

UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO

CARRERA:
INGENIERÍA DE SISTEMAS

Trabajo de titulación previo a la obtención del título de:
Ingeniero de Sistemas

TEMA:
APLICACIÓN MÓVIL PARA CONSULTAS DE TRANSPORTE
INTERPROVINCIAL CON LOS DATOS ALMACENADOS EN LA NUBE

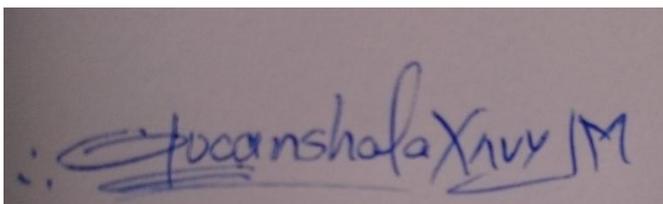
AUTOR:
XAVIER DAVID AUCANSHALA PILATUÑA

TUTOR:
FERNANDO JACINTO RODAS ORELLANA

Quito, septiembre del 2017

CESION DE DERECHOS DE AUTOR

Yo, Xavier David Aucanshala Pilatuña con documento de identificación N° 1722715214, manifiesto mi voluntad y cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy el autor del trabajo de titulación con el tema: “APLICACIÓN MÓVIL PARA CONSULTAS DE TRANSPORTE INTERPROVINCIAL CON LOS DATOS ALMACENADOS EN LA NUBE”, mismo que ha sido desarrollado para optar por el título de INGENIERO DE SISTEMAS en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente. En aplicación a lo determinado en la Ley de Propiedad Intelectual, en mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.



.....
XAVIER DAVID

AUCANSHALA PILATUÑA

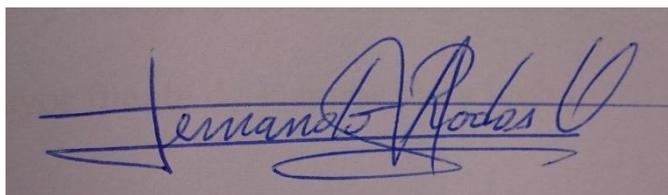
CI: 1722715214

Quito, septiembre del 2017

DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR

Yo declaro que bajo mi dirección y asesoría fue desarrollado el Proyecto Técnico, con el tema APLICACIÓN MÓVIL PARA CONSULTAS DE TRANSPORTE INTERPROVINCIAL CON LOS DATOS ALMACENADOS EN LA NUBE realizado por Xavier David Aucanshala Pilatuña, obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana, para ser considerados como trabajo final de titulación.

Quito, septiembre del 2017



.....
FERNANDO JACINTO RODAS ORELLANA

CI: 1708514821

Dedicatoria

Dedico este trabajo primeramente a mi Dios quien ha sido la fuerza espiritual y motivacional a lo largo de mi vida.

Mis padres Ángel y Carmen por todo su amor, comprensión y ese incondicional apoyo en cada etapa de mi vida, por ayudarme siempre en la parte espiritual y económica, sobre todo cuando estuve a punto de rendirme fueron mi mayor fuerza para poder empezar nuevamente en la lucha por mis sueños, sus consejos, valores, principios y sobre todo el espíritu de lucha y perseverancia han hecho de mi una persona decidida y valiente.

Mis hermanos Cristian, Ivan, Stalyn, hermanas Alicia, Verónica, Camila, sobrinas Ayleen y Alicia quienes con sus consejos, correcciones, compañía y apoyo han sido mi mayor fuente de inspiración y lucha.

Mis J&J, JM, PD fundamentales en mi vida y también mis amigas que con una pequeña palabra de aliento estuvieron al pendiente de mi vida.

Xavier David Aucanshala Pilatuña

Agradecimiento

Agradezco a la Universidad Politécnica Salesiana Campus Sur que ha contribuido en mi formación profesional y personal, también a todos mis profesores que con sus conocimientos supieron encaminar a mi formación intelectual, a mi tutor de proyecto de titulación el Ingeniero Fernando Rodas por haberme orientado y motivado para poder realizar mi trabajo.

Xavier David Aucanshala Pilatuña

ÍNDICE

Capítulo 1	1
INTRODUCCIÓN	1
1.1 Antecedentes	2
1.2 Problema	8
1.3 Justificación	9
1.4 Objetivo general	11
1.5 Objetivos específicos	11
1.6 Marco metodológico	12
1.6.1. Elementos de SCRUM.....	14
1.6.2. Los roles de SCRUM.....	16
1.6.3. Reuniones.....	16
1.7 Marco teórico	16
1.7.1. Programación orientada a objetos.....	18
1.7.2. Sistemas operativos	18
1.7.3. Android	19
1.7.4. Geolocalización	21
1.7.5. Aplicaciones Móviles	22
1.7.6. API de Google Maps	23
1.7.7. Google Maps	23
1.7.8. Base de datos relacional.....	24
Capítulo 2	25

ANÁLISIS Y DISEÑO	25
2.1. Análisis de requerimientos	26
2.1.1 Identificación de historias de usuario	26
2.1.2 Historias de usuario	27
2.1.3 Requerimientos del usuario de la aplicación móvil	28
2.2 Diseño	34
2.2.1. Diagramas de casos de uso.....	34
2.2.1.1. Diagrama de casos de uso para el usuario general de la Aplicación Móvil.	35
2.2.2. Diagrama físico de la base de datos MySQL	36
2.2.3. Diccionario de datos.	38
2.2.3.1 Diccionario de datos BDD Cooperativas Ecuador.....	38
2.2.3.2 Diccionario de datos BDD Compra Boletos	41
2.2.4. Diagrama de clases	43
2.2.4.1. Diagrama de clases para la Aplicación Móvil	43
2.2.5. Diagrama de estados UML.....	45
2.2.5.1. Diagrama de estados de la Aplicación Móvil.	45
2.2.6. Diagrama de despliegue	46
2.2.6.1. Diagrama de despliegue de la Aplicación Móvil.	46
2.2.7.1. Interfaz de Usuario para la Aplicación Móvil.	47
2.2.7.1.1. Pantalla Principal de la Aplicación Móvil.	47
2.2.7.1.2. Pantalla Menú principal de la Aplicación Móvil.	48
2.2.7.1.3. Pantalla Mensaje del Autor y desarrollador de la aplicación móvil.	49

2.2.7.1.4. Pantalla Módulo Cooperativas	49
2.2.7.1.5. Pantalla lista de ciudades destino.....	50
2.2.7.1.6. Pantalla lista de Cooperativas hacia un destino.	50
2.2.7.1.7. Pantalla información detallada de una cooperativa.	51
2.2.7.1.8. Pantalla Módulo Horarios.....	51
2.2.7.1.9. Pantalla lista de cooperativas.	52
2.2.7.1.10. Pantalla lista de horarios de una cooperativa	52
2.2.7.1.11. Pantalla Módulo Terminales	53
2.2.7.1.12. Pantalla Módulo Boletos.....	53
2.2.7.1.13. Pantalla lista de turnos.	54
2.2.7.1.14. Pantalla mensaje antes de comprar un asiento.	54
2.2.7.1.15. Pantalla calendario de viaje	55
2.2.7.1.16. Pantalla selección de asiento.	55
2.2.7.1.17. Pantalla pago con tarjeta de crédito.....	56
2.2.7.1.18. Pantalla Módulo cobro de boleto con tarjeta.....	56
2.2.7.1.19. Pantalla correo para facturar	57
2.2.7.1.20. Pantalla mensaje de agradecimiento.....	57
2.2.7.1.21. Pantalla lista de asientos ocupados.....	58
Capítulo 3	59
CODIFICACIÓN Y PRUEBAS	59
3.1 Arquitectura y tecnologías de la aplicación móvil	59
3.1.1. Arquitectura de la Aplicación Móvil.	59

3.1.1.1. Capa de Presentación	60
3.1.1.2. Capa Lógica.....	60
3.1.1.3. Capa de Persistencia	60
3.1.2. Entorno de desarrollo Android Studio IDE.....	62
3.1.3. Almacén de Claves de Android (KeyStore)	62
3.2. Codificación.....	63
3.2.1. Clases y métodos de la Aplicación Móvil.....	63
3.2.1.1 Conexión BDD MySQL y PHP.....	63
3.2.1.2. Clase Cooperativas	64
3.2.1.3. Módulo búsqueda de cooperativas.....	64
3.2.1.3.1. Búsqueda de cooperativas por ciudad destino	64
3.2.1.3.2. Información detallada de una cooperativa.....	65
3.2.1.4. Módulo búsqueda de horarios	66
3.2.1.4.1. Búsqueda de cooperativas por ciudad destino	66
3.2.1.4.2. Horarios y precios de una cooperativa	68
3.2.1.5. Módulo búsqueda de Terminales terrestres	69
3.2.1.6. Módulo Boletos	70
3.2.1.6.1. Búsqueda de turnos	70
3.2.1.6.2. Selección de fecha y asiento de viaje	72
3.2.1.6.3. Pago de boleto con tarjeta de crédito	73
3.3. Pruebas e implementación	74
3.3.1. Implementación de KeyStore Android.	74

3.3.2. Generación del APK	77
3.3.3. Pruebas de funcionamiento de la Aplicación Móvil.	77
3.3.3.1. Prueba en Búsqueda de Cooperativas	78
3.3.3.2. Prueba en Búsqueda de Horarios	80
3.3.3.3. Prueba en Búsqueda de Terminales Terrestres	82
3.3.3.4. Prueba en Compra de Boletos	83
3.3.4. Prueba de rendimiento de la Aplicación Móvil.	89
3.3.5. Pruebas de carga de la Aplicación Móvil.....	90
3.3.6. Pruebas de stress de la Aplicación Móvil.....	91
3.3.7. Resultados de la Aplicación Móvil.	91
CONCLUSIONES	93
RECOMENDACIONES	95
GLOSARIO DE TÉRMINOS	96
LISTA DE REFERENCIAS	97

ÍNDICE DE TABLAS

Tabla 1. Historia de usuario: mostrar un menú de opciones de búsqueda.....	28
Tabla 2. Historia de usuario: buscar las cooperativas seleccionando una ciudad destino	28
Tabla 3. Historia de usuario: mostrar la información detallada de una cooperativa	29
Tabla 4. Historia de usuario: buscar las cooperativas seleccionando una ciudad destino	30
Tabla 5. Historia de usuario: mostrar los horarios y precios de pasaje de una cooperativa ...	30
Tabla 6. Historia de usuario: para visualizar todas las terminales terrestres en el mapa.....	31
Tabla 7. Historia de usuario: buscar los turnos de una cooperativa	32
Tabla 8. Historia de usuario: escoger una fecha actual o posterior y un asiento disponible...32	
Tabla 9. Historia de usuario: Para realizar el pago de un boleto de bus interprovincial	33
Tabla 10. Descripción de las tablas de la base de datos Cooperativas Ecuador MySQL	37
Tabla 11. Descripción de las tablas de la base de datos Compra Boletos MySQL	38
Tabla 12. Diccionario de datos de la tabla Cantón.....	38
Tabla 13. Diccionario de datos de la tabla COOPERATIVAS.....	39
Tabla 14. Diccionario de datos de la tabla DESTINO	39
Tabla 15. Diccionario de datos de la tabla LOCALIZACIÓN	39
Tabla 16. Diccionario de datos de la tabla PROVINCIA	40
Tabla 17. Diccionario de datos de la tabla REGIÓN	40
Tabla 18. Diccionario de datos de la tabla TERMINALES	40
Tabla 19. Diccionario de datos de la tabla TURNO	41
Tabla 20. Diccionario de datos de la tabla TERMINALES	41
Tabla 21. Diccionario de datos de la tabla COOPERATIVAS.....	42
Tabla 22. Diccionario de datos de la tabla DESTINO	42
Tabla 23. Diccionario de datos de la tabla BOLETOS	42
Tabla 24. Archivo activity_cdestino.xml	64
Tabla 25. Cooperativas por ciudad destino.....	64

Tabla 26. Consulta de cooperativas por ciudad destino	65
Tabla 27. Archivo activity_descripcioncoop.xml	66
Tabla 28. Información detallada de una cooperativa	66
Tabla 29. Consulta de información detallada de una cooperativa.....	66
Tabla 30. Archivo activity_horario.xml	67
Tabla 31. Cooperativas por ciudad destino horario.....	67
Tabla 32. Consulta de cooperativas por ciudad destino horario	68
Tabla 33. Archivo activity_descripcionhorarios.xml	68
Tabla 34. Horarios y precios de una cooperativa	68
Tabla 35. Consulta de horarios y precios de una cooperativa.....	69
Tabla 36. Archivo activity_terminales.xml	69
Tabla 37. Marcadores de terminales en el mapa.....	70
Tabla 38. Archivo activity_boletos.xml	70
Tabla 39. Turnos para una cooperativa por ciudad destino	70
Tabla 40. Consulta de turnos para una cooperativa por ciudad destino	71
Tabla 41. Archivo activity_asientos.xml.....	72
Tabla 42. Fecha y asiento de una cooperativa	72
Tabla 43. Consulta de asientos ocupados en una fecha.....	73
Tabla 44. Archivo activity_tarjeta.xml.....	73
Tabla 45. Pago de un boleto con tarjeta	73
Tabla 46. Inserción de pago de un boleto con tarjeta	74

ÍNDICE DE FIGURAS

Figura 1. Fases Metodología SCRUM	13
Figura 2. Sistema de capas de Android	21
Figura 3. Desarrollo tradicional en cascada.....	25
Figura 4. Diagrama de caso de uso del usuario general de la aplicación móvil	35
Figura 5. Diagrama físico de la Base de Datos Cooperativas Ecuador en MySQL	36
Figura 6. Diagrama físico de la Base de Datos Compra Boletos en MySQL	37
Figura 7. Diagrama de clases de la Aplicación Móvil de Cooperativas	44
Figura 8. Diagrama de estado de la aplicación móvil	45
Figura 9. Diagrama de despliegue de la aplicación móvil.....	46
Figura 10. Pantalla principal.....	47
Figura 11. Pantalla menú principal	48
Figura 12. Pantalla mensaje de autores	49
Figura 13. Pantalla módulo Cooperativas.....	49
Figura 14. Pantalla lista de ciudades destino	50
Figura 15. Pantalla lista de cooperativas	50
Figura 16. Pantalla información de cooperativa	51
Figura 17. Pantalla módulo Horarios	51
Figura 18. Pantalla lista de cooperativas	52
Figura 19. Pantalla lista de horarios	52
Figura 20. Pantalla módulo Terminales	53
Figura 21. Pantalla módulo Boletos	53
Figura 22. Pantalla lista de turnos	54
Figura 23. Pantalla mensaje antes de compra	54
Figura 24. Pantalla calendario de viaje	55
Figura 25. Pantalla selección de asiento.....	55

Figura 26. Pantalla pago con tarjeta de crédito.....	56
Figura 27. Pantalla cobro de boleto con tarjeta.....	56
Figura 28. Pantalla correo para facturar	57
Figura 29. Pantalla mensaje de agradecimiento.....	57
Figura 30. Pantalla lista de asientos ocupados.....	58
Figura 31. Arquitectura de tres capas.....	60
Figura 32. Servidor Web de Android PHP / MySQL y JSON.....	61
Figura 33. Generación APK firmado	75
Figura 34. Ventana creación de nuevo Key Store.....	75
Figura 35. Ingresar datos para crear Key Store.....	76
Figura 36. Validación de credenciales	76
Figura 37. Selección de un tipo de firma.....	77
Figura 38. Mensaje de APK creada.....	77
Figura 39. Búsqueda de cooperativas por ciudad destino.....	79
Figura 40. Información detallada de una cooperativa	80
Figura 41. Búsqueda de cooperativas por ciudad destino horarios	81
Figura 42. Horarios y precios de una cooperativa.....	82
Figura 43. Búsqueda de terminales terrestres	83
Figura 44. Búsqueda de turnos disponibles	84
Figura 45. Selección de fecha y asiento para viaje	85
Figura 46. Pago de boleto con tarjeta de crédito.....	86
Figura 47. Envío de información de boleto a correo.....	87
Figura 48. Información recibida en correo electrónico	88
Figura 49. Gráfico prueba de rendimiento de la aplicación móvil.....	89
Figura 50. Pruebas de carga del servidor web	90
Figura 51. Pruebas de stress del servidor web	91

Figura 52. Gráfico de calificación de la app en Play Store92

Resumen

En la actualidad, el uso de teléfonos inteligentes (smartphone) Android, servicios web, almacén de datos en la nube, georreferenciación y aplicaciones móviles es esencial en la vida diaria de las personas, por lo que desarrollar una aplicación móvil que contenga información detallada de cooperativas de buses interprovinciales de manera rápida, confiable y sencilla ayudará a los ciudadanos a moverse en buses interprovinciales dentro del territorio ecuatoriano.

El propósito de este proyecto es aprovechar la tecnología para proporcionar una guía práctica a los usuarios para que puedan obtener información simple, efectiva y actualizada de las cooperativas de transporte interprovincial del Ecuador almacenado en una base de datos en la nube, sin la necesidad de acudir a una terminal terrestre, oficina de encomiendas, llamadas telefónicas o páginas web por dicha información. Para este proyecto se utiliza la metodología Scrum por su forma ágil y liviana de abordar el desarrollo de la aplicación móvil.

De esta manera se optimiza y automatiza la búsqueda de información de las cooperativas de transporte interprovincial del Ecuador, en cuanto a detalles generales, recorridos, horarios, que podrían llevarlo a una ciudad destino que el usuario elija, además ubicar las terminales terrestres dentro del territorio ecuatoriano a través de un mapa de georreferenciación, y por último un simulador de compra de boletos de asientos de las cooperativas de transporte interprovincial del Ecuador. El resultado obtenido en la Play Store muestra la gran aceptación de los usuarios con calificación de 4,6 estrellas y 50 descargas en una semana.

Abstract

Currently, the use of Android smartphones, web services, data warehouse in the cloud, georeferencing and mobile applications are essential in people's daily lives, So developing a mobile application containing detailed information on interprovincial transport cooperatives quickly, reliable and simple way will help citizens mobilize on interprovincial buses within Ecuadorian territory.

The purpose of this project is to take advantage of technology to provide a practical guide so that users can obtain simple, effective and updated information from Ecuador's interprovincial transport cooperatives stored in a database in the cloud without the need to go to a land terminal, Office of parcels, telephone calls or web pages by that information. Scrum methodology is used for this Project, for its agile and light way of approaching the development of the mobile application.

In this way, the information search of the interprovincial transport cooperatives of Ecuador is optimized and automated, as to a general details, schedules that could take you to a destination city that the user chooses, also the names, addresses and locations of the land terminals within the Ecuadorian territory through a map of georeferencing, and finally a simulator to buy Seating tickets of the interprovincial transport cooperatives of Ecuador. The result obtained in the Play Store shows the great acceptance of users with a rating of 4.6 stars and 50 downloads in a week.

Capítulo 1

INTRODUCCIÓN

En la actualidad las personas desean obtener información rápida, confiable y sencilla de las cooperativas de buses interprovinciales a pesar de contar con terminales terrestres, oficinas de encomiendas, llamadas telefónicas o páginas web, pero no siempre se encuentra disponible dicha información y es ahí donde intervienen las aplicaciones móviles. Esta aplicación móvil facilita y mejora el acceso a la información de cooperativas de transporte interprovincial del Ecuador.

Capítulo 1. Introducción y marco teórico

Para implementar la aplicación móvil ECUA Buses se realiza una investigación a través de información bibliográfica la cual permite analizar la situación actual del transporte de buses interprovinciales en el Ecuador y la utilidad que brinda la aplicación móvil, la misma que está dirigida a todos los usuarios que posean dispositivos móviles Android, que es un sistema operativo open source, por lo que no requiere de ninguna inversión fuerte, además que no se cuenta con el auspicio de una entidad pública ni privada.

Capítulo 2. Análisis y diseño

Se detallan el desarrollo de las fases de análisis y diseño del producto final, empezando con el análisis de requerimientos, que son necesarios para el desarrollo y el diseño de la aplicación móvil, un diagrama físico de entidad-relación de la base de datos en MySQL además de la construcción de una interfaz para la aplicación móvil Android. Aplicando las fases de desarrollo de software incremental que permite realizar pruebas y documentación después de cada etapa.

Capítulo 3. Codificación y pruebas

En este capítulo se muestra paso a paso todo el proceso que se lleva a cabo para el desarrollo de la aplicación móvil, todas las herramientas tecnológicas que se utilizaron previamente configuradas, la codificación general de la aplicación y por último las distintas pruebas a la que fue sometida la aplicación móvil para validar su funcionalidad, para así poder cumplir las expectativas de los usuarios generales en cuanto a sus requerimientos.

1.1 Antecedentes

El desarrollo de las Aplicaciones móviles día tras día ha ido en constante crecimiento tomando parte muy importante en la convivencia de las personas con la tecnología. La utilización de estas aplicaciones para teléfonos móviles por parte de las persona alrededor del mundo en los últimos años van teniendo gran aceptación por la gran cantidad de servicios que pueden brindar estas aplicaciones como por ejemplo en las redes sociales, juegos, música, georreferenciación, videos, edición, etc.

En este gran conjunto de aplicaciones móviles, es necesario indicar a las aplicaciones que utilizan Georreferenciación que son desarrolladas con el fin de evitar que un usuario que desea moverse de un lugar a otro pero no tiene conocimiento, deba ir a una fuente de información tradicional como pueden ser los puntos de información turística, guías telefónicas o mapas de papel, por información de dicho lugar pero que resulta una pérdida de tiempo, ineficiente, molesto y en la mayor parte resulta inexacto por la constante actualización de los datos.

Del mismo modo los organismos públicos del estado ecuatoriano han visto la necesidad de automatizar la mayor parte de sus procesos a través del desarrollo e implementación de aplicaciones móviles para gestionar sus servicios informativos, de

documentación o gestionar la georreferenciación de su personal o lugares de dichos organismos del Estado. Según la Ley Orgánica de Transporte Terrestre, Tránsito y Seguridad Vial – LOTTTSV, en el artículo 30.5 literal d se indica:

Planificar, regular y controlar las actividades y operaciones de transporte terrestre, tránsito y seguridad vial, los servicios de transporte público de pasajeros y bienes, transporte comercial y toda forma de transporte colectivo y/o masivo, en el ámbito intracantonal, conforme la clasificación de las vías definidas por el Ministerio del Sector. (Ministerio de Turismo, 2014)

También en el artículo 46 se indica:

El transporte terrestre automotor es un servicio público esencial y una actividad económica estratégica del Estado, que consiste en la movilización libre y segura de personas o de bienes de un lugar a otro, haciendo uso del sistema vial nacional, terminales terrestres y centros de transferencia de pasajeros y carga en el territorio ecuatoriano. Su organización es un elemento fundamental contra la informalidad, mejorar la competitividad y lograr el desarrollo productivo, económico y social del país, interconectado con la red vial internacional.

En el caso de la FENACOTIP, determina que:

El avance tecnológico, los procesos de globalización y las nuevas políticas gubernamentales, crean la necesidad de hacer una revisión de la organización en todos sus niveles y estructura, en la parte administrativa, organizativa, de gestión y financiera, agregando también el análisis de la imagen de la institución dentro del contexto nacional, toda vez que la actual administración

con una visión muy progresista lo considera pertinente e histórico realizar un plan estratégico que permita crear un nuevo horizonte institucional con una mejor planificación que aumente la calidad del servicio a sus filiales. (FENACOTIP, 2016)

El transporte terrestre pertenece a un sector estratégico en la economía nacional, al ser el motor que impulsa las actividades particulares y productivas del país. “Las personas necesitan movilizarse ya sea por razones de trabajo, estudio, turismo u otras. Por lo tanto, es de suma importancia que se realice investigaciones como la presente sobre este sector” (INEC, 2016).

La escasa información con el servicio de transporte terrestre de pasajeros e información disgregada en diferentes oficinas de las cooperativas de transporte terrestre, vuelven indispensable el desarrollo de esta aplicación móvil, en el cual se recoge y analiza información concreta sobre el servicio de transporte terrestre.

El transporte terrestre de pasajeros en el Ecuador no cuenta con un gran avance en cuestión de tecnología si se habla como FENACOTIP con respecto a aplicaciones móviles siendo algunas cooperativas de transporte terrestre que cuentan con la economía, estructura y organización estable son las que pueden implementar sus propias páginas web que en muchos casos no ayuda a lo que realmente el usuario busca como información específica y las demás cooperativas no pueden dar a conocer sus servicios, horarios y rutas. Además, por razones de logística y políticas de la FENACOTIP no se puede acceder a la Base de datos de las cooperativas de transporte terrestre del Ecuador, de manera que para la recolección de información se recurre a la toma de datos a través de fotografías en las ventanillas de cada cooperativa en contados Terminales Terrestres, y para desarrollar el análisis de requerimientos es

necesario contar con información que superen los treinta datos con la finalidad de tener una visión más amplia del comportamiento de las cooperativas de transporte en cuanto a recorridos u horarios.

Tomando como referencia distintas aplicaciones móviles desarrolladas en el Ecuador específicamente sobre georreferenciación, se cita una tesis realizada en el 2015, de Ángel Joel Sánchez Bonilla. Titulada como “Estudio de los conocimientos que tienen los ciudadanos de Guayaquil acerca de los recorridos de las líneas de Transporte urbano para determinar sus necesidades de información e implementar un prototipo para móviles con Android que sirva de guía a las personas que no conocen las rutas de transporte urbano.”, para la carrera de Ingeniería en Sistemas de la Universidad de Guayaquil.

El diseño de una aplicación de guía de recorrido de las líneas de transporte urbano de Guayaquil será de gran utilidad para que los ciudadanos o turistas que se movilizan haciendo uso de las líneas de transporte urbano, tengan una guía confiable y disponible en cualquier momento eliminando la inseguridad que tenían al no saber con exactitud los recorridos de las mismas. (Sánchez Bonilla, 2015)

Cabe recalcar que las características que posee este proyecto de tesis, está alineado al tema de georreferenciación y visualización de la información en una aplicación móvil, y será el inicio partida para la investigación, desarrollo e implementación del proyecto.

También en el área de aplicaciones móviles de georreferenciación se cita la tesis de Susana del Carmen Illescas Ochoa, titulada como “Proyecto de desarrollo para mejorar el sistema de transportación interprovincial análisis de ruta Guayaquil-Salitre-

Vinces” para la Escuela Superior Politécnica del Litoral. Esta tesis tiene como objetivo:

Cambiar la mentalidad de los transportistas que consideran que la satisfacción de los pasajeros no es importante y que no afecta sus ganancias...optimizar el servicio prestado e implantar innovaciones en la atención al público no es un costo irrecuperable sino, es una inversión que aumentará las ganancias a corto plazo, ya que los usuarios preferirán usar un transporte donde se sientan más cómodos y seguros. (Illescas Ochoa, 2007)

Con este objetivo se busca que los dirigentes y transportistas cambien su manera de pensar en cuanto a proporcionar la información requerida para el desarrollo de una aplicación móvil que podría a corto plazo resultarles muy rentables en cuanto al uso de sus servicios por parte de los usuarios frecuentes y ocasionales de sus buses de transporte.

Además dentro del área de las aplicaciones móviles de georreferenciación se cita la tesis de Loayza Abad Freddy Patricio, titulada como “Aplicación móvil para la georreferenciación de buses urbanos en la ciudad de Loja”, para la Universidad Nacional de Loja, esta tesis tiene como introducción:

En la actualidad en la ciudadanía Lojana existe un gran problema por la falta de información de las paradas y de la disponibilidad de información del transporte urbano en la ciudad de Loja, por lo que se ha propuesto como tema de tesis “Aplicación Móvil Para La Georreferenciación De Buses Urbanos En La Ciudad De Loja”, el cual me permite la georreferenciación en tiempo real de los buses urbanos y la visualización de información adicional como placa,

registro municipal, velocidad, hora y fecha de su última posición en el mapa.
(Loayza Abad, 2015)

Este artículo profundiza la georreferenciación en tiempo real de los buses urbanos y la visualización de toda la información que el usuario busca, también es otro punto de partida para la investigación, desarrollo e implementación del proyecto.

La oferta tecnológica en cuanto a aplicaciones móviles en el mercado local crece cada vez más, tanto así que hoy el negocio contempla desde aplicaciones que ayudan a tomar un taxi, buscar un número telefónico, redes sociales, realizar transferencias bancarias, hacer pedidos a domicilio, vigilar negocios a través de cámaras Wireless o simplemente revisar correos en tiempo actual ya sean personales o empresariales. “En el 2015, 3’084.886 ecuatorianos declararon tener un teléfono inteligente (Smartphone) casi cinco veces más que lo reportado en el 2011 cuando la cifra era de 522.640 personas” (www.ecuadorencifras.gob.ec, 2016), y más de 17,8 millones de usuarios de celulares.

En cuanto a la existencia de aplicaciones móviles que den información de buses urbanos e interprovinciales en el país son pocas, por lo tanto esta aplicación móvil sirve a los usuarios y a la mejora del servicio de las cooperativas de buses interprovinciales del Ecuador.

La aplicación móvil resultante está en capacidad de enlazar algunas funciones de las tesis descritas anteriormente en cuanto a la aplicación móvil que se desarrolla en este proyecto, pero que son desarrolladas con los requerimientos que propone el tutor de este proyecto de titulación ya que no se cuenta con el auspicio de una entidad pública ni privada, permitiendo la ubicación de los terminales terrestres, información de las

cooperativas de buses de transporte interprovincial en todo el territorio ecuatoriano, compra de boletos y pudiendo visualizar toda su información en la aplicación móvil.

Por lo tanto los beneficios y la utilidad de la aplicación móvil, en estos momentos solo es para las personas que posean un teléfono móvil Android, mostrando toda la información que puedan proporcionar las cooperativas de buses de transporte interprovincial, tratando de mantener actualizada la información de cada cooperativa o terminal terrestre, tratando constantemente de ir añadiendo más terminales o cooperativas a la base de datos de la aplicación móvil, mostrando la ubicación precisa de los terminales terrestres e información de cada cooperativa de bus interprovincial que se encuentre operando en el Ecuador. Además de la posibilidad de realizar la compra de boletos para las cooperativas de buses de transporte interprovincial que quieran adherirse a este proyecto.

1.2 Problema

La FENACOTIP cuenta con 300 cooperativas y empresas afiliadas, con un estimado de 30 socios por cooperativa, lo cual da como resultado una media de diez mil personas asociadas, convirtiéndola en la federación más grande del país. (FENACOTIP, 2016), siendo una gran problemática para los pasajeros (ciudadanos) frecuentes la falta de información acerca de dichas cooperativas en cuanto a sus rutas, unidades, horarios, frecuencias y otros. Quienes al no poder obtener información fiable y oportuna acerca de las cooperativas de buses que podrían transportarlos hacia su ciudad destino tienen que acudir a los terminales terrestres, sedes de las cooperativas o realizar llamadas telefónicas para obtener información de manera que el usuario debe hacer uso de un tiempo extra en sus actividades diarias.

La falta de desarrollo tecnológico en el país no cubre los aspectos informativos de la población que realiza viajes en transporte terrestres frecuentes o vacacionales, por ende, sus opciones de tener información oportuna y fiable son limitadas y están regidas a su capacidad de obtenerlas, a pesar del avance tecnológico, los procesos de globalización y las nuevas políticas gubernamentales, aun se tiene la necesidad de hacer una revisión de la organización en todos sus niveles y estructura, en la parte administrativa, organizativa, de gestión y financiera.

Los ciudadanos desean obtener información confiable y oportuna acerca de los buses de las cooperativas de transporte interprovincial del Ecuador que podrían llevarlos a su destino pese a contar con medios tradicionales de información como lo son: radio, TV, internet, periódicos y otros, pero esta información casi siempre se encuentra desactualizada o no se dispone a cada momento.

La poca información e ineficiencia en la difusión de los recorridos de las cooperativas de transporte interprovincial en Ecuador causa desorientación, incertidumbre e inseguridad en los ciudadanos y habituales usuarios de todo el país al momento de abordarlas para trasladarse de una ciudad a otra dentro del territorio ecuatoriano.

Actualmente son pocas las aplicaciones móviles en capacidad de entregar información específica y en tiempo real acerca de cooperativas de transporte interprovincial del Ecuador. Es así que se pretende cubrir esta carencia a través de una aplicación móvil que permitan la interacción de personas (pasajeros) con dichas cooperativas de transporte para lograr la satisfacción del usuario en cuanto a información.

1.3 Justificación

Una de las áreas de actividad más importante en el mercado de servicios es el Transporte Terrestre, un sector en crecimiento el cual, actualmente, está afrontando

retos importantes; el avance tecnológico ha sido de gran ayuda en este esfuerzo con el fin de contribuir al Sumak Kawsay a este sector de la población.

Dentro del Buen Vivir o Sumak Kawsay al que se refiere la Constitución de la República del Ecuador en el del Régimen del Buen Vivir en sus artículos 385 a 388, se detalla: “Desarrollar tecnologías e innovaciones que impulsen la producción nacional, eleven la eficiencia y productividad, mejoren la calidad de vida y contribuyan a la realización del buen vivir.” (Asamblea Constituyente, 2008)

En la actualidad existen pocos sistemas de organización, guía y ayuda en el cual las personas estén mejor informadas de las rutas, ciudad destino y horarios de las diferentes cooperativas de transporte terrestre interprovincial.

Basándose en este problema, y tomando en cuenta que hoy en día el uso de teléfonos móviles inteligentes (smartphone) está en crecimiento, desarrollar una aplicación móvil que contengan información de cooperativas que van hacia una ciudad destino, muestren la ruta, los horarios y otros detalles, de manera rápida, confiable y sencilla, puede ayudar a más de una persona que se quiera movilizar en transporte interprovincial dentro del territorio ecuatoriano.

Ante esta necesidad nace la idea de crear esta aplicación móvil donde el usuario será capaz de interactuar con información completa y actualizada de todas las cooperativas de transporte interprovincial, mediante el uso de herramientas open source, que permite ubicarse, informarse y trasladarse más seguro en su viaje evitando perder tiempo, permitiendo tener un mayor conocimiento de aquellas cooperativas de transporte que generalmente frecuentan los usuarios hacia su ciudad destino y de esta manera estar mejor informados en tiempo real.

El propósito del presente proyecto técnico es de entregar información simple, efectiva y disponible, donde por medio de una aplicación móvil al seleccionar el nombre de una ciudad destino la aplicación móvil debe dar al usuario todos los nombres de las cooperativas de transporte interprovincial que pueden llevar a un destino deseado con la posibilidad de obtener información más detallada de cada cooperativa seleccionada, además de todos los horarios existentes para la cooperativa seleccionada que están almacenados en una base de datos en la nube y también la aplicación a través de la geolocalización podrá obtener la ubicación actual, direcciones, nombres y el trazo de la ruta entre ubicación actual y terminal terrestre seleccionado dentro del mapa del Ecuador. Finalmente la aplicación puede realizar la compra de boletos de asientos para buses interprovinciales con una fecha determinada.

1.4 Objetivo general

Desarrollar e implementar una aplicación móvil para consulta de cooperativas de buses, con el fin de automatizar y gestionar la búsqueda de cooperativas de transporte interprovincial hacia un destino deseado en todo el territorio ecuatoriano.

1.5 Objetivos específicos

Realizar el levantamiento de requerimientos para obtener información de las necesidades que establece el cliente y satisfacerlas mediante la aplicación.

Obtener información detallada de las rutas, horarios y otros datos de las cooperativas de transporte interprovincial del Ecuador.

Desarrollar el diseño de las interfaces necesarias para la aplicación móvil, tomando en consideración los servicios y funcionalidades que se establecen en los requerimientos del cliente

Diseñar la base de datos que almacenará la información de cada cooperativa de transporte interprovincial, con un modelo entidad relación.

Realizar pruebas funcionales con los módulos desarrollados en la aplicación móvil para que trabaje sin inconvenientes.

Generar pruebas de usabilidad y rendimiento del aplicativo móvil a través de la opinión de los clientes que accedan a ella, evaluar los tiempos de obtención de información y conocer la aceptación y el grado de satisfacción que genera para evitar problemas a futuro en los módulos de la aplicación.

1.6 Marco metodológico

En la actualidad, el software es un común denominador entre las personas, ya sea en un nivel educativo, profesional, laboral o personal. La informática en todos los puntos de vista se ha vuelto indispensable en el vivir diario de las personas, esto ha hecho que sea todavía más importante el desarrollo de software, ya que cada día las personas demandan más calidad en los servicios informáticos, de tal manera que siempre se necesita estar actualizado en el software. Por lo tanto el producto a desarrollarse sirve para solventar problemas de automatización y cubrir necesidades específicas de las personas. Este desarrollo del software debe cumplir con ciertas especificaciones para obtener un producto final de calidad, siendo así la parte inicial del desarrollo el análisis del problema y la toma de requerimientos, siguiendo con la etapa de codificación y las pruebas respectivas, para finalmente implementar el producto.

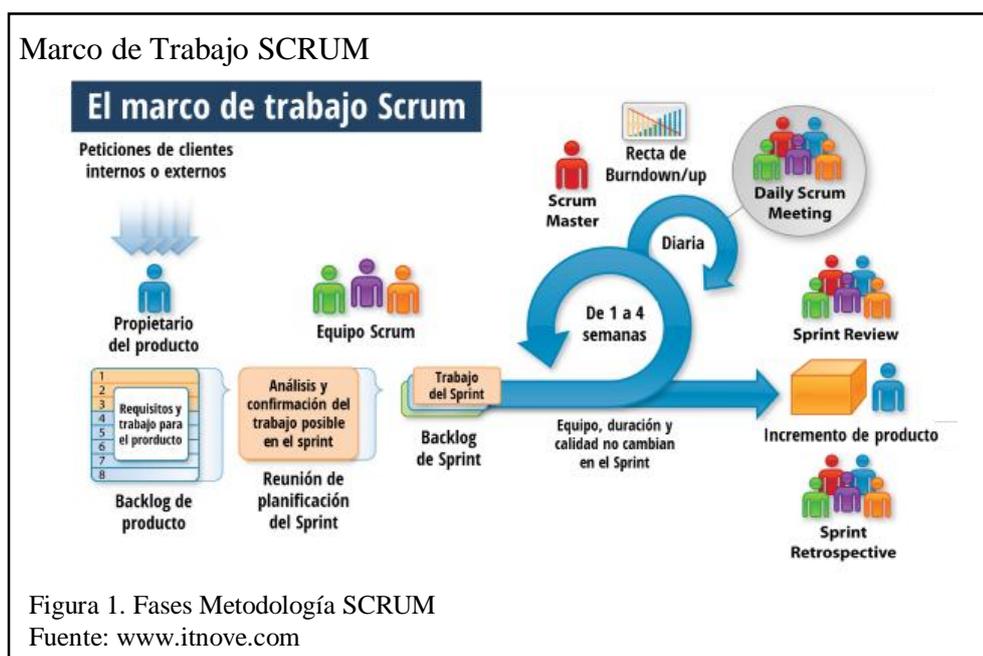
El presente proyecto desarrollado abarca temas de georreferenciación de las terminales terrestres e información de las cooperativas de buses interprovinciales con el desarrollo de una aplicativo móvil, para lo cual lo más conveniente es escoger una metodología de desarrollo acorde a la elaboración de la propuesta del proyecto, ya que se incorpora

una serie de buenas prácticas, para poder tomar decisiones y saber cómo enfrentarse a requerimientos o cambios inesperados que no se hayan detectado con anterioridad.

“Las metodologías pueden adaptarse según las necesidades de la empresa o proyecto. Todas promueven valores como comunicación, compromiso, seguimiento, empatía en los miembros del equipo, entre otras. (SG Buzz, 2012)” La metodología Scrum es una manera de trabajo en equipo en el que se busca obtener resultados funcionales de manera efectiva para proyectos de programación Web o de aplicaciones móviles.

“Este método ágil de trabajo es una manera de realizar proyectos complejos que inicialmente fue creado para el desarrollo de programación pero funciona con cualquier otro tipo de proyecto. (www.imaginanet.com, 2016)”

La metodología a utilizar dentro del desarrollo de la aplicación móvil es SCRUM, el cual se define como una metodología de desarrollo ágil, definiendo la forma de abordar un proceso de desarrollo de software de forma ágil y liviana, a través de la descripción de un conjunto de roles, componentes y organización de la actividad diaria (VASS digital, 2012).



Para que tengas una idea rápida, para que un proyecto ingrese al marco de lo que es el **modelo Scrum**, debe contar con las siguientes características (OK HOSTING, 2016):

- **Desarrollo Incremental.** Una metodología ágil sin desarrollo incremental, no puede ser considerada Scrum. Con incremental hago énfasis a olvidarnos de la planeación y de la ejecución de las líneas sin salirnos de lo pre establecido, pues con esta metodología, el desarrollo se irá incrementando poco a poco, sin importar el orden en el cual se lleven a cabo los procesos.
- **Calidad de las personas.** Básicamente la calidad de un producto, no será analizada en base a la calidad de cada uno de los procesos llevados a cabo. Al contrario, la calidad dependerá de las personas, la auto organización y el conocimiento de los equipos de trabajo.
- **Adiós al Secuencial y Cascada.** Aquí en el modelo Scrum, hay algo a lo que se le denomina, solapamiento. Esto consiste en que no importa en qué proceso te encuentres, si un proceso necesita ser trabajado, vuelves a él para realizar lo que tienes que hacer, a diferencia de las metodologías cascada o secuencial, donde no había vuelta atrás. Acá afortunadamente no hay ningún problema con eso y la ventaja es que se ahorran tiempos.
- **La comunicación es Fundamental.** Una de las cosas que se realizan, son los equipos de trabajo, sin embargo acá la ventaja que tendrás es que podrás estar en constante comunicación con los otros equipos de trabajo, nadie está envuelto en su propia burbuja y toda la información que se maneje o lleve a cabo, será comunicada sin problema.

1.6.1. Elementos de SCRUM

Los elementos de SCRUM son:

- **Product Backlog:** Es el inventario en el que se almacenan todas las funcionalidades o requisitos en forma de lista priorizada. Estos se encuentran definidos en el cronograma de actividades.
- **Historias de Usuario:** Son las descripciones de las funcionalidades que va a tener el software. Se realizará conforme se entregue los subproductos funcionales.
- **Sprint BackLog:** Es la lista de tareas que elabora el equipo durante la planificación de un Sprint. Estos se encuentran definidos en el cronograma de hitos.
- **Ejecución del Sprint:** Sería el periodo de entre 1 y 4 semanas (periodo definido previamente en el sprint backlog). Una vez iniciada la ejecución de un sprint definido, este no podrá ser modificado, y en caso de ser necesario introducir cambios estos se harán una vez concluido el periodo a través de la definición de otro sprint backlog.
- **Scrum diario:** Cada día, se realizará un daily scrum entre el equipo y el Scrum Master que hará un repaso del estado actual del sprint, dificultades que presenta el proyecto y cómo afrontarlas de la mejor manera.

Los objetivos de esta reunión son:

- Progreso realizado desde la última reunión
 - Sincronizar tareas con dependencias entre ellas
 - Planificar las prioridades de las siguientes tareas a realizar del product backlog.
 - Preparar futuras complicaciones que puedan suceder
- **Entrega:** Una vez concluida la ejecución del sprint, se dispondrá de una porción de la aplicación potencialmente definitiva.

1.6.2. Los roles de SCRUM

Se dividen de la siguiente manera:

- **Product Owner:** Es la única persona autorizada para decidir sobre cuáles funcionalidades y características funcionales tendrá el producto.
- **Scrum Master:** Es la persona que lidera las reuniones, guiando al equipo para que cumpla las reglas y procesos de la metodología.
- **Desarrollador:** Es un programador disciplinado que en forma auto-organizada será el encargado de desarrollar el producto comprometido a las historias de usuario al inicio de cada actividad (sprint).
- **Stakeholder:** Es la persona a la que el proyecto producirá un beneficio.

1.6.3. Reuniones

Las reuniones están planificadas de la siguiente manera:

- **Sprint Daily Meeting:** no durará más de 15 minutos, el objetivo de esta reunión es facilitar la transferencia de información y la colaboración entre los miembros del equipo (Scrum Master y Desarrollador) para aumentar su productividad.
- **Sprint Review Meeting:** Se presenta el producto entregable que han implementado, el Product Owner, lo analizan y escuchan sobre problemas durante el proceso. Máximo 2 horas

1.7 Marco teórico

El desarrollo de aplicaciones móviles supone un gran desafío debido a la gran disparidad de dispositivos móviles, sistemas operativos y características de los dispositivos ya que se pretende satisfacer al usuario final.

Una aplicación es un programa que se instala en un dispositivo móvil, puede ser un teléfono o una tablet, y que se integra a los elementos que componen el equipo, como su cámara o GPS. Proveen acceso directo a un contenido sin tener que buscarlo en Internet, y en ocasiones incluso sin acceso a la Red. (AS Technology, 2016)

Para poder entender un mejor este concepto, podemos decir que las aplicaciones son para los móviles lo que los programas son para los ordenadores de escritorio.

Un buen desarrollo de aplicaciones móviles además de requerir una interfaz de usuario atractiva. También debe implicar:

- Pruebas
- Integración
- Seguridad
- Control de calidad
- Gestión constante

El desarrollo de aplicaciones móviles para georreferenciación de lugares, ubicaciones o coordenadas requiere de insumos teóricos, las diferentes tecnologías, las técnicas de geolocalización o los sistemas de representación de mapas y que en esta investigación constituyen elementos importantes para su análisis, los cuales se relacionan con:

- Programación orientada a objetos
- Sistemas operativos
 - Android
- Geolocalización
 - Georreferenciación
 - Geocodificación

- Geoetiquetado
- Aplicaciones Móviles
 - Api de Google Maps
 - Google Maps
- Bases de datos relacionales

1.7.1. Programación orientada a objetos

La programación orientada a objetos es un paradigma de programación que usa los objetos que son entidades que combinan comportamiento, estado e identidad y sus interacciones para diseñar programas y aplicaciones informáticos.

Características específicas de la programación orientada a objetos (ALEGSA, 2017):

- Encapsulamiento: significa reunir todos los elementos que pueden considerarse pertenecientes a una misma entidad, al mismo nivel de abstracción. Esto permite aumentar la cohesión de los componentes del sistema.
- Polimorfismo: el polimorfismo se refiere a la propiedad por la que es posible enviar mensajes sintácticamente iguales a objetos de tipos distintos. El único requisito que deben cumplir los objetos que se utilizan de manera polimórfica es saber responder al mensaje que se les envía.
- Herencia: Las clases se relacionan entre sí y permiten formar una jerarquía de clasificación. Los objetos heredan propiedades y métodos (comportamientos) de las clases a las que pertenecen.

1.7.2. Sistemas operativos

Un sistema operativo o software de sistema es un conjunto de programas que administran de forma eficaz los recursos de un ordenador permitiendo la ejecución normal de todas las operaciones.

Dentro de las funcionalidades básicas de los sistemas operativos, se pueden describir tres categorías:

- Gestión de recursos de la computadora
- Ejecución de los servicios para los programas.
- Ejecución de los mandos de los usuarios.

El SO está formado conceptualmente por tres capas principales. La capa más cercana al hardware se denomina núcleo (kernel) y es la que gestiona los recursos hardware del sistema y la que suministra la funcionalidad básica del SO. Esta capa se ha de ejecutar en nivel núcleo, mientras que las otras se pueden ejecutar en niveles menos permisivos (Aguirre, 2011).

Los sistemas operativos para dispositivos móviles son menos robustos de manera que en un dispositivo móvil no se puede hacer lo mismo que una portátil, además son orientados hacia la conectividad inalámbrica y a necesidades específicas.

1.7.3. Android

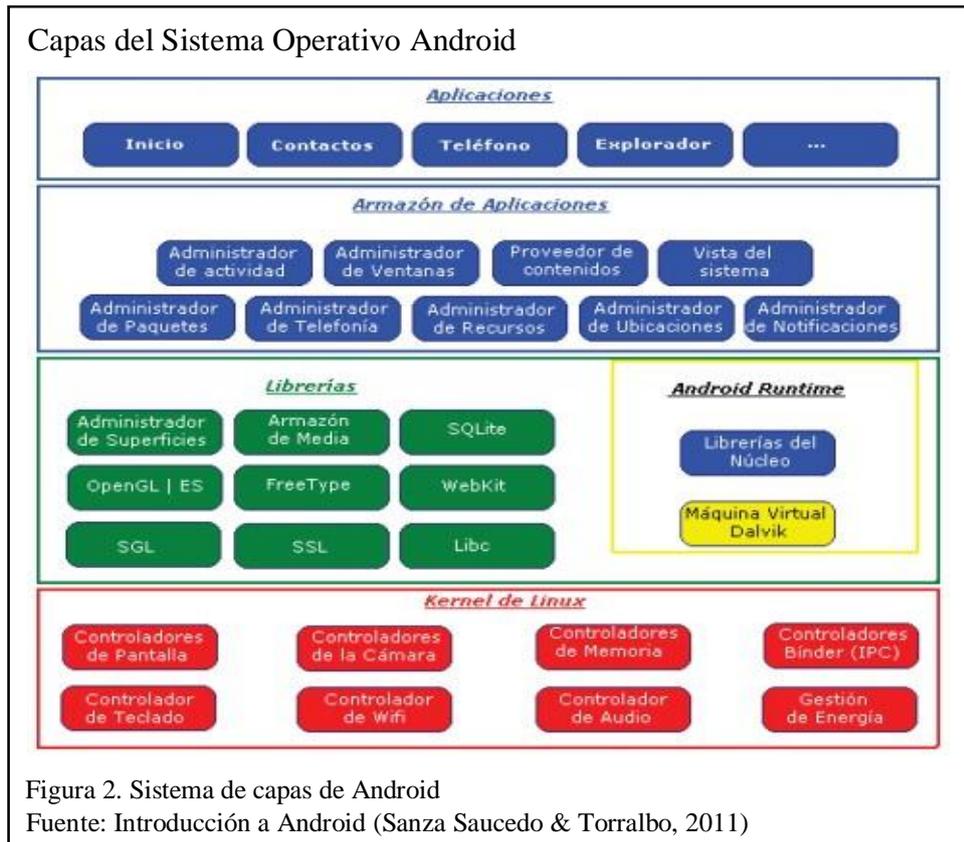
Android es un sistema operativo y una plataforma software, basado en Linux para teléfonos móviles. Además, usan este sistema operativo (aunque no es muy habitual), tablets, netbooks, reproductores de música e incluso PC's. Android permite programar en un entorno de trabajo (framework) de Java, aplicaciones sobre una máquina virtual Dalvik (una variación de la máquina de Java con compilación en tiempo de ejecución). Además, lo que le diferencia de otros sistemas operativos, es que cualquier persona que sepa programar puede crear nuevas aplicaciones, widgets, o incluso, modificar el propio sistema operativo, dado que Android es de código libre, por lo que sabiendo programar

en lenguaje Java, va a ser muy fácil comenzar a programar en esta plataforma (Sanz, Saucedo, & Torralbo, 2011).

Android posee las siguientes características:

- Navegador integrado
- Framework de aplicaciones
- Dalvik, máquina virtual
- Pantalla Táctil
- SQLite
- Multimedia: Soporte para medios con formatos comunes de audio, video e imágenes planas
- Bluetooth, 3G y wifi, cámara, GPS

A continuación se muestra las capas del sistema operativo Android:



En la Figura 2 se muestran cada una de las capas: que forma parte del propio Kernel de Linux, donde Android puede acceder a diferentes controladores, librerías creadas para el desarrollo de aplicaciones Android, la capa que sigue organiza los diferentes administradores de recursos, y la última capa de las aplicaciones a las que tiene acceso.

1.7.4. Geolocalización

Entendemos por geolocalización al conjunto de técnicas que permiten determinar la posición geográfica de un elemento (un ordenador, un teléfono móvil o cualquier dispositivo capaz de ser detectado) en el mundo real y hacer uso de esa información. Esta tecnología requiere de la perfecta sincronización entre hardware y software, es necesario un dispositivo con GPS o conexión a internet y un software que permita hacer uso de ellos en esta dirección.

En los últimos años los smartphone se han tornado el dispositivo ideal para la geolocalización gracias al hardware que incorporan y a que sus fabricantes han dotado

sus sistemas operativos de las herramientas necesarias para que los desarrolladores hagan uso de la geolocalización con facilidad y puedan centrarse en explorar sus múltiples utilidades. Entre ellas podemos diferenciar tres usos comunes:

Georreferenciación: Es el proceso mediante el que se localiza un objeto, lugar o persona en el espacio físico para posteriormente representarlo en un sistema de coordenadas o mapa. Un ejemplo habitual es la representación de tu posición en el mapa de tu ciudad y actualizarla a medida que te desplazas.

Geocodificación: Es el proceso de obtención de coordenadas geográficas a partir de otro tipo de datos geográficos, como la dirección o el código postal. Al proceso contrario, la obtención de direcciones postales a partir de coordenadas se le denomina geocodificación inversa. El ejemplo claro lo vemos en la aplicación Google Maps, que muestra en un mapa el punto donde quieres después de haber indicado la dirección postal.

Geoetiquetado: Es el proceso mediante el cual se añade información geográfica en forma de metadatos a otro tipo de contenido. Usualmente es un paso posterior a la georreferenciación. Un ejemplo de geoetiquetado sería incluir en una fotografía las coordenadas del lugar donde fue tomada (Sanchez, 2012).

1.7.5. Aplicaciones Móviles

Las aplicaciones móviles permiten crear y mantener una relación con el cliente. Es la mejor herramienta para acceder de manera rápida a la información, creando una valiosa interacción que posiciona a la marca y al mismo tiempo ofrece una imagen de innovación. Permite realizar ofertas especiales a clientes, obtener su opinión, facilitar su compra tanto online como física y motivar su

recomendación a otros clientes potenciales en redes sociales (kieranet.com, 2016).

Para el desarrollo de aplicaciones móviles se opta por el uso APIs que simplifican el trabajo del desarrollador y en este proyecto para la parte de georreferenciación se utiliza la API de Google Maps.

1.7.6. API de Google Maps

La API de Google Maps permite incorporar Google Maps en páginas web de desarrolladores externos, utilizando una interfaz JavaScript sencilla o una interfaz de Flash. Está diseñado para funcionar tanto en dispositivos móviles como en aplicaciones de navegador de escritorio tradicionales. La API incluye localización de idiomas para más de 50 idiomas, localización de regiones y Geocodificación, y tiene mecanismos para desarrolladores de empresas que desean utilizar la API de Google Maps dentro de una intranet. Los clientes API de Google Maps API pueden acceder a los servicios API HTTP a través de una conexión segura (HTTPS) (ProgrammableWeb, 2017).

1.7.7. Google Maps

Google Maps es una aplicación de Google Inc. servidor de aplicaciones de mapas en la web que ofrece imágenes de mapas desplazables, así como fotos satelitales del mundo, e incluso la ruta entre diferentes ubicaciones o imágenes a pie. Es idéntico a Google Earth, una aplicación que ofrece vistas del globo terráqueo, sea de día o de noche, pero que no es fácil de integrar a páginas Web. Está disponible para Android y Java ME (Plataforma para el lenguaje de Java). Google Maps ofrece la capacidad de hacer acercamientos o alejamientos para mostrar el mapa. El usuario puede controlar el mapa con el mouse o las teclas

de dirección para moverse a la ubicación que se desee. Los usuarios pueden ingresar una dirección, una intersección o un área en general para buscar en el mapa (Ruiz, Díaz, & Cervantes, 2014).

1.7.8. Base de datos relacional

Una base de datos relacional se compone de varias tablas o relaciones y al tener independencia lógica y física se opta el uso del diseño de la base de datos relacional en este proyecto por su facilidad de crear y acceder a sus tablas que poseen registros, filas o tuplas.

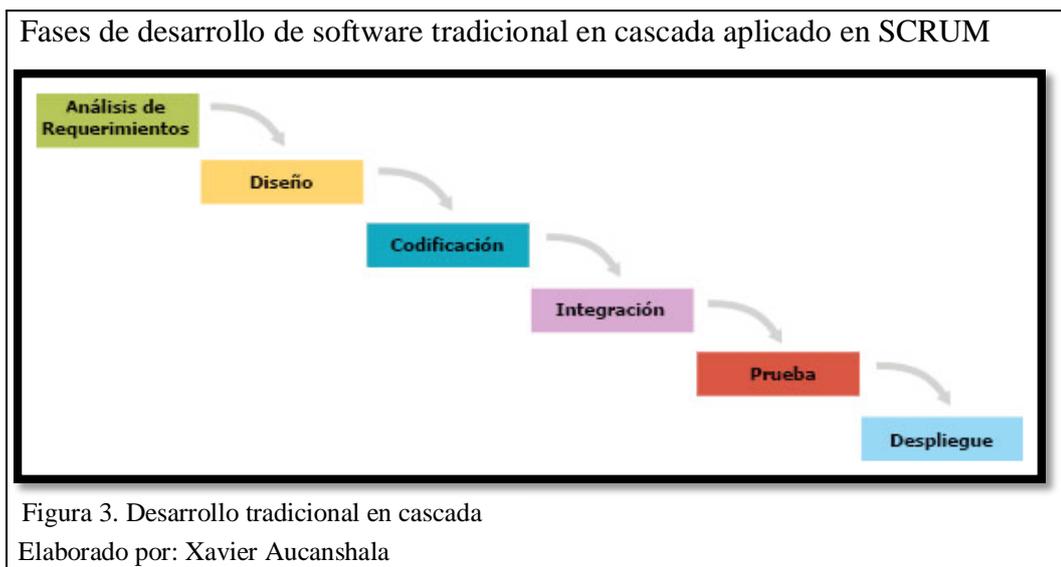
Una Base de datos relacional es una colección de datos cuya característica principal es que los datos pueden almacenarse y administrarse en forma de tablas. Al hablarse de bases de datos relacionales, significa que se pueden crear relaciones entre las tablas de las bases de datos. Una relación entre tablas consiste en que algunos registros de una tabla tengan datos en común con registros de otras tablas, permitiendo un manejo más eficiente y sin redundancia (Nevado, 2010).

Capítulo 2

ANÁLISIS Y DISEÑO

El producto final es una aplicación móvil y a continuación se detallan el desarrollo de las fases de análisis y diseño del producto final, empezando con el análisis de requerimientos, que son necesarios para el desarrollo y el diseño de la aplicación móvil, un diagrama físico de entidad-relación de la base de datos en MySQL y la construcción de una interfaz gráfica Android para la aplicación móvil, aplicando la metodología SCRUM.

En las reuniones de planificación se trata temas fundamentales como los requerimientos funcionales que tiene la aplicación móvil, la viabilidad del proyecto y la escalabilidad a futuro, para poder establecer los sprint, historias de usuario y demás elementos de la metodología descrita posteriormente. Así optimizar la responsabilidad del desarrollo por parte del programador, y la revisión de los avances por parte del Product Owner para poder realizar las observaciones al finalizar cada sprint.



2.1. Análisis de requerimientos

La finalidad del análisis de requerimientos es describir lo más detalladamente posible todos los requerimientos funcionales y no funcionales del software. A continuación se muestra todas las características que definen a la aplicación móvil, el rol del usuario, las interfaces y la funcionalidad que debe realizar todos los módulos de la aplicación móvil interactuando en conjunto.

2.1.1 Identificación de historias de usuario

En la aplicación móvil solo contiene un usuario, que puede ser una persona con un dispositivo móvil Android y está en capacidad de realizar estas funciones:

- Inicio de la aplicación
- Visualizar un menú principal que consta de cuatro opciones: Cooperativas, Horarios, Terminales y Boletos
- En la opción de cooperativas, buscar cooperativas de transporte interprovincial seleccionando su ciudad origen, su ciudad destino y mostrar una lista completa de todas las cooperativas que van hacia dicha ciudad destino
- Al seleccionar una cooperativa de transporte interprovincial en específico se muestra toda la información detalladamente de la cooperativa de transporte interprovincial antes seleccionada
- En la opción de horarios, buscar las cooperativas de transporte interprovincial seleccionando su ciudad origen, su ciudad destino y mostrar una lista completa de todas las cooperativas que van hacia dicha ciudad destino
- Al seleccionar una cooperativa de transporte interprovincial en específico se muestra la información de todos los horarios y precios correspondientes a dicha cooperativa de transporte interprovincial

- En la opción de terminales, se observa un mapa con todas las terminales terrestres disponibles mostrando nombre y dirección de cada terminal.
- En la opción de boletos, buscar todos los turnos disponibles seleccionando una cooperativa, una ciudad origen y una ciudad destino
- Al seleccionar un turno de una cooperativa de transporte interprovincial en específico, el usuario puede seleccionar una fecha actual o posterior así mismo un asiento que esté disponible
- Realizar el pago del boleto mediante una tarjeta de crédito a través de captura de datos mediante el uso de la cámara del Smartphone
- Enviar a un correo electrónico del usuario toda la información necesaria que garantice la compra del boleto

Todas las funcionalidades importantes que presenta la aplicación móvil se basan en los requerimientos iniciales que solicita el cliente.

2.1.2 Historias de usuario

Las historias de usuario se define como un conjunto de “tarjetas escritas en lenguaje de negocio que sirve como identificación y recordatorio del requerimiento y ayuda para la planificación mediante la priorización” (Mora, 2010), siendo una manera simple de describir una tarea concisa que aporta valor al usuario o al negocio permitiendo obtener todos los requerimientos del sistema a desarrollar. Para especificar todos los requerimientos que el cliente solicita se elaboran tablas estructuradas para cada historia de usuario.

Ahora se procede a detallar las historias de usuario del cliente, que corresponde al usuario general que va a ingresar a la aplicación móvil.

2.1.3 Requerimientos del usuario de la aplicación móvil

La finalidad de esta historia de usuario es describir los requerimientos del usuario general que accede a la aplicación móvil, el cual está registrado en la respectiva historia de usuario, acordado entre el cliente y el desarrollador.

La Tabla 1 explica los requerimientos del usuario general de la aplicación que muestra un menú de opciones de búsqueda de: Cooperativas, Horarios y Terminales con botones personalizados

Tabla 1. Historia de usuario: mostrar un menú de opciones de búsqueda

Mostrar menú de opciones de búsqueda	
No. 01	Usuario: Usuario general de la aplicación móvil
Nombre de historia: Mostar menú de opciones de búsqueda	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 02	Iteración asignada: 01
Programador responsable: Xavier Aucanshala	
Descripción: El usuario general de la aplicación móvil es capaz de visualizar un menú de búsqueda atractivo con cuatro opciones: Cooperativas, Horarios, Terminales y Boletos.	
Observaciones: Se Confirma con el cliente	

Nota: Historia de usuario para ver un menú de opciones de búsqueda

Elaborado por: Xavier Aucanshala

La Tabla 2 explica los requerimientos del usuario general de la aplicación que busca todas las cooperativas de transporte interprovincial que va hacia una ciudad destino que el usuario seleccione.

Tabla 2. Historia de usuario: buscar las cooperativas seleccionando una ciudad destino

Buscar cooperativas seleccionando una ciudad destino	
No. 02	Usuario: Usuario general de la aplicación móvil
Nombre de historia: Buscar cooperativas seleccionando una ciudad destino	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 03	Iteración asignada: 02
Programador responsable: Xavier Aucanshala	
Descripción: El usuario de la aplicación móvil es capaz de buscar todas las cooperativas de transporte interprovincial registradas seleccionando una ciudad destino.	
Observaciones: Se Confirma con el cliente	

Nota: Historia de usuario para buscar todas las cooperativas posibles seleccionando una ciudad destino

Elaborado por: Xavier Aucanshala

La Tabla 3 explica los requerimientos del usuario general de la aplicación que muestra la información detallada de una cooperativa de transporte interprovincial que el usuario seleccione.

Tabla 3. Historia de usuario: mostrar la información detallada de una cooperativa

Mostrar información de una cooperativa seleccionada	
No. 03	Usuario: Usuario general de la aplicación móvil
Nombre historia: Mostrar información de una cooperativa seleccionada	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 03	Iteración asignada: 01
Programador responsable: Xavier Aucanshala	
Descripción: El usuario de la aplicación móvil es capaz de ver toda la información detallada de una cooperativa de transporte interprovincial registrada a través de una ciudad destino seleccionada	

<p>Observaciones:</p> <p>Se Confirma con el cliente</p>

Nota: Historia de usuario para mostrar la información detallada de una cooperativa seleccionada

Elaborado por: Xavier Aucanshala

La Tabla 4 explica los requerimientos del usuario general de la aplicación que busca en la opción de horarios todas las cooperativas de transporte interprovincial que va hacia una ciudad destino que el usuario seleccione.

Tabla 4. Historia de usuario: buscar las cooperativas seleccionando una ciudad destino

Buscar cooperativas seleccionando una ciudad destino	
No. 04	Usuario: Usuario general de la aplicación móvil
Nombre historia: Buscar cooperativas seleccionando una ciudad destino	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 03	Iteración asignada: 01
Programador responsable: Xavier Aucanshala	
<p>Descripción:</p> <p>El usuario de la aplicación móvil en la opción horarios es capaz de buscar todas las cooperativas de transporte interprovincial registradas seleccionando una ciudad destino.</p>	
<p>Observaciones:</p> <p>Se Confirma con el cliente</p>	

Nota: Historia de usuario para buscar todas las cooperativas posibles seleccionando una ciudad destino

Elaborado por: Xavier Aucanshala

La Tabla 5 explica los requerimientos del usuario general de la aplicación que muestra todos los horarios y precios de pasaje de una cooperativa de transporte interprovincial que el usuario seleccione.

Tabla 5. Historia de usuario: mostrar los horarios y precios de pasaje de una cooperativa

Mostrar horarios y precios de una cooperativa seleccionada
--

No. 05	Usuario: Usuario general de la aplicación móvil	
Nombre historia: Mostrar horarios y precios de una cooperativa seleccionada		
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto	
Puntos estimados: 03	Iteración asignada: 01	
Programador responsable: Xavier Aucanshala		
Descripción: El usuario de la aplicación móvil es capaz de ver todos los horarios y precios de pasaje de una cooperativa de transporte interprovincial registrada a través de una ciudad destino seleccionada		
Observaciones: Se Confirma con el cliente		

Nota: Historia de usuario para mostrar los horarios y precios de una cooperativa seleccionada

Elaborado por: Xavier Aucanshala

La Tabla 6 explica los requerimientos del usuario general de la aplicación que visualiza en el mapa todas las terminales terrestres registradas en la aplicación sobre el territorio ecuatoriano a través de iconos sobre cada terminal terrestre.

Tabla 6. Historia de usuario: para visualizar todas las terminales terrestres en el mapa

Ver las terminales terrestres en el mapa		
No. 06	Usuario: Usuario general de la aplicación móvil	
Nombre historia: Ver las terminales terrestres en el mapa		
Prioridad en negocio: Media	Riesgo en desarrollo: Medio	
Puntos estimados: 02	Iteración asignada: 01	
Programador responsable: Xavier Aucanshala		
Descripción: El usuario de la aplicación móvil es capaz de visualizar todas las terminales terrestres registradas en la aplicación que se encontrarán dentro del mapa del Ecuador.		
Observaciones: Se Confirma con el cliente		

Nota: Historia de usuario para ver todas las terminales terrestres dentro del mapa de Ecuador

Elaborado por: Xavier Aucanshala

La Tabla 7 explica los requerimientos del usuario general de la aplicación que busca en la opción de boletos todos los turnos seleccionando una cooperativa, una ciudad de origen y una ciudad de destino que el usuario desee.

Tabla 7. Historia de usuario: buscar los turnos de una cooperativa

Buscar turnos de cooperativas seleccionando una ciudad origen y destino	
No. 07	Usuario: Usuario general de la aplicación móvil
Nombre historia: Buscar turnos de cooperativas seleccionando una ciudad origen y destino	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 03	Iteración asignada: 01
Programador responsable: Xavier Aucanshala	
Descripción: El usuario de la aplicación móvil en la opción boletos es capaz de buscar todos los turnos seleccionando una cooperativa de transporte interprovincial registrado, una ciudad origen y una ciudad de destino.	
Observaciones: Se Confirma con el cliente	

Nota: Historia de usuario para buscar todos los turnos seleccionando una cooperativa, una ciudad origen y una ciudad destino

Elaborado por: Xavier Aucanshala

La Tabla 8 explica los requerimientos del usuario general de la aplicación que escoge una fecha actual o posterior y un asiento disponible de entre 43 asientos de un bus para el turno seleccionado anteriormente.

Tabla 8. Historia de usuario: escoger una fecha actual o posterior y un asiento disponible

Escoger una fecha y un asiento disponible para un turno preseleccionado

No. 08	Usuario: Usuario general de la aplicación móvil
Nombre historia: Escoger una fecha y un asiento disponible para un turno seleccionado	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 03	Iteración asignada: 01
Programador responsable: Xavier Aucanshala	
Descripción: El usuario de la aplicación es capaz de escoger en un calendario la fecha actual o posterior y un asiento disponible dentro de un total de 43 asientos de un bus para el turno que ha seleccionado anteriormente.	
Observaciones: Se Confirma con el cliente	

Nota: Historia de usuario para escoger una fecha y un asiento disponible para un turno preseleccionado destino

Elaborado por: Xavier Aucanshala

La Tabla 9 explica los requerimientos del usuario general de la aplicación que realiza el pago en línea del boleto con tarjeta de crédito con toda la información seleccionada y se lo hace a través de la captura de datos de la tarjeta de crédito con la cámara del Smartphone y enviando todos los datos necesario que garanticen la compra del boleto a un correo electrónico personal

Tabla 9. Historia de usuario: Para realizar el pago de un boleto de bus interprovincial

Realizar el pago en línea de un boleto de bus interprovincial	
No. 09	Usuario: Usuario general de la aplicación móvil
Nombre historia: Realizar el pago en línea de un boleto de bus interprovincial	
Prioridad en negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 03	Iteración asignada: 01
Programador responsable: Xavier Aucanshala	

<p>Descripción: El usuario de la aplicación móvil es capaz de realizar el pago en línea de un boleto de bus interprovincial a través de una tarjeta de crédito y garantizando la compra del boleto con el envío de toda la información necesaria a un correo electrónico personal que el usuario ingrese</p>
<p>Observaciones: Se Confirma con el cliente</p>

Nota: Historia de usuario para el pago en línea con tarjeta de crédito de un boleto de bus interprovincial

Elaborado por: Xavier Aucanshala

2.2 Diseño

"El milagro más común de la ingeniería de software es la transición del análisis al diseño y del diseño al código" Richard Due.

El diseño del software es el proceso del diseño para planificar una solución de software, siendo una de las partes que componen las fases del ciclo de vida de la aplicación móvil. Una vez que se analizan y especifican los requisitos, el diseño del software es la última acción de la metodología correspondiente dentro de la actividad del modelado, la cual establece una plataforma para la construcción (generación de código y prueba), a través de los diferentes diagramas físicos de una base de datos, diagramas de casos de uso, diagramas de clases, diagramas de estados, diagramas de despliegue y por último el diseño de interfaces gráficas para la aplicación móvil. También una parte importante en la base de datos es el diccionario de datos que es un catálogo que detalla los campos y las características lógicas de cada tabla.

2.2.1. Diagramas de casos de uso.

El siguiente diagrama de caso de uso muestra los principales módulos que conforman la aplicación móvil conjuntamente con el único actor que interviene e interactúa.

2.2.1.1. Diagrama de casos de uso para el usuario general de la Aplicación

Móvil.

Un único actor que interviene es un usuario general que se representa en este caso de uso, el cual realiza varias funciones en la aplicación móvil.

La Figura 4 describe al diagrama de casos de uso del actor (usuario general) que realiza las actividades y sub actividades en la aplicación móvil. El actor (usuario general) puede realizar las siguientes actividades: iniciar la aplicación para consultas de cooperativas de transporte interprovincial y terminales terrestres, buscar una cooperativa por la ciudad de destino mostrando la información de rutas y horarios de la cooperativa seleccionada., visualizar en el mapa información general, ubicar una terminal terrestre y comprar un boleto de bus interprovincial con una tarjeta de crédito.

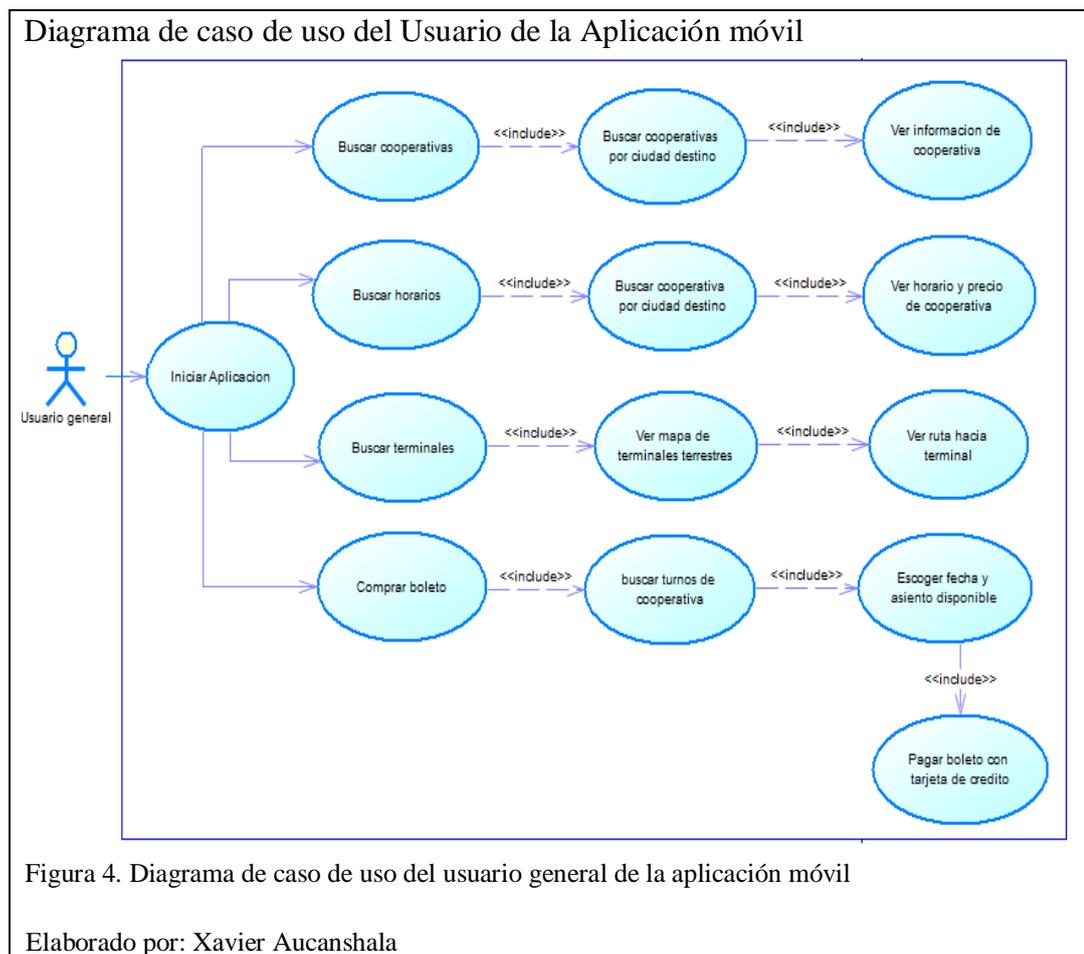
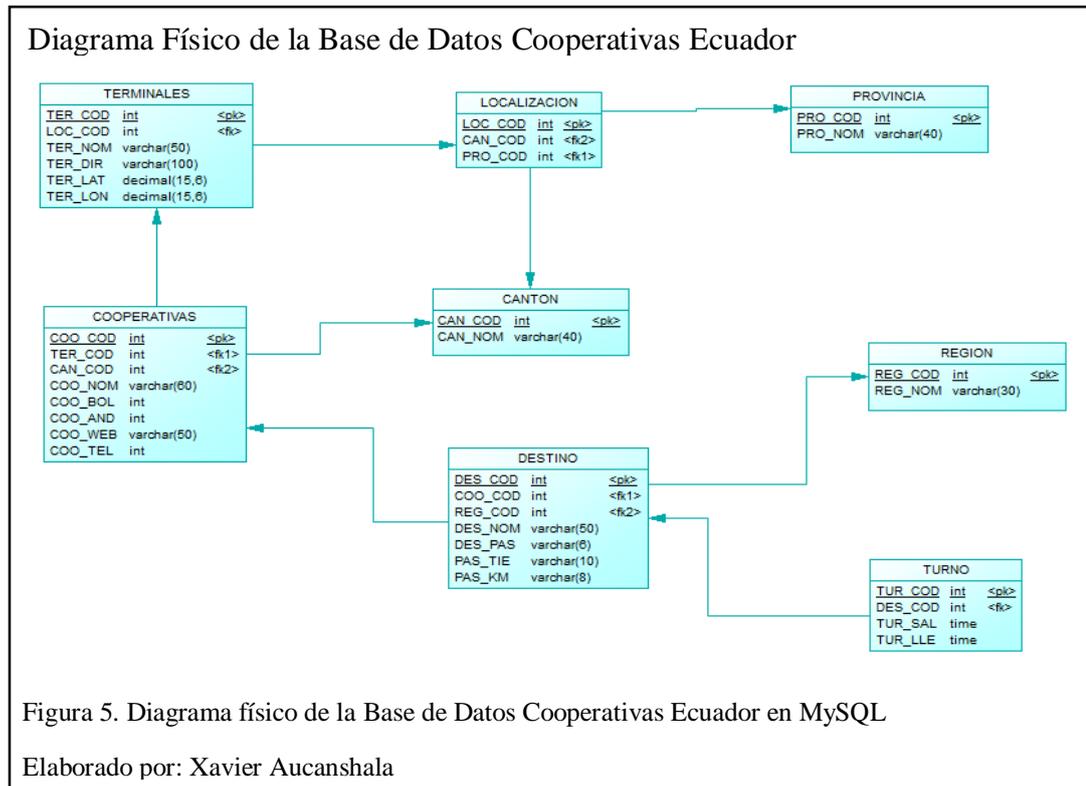


Figura 4. Diagrama de caso de uso del usuario general de la aplicación móvil

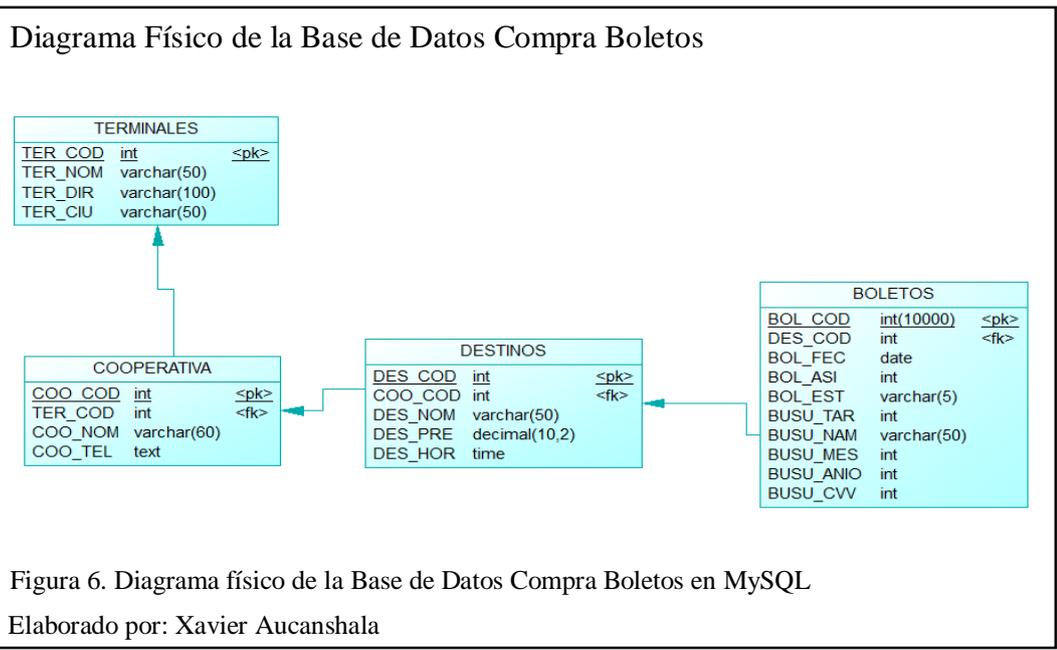
Elaborado por: Xavier Aucanshala

2.2.2. Diagrama físico de la base de datos MySQL

La base de datos MySQL se diseña acorde a los requerimientos que necesita el cliente, el cual se visualiza en el siguiente diagrama físico con las correspondientes tablas relacionadas para almacenar toda la información. Siendo un modelo Entidad-Relación.



En la figura No. 5 muestra una representación física de la base de datos Cooperativas Ecuador en MySQL, donde se visualiza todos los componentes de la base como son: las relaciones entre tablas de uno a uno y de uno a varios, los campos, las claves primarias y las claves foráneas.



En la figura No. 6 muestra una representación física de la base de datos Boletos en MySQL, donde se visualiza todos los componentes de la base como son: las relaciones entre tablas de uno a uno y de uno a varios, los campos, las claves primarias y las claves foráneas.

En la Tabla 10 se muestra la descripción de cada una de las tablas del diseño conceptual de la BDD Cooperativas Ecuador.

Tabla 10. Descripción de las tablas de la base de datos Cooperativas Ecuador MySQL

Tablas	Descripción
cantón	Guarda la información de los cantones
cooperativas	Guarda la información de las cooperativas de transporte interprovincial
destino	Guarda la información de las ciudades destino
localización	Guarda la información de la localización
provincia	Guarda la información de las provincias
región	Guarda la información de las regiones
terminales	Guarda la información de las terminales terrestres
turno	Guarda la información de los turnos

Nota. Descripción de los campos de las tablas de la BDD Cooperativas Ecuador

Elaborado por: Xavier Aucanshala

En la Tabla 11 se muestra la descripción de cada una de las tablas del diseño conceptual de la BDD Compra Boletos.

Tabla 11. Descripción de las tablas de la base de datos Compra Boletos MySQL

Tablas	Descripción
cooperativas	Guarda la información de las cooperativas de transporte interprovincial
destino	Guarda la información de las ciudades destino
terminales	Guarda la información de las terminales terrestres
boletos	Guarda la información de los boletos comprados

Nota. Descripción de los campos de las tablas de la BDD Compra Boletos

Elaborado por: Xavier Aucanshala

2.2.3. Diccionario de datos.

En esta sección se muestra el nombre del campo, el tipo de dato que tiene el campo y la descripción del campo de cada tabla de la base de datos Cooperativas Ecuador y de la base de datos Compra Boletos.

2.2.3.1 Diccionario de datos BDD Cooperativas Ecuador

En la Tabla 12 Se muestra la tabla Cantón que almacena la información de los cantones de cada provincia donde se encuentran las cooperativas de transporte interprovincial y las terminales terrestres.

Tabla 12. Diccionario de datos de la tabla Cantón

Tabla	cantón	
Campo	Tipo de dato	Descripción
CAN_COD	INT	Código del cantón
CAN_NOM	VARCHAR(40)	Nombre de cantón

Nota. Descripción de campos de la tabla Cantón

Elaborado por: Xavier Aucanshala

En la Tabla 13 Se muestra la tabla COOPERATIVAS que almacena la información de las cooperativas de transporte interprovincial.

Tabla 13. Diccionario de datos de la tabla COOPERATIVAS

Tabla	cooperativas	
Campo	Tipo de dato	Descripción
COO_COD	INT	Código de la cooperativa
TER_COD	INT	Código de la terminal (clave foránea)
CAN_COD	INT	Código del cantón (clave foránea)
COO_NOM	VARCHAR(60)	Nombre de la cooperativa
COO_BOL	INT	Boletería de la cooperativa
COO_AND	INT	Anden de la cooperativa
COO_WEB	VARCHAR(50)	Página web de la cooperativa
COO_TEL	TEXT	Teléfono de la cooperativa

Nota. Descripción de campos de la tabla Cooperativas

Elaborado por: Xavier Aucanshala

En la Tabla 14 Se muestra la tabla DESTINO que almacena la información de las ciudades destino que tendrá cada cooperativa.

Tabla 14. Diccionario de datos de la tabla DESTINO

Tabla	destino	
Campo	Tipo de dato	Descripción
DES_COD	INT	Código de la ciudad destino
COO_COD	INT	Código de la cooperativa (clave foránea)
REG_COD	INT	Código de la región (clave foránea)
DES_NOM	VARCHAR(50)	Nombre de la ciudad destino
DES_PAS	VARCHAR(6)	Pasaje de la ciudad destino
DES_TIE	VARCHAR(10)	Tiempo estimado de la ciudad destino
DES_KM	VARCHAR(8)	Distancia en Km de la ciudad destino

Nota. Descripción de campos de la tabla Destino

Elaborado por: Xavier Aucanshala

En la Tabla 15 Se muestra la tabla LOCALIZACIÓN que almacena las claves foráneas de los códigos de cantón y provincias.

Tabla 15. Diccionario de datos de la tabla LOCALIZACIÓN

Tabla	localización	
Campo	Tipo de dato	Descripción
LOC_COD	INT	Código de la ciudad destino
CAN_COD	INT	Código del cantón (clave foránea)
PRO_COD	INT	Código de la provincia (clave foránea)

Nota. Descripción de campos de la tabla Localización

Elaborado por: Xavier Aucanshala

En la Tabla 16 Se muestra la tabla PROVINCIA que almacena la información de cada Provincia del Ecuador.

Tabla 16. Diccionario de datos de la tabla PROVINCIA

Tabla	provincia	
Campo	Tipo de dato	Descripción
PRO_COD	INT	Código de la provincia
PRO_NOM	VARCHAR(40)	Nombre de la provincia

Nota. Descripción de campos de la tabla Provincia

Elaborado por: Xavier Aucanshala

En la Tabla 17 Se muestra la tabla REGIÓN que almacena la información de las regiones del Ecuador.

Tabla 17. Diccionario de datos de la tabla REGIÓN

Tabla	región	
Campo	Tipo de dato	Descripción
REG_COD	INT	Código de la región
REG_NOM	VARCHAR(30)	Nombre de la región

Nota. Descripción de campos de la tabla Región

Elaborado por: Xavier Aucanshala

En la Tabla 18 Se muestra la tabla TERMINALES que almacena la información de las terminales terrestres del Ecuador.

Tabla 18. Diccionario de datos de la tabla TERMINALES

Tabla	terminales
--------------	-------------------

Campo	Tipo de dato	Descripción
TER_COD	INT	Código de la terminal
LOC_COD	INT	Código de localización (clave foránea)
TER_NOM	VARCHAR(50)	Nombre de la terminal
TER_DIR	VARCHAR(100)	Dirección de la terminal
TER_LAT	DECIMAL(15,6)	Latitud de la terminal
TER_LON	DECIMAL(15,6)	Longitud de la terminal

Nota. Descripción de campos de la tabla Terminales

Elaborado por: Xavier Aucanshala

En la Tabla 19 Se muestra la tabla TURNO que almacena la información de los turnos diarios de las cooperativas de transporte interprovincial.

Tabla 19. Diccionario de datos de la tabla TURNO

Tabla	turno	
Campo	Tipo de dato	Descripción
TUR_COD	INT	Código de turno
DES_COD	INT	Código de destino (clave foránea)
TUR_SAL	TIME	Hora de salida
TUR_LLE	TIME	Hora de llegada

Nota. Descripción de campos de la tabla Turno

Elaborado por: Xavier Aucanshala

2.2.3.2 Diccionario de datos BDD Compra Boletos

En la Tabla 20 Se muestra la tabla TERMINALES que almacena la información de las terminales terrestres del Ecuador.

Tabla 20. Diccionario de datos de la tabla TERMINALES

Tabla	terminales	
Campo	Tipo de dato	Descripción
TER_COD	INT	Código de la terminal
TER_NOM	VARCHAR(50)	Nombre de la terminal
TER_DIR	VARCHAR(100)	Dirección de la terminal

TER_CIU	VARCHAR(50)	Nombre de la ciudad
---------	-------------	---------------------

Nota. Descripción de campos de la tabla Terminales

Elaborado por: Xavier Aucanshala

En la Tabla 21 Se muestra la tabla COOPERATIVAS que almacena la información de las cooperativas de transporte interprovincial.

Tabla 21. Diccionario de datos de la tabla COOPERATIVAS

Tabla	cooperativas	
Campo	Tipo de dato	Descripción
COO_COD	INT	Código de la cooperativa
TER_COD	INT	Código de la terminal (clave foránea)
COO_NOM	VARCHAR(60)	Nombre de la cooperativa
COO_TEL	TEXT	Teléfono de la cooperativa

Nota. Descripción de campos de la tabla Cooperativas

Elaborado por: Xavier Aucanshala

En la Tabla 22 Se muestra la tabla DESTINO que almacena la información de las ciudades destino que tendrá cada cooperativa.

Tabla 22. Diccionario de datos de la tabla DESTINO

Tabla	destino	
Campo	Tipo de dato	Descripción
DES_COD	INT	Código de la ciudad destino
COO_COD	INT	Código de la cooperativa (clave foránea)
DES_NOM	VARCHAR(50)	Nombre de la ciudad destino
DES_PRE	DECIMAL(10, 2)	Precio pasaje a la ciudad destino
DES_HOR	TIME	Tiempo estimado a la ciudad destino

Nota. Descripción de campos de la tabla Destino

Elaborado por: Xavier Aucanshala

En la Tabla 23 Se muestra la tabla BOLETOS que almacena la información de todos los boletos comprados de una cooperativa.

Tabla 23. Diccionario de datos de la tabla BOLETOS

Tabla	boletos	
Campo	Tipo de dato	Descripción
BOL_COD	SERIAL	Código Serial de los boletos
DES_COD	INT	Código de destino (clave foránea)
BOL_FEC	DATE	Fecha de viaje
BOL_ASI	INT	Numero de asiento
BOL_EST	VARCHAR(5)	Estado del boleto
BUSU_TAR	INT	Número de tarjeta de crédito
BUSU_NAM	VARCHAR(50)	Nombre de usuario de tarjeta de crédito
BUSU_MES	INT	Mes de tarjeta de crédito
BUSU_ANIO	INT	Año de tarjeta de crédito
BUSU_CVV	INT	Numero de CVV de tarjeta de crédito

Nota. Descripción de campos de la tabla Boletos

Elaborado por: Xavier Aucanshala

2.2.4. Diagrama de clases

Los diagramas de clases tiene como propósito el de representar los objetos fundamentales del sistema, pueden ser asociativas, de herencia, de uso y de agregación, ya que describe un conjunto de objetos con los mismos atributos, operaciones, métodos, relaciones y semántica, mostrando un conjunto de elementos estáticos.

2.2.4.1. Diagrama de clases para la Aplicación Móvil

En la Figura 7 se visualiza el diagrama de clase de la aplicación móvil con sus respectivas relaciones, métodos, clases serán invocados por la aplicación móvil y la base de datos remota MySQL.

Diagrama de clases de Aplicación Móvil de Cooperativas

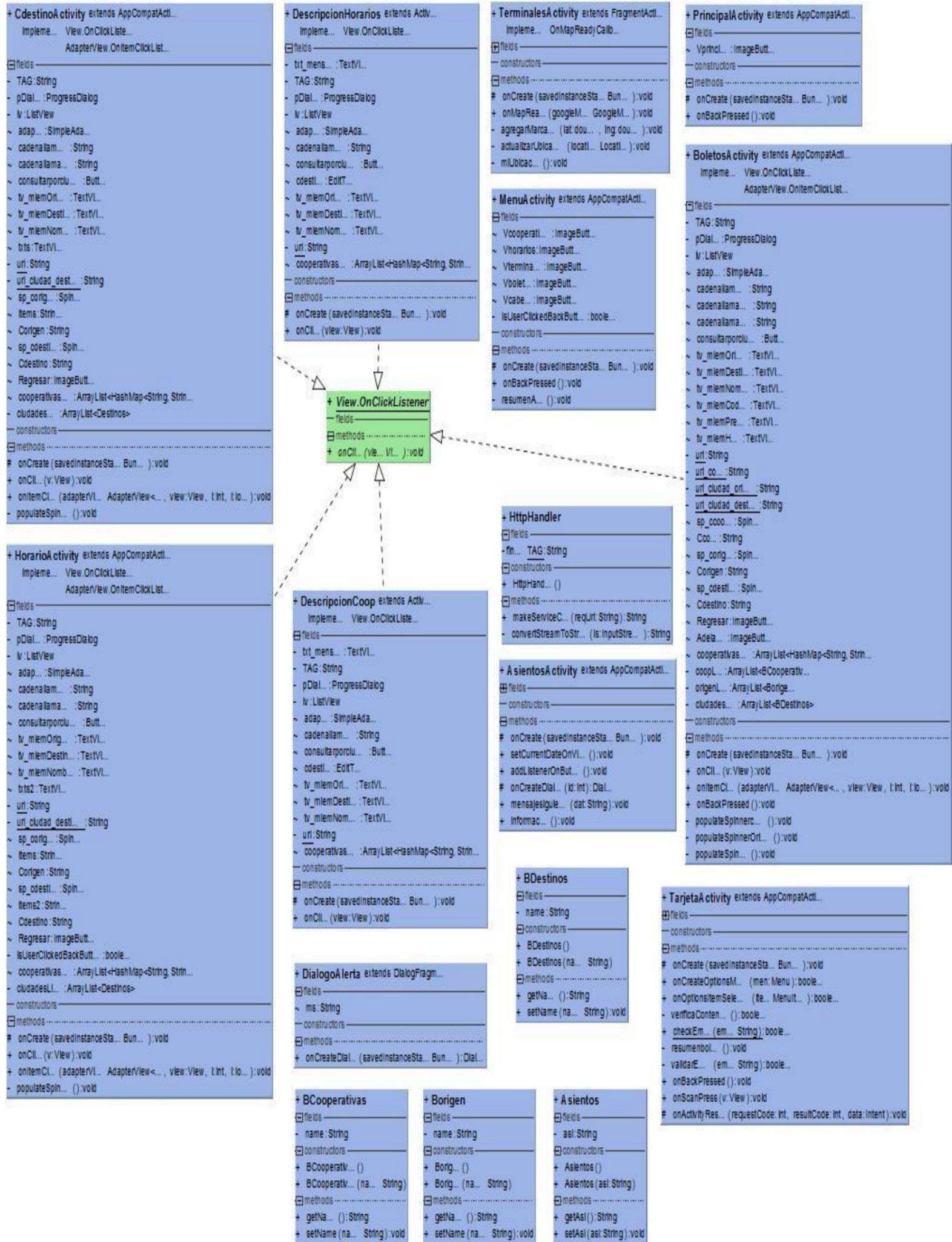


Figura 7. Diagrama de clases de la Aplicación Móvil de Cooperativas
Elaborado por: Xavier Aucanshala

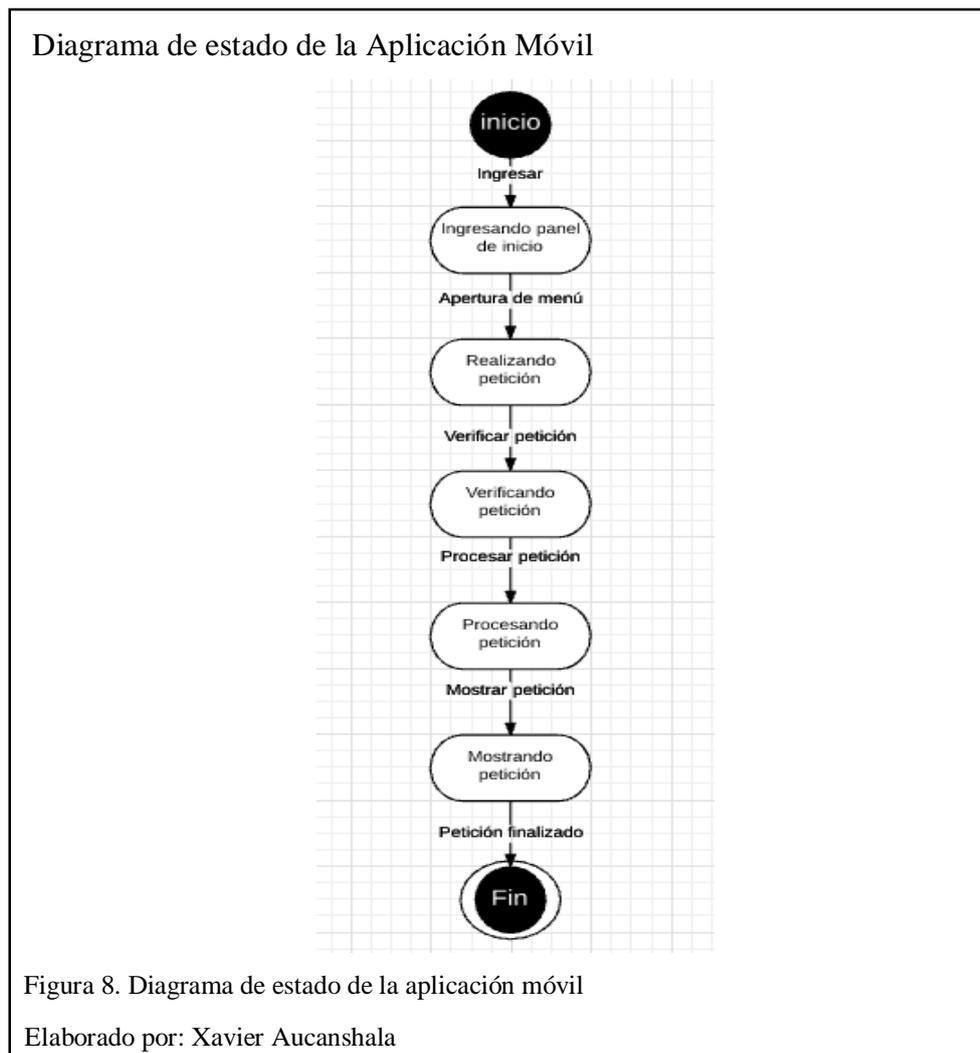
En la figura No. 7 muestra el diagrama de clases de la aplicación móvil ECUA Buses que describe la estructura del sistema, sus relaciones entre los objetos, sus atributos, operaciones (o métodos) y las clases del sistema.

2.2.5. Diagrama de estados UML.

Los diagramas de estado son utilizados para determinar cada una de las rutas o caminos que puede tomar un movimiento de información luego de ejecutarse cada proceso de la aplicación indicando que eventos hacen que se pase de un estado a otro, y las acciones que genera. De esta manera se ha desarrollado el diagrama de estado de la estructura de la aplicación móvil.

2.2.5.1. Diagrama de estados de la Aplicación Móvil.

Este diagrama de estados representa el comportamiento del sistema, las acciones o respuestas posibles que el usuario general puede realizar o pedir en la aplicación móvil junto a las peticiones de la Base de Datos Cooperativas Ecuador y Compra Boletos.



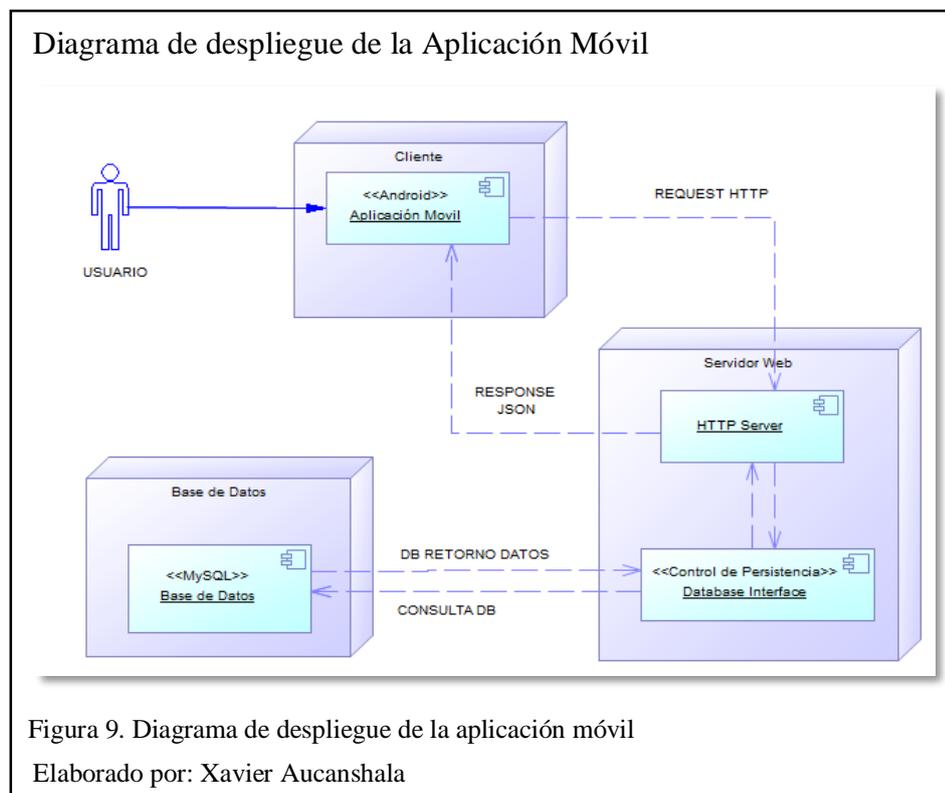
En la figura 8 se puede observar el estado de la información ejecutándose a través de todo el sistema cuando el usuario realiza acciones y peticiones en la aplicación móvil.

2.2.6. Diagrama de despliegue

Los diagramas de despliegue se encargan de modelar la arquitectura de un sistema en cuanto al tiempo de ejecución, mostrando la configuración de todos los elementos de hardware de manera que muestra como los artefactos y elementos del software se trazan en el hardware. De esta manera se ha elaborado el diagrama de despliegue de la estructura de la aplicación móvil.

2.2.6.1. Diagrama de despliegue de la Aplicación Móvil.

Este diagrama de despliegue representa el comportamiento del sistema, en este caso el servidor web en conjunto con el cliente y la base de datos, Los diagramas de despliegue muestran la configuración en funcionamiento del sistema incluyendo su software y su hardware en la aplicación móvil.



En la figura 9 se describe la topología del sistema, la estructura de los elementos de hardware y software que se ejecutan cada uno de ellos y son conectados por enlaces de red, TCP/IP.

2.2.7. Prototipos de interfaces gráficas.

En esta sección se procede a describir todos los prototipos en cuanto a las interfaces que tiene la aplicación móvil, dando una perspectiva real de las interfaces y sus funcionalidades donde el usuario general accede libremente. Las interfaces están diseñadas considerando en lo posible que sean lo más amigables para el usuario final.

2.2.7.1. Interfaz de Usuario para la Aplicación Móvil.

A continuación todas las interfaces de la aplicación móvil van a dar una visión más específica de cómo funciona cada módulo de la aplicación móvil.

2.2.7.1.1. Pantalla Principal de la Aplicación Móvil.



En la Figura 10 se muestra la interfaz principal de la aplicación que contiene en la parte de arriba una imagen en este caso el de la Universidad Politécnica Salesiana, a

continuación se muestra el logo de la aplicación Ecuador, Transporte de Buses (ECUA Buses) que viene a ser el botón de inicio y entrada al menú principal de la aplicación y finalmente se hace una mención a la marca del desarrollador de software Xotta studio.

2.2.7.1.2. Pantalla Menú principal de la Aplicación Móvil.



En la Figura 11 se puede observar la interfaz del menú principal de la aplicación que contiene en la parte de arriba el nombre de la aplicación ECUA Buses, además incluye el menú de cuatro botones con imagen que direccionan a sus respectivos módulos que posee la aplicación móvil: búsqueda de cooperativas, búsqueda de horarios, búsqueda de terminales terrestres y compra de boletos, cabe mencionar que este menú en posteriores versiones puede añadir más botones.

2.2.7.1.3. Pantalla Mensaje del Autor y desarrollador de la aplicación móvil.



En la Figura 12 se puede observar el mensaje acerca del autor y desarrollador de la aplicación, para poder acceder al mensaje basta con presionar la imagen del logotipo de la aplicación y en seguida se despliega dicha información y un botón de aceptar para cerrar la ventana de dialogo sobre nosotros.

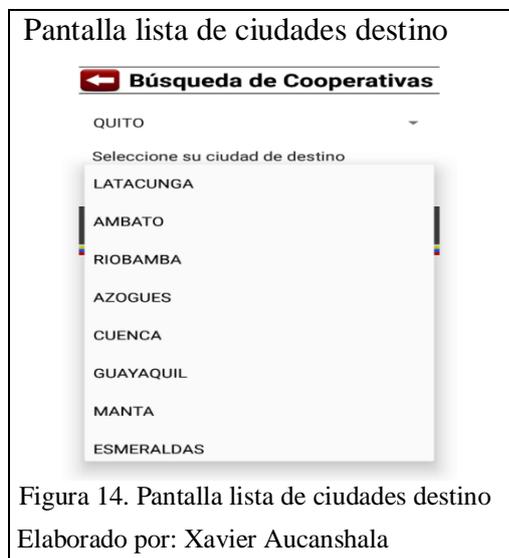
2.2.7.1.4. Pantalla Módulo Cooperativas.



En la Figura 13 al presionar en el botón Cooperativas se puede observar la interfaz de búsqueda de cooperativas que contiene un botón de regreso a menú, un spinner donde

se selecciona una ciudad de origen, un spinner que contiene las ciudades destino, un botón consultar que desplegara todas las cooperativas que van hacia dicho destino.

2.2.7.1.5. Pantalla lista de ciudades destino



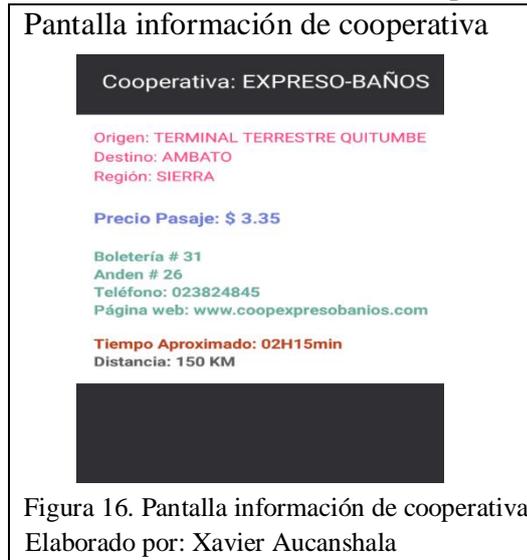
En la Figura 14 al seleccionar una ciudad origen se despliega una lista de todas las ciudades hacia donde las cooperativas tienen como destino posible o están almacenados en la BDD.

2.2.7.1.6. Pantalla lista de Cooperativas hacia un destino.



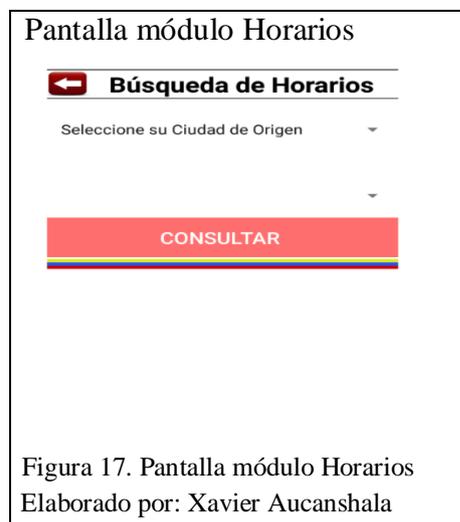
En la Figura 15 se puede observar una lista de todas las cooperativas que van hacia esa ciudad destino junto con la ciudad origen y región a la que pertenece la ciudad destino.

2.2.7.1.7. Pantalla información detallada de una cooperativa.



En la Figura 16 se puede observar la información completa en cuanto se refiere a una cooperativa seleccionada como por ejemplo: nombre de cooperativa, origen, destino, región, precio de pasaje, numero de boletería, numero de anden, teléfono, pagina web, tiempo aproximado de viaje y la distancia a recorrer.

2.2.7.1.8. Pantalla Módulo Horarios.



En la Figura 17 al presionar en el botón Horarios se puede observar la interfaz de búsqueda de cooperativas que contiene un botón de regreso a menú, un spinner donde se selecciona una ciudad de origen, un spinner que contiene las ciudades destino, un botón consultar que desplegara todas las cooperativas que van hacia dicho destino.

2.2.7.1.9. Pantalla lista de cooperativas.



Figura 18. Pantalla lista de cooperativas

En la Figura 18 se puede observar una lista de todas las cooperativas que van hacia esa ciudad destino junto con la ciudad origen y destino.

2.2.7.1.10. Pantalla lista de horarios de una cooperativa

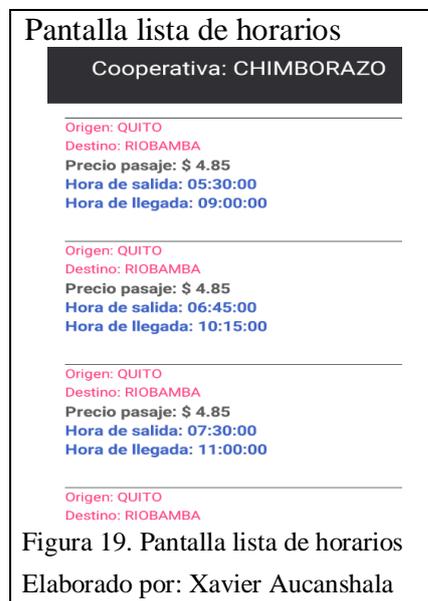


Figura 19. Pantalla lista de horarios

Elaborado por: Xavier Aucanshala

En la Figura 19 se puede observar los horarios detalladamente en cuanto se refiere a una cooperativa seleccionada como por ejemplo: nombre de cooperativa, origen, destino, precio de pasaje, hora de salida y hora de llegada hacia un destino.

2.2.7.1.11. Pantalla Módulo Terminales

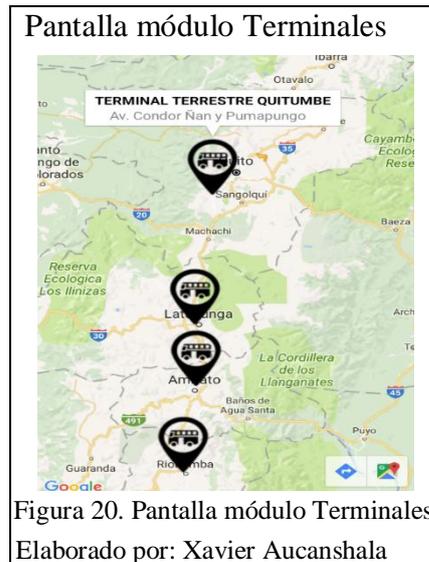


Figura 20. Pantalla módulo Terminales
Elaborado por: Xavier Aucanshala

En la Figura 20 se puede observar el mapa de ubicación de todas las terminales terrestres guardadas en la BDD, al presionar sobre un ícono se muestra un mensaje de información de del nombre de la terminal terrestre y su dirección física.

2.2.7.1.12. Pantalla Módulo Boletos.

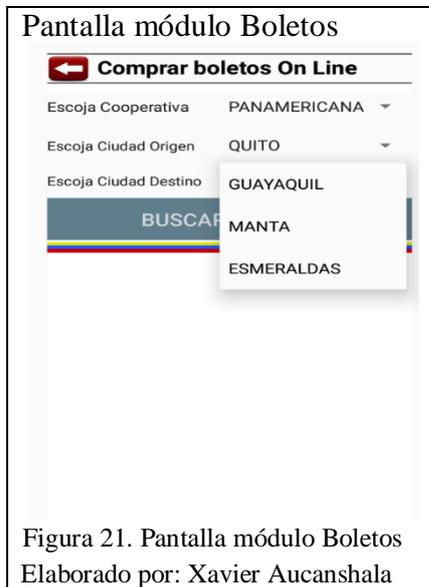


Figura 21. Pantalla módulo Boletos
Elaborado por: Xavier Aucanshala

En la Figura 21 al presionar en el botón Boletos se puede observar la interfaz de comprar de boletos para una cooperativa que contiene un botón de regreso a menú, un spinner donde se selecciona la cooperativa, un spinner para una ciudad de origen, un spinner que contiene las ciudades destino, un botón consultar que despliega todos los turnos disponibles para una cooperativa seleccionada que va hacia dicho destino.

2.2.7.1.13. Pantalla lista de turnos.

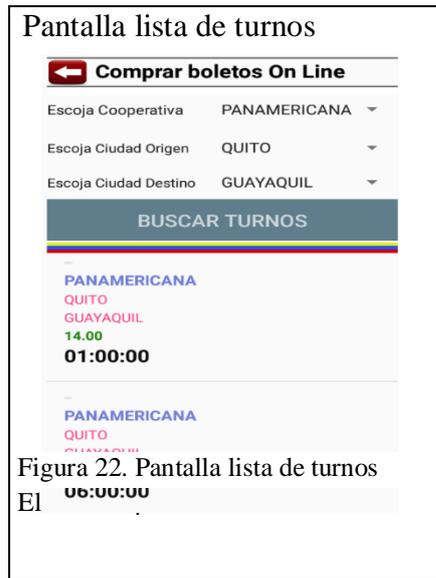


Figura 22. Pantalla lista de turnos
El

En la Figura 22 se puede observar una lista de todos los turnos y precios de pasaje para la cooperativa seleccionada junto con la ciudad origen y destino.

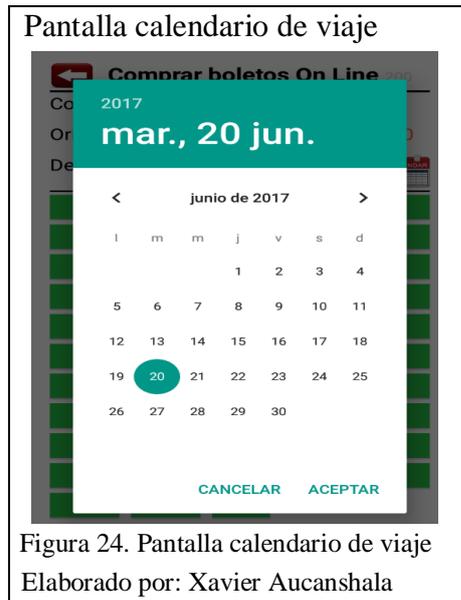
2.2.7.1.14. Pantalla mensaje antes de comprar un asiento.



Figura 23. Pantalla mensaje antes de compra
Elaborado por: Xavier Aucanshala

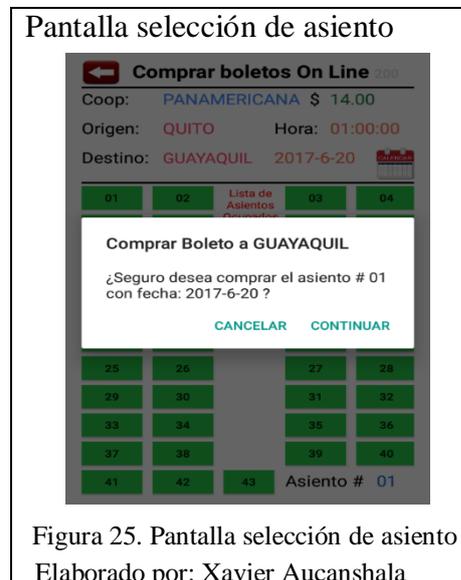
En la Figura 23 se puede observar el mensaje de advertencia antes de realizar la compra de un asiento, en dicho mensaje aclara que primero se debe seleccionar una fecha de viaje, luego solo se puede seleccionar un asiento y finalmente un botón aceptar para cerrar el mensaje de dialogo.

2.2.7.1.15. Pantalla calendario de viaje



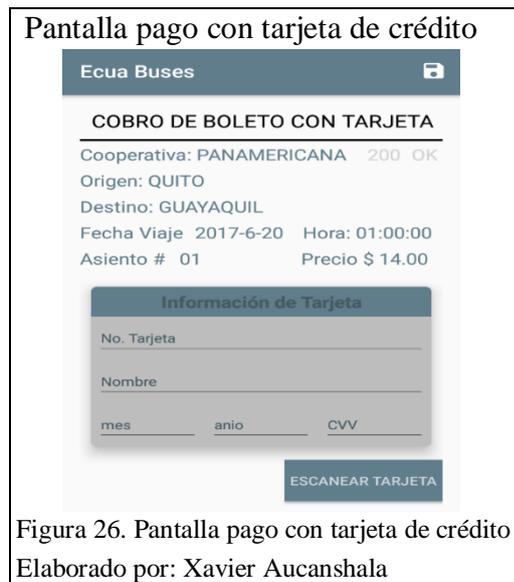
En la Figura 24 se puede observar un calendario donde el usuario solo puede seleccionar una fecha de viaje en el día actual o posterior, pero no una fecha anterior y dos botones para cancelar y aceptar la fecha de viaje.

2.2.7.1.16. Pantalla selección de asiento.



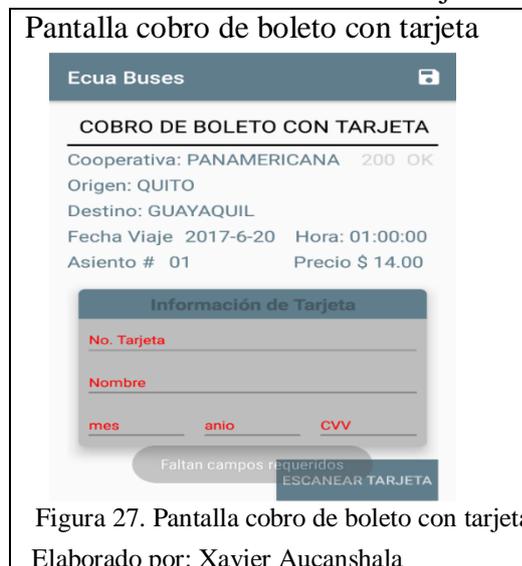
En la Figura 25 se puede observar el mensaje de confirmación para una fecha y un asiento seleccionado para un total de 43 asientos disponibles dando la alternativa de aceptar o cancelar la compra del asiento. Además al fondo se observa todos los datos previamente seleccionados por el usuario.

2.2.7.1.17. Pantalla pago con tarjeta de crédito



En la Figura 26 se puede observar toda la información del boleto a comprarse, además de un cuadro para la información de la tarjeta de crédito, un botón de escanear tarjeta que al presionar abre la cámara del celular para la toma de datos y finalmente un botón de guardar que se encuentra en la parte superior de la pantalla.

2.2.7.1.18. Pantalla Módulo cobro de boleto con tarjeta



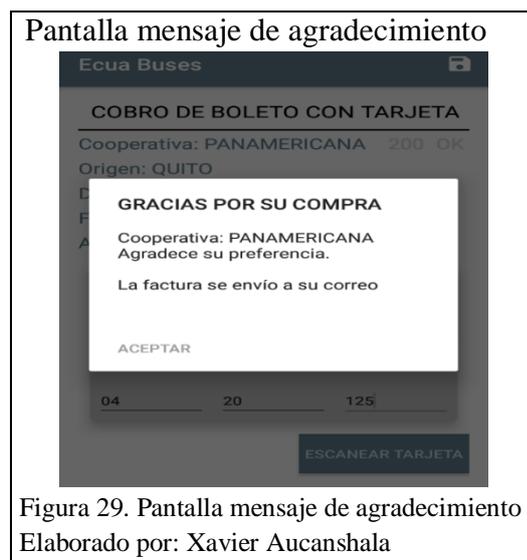
En la Figura 27 si se presiona el botón guardar antes de la toma de datos de la tarjeta se puede observar el mensaje que faltan campos requeridos y letras en rojo sobre cada caja de texto, de manera que para guardar la información de la compra se debe llenar todos los datos de la tarjeta.

2.2.7.1.19. Pantalla correo para facturar



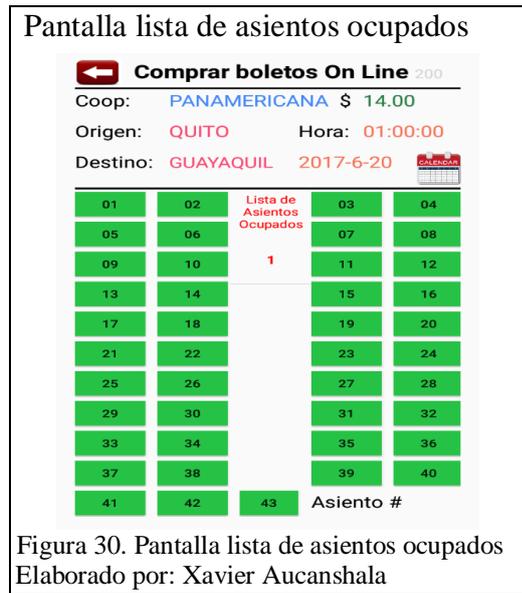
En la Figura 28 se puede observar una ventana con toda la información requerida para la factura que se envía a un correo electrónico personal del usuario, es necesario un correo, un asunto y el cuerpo del correo, para lo cual el correo debe estar debidamente validado y un botón de enviar datos hacia el correo ingresado.

2.2.7.1.20. Pantalla mensaje de agradecimiento



En la Figura 29 se puede observar el mensaje final de agradecimiento por la compra en la cooperativa, anunciando que la factura se envió a su correo y un botón aceptar que cerrara el mensaje de dialogo y este regresa a la interfaz de menú principal.

2.2.7.1.21. Pantalla lista de asientos ocupados



En la Figura 30 se puede observar con letras rojas en medio de los asientos una lista de asientos ocupados, este se va cargando de acuerdo a los asientos que se compran para un turno y fecha de viaje específicos.

Capítulo 3

CODIFICACIÓN Y PRUEBAS

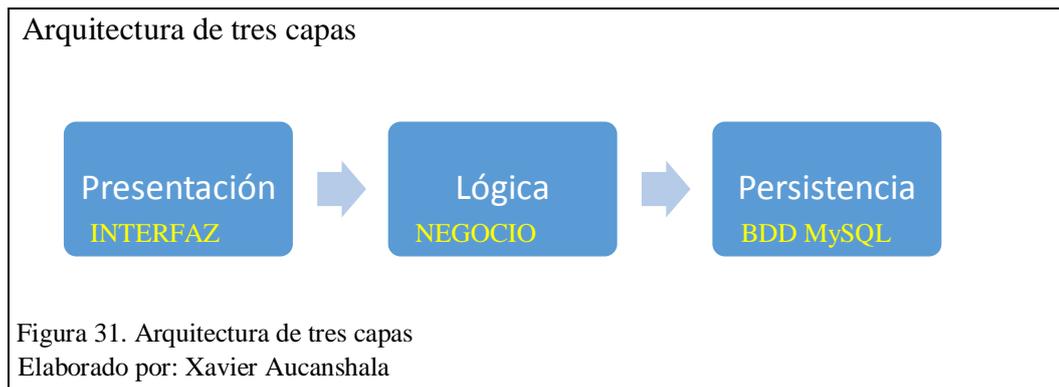
En este capítulo se muestra paso a paso todo el proceso que se lleva a cabo para el desarrollo de la aplicación móvil, todas las herramientas tecnológicas que se utilizan previamente configuradas, la codificación general de la aplicación y por último las distintas pruebas a las que se somete la aplicación móvil para validar su funcionalidad para así poder cumplir las expectativas de los usuarios en cuanto a sus requerimientos.

3.1 Arquitectura y tecnologías de la aplicación móvil

Para poder alojar en la nube la Base de datos MySQL y todos los archivos PHP necesarios tanto de conexión, configuración, consulta de la aplicación móvil se hace el uso de un hosting de paga y un dominio personal. En los siguientes subcapítulos se detalla las diferentes herramientas tecnológicas y la arquitectura utilizadas en el desarrollo del aplicativo móvil.

3.1.1. Arquitectura de la Aplicación Móvil.

La aplicación móvil ECUA Buses se basa en una arquitectura de tres capas: presentación (interfaz), lógica de negocios y persistencia, para así lograr una eficiente y correcta división entre el código y la dependencia de capas. Los sistemas en 3 capas son de más fácil ampliación, más robustos y flexibles, por lo que es posible ejecutar las tres capas en la misma máquina.



En la Figura 31 se describe una arquitectura de tres capas para una aplicación móvil: presentación (interfaz), lógica de negocios y persistencia (BDD).

3.1.1.1. Capa de Presentación

La capa de presentación o interfaz dentro de la arquitectura se refiere al mecanismo de interacción del usuario con el sistema. Es lo que puede ver el usuario, le comunica y captura la información del usuario en un mínimo proceso, la capa de presentación únicamente se comunica con la capa de negocio.

3.1.1.2. Capa Lógica

La capa lógica de negocios se encarga absolutamente de toda la lógica del proyecto y funcionamiento del mismo. Esta capa además de proteger el acceso directo a la información desde la capa de presentación, se encarga del manejo y suministro de información en las diferentes interfaces.

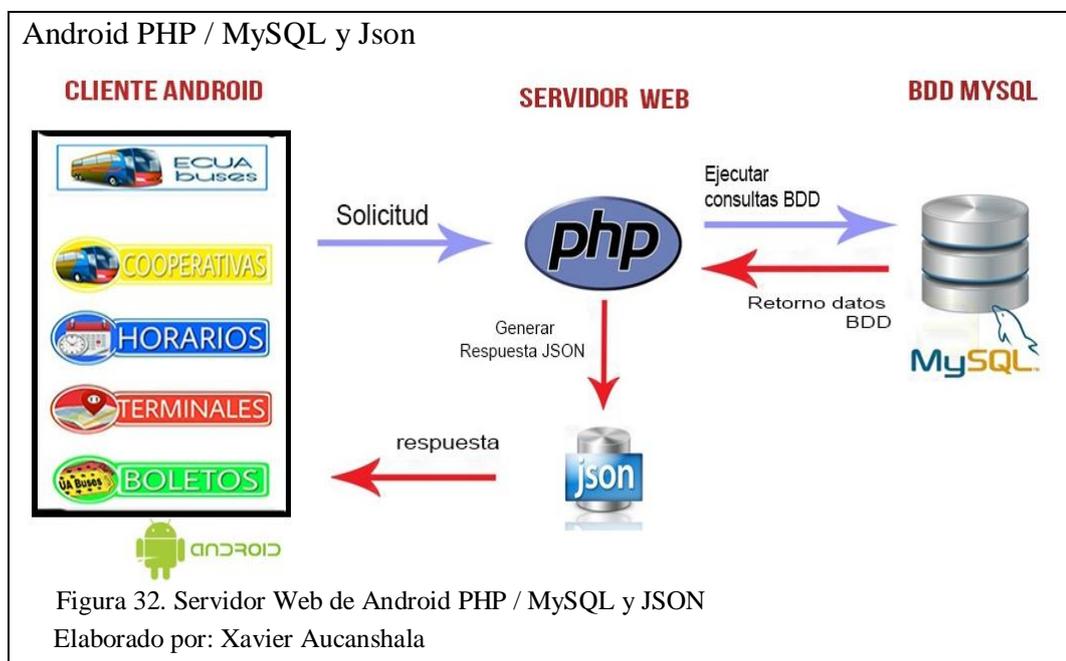
3.1.1.3. Capa de Persistencia

La capa de persistencia es la responsable del almacenamiento de los datos además de encargarse de la transmisión de la información entre la base de datos y la aplicación móvil con la capacidad de realizar inserciones o visualización de la información almacenada.

En Android no existe una forma nativa para acceder a bases de datos remotas.

En la aplicación móvil, para conectar entre la capa lógica y la capa de

persistencia, se ha optado por utilizar un puente que utiliza las tecnologías PHP y JSON para la comunicación. Partiendo de una base de datos MySQL en el servidor remoto, para enviar datos de una tabla a la aplicación, se tendrá dos secciones: la parte ejecutada en el servidor remoto y la parte Android. La parte del servidor remoto consiste en un fichero PHP que se conecta a la base de datos, realiza una consulta y devuelve los resultados codificados en formato JSON (la codificación a este formato permite obtener y enviar los datos de forma fluida y rápida). En la parte Android se utiliza un método HttpPost para obtener los datos, se transforma la respuesta a String JSON, y finalmente se transforman los datos JSON en el tipo de datos necesario (String, Int, etc). A continuación, en las siguientes imágenes, se puede observar el puente PHP/JSON utilizado y el esquema entidad-relación de la base de datos remota (Sáez, 2011).



En la Figura 32 se describe la estructura de un servidor web entre Android, PHP, MySQL y JSON el cual se encuentra entre la aplicación móvil y el servidor.

3.1.2. Entorno de desarrollo Android Studio IDE

“Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones para Android y se basa en IntelliJ IDEA” (developer.android.com, 2017), diseñado para ofrecer nuevas herramientas para el desarrollo de aplicaciones y alternativa al entorno Eclipse, hasta ahora el IDE más utilizado y trabaja través de una licencia Apache 2.0, es multiplataforma y programado en Java.

Esta IDE fue anunciada por Google el 16 de mayo del 2013, con el objetivo de crear un entorno exclusivo para la programación de aplicaciones móviles Android.

Cada proyecto en Android Studio contiene uno o más módulos con archivos de código fuente, archivos de recursos y cada módulo de la aplicación contiene las siguientes carpetas:

- **manifests:** contiene el archivo AndroidManifest.xml.
- **java:** contiene los archivos de código fuente de Java, incluido el código de prueba JUnit.
- **res:** Contiene todos los recursos, como diseños XML, cadenas de IU e imágenes de mapa de bits.

3.1.3. Almacén de Claves de Android (KeyStore)

El sistema Android Keystore es un repositorio de certificados de seguridad y permite almacenar claves criptográficas en un contenedor para que resulte más difícil extraerlo del dispositivo. Una vez que las claves se encuentran en el almacén de claves, se pueden usar para realizar operaciones criptográficas con el material de claves restante no exportable.

El sistema Android Keystore protege el material de claves contra usos no autorizados.

En las claves privadas solo una aplicación firmada puede mostrar todas sus funcionalidades en modo de liberación ya que al contrario una aplicación móvil que no ha sido firmada solo muestra estas funcionalidades en modo de depuración (Developers Android).

3.2. Codificación

En esta parte del capítulo, se va a mostrar todo lo relacionado a la programación y la organización de las clases que contiene la aplicación móvil.

El desarrollo de los módulos de la aplicación móvil se lo realizó en el siguiente orden:

1. Búsqueda de cooperativas
2. Búsqueda de horarios
3. Búsqueda de terminales terrestres en un mapa
4. Compra de boletos en línea

3.2.1. Clases y métodos de la Aplicación Móvil

Basados en la arquitectura de tres capas utilizados para el desarrollo de la aplicación móvil. Se procede a documentar todas las clases de la aplicación, es documentada respecto a la base de datos remota y todas las vistas necesarias para la comunicación entre usuario y aplicación móvil.

3.2.1.1 Conexión BDD MySQL y PHP

Para realizar la conexión entre la BDD MySQL y PHP se utilizan dos archivos Database.php que es una clase que envuelve una instancia de la clase PDO para el manejo de la base de datos y el otro archivo mysql_login.php que se encarga de proveer las constantes para conectarse a la base de datos Mysql tales como Hostname, Database, Username y Password.

3.2.1.2. Clase Cooperativas

La clase Cooperativas.php representa la estructura de las Cooperativas almacenadas en la base de datos y está conectada directamente al archivo Database.php y contiene todas las funciones necesarias para la aplicación móvil como por ejemplo: getAll(), getByCDestino, getByInfoCoop, getByCHorario, getByInfoHorarios, getByCiudadDestino y dentro de estas funciones se realiza la consulta SQL a través de parámetros hacia la BDD MySQL.

3.2.1.3. Módulo búsqueda de cooperativas

3.2.1.3.1. Búsqueda de cooperativas por ciudad destino

Para realizar la búsqueda de cooperativas por ciudad destino, se utiliza un archivo obtener_cooperativa_por_destino.php el cual a través de Cooperativas.php realiza las consultas SQL a la BDD devolviendo los registros en formato String JSON. Para luego mostrar una lista de todas las cooperativas que van hacia dicha ciudad destino desde una ciudad origen y que realiza las peticiones GET mediante la clase HttpHandler.java. La interacción entre servidor y usuario es realizada con la interfaz activity_cdestino.xml

Capa presentación o interfaz

Tabla 24. Archivo activity_cdestino.xml

Archivo	Módulo	Descripción
activity_cdestino.xml	Búsqueda de cooperativas	Pantalla para la búsqueda de cooperativas mediante origen y ciudad destino

Nota. Descripción de la interfaz Búsqueda de cooperativas

Elaborado por: Xavier Aucanshala

Capa lógica de negocios

Tabla 25. Cooperativas por ciudad destino

Archivo	Método	Descripción
---------	--------	-------------

CdestinoActivity.java	onCreate ()	Método invocado cuando inicia la actividad.
	onPreExecute()	Método asociado al dialogo de progreso y otras funcionalidades iniciales de la actividad.
	GetCooperativas	Es una clase multitarea, donde se implementa los métodos de inicio y funcionalidad de cada actividad.
	doInBackground ()	Es un método para el llamado de la URL del archivo obtener_cooperativa_por_destino.php con sus respectivos parámetros.
	GetCiudades	Es una clase multitarea, donde se implementa los métodos de inicio y funcionalidad de cada actividad.
	doInBackground ()	Es un método para el llamado de la URL del archivo obtener_ciudad_destino.php con sus respectivos parámetros.
	onPostExecute ()	Este método contiene toda la estructura de la lista informativa para ver todas las cooperativas para la ciudad destino seleccionado.

Nota. Descripción de la lógica Búsqueda de cooperativas

Elaborado por: Xavier Aucanshala

Capa de persistencia

Tabla 26. Consulta de cooperativas por ciudad destino

Archivo	Método	Descripción
obtener_cooperativa_por_destino.php	getByCDestino ()	Consulta a la base de datos el listado las cooperativas por ciudad destino seleccionado
	getByCiudadDestino ()	Consulta a la base de datos el listado de las ciudades destino desde una ciudad origen seleccionado

Nota. Descripción de la persistencia de búsqueda de cooperativas

Elaborado por: Xavier Aucanshala

3.2.1.3.2. Información detallada de una cooperativa

Para mostrar la información detallada de una cooperativa se utiliza un archivo obtener_informacion_por_coop.php el cual a través de Cooperativas.php realiza las consultas SQL a la BDD devolviendo los registros en formato String JSON. Para luego mostrar toda la información detallada de una cooperativa seleccionada y que realiza las peticiones GET mediante la clase HttpHandler.java. La interacción entre servidor y usuario es realizada con la interfaz activity_descripcioncoop.xml

Capa presentación o interfaz

Tabla 27. Archivo activity_descripcioncoop.xml

Archivo	Módulo	Descripción
activity_descripcioncoop.xml	Búsqueda de cooperativas	Pantalla para la información detallada de una cooperativa

Nota. Descripción de la interfaz información detallada de una cooperativa

Elaborado por: Xavier Aucanshala

Capa lógica de negocios

Tabla 28. Información detallada de una cooperativa

Archivo	Método	Descripción
DescripcionCoop.java	onCreate ()	Método invocado cuando inicia la actividad.
	onPreExecute()	Método asociado al dialogo de progreso y otras funcionalidades iniciales de la actividad.
	GetInformacion()	Es una clase multitarea, donde se implementa los métodos de inicio y funcionalidad.
	doInBackground ()	Es un método para el llamado de la URL del archivo obtener_informacion_por_coop.php con sus respectivos parámetros.
	onPostExecute ()	Este método contiene toda la estructura de la lista informativa para ver la información detallada de una cooperativa seleccionada.

Nota. Descripción de la lógica información detallada de una cooperativa

Elaborado por: Xavier Aucanshala

Capa de persistencia

Tabla 29. Consulta de información detallada de una cooperativa

Archivo	Método	Descripción
obtener_informacion_por_coop.php	getByInfoCoop ()	Consulta a la base de datos el listado la información detallada de una cooperativa seleccionada

Nota. Descripción de la persistencia de la información detallada de una cooperativa

Elaborado por: Xavier Aucanshala

3.2.1.4. Módulo búsqueda de horarios

3.2.1.4.1. Búsqueda de cooperativas por ciudad destino

Para realizar la búsqueda de cooperativas por ciudad destino, se utiliza un archivo obtener_cooperativa_por_destino_horario.php el cual a través de Cooperativas.php

realiza las consultas SQL a la BDD devolviendo los registros en formato String JSON. Para luego mostrar una lista de todas las cooperativas que van hacia dicha ciudad destino desde una ciudad origen y que realiza las peticiones GET mediante la clase `HttpHandler.java`. La interacción entre servidor y usuario es realizada con la interfaz `activity_horario.xml`

Capa presentación o interfaz

Tabla 30. Archivo `activity_horario.xml`

Archivo	Módulo	Descripción
<code>activity_horario.xml</code>	Búsqueda de horarios	Pantalla para la búsqueda de cooperativas mediante ciudad origen y ciudad destino

Nota. Descripción de la interfaz Búsqueda de horarios

Elaborado por: Xavier Aucanshala

Capa lógica de negocios

Tabla 31. Cooperativas por ciudad destino horario

Archivo	Método	Descripción
HorarioActivity.java	<code>onCreate ()</code>	Método invocado cuando inicia la actividad.
	<code>onPreExecute()</code>	Método asociado al dialogo de progreso y otras funcionalidades iniciales de la actividad.
	<code>GetCooperativas</code>	Es una clase multitarea, donde se implementa los métodos de inicio y funcionalidad.
	<code>doInBackground ()</code>	Es un método para el llamado de la URL del archivo <code>obtener_cooperativa_por_destino_horario.php</code> con sus respectivos parámetros.
	<code>GetCiudades</code>	Es una clase multitarea, donde se implementa los métodos de inicio y funcionalidad.
	<code>doInBackground ()</code>	Es un método para el llamado de la URL del archivo <code>obtener_ciudad_destino.php</code> con sus respectivos parámetros.
	<code>onPostExecute ()</code>	Este método contiene toda la estructura de la lista informativa para ver todas las cooperativas para la ciudad destino seleccionado.

Nota. Descripción de la lógica Búsqueda de cooperativas en horarios

Elaborado por: Xavier Aucanshala

Capa de persistencia

Tabla 32. Consulta de cooperativas por ciudad destino horario

Archivo	Método	Descripción
obtener_cooperativa_por_destino_horario.php	getByCHorario ()	Consulta a la base de datos el listado las cooperativas por ciudad destino seleccionado
	getByCiudadDestino ()	Consulta a la base de datos el listado de las ciudades destino desde una ciudad origen seleccionado

Nota. Descripción de la persistencia de búsqueda de cooperativas en horarios

Elaborado por: Xavier Aucanshala

3.2.1.4.2. Horarios y precios de una cooperativa

Para mostrar los horarios y precios de una cooperativa se utiliza un archivo obtener_informacion_horarios.php el cual a través de Cooperativas.php realiza las consultas SQL a la BDD devolviendo los registros en formato String JSON. Se realiza las peticiones GET mediante la clase HttpHandler.java. La interacción entre servidor y usuario es realizada con la interfaz activity_descripcionhorarios.xml

Capa presentación o interfaz

Tabla 33. Archivo activity_descripcionhorarios.xml

Archivo	Módulo	Descripción
activity_descripcionhorarios.xml	Búsqueda de horarios	Pantalla para los horarios y precios de una cooperativa

Nota. Descripción de la interfaz horarios y precios de una cooperativa

Elaborado por: Xavier Aucanshala

Capa lógica de negocios

Tabla 34. Horarios y precios de una cooperativa

Archivo	Método	Descripción
	onCreate ()	Método invocado cuando inicia la actividad.
	onPreExecute()	Método asociado al dialogo de progreso y otras funcionalidades iniciales de la actividad.
	GetHorarios ()	Es una clase multitarea, donde se implementa los métodos de inicio y funcionalidad.

DescripcionHorarios.java	doInBackground ()	Es un método para el llamado de la URL del archivo obtener_informacion_horarios.php con sus respectivos parámetros.
	onPostExecute ()	Este método contiene toda la estructura de la lista informativa para ver los horarios y precios de una cooperativa seleccionada.

Nota. Descripción de la lógica Horarios y precios de una cooperativa

Elaborado por: Xavier Aucanshala

Capa de persistencia

Tabla 35. Consulta de horarios y precios de una cooperativa

Archivo	Método	Descripción
obtener_informacion_horarios.php	getByInfoHorarios ()	Consulta a la base de datos el listado los horarios y precios de una cooperativa seleccionada

Nota. Descripción de la persistencia de Horarios y precios de una cooperativa

Elaborado por: Xavier Aucanshala

3.2.1.5. Módulo búsqueda de Terminales terrestres

Para realizar la búsqueda de terminales terrestres, lo primero es obtener una clave de Google Maps API, crear un nuevo proyecto, crear una nueva credencial, seleccionar clave de API, luego clave de Android, copiar el nombre del paquete desde el proyecto, y así mismo pegar la huella digital de certificado SHA-1 desde el proyecto que se encuentra en el archivo google_maps_api.xml, para finalmente obtener una clave guarda en la línea de código “google maps key”, sin antes habilitar las APIS importantes de un mapa en Android y en Android SDK Manager instalar el Google Play Services. Con esto ya podemos correr el mapa de google en la aplicación móvil.

Capa presentación o interfaz

Tabla 36. Archivo activity_terminales.xml

Archivo	Módulo	Descripción
activity_terminales.xml	Búsqueda de terminales terrestres	Pantalla para la búsqueda de terminales terrestres dentro de un mapa

Nota. Descripción de la interfaz Búsqueda de terminales terrestres

Elaborado por: Xavier Aucanshala

Capa lógica de negocios

Tabla 37. Marcadores de terminales en el mapa

Archivo	Método	Descripción
TerminalesActivity.java	onCreate ()	Método invocado cuando inicia la actividad.
	onMapReady()	Método donde se encuentran los marcadores de latitud y longitud, posición, título e icono de cada terminal.
	agregarMarcador ()	Método que inserta marcadores de latitud y longitud de posiciona actual del dispositivo móvil

Nota. Descripción de la lógica Búsqueda de terminales terrestres

Elaborado por: Xavier Aucanshala

3.2.1.6. Módulo Boletos

3.2.1.6.1. Búsqueda de turnos

Para realizar la búsqueda de turnos de cooperativas por ciudad origen y destino, se utiliza un archivo obtener_cooperativa_por_destino.php.php el cual a través de Cooperativas.php realiza las consultas SQL a la BDD devolviendo los registros en formato String JSON. Para luego mostrar una lista de todas las cooperativas que van hacia dicha ciudad destino desde una ciudad origen y que realiza las peticiones GET mediante la clase HttpHandler.java. La interacción entre servidor y usuario es realizada con la interfaz activity_boletos.xml

Capa presentación o interfaz

Tabla 38. Archivo activity_boletos.xml

Archivo	Módulo	Descripción
activity_boletos.xml	Búsqueda de boletos	Pantalla para la búsqueda de turnos mediante cooperativa, ciudad origen y ciudad destino

Nota. Descripción de la interfaz Búsqueda de boletos

Elaborado por: Xavier Aucanshala

Capa lógica de negocios

Tabla 39. Turnos para una cooperativa por ciudad destino

Archivo	Método	Descripción
BoletosActivity.java	onCreate ()	Método invocado cuando inicia la actividad.
	onPreExecute()	Método asociado al dialogo de progreso y otras funcionalidades iniciales de la actividad.
	GetCooperativas	Es una clase multitarea, donde se implementa los métodos de inicio y funcionalidad.
	doInBackground ()	Es un método para el llamado de la URL del archivo obtener_cooperativas.php con sus parámetros obligatorios.
	GetOrigen	Es una clase multitarea, donde se implementa los métodos de inicio y funcionalidad.
	doInBackground ()	Es un método para el llamado de la URL del archivo obtener_ciudad_origen.php con sus respectivos parámetros.
	GetDestino	Es una clase multitarea, donde se implementa los métodos de inicio y funcionalidad.
	doInBackground ()	Es un método para el llamado de la URL del archivo obtener_ciudad_destino.php con sus respectivos parámetros.
	GetTurnos	Es una clase multitarea, donde se implementa los métodos de inicio y funcionalidad.
	doInBackground ()	Es un método para el llamado de la URL del archivo obtener_cooperativa_por_destino.php con sus respectivos parámetros.
	onPostExecute ()	Este método contiene toda la estructura de la lista informativa para ver todos los turnos disponibles de la cooperativa para la ciudad destino seleccionado.

Nota. Descripción de la lógica turnos para una cooperativa por ciudad destino

Elaborado por: Xavier Aucanshala

Capa de persistencia

Tabla 40. Consulta de turnos para una cooperativa por ciudad destino

Archivo	Método	Descripción
obtener_cooperativas.php	getAll ()	Consulta a la base de datos el listado las cooperativas
obtener_ciudad_origen.php	getByCiudadOrigen ()	Consulta a la base de datos el listado las ciudades origen por cooperativa seleccionado
obtener_ciudad_destino.php	getByCiudadDestino ()	Consulta a la base de datos el listado de las ciudades destino desde una ciudad origen seleccionado

obtener_cooperativa_por_desti no.php	getByOrigenDestino ()	Consulta a la base de datos el listado de turnos de una cooperativa desde una ciudad origen y destino seleccionado
---	------------------------	--

Nota. Descripción de la persistencia de búsqueda de turnos para una cooperativa por ciudad destino
Elaborado por: Xavier Aucanshala

3.2.1.6.2. Selección de fecha y asiento de viaje

Para obtener la fecha y asiento de una cooperativa se utiliza un archivo obtener_asientos.php el cual a través de Cooperativas.php realiza las consultas SQL a la BDD devolviendo los registros en formato String JSON. Para luego mostrar toda la información detallada de una cooperativa seleccionada y que realiza las peticiones GET mediante la clase HttpHandler.java. La interacción entre servidor y usuario es realizada con la interfaz activity_asientos.xml

Capa presentación o interfaz

Tabla 41. Archivo activity_asientos.xml

Archivo	Módulo	Descripción
activity_asientos.xml	Búsqueda de boletos	Pantalla para escoger fecha y asiento de viaje de una cooperativa

Nota. Descripción de la interfaz fecha y asiento de una cooperativa
Elaborado por: Xavier Aucanshala

Capa lógica de negocios

Tabla 42. Fecha y asiento de una cooperativa

Archivo	Método	Descripción
AsientosActivity.java	onCreate ()	Método invocado cuando inicia la actividad.
	onPreExecute()	Método asociado al dialogo de progreso y otras funcionalidades iniciales de la actividad.
	GetAsientos ()	Es una clase multitarea, donde se implementa los métodos de inicio y funcionalidad.
	doInBackground ()	Es un método para el llamado de la URL del archivo obtener_asientos.php con sus respectivos parámetros.
	onPostExecute ()	Este método contiene toda la estructura de la lista informativa para ver los asientos ocupados y disponibles de una fecha seleccionada.

Nota. Descripción de la lógica fecha y asientos de una cooperativa
 Elaborado por: Xavier Aucanshala

Capa de persistencia

Tabla 43. Consulta de asientos ocupados en una fecha

Archivo	Método	Descripción
obtener_asientos.php	getByAsiento ()	Consulta a la base de datos el listado los asientos ocupados de una fecha seleccionada

Nota. Descripción de la persistencia de asientos ocupados en una fecha seleccionada
 Elaborado por: Xavier Aucanshala

3.2.1.6.3. Pago de boleto con tarjeta de crédito

Para realizar el pago de un boleto con tarjeta de crédito, se utiliza un archivo insertar_boleto.php el cual a través de Cooperativas.php realiza las consultas SQL a la BDD devolviendo los registros en formato String JSON. Para luego mostrar una lista de todas las cooperativas que van hacia dicha ciudad destino desde una ciudad origen y que realiza las peticiones GET mediante la clase HttpHandler.java. La interacción entre servidor y usuario es realizada con la interfaz activity_tarjeta.xml

Capa presentación o interfaz

Tabla 44. Archivo activity_tarjeta.xml

Archivo	Módulo	Descripción
activity_tarjeta.xml	Búsqueda de boletos	Pantalla para el pago de un boleto a través de tarjeta de crédito

Nota. Descripción de la interfaz Búsqueda de boletos
 Elaborado por: Xavier Aucanshala

Capa lógica de negocios

Tabla 45. Pago de un boleto con tarjeta

Archivo	Método	Descripción
	onCreate ()	Método invocado cuando inicia la actividad.
	onPreExecute()	Método asociado al dialogo de progreso y otras funcionalidades iniciales de la actividad.

TarjetaActivity.java	ObtenerWebService	Es una clase multitarea, donde se implementa los métodos de inicio y funcionalidad.
	doInBackground ()	Es un método para el llamado de la URL del archivo insertar_boleto.php con sus respectivos parámetros.
	onScanPress	Método que escanea los datos de una tarjeta de crédito a través de la cámara del dispositivo
	checkEmail ()	Método para validar un correo electrónico
	verificaContenido()	Método que valida datos de la tarjeta

Nota. Descripción de la lógica Pago de un boleto con tarjeta

Elaborado por: Xavier Aucanshala

Capa de persistencia

Tabla 46. Inserción de pago de un boleto con tarjeta

Archivo	Método	Descripción
insertar_boleto.php	insert ()	Inserta a la base de datos la información de compra de un boleto con tarjeta de crédito

Nota. Descripción de la persistencia de pago de un boleto con tarjeta

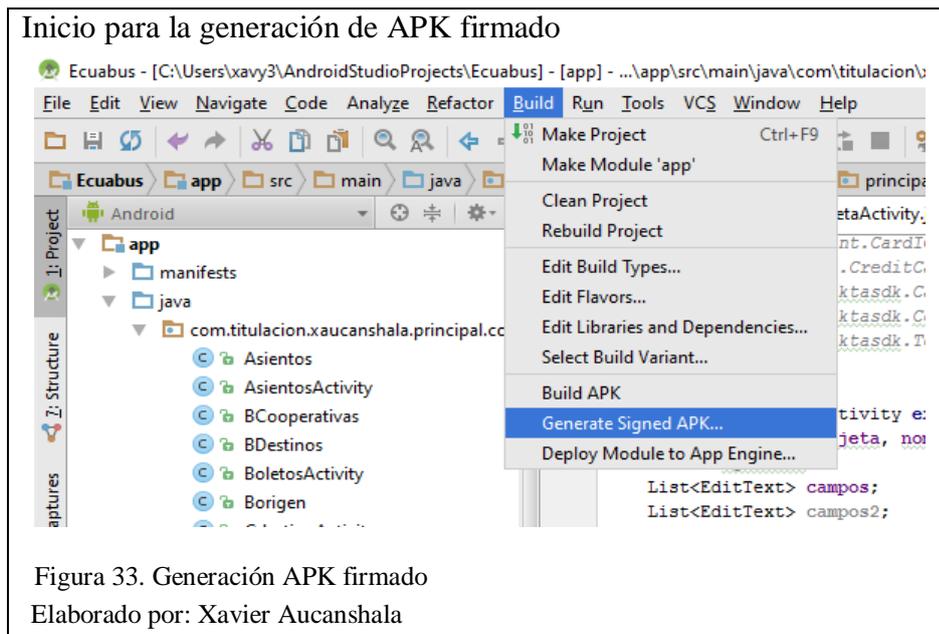
Elaborado por: Xavier Aucanshala

3.3. Pruebas e implementación

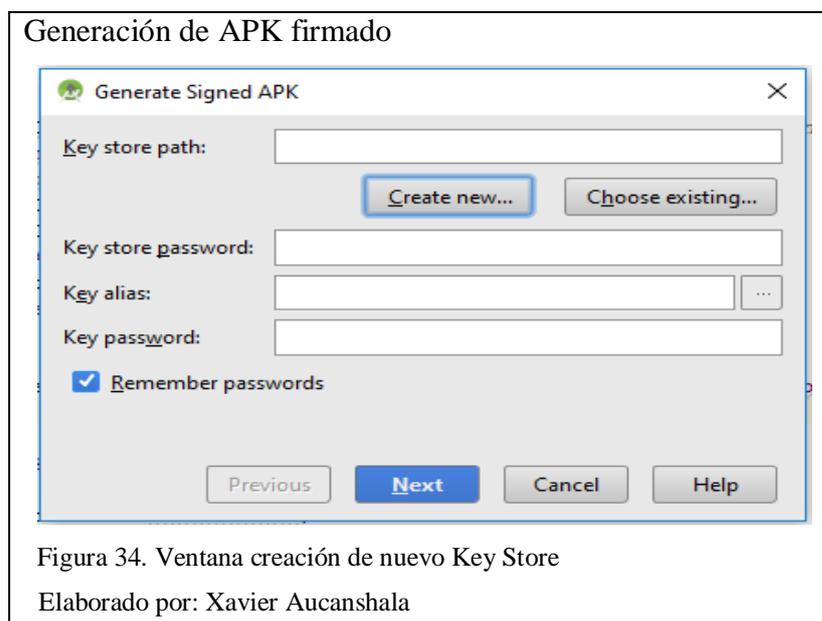
Las pruebas de funcionamiento y rendimiento de la aplicación móvil ECUA Buses se realizan tomando en consideración la versión de Android 6.0.1 o superior sobre la cual está desarrollada la aplicación móvil.

3.3.1. Implementación de KeyStore Android.

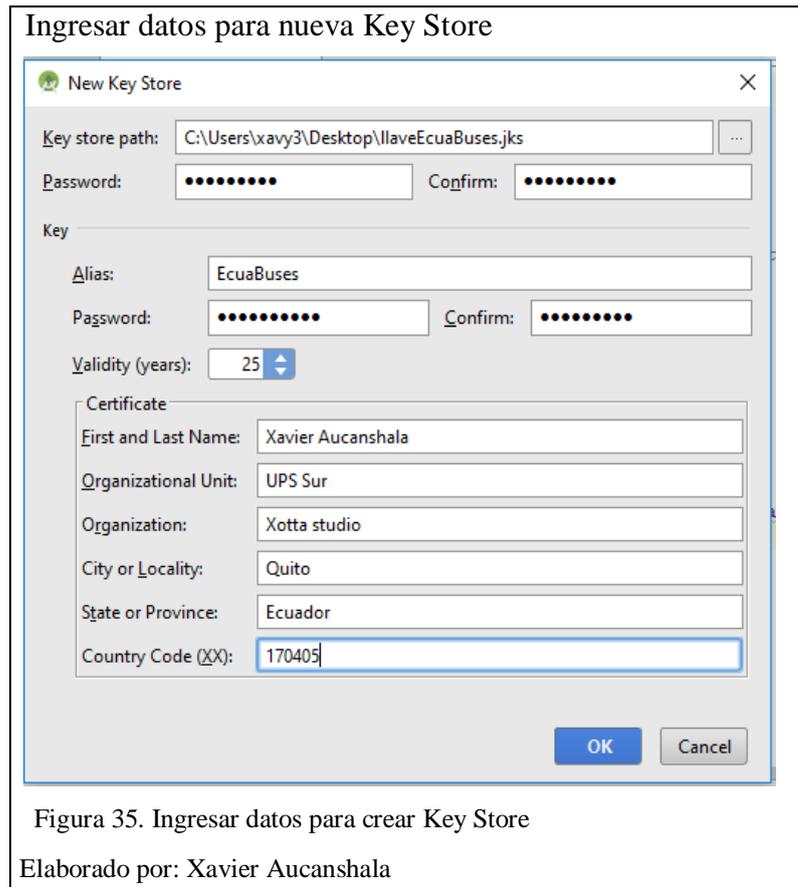
Para poder iniciar con la implementación de la aplicación móvil ECUA Buses, es primordial la creación de un nuevo almacén de datos.



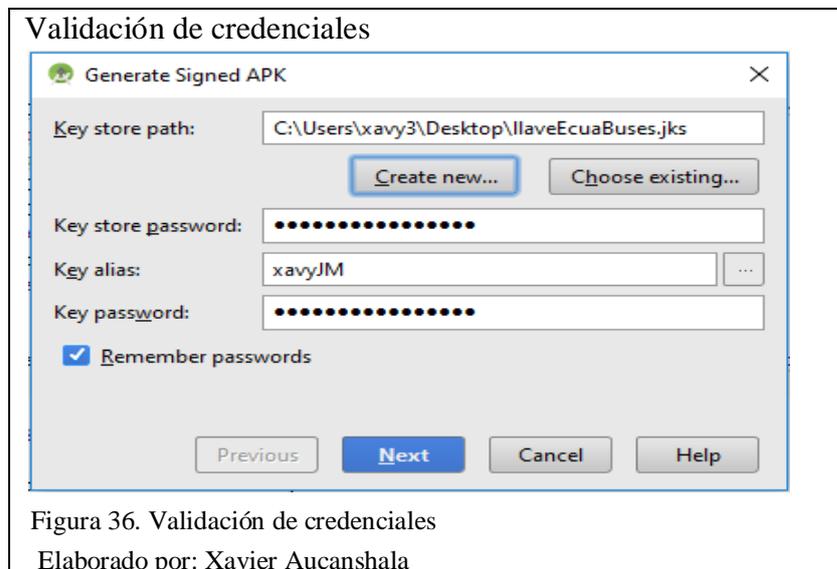
En la figura 33 se puede observar en herramientas que se escoge **Build** para buscar la opción **Generate Signed APK...**, que sirve para crear una APK firmada con todas las APIs que este incorporadas en la aplicación móvil.



En la figura 34 se muestra todos los parámetros que se requieren para generar una nueva APK firmada presionando **Create new...** o si ya se tienen una Key Store creada simplemente presionar **next**.



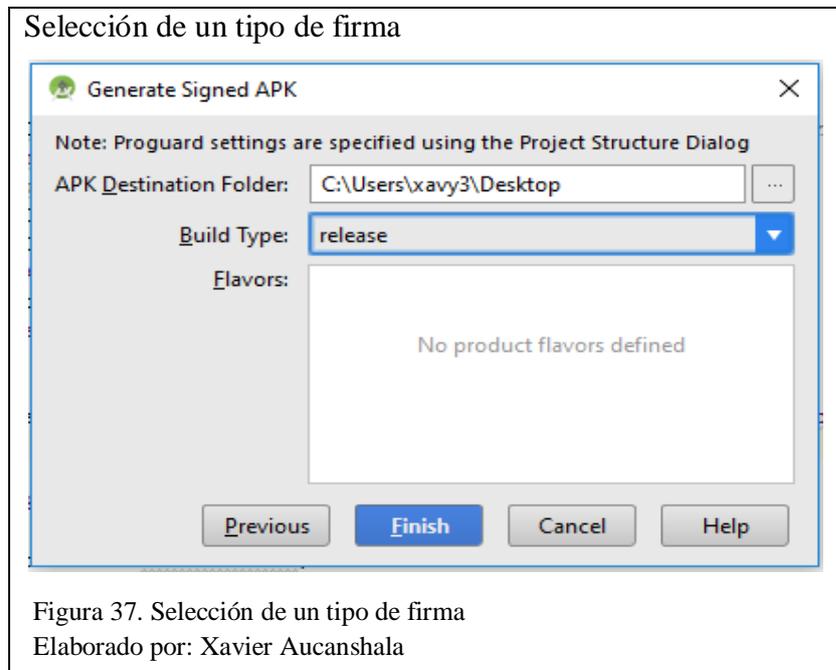
En la figura 35 se muestra que hay que ingresar un directorio raíz donde se guarda las claves privadas y el resto de información a ingresar, presionar **OK**.



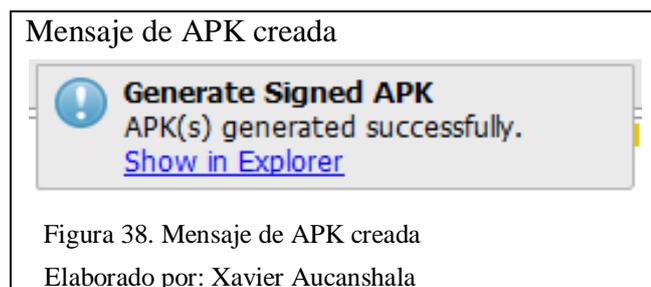
En la figura 36 se muestra como se ha creado el Key Store con el respectivo alias y clave privada.

3.3.2. Generación del APK

APK significa (Application Package File). Un archivo con extensión .apk es un paquete para el sistema operativo Android. A través de estos archivos es posible instalar programas que funcionan tanto en teléfonos móviles y tablets.



En la figura 37 muestra que se debe escoger entre dos tipos de firmas ya sea reléase o debug, además de escoger una carpeta donde se destinara la apk y presionar **Finish**.



En la figura 38 se muestra el mensaje que se ha creado la APK satisfactoriamente y si puede explorar donde esta almacenada la apk.

3.3.3. Pruebas de funcionamiento de la Aplicación Móvil.

Se utiliza dos Smartphone con sistema operativo Android, con versiones diferentes para la realización de pruebas en la aplicación móvil, a continuación se detallan las características de cada dispositivo móvil:

- | | |
|---|--|
| <ul style="list-style-type: none"> ▪ SONY XPERIA Z3 Compact <ul style="list-style-type: none"> • Sistema Operativo: Android 6.0.1 (Marshmallow) • Procesador: Quad-Core, 2.5 GHz • Memoria RAM: 2 GB • Memoria interna: 16 GB | <ul style="list-style-type: none"> ▪ SONY XPERIA Z4 <ul style="list-style-type: none"> • Sistema Operativo: Android 7.0 (Nougat) • Procesador: Octa-core, 6 GHz • Memoria RAM: 3 GB • Memoria interna: 32 GB |
|---|--|

3.3.3.1. Prueba en Búsqueda de Cooperativas

ECUA Buses (aplicación móvil) da al usuario general la posibilidad de buscar todas las cooperativas de acuerdo a la ciudad de origen y ciudad de destino que el usuario seleccione. Al realizar la búsqueda de las cooperativas en los dispositivos móviles la velocidad en el XPERIA Z3 Compact y el SAMSUNG GALAXY S4 son casi imperceptible debido a sus características de gama alta, y el tiempo de respuesta es muy aceptable para ambos dispositivos móviles, siendo la media del tiempo de respuesta de 1000ms.

Búsqueda de cooperativas por ciudad destino

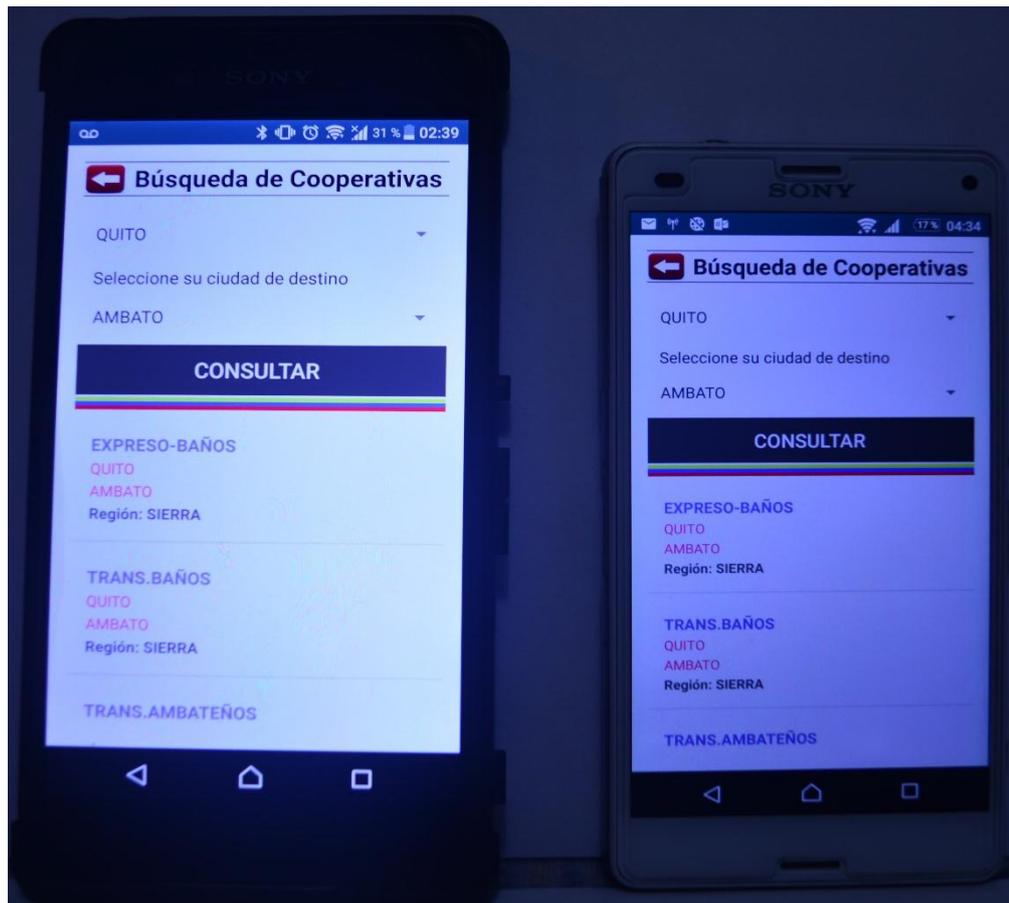


Figura 39. Búsqueda de cooperativas por ciudad destino

Elaborado por: Xavier Aucanshala

En la figura 39 se muestra la información de todas las cooperativas encontradas en la consulta, mediante la selección de una ciudad origen y una ciudad destino. Y por los tiempos de respuesta realmente rápidos y confiables no se presenta ningún inconveniente, por lo tanto el módulo Cooperativas trabaja satisfactoriamente cumpliendo con los requerimientos funcionales.

Información detallada de una cooperativa

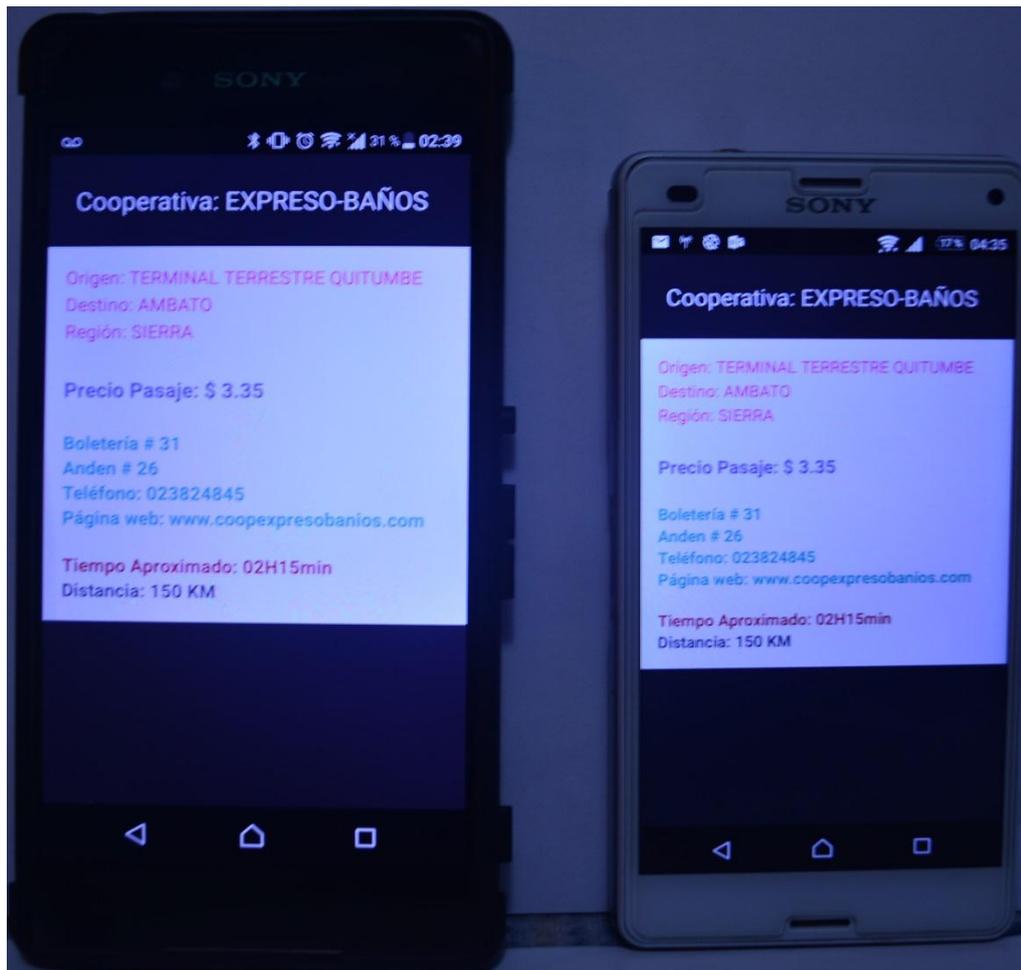


Figura 40. Información detallada de una cooperativa
Elaborado por: Xavier Aucanshala

En la figura 40 se muestra la información detallada de una cooperativa seleccionada mediante consulta, Y por los tiempos de respuesta realmente rápidos y confiables no se presenta ningún inconveniente, por lo tanto la información detallada de una cooperativa trabaja satisfactoriamente cumpliendo con los requerimientos funcionales.

3.3.3.2. Prueba en Búsqueda de Horarios

Otro de los módulos de la aplicación ECUA Buses da al usuario general la posibilidad de buscar los horarios disponibles para todas las cooperativas de acuerdo a la ciudad de origen y ciudad de destino que el usuario seleccione. De igual manera el tiempo de respuesta es muy aceptable para ambos dispositivos móviles.

Búsqueda de cooperativas por ciudad destino horarios

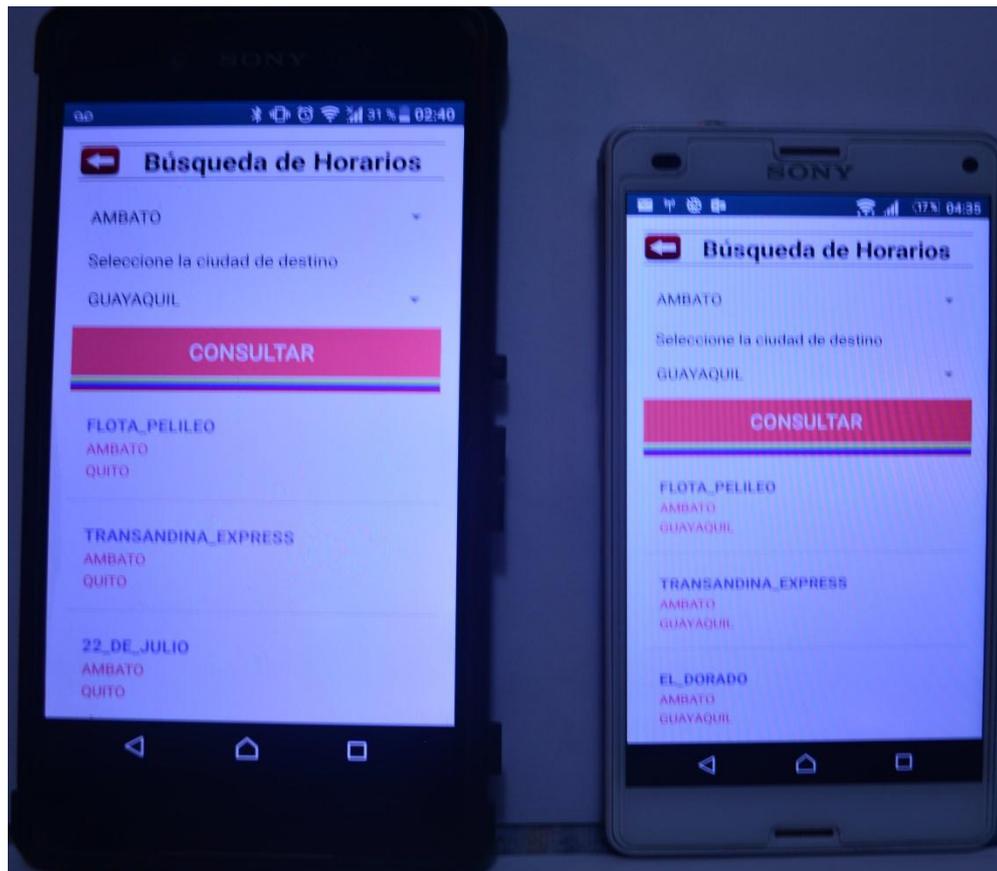


Figura 41. Búsqueda de cooperativas por ciudad destino horarios

Elaborado por: Xavier Aucanshala

En la figura 41 se muestra la información de todas las cooperativas encontradas en la consulta, mediante la selección de una ciudad origen y una ciudad destino. Y por los tiempos de respuesta realmente rápidos y confiables no se presenta ningún inconveniente, por lo tanto el módulo Horarios trabaja satisfactoriamente cumpliendo con los requerimientos funcionales.

Horarios y precios de una cooperativa

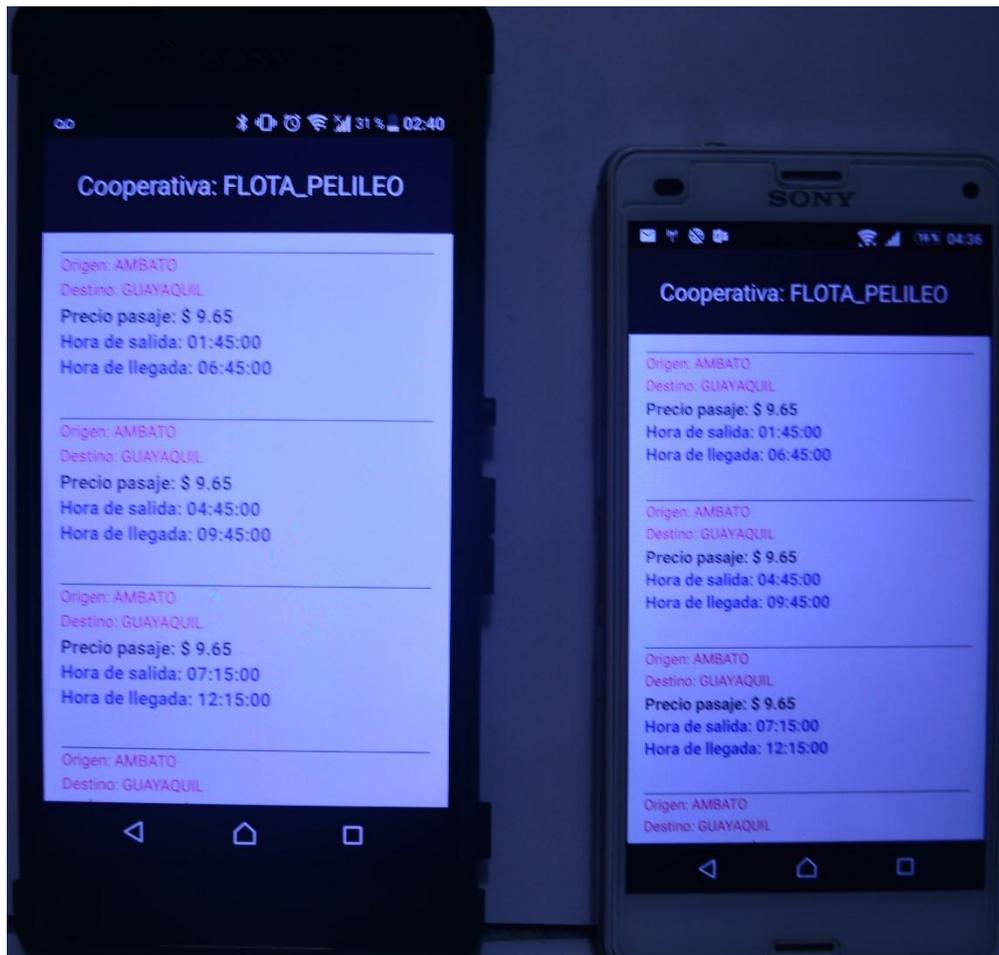


Figura 42. Horarios y precios de una cooperativa
Elaborado por: Xavier Aucanshala

En la figura 42 se muestra la información de horarios y precios de una cooperativa seleccionada mediante consulta, Y por los tiempos de respuesta realmente rápidos y confiables no se presenta ningún inconveniente, por lo tanto la información detallada de horarios y precios de una cooperativa trabaja satisfactoriamente cumpliendo con los requerimientos funcionales.

3.3.3.3. Prueba en Búsqueda de Terminales Terrestres

Otro de los módulos de la aplicación ECUA Buses da al usuario general la posibilidad de buscar las terminales terrestres disponibles en el mapa. De igual manera el tiempo de respuesta es muy aceptable para ambos dispositivos móviles.

Búsqueda de terminales terrestres

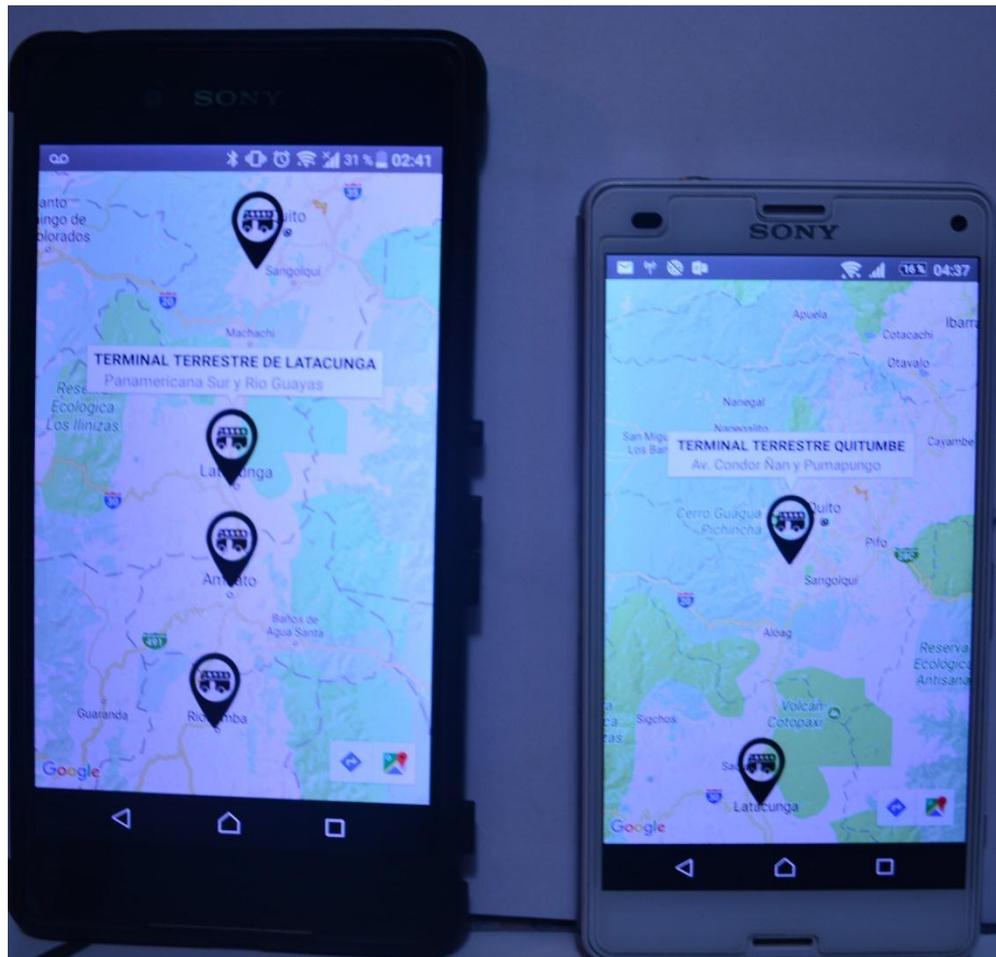


Figura 43. Búsqueda de terminales terrestres
Elaborado por: Xavier Aucanshala

En la figura 43 se muestra todos los terminales terrestres en el mapa junto a la información como dirección y nombre sobre cada icono. Y por los tiempos de respuesta realmente rápidos y confiables no se presenta ningún inconveniente, por lo tanto el módulo Cooperativas trabaja satisfactoriamente cumpliendo con los requerimientos funcionales.

3.3.3.4. Prueba en Compra de Boletos

Otro de los módulos de la aplicación ECUA Buses da al usuario general la posibilidad de comprar un boleto en línea para una cooperativa de acuerdo a la ciudad de origen y ciudad de destino que el usuario seleccione. De igual manera el tiempo de respuesta es muy aceptable para ambos dispositivos móviles.

Búsqueda de turnos disponibles

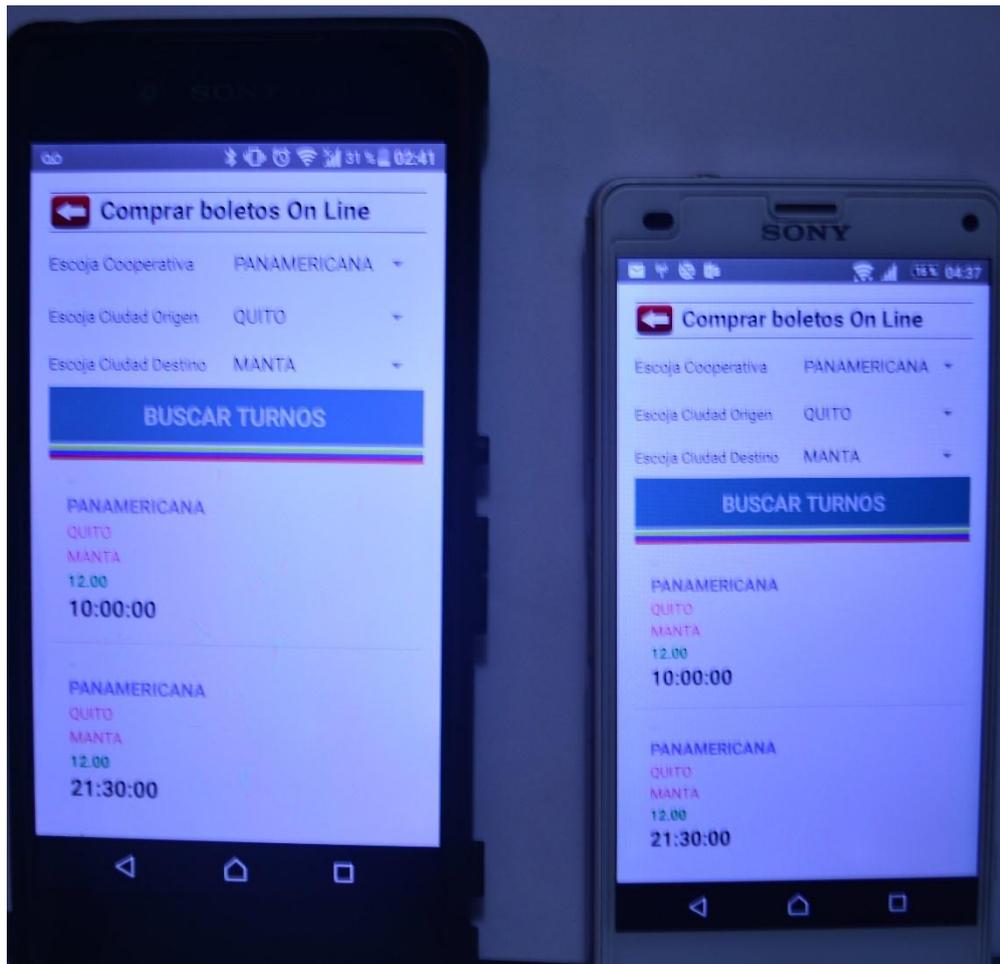


Figura 44. Búsqueda de turnos disponibles

Elaborado por: Xavier Aucanshala

En la figura 44 se muestra la información de todos los turnos disponibles para una cooperativa mediante la selección de una ciudad origen y una ciudad destino. Y por los tiempos de respuesta realmente rápidos y confiables no se presenta inconveniente alguno, por lo tanto el módulo Cooperativas cuenta con los requerimientos funcionales trabajando satisfactoriamente.

Selección de fecha y asiento para viaje

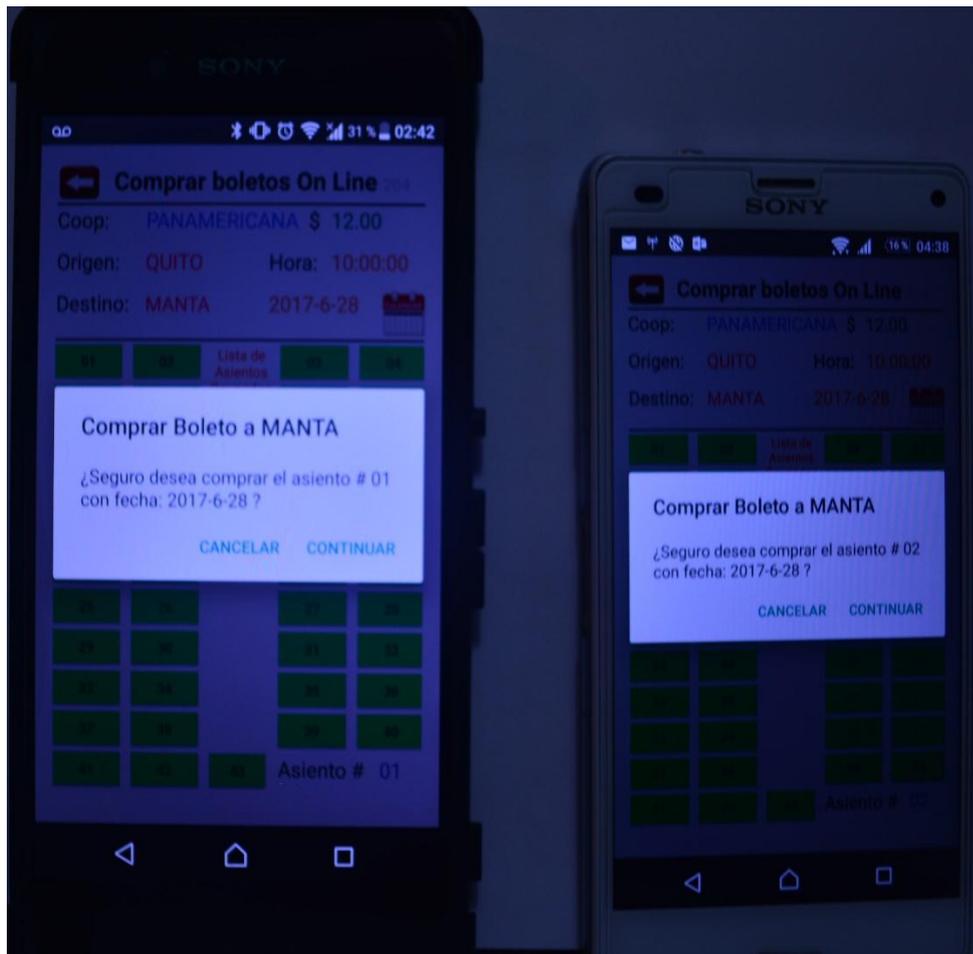


Figura 45. Selección de fecha y asiento para viaje
Elaborado por: Xavier Aucanshala

En la figura 45 se muestra la información de selección de fecha y un asiento para realizar la compra del boleto. Y por los tiempos de respuesta realmente rápidos y confiables no se presenta inconveniente alguno, por lo tanto la información detallada de una cooperativa cuenta con los requerimientos funcionales trabajando satisfactoriamente.

Pago de boleto con tarjeta de crédito

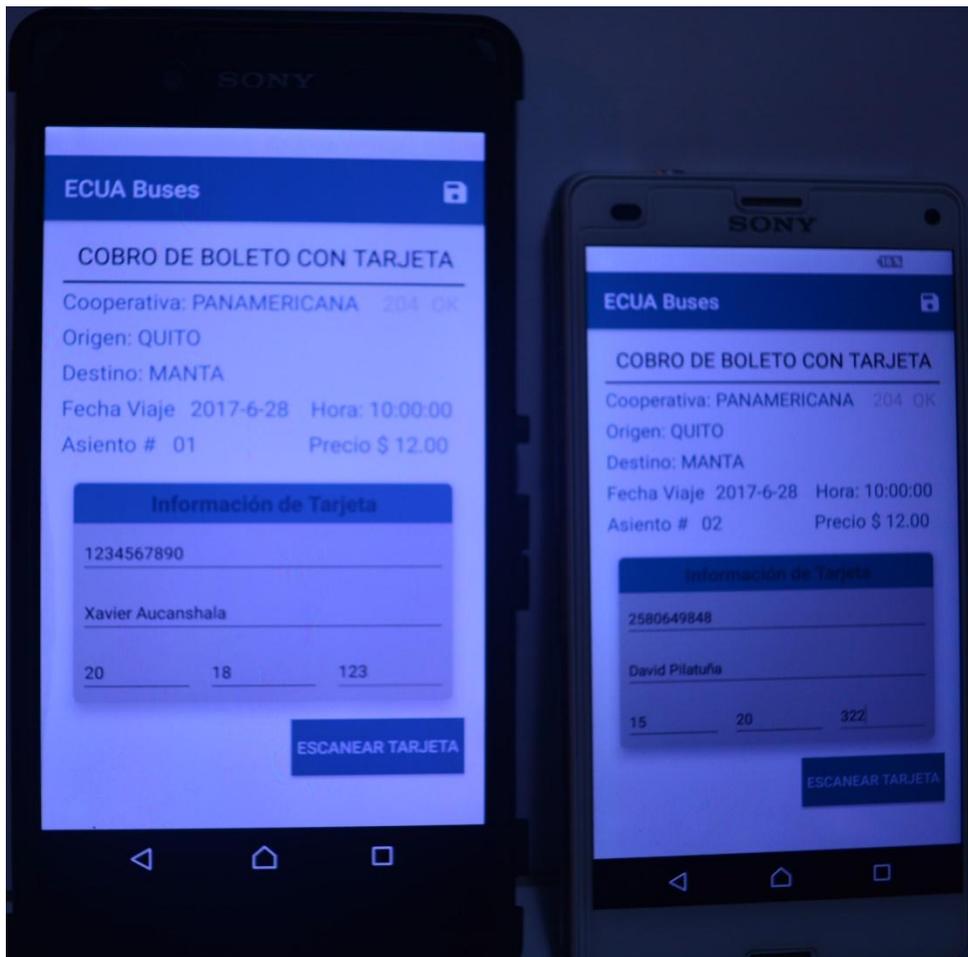


Figura 46. Pago de boleto con tarjeta de crédito

Elaborado por: Xavier Aucanshala

En la figura 46 se muestra la información detallada del boleto a comprar y el escaneo de la tarjeta para completar toda la información necesaria para posterior envío a correo personal. Y por los tiempos de respuesta realmente rápidos y confiables no se presenta inconveniente alguno, por lo tanto la información detallada de una cooperativa cuenta con los requerimientos funcionales trabajando satisfactoriamente.

Envío de información de boleto a correo

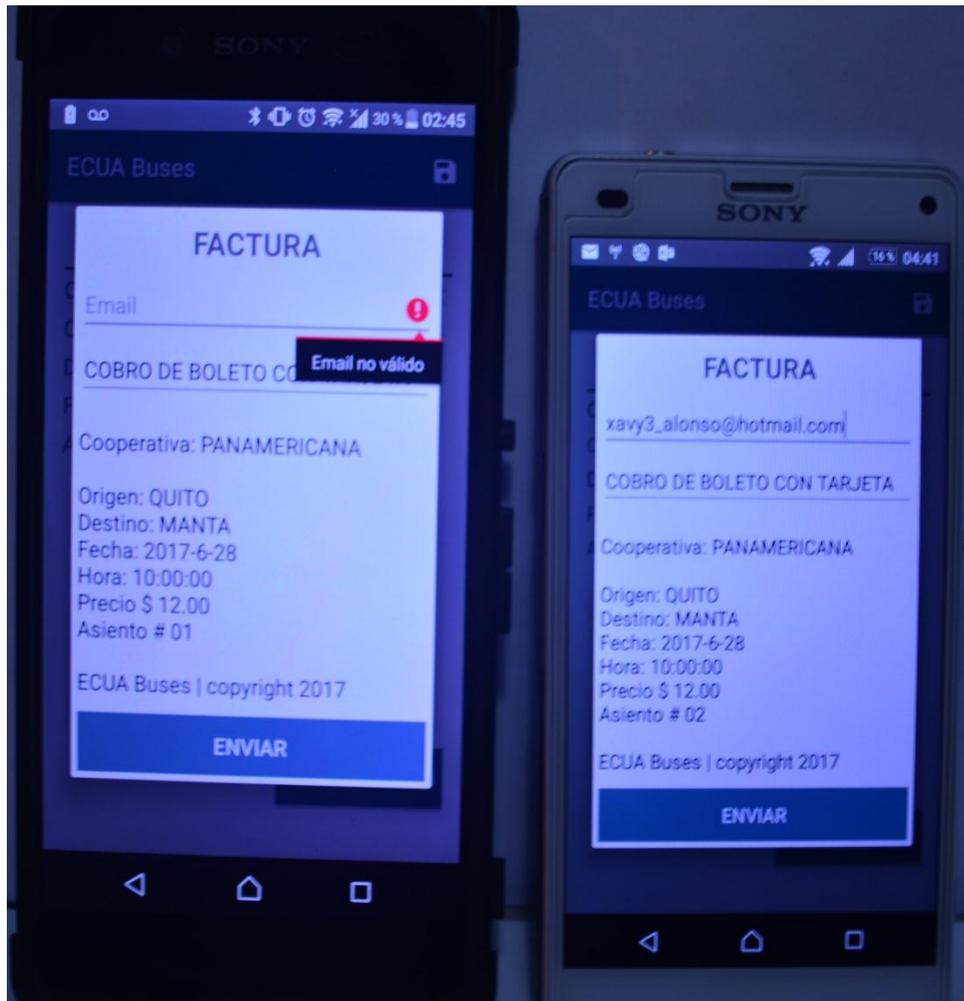
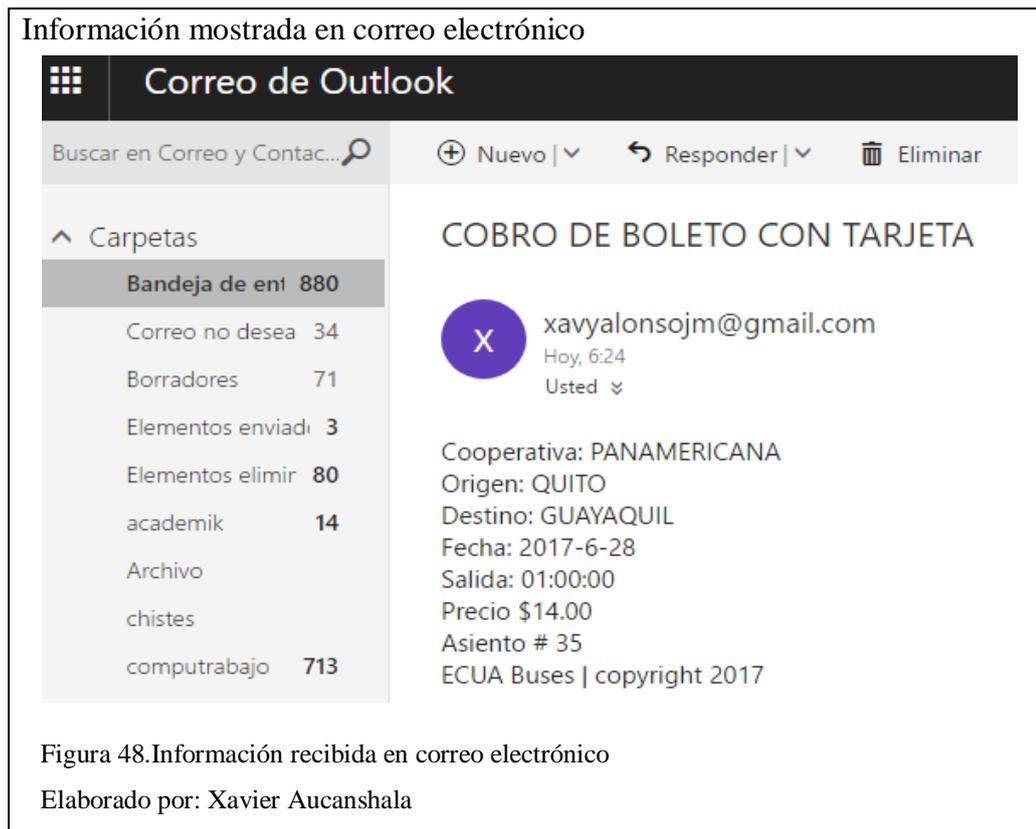


Figura 47. Envío de información de boleto a correo

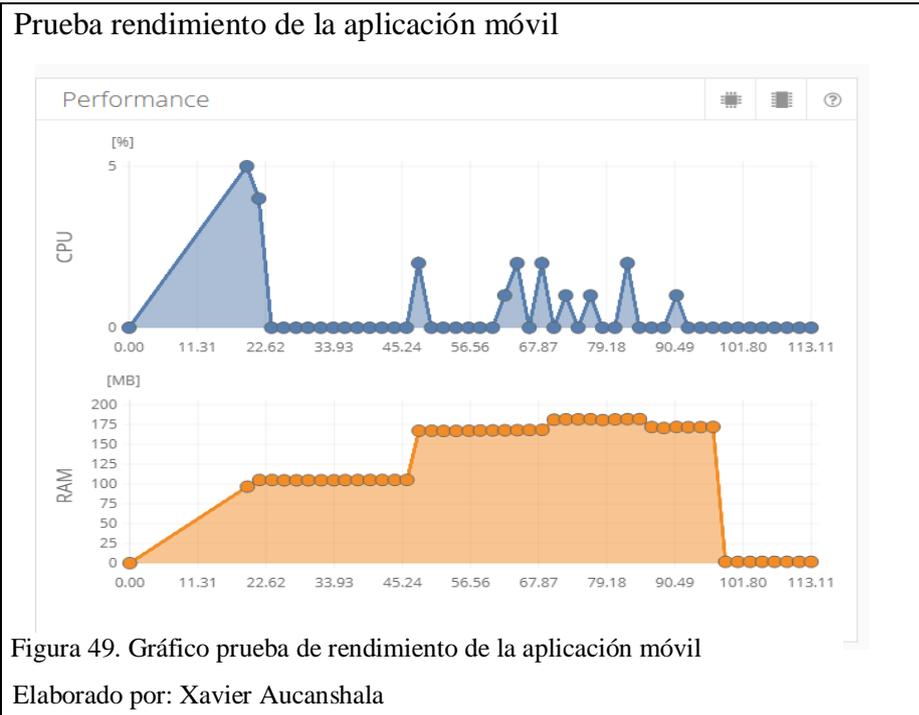
Elaborado por: Xavier Aucanshala

En la figura 47 se muestra toda la información detallada del boleto comprado a enviarse a un correo personal como respaldo de su compra. Los tiempos de respuesta son de 5000ms siendo rápidos y confiables no se presenta inconveniente alguno, por lo tanto la información detallada de una cooperativa cuenta con los requerimientos funcionales trabajando satisfactoriamente.



En la figura 48 se muestra toda la información detallada del boleto comprado que se recibe en el correo personal como respaldo de su compra. Y por los tiempos de respuesta realmente rápidos y confiables no se presenta inconveniente alguno, por lo tanto la información detallada de una cooperativa cuenta con los requerimientos funcionales trabajando satisfactoriamente.

3.3.4. Prueba de rendimiento de la Aplicación Móvil.



En la figura 49 se muestra la gráfica de la prueba de rendimiento de la aplicación móvil realizado en un dispositivo Android 6.0.1, de manera que se visualiza un desgaste de RAM al momento de ingresar y regresar al menú principal de la aplicación, y un bajo desgaste del procesador durante todo el uso de la aplicación. Estas pruebas se realizaron con el Software TestDroid.

3.3.5. Pruebas de carga de la Aplicación Móvil.

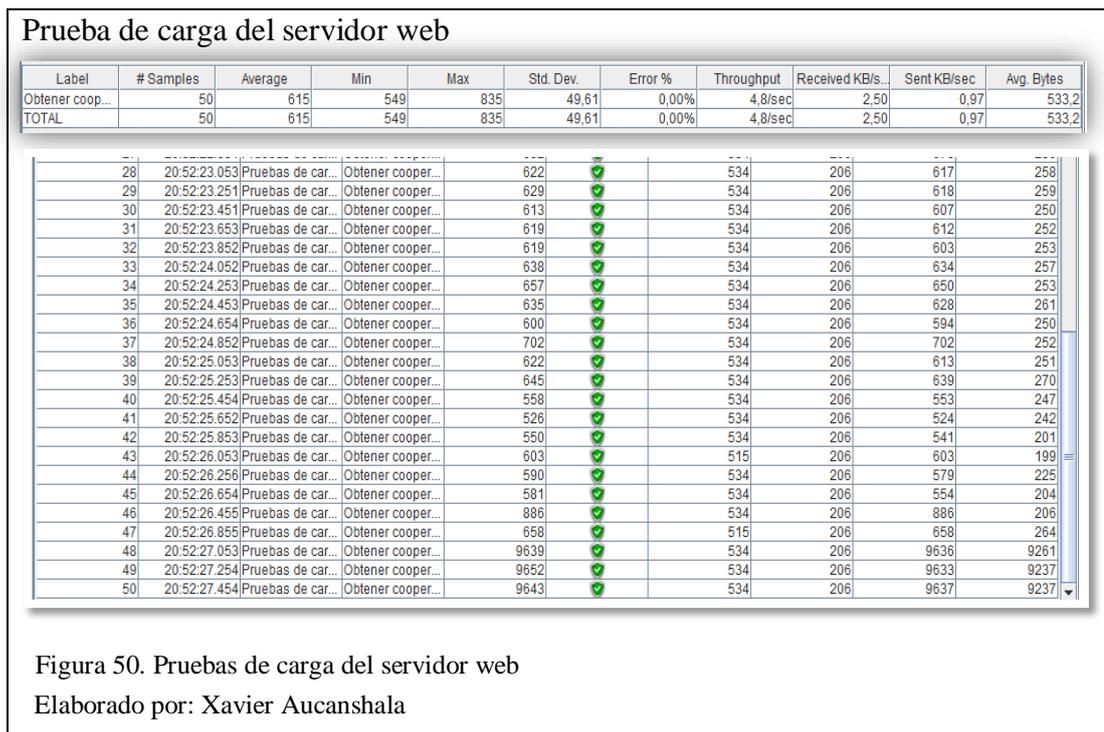
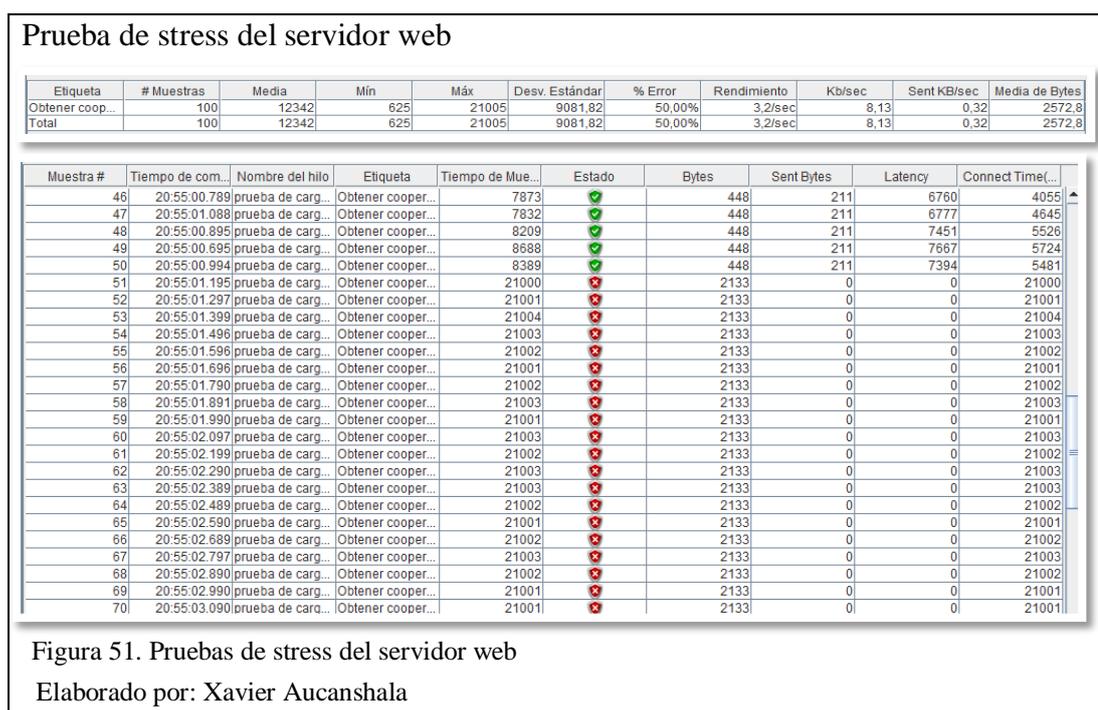


Figura 50. Pruebas de carga del servidor web

Elaborado por: Xavier Aucanshala

En la figura 50 se muestra la gráfica de la prueba de carga de la aplicación móvil realizado sobre el servidor web, la prueba se la realiza ante el alta de 50 usuarios cada 10 segundos de manera simultánea, verificando la correcta respuesta a las peticiones realizadas de parte del cliente hacia el servidor web. Estas pruebas se realizaron con el Software Jmeter.

3.3.6. Pruebas de stress de la Aplicación Móvil.



En la figura 51 se muestra la gráfica de la prueba de stress de la aplicación móvil realizado sobre el servidor web, la prueba se la realiza ante el alta de 100 usuarios cada 10 segundos de manera simultánea, verificando que la respuesta del servidor web hacia la aplicación comienza a fallar o es incapaz de responder a las peticiones a partir del usuario 51. Estas pruebas se realizaron con el Software Jmeter.

3.3.7. Resultados de la Aplicación Móvil.

La aplicación móvil ECUA Buses en su versión 1.0 ha obtenido buenas críticas de acuerdo a los comentarios que se visualizan en la Play Store, esto en base a la primera semana de haberse lanzado oficialmente la app en dicha tienda de aplicaciones, al ser una app gratuita y compatible con versiones de Android 5.0 y superior ya registra 50-100 descargas desde el 17 de julio hasta el 30 de agosto del 2017. De esta manera se está logrando resultados positivos en cuanto a la funcionalidad de la app y cumpliendo sus objetivos.

Algunos comentarios de la app ECUA Buses:

- “Es una app muy interesante.. Creo yo que nos servirá muchísimo en la vida cotidiana. Deberían de unirse mas cooperativas de transporte!”
- “Es muy útil la aplicación para los que viajamos constantemente. Con esta app podemos saber los horarios de los buses, y no perder tiempo de espera en los terminales”
- “Me parece funcional aunque para que sea bastante comercial me gustaría saber dónde está el bus como para planificar horarios, no he visto aplicaciones que hagan eso”



En la figura 52 se muestra el gráfico de barras de la calificación obtenida por la aplicación móvil ECUA Buses en su primera semana de lanzamiento, por lo que se logra dar a la ciudadanía en general una aplicación móvil informativa, que está disponible en tiempo real y con información actualizada.

CONCLUSIONES

- Utilizar sistemas operativos open source permiten crear aplicaciones móviles de buena calidad sin la necesidad de tener que comprar licencias, permitiendo reducir costos de presupuesto para el desarrollo de un proyecto, ya que las actualizaciones son constantes por el hecho de ser un software libre.
- La obtención de datos solo fue posible a través de fotografías en las distintas boleterías de las terminales terrestres ya que fue imposible obtener información directa de la FENACOTIP debido a sus políticas internas.
- Se ha logrado desarrollar e implementar la aplicación móvil para consulta de cooperativas de buses interprovinciales, ayudando a automatizar y gestionar la búsqueda de información de cooperativas de transporte interprovincial hacia una ciudad destino dentro del territorio ecuatoriano.
- Se ha logrado cumplir con los objetivos establecidos para el proyecto de manera que la aplicación está en capacidad de buscar información, horarios, precios de las cooperativas, además de ubicar los terminales terrestres dentro de un mapa a través de georreferenciación también comprar boletos de buses interprovinciales.
- Para realizar la conexión a la base de datos remota MySQL con la aplicación móvil no existe un método directo, por lo tanto se recurre a utilizar un servidor web entre PHP+JSON que permite el intercambio de datos con Android, logrando una comunicación eficaz entre la aplicación móvil ECUA Buses y el servidor que aloja la base de datos remota MySQL.
- Las pruebas realizadas a la aplicación móvil con dos dispositivos móviles de gama alta con versión 6.0.1 o más, permiten determinar el correcto

funcionamiento de la aplicación en todos sus módulos, en cuanto a tiempos de respuesta de 1000ms por consulta realizada.

- El modulo Boletos está desarrollado en un ambiente de prototipo ya que si la aplicación se pusiese en producción se establecería un tipo de cobro con tarjeta asociado a una plataforma que permita procesar pagos en línea ya que actualmente solo se realiza pagos a través de envío de datos a un correo electrónico.
- El desarrollo de aplicaciones móviles en forma gratuita permite acceder a cualquier usuario con un “smartphone” con sistema operativo Android, ayudando a los usuarios a que utilicen la aplicación de manera correcta, continúa y para el propósito que fue creada.

RECOMENDACIONES

- Debido a las distintas versiones de Android es necesario realizar las pruebas de esta aplicación con la mayor cantidad de emuladores o dispositivos posibles, de esta manera tener un panorama más claro sobre en qué dispositivos funcionará correctamente o incorrectamente esta aplicación.
- Para una próxima versión o mejora de esta aplicación es recomendable migrarla para el sistema operativo IOS ya que también es un campo muy utilizado por los usuarios de dispositivos móviles en el Ecuador.
- Se recomienda instalar la aplicación en un dispositivo móvil de gama alta con versión Android 6.0.1 (Marshmallow) o superior, para que la aplicación funcione correctamente en todos sus módulos.
- Es aconsejable mantener actualizado la información de las cooperativas para así evitar datos erróneos y generar confianza entre usuarios actuales o futuros usuarios de la aplicación móvil.

GLOSARIO DE TÉRMINOS

ECUA Buses: Nombre de la aplicación móvil Ecuador Transporte de Buses

LOTTTSV: Según la Ley Orgánica de Transporte Terrestre, Tránsito y Seguridad Vial

FENACOTIP: Federación Nacional De Cooperativas De Transporte Público De Pasajeros Del Ecuador.

BDD: Base de Datos

GPS: Sistema de Posicionamiento Global

POP: Process Oriented Programming o Programación orientada a Objetos

API: Application Programming Interface o Interfaz de Programación de Aplicaciones

Xotta studio: Marca personal del desarrollador de software Xavier Aucanshala

PHP: Hypertext Preprocessor

MySQL: Sistema de gestión de bases de datos relacional

JSON: Acrónimo de JavaScript Object Notation

IDE: Entorno De Desarrollo Integrado

XML: eXtensible Markup Language o Lenguaje de Marcado Extensible

KeyStore: Repository Of Security Certificates o Repositorio de Certificados de Seguridad

APK: Application Package File o Aplicación empaquetada de Android

PDO: Objetos de Datos de PHP

GET: Este método envía las variables dentro de la propia URL de la página

SDK: Software Development Kit o Kit de Desarrollo De Software

POST: Este método sólo se puede usar con formularios no por URL.

LISTA DE REFERENCIAS

- Aguirre, F. (2011). Diseño e implementación de un laboratorio de Software y redes mediante del uso de un servidor de terminales. Obtenido de <http://dspace.epoch.edu.ec/bitstream/123456789/171/1/38T00162.pdf>
- ALEGSA. (2017). Definición de programación orientada a objetos. Obtenido de http://www.alegsa.com.ar/Dic/programacion_orientada_a_objetos.php
- AS Technology. (05 de 01 de 2016). ¿Qué Son Y Para Qué Sirven Las Aplicaciones Móviles? Obtenido de <https://www.astechny.com/que-son-las-aplicaciones-moviles/>
- Asamblea Constituyente. (2008). CONSTITUCIÓN DEL ECUADOR. Obtenido de http://www.asambleanacional.gov.ec/documentos/constitucion_de_bolsillo.pdf
- developer.android.com. (20 de 06 de 2017). Conoce Android Studio. Obtenido de <https://developer.android.com/studio/intro/index.html?hl=es-419>
- Developers Android. (s.f.). Sistema Android Keystore. Obtenido de <https://developer.android.com/training/articles/keystore.html?hl=es-419>
- developers de google. (10 de 09 de 2016). Google Maps para cada plataforma. Obtenido de <https://developers.google.com/maps/>
- FENACOTIP. (02 de 09 de 2016). www.fenacotip.com.ec. Obtenido de <http://fenacotip.com.ec>
- Franco, A. (2011). Programación en el lenguaje Java. Obtenido de <http://www.sc.ehu.es/sbweb/fisica/cursoJava/Intro.htm>
- Illescas Ochoa, S. d. (2007). Proyecto de desarrollo para mejorar el sistema de transportacion interprovincial analisis de ruta Guayaquil-Salitre-Vinces. Obtenido de www.espol.edu.ec
- INEC. (18 de 12 de 2016). EL TRANSPORTE TERRESTRE DE PASAJEROS EN ECUADOR Y QUITO. Obtenido de PERSPECTIVA HISTÓRICA Y SITUACIÓN ACTUAL: http://inec.gob.ec/inec/index.php?option=com_remository&Itemid=420&func=startdown&id=414&lang=es
- kiaranet.com. (2016). APLICACIONES MÓVILES. Obtenido de Las aplicaciones móviles permiten crear y mantener tu relación con tus clientes: <http://kiaranet.com/38>
- Loayza Abad, F. P. (2015). APLICACIÓN MÓVIL PARA LA GEOREFERENCIACIÓN DE BUSES URBANOS EN LA CIUDAD DE LOJA". Loja, LOja, Ecuador: dspace.unl.edu.ec. Obtenido de <https://dspace.unl.edu.ec>
- Ministerio de Turismo. (31 de 12 de 2014). LEY ORGANICA DE TRANSPORTE TERRESTRE. Obtenido de <http://www.turismo.gob.ec/wp-content/uploads/2016/04/LEY-ORGANICA-DE-TRANSPORTE-TERRESTRE-TRANSITO-Y-SEGURIDAD-VIAL.pdf>
- Mora, M. (10 de 12 de 2010). Historias de usuario¿Por qué? ¿Qué son? ¿Cómo son? Obtenido de <http://es.slideshare.net/MiquelMora/historias-de-usuario>

- Nevado, V. (2010). *Introducción a Las Bases de Datos Relacionales*. Madrid: Vision Libros.
- OK HOSTING. (10 de 12 de 2016). *Metodologías del Desarrollo del Software*. Obtenido de <http://okhosting.com/blog/metodologias-del-desarrollo-de-software/>
- ProgrammableWeb. (10 de 08 de 2017). *API de Google Maps*. Obtenido de <https://www.programmableweb.com/api/google-maps>
- Ruiz, J., Díaz, J., & Cervantes, J. (2014). *API de Google Maps para un mapa de conocimiento de los asesores especializados de un Centro de Desarrollo Empresarial*. Obtenido de http://www.rcs.cic.ipn.mx/2014_73/API%20de%20Google%20Maps%20para%20un%20mapa%20de%20conocimiento%20de%20los%20asesores%20especializados.pdf
- Sáez, P. S. (2011). *GPSLoc. Localización y Relaciones Sociales en el ámbito de los Teléfonos Inteligentes*. Obtenido de <https://riunet.upv.es/bitstream/handle/10251/15641/GPSLoc.%20Localizaci%C3%B3n%20y%20Relaciones%20Sociales%20en%20el%20%C3%A1mbito%20de%20los%20Tel%C3%A9fonos%20Inteligentes.pdf?sequence=1>
- Sánchez Bonilla, A. J. (05 de 2015). *ESTUDIO DE LOS CONOCIMIENTOS QUE TIENEN LOS CIUDADANOS DE GUAYAQUIL ACERCA DE LOS RECORRIDOS DE LAS LINEAS DE TRANSPORTE URBANO PARA DETERMINAR SUS NECESIDADES DE INFORMACION E IMPLEMENTAR UN PROTOTIPO PARA MOVILES CON ANDROID QUE SIRVA DE GUIA A ...* Obtenido de repositorio.ug.edu.ec/bitstream/.../PTG-590-Sánchez%20Bonilla%20Angel%20Joel
- Sanchez, R. (11 de 06 de 2012). *Aplicación de geo-localización Forns IGP iOS / Android*. Obtenido de <http://upcommons.upc.edu/bitstream/handle/2099.1/15639/77863.pdf>
- Sanz, D., Saucedo, M., & Torralbo, P. (2011). *Introducción a Android*. Obtenido de <http://www.it-docs.net/ddata/18.pdf>
- SG Buzz. (09 de 2012). *La importancia de las metodologías ágiles*. Obtenido de https://sg.com.mx/buzz/la-importancia-las-metodologias-agiles#.WFduf_nhC00
- VASS digital. (20 de 09 de 2012). *SCRUM, desarrollo ágil por excelencia*. Obtenido de <http://www.vassdigital.com/scrum-la-metodologia-de-desarrollo-agil-por-excelencia/>
- www.ecuadorencifras.gob.ec. (20 de 06 de 2016). *En cinco años se quintuplicaron los usuarios de teléfonos inteligentes*. Obtenido de <http://www.ecuadorencifras.gob.ec/en-cinco-anos-se-quintuplicaron-los-usuarios-de-telefonos-inteligentes/>
- www.imaginanet.com. (10 de 12 de 2016). *SCRUM es una metodología para la*. Obtenido de <https://www.imaginanet.com/pdfinet/SCRUM%20es%20una%20metodolog%C3%A1a%20para%20la%20programaci%C3%B3n%20de%20aplicaciones%20m%C3%B3viles%20y%20Web.pdf>