



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

**IMPLEMENTACIÓN DE APLICACIÓN MÓVIL PARA PROMOCIÓN
TURÍSTICA UTILIZANDO TECNOLOGÍA ANDROID GUATE EXTENDIDA**

Carlos Alberto Prado Sanchez

Carlos Giovanni Barillas Colon

Asesorado por el Ing. Ricardo Israel Mazariegos Castillo

Guatemala, enero de 2015

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**IMPLEMENTACIÓN DE APLICACIÓN MÓVIL PARA PROMOCIÓN
TURÍSTICA UTILIZANDO TECNOLOGÍA ANDROID GUATE EXTENDIDA**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

CARLOS ALBERTO PRADO SANCHEZ

CARLOS GIOVANI BARILLAS COLON

ASESORADO POR EL ING. RICARDO ISRAEL MAZARIEGOS CASTILLO

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, ENERO DE 2015

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympto Paiz Recinos
VOCAL I	Ing. Angel Roberto Sic García
VOCAL II	Ing. Pedro Antonio Aguilar Polanco
VOCAL III	Inga. Elvia Miriam Ruballos Samayoa
VOCAL IV	Br. Narda Lucía Pacay Barrientos
VOCAL V	Br. Walter Rafael Véliz Muñoz
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

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

HONORABLE TRIBUNAL EXAMINADOR

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

**IMPLEMENTACIÓN DE APLICACIÓN MÓVIL PARA PROMOCIÓN
TURÍSTICA UTILIZANDO TECNOLOGÍA ANDROID GUATE EXTENDIDA**

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha 3 de julio de 2013.


Carlos Giovanni Barillas Colon


Carlos Alberto Prado Sanchez

Guatemala, 08 de mayo de 2014

**Ingeniero
Marlon Antonio Pérez Turk
Director
Escuela de Ciencias y Sistemas
Facultad de Ingeniería
Universidad de San Carlos de Guatemala**

Ingeniero Pérez Turk:

Me complace saludarle, haciendo referencia al trabajo de graduación titulado "IMPLEMENTACIÓN DE APLICACIÓN MÓVIL PARA PROMOCIÓN TURÍSTICA UTILIZANDO TECNOLOGÍA ANDROID GUATE EXTENDIDA", desarrollado por el estudiante universitario Carlos Giovani Barillas Colón con número de carné 200722384 y el estudiante universitario Carlos Alberto Prado Sanchez con número de carné 200818928, que como asesor apruebo el contenido del mismo.

Para su conocimiento y efectos, sin otro particular, me suscribo.



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

Atentamente,
Ing. Ricardo Mazariegos
Asesor



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

Guatemala, 6 de Agosto de 2014

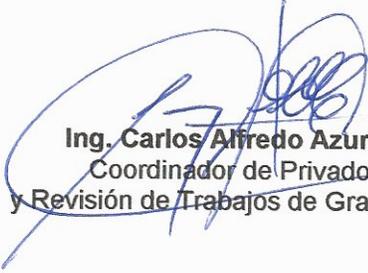
Ingeniero
Marlon Antonio Pérez Turk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación de los estudiantes **CARLOS ALBERTO PRADO SANCHEZ** con carné **2008-18928** y **CARLOS GIOVANI BARILLAS COLÓN** con carné **2007-22384**, titulado: **"IMPLEMENTACIÓN DE APLICACIÓN MÓVIL PARA PROMOCIÓN TURÍSTICA UTILIZANDO TECNOLOGÍA ANDROID GUATE EXTENDIDA"**, 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 APLICACIÓN MÓVIL PARA PROMOCIÓN TURÍSTICA UTILIZANDO TECNOLOGÍA ANDROID GUATE EXTENDIDA”**, realizado por los estudiantes CARLOS ALBERTO PRADO SANCHEZ y CARLOS GIOVANI BARILLAS COLON aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAD A TODOS”



*Ing. Maxlon Antonio Pérez Türk
Director, Escuela de Ingeniería en Ciencias y Sistemas*

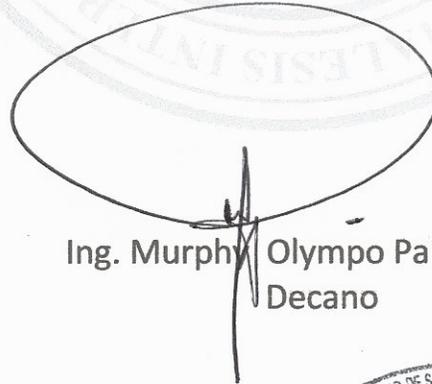
Guatemala, 26 de enero 2015



DTG. 019.2015

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 APLICACIÓN MÓVIL PARA PROMOCIÓN TURÍSTICA UTILIZANDO TECNOLOGÍA ANDROID GUATE EXTENDIDA**, presentado por los estudiantes universitarios: **Carlos Alberto Prado Sanchez y Carlos Giovani Barillas Colón**, y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, autoriza la impresión del mismo.

IMPRÍMASE:



Ing. Murphy Olympo Paiz Recinos
Decano

Guatemala, 27 de enero de 2015



/gdech

ACTO QUE DEDICO A:

Dios	Por ser el centro de mi vida y guiarme en cada paso que he dado.
Mis padres	Por darme la vida, su amor, su apoyo y por la buena educación brindada que me ha llevado a plantearme y cumplir con esta meta.
Mi hermana	Porque son parte importante de mi vida y quiero compartir este logro con ella.

Carlos Alberto Prado Sanchez

AGRADECIMIENTOS A:

Dios	Por permitirme cumplir las metas que me he propuesto y compartir este logro con las personas que quiero.
La Universidad de San Carlos de Guatemala	Por ser la casa de estudios que me ha permitido alcanzar este grado académico.
Mis Padres	Por todo su apoyo, por invertir en mis estudios y darme todo lo necesario para alcanzar este día.
Mis hermanas	Por su paciencia y cariño en esta etapa de mi vida.
Mis amigos	Por estar en los momentos de felicidad y adversidad.

Carlos Alberto Prado Sanchez

AGRADECIMIENTOS A:

- Dios** Por permitirme cumplir las metas que me he propuesto y compartir este logro con las personas que quiero.
- Mis padres** Por su apoyo incondicional al estar conmigo en las buenas y en las malas.
- Mis hermanas** Por darme apoyo y levantarme el ánimo en todo momento.
- Mis amigos** Werner Aldana, Fredy Velásquez, Selvin Naz, Carlos Prado, por ser mis hermanos y mis compañeros de batalla en todo momento

Carlos Giovani Barillas Colón

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	V
GLOSARIO	VII
RESUMEN.....	IX
OBJETIVOS.....	XI
INTRODUCCIÓN	XIII
1. CAPÍTULO DE ESTUDIO DE LA TECNOLOGÍA Y SU IMPACTO EN GUATEMALA	1
1.1. Identificar la teoría detrás que va a soportar la investigación	1
1.1.1. Teoría unificada de la aceptación y uso de la tecnología	1
1.1.2. Teoría de la acción razonada	3
1.1.3. Modelo de aceptación de tecnología	5
1.2. Teoría y la relación con la tecnología escogida.....	6
2. IDENTIFICACIÓN DE PROBLEMA Y SOLUCIÓN QUE LA APLICACIÓN REALIZARÁ.....	9
2.1. Mercado objetivo	10
2.2. Benchmark de la aplicación.....	10
3. DISEÑO DE LA APLICACIÓN BAJO LA NECESIDAD IDENTIFICADA	13
3.1. Prototipo	13
3.1.1. Galería.....	13
3.1.2. Información del sitio turístico	14

3.1.3.	Google Maps	16
3.1.4.	Galería de fotografías de un sitio precargado	17
3.1.5.	Subir imagen	18
3.2.	Validaciones, formularios	19
3.2.1.	Subir fotografía	19
3.2.2.	Registro	20
3.2.3.	Galería.....	21
3.3.	Diseño intuitivo y usabilidad	22
4.	DOCUMENTACIÓN Y TUTORIAL DE PROGRAMACIÓN DE LA APLICACIÓN	25
4.1.	Requisitos	25
4.1.1.	Herramientas.....	25
4.1.2.	Software	26
4.1.3.	Hardware.....	27
4.1.4.	Tutorial de desarrollo y referencias	27
4.1.4.1.	Configurar el entorno de desarrollo	27
4.1.4.2.	Menú principal	29
4.1.4.3.	Tomar fotografías	31
4.1.4.4.	GPS.....	32
4.1.4.5.	Servidor web	32
4.1.4.6.	Subir fotografía	35
4.1.4.7.	Galería <i>online</i>	36
4.1.5.	Consideraciones de implementación.....	37
4.1.5.1.	Escalar imágenes.....	37
4.1.5.2.	Async Task.....	38
4.1.5.3.	Multipart Entity.....	40
4.1.5.4.	GPS.....	43
4.1.5.5.	Manifest.....	44

4.1.5.6. Google Play	45
CONCLUSIONES	51
RECOMENDACIONES	53
BIBLIOGRAFÍA	55

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Teoría unificada de la aceptación y uso de la tecnología UTAUT	2
2.	Teoría de la acción razonada (TRA)	4
3.	Modelo de aceptación de una tecnología.....	6
4.	Aplicación descubre Guatemala.....	11
5.	Prototipo galería	14
6.	Prototipo información	15
7.	Prototipo Google Maps.....	16
8.	Prototipo galería local.....	17
9.	Prototipo subir imagen	18
10.	Navigation Drawer.....	22
11.	Actions.	23
12.	Android SDK	28
13.	Support Library.....	29
14.	Navigation Drawer.....	30
15.	Agregar las librerías	42
16.	Seleccionar las librerías	42
17.	Google Play Services	45
18.	Google Apis Console Create	46
19.	Nombre del proyecto	47
20.	Activar el servicio	47
21.	API Acess.....	48
22.	SHA1	49
23.	Configurar API Key.....	49

24. Archivo manifest50

TABLAS

I. Tabla Imagen33

GLOSARIO

Android	Android es un sistema operativo basado en Linux diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes o tabletas.
API	Es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Son usadas generalmente en las bibliotecas.
Debuguear	Probar y eliminar los errores del programa objetivo
GPS	Global Positioning System es un sistema de posicionamiento global que permite determinar en todo el mundo la posición de un objeto.
Loguear	Iniciar sesion en la aplicación a través de Facebook
Plug-in	Es una aplicación que se relaciona con otra para aportarle una función nueva y generalmente específica. Esta aplicación adicional es ejecutada por la aplicación principal e interactúan por medio de la API.

RESUMEN

El presente trabajo de graduación titulado Guate Extendida, es la documentación de la aplicación que lleva el mismo nombre, la cual está desarrollada en el Sistema Operativo Android y su objetivo es dar a conocer Guatemala, esto a través de información ya establecida en la aplicación sobre sitios turísticos conocidos, además que los usuarios pueden extender más la información subiendo sus propias fotografías.

Además de la aplicación se incluye un manual de desarrollo para todas aquellas personas que estén interesadas en el desarrollo móvil de aplicaciones para dispositivos Android. En este documento se detallan paso a paso como configurar el entorno de desarrollo y como desarrollar la aplicación, además de incluir una serie de recomendaciones para desarrollar aplicación en el Sistema Operativo Android.

OBJETIVOS

General

Proporcionar una herramienta para dar a conocer departamentos de Guatemala, a través de fotografías y ubicación de sitios turísticos.

Específicos

1. Dar a conocer los lugares turísticos de Guatemala por departamento.
2. Informar de cómo llegar a cada sitio turístico.
3. Permitir a usuarios compartir información de cada sitio turístico.
4. Permitir el acceso actualizado a información por medio de fotos de distintos lugares turísticos.

INTRODUCCIÓN

El uso de Smartphone ha tenido un gran crecimiento en los últimos años, por ello el proyecto “Guate extendida” proporciona un software en el cual los usuarios pueden tener a su alcance distintos sitios turísticos del interior del país.

La aplicación es intuitiva y sencilla de utilizar, permitiendo al usuario aumentar su catálogo de sitios turísticos a través de fotografías que pueden subir a la base de datos de la aplicación para que otros usuarios puedan verlas.

Al ser una aplicación móvil cualquier persona con un Smartphone y Sistema Operativo Android puede acceder a ella y ver los sitios turísticos que ofrece Guatemala.

La aplicación está enfocada en el departamento de San Marcos, ya que es un lugar con distintos lugares turísticos.

1. CAPÍTULO DE ESTUDIO DE LA TECNOLOGÍA Y SU IMPACTO EN GUATEMALA

1.1. Identificar la teoría detrás que va a soportar la investigación

A continuación se identificarán las teorías de investigación relacionadas con la aceptación e introducción de nuevas tecnologías en la sociedad.

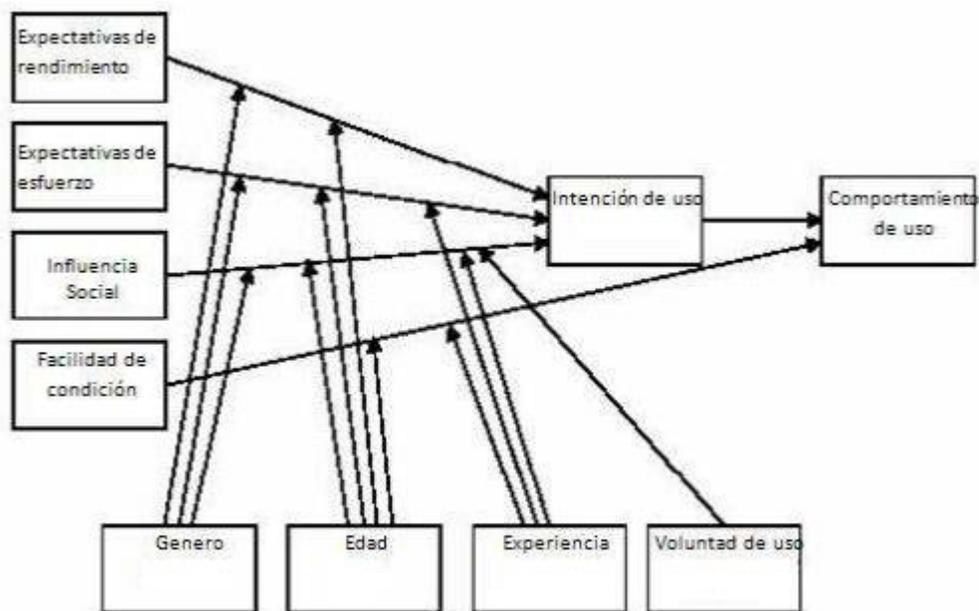
Las teorías que se tomarán en cuenta son las basadas en el modelo TAM (Technology Acceptance Model), siendo estas UTAUT (Unified Theory of Acceptance and Use of Technology), Theory of Reasoned Action (TRA) y TAM (Technology Acceptance Model).

1.1.1. Teoría unificada de la aceptación y uso de la tecnología

Esta teoría también es conocida por sus siglas en inglés UTAUT (Unified Theory of Acceptance and Use of Technology). Esta teoría tiene como objetivo explicar las intenciones de los usuarios al utilizar una nueva tecnología y el comportamiento de uso posterior. La teoría se basa en cuatro estructuras fundamentales: 1) La esperanza de desempeño, 2) la esperanza de lo posible, 3) la influencia social, y 4) facilitar las condiciones, las primeras tres son determinantes directos de la intención de uso y el comportamiento y el cuarto un determinante directo de la conducta de consumo. EL género, la edad, la experiencia y la voluntariedad de uso se postulan para medir el impacto de las cuatro estructuras fundamentales en la intención de uso y el comportamiento.

La teoría fue desarrollada a través de una revisión y consolidación de ocho modelos de investigación anterior se había empleado para explicar el comportamiento de uso de sistemas de información.

Figura 1. **Teoría unificada de la aceptación y uso de la tecnología
UTAUT**



Fuente: Venkatesh et al. Diagram/schematic of theory.

http://istheory.byu.edu/wiki/Unified_theory_of_acceptance_and_use_of_technology.

[Consulta: 5 de septiembre de 2013].

1.1.2. Teoría de la acción razonada

La teoría de la acción razonada conocida por sus siglas en inglés TRA (Theory of Reasoned Action), propone que el comportamiento individual es dirigido por las intenciones de comportamiento y están en función de la actitud de un individuo hacia el comportamiento y las normas subjetivas que rodean el desempeño de la conducta.

La actitud hacia la conducta se define como sentimientos positivos o negativos que la persona tiene sobre la realización de una conducta. Se determina a través de una evaluación de las creencias de uno con respecto a las consecuencias que pudieran derivarse de un comportamiento y una evaluación de la conveniencia de estas consecuencias.

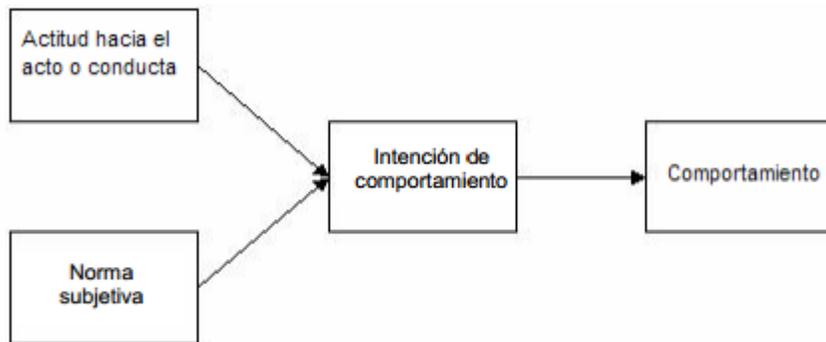
Formalmente, la actitud en general puede evaluarse como la suma de las evaluaciones de conveniencia y consecuencias individuales evaluadas para todas las consecuencias esperadas de la conducta.

La norma subjetiva se define como la percepción de un individuo de si las personas importantes para el individuo piensan que la conducta debe ser realizada. La contribución de la opinión de un referente dado se pondera por la motivación que un individuo tiene que cumplir con los deseos de ese referente. Por tanto, la norma subjetiva general se puede expresar como la suma de la percepción individual y la motivación evaluada por todos los referentes relevantes.

El modelo tiene algunas limitaciones, como un importante riesgo de confusión entre las actitudes y normas ya que las actitudes a menudo pueden ser reformuladas como normas y viceversa.

Una segunda limitación es el supuesto de que cuando alguien forma una intención de actuar, van a tener la libertad de actuar sin limitación. En la práctica, las restricciones tales como la capacidad limitada, el tiempo, los límites ambientales o de organización y los hábitos inconscientes limitarán la libertad de actuar.

Figura 2. **Teoría de la acción razonada (TRA)**



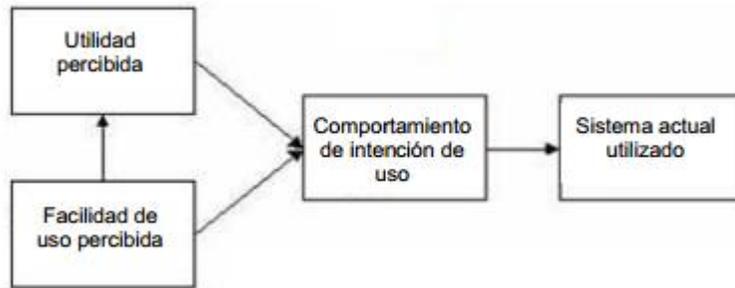
Fuente: Fishbein, Ajzen and Fishbein, Fishbein and Ajzen. *Diagram/schematic of theory*.
http://istheory.byu.edu/wiki/Theory_of_reasoned_action. [Consulta: 5 de septiembre de 2013].

1.1.3. Modelo de aceptación de tecnología

TAM es una adaptación de la Teoría de la Acción Razonada (TRA) en el campo de los sistemas informáticos. TAM postula que la utilidad percibida y la facilidad de uso percibida determinan la intención de un individuo para utilizar un sistema con la intención de uso sirviendo como un mediador del uso actual del sistema. La utilidad percibida también se ve directamente afectada por la facilidad de uso percibida. Los investigadores han simplificado TAM mediante la actitud de construir en la teoría de la acción razonada de la especificación actual. Los intentos de ampliar TAM han tomado generalmente uno de los tres métodos: mediante la introducción de factores de los modelos relacionados, mediante la introducción de factores de creencias adicionales o alternativos y mediante el examen de antecedentes y los asesores de utilidad percibida y la facilidad de uso percibida.

TRA y TAM, ambos tienen fuertes elementos de comportamiento, suponen en que cuando alguien forma una intención de actuar, es libre de actuar sin limitación. En la práctica existen limitaciones como la limitada capacidad o habilidad, el tiempo, los límites ambientales u organizacionales y los hábitos inconsistentes limitarán la libertad de actuar.

Figura 3. **Modelo de aceptación de una tecnología**



Fuente: Davis. http://istheory.byu.edu/wiki/Technology_acceptance_model. Diagram/schematic of theory. [Consulta: 5 de septiembre de 2013].

1.2. **Teoría y la relación con la tecnología escogida**

El proyecto de “Guate Extendida” surge por la necesidad de conocer Guatemala y sus bellos sitios turísticos, la forma de atraer el turismo será por medio de una aplicación fácil de utilizar, desarrollada para el Sistema Operativo Android. Debido a que es una aplicación fácil de utilizar y atrae a muchas personas ya que esta estará al alcance de cualquier persona que posea un Smartphone con este sistema.

TAM se basa en que la facilidad de uso percibida y la utilidad percibida generan un comportamiento de uso.

Lo que se pretende con la aplicación es que muchas personas la utilicen y puedan extender el contenido de esta, subiendo fotografías de sitios turísticos que las personas conozcan y con ello compartir su conocimiento y lograr que otras personas la utilicen ya que se pueden sentir identificados con el lugar.

La aplicación es sencilla e intuitiva para que el usuario se sienta cómodo utilizándola. Esta facilidad de utilización como dice la teoría tiene como resultado una utilidad de la aplicación, ya que el usuario notará el beneficio que obtiene, pues conocerá más lugares de Guatemala y podrá sentirse identificado con la aplicación ya que verá fotos que otros usuarios han compartido.

Como resultado de esta facilidad de uso y de la utilidad que se percibe de la aplicación se obtendrá una intención de uso de óptimo.

Si bien TAM se basa en la utilidad individual de la aplicación, UTAUT toma otros factores que afectan en la utilización de la aplicación, entre ellos factores individuales como la edad, género, experiencia, etcétera. Pero también toma en cuenta factores del entorno como las expectativas del esfuerzo y la influencia social, esto para generar una intención de uso y lograr un comportamiento de uso posterior.

En el caso de la aplicación “Guate extendida” se toman en cuenta factores individuales como la edad y la experiencia que tiene sobre haber vivido en cierto lugar y pueda contribuir con la aplicación compartiendo lugares conocidos para lograr una influencia social y que otras personas puedan conocer el lugar. Logrando con ello generar un comportamiento de uso posterior de la aplicación.

2. IDENTIFICACIÓN DE PROBLEMA Y SOLUCIÓN QUE LA APLICACIÓN REALIZARÁ

El turismo es una de las actividades económicas y culturales más importantes con las que cuenta Guatemala. Se entiende por turismo todas aquellas actividades que tengan que ver con conocer o disfrutar de ciertas regiones donde uno no vive permanentemente.

La importancia del turismo reside en dos pilares principales. El primero es aquel que tiene movimiento y la reactivación económica que genera en la región específica en la que se realiza. Así el país cuenta con una actividad económica más que genera empleos, obras de infraestructura, etcétera. Por otro lado, el turismo tiene otro pilar que se puede denominar sociocultural. Aquí, la importancia del turismo residirá en el hecho de que es a través suyo que el ser humano puede conocer de cerca y en vivo otras culturas, otras sociedades, otras formas de vivir y otros ambientes geográficos.

Actualmente hay en Latinoamérica 600 millones de celulares, de los cuales 200 millones son teléfonos inteligentes. Según Pyramid Research la venta de teléfonos inteligentes en América Latina debe avanzar en promedio un 30 % anual unos 60 millones cada año.

Hacia el 2014, IDC espera que en Latinoamérica, las ventas alcancen los 100 millones de unidades por año y que los precios de los aparatos bajen a la mitad de los que registran en la actualidad.

Dada la importancia que tiene el turismo y la creciente utilización de tecnologías móviles, se desea desarrollar la aplicación “Guate extendida”.

Donde el objetivo será dar a conocer los sitios turísticos de Guatemala y donde los usuarios puedan aumentar la información con fotografías de los sitios que conozcan.

2.1. Mercado objetivo

La aplicación “Guate Extendida” está dirigida a cualquier usuario que tenga un Smartphone con Sistema Operativo Android y quiera conocer Guatemala o dar a conocer algún sitio turístico a través de fotografías.

Cualquier persona con un teléfono inteligente o Smartphone con Sistema Operativo Android 3.1 en adelante puede utilizar la aplicación y ver los sitios preestablecidos en la aplicación, estos sitios son recopilaciones de internet sobre sitios turísticos de San Marcos.

Si los usuarios poseen una conexión a internet, podrán subir fotografías de sitios turísticos que conozcan y con ello aumentar la base de datos del sistema. Para subir fotografías los usuarios deberán poseer un perfil en Facebook, con el cual se deberán loguear previamente en la aplicación para subir una fotografía.

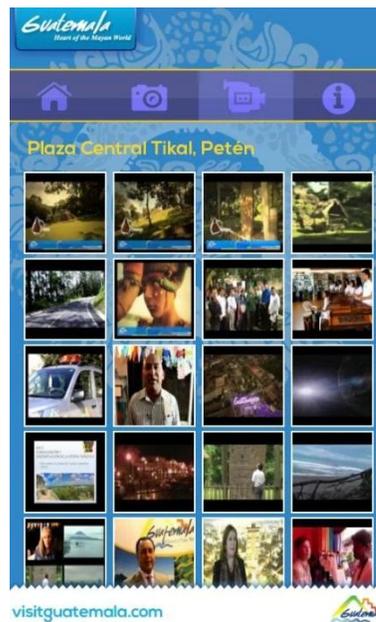
Si el usuario posee una conexión a internet podrá ver fotografías que otros usuarios subieron a la aplicación.

2.2. Benchmark de la aplicación

Actualmente no existen muchas aplicaciones para Android, las cuales estén enfocadas al turismo de Guatemala.

En Google Play existe una aplicación llamada “Descubre Guatemala” la cual tiene la funcionalidad similar a “Guate Extendida”. En esta aplicación es posible ver información y fotografías de distintos sitios turísticos de Guatemala. Esta aplicación contiene la mayoría de departamentos de Guatemala pero está limitada al contenido previamente cargado en la aplicación, impidiendo interacción con los usuarios.

Figura 4. **Aplicación descubre Guatemala**



Fuente: .Fox Networks. Descubre Guatemala.

[https://play.google.com/store/apps/details?id=com.foxnetworks.inguat.descubre&hl=es.](https://play.google.com/store/apps/details?id=com.foxnetworks.inguat.descubre&hl=es)

[Consulta: 16 de septiembre de 2013].

A diferencia de la aplicación “Descubre Guatemala”, “Guate extendida” permite una interacción con los usuarios permitiéndoles subir fotografías de lugares conocidos. Al subir la fotografía la aplicación guarda las coordenadas GPS donde fue tomada la fotografía, para que pueda ser ubicada la fotografía en Google Maps y con ello saber donde está ubicado el lugar.

3. DISEÑO DE LA APLICACIÓN BAJO LA NECESIDAD IDENTIFICADA

3.1. Prototipo

El prototipo de la aplicación muestra una serie de capturas de pantalla de lo que será la aplicación ya terminada, en estas capturas se observan las diferentes pantallas que tiene la aplicación como la galería, información, tomar fotografía, galería y Google Maps.

3.1.1. Galería

Pantalla donde se muestra el prototipo, cuando el usuario vea la galería de fotos subida por otros usuarios, esta galería se le denomina galería *online* ya que permite ver las diferentes fotografías que subió el usuario y un enlace para ver que pueda ver la ubicación en un mapa.

Figura 5. **Prototipo galería**

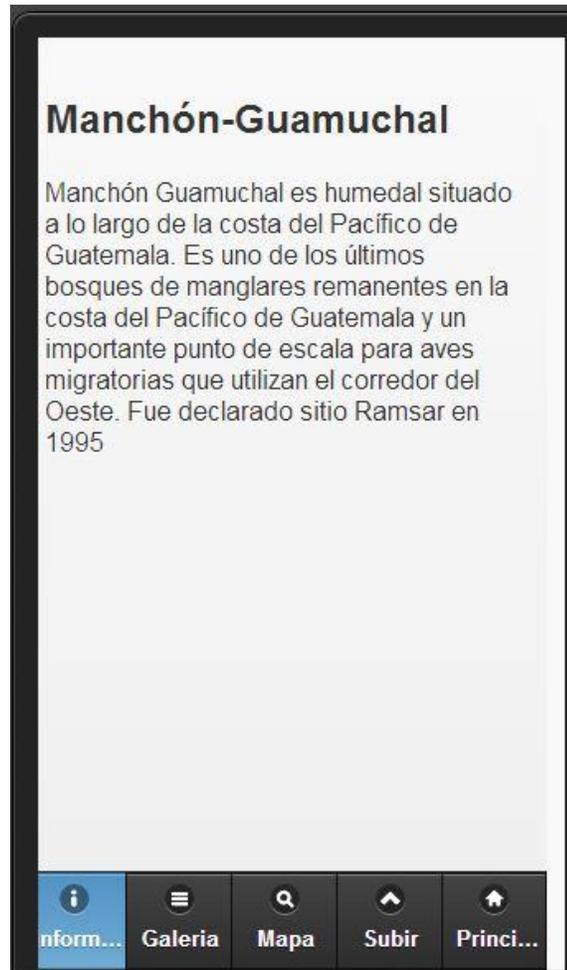


Fuente: elaboración propia.

3.1.2. Información del sitio turístico

Pantalla donde se muestra información del sitio turístico, seleccionado en la aplicación, esta es información ya cargada en la aplicación la cual describe de qué se trata el lugar y el nombre.

Figura 6. **Prototipo información**

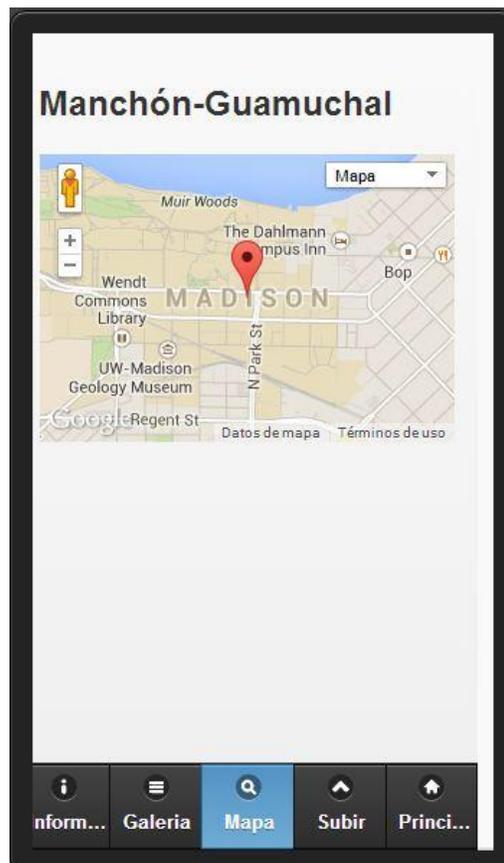


Fuente: elaboración propia.

3.1.3. Google Maps

Al momento de ver una fotografía en la galería, el usuario tendrá la opción de ver dónde fue tomada la fotografía y se mostrará en Google Maps la ubicación, al mostrar el mapa este tendrá una marca en las coordenadas exactas donde fue tomada la fotografía.

Figura 7. Prototipo Google Maps

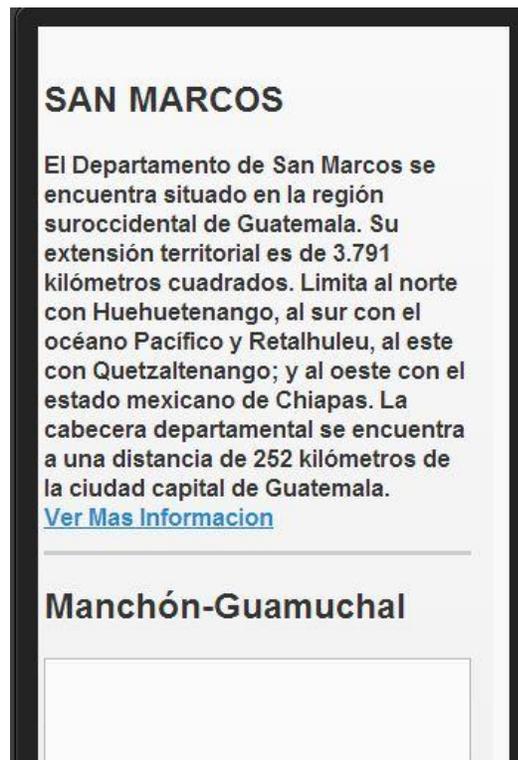


Fuente: elaboración propia.

3.1.4. Galería de fotografías de un sitio precargado

Cada sitio precargado en la aplicación posee su propia galería de imágenes, esta pantalla muestra información del sitio con una galería. Estas imágenes sirven de guía para que los usuarios puedan observar algunos lugares ya conocidos del departamento.

Figura 8. Prototipo galería local

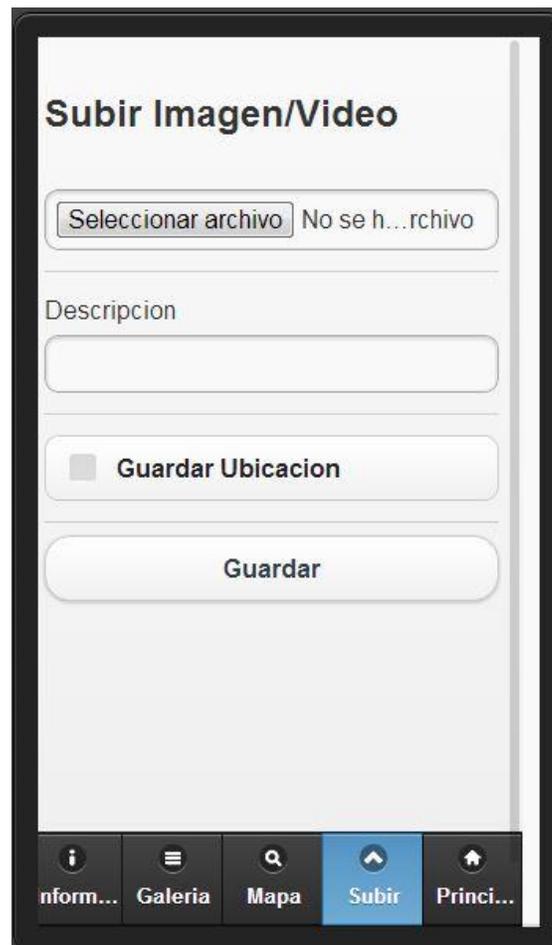


Fuente: elaboración propia.

3.1.5. Subir imagen

La aplicación permite tomar una fotografía y subirla a la aplicación para que pueda ser vista por otros usuarios al momento de tomar la fotografía la aplicación guardará la ubicación donde fue tomada para que se pueda mostrar a otros usuarios.

Figura 9. Prototipo subir imagen



Fuente: elaboración propia.

3.2. Validaciones, formularios

En la aplicación existen distintos formularios, donde se interactúa el usuario con la aplicación, a continuación se describen los formularios que forman parte de la aplicación “Guate Extendida”.

3.2.1. Subir fotografía

El formulario subir fotografía cuenta con todos los elementos que permiten al usuario tomar una fotografía, para posteriormente subir la fotografía a la aplicación y que otros usuarios puedan verla.

Al inicio al abrir la aplicación únicamente se mostrará la opción de tomar fotografía. Esto para forzar al usuario a que tome una fotografía antes de que presione el botón subir y provoque un error en la aplicación.

Cuando el usuario tome la fotografía, se habilitará el formulario con una vista previa de la imagen que capturo y un campo donde escriba una descripción de la fotografía que está tomando. Al finalizar con la descripción, el usuario puede presionar el botón “subir” para que la fotografía sea subida al sistema.

El usuario debe haber iniciado sesión en Facebook antes de poder subir una fotografía, de lo contrario se le mostrará un mensaje informándole que debe iniciar sesión y no podrá subir la fotografía.

Al tomar la fotografía la aplicación automáticamente guardará el nombre de usuario en Facebook con el que está logueado y la ubicación GPS donde fue tomada la fotografía.

La descripción de la fotografía no es necesaria y puede dejarse vacío el campo.

Al subir la fotografía se tendrá que esperar un momento en lo que sube al sistema (dependiendo de la conexión a internet y la calidad de la imagen) y si el proceso fue exitoso, se mostrará un mensaje indicando que la fotografía fue subida con éxito. De lo contrario se mostrará un mensaje de que ocurrió un error para que se vuelva a intentar.

3.2.2. Registro

La opción de registro permite iniciar sesión con su usuario de Facebook dentro de la aplicación.

Para registrarse es necesario ir a la opción de registro en el menú principal, en la pantalla principal aparece un botón en el cual si se presiona, procederá el proceso para registrarse con Facebook. Se pedirán los datos del usuario y una confirmación de permisos.

Si el proceso de registro es exitoso debería aparecer la foto de perfil en la pantalla principal.

En cualquier momento se puede cerrar sesión presionando registro en el menú principal y posteriormente cerrar sesión.

3.2.3. Galería

En la opción de galería se muestran las fotografías que han subido otros usuarios a la aplicación.

Para ver las fotografías hay que ir a la opción en el menú desplegable y se mostrará una pantalla con unos botones en la parte superior, en los cuales se encuentra un botón con una flecha hacia adelante donde el usuario podrá ver las fotografías en el sistema. En la parte superior se mostrará el usuario que subió la fotografía, en la parte central se muestra la fotografía y en la parte de abajo la descripción de la fotografía.

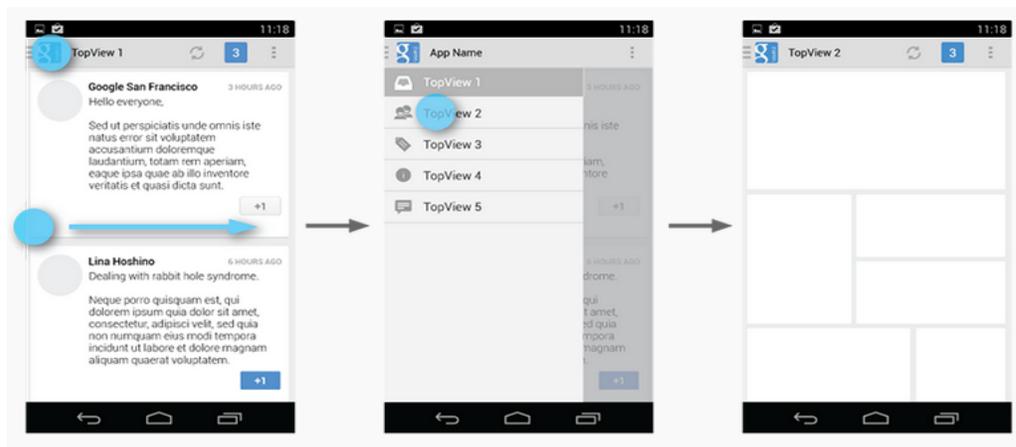
Arriba en la barra se muestra un botón de mostrar ubicación en el cual al darle *click* mostrará en pantalla la ubicación donde fue tomada la fotografía.

3.3. Diseño intuitivo y usabilidad

El diseño de la aplicación está basado en un menú desplegable en la parte izquierda de la aplicación donde se encuentran todas las opciones, para realizar dentro de la aplicación, este tipo de menús se conocen como Navigation Drawer.

Este tipo de menús de navegación ya llevaban tiempo utilizándose en numerosas aplicación, pero se hacían gracias a librerías externas que implementaban este componente, pero todas estas eran distintas e incoherentes entre si, tanto en diseño como en funcionalidad. Por ello Google ha querido acabar con este comportamiento aportando su propia implementación de este componente y definiendo guías diseño de la plataforma.

Figura 10. Navigation Drawer



Fuente: Creative Commons Attribution 2.5. Navigation Drawers.

<http://developer.android.com/design/patterns/navigation-drawer.html>.

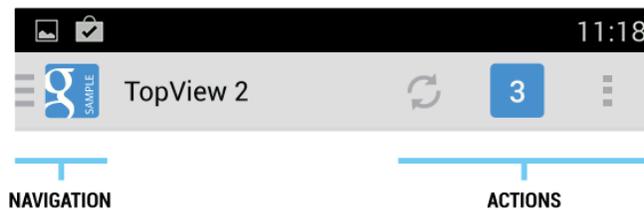
[Consulta: 19 de septiembre de 2013].

El navigation drawer está disponible como parte de la librería Android support.

Cada menú de la aplicación tiene personalizado su menú de acción, dependiendo de la opción del menú que se esté mostrando.

En la siguiente imagen se puede ver el estándar utilizado para el menú donde del lado izquierdo está el menú desplegable y del lado derecho están las acciones personalizadas para cada menú.

Figura 11. **Actions**



Fuente: Creative Commons Attribution 2.5. Navigation Drawers.

<http://developer.android.com/design/patterns/navigation-drawer.html>. [Consulta: 19 de septiembre de 2013].

Cada una de las pantallas posee un mismo fondo de pantalla para que la aplicación sea amigable a la vista y atractiva para el usuario.

Los temas en Android son mecanismos para aplicar un estilo consistente a la aplicación o a un activity. Los estilos especifican las propiedades visuales de los elementos los cuales maquillan la interfaz del usuario, tales como el color, alto y color de letra. Esto para que todas las aplicaciones tengan coherencia en la plataforma. Android provee dos temas para el desarrollo de aplicaciones:

- Holo Light
- Holo Dark

El tema utilizado en la aplicación es Holo Dark, es para continuar con los estándares de Android para el desarrollo de aplicaciones.

4. DOCUMENTACIÓN Y TUTORIAL DE PROGRAMACIÓN DE LA APLICACIÓN

4.1. Requisitos

Se describen todos los requisitos para desarrollar esta aplicación, se mencionan requisitos de *software*, por ejemplo, las herramientas y el entorno de desarrollo necesario, además de los requisitos de *hardware*.

4.1.1. Herramientas

Para el desarrollo de la aplicación se utilizó el Android SDK, el cual provee de las librerías y herramientas de desarrollo necesarias para compilar, probar y debuguear la aplicación.

Las herramientas que se utilizaron se describen a continuación:

- Eclipse + ADT Plugin
- Android SDK tools
- Android Platform-tools
- La última plataforma de Android
- La última imagen del sistema operativo Android para el emulador

4.1.2. Software

- PHP

Lenguaje utilizado en los servicios web.

- Apache

Servidor web con apache donde se alojaran los servicios web.

- MYSQL

Servidor de base de datos MYSQL alojado en la nube, el cual su funcionalidad será el almacenar la información correspondiente a la aplicación fotos e información de usuarios.

- SQLite

Sistema gestor de base de datos que se utiliza para guardar información sobre los distintos lugares turísticos que manejará la aplicación de manera local.

- Google Maps V2

Servicio de ubicación por medio de Google Maps, esto se utilizará con el fin de colocar la ubicación de ciertas fotografías identificando el lugar del cual proviene la fotografía.

- API de Facebook

API de Facebook el cual se utilizará para el manejo de usuarios, el usuario se loguea en la aplicación por el API de Facebook y así se llevará el control de los usuarios y el contenido que publiquen. Además de compartir fotos en dicha red social.

- Sistema Operativo

Sistema Operativo Android, este será el sistema base de la aplicación.

4.1.3. Hardware

Para el desarrollo de la aplicación se recomienda al menos un procesador Intel Core 2 Duo de 2.00 Ghz, 3Gb de memoria RAM y al menos 600 Mb para el SDK y al menos 100 MB para cada plataforma adicional utilizada.

4.1.4. Tutorial de desarrollo y referencias

A continuación se describe un tutorial de como fue realizado el proyecto de Guate Extendida para el Sistema Operativo Android.

Primero se describe cómo configurar el entorno de programación de la aplicación.

4.1.4.1. Configurar el entorno de desarrollo

Para comenzar el desarrollo de la aplicación es necesario empezar por descargar e instalar eclipse + el plugin necesario para desarrollar en Android. Para ello es necesario ir a la página de desarrolladores de Android (<http://developer.android.com/sdk/index.html>) y descargar el ADT bundle para

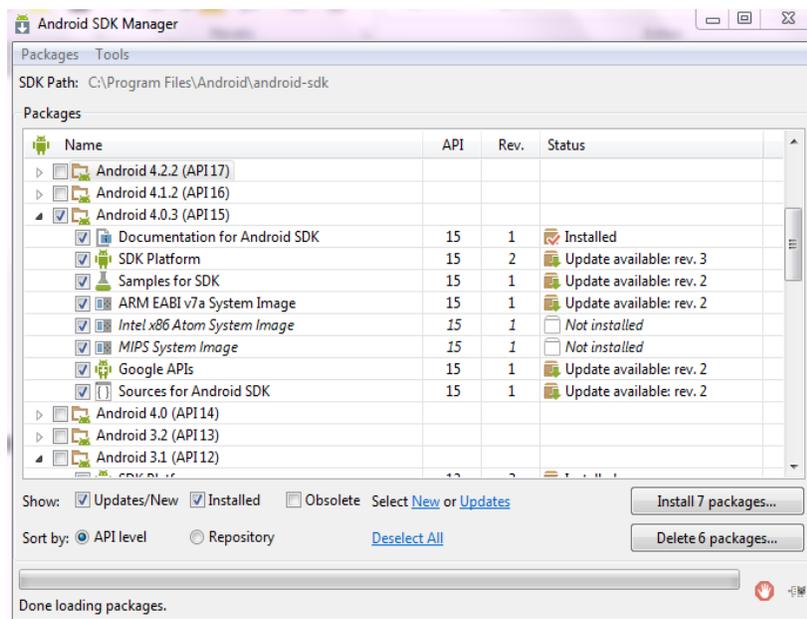
Windows. El bundle contiene todo lo necesario para que se pueda comenzar a trabajar.

- Eclipse + ADT plugin
- Android SDK Tools
- Android Platform-tools

Con eclipse instalado se necesita descargar la plataforma con la que se compilará el código del proyecto, en este caso se descargará Android 4.0.3 api level 15.

Para ello abrir el SDK y seleccionar el paquete necesario como se ve en la figura.

Figura 12. **Android SDK**



Fuente: elaboración propia.

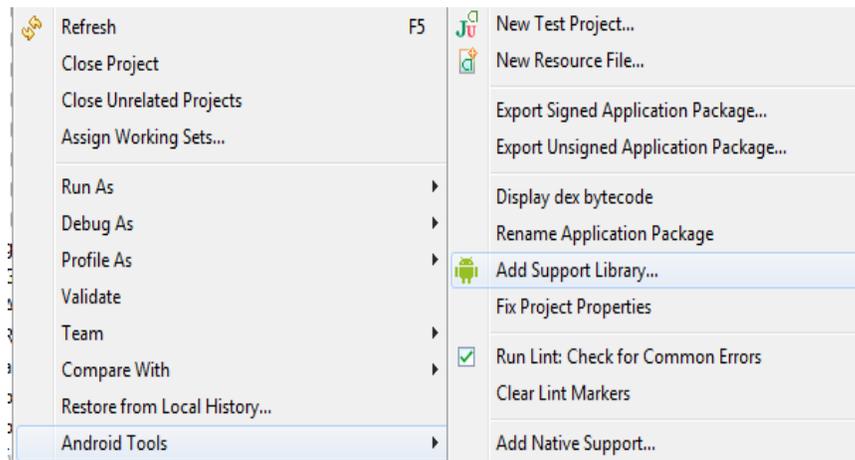
Después de tener instalada la plataforma 4.0.3 se crea un nuevo proyecto con el nombre “Guate Extendida”. Se selecciona el target y min SDK el api level 15 con el api de Google. Se selecciona el tema Holo Dark para la aplicación.

4.1.4.2. Menú principal

Para desarrollar el menú principal, de la aplicación se utiliza Navigation Drawer. Este consiste en un panel al lado izquierdo de la aplicación el cual posee todas las opciones a realizar dentro de la aplicación.

La librería Support es necesaria para que funcione el navigation drawer, para descargarla es necesario dar *click* derecho en el proyecto ir a AndroidTools y presionar la opción “Add Support Library”.

Figura 13. **Support Library**



Fuente: elaboración propia.

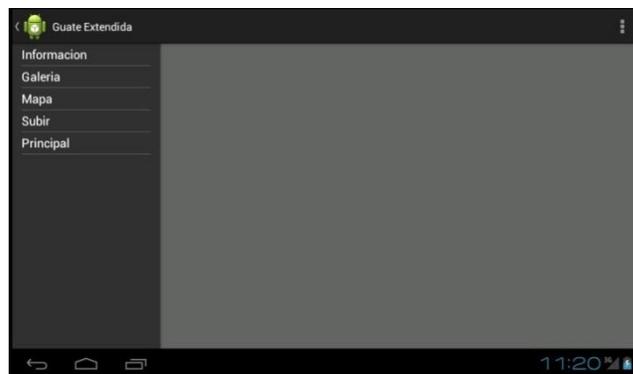
Navigation Drawer consiste en un activity principal el cual consta de un listview y un fragment de contenido. El listview representa las opciones del panel al lado izquierdo de la aplicación y el fragment de contenido es el que se cambia dependiendo de la opción seleccionada en el panel izquierdo. Cada vez que una opción del panel izquierdo es seleccionada se obtiene su posición y se realiza un remplazo del fragment por el nuevo seleccionado.

El activity principal debe heredar de “FragmentActivity” en lugar del clásico “Activity” esto debido a que este activity está conformado por distintos fragments.

Como se mencionó el xml del main activity debe contener un listview y un FrameLayout.

Cada clase que se desea mostrar en la aplicación debe heredar de Fragment y la clase en el método onCreateView debe inflar el xml correspondiente a la clase que se necesita.

Figura 14. **Navigation Drawer**



Fuente: elaboración propia.

4.1.4.3. Tomar fotografías

El *framework* de Android incluye soporte para cámaras de distintos dispositivos, permitiendo la captura de imágenes en la aplicación.

El *framework* de android puede capturar imágenes a través de la API o a través de un Intent. En la aplicación se utiliza un Intent para capturar las imágenes.

Se utiliza un Intent ya que es la forma más sencilla para tomar una fotografía sin mucho código sólo invocando la aplicación camara de Android. Un Intent camara hace una petición para realizar una imagen a través de la cámara app existente y luego retorna el control a la aplicación.

El procedimiento para utilizar el Intent Cámara se describe a continuación:

- Crear un Intent que realice una petición a una imagen con el siguiente tipo de intent "MediaStore.ACTION_IMAGE_CAPTURE"
- Se crea un archivo en la memoria del dispositivo utilizando el método Environment.getExternalStorageDirectory() para obtener la ruta y se guarda un archivo temporal llamado pic.jpg. El cual se utiliza para guardar temporalmente la fotografía tomada.
- Utilizar el método startActivityForResult() para ejecutar el intent cámara. Después de iniciar el intent, la aplicación de la cámara mostrara en el dispositivo y el usuario podrá tomar la fotografía.
- Declarar el método onActivityResult() en la aplicación para recibir la respuesta e información del intent cámara. Cuando el usuario termine de tomar la fotografía o cancele la operación, el sistema llamara este

método. Si el usuario toma la fotografía y el proceso es exitoso, se procede a tomar su ubicación por medio de gps.

4.1.4.4. GPS

Obtener la posición del usuario en Android funciona por medio de una llamada. Primero el usuario indica que desea recibir actualizaciones de ubicación de la clase "Location Manager" llamando "requestLocationListener", pasándole de parámetro un LocationListener. El location Listener debe implementar varios métodos de devolución de llamada que el gestor de localización llama cuando la ubicación del usuario cambia o cuando el estatus del servicio cambia.

Primero es necesario instanciar un objeto de la clase "Location Manager", con este objeto se realiza la llamada al método requestLocationUpdates el cual es un método que indica que se está solicitando una actualización de la posición en ese momento se le envía de parámetro cada cuánto tiempo se desea que vuelva a intentar actualizar la información en este caso será cada 30 000 milisegundos, el otro parámetro es un listener el cual cuando ubique la posición llamará el método onLocationChanged (Location location) el cual guardará la ubicación y finalizará el listener para que no siga intentando ubicar una posición, esto lo realiza a través de la llamada al método removeUpdates (listener).

4.1.4.5. Servidor web

La aplicación da una opción para subir fotografías de sitios turísticos conocidos en San Marcos, para subir las imágenes y guardarlos se utiliza un servidor web Apache con Php y Mysql.

En Mysql se tiene una base de datos con una tabla que se utiliza para almacenar la información de las imágenes guardadas. La tabla se describe a continuación.

Tabla I. **Tabla Imagen**

Nombre Campo	Tipo Dato	Descripción
idImagen(PRIMARY KEY)	INT(AutoIncrement)	Identificador único de cada registro
Imagen	VARCHAR	Nombre de la imagen que se almacena en el servidor
Descripción	VARCHAR	Descripción de la imagen tomada por el usuario
Latitud	VARCHAR	Coordenada latitud tomada desde el Gps
Longitud	VARCHAR	Coordenada longitud tomada desde el Gps
Usuario	VARCHAR	Nombre del usuario que subió la imagen

Fuente elaboración propia.

En el servidor existen 4 páginas php que se detallan a continuación:

Config.php en esta página se detallan las credenciales para conectarse a la base de datos mysql, en esta se define el nombre o ip del host, el usuario, el password y el nombre de la base de datos para conectarse.

Connectbd.php en esta página se realiza la conexión con la base de datos utilizando la página config.php y se retorna el objeto de la conexión.

funciones_bd.php en esta página se declaran dos métodos guardar foto y leer foto. El método guardar foto recibe como parámetro el nombre de la foto, descripción, latitud, longitud y nombre de usuario. Todos estos parámetros los guarda en la base de datos en la tabla imagen. El método leer foto recibe el índice de la imagen que se desea obtener y devuelve la información de la imagen en formato Json.

Access.php, esta página es la que es llamada por la aplicación al momento en que se desea guardar una imagen, en ella se leen todos los parámetros enviados desde la aplicación como el nombre de la imagen, descripción, latitud, longitud y usuario. Posteriormente se lee el archivo binario que viene de la aplicación y se guarda en el servidor en la carpeta "imágenes", si el proceso fue exitoso se hace una llamada al método guardar foto de la página funciones_bd.php.

Acces2.php esta página es llamada por la aplicación al momento que se abre la galería y se desea ver las fotos subidas por los usuarios. En esta página se recibe el parámetro index y se llama la función leer foto de la página funciones_bd.php.

4.1.4.6. Subir fotografía

El proceso de subir la fotografía al servidor se describe a continuación:

Camara.java en esta clase se define el método onClick del botón subir, en este método se declara un ByteArrayOutputStream en el cual se comprime la fotografía que se tomo y es convertido en un array de bytes, también se toma la descripción, latitud y longitud y todos los parámetros se envían a la clase TareaAsync para que se realice la tarea de subir el archivo en segundo plano y con ello no afectar el hilo de ejecución principal de la aplicación.

TareaAsync.java es una clase que hereda de AsyncTask la cual es una clase que se utiliza para realizar tareas en segundo plano al llamar a su método doInBackground. Cuando esta clase es llamada y se ejecuta el método execute y el primer parámetro es 1, indica que se desea subir una imagen al servidor, el método doInBackground atiende esta petición y crea un arraylist con los parámetros y hace una llamada al método httpPostConnect, este método crea un objeto MultipartEntity al cual se le añade la imagen que se tomo y esta guardada con el nombre "Pic.jpg", también se agregan los demás atributos de la imagen como el nombre de la imagen el cual será un identificador único generado con el método randomUUID de la clase java.util.UUID, la descripción, latitud, longitud y el id del usuario.

Cuando se termina de armar el objeto MultipartEntity se hace el post hacia el servidor en la pagina "Access.php" y se obtiene la respuesta.

4.1.4.7. Galería *online*

La galería permite ver fotografías que los demás usuarios han subido al servidor el proceso de desarrollo se describe a continuación.

Galería.java, esta clase hereda de fragment y solamente se infla cuando el usuario selecciona la galería del menú principal. Al momento de inflarla se carga el xml “galería” el cual contiene diferentes texview donde se muestra la descripción y el usuario que subió la fotografía. Además tiene un Image View que es el que muestra la fotografía tomada.

MainActivity.java, cuando el usuario selecciona galería se le muestran dos botones en el action bar de la aplicación, el primero tiene una imagen de un mapa y el segundo tiene una imagen de una flecha siguiente. Cuando el usuario desea ver la siguiente foto de la galería presionará la flecha siguiente, este método al ser un botón del ActionBar está definido en el mainActivity.java como onOptionsItemSelected, este método instancia un objeto de la clase TareaAsync y ejecuta el método execute para ejecutar el método en segundo plano. Cuando el usuario presiona el botón de mostrar ubicación este crea un *fragment* de la clase Mapa.java y lo reemplaza en el fragment principal.

Mapa.java esta clase hereda de Fragment, en la función onCreateView infla el xml que tiene declarado un *fragment* de la clase “com.google.android.gms.maps.SupportMapFragment” el cual contendrá el mapa donde se mostrará la ubicación de la fotografía esto utilizando Google maps. Esta clase tiene un método llamado ubicar el cual el *fragment* es mostrado obitene el mapa del xml y lo almacena en una variable de la clase GoogleMap.

Con esta variable llama el método `animateCamera` para mover la cámara a la posición de latitud y longitud dada de la fotografía setenado un zoom de 16.0f. Además agrega una marca en la posición indicada.

`TareaAsync.java` cuando esta clase es llamada desde el método del botón siguiente en el ActionBar ejecuta una acción en segundo plano donde forma un `MultipartEntity` con el índice de la fotografía que se desea mostrar hace la conexión con el servidor por la página "`Acces2.php`". Cuando obtiene la información del servidor este retorna un json con la información de la imagen. Posteriormente a esto se hace una petición "GET" de la imagen solicitada , esto a través de un `HttpURLConnection`, la imagen descargada se almacena en un `InputStream` que luego se pasa al dispositivo con el nombre "`downloadGTE.jpg`". Cuando el proceso se llevo con éxito se ejecuta el método `onPostExecute` el decodifica la imagen a un tamaño más pequeño para formar un bitmap de la imagen y que se pueda mostrar en la aplicación a través del `Imageview`.

4.1.5. Consideraciones de implementación

A continuación se describen diferentes consideración y buenas prácticas que se deben tener en cuenta al momento de desarrollar una aplicación para el Sistema Operativo Android y se utilicen herramientas como la captura de fotografías, mapas de Google y posicionamiento GPS.

4.1.5.1. Escalar imágenes

Uno de los grandes retos para los desarrolladores en Android es la innumerable cantidad de dispositivos que existen para Android, variando su RAM, el procesador, resolución de pantalla, etcétera.

Android maneja dos tipos de stack de memoria uno para las aplicaciones y otro para los gráficos. Cuando se desea convertir una imagen a bitmap esto lo almacena en la memoria de gráficos, funciona bien para una imagen de pocos kb, pero que pasa con las imágenes de 2MB o más, el proceso de convertir la imagen a un bitmap para mostrarlo en un ImageView consume demasiada memoria de gráficos y esta memoria está limitada dependiendo de cada dispositivo, si la memoria no es suficiente y la tarea se convierte en muy pesada para el dispositivo esto dará una excepción OutOfMemory. Para solucionar este problema es necesario redimensionar las imágenes antes de convertirlas. Para realizar esta tarea existen varias opciones.

Una de las opciones es cambiar el esquema de colores a RGB_565 para reducir el consumo de memoria pero la imagen perderá calidad también. Otra opción es escalar la imagen, con ello se logrará una disminución considerable del uso de memoria.

En la aplicación cuando se muestran las imágenes en la galería se tiene este problema ya que cuando se muestran las imágenes en el ImageView es necesario leer la imagen que se recibe del servidor y convertirla en un bitmap para mostrar, pero esta tarea era demasiado cargada para el dispositivo de prueba, por ello fue necesario utilizar el segundo método mencionado anteriormente donde se escala la imagen para mostrarla.

4.1.5.2. Async Task

Async Task es una clase abstracta proporcionada por Android que ayuda a utilizar el hilo de interfaz de usuario correctamente. Esta clase permite realizar operaciones largas en segundo plano y mostrar el resultado en el subproceso de interfaz de usuario sin tener que manipular los hilos.

Android implementa modelo de subproceso único y siempre que se inicia una aplicación Android, se crea un hilo. Suponiendo que se está haciendo una petición web en un clic de botón en la aplicación. El clic de botón realiza una solicitud al servidor y la respuesta será esperada. Debido al modelo de un solo hilo de Android, hasta que el tiempo de respuesta que esperaba la pantalla no responde. Así que se debe evitar la realización de operaciones de larga ejecución en el subproceso de interfaz de usuario. Esto incluye archivos y acceso a la red.

Para superar esto, se puede crear un nuevo hilo y aplicar método para realizar esta llamada web, así que la interfaz de usuario sigue siendo sensible.

AsyncTask consta de cuatro pasos:

- `doInBackground`: código de realizar la operación de larga duración va en este método. Cuando el método `onClick` se ejecuta al hacer clic con el botón, se llama al método que acepta los parámetros de ejecución y llama automáticamente al método `doInBackground` con los parámetros pasados.
- `onPostExecute`: este método se denomina método `doInBackground` después de completado el procesamiento. Resultado del `doInBackground` se pasa a este método.
- `onPreExecute`: este método se llama antes de que se llama al método `doInBackground`.
- `onProgressUpdate`: este método es invocado llamando `publishProgress` en cualquier momento desde `doInBackground`.

En la aplicación se utiliza esta clase en dos ocasiones la primera cuando se necesita subir una imagen al servidor y la segunda cuando se desea leer la imagen del servidor, en ambas situaciones se realizan conexiones hacia el servidor lo cual, si se hace desde el hilo principal de Android congelaría la aplicación, por ello es necesario realizar estas tareas en segundo plano utilizando esta clase que provee Android.

4.1.5.3. Multipart Entity

Existen diversas maneras para subir un archivo a un servidor remoto en Android, por ejemplo, ftp, http post o una solución openSource proporcionada por Apache Multipart Entity.

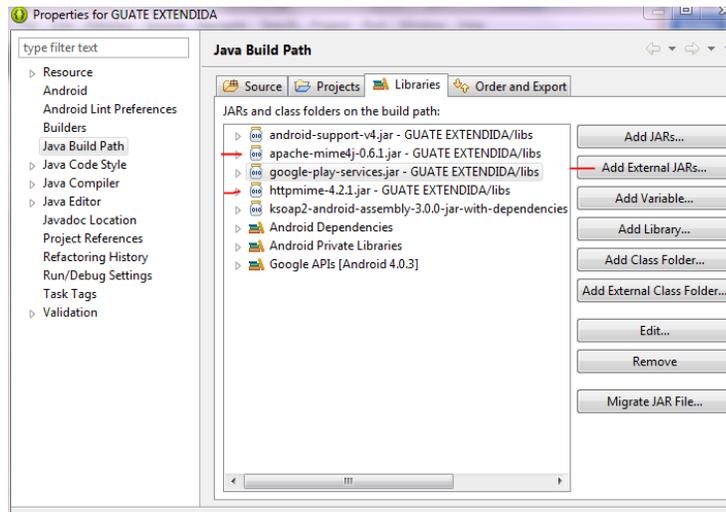
Carga de archivos y datos desde un dispositivo Android a un servidor no es un gran problema si se hace en un dispositivo con memoria alta. Pero si la memoria del dispositivo es baja y el tamaño del archivo que está subido es grande, entonces la aplicación para Android puede bloquearse en algún momento dando el error 'OutOfMemory'. Para evitar este error al cargar archivos a través del dispositivo Android, es necesario utilizar el cliente de Apache "Multipart Entity". Para utilizar este cliente es necesario para descargar los archivos jar y luego agregarlos al proyecto Android.

A continuación se describen los pasos para agregar los archivos del cliente Apache al proyecto Android.

- Descargar los jar necesarios para que funcione el cliente de Apache. Los jar utilizados son “httpmime-4.2.1.jar ” y “ apache-mime4j -0.6.1.jar”.
- Agregar los jar descargados a la carpeta “libs” del proyecto.
- Abrir el build Path del proyecto y agregar las librerías descargadas.
- En la pestaña “order and export” es necesario agregar las librerías.
- Realizar un “clean” del proyecto.

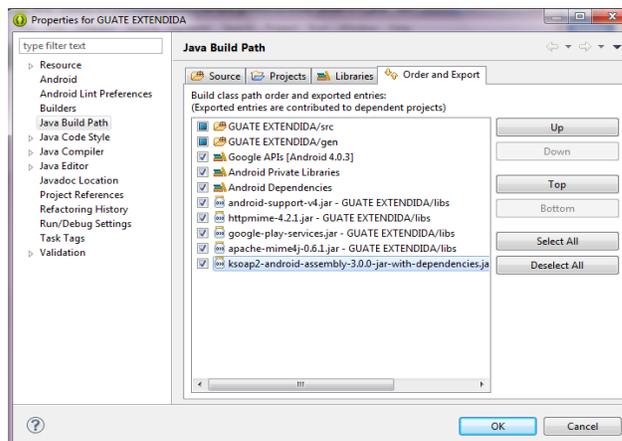
Estas librerías se utilizan en el proyecto al momento de subir imágenes al servidor.

Figura 15. **Agregar las librerías**



Fuente: elaboración propia.

Figura 16. **Seleccionar las librerías**



Fuente: elaboración propia.

4.1.5.4. GPS

Existen múltiples formas de obtener la localización de un dispositivo móvil, aunque la más popular es la localización por GPS, también es posible obtener la posición de un dispositivo por medio de las antenas telefónicas o mediante puntos Wi-Fi cercanos.

Para empezar con el GPS se debe aclarar que en Android no existe ningún método del tipo “obtenerPosicionActual()”.

Obtener la posición a través de un dispositivo de localización no es una tarea instantánea, sino que es un cierto tipo de procesamiento y espera que podría ser de segundos o minutos.

En la clase LocationManager lo más parecido que se puede encontrar para obtener la posición es el método getLastKnownLocation(String provider), que devuelve la última posición conocida del dispositivo. Es importante entender este método ya que devuelve la última posición conocida pero no se conecta a ningún proveedor en el momento para obtener la posición, si no que devuelve la última esta pudo ser hace minutos, hace horas o hace días.

Entonces la forma para obtener la posición gps del dispositivo se hace suscribiéndose al evento que se lanza cada vez que un proveedor recibe nuevos datos sobre la localización actual. Todo esto se realiza mediante la llamada al método requestLocationUpdates(), al que se debe pasar 4 parámetros.

- Nombre del proveedor de localización al que se quiere suscribir.
- Tiempo mínimo entre actualizaciones, en milisegundos.

- Distancia mínima entre actualizaciones, en metros.
- Instancia de un objeto LocationListener, que se tiene que implementar previamente para definir las acciones a realizar al recibir cada nueva actualización de la posición.

El listener es de tipo LocationListener y tiene unos métodos asociados estos son los siguientes, onLocationChanged(location), onProviderDisabled(provider), onProviderEnabled(provider) y onStatusChanged(provider, status, extras).

El método que se utiliza en la aplicación es onLocationChanged() que se ejecuta cada vez que se recibe una nueva localización desde el proveedor, entonces se guarda la posición donde fue tomada la fotografía y se le muestra al usuario que la ubicación GPS fue obtenida y ahora es posible subir la fotografía al servidor.

4.1.5.5. Manifest

Es importante definir una serie de permisos en el archivo manifest para que al momento de instalar la aplicación estos permisos sean concedidos y la aplicación pueda funcionar correctamente.

Entre ellos están el permiso para conectarse a internet “INTERNET”

Permiso para obtener su posición gps “ACCESS_FINE_LOCATION” y “ACCESS_COARSE_LOCATION”.

Permiso para utilizar la cámara del dispositivo “CAMERA” y “hardware.camera”

Permiso para leer y escribir en la memoria del dispositivo
WRITE_EXTERNAL_STORAGE y READ_EXTERNAL_STORAGE

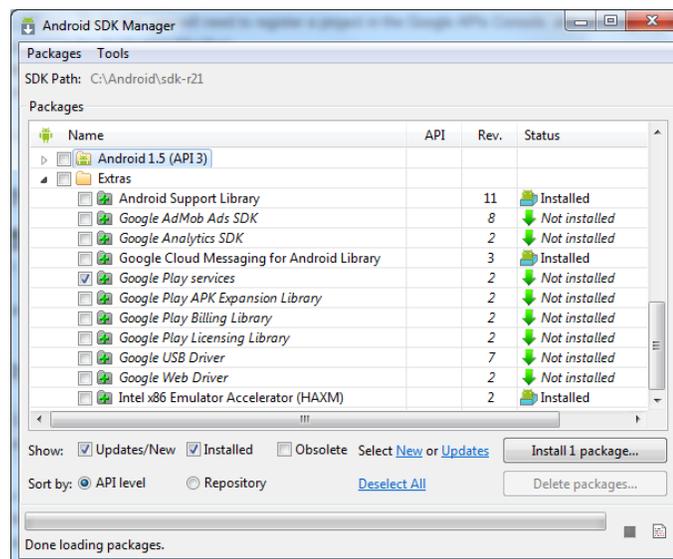
Y por último permisos para utilizar los mapas de Google MAPS_RECEIVE
y READ_GSERVICES.

4.1.5.6. Google Play

Para utilizar el API V2 de google play services es necesario configurar el
entorno.

Primero es necesario descargar el paquete llamado “Google Play
Services” desde el SDK Manager de eclipse.

Figura 17. Google Play Services

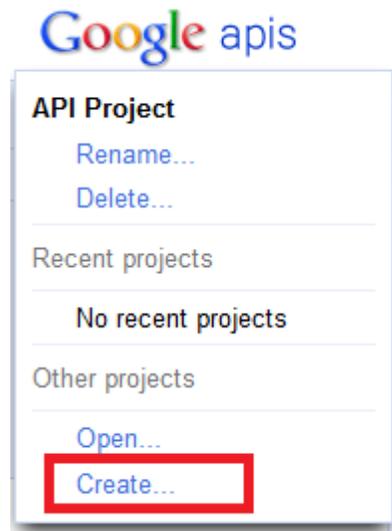


Fuente: elaboración propia.

El paquete quedará instalado en la ruta: <carpeta-sdk-android>/extras/google/google_play_services/.

El siguiente paso es obtener un API KEY en el servicio de mapas de Google en la aplicación. La nueva api se ha integrado a la consola de Apis de Google por lo que el primer paso es acceder a ella y crear un nuevo proyecto.

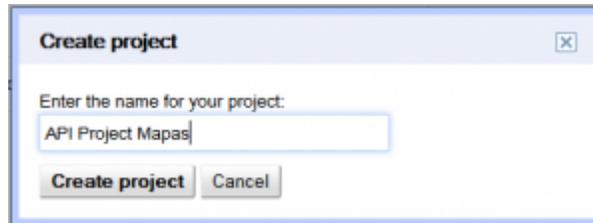
Figura 18. **Google Apis Console Create**



Fuente: elaboración propia.

Se introduce el nombre del proyecto.

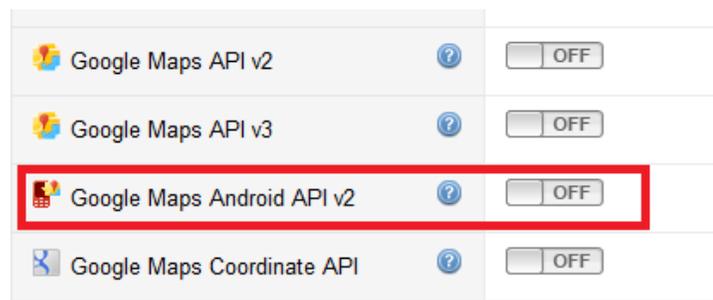
Figura 19. **Nombre del proyecto**



Fuente: elaboración propia.

Una vez creado el proyecto, hay que acceder a la opción “Services” del menú izquierdo y se activa el servicio “Google Maps Android API v2”.

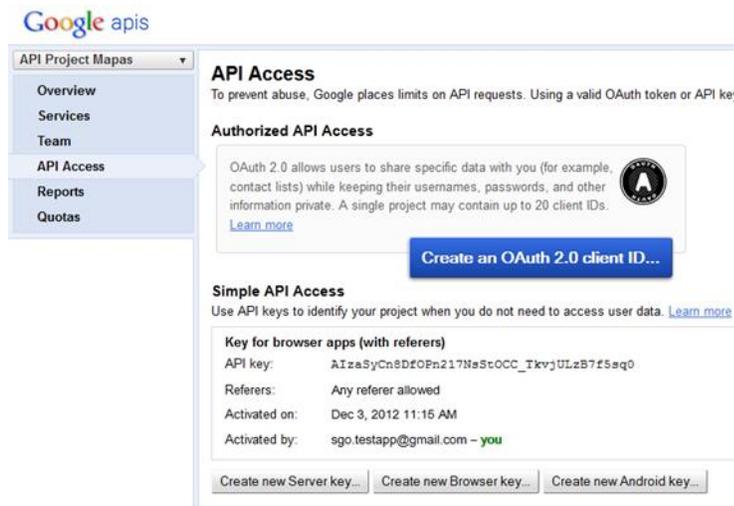
Figura 20. **Activar el servicio**



Fuente: elaboración propia.

Una vez activado el servicio aparecerá una nueva opción en el menú izquierdo llamado “API Access” accediendo a esta opción se puede obtener la nueva API KEY que permite utilizar los mapas en la aplicación Android.

Figura 21. API Access



Fuente: elaboración propia.

Para generar el API key se necesita firmar la aplicación con la keystore que provee eclipse para debuggar esta ruta se puede encontrar en propiedades->Build->Default debug Keystore. Con este keystore es necesario obtener la huella digital SHA1 esto se realiza con la aplicación keytool de java.

Para hacer este proceso se abre una consola de Windows y se ejecuta el siguiente comando "keytool.exe" -list -v -keystore debug.keystore -alias androiddebugkey -storepass android -keypass android". Esto devuelve algo parecido a la siguiente figura.

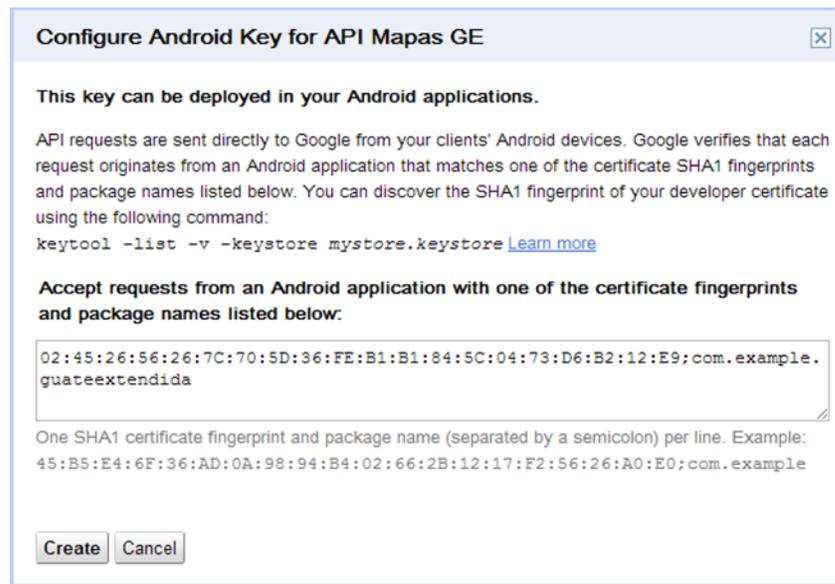
Figura 22. SHA1

```
00110 desde: Wed Nov 30 17:48:12 CET 2011 hasta: Fri Nov 22 17:48:12 CET 2011
Huellas digitales del Certificado:
MD5: 5A:F0:7B:4F:75:88:BE:3C:B3:F0:52:0C:FA:94:4F:E6
SHA1: B7:18:44:D6:A5:A2:EE:0B:86:86:72:E9:42:1A:2B:1D:67:5B:95:96
SHA256: 50:EF:2E:51:0D:19:69:60:E3:2D:AE:4C:AE:EF:44:39:84:24:47:CA:C3:
71:D4:C8:9A:F2:8D:BC:97:09:2A:E9
```

Fuente: elaboración propia.

Se copia el código SHA1 y se añade a la ventana de obtención de API Key con punto y coma se añade el nombre del proyecto.

Figura 23. Configurar API Key



Fuente: elaboración propia.

Ya con el API Key generado se ingresa al archivo manifest.

Figura 24. **Archivo manifest**

Key for Android apps (with certificates)	
API key:	AIzaSyBfXoc-vjp4CktV6Y7JDKFVnm9F4b1M3qo
Android apps:	02:45:26:56:26:7C:70:5D:36:FE:B1:B1:84:5C:04:73:D6:B2:12:E9;com.example.guateextendida
Activated on:	Oct 24, 2013 10:43 PM
Activated by:	carpra1@gmail.com – you

Fuente: elaboración propia.

```
<meta-data android:name="com.google.android.maps.v2.API_KEY"
Android : value= "AIzaSyBfXoc-vjp4CktV6Y7JDKFVnm9F4b1M3qo"/>
```

Por último, es necesario referenciar las librerías con el SDK de Google Play Services al proyecto. Para ello, desde eclipse se importa la librería de google services que se descargo al principio en “File / Import... / Existing Android Code Into Workspace”.

El siguiente paso es referenciar la librería de Google services desde la aplicación. Para ello ir a “Properties” del proyecto y accediendo a la sección Android de las preferencias. En esta ventana se pueden añadir una nueva librería en la sección inferior llamada Library. Cuando se pulsa el botón “Add...” aparecerá la librería recién importada y con ello queda la librería con referencia a el proyecto.

Con esto termina la configuración de la aplicación para que pueda utilizar Google Maps.

CONCLUSIONES

1. Con la aplicación Guate Extendida se da a conocer más Guatemala y se fomenta el turismo en la región de San Marcos.
2. A través de Google Maps es posible ubicar geográficamente cada sitio turístico mostrado en la aplicación, ya sean los que están cargados localmente en la aplicación como los que suben los usuarios.
3. La aplicación permite a los usuarios, compartir información y fotografías de sitios turísticos para que otros usuarios puedan conocerlos.
4. La aplicación posee un área de sitios ya precargados que son sitios turísticos conocidos del departamento.

RECOMENDACIONES

1. Si se desea desarrollar aplicaciones para Android, es necesario tomar en cuenta que existe gran cantidad de dispositivos en el mercado los cuales varían sus características. Por lo tanto, es necesario hacer pruebas en distintos dispositivos para lograr abarcar la mayoría de dispositivos.
2. Para desarrollar en Android es recomendable poseer un dispositivo para debuggear y realizar las pruebas. Ya que con el emulador dependiendo de la resolución de la pantalla y otras características se puede tornar demasiado lento, atrasando con ello las pruebas de la aplicación.
3. Para desarrollar en Android por comodidad se recomienda utilizar el IDE Eclipse ya que este posee un *plugin* para el desarrollo en Android. El proceso de configuración de entorno de desarrollo se hace realmente fácil gracias al bundle ADT que se descarga de la página de desarrolladores de Android.
4. En la página de desarrolladores de Android (<http://developer.android.com/index.html>) existe buena documentación, lo cual sirve como una guía para el proceso de desarrollo de cualquier aplicación que se desee implementar para este Sistema Operativo.

BIBLIOGRAFÍA

1. *Bases de Datos en Android (I)*. [en línea]
<<http://www.sgoliver.net/blog/?p=1611>> [Consulta: 15 de octubre de 2013].
2. *Controles de Selección en android: Gallery*. [en línea]
<<http://androideity.com/2011/08/29/controles-de-seleccion-en-android-gallery/>> [Consulta: 28 de octubre de 2013].
3. *Getting Started with the Facebook SDK for Android*. [en línea]
<<https://developers.facebook.com/docs/android/getting-started/>> [Consulta: 26 de octubre de 2013].
4. *Integrar SDK Facebook en Android*. [en línea]
<<http://www.nosinmiubuntu.com/2013/03/integrar-sdk-facebook-en-android.html>> [Consulta: 2 de agosto de 2013].
5. *Making ListView Scrolling Smooth*. [en línea]
<<http://developer.android.com/training/improving-layouts/smooth-scrolling.html#ViewHolder>> [Consulta: 27 de octubre de 2013].
6. *Pestañas en Android usando Eclipse*. [en línea]
<<http://androideity.com/2012/02/13/pestanas-en-android-usando-eclipse/>> [Consulta: 29 de octubre 2013].

7. *UI Fluidas y la propiedad weight en Android.* [en línea]
<<http://androideity.com/2012/06/01/ui-fluidas-y-la-propiedad-weight-en-android/>> [Consulta: 18 de septiembre de 2013].

8. *Usar nuestra propia base de datos SQLite en Android.* [en línea]
<<http://blog.netrunners.es/usar-nuestra-propia-base-de-datos-sqlite-en-android/>> [Consulta: 27 de octubre de 2013].