



## **ESCUELA DE INGENIERÍA EN SISTEMAS**

### **Tema:**

**“DESARROLLO DE APLICACIÓN MÓVIL, CON GEOLOCALIZACIÓN DE LÍNEAS DE AUTOBUSES Y SUS PARADAS PARA EL GOBIERNO AUTÓNOMO DESCENTRALIZADO MUNICIPALIDAD DE AMBATO”**

**Proyecto de investigación y desarrollo de grado previo a la obtención del título de Ingeniero en Sistemas y Computación**

### **Línea de Investigación:**

Sistemas de Información y/o Nuevas Tecnologías de la Información y Comunicación y sus Aplicaciones

### **Autor:**

**EDISON DAVID JARAMILLO ZAMBRANO**

### **Director:**

**Mg. ENRIQUE XAVIER GARCÉS FREIRE**

**Ambato – Ecuador**

**Enero - 2018**

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR AMBATO

HOJA DE APROBACIÓN

Tema:

“DESARROLLO DE APLICACIÓN MÓVIL, CON GEOLOCALIZACIÓN DE LÍNEAS DE AUTOBUSES Y SUS PARADAS PARA EL GOBIERNO AUTÓNOMO DESCENTRALIZADO MUNICIPALIDAD DE AMBATO.”

Línea de Investigación:

SISTEMAS DE INFORMACIÓN Y/O NUEVAS TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN Y SUS APLICACIONES.

Autor:

EDISON DAVID JARAMILLO ZAMBRANO



BIBLIOTECA

Enrique Xavier Garcés Freire, Ing. Mg.

f.

CALIFICADOR

Mena Hernández Liliana del Rocio, Ing. Mg.

f.

CALIFICADOR

Pailiacho Mena Verónica Maribel, Ing. Mg.

f.

CALIFICADORA

González Bucheli Andrea del Carmen, Ing. Mg.

f.

DIRECTORA DE LA ESCUELA

DE INGENIERÍA EN SISTEMAS

Hugo Rogelio Altamirano Villarroel, Dr.

f.



Pontificia Universidad Católica del Ecuador

SECRETARÍA GENERAL  
PROCURADURÍA

SECRETARIO GENERAL PUCESA

Ambato – Ecuador

Enero – 2018

## DECLARACIÓN DE AUTENTICIDAD Y RESPONSABILIDAD

Yo, Edison David Jaramillo Zambrano portador de la cédula de ciudadanía No.- 1804008108 declaro que los resultados obtenidos en la investigación que presento como informe final, previo la obtención del título de INGENIERO DE SISTEMAS Y COMPUTACIÓN son absolutamente originales, auténticos y personales.

En tal virtud, declaro que el contenido, las conclusiones y los efectos legales y académicos que se desprenden del trabajo propuesto de investigación y luego de la redacción de este documento son y serán de mi sola y exclusiva responsabilidad legal y académica.



Edison David Jaramillo Zambrano

C.I: 180400810-8



BIBLIOTECA

## **DEDICATORIA**

El presente trabajo lo quiero dedicar a todas esas personas que estuvieron conmigo apoyándome, especialmente a mis padres, el Sr. Edisson Jaramillo y la Sra. Mónica Zambrano, por ser el pilar fundamental que todo lo que soy, tanto en la educación como en la vida, a mi esposa Elizabeth Hidalgo, que, sin importar las circunstancias, supo darme su apoyo y dedicación incondicional a lo largo de todo este tiempo, a mi hijo Josué Nicolás, quien me motiva a soñar y triunfar en la vida y por último a mis hermanas Ana Jaramillo y Estefanía Jaramillo, ya que han demostrado, y lo siguen haciendo, ser un gran ejemplo a seguir.

Edison David Jaramillo Zambrano

## RECONOCIMIENTOS

Este trabajo de desarrollo se ha construido gracias al apoyo y soporte humano, motivacional y técnico de grandes personas y amigos. Es por ello que quiero extender un sincero agradecimiento al Ing. Gabriel Romero, al Ing. Enrique Garcés, al Ing. David Hidalgo y al Ing. Yu Lung Li quienes con su más sincera amistad y paciencia me han extendido su mano para la culminación de uno de mis sueños más importantes y trascendentales de mi vida, y que sin duda me abre grandes puertas hacia mis futuros triunfos profesionales.

Gracias amigos.

Edison David Jaramillo Zambrano

## RESUMEN

El objetivo principal del actual proyecto es la creación de una aplicación móvil que sirva como herramienta de información sobre el sistema de buses de la ciudad de Ambato para las personas que se encuentren dentro de esta ciudad. Mediante el uso de entrevistas a las autoridades de la Dirección de Tránsito, Transporte y Movilidad de la ciudad de Ambato, se logró obtener información precisa y actualizada, lo que fue de gran importancia para el desarrollo de la aplicación móvil. Esta aplicación fue desarrollada con *Android Studio*, *Visual Basic* y *Oracle DataBase*, lo que permitió implementar todas las funciones necesarias para que la aplicación funcione correctamente. La herramienta Google Maps Android API, permitió la debida gestión del mapa de la ciudad de Ambato, que con la ayuda de la metodología *Mobile-D*, se administró correctamente el tiempo del desarrollo del proyecto. Como resultado se obtuvo una aplicación móvil bajo el sistema operativo Android de viajes y guías única en la ciudad de Ambato, que permite conocer de manera precisa y detallada información de las diferentes rutas de las líneas de buses, además permite conocer la información necesaria de la mejor ruta a tomar para llegar a un lugar determinado.

**Palabras clave:** APIs de Google, aplicación de guías de viajes, desarrollo móvil, metodología Mobile-D, mapas de viajes, sistemas de buses.

## ABSTRACT

The main objective of this project is to develop a mobile application that could be used as an information tool about Ambato's city transportation system for people who come to the city. It was possible to gather precise and updated information thanks to the interviews to the authorities of the Department of Transit, Transportation and Mobility of Ambato, which was of great importance for the development of the mobile application. This application was developed using Android Studio, Visual Basis and Oracle Database, which enable to work with the features that are required for the application to run properly. The tool that allows the management of the maps was Google Maps Android API, that together with Mobile-D methodology enabled to manage the project development plan accordingly. As a result, a mobile application under Android operating system shows a unique travel guide and trips of Ambato city, which allow the user to get detailed and accurate information about bus routes of different lines. Moreover, it allows to get information about the most appropriate route to get to specific place.

**Key words:** Google APIs, travel application guide, mobile development, Mobile-D methodology, travel maps, transportation system.

## ÍNDICE DE CONTENIDO

### Preliminares

DECLARACIÓN DE AUTENTICIDAD Y RESPONSABILIDAD.....	iii
DEDICATORIA .....	iv
RECONOCIMIENTOS .....	v
RESUMEN.....	vi
ABSTRACT.....	vii
INTRODUCCIÓN.....	1
CAPÍTULO I.....	2
FUNDAMENTOS TEÓRICOS.....	2
1.1 Antecedentes.....	2
1.2 Problema .....	3
1.2.1 Descripción del problema.....	3
1.2.2 Preguntas básicas.....	3
1.3 Justificación.....	4
1.4 Objetivos.....	5
1.4.1 General.....	5
1.4.2 Específicos .....	5
1.5 Fundamentos teóricos .....	5
1.5.1 Transporte.....	5
1.5.2 Dispositivos móviles .....	8
1.5.3 Google Maps y API .....	13
1.5.4 Android Estudio .....	15
1.5.5 Base de Datos.....	16
1.5.6 <i>Web Services</i> .....	16
CAPÍTULO II.....	18
METODOLOGÍA .....	18
2.1 Metodología Mobile-D .....	18
2.1.1 Fase de exploración .....	18
2.1.2 Fase de inicialización .....	19
2.1.3 Fase de desarrollo o producto.....	19
2.1.4 Fase de estabilización.....	20
2.1.5 Fase de pruebas y corrección del sistema .....	20

CAPÍTULO III.....	21
RESULTADOS.....	21
3.1 Fase de exploración .....	21
3.1.1 Definición de las partes interesadas.....	21
3.1.2 Definición del alcance .....	22
3.1.3 Establecimiento del proyecto.....	23
3.2 Fase de inicialización .....	23
3.2.1 Preparación del proyecto.....	23
3.2.2 Planificación inicial .....	32
3.2.3 Día de lanzamiento.....	32
3.3 Fase de Desarrollo .....	32
3.3.1 Primera iteración .....	32
3.3.2 Segunda iteración .....	37
3.3.3 Tercera iteración .....	42
3.4 Fase de estabilización .....	70
3.4.1 Día de planificación .....	70
3.4.2 Día de trabajo.....	73
3.4.3 Día de lanzamiento .....	82
3.5 Fase de pruebas y lanzamiento.....	86
3.5.1 Día de planificación .....	86
3.5.2 Día de trabajo.....	90
3.5.3 Día de lanzamiento .....	102
CAPÍTULO IV.....	104
ANÁLISIS Y VALIDACIÓN DE RESULTADOS.....	104
4.1 Análisis de resultados.....	104
4.1.1 Interfaz gráfica de la aplicación.....	104
4.1.2 Desarrollo de la aplicación .....	107
4.2 Validación de resultados .....	109
4.2.1 Encuestas de satisfacción.....	110
CONCLUSIONES Y RECOMENDACIONES .....	112
CONCLUSIONES .....	112
RECOMENDACIONES.....	114
Bibliografía.....	115
ANEXOS.....	118

## ÍNDICE DE GRÁFICOS

### Ilustraciones

Ilustración 2.1 Ciclo de vida Mobile-D.....	18
Ilustración 3.1 Plataformas de Android Studio .....	24
Ilustración 3.2 Herramientas de Android Studio.....	26
Ilustración 3.3 Modo de desarrollador en teléfono Android.....	27
Ilustración 3.4 Activar la depuración por medio de USB .....	28
Ilustración 3.5 Librería ODBC en Visual Basic.....	29
Ilustración 3.6 Interfaz de Oracle SQL Developer.....	30
Ilustración 3.7 Instalación del servidor de Internet Information Services .....	30
Ilustración 3.8 Interfaz del servidor de Internet Information Services.....	31
Ilustración 3.9 Diagrama Entidad Relación .....	35
Ilustración 3.10 Creación de un proyecto web en Visual Studio .....	38
Ilustración 3.11 Creación de un servicio web (ASMX) .....	39
Ilustración 3.12 Conexión de la WS con la Base de Datos Oracle.....	39
Ilustración 3.13 Codificación de las consultas a la base de datos en la WS	40
Ilustración 3.14 Archivos para levantar la WS .....	41
Ilustración 3.15 Archivos dentro de la servidores IIS .....	41
Ilustración 3.16 Consultas de la WS en el IIS .....	41
Ilustración 3.17 Respuesta de las consultas de la WS en internet.....	42
Ilustración 3.18 Maquetado del SplashScreen.....	45
Ilustración 3.19 Diseño del SplashScreen .....	46
Ilustración 3.20 Cambio del Activity principal en el AndroidManifest.xml .....	47
Ilustración 3.21 Codificación en el archivo PantallaInicio.java .....	48
Ilustración 3.22 Resultado del SplashScreen .....	49
Ilustración 3.23 Maquetado de la aplicación .....	51
Ilustración 3.24 Identificación de las partes del maquetado.....	51
Ilustración 3.25 Secciones que se divide el menú principal .....	52
Ilustración 3.26 Diseño del menú principal .....	53
Ilustración 3.27 Google Cloud Platform .....	54
Ilustración 3.28 Restricción de la credencial para el uso de las APIs .....	55
Ilustración 3.29 Credencial para el uso de las APIs.....	55
Ilustración 3.30 Agregar credencia en el Manifest .....	56
Ilustración 3.31 Librería de Google Play Services .....	57
Ilustración 3.32 Servicios de Google Play en la aplicación .....	57
Ilustración 3.33 Open Graphics Library.....	58
Ilustración 3.34 Componente SupportMapFragment .....	59
Ilustración 3.35 Interfaz publica OnMapReadyCallback.....	59
Ilustración 3.36 Método onMapReady .....	60
Ilustración 3.37 Instanciación del fragment.....	60

Ilustración 3.38 Control IU My Location .....	61
Ilustración 3.39 Permisos de ubicación.....	62
Ilustración 3.40 Diseño del menú secundario .....	62
Ilustración 3.41 Tipos de mapas .....	63
Ilustración 3.42 Menú secundario .....	63
Ilustración 3.43 SoapAction .....	64
Ilustración 3.44 NameSpece.....	64
Ilustración 3.45 Method_Name .....	65
Ilustración 3.46 Url.....	65
Ilustración 3.47 Librerías para la gestión del SOAP.....	66
Ilustración 3.48 Resultado de rutas.....	67
Ilustración 3.49 Google Places API for Android .....	68
Ilustración 3.50 Google Maps Directions API.....	69
Ilustración 3.51 Interfaz del Theme Editor .....	74
Ilustración 3.52 Resultado de los nuevos colores de la interfaz.....	74
Ilustración 3.53 Maquetado del nuevo menú principal.....	75
Ilustración 3.54 Diseño del nuevo menú principal.....	76
Ilustración 3.55 Reubicación del botón para limpiar el mapa.....	77
Ilustración 3.56 Guía de usuario .....	77
Ilustración 3.57 Guía de usuario .....	78
Ilustración 3.58 Controles IU zoom y brújula .....	78
Ilustración 3.59 Botones de zoom y brújula .....	79
Ilustración 3.60 Limitación de la ciudad de Ambato en el mapa .....	80
Ilustración 3.61 Limitación del Zoom en el mapa.....	80
Ilustración 3.62 Visualización de la ciudad de Ambato en el mapa.....	81
Ilustración 3.63 Mensaje temporal de la línea a tomar.....	82
Ilustración 3.64 Inhabilitar el TextView origen.....	90
Ilustración 3.65 Mensaje guía en el TextView destino .....	91
Ilustración 3.66 Corrección del mensaje temporal de la línea a tomar.....	92
Ilustración 3.67 Eliminación de la imagen guía para el usuario .....	93
Ilustración 3.68 Cambio de color en la imagen guía para el usuario como en las rutas trazadas en el mapa .....	94
Ilustración 3.71 Añadir la línea a tomar en el marcador.....	95
Ilustración 3.72 Añadir la cooperativa de la línea a tomar en el marcador... 95	95
Ilustración 3.69 Venta informativa con una ruta .....	96
Ilustración 3.70 Venta informativa con dos rutas .....	97
Ilustración 3.73 Creación de la Key Store Path .....	98
Ilustración 3.74 Creación del APK de la aplicación.....	99
Ilustración 3.75 Ficha de Play Store .....	100
Ilustración 3.76 Cargar el APK de la aplicación .....	100
Ilustración 3.77 Clasificación de contenido de la aplicación .....	101
Ilustración 3.78 Precios y distribución de la aplicación .....	101

Ilustración 3.79 Aplicación AmbatoBus lista para descargarse mediante la tienda Play Store.....	102
Ilustración 4.1 Pruebas de la interfaz gráfica en la versión Android 5.1.....	105
Ilustración 4.2 Pruebas de la interfaz gráfica del menú principal en la versión Android 5.1 .....	106
Ilustración 4.3 Pruebas de la interfaz gráfica del menú secundario en la versión Android 5.1 .....	107
Ilustración 4.4 Prueba del resultado de la ruta más óptima que debe tomar el usuario en la versión Android 5.1.....	108
Ilustración 4.5 Prueba de la visualización de la información de cada línea de bus.....	109
Ilustración 4.6 Resultados de la pregunta 1 .....	110
Ilustración 4.7 Resultados de la pregunta 2.....	110
Ilustración 4.8 Resultados de la pregunta 3.....	111
Ilustración 4.9 Resultados de la pregunta 4.....	111

## Tablas

Tabla 3.1 Test de aceptación 1 .....	33
Tabla 3.2 Test de aceptación 2.....	33
Tabla 3.3 Información tentativa para la base de datos .....	34
Tabla 3.4 Test de aceptación 3.....	37
Tabla 3.5 Test de aceptación 4.....	43
Tabla 3.6 Test de aceptación 5.....	43
Tabla 3.7 Test de aceptación 6.....	44
Tabla 3.8 Lista de resumen de deficiencias 1 .....	70
Tabla 3.9 1er Taller post iteración.....	71
Tabla 3.10 Test de aceptación 7.....	72
Tabla 3.11 Test de aceptación 8.....	73
Tabla 3.12 Lista de resumen de deficiencias 2 .....	83
Tabla 3.13 Lista de resumen de deficiencias 3.....	83
Tabla 3.14 Lista de resumen de deficiencias 4.....	84
Tabla 3.15 Lista de resumen de deficiencias 5.....	85
Tabla 3.16 Lista de resumen de deficiencias 6.....	85
Tabla 3.17 2do Taller post iteración.....	86
Tabla 3.18 Test de aceptación 8.....	88
Tabla 3.19 Test de aceptación 9.....	89
Tabla 3.20 Test de aceptación 10.....	89

## **INTRODUCCIÓN**

Este proyecto de desarrollo e investigación tiene como propósito crear e implementar una herramienta digital informativa, representada en una aplicación móvil, sobre líneas, nombres, rutas y paradas, del sistema de buses de la ciudad de Ambato, que mejoren la forma de movilización de las personas que se encuentren en esta ciudad.

En el capítulo I se encuentra toda la información sobre la problemática actual y fundamentación teórica sobre las herramientas tecnológicas que se utilizaron a lo largo del desarrollo del presente proyecto, en el capítulo II y III se describe la metodología y se muestran los resultados de cada fase que integra dicha metodología, en el capítulo IV se evidencian los resultados de las pruebas y encuestas realizados con los usuarios, y finalmente en el último capítulo se muestran las conclusiones y recomendaciones, acompañadas del material bibliográfico y anexos.

# CAPÍTULO I

## FUNDAMENTOS TEÓRICOS

### 1.1 Antecedentes

En los últimos años el desarrollo y la utilización de aplicaciones móviles es cada vez más común en el convivir de las personas, según menciona [1], las aplicaciones móviles son compatibles con todo tipo de interfaz móvil las mismas que pueden ser instalados en *Smartphones*. Estas aplicaciones, se concretan en labores individuales, es decir, pueden direccionarse únicamente al campo profesional o solo al de entretenimiento y dependiendo del tipo de aplicación que necesitará requisitos como conexión a Internet o a una Base de Datos.

Existen aplicaciones en diferentes ámbitos como: informativa, entretenimiento, negocio, entre otras. Se pueden desarrollar aplicaciones basadas en plataformas como Google Maps que de acuerdo a [2] Google Maps es un plano a nivel mundial, la cual proporciona diferentes utilidades y beneficios al usuario para conocer su localización y la de lugares de su interés.

El presente proyecto pretende fusionar la utilidad que proporcionan las aplicaciones móviles con los beneficios que brindan plataformas como Google Maps, y con esto, desarrollar una aplicación móvil, con geolocalización de líneas de autobuses y sus paradas con la ayuda de varios departamentos del Gobierno Autónomo Descentralizado Municipalidad de Ambato y de esta

manera otorgar una herramienta que ayude al usuario a la movilidad por medio de los autobuses pertenecientes a la ciudad de Ambato.

## **1.2 Problema**

### **1.2.1 Descripción del problema**

La ciudad de Ambato no cuenta con una información detallada que permita a los ciudadanos, turistas y/o negociantes un conocimiento detallado sobre las diferentes rutas y paradas de los buses, en consecuencia, se han presentado problemas e inconvenientes que han afectado diariamente a las personas que requieren de estos servicios, además de lo anterior expuesto, existe un desorden en la ciudad por la falta de conocimiento de las propias ubicaciones que tienen las diferentes paradas de buses.

Mediante observación directa, se detecta que existe la necesidad de implementar una herramienta informativa como una aplicación móvil, en la cual, se detalle las rutas de las diferentes líneas de buses y sus respectivas paradas dentro de la ciudad.

### **1.2.2 Preguntas básicas**

**¿Por qué se origina?** Por la falta de información que la ciudadanía y los turistas tiene a respecto de las paradas y rutas de los buses.

**¿Qué lo origina?** La no existencia de una aplicación que muestre las rutas y paradas de los buses en la ciudad de Ambato

### **1.3 Justificación**

Se pudo identificar que las autoridades encargadas de emitir la información sobre las paradas tanto como las rutas de los buses, no han implementado ninguna herramienta adecuada para facilitar la adquisición de dicha información a la ciudadanía en general. Por otro lado, la poca información emitida por dichas autoridades mediante las redes sociales, no son lo bastante claras para dar a conocer a los usuarios en que lugar y que ruta pueden tomar para llegar a su destino final, e incluso las personas no están completamente seguras de que dicha información sea verídica o se encuentre actualizada, ya que constantemente se altera las rutas de los buses por obras realizadas dentro de la ciudad. Por tales motivos las personas que usan los autobuses, se encuentran confusas a la hora de tomar la ruta adecuada.

Tomando como referencia a [3] y [4] menciona que el 50,4% de la población de Tungurahua ocupan celulares inteligentes y de acuerdo a [4] menciona que el 51,3% de la población nacional tienen acceso al internet. Con lo mencionado anteriormente, se llegó a la conclusión de que es factible desarrollar una aplicación móvil para mejorar la movilidad de las personas que se encuentren en la ciudad de Ambato de forma práctica, rápida y segura, y con la seguridad de que toda la información que se muestre en dicha herramienta es verídica y actualizada.

## **1.4 Objetivos**

### **1.4.1 General**

Desarrollar una aplicación móvil, con geolocalización de líneas de autobuses y sus paradas para el Gobierno Autónomo Descentralizado Municipalidad de Ambato.

### **1.4.2 Específicos**

- Fundamentar teórica y metodológicamente sobre el desarrollo de aplicaciones móviles y georreferenciación.
- Analizar los diferentes requerimientos para el desarrollo de la aplicación.
- Diseñar la aplicación móvil, mediante la utilización del API (*Application Programming Interface*) de Google Maps y una herramienta de programación.
- Realizar pruebas de la aplicación móvil para verificación del cumplimiento de su funcionalidad.

## **1.5 Fundamentos teóricos**

### **1.5.1 Transporte**

De acuerdo a [5], se afirma que “transporte es un sistema de medios para poder conducir o transportar personas y cosas de un lugar a otro, lo cual viene dividido en el siguiente punto: transporte local.”

Con un criterio similar, pero más específico, define [6], el transporte traslada, conjuntamente con los visitantes, todo lo necesario para que el turismo se desenvuelva de óptima manera.

Desde la existencia evolutiva de la humanidad y más aún en los tiempos actuales, el transporte es muy importante en el progreso de la misma, pues es necesario para poder trasladar personas, así como cualquier clase de cosas, para de esta manera reducir el tiempo y acortar distancias, de acuerdo al tipo de transporte que se decida utilizar. Conforme a las necesidades que se presenten en las distintas actividades del ser humano, los tipos de transportes varía por su clase y capacidad, aplicando diferentes distribuciones en sus diseños.

- Transporte local o urbano

Como afirma [7], esta clase de transporte, tiene la finalidad de permitir la movilidad a los ciudadanos, con la que puedan acceder a todos los sectores de una ciudad. En los países desarrollados, varios factores actúan en razón del servicio a toda construcción por la red vial y la ramificación de las redes de transporte masivo, que sirven en su mayoría a todos los barrios, aseguran este acceso.

La principal característica del transporte urbano consta en prestar sus servicios solo dentro de una ciudad o de un área determinada. Sus beneficios son varios, ya que ayuda al ciudadano a movilizarse a largas o cortas distancias, ayuda a conservar el medio ambiente y facilita que el tráfico en la ciudad no sea abundante.

- Autobús

Como señala [8], un vehículo cuya función principal es transportar a varias personas durante un determinado lapso de tiempo y distancia, en el Ecuador este tipo de vehículos está bajo la norma NTE INEN 2205 y el reglamento RTE INEN 038, que le sirve para determinar su capacidad en un promedio de 60 pasajeros. Este tipo de transporte es el más usado en América Latina a nivel de transportes públicos.

Los autobuses son un medio de transporte terrestre que permite a cualquier ciudadano movilizarse en cortas o largas distancias, dependiendo de la necesidad que tenga. Como servicio público, debe estar regido por normas, establecidas por los organismos correspondientes, para el debido uso de los mismos.

- Línea de autobús

La idea principal que plantea el autor es que “ruta o línea, entendidas como un conjunto de vías o derechos de vía que sirven a los vehículos de transporte colectivo para su funcionamiento y servicio. El conjunto de rutas y líneas conforman la Red de Transporte Colectivo.” [9]

Las rutas son planificadas estratégicamente por los organismos que rigen en acuerdo con las cooperativas de autobuses para brindar el servicio a la mayor cantidad de ciudadanos. Estas rutas por lo general, atraviesan las ciudades o las rodean, dependiendo de la cantidad poblacional que habita en determinados sectores de la ciudad.

- Parada de buses

Empleando las palabras de [10], las paradas de autobús, son zonas dentro de las rutas de recorrido de los autobuses de transporte público, establecidas de acuerdo a la planificación urbana y a la accesibilidad segura del ascenso y descenso de los pasajeros. Actualmente, estos lugares se centran en ser puntos de recogida de pasajeros.

Una parada de autobús, es un espacio público. Su objetivo principal es proporcionar acceso libre al sistema de transporte público de la ciudad o del sector en donde se encuentre. Por lo general son identificados con la debida señalética, impuesta por la entidad reguladora de tránsito y transporte de la zona y ayuda al mejor ordenamiento y seguridad del ciudadano.

- Entidad reguladora

La entidad reguladora que rige el transporte y movilidad en la ciudad de Ambato es la Dirección de Transporte Terrestre y Movilidad (DTTM), que pertenece al Gobierno Autónomo Descentralizado de la Municipalidad de Ambato (GADMA), cuyas competencias le fueron otorgadas mediante resoluciones de la Agencia Nacional de Tránsito (ANT).

### **1.5.2 Dispositivos móviles**

[11], afirma que “un dispositivo móvil es un procesador con memoria que tiene muchas formas de entrada (teclado, pantalla, botones, etc.), también formas de salida (texto, gráficas, pantalla, vibración, audio, cable). Algunos dispositivos móviles ligados al aprendizaje son las laptops, teléfonos celulares e inteligentes, asistentes personales digitales (*Personal Digital Assistant*,

PDA, por sus siglas en ingles), reproductores de audio portátil, iPods, relojes con conexión, plataforma de juegos, etc. conectados o no a Internet.”

La principal característica de los dispositivos móviles, como su propio nombre lo indica, es la facilidad de movimiento que se le puede emplear. Ofrece varios servicios y en la actualidad son de gran importancia, debido a que contienen diversas funciones que ayudan a resolver problemas del diario vivir de las personas y son de gran ayuda en las instituciones educativas, para el mejoramiento del aprendizaje, por su facilidad de uso y buena interacción con el usuario.

- Sistemas operativos móviles

[12], anunció que, el software esencial de un dispositivo móvil es el sistema operativo, ya que este es capaz de realizar varias funciones como la administración de los recursos para que el funcionamiento del dispositivo sea de manera eficiente, cómoda y sin interrupciones, para que la persona que lo esté utilizando, se sienta satisfecha de los resultados que este le proporciona. Algunas de las empresas de gran desarrollo comercial, muy competitivas en el mercado mundial de los *smartphones* son Windows Mobile de Microsoft, iPhone OS de Apple, Blackberry OS de Research in Motion (RIM) y Android de Google, por otro lado, existe un sin número de Sistemas Operativos de teléfonos inteligentes como Ubuntu Touch, Firefox OS, Symbian, entre otros [1].

Existen gran variedad de sistemas operativos móviles que se utilizan alrededor de todo el mundo, cada uno de ellos con características diferentes y únicas, que les hacen ser adquiridos por varios usuarios, dependiendo de

sus necesidades y de cuán satisfecho se siente con el mismo. Los sistemas operativos móviles son mucho más simples que los sistemas operativos de las computadoras y están orientadas a las conexiones inalámbricas, formatos multimedia móviles y las diversas formas para introducir y recibir información por medio del internet o simplemente por redes de telecomunicación.

- Android

Desde el punto de vista de [13], Android es un conjunto de líneas de programación de código libre, para varios dispositivos móviles y un proyecto de código abierto, es un sistema operativo creado basándose en el sistema operativo Linux, diseñado originalmente para *smartphones*, pero consecutivamente su desarrollo fue empleado para soportar otros dispositivos como *tablets*, reproductores MP3, netbooks, PCs, televisores, lectores e-books, entre otros, éste sistema operativo se pone en marcha basándose a una arquitectura de capas llamada pila de software de Android.

En el presente proyecto, se decidió utilizar el sistema operativo móvil Android, por ser el líder del mercado móvil, esto lleva que, la mayoría de los ciudadanos, poseen un Smartphone o un celular inteligente con sistema operativo Android. Una de las características principales, es la facilidad de adquisición de las aplicaciones, pues la mayoría de ellas son gratuitas, útiles y de buena calidad.

- Lenguaje de programación (IDE)

Tal como [14], un lenguaje de programación surge a raíz del hecho de contar con un código especial para introducir datos, operaciones y procesos en la computadora.

Parafraseando a [15], se dice que, cada ordenador, según su diseño, *entiende* un determinado conjunto de instrucciones elementales (lenguaje máquina). Sin embargo, para agilizar el desempeño del programador, se dispone también de un lenguaje de programación superior, que resulta de mayor confiabilidad y manejo, sin depender de algún diseño específico por cada ordenador. Los programas escritos en un lenguaje de programación superior, no podrán ser ejecutados por un ordenador, mientras no sean traducidos al lenguaje propio de este.

La definición de un lenguaje de programación necesita la especificación de un conjunto de símbolos y palabras claves utilizables, tomando en consideración reglas gramaticales sintácticas y semánticas, con el fin de construir sentencias o procesos de una forma correcta, de acuerdo a las normativas con que trabaje el lenguaje de programación.

Un lenguaje de programación es el conjunto de líneas de código que pueden ser palabras claves y símbolos, de esta manera se forma la base fundamental de cualquier programa o sistema operativo que existe en la actualidad, ya que, en él se realizan operaciones y procesos para realizar funciones específicas para el usuario final. Cada lenguaje de programación, tiene su propio compilador, cuya función es entender o traducir el conjunto de líneas de

código hacia otro lenguaje de programación para que la máquina sea capaz de interpretar.

- Software libre

Desde el punto de vista conceptual de [16] , software libre es cualquier programa que permita a los usuarios disfrutar de libertades de uso, es decir que pueden libremente redistribuir copias modificadas o no, de forma gratuita o a la vez, cobrando por su distribución, indistintamente de alguna persona o lugar. Tener esta libertad, representa la potestad de actuar autónomamente sin permisos ni pagos por el servicio. De igual modo, esta libertad se tiene que dar cuando el usuario introduzca sus modificaciones y las utilice con toda privacidad, sea cual fuese su aplicación privada, tanto en el campo laboral como en el personal. Dichas modificaciones no están obligadas a ser notificadas hacia otras entidades.

La principal razón, para utilizar software libre, se enfoca en que los usuarios puedan acceder a él sin ningún tipo de restricción ya que este proyecto va enfocado a facilitar el reconocimiento de las diferentes rutas de autobuses en la ciudad, sin restricción alguna para los usuarios finales.

- Aplicaciones móviles

De acuerdo con [1], las aplicaciones móviles se pueden adaptar a la interfaz de un celular y que pueden ser instalados en dispositivos inteligentes *Smartphone*, se encuentran generalmente disponibles y con libre acceso en plataformas de distribución propietarias de los Sistemas Operativos móviles. Estas aplicaciones, se concretan en labores individuales, es decir, pueden

direccionarse únicamente al campo profesional o solo al de entretenimiento, pero que requieren necesariamente de Internet o una conexión a Base de Datos.

Las aplicaciones móviles son muy productivas, porque proporcionan muchas ventajas, la principal es que se pueden ejecutar en teléfonos inteligentes, tabletas y otros dispositivos móviles, para que el usuario utilice este beneficio en cualquier lugar. Las plataformas en donde se encuentran alojadas, pertenecen a los propietarios de los sistemas operativos móviles como Android, iOS, BlackBerry OS, Windows Phone, entre otros.

Permite, además, acceder en forma rápida y fácil a la información proporcionada desde la aplicación, sin tener que introducir datos de autenticación cada vez que se acceda a la misma.

### **1.5.3 Google Maps y API**

- Google Maps

Según lo mencionado en [2] Google Maps es un plano a nivel mundial, cuya función principal es proporcionar al usuario la localización de lugares que necesite, sin importar el espacio o el tiempo en el que se encuentre. La aplicación es tan efectiva, que basta con introducir en el buscador el nombre de una avenida, una dirección completa o un lugar turístico, para que esta aplicación se lo indique inmediatamente, utilizando vistas callejeras o satelitales que, al pulsar en dichos puntos, se despliegan bastas informaciones relacionadas a la búsqueda, con información adicional sobre el lugar o las imágenes reportadas por los usuarios.

“Este despliegue de la información, proporciona al usuario visualizar los resultados de tres maneras diferentes:

**Vista de callejero:** Muestra una representación del territorio donde se ha realizado la búsqueda como si de un callejero se tratase, utilizando representaciones sencillas de los elementos propios del entorno urbano y ofreciendo datos también básicos para la identificación del espacio, como los nombres de las calles y de los edificios más representativos.

**Vista de Satélite:** son fotografías tomadas de la Tierra desde un satélite que Google Maps elabora y coloca para generar una representación de la superficie terrestre. Aquí es como si el usuario sobrevolara la zona adyacente al punto que ha buscado y pudiera ver su aspecto como si lo estuviera observando desde el cielo.

**Street View:** Este sistema permite a los usuarios caminar virtualmente por el mapa, recorrer las calles circundantes al punto que ha seleccionado” [2].

Google Maps, no solo ofrece la facilidad de mostrar los mapas de diferentes sitios del mundo, ofrece también diversas herramientas que permiten montar mapas de cualquier zona del planeta y editarlas conforme a las necesidades del usuario, en diferentes plataformas móviles.

Google Maps es una aplicación tanto web como móvil, que indica el mapa de la zona en donde el usuario se encuentre o de los diversos lugares que requiera visitar o conocer como iglesias, restaurantes, lugares turísticos, hoteles, centros comerciales, hospitales, entre otros. Con la ayuda de sus diferentes tipos de vistas, facilitan el mejor reconocimiento de los lugares y

ubicación de los mismo para evitar posibles pérdidas de tiempo y recursos tantos humanos como económicos.

- API Google Maps

Con referencia a [17], Google Maps Android API, permite insertar diversos mapas a las aplicaciones de Android con características únicas como mapas básicos que visualicen edificios en 3D y planos para pisos de interiores, a más de sus presentaciones básicas. Otro beneficio es, la inserción de imágenes de Street View y satelitales y en cada una de las mencionadas se puede agregar marcadores personalizados, ventanas de información y polilíneas. La última versión de esta API está diseñada para proporcionar a todos sus usuarios una velocidad mayor y que su aplicación sea más fácil en los dispositivos móviles.

- Geolocalización

De acuerdo con [18], este sistema está desarrollado con la finalidad de ubicar geográficamente un dispositivo de usuario o de computación a través de algunos mecanismos de recolección de datos. La mayoría de estos sistemas emplean direcciones de enrutamiento de red o dispositivos GPS interno para determinar una ubicación determinada. No todos los navegadores ni dispositivos, son compatibles con el sistema de geolocalización, situación que limita dicho servicio al usuario.

#### **1.5.4 Android Estudio**

Como señala [19], dentro de los lenguajes de programación para el desarrollo de aplicaciones móviles se encuentra Android Studio, que facilita

herramientas más rápidas para la creación de aplicaciones en los dispositivos que trabajen bajo este sistema operativo. Ofrece una variedad de características que optimizan su desarrollo en el diseño e implementación de sus productos de alta calidad.

### **1.5.5 Base de Datos**

Según lo manifiesta [20], una base de datos es una agrupación de datos estáticos que las empresas utilizan mediante plataformas especializadas para leer este tipo de información.

Por otro lado, como lo menciona en [21], una base de datos se define como una herramienta que permite organizar y recopilar todo tipo de información tal como datos de personas, productos, pedidos, entre otros. En toda base de datos existen redundancias e inconsistencias y esto se debe a que se comienza a tener un incremento de datos los cuales se encuentran almacenados en repositorios no adecuados. Con el transcurso del tiempo se torna difícil la gestión de dicha herramienta, por lo cual se recomienda transferir la información a una base de datos creada con un sistema de administración de base de datos (DBMS).

### **1.5.6 Web Services**

Según [22], una de las nuevas generaciones de aplicaciones web son los llamados *Web Services* cuyas características principales se basan en que son mecanismos de *software* autocontenidas, autodescriptivas y modulares en las cuales se tiene la posibilidad de realizar acciones como acceso, localización e invocación desde cualquier sitio sobre el Internet. Los estándares que ayudan a construir una *Web Services* son:

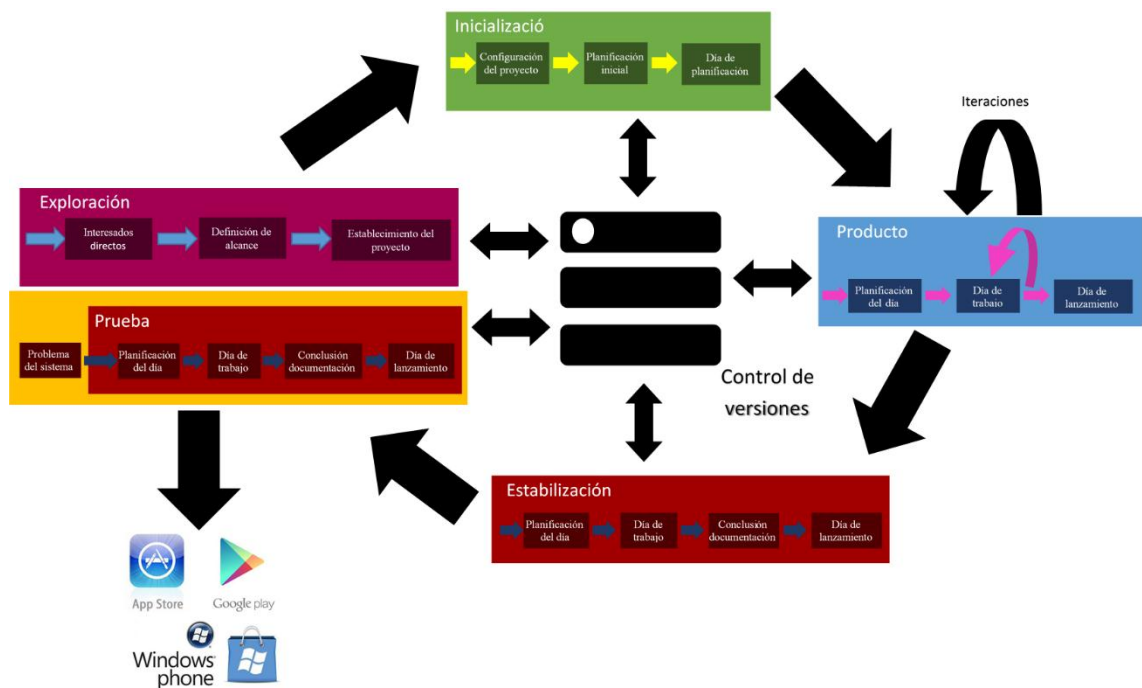
- **UDDI (*Universal Description, Discovery and Integration*)**: permite publicar y localizarlas.
- **WSDL (*Web Service Description Language*)**: sirve para describirlas.
- **SOAP (*Simple Object Access Protocol*)**: sirve para invocarlos sobre HTTP.

Por último, las respuestas que se obtienen de las consultas realizadas a la *Web Services* son presentadas a través de XML (*extensible markup language*) que es un lenguaje de descripción de datos.

## CAPÍTULO II METODOLOGÍA

### 2.1 Metodología Mobile-D

Según lo expresado en [23], la metodología consta de 5 fases que se muestran a continuación:



**Ilustración 2.1** Ciclo de vida Mobile-D  
**Fuente:** elaboración propia a partir de (Jean Williams)

#### 2.1.1 Fase de exploración

Como lo mencionan en [24], el objetivo principal de esta fase es el levantamiento de toda la información necesaria mediante la comunicación con las personas interesadas, para concretar los objetivos del proyecto como la fecha de entrega del mismo, definir todos los recursos materiales y humanos,

de esta manera se asegura que el proyecto inicie sin ningún tipo de retrasos por falta de equipos ni de conocimientos.

El levantamiento de información se la puede realizar mediante diferentes tipos de recolección de información como: reuniones, entrevista, encuestas, entre otras.

### **2.1.2 Fase de inicialización**

Con referente a lo expresado en [25], en esta fase se asegura el éxito del proyecto, ya que se realiza un estudio sobre todos los datos obtenidos en la fase anterior, para que de esta manera se escoja de manera correcta, las herramientas adecuadas para proceder a instalar, configurar y probar los ambientes tanto físicos como técnicos, los recursos que contribuyan en el proceso de desarrollo de la aplicación tomando en cuenta las correctas librerías a utilizar, sin dejar a un lado la verificación de la compatibilidad entre software y hardware que se necesitan para la elaboración de la misma.

### **2.1.3 Fase de desarrollo o producto**

Durante esta fase, como lo expresa el autor [26], se implementa la funcionalidad de la aplicación, tomando en cuenta los requerimientos del cliente, y con la utilización de ciclos de desarrollo iterativos e incrementales, permitir la mejora continua del producto.

El desarrollo de esta fase se lo realiza mediante etapas representadas en día de planificación en donde se ejecuta los test de aceptación, día de trabajo en donde se desarrolla el proyecto con su debida documentación y por último el

día de entrega, en el cual se procede al llenado de la lista de resumen de deficiencias.

#### **2.1.4 Fase de estabilización**

Según [27], en esta etapa se asegura la calidad de la implementación o unión de todos los módulos del proyecto con el objetivo de obtener un solo producto. Para ello, se lo realiza de la misma manera que la fase de producción, basándose en el desarrollo mediante las 3 etapas: día de planificación, día de trabajo y día de entrega.

La documentación en esta fase es similar a la fase de desarrollo con la diferencia que en el día de planificación se debe llenar el primer taller post iteración, cuyo objetivo es documentar la resolución de las deficiencias obtenidas en la iteración anterior.

#### **2.1.5 Fase de pruebas y corrección del sistema**

En esta última fase, se comienza una exploración de deficiencias en el software. Se realiza la planificación, trabajo y entrega del software completo.

De igual manera que la fase anterior, se realiza el un taller post iteración cuyo objetivo es corregir las últimas deficiencias para que el producto cumpla correctamente con las funciones principales que el cliente las estableció al inicio del proyecto [23].

## **CAPÍTULO III**

### **RESULTADOS**

#### **3.1 Fase de exploración**

##### **3.1.1 Definición de las partes interesadas**

Para el desarrollo del actual proyecto se establece una parte o usuario interesado:

- Gobierno Autónomo Descentralizado Municipalidad de Ambato: al ser la entidad encargada del mejoramiento continuo de la ciudad de Ambato, es la parte que muestra más interés en el proyecto, ya que la información necesaria para realizar el mismo, fue impartida por La Dirección de Tránsito, Transporte y Movilidad (DTTM) perteneciente al GADMA basándose a los requerimientos iniciales.

Para ellos se realiza una reunión, cuyo objetivo principal es dar a conocer la idea del proyecto de investigación y solicitar la información necesaria para poner en marcha el desarrollo del mismo. En una segunda reunión se recibe toda la información oficial requerida.

Las personas involucradas en dicha reunión son el Ing. Edgar Fabián Muyulema Galarza, director del Departamento de Tecnologías y Comunicación, Bolívar Xavier Paredes Calero, técnico informático y Katerine Alejandra Latorre Hernández, técnica de la unidad de Gestión de Tránsito, Transporte y Seguridad Vial.

- Usuarios de la ciudad de Ambato: son los ciudadanos o turistas que se encuentran en la ciudad de Ambato que carecen de información sobre las paradas y rutas de los buses que existen dentro de la zona urbana de la ciudad.

### 3.1.2 Definición del alcance

Las limitaciones del sistema son:

- Los datos de la aplicación están almacenados en una base de datos Oracle
- La conexión entre la aplicación y la base de datos es por medio de un *Web Service*.
- La aplicación es desarrollada para el sistema operativo Android.
- Tiene la opción de geolocalización de las paradas de buses como del usuario que esté utilizando la aplicación.
- Cada una de las paradas muestran su propio nombre que los identifica.
- La aplicación está almacenada en la tienda de descarga de aplicaciones que ofrece Google llamada Play Store.
- Se indica cada una de las rutas de líneas de buses con sus respectivos nombres y paradas.
- La aplicación funciona con acceso a internet.
- Tienes datos solo referentes a la zona urbana y rural de la ciudad de Ambato.
- Establece la ruta más óptima de acuerdo a los posibles destinos que el usuario final desee movilizarse mediante este medio de transporte.

### 3.1.3 Establecimiento del proyecto

Después de un estudio de los requerimientos necesarios para crear la aplicación, se procedió a realizar entregables en un cierto periodo de tiempo con su debida documentación:

- Creación de la base de datos en la plataforma Oracle.
- Creación de un Web Service en Visual Studio 2015.
- Creación de la aplicación en el entorno de *Android Studio* con su respectiva configuración y librerías necesarias.
- Incorporación del mapa con ayuda de la *API Google Maps Android*.
- Diseño de los diferentes menús que permiten interactuar y usar las diferentes funcionalidades que la aplicación ofrece.
- Visibilidad de las paradas de buses de las diferentes líneas de la ciudad de Ambato.
- Visibilidad de los nombres de las líneas y la identificación de cada una de las paradas.

## 3.2 Fase de inicialización

### 3.2.1 Preparación del proyecto

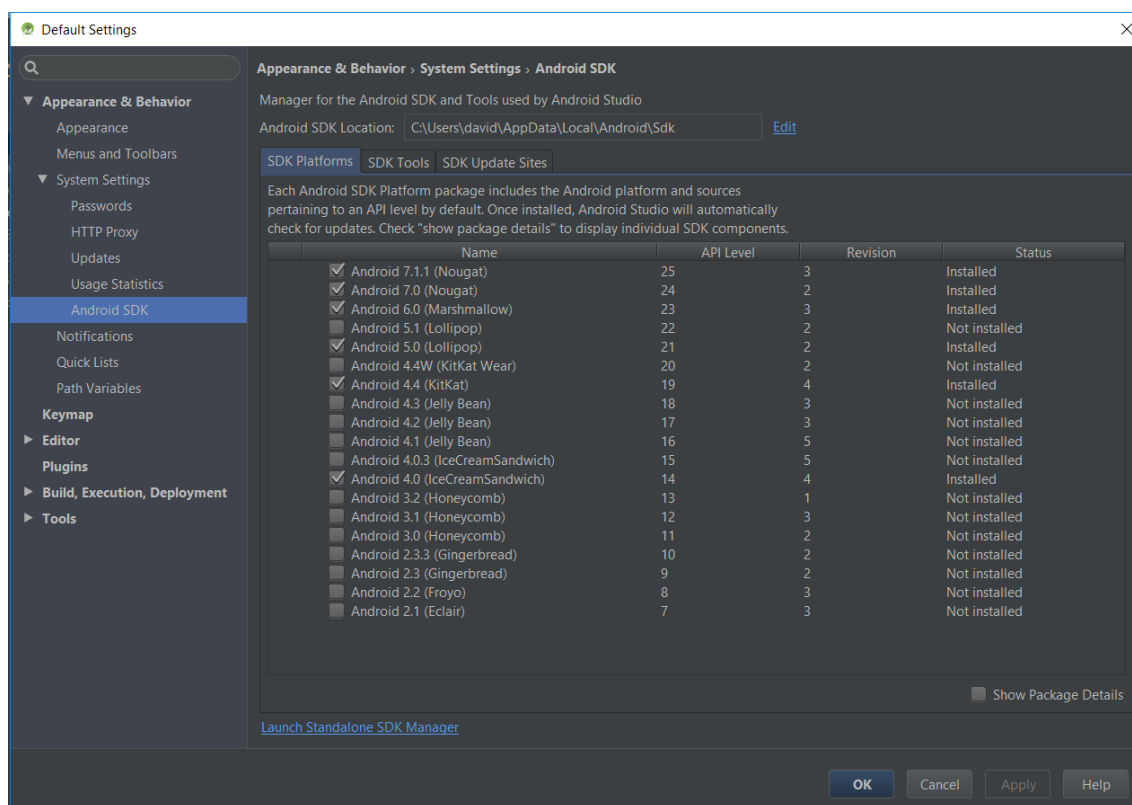
Para comenzar a realizar este proyecto, fue necesario establecer tres partes principales:

- **Preparación del ambiente:** consta en dejar claro que herramientas se va a utilizar en el transcurso del proyecto de manera que, al iniciar el desarrollo del mismo, todo transcurra sin ninguna novedad y mucho menos con retrasos. Para ello se instaló lo siguiente:

- El lenguaje de programación para dispositivos móviles *Android Studio*, con las librerías y plataformas necesarias para el funcionamiento correcto de las *APIs de Google*:

### Plataformas

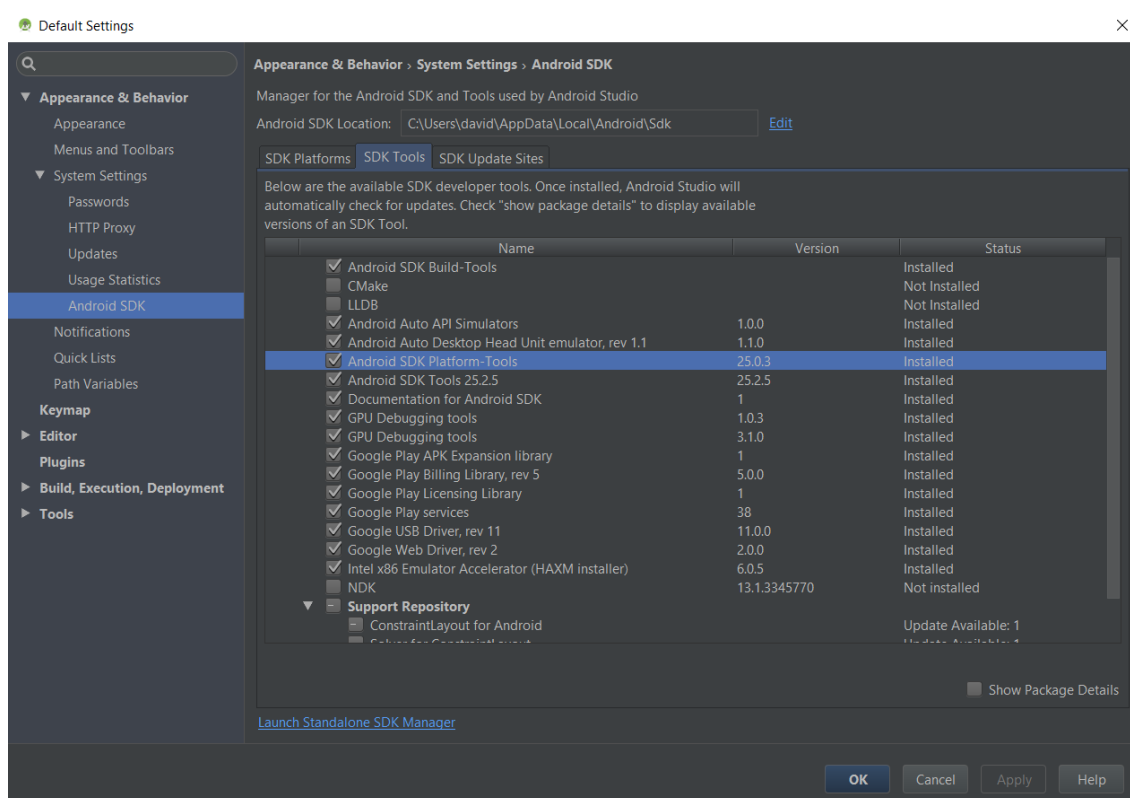
- Android 4.0 (*IceCreamSandwich*).
- Android 4.4 (*KitKat*).
- Android 5.0 (*Lollipop*).
- Android 6.0 (*Marshmallow*).
- Android 7.0 (*Nougat*).
- Android 7.1.1 (*Nougat*).



**Ilustración 3.1** Plataformas de Android Studio  
Fuente: elaboración propia

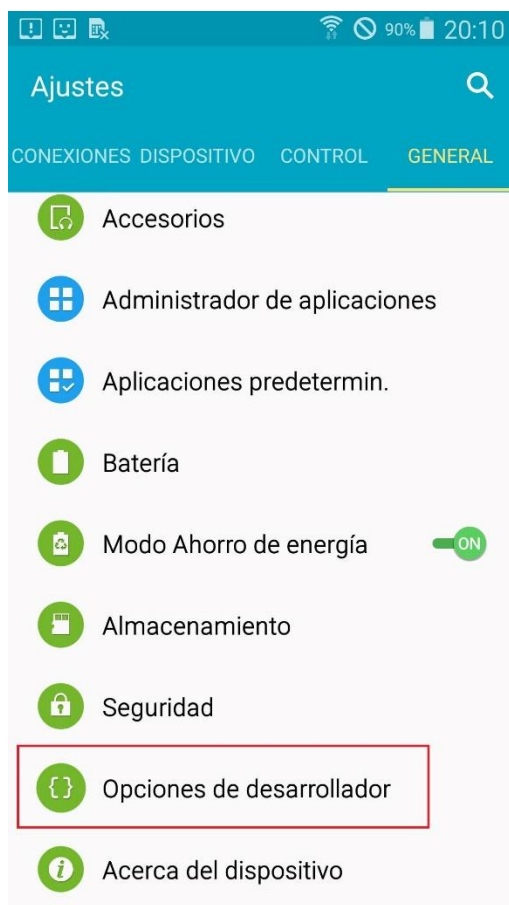
**Librerías:**

- Android SDK Build-Tools
- Android Auto API Simulators V. 1.0.0
- Android Auto Desktop Head Unit emulator, rev 1.1 V. 1.1.0
- Android SDK Platform-Tools V. 25.0.3
- Android SDK Build-Tools V. 25.2.5
- Documentation for Android SDK V.1
- GPU Debugging tools V.1.0.3
- GPU Debugging tools V.3.1.0
- Google Play APK Expansion Library V. 1
- Google Play Billing Library, rev 5 V. 5.0.0
- Google Play Licensing Library V.1
- Google Play service V. 38
- Google USB Driver, rev 11 V.11.0.0
- Google Web Driver, rev 2 V.2.0.0
- Intel x86 Emulator Accelerator (HAXM installer) V. 6.0.5
- Android Support Repository V. 43.0.0
- Google Repository V.43



**Ilustración 3.2** Herramientas de Android Studio  
Fuente: elaboración propia

- Otra herramienta importante que se debió preparar es un dispositivo móvil Samsung Galaxy Note 3 con sistemas operativo Android 5.0., activando el modo desarrollador y depuración USB para realizar las pruebas pertinentes de la aplicación enlazado con *Android Studio*.

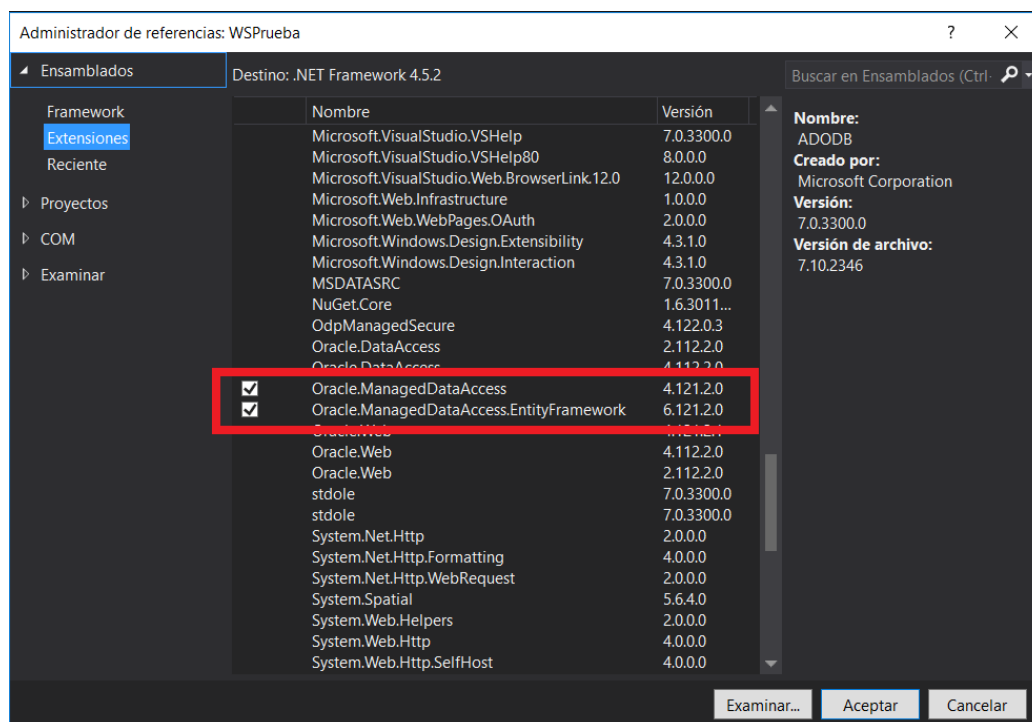


**Ilustración 3.3** Modo de desarrollador en teléfono Android  
**Fuente:** elaboración propia



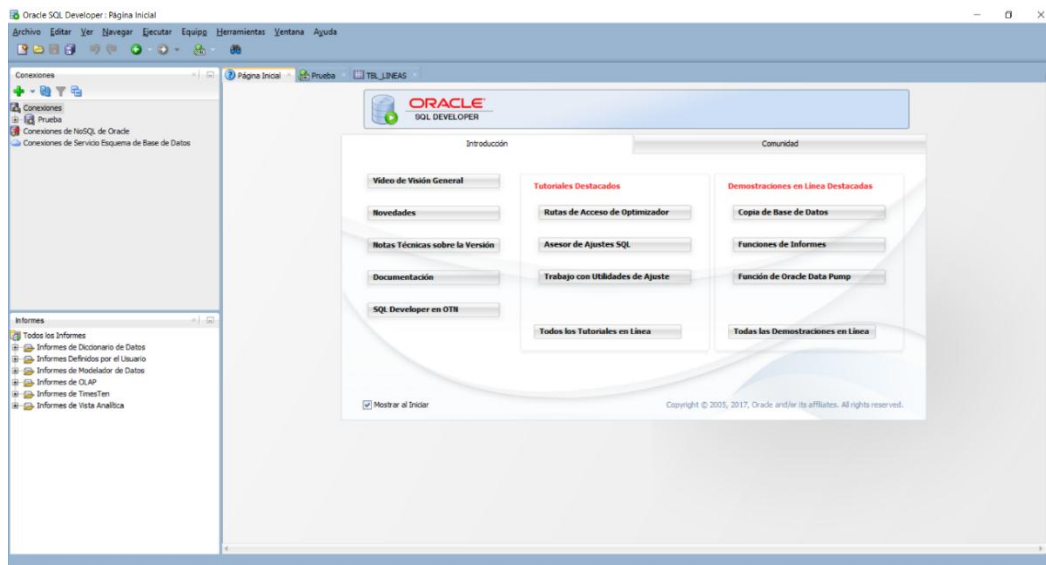
**Ilustración 3.4** Activar la depuración por medio de USB  
**Fuente:** elaboración propia

- Se preparó el lenguaje de programación *Visual Studio 2015* para la creación de *Web Service* con su respectiva librería ODBC (*Open DataBase Connectivity*) para que se pueda conectar con una base de datos *Oracle*.



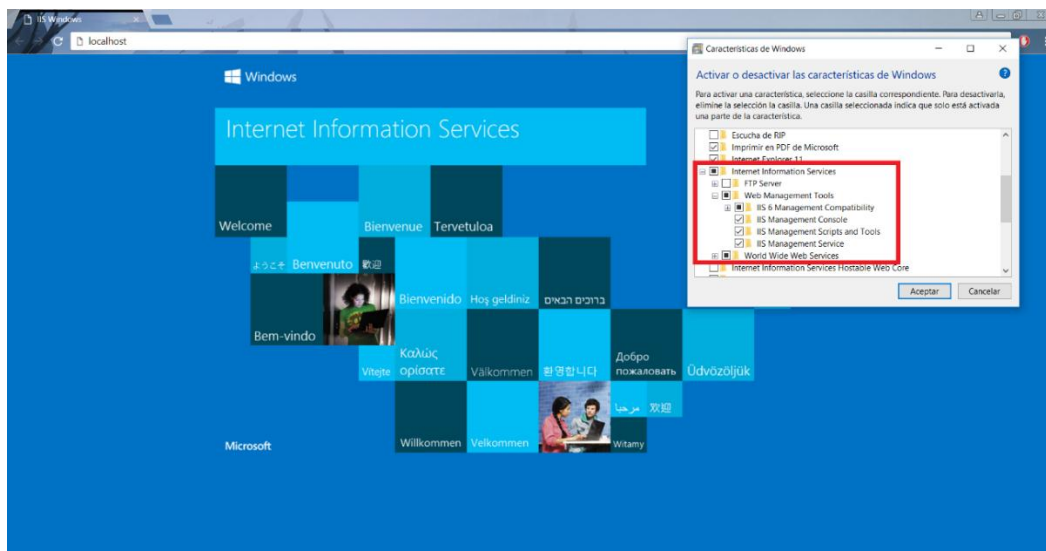
**Ilustración 3.5** Librería ODBC en Visual Basic  
Fuente: elaboración propia

- Instalación de la base de datos *Oracle 11g Express* y *Oracle SQL Developer* para la adecuada administración de la información.

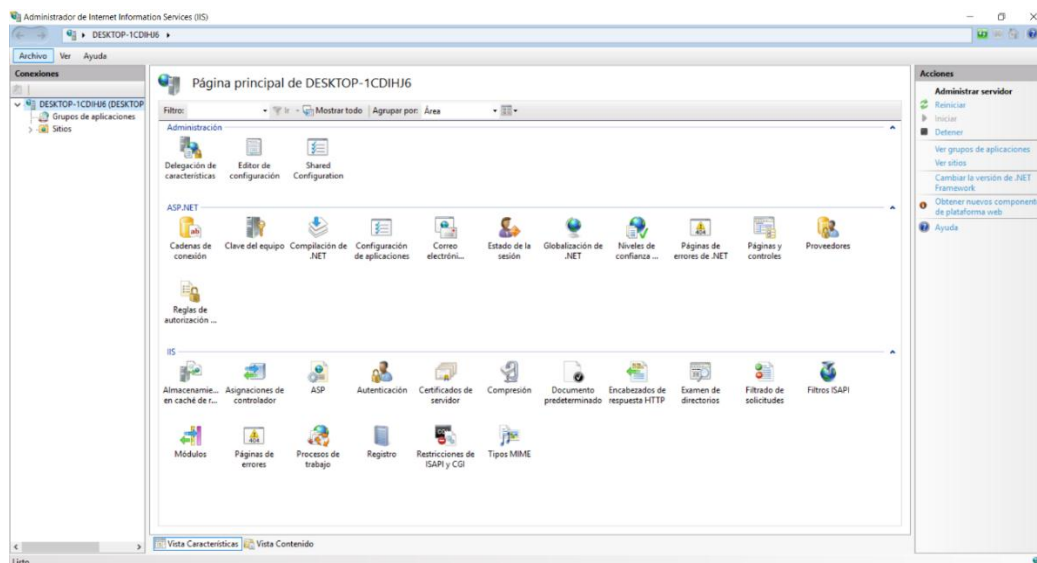


**Ilustración 3.6** Interfaz de Oracle SQL Developer  
Fuente: elaboración propia

- Instalación y levantamiento del servidor Internet Information Services (IIS) para las pruebas respectivas del uso de las *Web Services*.



**Ilustración 3.7** Instalación del servidor de Internet Information Services  
Fuente: elaboración propia



**Ilustración 3.8** Interfaz del servidor de Internet Information Services  
**Fuente:** elaboración propia

- **Capacitación del personal:** en cuestión al conocimiento que se debe tener para utilizar las diferentes herramientas expuestas anteriormente, se usó el contenido y las enseñanzas impartidas de [19], en donde se explica cada uno de los elementos dentro del entorno de *Android Studio*, por otro lado, en la página web [28], se encuentra documentación donde se muestran los diferentes códigos que se pueden emplear en el desarrollo de la aplicación, de igual manera videos tutoriales en línea sirvieron para la debida capacitación en el manejo de las herramientas.
- **Establecer la comunicación con los interesados:** en cuanto a la comunicación con los interesados, se resolvió realizar una reunión por cada entregable que se concluya con el fin de encontrar y corregir errores, realizar mejoras y de esta manera satisfacer los requerimientos antes mencionados.

### **3.2.2 Planificación inicial**

Se planifica el tiempo de duración de cada uno de los entregables y se asignan los recursos necesarios para que el desarrollo de cada uno de ellos tenga éxito y no ocurra pérdida de tiempo.

### **3.2.3 Día de lanzamiento**

Para obtener un mejor resultado al final del desarrollo de la aplicación es necesario realizar una serie de pruebas de las herramientas escogidas hasta familiarizarse con ayuda de documentación y videos tutoriales impartidos de las páginas [19] y [28].

## **3.3 Fase de Desarrollo**

Esta fase consta de varias tareas, de las cuales dos son las más importantes:

- Documentación de los avances que se van realizando.
- La comunicación con los interesados mediante reuniones exponiendo los avances de acuerdo a los requisitos dados.

### **3.3.1 Primera iteración**

#### **3.3.1.1 Día de planificación**

La intención de este día es que, por medio de la comunicación con las personas interesadas, se pueda escoger y planificar el trabajo para la iteración, y mediante el llenado de los test de aceptación de los diferentes avances o pasos que se vayan dando en la iteración, especificando con sus respectivos resultados.

**Tabla 3.1** Test de aceptación 1

TEST DE ACEPTACIÓN
TEST ID: 1
HISTORIA: recopilación de información
FECHA DE REDACCION: 24.10.2016
FECHAS DE EJECUCIÓN: 25.10.2016
DESCRIPCIÓN: <ul style="list-style-type: none"> <li>• Cantidad de paradas innovadoras oficiales de la ciudad de Ambato, con sus respectivas coordenadas geográfica y el ID o nombre que lo represente.</li> <li>• La cantidad de líneas de buses en el sector urbano de Ambato y un identificador que los caracterice a cada uno (Número, color, entre otros).</li> <li>• Nombres de las cooperativas de buses que laboran dentro de la Ciudad de Ambato.</li> <li>• La cantidad de líneas que existen por cada una de las cooperativas de buses.</li> <li>• Cantidad de paradas oficiales no innovadoras de la ciudad de Ambato, con sus respectivas coordenadas geográfica y que líneas de buses pasan por cada una de ellas.</li> <li>• Las rutas respectivas de cada una de las líneas de transporte de la ciudad de Ambato representadas gráficamente o mediante coordenadas geográficas.</li> </ul>
RESULTADO ESPERADO: <ul style="list-style-type: none"> <li>• Información actualizada, correcta y completa.</li> <li>• Respuesta inmediata.</li> </ul>

**Fuente:** elaboración propia

**Tabla 3.2** Test de aceptación 2

TEST DE ACEPTACIÓN
TEST ID: 2
HISTORIA: creación de la base de datos
FECHA DE REDACCION: 24.10.2016
FECHAS DE EJECUCIÓN: 25.10.2016
DESCRIPCIÓN: <ul style="list-style-type: none"> <li>• Deben analizarse que campos deben ir en cada tabla creada</li> <li>• Creación de filas y columnas.</li> <li>• Deben estar correctamente relacionas mediante el MER (Modelo Entidad Relación)</li> <li>• Cargar la información en la base de datos</li> </ul>
RESULTADO ESPERADO: <ul style="list-style-type: none"> <li>• Toda la información debe ser coherente.</li> <li>• Los datos deben relacionarse adecuadamente.</li> <li>• Los resultados de las consultas deben dar como resultado la información correcta.</li> </ul>

**Fuente:** elaboración propia

### 3.3.1.2 Día de trabajo

La prioridad principal de esta fase es culminar todo el trabajo planeado, tomando en cuenta los requisitos mencionados en el día de planificación para que al culminar la primera iteración los resultados sean los esperados.

#### 3.3.1.2.1 Clasificación de información

La información adquirida según los requerimientos mencionados, se deben clasificar de acuerdo a las posibles tablas que pueden ser creadas para conformar la base de datos. A continuación, se muestra una tabla en donde se especifica lo dicho anteriormente.

**Tabla 3.3** Información tentativa para la base de datos

Posibles tablas	Información
Rutas	Latitud, Longitud, Dirección
Paradas	Nombre, Latitud, Longitud
Líneas	Cooperativa, Nombre, Descripción

**Fuente:** elaboración propia

**Latitud-Longitud:** coordenadas geográficas.

**Rutas\_Dirección:** sentido en el cual el bus debe ir (Salida-Retorno).

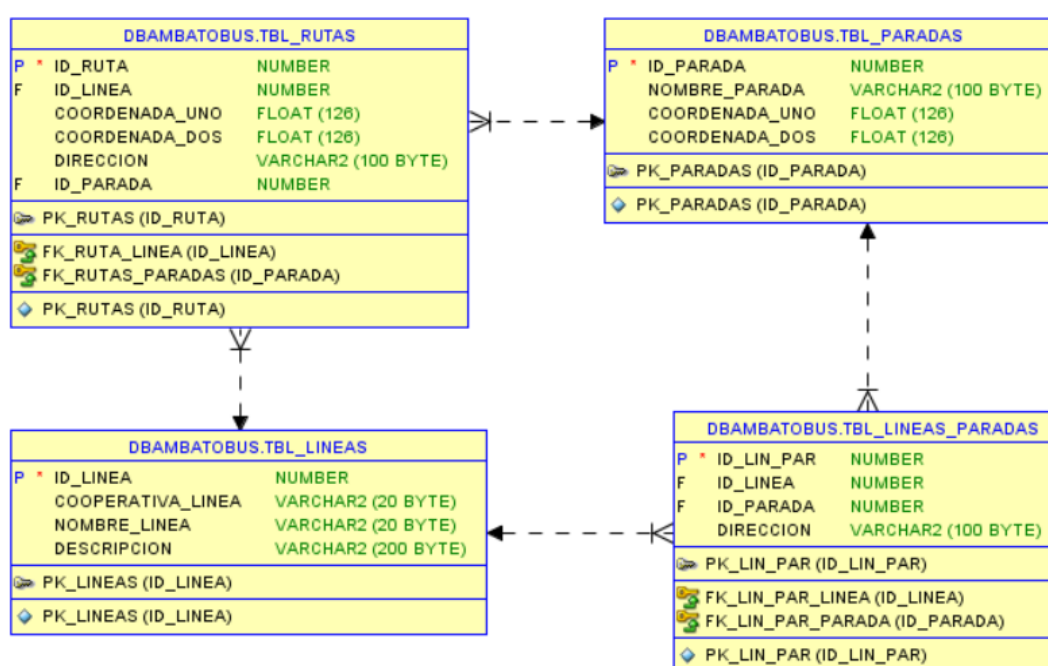
**Paradas\_Nombre:** son los nombres dados a cada una de las paradas que existen tomando como referencia el sector en el que se encuentran situadas.

**Líneas\_Nombre:** son los nombres que representan a cada una de las líneas pertenecientes a las cooperativas existentes.

**Lineas\_Descripción:** son los nombres de los lugares en general por donde cada una de las líneas realizan sus rutas.

### 3.3.1.2.2 Creación de la base de datos

Una vez obtenida la información clasificada, se empieza rápidamente a la creación de las tablas dentro de la base de datos llamada “DBAmbatoBus” con su respectivo usuario y contraseña, por medio de código SQL especificado en el Anexo 1. A continuación, se representa el MER de la base de datos creada:



**Ilustración 3 9** Diagrama Entidad Relación  
Fuente: elaboración propia

- La tabla TBL\_RUTAS está conformada por su clave primaria ID\_RUTA y dos claves foráneas que son la relación con las tablas TBL\_PARADAS y TBL\_LINEAS. Aparte de lo mencionado, contiene toda la información de las coordenadas (latitud, longitud), que trazan las rutas de cada una de las líneas de autobuses que existen dentro de

la ciudad de Ambato y existe otro campo llamado Dirección, en el cual se almacena el sentido que se dirige el autobús (Salida, Retorno).

- La tabla TBL\_PARADAS almacena toda la información referente a cada una de las paradas que existen dentro de la ciudad de Ambato en las cuales se obtiene el nombre de referencia del lugar en donde se encuentran situadas y sus respectivas coordenadas (latitud, longitud) de posicionamiento.
- La tabla TBL\_LINEAS guarda los nombres con que se identifica la línea (pueden ser números y letras), con el respectivo nombre de la cooperativa de autobús al que pertenece y su descripción que son los nombres de los lugares por donde realiza su ruta.
- La tabla TBL\_LINEAS\_PARADAS está conformada solamente por una clave primaria ID\_LIN\_PAR, dos claves foráneas que son la relación con las tablas TBL\_PARADAS y TBL\_LINEAS, cuya función es repetir la información en cadena de acuerdo a la información previamente establecida y, por último, un campo Dirección cuya información fue explicada anteriormente.

### **3.3.1.3 Día de lanzamiento**

Una vez culminado el día de trabajo, llega el día de presentación del primer entregable y para ello es esencial informar que la base de datos creada cumpla con todos los resultados esperados mencionados en los test de aceptación. Para ello se realiza un resumen en donde se especifican las deficiencias encontradas durante la primera iteración, en este caso, no se

hallaron ningún tipo de deficiencias por lo cual no se realiza el cuadro de resumen de deficiencias.

### 3.3.2 Segunda iteración

#### 3.3.2.1 Día de planificación

La intención de este día es que, por medio de la comunicación con las personas interesadas, se pueda escoger y planificar el trabajo para la iteración, y mediante el llenado de los test de aceptación de los diferentes avances o pasos que se vayan dando en la iteración, ir especificando con sus respectivos resultados.

**Tabla 3.4** Test de aceptación 3

TEST DE ACEPTACIÓN
TEST ID: 3
HISTORIA: Creación de una <i>Web Services</i>
FECHA DE REDACCION: 31.10.2016
FECHAS DE EJECUCIÓN: 01.11.2016
DESCRIPCIÓN: <ul style="list-style-type: none"> <li>• Conexión a la base de datos Oracle.</li> <li>• Establecer diferentes consultas para obtener la información de la base de datos.</li> <li>• Debe estar alojada en el IIS (<i>Internet Information Services</i>).</li> </ul>
RESULTADO ESPERADO: <ul style="list-style-type: none"> <li>• La <i>web services</i> se conectará automáticamente con la base de datos.</li> <li>• Las consultas deben ser apropiadas para adquirir la información precisa de la base de datos y de esta manera los resultados se visualicen de forma apropiada.</li> <li>• Al probar el funcionamiento de la <i>web service</i> mediante el IIS, debe funcionar correctamente.</li> </ul>

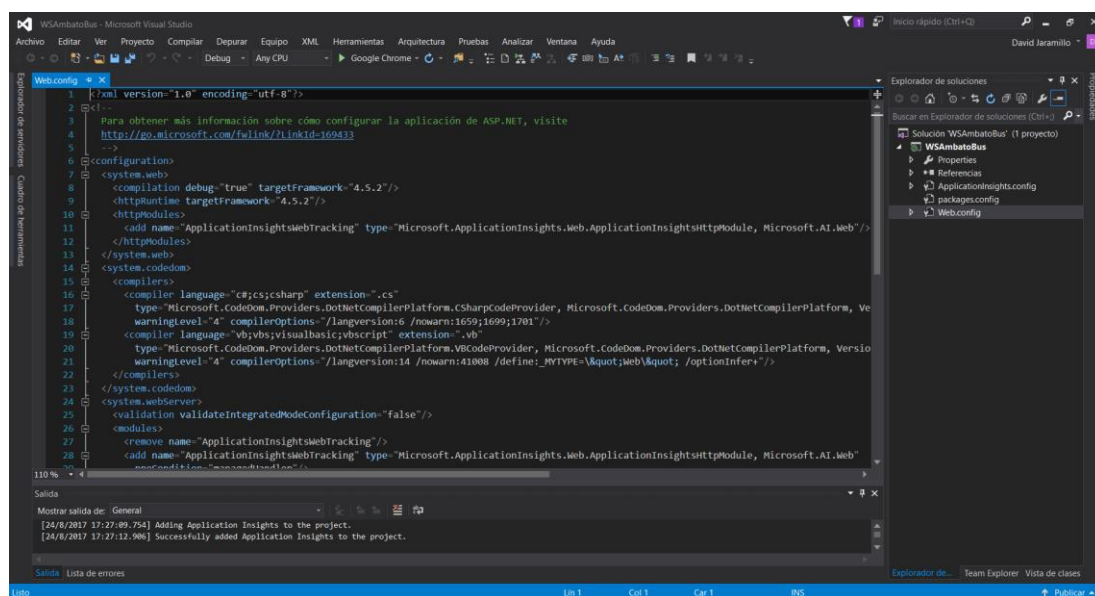
**Fuente:** elaboración propia

#### 3.3.2.2 Día de trabajo

La prioridad principal de esta fase es culminar todo el trabajo planeado, tomando en cuenta los requisitos mencionados en el día de planificación para que al culminar la segunda iteración los resultados sean los esperados.

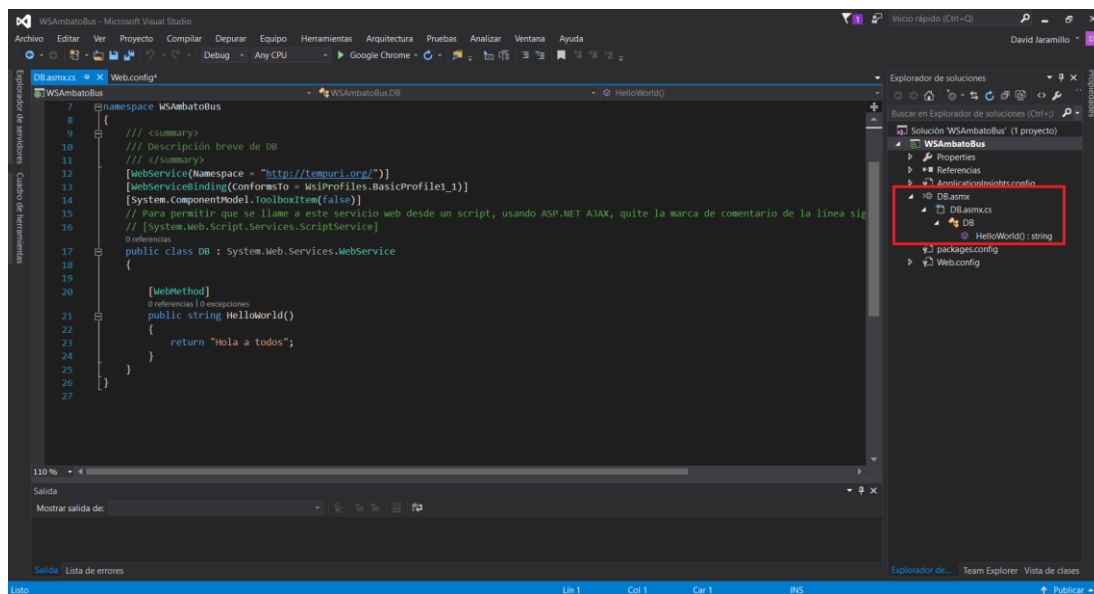
### 3.3.2.2.1 Creación de la *web services*

Dentro del entorno de Visual Basic 2015, se crea un nuevo proyecto web con el nombre de WSAmbatoBus especificando el lenguaje de programación Visual C#.



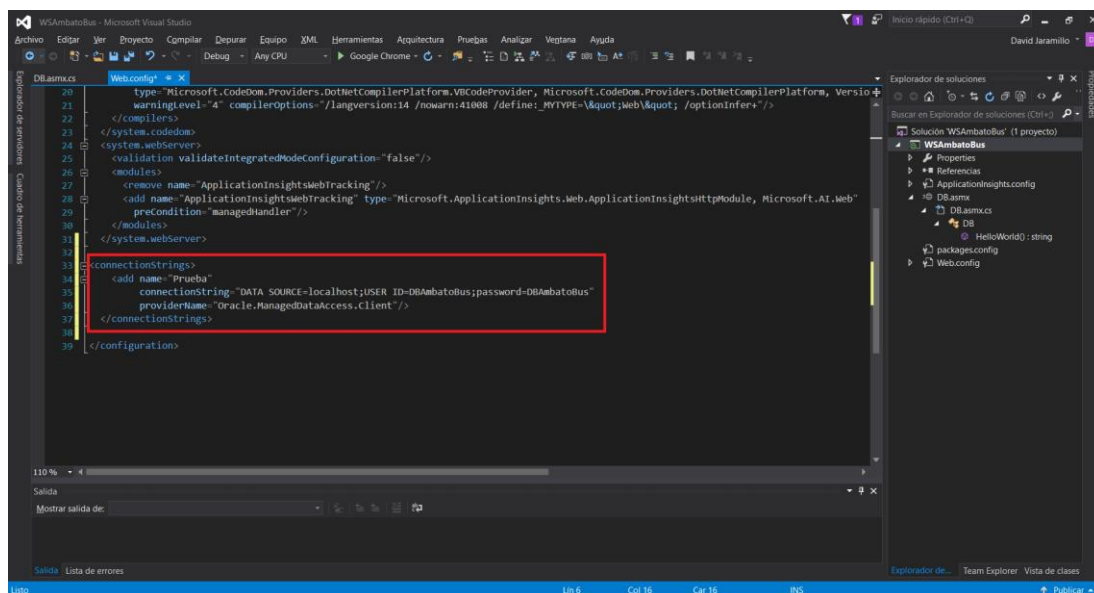
**Ilustración 3.10** Creación de un proyecto web en Visual Studio  
Fuente: elaboración propia

A continuación, se crea dentro del proyecto un servicio web (ASMX) con el nombre de DB el cual servirá para realizar las debidas consultas a la base de datos.



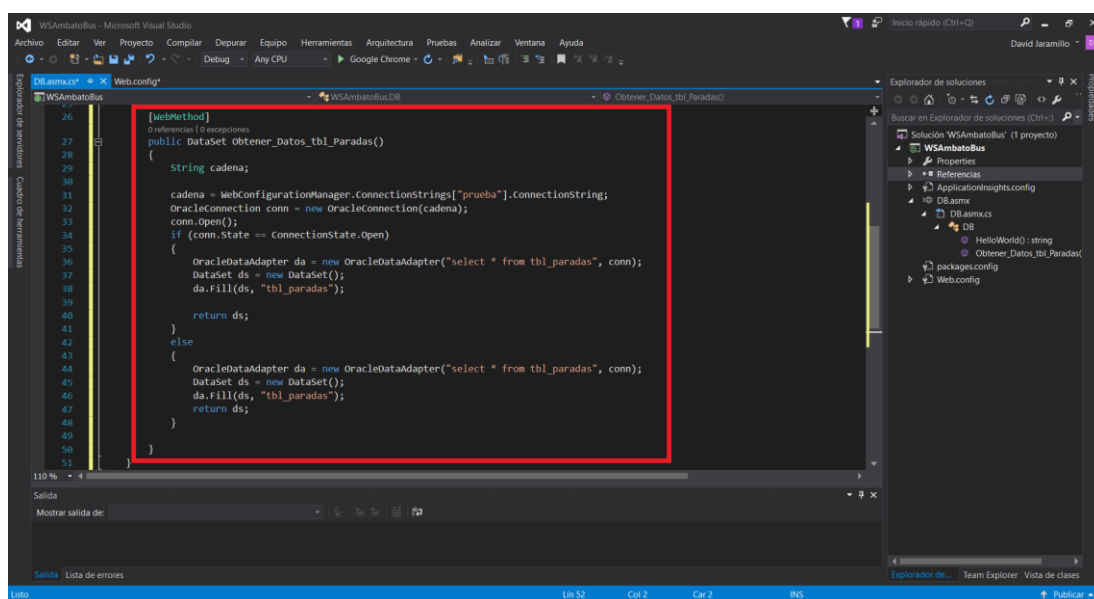
**Ilustración 3.11** Creación de un servicio web (ASMX)  
Fuente: elaboración propia

Es recomendable añadir el código de conectividad con la base de datos dentro del archivo “Web.config”, ya que, al momento de utilizar el servicio web, la conexión a la base de datos se realiza una sola vez.



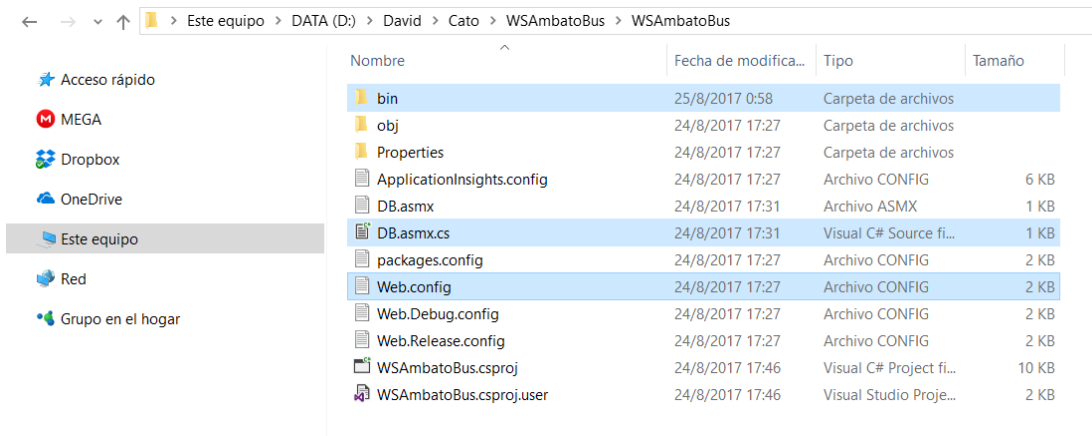
**Ilustración 3.12** Conexión de la WS con la Base de Datos Oracle  
Fuente: elaboración propia

Dentro del archivo DB.asmx.cs, se codifican las consultas necesarias para que la aplicación funcione correctamente. En el Anexo 2 se muestran detalladamente todas las consultas que se utilizan para que la App funcione correctamente.

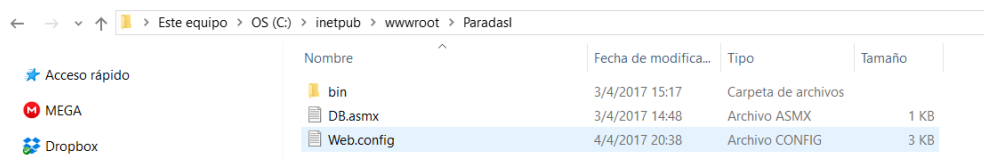


**Ilustración 3.13** Codificación de las consultas a la base de datos en la WS  
**Elaborado por:** Investigador

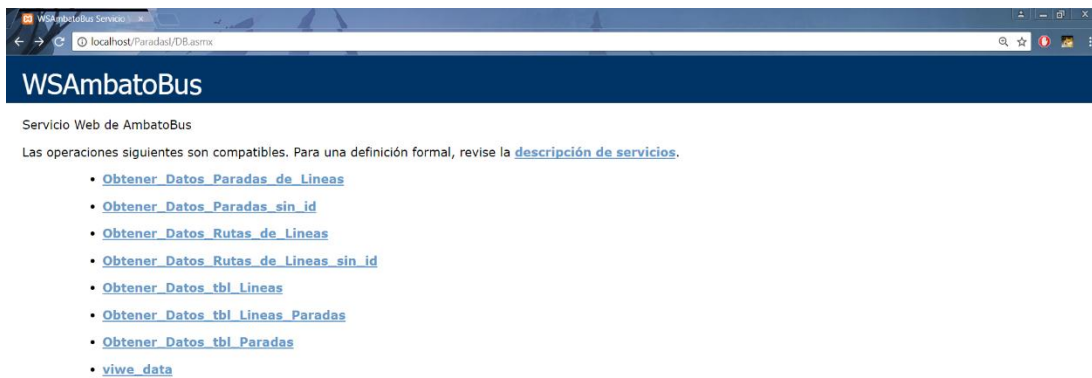
Para subir el *Web Services* al servidor local IIS, es necesario crear una nueva carpeta dentro del directorio `C:\inetpub\wwwroot` con el nombre `ParadasI`, y dentro de la misma se deben copiar los archivos generados al momento de terminar y guardar el servicio web en Visual Basic 2015 como muestran las imágenes siguientes.



**Ilustración 3.14** Archivos para levantar la WS  
Fuente: elaboración propia

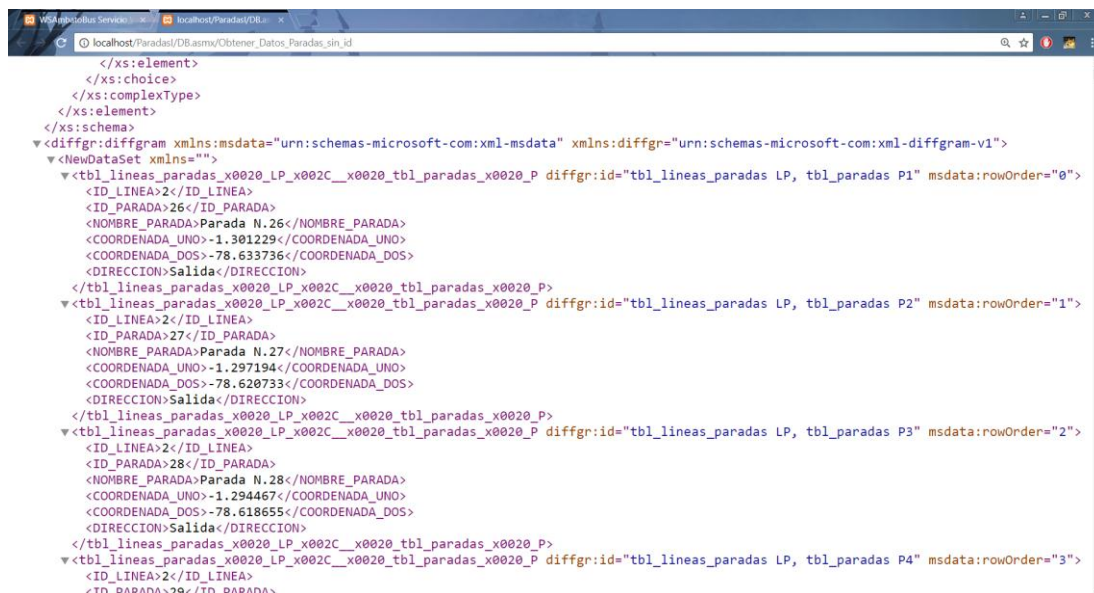


**Ilustración 3.15** Archivos dentro de la servidores IIS  
Fuente: elaboración propia



**Ilustración 3.16** Consultas de la WS en el IIS  
Fuente: elaboración propia

La respuesta que debe dar al probar la Web Services, debe ser en código XML especificando la información solicitada mediante la consulta escogida.



```

</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>
<diffgr:diffgram xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" xmlns:diffgr="urn:schemas-microsoft-com:xml-diffgram-v1">
  <NewDataSet xmlns="">
    <tbl_lineas_paradas_x0020_LP_x002C_x0020_tbl_paradas_x0020_P diffgr:id="tbl_lineas_paradas LP, tbl_paradas P1" msdata:rowOrder="0">
      <ID LINEA>2</ID LINEA>
      <ID PARADA>26</ID PARADA>
      <NOMBRE_PARADA>Parada N.26</NOMBRE_PARADA>
      <COORDENADA_UNO>-1.301229</COORDENADA_UNO>
      <COORDENADA_DOS>-78.633736</COORDENADA_DOS>
      <DIRECCION>Salida</DIRECCION>
    </tbl_lineas_paradas_x0020_LP_x002C_x0020_tbl_paradas_x0020_P>
    <tbl_lineas_paradas_x0020_LP_x002C_x0020_tbl_paradas_x0020_P diffgr:id="tbl_lineas_paradas LP, tbl_paradas P2" msdata:rowOrder="1">
      <ID LINEA>2</ID LINEA>
      <ID PARADA>27</ID PARADA>
      <NOMBRE_PARADA>Parada N.27</NOMBRE_PARADA>
      <COORDENADA_UNO>-1.297194</COORDENADA_UNO>
      <COORDENADA_DOS>-78.620733</COORDENADA_DOS>
      <DIRECCION>Salida</DIRECCION>
    </tbl_lineas_paradas_x0020_LP_x002C_x0020_tbl_paradas_x0020_P>
    <tbl_lineas_paradas_x0020_LP_x002C_x0020_tbl_paradas_x0020_P diffgr:id="tbl_lineas_paradas LP, tbl_paradas P3" msdata:rowOrder="2">
      <ID LINEA>2</ID LINEA>
      <ID PARADA>28</ID PARADA>
      <NOMBRE_PARADA>Parada N.28</NOMBRE_PARADA>
      <COORDENADA_UNO>-1.294467</COORDENADA_UNO>
      <COORDENADA_DOS>-78.618655</COORDENADA_DOS>
      <DIRECCION>Salida</DIRECCION>
    </tbl_lineas_paradas_x0020_LP_x002C_x0020_tbl_paradas_x0020_P>
    <tbl_lineas_paradas_x0020_LP_x002C_x0020_tbl_paradas_x0020_P diffgr:id="tbl_lineas_paradas LP, tbl_paradas P4" msdata:rowOrder="3">
      <ID LINEA>2</ID LINEA>
      <ID PARADA>29</ID PARADA>
    </tbl_lineas_paradas_x0020_LP_x002C_x0020_tbl_paradas_x0020_P>
  </NewDataSet>
</diffgr:diffgram>

```

**Ilustración 3.17** Respuesta de las consultas de la WS en internet

Fuente: elaboración propia

### 3.3.2.3 Día de lanzamiento

Una vez culminado el día de trabajo, llega el día de presentación del segundo entregable y para ello es esencial informar que la *Web Services* cumpla con todos los resultados esperados mencionados en los test de aceptación. En ese caso, no existen problemas para realizar el test de deficiencias que menciona la metodología.

### 3.3.3 Tercera iteración

#### 3.3.3.1 Día de planificación

La intención de este día es que, por medio de la comunicación con las personas interesadas, se puede escoger y planificar el trabajo para la iteración, y mediante el llenado de los test de aceptación de los diferentes

avances o pasos que se vayan dando en la iteración, ir especificando con sus respectivos resultados.

**Tabla 3.5** Test de aceptación 4

TEST DE ACEPTACIÓN
TEST ID: 4
HISTORIA: diseño de la interfaz de la App
FECHA DE REDACCION: 7.11.2016
FECHAS DE EJECUCIÓN: 8.11.2016
DESCRIPCIÓN: <ul style="list-style-type: none"> <li>• Debe tener una pantalla de carga.</li> <li>• La pantalla principal debe tener un diseño apropiado.</li> <li>• En la pantalla principal debe constar los botones que cumplan las funciones básicas de la aplicación.</li> <li>• Debe tener un menú donde haya todas las líneas de buses</li> </ul>
RESULTADO ESPERADO: <ul style="list-style-type: none"> <li>• Antes de iniciar la aplicación, se debe mostrar una pantalla de carga.</li> <li>• El diseño de la pantalla principal.</li> <li>• Los botones deben ser estandarizados.</li> <li>• El menú debe tener todas las líneas con sus respectivas cooperativas.</li> </ul>

**Fuente:** elaboración propia

**Tabla 3.6** Test de aceptación 5

TEST DE ACEPTACIÓN
TEST ID: 5
HISTORIA: implementar el mapa en la app por medio del API de Google Maps para Android
FECHA DE REDACCION: 7.11.2016
FECHAS DE EJECUCIÓN: 8.11.2016
DESCRIPCIÓN: <ul style="list-style-type: none"> <li>• El mapa debe visualizarse correctamente.</li> <li>• Debe mostrar la ubicación del usuario.</li> <li>• El mapa debe ser funcional.</li> </ul>
RESULTADO ESPERADO: <ul style="list-style-type: none"> <li>• El mapa debe mostrarse en un tamaño adecuado.</li> <li>• Se debe habilitar un botón el cual funcione para saber la ubicación del usuario mediante un marcador.</li> <li>• Debe tener los diferentes estilos de mapas: <ul style="list-style-type: none"> <li>• Satelital</li> <li>• Hibrido</li> <li>• Normal</li> <li>• Tierra</li> </ul> </li> </ul>

**Fuente:** elaboración propia

**Tabla 3.7** Test de aceptación 6

TEST DE ACEPTACIÓN
TEST ID: 6
HISTORIA: codificación de la App e implementación de las demás APIs de Google
FECHA DE REDACCION: 7.11.2016
FECHAS DE EJECUCIÓN: 8.11.2016
DESCRIPCIÓN: <ul style="list-style-type: none"> <li>• Desarrollar la conexión a la <i>Web Services</i>.</li> <li>• Desarrollar el funcionamiento correcto de la App.</li> <li>• Implementar todas las demás APIs necesarias.</li> </ul>
RESULTADO ESPERADO: <ul style="list-style-type: none"> <li>• Toda la información de la base de datos sea receptada correctamente por la App.</li> <li>• Las rutas se muestren correctamente en el mapa.</li> <li>• La funcionalidad de cada una de las APIs sea correcta.</li> </ul>

**Fuente:** elaboración propia

### 3.3.3.2 Día de trabajo

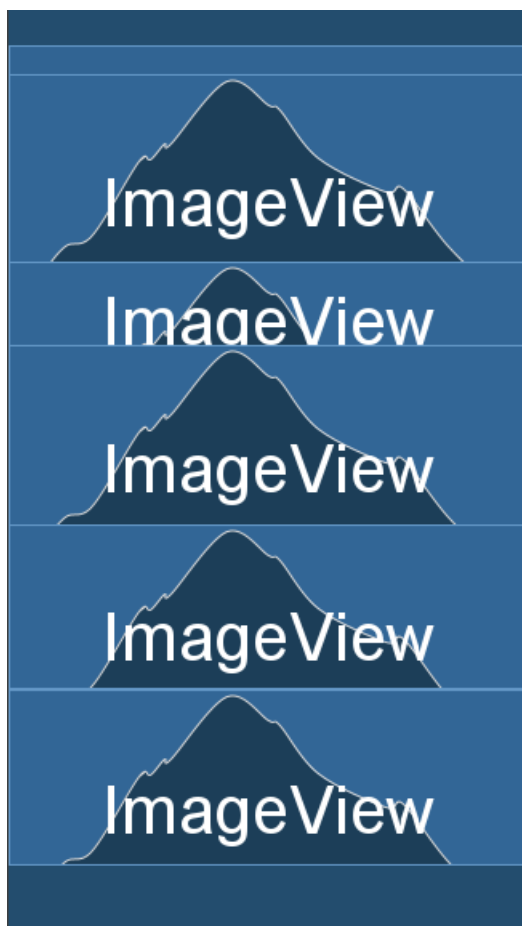
La prioridad principal de esta fase es culminar todo el trabajo planeado, tomando en cuenta los requisitos mencionados en el día de planificación para que al culminar la tercera iteración los resultados sean los esperados.

#### 3.3.3.2.1 Diseño de la interfaz de la App

En esta iteración es donde se empieza a trabajar dentro del entorno de Android Studio, lo cual, al momento de crear un nuevo proyecto y después de haber puesto el nombre del proyecto, la ruta de donde se va a guardar y escoger los dispositivos Android a los que va dirigida la aplicación, se escoge el modelo de la interfaz gráfica que ya viene prediseñada llamada *Navigation Drawer Activity*, la razón por la que se escoge este diseño es porque la aplicación necesita dicha interfaz para satisfacer los resultados esperados descritos en el Test de Aceptación 4.

Antes de rediseñar la interfaz gráfica escogida, se crea una nueva *Activity* en blanco con el nombre de PantallaInicio, dentro del mismo proyecto, con el propósito de realizar la pantalla de carga o *SplashScreen* que se mostrará antes de que la aplicación inicie completamente.

El momento de crear el nuevo Activity, se origina automáticamente un archivo llamado *activity\_pantalla\_inicio.xml*, dentro del cual se realiza el diseño del *SplashScreen*. A continuación, se muestra el maquetado de la pantalla de carga:



**Ilustración 3.18** Maquetado del *SplashScreen*  
Fuente: elaboración propia

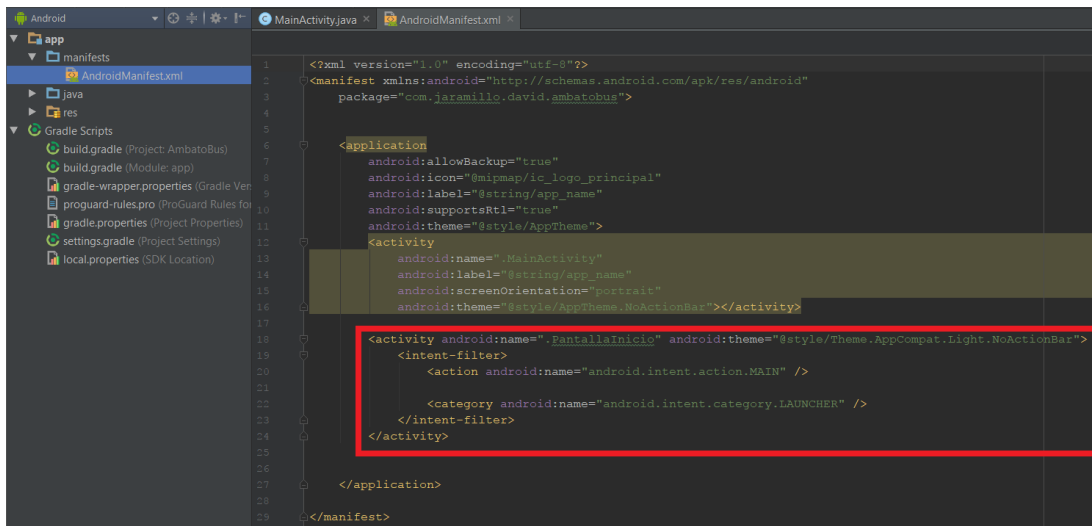
Al tener el maquetado listo, el diseño final se presenta en la siguiente ilustración:



**Ilustración 3.19** Diseño del *SplashScreen*  
Fuente: elaboración propia

Al tener culminado el diseño, se procede a transformarlo en *Activity* principal de la aplicación para que, al momento de ejecutarla, esta sea la primera en mostrarse. Para ello, dentro del archivo *AndroidManifest.xml*, se aumenta automáticamente una nueva sección *<activity>*, en la cual se aumenta las instrucciones para que su funcionamiento sea acorde lo mencionado anteriormente.

Adicionalmente, se le añade un tema el cual indicará que el *ActionBar* se desaparezca del *SplashScreen*.



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.jaramillo.david.ambatobus">
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_logo_principal"
        android:label="@string/app_name"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity
            android:name=". MainActivity"
            android:label="@string/app_name"
            android:screenOrientation="portrait"
            android:theme="@style/AppTheme.NoActionBar"></activity>
        <activity android:name=". PantallaInicio" android:theme="@style/Theme.AppCompat.Light.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

**Ilustración 3.20** Cambio del *Activity principal* en el *AndroidManifest.xml*  
**Fuente:** elaboración propia

Dentro del archivo *PantallaInicio.java*, se añade las instrucciones necesarias para que el funcionamiento del *SplashScreen* sea adecuada.

```

protected static final int TIMER_RUNTIME=5000;
protected boolean mbActive;
protected ProgressBar mProgressBar;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_pantalla_inicio);
    mProgressBar = (ProgressBar) findViewById(R.id.ProgressBarH);
    final Thread timerThread = run() → {
        mbActive = true;
        try{
            int waited = 0;
            while(mbActive && (waited<TIMER_RUNTIME)) {
                sleep(200);
                if (mbActive){
                    waited+=200;
                    updateProgress (waited);
                }
            }
        } catch (InterruptedException e) {
            e.printStackTrace();
        } finally {
            onCounted();
        }
    };
    timerThread.start();
}
public void updateProgress(final int timePassed){
    if(null != mProgressBar){
        final int progress = mProgressBar.getMax()*timePassed / TIMER_RUNTIME;
        mProgressBar.setProgress (progress);
    }
}
public void onCounted(){
    Intent intent = new Intent(PantallaInicio.this, MainActivity.class);
    startActivity(intent);
    finish();}

```

**Ilustración 3.21** Codificación en el archivo *PantallaInicio.java*  
**Fuente:** elaboración propia

Al realizar todo correctamente, la aplicación deberá lucir como se muestra a continuación:



**Ilustración 3.22** Resultado del SplashScreen  
**Fuente:** elaboración propia

Una vez culminado la pantalla de carga, se procede a rediseñar la interfaz del *Navigation Drawer Activity*. Se añaden algunos elementos necesarios, lo que en otras palabras viene siendo el maquetado, el cual, se lo realiza en el propio editor de interfaz de Android Studio, que permite editar a su gusto la interfaz de la aplicación.

Para la realización del maquetado, se tomaron en cuenta las funciones principales que debe cumplir la aplicación tales como:

- Un *AutoCompleteTextView* con la función de origen, que sirve para mostrar como referencia de la ubicación en donde se encuentra el usuario.

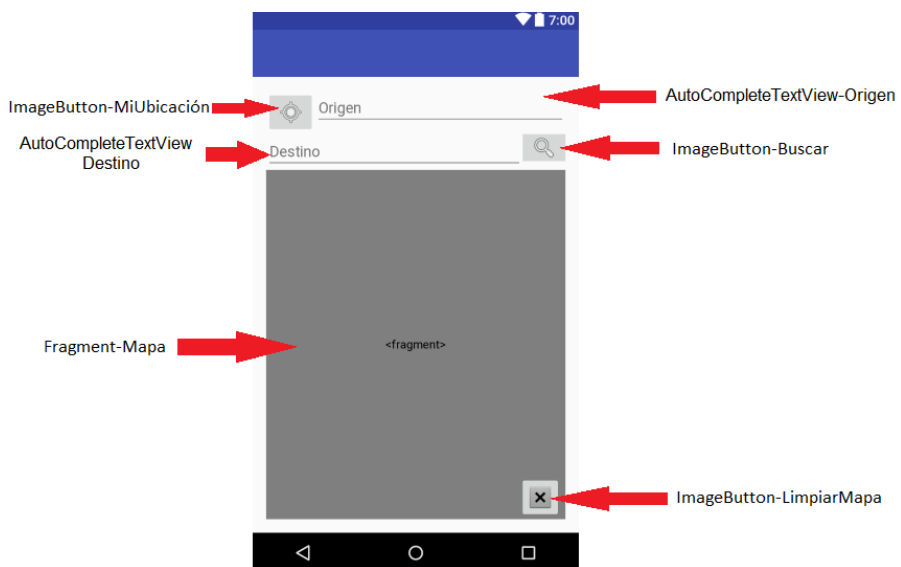
- Un *AutoCompleteTextView* con la función de destino, que sirve para que el usuario escriba el nombre del destino al que desee transportarse.
- Un *ImageButton*, cuya función es mandar a buscar la mejor ruta entre el origen y el destino del cual el usuario especifique.
- Un *ImageButton*, cuya función es colocar una referencia de la ubicación del usuario.
- Un *ImageButton*, que sirve para limpiar el mapa.
- Un *fragment*, en el cual se mostrará el mapa subido por la API de Google Maps Android

En la siguiente ilustración, se muestra el maquetado de la interfaz principal de la aplicación:



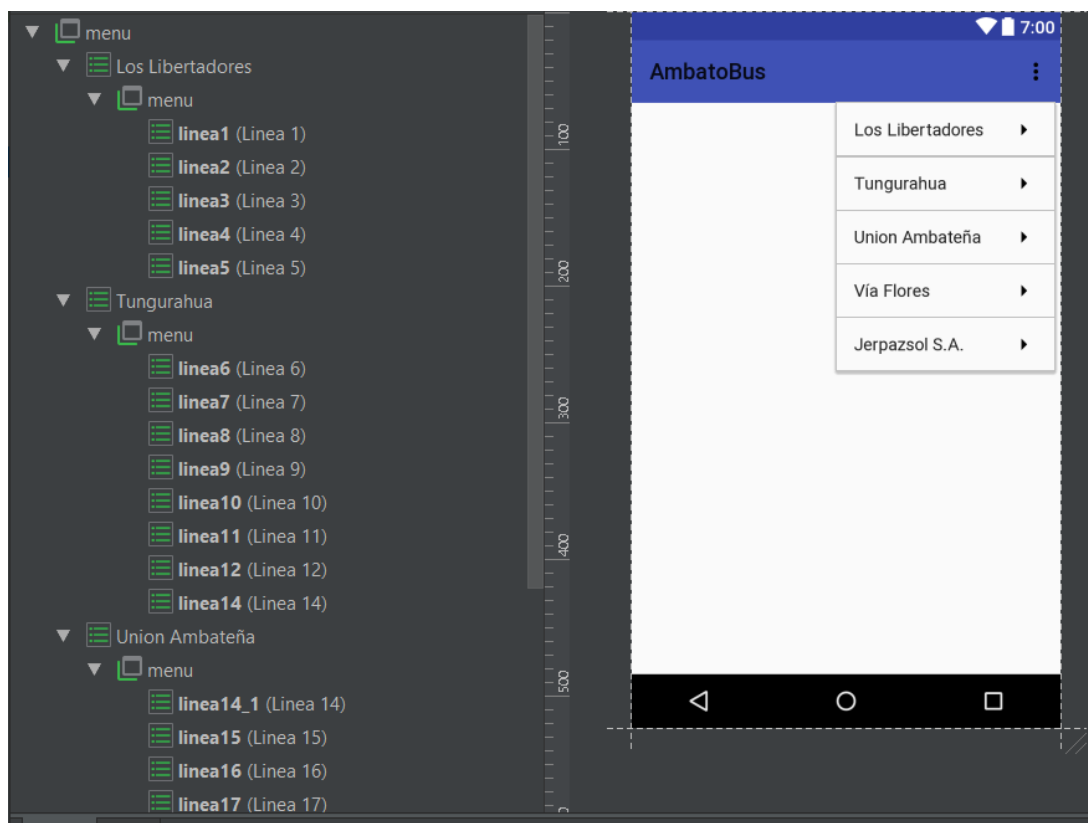
**Ilustración 3.23** Maquetado de la aplicación  
Fuente: elaboración propia

A continuación, se indica cada uno de los componentes que conforman la interfaz principal de la aplicación:

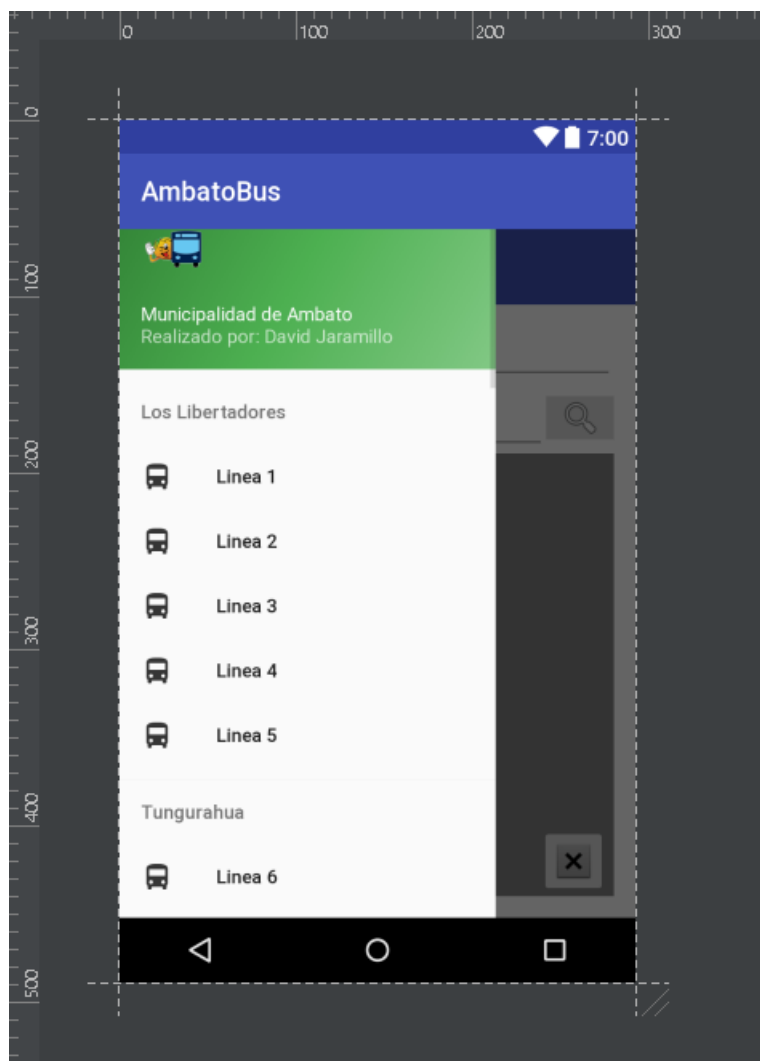


**Ilustración 3.24** Identificación de las partes del maquetado  
Fuente: elaboración propia

Para desarrollar el menú requerido, se edita el diseño predefinido, en donde se divide en 5 secciones, cada una perteneciente a cada cooperativa de transporte de bus existente en la ciudad de Ambato. Cada una de estas secciones consta de varios ítems, los cuales representan a cada una de las líneas que forman parte de las cooperativas.



**Ilustración 3.25** Secciones que se divide el menú principal  
**Fuente:** elaboración propia

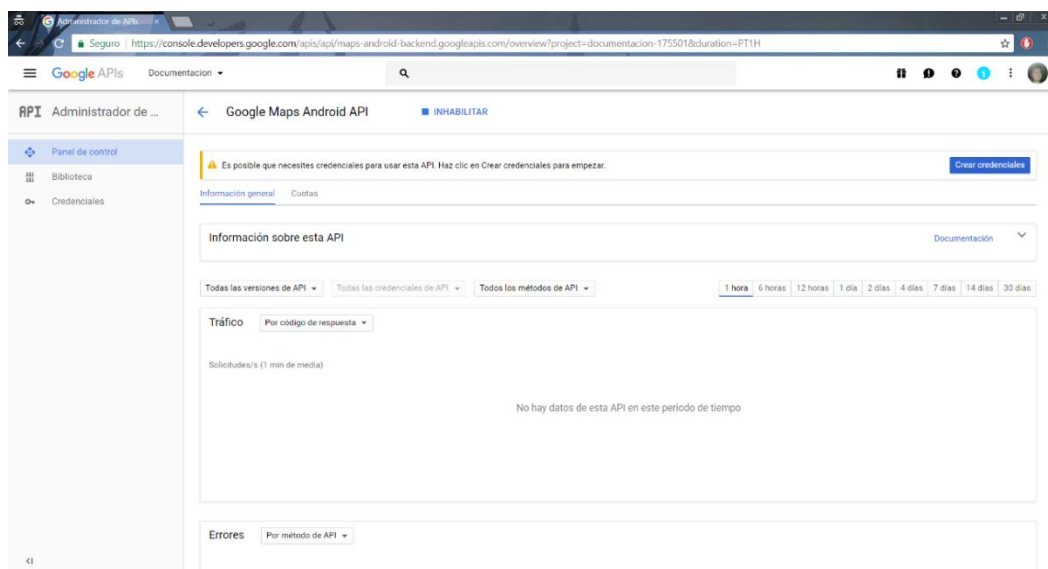


**Ilustración 3.26** Diseño del menú principal  
**Fuente:** elaboración propia

### 3.3.3.2.2 Implementar el mapa en la App.

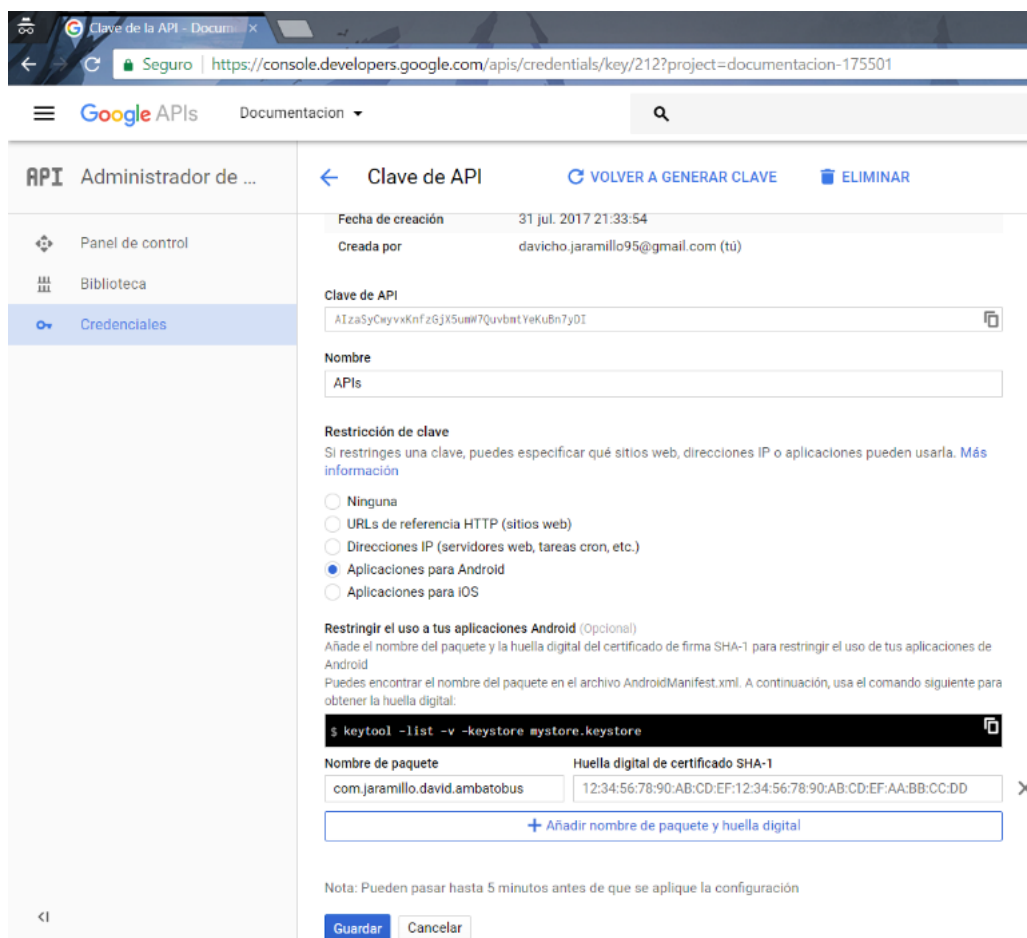
Para implementar el mapa en la aplicación, es necesario ocupar el API de Google Maps para Android. Se accede a *Google Cloud Platform* en donde se puede habilitar y obtener una clave que sirve para la comunicación entre la aplicación y un servicio web proporcionado por Google.

Para ello se debe crear un nuevo proyecto y después en el submenú “Biblioteca” buscar “Google Maps Android API” y habilitarlo.

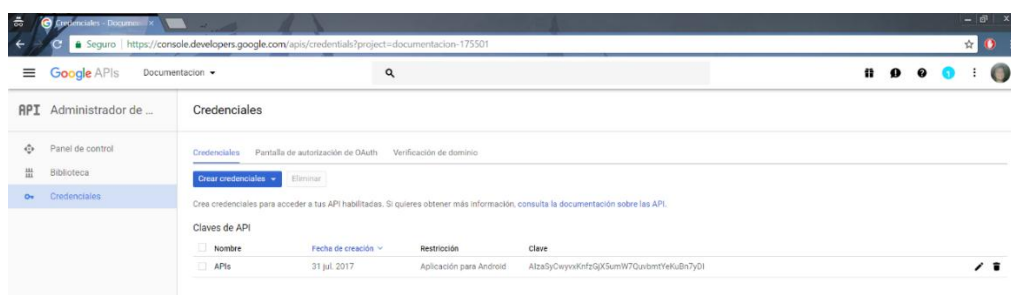


**Ilustración 3.27** Google Cloud Platform  
**Fuente:** elaboración propia

Al habilitar la API, es necesario crear una credencial para el uso de la misma, esta es la clave la cual se ocupa como medio de comunicación como se lo explicó anteriormente. Es importante restringir dicha credencial para que solo funcione con la aplicación que se está creando y así asegurarse el uso no autorizado y robo de cuotas.



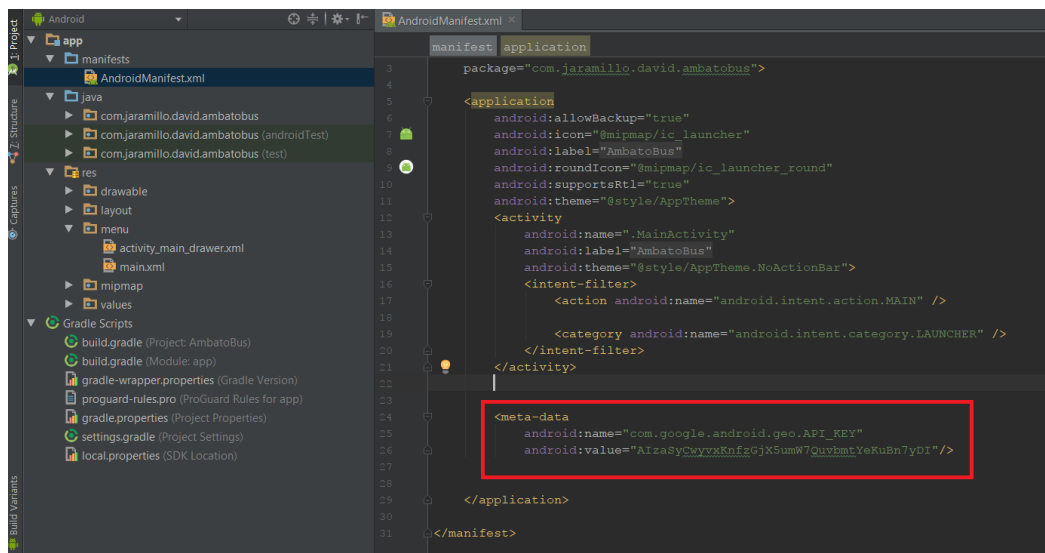
**Ilustración 3.28** Restricción de la credencial para el uso de las APIs  
Fuente: elaboración propia



**Ilustración 3.29** Credencial para el uso de las APIs  
Fuente: elaboración propia

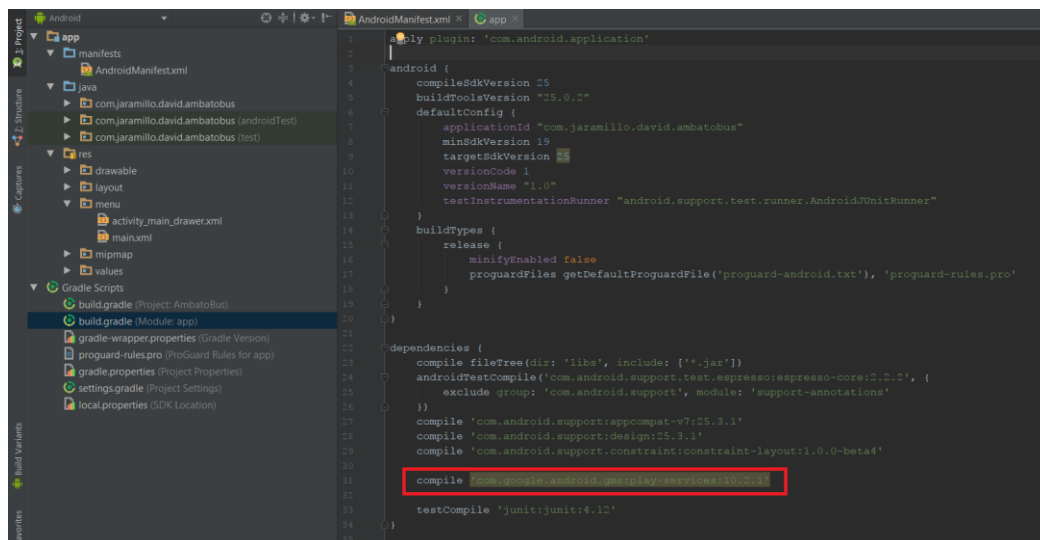
Teniendo ya la clave lista, en el archivo AndroidManifest.xml que encuentra dentro del proyecto creado en Android Studio, se agrega el elemento `<meta-`

*data*> dentro de la sección *<application>*, que sirve para agregar la credencial de la API en la aplicación.



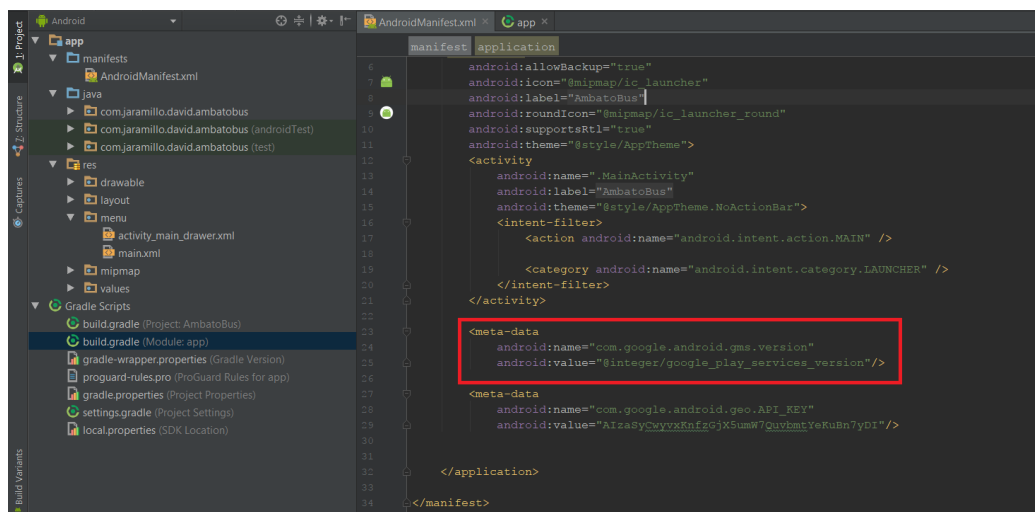
**Ilustración 3.30** Agregar credencia en el *Manifest*  
Elaborado por: Investigación

A continuación en el archivo *build.gradle(Module:app)*, se agrega la librería de *Google Play Services* con la versión actualizada y se procede a sincronizar para que el proyecto reconozca la misma. Cabe recalcar que dicha librería es importante añadirla, ya que su objetivo es que se pueda cargar todos los elementos que permiten personalizar y así mostrar el mapa en la Aplicación.



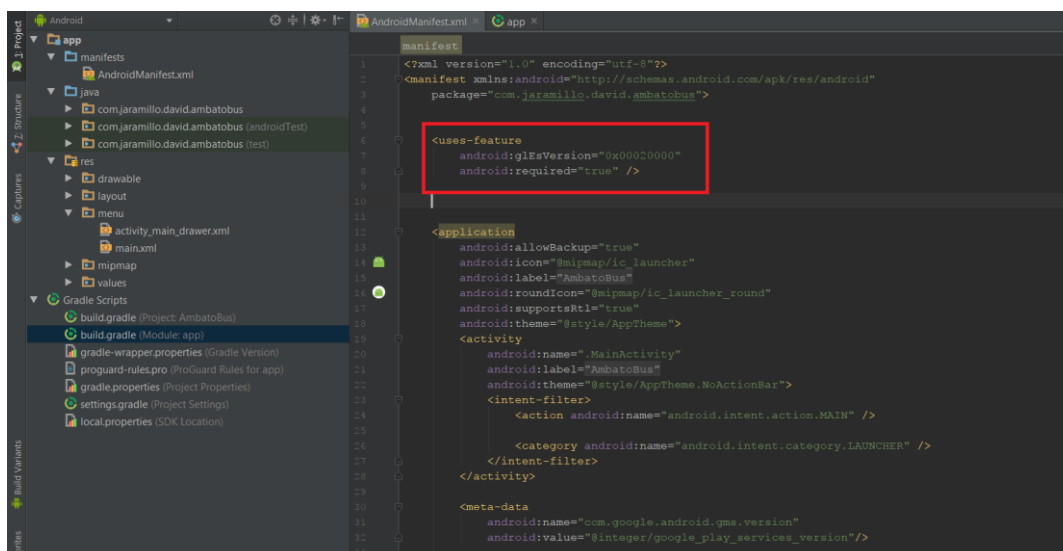
**Ilustración 3.31** Librería de Google Play Services  
Fuente: elaboración propia

Una vez sincronizada correctamente la librería de *Google Play Services*, dentro del archivo `AndroidManifest.xml`, se agrega otro elemento `<meta-data>` dentro de la sección `<application>`, que sirve para dirigir el SDK de servicios de Google Play hacia la aplicación.



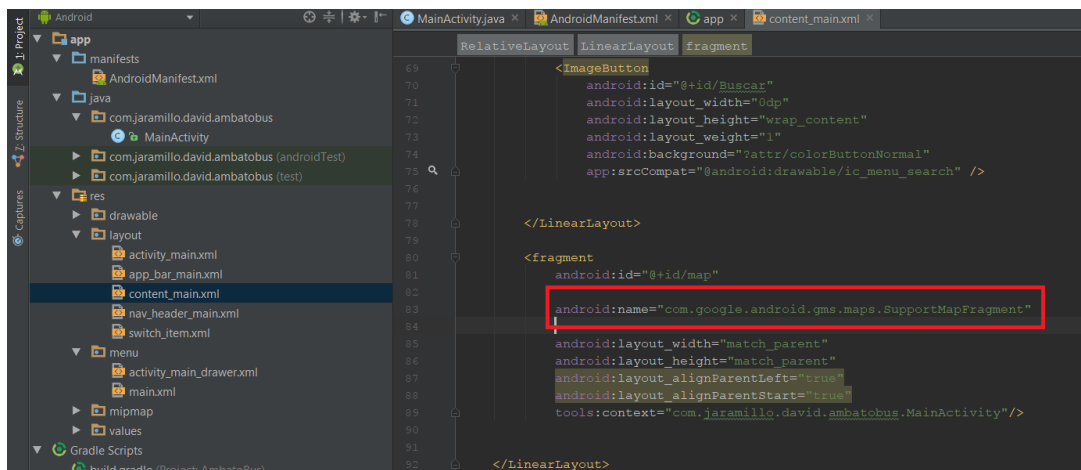
**Ilustración 3.32** Servicios de Google Play en la aplicación  
Fuente: elaboración propia

Para que se visualice correctamente el mapa, se debe agregar el soporte para gráficos 2D y 3D de alto rendimiento con la Open Graphics Library (OpenGL), específicamente, la API de OpenGL ES versión 2 dentro del archivo AndroidManifest.xml, como lo menciona en [29].



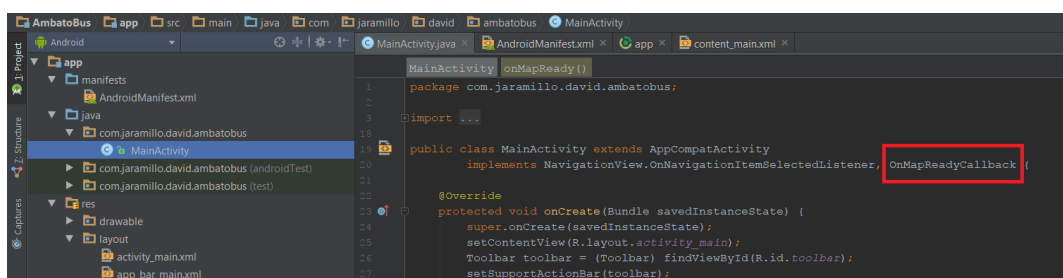
**Ilustración 3.33** Open Graphics Library  
Fuente: elaboración propia

En el archivo content\_main.xml, en la sección del *fragment*, se agrega el componente SupportMapFragment, con el objetivo de que el mapa aparezca dentro del *fragment*.



**Ilustración 3.34** Componente SupportMapFragment  
Fuente: elaboración propia

Dentro del MainActivity.java en la clase MainActivity, se añade una nueva interfaz pública llamada *OnMapReadyCallback*. Al momento de insertarlo, es importante añadir el método que pide, llamado *onMapReady*. Para que funcione correctamente dicho método es necesario que en el dispositivo estén instalados los servicios de Google Play, caso contrario el usuario debe instalarlos.



**Ilustración 3.35** Interfaz pública OnMapReadyCallback  
Fuente: elaboración propia

```

41 MainActivity.onNavigationItemSelected()
42 // as you specify a parent activity in AndroidManifest.xml.
43 int id = item.getItemId();
44
45 //noinspection SimplifiableIfStatement
46 if (id == R.id.action_settings) {
47     return true;
48 }
49
50 return super.onOptionsItemSelected(item);
51
52 }
53
54 @SuppressWarnings("StatementWithEmptyBody")
55 @Override
56 public boolean onNavigationItemSelected(MenuItem item) {
57     // Handle navigation view item clicks here.
58     int id = item.getItemId();
59
60     DrawerLayout drawer = (DrawerLayout) findViewById(R.id.drawer_layout);
61     drawer.closeDrawer(GravityCompat.START);
62     return true;
63 }
64
65
66 @Override
67 public void onMapReady(GoogleMap googleMap) {
68 }
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

**Ilustración 3.36** Método `onMapReady`  
Fuente: elaboración propia

Dentro de la clase `onCreate`, se crea una nueva instancia del `fragment` para que la App sepa en donde se debe visualizar el mapa.

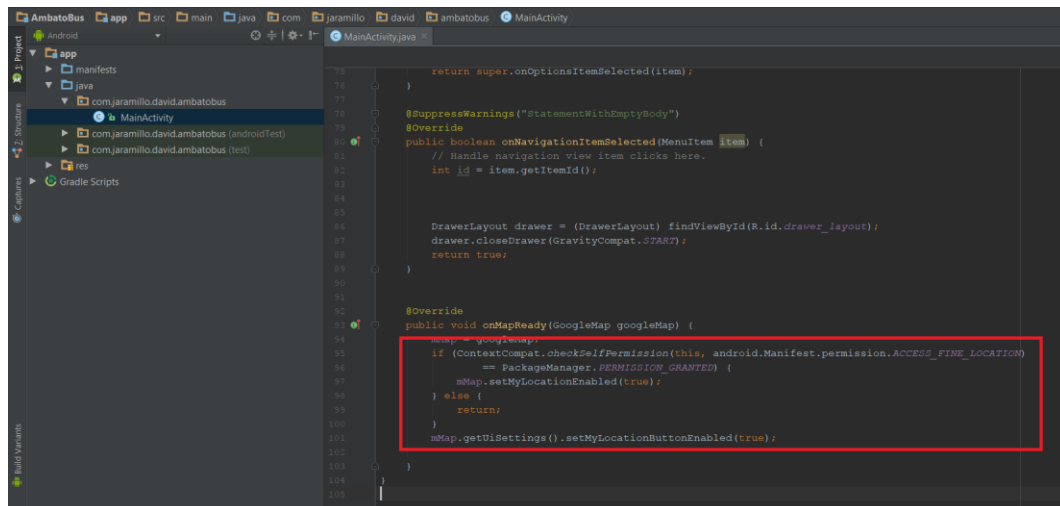
```

21 MainActivity.onCreate()
22 super.onCreate(savedInstanceState);
23 setContentView(R.layout.activity_main);
24 Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
25 setSupportActionBar(toolbar);
26
27 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.drawer_layout);
28 ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(
29     this, drawer, toolbar, "Open navigation drawer", "Close navigation drawer");
30 drawer.setDrawerListener(toggle);
31 toggle.syncState();
32
33 NavigationView navigationView = (NavigationView) findViewById(R.id.nav_view);
34 navigationView.setNavigationItemSelectedListener(this);
35
36 SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()
37     .findFragmentById(R.id.map);
38 mapFragment.getMapAsync(this);
39
40 }
41
42
43
44 @Override
45 public void onBackPressed() {
46     DrawerLayout drawer = (DrawerLayout) findViewById(R.id.drawer_layout);
47     if (drawer.isDrawerOpen(GravityCompat.START)) {
48         drawer.closeDrawer(GravityCompat.START);
49     } else {
50         super.onBackPressed();
51     }
52 }
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

**Ilustración 3.37** Instanciación del `fragment`  
Fuente: elaboración propia

En el método `onMapReady`, se añade un control IU (Interfaz de Usuario) para mostrar el botón de *My Location*.



```

75         return super.onOptionsItemSelected(item);
76     }
77
78     @SuppressWarnings("StatementWithEmptyBody")
79     @Override
80     public boolean onOptionsItemSelected(MenuItem item) {
81         // Handle navigation view item clicks here.
82         int id = item.getItemId();
83
84
85         DrawerLayout drawer = (DrawerLayout) findViewById(R.id.drawer_layout);
86         drawer.closeDrawer(GravityCompat.START);
87         return true;
88     }
89
90     @Override
91     public void onMapReady(GoogleMap googleMap) {
92         mMap = googleMap;
93         if (ContextCompat.checkSelfPermission(this, android.Manifest.permission.ACCESS_FINE_LOCATION)
94             == PackageManager.PERMISSION_GRANTED) {
95             mMap.setLocationEnabled(true);
96         } else {
97             return;
98         }
99
100         mMap.getUiSettings().setMyLocationButtonEnabled(true);
101     }
102
103
104
105

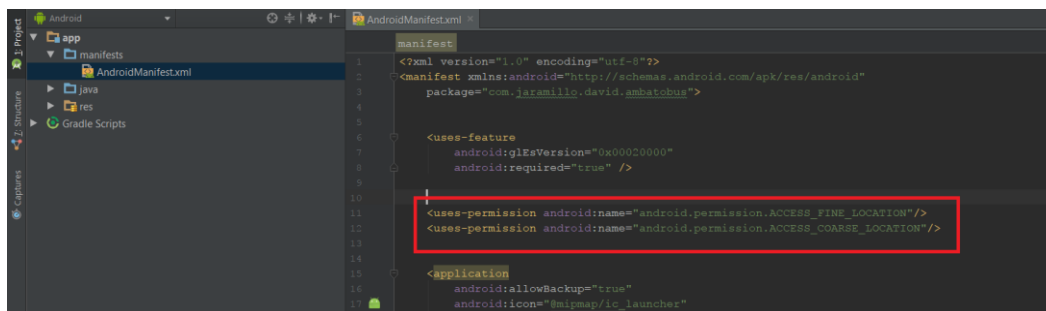
```

**Ilustración 3.38** Control IU My Location  
Fuente: elaboración propia

Para que la aplicación pueda acceder a las coordenadas de ubicación del usuario, es necesario darle 2 diferentes permisos que se los lista a continuación:

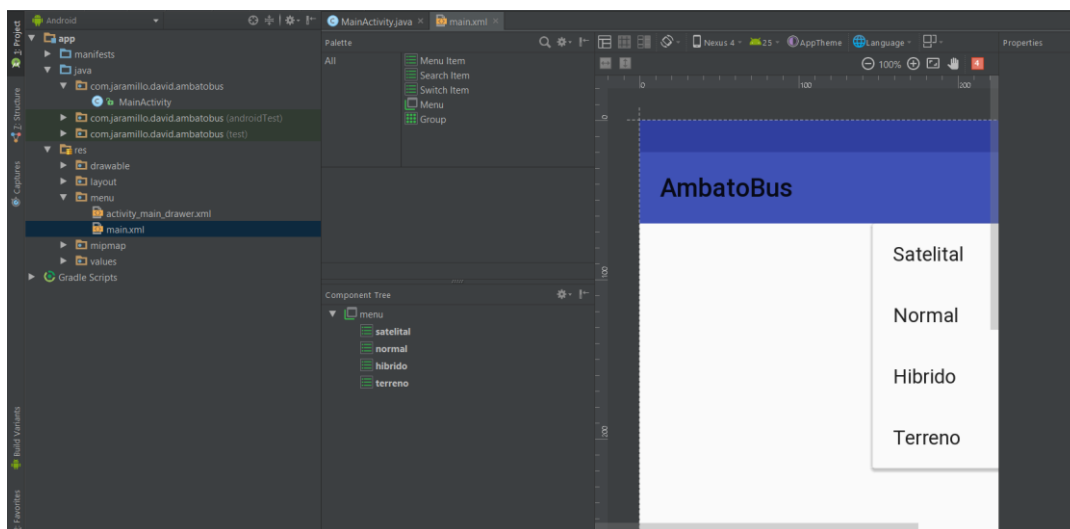
- **ACCESS\_COARSE\_LOCATION:** Permite que la App acceda a una ubicación aproximada.
- **ACCESS\_FINE\_LOCATION:** Permite que la App acceda a una ubicación precisa.

Esos permisos se los aumenta en el `androidManifest.xml` antes de la sección de `<application>`.



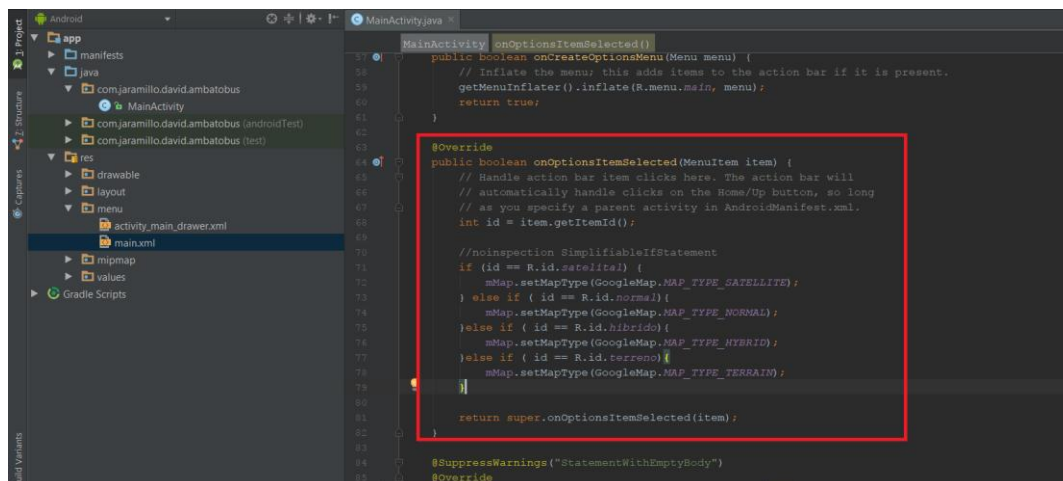
**Ilustración 3.39** Permisos de ubicación  
Fuente: elaboración propia

Para mostrar los diferentes tipos de mapas, en la aplicación debe existir un menú secundario en el cual se muestren los tipos de mapas a elegir. Para ello se edita en el archivo main.xml, los 4 nombres de los mapas que Google Maps nos ofrece.



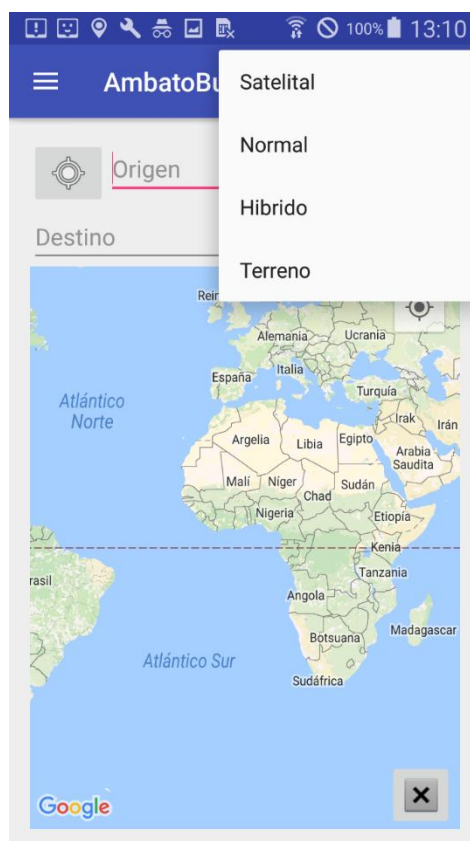
**Ilustración 3.40** Diseño del menú secundario  
Fuente: elaboración propia

En la clase *onOptionsItemSelected*, se codifica la función de cada uno de los ítems creados en el menú anterior, los cuales son los tipos de mapas que se pueden visualizar en la aplicación:



**Ilustración 3.41** Tipos de mapas  
Fuente: elaboración propia

Al realizar todo correctamente, en el dispositivo, la aplicación deberá lucir como se muestra a continuación:



**Ilustración 3.42** Menú secundario  
Fuente: elaboración propia

### 3.3.3.2.3 Codificación de la App e implementación de las demás APIs

Se identifican las cuatro variables básicas que se necesitan para crear la conexión entre la App y la *Web Services*.

- SoapAction: es un encabezado de HTTP que se envía al comienzo del mensaje SOAP real.



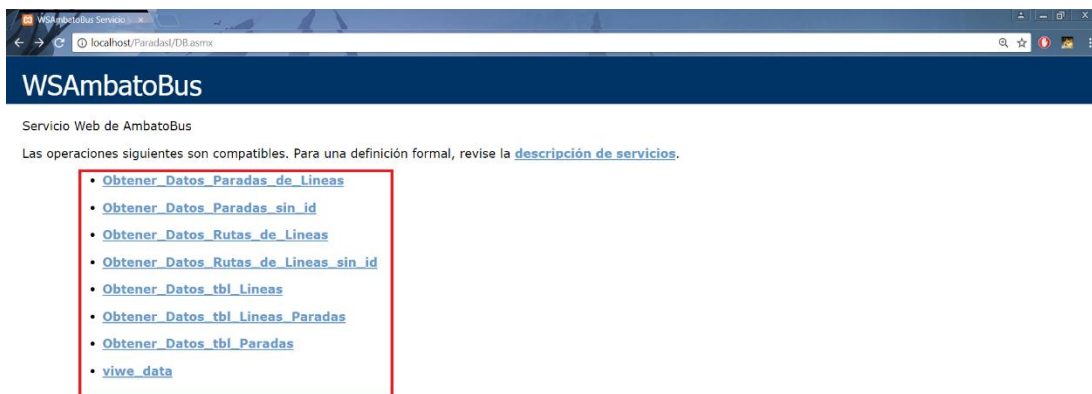
**Ilustración 3.43 SoapAction**  
Fuente: elaboración propia

- NameSpece: esta información se la puede encontrar en la descripción de servicios de la *web services*.



**Ilustración 3.44 NameSpece**  
Fuente: elaboración propia

- **Method\_Name**: es el nombre del método o de la consulta que se va a realizar a la base de datos.



**Ilustración 3.45** Method\_Name  
Fuente: elaboración propia

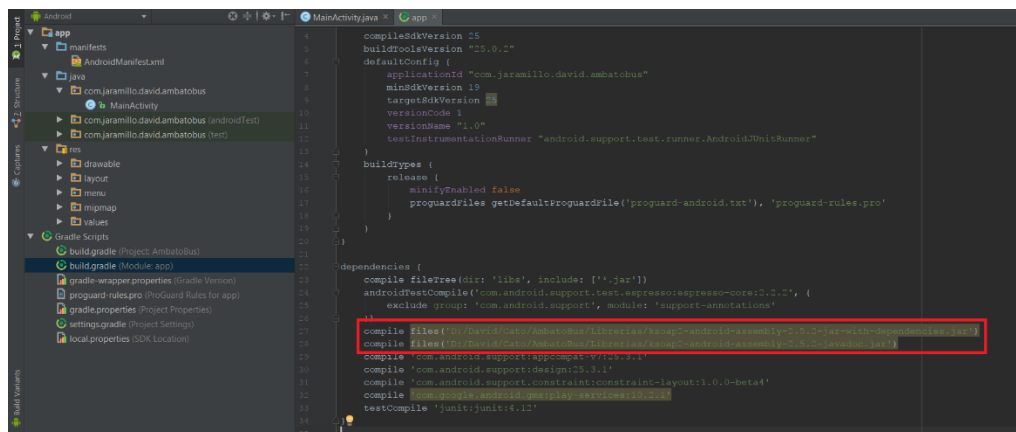
- **Url**: es el enlace el cual se utiliza para entrar a la web services, y se toma en cuenta desde el nombre de la carpeta creada para almacenar la web service. Se reemplaza la palabra *localhost* por la dirección IP con la que el servidor esté conectada a internet, seguido del puerto con el que sea la puerta de enlace del IIS.



**Ilustración 3.46** Url  
Fuente: elaboración propia

Se deben incorporar dos librerías especializadas para poder gestionar el SOAP en Android Studio y para sincronizarlas se deben agregar sus rutas de ubicación en el archivo `build.gradle(Module:app)`, como se muestra a continuación:

- ksoap2-android-assembly-2.5.2-jar-with-dependencies.jar
- ksoap2-android-assembly-2.5.2-javadoc.jar

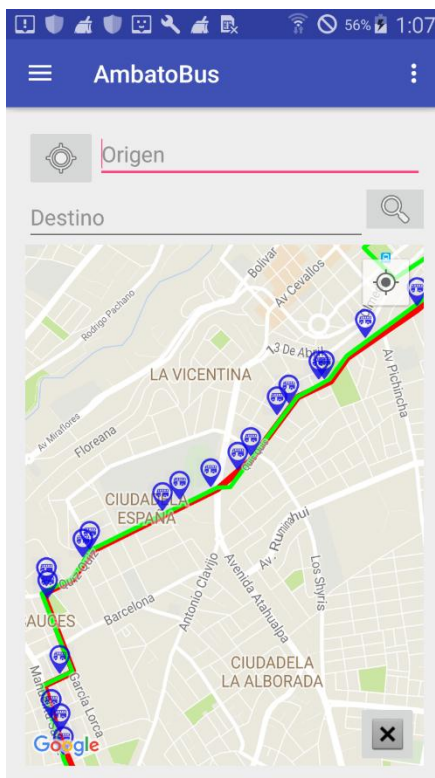


**Ilustración 3.47** Librerías para la gestión del SOAP  
Fuente: elaboración propia

Se crea una clase tipo *Boolean* para la conexión y almacenar la información dentro de vectores y matrices, y esto sirve para leer las coordenadas y así trazar las rutas en el mapa, este tipo de clases se las crea por cada uno de los métodos existentes en la *Web Services (WS)*.

Se hace el uso de una clase *AsyncTask* para realizar la conexión a la *WS*, almacenamiento y lectura de los datos adquiridos en un subproceso y así mismo mostrar los resultados en la interfaz del usuario sin tener que manipular dichos subprocesos. Cabe recalcar que, en dicha clase, es donde se desarrolla la lógica pertinente para que el usuario pueda observar la ruta del bus, sepa que bus coger y cuál es la parada más cercana, y trazar la ruta de cada una de las líneas con las paradas como información general.

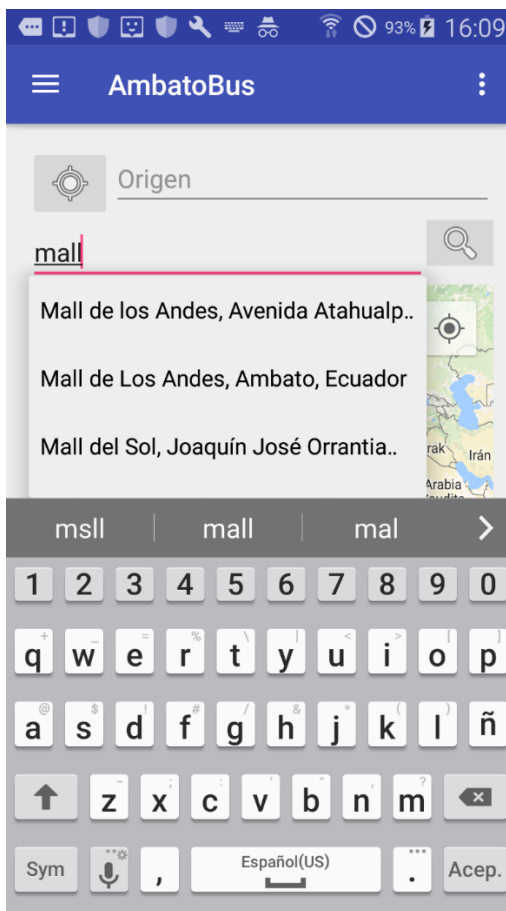
La clase *AsyncTask* es ejecutada al momento de que el usuario presione algún ítem del menú principal o del botón para buscar la mejor ruta.



**Ilustración 3.48** Resultado de rutas  
**Fuente:** elaboración propia

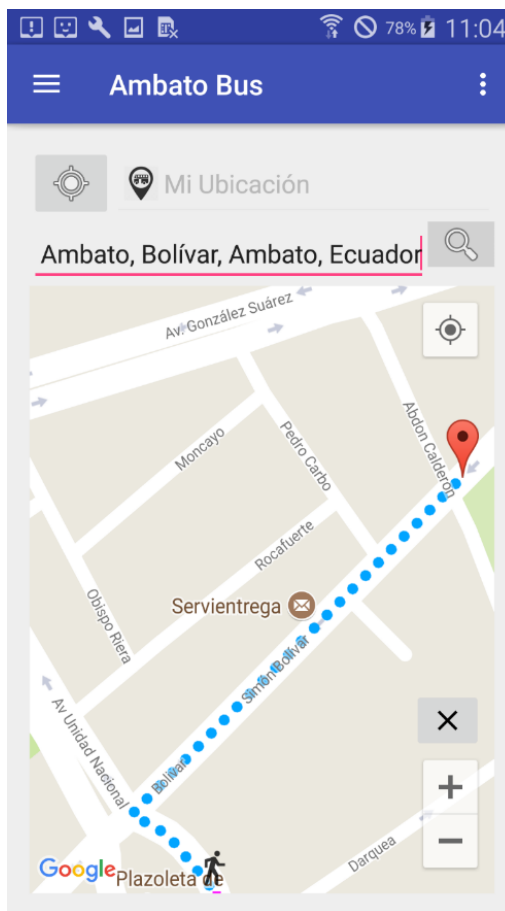
Se necesita insertar 2 APIs más en la aplicación:

- Google Places API for Android: esta API se utiliza para el autocomplementado de sitios cuya función es completar automáticamente el nombre o la dirección de un lugar específico mientras el usuario lo escribe. El código y las clases correspondientes fueron encontradas en [30].



**Ilustración 3.49** *Google Places API for Android*  
**Fuente:** elaboración propia

- Google Maps Directions API: se utiliza esta API para planear una ruta que sirve para dirigirle al usuario desde su punto de partida hasta la parada más cercana. El código y las clases correspondientes fueron encontradas en [31].



**Ilustración 3.50** Google Maps Directions API  
**Fuente:** elaboración propia

- Google Maps Geocoding API: esta API permite saber las coordenadas exactas de cualquier tipo de lugar o dirección que se encuentre dentro de la base de datos de Google. El código y las clases correspondientes fueron encontradas en [32].

### 3.3.3.3 Día de lanzamiento

Una vez culminado el día de trabajo, llega el día de presentación del tercer entregable y para ello es esencial informar que la aplicación cumpla con todos los resultados esperados mencionados en los test de aceptación. Para ello se realiza un resumen en donde se especifiquen las deficiencias encontradas durante la tercera iteración como se muestra a continuación.

Proyecto: AmbatoBus

Fecha: 10.02.2017

**Tabla 3.8** Lista de resumen de deficiencias 1

Lista de Resumen de Deficiencias		
Requerimientos	Descripción	Acción Correctiva
1	El color del diseño de la App no es atractivo.	Cambiar el color de la interfaz del usuario.
2	El botón de limpiar el mapa se sobrepone con los botones de zoom.	Reubicar el botón de limpiar el mapa.
3	Falta de controles en el mapa para mejor manejo.	Aumentar botones de zoom y brújula en el mapa.
4	El mapa muestra otros lugares que no se encuentran dentro de la ciudad de Ambato.	Limitar la visualización del mapa que se muestre solo la ciudad de Ambato.
5	No se puede interactuar correctamente en el mapa	Limitar los niveles de zoom.
6	No se sabe el significado de cada trazo de las rutas.	Aumentar referencias de que significa cada uno de los trazos.
7	No se muestra que línea se debe coger.	Aumentar un aviso del número de línea que se debe coger.

Fuente: elaboración propia

### 3.4 Fase de estabilización

#### 3.4.1 Día de planificación

Teniendo presente la lista de resumen de deficiencias del segundo y tercer entregable realizado en la anterior fase de producción, la fase de estabilización de la aplicación cuenta con los mismos pasos, es necesario llenar el 1er taller post iteración con el objetivo de mejorar iterativamente el proceso de desarrollo de la aplicación. Esto incluye la identificación de fortalezas y debilidades en proceso, así como la generación de acciones de mejora para la siguiente iteración, basándose en la lista de resumen de deficiencias.

## 1er Taller post iteración

Proyecto: AmbatoBus

Fecha: 13/02/2017

Iteración #: 1

Participante: David Jaramillo

Tabla 3.9 1er Taller post iteración

Tema de mejora: Diseño de interfaz				
Problema	Acción	Responsable	Seguimiento del plan	Realización
El color del diseño de la App no es atractivo.	Cambiar el color de la interfaz del usuario tonos más atractivos.	David Jaramillo	Revisar las combinaciones de colores que en Android Studio ofrece.	El usuario podrá apreciar un diseño más atractivo.
Tema de mejora: Diseño de interfaz				
Problema	Acción	Responsable	Seguimiento del plan	Realización
El botón de limpiar el mapa se sobrepone con los botones de zoom.	Reubicar el botón de limpiar el mapa.	David Jaramillo	Se verifica la posición en la que este ubicado el botón para poder cambiar.	El usuario podrá usar correctamente todos los botones que se muestran en la aplicación.
Tema de mejora: Botones en el mapa implementado				
Problema	Acción	Responsable	Seguimiento del plan	Realización
Falta de controles en el mapa para mejor manejo.	Aumentar botones de zoom y brújula en el mapa mediante los controles IU.	David Jaramillo	Revisar la página oficial de <i>Google Developers</i> para <i>Android Studio</i>	El usuario tendrá a su disposición funciones para poder manipular el mapa de mejor manera.
Tema de mejora: Limitaciones en el mapa				
Problema	Acción	Responsable	Seguimiento del plan	Realización
El mapa muestra otros lugares que no se encuentran dentro de la ciudad de Ambato.	Limitar la visualización del mapa para que se muestre solo la ciudad de Ambato.	David Jaramillo	Revisar la página oficial de <i>Google Developers</i> para <i>Android Studio</i>	El usuario solo tendrá disponible el mapa de la ciudad de Ambato.
Tema de mejora: Limitación en las funciones de los botones del mapa				
Problema	Acción	Responsable	Seguimiento del plan	Realización

Se muestran lugares fuera de los límites de la ciudad.	Limitar los niveles de zoom.	David Jaramillo	Revisar la página oficial de <i>Google Developers</i> para <i>Android Studio</i>	El usuario tendrá restricciones al momento de utilizar el zoom en el mapa.
Tema de mejora: indicaciones de significados de cada línea trazada				
Problema	Acción	Responsable	Seguimiento del plan	Realización
No se sabe el significado de cada trazo de las rutas.	Aumentar referencias como imágenes de que significa cada uno de los trazos.	David Jaramillo	Diseñar imágenes para representar cada uno de los trazos en el mapa.	El usuario sabrá que significa cada uno de los trazos que se muestre en el mapa de la aplicación.
Tema de mejora: indicaciones de líneas a coger				
Problema	Acción	Responsable	Seguimiento del plan	Realización
No se muestra que línea se debe coger.	Aumentar un aviso del número de línea que se debe tomar mediante un mensaje temporal en pantalla.	David Jaramillo	Revisar la página oficial de <i>Google Developers</i> para <i>Android Studio</i>	El usuario sabrá que línea debe coger para llegar a su destino.

Fuente: elaboración propia

Los test de aceptación se formaron basándose en la información impartida del test de post iteración se muestran a continuación:

**Tabla 3.10** Test de aceptación 7

TEST DE ACEPTACIÓN
TEST ID: 7
HISTORIA: diseño de la interfaz gráfica
FECHA DE REDACCION: 14.02.2017
FECHAS DE EJECUCIÓN: 15.02.2017
DESCRIPCIÓN: <ul style="list-style-type: none"> <li>Se debe mostrar la interfaz con un mejor tono.</li> <li>Se debe mostrar los botones correctamente posicionados.</li> <li>Se debe mostrar guías del significado de cada trazo.</li> </ul>
RESULTADO ESPERADO: <ul style="list-style-type: none"> <li>Llamar la atención del usuario con la nueva interfaz.</li> <li>El usuario tendrá una aplicación mejor estructurada.</li> <li>El usuario sabrá el significado de cada uno de los trazos que se muestren en el mapa.</li> </ul>

Fuente: elaboración propia

**Tabla 3.11** Test de aceptación 8

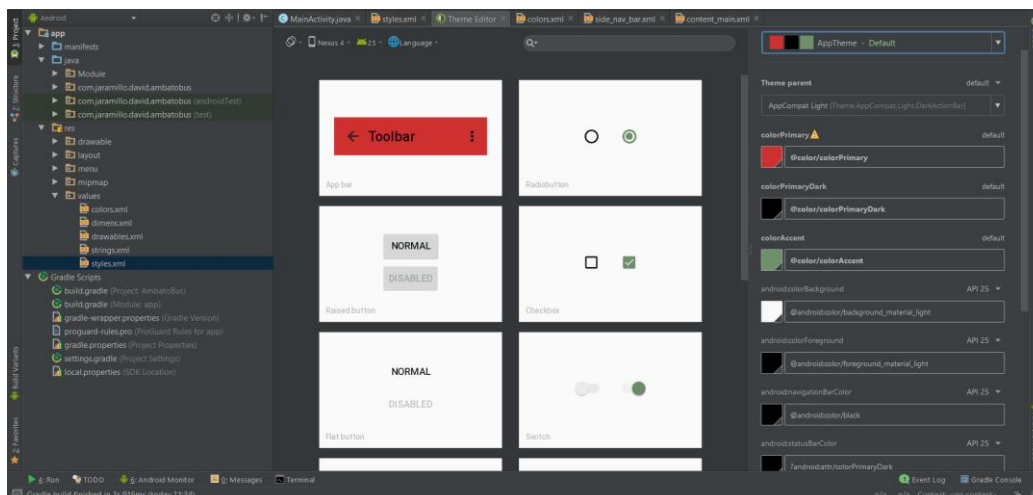
TEST DE ACEPTACIÓN
TEST ID: 8
HISTORIA: implementación de nuevas funcionalidades y correcciones de la App
FECHA DE REDACCION: 14.02.2017
FECHAS DE EJECUCIÓN: 15.02.2017
DESCRIPCIÓN: <ul style="list-style-type: none"> <li>• El mapa debe mostrar las funciones de zoom y brújula.</li> <li>• En el mapa solo se deberá visualizar solo la ciudad de Ambato.</li> <li>• La funcionalidad del zoom debe limitarse.</li> <li>• Se debe mostrar un identificador de que línea debe tomar el usuario.</li> </ul>
RESULTADO ESPERADO: <ul style="list-style-type: none"> <li>• El usuario podrá interactuar de mejor manera con el mapa.</li> <li>• El usuario no se desorientará.</li> <li>• El usuario sabrá que bus tomar para llegar a su destino final.</li> </ul>

**Fuente:** elaboración propia

### 3.4.2 Día de trabajo

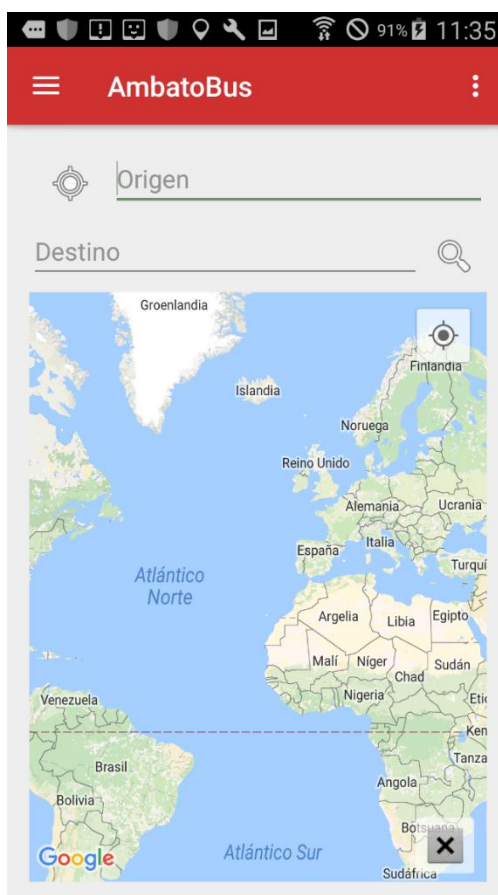
#### 3.4.2.1 Diseño de la interfaz gráfica

Se toma la decisión de darle colores acordes los que representan al Municipio de Ambato. Con la ayuda del *Theme Editor* se cambiaron los colores principales de la aplicación que conlleva al *App bar*, *Widgets* y cajas de texto, recalando que no se cambiarán los botones pertenecientes a las funcionalidades del mapa, y por último se cambiará el color del *Navigation Drawer*, que viene a ser el menú principal de la App.



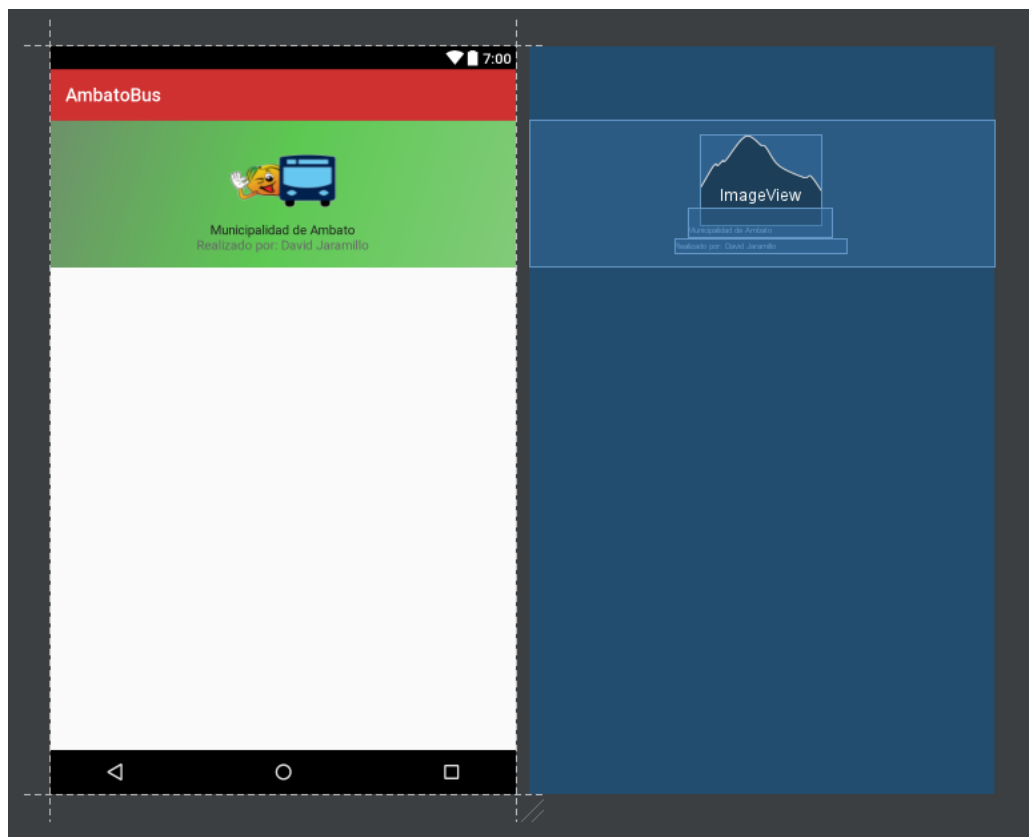
**Ilustración 3.51** Interfaz del Theme Editor  
Fuente: elaboración propia

A continuación, se muestra el resultado de la pantalla principal con los nuevos colores:



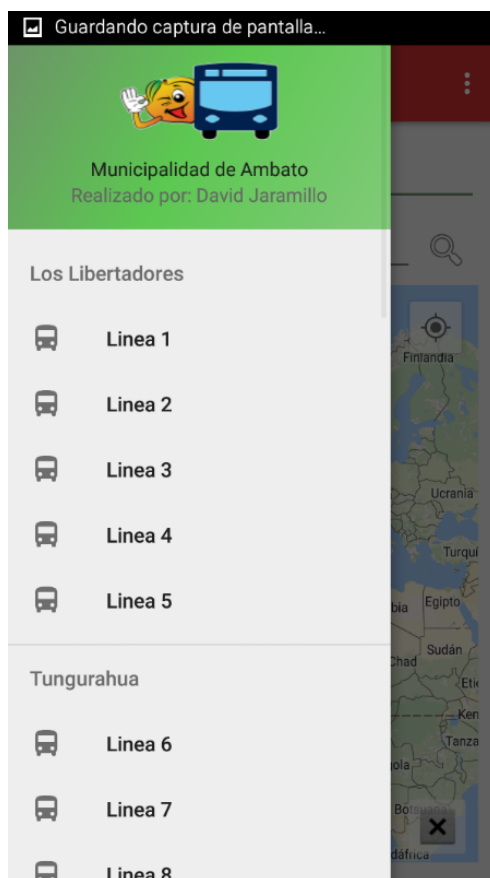
**Ilustración 3.52** Resultado de los nuevos colores de la interfaz  
Fuente: elaboración propia

En la ilustración siguiente, se muestra el maquetado con los nuevos colores dados al menú principal:



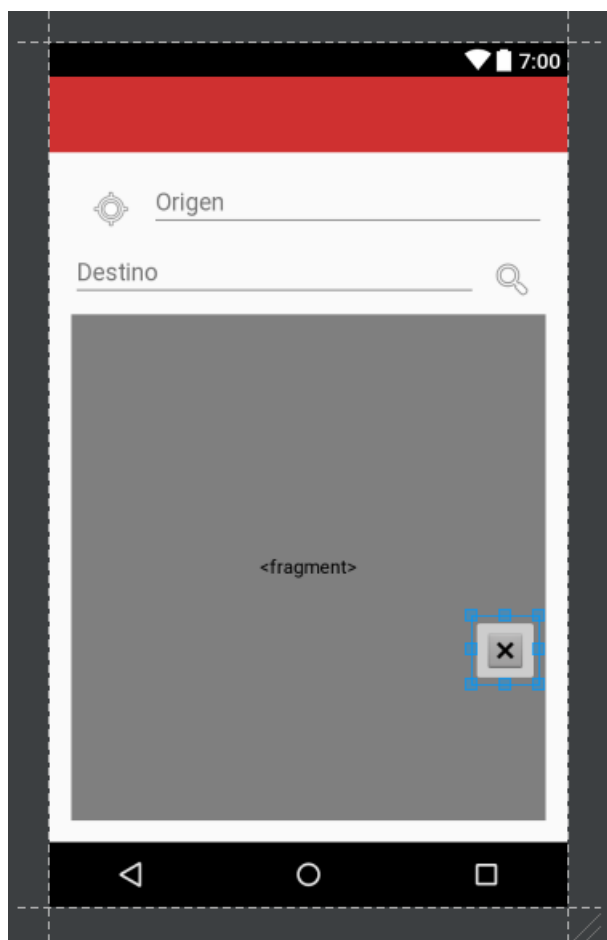
**Ilustración 3.53** Maquetado del nuevo menú principal  
**Fuente:** elaboración propia

A continuación, se muestra el resultado del menú principal con los nuevos colores y la reacomodación de los elementos:



**Ilustración 3.54** Diseño del nuevo menú principal  
**Fuente:** elaboración propia

Se procede a reubicar el botón de limpiar mapa, ya que al momento de implementar los propios botones de zoom que ofrece el API de Google Maps para Android, se superponen con el botón de limpiar mapa.

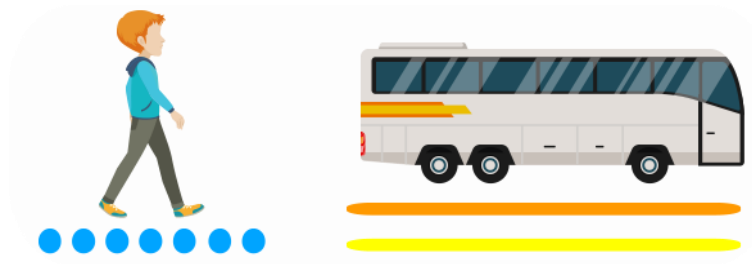


**Ilustración 3.55** Reubicación del botón para limpiar el mapa  
**Fuente:** elaboración propia

Para mejor entendimiento del usuario sobre los significados de cada trazo que se muestre en el mapa, se tomó la decisión de añadir imágenes indicando el significado de los mismos.



**Ilustración 3.56** Guía de usuario  
**Elaborada por:** Investigador



**Ilustración 3.57** Guía de usuario  
Fuente: elaboración propia

### 3.4.2.2 Implementación de nuevas funcionalidades y correcciones de la App

Es necesario que el usuario tenga mayor facilidad de manipular el mapa, para ello se implementan diversas funcionalidades en el mismo, como se mencionó anteriormente, son conocidos como controles IU. Se implementan los botones de Zoom y brújula.

```

MainActivity onMapReady()
    } else if (id == R.id.linea22) {
        GuiaPrint.setVisibility(View.INVISIBLE);
        mMap.clear();
        new asyntodos().execute("22");
        cargardatos2.clear();
        cargardatos3.clear();
        GuiaInfo.setVisibility(View.VISIBLE);
    }

    DrawerLayout drawer = (DrawerLayout) findViewById(R.id.drawer_layout);
    drawer.closeDrawer(GravityCompat.START);
    return true;
}

@Override
public void onMapReady(GoogleMap googleMap) {
    mMap = googleMap;
    if (ContextCompat.checkSelfPermission(this, android.Manifest.permission
        == PackageManager.PERMISSION_GRANTED) {
        mMap.setMyLocationEnabled(true);
    } else {
        return;
    }
    mMap.getUiSettings().setMyLocationButtonEnabled(true);
    mMap.getUiSettings().setCompassEnabled(true);
    mMap.getUiSettings().setZoomControlsEnabled(true);
}

```

**Ilustración 3.58** Controles IU zoom y brújula  
Fuente: elaboración propia

Añadiendo el código adecuado, el resultado es el siguiente:

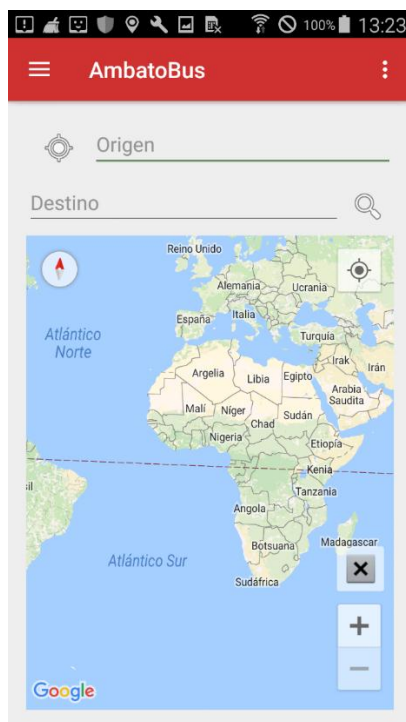


Ilustración 3.59 Botones de zoom y brújula  
**Fuente:** elaboración propia

Al ser una aplicación que se implementa solo en la ciudad de Ambato, se muestra a los usuarios solo la ciudad antes mencionada.

```

@Override
public void onMapReady(GoogleMap googleMap) {
    mMap = googleMap;

    //Limitar el mapa solo para ver Ambato
    LatLngBounds LimiteAmbato = new LatLngBounds(
        new LatLng(-1.341724, -78.721963), new LatLng(-1.197878, -78.564285));
    mMap.setLatLngBoundsForCameraTarget(LimiteAmbato);
    // Mover camara Al centro de Ambato
    LatLng ambato = new LatLng(-1.2543408, -78.6228504);
    mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(ambato, 13));

    //controles IU
    if (ContextCompat.checkSelfPermission(this, android.Manifest.permission.ACCESS_FINE_LOCATION)
        == PackageManager.PERMISSION_GRANTED) {
        mMap.setMyLocationEnabled(true);
    } else {
        return;
    }
    mMap.getUiSettings().setMyLocationButtonEnabled(true);
    mMap.getUiSettings().setCompassEnabled(true);
    mMap.getUiSettings().setZoomControlsEnabled(true);
}
}

```

**Ilustración 3.60** Limitación de la ciudad de Ambato en el mapa  
Fuente: elaboración propia

Para facilitar el uso de la aplicación, el zoom del mapa se limita a la ciudad de Ambato.

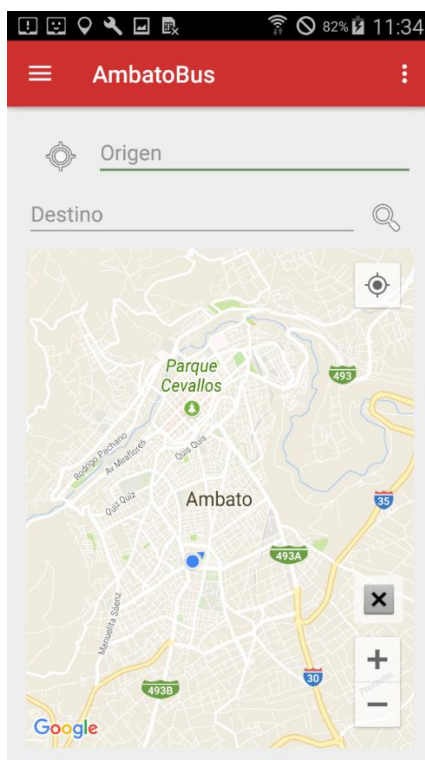
```

MainActivity onMapReady()
804     return true;
805 }
806
807
808
809 @Override
810 public void onMapReady(GoogleMap googleMap) {
811     mMap = googleMap;
812     //Limitar el mapa solo para ver Ambato
813     LatLngBounds LimiteAmbato = new LatLngBounds(
814         new LatLng(-1.341724, -78.721963), new LatLng(-1.197878, -78.564285));
815     mMap.setLatLngBoundsForCameraTarget(LimiteAmbato);
816     // Mover camara Al centro de Ambato
817     LatLng ambato = new LatLng(-1.2543408, -78.6228504);
818     mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(ambato, 13));
819
820     //Limitar zoom
821     mMap.setMinZoomPreference(13.0f);
822     mMap.setMaxZoomPreference(20.0f);
823
824
825
826     //controles IU
827     if (ContextCompat.checkSelfPermission(this, android.Manifest.permission.ACCESS_FINE_LOCATION)
828         == PackageManager.PERMISSION_GRANTED) {
829         mMap.setMyLocationEnabled(true);
830     } else {
831         return;
832     }
833     mMap.getUiSettings().setMyLocationButtonEnabled(true);
834     mMap.getUiSettings().setCompassEnabled(true);
835     mMap.getUiSettings().setZoomControlsEnabled(true);
836 }
837 }
838

```

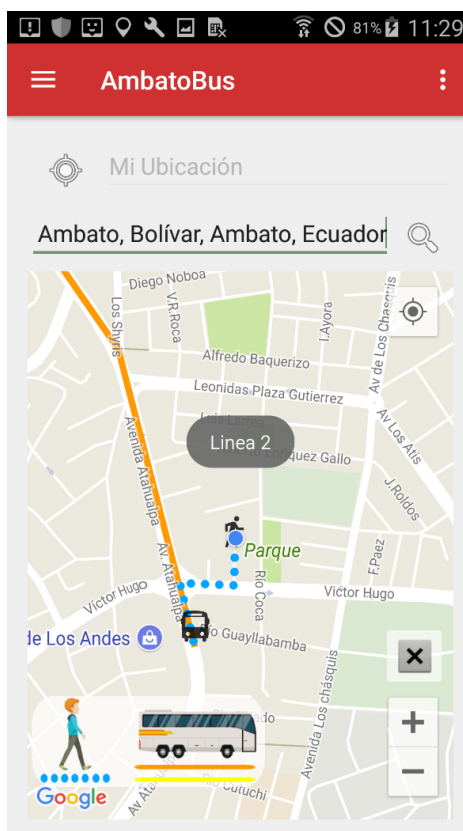
**Ilustración 3.61** Limitación del Zoom en el mapa  
Fuente: elaboración propia

Una vez que se realizan los cambios adecuados en el código, el resultado es el siguiente:



**Ilustración 3.62** Visualización de la ciudad de Ambato en el mapa  
**Fuente:** elaboración propia

Para dar una mejor referencia al usuario, se muestra un mensaje temporal de la línea de bus que deberá tomar para llegar a su destino.



**Ilustración 3.63** Mensaje temporal de la línea a tomar  
Fuente: elaboración propia

### 3.4.3 Día de lanzamiento

En este día se decide realizar las pruebas de la aplicación con algunos usuarios que se prestaron para las mismas. Esto consiste en que las personas utilicen la aplicación y de esta manera hagan notar errores que se presenten durante la ejecución de la App, en el Anexo 3 se muestran a los usuarios que probaron la aplicación. A continuación, se presenta un cuadro de deficiencias de las respuestas obtenidas de cada usuario.

Proyecto: AmbatoBus

Fecha: 27.02.2017

**Tabla 3.12** Lista de resumen de deficiencias 2

Lista de Resumen de Deficiencias		
Usuario: Estefanía Jaramillo		
Requerimientos	Descripción	Acción Correctiva
1	El TextView debería estar bloqueado con las palabras Mi Ubicación	Inhabilitar el TextView del origen para evitar la confusión de los usuarios y no puedan ingresar direcciones, nombres de calles y lugares.
2	Debe existir un mensaje permanente en pantalla de que bus se debe tomar o que, al momento de dar clic en el marcador con la figura de un bus, indique el número de línea que se deberá coger	En el marcador con la figura de un bus, se añadirá un título con el número de línea que el usuario deberá tomar para llegar a su destino, que, al momento de pulsarlo, se hará visible.
3	Debe mostrar el tiempo que dura al trasladarse y la distancia que se debe recorrer	En la parte inferior se mostrará una pantalla emergente con la información del tiempo y distancia.

**Fuente:** elaboración propia

Proyecto: AmbatoBus

Fecha: 27.02.2017

**Tabla 3.13** Lista de resumen de deficiencias 3

Lista de Resumen de Deficiencias		
Usuario: Gabriel Romero		
Requerimientos	Descripción	Acción Correctiva
1	El TextView debería estar bloqueado con las palabras Mi Ubicación	Inhabilitar el TextView del origen para evitar la confusión de los usuarios y no puedan ingresar direcciones, nombres de calles y lugares.
2	El mensaje de guía del TextView del destino no logra hacerle entender	Se cambia el mensaje por uno que llame más la atención e indique la función que el TextView posee.

	perfectamente al usuario que debe escribir.	
3	El mensaje temporal de la línea de bus que deberá tomar el usuario no es muy claro.	Se debe cambiar el mensaje para que el usuario entienda de mejor manera que debe hacer con esa línea de bus.
4	La imagen que indica el significado de que trazo es para caminar y que trazo indica la ruta del bus, no debería existir ya que se muestran marcadores que indican lo mismo.	Se elimina la guía que se implementó a la aplicación, para que el usuario no se confunda.
5	No existe una referencia para indicar a que cooperativa de bus pertenece la línea de bus a tomar.	Se debe implementar en el marcador la información de que cooperativa pertenece la línea a tomar.

**Fuente:** elaboración propia

Proyecto: AmbatoBus

Fecha: 27.02.2017

**Tabla 3.14** Lista de resumen de deficiencias 4

Lista de Resumen de Deficiencias		
Usuario: Ana Jaramillo		
Requerimientos	Descripción	Acción Correctiva
1	El TextView debería estar bloqueado con las palabras Mi Ubicación	Inhabilitar el TextView del origen para evitar la confusión de los usuarios y no puedan ingresar direcciones, nombres de calles y lugares.
2	Existe confusión en el color de líneas trazadas al momento de dar clic en cualquier opción del menú principal.	Se debe intercambiar el color de las líneas de buses que indican información en general de todas las rutas de buses.
3	Debe mostrar el tiempo que dura al trasladarse y la distancia que se debe recorrer	En la parte inferior se mostrará una pantalla emergente con la información del tiempo y distancia.

**Fuente:** elaboración propia

Proyecto: AmbatoBus

Fecha: 27.02.2017

**Tabla 3.15** Lista de resumen de deficiencias 5

Lista de Resumen de Deficiencias		
Usuario: Luis Felipe Zambrano		
Requerimientos	Descripción	Acción Correctiva
1	La imagen que indica el significado de que trazo es para caminar y que trazo indica la ruta del bus, no debería existir ya que se muestran marcadores que indican lo mismo.	Se elimina la guía que se implementó a la aplicación, para que el usuario no se confunda.
2	En el menú principal, no existen referencias por los lugares que pasa la línea de bus.	Se debe añadir información en donde se especifique los lugares principales por donde pasa la línea de bus.
3	El mensaje temporal de la línea de bus que deberá tomar el usuario no es muy claro.	Se debe cambiar el mensaje para que el usuario entienda de mejor manera que debe hacer con esa línea de bus.

**Fuente:** elaboración propia

Proyecto: AmbatoBus

Fecha: 27.02.2017

**Tabla 3.16** Lista de resumen de deficiencias 6

Lista de Resumen de Deficiencias		
Usuario: Elizabeth Hidalgo		
Requerimientos	Descripción	Acción Correctiva
1	El <i>TextView</i> debería estar bloqueado con las palabras Mi Ubicación	Inhabilitar el <i>TextView</i> del origen para evitar la confusión de los usuarios y no puedan ingresar direcciones, nombres de calles y lugares.
2	La imagen que indica el significado de que trazo es para caminar y que trazo indica la ruta del bus, no debería existir ya que se muestran marcadores que indican lo mismo.	Se elimina la guía que se implementó a la aplicación, para que el usuario no se confunda.
3	No existe una referencia para indicar a que cooperativa de bus	Se debe implementar en el marcador la información de que cooperativa pertenece la línea a tomar.

	pertenece la línea de bus a tomar.	
--	------------------------------------	--

Fuente: elaboración propia

### 3.5 Fase de pruebas y lanzamiento

#### 3.5.1 Día de planificación

Teniendo presente la lista de resumen de deficiencias de cada uno de los usuarios escogidos, es necesario llenar el 2do taller post iteración con el objetivo de mejorar la aplicación para que esté lista al momento de subirla a la tienda de aplicación *Play Store* que ofrece *Google*.

2do Taller post iteración

Proyecto: AmbatoBus

Fecha: 28/02/2017

Iteración #: 2

Participante: David Jaramillo

**Tabla 3.17** 2do Taller post iteración

Tema de mejora: Diseño de interfaz				
Problema	Acción	Responsable	Seguimiento del plan	Realización
Confusión con en el casillero de origen, debería estar bloqueado.	Inhabilitar el <i>TextView</i> del origen para evitar la confusión de los usuarios y no puedan ingresar direcciones, nombres de calles y lugares.	David Jaramillo	Revisar la página oficial de <i>Google Developers</i> para <i>Android Studio</i>	El usuario no se confunde ni tiene dudas del uso del <i>TextView</i> origen.
El mensaje de guía del <i>TextView</i> del destino no logra hacerle entender perfectamente	Se cambiará el mensaje por uno que llame más la atención e indique la función que el <i>TextView</i> posee.	David Jaramillo	Revisar la página oficial de <i>Google Developers</i> para <i>Android Studio</i>	El usuario sabe perfectamente que función cumple el <i>TextView</i> destino

al usuario que debe escribir.				
La imagen que indica el significado de que trazo es para caminar y que trazo indica la ruta del bus, no debería existir ya que se muestran marcadores que indican lo mismo.	Se eliminará la guía que se implementó a la aplicación, para que el usuario no se confunda.	David Jaramillo	Revisar la codificación ya hecha para eliminar el código que concierne a la imagen.	El usuario tendrá una mejor aplicación.
Existe confusión en el color de líneas trazadas al momento de dar clic en cualquier opción del menú principal.	Editar y remplazar la imagen de guía con los colores intercambiados.	David Jaramillo	Revisar la codificación ya hecha para editar la imagen.	El usuario no se confundirá en el color de las líneas trazadas.
Tema de mejora: información que se muestra al usuario				
Problema	Acción	Responsable	Seguimiento del plan	Realización
Al dar clic en el marcador con la figura de bus, debe aparecer la línea de bus a tomar	En el marcador con la figura de un bus, se añadirá un título con el número de línea que el usuario deberá tomar para llegar a su destino, que, al momento de pulsarlo, se hará visible.	David Jaramillo	Revisar la página oficial de <i>Google Developers</i> para <i>Android Studio</i>	El usuario tiene a su disposición la línea de bus que debe tomar.
El mensaje temporal de la línea de bus que deberá tomar el usuario, no es muy claro.	Se debe cambiar el mensaje que se muestra para que el usuario sepa que bus debe hacer.	David Jaramillo	Revisar la codificación ya hecha para editar el mensaje.	El usuario entiende de mejor manera que debe hacer con la línea de bus que se muestre en el mensaje.
Debe mostrar el tiempo que dura al trasladarse y la distancia que se debe recorrer	En la parte inferior se mostrará una pantalla emergente con la información del tiempo y distancia.	David Jaramillo	Realizar el algoritmo correcto para calcular tiempos y distancias y así poder mostrar al usuario.	El usuario podrá saber el tiempo y distancia a recorrer.
No existe una referencia para indicar a que cooperativa de bus pertenece la	Se deberá implementar en el marcador la información de que cooperativa	David Jaramillo	Realizar una nueva consulta para saber la cooperativa que pertenece	El usuario tendrá a su disposición más información sobre la línea

línea de bus a tomar.	pertenece la línea a tomar.		la línea de bus a tomar.	que deberá tomar para llegar a su destino.
Existe confusión en el color de líneas trazadas al momento de dar clic en cualquier opción del menú principal.	Editar el código para que el color de las rutas sean las correctas.	David Jaramillo	Revisar la codificación ya hecha para editar los colores de las rutas.	El usuario entenderá de mejor manera las rutas trazadas en la aplicación.
Tema de mejora: Subir la App en la <i>PlayStore</i>				
Problema	Acción	Responsable	Seguimiento del plan	Realización
La aplicación no se encuentra en la tienda de aplicación que ofrece Google llamada <i>PlayStore</i> .	Subir la aplicación en la tienda <i>PlayStore</i> .	David Jaramillo	Realizar los pasos necesarios para subir la aplicación en la tienda.	El usuario podrá descargarse la aplicación en su dispositivo <i>Android</i> desde la tienda <i>PlayStore</i> .

Fuente: elaboración propia

A continuación, se muestran los test de aceptación:

**Tabla 3.18** Test de aceptación 8

TEST DE ACEPTACIÓN
TEST ID: 8
HISTORIA: correcciones de la interfaz gráfica
FECHA DE REDACCION: 06.03.2017
FECHAS DE EJECUCIÓN: 07.03.2017
DESCRIPCIÓN: <ul style="list-style-type: none"> <li>• Se debe inhabilitar el <i>TextView</i> de origen.</li> <li>• Se debe expresar correctamente los mensajes de guía hacia el usuario</li> <li>• No deben duplicarse las guías al usuario.</li> <li>• Se corregirá los colores de la guía y de las rutas que se muestra al momento de ingresar a las opciones del menú principal.</li> </ul>
RESULTADO ESPERADO: <ul style="list-style-type: none"> <li>• El usuario no se confundirá ni tendrá dudas del uso del <i>TextView</i> origen.</li> <li>• El usuario entenderá correctamente lo que debe realizar.</li> <li>• No habrá incomodidades al momento de usar la aplicación.</li> <li>• El usuario no se confundirá en el color de las líneas trazadas.</li> </ul>

Fuente: elaboración propia

**Tabla 3.19** Test de aceptación 9

TEST DE ACEPTACIÓN
TEST ID: 9
HISTORIA: correcciones de la información que se muestra al usuario
FECHA DE REDACCION: 06.03.2017
FECHAS DE EJECUCIÓN: 07.06.2017
DESCRIPCIÓN: <ul style="list-style-type: none"> <li>• Se muestra la información sobre el tiempo y distancia a recorrer del usuario.</li> <li>• El número de la línea de bus a tomar deberá aparecer al dar clic en el marcador correspondiente.</li> <li>• Al dar clic en el marcador con la figura del bus, se deberá mostrar la cooperativa de bus que pertenece la línea a tomar por el usuario.</li> </ul>
RESULTADO ESPERADO: <ul style="list-style-type: none"> <li>• El usuario podrá saber el tiempo y distancia a recorrer.</li> <li>• El usuario tiene a su disposición el número de la línea de bus que debe tomar.</li> <li>• El usuario se podrá guiar de mejor manera al momento de tomar el bus que lo llevará a su destino.</li> </ul>

**Elaborado por:** Investigador

**Tabla 3.20** Test de aceptación 10

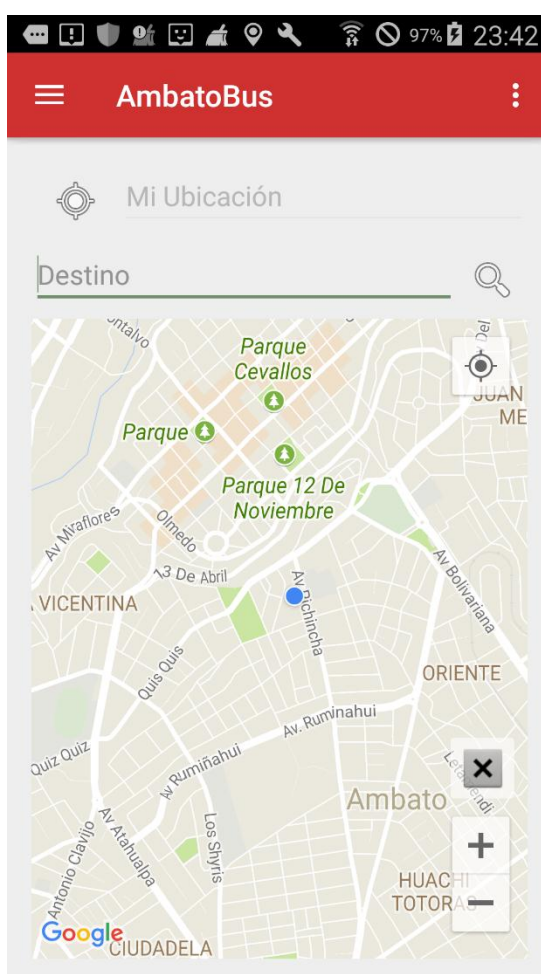
TEST DE ACEPTACIÓN
TEST ID: 10
HISTORIA: subir la aplicación en la <i>PlayStore</i>
FECHA DE REDACCION: 06.03.2017
FECHAS DE EJECUCIÓN: 07.06.2017
DESCRIPCIÓN: <ul style="list-style-type: none"> <li>• Se debe subir la aplicación en la tienda de aplicación que ofrece Google llamada <i>PlayStore</i>.</li> </ul>
RESULTADO ESPERADO: <ul style="list-style-type: none"> <li>• El usuario podrá descargarse la aplicación en su dispositivo Android desde la tienda <i>PlayStore</i>.</li> </ul>

**Fuente:** elaboración propia

## 3.5.2 Día de trabajo

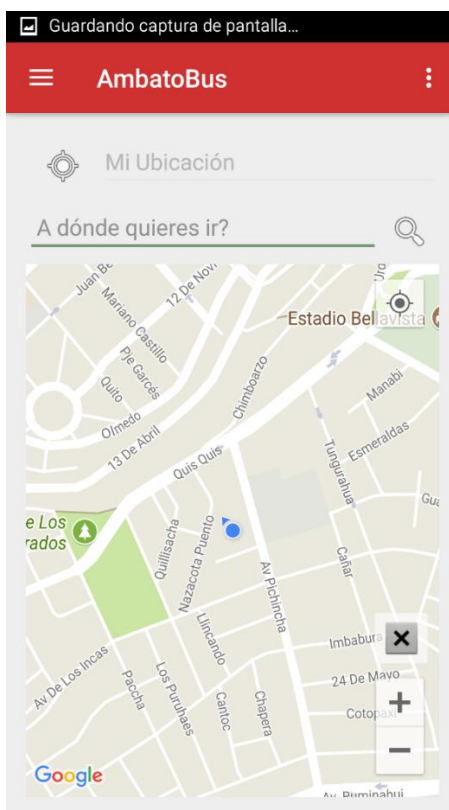
### 3.5.2.1 Correcciones de la interfaz gráfica

Al ser una aplicación que muestra la ruta desde la ubicación del usuario hacia un destino predeterminado, se decide inhabilitar el *TextView* origen, con el propósito de que el usuario no se confunda y la App funcione de mejor manera.



**Ilustración 3.64** Inhabilitar el *TextView* origen  
Fuente: elaboración propia

Un objetivo que la aplicación desea llegar, es que el usuario entienda claramente el funcionamiento de la misma, para ello, se cambian los mensajes anteriores a unos más específicos.



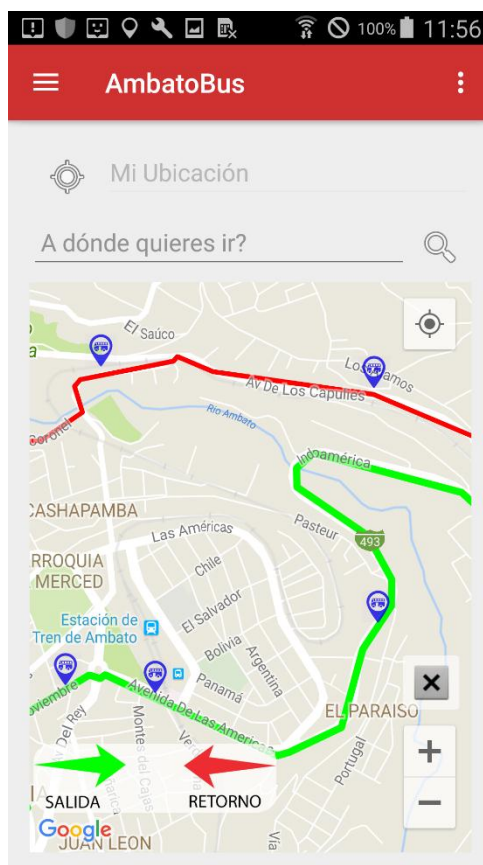
**Ilustración 3.65** Mensaje guía en el *TextView* destino  
**Fuente:** elaboración propia





**Ilustración 3.67** Eliminación de la imagen guía para el usuario  
**Fuente:** elaboración propia

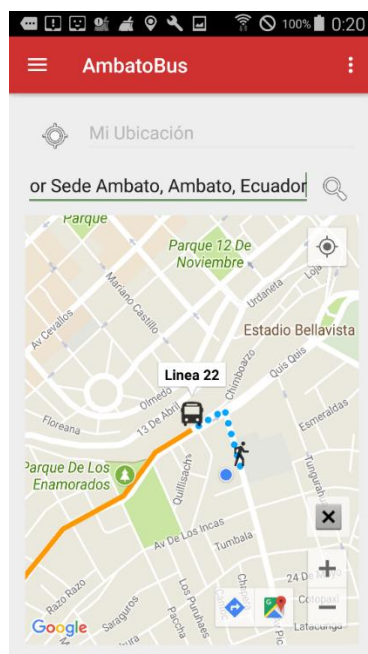
La aplicación cuenta con un menú en donde se muestran todas las rutas de los buses de la ciudad con sus respectivas paradas, al momento de trazarlas en el mapa, su simbología con los colores presentaba incomodidad con los usuarios, ya que, por costumbre, el color verde se lo representa como salida. Para ello se realiza el cambio de colores tanto en el trazo como en la imagen de guía.



**Ilustración 3.68** Cambio de color en la imagen guía para el usuario como en las rutas trazadas en el mapa  
**Fuente:** elaboración propia

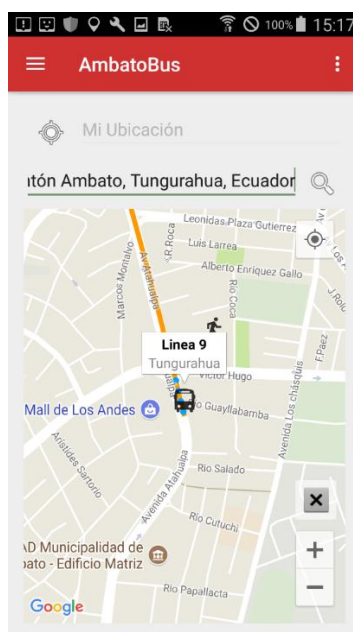
### 3.5.2.2 Correcciones de la información que se muestra al usuario

Para mejor reconocimiento de la línea a tomar y que el usuario pueda llegar a su destino, se añade un título al marcado con el número de la línea.



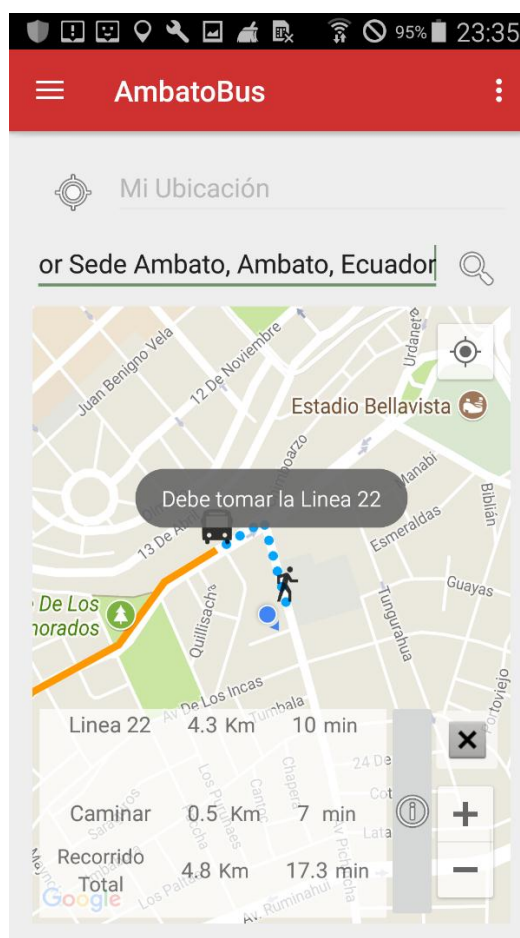
**Ilustración 3.69** Añadir la línea a tomar en el marcador  
**Fuente:** elaboración propia

Se añade el nombre de la cooperativa a la que pertenece la línea de bus.

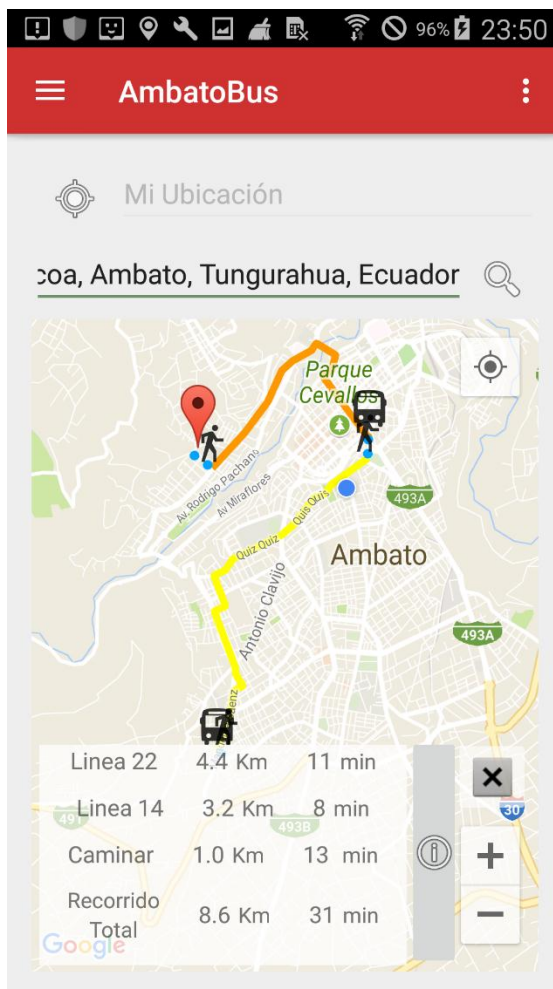


**Ilustración 3.70** Añadir la cooperativa de la línea a tomar en el marcador  
**Fuente:** elaboración propia

Para dar mayor facilidad e información al usuario, se agrega un *layout* en el cual se describirá toda la información necesaria sobre tiempos y distancias a recorrer del usuario, tanto a pie como en el recorrido del bus, tomando en cuenta que, en algunas situaciones, el usuario deberá tomar dos buses para llegar a su destino.



**Ilustración 3.71** Venta informativa con una ruta  
Fuente: elaboración propia

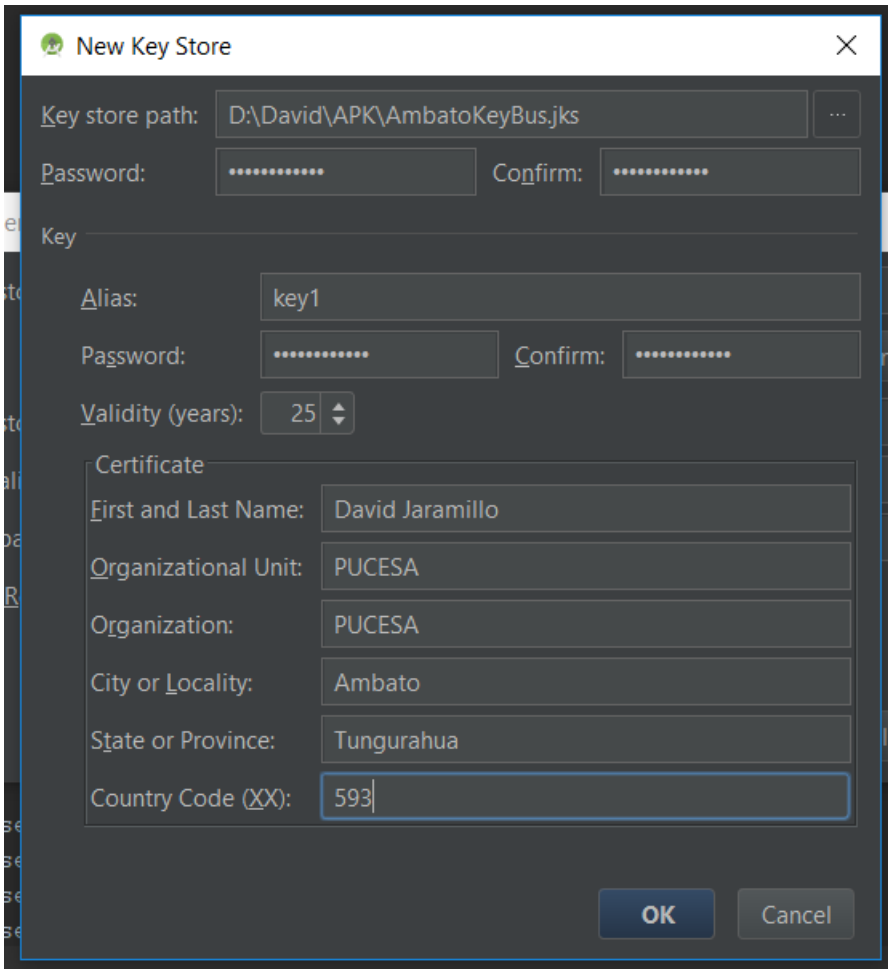


**Ilustración 3.72** Venta informativa con dos rutas  
**Fuente:** elaboración propia

### 3.5.2.3 Subir la aplicación en la *PlayStore*.

Antes de subir la aplicación a la tienda, es necesario exportar el APK desde *Android Studio* para subirlo y esté disponible. Para ello se siguen los siguientes pasos:

- Se crea una nueva *Key store path*, en donde se tiene que especificar algunos datos del desarrollador como se observa a continuación:



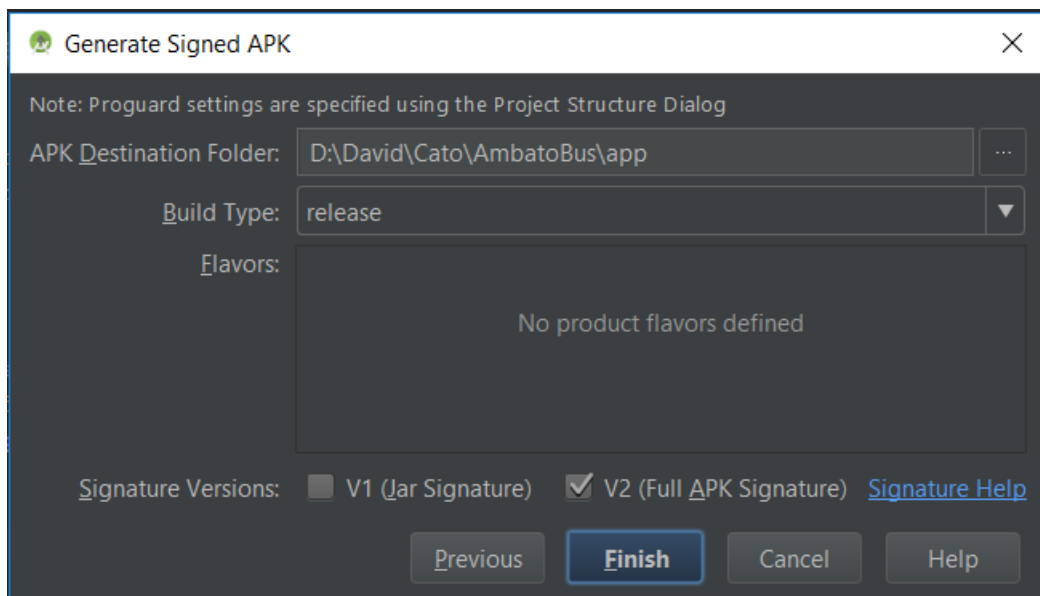
The image shows a 'New Key Store' dialog box with the following fields and values:

- Key store path: D:\David\APK\AmbatoKeyBus.jks
- Password: [masked]
- Confirm: [masked]
- Key section:
  - Alias: key1
  - Password: [masked]
  - Confirm: [masked]
- Validity (years): 25
- Certificate section:
  - First and Last Name: David Jaramillo
  - Organizational Unit: PUCESA
  - Organization: PUCESA
  - City or Locality: Ambato
  - State or Province: Tungurahua
  - Country Code (XX): 593

**Ilustración 3.73** Creación de la *Key Store Path*

**Fuente:** elaboración propia

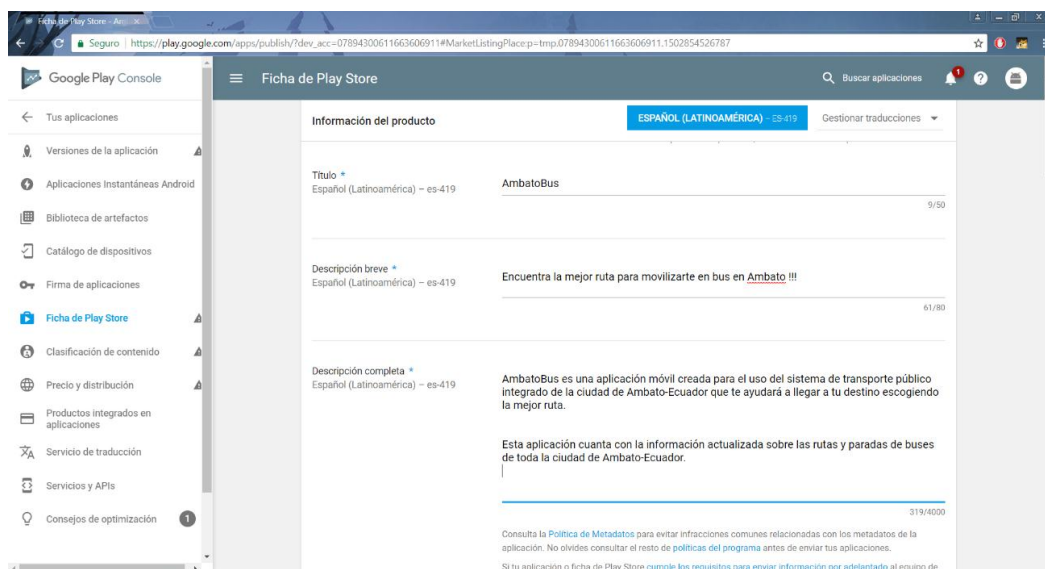
- Finalmente se crea el APK de la aplicación asignándole una ubicación para que se guarde ahí.



**Ilustración 3.74** Creación del APK de la aplicación  
**Fuente:** elaboración propia

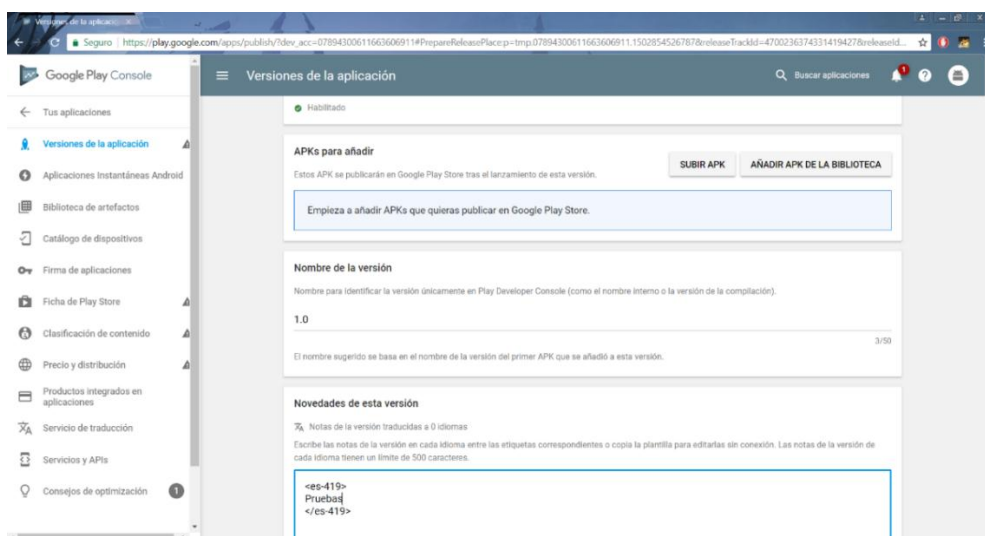
A continuación, en la página *Google Play Developer Console*, se registra con la misma cuenta de desarrollador que se usa para habilitar las APIs y se sigue los pasos que se muestran en la misma página. Después de especificar la forma de pago y llenar la información del desarrollador, se sube el apk de la aplicación y se llena la información necesaria que pide la página. Se debe llenar una ficha de Play Store, en donde se especifique:

- Título de la aplicación,
- descripción breve,
- descripción completa,
- imágenes representativas de la App,
- tipo de aplicación,
- categoría a la que pertenece,
- política de privacidad.



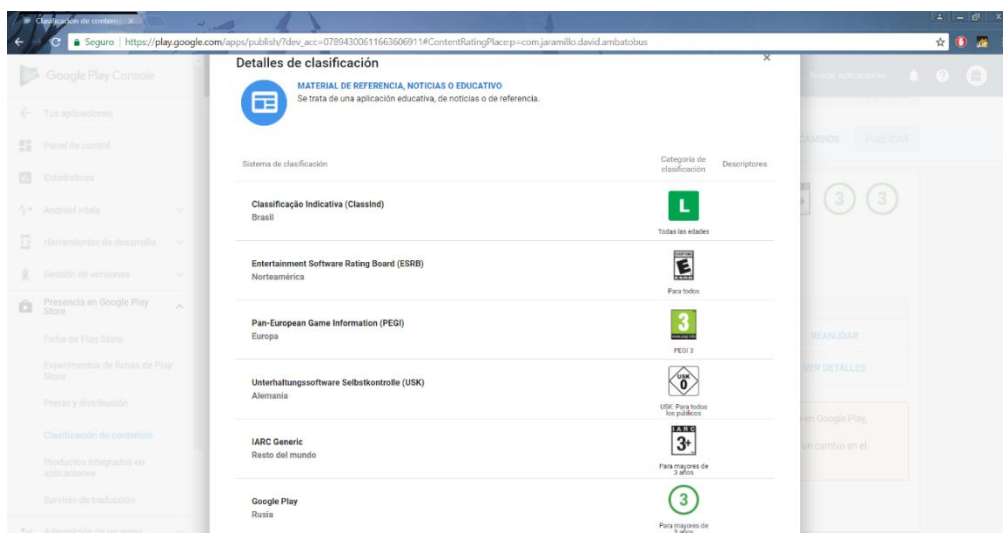
**Ilustración 3.75** Ficha de Play Store  
Fuente: elaboración propia

En el submenú versiones de la aplicación, se crea una versión de producción para subir el APK creado de la aplicación AmbatoBus.



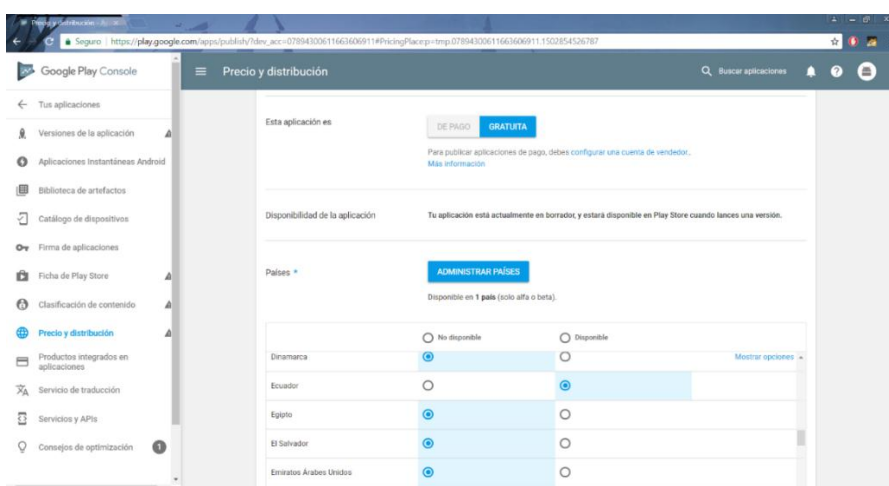
**Ilustración 3.76** Cargar el APK de la aplicación  
Fuente: elaboración propia

En la pestaña de Clasificación de contenido, se elige una categoría para la aplicación y se llena un cuestionario para verificar que la App pertenezca a dicha categoría.



**Ilustración 3.77** Clasificación de contenido de la aplicación  
Fuente: elaboración propia

En la pestaña de precios y distribución, se especifica que la aplicación es gratuita y que va dirigida solo al país de Ecuador.



**Ilustración 3.78** Precios y distribución de la aplicación  
Fuente: elaboración propia

Realizados todos los pasos, se lanzó la aplicación para que los usuarios se la puedan descargar mediante la tienda *Play Store*.



**Ilustración 3.79** Aplicación AmbatoBus lista para descargarse mediante la tienda Play Store  
**Fuente:** elaboración propia

### 3.5.3 Día de lanzamiento

Al culminar el día de trabajo, se realizaron las pruebas finales para verificar que la lista de deficiencias conseguidas por los usuarios al usar la aplicación, se hayan corregido satisfactoriamente.

Una vez culminadas las pruebas y verificando que ya no existían errores en la ejecución, se realizó la entrega de la aplicación publicada en la tienda *Play Store*, en donde está disponible para todos los usuarios del Ecuador.

En el transcurso del tiempo, se verificará la calificación y comentarios que los usuarios le den a la aplicación, combinados con sus respectivos *FeedBack*.

## **CAPÍTULO IV**

### **ANÁLISIS Y VALIDACIÓN DE RESULTADOS**

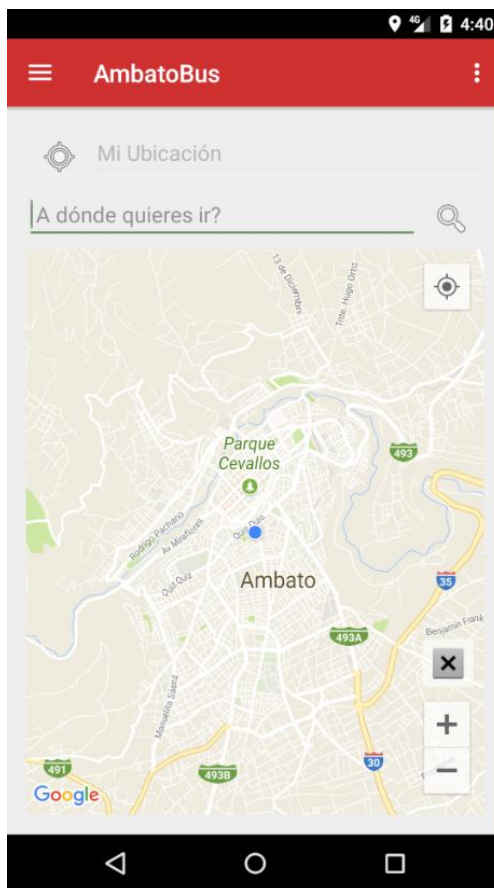
#### **4.1 Análisis de resultados**

Para culminar el presente proyecto, se analizan todos los test de deficiencias con sus respectivos resultados, abarcándolo en dos aspectos que son: la interfaz gráfica, y la codificación de la aplicación.

La aplicación funciona correctamente en todos los *Smartphone* con sistema operativo Android con versiones superiores a 4.4, por lo que las pruebas que se muestran a continuación serán con la versión 5.1.

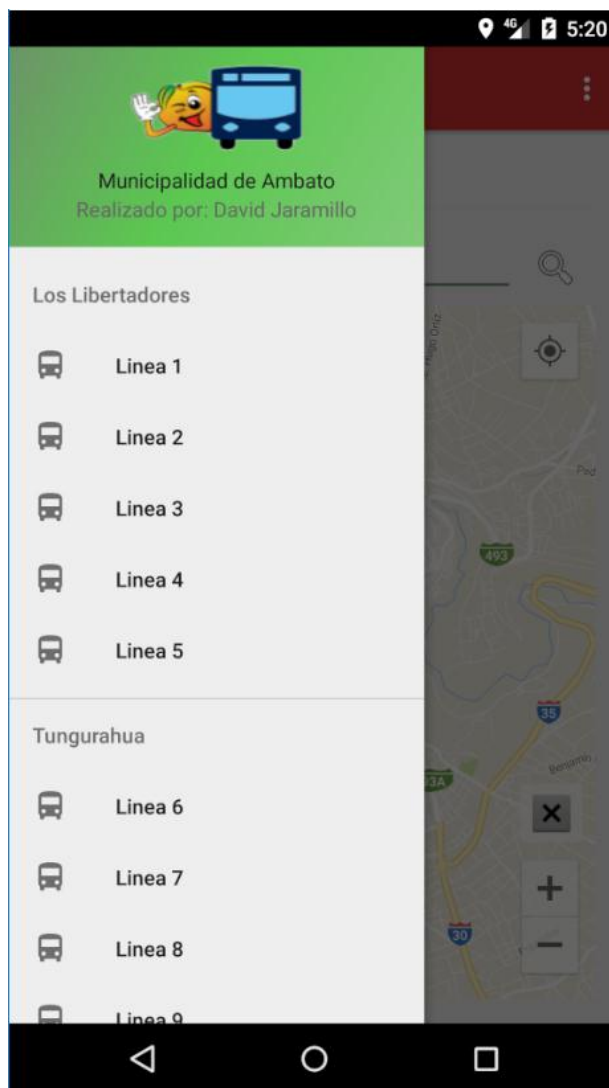
##### **4.1.1 Interfaz gráfica de la aplicación**

En la siguiente ilustración se muestra lo que observa el usuario al momento de abrir la aplicación móvil.



**Ilustración 4.1** Pruebas de la interfaz gráfica en la versión Android 5.1  
**Fuente:** elaboración propia

En la aplicación existen dos menús que cumplen diversas funciones, y al momento de corregir la aplicación basándose en los resúmenes de los test de deficiencias, se cambió los colores del menú principal y del texto, y la posición de la imagen de la aplicación fue reubicada para que se visualice un mejor diseño. A continuación, se observa el resultado obtenido al realizar los cambios:



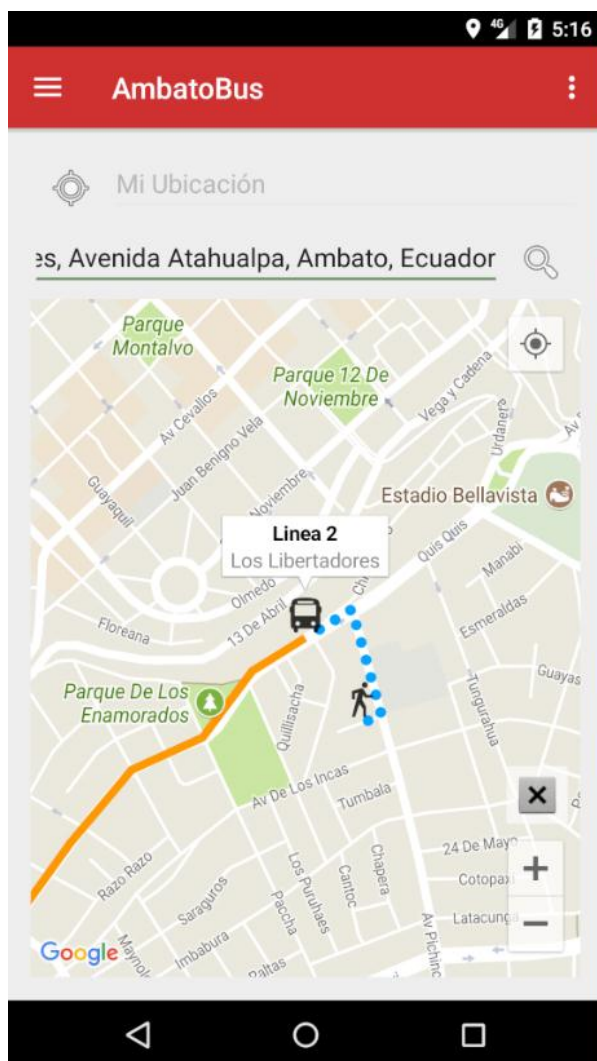
**Ilustración 4.2** Pruebas de la interfaz gráfica del menú principal en la versión Android 5.1  
**Fuente:** elaboración propia



**Ilustración 4.3** Pruebas de la interfaz gráfica del menú secundario en la versión Android 5.1  
**Fuente:** elaboración propia

#### 4.1.2 Desarrollo de la aplicación

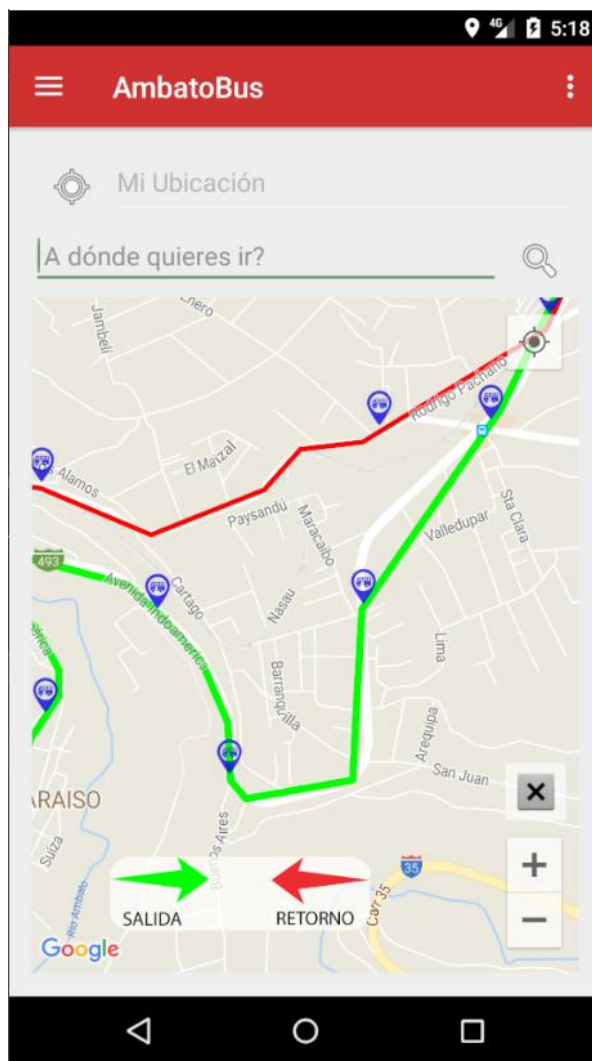
Mediante la codificación, se obtuvo algunos cambios requeridos por los usuarios en los test de deficiencias que los mismos realizaron. En las ilustraciones siguientes, se representan los cambios que se notaron más al momento de hacerlos: así como son los botones IU y la aplicación de las diferentes APIs para complementar a la aplicación.



**Ilustración 4.4** Prueba del resultado de la ruta más óptima que debe tomar el usuario en la versión Android 5.1

**Fuente:** elaboración propia

A continuación, se representa el cambio en las guías para el usuario y en el color de las rutas:



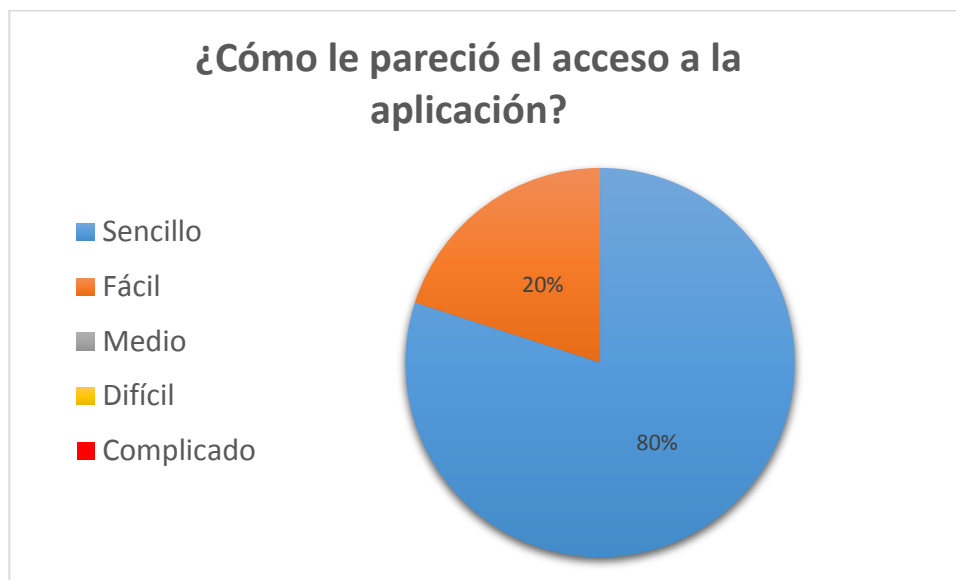
**Ilustración 4.5** Prueba de la visualización de la información de cada línea de bus.  
**Fuente:** elaboración propia

## 4.2 Validación de resultados

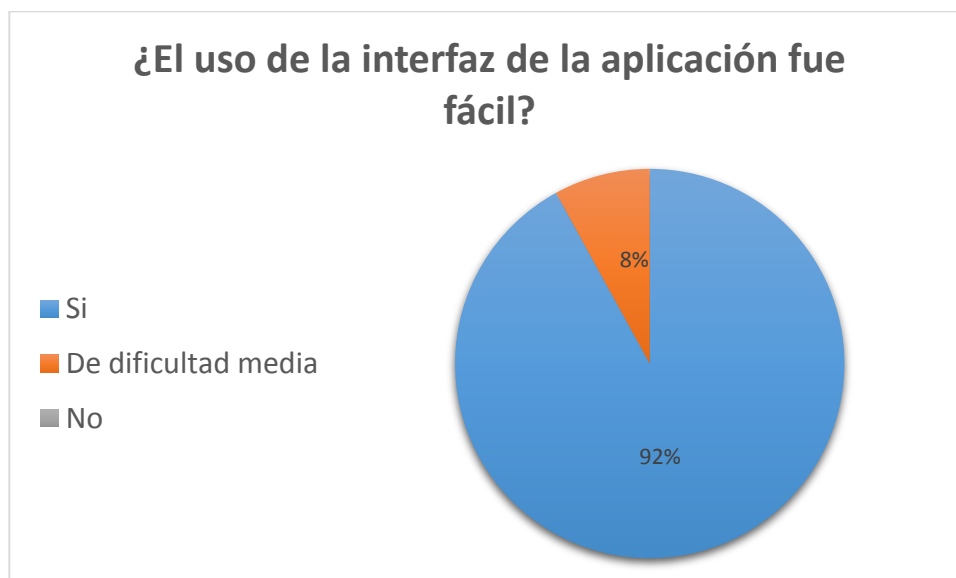
Para la validación de resultados, se realizó una encuesta de satisfacción veinticinco usuarios interesados en usar la aplicación, algunos de ellos, fueron de ayuda para realizar el resumen de deficiencias de la aplicación, y de dichos usuarios, se pudo obtener los requerimientos funcionales de la aplicación. En el Anexo 4 se detalla el modelo de la encuesta de satisfacción.

### 4.2.1 Encuestas de satisfacción

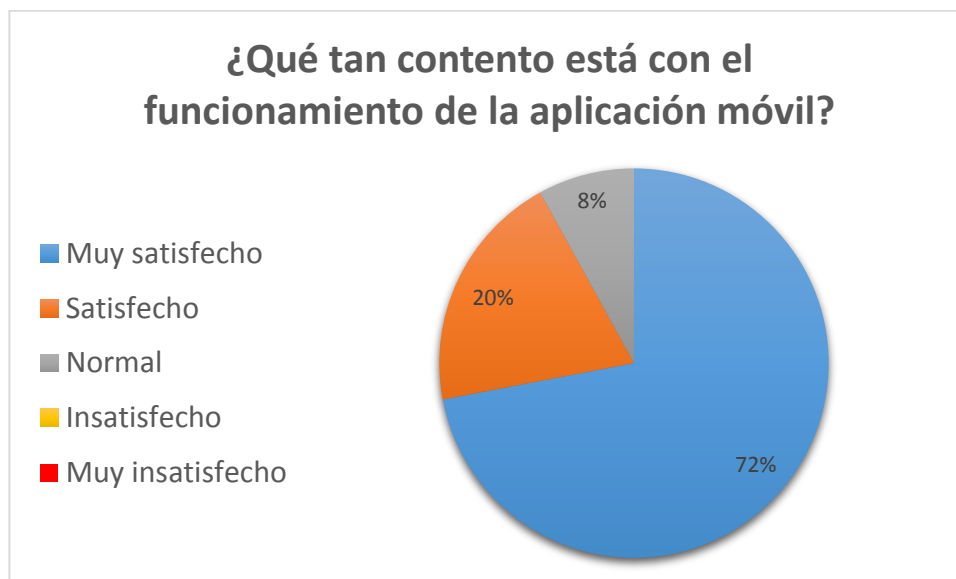
Las respuestas de los veinticinco usuarios encuestados fueron tabuladas de acuerdo a las diversas opciones que se les dio en las encuestas:



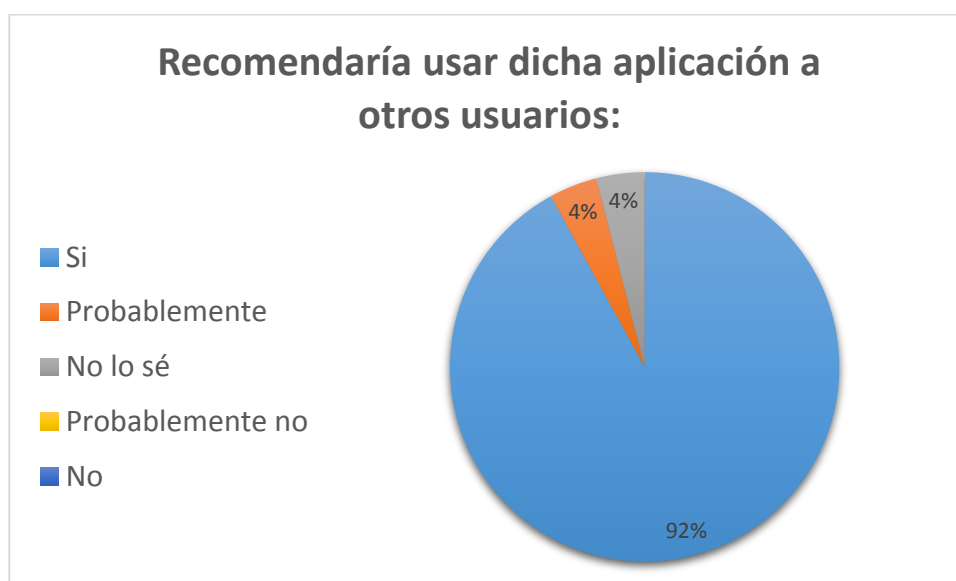
**Ilustración 4.6** Resultados de la pregunta 1  
Fuente: elaboración propia



**Ilustración 4.7** Resultados de la pregunta 2  
Fuente: elaboración propia



**Ilustración 4.8** Resultados de la pregunta 3  
**Fuente:** elaboración propia



**Ilustración 4.9** Resultados de la pregunta 4  
**Fuente:** elaboración propia

## CONCLUSIONES Y RECOMENDACIONES

### CONCLUSIONES

- Mediante la investigación y posterior implementación de la metodología Mobile-D, se llegó a determinar, que dicha metodología es excelente para proyectos con recursos humanos reducidos y tiempo limitado, ya que por medio del uso de talleres post iteración y test de aceptación, se describen los procesos necesarios para llevar a cabo los objetivos de la aplicación, las cuales son: indicar al usuario la ruta más óptima para llegar a su destino mediante el transporte de buses y mostrar las información general de cada una de las líneas de buses que existen en la ciudad de Ambato.
- Para el desarrollo del presente proyecto, se utilizó software especializados en desarrollo de aplicaciones móviles para Android y creación de *Web Services*, un sistema gestor de base de datos y un servidor de internet (IIS), lo que permitió desarrollar la aplicación de manera correcta, cumpliendo con todos los propósitos planteados como la implementación de una aplicación móvil, que permita conocer la mejor ruta disponible para llegar a un lugar mediante el sistema de buses urbanos, con sus respectivas pruebas de manera local.
- Mediante el uso del Google Maps Android API, se logró implementar funcionalidades de georreferenciación y diseño de las rutas en la aplicación AmbatoBus, lo que hizo posible determinar la ubicación del usuario y con ello mostrarle la ruta óptima en tiempo real.

- Más del 80% de los usuarios encuestados, mostraron satisfacción del funcionamiento, diseño y accesibilidad de la aplicación, lo cual permitió lanzarla a Google Play Store, con la seguridad de que AmbatoBus satisface las necesidades de las personas.

## RECOMENDACIONES

- Promocionar la aplicación AmbatoBus mediante los diferentes canales de las autoridades de la ciudad de Ambato como: La Dirección de Tránsito, Transporte y Movilidad, y el Gobierno Autónomo Descentralizado Municipalidad de Ambato.
- Mantener la información actualizada tanto de paradas como de rutas de los buses que pertenezcan a la ciudad de Ambato
- Desarrollar e implementar este tipo de aplicación en las demás ciudades del Ecuador, para que los visitantes tengan una herramienta útil al momento de movilizarse.
- Utilizar las demás APIs que Google ofrece para que las aplicaciones de este tipo tengan mayor usabilidad y accesibilidad hacia los usuarios, y se sientan satisfechos con las mismas.
- Para los desarrolladores que usen la metodología Mobile-D, deben realizar todas las iteraciones de la fase de desarrollo mediante los tres escenarios representados en: día de planificación, producción y entrega. De esta manera se tendrá mejor organizado el proyecto.

## Bibliografía

- [1] O. E. Acosta Mayorga, «Repositorio Digital Universidad Técnica de Ambato,» Junio 2015. [En línea]. Available: <http://repo.uta.edu.ec/handle/123456789/10999>.
- [2] M. Pérez Iglesias, Google, SC: Profit, 2012.
- [3] Instituto Nacional de Estadísticas y Censos, «Portal INEC,» 2010. [En línea]. Available: <http://www.ecuadorencifras.gob.ec/wp-content/descargas/Manu-lateral/Resultados-provinciales/tungurahua.pdf>.
- [4] Instituto Nacional de Estadísticas y Censos, «Portal INEC,» 20 Febrero 2013. [En línea]. Available: [www.inec.gob.ec](http://www.inec.gob.ec).
- [5] Real Academia Española, «Diccionario de la lengua Española,» 2016. [En línea]. Available: <http://dle.rae.es/?id=aMhvXp>.
- [6] M. Mauleón Torres, Transporte, operadores, redes, Diaz de Santos, 2013.
- [7] J. Claude Lablée y R. Montezuma , El transporte urbano: un desafío para el próximo milenio, Santafé de Bogotá : Javeriano, 1996.
- [8] J. F. Hernández Villena y H. D. Verdugo Ibarra , «Repositorio Institucional de la Escuela Superior Politécnica de Chimborazo,» 4 Mayo 2016. [En línea]. Available: <http://dspace.esPOCH.edu.ec/bitstream/123456789/4657/1/15T00628.pdf>.
- [9] J. Mundó Tejada, «Ecotrópicos Revista de la Sociedad Venezolana de Ecología,» 2002. [En línea]. Available: [http://ecotropicos.saber.ula.ve/db/ssaber/Edocs/pubelectronicas/fermentum/numero\\_34/articulo3.pdf](http://ecotropicos.saber.ula.ve/db/ssaber/Edocs/pubelectronicas/fermentum/numero_34/articulo3.pdf).
- [10] N. Olazabal Abal, «Dadun Depósito Académico Digital Universitario de Navarra,» 2014. [En línea]. Available: [http://dadun.unav.edu/bitstream/10171/37379/1/903127Nerea\\_Olazabal\\_DI.pdf](http://dadun.unav.edu/bitstream/10171/37379/1/903127Nerea_Olazabal_DI.pdf).
- [11] M. S. Ramírez Montoya, «UNED Revistas Científicas,» 2009. [En línea]. Available: [http://repo.uta.edu.ec/bitstream/123456789/10999/1/Tesis\\_t1014si.pdf](http://repo.uta.edu.ec/bitstream/123456789/10999/1/Tesis_t1014si.pdf).

- [12] K. Malave Polanco y J. L. Beauperthuy Taibo, «UAEM redalyc,» 2011. [En línea]. Available: <http://www.redalyc.org/articulo.oa?id=78219156004>.
- [13] Android by Google Inc., «Android Open Source Project,» SF. [En línea]. Available: <http://source.android.com/>.
- [14] M. Corrales Mora, Lenguaje Lgo III: Explorando la programación, San José, Costa Rica: Universidad Estatal a Distancia, 1994.
- [15] J. J. Rodríguez Salazar, Introducción a la programación, San Vicente: Clb Universitario, 2003.
- [16] R. M. Stallman, «Software libre para una sociedad libre,» Diciembre 2004. [En línea]. Available: <http://bibliotecadigital.org/bitstream/001/144/8/84-933555-1-8.pdf>.
- [17] Google Developer, «API de Google Maps,» 24 Marzo 2016. [En línea]. Available: <https://developers.google.com/maps/documentation/android-api/?hl=es>.
- [18] Google Developer, «API de Google Maps,» 1 Junio 2016. [En línea]. Available: <https://developers.google.com/maps/articles/geolocation#SpecifyingSensor>.
- [19] Android Developer, «Android Studio,» 6 Dicimebre 2016. [En línea]. Available: <https://developer.android.com/studio/index.html>. [Último acceso: 13 Febrero 2017].
- [20] Pearson Educación, «unefazuliasistemas.files.wordpress.com,» 2001. [En línea]. Available: <https://unefazuliasistemas.files.wordpress.com/2011/04/introduccion-a-los-sistemas-de-bases-de-datos-cj-date.pdf>. [Último acceso: 20 Julio 2017].
- [21] Microsoft, «support.office.com,» Microsoft, 2017. [En línea]. Available: <https://support.office.com/es-es/article/Conceptos-b%C3%A1sicos-sobre-bases-de-datos-a849ac16-07c7-4a31-9948-3c8c94a7c204>. [Último acceso: 20 Julio 2017].
- [22] P. Bazán, «BPEL: una propuesta para el uso de Web Services,» 2006. [En línea]. Available: [http://sedici.unlp.edu.ar/bitstream/handle/10915/21788/Documento\\_completo.pdf?sequence=1](http://sedici.unlp.edu.ar/bitstream/handle/10915/21788/Documento_completo.pdf?sequence=1). [Último acceso: 20 Julio 2017].

- [23] F. Durán Lasso, «Escuela Politécnica Nacional,» Julio 2013. [En línea]. Available: <http://bibdigital.epn.edu.ec/bitstream/15000/6546/1/CD-5012.pdf>. [Último acceso: 25 Julio 2016].
- [24] Y. D. Amaya Balaguera, «Gestion IT,» 14 Novimebre 2013. [En línea]. Available: [http://gestion-it.260mb.net/docs/pdf/12Articulo\\_Rev-Tec-Num-2.pdf?i=1](http://gestion-it.260mb.net/docs/pdf/12Articulo_Rev-Tec-Num-2.pdf?i=1). [Último acceso: 26 Septiembre 2017].
- [25] P. Blanco, J. Camarero, A. Fumero, A. Werteski y P. Rodríguez, «ResearchGate,» 2009. [En línea]. Available: [https://www.researchgate.net/profile/Antonio\\_Fumero/publication/267795011\\_Metodologia\\_de\\_desarrollo\\_agil\\_para\\_sistemas\\_moviles\\_Introduccion\\_al\\_desarrollo\\_con\\_Android\\_y\\_el\\_iPhone/links/577009d108ae842225aa444b/Metodologia-de-desarrollo-agil-para-sistemas-m](https://www.researchgate.net/profile/Antonio_Fumero/publication/267795011_Metodologia_de_desarrollo_agil_para_sistemas_moviles_Introduccion_al_desarrollo_con_Android_y_el_iPhone/links/577009d108ae842225aa444b/Metodologia-de-desarrollo-agil-para-sistemas-m). [Último acceso: 9 Septiembre 2017].
- [26] A. C. Spataru, «SemanticScholar,» 2010. [En línea]. Available: <https://pdfs.semanticscholar.org/47ed/deef28379c7cb5614557ea40c58e8d55c6ce.pdf>. [Último acceso: 9 Septiembre 2017].
- [27] R. M. Cruz Velandia, «Pontificia Universidad Javeriana,» Enero 2011. [En línea]. Available: <http://pegasus.javeriana.edu.co/~PA133-05-PMovVidaAutomotor/entregables/propuesta.pdf>. [Último acceso: 9 Septiembre 2017].
- [28] Android Google Developer, «API de Google Maps Android,» 24 Marzo 2016. [En línea]. Available: <https://developers.google.com/maps/documentation/android-api/intro?hl=es-419>. [Último acceso: 13 Febrero 2017].
- [29] Android O Developer Preview, «Android O Developer Preview,» Google developer, 2017. [En línea]. Available: <https://developer.android.com/guide/topics/graphics/opengl.html>. [Último acceso: 31 Julio 2017].
- [30] M. Gupt, «GitHub,» 1 Enero 2016. [En línea]. Available: <https://github.com/Truiron/PlaceAPIAutocomplete/tree/master/app/src/main/java/com/truiron/placeapiautocomplete>. [Último acceso: 16 Agosto 2017].
- [31] M. Thành Hiệp, «GitHub,» 3 4 2016. [En línea]. Available: <https://github.com/hiepquan2008/GoogleMapDirectionSimple/tree/master/app/src/main/java>. [Último acceso: 16 Agosto 2017].
- [32] EDMTDev, «GitHub,» 30 11 2016. [En línea]. Available: <https://github.com/eddydn/GetCoordinatesGeocode/tree/e9eb5d10c7f1c4f0ef8ca32da3046c1a55e8b1a7/app/src/main/java/dev/edmt/getcoordinatesgeocode>. [Último acceso: 16 Agosto 2017].

## ANEXOS

### Anexo 1: código de creación de la base de Datos Oracle

```

REM CREACION DE LA TABLA RUTAS Y UN TRIGGER PARA EL INCREMENTO AUTOMÁTICO DEL ID

CREATE TABLE TBL_RUTAS
(ID_RUTA NUMBER,
ID_LINEA,
COORDENADA_UNO DOUBLE PRECISION,
COORDENADA_DOS DOUBLE PRECISION,
DIRECCION VARCHAR(100),
ID_PARADA,
CONSTRAINT PK_RUTAS PRIMARY KEY (ID_RUTA),
CONSTRAINT FK_RUTA_LINEA FOREIGN KEY(ID_LINEA) REFERENCES TBL_LINEAS (ID_LINEA),
CONSTRAINT FK_RUTAS_PARADAS FOREIGN KEY(ID_PARADA) REFERENCES TBL_PARADAS (ID_PARADA));

CREATE SEQUENCE SEQ_RUTAS_IDRUTA
START WITH 1
INCREMENT BY 1
NOMAXVALUE;

CREATE OR REPLACE TRIGGER trig_rutas_seq
BEFORE INSERT ON TBL_RUTAS
FOR EACH ROW
BEGIN
    SELECT SEQ_RUTAS_IDRUTA.NEXTVAL INTO :NEW.ID_RUTA FROM DUAL;
END
;
.
/

REM CREACION DE LA TABLA LINEAS PARADAS Y UN TRIGGER PARA EL INCREMENTO AUTOMÁTICO DEL ID

CREATE TABLE TBL_LINEAS_PARADAS
(ID_LIN_PAR NUMBER,
ID_LINEA,
ID_PARADA,
DIRECCION VARCHAR (100),
CONSTRAINT PK_LIN_PAR PRIMARY KEY (ID_LIN_PAR),
CONSTRAINT FK_LIN_PAR_LINEA FOREIGN KEY(ID_LINEA) REFERENCES TBL_LINEAS (ID_LINEA),
CONSTRAINT FK_LIN_PAR_PARADA FOREIGN KEY(ID_PARADA) REFERENCES TBL_PARADAS (ID_PARADA));

CREATE SEQUENCE SEQ_LINPAR_IDLINPAR
START WITH 1
INCREMENT BY 1
NOMAXVALUE;

CREATE OR REPLACE TRIGGER trig_lin_par_seq
BEFORE INSERT ON TBL_LINEAS_PARADAS
FOR EACH ROW
BEGIN
    SELECT SEQ_LINPAR_IDLINPAR.NEXTVAL INTO :NEW.ID_LIN_PAR FROM DUAL;
END
;
.
/

```

---

REM CREACION DE LA TABLA LINEAS Y UN TRIGGER PARA EL INCREMENTO AUTOMÁTICO DEL ID

```
CREATE TABLE TBL_LINEAS
(ID_LINEA NUMBER,
COOPERATIVA_LINEA VARCHAR(20),
NOMBRE_LINEA VARCHAR (20),
DESCRIPCION VARCHAR (200),
CONSTRAINT PK_LINEAS PRIMARY KEY (ID_LINEA));

CREATE SEQUENCE SEQ_LINEAS_IDLINEA
START WITH 1
INCREMENT BY 1
NOMAXVALUE;

CREATE OR REPLACE TRIGGER trig_lineas_seq
BEFORE INSERT ON TBL_LINEAS
FOR EACH ROW
BEGIN
    SELECT SEQ_LINEAS_IDLINEA.NEXTVAL INTO :NEW.ID_LINEA FROM DUAL;
END
;
.
/
```

---

REM CREACION DE LA TABLA PARADAS Y UN TRIGGER PARA EL INCREMENTO AUTOMÁTICO DEL ID

```
CREATE TABLE TBL_PARADAS
(ID_PARADA NUMBER,
NOMBRE_PARADA VARCHAR(100),
COORDENADA_UNO DOUBLE PRECISION,
COORDENADA_DOS DOUBLE PRECISION,
CONSTRAINT PK_PARADAS PRIMARY KEY (ID_PARADA));

CREATE SEQUENCE SEQ_PARADAS_IDPARADA
START WITH 1
INCREMENT BY 1
NOMAXVALUE;

CREATE OR REPLACE TRIGGER trig_paradas_seq
BEFORE INSERT ON TBL_PARADAS
FOR EACH ROW
BEGIN
    SELECT SEQ_PARADAS_IDPARADA.NEXTVAL INTO :NEW.ID_PARADA FROM DUAL;
END
;
.
/
```

## Anexo 2: Código de consultas de la WS

```

[WebMethod]
0 referencias | 0 excepciones
public DataSet Obtener_Datos_tbl_Lineas()
{
    String cadena;
    cadena = WebConfigurationManager.ConnectionStrings["DBAmbatoBus"].ConnectionString;
    OracleConnection conn = new OracleConnection(cadena);
    conn.Open();
    if (conn.State == ConnectionState.Open)
    {
        OracleDataAdapter da = new OracleDataAdapter("select * from tbl_lineas", conn);
        DataSet ds = new DataSet();
        da.Fill(ds, "tbl_lineas");

        return ds;
    }
    else
    {
        return null;
    }
}

```

```

[WebMethod]
0 referencias | 0 excepciones
public DataSet Obtener_Datos_Paradas_de_Lineas(String id)
{
    String cadena;
    cadena = WebConfigurationManager.ConnectionStrings["DBAmbatoBus"].ConnectionString;
    OracleConnection conn = new OracleConnection(cadena);
    conn.Open();
    if (conn.State == ConnectionState.Open)
    {
        OracleDataAdapter da = new OracleDataAdapter("select descripcion, nombre_parada, coordenada_uno, coordenada_dos from tbl_lineas L,"
            + " tbl_lineas_paradas LP, tbl_paradas P where LP.id_linea = " + id + " and L.id_linea = " + id
            + " and P.id_parada= LP.id_parada", conn);
        DataSet ds = new DataSet();
        da.Fill(ds, "tbl_lineas L, tbl_lineas_paradas LP, tbl_paradas P");
        return ds;
    }
    else
    {
        return null;
    }
}

```

```

[WebMethod]
0 referencias | 0 excepciones
public DataSet Obtener_Datos_Paradas_sin_id()
{
    String cadena;
    cadena = WebConfigurationManager.ConnectionStrings["DBAmbatoBus"].ConnectionString;
    OracleConnection conn = new OracleConnection(cadena);
    conn.Open();
    if (conn.State == ConnectionState.Open)
    {
        OracleDataAdapter da = new OracleDataAdapter("select LP.id_linea, LP.id_parada, nombre_parada, coordenada_uno, coordenada_dos,"
            + " direccion from tbl_lineas_paradas LP INNER JOIN tbl_paradas P on LP.id_parada = P.id_parada", conn);
        DataSet ds = new DataSet();
        da.Fill(ds, "tbl_lineas_paradas LP, tbl_paradas P");
        return ds;
    }
    else
    {
        return null;
    }
}

```

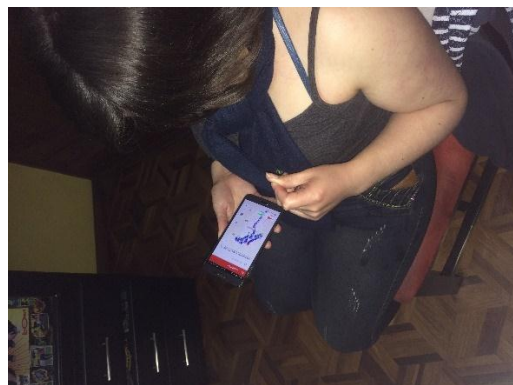
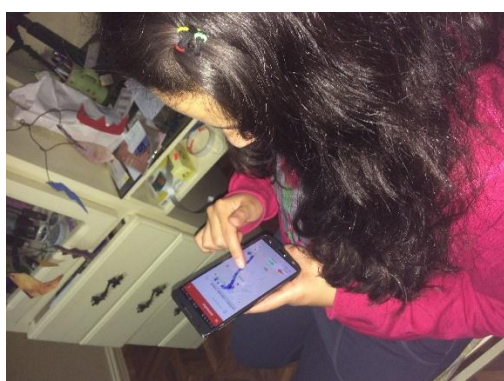
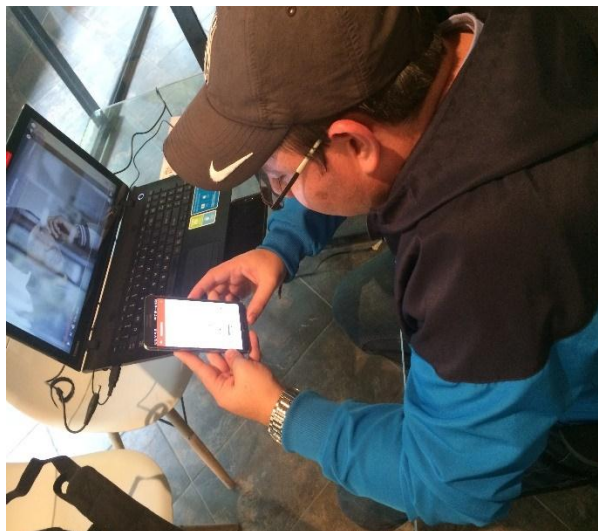
```
[WebMethod]
0 referencias | 0 excepciones
public DataSet Obtener_Datos_Rutas_de_Lineas(String id)
{
    String cadena;
    cadena = WebConfigurationManager.ConnectionStrings["DBAmbatoBus"].ConnectionString;
    OracleConnection conn = new OracleConnection(cadena);
    conn.Open();
    if (conn.State == ConnectionState.Open)
    {
        OracleDataAdapter da = new OracleDataAdapter("select coordenada_uno, coordenada_dos, direccion from tbl_rutas R where R.id_linea = "
            + id, conn);
        DataSet ds = new DataSet();
        da.Fill(ds, "tbl_rutas");
        return ds;
    }
    else
    {
        return null;
    }
}
```

```
[WebMethod]
0 referencias | 0 excepciones
public DataSet Obtener_Datos_Rutas_de_Lineas_sin_id()
{
    String cadena;

    cadena = WebConfigurationManager.ConnectionStrings["DBAmbatoBus"].ConnectionString;
    OracleConnection conn = new OracleConnection(cadena);
    conn.Open();
    if (conn.State == ConnectionState.Open)
    {
        OracleDataAdapter da = new OracleDataAdapter("select id_linea, coordenada_uno, coordenada_dos, direccion, id_parada from tbl_rutas", conn);
        DataSet ds = new DataSet();
        da.Fill(ds, "tbl_rutas");

        return ds;
    }
    else
    {
        return null;
    }
}
```

Anexo 3: ilustraciones de los usuarios que realizaron las pruebas de la aplicación



## Anexo 4: Encuesta de satisfacción

### Encuesta de satisfacción de la aplicación móvil AmbatoBus

Nombre \_\_\_\_\_

Carrera \_\_\_\_\_

¿Cómo le pareció el acceso a la aplicación?

Sencillo	Fácil	Medio	Difícil	Complicado

¿El uso de la interfaz de la aplicación fue fácil?

Si	De dificultad media	No

¿Qué tan contento está con el funcionamiento de la aplicación móvil:

Muy satisfecho	Satisfecho	Normal	Insatisfecho	Muy insatisfecho

Recomendaría usar dicha aplicación a otros usuarios:

Si	Probablemente	No lo sé	Probablemente no	No

Firma: