PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE INGENIERÍA

MAESTRÍA EN REDES DE COMUNICACIÓN

TESIS DE GRADO PREVIA LA OBTENCIÓN DEL TÍTULO DE:

MASTER EN REDES DE COMUNICACIÓN

TEMA:

"DESARROLLO DE UNA APLICACIÓN PARA EQUIPOS ANDROID, BASADA EN GEOLOCALIZACIÓN PARA OBTENER INFORMACIÓN DE ATRACTIVOS TURÍSTICOS EN LA CIUDAD DE TULCÁN"

AUTOR: **Ing. JUAN CARLOS ROBLES RUANO**

DIRECTOR: **Ing. FRANCISCO BALAREZO, MBA**

Quito – 2015

AGRADECIMIENTO

Ante todo quiero dar gracias a mi Dios por darme la vida, la inteligencia con otras virtudes y habilidades, las cuales me han servido para destacarme a nivel académico, profesional y deportivo.

A mis padres por su amor y apoyo incondicional, que me han permitido salir adelante ante las dificultades que se me han presentado en esta vida.

A mis hermanos, tíos, primos y amigos por su ayuda desinteresada y sus sugerencias que me han servido para llevar a cabo este proyecto.

Al Ingeniero Francisco Balarezo y a los ingenieros correctores, por su colaboración para culminar con éxito este posgrado y conseguir satisfactoriamente otra meta a nivel profesional.

Juan Carlos

DEDICATORIA

El presente proyecto dedico con todo mi amor a mis padres: Carlos y Mariana; a mis hermanos: Jenny, Rubén, Katherinn y Marianela; y especialmente a mis sobrinos: Bryan, Emily, Brithany, Mabel y Génesis.

Juan Carlos

RESUMEN

El presente proyecto titulado “DESARROLLO DE UNA APLICACIÓN PARA EQUIPOS ANDROID, BASADA EN GEOLOCALIZACIÓN PARA OBTENER INFORMACIÓN DE ATRACTIVOS TURÍSTICOS EN LA CIUDAD DE TULCÁN”, fue creado con la intención de poner a disposición de los usuarios que visiten la Ciudad de Tulcán, una aplicación que aprovecha la tecnología disponible en los dispositivos móviles inteligentes y la facilidad con la que estos se conectan a las redes de telecomunicaciones.

El proyecto consta de la app llamada “Conoce Tulcán” y del sitio web cuya URL es “https://sites.google.com/site/conocetulcan”. La finalidad de la aplicación móvil es proporcionar una breve información de los atractivos turísticos urbanos en la Ciudad de Tulcán y el propósito del sitio web es contener la información de cada uno de los atractivos y permitir a los usuarios descargar la app.

La aplicación al ejecutarse localiza geográficamente el dispositivo móvil del usuario; a continuación se despliega en la pantalla del dispositivo los atractivos turísticos aledaños a él. Acto seguido, el usuario debe tomar una fotografía del atractivo de su interés y en la pantalla aparece la imagen con el nombre del atractivo turístico sobreimpreso. Si el usuario desea tener una breve información del atractivo, debe pulsar el botón destinado para esta función. La información de cada atractivo turístico se la obtiene del sitio web creado para brindar este servicio. Para el funcionamiento de la aplicación, el usuario debe tener acceso a Internet.

De esta manera, con esta innovadora aplicación móvil se beneficiarán los turistas que visiten la Ciudad de Tulcán, tratando de evitarles adquirir una guía informativa, preguntar a los transeúntes locales o buscar información en otros medios.

ABSTRACT

This project titled "DEVELOPMENT OF AN APPLICATION FOR DEVICES ANDROID BASED ON GEOLOCATION TO BE OBTAINED INFORMATION ON TOURIST ATTRACTIONS IN TULCAN CITY " was created with the intention to make available to users who visit Tulcan City, an application that leverages technology available in smart mobile devices and the ease to connect to the telecommunications networks.

The project includes the app called "Know Tulcan" and the website whose URL is "https://sites.google.com/site/conocetulcan". The purpose of the mobile application is to provide brief information of urban tourist attractions in Tulcan City and purpose of the website is to contain information of each attraction and allow users to download the app.

The application to run located geographically user mobile device; is then displayed on the screen of the device the surrounding him tourist attractions. Then the user has to take a picture attractiveness of their interest and in the display image is superimposed with the name of tourist attraction. If user wants brief information of the attractiveness, you must press the button designed for this function. The information for each tourist attraction is obtained from the website created to provide this service. For the operation of the application, the user must have access to the Internet.

Thereby, with this innovative mobile application tourists visiting Tulcan City will benefit, trying to spare them acquire an informative guide, ask local passers or for information on other means.

ÍNDICE DE CONTENIDO

AGRADECIMIENTO **1**

DEDICATORIA **2**

RESUMEN **3**

ABSTRACT **4**

ÍNDICE DE CONTENIDO **5**

ÍNDICE DE FIGURAS **8**

ÍNDICE DE TABLAS **10**

ÍNDICE DE ECUACIONES **10**

CAPÍTULO 1 ..... **11**

INTRODUCCIÓN **11**

1.1 DESCRIPCIÓN GENERAL DEL PROYECTO 11
1.2 JUSTIFICACIÓN 12
1.3 ANTECEDENTES 13
1.4 OBJETIVOS 15
1.4.1 General 15
1.4.2 Específicos 15

CAPÍTULO 2 ..... **16**

MARCO TEÓRICO **16**

2.1 DISPOSITIVOS MÓVILES 16
2.1.1 Definición 16
2.1.2. Características generales 17
2.1.2.1 Movilidad 18
2.1.2.2 Tamaño reducido 18
2.1.2.3 Comunicación inalámbrica 19
2.1.2.4 Interacción con las personas 19
2.1.3 Teléfonos móviles o celulares 20
2.1.4 Teléfonos inteligentes 21
2.1.5 Tabletas o tablets 23
2.2 SISTEMA OPERATIVO ANDROID 24
2.2.1 Definición 24
2.2.2 ¿Por qué utilizarlo? 25
2.2.3 Características generales 28
2.2.4 Arquitectura 30
2.2.4.1 Kernel de Linux 30
2.2.4.2 Librerías y ejecución 31

2.2.4.3 Estructura de aplicaciones 31
2.2.4.4 Aplicaciones 31
2.3 APLICACIONES MÓVILES ANDROID 32
2.3.1 Componentes básicos 33
2.3.1.1 Actividades 34
2.3.1.2 Servicios 34
2.3.1.3 Proveedores de contenido 34
2.3.1.4 Receptores de difusión 35
2.3.2 Archivo Manifest 36
2.3.3 Recursos 37
2.4 KIT DE DESARROLLO DE SOFTWARE 37
2.5 EMULADORES 39
2.5.1 Dispositivo Virtual Android, AVD 39
2.5.2 Genymotion 41
2.6 GEOLOCALIZACIÓN 42
2.6.1 GPS 43
2.7 GOOGLE SITES 44
2.7.1 Características 45
2.7.2 Privacidad 46
2.7.3 Seguridad 47
2.8 REDES DE DATOS MÓVILES 48
2.9 RED DE DATOS INALÁMBRICA, WI-FI 51

CAPÍTULO 3 53

ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DEL SITIO WEB 53

3.1 ANÁLISIS 53
3.1.1 Requerimientos para la creación del sitio web 53
3.1.1.1 Requerimientos técnicos 53
3.1.1.2 Requerimientos funcionales 54
3.1.1.3 Requerimientos no funcionales 55
3.1.2 Seguridad del sitio web 56
3.1.3 Selección de los atractivos turísticos 56
3.1.3.1 Iglesias 57
3.1.3.2 Parques y plazoletas 57
3.1.3.3 Edificaciones patrimoniales y culturales 57
3.1.3.4 Monumentos, estatuas y bustos 58
3.2 DISEÑO 58
3.2.1 Interfaz gráfica 58
3.3 IMPLEMENTACIÓN 60
3.3.1 Recopilación de información, fotografías e imágenes 60
3.3.2 Creación del sitio web 60

CAPÍTULO 4 67

ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE LA APLICACIÓN MÓVIL 67

4.1 ANÁLISIS 67
4.1.1 Requerimientos para la creación de la app 67
4.1.1.1 Requerimientos técnicos 67
4.1.1.2 Requerimientos funcionales 67
4.1.1.3. Requerimientos no funcionales 68
4.2 DISEÑO 69
4.2.1 Arquitectura de la aplicación 72
4.2.1.1 Clase Portada 74
4.2.1.2 Clase Ubica 74
4.2.1.3 Clase Selec 74
4.2.1.4 Clase Foto1 74
4.2.1.5 Clase Foto2 75
4.2.1.6 Clase Foto3 75
4.2.1.7 Clase Acerca 76
4.3 IMPLEMENTACIÓN 76
4.3.1 Obtención de coordenadas de los atractivos turísticos 76
4.3.2 Instalación del entorno de desarrollo Eclipse con SDK Android 78
4.3.3 Instalación del emulador Genymotion 80
4.3.4 Creación de la app “Conoce Tulcán” 85
4.3.5 Emulación de la app en Genymotion 94
4.3.6 Generación del archivo APK 100

CAPÍTULO 5 104

INSTALACIÓN Y PRUEBAS DE LA APP EN DISPOSITIVOS MÓVILES 104

5.1 INSTALACIÓN 104
5.2 FUNCIONAMIENTO 108
5.3 PRUEBAS 115
5.3.1 Prueba de usabilidad 115
5.3.2 Prueba de funcionalidad 116
5.3.3 Pruebas técnicas 116

CAPÍTULO 6 119

CONCLUSIONES Y RECOMENDACIONES 119

6.1 CONCLUSIONES 119
6.2 RECOMENDACIONES 121

REFERENCIAS BIBLIOGRÁFICAS 122

ACRÓNIMOS 126

ANEXOS.....130

ANEXO 1: CÓDIGO FUENTE DE LA APP.....130
ANEXO 2: MANUAL DE USUARIO.....156
ANEXO 3: FOTOGRAFÍAS DE LA EJECUCIÓN DE LA APLICACIÓN "CONOCE TULCÁN" EN ALGUNOS DISPOSITIVOS MÓVILES INTELIGENTES.....171

ÍNDICE DE FIGURAS

FIGURA 2.1.1 ALGUNOS DISPOSITIVOS MÓVILES.....16
FIGURA 2.1.2 MOVILIDAD.....18
FIGURA 2.1.3 TAMAÑO REDUCIDO.....19
FIGURA 2.1.4 COMUNICACIÓN INALÁMBRICA.....19
FIGURA 2.1.5 INTERACCIÓN CON LAS PERSONAS.....20
FIGURA 2.1.6 TELÉFONOS MÓVILES.....20
FIGURA 2.1.7 TELÉFONOS INTELIGENTES.....22
FIGURA 2.1.8 TABLETAS.....24
FIGURA 2.2.1 VERSIONES DE ANDROID.....26
FIGURA 2.2.2 SISTEMAS OPERATIVOS EN TELÉFONOS MÓVILES Y TABLETAS.....26
FIGURA 2.2.3 FABRICANTES DE DISPOSITIVOS MÓVILES.....27
FIGURA 2.2.4 NÚMERO DE APLICACIONES EN GOOGLE PLAY.....28
FIGURA 2.2.5 ARQUITECTURA DE ANDROID.....30
FIGURA 2.3.1 ELEMENTOS DE UNA APLICACIÓN MÓVIL.....33
FIGURA 2.5.1 VENTANA DE EDICIÓN DE UN AVD.....40
FIGURA 2.5.2 DISPOSITIVO VIRTUAL SAMSUNG S3.....42
FIGURA 2.8.1 ICONOS DE REDES MÓVILES.....49
FIGURA 3.2.1 ESTRUCTURA DE LA PÁGINA PRINCIPAL.....59
FIGURA 3.2.2 ESTRUCTURA DE LAS PÁGINAS INFORMATIVAS.....59
FIGURA 3.3.1 INGRESO A GOOGLE SITES.....60
FIGURA 3.3.2 INTERFAZ GRÁFICA DE GOOGLE SITES.....61
FIGURA 3.3.3 CREACIÓN DEL SITIO WEB.....61
FIGURA 3.3.4 PLANTILLA DEL SITIO WEB.....62
FIGURA 3.3.5 INTERFAZ DE EDICIÓN.....62
FIGURA 3.3.6 CREACIÓN DE UNA NUEVA PÁGINA.....63
FIGURA 3.3.7 MENÚ DE ACCIONES.....63
FIGURA 3.3.8 COMPARTICIÓN DEL ACCESO AL SITIO WEB.....64
FIGURA 3.3.9 PÁGINA PRINCIPAL DEL SITIO.....65
FIGURA 3.3.10 PÁGINA INFORMATIVA PARA LA CATEDRAL DE TULCÁN.....65
FIGURA 3.3.11 MAPA DEL SITIO.....66
FIGURA 4.2.1 ÁREA DE ACCIÓN PARA UN USUARIO.....71

FIGURA 4.2.2 DIAGRAMA DE FLUJO DE LA APLICACIÓN CONOCE TULCÁN 72
FIGURA 4.2.3 ARQUITECTURA DE LA APP 73
FIGURA 4.2.4 DIAGRAMA DE CLASES 73
FIGURA 4.3.1 SITIO DE DESCARGA DE LAS HERRAMIENTAS DE DESARROLLO ANDROID ... 78
FIGURA 4.3.2 CARPETAS DE LAS HERRAMIENTAS DE DESARROLLO 78
FIGURA 4.3.3 CREACIÓN DE LA CARPETA "WORKSPACE" 79
FIGURA 4.3.4 VENTANA DE BIENVENIDA AL DESARROLLO ANDROID 79
FIGURA 4.3.5 INTERFAZ GRÁFICA DE ECLIPSE 80
FIGURA 4.3.6 PÁGINA DE DESCARGA DE GENYMOTION 80
FIGURA 4.3.7 VENTANA PARA INSTALAR NUEVO SOFTWARE 81
FIGURA 4.3.8 VENTANA PARA AÑADIR EL PLUGIN DE GENYMOTION 81
FIGURA 4.3.9 UBICACIÓN DEL ICONO DEL EMULADOR GENYMOTION 82
FIGURA 4.3.10 EMULADOR GENYMOTION 82
FIGURA 4.3.11 CONFIGURACIÓN DE EMULADOR 83
FIGURA 4.3.12 SELECCIÓN DEL DISPOSITIVO VIRTUAL PARA DESCARGAR 83
FIGURA 4.3.13 PROPIEDADES DEL DISPOSITIVO VIRTUAL 84
FIGURA 4.3.14 PROCESO DE DESCARGA DEL DISPOSITIVO VIRTUAL 84
FIGURA 4.3.16 VENTANA PARA CREAR UN PROYECTO 86
FIGURA 4.3.17 VENTANA PARA CONFIGURAR UN PROYECTO 86
FIGURA 4.3.18 VENTANA PARA CONFIGURAR EL ICONO 87
FIGURA 4.3.19 VENTANA PARA CREAR UNA ACTIVIDAD 88
FIGURA 4.3.20 VENTANA PARA CREAR UNA ACTIVIDAD VACÍA 89
FIGURA 4.3.21 ACTIVIDAD PORTADA DEL PROYECTO CONOCE TULCÁN 89
FIGURA 4.3.22 LAYOUT "PORTADA.XML" 90
FIGURA 4.3.23 CREACIÓN DE LA ACTIVIDAD "UBICA" 91
FIGURA 4.3.24 ACTIVIDAD UBICA EN EL PROYECTO CONOCE TULCÁN 91
FIGURA 4.3.25 CREACIÓN DEL ARCHIVO UBICA.XML 92
FIGURA 4.3.26 ARCHIVO "ANDROIDMANIFEST.XML" 93
FIGURA 4.3.27 DISPOSITIVOS VIRTUALES DE GENYMOTION 94
FIGURA 4.3.28 CONFIGURACIÓN PARA EJECUTAR LA APP 94
FIGURA 4.3.29 SELECCIÓN DEL DISPOSITIVO VIRTUAL 95
FIGURA 4.3.30 VENTANA DE INICIO 95
FIGURA 4.3.31 SELECCIÓN DE CÁMARA 96
FIGURA 4.3.32 COLOCACIÓN DE COORDENADAS 96
FIGURA 4.3.33 VENTANA DE UBICACIÓN GEOGRÁFICA 97
FIGURA 4.3.34 LISTA DE LUGARES CERCANOS 97
FIGURA 4.3.35 VENTANA PARA TOMAR FOTO Y MOSTRARLA 98
FIGURA 4.3.36 IMAGEN CON NOMBRE DEL ATRACTIVO TURÍSTICO 99
FIGURA 4.3.37 BREVE INFORMACIÓN DE LA CATEDRAL DE TULCÁN 99
FIGURA 4.3.38 CORTA INFORMACIÓN DE LA APP 100
FIGURA 4.3.39 VENTANA PARA EXPORTAR ARCHIVOS 101

FIGURA 4.3.40 VENTANA DE VERIFICACIÓN DEL PROYECTO 101
FIGURA 4.3.41 VENTANA DE "KEYSTORE" 102
FIGURA 4.3.42 VENTANA DE CREACIÓN DE LA CLAVE 102
FIGURA 4.3.43 VENTANA DE DESTINACIÓN DEL ARCHIVO APK 103
FIGURA 5.1.1 PÁGINA DE DESCARGA DE LA APLICACIÓN "CONOCE TULCÁN" 104
FIGURA 5.1.2 PROCESO DE LA DESCARGA 105
FIGURA 5.1.3 HABILITACIÓN DE "FUENTES DESCONOCIDAS" 105
FIGURA 5.1.4 UBICACIÓN DEL ARCHIVO DESCARGADO "CONOCETULCÁN.APK" 106
FIGURA 5.1.5 INSTALADOR DE PAQUETE 106
FIGURA 5.1.6 HARDWARE A UTILIZAR LA APP 107
FIGURA 5.1.7 PROCESO DE INSTALACIÓN 107
FIGURA 5.1.8 FINALIZACIÓN DE LA INSTALACIÓN 108
FIGURA 5.2.1 ICONO DE LA APP 108
FIGURA 5.2.2 VENTANA DE INICIO 109
FIGURA 5.2.3 BÚSQUEDA Y RESULTADO DE LA UBICACIÓN GEOGRÁFICA 110
FIGURA 5.2.4 ATRACTIVOS TURÍSTICOS CERCANOS AL USUARIO 110
FIGURA 5.2.5 VENTANA PARA TOMAR UNA FOTO Y MOSTRARLA 111
FIGURA 5.2.6 ENFOCANDO Y CAPTURANDO EL ATRACTIVO TURÍSTICO 111
FIGURA 5.2.7 IMAGEN CAPTURADA 112
FIGURA 5.2.8 IMAGEN DEL ATRACTIVO TURÍSTICO CON SU NOMBRE SOBREIMPESO 112
FIGURA 5.2.9 BREVE INFORMACIÓN DEL "MONUMENTO A ABDÓN CALDERÓN" 113
FIGURA 5.2.10 CORTA INFORMACIÓN DE LA APP 113
FIGURA 5.2.11 CARPETA "MIS ARCHIVOS" 114
FIGURA 5.2.12 TODAS LAS CARPETAS DEL DISPOSITIVO 114
FIGURA 5.2.13 ARCHIVOS DE LAS FOTOGRAFÍAS DE CIERTOS ATRACTIVOS TURÍSTICOS 115

ÍNDICE DE TABLAS

TABLA 2.9.1 DIFERENCIAS ENTRE UNA RED MÓVIL Y UNA RED WI-FI 52
TABLA 4.3.1 COORDENADAS GEOGRÁFICAS DE LOS ATRACTIVOS TURÍSTICOS 77
TABLA 5.3.1 RESULTADOS DE LA PRUEBA DE USABILIDAD 115
TABLA 5.3.2 RESULTADOS DE LA PRUEBA DE FUNCIONALIDAD 116
TABLA 5.3.3 RESULTADOS DEL FUNCIONAMIENTO EN VARIOS DISPOSITIVOS MÓVILES ... 117
TABLA 5.3.4 COMPARACIÓN DE TIEMPOS DE RESPUESTA 117

ÍNDICE DE ECUACIONES

ECUACIÓN 4.2.1 FÓRMULA DE LA DISTANCIA ENTRE DOS PARALELOS 70

CAPÍTULO 1

INTRODUCCIÓN

1.1 DESCRIPCIÓN GENERAL DEL PROYECTO

En la actualidad se ofrece un sinnúmero de aplicaciones para dispositivos móviles, orientadas al turismo. Estas aplicaciones suelen facilitar a los usuarios realizar las actividades siguientes: acceder a datos de geolocalización, proporcionar información de puntos de interés, trazar rutas en mapas, reservar una habitación de hotel, entre otras. Todo esto aprovechando las características que ofrecen hoy en día la mayoría de dispositivos móviles inteligentes, como: disponer de GPS, acelerómetro, giroscopio, sensor de proximidad, sensor de luz, cámara frontal, cámara posterior, conexión Wi-Fi, conexión a redes de datos, entre otras funciones que facilitan la utilización de múltiples aplicaciones. [1]

El presente trabajo presenta el desarrollo de una aplicación móvil Android, basada en geolocalización, para proporcionar información de atractivos turísticos en la Ciudad de Tulcán, obteniéndola de un sitio web creado para dicha finalidad, la app y el portal web llevan el nombre de “Conoce Tulcán”. El propósito de dicha aplicación es presentar una breve información de: iglesias, parques, monumentos, entre otros sitios patrimoniales y culturales ubicados en la zona urbana; evitándole al usuario adquirir una guía turística o preguntar a los transeúntes locales. Todo esto aprovechando las características ofrecidas por los dispositivos móviles con sistema operativo Android.

El funcionamiento general de la aplicación consiste en localizar geográficamente el dispositivo móvil del usuario; a continuación se despliega en la pantalla los atractivos turísticos aledaños a él; enseguida debe escoger un atractivo; acto seguido tiene que tomar una fotografía del atractivo de su interés y en la pantalla debe aparecer la imagen con el nombre del atractivo turístico sobreimpreso. Si el usuario desea tener una breve información detallada del atractivo, tiene que pulsar el botón destinado para ejecutar esta acción. La información más detallada de los atractivos turísticos se la obtiene de un sitio web creado para brindar este

servicio. Para la ejecución de la aplicación, el usuario debe tener conexión a internet, sea por red Wi-Fi o red móvil.

1.2 JUSTIFICACIÓN

Actualmente en la Ciudad de Tulcán únicamente existen guías turísticos en el Cementerio Municipal José María Azaél Franco y oficinas de turismo del Ministerio de Turismo, GAD Municipal y GAD Provincial. El resto de atractivos turísticos no cuentan con personal especializado en el ramo para que proporcionen información in situ a los turistas nacionales y extranjeros. Además, cabe señalar que no existe ningún operador turístico en toda la provincia. Ante esta situación tomé la decisión personal, libre y voluntaria de desarrollar una aplicación móvil para equipos Android, contribuyendo, en parte, con una solución al inconveniente antes mencionado.

Hoy en día, a los dispositivos móviles se los ha dotado de una gran capacidad de cómputo con varias funciones, utilizando Android como sistema operativo con licencia libre y código abierto. Los desarrolladores de aplicaciones Android tienen a disposición algunas herramientas de desarrollo de software. Además tienen varias opciones para distribuir y comercializar sus aplicaciones. [1]

La importancia de este proyecto radica en la utilización de la tecnología que disponen los dispositivos móviles inteligentes como pantalla táctil, cámara, GPS, conexión a internet, [2] entre otras funciones y en el empleo de recursos de Internet, como un sitio web que se debe crear y tiene que servir de soporte, ya que va a alojar la información de los atractivos turísticos y facilitar a los usuarios descargar la aplicación “Conoce Tulcán”.

Para la creación de la app se debe tener cuenta conceptos de desarrollo de software, con un lenguaje de programación propio de sistemas Android y para la creación del sitio web se va a emplear Google Sities, una plataforma en línea que facilita crear sitios, para no tener problemas de alojamiento en la web, ya que se

trata de un proyecto netamente con fines académicos. Además cabe mencionar, el funcionamiento general de dicha aplicación se basa en conceptos relacionados a Internet, Sistemas de Comunicaciones, Redes Inalámbricas y Comunicaciones Móviles, que son módulos que se los trataron en su momento a lo largo de la Maestría en Redes de Comunicación.

1.3 ANTECEDENTES

La Ciudad de Tulcán es la capital de la Provincia del Carchi, es la más norteña de todas las capitales de provincia, está ubicada en la frontera con la República de Colombia. Se encuentra a una altitud de 2957 m sobre el nivel del mar, posee un clima templado frío con una temperatura aproximadamente de 12° C. Es el centro de la infraestructura de servicios turísticos del Carchi; cuenta con aeropuerto, terminal terrestre de transporte interprovincial, intercantonal e interparroquial, hoteles, restaurantes, discotecas, entre otros servicios, desde donde es posible visitar los lugares turísticos de la provincia y del país, para un verdadero deleite de propios y extraños. [3]

Tulcán es considerada una ciudad eminentemente comercial, la principal motivación para visitarla es realizar compras o establecer transacciones de negocios. Tradicionalmente, desde hace muchos años existe un gran movimiento en el intercambio de mercaderías entre las Repúblicas de Ecuador y Colombia. Para promover su desarrollo turístico, se debe girar alrededor de su principal segmento de visitantes, que son personas con una estadía corta en la ciudad y que la visitan con intenciones de hacer compras y a su vez conocerla. [4]

El mayor atractivo turístico de esta ciudad es el Cementerio Municipal José María Azaél Franco, esta joya arquitectónica lleva el título de Patrimonio Cultural del Estado y es una de las bellezas más representativas a nivel de América Latina. [5]

Es el único sitio turístico que cuenta con guías turísticos, el resto de atractivos turísticos de la ciudad como iglesias, monumentos, parques, entre otros sitios patrimoniales o culturales, no cuentan con personal capacitado que proporcione información a los turistas nacionales y extranjeros.

En lo que se refiere a dispositivos móviles inteligentes, debido a los avances de la tecnología cada vez brindan mayores y mejores prestaciones a los usuarios. Actualmente un equipo móvil inteligente básicamente dispone de GPS, acelerómetro, giroscopio, sensor de proximidad, doble cámara, Wi-Fi, [2] entre otras funciones que facilitan la utilización de múltiples aplicaciones, para la realización de video llamadas, entretenerse con juegos, utilización de geolocalización, disfrutar de realidad aumentada, disponer de información en tiempo real, entre otras utilidades desarrolladas para el sistema operativo Android. Este sistema se basa en el núcleo de Linux, disponible con licencia libre y código abierto para cualquier fabricante; actualmente es el más utilizado a nivel mundial por los productores de dispositivos móviles.

Los desarrolladores de aplicaciones Android disponen de un Kit de Desarrollo de Software bajo el lenguaje Java, un Kit de Desarrollo Nativo para aplicaciones o extensiones en lenguaje C o C++, entre otros. Los programadores novatos pueden emplear un entorno visual llamado Google App Inventor u otras plataformas disponibles en la web. [6]

En lo concerniente a la geolocalización, que se basa en el Sistema de Posicionamiento Global (GPS) cuyo objetivo es la determinación de las coordenadas geográficas de lugares respecto de un sistema de referencia mundial. Los puntos pueden estar ubicados en cualquier parte del planeta, pueden permanecer estáticos o estar en movimiento y las observaciones pueden obtenerse en cualquier momento del día. La difusión de la tecnología GPS se ha convertido de hecho en un servicio público de carácter mundial de enorme importancia y con la disponibilidad de un sinnúmero de aplicaciones con geolocalización para dispositivos de uso personal. [7]

Las apps disponibles en Google Play, basadas en geolocalización y orientadas al turismo son muy variadas a nivel mundial, tales como guías turísticas de algunas ciudades del mundo, mapas, rutas, entre otras. También existen aplicaciones basadas en las plataformas de Layar, Wikitude, Junaio, entre otras, que introducen el concepto de la realidad aumentada, convirtiéndolas en más llamativas y novedosas. Pero realizar una aplicación utilizando dichas plataformas, tiene un costo económico muy alto para su publicación o distribución.

En el Ecuador, se han desarrollado aplicaciones como las mencionadas anteriormente, para las principales ciudades y se han realizado algunos trabajos de tesis de grado.

1.4 OBJETIVOS

1.4.1 General

Desarrollar la aplicación móvil “Conoce Tulcán” para equipos Android, basada en geolocalización para obtener información de atractivos turísticos en la Ciudad de Tulcán.

1.4.2 Específicos

- Analizar las tecnologías y recursos disponibles para llevar a cabo el desarrollo de la aplicación móvil basada en geolocalización.

- Crear un sitio web que sirva como soporte para la aplicación móvil creada, donde se va a encontrar la información de los atractivos turísticos y las personas interesadas puedan descargar la app.

- Analizar las seguridades que ofrece la plataforma Google Sities al sitio web creado.

- Realizar pruebas de funcionamiento de la app en algunos modelos con el sistema operativo Android.

- Comparar el funcionamiento de la app en una red móvil y en una red Wi-Fi.

CAPÍTULO 2

MARCO TEÓRICO

2.1 DISPOSITIVOS MÓVILES

2.1.1 Definición

El término dispositivo móvil se refiere a un equipo que tiene capacidad de procesamiento, conexión a Internet, almacenamiento y está diseñado específicamente para una función, pero que puede llevar a cabo otras funciones más generales. No obstante, algunas veces también se clasifican las cámaras digitales y los reproductores MP3 o MP4 como dispositivos móviles, su clasificación se presenta a continuación:

- Teléfonos móviles o celulares
- Teléfonos inteligentes (Smartphone)
- Tabletas (Tablet)
- Asistentes personales digitales (PDA)
- Computador de mano (Handheld PC)
- Tablet PC
- Lector de libros electrónicos
- Sistemas de navegación de automóviles
- Consolas portátiles de juegos



Figura 2.1.1 Algunos dispositivos móviles[1]

[1] Fuente: Imágenes de Google. *Dispositivos móviles.* http://www.google.com.ec/imghp?hl=es. Editado por el autor

Más adelante se hará una descripción general solamente de los teléfonos móviles, inteligentes y tabletas, debido a que estos dispositivos utilizan el sistema operativo Android.

2.1.2. Características generales

Las características esenciales que poseen los dispositivos móviles son las que se detallan enseguida:

- Son aparatos pequeños
- La mayoría de estos aparatos se pueden transportar en el bolsillo del propietario o en un pequeño bolso
- Funcionalidad limitada
- En pocos años el usuario deberá cambiarlo
- Tienen capacidad de procesamiento
- Tienen conexión permanente o limitada a una red
- Tienen memoria (RAM, tarjetas MicroSD, entre otras)
- Fácil de aprender su operación, no se requieren usuarios expertos
- Uso individual de una persona, tanto en posesión como en operación
- Tienen una alta capacidad de interacción mediante la pantalla o el teclado [7]

Generalmente, un dispositivo móvil puede definirse con sólo cuatro características que lo diferencian de otros dispositivos que, aunque pudieran parecer similares, carecen de algunas de las especificaciones de los verdaderos dispositivos móviles. Estas cuatro características son:

- Movilidad.
- Tamaño reducido.
- Comunicación inalámbrica.
- Interacción con las personas. [1]

2.1.2.1 *Movilidad*

Se entiende a la cualidad que tiene un dispositivo para ser transportado frecuentemente con facilidad. Por consiguiente, el concepto de movilidad es una característica básica. Estos dispositivos son lo suficientemente pequeños para ser trasladados y utilizados durante su transporte.

Todo dispositivo móvil debería funcionar y poder ser usado de forma fiable mientras es transportado. Para ofrecer la portabilidad, los dispositivos móviles contienen baterías recargables, que permiten varias horas de operación sin necesidad de acceso a un cargador. Aunque son móviles, estos dispositivos pueden sincronizarse con un sistema de sobremesa para actualizar aplicaciones y datos.



Figura 2.1.2 Movilidad[2]

2.1.2.2 *Tamaño reducido*

Se refiere a la particularidad que posee un dispositivo móvil para ser usado fácilmente con una o dos manos. El tamaño reducido también permite ser transportado cómodamente por una persona. Se lo puede llevar en una mano y cabe en un bolsillo o en un pequeño estuche.

Los dispositivos móviles están dotados de pantallas táctiles o pequeños teclados numéricos para recibir datos de entrada y mantener su tamaño compacto.

[2] Fuente: *Imágenes de Google. Dispositivos móviles con personas.* http://www.worldanime.tv/wp-content/uploads/2012/01/20120122-064153-p.m.-1024x764.jpg



Figura 2.1.3 Tamaño reducido[3]

2.1.2.3 *Comunicación inalámbrica*

Se trata sobre la capacidad que tiene un dispositivo para enviar o recibir datos, sin la necesidad de un enlace cableado.

Los términos de móvil y sin cables muchas veces suelen confundirse. Por ejemplo, un dispositivo móvil con datos en él y aplicaciones para administrar dichos datos puede ser móvil, pero no tiene por qué ser inalámbrico, ya que puede necesitar un cable para conectarse a un ordenador y obtener o enviar datos y aplicaciones.

Si el dispositivo puede enlazarse a una red inalámbrica para obtener datos mientras se moviliza por la calle o se encuentra en la oficina, entonces será también inalámbrico.

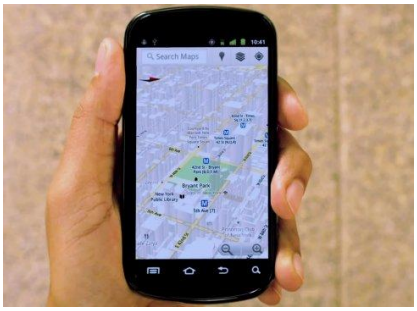


Figura 2.1.4 Comunicación inalámbrica[4]

2.1.2.4 *Interacción con las personas*

Consiste en el proceso de uso que establece un usuario con su dispositivo. Entre otros aspectos que intervienen en la interacción son la usabilidad y la ergonomía.

[3] Fuente: *Imágenes de Google. Dispositivos móviles.* http://tecnoalt.com/wp-content/uploads/2013/11/wpid-internet-movil-acceso-internet_624x351.jpg

[4] Fuente*:* Imágenes de Google. *Dispositivos móviles.* http://globedia.com/imagenes/noticias/2011/5/28/google-maps-consultas-dispositivos-moviles-pcs_1_731576.jpg

Un usuario interactúa con su dispositivo móvil mediante la interfaz gráfica de usuario, teniendo en cuenta la distancia a la pantalla, el orden de los elementos, el tamaño de los textos y las imágenes, la apariencia visual de las páginas, entre otras.

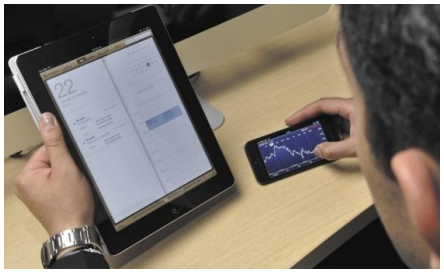


Figura 2.1.5 Interacción con las personas[5]

2.1.3 Teléfonos móviles o celulares

Un teléfono móvil se trata de un artefacto electrónico inalámbrico que sirve para acceder y utilizar los servicios de una red de telefonía móvil o celular. Este tipo de dispositivos se encuentran constituido por los elementos siguientes:

- Microprocesador DSP
- Módulo de recepción / transmisión
- Placa de circuitos
- Pantalla de cristal líquido
- Altavoz
- Micrófono
- Teclado
- Batería



Figura 2.1.6 Teléfonos móviles[6]

[5] Fuente: Imágenes de Google. *Uso de dispositivos móviles.* *http://st.elespectador.co/files/imagenprincipal/0afe3a138c16425723e6e2eb0bf24926.jpg*

[6] Fuente: Imágenes de Google. *Teléfonos móviles.* http://www.smartblog.es/wp-content/uploads/2011/09/historia-telefonos-moviles.jpg

Los teléfonos móviles facilitan la comunicación con otros equipos celulares y también pueden conectarse a la red telefónica convencional. Además, disponen de otros servicios como: mensajería SMS, calculadora, agenda, juegos, alarma, recepción de radio FM, entre otros; dependiendo del modelo del dispositivo móvil. [8]

2.1.4 Teléfonos inteligentes

Los teléfonos inteligentes también conocidos como Smartphone, integran los conceptos de teléfono móvil y ordenador en un solo dispositivo. Estos dispositivos presentan una alta funcionalidad que un teléfono móvil, debido a que disponen de un hardware más avanzado, como el que se describe a continuación:

- Procesador de doble núcleo con una velocidad superior a 500 MHz. Los últimos modelos tienen incorporado un procesador de cuatro núcleos con una velocidad de 2,5 GHz.
- Procesador de gráficos.
- Memoria RAM no superior a 1 GB, memoria interna de 4 GB y un slot para memoria externa microSD no superior a 32GB. Los últimos modelos disponen de una RAM de 2 GB, una memoria interna de 16 o 32 GB y un slot para memoria externa microSD hasta de 128 GB.
- Pantalla táctil de cristal líquido TFT de 3,3’ a 5’, con varias resoluciones dependiendo del modelo. En modelos más avanzados se utiliza la tecnología AMOLED en pantallas de hasta 5,5’.
- Sensores de proximidad, acelerómetro, giroscopio y brújula. En los últimos modelos se agregan otros sensores como: barómetro, ritmo cardíaco y de gestos.
- Sistema de posicionamiento global GPS.
- Cámara posterior de 5 MP o 8 MP, algunos modelos poseen cámara frontal con una resolución inferior a 2 MP. Los modelos más sofisticados tienen una cámara posterior de 16 MP o más y una frontal de 2,2 MP.
- Conexión a internet a través de paquetes de datos o Wi-Fi.

- Conexión con otros dispositivos a través de Bluetooth, NFC, infrarrojo y USB, la disponibilidad varía de acuerdo al modelo. [9]

Gracias al avance de la tecnología, todas las mejoras en el hardware de un Smartphone, permiten al usuario, de acuerdo al modelo que disponga, realizar algunas funciones como: revisar el correo electrónico, gestionar el organizador personal, disponer de información en tiempo real, realizar videollamadas, entretenerse con videojuegos, escuchar música, observar videos, acceder a redes sociales, editar videos e imágenes, usar la mensajería instantánea, utilizar la geolocalización, disfrutar de realidad aumentada, revisar archivos de Word, Excel, PowerPoint y PDF e instalar otros programas para realizar diversas funciones que permita el sistema operativo del teléfono.



Figura 2.1.7 Teléfonos inteligentes[7]

Estos dispositivos están producidos bajo una plataforma informática y dependiendo del fabricante utilizan diferentes sistemas operativos, como: Android de Google, iOS de Apple, Windows Phone de Microsoft, BlackBerry OS de RIM, Bada de Samsung, Symbian de Nokia, entre otros. Definitivamente la posibilidad de acceso a internet es el factor que más ha incidido para que los Smartphone logren tener el nivel de penetración alcanzado en el mercado. [10]

[7] Fuente: Imágenes de Google. *Teléfonos inteligentes.* http://nexusmaniacs.com/wp-content/uploads/2014/07/Smartphones.jpg

2.1.5 Tabletas o tablets

Una tableta es un dispositivo de mayor tamaño que un Smartphone, es un aparato que integra las mejores funcionalidades de un teléfono inteligente y de un ordenador, también se las conoce como tabletas táctiles. Ciertos especialistas se atreven a afirmar que, en un futuro no muy lejano, conseguirán desplazar a los actuales netbooks.

Estos dispositivos destacan por su ligereza, versatilidad y reducidas dimensiones, lo que facilita enormemente su portabilidad. Por ser táctil no se necesita teclado físico o ratón para que el usuario pueda interactuar, esto los convierte en herramientas intuitivas, rápidas y que no precisan de aprendizaje instrumental por parte del usuario. Por primera vez, es la tecnología la que se adapta al usuario y no al contrario. [11]

Estos dispositivos se encuentran diseñados con el hardware que se detalla enseguida:

- Procesador de doble núcleo con una velocidad inferior a 1,2 GHz. Los últimos modelos tienen incorporado un procesador de cuatro núcleos con una velocidad de hasta 2,5 GHz.
- Procesador de gráficos.
- Memoria RAM no superior a 1GB, memoria interna de 8GB, algunos modelos poseen un slot para memoria externa microSD no superior a 32GB. Los últimos modelos disponen de una RAM de 3GB, una memoria interna de 16 o 32 GB y un slot para memoria externa microSD hasta de 128GB.
- Pantalla táctil de cristal líquido TFT de 7’ a 10,1’, con varias resoluciones dependiendo del modelo. En modelos más avanzados se utiliza la tecnología AMOLED.
- Sensores como acelerómetro, giroscopio y brújula, algunos modelos poseen también un sensor de proximidad.
- Sistema de posicionamiento global GPS, algunos modelos carecen de esta función.
- Cámara posterior con una resolución de hasta 8MP que varía según del modelo, algunos modelos también poseen cámara frontal con una resolución inferior a 2,2 MP.

- Conexión a internet a través Wi-Fi, algunos modelos también tienen acceso a la web por medio de paquetes de datos.
- Conexión con otros dispositivos a través de Bluetooth, NFC, infrarrojo y USB, la disponibilidad varía de acuerdo al modelo. [12]



Figura 2.1.8 Tabletas[8]

Todo este permite al usuario, según el modelo que utilice, realizar algunas tareas como: ver el correo electrónico, emplear el organizador personal, recibir información en tiempo real, efectuar videollamadas, entretenerse con videojuegos, escuchar música, observar videos, acceder a redes sociales, editar videos e imágenes, usar la mensajería instantánea, utilizar la geolocalización, disfrutar de realidad aumentada, revisar archivos de Office y PDF, editar archivos de Office e instalar otras aplicaciones que permita el sistema operativo de la tableta.

Según el fabricante, las tabletas pueden utilizar los siguientes sistemas operativos: Android de Google, iOS de Apple, WebOS de HP, ChromeOS proyecto de Google, Windows Phone de Microsoft, Ubuntu Touch basado en Linux, entre otros. [13]

2.2 SISTEMA OPERATIVO ANDROID

2.2.1 Definición

El sistema operativo Android se basa en el kernel de Linux 2.6, es un producto de la Open Handset Alliance, un grupo de organizaciones que

[8] Fuente: Imágenes de Google. *Tabletas.* http://static.digit.in/fckeditor/besttablets2013.jpg.

colaboran para desarrollar un mejor teléfono móvil para los consumidores. Este grupo, dirigido por Google, incluye operadores de telefonía móvil, fabricantes de teléfonos móviles, industrias de semiconductores, empresas de software y compañías de marketing. Android es una plataforma de código abierto, lo que significa que no solamente pueden mejorarla los desarrolladores de Google, sino que también se nutre de las aportaciones de desarrolladores externos. Originalmente, el objetivo de implementación para Android fue el área de teléfonos móviles, incluso teléfonos inteligentes. Sin embargo, el rango completo de servicio de computación de Android y el amplio soporte funcional tienen el potencial para extenderse más allá del mercado de teléfonos móviles. Android puede ser útil para otras plataformas y aplicaciones. [14]

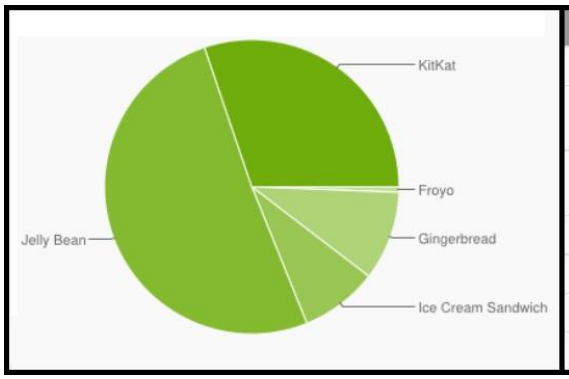
2.2.2 ¿Por qué utilizarlo?

A continuación se detallan las razones más importantes porque es conveniente la utilización del sistema operativo Android:

- Cuenta con su propio sitio web, “https://android.com”, que pone a disposición de los usuarios lo siguiente:
  - Información del sistema operativo
  - Herramientas de desarrollo
  - Guía de las API’s
  - Detalle de clases y funciones del sistema
  - Ejemplos de aplicaciones
  - Entre otros servicios

- Este sistema operativo al ser de código abierto, se encuentra libre de licencias y no existe pago alguno por distribución o desarrollo, por consiguiente, los desarrolladores pueden modificarlo y mejorarlo, produciéndose un sinnúmero de posibilidades y mejoras. En la Figura 2.2.1 se puede observar las diferentes versiones en funcionamiento de acuerdo al número de usuarios registrados en Google Play, según

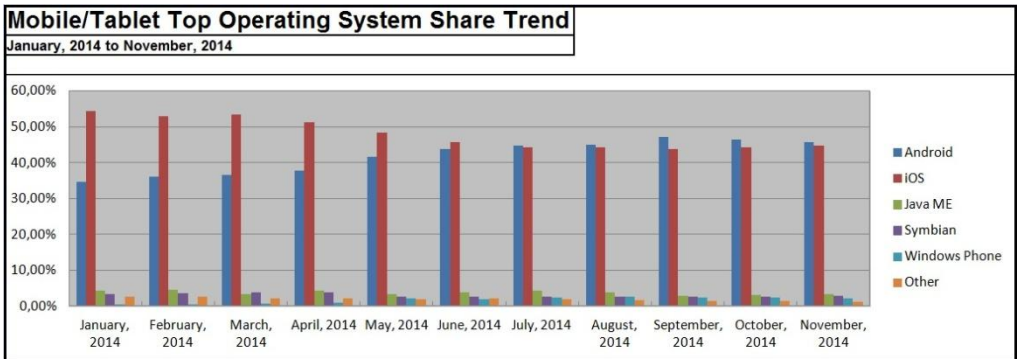
datos obtenidos el 3 de noviembre del 2014 del sitio web oficial de Android.



Version	Codename	API	Distribution
2.2	Froyo	8	0.6%
2.3.3 - 2.3.7	Gingerbread	10	9.8%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	8.5%
4.1.x	Jelly Bean	16	22.8%
4.2.x		17	20.8%
4.3		18	7.3%
4.4	KitKat	19	30.2%

Figura 2.2.1 Versiones de Android[9]

- Actualmente es el sistema operativo más utilizado en tabletas y teléfonos móviles alrededor del mundo, según datos obtenidos del sitio web de la compañía Net Applications, tal como se muestra en la Figura 2.2.2.



Month	Android	iOS	Java ME	Symbian	Windows Phone	Other
January, 2014	34,60%	54,46%	4,26%	3,41%	0,56%	2,72%
February, 2014	36,14%	52,96%	4,44%	3,50%	0,45%	2,51%
March, 2014	36,58%	53,29%	3,36%	3,92%	0,69%	2,17%
April, 2014	37,75%	51,11%	4,36%	3,77%	0,83%	2,18%
May, 2014	41,58%	48,34%	3,46%	2,52%	2,10%	2,00%
June, 2014	43,75%	45,61%	3,77%	2,73%	1,99%	2,15%
July, 2014	44,62%	44,19%	4,19%	2,57%	2,49%	1,94%
August, 2014	45,01%	44,34%	3,77%	2,61%	2,69%	1,57%
September, 2014	47,06%	43,86%	2,78%	2,56%	2,38%	1,37%
October, 2014	46,38%	44,23%	3,05%	2,60%	2,38%	1,36%
November, 2014	45,78%	44,61%	3,41%	2,78%	2,13%	1,29%

Figura 2.2.2 Sistemas operativos en teléfonos móviles y tabletas[10]

A partir del mes de julio del 2014 el sistema operativo Android empieza a superar al iOS de Apple y se ha mantenido así hasta la fecha actual.

[9] Fuente: Developer Android. *Dashboards.* http://developer.android.com/about/dashboards/index.html. Editado por el autor

[10] Fuente: Net Applications. *Mobile/Tablet TopOperating System Share Trend.* http://marketshare.hitslink.com/operating-system-market-share.aspx?qprid=9&qpcustomb=1&qpct=2&qpsp=180&qpnp=11&qptimeframe=M.

- Es el sistema operativo utilizado por varios fabricantes de teléfonos móviles y tabletas, como: Samsung, LG, HTC, Motorola, Sony, Micromax y otros. Por lo tanto el usuario cuenta con amplias posibilidades de elección y cada una de estas marcas cuenta con dispositivos de diferente gama en prestaciones y costo. De acuerdo a los datos conseguidos del sitio web de la compañía StatCounter, como se pueden ver en la figura 2.2.3.

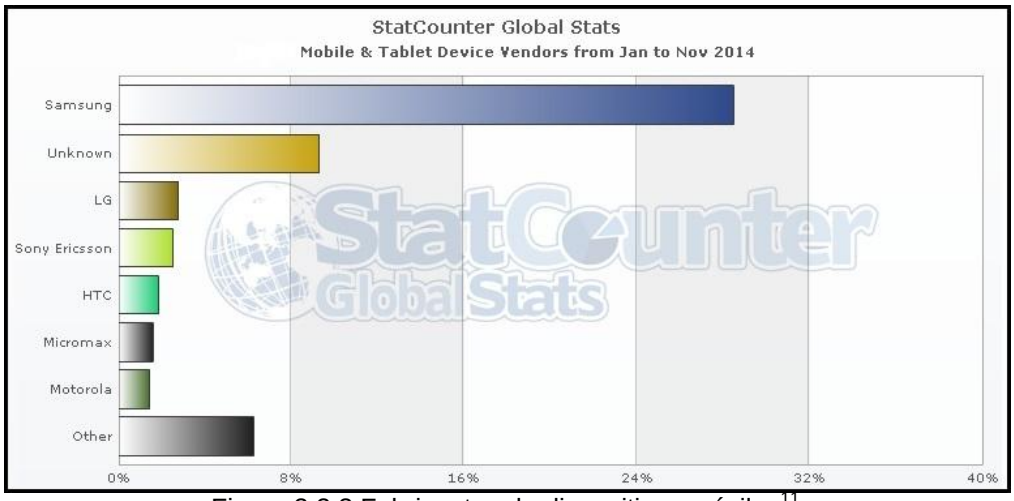


Figura 2.2.3 Fabricantes de dispositivos móviles[11]

- Dispone de un sitio web propio, Google Play, para la descarga de aplicaciones. Actualmente cuenta con alrededor de 1412000 apps disponibles, entre gratuitas y pagadas. En la figura 2.2.4 se muestra el crecimiento de las apps, según datos tomados del sitio web de AppBrain en noviembre del 2014.

[11] Fuente: StatCounter. http://gs.statcounter.com/#mobile+tablet-vendor-ww-monthly-201401-201411-bar. Editado por el autor

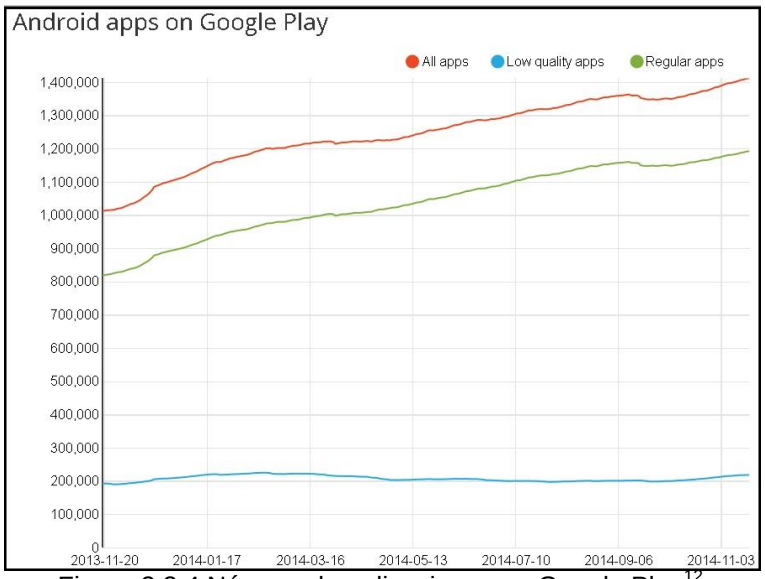


Figura 2.2.4 Número de aplicaciones en Google Play[12]

- Se encuentra disponible en Internet gran cantidad de información, tutoriales, cursos, ejemplos de aplicaciones y libros electrónicos sobre el sistema operativo Android.

En conclusión el sistema operativo Android es más ventajoso para desarrollar aplicaciones móviles; por tratarse de un sistema de código abierto, estar libre de licencias, no existir pago alguno por distribución o desarrollo, por encontrarse a disposición una gran cantidad de información en Internet y ser el más utilizado en los dispositivos móviles de los fabricantes más reconocidos a nivel mundial.

2.2.3 Características generales

El real atractivo de Android, como un entorno de desarrollo, se encuentra en la interfaz de programación de aplicaciones (API) que pone a disposición de los desarrolladores novatos, dándoles la oportunidad de crear sus propias aplicaciones para teléfonos móviles y tabletas. Android posee un conjunto de aplicaciones nativas como:

- Un cliente de correo electrónico.
- Una aplicación de gestión de SMS.

[12] Fuente: AppBrain. http://www.appbrain.com/stats/number-of-android-apps. Editado por el autor

- Un conjunto completo para la administración de la información personal que incluye una lista de contactos y un calendario.
- Un navegador web basado en WebKit.
- Un reproductor de música y galería de fotos.
- Una aplicación de la cámara y grabación de vídeo.
- Una calculadora.
- Pantalla de inicio.
- Un reloj con alarma.

A continuación se detallan las características del sistema operativo Android más destacables:

- Redes GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, LTE y WiMAX para telefonía o transferencia de datos.
- Hardware de acceso Wi-Fi.
- SQLite, una base de datos ligera para el almacenamiento de datos.
- API completa para los servicios basados en localización, GPS.
- Total control de hardware multimedia, incluyendo la reproducción y la grabación con la cámara y micrófono.
- Soporte de archivos multimedia: H.263, H.264, MPEG-4, AMR, AMR-WB, AAC, HE-AAC, MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF y BMP.
- API para el uso de hardware de sensores, como acelerómetro, giroscopio, brújula y de proximidad.
- Librerías para el uso de Bluetooth y NFC en la transferencia de datos punto a punto.
- Paso de mensajes en la Comunicación entre Procesos, IPC.
- Aplicaciones y procesos en segundo plano.
- Widgets en pantalla de inicio, carpetas y fondos de pantalla.
- Soporta la multitarea de aplicaciones.
- Un navegador integrado de código abierto basado en WebKit HTML5.
- Soporte completo para las aplicaciones de mapas, que integran controles en la interfaz de usuario.

- Acelerador gráfico móvil optimizado, incluye la ruta a la librería de gráficos 2D y soporte para gráficos en 3D usando OpenGL ES 2.0.
- Un sistema que fomenta la reutilización de los componentes de la aplicación y la sustitución de aplicaciones nativas. [15]

2.2.4 Arquitectura

La arquitectura interna de la plataforma Android, se encuentra fundamentalmente constituida por cuatro capas, de la forma que se detalla en la figura 2.2.5.

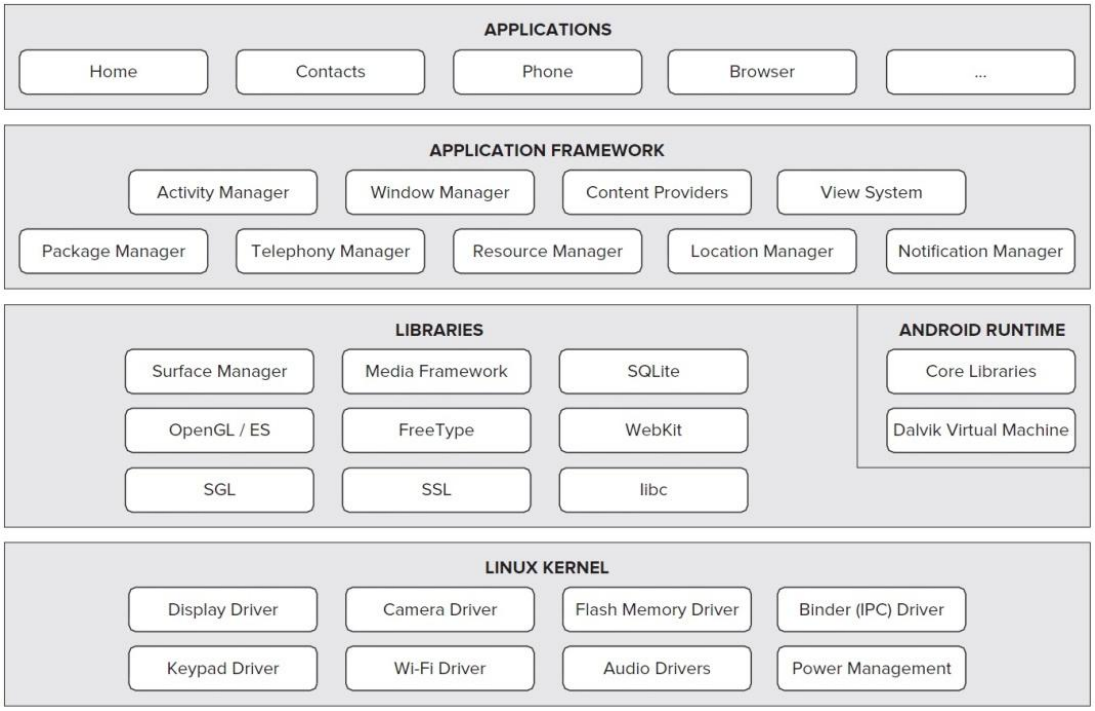


Figura 2.2.5 Arquitectura de Android[13]

2.2.4.1 *Kernel de Linux*

El kernel de Linux es la base de Android, utilizado por su robustez demostrada y por la implementación de funciones básicas para cualquier sistema operativo, como: seguridad, administración de memoria y procesos, implementación de conectividad de red y varios controladores para la comunicación con los dispositivos físicos.

[13] Fuente: Lee, Wei-Meng. *Beginning Android 4 Application Development*. Disponible en: http://filepi.com/i/38lghlw

Android emplea como base el kernel de Linux, pero los dos sistemas no son lo mismo, Android no cuenta con un sistema nativo de ventanas de Linux, ni tiene soporte para la librería estándar de C, ni tampoco es posible utilizar la mayoría de aplicaciones de GNU de Linux.
Además sobre todo lo implementado en el kernel de Linux, Android agrega algunas cosas específicas para plataformas móviles como la comunicación entre procesos, la forma de manejar la memoria compartida y la administración de energía.

2.2.4.2 *Librerías y ejecución*

Esta capa comprende un conjunto de librerías de C y C++ utilizadas por el sistema para varios fines como: el manejo de la pantalla (surface manager), el uso de mapas de bits y tipos de letra (Free Type), el despliegue de gráficos en 2D y 3D (SGL y OpenGL), la utilización de multimedia (Media Framework), el almacenamiento de datos (SQLite) y el empleo del navegador web (WebKit).
Junto a estas librerías, se encuentra lo necesario para la ejecución de las aplicaciones a través de la máquina virtual Dalvik. Cada aplicación utiliza una instancia de la máquina virtual ejecutando un archivo DEX y el sistema está optimizado para que se ejecuten múltiples instancias de la máquina virtual. Se desarrolla en Java, pero no se utiliza una máquina virtual de Oracle para su ejecución, ni tampoco archivos CLASS.

2.2.4.3 *Estructura de aplicaciones*

Esta estructura brinda un contexto para crear aplicaciones, permite a los desarrolladores aprovechar un sistema de vistas ya construido, administrar notificaciones, obtener datos a través de proveedores de contenido, entre otros beneficios.

2.2.4.4 *Aplicaciones*

Esta capa contiene un conjunto de aplicaciones que incluye el sistema operativo por defecto como: teléfono, navegador web, manejo de contactos, reloj, entre otras. Además se encuentra el grupo de las aplicaciones descargadas desde Google Play Store. [16]

2.3 APLICACIONES MÓVILES ANDROID

Las aplicaciones móviles están escritas en lenguaje de programación Java adecuado para Android. El software de desarrollo compila el código con todos los datos y archivos necesarios en un paquete APK. Un archivo APK incluye todo el contenido de una aplicación y es el utilizado por los dispositivos móviles para su instalación.

Una vez instalada en un dispositivo, cada aplicación tiene su propio entorno limitado de seguridad:

- El sistema operativo Android es un sistema Linux para múltiples usuarios en el que cada aplicación es un usuario distinto.
- Automáticamente, el sistema operativo establece a cada aplicación un identificador único de usuario de Linux (el ID es usado sólo por el sistema y es desconocido para la aplicación). El sistema establece los permisos para todos los archivos de la aplicación, para que sólo el ID de usuario designado a la aplicación en cuestión pueda acceder a sus archivos.
- Cada procedimiento posee su específica máquina virtual, por lo que el código de una aplicación se ejecuta en forma aislada de otras aplicaciones.
- De forma predeterminada, cada aplicación se ejecuta en su propio proceso de Linux. Android inicia el procedimiento cuando determinado componente de la aplicación debe ser ejecutado, luego cierra el proceso cuando ya no es necesario o cuando el sistema tiene que restablecer la memoria para otras aplicaciones.

De este modo, el sistema operativo Android utiliza el principio de mínimo privilegio. Es decir, cada aplicación, por defecto, sólo puede acceder a los elementos que requiere para realizar su trabajo y nada más. Esto crea un ambiente muy seguro, donde una aplicación no puede tener acceso a partes del sistema sin tener permiso.

No obstante, existen formas para que una aplicación pueda compartir datos con otras aplicaciones o tenga acceso a los servicios del sistema operativo:

- Es viable establecer que dos aplicaciones compartan el mismo ID de usuario, en donde cada una es capaz de acceder a sus propios archivos. Para conservar los recursos del sistema, las aplicaciones con un mismo ID

de usuario, también pueden ejecutarse en un mismo proceso de Linux y compartir la misma máquina virtual, para esto, las aplicaciones deben estar validadas con el mismo certificado.

- Una aplicación puede pedir permiso para tener acceso a la infromación del equipo móvil, como a los contactos del usuario, mensajes SMS, almacenamiento externo en tarjeta SD, cámara, Bluetooth y otros. Todos los permisos de la aplicación deben ser concedidos por el usuario durante la instalación.

Los elementos fundamentales que permiten el funcionamiento y la existencia de una aplicación en un sistema Android, se presentan en la figura 2.3.1:



Figura 2.3.1 Elementos de una aplicación móvil

2.3.1 Componentes básicos

Los componentes esenciales de una aplicación Android constituyen un punto diferente a través del cual el sistema puede entrar a la aplicación. No todos los componentes son puntos de entrada reales para el usuario y algunos dependen unos de otros, pero existen como una entidad propia y cada uno desempeña un papel específico, como un bloque único que brinda apoyo para especificar el comportamiento general de la aplicación.

Cada componente tiene un propósito particular y un ciclo de vida distinto, que puntualiza cómo el componente se forma y se extingue; dichos componentes son los siguientes:

2.3.1.1 *Actividades*

Una actividad representa una pantalla o ventana única con una interfaz gráfica de usuario. Como por ejemplo, una aplicación de correo electrónico puede tener una actividad que muestra la lista de mensajes nuevos, otra actividad para redactar un correo, otra para la lectura de mensajes. Si bien las actividades trabajan en conjunto para crear una experiencia de usuario razonable, en la aplicación de correo electrónico, cada una es independiente de las demás. Como tal, una aplicación diferente puede iniciar dichas actividades (si la aplicación de correo electrónico lo permite). Por ejemplo, una aplicación de cámara puede iniciar la actividad en la aplicación de correo electrónico que compone un correo nuevo, con la finalidad que el usuario envíe una fotografía.

Una actividad se implementa como una subclase *Activity*.

2.3.1.2 *Servicios*

Un servicio es un elemento, cuya ejecución se hace en segundo plano para efectuar acciones de larga realización o para cumplir trabajos de procedimientos remotos. Este elemento no tiene una interfaz gráfica de usuario. Por ejemplo, la reproducción de música, es un servicio en segundo plano, mientras el usuario está en una aplicación diferente; o la obtención de datos mediante la red sin cortar la interacción del usuario con una actividad.

La implementación de un servicio se efectúa como una subclase *Service*.

2.3.1.3 *Proveedores de contenido*

Un proveedor de contenido administra un grupo de aplicaciones con información compartida. Puede contener datos en el sistema de archivos, en una base de datos SQLite, en la web, o en cualquier otro lugar de almacenamiento permanente, donde la aplicación pueda acceder. A través del proveedor de contenido, otras aplicaciones pueden buscar o inclusive cambiar la información (si el proveedor de contenido lo permite). Por ejemplo, el sistema operativo Android proporciona un proveedor que permite gestionar el contenido de los contactos de usuario. Como tal, cualquier aplicación con las aprobaciones respectivas puede leer y

escribir información de una persona en particular de la lista de contactos de usuario.

Los proveedores de contenido también son útiles para la lectura y escritura de datos para uso personal en una aplicación y no se los comparte. Por ejemplo, el *Bloc de Notas* es una aplicación que utiliza un proveedor de contenido para archivar notas.

Un proveedor de contenido se implanta como una subclase *ContentProvider* y tiene que implementar un grupo estándar de API que facilite a otras aplicaciones efectuar transacciones.

2.3.1.4 *Receptores de difusión*

Un receptor de difusión es un componente que permite transmitir mensajes. Diversas difusiones se producen en el sistema, por ejemplo, una emisión para avisar, por ejemplo: la pantalla se ha apagado, la batería está baja o una imagen fue capturada. Las aplicaciones también consiguen empezar las emisiones, por ejemplo, para admitir que otras aplicaciones empleen ciertos datos que se han descargado al equipo móvil y éstos pueden ser aprovechados. Sin embargo los receptores de difusión no exponen una interfaz gráfica de usuario, es posible crear una notificación en la barra de estado, para advertir al usuario cuando se ocasiona un evento de difusión. Sin embargo, un receptor de difusión es sólo un ingreso a otros elementos y tiene la intención de hacer muy mínima la cantidad de trabajo.

Un receptor de difusión se implementa como una subclase *BroadcastReceiver* y cada difusión se transfiere como un objeto *Intent*.

Las actividades, servicios y receptores de difusión se activan mediante un mensaje asíncrono llamado Intent (intención). Las intenciones unen componentes individuales entre sí en tiempo de ejecución, si el componente pertenece a la aplicación o a otra.

Una intención se crea con un objeto *Intent*, que define un mensaje para activar un componente específico en forma explícita o un componente con una intención específica en forma implícita.

Para las actividades y servicios, la intención define la acción a realizar, por ejemplo una "vista" o "enviar" y puede especificar el URI de los datos para actuar con otras cosas del componente. Por ejemplo, la intención podría transmitir una solicitud de actividad para exponer una imagen o desplegar una página web. En unas ocasiones, puede comenzar una actividad para presentar un resultado.

Para los receptores de difusión, la intención simplemente define el mensaje que se está emitiendo, por ejemplo una emisión para indicar que la batería del dispositivo está baja.

En cambio un proveedor de contenido, no se activa por intenciones, se activa por los métodos de objeto *ContentResolver*. Lo que deja una capa de abstracción entre el proveedor de contenido y la información del componente que la solicita, esto por aspectos de seguridad.

2.3.2 Archivo Manifest

Antes de que Android pueda iniciar un componente de una aplicación, el sistema debe saber que el componente existe leyendo el archivo “AndroidManifest.xml”. El programador debe declarar todos los componentes de la aplicación en dicho archivo, el cual debe estar en la raíz del directorio del proyecto.

El manifiesto realiza otras funciones, como las siguientes:

- Identificar los permisos de usuario que la aplicación requiere, tales como acceso a Internet, permiso de lectura a los contactos de usuario, entre otros.
- Señalar el mínimo nivel requerido de API para que la aplicación funcione en un dispositivo móvil.
- Declarar las características de hardware y software utilizadas o requeridas por la aplicación, tales como la cámara, los servicios de bluetooth, una pantalla multitouch, entre otras.
- Establecer las librerías API que la aplicación necesita, tales como las librerías de Google Maps.
- Detallar la actividad o las actividades que conforman la aplicación.

2.3.3 Recursos

Una aplicación se compone de algo más que código, necesita recursos que están separados del código fuente, como imágenes, archivos de audio y otros relacionados con la presentación visual de la aplicación. Por ejemplo, se debe definir animaciones, menús, estilos, colores y diseño de interfaces de usuario con archivos XML. El uso de los recursos por una aplicación hace que sea fácil actualizar varias características sin modificar el código y proporcionando un conjunto de alternativas, los recursos permiten al desarrollador optimizar la aplicación para una variedad de configuraciones de dispositivos, como diferentes lenguajes y tamaños de pantalla.

Para cada recurso que se incluya en un proyecto Android, las herramientas de desarrollo SDK definen un número único ID, que se lo puede utilizar en el código para hacer referencia a los recursos que emplea la aplicación o a otros recursos definidos en el formato *XML*. Como por ejemplo, si la aplicación dispone de un archivo de imagen llamado *figura.png*, debe estar guardado en la carpeta *res/drawable*, las herramientas SDK generan un identificador de recurso llamado *R.drawable.figura*, para que sirva de referencia de dicha imagen y poder colocarla en la interfaz gráfica de usuario.

Una de las propiedades más importantes del suministro de recursos por separado a partir del código fuente, es la capacidad para que el programador pueda proporcionar recursos alternativos a diferentes configuraciones de dispositivos. [17]

2.4 KIT DE DESARROLLO DE SOFTWARE

Se trata en una Interfaz de Programación de Aplicaciones, API, creada para permitir el uso de cierto lenguaje de programación o puede también incluir hardware sofisticado para comunicarse con un determinado sistema embebido. Las herramientas más comunes incluyen: soporte para la detección de errores de programación, un Entorno de Desarrollo Integrado, IDE, y otras utilidades. Los

SDK frecuentemente contienen códigos de ejemplo y notas técnicas de soporte u otra documentación, para ayudar a clarificar ciertas dudas. [18]

Google Labs pone a disposición de los desarrolladores el programa App Inventor para Android en versión beta, para la creación de aplicaciones de forma visual a partir de un conjunto de herramientas básicas. El usuario puede ir enlazando una serie de bloques para crear la app. El sistema es gratuito y se puede descargar fácilmente de la web. [19]

Android ofrece a los programadores las herramientas siguientes para el desarrollo de apps:

- Android SDK
- Android Studio
- Android NDK

Android SDK, utiliza el lenguaje de programación Java, con herramientas que facilitan la creación de aplicaciones y depuración de su ejecución con utilidades adecuadas. Éstas se encuentran integradas en el programa Eclipse, el cual utiliza el complemento de Herramientas de Desarrollo Android, ADT, pero es posible ejecutarlas individualmente.
ATD viene incluido en el IDE Eclipse, para conseguir un mayor potencial en el desarrollo e integración de aplicaciones, confiriendo, en líneas generales, una mayor versatilidad para una mejor construcción de aplicaciones. Estas herramientas amplían las capacidades ya existentes de Eclipse, permitiendo configurar rápidamente nuevos proyectos para Android, crear una interfaz de usuario de aplicación, agregar los paquetes basados en la estructura API de Android, depurar sus aplicaciones utilizando las herramientas de SDK e incluso exportar con firmas las APK generadas a fin de distribuirlas adecuadamente. Soporta NDK. [18]

Android Studio, es un nuevo entorno de desarrollo de Android basado en IntelliJ IDEA. Ofrece nuevas características y mejoras con respecto a Eclipse ADT y será el IDE oficial de Android, actualmente se encuentra en versión beta. Algunas

características todavía no están implementadas y pueden presentar errores. No soporta NDK. [20]

Android NDK, es un conjunto de herramientas que le permite implementar apps utilizando lenguaje nativo C y C ++. Se lo recomienda para aplicaciones con cargas de trabajo intensivas de la CPU, como: motores de juego, procesamiento de señales, simulación física, entre otras. Para desarrollar en código nativo, se debe pensar en las necesidades de la aplicación a crear y ver si NDK proporciona la funcionalidad que se requiere. [21]

En conclusión, de las cuatro opciones descritas anteriormente, Android SDK es la mejor alternativa para el desarrollo de apps, porque cuenta con todas herramientas necesarias para el desarrollo de aplicaciones y de una estructura de API, está disponible en versiones estables, soporta NDK, existe gran cantidad de material de ayuda o consulta en Internet, se hallan muchísimos ejemplos de aplicaciones en la web, ofrece una interfaz gráfica de usuario amigable, entre lo más destacado de este kit de desarrollo.

2.5 EMULADORES

2.5.1 Dispositivo Virtual Android, AVD

Un AVD es una configuración de emulador que permite simular un dispositivo real; mediante la definición de las opciones de software y un hardware determinado, emula una versión de Android, el dispositivo virtual se encuentra integrado en el IDE Eclipse. Este emulador se trata de:

- Un perfil de hardware que define las características del dispositivo virtual. Por ejemplo, se puede definir si el dispositivo tiene una cámara, si utiliza un teclado QWERTY físico o un teclado para marcar, que cantidad de memoria dispone, entre otras.
- Un mapeo de una imagen del sistema, donde se puede definir qué versión de Android se ejecutará en el dispositivo virtual. Se puede

especificar el aspecto del emulador que se desea utilizar con la AVD, dimensiones de pantalla, apariencia, entre muchas otras.

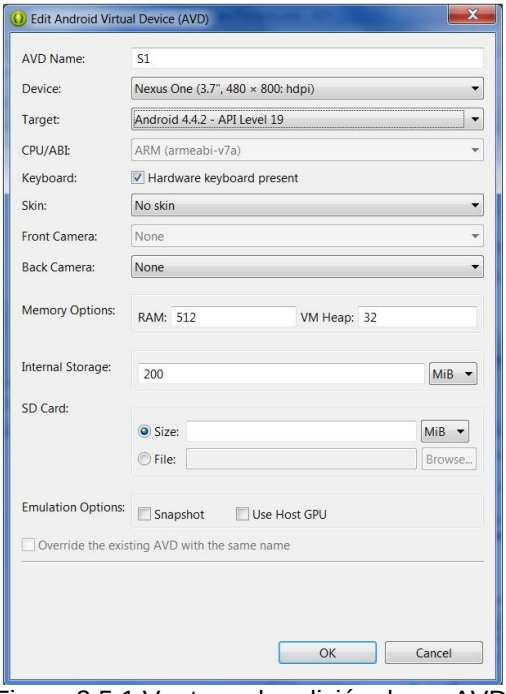


Figura 2.5.1 Ventana de edición de un AVD

AVD despliega una interfaz que permite modelar diferentes configuraciones de dispositivos virtualizados, emulando la ejecución de las distintas versiones, KitKat, Jelly Bean, Ice Cream Sandwich, entre otras, para conocer el funcionamiento de las aplicaciones creadas en los distintos entornos y poder depurarlas adecuadamente.

Para sacar el máximo provecho a las herramientas SDK y aprender e interiorizar correctamente sus funcionalidades, es importante centrarse en los apartados siguientes:

- Usar la documentación Android para tener los recursos suficientes y poder obtener respuestas a las inquietudes que se presenten en el desarrollo de aplicaciones, ya que la programación es un proceso complejo que requiere de retroalimentación continua y ayuda, para poder llevarla a cabo sin perder el ánimo y enfrentarse a obstáculos importantes.
- Depurar aplicaciones con el Servicio de Monitoreo de Depuración Dalvik, DDMS, o con el Monitoreo de Depuración Android, ADM.
- Emular con AVD para reproducir el comportamiento de las distintas versiones de Android, tratando de imitar de la forma más exactamente

posible el funcionamiento de un dispositivo, como si se estuviese usado un real.

- Conocer la estructura del sistema Android, como: configuración, menús, pantallas, entre otras disposiciones. [18]

2.5.2 Genymotion

Es un emulador de Android que aprovecha la arquitectura x86 para ejecutar de forma fluida y rápida diferentes dispositivos Android. Olvidando la lentitud del emulador nativo de Android, se puede ejecutar todo tipo de aplicaciones y juegos en Windows, Mac o Linux.

Uno de los principales usos de Genymotion es facilitar el desarrollo de aplicaciones Android. Se integran perfectamente con el Puente de Depuración Android, línea de comandos y los diferentes entornos de desarrollo. Se pueden lanzar en apenas un par de segundos las aplicaciones conectadas al ATD desde Eclipse para probarlas.

Genymotion está basado en el uso de máquinas virtuales x86 optimizadas para correr sobre Virtualbox. Los creadores de este emulador han conseguido crear una interfaz simple capaz de soportar distintas funcionalidades accesibles a cualquier usuario, sin olvidar a los desarrolladores.

Al descargar Genymotion, se lo instala fácilmente realizando un par de clics, y luego se selecciona cualquiera de las máquinas virtuales que emulan dispositivos como: Nexus 4, Nexus 5, Nexus 7, Galaxy Nexus, HTC One, Motorola X, Galaxy S5, Xperia Z, entre otros, para distintas versiones de Android con diferentes resoluciones de pantalla.

Entre todas de las características que aporta el emulador de Genymotion se encuentran: el uso de conexión a internet, la simulación de ubicaciones GPS con el widget que facilita la búsqueda y el posicionamiento sobre un mapa, simulación de la cámara, estado de la batería, rotación del dispositivo, entre otras funciones.



Figura 2.5.2 Dispositivo virtual Samsung S3

Por problemas de licencia, este emulador no incluye por defecto las aplicaciones de Google, aunque se está trabajando para conseguir un acuerdo con Google. Pero no hay problema, ya que Genymotion permite instalar cualquier APK, simplemente arrastrándolo sobre el emulador. [22]

De los dos emuladores detallados, se concluye que Genymotion es la mejor opción que permite reproducir un dispositivo real de manera virtual, permitiendo seleccionar algunos modelos de varios fabricantes como Samsung, Sony, HTC, Motorola, Google y LG con distintas versiones de Android. Este emulador es excelente debido a su ejecución de forma más rápida y fluida, aprovechando las bondades de la arquitectura x86 con máquinas virtuales, en comparación a la lentitud del emulador nativo del SDK Android.

2.6 GEOLOCALIZACIÓN

La geolocalización es una tecnología que en la actualidad se encuentra en boga, siendo utilizada en el sector automotriz, aplicaciones móviles, redes sociales,

entre otros ámbitos. Donde se hace referencia al conocimiento de la ubicación geográfica de un objeto de manera automática.

También denominada georreferenciación, se trata de un conjunto organizado de hardware y software, más datos geográficos, que se encuentra diseñado especialmente para capturar, almacenar, manipular y analizar en todas las posibles formas la información geográfica referenciada, con la clara misión de resolver problemas de gestión y planificación.

Existen varias alternativas para conocer dicha ubicación, aunque claramente, son los dispositivos móviles que por su portabilidad, permiten más fácilmente conocer la ubicación y actualizarla a medida que el usuario se vaya movilizando y por tanto, cambiando de ubicación geográfica.

Entre las varias opciones que permiten encontrar la geolocalización, se destacan los teléfonos inteligentes, que cuentan con funciones especializadas, en este caso particular traen integrados receptores de GPS, los cuales y gracias a la red de satélites que rodea al planeta, éstos podrán ubicar en cualquier lugar del globo terráqueo donde se encuentra el dispositivo.

Otra alternativa ampliamente usada y a disposición de aquellos que no poseen dispositivos móviles, es Google Earth, que se trata de un programa informático, que permite obtener imágenes del planeta en tecnología 3D en combinación con imágenes de satélite, mapas y el motor de búsqueda de Google y así facilitar la visualización de imágenes a escala de un lugar del planeta, que se pretenda descubrir o identificar. [23]

2.6.1 GPS

Un sistema de posicionamiento global tiene como objetivo la determinación de las coordenadas espaciales de puntos, respecto de un sistema de referencia mundial. Los puntos pueden estar ubicados en cualquier lugar del planeta, pueden permanecer estáticos o en movimiento y las observaciones pueden realizarse en cualquier momento del día.

Para la obtención de coordenadas geográficas, el sistema se basa en la determinación simultánea de las distancias a cuatro satélites de coordenadas conocidas, como mínimo. Estas distancias se obtienen a partir

de las señales emitidas por los satélites, las que son recibidas por receptores especialmente diseñados. Las coordenadas de los satélites son provistas al receptor por el sistema.

Desde el punto de vista geodésico-topográfico, este sistema responde a dos requerimientos básicos:

- *Planteo directo o levantamiento*: Se tiene en el terreno un punto materializado, un pilar con placa, un hito o cualquier otra marca. Se piden sus coordenadas en un sistema de referencia preestablecido.
- *Planteo inverso o replanteo:* Se dan las coordenadas de un punto en un sistema de referencia determinado y se pide la localización de dicho punto, que de no estarlo ya, será materializado en el terreno.

El acceso a las señales que emiten los satélites es de carácter público, no requiriéndose licencia o autorización alguna. Actualmente, el uso civil de GPS ha sobrepasado largamente al uso militar, convirtiéndose de hecho en un servicio público de carácter mundial de enorme importancia y con innumerables apps y dispositivos. [24]

2.7 GOOGLE SITES

Se trata de una aplicación online que permite crear intranets, extranets para clientes, sitios propios y de colaboración online, wikis, pequeños sitios de seguimiento de proyectos; todo esto tan fácil como editar un documento de texto. Admite recopilar con rapidez, información de distinta naturaleza en una misma ubicación (vídeos, calendarios, presentaciones, archivos adjuntos y texto) y compartirla fácilmente con un pequeño grupo de personas, toda una organización o con cualquier usuario de internet para verla o editarla, según sea el caso.

2.7.1 Características

- Personalización de la interfaz del sitio al gusto del creador para que se adapte a sus necesidades. No se requiere conocimientos de programación HTML u otro lenguaje.
- Elección de distintos tipos de página; como: página web, anuncios, archivador y lista.
- Cambio de la URL de Google Sites por un dominio propio. Por ejemplo: "example.com" como dominio, y "tusitio" como sitio, la URL estándar para un sitio es https://sites.google.com/site/tusitio. Esto se podría cambiar a http://www.example.com a través de la Consola del administrador de Google.
- Centralización de la información compartida, insertar contenido enriquecido (vídeos de YouTube; documentos, hojas de cálculo, presentaciones de Google Docs; presentaciones de diapositivas de fotos de Picasa, gadgets de iGoogle) en cualquier página y subir archivos adjuntos.
- Administración de la configuración de permisos para mantener un grado de privacidad, de libertad de edición o de visualización que desee el creador del sitio. Solo los usuarios asignados como colaboradores, tendrán derechos de edición.
- Realización de búsquedas en su contenido de Google Sites mediante la tecnología de búsqueda de Google.
- Asignación de almacenamiento 100 MB/sitio. Los documentos insertados, incluidos los documentos de Google Docs, no se contabilizan para la cuota de almacenamiento, pero los elementos que se añaden como un archivo adjunto o en la página de almacenamiento de archivos, sí. El tamaño máximo de un archivo adjunto es de 20 MB.
- Número de páginas por sitio ilimitado. El diseño de un sitio con miles de páginas y archivos adjuntos, puede ralentizarlo y crear una experiencia de usuario deficiente. Estos sitios, o sus páginas individuales, pueden inhabilitarse o en ocasiones, suprimirse temporalmente. Se recomienda dividir estos sitios de gran tamaño en varios sitios relacionados o inhabilitar los elementos de la barra lateral

dinámica como, por ejemplo, la navegación organizada automáticamente y la actividad reciente de los sitios. [25]

2.7.2 Privacidad

La privacidad es un aspecto que Google, empresa multinacional estadounidense, ha tomado muy en serio y cree firmemente que los productos y servicios que ofrece son una excelente opción en este sentido. Esta empresa se basa en la confianza de los usuarios, fundamentada en la capacidad de Google para proteger los datos adecuadamente y el compromiso de respetar la privacidad de la información que aportan a sus sistemas, sin proporcionarla a otros usuarios y sin hacer un uso inadecuado de ella.

Google no es el propietario de los datos de los usuarios. No se posiciona respecto a si los datos pertenecen a una institución registrada o a un usuario, pero sí está claro, que no le pertenecen.

Los datos que introducen los usuarios a los sistemas de Google son de propiedad de los usuarios y la empresa cree que así debe ser. Esto implica tres cuestiones fundamentales:

- No se los comparte con nadie, excepto en los casos que se especifican en la política de privacidad.
- Guarda los datos el tiempo que el usuario quiera.
- Debería poder migrar dichos datos, en caso de que opte por utilizar servicios externos junto con los de Google o si deja de utilizar los servicios por completo.

Sólo puede compartir información con terceros en virtud de lo establecido en la política de privacidad y en el acuerdo con el cliente. No facilita a terceros, ni comparte con ellos, información confidencial de los usuarios como por ejemplo: datos personales o del correo electrónico, salvo si la legislación vigente así lo requiere, si un usuario del sistema lo solicita o si la finalidad de ello es proteger los sistemas de Google. [26]

2.7.3 Seguridad

Google, como proveedor consolidado de servicios basados en web, hace todo lo posible para proteger su infraestructura de las amenazas. Para ello, utiliza hardware personalizado con un sistema operativo y un sistema de archivos exclusivos en sus centros de datos. Todos y cada uno de estos sistemas se han optimizado para ofrecer la máxima seguridad y rendimiento. El equipo de protección de la información de Google cuenta con colaboradores externos para probar y mejorar continuamente su infraestructura de seguridad y, de este modo, garantizar su impermeabilidad ante cualquier ataque. Además, puesto que Google controla todos los elementos que conforman sus sistemas, puede responder rápidamente a cualquier amenaza o peligro que pueda surgir.

Google cuenta con varios centros de datos repartidos por distintas zonas del planeta. Los clústeres de equipos se han diseñado para ofrecer la máxima resistencia y redundancia, así como para eliminar los puntos únicos de fallo y minimizar las posibles consecuencias de los errores habituales de los equipos y los riesgos medioambientales.

Las aplicaciones que ofrece Google disponen de certificados de seguridad. Esto significa que un auditor externo independiente ha concedido dichos certificados por haber superado satisfactoriamente la auditoría de seguridad realizada. Google está orgulloso de proporcionar a los usuarios la tranquilidad de saber que sus datos están protegidos por los estándares de auditoría SSAE 16, ISAE 3402 tipo II y SAS 70 tipo II.

El auditor independiente externo ha comprobado que Google aplica los controles y protocolos siguientes:

- *Seguridad lógica:* Los controles proporcionan una garantía razonable de que el acceso lógico a los sistemas de producción y a los datos de las aplicaciones de Google, está restringido, inclusive a las personas autorizadas.
- *Privacidad:* Los controles proporcionan una garantía razonable de que Google ha implementado políticas y procedimientos, en torno a la privacidad de los datos de los clientes de las aplicaciones de Google.

- *Seguridad física de los centros de datos:* Los controles proporcionan una garantía razonable de que los centros de datos, donde se aloje la información de las aplicaciones de Google y las oficinas corporativas están protegidos.
- *Gestión de incidentes y disponibilidad:* Los controles proporcionan una garantía razonable de que los sistemas de Google son redundantes y de que los incidentes se notifican, se tratan y se registran correctamente.
- *Gestión de cambios:* Los controles proporcionan una garantía razonable de que el desarrollo de las aplicaciones de Google y las modificaciones que se aplican a este servicio, se someten a pruebas y a revisiones de código independientes antes de su lanzamiento.
- *Organización y administración:* Los controles proporcionan una garantía razonable de que la administración ofrece la infraestructura y los mecanismos necesarios, para realizar el seguimiento de las iniciativas de la empresa que afectan a las aplicaciones de Google y para comunicarlas.

Google ofrece una conectividad segura con el protocolo SSL/TLS, permitiendo establecer comunicaciones seguras en Internet, como: navegación web, correo electrónico, mensajería instantánea y de otros sistemas de transferencia de datos. Si se habilita la conexión HTTPS, Google aplicará este protocolo cuando los usuarios accedan a la mayoría de los servicios de Google. Dicho protocolo varía en función del servicio y está disponible para Gmail, Google Calendar, Google Drive, Google Sites y chat. [26]

2.8 REDES DE DATOS MÓVILES

La telefonía móvil ha avanzado muy rápidamente, lo que ha permitido saltar de la básica tarea de comunicación por voz, hasta la descarga de datos de Internet, ver películas, realizar videoconferencias, utilizar la geolocalización, disfrutar de la

realidad aumentada, entre otras aplicaciones. Todo, gracias al desarrollo de la tecnología, que entrega dispositivos cada vez más sofisticados y con la implementación de nuevos estándares técnicos.



Figura 2.8.1 Iconos de redes móviles[14]

Los usuarios para disfrutar los beneficios de las redes móviles deberán disponer de un teléfono celular y contratar un plan de datos con una operadora de telefonía móvil. De acuerdo a la tecnología empleada por la proveedora del servicio el suscriptor tendrá a su disposición las siguientes tecnologías móviles: GPRS, EDGE, 3G, H, H+ y 4G.

GPRS: La tecnología de Servicio General de Paquetes vía Radio es una extensión del sistema Global para comunicaciones Móviles, GSM, para la transmisión de datos no conmutada o por paquetes. Permite velocidades de transferencia de 56 a 144 kbps. Un dispositivo móvil al conectarse a una red GPRS, en la parte superior aparece la letra G.

EDGE: La tecnología de Tasa de Datos Mejorada para la Evolución de GSM es un sistema basado en GPRS y puede funcionar en un máximo de 384 kbps. Si un teléfono celular es compatible con EDGE, puede ser utilizado para la transmisión de datos móviles, tales como la recepción de archivos adjuntos en el correo electrónico y navegar por sitios web complejos. Cuando un dispositivo móvil accede a una red EDGE, en la parte superior aparece la letra E.

[14] Fuente: Emezeta, *¿Qué significa G, E, 3G, H/3G+, H+, 4G?*. http://i.emezeta.com/weblog/redes-moviles/redes-moviles-g-e-3g-h-4g.png

3G: Utiliza el Sistema Universal Telecomunicaciones Móviles, UMTS, llamado 3G por ser la tercera generación de sistemas para móviles. Los servicios asociados con esta generación proporcionan la posibilidad de transferir tanto voz (llamada telefónica) y datos (descarga de programas, envío/recepción de correo electrónico y mensajería instantánea). Permite velocidades de conexión de hasta 2 Mbps. Al conectarse un teléfono móvil a una red 3G, en la parte superior aparece 3G.

H: Se basa en el Acceso Descendente de Paquetes a Alta Velocidad, HSDPA, también conocida como 3,5G, 3G+ o Turbo3G. Es la optimización de la tecnología espectral UMTS/WCDMA, incluida en las especificaciones de 3GPP Release 5 y consiste en un nuevo canal compartido en el enlace descendente, que mejora significativamente la capacidad máxima de transferencia de información, pudiéndose alcanzar tasas de hasta 14 Mbps de descarga, teóricamente hablando, puesto que luego se está sujeto a la voluntad de las operadoras. Un dispositivo móvil al conectarse a una red HSDPA, en la parte superior aparece la letra H o 3G+.

H+: Se fundamenta en el Acceso Ascendente de Paquetes a Alta Velocidad, HSUPA, que es una tecnología de acceso de datos para redes de telefonía móvil con una alta tasa de transferencia, teóricamente de hasta 22 Mbps de subida y 84 Mbps de descarga. Calificado como generación 3,75; 3,75G o 3,5G+, es una evolución de HSDPA. La solución HSUPA potencia inicialmente la conexión de subida del 3G. HSUPA está definido en el Release 6, estándar publicado por 3GPP, como una tecnología que ofrece una mejora sustancial en la velocidad para el tramo de subida, desde el terminal hacia la red. HSDPA y HSUPA, ofrecen altas prestaciones de voz y datos, y permiten la creación de un gran mercado de servicios IP multimedia móvil. HSUPA mejora las aplicaciones de datos avanzados persona a persona, con mayores y más simétricos ratios de datos, como el e-mail en el móvil y juegos en tiempo real con otro jugador. Cuando un dispositivo móvil accede a una red HSUPA, en la parte superior aparece H+.

4G: Se trata de la cuarta generación de conexiones móviles conocida como Evolución a Largo Plazo, LTE. Consiste en el siguiente paso evolutivo de la tecnología UMTS, cuya interfaz radioeléctrica se basa en el Acceso Múltiple por

División de Frecuencia Ortogonal, OFDMA, facilitando mayor velocidad de transferencia de subida y bajada, junto con una importante reducción de la latencia. Las velocidades teóricas de conexión que permite, son de 100 Mbps de descarga y 50 Mbps de subida. Si un dispositivo se conecta a este tipo de red, en la parte superior aparece 4G. [27]

LTE-Advanced es la siguiente evolución de LTE, una tecnología basada en OFDMA, especificada en el Release 8 y 9, soportada por un gran número de fabricantes y operadores en todo el mundo. Esta tecnología es compatible tanto hacia atrás como hacia adelante con LTE, lo que significa que los dispositivos LTE operan en redes LTE-Advanced y los dispositivos LTE-Advanced funcionan en redes LTE. Las velocidades teóricas de conexión son de 1 Gbps de descarga y 500 Mbps de subida. [28]

2.9 RED DE DATOS INALÁMBRICA, WI-FI

La tecnología Wi-Fi permite crear redes informáticas inalámbricas. Es un estándar de la IEEE llamado 802.11. Esta tecnología es cada vez más utilizada por diversos equipos como: laptops, PDAs, teléfonos móviles, tablets, smart TV, consolas de videojuegos, entre otros aparatos e incluso en impresoras para facilitar su conexión.

Su alcance varía de un dispositivo a otro de algunas decenas de metros a varias centenas de metros, lo que la convierte en una tecnología primordial para las redes domésticas con conexión a Internet. [29]

Actualmente en nuestro medio existen redes Wi-Fi abiertas, disponibles a los usuarios en parques, plazas u otros lugares de acuerdo a la implementación que hayan realizado los municipios en sus respectivas ciudades. También la mayoría de centros educativos disponen de esta tecnología en sus instalaciones, siendo además utilizada por algunas organizaciones públicas y privadas.

A continuación se presentan las diferencias al utilizar un dispositivo móvil en una red móvil y en una red Wi-Fi abierta:

Red móvil	Red Wi-Fi
Se requiere la contratación de un plan de datos para su utilización	Totalmente gratis
Se conecta a la red de la operadora	Se conecta a un ruteador
Amplia área de cobertura, permitiendo total movilidad	Cobertura reducida con poca movilidad
Mayor seguridad y privacidad en una conexión	Menor seguridad y privacidad
Gran cantidad de suscriptores usando la red	Cantidad limitada de usuarios
Distintas velocidades teóricas de conexión, según la tecnología usada en la red: 56; 144; 384 kbps; 2; 14; 22; 50; 84; 100; 500 Mbps y 1 Gbps	Velocidades teóricas según el estándar utilizado en la red: 11; 54; 108; 300; 600 Mbps; 1 Gbps

Tabla 2.9.1 Diferencias entre una red móvil y una red Wi-Fi

De la tabla anterior se concluye que al conectarse a una red móvil, se disfruta de una amplia área de cobertura, gozando de una gran movilidad con una mayor seguridad y privacidad, todo esto bajo la contratación de un plan de datos. En cambio al conectarse a una red Wi-Fi abierta, se tiene mayores velocidades de conexión con poca seguridad y privacidad.

CAPÍTULO 3

ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DEL SITIO WEB

3.1 ANÁLISIS

3.1.1 Requerimientos para la creación del sitio web

3.1.1.1 *Requerimientos técnicos*

Hosting

El sitio web tiene que ser creado en Google Sites, donde toda la información que contenga el sitio, estará guardada en los servidores de la empresa Google. Se escogió esta aplicación online para evitar los problemas de alojamiento del sitio web y el pago por dicho servicio. Dicho sitio forma parte de un proyecto netamente con fines académicos.

Software

Se va a utilizar las aplicaciones y programas siguientes:

- Google Sites para la creación del sitio web.
- Adobe Photoshop CS6 para la creación de collages.
- Microsoft Office Picture Manager y Paint para la edición de imágenes.
- UltraSlideshow Flash Creator para la creación de la galería de imágenes.
- Sothink SWF Easy para la creación del banner de bienvenida.

Lenguaje de programación web

Al utilizar la aplicación online Google Sites, no se necesita saber ningún tipo específico de lenguaje de programación, solamente ser ingenioso e imaginativo para diseñar el sitio.

Hardware

- Computador de escritorio o laptop con procesador de 2GHz, memoria de 2GB y 10GB de espacio en el disco duro como requisitos mínimos y con conexión a internet.

- Cámara fotográfica digital.

3.1.1.2 *Requerimientos funcionales*

Interfaz gráfica

El sitio web debe ofrecer una interfaz gráfica comprensible y de fácil utilización para los usuarios.

Funcionalidades

- El sitio tiene que contar básicamente con las secciones siguientes:
  - Sección principal: Va a contener las categorías de los atractivos urbanos con sus correspondientes submenús. Se debe realizar la actualización de la información de cada uno de los lugares, según sea el caso.
  - Sección lateral: Debe estar formada por módulos con sus respectivos submenús, relacionados con información específica del Cantón Tulcán. Se puede gestionar y actualizar el contenido de esta sección, de acuerdo a las necesidades del sitio.
  - Sección de contenido: En la página principal se debe exponer un banner de bienvenida y una galería de imágenes. En las otras páginas se tiene que mostrar el respectivo contenido informativo, con imágenes, texto y archivos adjuntos según sea el caso.

- El sitio web va a permitir la gestión de usuarios, ya que se trata de un proyecto académico:
  - Colaboradores: Individuos a los cuales se les da privilegios para que puedan ayudar a mejorar el sitio.
  - Invitados: Personas para que puedan visitar el sitio web.

  En ambos casos, deben constar en los registros del creador o dueño del sitio web.

3.1.1.3 *Requerimientos no funcionales*

Usabilidad

El portal web debe tener una interfaz comprensible para el usuario, de tal manera que éste entienda todo lo ve en el sitio web y sepa lo que debe hacer. Para ello se tiene que tomar en cuenta lo siguiente:

- El idioma debe ser el español.
- El formato de cada página debe ser el apropiado. Con colores que no afecten la presentación del contenido del sitio. Con un tipo de letra legible de tamaño adecuado. Con títulos en cada una de las páginas.

Disponibilidad

- El sitio web va a estar disponible los 365 días del año, las 24 horas del día.
- El usuario puede ingresar al sitio utilizando cualquier navegador web.

Rendimiento

- Las páginas que componen el portal web deben tener el menor tamaño posible, los textos e imágenes van a tener un adecuado formato para lograr una rapidez en la navegación.
- El sitio web tiene que soportar los navegadores web más conocidos como: Firefox de Mozilla, Chrome de Google, Internet Explorer de Microsoft, entre otros.
- El sistema puede tener la posibilidad de evolucionar en base a las necesidades que se presenten.

Desempeño

El sitio web va a brindar confiabilidad a los diferentes usuarios, así toda la información disponible, puede ser consultada en todo momento y actualizada sin tener contratiempos en el servicio.

Escalabilidad

El portal web debe permitir en un futuro las acciones de eliminar, modificar y crear funcionalidades una vez que el sitio web este en funcionamiento sin afectar su desempeño actual.

3.1.2 Seguridad del sitio web

Google Sites dispone de certificados de seguridad. Google proporciona a todos sus usuarios la tranquilidad de saber que sus datos están protegidos por estándares de auditoría externa SSAE 16, ISAE 3402 tipo II y SAS 70 tipo II.

Además ofrece una conectividad segura con el protocolo SSL / TLS, permitiendo establecer comunicaciones seguras en Internet, es decir una conexión HTTPS, a los usuarios que acceden a las aplicaciones de Google.

El sitio web, al ser parte de un proyecto totalmente académico está concebido de manera siguiente: la información no va a estar disponible a cualquier persona, sino que van a tener acceso solamente los usuarios que el creador o dueño del sitio tenga previamente en sus registros.

3.1.3 Selección de los atractivos turísticos

Se entiende por atractivo turístico, un lugar de interés que los turistas visitan, normalmente por su valor patrimonial o cultural, su significado histórico, su belleza o brindar ocio, aventura y diversión. [30]

Los atractivos turísticos de la Ciudad de Tulcán que se seleccionaron debían cumplir los aspectos siguientes:

- Belleza arquitectónica
- Número de visitantes que recibe
- Conservación de sus instalaciones
- Representación de carácter cultural o patrimonial
- Importancia histórica local o nacional, entre otros.

Además estos sitios se clasifican en las categorías siguientes:

- Iglesias
- Parques y plazoletas
- Edificaciones patrimoniales y culturales
- Monumentos, estatuas y bustos

3.1.3.1 *Iglesias*

En la Ciudad de Tulcán existen algunas iglesias y capillas, pero se prefirieron tres, debido a su belleza arquitectónica y representación patrimonial. A continuación se mencionan dichos templos:

- Catedral de Tulcán
- San Francisco
- La Dolorosa

3.1.3.2 *Parques y plazoletas*

Para elegir los parques y plazoletas se tomó en cuenta su importancia histórica, número de visitantes que acuden a las instalaciones que ofrecen y su conservación. Las áreas que cumplieron con estos requerimientos, son las siguientes:

- Parque Central
- Parque Isidro Ayora
- Parque Julio Andrade
- Plazoleta Padre Carlos de La Vega
- Plazoleta La Concordia
- Parque Infantil Bolivia
- Parque Recreacional La Laguna
- Complejo Recreacional Los Pastos

3.1.3.3 *Edificaciones patrimoniales y culturales*

En la clasificación de las edificaciones se tomaron en cuenta aspectos como: belleza arquitectónica, conservación y su representación patrimonial o cultural. Entre estos lugares se tiene los siguientes:

- Cementerio Municipal José Azaél Franco

- Teatro Municipal Lemarie
- Casa de Aduana
- Mercado Plaza Central
- Casa de la Cultura Ecuatoriana Benjamín Carrión Núcleo del Carchi
- Museo Arqueológico, de Arte Moderno y Contemporáneo
- Bulevar de la Juventud
- Puente Internacional Rumichaca

3.1.3.4 *Monumentos, estatuas y bustos*

La selección de monumentos, estatuas y bustos se la hizo tomando en consideración su importancia histórica local o nacional, su representación cultural y su conservación. Las esculturas seleccionadas fueron:

- Monumento al Civismo Carchense
- Monumento a La Libertad
- Monumento al Guerrero Pasto
- Monumento al 26 de Mayo de 1971
- Monumento a Abdón Calderón
- Monumento al Minguero
- Monumento al Juego de las Bolas
- Estatua al Pupo
- Estatua al Padre Carlos de La Vega
- Estatua a San Francisco de Asís
- Busto a Julio Andrade
- Busto a Simón Bolívar
- Busto a Juan Montalvo

3.2 DISEÑO

3.2.1 Interfaz gráfica

El diseño de la estructura de las páginas que conforman el sitio web, no incluye ningún elemento gráfico, solamente líneas y bloques, de esta manera

se facilita la creación del contenido de cada una de las páginas, cumpliendo con los requerimientos anteriormente mencionados.

A continuación se presentan los modelos del diseño de la estructura de la página principal y páginas informativas:

- Página principal del sitio

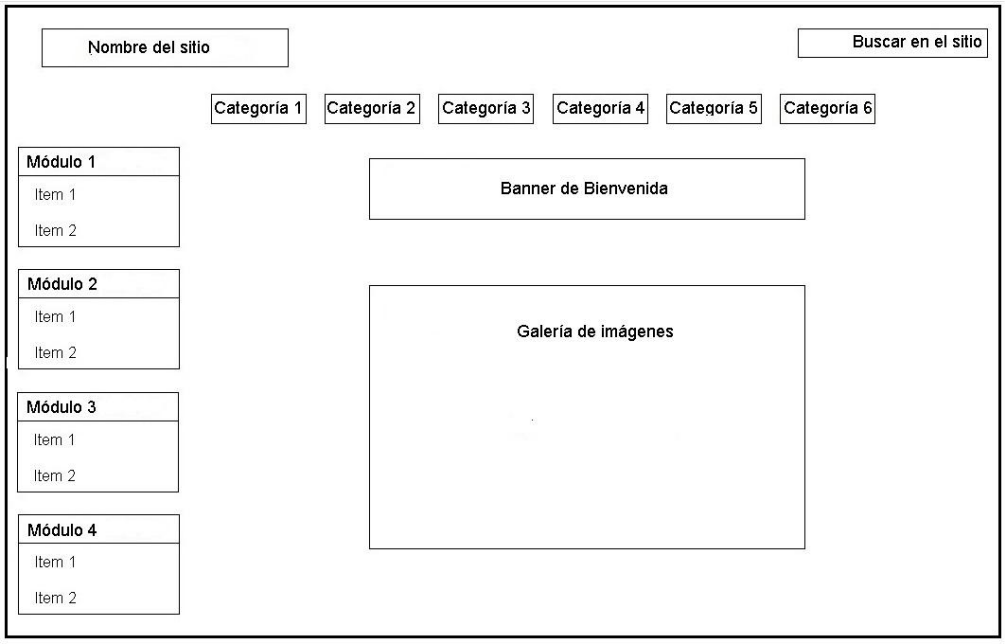


Figura 3.2.1 Estructura de la página principal

- Páginas informativas

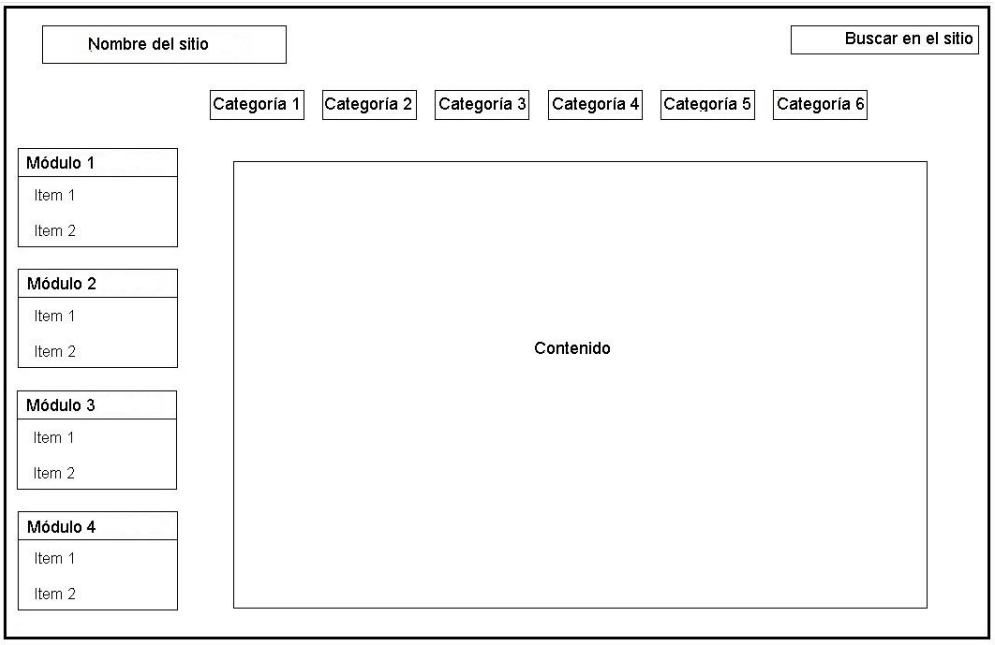


Figura 3.2.2 Estructura de las páginas informativas

3.3 IMPLEMENTACIÓN

3.3.1 Recopilación de información, fotografías e imágenes

Con la clasificación de los atractivos turísticos se procedió a buscar la información relacionada a cada uno de éstos, para lo cual se recurrió a la Biblioteca Municipal, Museo arqueológico y de arte, Unidad de turismo del GAD Municipal, Despacho Parroquial de La Dolorosa, revistas, folletos, diarios locales e Internet. Posteriormente se prosiguió a la selección y depuración de la información encontrada.

Finalmente se tomó las fotografías de cada uno de los lugares y la búsqueda de algunas imágenes para completar el contenido de las páginas del sitio web.

3.3.2 Creación del sitio web

Primeramente se debe abrir un navegador web e ingresar a la URL “https://sites.google.com”, apareciendo una ventana de autentificación tal como se muestra en la figura 3.3.1.

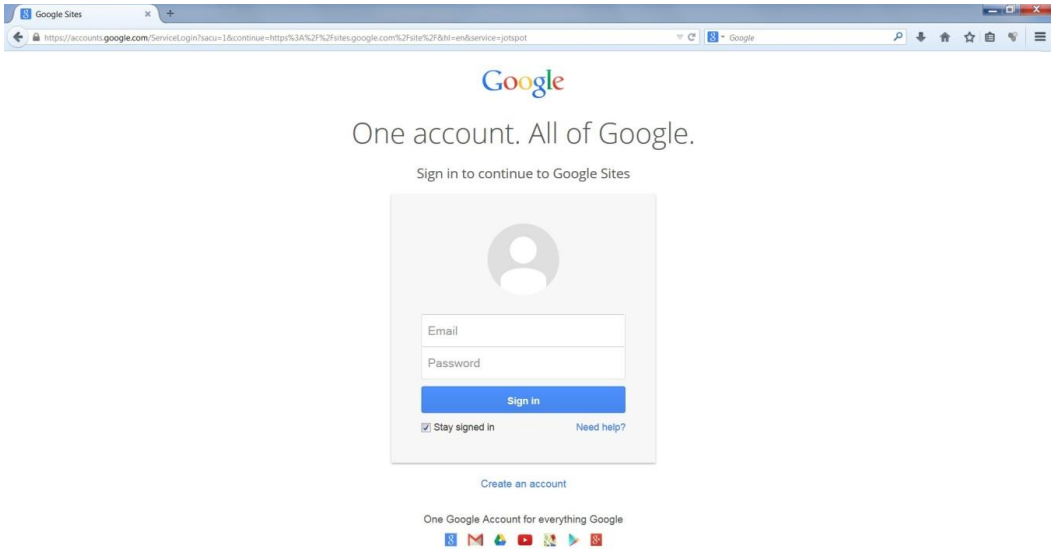


Figura 3.3.1 Ingreso a Google Sites

Para ingresar se debe tener una cuenta activa en Google, con esto, se digita el correo electrónico y la contraseña, accediendo a la interfaz gráfica de Google Sites, tal como se puede ver en la figura 3.3.2.

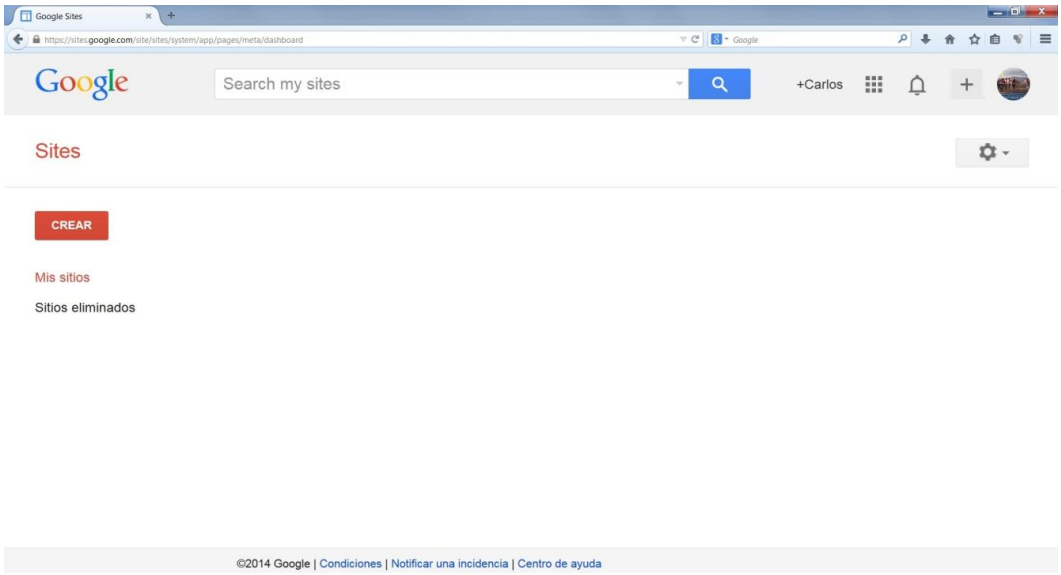


Figura 3.3.2 Interfaz gráfica de Google Sites

A continuación se pulsa el botón “Crear” y se presenta una ventana, tal como se muestra en la figura 3.3.3.

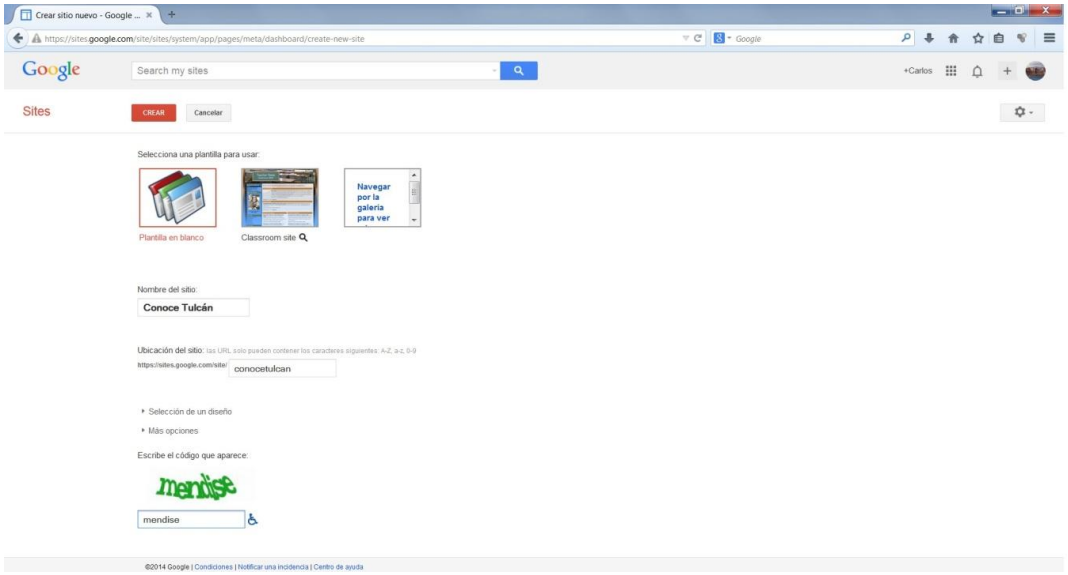


Figura 3.3.3 Creación del sitio web

En “Nombre del sitio”, se escribe como se va a llamar el sitio, en este caso “*Conoce Tulcán*”, el siguiente campo de la URL se completa automáticamente o se lo puede modificar, se selecciona un diseño de plantilla, se escribe un código que aparece y para terminar se pulsa el botón “Crear”, presentándose la plantilla seleccionada para el sitio, como se puede apreciar en la figura 3.3.4.

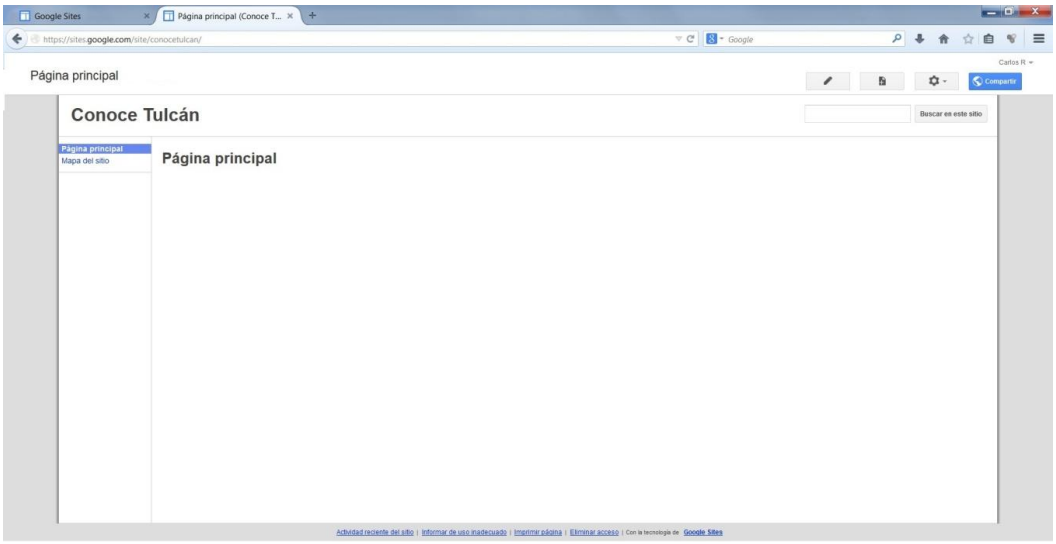


Figura 3.3.4 Plantilla del sitio web

Continuando con la creación del sitio, se debe tener en cuenta para qué sirve cada uno de los iconos presentes en la interfaz gráfica.



Al pulsar el icono “Editar página” se ingresa a una interfaz similar a un editor de texto, para facilitar la modificación de la página actual y guardar los cambios realizados, como se puede mirar en la figura 3.3.5.

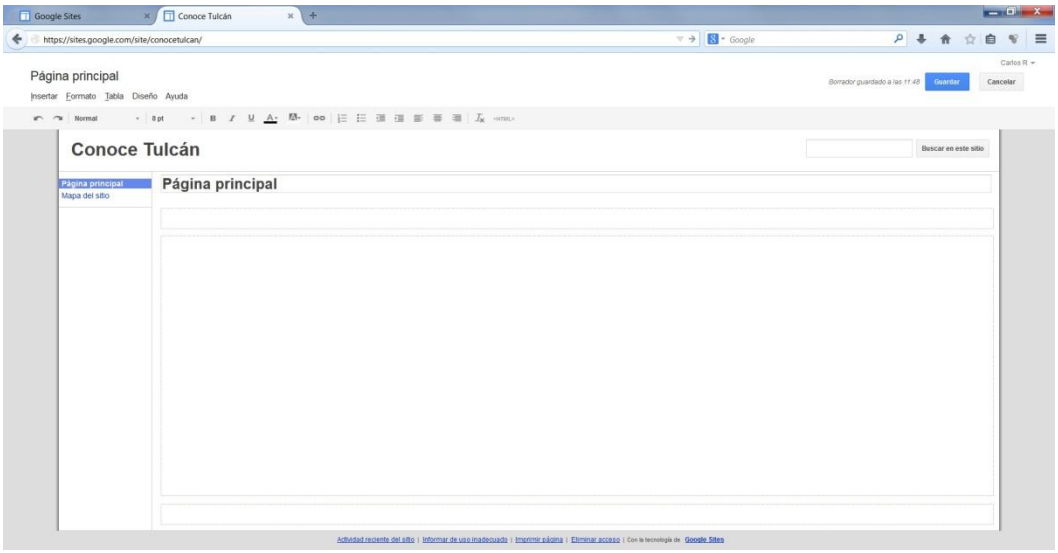


Figura 3.3.5 Interfaz de edición



Al hacer clic en este icono se crea una nueva página en el sitio web, desplegándose una ventana, como se observa en la figura 3.3.6. Inmediatamente se debe escribir el nombre de la página, seleccionar una plantilla y elegir la ubicación que va a tener en el sitio web.

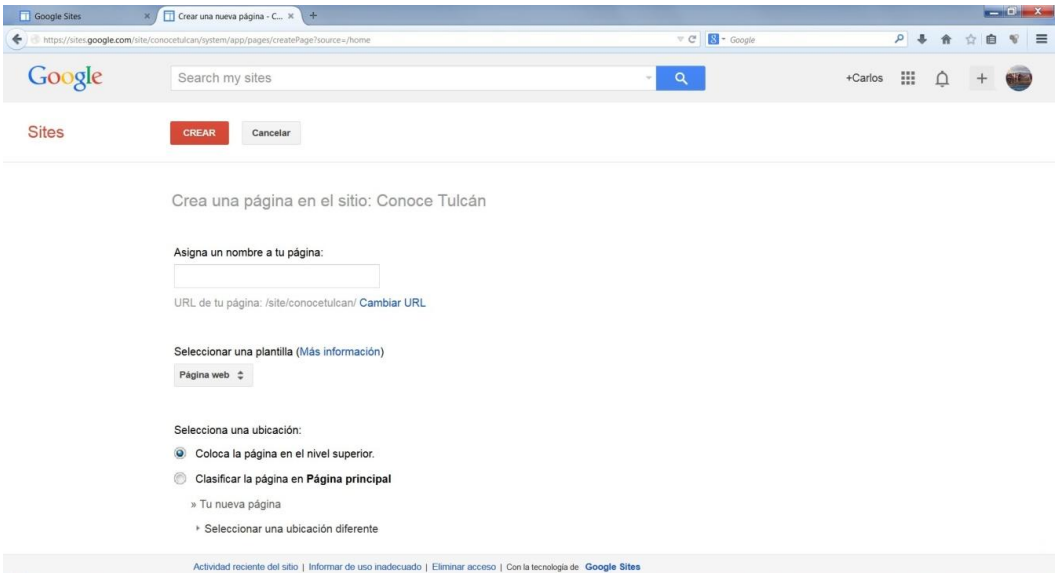


Figura 3.3.6 Creación de una nueva página



Al pulsar en este icono asoma un menú de acciones, como se ve en la figura 3.3.7, donde se puede escoger cualquier ítem que permite realizar acciones en la página actual y sobre el sitio web en general.

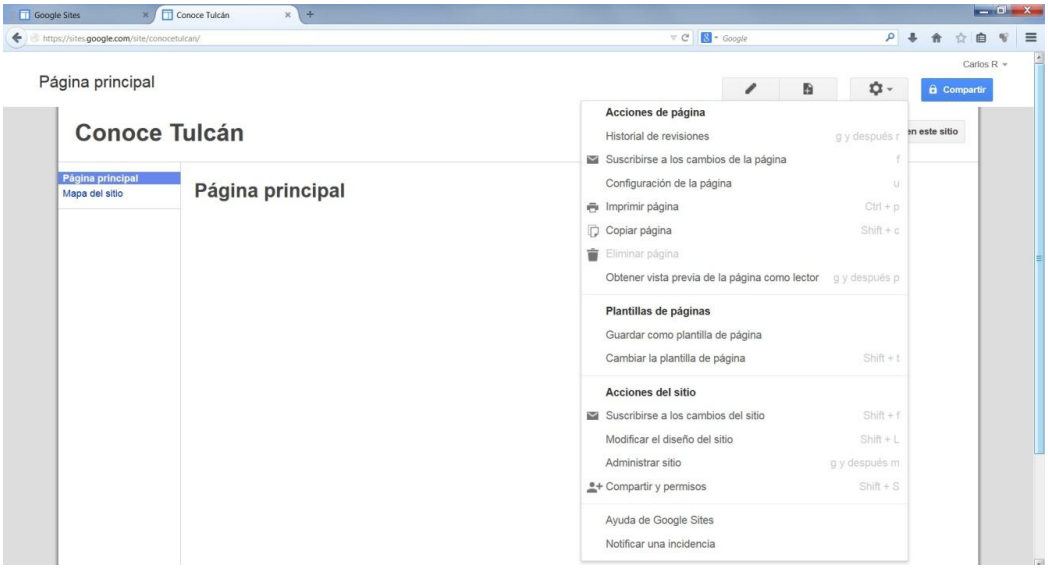
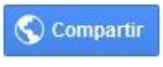


Figura 3.3.7 Menú de acciones



Al hacer clic en este icono se ingresa a una ventana, como se muestra en la figura 3.3.8, en la cual se puede cambiar quién puede tener acceso al sitio web:

- Público en la web, cualquier usuario de Internet pude encontrar el sitio y acceder a él sin necesidad de iniciar sesión.

- Cualquier usuario que tenga el enlace, el usuario que conozca el enlace puede acceder al sitio sin necesidad de iniciar sesión.
- Ciertos usuarios, éstos necesitan iniciar una sesión de acceso, para ingresar al sitio.

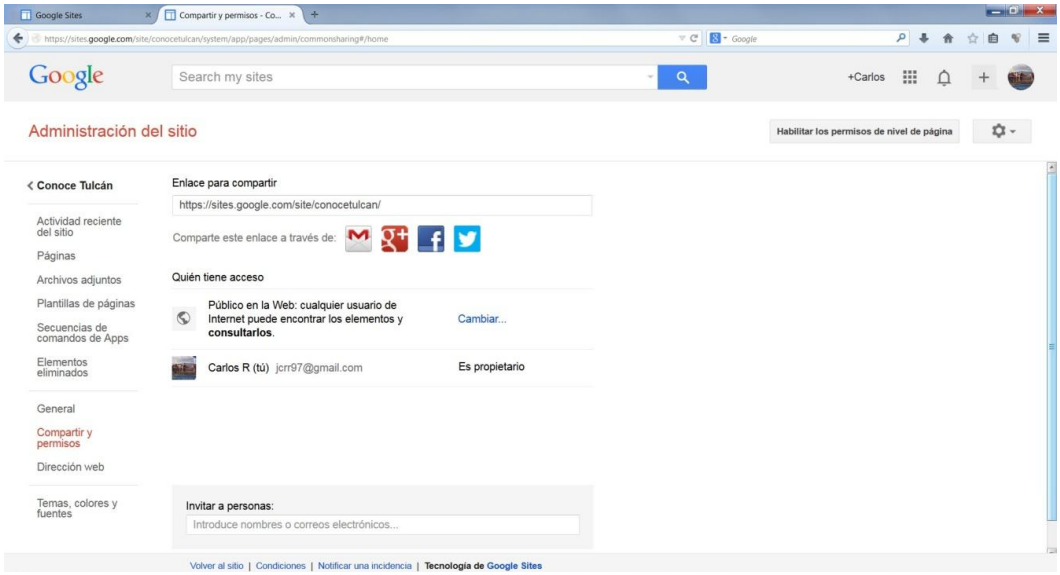


Figura 3.3.8 Compartición del acceso al sitio web

Finalmente se modificó la plantilla para adaptarla a la estructura diseñada anteriormente, se crearon las páginas necesarias con el contenido adecuado para el sitio web. Los colores utilizados fueron el celeste (C0E3F4), el blanco (FFFFFF), el azul (0000FF), el azul (5671FA) y el negro (000000). El tipo de letra empleado en toda la plantilla es Verdana. El tamaño de la letra del título del sitio web es de 24 puntos, los títulos de las páginas informativas van de 12, ambos casos en color azul (0000FF) y la redacción de las páginas informativas con otros textos de la plantilla tienen un tamaño de 10 en color negro y azul (0000FF), respectivamente.

Por último para completar el diseño del sitio, se crearon el banner de bienvenida, utilizando el programa Sothink SWF Easy y la galería de imágenes empleando la aplicación UltraSlideshow Flash Creator.

Vista final de la página principal:

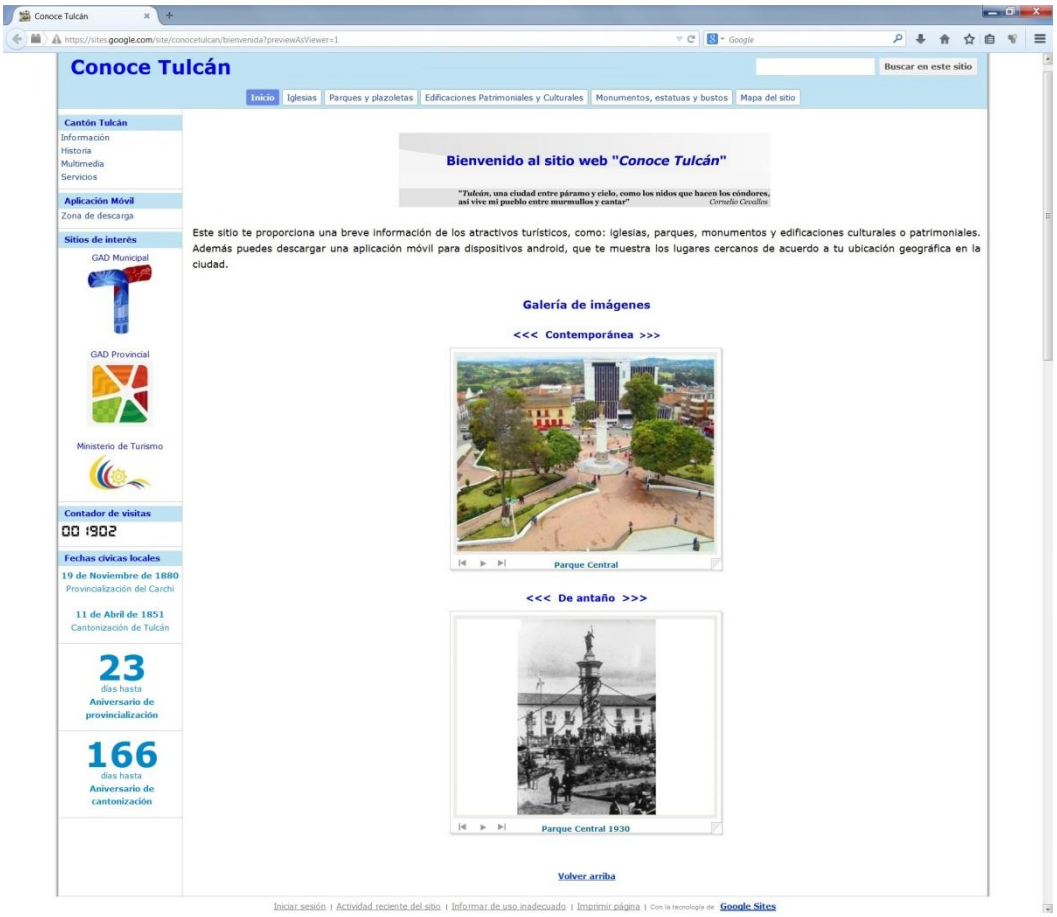


Figura 3.3.9 Página principal del sitio

Vista final de una página informativa:

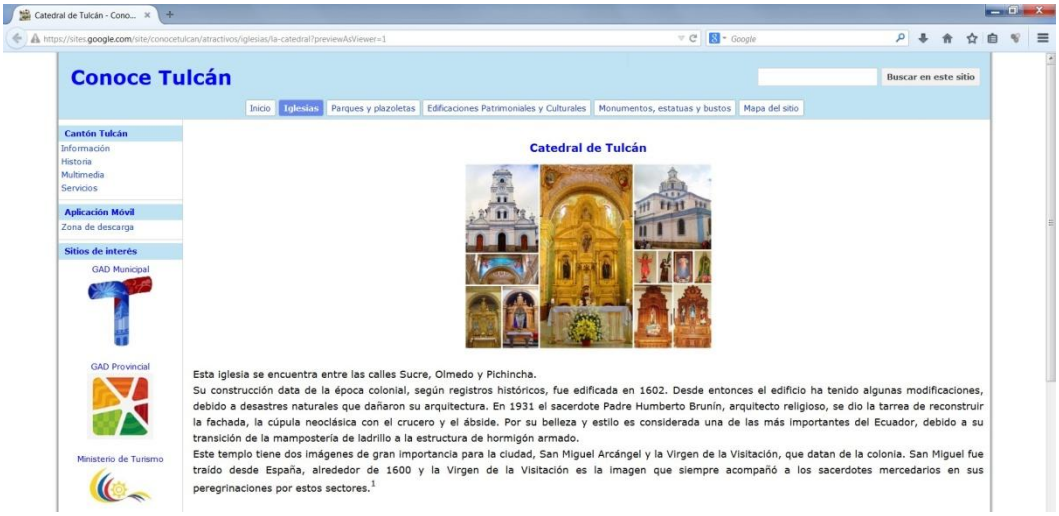


Figura 3.3.10 Página informativa para la Catedral de Tulcán

Mapa del sitio:

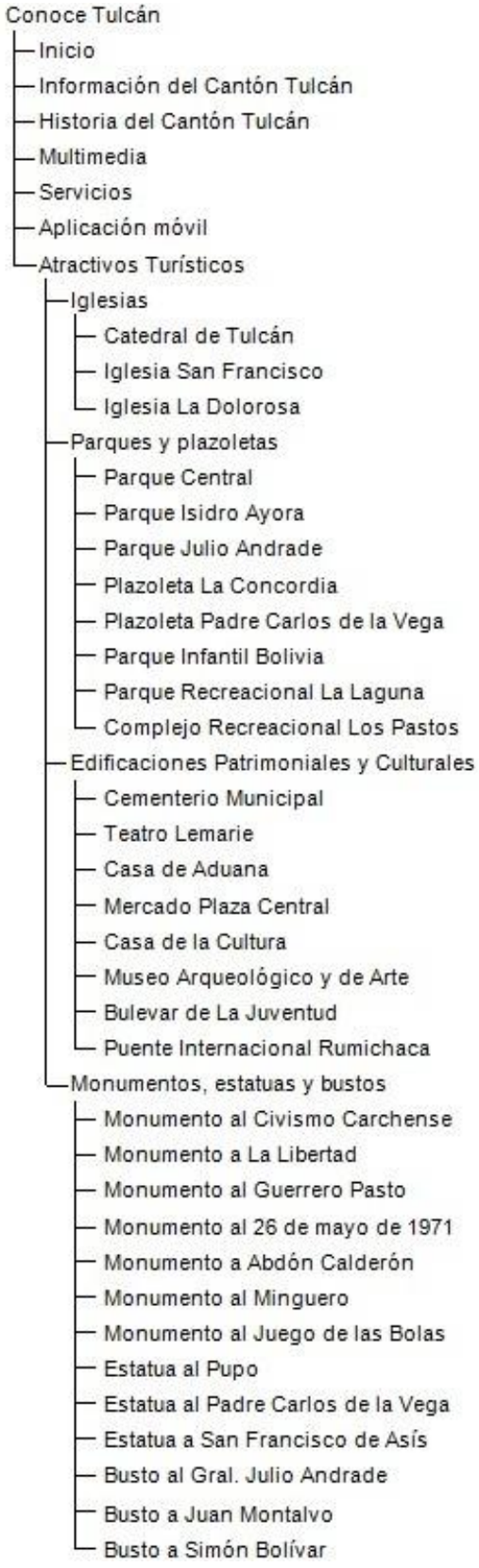


Figura 3.3.11 Mapa del sitio

CAPÍTULO 4

ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE LA APLICACIÓN MÓVIL

4.1 ANÁLISIS

4.1.1 Requerimientos para la creación de la app

4.1.1.1 *Requerimientos técnicos*

Software

Se va a utilizar los programas siguientes:

- IDE Eclipse con el complemento ADT y SDK Android.
- Genymotion.
- VirtualBox.
- Sitio web “www.flashearth.com”

Lenguaje de programación

El lenguaje que se utiliza para desarrollar aplicaciones móviles utilizando IDE Eclipse es Java, adecuado para Android.

Hardware

- Computador de escritorio o laptop con procesador de 2GHz, memoria de 2GB y 10GB de espacio en el disco duro, como requisitos mínimos y conexión a Internet.
- Receptor GPS.
- Smartphone con cámara, GPS y conexión a Internet.

4.1.1.2 *Requerimientos funcionales*

Interfaz gráfica

- Las ventanas que se muestren al usuario van a ser comprensibles y amigables, indicando al usuario lo que debe hacer para proceder a usar la app, en caso de omitir las indicaciones, la aplicación no le permite continuar. No tendrá ventanas deslizantes. Los textos en la interfaz gráfica van a tratar al usuario de tú, de esta manera se hace la aplicación más agradable.

- La sobreimpresión del nombre del atractivo turístico en la imagen resultante mostrada en la pantalla, debe ser en la parte superior izquierda.
- La breve información de los atractivos turísticos se va a observar en una interfaz gráfica, adecuada para dispositivos móviles, del sitio web creado para esta finalidad.

Funcionalidades

- La app tiene que proveer al usuario una breve información sobre los distintos atractivos turísticos que estén cercanos a él. Dicha información debe mostrarse en una interfaz adecuada del sitio web creado para tal propósito.
- La aplicación tiene que usar la cámara de fotos del dispositivo móvil. También va a almacenar las fotografías tomadas, emplear las señales GPS y la conexión a Internet, ya sea mediante redes de telefonía móvil o redes inalámbricas Wi-Fi.
- La app va a desplegar mensajes en la pantalla, en caso de que el GPS se encuentre desactivado y no se tenga conexión a Internet.
- La aplicación debe ser capaz de geolocalizar al dispositivo móvil y de acuerdo a sus coordenadas geográficas, se debe mostrar los atractivos turísticos cercanos al usuario.

4.1.1.3. *Requerimientos no funcionales*

Usabilidad

- La aplicación móvil va a tener una interfaz gráfica intuitiva, para que el usuario sepa lo que debe hacer. Para ello se debe tomar en cuenta los puntos siguientes:
  - El idioma tiene que ser español.
  - Las imágenes y colores utilizados no va a afectar la presentación del contenido de la app.
  - El tipo de letra debe ser legible de tamaño adecuado, esto puede variar según el tamaño de la pantalla del dispositivo móvil.

- La orientación del teléfono siempre va a ser vertical en todas las ventanas, excepto cuando se usa la cámara de fotos.

Disponibilidad

- La app puede ejecutar todas sus funcionalidades en un dispositivo móvil, siempre que se den las siguientes condiciones:
  - Habilitar la conexión a Internet (Wi-Fi o móvil)
  - Disponer de servicio GPS y esté activado
  - Poseer cámara de fotos
  - Tener energía en la batería
- El hardware que la aplicación ocupe del dispositivo móvil no debe generar conflictos con otras apps instaladas en éste.
- La app en funcionamiento no va a colapsar en ningún momento de su utilización.

Rendimiento

- Debe ser capaz de ejecutarse en un dispositivo móvil con sistema operativo Android Ice Cream Sandwich API 14, como mínimo. Antes de instalarla se debe solicitar al usuario el permiso para que la app pueda conectarse a Internet, utilizar el GPS, controlar la cámara y modificar o eliminar contenido de la tarjeta SD del terminal.
- La cantidad de información que va a manejar la app será mínima, y está relacionada al tamaño de las fotografías que se tomen de los atractivos turísticos, las cuales se guardarán en una carpeta en la memoria interna del dispositivo móvil.

4.2 DISEÑO

La aplicación móvil “Conoce Tulcán”, primordialmente debe contener los datos de los atractivos turísticos, como: nombre, latitud, longitud y URL. Con las coordenadas geográficas de los lugares, se establece un rango que abarque un

área en la que se encuentre el usuario cerca a los atractivos turísticos. Para lo cual se debe tener en cuenta el dato siguiente del planeta:

- Radio ecuatorial = 6378,137 km

Para calcular la distancia que representa un grado, se utiliza la fórmula para encontrar la distancia. [31]

Donde d: Distancia, r: Radio de la Tierra, β_A y β_B: Latitudes, θ_A y θ_B: Longitudes:

$$d = \frac{2\pi\, r}{360°} \,.\arccos\{\sin\beta_A.\sin\beta_B + \cos\beta_A.\cos\beta_B.\cos(\theta_A - \theta_B)\}$$

Ecuación 4.2.1 Fórmula de la distancia entre dos paralelos

Para realizar el cálculo de la distancia entre dos paralelos separados por un grado, se tomará las coordenadas siguientes:

Lugar A: Latitud = 1° y longitud = θ

Lugar B: Latitud = 0° y longitud = θ

$$d = \frac{2\pi\ (6378{,}137\ \text{km})}{360°} \,.\arccos\{\sin 1°.\sin 0° + \cos 1°.\cos 0°.\cos(\theta - \theta)\}$$

$$d = \frac{2\pi\ (6378{,}137\ \text{km})}{360°} \,.\arccos\{0 + \cos 1°\}$$

$$d = \frac{2\pi\ (6378{,}137\ \text{km})}{360°} \,.\arccos\{0.9998476952\}$$

$$d = \ 111{,}31949\text{km}$$

Con esta fórmula se puede encontrar la distancia entre dos lugares cualesquiera sobre la superficie terrestre. Con esto se tiene que 1° de latitud equivale a una distancia aproximadamente de 111,319 km. De forma análoga 0,001° vale alrededor de 111,319 m.

Por consiguiente, el área alrededor donde se encuentre el usuario debe ser aproximadamente de 100 m^2, dicha área puede disminuir o aumentar dependiendo del sector de la ciudad. Por lo tanto se debe utilizar cuatro decimales en los valores de las coordenadas geográficas.

Solamente se va a presentar máximo tres atractivos turísticos en la pantalla del equipo, para que el usuario seleccione uno de ellos, dependiendo de su ubicación en la ciudad. Se impuso esta condición debido a la cercanía de los lugares y

sobre todo para que el usuario se encuentre contemplando el sitio elegido a una distancia adecuada para poder fotografiarlo.

A continuación en la figura 4.2.1 se muestra el área en la que se puede encontrar un usuario.

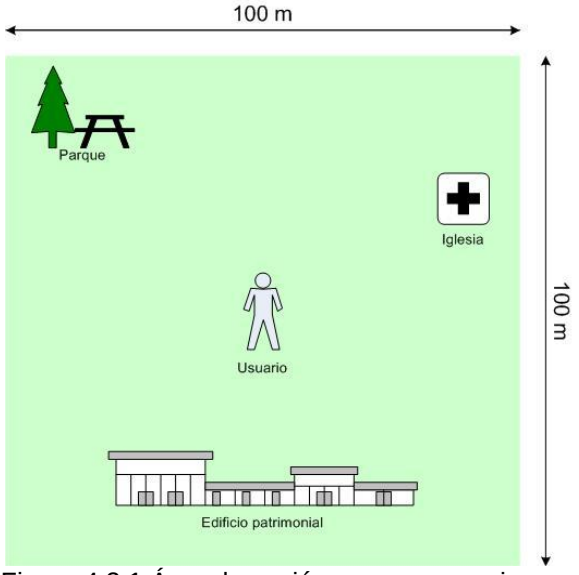


Figura 4.2.1 Área de acción para un usuario

Para desarrollar la aplicación móvil, ésta debe seguir la siguiente secuencia de ejecución:

- Presentar en la pantalla inicial las indicaciones generales antes de comenzar la app.
- Activar el GPS y la conexión a Internet.
- Buscar la ubicación geográfica del usuario y mostrar la visualización de la cámara.
- Indicar al usuario si existen lugares cercanos.
- Listar máximo tres atractivos turísticos cercanos.
- Seleccionar un atractivo turístico.
- Tomar una foto del atractivo turístico seleccionado.
- Guardar la foto.
- Mostrar en la pantalla la imagen del atractivo turístico, con su nombre sobreimpreso en la parte superior.
- Pulsar un botón para obtener una breve información del atractivo turístico.
- Tocar el botón salir para finalizar la aplicación.

El usuario puede oprimir la tecla “back” del dispositivo móvil para regresar a la anterior ventana. Al presionar la tecla “menú” el usuario va a encontrar la opción “Acerca de” que despliega una corta información de la aplicación.

A continuación, en la figura 4.2.2 se detalla el diagrama de flujo de la aplicación.

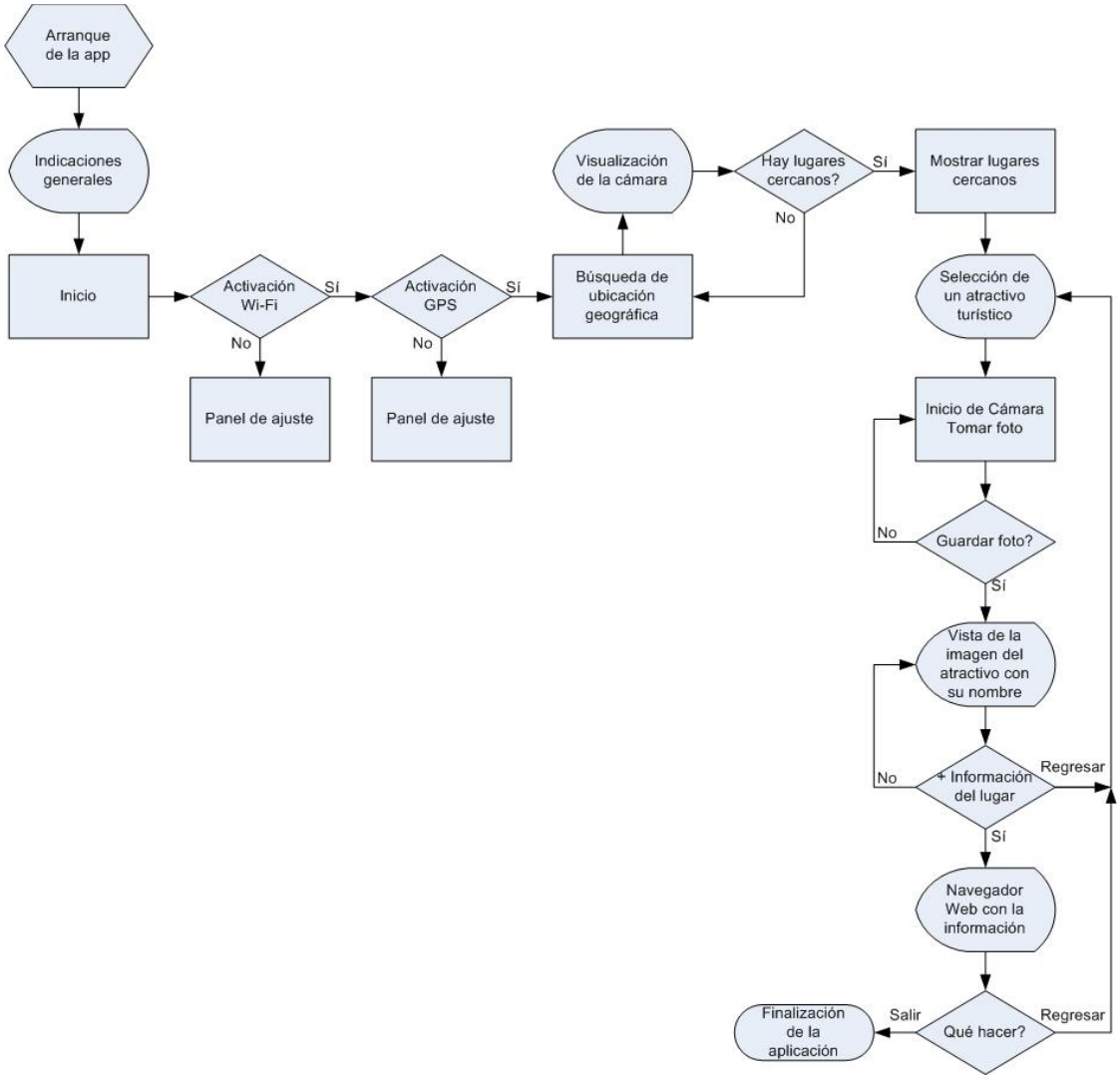


Figura 4.2.2 Diagrama de flujo de la aplicación Conoce Tulcán

4.2.1 Arquitectura de la aplicación

La arquitectura general de la app a desarrollar, va a utilizar el hardware y aplicaciones nativas de Android que se muestran en la figura 4.2.3.

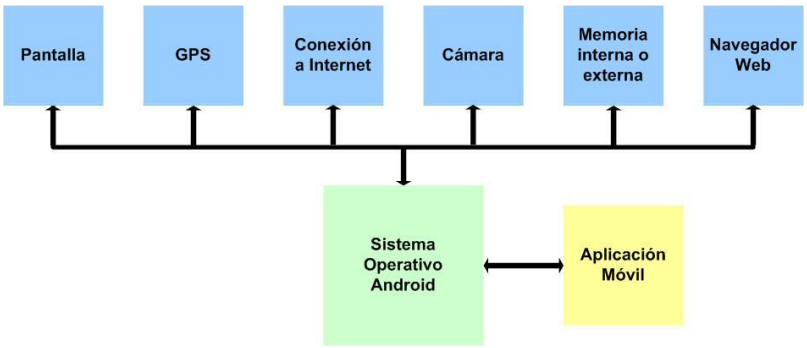


Figura 4.2.3 Arquitectura de la app

Como se puede apreciar en el diagrama, el hardware que posee el dispositivo móvil y las aplicaciones nativas propias de Android, van a ser gestionados por clases en el programa de la aplicación, dichas clases se detallan enseguida en el diagrama de la figura 4.2.4.

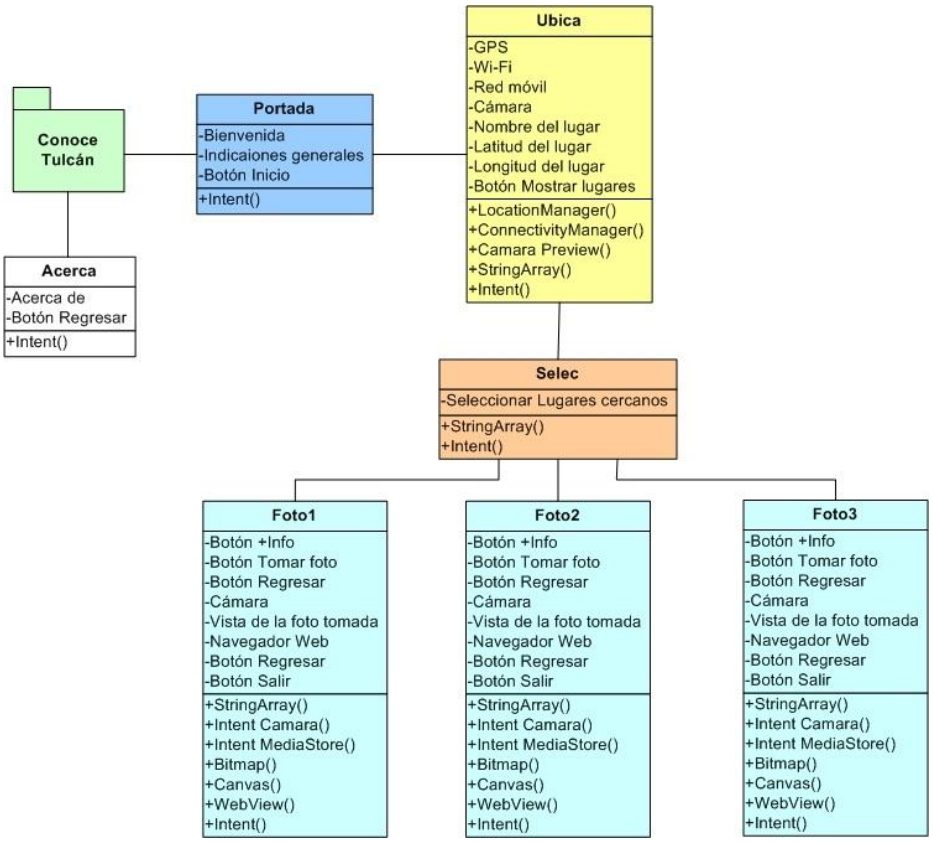


Figura 4.2.4 Diagrama de clases

Las clases o actividades que se van a emplear en el desarrollo de la app son siete. Cada una de éstas corresponde a una interfaz gráfica de usuario. Esto se debe a la arquitectura de Android, fundamentada en crear clases por cada ventana en la que el usuario puede interactuar. Si se sigue la secuencia desde el inicio, se ve que cada actividad crea a la siguiente, hasta

llegar al final de la aplicación, excepto la clase Acerca. En todas las clases se debe utilizar un *Intent* para pasar de una actividad a otra.

4.2.1.1 *Clase Portada*

En esta actividad se debe dar la bienvenida al usuario, presentar las indicaciones generales para el debido funcionamiento de la aplicación y un botón de “Inicio” para continuar a la siguiente clase.

4.2.1.2 *Clase Ubica*

Es la actividad principal de la aplicación, tiene que contener los datos de los atractivos turísticos, como: nombre, latitud, longitud y URL. Debe mostrar lo siguiente: mensajes al usuario, ubicación geográfica del usuario, visualización de la cámara y un botón “Mostar lugares cercanos”.
Se encarga de buscar la ubicación geográfica del usuario, compararla con la latitud y longitud de los lugares, almacenar el nombre y la URL, de máximo tres lugares cercanos, en un arreglo de cadena de caracteres para enviarlos a las siguientes actividades. Al presionar el botón mencionado anteriormente, se debe avanzar a la siguiente clase.

4.2.1.3 *Clase Selec*

Esta actividad recibe un arreglo de cadena de caracteres con el nombre de máximo tres atractivos turísticos, cercanos a la ubicación geográfica del usuario y los despliega en una lista para ser mostrados en la pantalla. El usuario debe seleccionar un lugar para seguir a la próxima actividad.

4.2.1.4 *Clase Foto1*

Esta clase recibe un arreglo de cadena de caracteres con el nombre y URL del primer atractivo turístico de la lista de la actividad anterior para utilizarlos en el proceso. Se presenta una pantalla con un espacio para mostrar una imagen y tres botones en la parte inferior: “+ Info”, “Tomar foto” y “Regresar”. Al pulsar el botón “Tomar foto”, se activa la cámara del dispositivo móvil para que el usuario tome una foto del lugar seleccionado en la actividad anterior. Se crea una carpeta con el nombre “Conoce

Tulcán” en la memoria del equipo para guardar dicha foto. Enseguida en la pantalla debe aparecer la imagen del atractivo turístico con el nombre sobreimpreso en la parte superior izquierda.
Al presionar el botón “Regresar” se vuelve a la clase Selec.
El botón “+ Info” solamente se puede oprimir, una vez que aparezca la imagen del lugar elegido, en este caso se muestra una pantalla con un navegador web y dos botones “Regresar” y “Salir”. En el navegador se debe cargar la URL del lugar en cuestión, desplegándose una breve información. Al tocar el botón “Regresar”, se vuelve a la clase Selec y al teclear el botón “Salir”, se abandona la aplicación.

4.2.1.5 *Clase Foto2*

Esta clase recibe un arreglo de cadena de caracteres con el nombre y URL del segundo atractivo turístico de la lista de la actividad Selec para utilizarlos en el proceso. Se presenta una pantalla con un espacio para mostrar una imagen y tres botones en la parte inferior: “+ Info”, “Tomar foto” y “Regresar”. Al pulsar el botón “Tomar foto”, se activa la cámara del dispositivo móvil para que el usuario tome una foto del lugar seleccionado en la actividad anterior. Se crea una carpeta con el nombre “Conoce Tulcán” en la memoria del equipo para guardar dicha foto. Enseguida en la pantalla debe aparecer la imagen del atractivo turístico con el nombre sobreimpreso en la parte superior izquierda.
Al presionar el botón “Regresar” se vuelve a la clase Selec.
El botón “+ Info” solamente se puede oprimir, una vez que aparezca la imagen del lugar elegido, en este caso se muestra una pantalla con un navegador web y dos botones “Regresar” y “Salir”. En el navegador se debe cargar la URL del lugar en cuestión, desplegándose una breve información. Al tocar el botón “Regresar”, se vuelve a la clase Selec y al teclear el botón “Salir”, se abandona la aplicación.

4.2.1.6 *Clase Foto3*

Esta clase recibe un arreglo de cadena de caracteres con el nombre y URL del tercer atractivo turístico de la lista de la actividad Selec para utilizarlos en el proceso. Se presenta una pantalla con un espacio para

mostrar una imagen y tres botones en la parte inferior: “+ Info”, “Tomar foto” y “Regresar”. Al pulsar el botón “Tomar foto”, se activa la cámara del dispositivo móvil para que el usuario tome una foto del lugar seleccionado en la actividad anterior. Se crea una carpeta con el nombre “Conoce Tulcán” en la memoria del equipo para guardar dicha foto. Enseguida en la pantalla debe aparecer la imagen del atractivo turístico con el nombre sobreimpreso en la parte superior izquierda.

Al presionar el botón “Regresar” se vuelve a la clase Selec.

El botón “+ Info” solamente se puede oprimir, una vez que aparezca la imagen del lugar elegido, en este caso se muestra una pantalla con un navegador web y dos botones “Regresar” y “Salir”. En el navegador se debe cargar la URL del lugar en cuestión, desplegándose una breve información. Al tocar el botón “Regresar”, se vuelve a la clase Selec y al teclear el botón “Salir”, se abandona la aplicación.

4.2.1.7 *Clase Acerca*

Esta actividad expone en la pantalla una corta información sobre la aplicación y debe aparecer cada vez que se seleccione la opción “Acerca de”, al pulsar la tecla menú del dispositivo móvil.

4.3 IMPLEMENTACIÓN

4.3.1 Obtención de coordenadas de los atractivos turísticos

Para la adquisición de las coordenadas geográficas, se utilizó un receptor GPS marca Garmin Etrex Legend. Se acudió a cada uno de los atractivos turísticos y se hizo la lectura de dichas coordenadas. Se las comparó con los valores obtenidos del sitio web “http://www.flashearth.com”, teniendo una variación de alrededor de 3 milésimas de grado.

Atractivo Turístico	Latitud (Norte)	Longitud (Oeste)
Monumento al Civismo Carchense	0,7758°	-77,7411°
Monumento al Pupo	0,7794°	-77,7432°
Monumento al Guerrero Pasto	0,7941°	-77,7397°

Monumento al Minguero	0,7977°	-77,7348°
Bulevar de la Juventud	0,8020°	-77,7309°
Parque Recreacional La Laguna	0,8022°	-77,7286°
Monumento al 26 de Mayo de 1971	0,8033°	-77,7291°
Monumento al Juego de las Bolas	0,8044°	-77,7248°
Parque Julio Andrade	0,8048°	-77,7269°
Busto a Julio Andrade	0,8049°	-77,7268°
Estatua al Padre Carlos de la Vega	0,8116°	-77,7214°
Plazoleta Padre Carlos de la Vega	0,8118°	-77,7210°
Catedral de Tulcán	0,8115°	-77,7182°
Plazoleta la Concordia	0,8118°	-77,7183°
Busto a Simón Bolívar	0,8118°	-77,7184°
Busto a Juan Montalvo	0,8120°	-77,7118°
Monumento a la Libertad	0,8120°	-77,7172°
Parque Central	0,8121°	-77,7173°
Teatro Municipal Lemarie	0,8124°	-77,7180°
Mercado Plaza Central	0,8142°	-77,7161°
Estatua a San Francisco de Asís	0,8150°	-77,7153°
Iglesia San Francisco	0,8151°	-77,7154°
Iglesia La Dolorosa	0,8161°	-77,7136°
Monumento a Abdón Calderón	0,8162°	-77,7141°
Parque Isidro Ayora	0,8167°	-77,7141°
Museo Arqueológico y de Arte	0,8173°	-77,7163°
Casa de la Cultura Ecuatoriana	0,8174°	-77,7160°
Parque Infantil Bolivia	0,8171°	-77,7122°
Cementerio Municipal	0,8194°	-77,7144°
Casa de Aduana	0,8152°	-77,6651°
Puente Internacional Rumichaca	0,8140°	-77,6640°
Complejo Recreacional Los Pastos	0,8379°	-77,7023°

Tabla 4.3.1 Coordenadas geográficas de los atractivos turísticos

4.3.2 Instalación del entorno de desarrollo Eclipse con SDK Android

Para obtener las herramientas necesarias para el desarrollo de la aplicación “Conoce Tulcán”, se debe descargarlas de la siguiente dirección web: “https://developer.android.com/sdk/index.html”, tal como se puede observar en la figura 4.3.1.

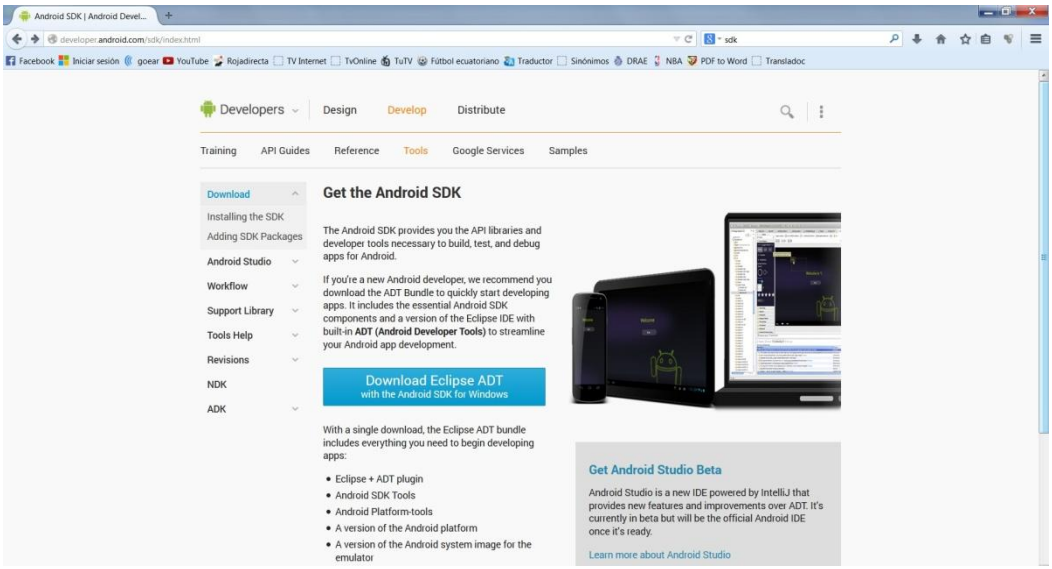


Figura 4.3.1 Sitio de descarga de las herramientas de desarrollo Android

Se obtiene un archivo zipiado, se debe descomprimir su contenido en una nueva carpeta en explorador de Windows, en la ubicación y con el nombre que se desee, no necesita instalarse nada, como se ve en la figura 4.3.2.

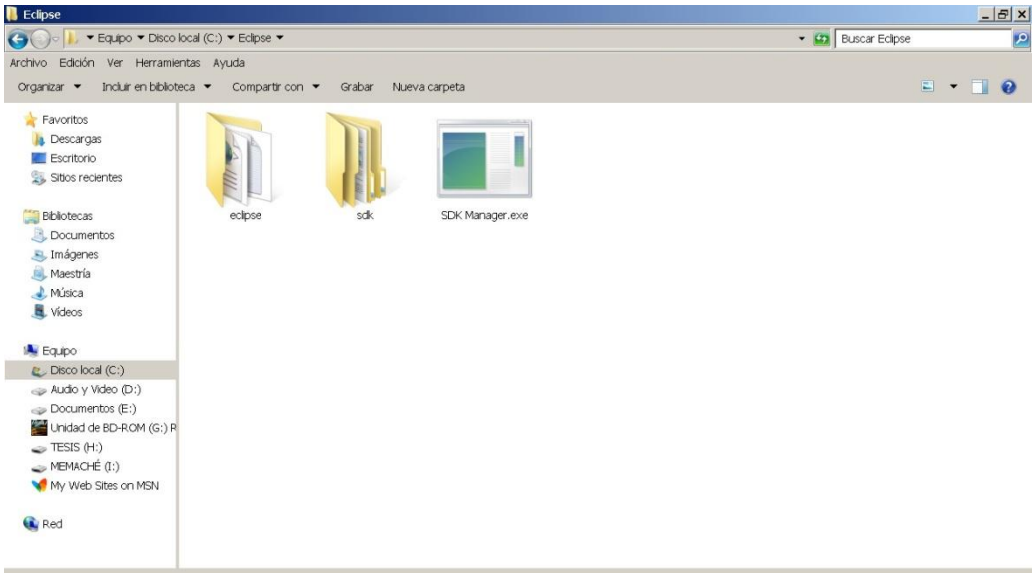


Figura 4.3.2 Carpetas de las herramientas de desarrollo

A continuación se entra a la carpeta “eclipse” y se da doble clic en el archivo ejecutable “eclipse”, apareciendo una ventana que pide crear la carpeta “workspace”, de preferencia en la misma carpeta que se creó anteriormente y se selecciona la casilla inferior para que no aparezca cada vez que se ejecuta este programa, de la manera que se indica en la figura 4.3.3. Dicha carpeta contendrá los proyectos que se desarrollen.

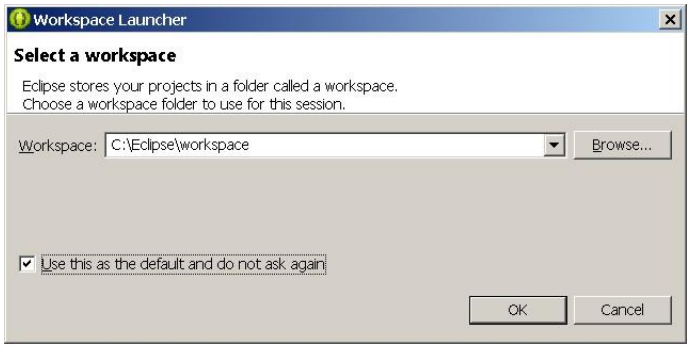


Figura 4.3.3 Creación de la carpeta “workspace”

En la ventana anterior se da clic en “OK”, asomando una ventana de bienvenida y con una pregunta “Enviar estadísticas de uso a Google?”, como se observa en la figura 4.3.4.

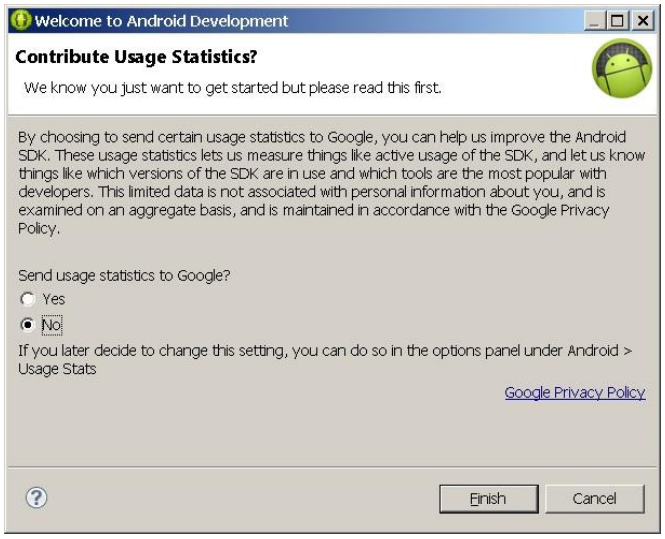


Figura 4.3.4 Ventana de bienvenida al desarrollo Android

Dando clic en “Finish” de la ventana anterior, se entra a la interfaz gráfica del programa Eclipse, como se mira en la figura 4.3.5.

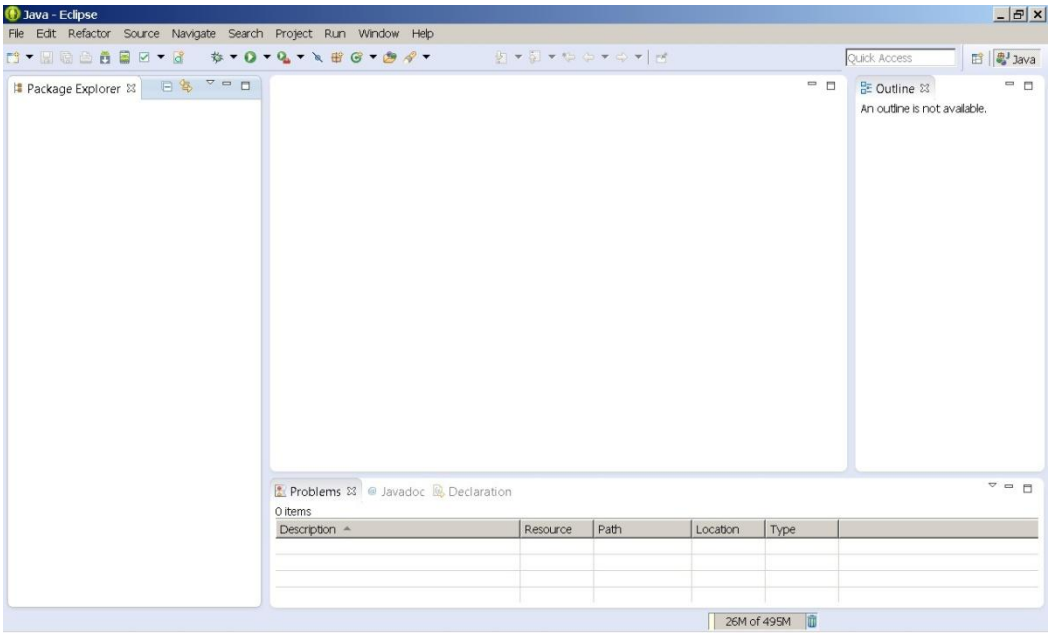


Figura 4.3.5 Interfaz gráfica de Eclipse

4.3.3 Instalación del emulador Genymotion

Dicho emulador sirve para probar la app virtualmente, se lo tiene que descargar de la siguiente dirección web, previa la creación de un registro de usuario: “https://cloud.genymotion.com/page/launchpad/download”, como se puede ver en la figura 4.3.6.

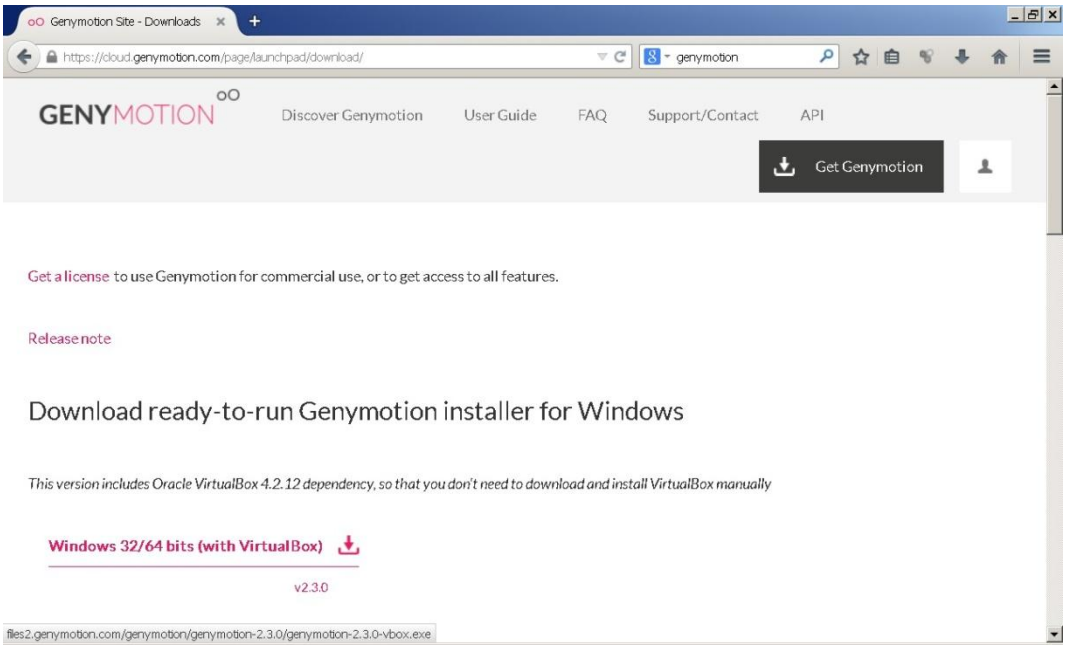


Figura 4.3.6 Página de descarga de Genymotion

Enseguida se da clic en “Windows 32/64 bits (with VirtualBox)”, indicando que se debe guardar el archivo ejecutable de instalación.

Una vez instalado el programa, se debe obtener un plugin en el IDE Eclipse. Para que Genymotion sea asociado y se cree un icono en la barra de herramientas, se da clic en “Help”, y luego en “Install New Software”, apareciendo la ventana que se ve en la figura 4.3.7.

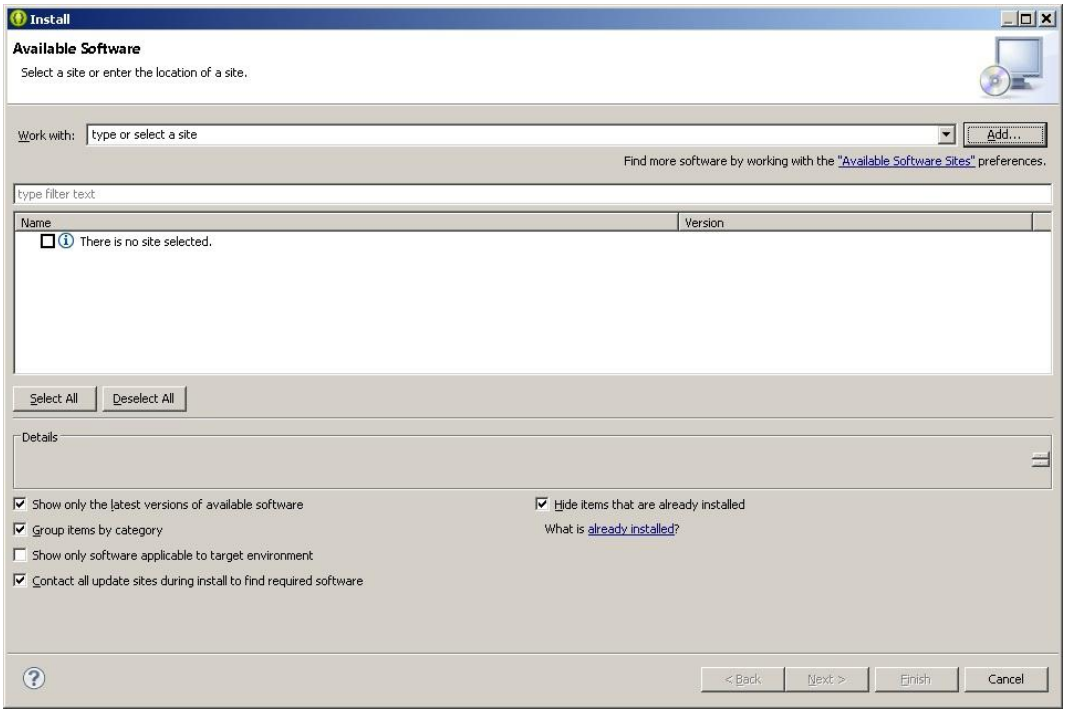


Figura 4.3.7 Ventana para instalar nuevo software

Para proseguir se pulsa el botón “Add”, desplegándose una ventana como se muestra en la figura 4.3.8. En “Name” se escribe “Genymotion” y en “Location” se pone “http://plugins.genymotion.com/eclipse”.

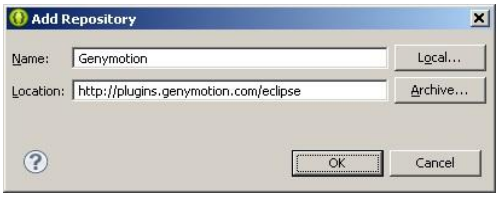


Figura 4.3.8 Ventana para añadir el plugin de Genymotion

Se pulsa el botón “OK”, apareciendo el repositorio de Genymotion en la ventana de la figura 4.3.7 y para terminar, se presiona “Finish”. Creándose de esta manera el icono del emulador, como se observa en la figura 4.3.9.

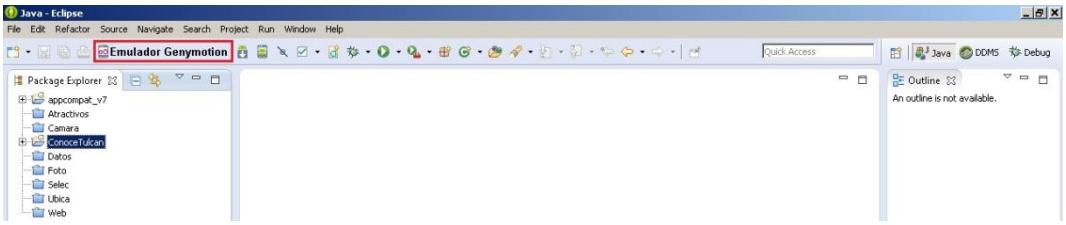


Figura 4.3.9 Ubicación del icono del emulador Genymotion

A continuación se debe abrir el emulador, dando clic en “Inicio”, luego en “Todos los programas”, enseguida en la carpeta “Genymotion” y finalmente en “Genymotion”, asomando la ventana que se mira en la figura 4.3.10.

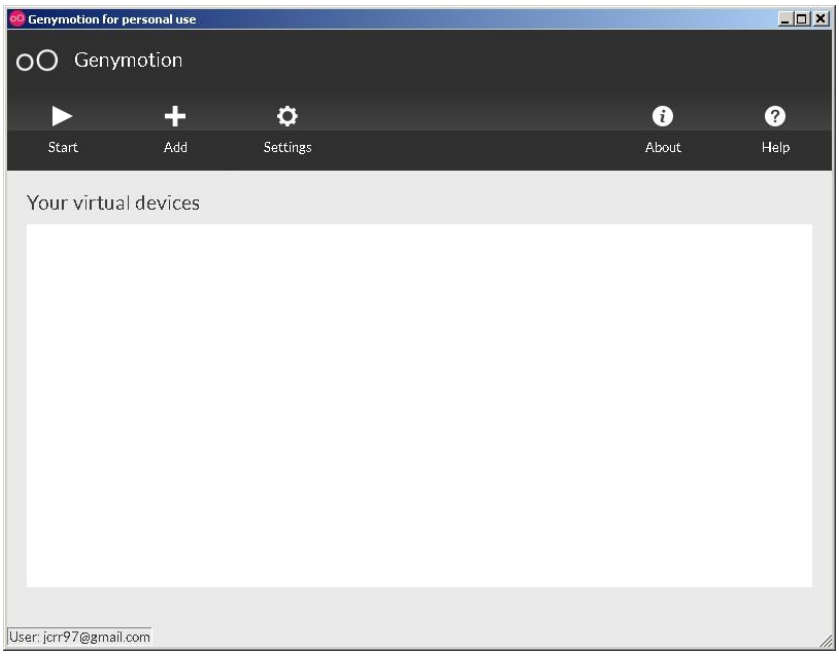


Figura 4.3.10 Emulador Genymotion

Continuando, se pulsa sobre “Settings”, apareciendo la ventana que se indica en la figura 4.3.11. Se elige la pestaña “ADB” y se debe especificar en “Android SDK” la ubicación de la carpeta “sdk”; sí el path es correcto, aparece el mensaje “Android SDK tools found successfully” y se oprime “OK”.

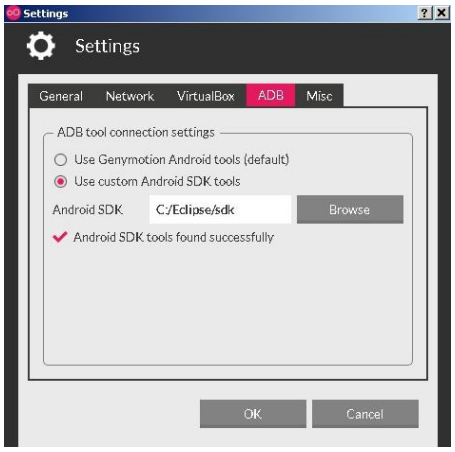


Figura 4.3.11 Configuración de emulador

Una vez que se sale de la configuración, se puede añadir un nuevo dispositivo virtual, haciendo clic en “Add” o también se lo puede hacer desde el IDE Eclipse, haciendo clic sobre el icono del emulador y luego presionando “New”. En ambos casos se despliega la ventana que se muestra en la figura 4.3.12. Se debe elegir por versión (Android versión) o por modelo (Device model).

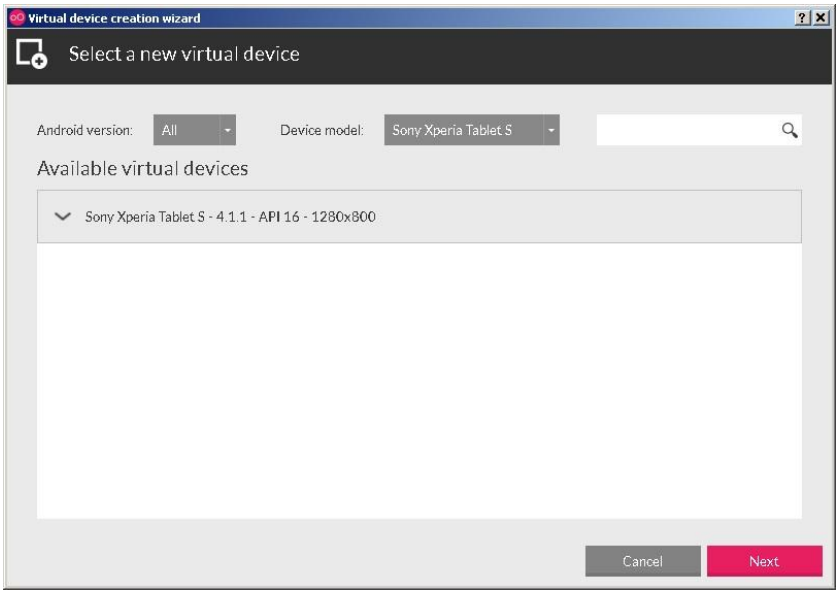


Figura 4.3.12 Selección del dispositivo virtual para descargar

Al pulsar el botón “Next”, se muestra una ventana con las características del dispositivo virtual, como se ve en la figura 4.3.13.

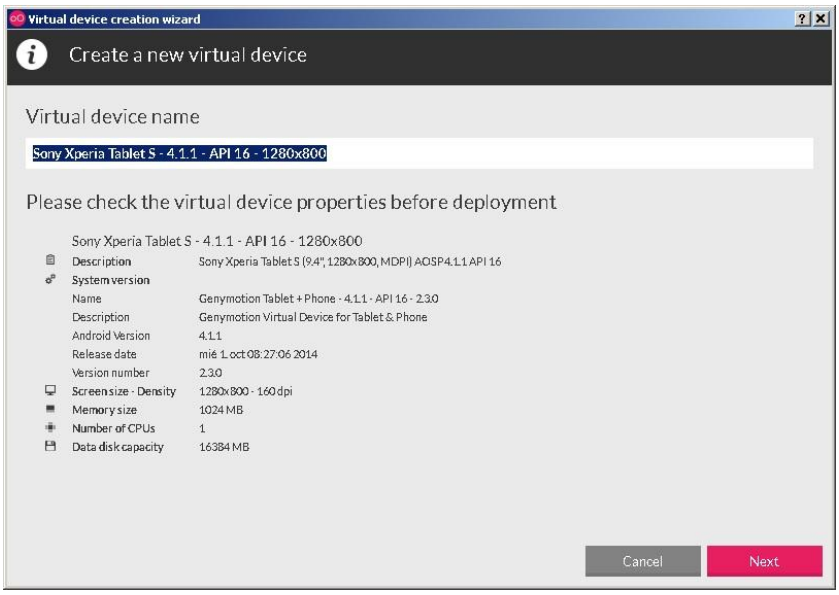


Figura 4.3.13 Propiedades del dispositivo virtual

Al oprimir el botón “Next”, se empieza a descargar el dispositivo virtual como se contempla en la figura 4.3.14.

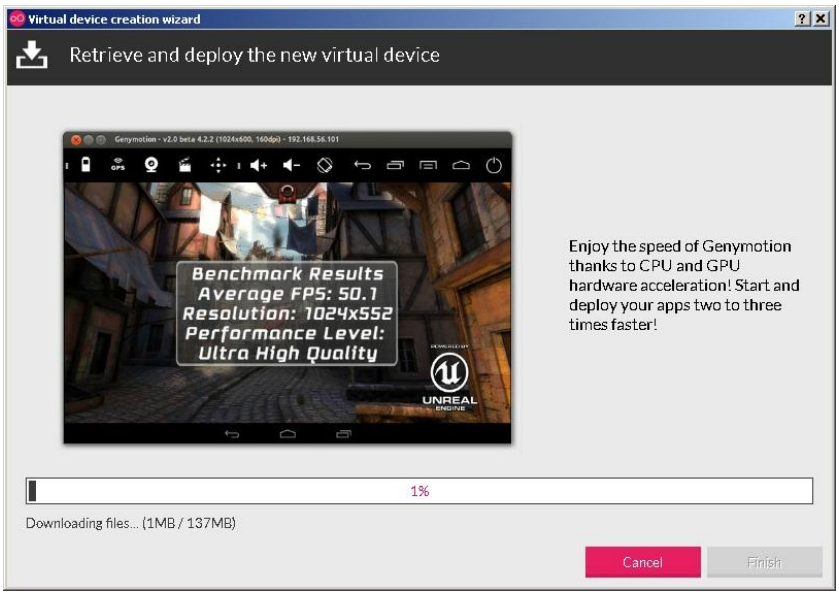


Figura 4.3.14 Proceso de descarga del dispositivo virtual

Al terminar la descarga se presiona el botón “Finish”. Se pueden añadir los dispositivos virtuales que se necesite, repitiendo los pasos anteriores. Asomando dichos dispositivos como se ve en la figura 4.3.15.

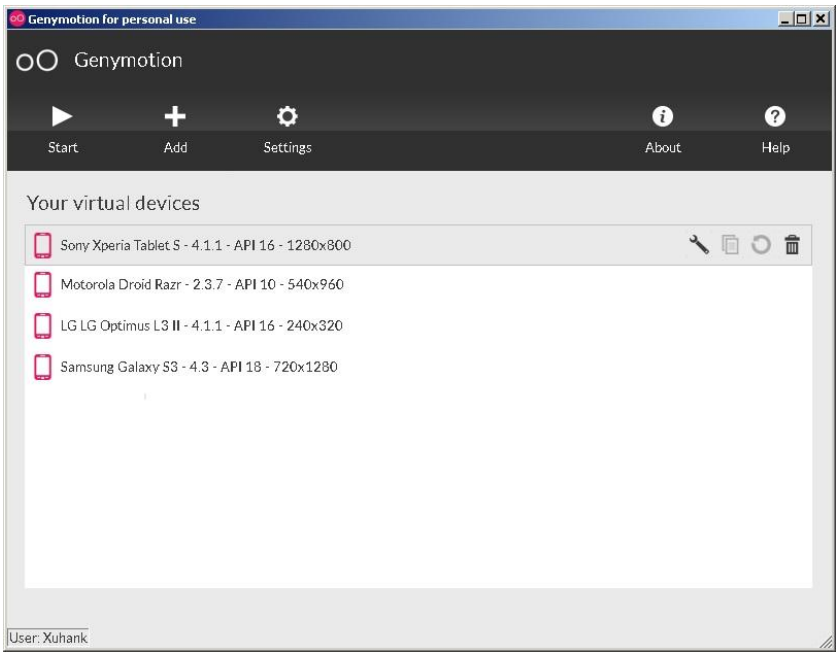


Figura 4.3.15 Dispositivos virtuales disponibles

4.3.4 Creación de la app “Conoce Tulcán”

Para comenzar el desarrollo de la app “Conoce Tulcán”, se debe crear un nuevo proyecto, para lo cual se da un clic en “File”, se selecciona “New” y se escoge “Android Application Proyect”. En la ventana que aparece se debe escribir el nombre de la aplicación (Application Name), el nombre del proyecto (Proyect Name) y el identificador de la aplicación (Package Name), elegir la versión mínima de Android compatible (Minimum Required SDK), la versión máxima (Target SDK), la versión a utilizar en el desarrollo de la app (Compile With) y escoger un tema (Theme). De la manera que se ve en la ventana de la figura 4.3.16.

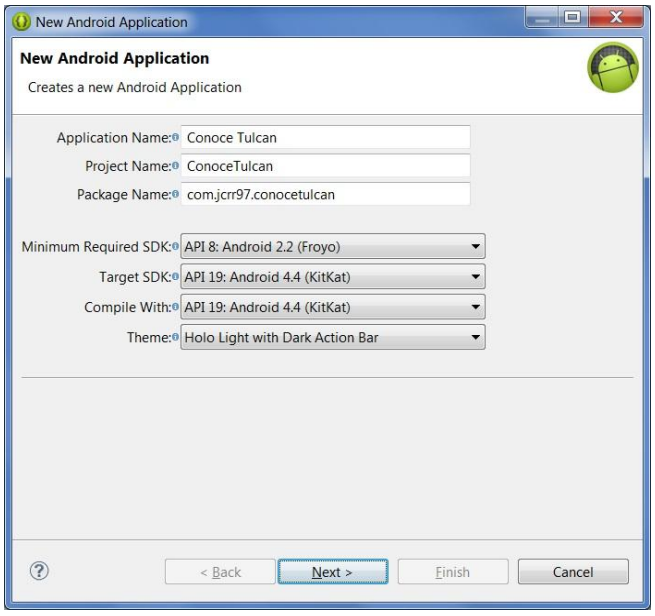


Figura 4.3.16 Ventana para crear un proyecto

Una vez que se han introducido los datos, se hace clic en el botón “Next”. Asomando la ventana de la figura 4.3.17. En esta ventana se puede activar las casillas crear icono personalizado (Create Custom launcher icon), crear una actividad (Create activity), definir el comportamiento del proyecto mediante la opción "Mark this project as a library", informar la ubicación del proyecto (Create Proyect in Workspace) o añadir el proyecto a working set (Add proyect to working sets).

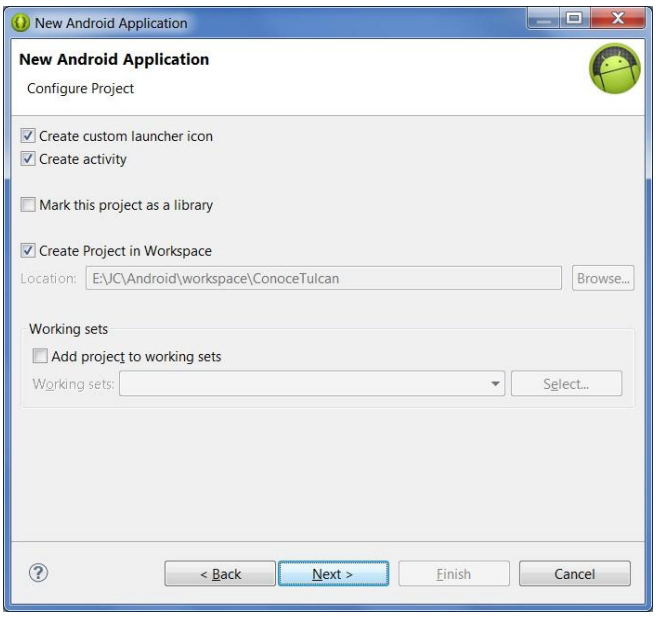


Figura 4.3.17 Ventana para configurar un proyecto

Al pulsar el botón “Next” se presenta la ventana de la figura 4.3.18. Aquí se configura el icono para la aplicación, con las siguientes opciones:

- Crear el icono desde una imagen personalizada (Image).
- Crear el icono desde una imagen propuesta por el SDK Android (Clipart).
- Crear el icono desde texto (Text).

También se puede especificar algunas propiedades del icono:

- Margen interno (Padding)
- Tamaño (Foreground Scaling)
- Forma (Shape)
- Color de fondo (Background Color)
- Color (Foreground Color)

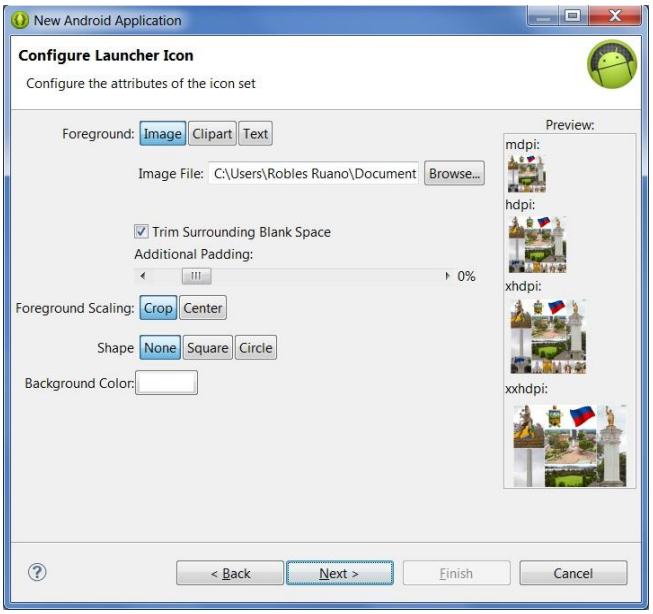


Figura 4.3.18 Ventana para configurar el icono

Para continuar se presiona el botón “Next”, desplegándose la ventana de la figura 4.3.19. Aquí se crea una actividad, seleccionado una de las opciones siguientes:

- Actividad vacía (Blank Activity)
- Actividad a pantalla completa (Fullscreen Activity)
- Actividad maestro/detalle (Master/Detail Flow)

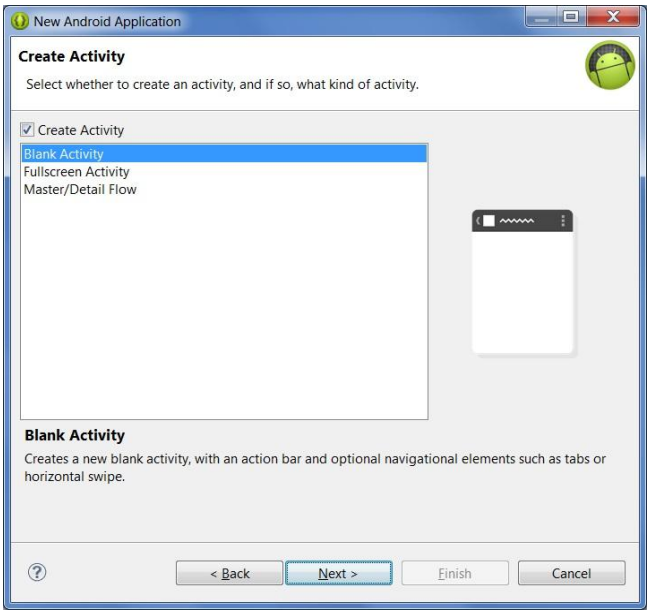


Figura 4.3.19 Ventana para crear una actividad

Al preferir la opción “Blank Activity” aparece la ventana de la figura 4.3.20.

Se llenan los campos con nombres que se desee, en este caso *Portada.*

En esta ventana se indica el estado inicial de la actividad:

- Nombre de la actividad (Activity Name)
- Nombre del archivo que sirve para crear la ventana (Layout Name)
- Nombre del archivo para crear la vista fragmento (Fragment Layout Name)
- Tipo de navegación (Navigation Type)
    - None
    - Swipes Views
    - Action Bar Tabs
    - Action Bar Spinner
    - Navigation Drawer

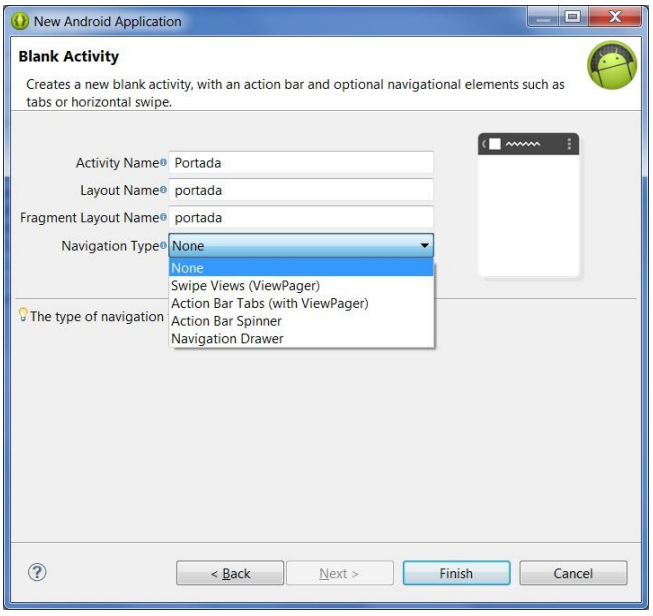


Figura 4.3.20 Ventana para crear una actividad vacía

Para terminar la creación del proyecto, tiene que hacer clic en el botón “Finish”. El proyecto aparecerá en la pestaña Package Explorer, tal como se puede observar en la figura 4.3.21.

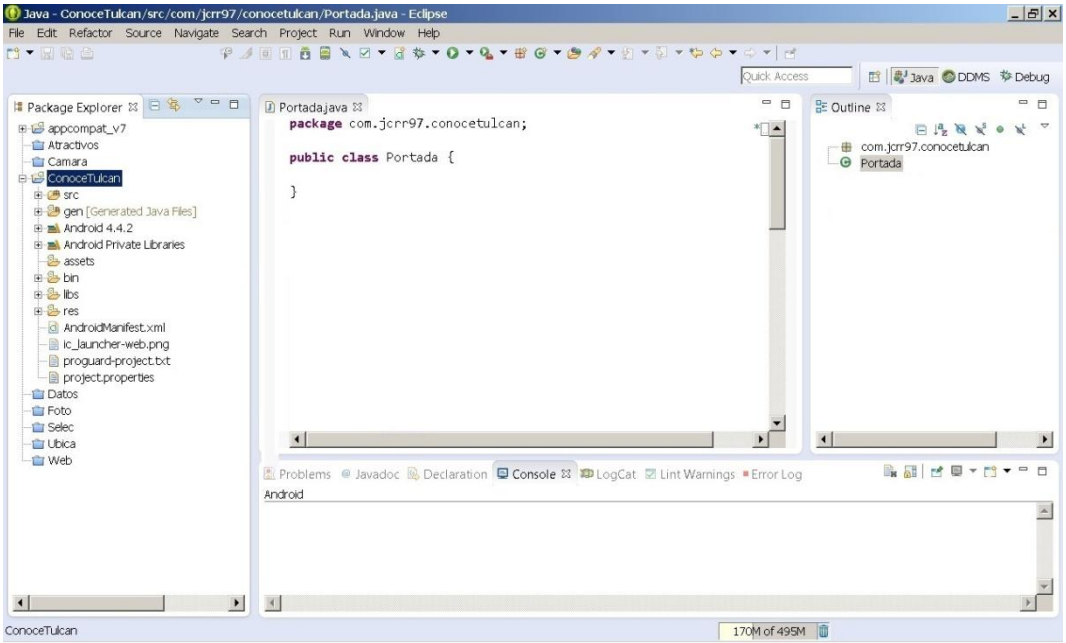


Figura 4.3.21 Actividad Portada del proyecto Conoce Tulcán

Para ver el layout creado, se dirige a la carpeta del proyecto “ConoceTulcan” tiene que dar doble clic en la carpeta “res”, luego hacer doble clic en la carpeta “layout” y por último doble clic en el archivo “portada.xml”.

Apareciendo dicho archivo en la interfaz de Eclipse, como se muestra en la figura 4.3.22.

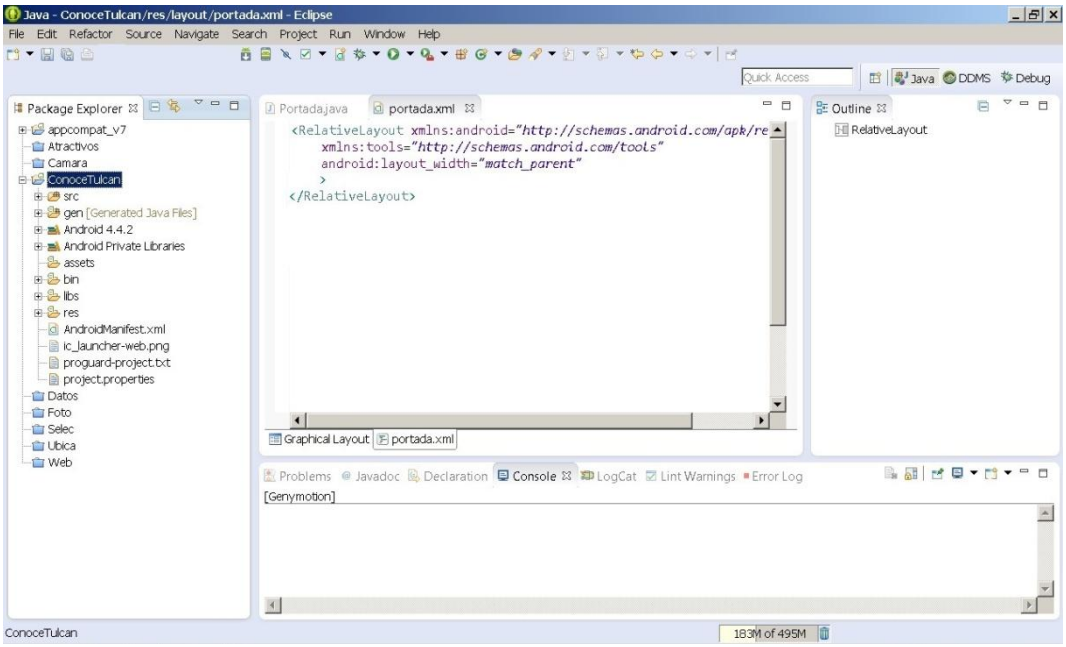


Figura 4.3.22 Layout “portada.xml”

En la clase “Portada” y en el archivo “portada.xml” se debe escribir el código necesario para cumplir con el diseño anteriormente detallado.

Para crear una nueva actividad o clase, se da un clic en “File” o clic derecho sobre la carpeta del proyecto, luego se elige “New” y se escoge “Class”, presentándose una ventana como se ve en la figura 4.3.23. Se tiene que buscar el paquete (Package) donde se quiere crear la nueva clase, en este caso se busca “com.jcrr97.conocetulcan” y se escribe el nombre de la actividad (Name).



Figura 4.3.23 Creación de la actividad “Ubica”

Al pulsar el botón “Finish” aparece la ventana que se puede apreciar en la figura 4.3.24.

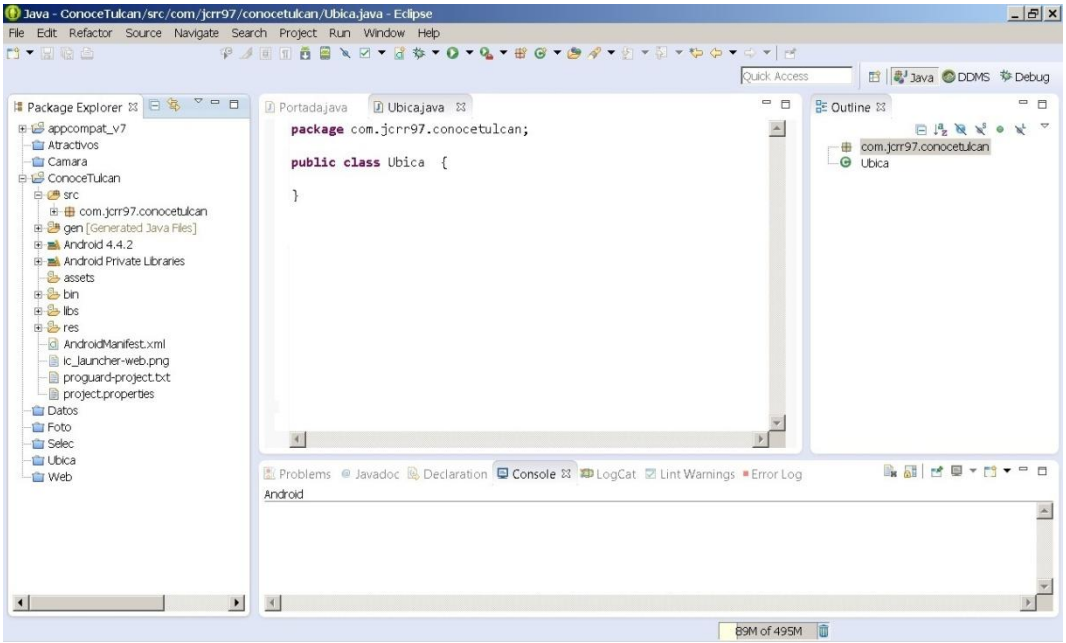


Figura 4.3.24 Actividad Ubica en el proyecto Conoce Tulcán

En esta clase se escribe el correspondiente código fuente para llevar a cabo los requerimientos de diseño.

Para crear el layout correspondiente a la actividad “Ubica”, se da clic derecho sobre la carpeta del proyecto, se escoge “New” y luego “Android XML File”; desplegándose la ventana que se mira en la figura 4.3.25, en la cual se escribe el nombre del archivo en “File”, se elige el elemento raíz a crear (Root Element) y se presiona el botón “Finish”.

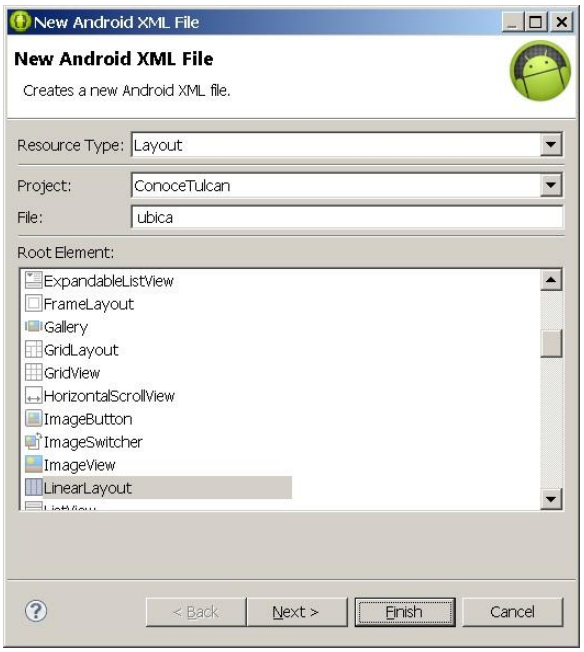


Figura 4.3.25 Creación del archivo ubica.xml

Se repite los pasos anteriores para crear las actividades y layouts necesarios para efectuar el desarrollo de la app “Conoce Tulcán”, cumpliendo los requerimientos planteados en el diseño. Todo el código fuente de la aplicación se encuentra en el Anexo 1.

En el archivo “AndroidManifest.xml”, ubicado en la carpeta del proyecto, se debe especificar la versión mínima y máxima de Android que soporta la app, los permisos de uso de hardware, todas las actividades creadas, la orientación de la aplicación, entre otras. Como se muestra en la figura 4.3.26.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.jcrr97.conocetulcan"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="14"
        android:targetSdkVersion="19" />

    <uses-permission android:name="android.permission.CAMERA" />
    <uses-feature android:name="android.hardware.camera" />
    <uses-feature android:name="android.hardware.camera.autofocus" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
    <uses-permission android:name="android.permission.INTERNET" />

    <application
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name="com.jcrr97.conocetulcan.Portada"
            android:label="@string/app_name"
            android:screenOrientation="portrait" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name=".Ubica"
            android:label="@string/app_name"
            android:screenOrientation="portrait">
            </activity>
         <activity
            android:name=".Selec"
            android:label="@string/app_name"
            android:screenOrientation="portrait">
            </activity>
             <activity
            android:name=".Foto1"
            android:label="@string/app_name"
            android:screenOrientation="portrait">
            </activity>
         <activity
            android:name=".Foto2"
            android:label="@string/app_name"
            android:screenOrientation="portrait">
            </activity>
            <activity
            android:name=".Foto3"
            android:label="@string/app_name"
            android:screenOrientation="portrait">
            </activity>
            <activity
            android:name=".Acerca"
            android:label="@string/acerc"
            android:screenOrientation="portrait">
            </activity>
    </application>

</manifest>
```

Figura 4.3.26 Archivo “AndroidManifest.xml”

4.3.5 Emulación de la app en Genymotion

Primeramente se ingresa al IDE Eclipse, se da clic sobre el icono del emulador Genymotion, abriéndose la ventana que se muestra en la figura 4.3.27.

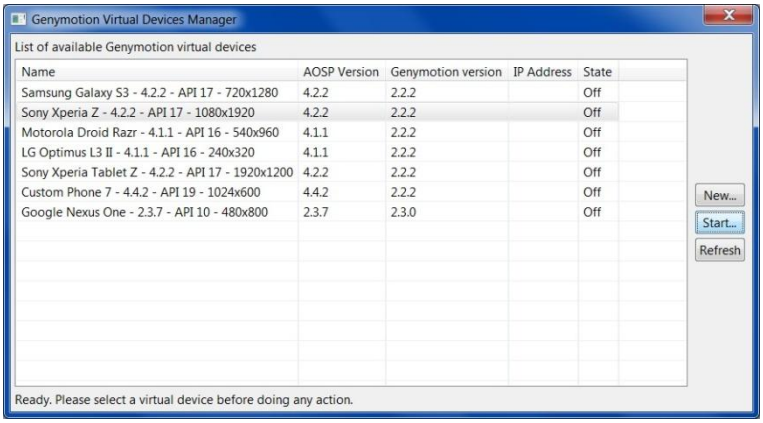


Figura 4.3.27 Dispositivos virtuales de Genymotion

Enseguida se escoge un terminal virtual y se pulsa el botón “Star”, dando inicio a la carga del dispositivo. A continuación en el IDE Eclipse, se hace clic sobre “Run” y luego en “Run Configuration”, asomándose la ventana que se ve en la figura 4.3.28. Se debe buscar el proyecto (Proyect), se elige la primera actividad de la app en “Launch”.

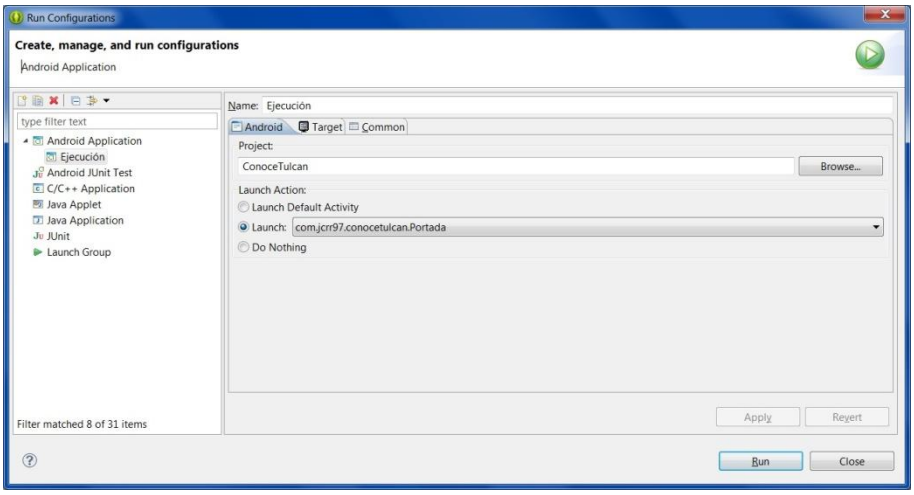


Figura 4.3.28 Configuración para ejecutar la app

Al oprimir “Run” sale la ventana que se observa en la figura 4.3.29, en donde se selecciona el dispositivo virtual anteriormente elegido.

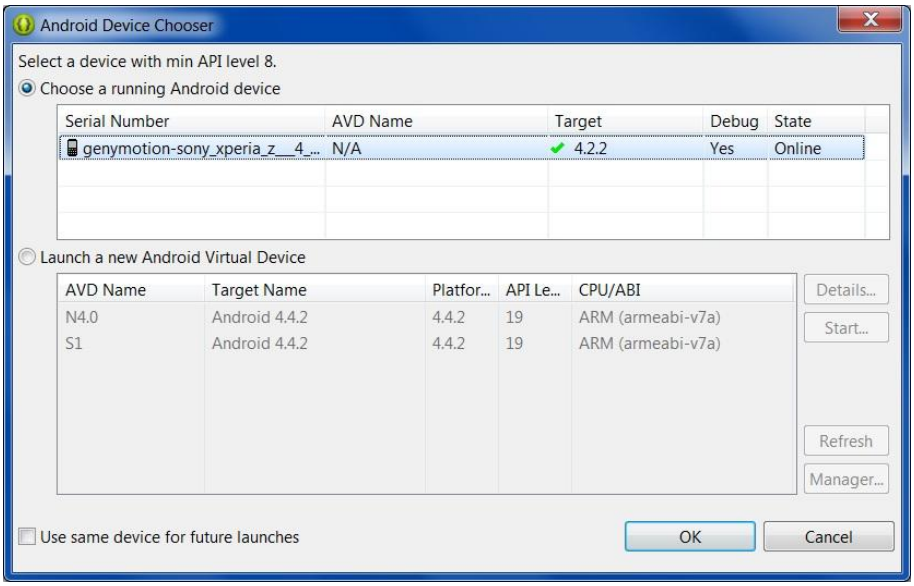


Figura 4.3.29 Selección del dispositivo virtual

Al pulsar el botón “OK” aparece la app en el dispositivo virtual, como se muestra en la figura 4.3.30.



Figura 4.3.30 Ventana de inicio

Antes de comenzar con la utilización de la app, se debe activar la cámara y el GPS, pulsando sobre los iconos respectivos que se encuentran en la parte lateral superior derecha del dispositivo virtual.

En la ventana correspondiente a la cámara que se observa en la figura 4.3.31, se debe seleccionar “Webcam1” y “Back”, para salir se pulsa “X”.

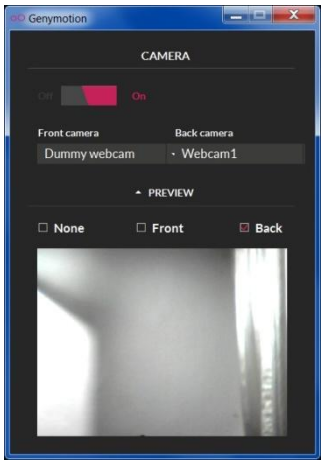


Figura 4.3.31 Selección de cámara

En la ventana perteneciente al GPS que se ve en la figura 4.3.32, se debe colocar las coordenadas geográficas del atractivo turístico que permiten realizar la emulación. Para salir se pulsa “X”, volviendo a la ventana de la figura 4.3.30.



Figura 4.3.32 Colocación de coordenadas

En la mencionada ventana se presiona “Inicio”, pasando a la ventana que se observa en la figura 4.3.33.

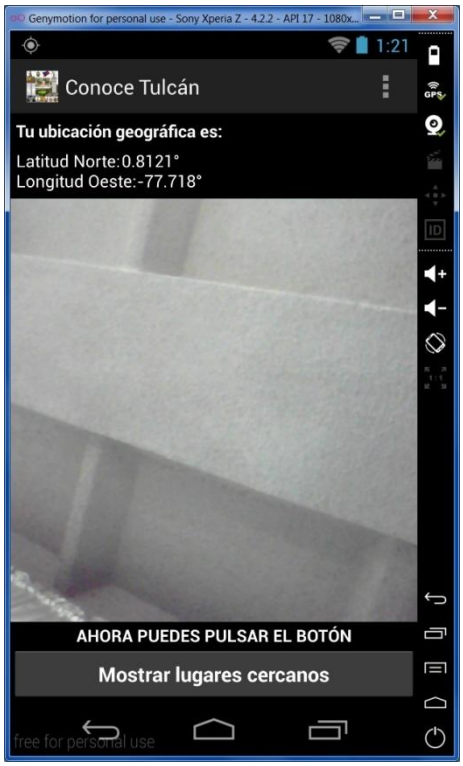


Figura 4.3.33 Ventana de ubicación geográfica

Para continuar se oprime el botón “Mostrar lugares cercanos”, asomando la ventana que se mira en la figura 4.3.34.



Figura 4.3.34 Lista de lugares cercanos

Enseguida se selecciona el atractivo turístico que se desee, por ejemplo “Catedral de Tulcán”, avanzando a la ventana que se aprecia en la figura 4.3.35.

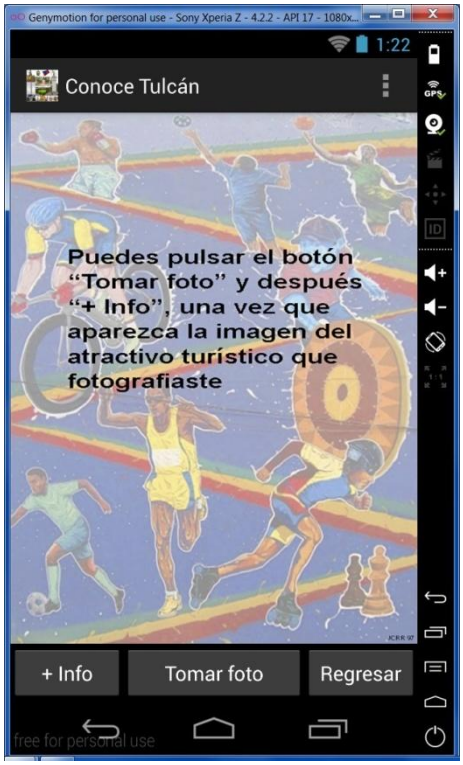


Figura 4.3.35 Ventana para tomar foto y mostrarla

Al pulsar el botón “Tomar foto” se despliega la cámara para fotografiar el atractivo turístico elegido anteriormente, se guarda la fotografía tomada y aparece la imagen con el nombre del atractivo sobreimpreso en la parte superior izquierda, como se muestra en figura 4.3.36.



Figura 4.3.36 Imagen con nombre del atractivo turístico

A continuación se pulsa el botón “+ Info”, abriéndose el navegador web con una breve información del atractivo turístico, como se muestra en la figura 4.3.37.



Figura 4.3.37 Breve información de la Catedral de Tulcán

Por último, se puede oprimir el botón “Regresar” para volver a la ventana que permite seleccionar un atractivo turístico o tocar el botón “Salir” para finalizar la app.

Al pulsar el botón “Menú” del dispositivo virtual en cualquier actividad, se abre una ventana con una corta información sobre la app, como se muestra en la figura 4.3.38. Al presionar “Regresar” se vuelve a la actividad que se encontraba.



Figura 4.3.38 Corta información de la app

De esta manera se ha conseguido emular exitosamente la app, indicando paso a paso cómo se debe interactuar en cada actividad para tener una manipulación adecuada de la aplicación “Conoce Tulcán”.

4.3.6 Generación del archivo APK

Una vez que se ha emulado la app se debe proceder a la obtención del archivo APK, para lo cual se debe seguir los siguientes pasos:

- En la interfaz del IDE Eclipse se pulsa “File” y luego “Export”, abriéndose la ventana que se ve en la figura 4.3.39.

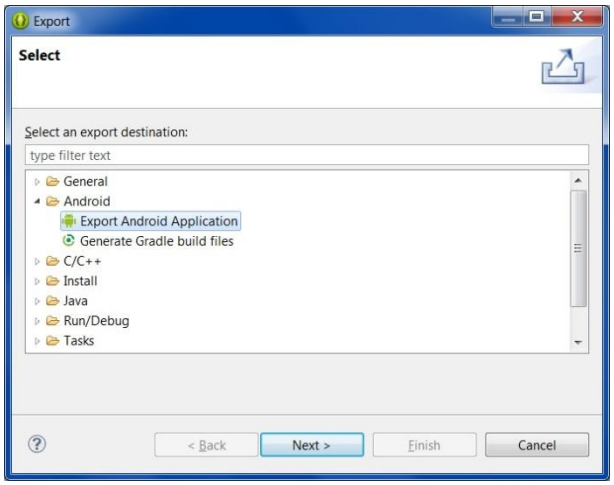


Figura 4.3.39 Ventana para exportar archivos

- En dicha ventana se despliega la carpeta “Android”, se escoge “Export Android Application” y se presiona el botón “Next”; pasando a la ventana que se muestra en la figura 4.3.40.

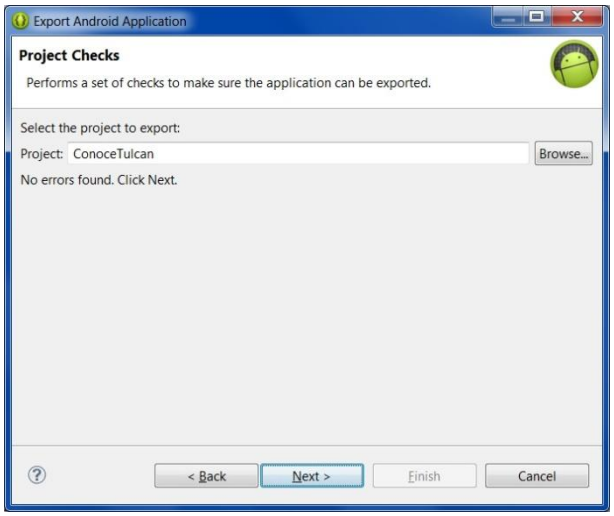


Figura 4.3.40 Ventana de verificación del proyecto

- En la ventana anterior se oprime el botón “Browse” para ubicar el proyecto, a continuación aparece un mensaje de verificación “No errors found. Click Next”, se da clic en el botón “Next”, asomando la ventana que se observa en la figura 4.3.41.

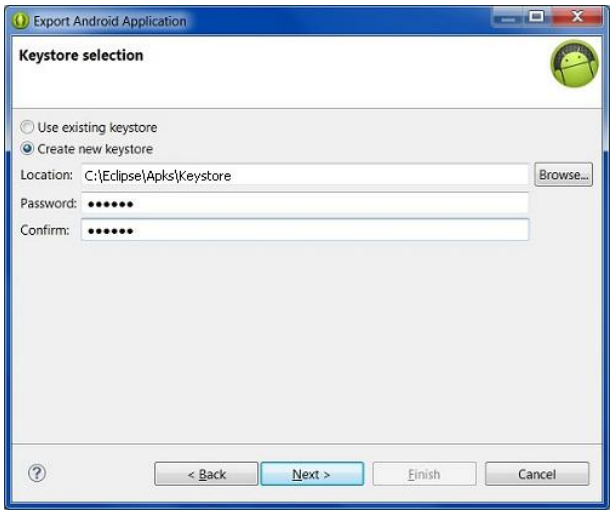


Figura 4.3.41 Ventana de “Keystore”

- En la ventana previa se debe crear una nueva clave para archivar, para lo cual se busca donde se quiere guardar la clave pulsando el botón “Browse”, se escribe una contraseña, se confirma y se oprime el botón “Next” presentándose la ventana que se puede mirar en la figura 4.3.42.

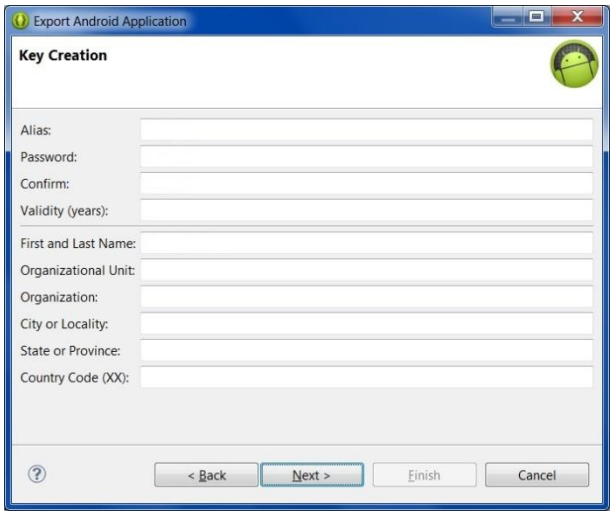


Figura 4.3.42 Ventana de creación de la clave

- En la mencionada ventana se tiene que escribir los datos de usuario, en validez “Validity (years)” se coloca 25 y se da clic en el botón “Next” apareciendo la ventana que se indica en la figura 4.3.43.

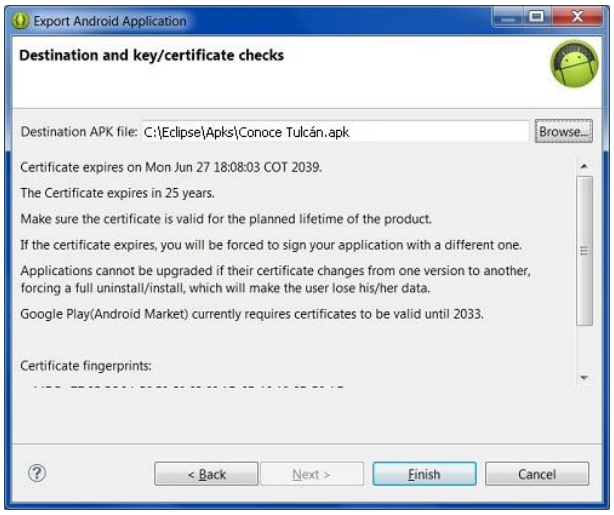


Figura 4.3.43 Ventana de destinación del archivo APK

- En la última ventana se elige la carpeta de destino del archivo APK, presionando el botón “Browse” y para salir se pulsa “Finish”.

De esta manera se ha conseguido el archivo de la app “Conoce Tulcán.apk”, el cual tiene que ser subido a la página de descarga, “https://sites.google.com/site/conocetulcan/app-android”, del sitio web creado que sirve como soporte a la aplicación móvil.

CAPÍTULO 5

INSTALACIÓN Y PRUEBAS DE LA APP EN DISPOSITIVOS MÓVILES

5.1 INSTALACIÓN

La app debe ser instalada en dispositivos móviles que cumplan con los requisitos siguientes:

- Sistema operativo Android, mínimo Ice Cream Sandwich API 14 hasta KitKat API 19
- Receptor de GPS
- Conexión a Internet
- Cámara posterior
- Almacenamiento de archivos

Para proceder con la instalación, con el dispositivo móvil en sus manos, se tiene que abrir el navegador web, luego se digita la URL siguiente "https://sites.google.com/site/conocetulcan/app-android". En esta página debe encontrarse el archivo "Conoce Tulcán.apk", dar un clic en la flecha azul para descargarlo, como se presenta en la figura 5.1.1.



Figura 5.1.1 Página de descarga de la aplicación "Conoce Tulcán"

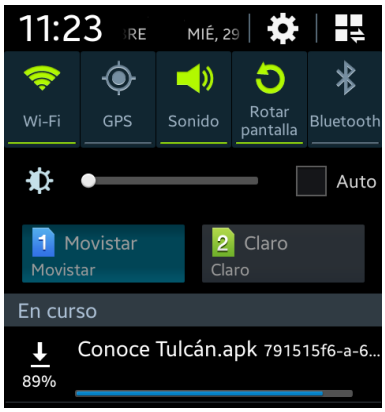


Figura 5.1.2 Proceso de la descarga

Antes de instalar la app, se debe habilitar la opción que reconoce aplicaciones desconocidas a Google Play, para lo cual se oprime el icono de “Ajustes” o “Configuración” del dispositivo móvil, a continuación se selecciona “Seguridad” y finalmente se habilita la opción “Fuentes desconocidas”, como se indica en la figura 5.1.3.

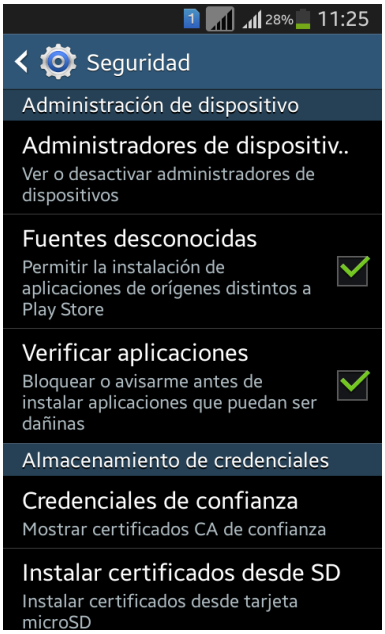


Figura 5.1.3 Habilitación de “Fuentes desconocidas”

Enseguida tiene que dirigirse al explorador de archivos o presionar en el icono “Mis archivos” y buscar la carpeta “Download”, como se observa en la figura 5.1.4.

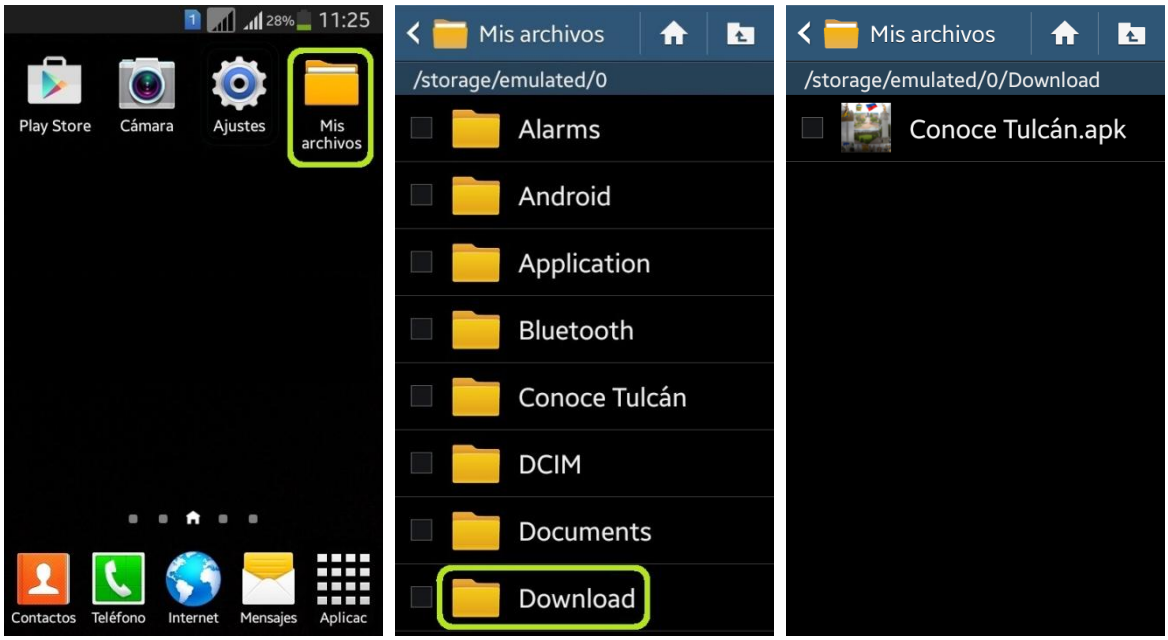


Figura 5.1.4 Ubicación del archivo descargado “ConoceTulcán.apk”

En esta ventana se presiona sobre el archivo “Conoce Tulcán.apk”, luego se elige la opción “Instalador del paquete” y por último se toca el botón “Siempre” o “Solo una vez” para iniciar el proceso de instalación, como se muestra en la figura 5.1.5.

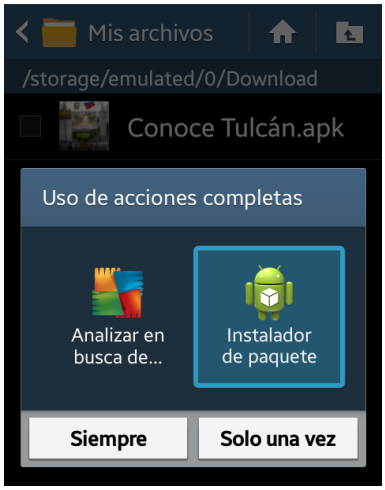


Figura 5.1.5 Instalador de paquete

Después aparece una ventana, mostrando el hardware que va a utilizar la app para su funcionamiento, como se ve en la figura 5.1.6.

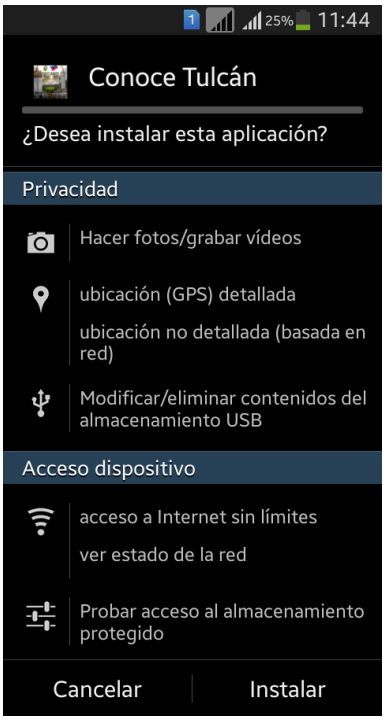


Figura 5.1.6 Hardware a utilizar la app

Finalmente se pulsa el botón “Instalar”, apareciendo las ventanas que se presentan en la figura 5.1.7 y 5.1.8.

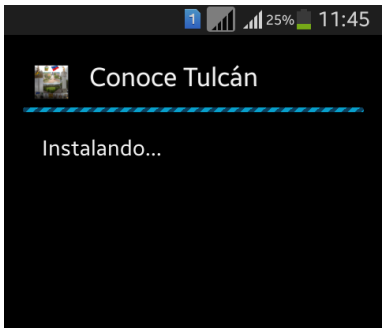


Figura 5.1.7 Proceso de instalación

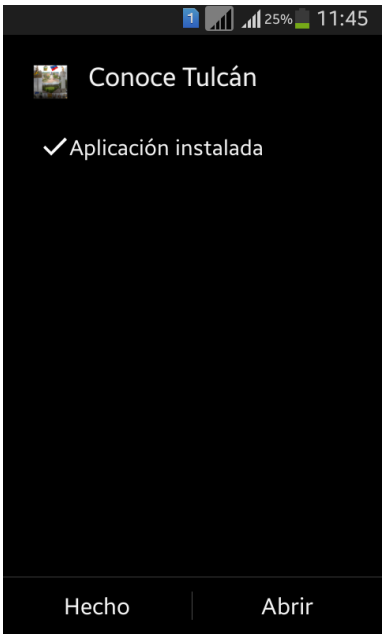


Figura 5.1.8 Finalización de la instalación

Una vez instalada la app se pulsa el botón “Hecho” para finalizar la instalación.

5.2 FUNCIONAMIENTO

Para acceder a la app se ingresa a las aplicaciones instaladas en el dispositivo móvil, como se puede contemplar en la figura 5.2.1.



Figura 5.2.1 Icono de la app

Acto seguido se oprime sobre el icono de la app, se abre la ventana de inicio, como se ve en la figura 5.2.2. Se lee las indicaciones generales.



Figura 5.2.2 Ventana de inicio

Antes de continuar se debe activar el GPS y la conexión a Internet, lo cual se lo puede hacer desde el panel de notificaciones, para abrirlo se desliza el dedo desde la parte superior de la pantalla hacia abajo, aquí se procede a la activación de GPS y Wi-Fi o Datos móviles, para volver a la ventana de inicio de la app se desliza el dedo desde la parte inferior de la pantalla hacia arriba o se pulsa la tecla “Home” o la tecla “Back” del dispositivo.

Ahora se puede pulsar el botón “Iniciar”, la app comienza a buscar la ubicación geográfica del dispositivo móvil, apareciendo la ventana como se muestra en la figura 5.2.3.



Figura 5.2.3 Búsqueda y resultado de la ubicación geográfica

Luego se presiona el botón “Mostrar lugares cercanos”, desplegándose una ventana con la lista de los atractivos turísticos aledaños a la ubicación del usuario, como se puede ver en la figura 5.2.4.



Figura 5.2.4 Atractivos turísticos cercanos al usuario

Acto seguido se elige el atractivo turístico que se desee, pulsando sobre su nombre, pasando a la ventana que se aprecia en la figura 5.2.5.

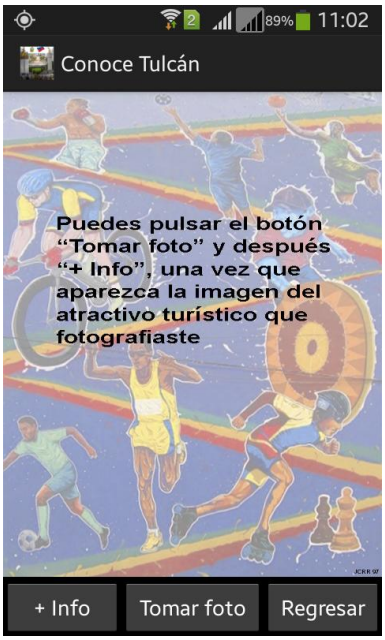


Figura 5.2.5 Ventana para tomar una foto y mostrarla

Para proseguir se oprime el botón "Tomar foto", abriéndose la cámara del dispositivo, como se muestra en la figura 5.2.6.



Figura 5.2.6 Enfocando y capturando el atractivo turístico

Una vez hecha la captura, se pide "Descartar" o "Guardar" la imagen, como se ve en la figura 5.2.7.



Figura 5.2.7 Imagen capturada

Para continuar se presiona el botón “Guardar”, mostrándose en una ventana con la imagen del atractivo turístico elegido anteriormente con su nombre sobreimpeso, como se puede observar en la figura 5.2.8.

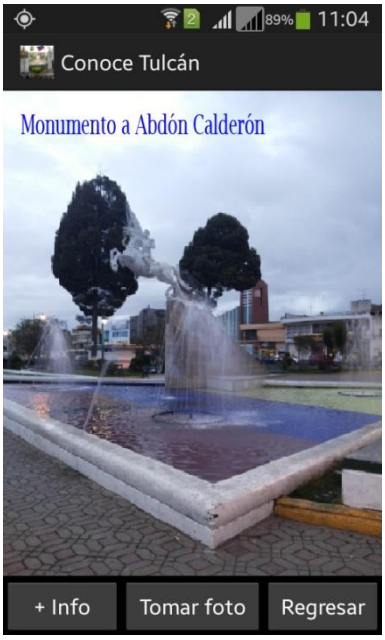


Figura 5.2.8 Imagen del atractivo turístico con su nombre sobreimpeso

Si se quiere más información sobre el “Monumento a Abdón Calderón”, se pulsa el botón “+ Info”, abriéndose el navegador web con una breve información del mencionado monumento, como se presenta en la figura 5.2.9.

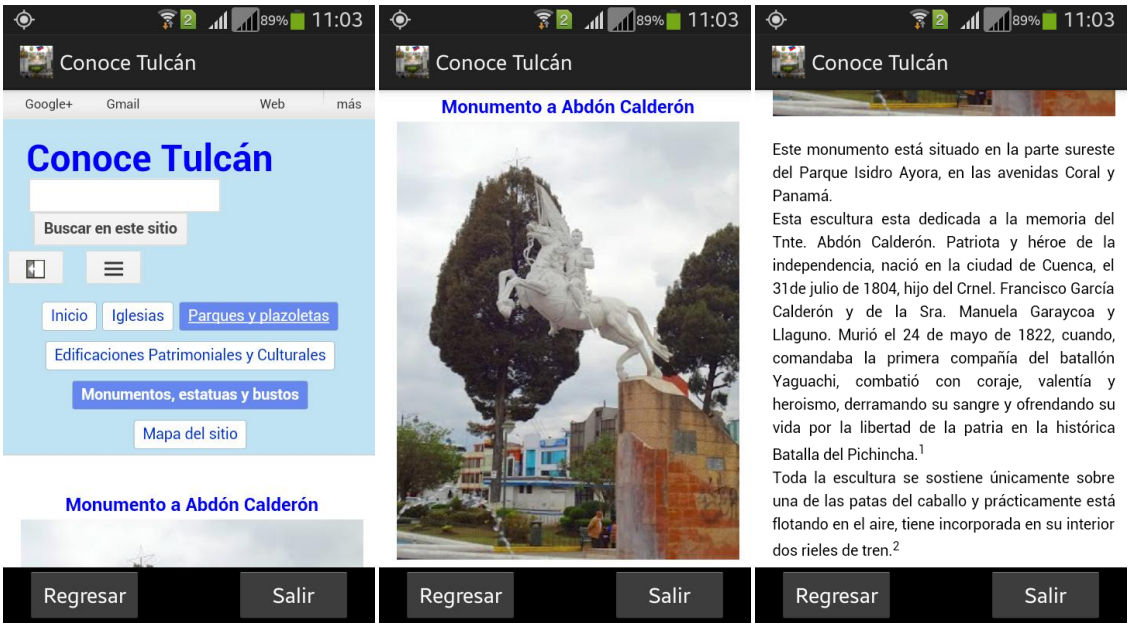


Figura 5.2.9 Breve información del “Monumento a Abdón Calderón”

En la ventana anterior, al presionar el botón “Regresar”, se vuelve a la actividad que exhibe la lista de atractivos turísticos aledaños al usuario o al oprimir el botón “Salir” se finaliza y se abandona la app.

Durante la ejecución de la aplicación, en cualquier ventana se puede pulsar el botón “Menú” del dispositivo para acceder a una corta información de la app, como se puede mirar en la figura 5.2.10.



Figura 5.2.10 Corta información de la app

En la ventana anterior, al oprimir “Regresar” se vuelve la actividad respectiva.

De esta manera se ha explicado el correcto funcionamiento de la aplicación móvil, como se pudo observar toda la ejecución dura aproximadamente dos minutos.

Por último se va a indicar la ubicación donde se guardan los archivos de las fotografías tomadas a los atractivos turísticos. Para lo cual se debe ingresar a la carpeta “Mis archivos” del dispositivo móvil, como se ve en la figura 5.2.11.

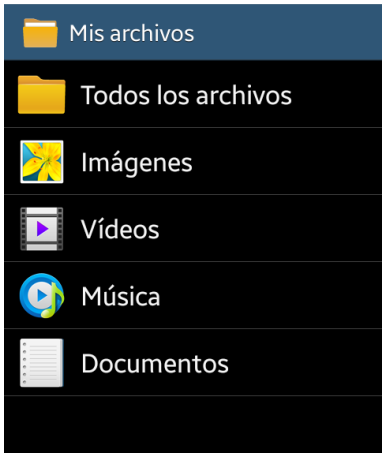


Figura 5.2.11 Carpeta “Mis archivos”

Luego se presiona sobre “Todos los archivos”, apareciendo una ventana como se muestra en la figura 5.2.12.

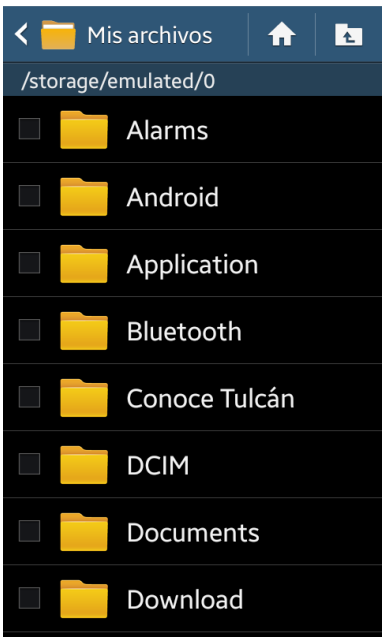


Figura 5.2.12 Todas las carpetas del dispositivo

Se oprime sobre la carpeta “Conoce Tulcán”, pasando a una ventana con los archivos de las fotografías tomadas a ciertos atractivos turísticos, como se indica

en la figura 5.2.13. Dicha carpeta se crea automáticamente al momento de la primera ejecución de la app.

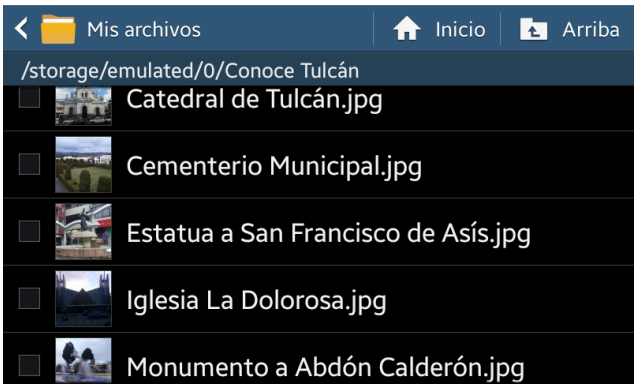


Figura 5.2.13 Archivos de las fotografías de ciertos atractivos turísticos

5.3 PRUEBAS

5.3.1 Prueba de usabilidad

Se llevó a cabo esta prueba con la finalidad de verificar la manipulación de la app, por un grupo de diez personas con diferentes edades. Para lo cual se utilizó un Smartphone Samsung Galaxy Grand Duos conectado a una red Wi-Fi y se les realizó tres preguntas sobre la ejecución de la aplicación, consiguiendo los resultados siguientes:

Persona	Edad	Dificultad en el manejo Alta, media, baja	Presentación de la información Excelente, buena, mala	Tiempo de respuesta Largo, medio, corto
1	33	Baja	Excelente	Corto
2	14	Baja	Excelente	Corto
3	26	Baja	Excelente	Corto
4	62	Baja	Excelente	Corto
5	57	Baja	Excelente	Corto
6	37	Baja	Excelente	Corto
7	44	Baja	Excelente	Corto
8	29	Baja	Excelente	Corto
9	18	Baja	Excelente	Corto
10	50	Baja	Excelente	Corto

Tabla 5.3.1 Resultados de la prueba de usabilidad

De acuerdo a los resultados, todas las personas que colaboraron no tienen ninguna dificultad en el manejo de la app, manifestaron que la información

expuesta es excelente y los tiempos de respuesta son cortos. Esto indica que las ventanas de la aplicación son comprensibles, amigables e intuitivas.

5.3.2 Prueba de funcionalidad

Se hizo esta prueba para confirmar el cumplimiento de los requerimientos establecidos previamente en el diseño de la app.

Requerimiento funcional	Resultado
Utilización del GPS	OK
Conexión a Internet	OK
Empleo de cámara	OK
Almacenamiento de las fotografías tomadas	OK
Ventanas comprensibles, amigables e intuitivas	OK
No tiene ventanas deslizantes	OK
Sobreimpresión del nombre del atractivo en la imagen resultante mostrada en la pantalla	OK
Interfaz gráfica del sitio web con la breve información de los atractivos turísticos	OK
No genera ningún conflicto con otras apps instaladas	OK
No colapsa durante su ejecución	OK

Tabla 5.3.2 Resultados de la prueba de funcionalidad

5.3.3 Pruebas técnicas

- La primera prueba que se efectuó fue la comprobación del funcionamiento de aplicación móvil en varios equipos, con versiones diferentes del sistema operativo Android. Obteniéndose los resultados que se muestran en la tabla 5.3.3.

Dispositivo móvil		Sistema Operativo	Resultado
Tipo	**Marca**		
Smartphone	Huawei Y530	Jelly Bean 4.3 API 18	OK
Smartphone	Samsung GT-I90822L	Jelly Bean 4.2.2 API 17	OK
Smartphone	LG E612g	Ice Cream Sandwich 4.0.3 API 15	OK
Tablet	Acer Iconia B1-710	Jelly Bean 4.1.2 API 16	OK
Tablet	Samsung GT-P3100	Ice Cream Sandwich 4.0.3 API 15	OK

Tabla 5.3.3 Resultados del funcionamiento en varios dispositivos móviles

Según los resultados obtenidos, la aplicación móvil sí es compatible con diferentes versiones del sistema operativo Android en varios equipos, tal como se planteó en el diseño de la misma.

- La siguiente prueba consistió en comparar el desempeño de la app en una red Wi-Fi y en un red móvil. Para lo cual se empleó un Smartphone Samsung Galaxy Grand Duos y se cronometró el tiempo de respuesta en dos actividades de la aplicación móvil, durante su ejecución en tres lugares diferentes de la Ciudad de Tulcán. Consiguiendo los resultados que se observan en la tabla 5.3.4.

Lugar	Actividad de la aplicación móvil	Red	
		Wi-Fi	**Móvil**
Parque Ayora	Determinación de la ubicación geográfica	3,52”	9,81”
	Información web del atractivo turístico	31,32”	39,34”
Parque Central	Determinación de la ubicación geográfica	4,15”	5,18”
	Información del atractivo turístico	13,49”	16,90”
Parque La Laguna	Determinación de la ubicación geográfica	4,87”	6,23”
	Información del atractivo turístico	6,91”	10,26”

Tabla 5.3.4 Comparación de tiempos de respuesta

La red Wi-Fi abierta, presente en cada uno de estos lugares, proporcionó una intensidad de la señal muy buena. Mientras que la red móvil que se utilizó fue de la operadora “Claro”, conectándose el dispositivo móvil a una red HSUPA o H+. Los tiempos pueden variar según la disponibilidad de las redes y de acuerdo al lugar donde se encuentre el usuario.

Los tiempos obtenidos en una red Wi-Fi abierta son relativamente bajos, pero este tipo de red permite una limitada movilización y no se encuentra disponible en todos los atractivos turísticos. En cambio con una red móvil, se consigue una gran movilidad y disponibilidad.

CAPÍTULO 6

CONCLUSIONES Y RECOMENDACIONES

6.1 CONCLUSIONES

- Se consiguió exitosamente el desarrollo de la app “Conoce Tulcán” para dispositivos móviles con el sistema operativo Android, cumpliéndose con todos los requerimientos establecidos. La aplicación se basa en la geolocalización y proporciona una breve información de los atractivos turísticos a los usuarios que visiten la Ciudad de Tulcán. Se consideró como atractivos turísticos urbanos a los siguientes puntos de interés: iglesias, parques, plazoletas, monumentos, estatuas, bustos, edificaciones patrimoniales y culturales.

- Se determinó que el sistema operativo Android es el más ventajoso para desarrollar aplicaciones móviles; por ser un sistema de código abierto, estar libre de licencias, no existir pago alguno por distribución o desarrollo, por encontrarse gran cantidad de información en Internet y ser el más utilizado en los dispositivos móviles por los fabricantes más reconocidos a nivel mundial.

- Se utilizó la tecnología necesaria disponible en los dispositivos móviles inteligentes, como: receptor de señales GPS, cámara posterior, conexión a Internet, almacenamiento de archivos, pantalla táctil, entre otros. Los cuales permiten el correcto funcionamiento de la aplicación “Conoce Tulcán”; además dicho funcionamiento se fundamenta en las redes de telecomunicaciones como: Internet, redes inalámbricas Wi-Fi, redes móviles y servicio de GPS.

- Se empleó las herramientas de desarrollo para los programadores proporcionadas por Android en su sitio web. Específicamente se usó IDE Eclipse con Android SDK. Se recurrió a la utilización del emulador Genymotion, ya que brinda un mejor desempeño respecto al emulador nativo. Todas estas herramientas facilitaron el desarrollo de la aplicación

"Conoce Tulcán" y el emulador permitió realizar pruebas de funcionamiento en dispositivos virtuales.

- Se creó correctamente el sitio web que sirve como soporte a la aplicación "Conoce Tulcán", cumpliéndose con todos los requerimientos establecidos. Para su creación se prefirió la aplicación online Google Sites, para evitar los problemas de alojamiento del sitio web, registro de dominio y el pago por dichos servicios. En este sitio se encuentra la información de los atractivos turísticos y la aplicación móvil con su manual de usuario para descargarlos. El sitio web se concibió para que la información no esté disponible a cualquier persona, sino que van a tener acceso solamente los usuarios que el dueño del sitio tenga previamente en sus registros.

- Se determinó que Google proporciona una tranquilidad a todos los usuarios de sus aplicaciones al saber que sus datos están protegidos, ya que dispone de certificados de seguridad cumpliendo con estándares de auditoría externa. Además ofrece una conectividad segura con el protocolo SSL / TLS, permitiendo establecer comunicaciones seguras en Internet, es decir una conexión HTTPS a los usuarios que acceden a Google Sites y a otras aplicaciones de Google.

- Se realizaron exitosamente las pruebas de funcionamiento de la aplicación "Conoce Tulcán". La prueba de usabilidad arrojó extraordinarios resultados, dejando satisfechos a los participantes. La prueba de funcionalidad permitió la verificación de los requerimientos establecidos, cumpliéndose en su totalidad. La aplicación fue instalada y ejecutada en varios Smartphones y Tablets de diferentes fabricantes con versiones distintas del sistema operativo Android, obteniéndose un excelente funcionamiento.

- Se comparó el funcionamiento de la aplicación "Conoce Tulcán" en una red Wi-Fi y en una móvil. Los tiempos que se obtuvieron en una red Wi-Fi abierta, fueron relativamente bajos, pero este tipo de red permitió una

limitada movilización y no se encuentra disponible en todos los atractivos turísticos. En cambio una red móvil, facilita gran movilidad y disponibilidad con tiempos relativamente altos.

6.2 RECOMENDACIONES

- La instalación de la aplicación “Conoce Tulcán” en un dispositivo móvil inteligente, se puede realizar teniendo en cuenta los requerimientos mínimos necesarios para que funcione óptimamente la app.

- Para una adecuada manipulación de la aplicación “Conoce Tulcán” y su correcto funcionamiento, se tiene que leer detenidamente los mensajes que aparecen en algunas ventanas de la aplicación móvil, durante el transcurso de su ejecución. Podrá ser manejada por niños, jóvenes y adultos que estén familiarizados con la tecnología disponible en los dispositivos móviles inteligentes.

- La utilización de la aplicación “Conoce Tulcán” se debe efectuar en lugares abiertos, donde el usuario observe los atractivos turísticos que le interesan y se ubique a una distancia adecuada para poder fotografiarlos; en el caso de los parques, plazoletas, bulevar y cementerio, el usuario tiene que estar en el área que ocupan dichos lugares.

- Las posibles mejoras de la aplicación “Conoce Tulcán” para versiones futuras podrían ser la inclusión de otros sitios de interés para los turistas como: restaurantes, hoteles, discotecas, farmacias, centros comerciales, aeropuerto, terminal de transporte, hospital, clínicas, entre otros. También se recurriría a la utilización de la realidad aumentada, para que las próximas versiones de la app sean más llamativas, proporcionando rápidamente información en tiempo real.

REFERENCIAS BIBLIOGRÁFICAS

[1] Morillo, David. *Introducción a los dispositivos móviles*. Barcelona, Editorial FUOC, 2011. Disponible en: http://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_moviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_%28Modulo_2%29.pdf. Acceso 22 de febrero del 2014

[2] Visita Ecuador. *Tulcán*. http://www.visitaecuador.com/ve/mostrarRegistro.php?idRegistro=489. Acceso 22 de febrero del 2014

[3] Gobierno Autónomo Descentralizado Municipal de Tulcán. *Cantón Tulcán*. http://www.gmtulcan.gob.ec/index.php/2012-06-06-13-23-58/2012-06-06-13-24-59. Acceso 22 de febrero del 2014

[4] Gobierno Autónomo Descentralizado Provincial del Carchi. *Cementerio Municipal de Tulcán.* http://www.carchi.gob.ec/index.php/turismo-canton-tulcan/cementerio-municipal-de-tulcan. Acceso 22 de febrero del 2014

[5] Garrido, Juan. *Desarrollo de Aplicaciones Móviles*. Cataluña, UOC, 2013. Disponible en: http://openaccess.uoc.edu/webapps/o2/bitstream/10609/18528/6/jugarridocoTFC0113memoria.pdf. Acceso 23 de febrero del 2014

[6] Google play. *Geolocalización*. https://play.google.com/store/search?q=geolocalizacion&c=apps. Acceso 25 de febrero del 2014
Wikitude. https://play.google.com/store/search?q=wikitude&c=apps&hl=es-419. Acceso 25 de febrero del 2014

[7] Departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Granada.*Tipos de dispositivos móviles.* http://leo.ugr.es/J2ME/INTRO/intro_4.htm. Acceso 8 de septiembre del 2014

[8] Imaginar, *Telefonía móvil para el desarrollo.* http://www.imaginar.org/iicd/fichas/04_Telefonia_movil.pdf. Acceso 9 de septiembre del 2014

[9] Gsmarena. *Samsung Galaxy S5.* http://www.gsmarena.com/samsung_galaxy_s5-6033.php *Motorola DEFY.* http://www.gsmarena.com/motorola_defy-3514.php *Sony Ericsson Xperia ray*. http://www.gsmarena.com/sony_ericsson_xperia_ray-3972.php *Samsung Galaxy Grand I9082.* http://www.gsmarena.com/samsung_galaxy_grand_i9082-5163.php. Acceso 9 de septiembre del 2014

[10] SUPERTEL, "Smartphone". *Revista Institucional SUPERTEL*, Número 16, Quito, 2012. Disponible en: http://www.supertel.gob.ec/pdf/publicaciones/revista_supertel_16_final.pdf Acceso 9 de septiembre del 2014

[11] Ortega, Rosario. *Tablets. La revolución táctil.* http://recursostic.educacion.es/observatorio/web/es/equipamiento-tecnologico/hardware/1012-tablets-la-revolucion-tactil-. Acceso 10 de septiembre del 2014

[12] Gsmarena. *Samsung Galaxy Tab S 8.4 LTE*. http://www.gsmarena.com/samsung_galaxy_tab_s_8_4_lte-6435.php. *Apple iPad Air*. http://www.gsmarena.com/apple_ipad_air-5797.php *Acer Iconia Tab B1-A71*. http://www.gsmarena.com/acer_iconia_tab_b1_a71-5239.php *Asus Google Nexus 7*. http://www.gsmarena.com/asus_google_nexus_7-4850.php. Acceso 11 de septiembre del 2014

[13] Wikipedia. *Tableta.* http://es.wikipedia.org/wiki/Tableta_%28computadora%29#Lista_de_sistemas_operativos Acceso 11 de septiembre del 2014

[14] Ableson, Frank. *Introducción al desarrollo de Android.* http://www.ibm.com/developerworks/ssa/library/os-android-devel. Acceso 11 de septiembre del 2014

[15] Meier, Reto. *Professional Android 2 Application Development.* Indianapolis. Editorial Wiley. 2010. Disponible en: http://filepi.com/i/7PXRNd4. Acceso 11 de septiembre del 2014

[16] Catalán, Adrián. *Curso Android: Desarrollo de aplicaciones móviles.* 2011. Disponible en: http://www.etnassoft.com/biblioteca/curso-android/. Acceso 15 de septiembre del 2014

[17] Developer Android. *Application Fundamentals*. http://developer.android.com/guide/components/fundamentals.html. Acceso 22 de septiembre del 2014

[18] Roldayan. *Desarrollo de aplicaciones en Android (II). Herramientas SDK.* http://www.mibqyyo.com/articulos/2013/09/20/desarrollo-de-aplicaciones-en-android-ii-herramientas-sdk/#/vanilla/discussion/embed/?vanilla_discussion_id=0. Acceso 25 de septiembre del 2014

[19] Wikipedia, *App Inventor*. http://es.wikipedia.org/wiki/App_Inventor. Acceso 25 de septiembre del 2014

[20] Developer Android. *Android Studio*. http://developer.android.com/sdk/installing/studio.html. Acceso 25 de septiembre del 2014

[21] Developer Android. *Android NDK*. http://developer.android.com/tools/sdk/ndk/index.html. Acceso 25 de septiembre del 2014

[22] Rodríguez, Txema. *Genymotion, el emulador más rápido de Android. http://www.xatakandroid.com/roms-android/genymotion-el-emulador-mas-rapido-de-android#comments.* Acceso 25 de septiembre del 2014

[23] Definición ABC. *Definición de Geolocalización*. http://www.definicionabc.com/geografia/geolocalizacion.php#ixzz3ENSvbTKN. Acceso 25 de septiembre del 2014

[24] Huerta, Eduardo. Mangiaterra, Aldo. Noguera, Gustavo. *GPS Posicionamiento Satelital*. Rosario, Editoral UNR, 2005. Disponible en: http://www.fceia.unr.edu.ar/gps/GGSR/libro_gps.pdf.

[25] Google. *Sitios Web*. https://support.google.com/a/answer/90915. Acceso 5 de octubre del 2014

[26] Google. *Tu seguridad y privacidad*. https://support.google.com/a/answer/60762?hl=es&ref_topic=29818. Acceso 5 de octubre del 2014

[27] Sierras, Cedric. *¿No sabes qué son las letras G, E, 3G, H y H+? Te lo explicamos*. http://atombit.es/no-sabes-que-son-las-letras-g-e-3g-h-y-h-te-lo-explicamos. Acceso 6 de octubre del 2014

[28] 4g Américas. *LTE-Advanced*. http://www.4gamericas.org/index.cfm?fuseaction=page&pageid=1407. Acceso 6 de octubre del 2014

[29] Kioskea. *WiFi - Redes inalámbricas y seguridad*. http://es.kioskea.net/faq/1160-wifi-redes-inalambricas-y-seguridad. Acceso 6 de octubre del 2014

[30] Wikipedia. *Atracción turística.* Disponible en: http://es.wikipedia.org/wiki/Atracci%C3%B3n_tur%C3%ADstica. Acceso 8 de octubre del 2014

[31] Urrutia, Javier. *Curso de Cartografía y Orientación*. Disponible en: http://www.mendikat.net/descargas/Curso%20de%20Cartogarfia%20y%20Orientacion%20%5Bv1.1%5D%5BPor%20Javier%20Urrutia%5D%5BManuales%20Mendikat%5D.pdf. Acceso 13 de octubre del 2014

ACRÓNIMOS

ADM: Android Debug Monitor, monitoreo de depuración android.

AMOLED: Active Matrix Organic Light Emitting Diode, diodo orgánico emisor de luz con matriz activa.

API: Applications Programmation Interface, interfaz de programación de aplicaciones.

APK: Application Package File, archivo de paquete de aplicación móvil.

App: Aplicación Móvil.

ATD: Android Tools Developer, herramientas de desarrollo Android.

AVD: Android Virtual Devices, dispositivos virtuales Android.

CDMA: Code División Múltiple Access, acceso múltiple por división de código.

CPU: Central Processing Unit, unidad central de procesamiento.

DDMS: Dalvik Debug Monitor Server, servidor de monitoreo de depuración Dalvik.

DEX: Dalvik Executable File, archivo ejecutable Dalvik.

DSP: Digital Signal Processor, procesador de señal digital.

EDGE: Enhanced Data Rates for GSM Evolution, tasas de datos mejoradas para la evolución de GSM.

EV-DO: Evolution-Data Optimized, evolución de datos optimizados.

FM: Frequency Modulaty, frecuencia modulada.

GB: GigaBytes.

Gbps: Gigabits por segundo.

GHz: GigaHertz.

GPRS: General Packet Radio Service, servicio general de paquetes vía radio.

GPS: Global Positioning System, sistema de posicionamiento global.

GSM: Global System Mobile, sistema global de comunicaciones móviles.

HSDPA: High Speed Downlink Packet Access, acceso descendente de paquetes a alta velocidad.

HSUPA: High Speed Uplink Packet Access, acceso ascendente de paquetes a alta velocidad.

HTML: HyperText Markup Language, lenguaje de marcas de hipertexto

HTTPS: Hypertext Transfer Protocol Secure, Protocolo seguro de transferencia de hipertexto.

ID: Identificador.

IDE: Integrated Development Environment, entorno de desarrollo integrado.

IDEN: Integrated Digital Enhanced Network, red mejorada digital integrada.

IEEE: Institute of Electrical and Electronics Engineers, Instituto de Ingenieros Eléctricos y Electrónicos.

iOS: Sistema operativo móvil de Apple.

IPC: Inter-Process Communication, comunicación entre procesos.

ISAE: International Standard on Assurance Engagements, Estándar Internacional sobre los Compromisos de Aseguramiento.

kbps: Kilobits por segundo.

LTE: Long Term Evolution, evolución a largo plazo.

MB: MegaBytes.

Mbps: Megabits por segundo.

MHz: MegaHertz.

MP: Megapixels.

NDK: Native Developer Kit, kit de desarrollo nativo.

NFC: Near Field Comunication, comunicación de corto alcance.

OFDMA: Orthogonal Frequency-Division Multiple Access, acceso múltiple por división de frecuencia ortogonal.

OS: Operaty System, sistema operativo.

PC: Personal Computer, computador personal.

PDA: Personal Digital Assistant, asistente personal digital.

PDF: Portable Document Format, formato de documento portátil.

RAM: Random Access Memory, memoria de acceso aleatorio.

SAS: Statement on Auditing Standards, Declaración de Estándares sobre Auditoría.

SD: Secure Digital, seguro digital.

SDK: Sotfware Developer Kit, kit de desarrollo de software.

SMS: Short Message Service, servicio de mensajes cortos.

SSAE: Statement on Standards for Attestation Engagements, Declaración de Estándares para los Compromisos de Certificación.

SSL: Secure Sockets Layer, capa de conexión segura.

TFT – LCD: Thin Film Transistor - Liquid Crystal Display, pantalla de cristal líquido de transistores de película fina.

TLS: Transport Layer Security, seguridad de la capa de transporte.

UMTS: Universal Mobile Telecommunications System, sistema universal de telecomunicaciones móviles.

URI: Uniform Resource Identifier, identificador de recursos uniforme.

URL: Uniform Resource Locator, localizador de recursos uniforme.

USB: Universal Serial Bus, bus universal en serie.

WCDMA: Wideband Code Division Multiple Access, acceso múltiple por división de código de banda ancha.

Wi-Fi: Wireless Fidelity, conexión inalámbrica.

WiMAX: Worldwide Interoperability for Microwave Access, interoperabilidad mundial para acceso por microondas.

XML: eXtensible Markup Language, lenguaje de marcas extensible.

ANEXOS

Anexo 1: CÓDIGO FUENTE DE LA APP

```
/*********************************
 *                               *
 *     Aplicación "Conoce Tulcán"  *
 *                               *
 *            PUCE               *
 *                               *
 *  Maestría en Redes de Comunicaciones *
 *                               *
 *                               *
 *        Autor: Ing. Juan Carlos Robles Ruano *
 *                               *
 *        Tutor: Ing. Francisco Balarezo *
 *                               *
 *                               *
 *           Quito - 2014        *
 *                               *
 *********************************/
package com.jcrr97.conocetulcan;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.KeyEvent;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;

// Está actividad da la bienvenida a la aplicación y permite su inicio

public class Portada extends Activity {
     private Button ini;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.portada);
    ini = (Button) this.findViewById(R.id.button1);

    // Botón Iniciar: Permite pasar a la actividad de ubicación geográfica
    ini.setOnClickListener(new OnClickListener() {
       public void onClick(View v) {
            Intent a = new Intent(Portada.this, Ubica.class);
            startActivity(a);
           }
         });
     }

    // Tecla return del dispositivo
    public boolean onKeyDown(int keyCode, KeyEvent event) {
    if ((keyCode == KeyEvent.KEYCODE_BACK)) {
        finish();
```

```
        }
    return super.onKeyDown(keyCode, event);
    }

    // Tecla menú del dispositivo
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.portada, menu);
        return true;
        }
    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();
        if (id == R.id.acerc) {
        Intent a = new Intent(Portada.this, Acerca.class);
        startActivity(a);
        }
     return super.onOptionsItemSelected(item);
     }
}


package com.jcrr97.conocetulcan;

import java.io.IOException;
import android.app.Activity;
import android.content.Context;
import android.content.Intent;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.os.Bundle;
import android.util.Log;
import android.view.KeyEvent;
import android.view.Menu;
import android.view.MenuItem;
import android.view.SurfaceHolder;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import android.content.res.Configuration;
import android.hardware.Camera;
import android.view.SurfaceView;

//Está actividad localiza al dispositivo, despliega en pantalla sus coordenadas geográficas
//y permite al usuario pulsar el botón "Mostrar lugares cercanos"

public class Ubica extends Activity implements SurfaceHolder.Callback, LocationListener {
    protected LocationManager locationManager;
    protected LocationListener locationListener;
    protected Context context;
    String[] lug;
    TextView coord, mens, puls;
    String provider;
```

```
Button most;
SurfaceView cameraView;
SurfaceHolder surfaceHolder;
Camera camera;
private static boolean wifiConnected = false;
private static boolean mobileConnected = false;
float lng,lat;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.ubica);
    mens = (TextView) findViewById(R.id.textView1);
    coord = (TextView) findViewById(R.id.textView2);
    puls = (TextView) findViewById(R.id.textView3);
    lug = new String[6];
    cameraView = (SurfaceView) this.findViewById(R.id.preview);
    surfaceHolder = cameraView.getHolder();
    surfaceHolder.addCallback(this);

    //Inicio de ubicación geográfica
    locationManager = (LocationManager) getSystemService(Context.LOCATION_SERVICE);
    locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0, this);

    //Si el GPS no está habilitado
    if (!locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER)){
       Toast toast = Toast.makeText(getApplicationContext(), "Gps Deshabilitado",
       Toast.LENGTH_LONG);
       toast.show();
       }

    //Si existe conexion a internet
    ConnectivityManager connMgr= (ConnectivityManager)getSystemService
    (Context.CONNECTIVITY_SERVICE);
    NetworkInfo activeInfo = connMgr.getActiveNetworkInfo();
    if (activeInfo != null && activeInfo.isConnected()) {
       wifiConnected = activeInfo.getType() == ConnectivityManager.TYPE_WIFI;
       mobileConnected = activeInfo.getType() == ConnectivityManager.TYPE_MOBILE;
       if (wifiConnected) {
          }
       else if(mobileConnected){
                }
       }
     else{
          Toast toast = Toast.makeText(getApplicationContext(), "No hay conexión a  Internet",
          Toast.LENGTH_LONG);
          toast.show();
          }
    }

//Inicio de vista previa en la cámara
public void surfaceChanged(SurfaceHolder holder, int format, int w, int h) {
    camera.startPreview();
     }

public void surfaceCreated(SurfaceHolder holder) {
      camera = Camera.open();
```

```
        try {
            camera.setPreviewDisplay(holder);
            Camera.Parameters parameters = camera.getParameters();
            if (this.getResources().getConfiguration().orientation !=Configuration
                .ORIENTATION_LANDSCAPE) {parameters.set("orientation", "portrait");
                camera.setDisplayOrientation(90);
                }
            camera.setParameters(parameters);
            }
        catch (IOException exception) {
            camera.release();
            }
        }

public void surfaceDestroyed(SurfaceHolder holder) {
    camera.stopPreview();
    camera.release();
    }

@Override
public void onPause () {
    super .onPause () ;
    camera.stopPreview () ;
    }

@Override
public void onDestroy () {
    super .onDestroy () ;
    camera.release () ;
    }

@Override
public void onLocationChanged(Location location) {
    //Latitud y Longitud con cuatro decimales
    lng = (float) (Math.round(location.getLongitude()*10000.0)/10000.0);
    System.out.println("location.getLongitude() (Math.round) : " + lng);
    lat = (float) (Math.round(location.getLatitude()*10000.0)/10000.0);
    System.out.println("location.getLatitude() (Math.round) : " + lat);

    //Determinación de latitud sur y longitud oeste
    if (lng < 0 & lat < 0) {
        mens.setText("Tu ubicación geográfica es:");
        coord.setText("Latitud Sur:"+"\t"+lat+"°"+"\n"+"Longitud Oeste:"+"\t"+lng+"°");
        }

    //Determinación de latitud norte y longitud oeste
    if (lng < 0 & lat > 0) {
        mens.setText("Tu ubicación geográfica es:");
        coord.setText("Latitud Norte:"+"\t"+lat +"°"+"\n"+"Longitud Oeste:"+"\t"+lng+"°");
        }

    //Determinación de latitud sur y longitud este
    if (lng > 0 & lat < 0) {
        mens.setText("Tu ubicación geográfica es:");
        coord.setText("Latitud Sur:"+"\t"+lat+"°"+"\n"+"Longitud Este:"+"\t"+lng+"°");
     }
```

```
//Determinación de latitud norte y longitud este
if (lng > 0 & lat > 0) {
    mens.setText("Tu ubicación geográfica es:");
    coord.setText("Latitud Norte:"+"\t"+lat+"°"+"\n"+"Longitud Este:"+"\t"+lng+"°");
    }

//Si el usuario está cerca al monumento del Civismo Carchense
if ((lng <= -77.7406f & lng >= -77.7417f) & (lat <= 0.7764f & lat >= 0.7753f)) {
    lug[0] = "Monumento al Civismo Carchense";
    lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/monumentosestatuasybustos/
              monumentos-al-civismo-carchense";
    puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
    most = (Button)this.findViewById(R.id.button1);
    most.setOnClickListener(new OnClickListener() {
          public void onClick(View v) {
               //Se envía el nombre del lugar y su url, a la siguiente actividad
               Intent a = new Intent(Ubica.this, Selec.class);
               a.putExtra("lugares", lug);
               startActivity(a);
               }
          });
    }

//Si el usuario está cerca a la estatua del Pupo
else if ((lng <= -77.7427f & lng >= -77.7437f) & (lat <= 0.7799f & lat >= 0.7789f)) {
     lug[0] = "Estatua al Pupo";
     lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/monumentosestatuasybustos/
               estatua-a-el-pupo";
     puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
     most = (Button)findViewById(R.id.button1);
     most.setOnClickListener(new OnClickListener() {
          public void onClick(View v) {
               //Se envía el nombre del lugar y su url, a la siguiente actividad
               Intent a = new Intent(Ubica.this, Selec.class);
               a.putExtra("lugares", lug);
               startActivity(a);
               }
          });
    }

//Si el usuario está cerca al monumento del Guerrero Pasto
else if ((lng <= -77.7393f & lng >= -77.74f) & (lat <= 0.7942f & lat >= 0.7936f)) {
     lug[0] = "Monumento al Guerrero Pasto";
     lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/monumentosestatuasybustos/
               estatua-al-guerrero-pasto";
     puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
     most = (Button)this.findViewById(R.id.button1);
     most.setOnClickListener(new OnClickListener() {
          public void onClick(View v) {
               //Se envía el nombre del lugar y su url, a la siguiente actividad
               Intent a = new Intent(Ubica.this, Selec.class);
               a.putExtra("lugares", lug);
               startActivity(a);
               }
          });
    }
```

```
//Si el usuario está cerca al monumento del Minguero
else if ((lng <= -77.734f & lng >= -77.7352f) & (lat <= 0.798f & lat >= 0.7972f)) {
    lug[0] = "Monumento al Minguero";
    lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/monumentosestatuasybustos/
              estatuas-a-los-mingueros";
    puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
    most = (Button)this.findViewById(R.id.button1);
    most.setOnClickListener(new OnClickListener() {
        public void onClick(View v) {
            //Se envía el nombre del lugar y su url, a la siguiente actividad
            Intent a = new Intent(Ubica.this, Selec.class);
            a.putExtra("lugares", lug);
            startActivity(a);
            }
        });
    }

//Si el usuario está cerca al Buvelar de la Juventud
else if ((lng <= -77.7296f & lng >= -77.7336f) & (lat <= 0.8022f & lat >= 0.8018f)) {
    lug[0] = "Bulevar de la Juventud";
    lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/edificaciones/
              boulevard-de-la-juventud";
    puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
    most = (Button)this.findViewById(R.id.button1);
    most.setOnClickListener(new OnClickListener() {
        public void onClick(View v) {
            //Se envía el nombre del lugar y su url, a la siguiente actividad
            Intent a = new Intent(Ubica.this, Selec.class);
            a.putExtra("lugares", lug);
            startActivity(a);
            }
        });
    }

//Si el usuario está cerca al monumento del 26 de mayo de 1971 y al parque la Laguna
else if ((lng <= -77.7274f & lng >= -77.7294f) & (lat <= 0.8036f & lat >= 0.8011f)) {
    lug[0] = "Monumento al 26 de mayo de 1971";
    lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/monumentosestatuasybustos/
              estatua-al-veintiseis-de-mayo";
    lug[2] = "Parque Recreacional La Laguna";
    lug[3] = "https://sites.google.com/site/conocetulcan/atractivos/parquesyplazoletas/
              parque-recreativo-la-laguna";
    puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
    most = (Button)this.findViewById(R.id.button1);
    most.setOnClickListener(new OnClickListener() {
        public void onClick(View v) {
            //Se envían los nombres de los lugares y sus respectivas url, a la siguiente actividad
            Intent a = new Intent(Ubica.this, Selec.class);
            a.putExtra("lugares", lug);
            startActivity(a);
            }
        });
    }

//Si el usuario está cerca al parque y al busto a Julio Andrade
else if ((lng <= -77.7264f & lng >= -77.7274f) & (lat <= 0.8052f & lat >= 0.8042f)) {
    lug[0] = "Parque Julio Andrade";
```

```
        lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/parquesyplazoletas/
                  parque-julio-andrade";
        lug[2] = "Busto al Gral. Julio Andrade";
        lug[3] = "https://sites.google.com/site/conocetulcan/atractivos/monumentosestatuasybustos/
                  busto-al-general-julio-andrade";
        puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
        most = (Button)this.findViewById(R.id.button1);
        most.setOnClickListener(new OnClickListener() {
              public void onClick(View v) {
                   //Se envían los nombres de los lugares y sus respectivas url, a la siguiente actividad
                   Intent a = new Intent(Ubica.this, Selec.class);
                   a.putExtra("lugares", lug);
                   startActivity(a);
                   }
              });
         }

    //Si el usuario está cerca al monumento del Juego de las Bolas
    else if ((lng <= -77.7244f & lng >= -77.7251f) & (lat <= 0.8046f & lat >= 0.8040f)) {
        lug[0] = "Monumento al Juego de las Bolas";
        lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/monumentosestatuasybustos/
                  monumento-al-juego-tradicional-de-las-bolas";
        puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
        most = (Button)this.findViewById(R.id.button1);
        most.setOnClickListener(new OnClickListener() {
              public void onClick(View v) {
                   //Se envía el nombre del lugar y su url, a la siguiente actividad
                   Intent a = new Intent(Ubica.this, Selec.class);
                   a.putExtra("lugares", lug);
                   startActivity(a);
                   }
              });
        }

    //Si el usuario está cerca a la plazoleta y a la estatua del Padre Carlos de la Vega
    else if ((lng <= -77.7206f & lng >= -77.7219f) & (lat <= 0.8121f & lat >= 0.8111f)) {
        lug[0] = "Plazoleta Padre Carlos de la Vega";
        lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/parquesyplazoletas/
                  plazoleta-padre-carlos-de-la-vega";
        lug[2] = "Estatua al Padre Carlos de la Vega";
        lug[3] = "https://sites.google.com/site/conocetulcan/atractivos/monumentosestatuasybustos/
                  estatua-al-padre-carlos-de";
        puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
        most = (Button)this.findViewById(R.id.button1);
        most.setOnClickListener(new OnClickListener() {
              public void onClick(View v) {
                   //Se envían los nombres de los lugares y sus respectivas url, a la siguiente actividad
                   Intent a = new Intent(Ubica.this, Selec.class);
                   a.putExtra("lugares", lug);
                   startActivity(a);
                   }
             });
       }

    //Si el usuario está cerca a la Catedral, a la plazoleta la Concordia y al busto a Simón Bolívar
    else if ((lng <= -77.7182f & lng >= -77.7185f) & (lat <= 0.8118f & lat >= 0.8109f)) {
        lug[0] = "Catedral de Tulcán";
```

```
        lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/iglesias/la-catedral";
        lug[2] = "Plazoleta la Concordia";
        lug[3] = "https://sites.google.com/site/conocetulcan/atractivos/parquesyplazoletas/
                  plazoleta-la-concordia";
        lug[4] = "Busto a Simón Bolívar";
        lug[5] = "https://sites.google.com/site/conocetulcan/atractivos/monumentosestatuasybustos/
                  busto-a-simon-bolivar";
        puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
        most = (Button)this.findViewById(R.id.button1);
        most.setOnClickListener(new OnClickListener() {
              public void onClick(View v) {
                    //Se envían los nombres de los lugares y sus respectivas url, a la siguiente actividad
                    Intent a = new Intent(Ubica.this, Selec.class);
                    a.putExtra("lugares", lug);
                    startActivity(a);
                    }
             });
        }

    //Si el usuario está cerca al Teatro Lemarie
    else if ((lng <= -77.7181f & lng >= -77.7182f) & (lat <= 0.8125f & lat >= 0.8123f)) {
        lug[0] = "Teatro Municipal Lemarie";
        lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/edificaciones/teatro-lemarie";
        puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
        most = (Button)this.findViewById(R.id.button1);
        most.setOnClickListener(new OnClickListener() {
              public void onClick(View v) {
                    //Se envía el nombre del lugar y su url, a la siguiente actividad
                    Intent a = new Intent(Ubica.this, Selec.class);
                    a.putExtra("lugares", lug);
                    startActivity(a);
                    }
             });
         }

    //Si el usuario está cerca al parque Central y monumento a la Libertad
    else if ((lng <= -77.7167f & lng >= -77.7180f) & (lat <= 0.8126f & lat >= 0.8116f)) {
        lug[0] = "Parque Central";
        lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/parquesyplazoletas/
                  parque-central";
        lug[2] = "Monumento a la Libertad";
        lug[3] = "https://sites.google.com/site/conocetulcan/atractivos/monumentosestatuasybustos/
                  monumento-a-la-libertad";
        puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
        most = (Button)this.findViewById(R.id.button1);
        most.setOnClickListener(new OnClickListener() {
              public void onClick(View v) {
                    //Se envían los nombres de los lugares y sus respectivas url, a la siguiente actividad
                    Intent a = new Intent(Ubica.this, Selec.class);
                    a.putExtra("lugares", lug);
                    startActivity(a);
                    }
             });
       }
    //Si el usuario está cerca al mercado Plaza Central
    else if ((lng <= -77.7155f & lng >= -77.7164f) & (lat <= 0.8145f & lat >= 0.8134f)) {
        lug[0] = "Mercado Plaza Central";
```

```
        lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/edificaciones/
                  mercado-plaza-central";
        puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
        most = (Button)this.findViewById(R.id.button1);
        most.setOnClickListener(new OnClickListener() {
              public void onClick(View v) {
                    //Se envía el nombre del lugar y su url, a la siguiente actividad
                    Intent a = new Intent(Ubica.this, Selec.class);
                    a.putExtra("lugares", lug);
                    startActivity(a);
                    }
              });
        }

    //Si el usuario está cerca a la Iglesia y estatua de San Francisco
    else if ((lng <=- 77.7151f & lng >= -77.7157f) & (lat <= 0.8151f & lat >= 0.8147f)) {
        lug[0] = "Iglesia San Francisco";
        lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/iglesias/san-francisco";
        lug[2] = "Estatua a San Francisco de Asís";
        lug[3] = "https://sites.google.com/site/conocetulcan/atractivos/monumentosestatuasybustos/
                  estatua-a-san-francisco";
        puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
        most = (Button)this.findViewById(R.id.button1);
        most.setOnClickListener(new OnClickListener() {
              public void onClick(View v) {
                    //Se envían los nombres de los lugares y sus respectivas url, a la siguiente actividad
                    Intent a = new Intent(Ubica.this, Selec.class);
                    a.putExtra("lugares", lug);
                    startActivity(a);
                    }
              });
        }

    //Si el usuario está cerca al busto de Juan Montalvo
    else if ((lng <= -77.7114f & lng >= -77.7125f) &(lat <= 0.8123f & lat >= 0.8114f)) {
        lug[0] = "Busto a Juan Montalvo";
        lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/monumentosestatuasybustos/
                  busto-a-juan-montalvo";
        puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
        most = (Button)this.findViewById(R.id.button1);
        most.setOnClickListener(new OnClickListener() {
              public void onClick(View v) {
                    //Se envía el nombre del lugar y su url, a la siguiente actividad
                    Intent a = new Intent(Ubica.this, Selec.class);
                    a.putExtra("lugares", lug);
                    startActivity(a);
                    }
              });
        }

    //Si el usuario está cerca al parque Ayora, monumento a Abdón Calderón e iglesia La Dolorosa
    else if ((lng <= -77.7134f & lng >= -77.7153f) & (lat <= 0.8178f & lat >= 0.8162f)) {
        lug[0] = "Parque Isidro Ayora";
        lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/parquesyplazoletas/
                  parque-isidro-ayora";
        lug[2] = "Monumento a Abdón Calderón";
        lug[3] = "https://sites.google.com/site/conocetulcan/atractivos/monumentosestatuasybustos/
```

```
                monumento-a-abdon-calderon";
        lug[4] = "Iglesia La Dolorosa";
        lug[5] = "https://sites.google.com/site/conocetulcan/atractivos/iglesias/la-dolorosa";
        puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
        most = (Button)this.findViewById(R.id.button1);
        most.setOnClickListener(new OnClickListener() {
            public void onClick(View v) {
                //Se envían los nombres de los lugares y sus respectivas url, a la siguiente actividad
                Intent a = new Intent(Ubica.this, Selec.class);
                a.putExtra("lugares", lug);
                startActivity(a);
                }
            });
        }

    //Si el usuario está cerca a la Casa de la Cultura y museo arqueológico
    else if ((lng <= -77.7165f & lng >= -77.7156f) & (lat <= 0.8182f & lat >= 0.8172f)) {
        lug[0] = "Casa de la Cultura Ecuatoriana";
        lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/edificaciones/
                casa-de-la-cultura-benjamin-carrion-nucleo-del-carchi";
        lug[2] = "Museo Arqueológico y de Arte";
        lug[3] = "https://sites.google.com/site/conocetulcan/atractivos/edificaciones/
                museo-german-bastidas";
        puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
        most = (Button)this.findViewById(R.id.button1);
        most.setOnClickListener(new OnClickListener() {
            public void onClick(View v) {
                //Se envían los nombres de los lugares y sus respectivas url, a la siguiente actividad
                Intent a = new Intent(Ubica.this, Selec.class);
                a.putExtra("lugares", lug);
                startActivity(a);
                }
            });
        }

    //Si el usuario está cerca al cementerio municipal
    else if ((lng <= -77.7134f & lng >= -77.7157f) & (lat <= 0.8208f & lat >= 0.8179f)) {
        lug[0] = "Cementerio Municipal";
        lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/edificaciones/
                cementerio-municipal";
        puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
        most = (Button)this.findViewById(R.id.button1);
        most.setOnClickListener(new OnClickListener() {
            public void onClick(View v) {
                //Se envía el nombre del lugar y su url, a la siguiente actividad
                Intent a = new Intent(Ubica.this, Selec.class);
                a.putExtra("lugares", lug);
                startActivity(a);
                }
            });
        }

    //Si el usuario está cerca al parque infantil Bolivia
    else if ((lng <= -77.7117f & lng >= -77.7124f) & (lat <= 0.8174f & lat >= 0.8165f)) {
        lug[0] = "Parque Infantil Bolivia";
        lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/parquesyplazoletas/
                parque-infantil-bolivia";
```

```
        puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
        most = (Button)this.findViewById(R.id.button1);
        most.setOnClickListener(new OnClickListener() {
              public void onClick(View v) {
                  //Se envía el nombre del lugar y su url, a la siguiente actividad
                  Intent a = new Intent(Ubica.this, Selec.class);
                  a.putExtra("lugares", lug);
                  startActivity(a);
                  }
             });
       }

//Si el usuario está cerca al Complejo Recreacional Los Pastos
else if ((lng <= -77.7014f & lng >= -77.7034f) & (lat <= 0.8388f & lat >= 0.8367f)) {
     lug[0] = "Complejo Recreacional Los Pastos";
     lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/parquesyplazoletas/
               parque-recreativo-los-pastos";
     puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
     most = (Button)this.findViewById(R.id.button1);
     most.setOnClickListener(new OnClickListener() {
           public void onClick(View v) {
                //Se envía el nombre del lugar y su url, a la siguiente actividad
                Intent a = new Intent(Ubica.this, Selec.class);
                a.putExtra("lugares", lug);
                startActivity(a);
                }
           });
      }

//Si el usuario está cerca a la casa de aduana
else if ((lng <= -77.6632f & lng >= -77.6653f) & (lat <= 0.8156f & lat >= 0.8135f)) {
     lug[0] = "Casa de Aduana";
     lug[1] = "https://sites.google.com/site/conocetulcan/atractivos/edificaciones/
               casa-de-aduana";
     lug[2] = "Puente Internacional Rumichaca";
     lug[3] = "https://sites.google.com/site/conocetulcan/atractivos/edificaciones/rumichaca";
     puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
     most = (Button)this.findViewById(R.id.button1);
     most.setOnClickListener(new OnClickListener() {
           public void onClick(View v) {
                //Se envía el nombre del lugar y su url, a la siguiente actividad
                Intent a = new Intent(Ubica.this, Selec.class);
                a.putExtra("lugares", lug);
                startActivity(a);
                }
           });
     }

//Si el usuario está cerca a la Facultad de Sistemas de la PUCE
else if((lng <= -78.4872f & lng >= -78.4971f) & (lat <= -0.2074f & lat >= -0.2125f))          {
     lug[0] = "Escuela Politécnica Nacional";
     lug[1] = "http://www.epn.edu.ec";
     lug[2] = "Universidad Politécnica Salesiana";
     lug[3] = "http://www.ups.edu.ec";
     lug[4] = "Casa de la Cultura Ecuatoriana";
     lug[5] = "http://www.casadelacultura.gob.ec";
     puls.setText("AHORA PUEDES PULSAR EL BOTÓN");
```

```
                most = (Button)this.findViewById(R.id.button1);
                most.setOnClickListener(new OnClickListener() {
                        public void onClick(View v) {
                            //Se envían los nombres de los lugares y sus respectivas url, a la siguiente actividad
                            Intent a = new Intent(Ubica.this, Selec.class);
                            a.putExtra("lugares", lug);
                            startActivity(a);
                            }
                        });
                 }
              else{
                    puls.setText("NO HAY LUGARES CERCANOS");
                   }
              }

        // Tecla return del dispositivo
        public boolean onKeyDown(int keyCode, KeyEvent event) {
              if ((keyCode == KeyEvent.KEYCODE_BACK)) {
                 System.exit(0);
                 }
              return super.onKeyDown(keyCode, event);
              }

        @Override
        public void onProviderDisabled(String provider) {
              Log.d("Latitude","disable");
              }

        @Override
        public void onProviderEnabled(String provider) {
              Log.d("Latitude","enable");
              }

        @Override
        public void onStatusChanged(String provider, int status, Bundle extras) {
              Log.d("Latitude","status");
              }

        // Tecla menú del dispositivo
       public boolean onCreateOptionsMenu(Menu menu) {
       getMenuInflater().inflate(R.menu.portada, menu);
       return true;
              }
       @Override
        public boolean onOptionsItemSelected(MenuItem item) {
              int id = item.getItemId();
              if (id == R.id.acerc) {
                  Intent a = new Intent(Ubica.this, Acerca.class);
                  startActivity(a);
                 }
              return super.onOptionsItemSelected(item);
              }
}
```

```
package com.jcrr97.conocetulcan;

import android.app.Activity;
import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import android.text.SpannableString;
import android.text.style.UnderlineSpan;
import android.view.KeyEvent;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.CheckedTextView;
import android.widget.TextView;

//Está actividad despliega un listado de los lugares cercanos para poder seleccionarlos

public class Selec extends Activity {
	String[] lug;
	String[] lug0;
	String[] lug1;
	String[] lug2;
	CheckedTextView noma;
	CheckedTextView nomb;
	CheckedTextView nomc;
	TextView nom01;
	TextView nom02;
	TextView nom03;
	String url01;
	String url02;
	String url03;

	@Override
	public void onCreate(Bundle savedInstanceState) {
		super.onCreate(savedInstanceState);
		setContentView(R.layout.selec);
		addListenerOnButton();
		nom01 = (TextView)this.findViewById(R.id.checkedTextView1);
		nom02 = (TextView)this.findViewById(R.id.checkedTextView2);
		nom03 = (TextView)this.findViewById(R.id.checkedTextView3);
		lug0 = new String[2];
		lug1 = new String[2];
		lug2 = new String[2];

		//Se obtiene los lugares enviados de la actividad anterior
		Intent men = getIntent();
		String[] lug = men.getStringArrayExtra("lugares");
		nom01.setText(lug[0]);
		nom02.setText(lug[2]);
		nom03.setText(lug[4]);
		url01 = lug[1];
		url02 = lug[3];
		url03 = lug[5];

		//Primer lugar cercano con su nombre y url
		lug0[0] = lug[0];
```

```
        lug0[1] = lug[1];

        //Segundo lugar cercano con su nombre y url
        lug1[0] = lug[2];
        lug1[1] = lug[3];

        //Tercer lugar cercano con su nombre y url
        lug2[0] = lug[4];
        lug2[1] = lug[5];
        }

    public void addListenerOnButton() {
        //Visualización del nombre del primer lugar cercano
        noma = (CheckedTextView) findViewById(R.id.checkedTextView1);
        SpannableString content = new SpannableString("");
        content.setSpan(new UnderlineSpan(), 0, content.length(), 0);
        noma.setText(content);
        noma.setOnClickListener(new OnClickListener() {
                public void onClick(View v) {
                    noma.toggle();

                    //Si el primer lugar es seleccionado se envía su nombre y url a la siguiente actividad
                    if (noma.isChecked()) {
                        Intent a = new Intent(Selec.this, Foto1.class);
                        a.putExtra("url0", lug0);
                        startActivity(a);
                        }
                    }
            });

        //Visualización del nombre del segundo lugar cercano
        nomb = (CheckedTextView) findViewById(R.id.checkedTextView2);
        SpannableString content1 = new SpannableString("");
        content1.setSpan(new UnderlineSpan(), 0, content1.length(), 0);
        nomb.setText(content1);
        nomb.setOnClickListener(new OnClickListener() {
                public void onClick(View v) {
                    nomb.toggle();

                    //Si url02 existe continua, caso contrario no avanza
                    if ("".compareTo(url02) != 0) {

                        //Si el segundo lugar es seleccionado se envía su nombre y url a la siguiente actividad
                        if (nomb.isChecked()) {
                            Intent a = new Intent(Selec.this, Foto2.class);
                            a.putExtra("url1", lug1);
                            startActivity(a);
                            }
                        }
                    }
              });

        //Visualización del nombre del tercer lugar cercano
        nomc = (CheckedTextView) findViewById(R.id.checkedTextView3);
        SpannableString content2 = new SpannableString("");
        content2.setSpan(new UnderlineSpan(), 0, content2.length(), 0);
        nomc.setText(content2);
```

```
            nomc.setOnClickListener(new OnClickListener() {
                    public void onClick(View v) {
                        nomc.toggle();

                        //Si url03 existe continua, caso contrario no avanza
                        if ("".compareTo(url03) != 0) {

                           //Si el tercer lugar es seleccionado se envía su nombre y url a la siguiente actividad
                           if (nomc.isChecked()) {
                               Intent a = new Intent(Selec.this, Foto3.class);
                               a.putExtra("url2", lug2);
                               startActivity(a);
                               }
                            }
                         }
                    });
            }

        //Tecla return del dispositivo
        public boolean onKeyDown(int keyCode, KeyEvent event) {
             if ((keyCode == KeyEvent.KEYCODE_BACK)) {
                System.exit(0);
                }
            return super.onKeyDown(keyCode, event);
            }

        public boolean onCreateOptionsMenu(Menu menu) {
             getMenuInflater().inflate(R.menu.portada, menu);
             return true;
             }

        @Override
         public boolean onOptionsItemSelected(MenuItem item) {
              int id = item.getItemId();
              if (id == R.id.acerc) {
                 Intent a = new Intent(Selec.this, Acerca.class);
                 startActivity(a);
                 }
              return super.onOptionsItemSelected(item);
              }
}


package com.jcrr97.conocetulcan;

import java.io.File;
import android.app.Activity;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Matrix;
import android.graphics.Paint;
import android.graphics.Typeface;
import android.net.Uri;
import android.os.Bundle;
```

```
import android.os.Environment;
import android.provider.MediaStore;
import android.view.KeyEvent;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.View.OnClickListener;
import android.webkit.WebView;
import android.webkit.WebViewClient;
import android.widget.Button;
import android.widget.ImageView;

//Está actividad permite iniciar la cámara del dispositivo para fotografiar el primer lugar cercano
//seleccionado previamente, facilita visualizar en la pantalla la imagen capturada del lugar seleccionado con
//su respectivo nombre y dispone de un botón para obtener mayor información del lugar seleccionado en
//un sitio web creado para dicho servicio

public class Foto1 extends Activity {
    private Button foto;
    private Button info;
    private Button reg;
    private Button regw;
    private Button salw;
    private ImageView img;
    String[] lug0;
    String nom01;
    String url01;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.foto);
        img = (ImageView)this.findViewById(R.id.imageView1);
        foto = (Button) this.findViewById(R.id.button2);
        info = (Button) this.findViewById(R.id.button1);
        reg = (Button) this.findViewById(R.id.button3);

        //Se obtiene el nombre y la url del primer lugar cercano selecionado
        Intent m0 = getIntent();
        String[] lug0 = m0.getStringArrayExtra("url0");
        nom01 = lug0[0];
        url01 = lug0[1];

        //Botón Regresar: Vuelve al listado de lugares cercanos
        reg.setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
                System.exit(0);
                }
            });

        //Botón Tomar foto: Aparece la cámara para fotografiar el lugar cercano selecionado previamente
        foto.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                //Se crea el Intent para llamar a la cámara
                Intent cameraIntent = newIntent(android.provider.MediaStore.
```

```
                                        ACTION_IMAGE_CAPTURE);

            //Se Crea una carpeta en la memoria del dispositivo
            File imagesFolder = new File(Environment.getExternalStorageDirectory(),
                                    "Conoce Tulcán");
            imagesFolder.mkdirs();

            //Se coloca el nombre del lugar seleccionado en la imagen capturada
            File image = new File(imagesFolder, nom01 + ".jpg");
            Uri uriSavedImage = Uri.fromFile(image);

            //Se ordena al Intent para guardar la imagen
            cameraIntent.putExtra(MediaStore.EXTRA_OUTPUT, uriSavedImage);

            //Se lanza la aplicacion de la cámara con retorno (forResult)
            startActivityForResult(cameraIntent, 1);
            }
        });
}

protected void onActivityResult(int requestCode, int resultCode, Intent data) {

            //Se comprueba que la foto ha sido tomada con éxito
            if (requestCode == 1 && resultCode == RESULT_OK) {

                //Se crea un bitmap con la imagen recientemente almacenada en la memoria
                Bitmap bMap = BitmapFactory.decodeFile(Environment.getExternalStorageDirectory()+
                                "/Conoce Tulcán/"+ nom01 +".jpg");

                //Se comprime la imagen y se la rota para visualizarla en la pantalla
                int Height = bMap.getHeight();
                int Width = bMap.getWidth();
                int newHeight = 430;
                int newWidth = 320;
                float scaleWidth = ((float) newWidth) / Width;
                float scaleHeight = ((float) newHeight) / Height;
                Matrix matrix = new Matrix();
                matrix.postScale(scaleWidth, scaleHeight);
                matrix.postRotate(90);
                Bitmap resizedBitmap = Bitmap.createBitmap(bMap, 0, 0, Width, Height, matrix, true);

                //Se escribe en la imagen el nombre del lugar seleccionado
                Canvas canvas = new Canvas(resizedBitmap);
                Paint paint = new Paint();
                paint.setColor(Color.BLUE);
                paint.setTextSize(18);
                paint.setTypeface(Typeface.MONOSPACE);
                canvas.drawText(nom01, 20, 30, paint);

                //Se visualiza la imagen con el nombre del lugar seleccionado
                img.setImageBitmap(resizedBitmap) ;

                //Botón +Info: Proporciona información detallada del lugar seleccionado
                info.setOnClickListener(new View.OnClickListener() {
                    public void onClick(View v) {
                        setContentView(R.layout.web);
                        WebView webView = (WebView) findViewById(R.id.webView1);
```

```
                    salw = (Button) findViewById(R.id.button2);
                    regw = (Button) findViewById(R.id.button1);

                    //Se carga la url del lugar selecionado, en un navegador personalizado
                    webView.loadUrl(url01);
                    webView.setWebViewClient(new WebViewClient() {
                });

                    //Botón Regresar: Regresa al listado de lugares cercanos
                    regw.setOnClickListener(new OnClickListener() {
                        @Override
                        public void onClick(View v) {
                            System.exit(0);
                            }
                        });

                    //Botón Salir: Permite abandonar y cerrar la aplicación
                    salw.setOnClickListener(new OnClickListener() {
                        public void onClick(View v) {
                            startActivity(new Intent(getBaseContext(), Portada.class)
                            .addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP |
                                    Intent.FLAG_ACTIVITY_SINGLE_TOP));
                            Intent a = new Intent(Intent.ACTION_MAIN);
                            a.addCategory(Intent.CATEGORY_HOME);
                            a.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
                            a.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
                            a.putExtra("Salir", true);
                            startActivity(a);
                            finish();
                        }
                    });
                }
            });
        }
    }

public boolean onKeyDown(int keyCode, KeyEvent event) {
    if ((keyCode == KeyEvent.KEYCODE_BACK)) {
        System.exit(0);
        }
    return super.onKeyDown(keyCode, event);
    }

public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.portada, menu);
    return true;
    }

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    int id = item.getItemId();
    if (id == R.id.acerc) {
        Intent a = new Intent(Foto1.this, Acerca.class);
        startActivity(a);
        }
    return super.onOptionsItemSelected(item);
    }
```

```
}


package com.jcrr97.conocetulcan;

import java.io.File;
import android.app.Activity;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Matrix;
import android.graphics.Paint;
import android.graphics.Typeface;
import android.net.Uri;
import android.os.Bundle;
import android.os.Environment;
import android.provider.MediaStore;
import android.view.KeyEvent;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.View.OnClickListener;
import android.webkit.WebView;
import android.webkit.WebViewClient;
import android.widget.Button;
import android.widget.ImageView;

//Está actividad permite iniciar la cámara del dispositivo para fotografiar el segundo lugar cercano
//seleccionado previamente, facilita visualizar en la pantalla la imagen capturada del lugar seleccionado con
//su respectivo nombre y dispone de un botón para obtener mayor información del lugar seleccionado en
//un sitio web creado para dicho servicio

public class Foto2 extends Activity {
	private Button foto;
	private Button info;
	private Button reg;
	private Button regw;
	private Button salw;
	private ImageView img;
	String[] lug1;
	String nom02;
	String url02;

	@Override
	protected void onCreate(Bundle savedInstanceState) {
		super.onCreate(savedInstanceState);
		setContentView(R.layout.foto);
		img = (ImageView)this.findViewById(R.id.imageView1);
		foto = (Button) this.findViewById(R.id.button2);
		info = (Button) this.findViewById(R.id.button1);
		reg = (Button) this.findViewById(R.id.button3);

		//Se obtiene el nombre y la url del segundo lugar cercano selecionado
		Intent m1 = getIntent();
		String[] lug1 = m1.getStringArrayExtra("url1");
```

```
nom02 = lug1[0];
url02 = lug1[1];

//Botón Regresar: Vuelve al listado de lugares cercanos
reg.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View v) {
        System.exit(0);
        }
    });

//Botón Tomar foto: Aparece la cámara para fotografiar el lugar cercano selecionado previamente
foto.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        //Se crea el Intent para llamar a la cámara
        Intent cameraIntent = newIntent(android.provider.MediaStore.
                              ACTION_IMAGE_CAPTURE);

        //Se Crea una carpeta en la memoria del dispositivo
        File imagesFolder = new File(Environment.getExternalStorageDirectory(),
                            "Conoce Tulcán");
        imagesFolder.mkdirs();

        //Se coloca el nombre del lugar seleccionado en la imagen capturada
        File image = new File(imagesFolder, nom02 + ".jpg");
        Uri uriSavedImage = Uri.fromFile(image);

        //Se ordena al Intent para guardar la imagen
        cameraIntent.putExtra(MediaStore.EXTRA_OUTPUT, uriSavedImage);
        //Se lanza la aplicacion de la cámara con retorno (forResult)
        startActivityForResult(cameraIntent, 1);
        }
    });
}

protected void onActivityResult(int requestCode, int resultCode, Intent data) {

        //Se comprueba que la foto ha sido tomada con éxito
        if (requestCode == 1 && resultCode == RESULT_OK) {

            //Se crea un bitmap con la imagen recientemente almacenada en la memoria
            Bitmap bMap = BitmapFactory.decodeFile(Environment.getExternalStorageDirectory()+
                          "/Conoce Tulcán/"+ nom02 +".jpg");

            //Se comprime la imagen y se la rota para visualizarla en la pantalla
            int Height = bMap.getHeight();
            int Width = bMap.getWidth();
            int newHeight = 430;
            int newWidth = 320;
            float scaleWidth = ((float) newWidth) / Width;
            float scaleHeight = ((float) newHeight) / Height;
            Matrix matrix = new Matrix();
            matrix.postScale(scaleWidth, scaleHeight);
            matrix.postRotate(90);
            Bitmap resizedBitmap = Bitmap.createBitmap(bMap, 0, 0, Width, Height, matrix, true);
```

```
                //Se escribe en la imagen el nombre del lugar seleccionado
                Canvas canvas = new Canvas(resizedBitmap);
                Paint paint = new Paint();
                paint.setColor(Color.BLUE);
                paint.setTextSize(18);
                paint.setTypeface(Typeface.MONOSPACE);
                canvas.drawText(nom02, 20, 30, paint);

                //Se visualiza la imagen con el nombre del lugar seleccionado
                img.setImageBitmap(resizedBitmap) ;

                //Botón +Info: Proporciona información detallada del lugar seleccionado
                info.setOnClickListener(new View.OnClickListener() {
                    public void onClick(View v) {
                        setContentView(R.layout.web);
                        WebView webView = (WebView) findViewById(R.id.webView1);
                        salw = (Button) findViewById(R.id.button2);
                        regw = (Button) findViewById(R.id.button1);

                        //Se carga la url del lugar selecionado, en un navegador personalizado
                        webView.loadUrl(url02);
                        webView.setWebViewClient(new WebViewClient() {
                    });

                        //Botón Regresar: Regresa al listado de lugares cercanos
                        regw.setOnClickListener(new OnClickListener() {
                            @Override
                            public void onClick(View v) {
                                System.exit(0);
                                }
                            });

                        //Botón Salir: Permite abandonar y cerrar la aplicación
                        salw.setOnClickListener(new OnClickListener() {
                            public void onClick(View v) {
                                startActivity(new Intent(getBaseContext(), Portada.class)
                                .addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP |
                                        Intent.FLAG_ACTIVITY_SINGLE_TOP));
                                Intent a = new Intent(Intent.ACTION_MAIN);
                                a.addCategory(Intent.CATEGORY_HOME);
                                a.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
                                a.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
                                a.putExtra("Salir", true);
                                startActivity(a);
                                finish();
                            }
                        });
                    }
                });
            }
        }

    public boolean onKeyDown(int keyCode, KeyEvent event) {
        if ((keyCode == KeyEvent.KEYCODE_BACK)) {
            System.exit(0);
            }
```

```
            return super.onKeyDown(keyCode, event);
            }

        public boolean onCreateOptionsMenu(Menu menu) {
            getMenuInflater().inflate(R.menu.portada, menu);
            return true;
            }

        @Override
        public boolean onOptionsItemSelected(MenuItem item) {
            int id = item.getItemId();
            if (id == R.id.acerc) {
                Intent a = new Intent(Foto2.this, Acerca.class);
               startActivity(a);
               }
            return super.onOptionsItemSelected(item);
            }
}


package com.jcrr97.conocetulcan;

import java.io.File;
import android.app.Activity;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Matrix;
import android.graphics.Paint;
import android.graphics.Typeface;
import android.net.Uri;
import android.os.Bundle;
import android.os.Environment;
import android.provider.MediaStore;
import android.view.KeyEvent;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.View.OnClickListener;
import android.webkit.WebView;
import android.webkit.WebViewClient;
import android.widget.Button;
import android.widget.ImageView;

//Está actividad permite iniciar la cámara del dispositivo para fotografiar el tercer lugar cercano
//seleccionado previamente, facilita visualizar en la pantalla la imagen capturada del lugar seleccionado con
//su respectivo nombre y dispone de un botón para obtener mayor información del lugar seleccionado en
//un sitio web creado para dicho servicio

public class Foto3 extends Activity {
    private Button foto;
    private Button info;
    private Button reg;
    private Button regw;
    private Button salw;
```

```
private ImageView img;
String[] lug2;
String nom03;
String url03;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.foto);
    img = (ImageView)this.findViewById(R.id.imageView1);
    foto = (Button) this.findViewById(R.id.button2);
    info = (Button) this.findViewById(R.id.button1);
    reg = (Button) this.findViewById(R.id.button3);

    //Se obtiene el nombre y la url del tercer lugar cercano selecionado
    Intent m2 = getIntent();
    String[] lug2 = m2.getStringArrayExtra("url2");
    nom03 = lug2[0];
    url03 = lug2[1];

    //Botón Regresar: Vuelve al listado de lugares cercanos
    reg.setOnClickListener(new OnClickListener() {
        @Override
        public void onClick(View v) {
            System.exit(0);
            }
        });

    //Botón Tomar foto: Aparece la cámara para fotografiar el lugar cercano selecionado previamente
    foto.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {

            //Se crea el Intent para llamar a la cámara
            Intent cameraIntent = newIntent(android.provider.MediaStore.
                                    ACTION_IMAGE_CAPTURE);

            //Se Crea una carpeta en la memoria del dispositivo
            File imagesFolder = new File(Environment.getExternalStorageDirectory(),
                                "Conoce Tulcán");
            imagesFolder.mkdirs();

            //Se coloca el nombre del lugar seleccionado en la imagen capturada
            File image = new File(imagesFolder, nom03 + ".jpg");
            Uri uriSavedImage = Uri.fromFile(image);

            //Se ordena al Intent para guardar la imagen
            cameraIntent.putExtra(MediaStore.EXTRA_OUTPUT, uriSavedImage);

            //Se lanza la aplicacion de la cámara con retorno (forResult)
            startActivityForResult(cameraIntent, 1);
            }
        });
    }

    protected void onActivityResult(int requestCode, int resultCode, Intent data) {
```

```
//Se comprueba que la foto ha sido tomada con éxito
if (requestCode == 1 && resultCode == RESULT_OK) {

    //Se crea un bitmap con la imagen recientemente almacenada en la memoria
    Bitmap bMap = BitmapFactory.decodeFile(Environment.getExternalStorageDirectory()+
                    "/Conoce Tulcán/"+ nom03 +".jpg");

    //Se comprime la imagen y se la rota para visualizarla en la pantalla
    int Height = bMap.getHeight();
    int Width = bMap.getWidth();
    int newHeight = 430;
    int newWidth = 320;
    float scaleWidth = ((float) newWidth) / Width;
    float scaleHeight = ((float) newHeight) / Height;
    Matrix matrix = new Matrix();
    matrix.postScale(scaleWidth, scaleHeight);
    matrix.postRotate(90);
    Bitmap resizedBitmap = Bitmap.createBitmap(bMap, 0, 0, Width, Height, matrix, true);

    //Se escribe en la imagen el nombre del lugar seleccionado
    Canvas canvas = new Canvas(resizedBitmap);
    Paint paint = new Paint();
    paint.setColor(Color.BLUE);
    paint.setTextSize(18);
    paint.setTypeface(Typeface.MONOSPACE);
    canvas.drawText(nom03, 20, 30, paint);

    //Se visualiza la imagen con el nombre del lugar seleccionado
    img.setImageBitmap(resizedBitmap) ;

    //Botón +Info: Proporciona información detallada del lugar seleccionado
    info.setOnClickListener(new View.OnClickListener() {
        public void onClick(View v) {
            setContentView(R.layout.web);
            WebView webView = (WebView) findViewById(R.id.webView1);
            salw = (Button) findViewById(R.id.button2);
            regw = (Button) findViewById(R.id.button1);

            //Se carga la url del lugar selecionado, en un navegador personalizado
            webView.loadUrl(url03);
            webView.setWebViewClient(new WebViewClient() {
        });

            //Botón Regresar: Regresa al listado de lugares cercanos
            regw.setOnClickListener(new OnClickListener() {
                @Override
                public void onClick(View v) {
                    System.exit(0);
                    }
                });

            //Botón Salir: Permite abandonar y cerrar la aplicación
            salw.setOnClickListener(new OnClickListener() {
                public void onClick(View v) {
                    startActivity(new Intent(getBaseContext(), Portada.class)
                    .addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP |
                                Intent.FLAG_ACTIVITY_SINGLE_TOP));
```

```
                                        Intent a = new Intent(Intent.ACTION_MAIN);
                                        a.addCategory(Intent.CATEGORY_HOME);
                                        a.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
                                        a.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
                                        a.putExtra("Salir", true);
                                        startActivity(a);
                                        finish();
                                    }
                                });
                            }
                        });
                    }
                }

    public boolean onKeyDown(int keyCode, KeyEvent event) {
        if ((keyCode == KeyEvent.KEYCODE_BACK)) {
            System.exit(0);
            }
        return super.onKeyDown(keyCode, event);
        }

    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.portada, menu);
        return true;
        }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();
        if (id == R.id.acerc) {
            Intent a = new Intent(Foto3.this, Acerca.class);
            startActivity(a);
            }
        return super.onOptionsItemSelected(item);
        }
}


package com.jcrr97.conocetulcan;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;

//Esta aplicación despliega una corta información acerca de la app

public class Acerca extends Activity {
    Button reg;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.acerca);
        reg = (Button) this.findViewById(R.id.button1);
        reg.setOnClickListener(new OnClickListener() {
```

```
            //Botón Regresar: Permite volver a la actividad previa
            @Override
             public void onClick(View v) {
                  finish();
                  }
            });
    }
}
```

Anexo 2: MANUAL DE USUARIO

INTRODUCCIÓN

En este documento se describen el objetivo y una información clara y concisa de cómo instalar y utilizar la aplicación móvil “Conoce Tulcán”, para su correcta manipulación.

La aplicación móvil “Conoce Tulcán” fue creada con la finalidad de proporcionar una breve información de los atractivos turísticos urbanos, como: iglesias, parques, plazoletas, monumentos, estatuas, bustos, edificaciones patrimoniales y culturales. Es muy importante consultar este manual, ya que guía al usuario paso a paso en la manipulación de la aplicación. Con el fin de facilitar la comprensión del manual, se incluyen gráficos explicativos.

OBJETIVO DE ESTE MANUAL

El objetivo primordial es ayudar y guiar al usuario a instalar y utilizar la aplicación móvil “Conoce Tulcán”, para lo cual se llevara a cabo las acciones siguientes:

- Explicar cómo descargar e instalar la aplicación.
- Enseñar cómo utilizar la app, mediante una descripción detallada e ilustrada de cada una de las ventanas.

DIRIGIDO A

Este manual está orientado a los turistas que visiten la Ciudad de Tulcán o a las personas interesadas en obtener una breve información de los atractivos turísticos.

Las personas que vayan a manipular la aplicación, deberán utilizar este manual teniendo conocimientos mínimos sobre:

- Manejo elemental de un Smartphone o Tablet.
- Navegación básica de web móvil.

ESPECIFICACIONES TÉCNICAS DEL SMARPHONE O TABLET

- Sistema operativo Android, mínimo Ice Cream Sandwich API 14 hasta KitKat API 19
- Receptor de GPS
- Conexión a Internet
- Cámara posterior
- Espacio menor a 10 MB para la instalación de la app
- Almacenamiento de archivos para las fotografías tomadas

DESCARGA E INSTALACIÓN DE LA APLICACIÓN

Los pasos necesarios que se deben seguir para lograr una correcta instalación de la aplicación móvil “Conoce Tulcán”, se describen a continuación:

1. Con el dispositivo móvil en sus manos, tiene que pulsar sobre el icono “Aplicaciones”, enseguida buscar el icono del navegador web, de la forma que se muestra en la figura 1.

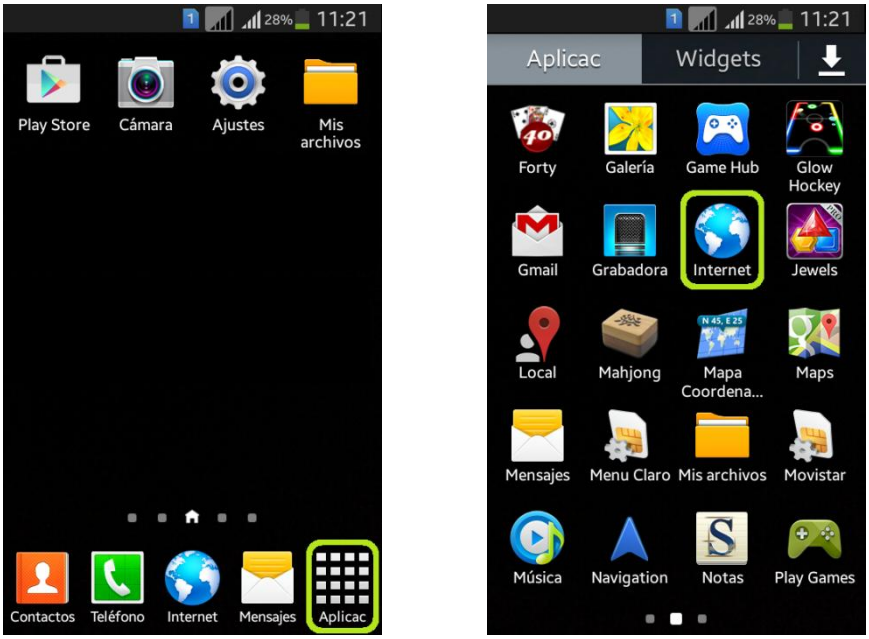


Figura 1. Aplicaciones instaladas en el dispositivo móvil

2. Presionar en el icono “Internet” y luego digitar la URL siguiente “https://sites.google.com/site/conocetulcan/app-android”, abriéndose la página web que se ve en la figura 2.



Figura 2. Página de descarga de la aplicación móvil “Conoce Tulcán”

3. En esta página tiene que encontrarse el archivo “Conoce Tulcán.apk”, debe pulsar sobre la flecha azul para proceder a descargarlo, como se presenta en la figura 3.

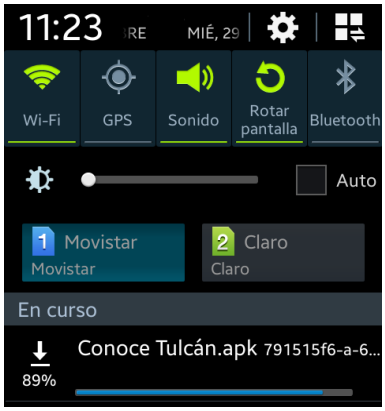


Figura 3. Proceso de la descarga

4. Antes de instalar la app, se debe habilitar las opciones que permiten instalar aplicaciones desconocidas a Google Play, para lo cual se oprime el icono de “Ajustes” o “Configuración” en el dispositivo móvil, a continuación se selecciona “Seguridad” y finalmente se habilita las opciones “Fuentes desconocidas” y “Verificar aplicaciones”, como se indica en la figura 4.

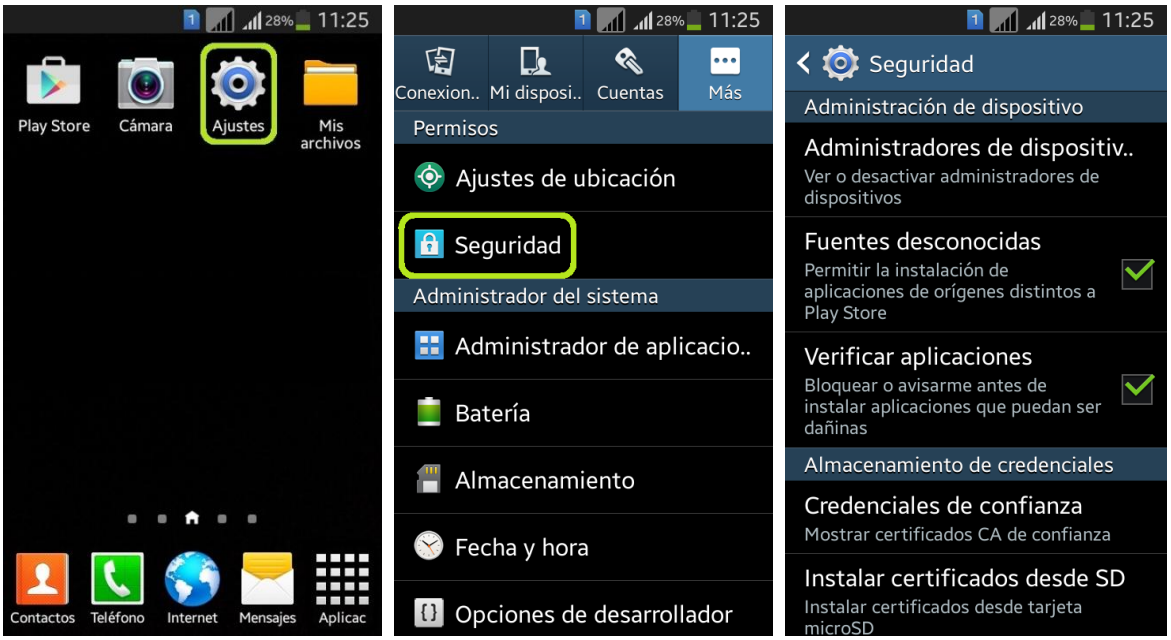


Figura 4. Habilitación de “Fuentes desconocidas” y “Verificar aplicaciones”

5. A continuación tiene que dirigirse al explorador de archivos, para lo cual debe oprimir el icono “Mis archivos”, luego presionar en “Todos los archivos”, enseguida se debe encontrar y pulsar sobre la carpeta “Download”, como se observa en la figura 5.

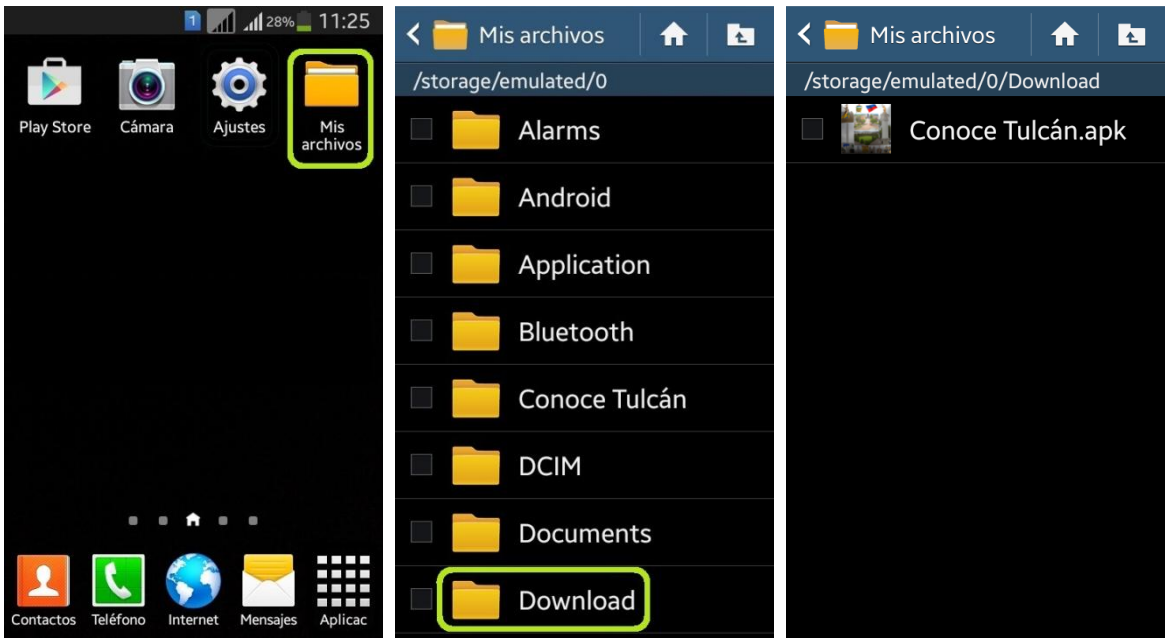


Figura 5. Ubicación del archivo descargado “Conoce Tulcán.apk”

6. En la última ventana se tiene que presionar sobre el archivo “Conoce Tulcán.apk”, luego se elige la opción “Instalador del paquete” y por último se

toca el botón “Siempre” o “Solo una vez” para iniciar el proceso de instalación, como se observa en la figura 6.

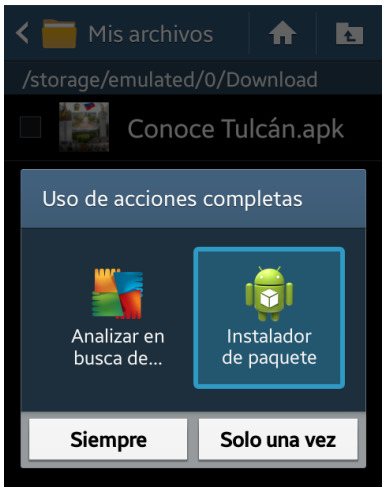


Figura 6. Instalador de paquete

7. Acto seguido aparece una ventana, mostrando el hardware que va a utilizar la app para su funcionamiento, como se ve en la figura 7.

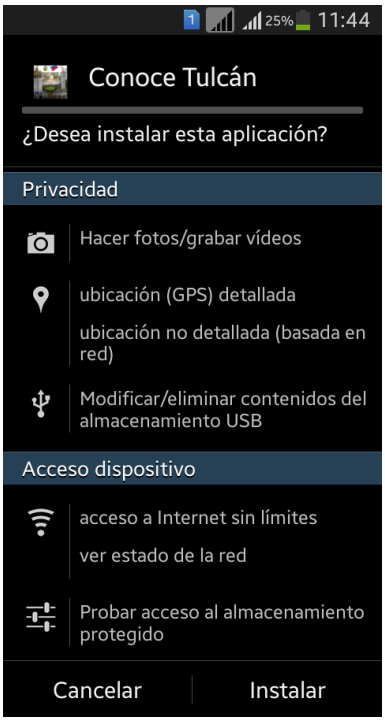


Figura 7. Hardware a utilizar la app

8. Se pulsa el botón “Instalar” apareciendo la ventana que se presenta en la figura 8.

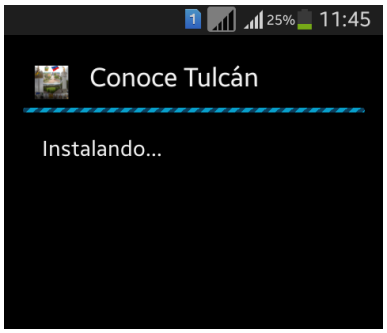


Figura 8. Proceso de instalación

9. Una vez instalada la app se pulsa el botón “Hecho” para finalizar la instalación.

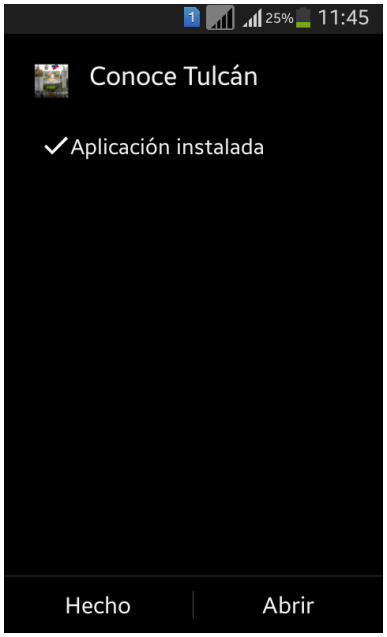


Figura 9. Finalización de la instalación

MANIPULACIÓN DE LA APLICACIÓN

La aplicación móvil “Conoce Tulcán” debe ejecutarse en lugares descubiertos, donde el usuario se encuentre cercano al atractivo turístico para poder fotografiarlo. Los pasos para una adecuada manipulación se detallan enseguida:

1. Para acceder a la app se ingresa a las aplicaciones instaladas en el dispositivo móvil, como se puede contemplar en la figura 10.



Figura 10. Icono de la app

2. Luego se oprime sobre el icono “Conoce Tulcán, abriéndose la ventana de inicio, como se ve en la figura 11. Se debe leer las indicaciones generales.



Figura 11. Ventana de inicio

3. Enseguida se debe activar el GPS y la conexión a Internet. Para lo cual se desliza el dedo desde la parte superior de la pantalla hacia abajo, apareciendo el panel de notificaciones como se indica en la figura 12.

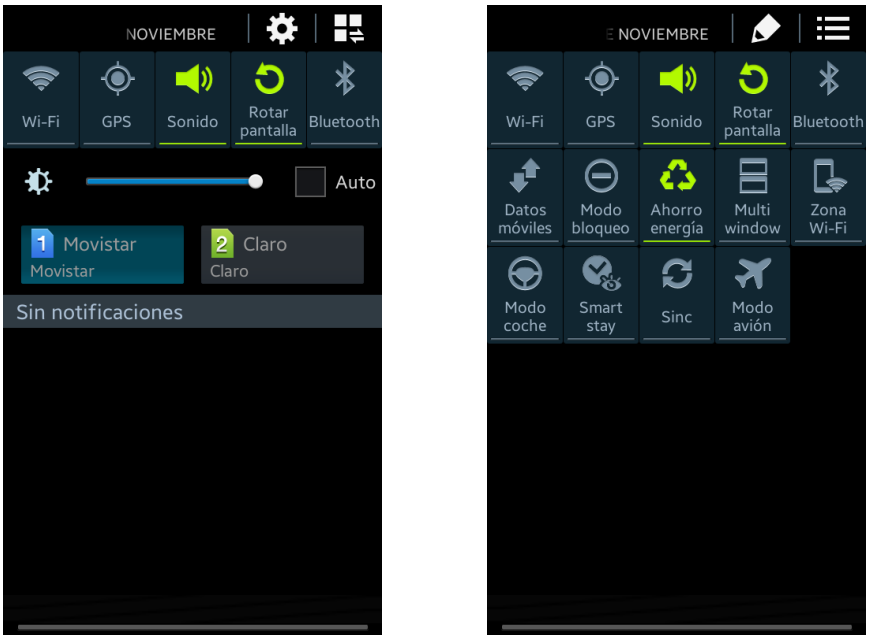


Figura 12. Panel de notificaciones y botones de ajuste rápido

4. A continuación se activan los iconos de “GPS” y “Wi-Fi” o “Datos móviles”, pulsando sobre dichos iconos, como se aprecia en la figura 13.

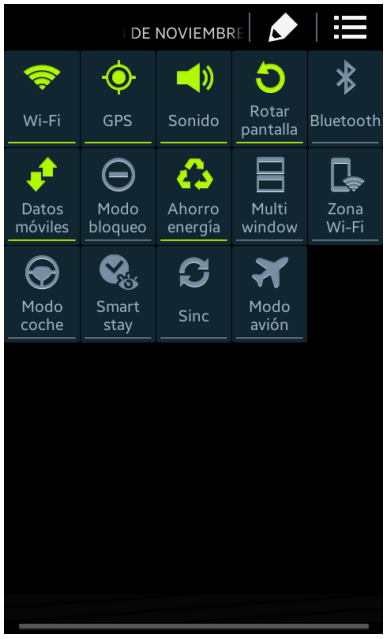


Figura 13. Activación de GPS y conexión a Internet

5. Para volver a la ventana de inicio de la aplicación, se desliza el dedo desde la parte inferior de la pantalla hacia arriba o se oprime tecla “Home” o el botón “Back”.

6. En caso de omitir los pasos 3 y 4, aparecen en la pantalla dos mensajes: “GPS Deshabilitado” y “No hay conexión a Internet”, como se indica en la figura 14.

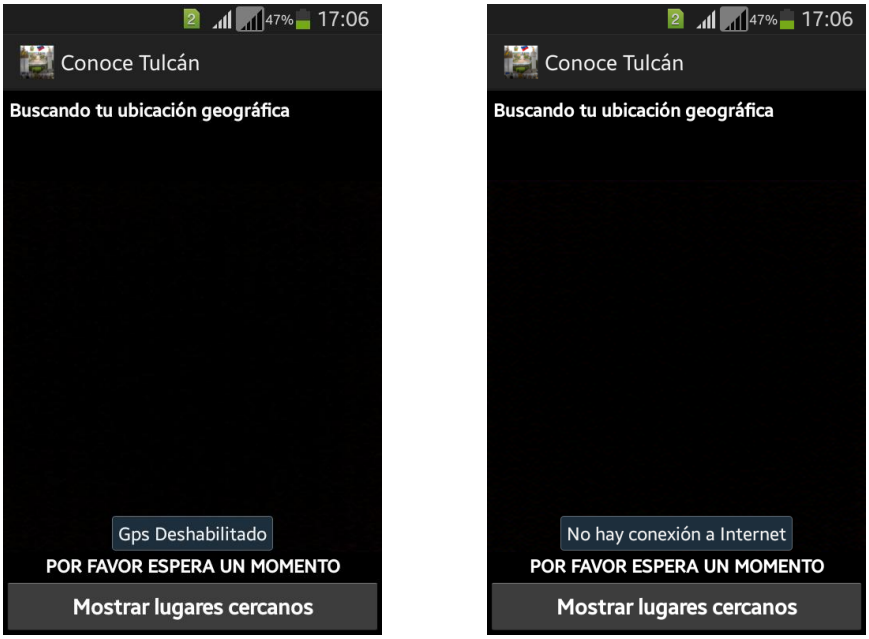


Figura 14. Mensajes que avisan “GPS Deshabilitado” y “No hay conexión a Internet”

7. Ahora se puede pulsar el botón “Iniciar”, la aplicación comienza a buscar la ubicación geográfica del dispositivo móvil, apareciendo la ventana que se muestra en la figura 15.

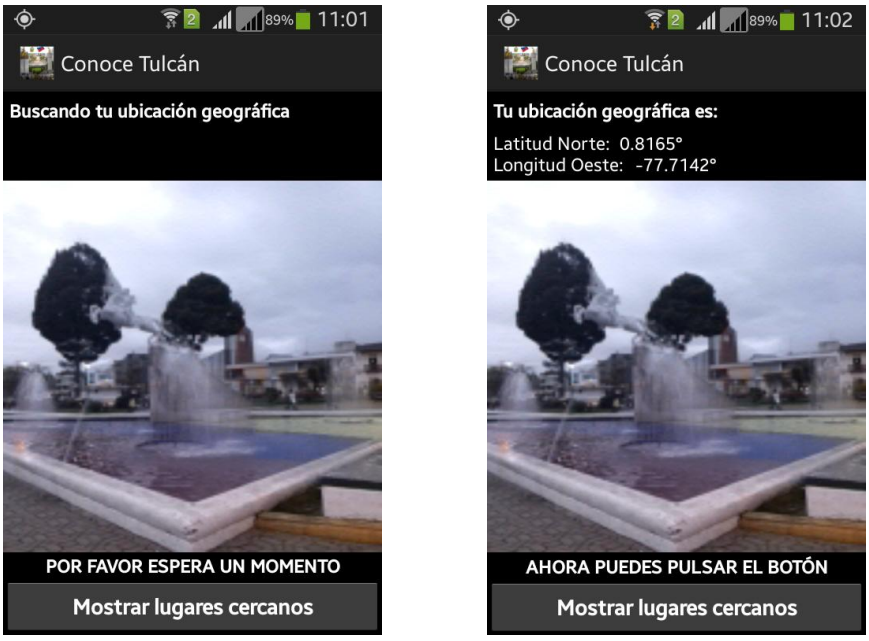


Figura 15. Búsqueda y resultado de la ubicación geográfica

8. Antes de continuar, debe aparecer el mensaje “AHORA PUEDES PULSAR EL BOTÓN”, de ser así, se presiona el botón “Mostrar lugares cercanos”, desplegándose una ventana con la lista de los atractivos turísticos aledaños a la ubicación del usuario, como se puede ver en la figura 16.



Figura 16. Atractivos turísticos cercanos al usuario

9. Acto seguido se elige el atractivo turístico que se desee, pulsando sobre su nombre, pasando a la ventana que se aprecia en la figura 17.

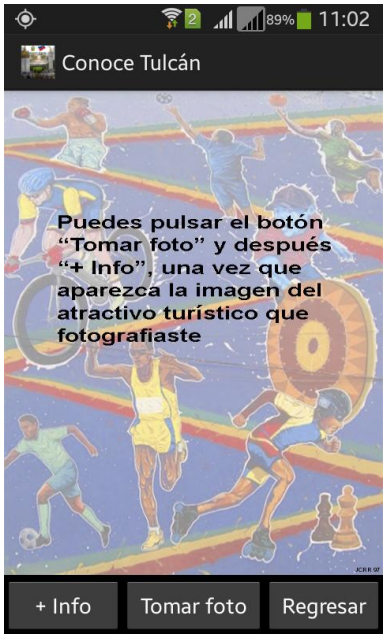


Figura 17. Ventana para tomar una fotografía y mostrarla

10. Para proseguir se oprime el botón “Tomar foto”, abriéndose la cámara del dispositivo, como se muestra en la figura 18.



Figura 18. Enfocando para capturar el atractivo turístico

11. Una vez hecha la captura, aparecerá una ventana que pide “Descartar” o “Guardar” la imagen, como se ve en la figura 19.



Figura 19. Imagen capturada

12. Para continuar se presiona el botón “Guardar”, asomando una ventana con la imagen del atractivo turístico elegido anteriormente con su nombre sobreimpeso, como se puede observar en la figura 20.

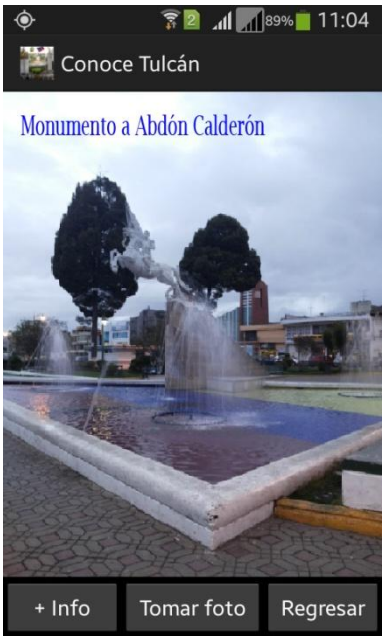


Figura 20. Imagen del atractivo turístico con su nombre sobreimpeso

13. Si se quiere más información del atractivo turístico, en este caso sobre el "Monumento a Abdón Calderón", se pulsa el botón "+ Info", abriéndose el navegador web con una breve información del mencionado monumento, como se presenta en la figura 21.

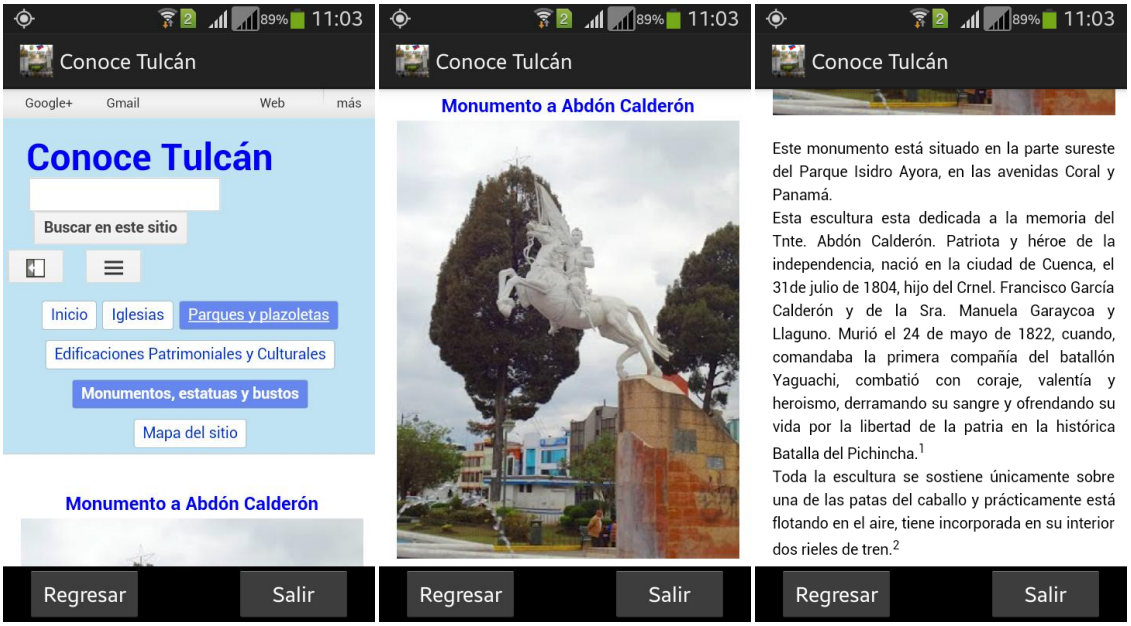


Figura 21. Breve información del "Monumento a Abdón Calderón"

14. En la ventana anterior, al presionar el botón "Regresar", se vuelve a la actividad que exhibe la lista de atractivos turísticos aledaños al usuario, o al oprimir el botón "Salir", se finaliza y se abandona la aplicación.

15. Durante la ejecución de la aplicación, en cualquier ventana se puede pulsar el botón “Menú” del dispositivo para acceder a una corta información de la app, como se puede mirar en la figura 22.



Figura 22. Corta información de la app

En la ventana anterior, al oprimir “Regresar” se vuelve la actividad previa respectiva.

De esta manera se ha explicado claramente la correcta manipulación de la aplicación móvil “Conoce Tulcán”.

UBICACIÓN DE LOS ARCHIVOS DE LAS FOTOGRAFÍAS TOMADAS

Para ubicar la carpeta donde se guardan los archivos de las fotografías tomadas a los atractivos turísticos, se debe ingresar a la carpeta “Mis archivos”, como se indicó anteriormente. Pero en caso se busca la carpeta “Conoce Tulcán”, como se ve en la figura 23.

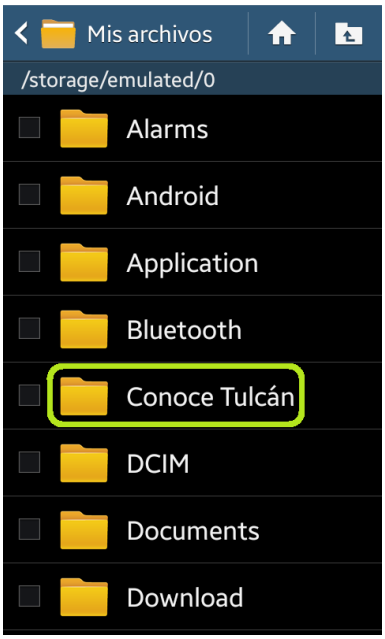


Figura 23. Ubicación de la carpeta “Conoce Tulcán”

Se oprime sobre la carpeta “Conoce Tulcán”, pasando a una ventana con los archivos de las fotografías tomadas a ciertos atractivos turísticos, como se indica en la figura 24. Dicha carpeta se crea automáticamente al momento de la primera ejecución de la aplicación.

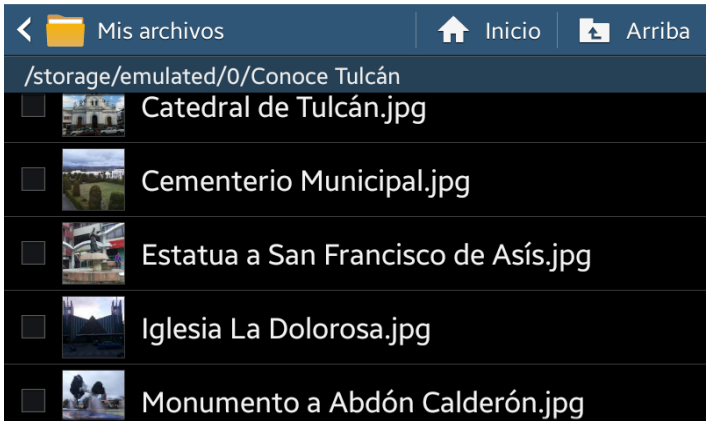


Figura 24. Archivos de las fotografías tomadas a ciertos atractivos turísticos

SOLUCIÓN A POSIBLES PROBLEMAS

1. **No se puede instalar la aplicación móvil**
    - Verificar sí el dispositivo móvil tiene espacio suficiente para la instalación y confirmar si el sistema operativo, es compatible con la aplicación, ver especificaciones técnicas del Smartphone o Tablet .
    - Constatar sí está activada la casilla de “Fuentes desconocidas”, repetir el paso 4 de la sección “Descarga e instalación de la aplicación”.

2. **La aplicación no proporciona ninguna ubicación geográfica**
    - Comprobar sí se encuentran activados el GPS y la conexión a Internet, leer los pasos de la sección “Manipulación de la aplicación”.
    - No debe ejecutarse la aplicación en lugares cubiertos.

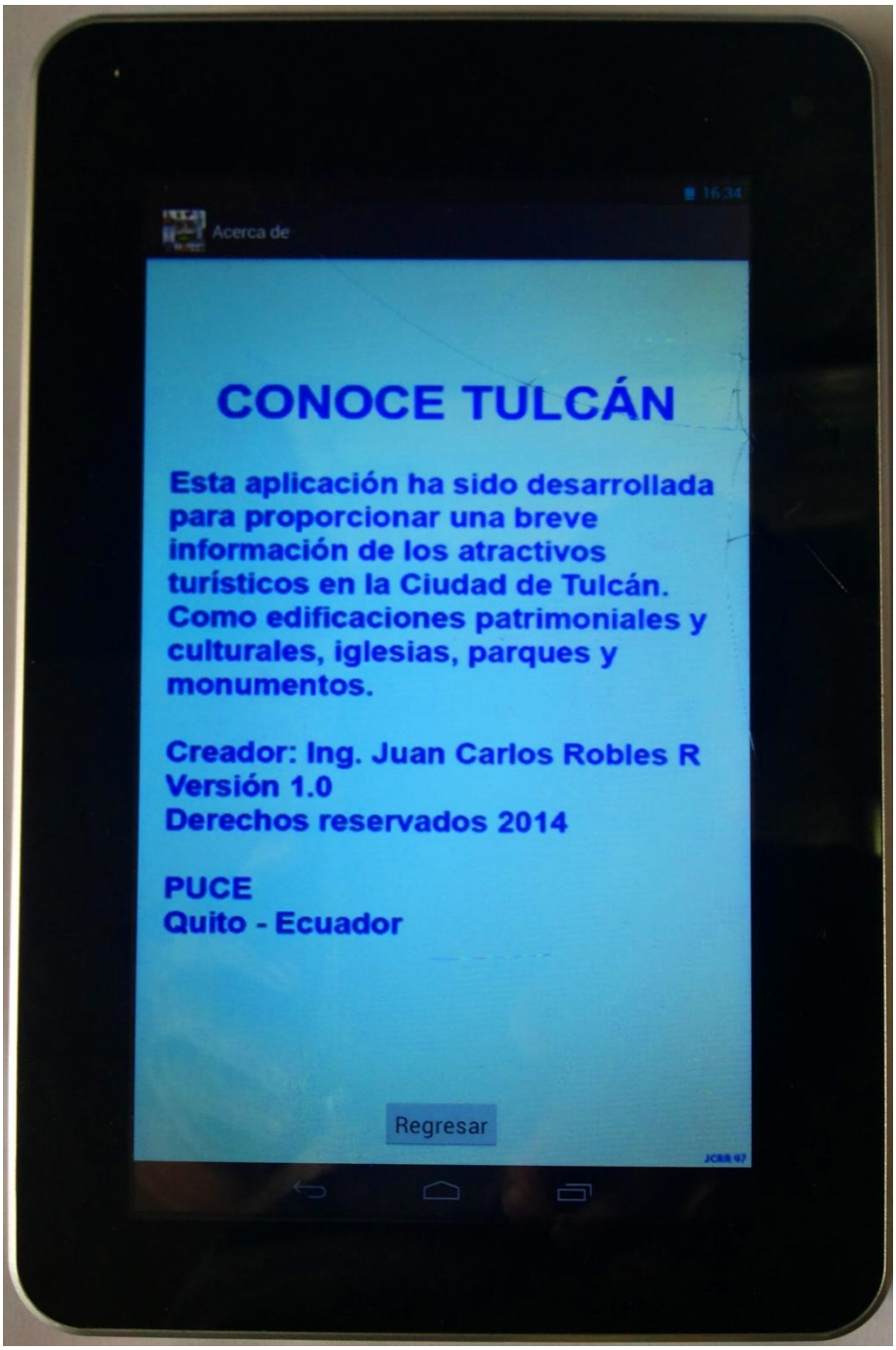
3. **La aplicación da la ubicación geográfica, pero no se puede pulsar el botón “Mostar lugares cercanos”**
    - En la ubicación geográfica donde se encuentra, no existe ningún atractivo turístico cercano o está muy lejos del mismo. Recuerde que debe estar cerca al atractivo turístico para poder fotografiarlo.
    - Diríjase hacia otra ubicación.

4. **No aparece en la pantalla, la imagen del atractivo turístico con su nombre sobreimpreso**
    - No se guardó correctamente la fotografía tomada al atractivo turístico, vuelva a fotografiarlo.
    - Revisar los pasos 9, 10, 11 y 12 de la sección “Manipulación de la aplicación”.

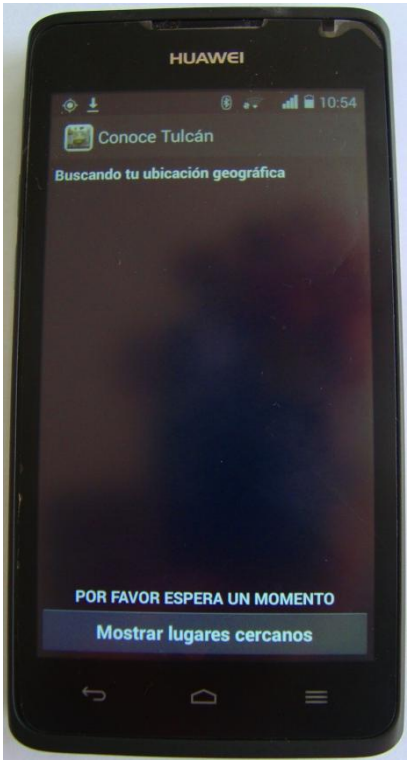
Anexo 3: FOTOGRAFÍAS DE LA EJECUCIÓN DE LA APLICACIÓN “CONOCE TULCÁN” EN ALGUNOS DISPOSITIVOS MÓVILES INTELIGENTES



Tablet Samsung GT-P3100



Tablet Acer B1-710



Smartphone Huawei Y530



Smartphone LG E621g