: Si el usuario no quiere registrarse o si quiere regresar a la página principal para ingresar a la aplicación con una cuenta existente, debe presionar

la etiqueta “¿Ya tienes cuenta? Ingresar aquí...”. A continuación se muestra una figura que describe la opción comentada anteriormente.

Figura 28. **Opción de ingreso a la aplicación en pantalla de registro**



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

Si la autenticación a través de la aplicación de registro es satisfactoria, se pide que el estudiante inicie sesión con un usuario y contraseña creada a través de la página de Twitter. Esto sirve para que la aplicación tenga relacionado el carné del estudiante con un usuario específico de Twitter (el cual debe ser único para utilizarlo en la aplicación).

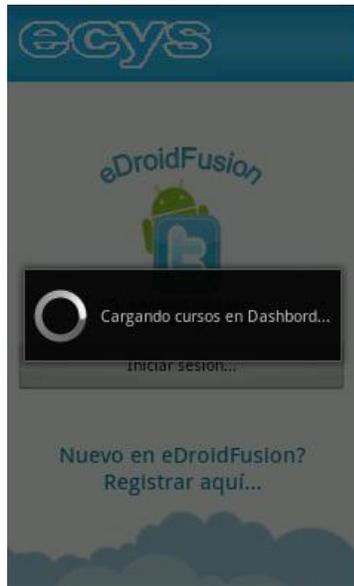
Figura 29. **Iniciar sesión en Twitter**

The image shows a login screen for an application named 'eDroidFusion - USAC'. At the top left is a small Twitter bird icon. To its right, the text reads 'eDroidFusion - USAC', 'Por Universidad de San Carlos de Guatemala', and 'edroidfusion.wix.com/edroidfusion'. Below this is a horizontal line. Under the line, the text says 'Esta aplicación **será capaz de:**' followed by a bulleted list: 'Leer Tweets de tu cronología.', 'Ver a quién sigues y seguir a nuevas personas.', 'Actualizar tu perfil.', and 'Publicar Tweets por ti.' Below the list are two input fields: 'Nombre de usuario o correo electrónico' and 'Contraseña'. Under the password field is a checkbox labeled 'Recordar mis datos' and a link '¿Olvidaste tu contraseña?'. At the bottom of the form are two buttons: 'Inicia Sesión' (highlighted in blue) and 'Cancelar' (highlighted with a red border). Below the buttons, the text says 'Esta aplicación **no tendrá capacidad para:**' followed by a bulleted list: 'Acceder a tus mensajes directos.' and 'Ver tu contraseña de Twitter.'

Fuente: elaboración propia a partir de la aplicación Android desarrollada.

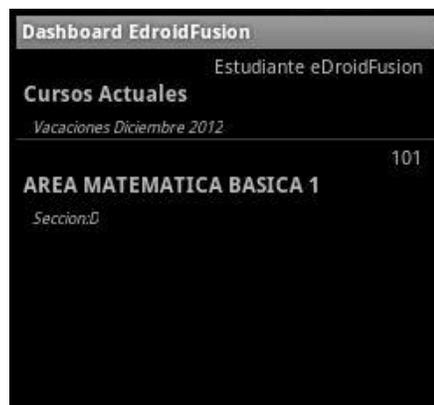
Quando se finaliza el inicio de sesión en Twitter, la aplicación realiza automáticamente el proceso de registro del usuario en la aplicación. Mientras tanto, prepara la interfaz donde el usuario puede administrar los cursos.

Figura 30. **Cargando cursos en interfaz principal**



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

Figura 31. **Pantalla principal del usuario en la aplicación**



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

En la figura anterior se puede notar claramente que muestra el nombre del usuario (estudiante) en la parte superior derecha y los cursos asignados actualmente.

#### 5.4.2. Proceso de inicio de sesión

Por otro lado, como se menciona anteriormente, en la pantalla principal se tiene la opción de iniciar sesión cuando el usuario ya está registrado en la aplicación. Para realizar esto se debe presionar el botón “Iniciar sesión...” en la pantalla de inicio de la aplicación.

Figura 32. Opción para iniciar sesión



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

Después, la aplicación muestra la página principal de Twitter para poder iniciar sesión con la cuenta única de Twitter que se asigna al momento de registrar el usuario en la aplicación.

Figura 33. **Iniciar sesión con un usuario registrado en la aplicación**

eDroidFusion - USAC  
Por Universidad de San Carlos de Guatemala, Ingeniería  
edroidfusion.wix.com/edroidfusion

Esta aplicación **será capaz de:**

- Leer Tweets de tu cronología.
- Ver a quién sigues y seguir a nuevas personas.
- Actualizar tu perfil.
- Publicar Tweets por ti.

edroidfusion

\*\*\*\*\*

Recordar mis datos · [¿Olvidaste tu contraseña?](#)

**Inicia Sesión**    Cancelar

Esta aplicación **no tendrá capacidad para:**

- Acceder a tus mensajes directos.
- Ver tu contraseña de Twitter.

Fuente: elaboración propia a partir de la aplicación Android desarrollada.

Cuando se inicia sesión en la página de Twitter, la aplicación realiza automáticamente las validaciones necesarias para poder cargar los cursos y abrir la pantalla principal del usuario. Se recomienda ver las figuras 29 y 30.

Si ocurre un error en el proceso de inicio de sesión en la aplicación, se muestra el siguiente mensaje: “No existe registrado el usuario Twitter en la aplicación”. Esto indica claramente que no está registrado el usuario en la

aplicación y por lo tanto no se puede iniciar sesión con dicho usuario. Para poder iniciar sesión en la aplicación, el usuario deberá estar registrado dentro de la aplicación.

## **5.5. Pantalla principal de usuario**

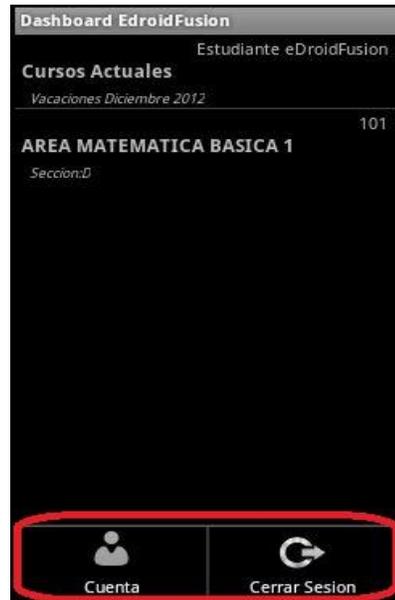
La pantalla principal de usuario muestra todos los cursos asignados por el estudiante en el semestre actual. El aspecto de esta pantalla se puede observar en la figura 30.

Si el estudiante no tiene cursos asignados en el semestre actual, pues no aparecen cursos, entonces debe el estudiante debe asignarse los cursos. Para la asignación de cursos del estudiante, ver el apartado 5.7 que corresponde a la pantalla para cursos.

### **5.5.1. Menú para la pantalla principal de usuario**

El menú para la pantalla principal de usuario se activa al presionar la tecla “Menú” del dispositivo móvil (dentro de la pantalla principal de usuario). El menú aparece en la parte inferior de la pantalla.

Figura 34. Menú para la pantalla principal de usuario



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

El menú para la pantalla principal de usuario incluye las siguientes opciones:

- Cerrar sesión: sirve para finalizar la sesión del usuario en la aplicación, permitiendo que el usuario pueda iniciar sesión nuevamente desde la pantalla inicial de la aplicación. Al presionar esta opción, la aplicación cierra la sesión del usuario y redirecciona a la pantalla inicial.
- Cuenta: muestra la información básica del usuario. Esta información es la que se asigna cuando el usuario solicita crear un nuevo permiso para utilizar el sistema. Es conocida comúnmente como perfil de usuario.

## 5.6. Perfil de usuario

La pantalla del perfil de usuario tiene un aspecto como se muestra en la siguiente figura.

Figura 35. Perfil de usuario



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

En el perfil de usuario aparece la información básica, por ejemplo: nombre, carné, usuario Twitter, carrera, *email*, número de móvil, dirección de domicilio y la foto respectiva que se encuentra en el perfil principal de la página de Twitter.

### 5.6.1. Menú para el perfil de usuario

El menú para el perfil de usuario aparece activo de forma fija en la parte inferior de la pantalla. En la siguiente figura se puede observar el menú en la parte inferior.

Figura 36. Menú para el perfil de usuario



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

El menú del perfil de usuario incluye tres opciones, estas opciones se describen a continuación:

- Perfil: muestra la información básica del usuario. En este caso, es la opción en la cual se encuentra por configuración predeterminada.

- Cursos: pantalla encargada de realizar las gestiones principales de los cursos; por ejemplo: consultar, asignar y eliminar.
- Dashboard: nombre asignado para hacer referencia a la pantalla principal del usuario, en la que se muestran todos los cursos asignados en el semestre actual.

### 5.7. Pantalla para cursos

La pantalla para cursos tiene un aspecto como se muestra en la siguiente figura.

Figura 37. **Pantalla para cursos**



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

La pantalla para cursos incluye tres opciones para administrar los cursos. A continuación se describen brevemente esas opciones:

- Ir a cursos: permite consultar los cursos asignados por el usuario en los distintos años y semestres disponibles.

- Agregar cursos: permite agregar un nuevo curso en los distintos años y semestres disponibles.
- Eliminar cursos: permite eliminar cursos asignados por el usuario en los distintos años y semestres disponibles.

Para cada una de las tres opciones anteriores, cuando el usuario selecciona alguna, muestra el listado de años permitidos, de tal forma que el usuario pueda seleccionar el año que corresponde a la gestión que desea realizar.

Figura 38. **Seleccionar año para gestionar cursos**



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

Asimismo, después de seleccionar el año respectivo, para cada una de las tres opciones, se muestra la pantalla que contiene los semestres disponibles, permitiendo al usuario seleccionar el semestre.

Figura 39. **Seleccionar el semestre para gestionar cursos**



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

Luego, dependiendo la gestión que realiza el usuario, podrá seleccionar el curso al cual se le aplica la acción. La pantalla muestra los cursos disponibles en el semestre y año seleccionado.

## **5.8. Consultar y crear mensajes (*tweets*)**

A continuación se explica la manipulación de los *tweets* dentro de la aplicación.

### **5.8.1. Consultar *tweets***

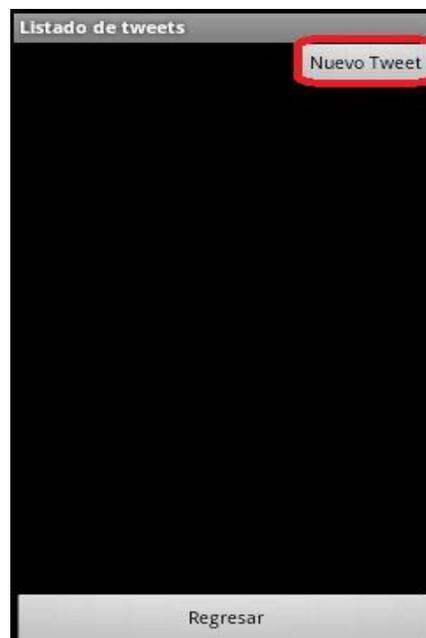
Para poder consultar los *tweets* realizados en un curso específico, el usuario debe seleccionar el curso correspondiente. Esto se puede realizar

desde la pantalla principal del usuario (para ver los *tweets* de los cursos asignados actualmente) o desde el menú de cursos (para ver los cursos asignados en otros años y semestres). Entonces, cuando el usuario selecciona el curso, automáticamente aparece el listado de *tweets* relacionados a dicho curso.

### 5.8.2. Crear *tweets*

Para poder crear *tweets*, el usuario debe seleccionar la etiqueta llamada: “Crear Tweet”. Esta etiqueta se encuentra en la parte superior derecha dentro del listado de *tweets* que se consultan.

Figura 40. Crear *tweet* para un curso específico



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

La opción de poder crear *tweets* solamente está habilitada para los cursos asignados en dicho momento. Para los cursos asignados en años y semestres anteriores solamente está habilitada la opción de poder consultar *tweets* (ver listado de *tweets*) relacionados al curso.

Cuando el usuario selecciona el botón llamado: “Nuevo Tweet”, podrá observar un área de texto en el cual debe ingresar el estatus (*tweet*) que desea realizar en el curso específico. También podrá observar el *hashtag* en la parte inferior (debajo del área de texto).

Figura 41. **Pantalla para crear nuevo *tweet***



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

Para crear un *tweet*, este debe cumplir las siguientes condiciones:

- Deberá contener al menos un carácter como parte del texto a ingresar en el área de texto.
- Deberá tener a lo máximo 125 caracteres como parte del texto a ingresar en el área de texto.

Después de ingresar el texto para el *tweet*, el usuario debe presionar el botón “Aceptar” para actualizar el *tweet* en la página de Twitter y presionar el botón “Regresar” para cancelar el proceso de actualización de estado.

Figura 42. **Actualizar en Twitter al crear un nuevo *tweet***



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

Al finalizar el proceso de actualización, la aplicación muestra el mensaje de confirmación respectivo.

Figura 43. **Tweet creado de forma correcta**

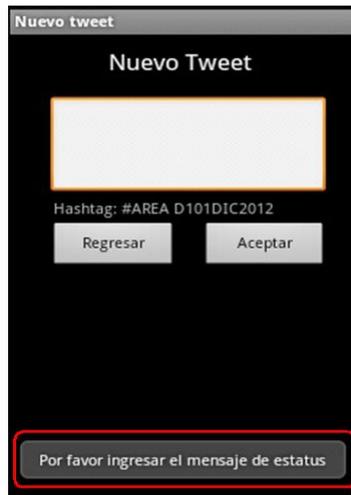


Fuente: elaboración propia a partir de la aplicación Android desarrollada.

Si el *tweet* no cumple con alguna de las dos condiciones mencionadas anteriormente, entonces no se puede crear, lo cual conlleva a mostrar un mensaje de error.

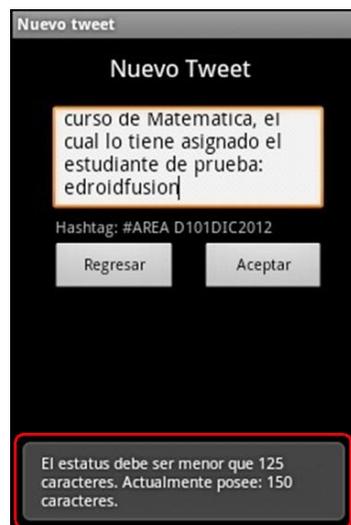
A continuación se muestran dos figuras mostrando estos tipos de mensajes de error.

Figura 44. **Tweet con mensaje de estatus vacío**



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

Figura 45. **Tweet con estatus que excede el tamaño máximo**



Fuente: elaboración propia a partir de la aplicación Android desarrollada.

## CONCLUSIONES

1. El estudio permite establecer la necesidad de implementar en la Facultad de Ingeniería este tipo de aplicación, con el fin de aprovechar la tecnología y actualizar sus métodos para difundir la información relacionada a la vida académica y desarrollo informativo, así como, facilitar la comunicación entre facultad, docente y población estudiantil.
2. Se genera el manual básico del desarrollo de aplicación, que sirve de guía para estudiantes con conocimientos de Programación Orientada a Objetos (POO) que quieran desarrollar sobre la plataforma Android.
3. El estudio demuestra, a través de la aplicación, cómo cohesionar las tecnologías: Android, PHP, SQL y Twitter, en una sola aplicación; de tal forma que los interesados puedan continuar el desarrollo o bien sirva de idea para integrar varias tecnologías en una sola aplicación.
4. Se crea la aplicación capaz de difundir rápidamente las noticias de cursos gracias al motor de la red social de Twitter y un servidor de autenticación para validación de usuarios, de manera que todos los interesados puedan utilizarla para obtener noticias en tiempo real.
5. La modernización de la facultad a través del uso de la aplicación será un proceso gradual conforme los usuarios la utilicen y la funcionalidad que ellos obtengan a través de su utilización, asegurando el éxito a mediano plazo debido a que la necesidad es latente, pues a la fecha no existe un sistema con este tipo de aplicación.



## RECOMENDACIONES

1. La Facultad de Ingeniería puede considerar esta propuesta, pues la aplicación da un avance y actualización a su sistema de difusión y comunicación, para que toda la población estudiantil posea acceso a información actualizada, de manera rápida y efectiva.
2. Dar continuidad al proyecto actual para que sea utilizado en toda la Facultad de Ingeniería y en un futuro sea utilizada por todos los usuarios de la Universidad de San Carlos de Guatemala, dando así un avance de modernización tecnológica a la universidad.
3. Automatizar el proceso de carga de cursos por semestres mediante la creación de un servicio web entre la Facultad de Ingeniería y la persona que brinde continuidad al proyecto actual, tomando en cuenta además la definición de los periodos por semestre.
4. Para desarrollar esta aplicación se debe tener conocimientos en: Programación Orientada a Objetos (POO) y Programación Web (PW), usando como preferencia programación PHP; asimismo, visitar el sitio oficial de desarrollo de aplicaciones Android y utilizar este material como complemento para poder conocer cómo estructurar un nuevo proyecto.
5. Tomar como base el actual proyecto para hacer un desarrollo sobre sistemas operativos iOS, de manera que se pueda abarcar más población de usuarios, pues esta herramienta beneficiará en su totalidad a la población estudiantil.



## BIBLIOGRAFÍA

1. Android Developers [en línea]. <<http://developer.android.com/>>. [Consulta: 15 de agosto de 2012].
2. CHALI MONROY, Rolando Iván. *Desarrollo de una herramienta móvil de software para apoyar a estudiantes universitarios en su organización*. Trabajo de graduación de Ing. en Ciencias y Sistemas. Universidad de San Carlos de Guatemala, Facultad de Ingeniería, 2012. 106 p.
3. DAMIÁN MARROQUÍN, Félix Javier. *Programación de dispositivos móviles*. Trabajo de graduación de Ing. en Ciencias y Sistemas. Universidad de San Carlos de Guatemala, Facultad de Ingeniería, 2006. 139 p.
4. Superintendencia de Telecomunicaciones de Guatemala. *Crecimiento de la telefonía fija y móvil en Guatemala* [en línea]. <[http://www.sit.gob.gt/oldSite/uploads/docs/stats/ctfm/CreTelefonia\\_1sem11.pdf](http://www.sit.gob.gt/oldSite/uploads/docs/stats/ctfm/CreTelefonia_1sem11.pdf)>. [Consulta: 10 de septiembre de 2012].
5. TAMADA, Ravi. *AndroidHive* [en línea]. <<http://www.androidhive.info/>>. [Consulta: 20 de agosto de 2012].