



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

IMPLEMENTACIÓN DE UNA PLATAFORMA MÓVIL PARA LA OFERTA DE
PARQUEADEROS EN EL DISTRITO METROPOLITANO DE QUITO

AUTORAS

Betty Alejandra Burbano Quiñonez
Tabata Fernanda Jardim Rodríguez

AÑO

2020



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

IMPLEMENTACIÓN DE UNA PLATAFORMA MÓVIL PARA LA OFERTA DE
PARQUEADEROS EN EL DISTRITO METROPOLITANO DE QUITO

Trabajo de Titulación presentado en conformidad con los requisitos establecidos
para optar por los títulos de Ingeniera en Electrónica y Redes de Información e
Ingeniera en Redes y Telecomunicaciones

Profesor Guía

MSc. Iván Patricio Ortiz Garcés

Autoras

Betty Alejandra Burbano Quiñonez

Tabata Fernanda Jardim Rodríguez

Año

2020

DECLARACIÓN DEL PROFESOR GUÍA

"Declaro haber dirigido el trabajo, Implementación de una plataforma móvil para la oferta de parqueaderos en el Distrito Metropolitano de Quito, a través de reuniones periódicas con las estudiantes Betty Alejandra Burbano Quiñonez y Tabata Fernanda Jardim Rodríguez, en el semestre 202010, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación".



Iván Patricio Ortiz Garcés

Magister en Redes de Comunicaciones

CI: 0602356776

DECLARACIÓN DEL PROFESOR CORRECTOR

“Declaro haber revisado este trabajo, Implementación de una plataforma móvil para la oferta de parqueaderos en el Distrito Metropolitano de Quito, de las estudiantes Betty Alejandra Burbano Quiñonez y Tabata Fernanda Jardim Rodríguez, en el semestre 202010, dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación”.

A handwritten signature in blue ink, appearing to be 'W. Eduardo Villegas Chilibingua', written over a horizontal line.

William Eduardo Villegas Chilibingua

Magister en Redes de Comunicaciones

CI: 1715338263

DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

“Declaramos que este trabajo es original, de nuestra autoría, que se han citado las fuentes correspondientes y que en su ejecución respetaron las disposiciones legales que protegen los derechos de autor vigentes.”



Betty Alejandra Burbano Quiñonez

CI: 1718802380



Tabata Fernanda Jardim Rodríguez

CI: 1758253189

AGRADECIMIENTOS

Agradezco a mis padres por ser los principales motores para seguir mis sueños, gracias a ellos por cada consejo y por cada palabra que me motivaron y me guiaron durante toda mi vida.

Betty Burbano

AGRADECIMIENTOS

Agradezco a mis padres y familia por ser los motores de mi vida, los cuales siempre han estado apoyándome en cada momento y han creído en todas las metas que me he propuesto.

Tabata Jardim

DEDICATORIA

Dedico esta tesis a Dios, por darme la fuerza necesaria para continuar en este proceso y por inspirarme a conseguir mis metas.

Betty Burbano

DEDICATORIA

Este proyecto está dedicado a mis padres por enseñarme cada día a ser mejor persona y que cada vez que te caes puedes levantarte con una mejor cara, sacando un aprendizaje.

Tabata Jardim

RESUMEN

El uso de aplicaciones móviles se ha incrementado en los últimos años debido a que el proceso de desarrollo es sencillo y ofrece una solución rápida a distintas problemáticas. Las aplicaciones móviles se centran en el aprovechamiento de las nuevas tecnologías orientadas a las comunicaciones y redes, ganándose un gran puesto en el mercado de software.

El trabajo de titulación se basa en la implementación de una plataforma móvil capaz de gestionar la oferta y demanda de parqueaderos tanto públicos como privados en el Distrito Metropolitano de Quito. Esta plataforma está orientada tanto a usuarios que utilizan parqueaderos como a usuarios que ofrecen dicho servicio, facilitándoles la búsqueda y la administración de plazas de parqueo y reduciendo la dificultad de encontrar parqueaderos en la ciudad.

La aplicación móvil es desarrollada con los requisitos necesarios orientados a servicios de conexión a internet, geolocalización y procesamiento. Una referencia clave es la situación actual del problema y las mejores técnicas que se pueden emplear para solventarlo. Las mejores tecnologías en el ámbito de desarrollo de software, almacenamiento y análisis de datos son un factor importante para el funcionamiento óptimo del aplicativo móvil.

El resultado final es la implementación de un prototipo capaz de facilitar el proceso de búsqueda, adquisición y administración de parqueaderos para los usuarios y crear una base para trabajos futuros.

ABSTRACT

The use of mobile applications has increased in recent years because the development process is simple and offers a quick solution to different problems. Mobile applications focus on the use of new technologies aimed at communications and networks, earning a great position in the software market.

The titling work is based on the implementation of a mobile platform capable of managing the supply and demand of public and private parking lots in the Metropolitan District of Quito. This platform is aimed at both users who use parking lots and users who offer this service, facilitating the search and administration of parking spaces and reducing the difficulty of finding parking lots in the city.

The mobile application is developed with the necessary requirements aimed at internet connection, geolocation and processing services. A key reference is the current situation of the problem and the best techniques that can be used to solve it. The best technologies in the field of software development, storage and data analysis are an important factor for the optimal operation of the mobile application.

The result is the implementation of a prototype capable of facilitating the process of search, acquisition and administration of parking lots for users and create a basis for future work.

ÍNDICE

1. CAPÍTULO I. INTRODUCCIÓN	1
1.1.Antecedentes	1
1.2.Justificación.....	2
1.3.Alcance.....	3
1.4.Objetivo General	4
1.5.Objetivos Específicos.....	4
2. CAPÍTULO II. MARCO TEÓRICO	5
2.1.Conceptos.....	5
2.1.1. <i>Cloud Computing</i>	5
2.2.Metodología empleada	8
2.2.1.Scrum	8
2.3.Herramientas de Desarrollo.....	12
2.3.1.Android Studio	12
2.4.Sistema Operativo	14
2.4.1.Android S.O	15
2.4.1.1.Características.....	15
2.4.1.2.Arquitectura	15
2.5.Lenguaje de Programación	16
2.5.1.Java	16
2.5.1.1.Componentes	16
2.5.2.eXtensible Markup Language - XML.....	17

2.5.3.JavaScript Object Notation – JSON.....	17
2.6.Almacenamiento	18
2.6.1.MySQL.....	18
2.6.2.Encryptación SHA.....	19
2.7.Herramientas de trabajo y de análisis.....	19
2.7.1. <i>Dashboard</i>	19
2.7.2.Requerimientos.....	20
2.7.2.1.Requerimientos funcionales	20
2.7.2.2.Requerimientos no funcionales	20
2.7.3.Modelo entidad relación.....	21
2.7.3.1.Entidad	22
2.7.3.2.Relaciones.....	22
3. CAPÍTULO III. ANÁLISIS DE REQUERIMIENTOS PARA LA SOLUCIÓN DEL PROBLEMA	22
3.1.Recolección de Datos.....	23
3.2.Procesamiento de Información	25
3.3.Resultado del Análisis.....	27
3.4.Enfoques de seguridad.....	29
3.4.1.Confidencialidad	29
3.4.2.Integridad.....	29
3.4.3.Disponibilidad	29
4. CAPÍTULO IV. DESARROLLO DEL PROTOTIPO CON PARÁMETROS DE SEGURIDAD	30

- 4.1.Diseño..... 30
 - 4.1.1.Diagrama de Arquitectura..... 30
 - 4.1.2.Diagrama Entidad – Relación 31
 - 4.1.3.Esquema Lógico Estándar..... 31
- 4.2.Implementación 32
 - 4.2.1.Product Backlog..... 33
 - 4.2.2.Historias de Usuario 34
 - 4.2.3.Sprints 36
 - 4.2.3.1.Sprint 1 37
 - 4.2.3.2.Sprint 2 42
 - 4.2.3.3.Sprint 3 47
 - 4.2.3.4.Sprint 4 50
 - 4.2.4.Tabla de requerimientos no funcionales 54
- 5. CAPÍTULO IV. ANÁLISIS DE RESULTADOS 55**
 - 5.1.Verificación de funcionalidad 55
 - 5.1.1.Primer historia de usuario: Ingresar Usuario..... 56
 - 5.1.2.Segunda historia de usuario: Modificar Usuario 58
 - 5.1.3.Tercera historia de usuario: Login Usuario 60
 - 5.1.4.Cuarta historia de usuario: Mostrar información del Usuario 60
 - 5.1.5.Quinta historia de usuario: Ingresar Parqueadero 61
 - 5.1.6.Sexta historia de usuario: Modificar Parqueadero 64
 - 5.1.7.Séptima historia de usuario: Eliminar Parqueadero..... 66
 - 5.1.8.Octava historia de usuario: Mostrar Parqueadero del Usuario 66

5.1.9. Novena historia de usuario: Mostrar Parqueadero registrados en la aplicación móvil.....	67
5.1.10. Décima historia de usuario: Ingresar Reserva.....	68
5.1.11. Décima primera historia de usuario: Eliminar Reserva.....	70
5.1.12. Décima segunda historia de usuario: Confirmar Reserva.....	71
5.1.13. Décima tercera historia de usuario: Contactar a Usuario.....	73
5.1.14. Décima cuarta historia de usuario: Análisis de Reservas Realizadas.....	75
5.1.15. Décima quinta historia de usuario: Análisis de Reservas.....	75
5.1.16. Décima sexta historia de usuario: Análisis de Gastos.....	76
5.1.17. Décima sexta historia de usuario: Análisis de Ganancias.....	77
5.2. Análisis de Pruebas.....	78
5.2.1. Análisis 1: Análisis de Requerimientos no Funcionales.....	79
5.2.2. Análisis 2: Tiempos de Respuesta.....	80
5.2.3. Análisis 3: Seguridad de uso.....	81
5.2.4. Análisis 4: Manejo de interfaz.....	81
6. CONCLUSIONES Y RECOMENDACIONES.....	82
6.1. Conclusiones.....	82
6.2. Recomendaciones.....	84
REFERENCIAS.....	85
ANEXOS.....	91

ÍNDICE DE FIGURAS

Figura 1. Representación gráfica de Cloud Computing.....	6
Figura 2. Representación de metodología Scrum.	9
Figura 3. Ciclo de Deming.....	10
Figura 4. Representación de modelo de desarrollo tradicional.....	12
Figura 6. Tipos de requerimientos no funcionales.....	21
Figura 7. Diagrama de Arquitectura de aplicación móvil.	31
Figura 8. Esquema Lógico Estándar.	32
Figura 9. Diagrama de Flujo proceso de ingreso de Usuario.	38
Figura 10. Tablas de base de datos utilizadas en la primera historia de usuario. .	39
Figura 11. Registro y validación de los datos del usuario.	39
Figura 12. Visualización y modificación de los datos del usuario.....	40
Figura 13. Pantalla de Logeo.	41
Figura 14. Pantalla de visualización información proporcionada por el usuario. ...	42
Figura 15. Registro de un nuevo parqueadero.....	43
Figura 16. Modificar los datos de un parqueadero existente.....	44
Figura 17. Eliminar un parqueadero existente.....	45
Figura 18. Parqueaderos creados por el usuario.	46
Figura 19. Parqueaderos existentes en la plataforma móvil.....	47
Figura 20. Crear una reserva.	48
Figura 21. Eliminar una reserva.	48
Figura 22. Confirmación de una reserva.	49
Figura 23. Contactar al dueño del parqueadero.....	50
Figura 24. Dashboard que representa mediante gráficos la interacción del usuario con la aplicación móvil.	51
Figura 25. Gráfico de la cantidad de reservas realizadas a los parqueaderos de un usuario.	52
Figura 26. Gráfico de los gastos que tiene un usuario al arrendar parqueaderos de otros.	53

Figura 27. Gráfico de las ganancias obtenidas al alquilar su parqueadero a otros.	54
Figura 28. Ingreso del usuario por primera vez.....	56
Figura 29. Proceso de registro de un usuario.....	57
Figura 30. Almacenamiento de datos en la base de datos.....	57
Figura 31. Proceso para modificación de los datos de un usuario.	58
Figura 32. Información cambiada	59
Figura 33. Cambiar información personal	59
Figura 34. Cambio realizado que se muestra en la base de datos.....	59
Figura 35. Login	60
Figura 36. Información detallada del usuario logeado.....	61
Figura 37. Ingresar un parqueadero.....	62
Figura 38. Proceso para el ingreso de un parqueadero nuevo.....	63
Figura 39. Creación de un parqueadero en la base de datos.....	64
Figura 40. Modificar un parqueadero existente.	65
Figura 41. Modificación del parqueadero en la base de datos.	66
Figura 42. Eliminar un parqueadero existente.....	66
Figura 43. Mostrar los parqueaderos de un usuario.....	67
Figura 44. Buscar parqueaderos.	68
Figura 45. Realizar una reserva	69
Figura 46. Formulario para ingreso de una reserva.	70
Figura 47. Ingreso de los datos de la reserva en la base de datos.	70
Figura 48. Eliminar una reserva	71
Figura 49. Confirmación para eliminar una reserva.....	71
Figura 50. Confirmación de una reserva	72
Figura 51. Acción ante una reserva.....	72
Figura 52. Almacenamiento de la reserva en la base de datos.....	73
Figura 53. Contactar a un usuario.	74
Figura 54. Gráfico de reservas realizadas.....	75
Figura 55. Gráficos de reservas recibidas.	76

Figura 56. Gastos realizados.....	77
Figura 57. Ganancias generadas.	78
Figura 58. Suscripción a Google Play.	79

ÍNDICE DE TABLAS

Tabla 1. Comparación entre la metodología Scrum y modelos de desarrollo tradicionales.	11
Tabla 2. Product Backlog.	33
Tabla 3. Primera Historia de Usuario.	34
Tabla 4. Tabla de Sprints con Historias de Usuario.	36
Tabla 5. Requerimientos no funcionales.	55

1. CAPÍTULO I. INTRODUCCIÓN

1.1. Antecedentes

El mundo presenta cada vez más a una sociedad que se aproxima a la tecnificación para facilitar la resolución de problemas cotidianos con el empleo de la tecnología, el alto uso de aplicaciones móviles es un claro ejemplo de la introducción de mecanismos tecnológicos en el ámbito social ya que, son herramientas que permiten realizar tareas de una manera más sencilla, su desarrollo no presenta mayor dificultad debido a su simpleza en el proceso de implementación y la obtención de grandes cantidades de información que se prestan a facilitar dicho proceso.

En los últimos años el parque automotor ha crecido considerablemente en el país, especialmente en las ciudades más pobladas como el Distrito Metropolitano de Quito, debido al aumento de posibilidades de compras de automóviles, esta situación ha traído consigo un gran problema llamado tráfico.

Gracias al aumento de la circulación de medios de transporte, el tráfico en Quito se ha agravado a tal punto que un conductor promedio pierde aproximadamente 173 horas al año en el tráfico según un estudio realizado por la consultora de transporte Inrix (El Universo, 2019); una de las principales razones por la cual el tráfico se ha intensificado es la gran dificultad de conseguir un parqueadero en la ciudad, especialmente en lugares concurridos y horas pico.

En la ciudad de Quito existen aproximadamente 13200 plazas de parqueadero para darle abastecimiento a la población que posee un automóvil para su desplazamiento dentro de la ciudad.

La ciudad de Quito está habitada por 2'644.145 ocupando el segundo lugar después de la ciudad de Guayaquil, donde “si se calcula que por cada 10 personas el indicador se reduce a 2 autos” tal como lo describe Andrea Medina en El Comercio

en el artículo: En Pichincha hay 2 autos por cada 10 personas desde el 2016, publicado en el 2018 (El comercio, 2018).

Se han desarrollado varias soluciones para la problemática, pero las cuales no han podido ser implementadas por varios inconvenientes, por ejemplo, una aplicación móvil en conjunto con un software automatizado iba a ser implementado en varios parqueaderos municipales de Quito, así se buscarían plazas de parqueo libres en el Centro y Norte de Quito y Cumbayá, logrando abarcar una cantidad limitada de la ciudad. (El comercio, s.f.).

Por otro lado, Parkcero buscaba ser una solución desarrollada en una aplicación móvil para la ubicación de parqueaderos y el alquiler de estos en la ciudad de Bogotá, Colombia, cabe recalcar que dicha solución no ha sido implementada hasta la fecha, pero es referenciada en este documento debido a que la solución planeada tiene una ligera semejanza con la plataforma planteada en el presente trabajo de titulación (Parkcero, 2017).

1.2. Justificación

Buscar un parqueadero en la ciudad de Quito se convierte en una tarea difícil y muchas veces imposible debido a la gran cantidad de vehículos que circulan en las calles, la gran demanda de plazas de parqueo existentes y la falta de mecanismos para conocer plazas de parqueaderos disponibles; lo que significa que una persona desperdicia una gran cantidad de tiempo en la movilización y búsqueda de un parqueadero.

Por esta razón este proyecto de titulación busca disminuir el tiempo que el usuario emplea en encontrar una plaza de parqueo de forma sencilla y rápida tomando como factor importante la cercanía del lugar de destino del usuario.

Su desarrollo se realizará en el entorno Android Estudio, ya que permite aprovechar al máximo los beneficios del software como la visualización de soluciones en vivo,

entorno robusto, emuladores que permiten mostrar las actividades en una gran cantidad de sistemas operativos, dispositivos y plantillas que son utilizadas para el desarrollo de diferentes elementos, esto se suma a su facilidad y confiabilidad de uso y pruebas.

Está enfocado no solo para los habitantes de la ciudad de Quito, sino también hacia los miembros de la comunidad de la Universidad de las Américas, tanto estudiantes, docentes y personal administrativo, ya que el problema de la cantidad de plazas de parqueo también se ve reflejado en los parqueaderos de los distintos campus de la Universidad.

Muchas veces se ve una ligera escases de parqueaderos en horas específicas como en la mañana y la tarde y en días en los que hay una gran afluencia de estudiantes como en días de exámenes, por lo cual, el hecho de que se tenga una alternativa para la solución de este problema es de gran ayuda para toda la comunidad de la Universidad de las Américas, así se logrará evitar problemas relacionados con retrasos o con el parqueo en lugares poco confiables para la persona.

Con el proyecto de titulación se busca que el usuario genere confianza por la plataforma móvil, ya que esta protege la información manejada por medio de la implementación de medidas de seguridad de la información.

1.3. Alcance

El presente proyecto de titulación busca realizar una plataforma móvil para dispositivos que emplean el sistema operativo Android para mostrar la oferta y demanda de plazas de parqueo en el Distrito Metropolitano de Quito, la plataforma móvil será desarrollada con el lenguaje de programación Java en conjunto con Json y MySQL en el entorno de desarrollo Android Studio para el correcto almacenamiento de la información en la nube.

La aplicación móvil se centra en la resolución de tres aspectos importantes. El primer aspecto abarca las actividades realizadas por usuarios que desean obtener una plaza de parqueo en donde la plataforma móvil ubicará los parqueaderos más cercanos al destino a donde el usuario desea movilizarse tomando en cuenta las condiciones de su medio de transporte, tiempo, costos y disponibilidad.

El segundo aspecto se refiere a las actividades que pueden realizar los usuarios, los cuales poseen parqueaderos, la plataforma maneja la disponibilidad, el tiempo, cobros y ganancias generadas por medio del control de reservas.

El tercer aspecto abarca las acciones realizadas en relación con usuarios que poseen parqueaderos particulares y debido a la falta de uso desean ponerlo en arriendo, la plataforma móvil contactará al dueño de la plaza de parqueo con el usuario que desea usarlo, así se manejan las cuestiones de lugar, tiempo de parqueo, cobros, etc.

Con la plataforma móvil es posible obtener análisis de la información dada por los usuarios por medio de *dashboards* los cuales muestran ganancias, gastos, etc., para que los usuarios tengan conocimiento de su progreso en el uso de la aplicación móvil.

1.4. Objetivo General

Implementar un prototipo de aplicación móvil para la oferta y demanda de parqueaderos ubicados geográficamente dentro del Distrito Metropolitano de Quito.

1.5. Objetivos Específicos

- Determinar los requerimientos funcionales y no funcionales del usuario final para solucionar la problemática planteada.

- Desarrollar el prototipo mediante la implementación de características mínimas de seguridad de información, para el fácil manejo de la plataforma por parte de los usuarios.
- Verificar el correcto funcionamiento del prototipo mediante pruebas de campo.

2. CAPÍTULO II. MARCO TEÓRICO

En este capítulo se detallan los conceptos más importantes que fueron empleados como base para el desarrollo del trabajo de titulación.

2.1. Conceptos

En las siguientes literales se definen una serie de temas como la computación en la nube, metodología para la recolección de información, herramientas de desarrollo y lenguaje de programación, los cuales son relevantes para llevar a cabo el desarrollo de la aplicación móvil.

2.1.1. *Cloud Computing*

Cloud abarca una gran cantidad de conceptos relacionados con servicios y aplicaciones de ámbito informático que utilizan Internet como el principal medio (Luna, 2014, p.3), esto se puede visualizar en la figura 1, en el cual se detallan los modelos de servicios, los cuales se dividen en: software como servicio (SaaS), plataforma como servicio (PaaS) e infraestructura como servicio (IaaS), éstos se desarrollan más adelante.



Figura 1. Representación gráfica de Cloud Computing.

Tomado de INCIBE, 2017.

Cloud Computing es un modelo para permitir el acceso adecuado, conveniente y bajo demanda a una red compartida con un conjunto de recursos informáticos configurables (por ejemplo, redes, servidores, almacenamiento, aplicaciones y servicios) que se puede proveer y publicar rápidamente con un mínimo esfuerzo de gestión o interacción por parte del proveedor de servicios. (NIST, 2011, p.2).

El Instituto Nacional de Estándares y Tecnologías estadounidense (NIST) define las siguientes características de Cloud Computing:

- En demanda – autoservicio: Un consumidor puede proporcionar de manera unilateral servicios informáticos, como uso de servidores, almacenamiento en red, etc., de forma automática según sea necesario sin la necesidad de la intervención humana con cada proveedor de servicios.
- Amplio acceso a la red: Los servicios están disponibles a través de la red y se accede a estos mediante mecanismos estándar que promueven el uso de plataformas heterogéneas como teléfonos móviles, tablets, computadoras portátiles y estaciones de trabajo.

- Agrupación de recursos: Los recursos informáticos del proveedor se agrupan para proporcionarlos a varios clientes utilizando un modelo de múltiples inquilinos con el que adquieren diferentes recursos tanto físicos como virtuales asignados de forma dinámica de acuerdo con la demanda del consumidor.
- Rápida escalabilidad: Los servicios pueden proveerse de tal forma que puedan escalar rápidamente de acuerdo con la demanda.
- Servicio medido: Los sistemas alojados en la nube son capaces de controlar y optimizar de forma automática el uso de recursos con la capacidad de medición en los niveles de abstracción apropiados para cada servicio.

De igual manera el NIST presenta las siguientes definiciones sobre los modelos de servicios:

- Software como servicio (SaaS): Capacidad que se brinda al cliente de usar las aplicaciones del proveedor que se encuentran en una infraestructura en la nube.
- Plataforma como servicio (PaaS): Capacidad proporcionada al cliente de implementar en la infraestructura de la nube aplicaciones creadas o adquiridas usando lenguajes de programación, bibliotecas, servicios y herramientas compatibles con el proveedor.
- Infraestructura como servicio (IaaS): Capacidad que se brinda al cliente para proporcionar procesamiento, almacenamiento, redes y otros recursos informáticos fundamentales para la implementación de software.

2.2. Metodología empleada

En el siguiente ítem se expondrá de forma detallada la metodología utilizada en el proyecto, con la finalidad de obtener el mejor resultado posible en el proyecto realizado.

2.2.1. Scrum

Scrum es una metodología basada en conceptos de control efectivo para la administración de sistemas complejos y cambiantes, sus tres pilares definidos por Galiano (2016, p.21) son los siguientes:

- **Transparencia:** Todos los involucrados en el proceso deben conocer los puntos más importantes del mismo, por lo tanto, deben ser expuestos siguiendo un estándar para que todos los individuos tengan la misma captación del objetivo.
- **Inspección:** Es importante realizar una evaluación continua de las actividades realizadas a través del proceso con la finalidad de identificar posibles cambios de manera ágil.
- **Adaptación:** Una vez identificado un cambio se deben realizar las acciones pertinentes para poder corregirlo o reestructurar el objetivo central.

Esta metodología fija un conjunto de acciones que deben ser realizadas para la correcta implementación de esta metodología:

- Pila de Productos
- Reunión Planificación del Sprint
- Pila de Sprint
- Revisión del Sprint

- Retrospectiva del Sprint

El concepto Sprint usado en la definición anterior hace referencia a un conjunto de eventos en un proyecto.

Una de las principales ventajas de la metodología Scrum es la flexibilidad para manejar tanto objetivos individuales como corporativos, también impulsa a la creación de culturas basadas en el rendimiento óptimo y la buena comunicación (Socconini & Reato, 2019, p.29).

En la figura 2 se puede visualizar de forma más detallada en que se basa la metodología Scrum.

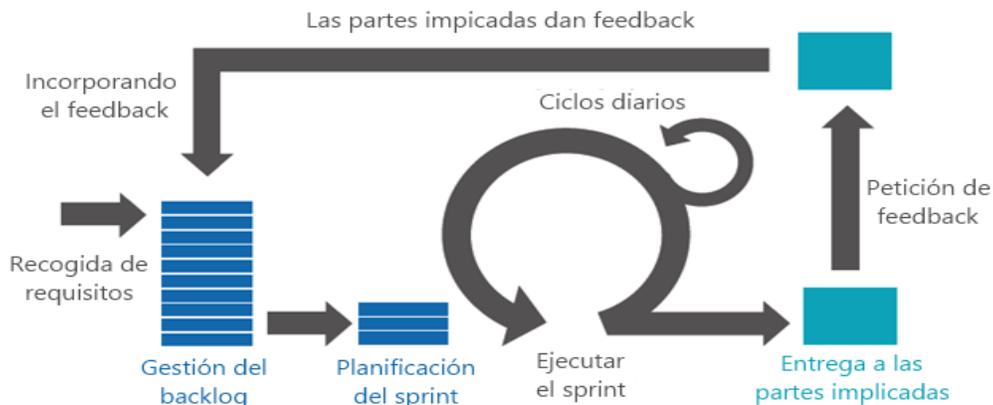


Figura 2. Representación de metodología Scrum.

Tomado de Leal, 2017

Scrum utiliza el ciclo de Deming, el cual consiste en realizar procesos de mejora continua enfocándose en la calidad del producto a realizar, este consta de cuatro fases las cuales se encuentran detalladas a continuación y se pueden visualizar en la figura 3 de este documento:

- Planificar: En esta fase se definen las actividades a realizar para el proceso y como deben ser realizadas cada una de estas acciones.

- Hacer: Esta fase corresponde a la realización de las actividades planteadas en la fase anterior.
- Revisar: En la tercera fase se comprueba si se ha podido realizar las actividades correspondientes de la manera planificada.
- Realizar: En la última fase se define la manera en que las actividades pueden ser realizadas de mejor manera en el siguiente ciclo.



Figura 3. Ciclo de Deming.

Adaptado de Roser, 2016.

Scrum propone una serie de valores centrados en una comunicación eficaz durante el proceso de desarrollo:

- Compromiso: Este valor define que cada miembro del equipo debe comprometerse en el proceso de desarrollo.

- Enfoque: En caso de que exista un problema en el proceso de desarrollo este debe ser resuelto dividiéndolo en pequeñas partes que puedan ser manejables por el equipo.
- Organización abierta: El equipo debe estar abierto a una correcta comunicación de las actividades y experiencias que se obtienen durante el periodo de desarrollo.
- Respeto: Este valor es fundamental ya que garantiza que el trabajo de todos los miembros del equipo sea reconocido de manera correcta.
- Coraje: Todos los miembros del equipo deben afrontar cada reto que se presenta en el proceso de desarrollo.

La metodología Scrum presenta una gran cantidad de ventajas sobre los métodos tradicionales de desarrollo, algunas de estas ventajas se encuentran detalladas a continuación en la tabla 1:

Tabla 1. Comparación entre la metodología Scrum y modelos de desarrollo tradicionales.

Metodologías Tradicionales	Metodología Scrum
Modelo predictivo	Modelo adaptativo
Fases en serie	Fases simultáneas
Jerarquía	Autogestión
Objetivos completos	Entregas incrementales

Adaptado de Galiano, 2016, p.28.

La metodología tradicional sigue cada etapa de manera secuencial, los pasos que se deben seguir se pueden visualizar en la figura 4.

En la etapa de requisitos se definen los requerimientos necesarios para el desarrollo del proyecto, la etapa de diseño se encarga de la definición del modelo del proyecto,

la etapa de implementación es del desarrollo del proyecto propiamente dicho, la etapa de verificación corresponde a pruebas y la etapa de mantenimiento se encarga de las mejoras necesarias.

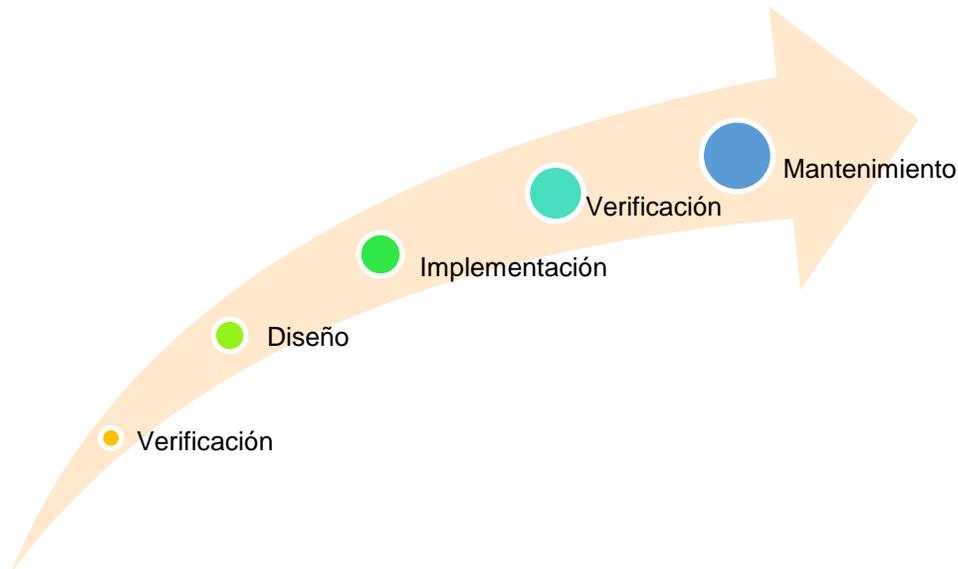


Figura 4. Representación de modelo de desarrollo tradicional.

Adaptado de Openclassrooms, 2017.

2.3. Herramientas de Desarrollo

Las herramientas de desarrollo son mecanismos que se enfocan en el ciclo de vida para el desarrollo de tareas de un proyecto de software, éstas son utilizadas para llevar a cabo el prototipo móvil, los cuales se enfocan principalmente en el modelaje y creación de sistemas informáticos, que poseen diferentes etapas tomando en cuenta como guía las tecnologías planteadas en los puntos anteriores.

2.3.1. Android Studio

Friesen (2019) manifiesta que Android Studio es un entorno de desarrollo para aplicaciones que se emplean en el sistema operativo Android, se caracteriza por la

simplificación del proceso de implementación y su kit de herramientas está basado en el IDE IntelliJ IDEA. Éste es completamente intuitivo y de código abierto, está desarrollado en Java y puede ser utilizado en cualquier plataforma.

En el artículo "How To Developers: Android Studio: A Platform for Android Development" (2015) se definen algunas propiedades únicas de este entorno de desarrollo:

- Existen ejemplos de código alojados en GitHub para facilitar la búsqueda de soluciones de implementaciones de software.
- Cuenta con un conjunto de plantillas que permiten crear y administrar actividades de manera sencilla.
- El editor cuenta con la funcionalidad de agregar componentes fácilmente y es posible realizar previsualizaciones del diseño de una implementación de software.

Hohensee (2014) expresa que las piezas claves de Android Studio es su editor de código y sus elementos como *Smart Editing* para obtener código más legible y su sistema de construcción basado en Gradle, este permite realizar configuraciones sobre código para producir distintas versiones de este.

Android Studio presenta una serie de requerimientos mínimos para ser usado en varias plataformas, en el caso del sistema operativo Windows los requisitos para la última versión del entorno de desarrollo son los siguientes ("Download Android Studio and SDK tools | Android Developers", s.f.):

- Sistema operativo Microsoft Word 7/8/10 de 32 o 64 bits, se debe tener en cuenta que los Emuladores de Android solo pueden ser soportados por sistemas operativos de 64 bits.

- 4 GB de memoria RAM como requerimiento mínimo, aunque se recomienda tener 8 GB.
- 2GB de almacenamiento interno, aunque se recomienda tener 4GB.
- Resolución de pantalla mínima de 1280 x 800.

De igual forma Android Studio presenta una serie de características que los vuelven un entorno de desarrollo robusto:

- Cuenta con un Analizador APK el cual permite analizar el archivo APK incluso si el proyecto no fue creado directamente en Android Studio, este análisis permite reducir el código de una aplicación.
- Con Android Studio es posible instalar emuladores los cuales manejan aplicaciones de una forma más rápida que un dispositivo físico, esto es usado para realizar pruebas de manera más eficiente.
- El entorno de desarrollo presenta un editor de código inteligente para los lenguajes de programación Java, Kotlin, C y C++, el cual realiza sugerencias de código y permite desarrollar de forma más rápida y productiva.

2.4. Sistema Operativo

A continuación, se detalla información relevante sobre el sistema operativo utilizado en la solución planteada, los cuales son programas de software que sirven esencialmente para administrar recursos y coordinar tareas que desempeña el hardware.

2.4.1. Android S.O

“Android es un sistema operativo móvil que se basa en una versión modificada de Linux. Fue desarrollado originalmente por un startup del mismo nombre, Android, Inc.” Lee y DiMarzio (2016, p.2).

2.4.1.1. Características

- Almacenamiento: El sistema operativo utiliza SQLite, una base de datos relacional caracterizada por su estructura ligera.
- Conectividad: Este sistema trabaja con las tecnologías GSM/EDGE, IDEN, CDMA, Bluetooth, Wi-Fi, WiMAX, etc.
- Mensajería: Android usa SMS y MMS.
- Soporte de medios: El sistema operativo utiliza los medios H.263, H.624, MPEG-4 SP, MP3, WAV, JPEG, GIF, PNG, BMP, entre otros.
- Soporte de Hardware: Los componentes de hardware que puede sostener Android son el sensor acelerómetro, cámara, brújula, pantallas multitáctiles, GPS, conexiones a internet mediante medios inalámbricos, cableados, etc.

2.4.1.2. Arquitectura

- Kernel de Linux: El sistema operativo Android se basa en el kernel de Linux, aquí se encuentran alojados los controladores de bajo nivel que trabajan con los componentes de hardware de los diferentes dispositivos.
- Bibliotecas: Las bibliotecas almacenan las características necesarias para el sistema operativo.

- **Tiempo de ejecución de Android:** El tiempo de ejecución provee una variedad de bibliotecas que permiten que aplicaciones de Android puedan ser creadas por desarrolladores utilizando el lenguaje de programación Java. El sistema operativo también contiene a Dalvik, una máquina virtual que permite que las aplicaciones sean ejecutadas en su propio proceso e instancia.
- **Framework de aplicación:** Muestra las diferentes capacidades del sistema operativo para que los desarrolladores puedan aprovecharlas e incluirlas en aplicaciones.

2.5. Lenguaje de Programación

Las herramientas de programación son programas que poseen una sintaxis determinada formada por diversas instrucciones, éstas son utilizadas para el desarrollo de plataformas, con el objetivo de lograr una solución óptima.

2.5.1. Java

Es un lenguaje de programación nacido en los años 90 por Sun Microsystems, como lenguaje orientado a objetos, de fácil uso, potente y muy bien adaptado para la programación de aplicaciones en red. (Francisco Durán, 2007).

2.5.1.1. Componentes

- **Clases:** Describen la categoría de un objeto, la cual puede ser instanciada a través de los objetos. Las clases contienen campos y métodos.
- **Campos:** Son variables de datos que pertenece a una clase.
- **Métodos:** Son las características propias de una clase y sirven para la comunicación entre objetos.
- **Parámetros:** Pertenecen a los métodos y le dan valor agregado.

- Atributos: Son los estados de un objeto.
- Objetos: Son instancias de una clase.

2.5.2. eXtensible Markup Language - XML

XML, es un lenguaje que nació en los años 90 apoyado por W3C, con el objetivo de crear un lenguaje de marcas para Internet, éste tenía el reto de combinar el HTML con SGML.

Propone un formato con el cual se entrelazan marcas de textos para representar información estructurada en Internet, para que pueda ser procesada y visualizada por distintos tipos de aplicaciones y dispositivos.

2.5.3. JavaScript Object Notation – JSON

Es un formato utilizado para el intercambio de datos ligeros basado en texto plano el cual es legible para los usuarios y fácil para el debido análisis de las computadoras.

No contempla espacios de nombres, no se puede extender y no tiene validador, este formato es utilizado en la mayoría de los lenguajes de programación de hoy en día.

El JSON posee dos tipos de estructuras básicas las cuales dependen de los siguientes tipos de datos:

- String: Es un grupo de caracteres Unicode por defecto UTF-8, los cuales son utilizados mediante comillas.
- Number: Tipo de dato utilizado para representar números, se utiliza la misma sintaxis que se emplea en lenguaje C y Java.

- Value: Tipo de dato que puede definirse como un string, number, array, true, false, etc.
- Object: Grupo no ordenado de variables con sus valores.
- Array: Valores ordenados, los cuales están delimitados por corchetes.

2.6. Almacenamiento

En esta sección se detalla todo lo relacionado al proceso de almacenamiento de los datos para la aplicación móvil.

2.6.1. MySQL

“MySQL es un sistema de gestor de bases de datos (SGBD, DBMS) muy conocido y ampliamente usado por su simplicidad y notable rendimiento (...)” (Casillas, Gibert Pérez, s.f).

Este sistema ha ganado gran popularidad por tener varias características favorables para la manipulación del usuario final, las características más relevantes son:

- Es multiplataforma.
- Está desarrollado en el lenguaje de programación C/C++.
- Tiene alta velocidad de respuesta.
- Soporta varios métodos de almacenamiento de tablas.
- Tiene una gran biblioteca de tipos de datos
- Su administración se basa en usuarios y los privilegios que se les otorguen a éstos.

2.6.2. Encriptación SHA

La encriptación es la forma en la que los datos se cifran con clave con el objetivo de aumentar la seguridad de un sistema.

El Instituto Nacional de Estándares y Tecnologías estadounidense, NIST, lanzó un algoritmo basado en técnicas similares a MD5, en 1993 el NIST lo nombró oficialmente SHA, Secure Hash Algorithm, aunque hoy en día es conocido como SHA-0 debido a la existencia de sucesores del algoritmo.

La NSA cambió este algoritmo por una versión mejorada cuya seguridad se considera mejor que MD5 ya que el código hash de este tiene 160 bits para manejar mensajes de hasta 264 bs.

2.7. Herramientas de trabajo y de análisis

En la sección de a continuación se detallarán las herramientas de trabajo utilizadas, como los requerimientos funcionales y no funcionales y de la misma manera el análisis previo realizado para poder diseñar la solución planteada.

2.7.1. *Dashboard*

Un panel de instrumentos, mejor conocido por su término en inglés, Dashboard, es un conjunto de gráficos que muestran valores importantes de manera interactiva para ser usados en la toma de decisiones de un objetivo determinado. En este documento se utiliza el término *Dashboard* para hacer referencia al concepto antes propuesto.

El objetivo de esta herramienta es mostrar valores de factores claves de ámbitos en donde es necesario tomar decisiones, ilustrando correctamente la decisión final a tomar dependiendo de objetivos propuestos.

2.7.2. Requerimientos

Los requerimientos son todas las actividades que están relacionadas a obtener y definir las necesidades de un usuario para el desarrollo de una solución tecnológica.

2.7.2.1. Requerimientos funcionales

“Son declaraciones de los servicios que proveerá el sistema, de manera en que éste reaccionará en situaciones particulares” (Sommerville, 2005). Los servicios y/o funciones solicitados previamente por el usuario deben estar bien definidos y no deben tener definiciones que sean contradictorias entre sí.

2.7.2.2. Requerimientos no funcionales

“Son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo, estándares, etc.” (Sommerville, 2005).

Este tipo de requerimientos hacen referencia a las propiedades que el sistema debe tener en cuanto a fiabilidad, respuesta oportuna en base al tiempo y la capacidad de almacenamiento, éstos se obtienen gracias a las necesidades del cliente.

Los requerimientos no funcionales tienen la siguiente clasificación, la cual se puede observar en la figura 6 del presente documento.

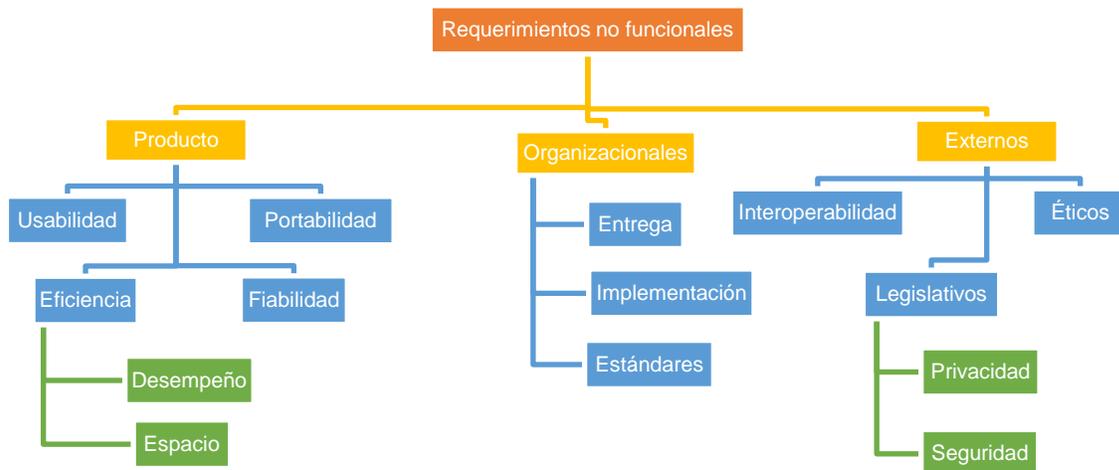


Figura 5. Tipos de requerimientos no funcionales.

Adaptado de Sommerville I, 2005.

2.7.3. Modelo entidad relación

Es un modelo de datos que se basa en la percepción de lo real, el cual consta de un conjunto de objetos, también llamados entidades y las relaciones que existen entre ellos. Éste nace con el fin de facilitar y disminuir el tiempo empleado en el diseño de las bases de datos, es importante saber que al utilizar este modelo para el diseño y dimensionamiento de una base de datos se debe tomar en cuenta y elegir de manera estratégica una “Llave primaria”, lo cual sirve para hacer distinciones entre una entidad que conforma un conjunto de entidades.

El diagrama Entidad – Relación es una técnica utilizada en Ingeniería de la Información para crear modelos de datos de alta calidad, éste define los datos y las relaciones de éstos de manera estandarizada para cualquier tipo de sistema de información (Barker, 1994).

2.7.3.1. Entidad

Las entidades son objetos que tienen características propias las cuales hacen que se diferencien de los otros, ésta puede ser concreta o abstracta, y están definidas por un conjunto de atributos los cuales poseen valores predeterminados, llamados dominios.

2.7.3.2. Relaciones

Las relaciones se dan entre varias entidades, éstas están definidas en base a una cardinalidad de mapeo, es decir, que expresan el número de entidades con las cuales se pueden asociar unas con otras, y se dividen en las siguientes:

- Una a una: Las entidades del conjunto A solo pueden estar asociadas a una entidad del conjunto B, y las entidades del conjunto B solo pueden estar asociadas a un único valor de las entidades del conjunto A.
- Una a muchas: Las entidades del conjunto A pueden estar asociadas a 1 o más de una entidad del conjunto B, sin embargo, las entidades del conjunto B solo pueden estar asociadas a una única entidad del conjunto A.
- Muchas a una: Las entidades del conjunto A solo están asociadas a una única entidad del conjunto B, pero las entidades del conjunto B pueden estar asociadas a 1 o más de una entidad del conjunto A.
- Muchas a muchas: Las entidades tanto del conjunto A como del conjunto B están asociadas entre sí a más de una.

3. CAPÍTULO III. ANÁLISIS DE REQUERIMIENTOS PARA LA SOLUCIÓN DEL PROBLEMA

En este capítulo se detalla la información recolectada como antecedentes del problema planteado e información dada por los usuarios encuestados para poder

observar la factibilidad de la solución propuesta, con los cuales se definen los requerimientos funcionales y no funcionales con el objetivo de definir los parámetros más importantes que contendrá la aplicación.

De la misma manera se define su arquitectura y estructura, el *Product Backlog* que contendrá las acciones que debe realizar la aplicación móvil, junto con todas las historias de usuario que detallarán de qué forma serán desarrolladas dichas acciones y los sprints correspondientes para cada una de las historias de usuario.

3.1. Recolección de Datos

Para poder obtener la información pertinente y necesaria sobre la problemática planteada se ha realizado una investigación profunda acerca de los parqueaderos existentes actualmente en el Distrito Metropolitano de Quito.

Según la Agencia Nacional de Tránsito - AMT en el año 2017, 26 863 autos entraron como nuevos a las calles y en entre el mes de enero del mismo año y el 2016 ya se encontraban 32 431 autos en las calles de la capital, es decir, el aumento de autos en ese periodo fue del 20,7%, equivalente a 5 568 automóviles. (Orozco, 2017).

Una de las razones por la cual se ha visto un aumento en el parque automotor es la gran participación que tiene la ciudad en la venta de vehículos, especialmente en vehículos livianos ya que un estudio realizado en el año 2018 muestra que el 38% del mercado automotriz se centra en el Distrito Metropolitano de Quito. (Tecniseguros, 2019).

Muchas personas consideran que la razón por la cual el mercado automotriz es tan grande en la ciudad es la medida Pico y Placa impuesta en el año 2010, ya que, debido a la imposibilidad de los ciudadanos de desplazarse con sus propios vehículos se han dado a la obligación de buscar otras alternativas para movilizarse, una de ellas es adquirir otro automóvil.

Los antecedentes antes expuestos sirvieron para concluir que en la provincia de Pichincha existen al menos 2 autos por cada 10 personas, y para el 2018 parque automotor aumentó en un 128% llegando la cantidad de vehículos a 270 024. (Orozco, 2018).

En el Distrito Metropolitano de Quito existe una cantidad limitada de parqueaderos, un ejemplo de esto es la red de estacionamientos municipales de la ciudad, la cual cuenta con solo 2.130 plazas que se ubican de manera estratégica en lugares concurridos de la ciudad. (Conmicelu, 2018).

Debido a que la oferta de parqueaderos en la ciudad de Quito es muy escasa en comparación a la demanda de automóviles, la mayoría de las veces los parqueaderos de lugares como restaurantes, cines y oficinas que son muy concurridos, solo tienen una cantidad limitada de plazas, por lo que para conseguir una plaza por estos lugares es complicado, las personas deben parquearse en zonas alejadas de su lugar de destino. (El Telégrafo - Noticias del Ecuador y del mundo, 2018).

La búsqueda de parqueaderos hace que las personas pierdan tiempo mientras están en las calles debido al tráfico, ya que, éste es un serio problema en la ciudad de Quito. Un estudio realizado en el año 2019 determinó que un ciudadano promedio en el Distrito Metropolitano de Quito puede perder hasta 173 horas al año en el tráfico.

Existen muchas razones por las cuales se genera tráfico en la ciudad, una de ellas es la falta de comunicación a la hora de hacer obras que impiden el tránsito, otra razón es la falta de diferenciación horaria entre el sector educativo y empresarial, ya que, esto genera una gran cantidad de tráfico en horas pico, esto sumado a la exhaustiva búsqueda de parqueaderos genera tráfico en muchos sectores concurridos de la ciudad.

Para recabar más información sobre el tema, se realizó una encuesta compuesta por 10 preguntas; realizada en el mes de octubre del 2019 a 48 personas, enfocada a conocer los gustos y preferencias del mercado objetivo respecto al servicio a ofertarse, todos residentes de la ciudad de Quito, mayores de 18 años que tenían al menos un auto y/o que posean un parqueadero libre para su oferta. Para la obtención de los resultados de la encuesta no probabilística con muestreo por conveniencia fueron enviadas por canales electrónicos. La misma se puede ver en el Anexo 1.

3.2. Procesamiento de Información

Actualmente en la ciudad de Quito se ha visto un aumento considerable de la cantidad de autos que circulan día a día, lo cual causa una terrible situación ya que el número de parqueaderos en relación a la cantidad de autos que circulan diariamente es limitado, en especial en lugares concurridos como el Centro Histórico, en el cual existen varios parqueaderos que tienen gran afluencia en los fines de semana y la demanda de los mismos es exorbitante, por tal razón, en éste como en muchos otros lugares de la ciudad se genera una gran cantidad de tráfico, generando estrés en los ciudadanos y de la misma manera retrasos en el tiempo de llegada sus lugares de destino.

Debido a que la oferta de parqueaderos es limitada muchas personas se ven obligadas a buscar plazas de parqueos lejos de sus lugares de destino, generando inconformidad e incertidumbre. Muchos ciudadanos incluso no logran encontrar plazas de parqueos, viéndose obligados a dejar sus automóviles en lugares completamente alejados de sus destinos que para llegar a ellos deben tomar un medio de transporte público.

Todos estos inconvenientes detallados anteriormente deben ser solventado de tal manera que un ciudadano pueda dirigirse a un destino teniendo conocimiento de la disponibilidad o la existencia de plazas de parqueaderos cerca del mismo, para no

generar tiempo innecesario mientras se dirige a su lugar de destino, a medida que esto es solventado el ciudadano ya no se ve en la necesidad de pasar grandes cantidades de tiempo en el tráfico ya que se sabe exactamente al lugar en donde se encuentra una plaza de parqueo.

Como resultado de las encuestas respondidas se obtuvo lo siguiente:

1. ¿Con cuál sistema operativo se familiariza más al utilizar un dispositivo inteligente?

El 79.2% de los encuestados respondieron que el sistema operativo con el que más se familiariza el Android, el 18.8% respondieron que es iOS y el 2% restante respondieron que es Windows Phone.

2. ¿Estaría dispuesto a utilizar una aplicación desarrollada por estudiantes?

La encuesta muestra que el 89.6% sí estaría dispuesto a utilizar una aplicación desarrollada por estudiantes, el 6.3% no lo estaría y el 4.2% tal vez lo estaría.

3. ¿Qué tipo de dispositivo utiliza más?

El 97.9% utiliza más un dispositivo Smartphone, mientras que el 2.1% restante utiliza más Tablet.

4. ¿Estaría dispuesto a descargar una aplicación móvil que le facilita la búsqueda de parqueaderos?

Según la encuesta el 97.9% estaría dispuesto a descargar la encuesta, y el 2.1% tal vez lo estaría.

5. ¿Es difícil para usted encontrar un parqueadero en la ciudad de Quito?

El 77.1% piensa que es difícil encontrar un parqueadero en la ciudad de Quito, el 18.8% piensa que tal vez es difícil y el 3.5 piensa que no es difícil

6. ¿Diariamente por cuánto tiempo utiliza un puesto de parqueadero?

Según la encuesta realizada el 62.5% utiliza un puesto de parqueadero de 1 a 3 horas, el 20.8% utiliza de 4 a 6 horas y el 16.7% de 6 o más horas.

7. ¿Qué tipo de parqueadero utiliza usualmente?

Para el 50% utilizan usualmente parqueaderos públicos, el 33.3% utiliza parqueaderos privados (de terceros) y el 16.7% utiliza parqueaderos propios.

8. ¿Cuánto tiempo gasta a diario ubicando un lugar de parqueadero?

El 39.6% pasa de 10 a 15 minutos ubicando un lugar de parqueadero, el 33.3% pasa más de 15 minutos y el 27.1% pasa de 5 a 7 minutos.

9. ¿Le resulta molesto arrendar su parqueadero a extraños?

Según el 54.2% no les resultaría molesto, el 35.4% sí les resultaría molesto y el 10.4% tal vez les resultaría molesto.

10. ¿No encontrar un puesto de parqueadero cuánto estrés le genera?

Los resultados obtenidos reflejan que el 41.7% de los encuestados le genera mucho estrés, al 27.1% le genera estrés, al 20.08% le genera poco de estrés, y al 10.4% le genera algo de estrés.

Los gráficos de la encuesta se pueden visualizar en el Anexo 2 del presente documento.

3.3. Resultado del Análisis

Con la información obtenida se tomó la decisión de realizar un producto que sea capaz de resolver la problemática planteada, de manera que el usuario pase menos tiempo en el tráfico y buscando una plaza de parqueo para llegar a su destino, dando

una respuesta directa para que el usuario no se vea obligado a tomar otras alternativas debido a la incapacidad de encontrar un parqueadero en la ciudad.

La solución debe dar respuesta inmediata y el producto debe ofrecerse de tal modo que sea accesible para la mayoría de los ciudadanos que estén en el Distrito Metropolitano de Quito, por lo que se realizará una aplicación móvil la cual estará al alcance de la mayoría de los usuarios y ofreciendo una forma intuitiva y rápida de resolver el problema de la búsqueda de parqueaderos en la ciudad.

Para poder solventar el inconveniente que se vive día a día en la ciudad de Quito, se busca ofertar una aplicación desarrollada en Android Studio, la cual podrá ser descargada mediante Google Play.

Esta aplicación tendrá el objetivo de ser lo más intuitiva posible, la cual tendrá una interfaz de usuario muy amigable, con el fin de reducir el tiempo perdido en el tráfico de los usuarios, concentrándose en brindarle bienestar y seguridad en el desarrollo transaccional, es decir estará limitada a poner en riesgo sus pertenencias.

Con ésta se podrá de la misma manera poder evaluar su desempeño y poder sacar estadísticas para así poder brindar mejoras cuando el sistema lo requiera, siempre pensando en la satisfacción del usuario final. Estará centrada en el posicionamiento global de los costos de parqueaderos, para que el cliente tenga la capacidad de pagar un precio justo por el servicio brindado, el cual está en ofertar un parqueadero o arrendarlo.

Para que el usuario pueda utilizar las funcionalidades de la aplicación es necesario que tenga compatibilidad con Android 3.5, versión en la cual estará desarrollada la misma.

Se utilizará *Runhosting* es una plataforma en la nube, utilizada para el almacenamiento de información, la cual es de software libre y de código abierto. De la misma manera para el diseño de las bases de datos será empleado MySQL, el

cual tiene alta compatibilidad con *Runhosting*, y de la misma manera es un lenguaje de fácil manejo y entendimiento.

La aplicación debe ser eficiente, la cual tenga una conexión de GPS que dependerá del dispositivo integrado en cada uno de los *Smartphone*, de la misma manera debe contar con conexión a internet para poder utilizar la aplicación y poder realizar las transacciones que desee.

3.4. Enfoques de seguridad

Se basa en tres pilares importantes que todo sistema debe tener, con el objetivo de brindar una protección suficiente de la información almacenada y/o utilizada en el sistema.

3.4.1. Confidencialidad

Asegurar la confidencialidad de un sistema consiste en evitar que personas y sistemas no autorizados accedan a la información.

3.4.2. Integridad

Es una característica de la seguridad de la información la cual consiste en que la información no sea modificada, alterada o eliminada por usuarios que no tengan autorización para realizarlo.

3.4.3. Disponibilidad

Este ítem hace referencia a que la información sea accesible 24/7, para que pueda ser utilizada por el usuario autorizado cuando éste necesite.

4. CAPÍTULO IV. DESARROLLO DEL PROTOTIPO CON PARÁMETROS DE SEGURIDAD

4.1. Diseño

El proceso de diseño consiste en la recopilación de datos la cual es procesada y analizada con el objetivo de crear y plasmar una solución viable en un modelo de software que permita satisfacer las acciones de dicha solución, para esto se utilizaron los diagramas de Arquitectura y Entidad-Relación, junto con el Esquema Lógico Estándar, los cuales son herramientas propias para este tipo de diseños.

4.1.1. Diagrama de Arquitectura

El diagrama de arquitectura es un esquema que permite obtener una visión gráfica y simplificada lo que va a realizar una aplicación, este diagrama es utilizado para entender la funcionalidad general de un sistema o aplicación.

El diagrama de arquitectura indicado en la figura 7 describe el funcionamiento de la aplicación móvil, la cual es desarrollada en Android Studio y el usuario puede utilizarla por medio de un dispositivo móvil, mismo que se conecta a un servidor web alojado en la nube que tiene una base de datos MySQL.

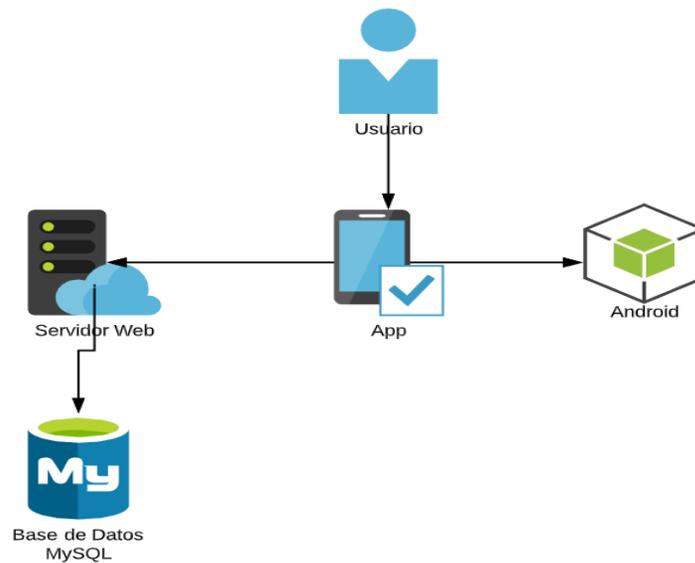


Figura 6. Diagrama de Arquitectura de aplicación móvil.

4.1.2. Diagrama Entidad – Relación

El diagrama Entidad – Relación de la aplicación móvil que se visualiza en el Anexo 3 presenta las entidades utilizadas por la misma; la entidad Persona, la cual almacena la información del cliente; la entidad Usuario, que almacena los datos principales para el acceso a las distintas funciones de la aplicación móvil; la entidad Parqueadero almacena la información general de los parqueaderos y la entidad Reserva, la cual maneja el proceso de reservas de plazas de parqueo. Todas las entidades se presentan junto con sus respectivos atributos y relaciones.

4.1.3. Esquema Lógico Estándar

El esquema lógico estándar es un modelo de representación de datos que despliega los componentes que tiene una base de datos de un tipo específico, sin considerar el gestor de base de datos a usar.

En la figura 8 se muestra el esquema lógico estándar de la base de datos de la aplicación móvil, las tablas representan las entidades mencionadas en el punto anterior, adicional a esto se suma los campos que le pertenecen a cada tabla.

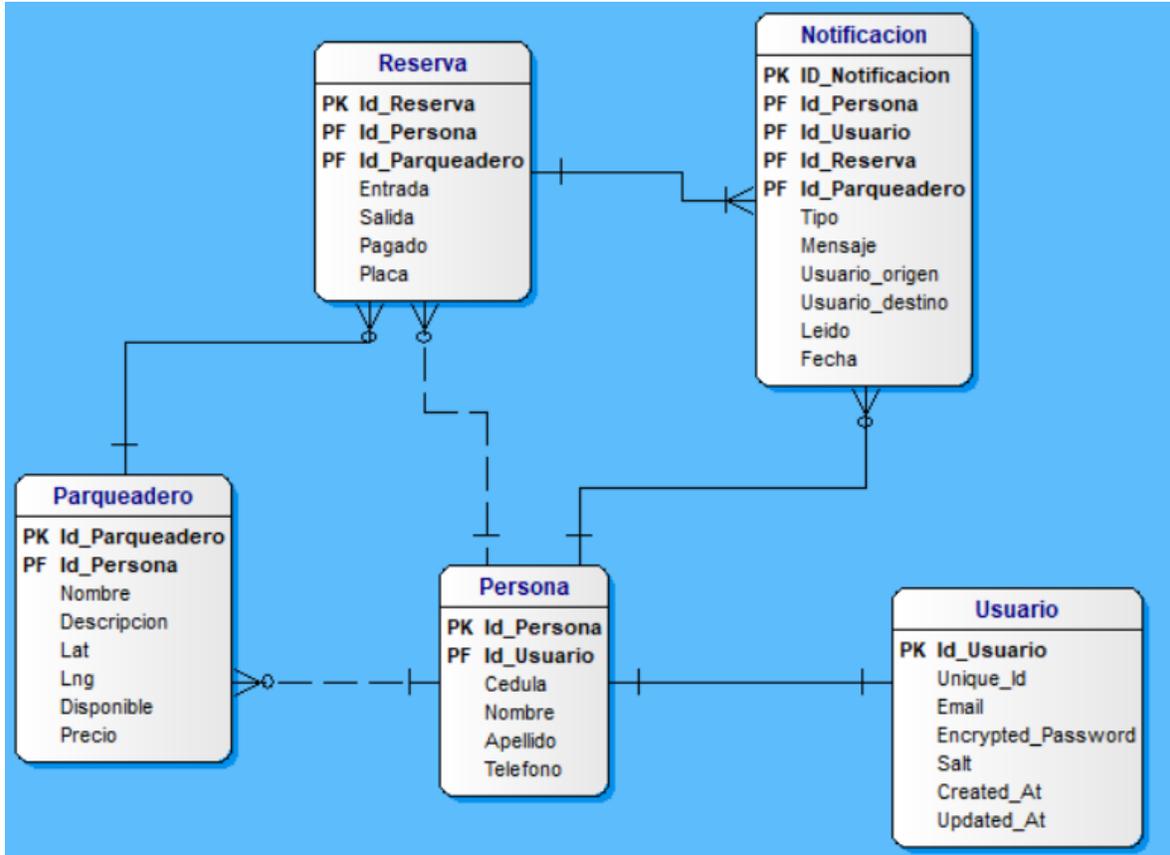


Figura 7. Esquema Lógico Estándar.

El esquema presenta la tabla Persona que tiene una relación de uno a uno con la tabla Usuario y la tabla Parqueadero tiene una relación de uno a muchos con la tabla Persona y Reserva. Todas las tablas son presentadas con sus respectivos campos.

4.2. Implementación

El proceso de implementación detalla de qué forma son desarrolladas las historias del usuario a través de los distintos sprints, éstos a su vez son definidos en el

Product Backlog que representa las acciones que la aplicación móvil debe realizar, para solventarlas se define el proceso realizado y el resultado de éste.

4.2.1. *Product Backlog*

Es una lista ordenada de las necesidades del usuario, dicha lista evoluciona en función del proceso de desarrollo de proyecto y el miembro encargado como Dueño del Proyecto debe monitorizar el desarrollo y cambios que se implementen en el mismo.

La tabla 2 muestra el *Product Backlog* de la aplicación móvil junto con las distintas historias de usuario, el sprint correspondiente a cada historia de usuario y el esfuerzo calculado para cada uno.

Tabla 2. Product Backlog.

Número	Historia de Usuario	Sprint	Esfuerzo
1	Ingresar Usuario	1	1
2	Modificar Usuario	1	1
3	Login Usuario	1	2
4	Mostrar Información Usuario	1	1
5	Ingresar Parqueadero	2	1
6	Modificar Parqueadero	2	1
7	Eliminar Parqueadero	2	1
8	Mostrar Parqueaderos Usuario	2	1
9	Mostrar Parqueaderos	2	1
10	Ingresar Reserva	3	1
11	Eliminar Reserva	3	1
12	Confirmar Reserva	3	1
13	Contactar Usuario	3	2

14	Análisis de Reservas Realizadas	4	3
15	Análisis de Reservas	4	3
16	Análisis de Gastos	4	3
17	Análisis de Ganancias	4	3

4.2.2. Historias de Usuario

Las historias de usuario representan los requerimientos funcionales de un proyecto, es decir, las funciones que deben realizarse con la finalidad de satisfacer al cliente, las historias de usuario deben definirse de manera detallada y organizada para tener conocimientos sobre los miembros del proyecto que lo realizarán, el sprint al cual pertenece, el riesgo que representa la adición de la funcionalidad al proyecto y datos adicionales que facilitan el desarrollo de éste.

La tabla 3 detalla la primera historia de usuario, la cual se centra en el registro de un usuario a la aplicación móvil por medio de la creación de credenciales para el posterior uso de ésta.

La tabla muestra una serie de conceptos que son necesarios definir para cada historia de usuario como el número de ésta, a quien le pertenece dicha funcionalidad, el miembro del equipo de desarrollo encargado de implementarla, etc.

Tabla 3. Primera Historia de Usuario.

Historia de Usuario	
Número: 1	Usuario: Usuario
Nombre Historia: Ingresar Usuario	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 1

Programador Responsable: Betty Burbano, Tabata Jardim
Descripción: Se presenta un formulario en donde el usuario puede crear las credenciales que le serán útiles para utilizar la aplicación.
Validación: El usuario crea nuevas credenciales llenando los campos del formulario con la información necesaria y al presionar un botón se valida que todos los datos se encuentren correctos y que las credenciales no existan.

En los Anexos del 4 al 19 se muestran las otras historias de usuario las cuales hacen referencia a la funcionalidad del sistema, en las cuales se detallan las siguientes funciones y características:

1. Modificación personal a la hora del registro y/o mantenimiento de un usuario.
2. Proceso de logueo en la aplicación, en las cuales se utilizan las credenciales del usuario una vez que éste está registrado.
3. Visualización de la información relevante dentro de la aplicación móvil.
4. Ingreso, modificación y/o eliminación de varios parqueaderos por parte del usuario en el cual puede colocar sus características como ubicación, valor y disponibilidad, para el uso de otros usuarios.
5. Proceso de ingreso, confirmación y/o eliminación de la reserva a un parqueadero en la zona solicitada por el usuario, lo cual permite la gestión oportuna de la transacción, con los debidos involucrados.
6. Visualización de un *dashboard* entendible en el que se detallan las negociaciones realizadas por los usuarios, dinero invertido, dinero ganado, mediante la implementación de gráficas.

4.2.3. Sprints

Un sprint representa un conjunto de actividades desarrolladas en un proyecto que permite completar una función de éste, los sprints deben ser definidos en base a los requerimientos planteados para el proceso de implementación del proyecto y son usados como una forma de repartición de tareas entre los miembros del equipo de desarrollo.

La tabla 4 muestra los sprints definidos para el proyecto, cada sprint muestra las historias de usuario que pertenecen a cada bloque de tareas.

Tabla 4. Tabla de Sprints con Historias de Usuario.

Spint 1
Ingresar Usuario
Modificar Usuario
Login Usuario
Mostrar Información Usuario
Spint 2
Ingresar Parqueaderos
Modificar Parqueaderos
Eliminar Parqueaderos
Mostrar Parqueaderos Usuario
Mostrar Parqueaderos
Spint 3
Ingresar Reserva

Eliminar Reserva
Confirmar Reserva
Contactar Usuario
Spint 4
Análisis de reservas realizadas
Análisis de reservas
Análisis de Gastos
Análisis de Ganancias

4.2.3.1. Sprint 1

El Sprint 1 representa el conjunto de tareas que se encarga de la gestión del usuario en la aplicación móvil, en éste se definen las primeras cuatro historias de usuario correspondientes al registro de un usuario en la aplicación, modificación de sus datos, logueo y despliegue de la información de éste.

- **Historia 1**

La primera historia de usuario detalla el proceso de registro y validación de los datos de un usuario en la aplicación móvil, el proceso se visualiza en la figura 9. Cabe recalcar que para esto se utiliza el método de encriptación SHA de ida, para almacenar la contraseña encriptada.

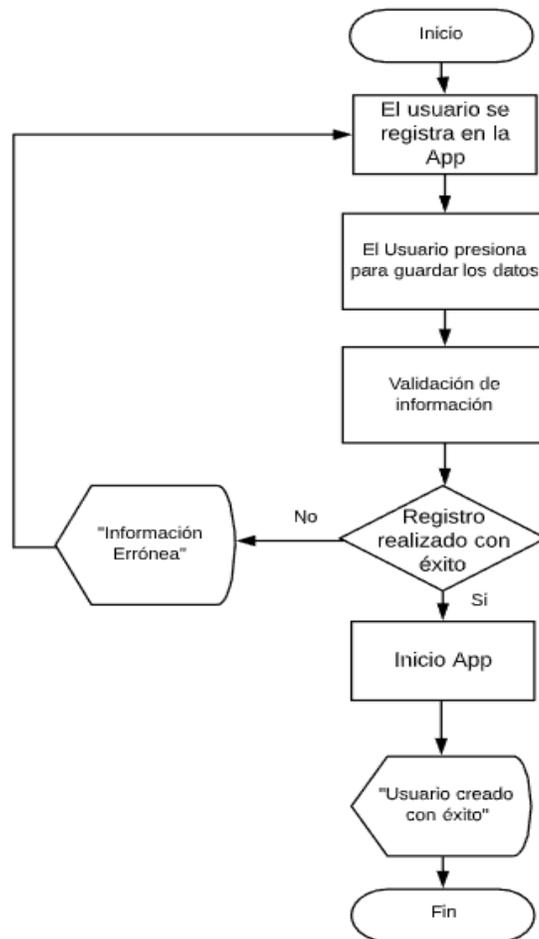


Figura 8. Diagrama de Flujo proceso de ingreso de Usuario.

La estructura de los diagramas de este documento muestra un inicio y fin representados con óvalos, las actividades son definidos en rectángulos, si una actividad puede derivarse a más de una actividad siguiente se usa un rombo como símbolo de decisión y los despliegues de mensajes en pantalla son mostrados con su figura característica, usada en diagramas de flujo.

La figura 10 muestra las tablas de la base de datos involucradas en la primera historia de usuario y la relación entre las mismas.

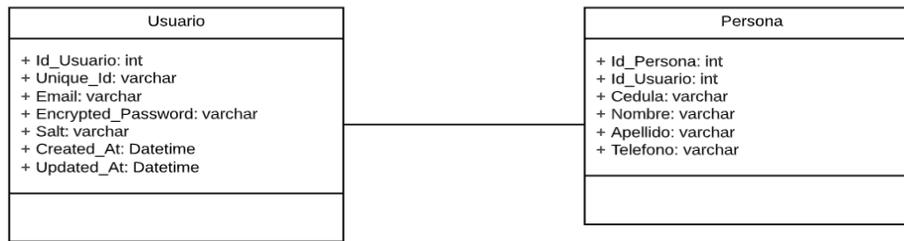


Figura 9. Tablas de base de datos utilizadas en la primera historia de usuario.

Es importante mencionar que las tablas son mostradas con sus respectivos campos y relaciones.

- **Muestra de solución**

En la figura 11 se puede observar el resultado final de la historia 1, logrando obtener la pantalla para el registro y validación de los datos del usuario.

Formulario de registro de usuario:

- Campos de entrada: Email, Contraseña, Nombre, Apellido, Cédula, Teléfono.
- Botón: REGISTRAR
- Botón: ¡Ya estoy registrado! Iniciar Sesión
- Ilustración de un skyline urbano.

Figura 10. Registro y validación de los datos del usuario.

- **Historia 2**

La segunda historia de usuario indica como un usuario puede modificar sus datos dentro de la aplicación, el diagrama de flujo del proceso junto con las tablas de la base de datos correspondiente a esta historia se encuentra en el Anexo 20.

- **Muestra de solución**

En la figura 12 se puede observar el resultado final de la historia 2, logrando obtener la pantalla para la visualización y modificación de los datos del usuario.

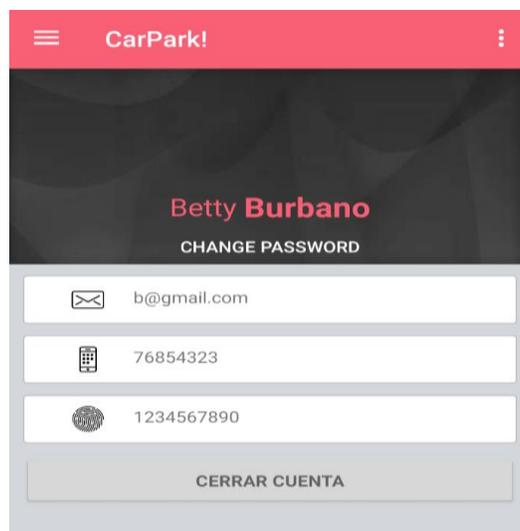


Figura 11. Visualización y modificación de los datos del usuario.

- **Historia 3**

En la tercera historia de usuario se visualiza el proceso para que un usuario pueda loguearse en la aplicación, el diagrama de flujo y las tablas de la base de datos que pertenecen a esta historia se encuentran en el Anexo 21. Para el proceso de logueo se encripta la contraseña escrita y se compara con la contraseña almacenada en la base de datos.

- **Muestra de solución**

En la figura 13 se puede observar el resultado final de la historia 2, logrando obtener la pantalla para la visualización y modificación de los datos del usuario.

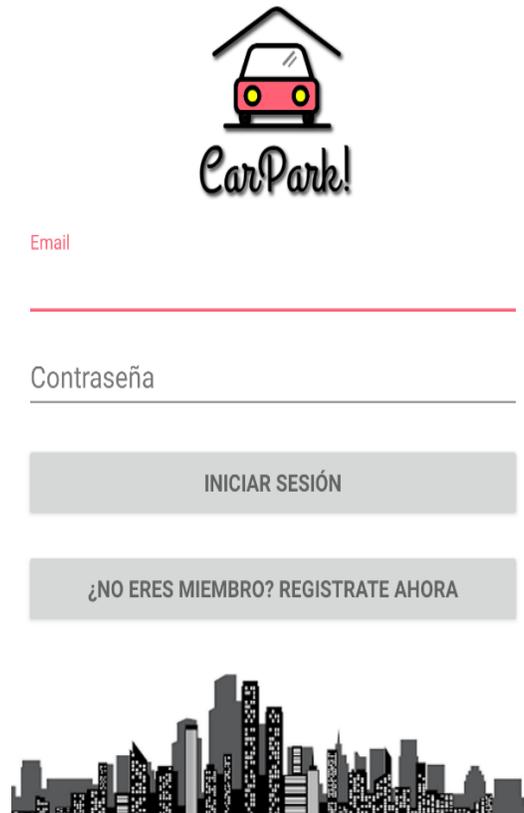


Figura 12. Pantalla de Logeo.

- **Historia 4**

El proceso para la cuarta historia de usuario se encuentra en el Anexo 22 y detalla como un usuario puede visualizar su información en la aplicación móvil.

- **Muestra de solución**

En la figura 14 se puede observar el resultado final de la historia 4, logrando obtener la pantalla para la visualización de la información proporcionada por el usuario.

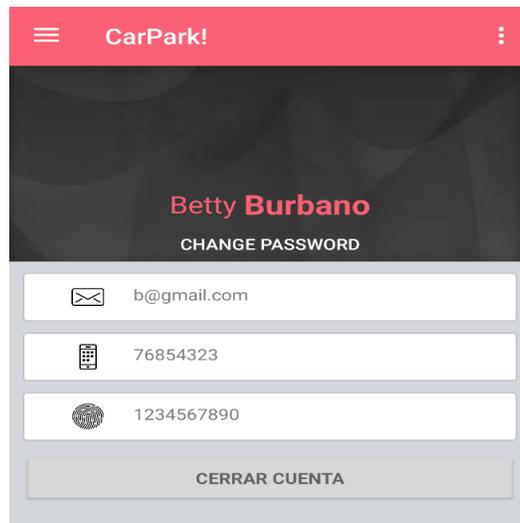


Figura 13. Pantalla de visualización información proporcionada por el usuario.

4.2.3.2. Sprint 2

El segundo Sprint representa las acciones relacionadas con los parqueaderos, abarca la quinta, sexta, séptima, octava y novena historia de usuario que realizan las acciones de registro, modificación y eliminación de parqueaderos junto con el despliegue de información de estos.

- **Historia 5**

La quinta historia de usuario explica el proceso de registro de un parqueadero, su diagrama de flujo y la tabla que utiliza se encuentra detallada en el Anexo 23.

- **Muestra de solución**

En la figura 15 se puede observar el resultado final de la historia 5, logrando obtener la pantalla para el registro de un nuevo parqueadero.



Figura 14. Registro de un nuevo parqueadero.

- **Historia 6**

La sexta historia de usuario detalla como un usuario puede modificar los datos de su parqueadero, esto se visualiza en el diagrama de flujo del y las tablas del Anexo 24.

- **Muestra de solución**

En la figura 16 se puede observar el resultado final de la historia 6, logrando obtener la pantalla para modificar los datos de un parqueadero ya creado anteriormente.

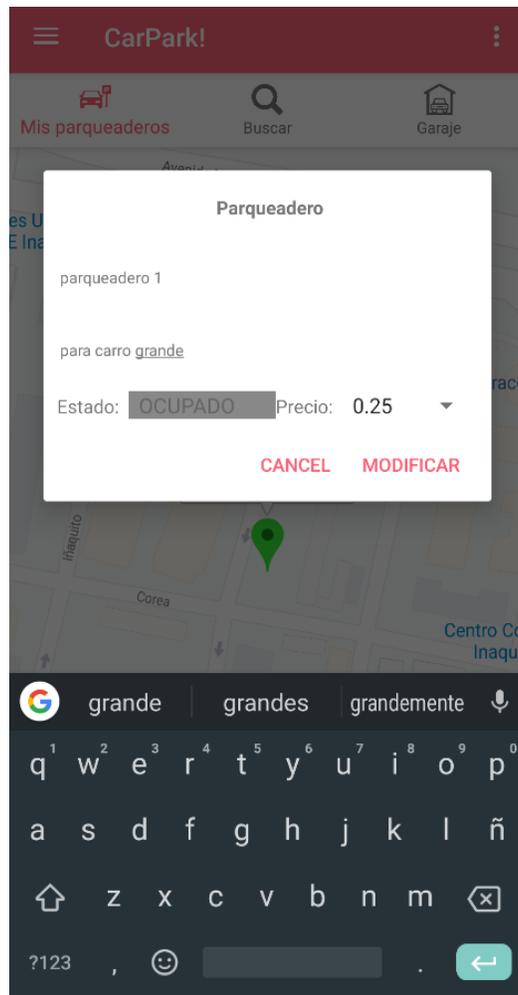


Figura 15. Modificar los datos de un parqueadero existente.

- **Historia 7**

La séptima historia explica de qué forma un usuario puede eliminar un parqueadero, el diagrama de flujo y las tablas de la base de datos correspondientes se encuentran en el Anexo 25.

- **Muestra de solución**

En la figura 17 se puede observar el resultado final de la historia 7, logrando obtener la pantalla para eliminar un parqueadero creado con anterioridad.

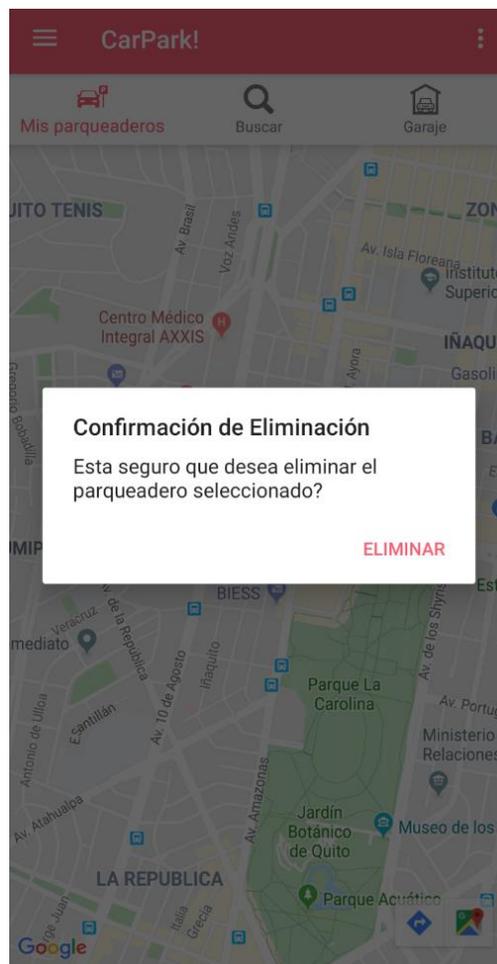


Figura 16. Eliminar un parqueadero existente.

- **Historia 8**

Un usuario puede visualizar sus parqueaderos con la información detallada de los mismos, el proceso se muestra en el Anexo 26 que tiene el diagrama de flujo y las tablas correspondientes.

- **Muestra de solución**

En la figura 18 se puede observar el resultado final de la historia 8, logrando obtener la pantalla para la visualización de los parqueaderos creados por el usuario.



Figura 17. Parqueaderos creados por el usuario.

- **Historia 9**

Todos los parqueaderos se despliegan junto con su información para que puedan ser revisados por los distintos usuarios de la aplicación móvil, el proceso de esta acción esta detallado en el Anexo 27.

- **Muestra de solución**

En la figura 19 se puede observar el resultado final de la historia 9, logrando obtener la pantalla para la visualización de todos los parqueaderos creados en la plataforma móvil.



Figura 18. Parqueaderos existentes en la plataforma móvil.

4.2.3.3. Sprint 3

El Sprint 3 se centra en las funciones de reservas de parqueaderos, en éste se encuentran la décima, decimoprimer, decimosegunda y decimotercera historia correspondiente al ingreso, eliminación y confirmación de la reserva y el contacto con el dueño del parqueadero.

- **Historia 10**

Esta historia de usuario se centra en el proceso de registro de una reserva de un parqueadero, la explicación de este proceso se encuentra en el Anexo 28.

- **Muestra de solución**

En la figura 20 se puede observar el resultado final de la historia 10, logrando obtener la pantalla para realizar el registro de la reserva de un parqueadero.

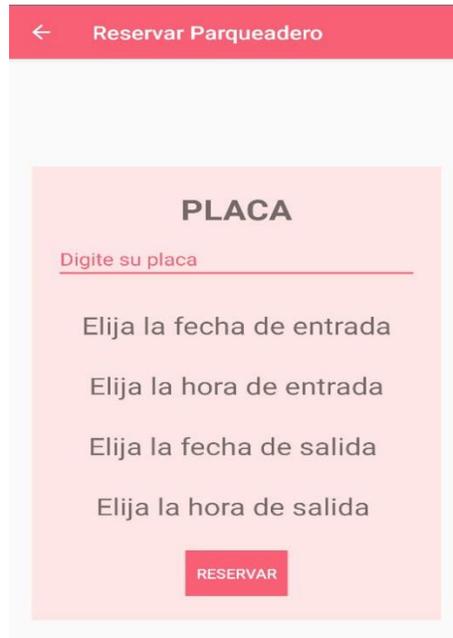


Figura 19. Crear una reserva.

- **Historia 11**

En esta historia de usuario se detalla la forma en que un usuario puede eliminar una reserva, el proceso correspondiente se encuentra en el Anexo 29.

- **Muestra de solución**

En la figura 21 se puede visualizar el resultado final de la historia 11, logrando obtener la pantalla para eliminar una reserva no deseada.



Figura 20. Eliminar una reserva.

- **Historia 12**

En el Anexo 30 se encuentra el proceso correspondiente para la confirmación de la reserva por parte del dueño del parqueadero que se detalla en la decimosegunda historia.

- **Muestra de solución**

En la figura 22 se puede visualizar el resultado final de la historia 12, obteniendo la pantalla para cuando se realiza la confirmación de una reserva de un parqueadero entre usuarios de la plataforma.



Figura 21. Confirmación de una reserva.

- **Historia 13**

La decimotercera historia detalla como un usuario puede comunicarse con el dueño del parqueadero al cual desea acceder, esto se muestra en el Anexo 31.

- **Muestra de solución**

En la figura 23 se puede visualizar el resultado final de la historia 13, obteniendo la pantalla que direcciona al usuario para que éste se pueda contactar cuando desee con el dueño del parqueadero.



Figura 22. Contactar al dueño del parqueadero.

4.2.3.4. Sprint 4

El Sprint 4 tiene un bloque de funciones relacionadas con el análisis de datos mediante gráficos estadísticos, dentro del mismo se encuentran las historias de usuario número catorce, quince, dieciséis y diecisiete que detallan los procesos de análisis de reservas realizadas, reservas hechas a un parqueadero específico, gastos y ganancias.

- **Historia 14**

En el Anexo 32 detalla el proceso de visualización de varios gráficos estadístico que le muestra al usuario las reservas realizadas, las reservas que le han hecho a sus

parqueaderos, cuando ha invertido en la aplicación y cuánto dinero ha ganado en ella.

- **Muestra de solución**

En la figura 24 se puede visualizar el resultado final de la historia 14, donde el usuario tendrá una vista amplia de un dashboard, para elegir que gráfico desea observar.

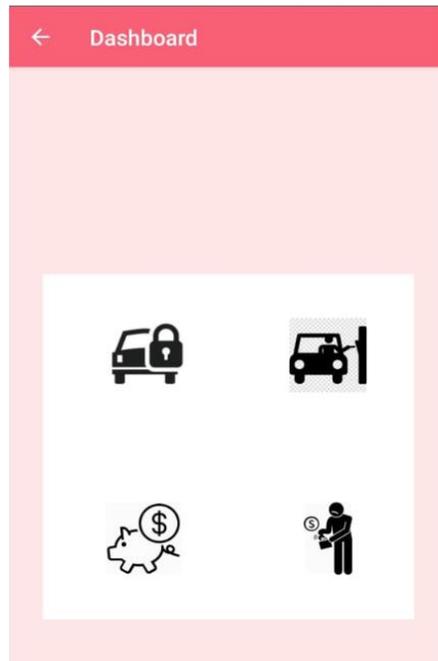


Figura 23. Dashboard que representa mediante gráficos la interacción del usuario con la aplicación móvil.

- **Historia 15**

Un usuario, dueño de un parqueadero puede visualizar con gráficas la cantidad de reservas que se han realizado en su plaza de parqueo, este proceso se define en el Anexo 33.

- **Muestra de solución**

En la figura 25 se puede visualizar el resultado final de la historia 15, obteniendo la pantalla donde el usuario puede ver la cantidad de reservas que le han realizado en sus parqueaderos, mediante un gráfico amigable.

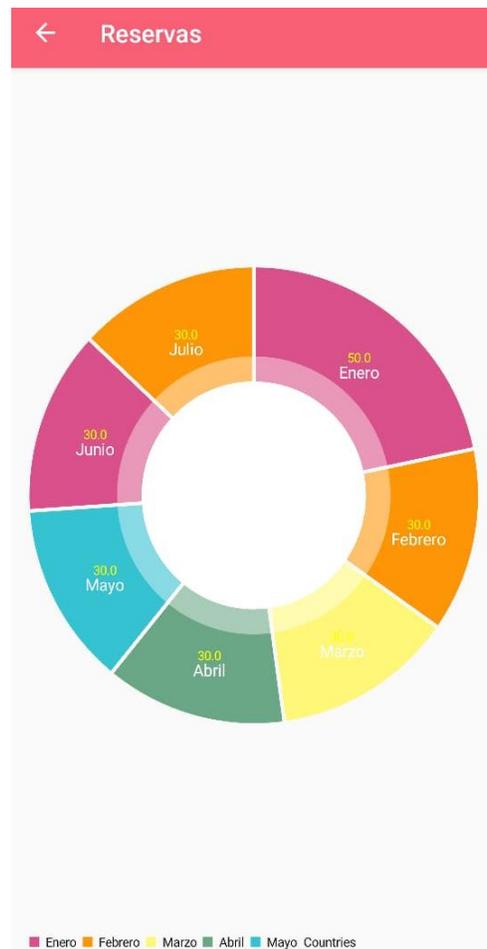


Figura 24. Gráfico de la cantidad de reservas realizadas a los parqueaderos de un usuario.

- **Historia 16**

Un usuario puede visualizar los gastos que ha realizado en un tiempo específico con el uso de la aplicación, esto se encuentra detallado en el Anexo 34.

- **Muestra de solución**

En la figura 26 se puede visualizar el resultado final de la historia 16, obteniendo la pantalla donde el usuario puede ver el gasto generado en arrendar parqueaderos dentro de la aplicación, mediante un gráfico amigable.



Figura 25. Gráfico de los gastos que tiene un usuario al arrendar parqueaderos de otros.

- **Historia 17**

El Anexo 35 muestra como un usuario puede determinar las ganancias que ha obtenido un usuario alquilando su parqueadero a otros.

- **Muestra de solución**

En la figura 27 se puede visualizar el resultado final de la historia 17, obteniendo la pantalla donde el usuario puede ver mediante un gráfico amigable todas las ganancias obtenidas al alquilar su parqueadero a otros usuarios.

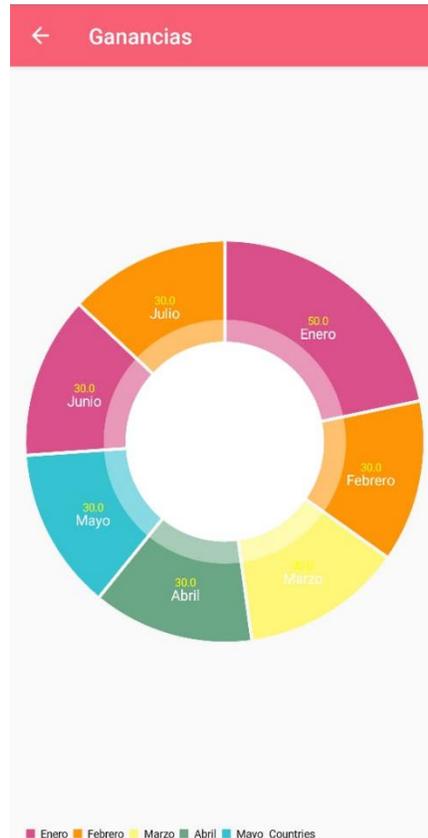


Figura 26. Gráfico de las ganancias obtenidas al alquilar su parqueadero a otros.

4.2.4. Tabla de requerimientos no funcionales

Los requerimientos no funcionales son requisitos para un proyecto que no necesitan la especificación de una tarea específica, se refiere a las propiedades que surgen de las mismas, están pueden ser definiciones de la forma en que operará el proyecto o restricciones en su funcionamiento (Sommerville & Galipienso, 2005).

Para la aplicación móvil se han definido algunos requerimientos no funcionales con el uso de métricas, los cuales se encuentran detallados en la tabla 5:

Tabla 5. Requerimientos no funcionales.

Requerimientos no funcionales
Facilidad de uso
La aplicación móvil será desarrollada pensando plenamente en experiencia del usuario, mostrando una interfaz amigable, con colores simpáticos que se vuelven agradables para la vista. Adicional a esto se sumarán cuadros de ayuda que permiten que el usuario pueda comprender de mejor forma las funcionalidades de la aplicación móvil.
Integridad
Los datos del usuario serán mostrados únicamente a quien le pertenece, no se mostrarán datos sensibles de otro usuario y estos serán almacenados de tal forma que se encuentren protegidos.
Portabilidad
El desarrollo de la aplicación móvil se enfocará en el uso de esta en el sistema Android, para ello se usará el lenguaje de programación Java y la base de datos MySQL, con esto se busca abarcar al 70 por ciento del mercado quiteño.

5. CAPÍTULO V. ANÁLISIS DE RESULTADOS

En este capítulo se encuentra el análisis realizado para el prototipo desarrollado, se despliega el detalle de cada historia de usuario junto con los pasos necesarios para desarrollar la acción.

5.1. Verificación de funcionalidad

Esta sección muestra la funcionalidad del prototipo realizado, comprobando cada acción con la historia de usuario correspondiente.

5.1.1. Primera historia de usuario: Ingresar Usuario

Para que el usuario pueda registrarse debe ingresar a la aplicación móvil, ésta se abrirá mostrando la siguiente pantalla, debiendo hacer clic en el botón ¿NO ERES MIEMBRO? REGISTRATE AHORA, lo detallado se visualiza en la figura 28.



Figura 27. Ingreso del usuario por primera vez.

Después de esto aparecerá el siguiente formulario, el usuario debe ingresar sus datos y hacer clic en el botón REGISTRAR, lo cual se detalla en la figura 29.

Email
 alejandrabq2497@gmail.com

Contraseña
 ••••••••

Nombre
 Betty

Apellido
 Burbano

Cédula
 1718802380

Teléfono
 0987448618

REGISTRAR

¡Ya estoy registrado! Iniciar Sesión



Figura 28. Proceso de registro de un usuario.

En la base de datos se almacenará los datos ingresados por el usuario en las tablas Usuario y Persona, se muestra un ejemplo en la figura 30.

Tabla Usuario

ID_USUARIO	UNIQUE_ID	EMAIL	ENCRYPTED_PASSWORD
24	5dedd1e1c2faf2.45030803	alejandrabq2497@gmail.com	CqqFdDQZIGXuqACKLkfe7+Y1u41mMDI3MwVkyWJh

Tabla Persona

ID_PERSONA	ID_USUARIO	CEDULA	NOMBRE	APELLIDO	TELEFONO
14	24	1718802380	Betty	Burbano	0987448618
11	21	333333333	Heor3	Heor3	333333333

Figura 29. Almacenamiento de datos en la base de datos.

5.1.2. Segunda historia de usuario: Modificar Usuario

Para que el usuario modifique sus datos debe ingresar en la opción Mi Perfil que se encuentra en el menú lateral, tal como se detalla en la figura 31.

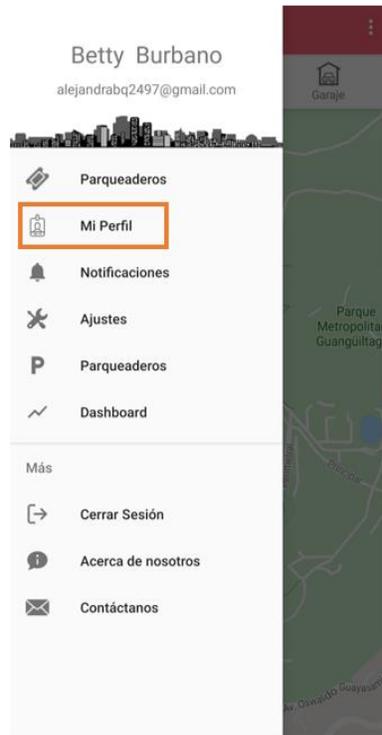


Figura 30. Proceso para modificación de los datos de un usuario.

Una vez elegida la opción aparecerá la siguiente ventana con los datos del usuario, el podrá modificarlos cuando desee haciendo clic en el botón CAMBIAR INFORMACIÓN. Lo detallado de puede visualizar en las figuras 32 y 33.

Figura 32. Cambiar información personal

Figura 31. Información cambiada

Una vez realizado esto se modificarán los datos en el registro del usuario correspondiente en la tabla Persona.

Por seguridad se cerrará sesión una vez modificados los datos del usuario. Los cambios se visualizan en la figura 34.

ID_PERSONA	ID_USUARIO	CEDULA	NOMBRE	APELLIDO	TELEFONO
14	24	1718802380	Betty Alejandra	Burbano	0987448618

Figura 33. Cambio realizado que se muestra en la base de datos.

5.1.3. Tercera historia de usuario: Login Usuario

Para que un usuario pueda loguearse debe ingresar a la aplicación móvil e ingresar el correo y la contraseña que ingresó anteriormente. El proceso de login está representado en la figura 35.

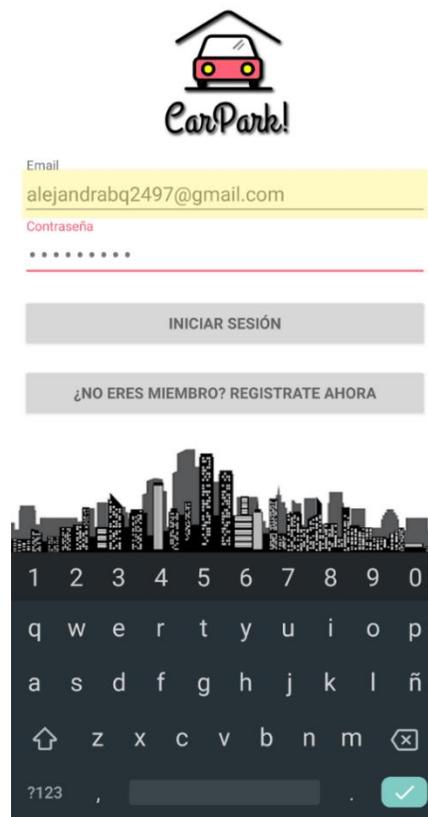


Figura 34. Login

5.1.4. Cuarta historia de usuario: Mostrar información del Usuario

Para que un usuario pueda visualizar su información debe elegir la opción Mi Perfil en el menú lateral. En la figura 36 se detalla la información del usuario.

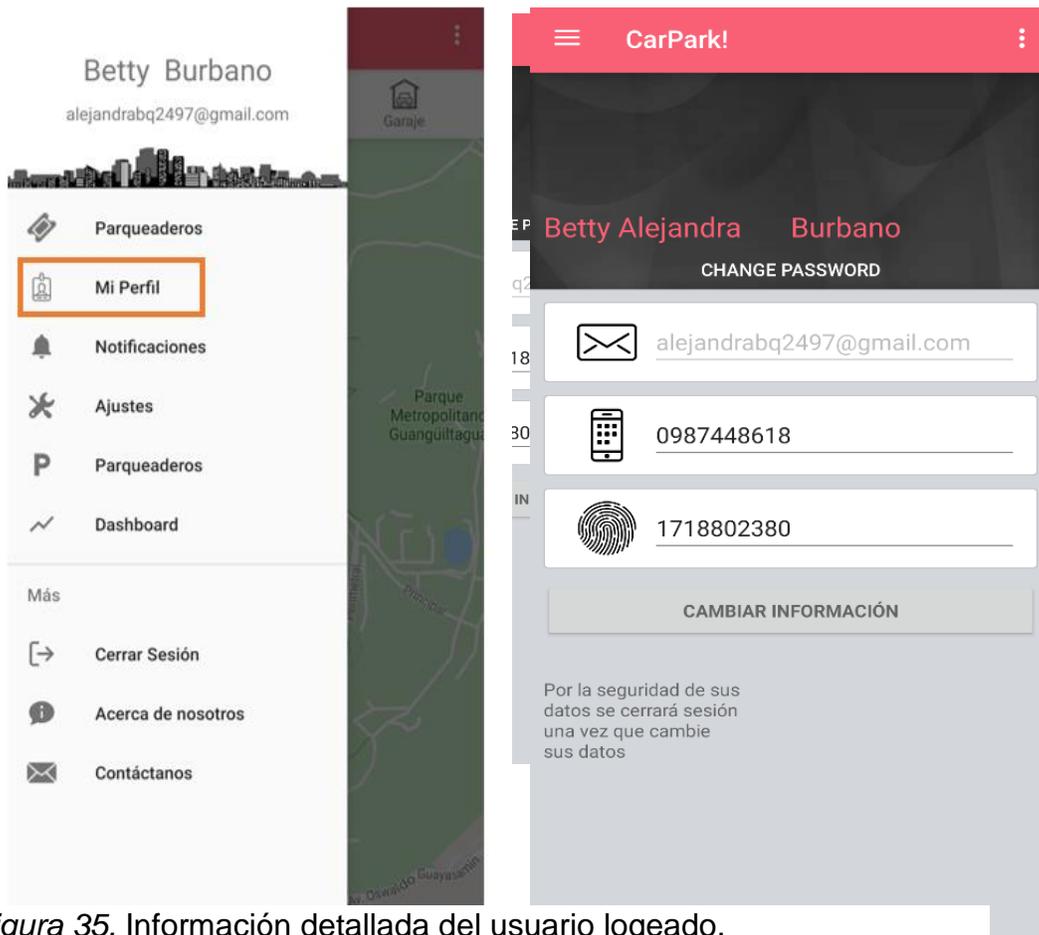


Figura 35. Información detallada del usuario logeado.

5.1.5. Quinta historia de usuario: Ingresar Parquero

Para que un usuario pueda ingresar un parquero debe posicionarse en la ventana Mis Parqueros, lo cual se visualiza en la figura 37.



Figura 36. Ingresar un parqueadero.

El usuario debe hacer clic en el lugar en donde desea ingresar el parqueadero y debe ingresar los datos de este, el proceso detallado se encuentra en la figura 38.

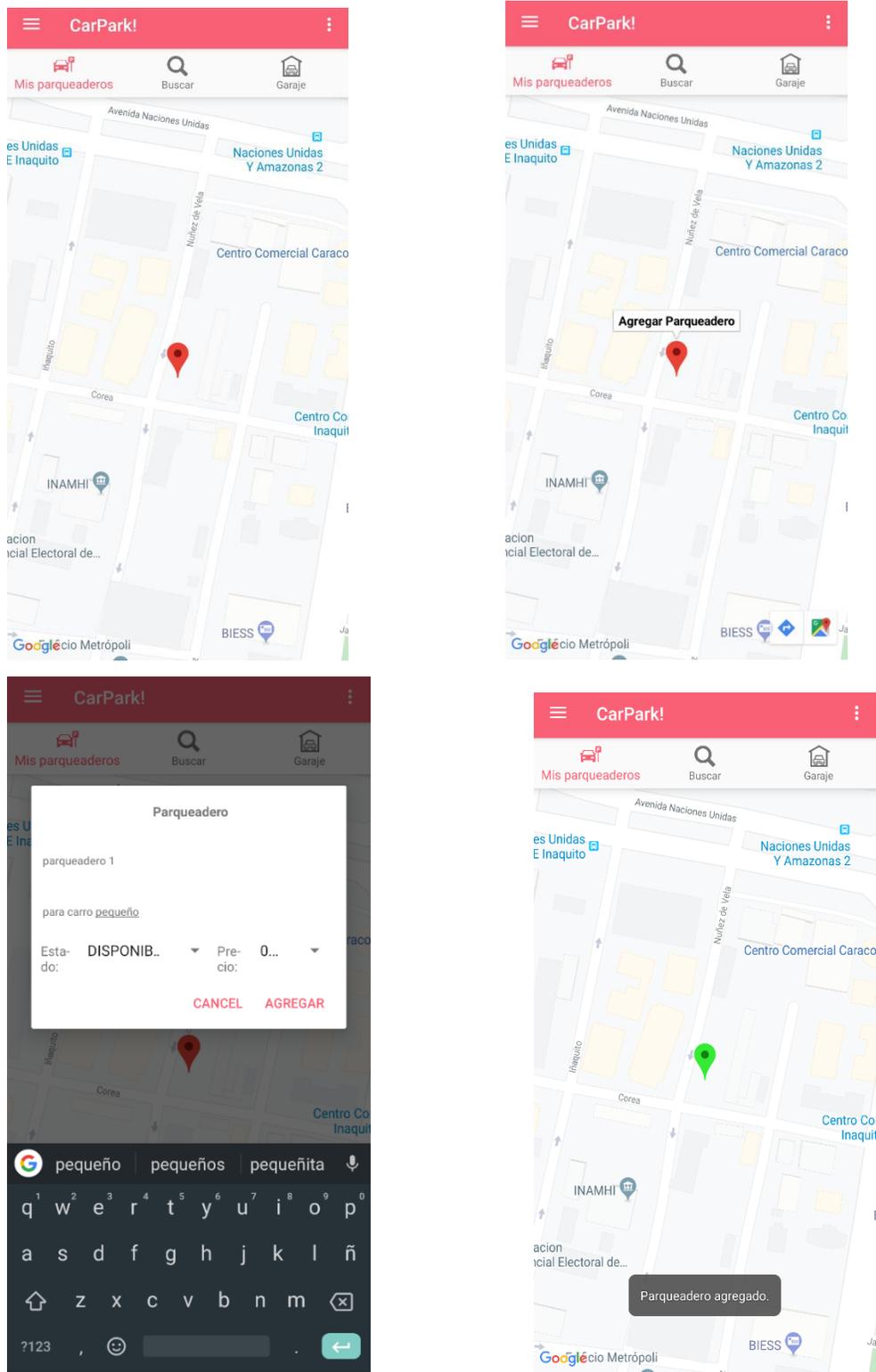


Figura 37. Proceso para el ingreso de un parqueadero nuevo.

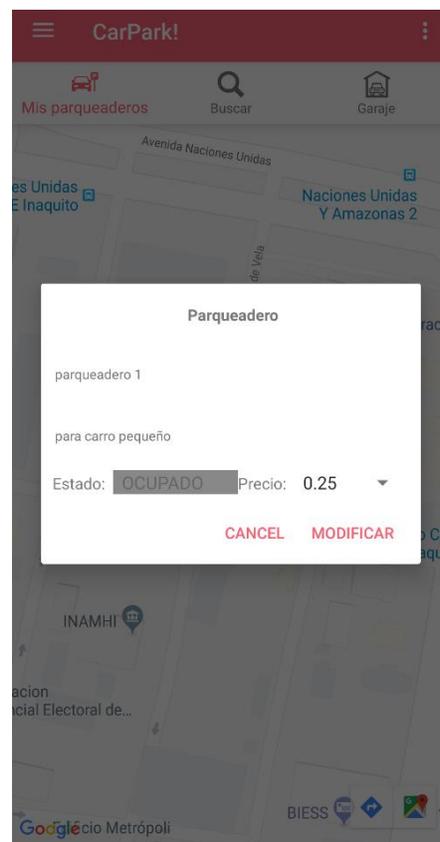
El registro se almacenará en la tabla Parquadero. A continuación, se despliega la figura 39 para su visualización

ID_PARQUEADERO	ID_PERSONA	NOMBRE	DESCRIPCION	LAT	LNG	DISPONIBLE	PRECIO
1	14	parquadero 1	para carro pequeño	-0.17704428870485675	-78.48700437694788	1	0.75

Figura 38. Creación de un parquadero en la base de datos.

5.1.6. Sexta historia de usuario: Modificar Parquadero

Un usuario puede modificar la información de sus parquaderos seleccionándolos y cambiando la información de éstos, tal como se puede visualizar en las figuras 40 y 41.



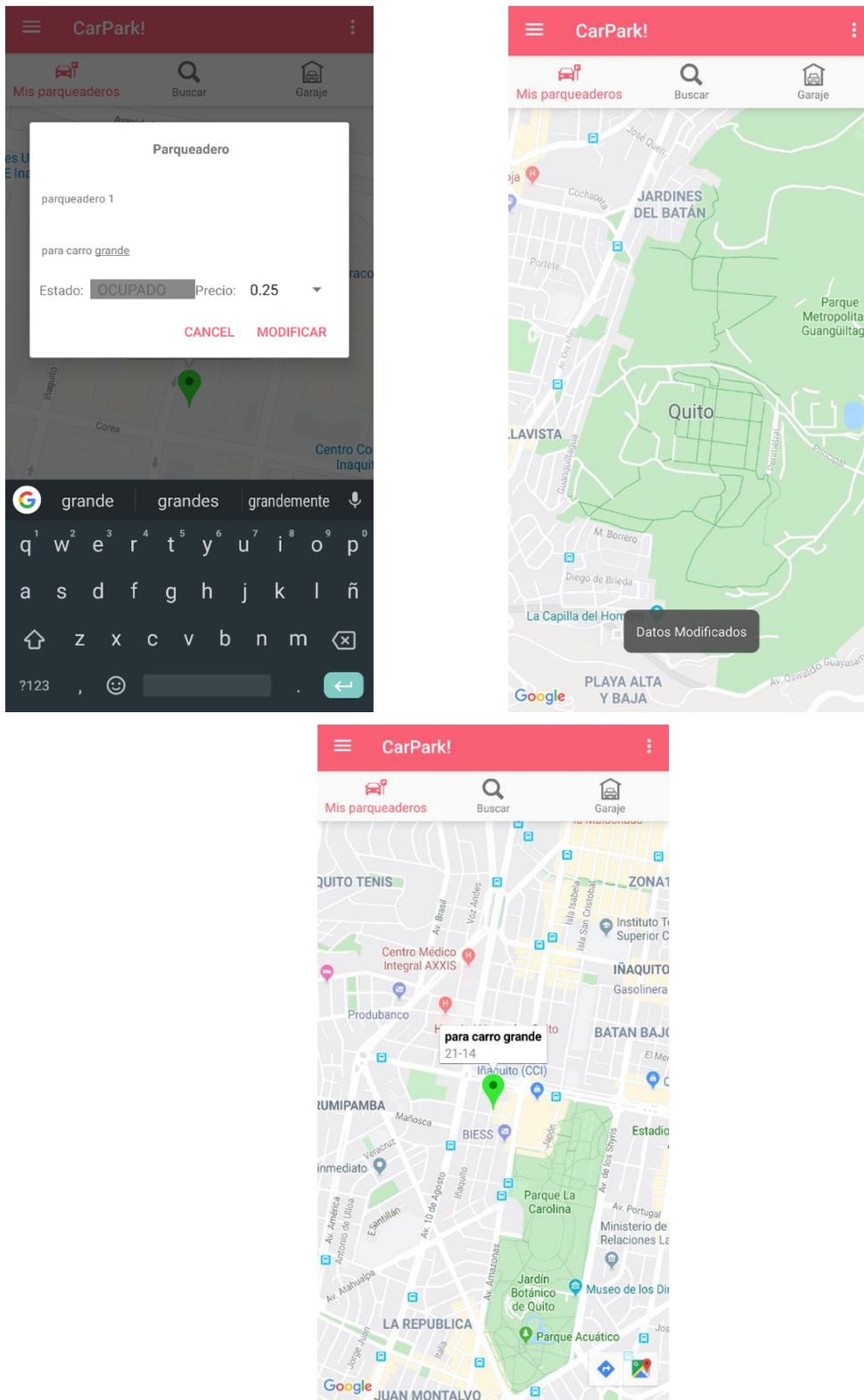


Figura 39. Modificar un parqueadero existente.

ID_PARQUEADERO	ID_PERSONA	NOMBRE	DESCRIPCION	LAT	LNG	DISPONIBLE	PRECIO
21	14	parqueadero 1	para carro grande	-0.17704428870485675	-78.48700437694788	1	0.25

Figura 40. Modificación del parqueadero en la base de datos.

5.1.7. Séptima historia de usuario: Eliminar Parqueadero

Para eliminar un parqueadero se debe hacer un clic largo sobre el mismo y confirmar la eliminación, el siguiente proceso se visualiza en la figura 42.

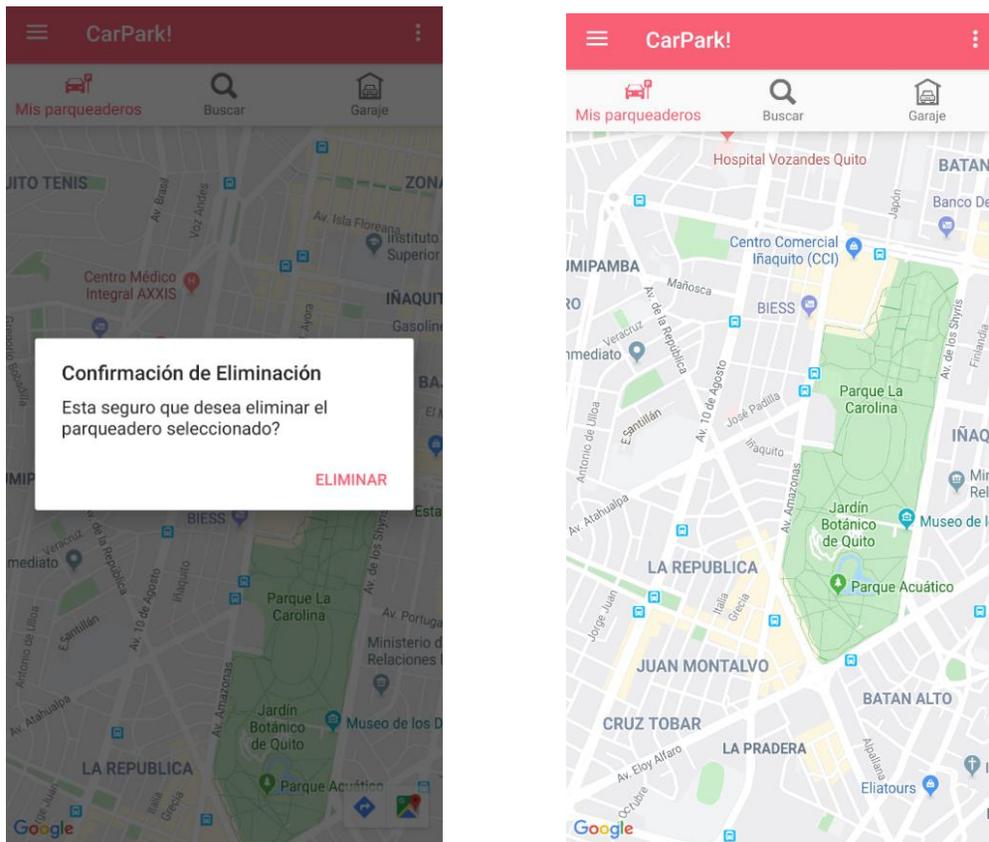


Figura 41. Eliminar un parqueadero existente.

5.1.8. Octava historia de usuario: Mostrar Parqueadero del Usuario

Para que un usuario pueda visualizar sus parqueaderos debe estar en la pestaña Mis Parqueaderos, esto se define en la figura 43.



Figura 42. Mostrar los parqueaderos de un usuario.

5.1.9. Novena historia de usuario: Mostrar Parqueadero registrados en la aplicación móvil

En la pestaña buscar se muestran todos los parqueaderos disponibles en el Distrito Metropolitano de Quito, lo cual se encuentra en la figura 44.



Figura 43. Buscar parqueaderos.

5.1.10. Décima historia de usuario: Ingresar Reserva

Para que un usuario realice una reserva debe seleccionar el parqueadero deseado y hacer clic en la etiqueta. En la figura 45 se encuentra como realizar una reserva de un parqueadero.



Figura 44. Realizar una reserva

Después de esto aparecerá el siguiente formulario en donde se debe ingresar la fecha y hora de entrada al parqueadero y la fecha y hora de salida junto con la placa del auto del usuario. El formulario de la reserva se encuentra señalado en la figura 46.

Figura 45. Formulario para ingreso de una reserva.

Una vez ingresados los datos, el usuario debe presionar el botón Reservar para hacer la petición de reservación al dueño del parqueadero.

Esto se almacenará en la tabla Reserva, lo cual se visualiza en la figura 47.

ID_RESERVA	ID_DEMANDANTE	ID_DUENO	ID_PARQUEADERO	ENTRADA	SALIDA	PAGADO	PLACA	fechaEntrada	fechaSalida	favorito	P_C	valorTotal	
1	8	8	4	16	13:20	14:30	NULL	ASD-567	2019/12/16	2019/12/16	NULL	1	1.25

Figura 46. Ingreso de los datos de la reserva en la base de datos.

5.1.11. Décima primera historia de usuario: Eliminar Reserva

En caso de que el usuario desee eliminar una reserva ya solicitada debe cancelar la confirmación presionando el botón CANCELAR en la pestaña Notificaciones, lo cual se encuentra representado en la figura 48.



Figura 47. Eliminar una reserva

Luego de esto con fines de mantener la integridad del funcionamiento de la aplicación móvil, el usuario debe eliminar la solicitud realizada en la ventana Reservas presionando sobre los datos de esta y confirmando su eliminación, esto se visualiza en la figura 49.

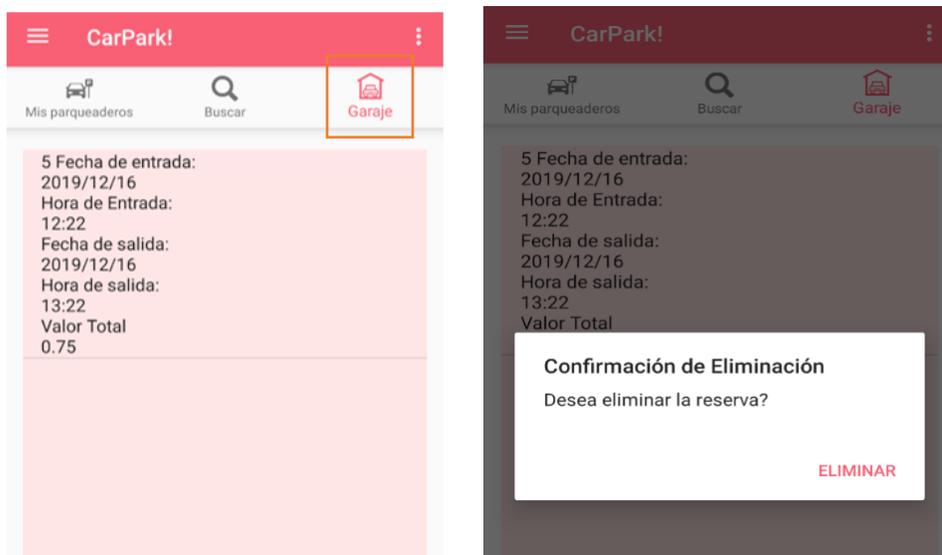


Figura 48. Confirmación para eliminar una reserva

5.1.12. Décima segunda historia de usuario: Confirmar Reserva

Una vez realizada la solicitud el usuario debe dirigirse a la pestaña Notificaciones en donde deberá confirmar que desea realizar la reserva, esto lo hará presionando el botón CONFIRMAR, tal como se puede confirmar en la figura 50.



Figura 49. Confirmación de una reserva

Al usuario le llegará una notificación mostrando si el dueño del parqueadero que deseo reservar acepta la solicitud o no. En la figura 51 se puede visualizar el proceso para aceptar o denegar una reserva.

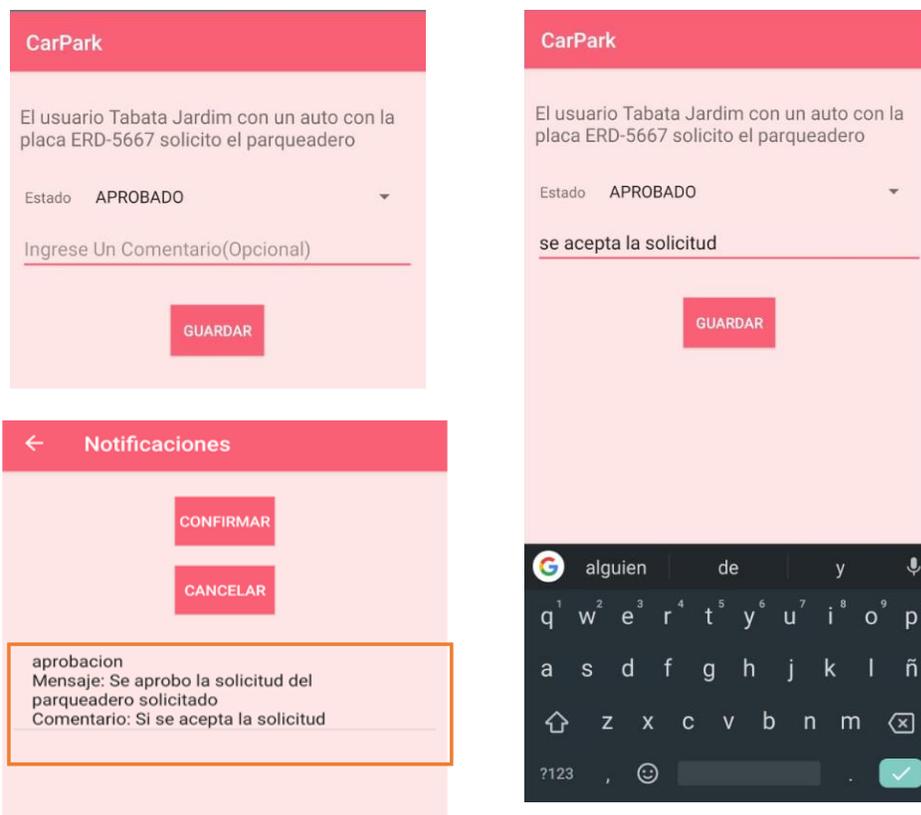


Figura 50. Acción ante una reserva.

Esto se almacena como una solicitud en la tabla notificación. El almacenamiento de los datos de la reserva se encuentra detallada en la figura 52.

id	tipo	mensaje	usuario_origen	usuario_destino	leido	id_reserva	fecha
1	solicitud	El usuario User2 User2 con un auto con la placa SD...	9	8	1	24	2019-12-16
2	aprobacion	Se aprobo la solicitud del parqueadero solicitado	8	9	0	24	2019-12-16
3	solicitud	El usuario User1 User1 con un auto con la placa WD...	8	9	0	20	2019-12-16
4	solicitud	El usuario User1 User1 con un auto con la placa AS...	8	9	1	20	2019-12-16
5	aprobacion	Se aprobo la solicitud del parqueadero solicitado	9	8	0	20	2019-12-16
9	solicitud	El usuario Tabata Jardim con un auto con la placa ...	15	14	1	27	

Figura 51. Almacenamiento de la reserva en la base de datos.

5.1.13. Décima tercera historia de usuario: Contactar a Usuario

Si un usuario desea contactar al dueño de un parqueadero puede llamarlo haciendo clic sobre la etiqueta de información de un parqueadero en la pestaña Buscar y confirmar la acción. El proceso de contactar a otro usuario dentro de la aplicación se puede visualizar en la figura 53.

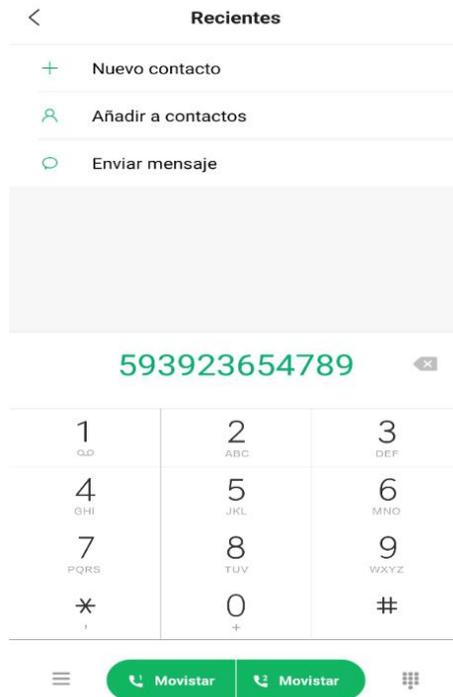
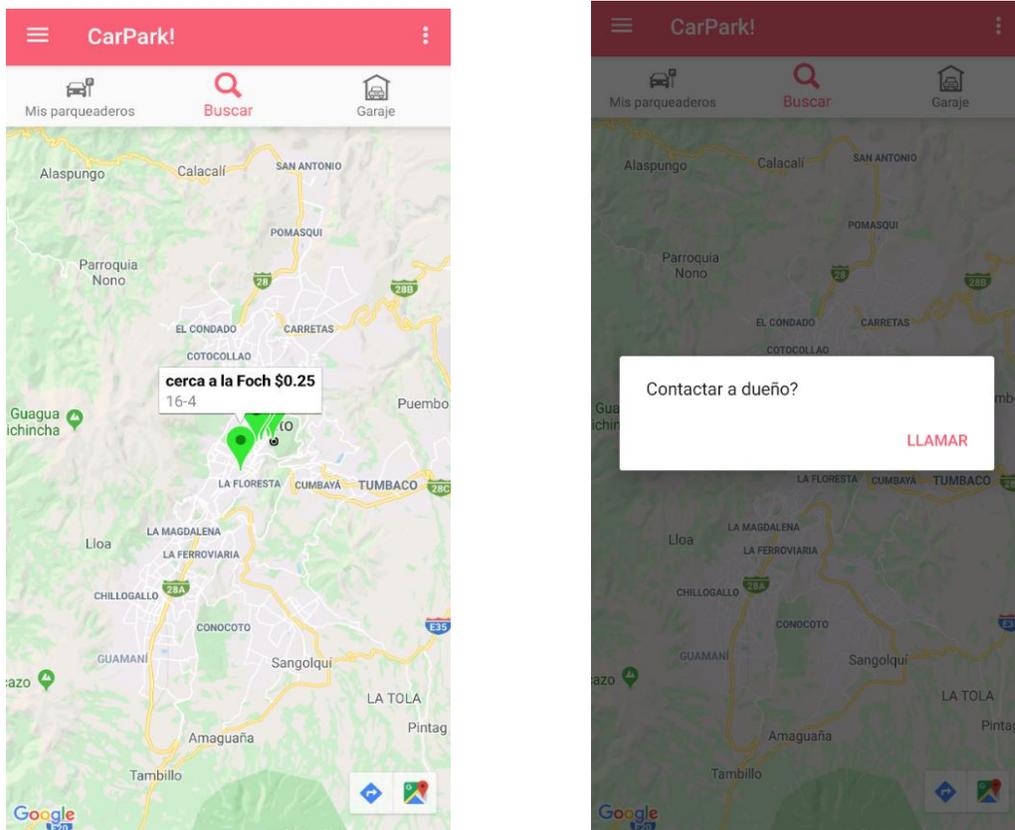


Figura 52. Contactar a un usuario.

5.1.14. Décima cuarta historia de usuario: Análisis de Reservas Realizadas

Un usuario puede ver la cantidad de reservas que ha realizado en cada mes en la opción *Dashboard* del menú lateral y seleccionando la opción Reservas Realizadas, lo cual se encuentra detallado en la figura 54.

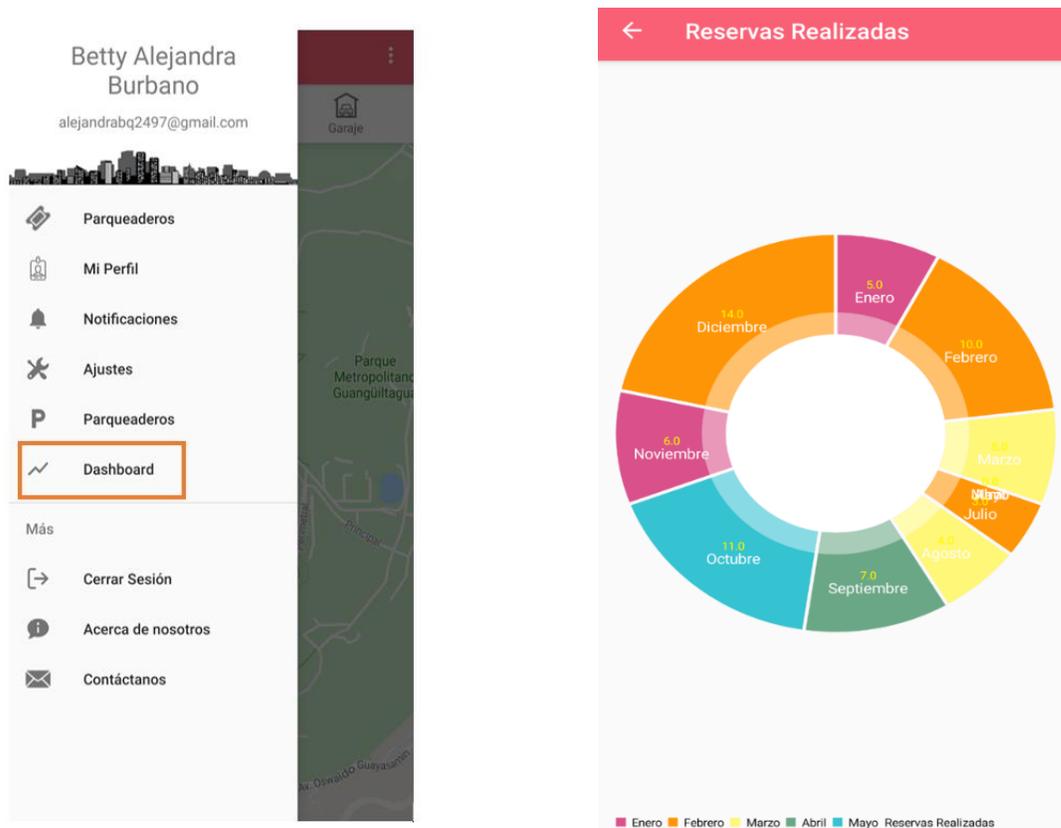


Figura 53. Gráfico de reservas realizadas.

5.1.15. Décima quinta historia de usuario: Análisis de Reservas

Un usuario puede ver la cantidad de reservas que se han realizado en sus parqueaderos en cada mes en la opción *Dashboard* del menú lateral y seleccionando la opción Reservas. El gráfico de las reservas recibidas se encuentra en la figura 55.

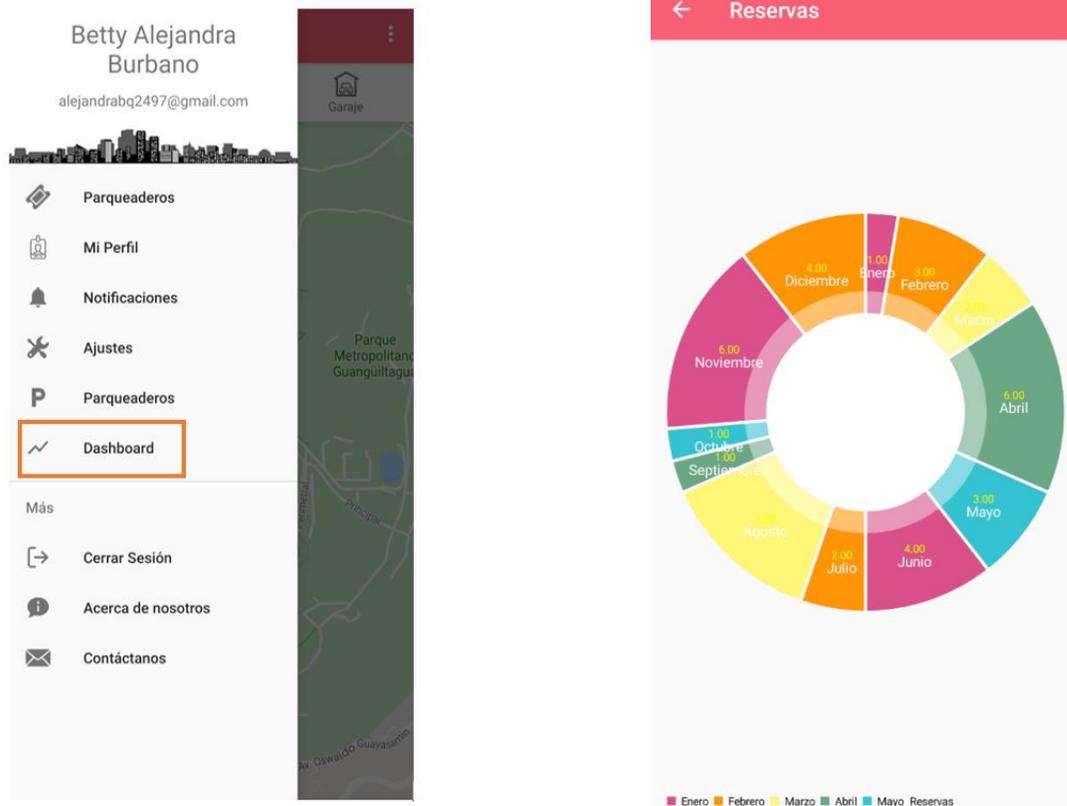


Figura 54. Gráficos de reservas recibidas.

5.1.16. Décima sexta historia de usuario: Análisis de Gastos

Un usuario puede ver los gastos que ha tenido haciendo reservas en cada mes en la opción *Dashboard* del menú lateral y seleccionando la opción Gastos, lo cual se visualiza en la figura 56.

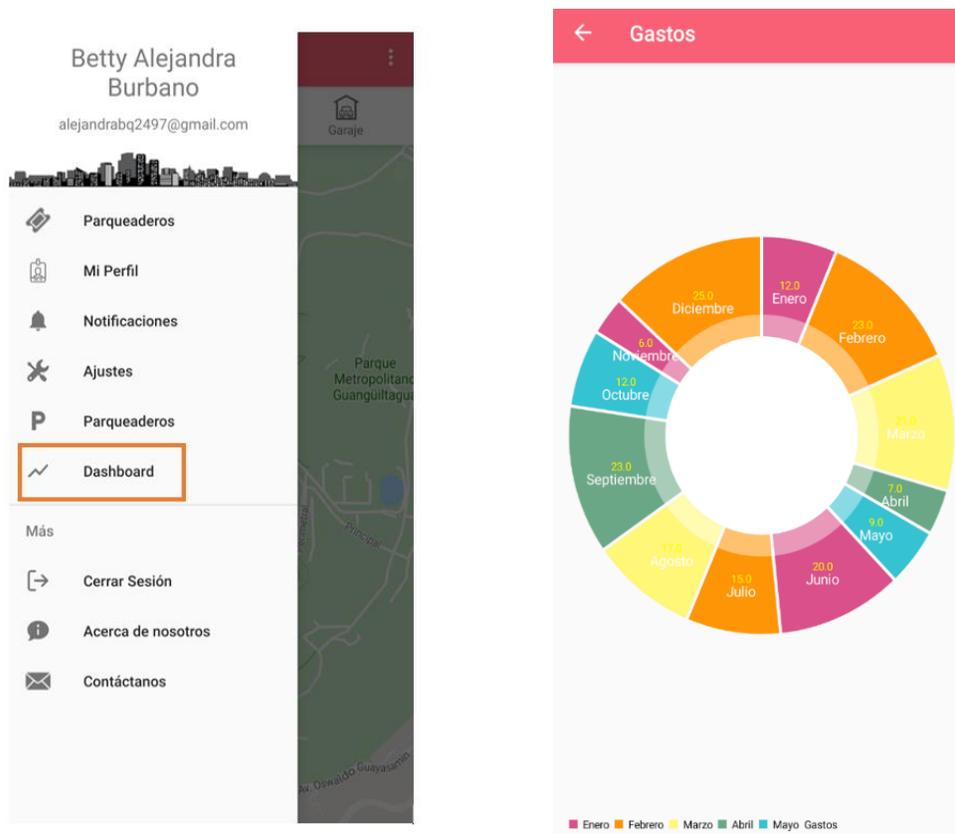


Figura 55. Gastos realizados.

5.1.17. Décima sexta historia de usuario: Análisis de Ganancias

Un usuario puede ver las ganancias que ha obtenido reservando sus parqueaderos en cada mes en la opción *Dashboard* del menú lateral y seleccionando la opción Ganancias, esto se visualiza en la figura 57.

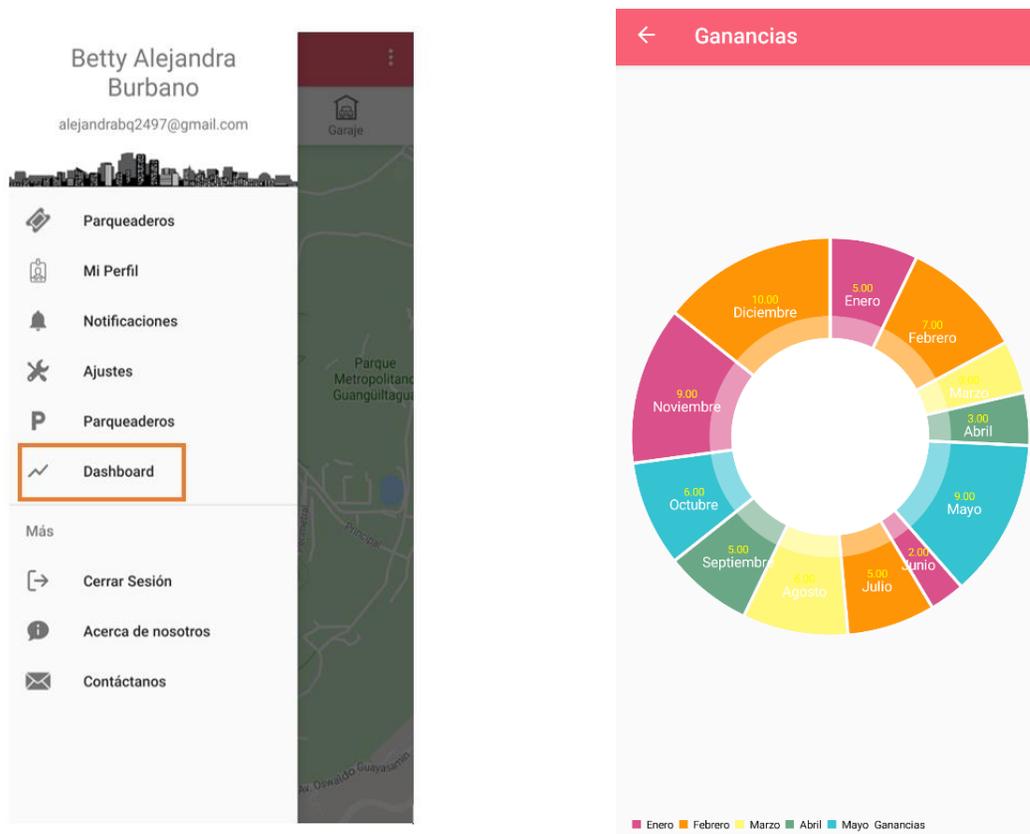


Figura 56. Ganancias generadas.

5.2. Análisis de Pruebas

Mediante las pruebas realizadas en la sección anterior se pudo verificar el buen funcionamiento de la aplicación móvil, teniendo como resultado el almacenamiento correcto de la información ingresada por los usuarios de forma dinámica en la base de datos que se encuentra en la nube. De la misma manera la transaccionalidad realizada entre los usuarios registrados para el alquiler y oferta de parqueaderos se realiza de buena manera.

Lo único que se encontró como inconveniente a la hora de utilizar la aplicación es que la misma solo podrá ser utilizada localmente, es decir que los usuarios no podrán descargar la API desde Google Play debido a que los desarrolladores deben

pagar una suscripción previa, para poder utilizar el servicio de la tienda, lo cual se puede visualizar en la figura 58.

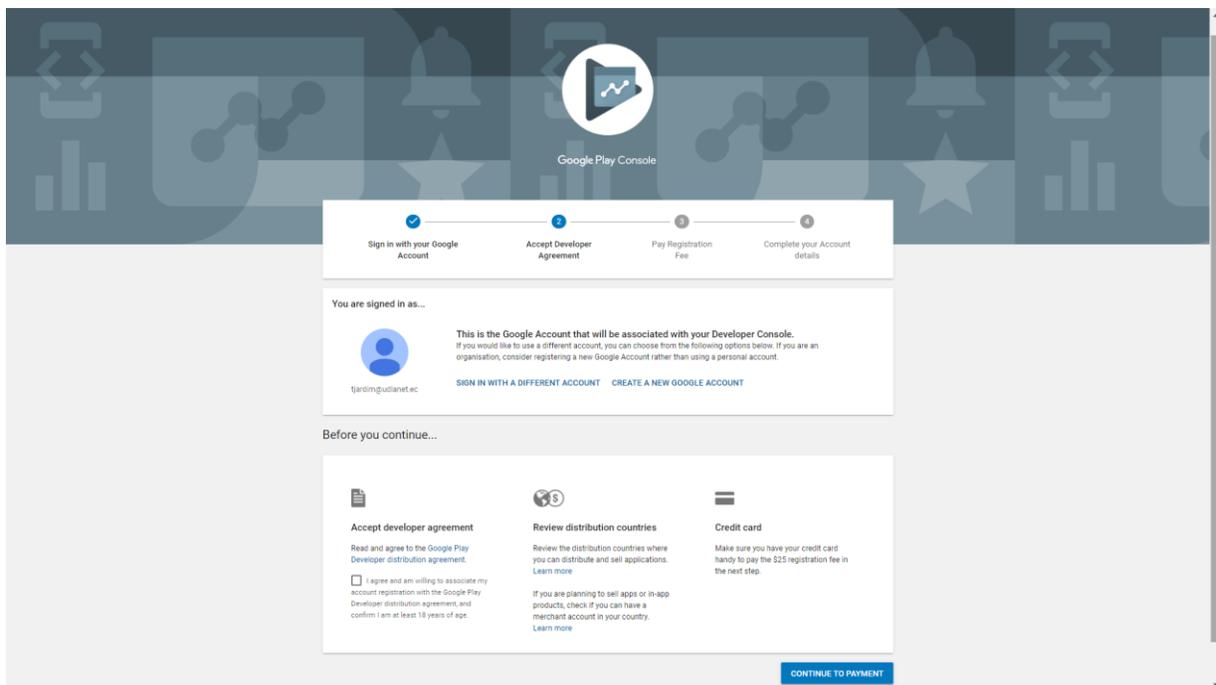


Figura 57. Suscripción a Google Play.

5.2.1. Análisis 1: Análisis de Requerimientos no Funcionales

El funcionamiento de la aplicación se encuentra acorde con los requerimientos no funcionales planteados anteriormente en este documento.

El prototipo es de fácil uso ya que, en su etapa de desarrollo se ha procurado disminuir la mayor cantidad de ventanas y pestañas para dar al usuario una aplicación de uso simple, la interfaz de ésta junto con el uso de colores pasteles de baja intensidad y los elementos de diseño, como imágenes y gráficos dan una apariencia agradable a la vista.

Se ha utilizado una gran cantidad de cuadros y mensajes que indican información adicional al usuario en el momento en que se encuentre realizando una acción.

Se han plasmado varias funciones dentro del uso de la aplicación móvil para garantizar que datos sensibles no se encuentren expuestos como números de teléfonos, contraseñas e identificaciones de otros usuarios, de igual manera se han desarrollado varios mecanismos de recarga de la aplicación en casos de modificación de datos para evitar incoherencias debido al manejo de datos caches.

Por seguridad se ha implementado un método de encriptación de contraseñas basado Hash para garantizar la integridad de la información de los usuarios.

La aplicación puede ser utilizada en cualquier teléfono móvil que tenga un API mayor o igual a 25 ya que las tecnologías utilizadas son compatibles con dichas características.

5.2.2. Análisis 2: Tiempos de Respuesta

La latencia del prototipo es aceptable, las solicitudes y respuestas a la base de datos se realizan de manera rápida, cabe recalcar que el funcionamiento de la app será óptimo con teléfonos móviles con un sistema operativo posterior al API 27 debido a la compatibilidad de los mismos con las distintas librerías utilizadas en la aplicación móvil, esto fue realizado considerando las constantes actualizaciones de las librerías utilizadas por Android Studio, principalmente las relacionadas con el API de Google Maps.

Para mejorar la experiencia del usuario se ha considerado el uso de *Shared Preferences* y datos caches para el almacenamiento de información de poca modificación, esto permite aumentar la velocidad de la respuesta a varias acciones debido a que se limita el acceso a la base de datos.

5.2.3. Análisis 3: Seguridad de uso

Por seguridad se ha agregado información de contacto a la aplicación móvil en caso de que cualquier usuario requiera contactarse con otro, al momento de realizar una reserva.

De la misma manera, los datos personales ingresados son encriptados con SHA, es decir que a la hora de revisar dicha información en la base de datos esa información no se puede visualizar como lo colocan los usuarios realmente.

5.2.4. Análisis 4: Manejo de interfaz

El mapa fue desarrollado con la finalidad de que sea manejado de manera simple, se implementó diferentes APIs y librerías para asemejar su imagen a la aplicación Google Maps, así los usuarios pueden estar más familiarizados con el aspecto del prototipo.

Los gráficos estadísticos usados para el análisis de información fueron implementados de tal forma que el usuario pueda visualizar fácilmente sus movimientos mensuales y de manera automática, no habrá necesidad de que el usuario realice acciones extras.

6. CONCLUSIONES Y RECOMENDACIONES

6.1. Conclusiones

Carpark, el prototipo de aplicación móvil; fue exitosamente desarrollado entorno a la implementación de una forma simple y sencilla de búsqueda, oferta y demanda de parqueaderos en el Distrito Metropolitano de Quito para solventar uno de sus mayores problemas, la dificultad para encontrar parqueaderos.

Se diseñó una aplicación enfocada a las necesidades y problemática que presentan los quiteños, pero se debe dejar claro que la misma solo puede ser utilizada por personas que sean mayores de edad, que cuenten con un automóvil y/o con un puesto de parqueadero.

El prototipo móvil fue desarrollado con características mínimas de seguridad, lo que quiere decir que en cuanto a la seguridad de información llega netamente hasta almacenamiento de datos y transaccionalidad entre usuarios, no tiene nada que ver con las medidas de seguridad que deben tener los usuarios después de haber utilizado la aplicación, por lo que la misma no se hace responsable de daños y pérdidas que presenten los clientes al momento de rentar parqueaderos.

La sección *Dashboard* del prototipo está enfocada en las transacciones que realiza cada usuario diariamente, las mismas pueden ser visualizadas de forma mensual con el objetivo de determinar ganancias y gastos, lo cual influye en el dinero del usuario, puesto que puede aprovechar el uso de la aplicación para ganar dinero de forma rápida y segura.

La precisión funcional del aplicativo depende de las prestaciones del teléfono móvil del usuario ya que usa características físicas como GPS y memoria RAM, por lo tanto, el rendimiento de la aplicación móvil dependerá en cierta medida del tipo de teléfono móvil.

Para preservar la integridad de la información del usuario en el proceso de modificación de datos se debe actualizar los archivos caches manejados por la aplicación para evitar incongruencias, por tal motivo el usuario debe volver a iniciar sesión una vez realizada la acción mencionada, de igual forma los datos modificados dentro de la aplicación se visualizan una vez que ésta se recarga.

Para evitar que los usuarios realicen un mal uso de la aplicación, en el proceso de eliminación de una solicitud de reserva, se debe realizar la acción de dos pasos de verificación, con el objetivo de poder garantizar que el usuario haya creado la reserva por un motivo acorde a la finalidad de aplicación, si esta doble acción no se realiza se considerará el proceso como una falsa reserva.

Por temas relacionados a costos el prototipo de aplicación móvil será presentado como un archivo en formato *.apk* puesto que la creación de una cuenta de desarrollador en la plataforma Google Play tiene un costo de USD 25.00 y la finalidad del proyecto se centra en el desarrollo de un modelo previo al de producción.

El proceso de diseño e implementación de una aplicación móvil es un trabajo que necesita de un aporte riguroso, tomando en cuenta los requerimientos de los usuarios, y la forma en la que los programadores los entienden para plasmarlos de la mejor manera posible.

Con las encuestas realizadas se sabe que los puntos fuertes del aplicativo móvil son su enfoque al sistema operativo Android debido a la gran demanda de éste, la prestación del mercado al uso de aplicaciones móvil realizadas por estudiantes universitarios, la dificultad al momento de conseguir una plaza de parqueo en la ciudad de Quito y la disposición de obtener ganancias a raíz de un parqueadero propio poco usado.

6.2. Recomendaciones

Para versiones posteriores del prototipo el enfoque se centrará en la evaluación completa de la viabilidad, funcionamiento, producción y distribución de la aplicación móvil, por tal razón esta será publicada en la plataforma Google Play.

En futuros procesos de producción y distribución del aplicativo móvil se desarrollará un prototipo orientado a teléfonos móviles que utilizan el sistema operativo iOS con la finalidad de cubrir una mayor parte del mercado.

Se debería contratar un servicio de Cloud pagado, para disminuir la latencia existente en las transacciones que se realizan con la aplicación, ya que la utilizada soporta un límite de usuarios, que no abarca el Distrito Metropolitano de Quito en sus horas pico.

Para expandir el mercado objetivo se puede realizar convenios con instituciones públicas como el municipio del Distrito Metropolitano de Quito para implementar una forma de más sencilla de control de parqueaderos y extender el uso del aplicativo móvil.

Para futuras mejoras a la aplicación el proceso de eliminación de reserva será monitoreado con mayor énfasis, se realizarán análisis de falsas reservas junto con los usuarios que las realizan, en caso de que realicen esta acción constantemente será respectivamente notificados, adicional, se pedirá una mayor especificación para el motivo por el cual se realizan cancelaciones de reservas, estos datos podrán ser usados en futuros análisis que ayudarán al crecimiento de la aplicación y mejorarán la experiencia de usuario.

Durante el proceso de pruebas se puede recomendar para futuros proyectos de la misma índole aplicar técnicas como el Grupo Focal para la verificación de la funcionalidad e interfaz de usuario.

REFERENCIAS

Android 4.4 API | Android Developers. (s.f.). Recuperado el 17 de diciembre de 2019 de <https://developer.android.com/about/versions/android-4.4?hl=es-419>

android.support.v4.content.FileProvider not found. (2018). Recuperado el 5 de octubre de 2019 de <https://stackoverflow.com/questions/48534293/android-support-v4-content-fileprovider-not-found>

Attempt to get length of null array error, How to solve? (2016). Recuperado el 2 de noviembre de 2019 de <https://stackoverflow.com/questions/37258874/attempt-to-get-length-of-null-array-error-how-to-solve>

Barker, R. (1994). El modelo entidad-relación CASE*methodtm. Madrid, España: Addison-Wesley: Diaz de Santos.

Calleja, M. A. (s.f.). Tecnologías de Programación Integrativas. Madrid: Editorial Universitaria Ramón Areces.

Celaya, L. A. (2014). Cloud: herramientas para trabajar en la nube (2ª ed.). Málaga, España: Editorial ICB.

Conmicelu. (2018). Red de Parqueaderos o estacionamientos públicos en Quito. Recuperado el 28 de octubre de 2019 de <http://www.conmicelu.com/tramites/red-de-parqueaderos-o-estacionamientos-publicos-en-quito/>

Crockford, D. (s.f.). JSON. Recuperado el 28 octubre de 2019 de <https://json.org>

Didn't find class "androidx.core.content.FileProvider". (2018). Recuperado el 7 de diciembre de 2019 de <https://stackoverflow.com/questions/52696574/didnt-find-class-androidx-core-content-fileprovider>

DiMarzio, J., & Lee, W. (2016). *Beginning Android Programming with Android Studio*. Indianapolis, Estados Unidos: Wiley.

Download Android Studio and SDK tools | Android Developers. (s.f.). Recuperado el 12 de octubre de 2019 de https://developer.android.com/studio/?gclid=Cj0KCQjw0IDtBRC6ARIsAIA5gWvwxcTXQiBGH4TKwuAFubjeenZa8UKghzUm-C9erC4_oNhFXTJWmpEaAgE4EALw_wcB

Drawables overview | Android Developers. (s.f.). Recuperado el 22 de diciembre de 2019 de <https://developer.android.com/guide/topics/graphics/drawables>

El Comercio. En Pichincha hay 2 autos por cada 10 personas desde el 2016 (2018) Recuperado el 10 de abril de 2019 de <https://www.elcomercio.com/datos/pichincha-tasa-autos-personas-vehiculos.html>

El Comercio. Parqueaderos de borde con más demanda. (s.f.). Recuperado el 10 de abril de 2019 de <https://www.elcomercio.com/datos/pichincha-tasa-autos-personas-vehiculos.html>

El Telégrafo - Noticias del Ecuador y del mundo. (2018). Parquear es difícil y caro en Quito. Recuperado el 28 de octubre de 2019 de <https://www.eltelegrafo.com.ec/noticias/punto/1/parquear-es-dificil-y-caroen-quito>

El Universo. Conductores de Quito y Guayaquil pasan más de 160 horas al año atrapados en el tráfico, dice estudio (2019). Recuperado el 10 de abril de 2019 de <https://www.eluniverso.com/noticias/2019/12/20/nota/7659138/conductores-quito-guayaquil-pasan-mas-160-horas-atrapados-trafico>

Francisco Durán, F. G. (2007). Programación Orientada a Objetos con Java. Editorial Paraninfo.

Friesen, J. (2019). Tutorial series: Android Studio for beginners: A complete beginner's guide to Android Studio, from installing Android Studio to coding and debugging your first Android app. Includes source code for the example app. Recuperado el 3 de octubre de 2019 de <https://search-proquest-com.bibliotecavirtual.udla.edu.ec/docview/2181634133?accountid=33194>

Galiano, J. (2016). Implantar scrum con éxito. Barcelona, España: Editorial UOC.

Google Play Console (s.f). Recuperado el 12 de octubre de 2019 de <https://play.google.com/apps/publish/signup/#>

Hohensee, B., & Hidalgo, I. (2014). Introducción A Android Studio. Incluye Proyectos Reales Y El Código Fuente. Nueva Jersey, Estados Unidos: Babelcube.

How To Developers: Android Studio: A Platform for Android Development. (2015, 1 octubre). Recuperado el 3 de octubre de 2019 de <https://search-proquest-com.bibliotecavirtual.udla.edu.ec/docview/1719231971?accountid=33194>

Imagenes, Botones, Sonidos y Share en Android. (2018). Recuperado el 4 de diciembre de 2019 de <https://es.stackoverflow.com/questions/201413/imagenes-botones-sonidos-y-share-en-android?noredirect=1>

INCIBE. (2017). Cloud computing: una guía de aproximación para el empresario. Recuperado el 27 octubre de 2019 de <https://www.incibe.es/protege-tu-empresa/blog/cloud-computing-guia-aproximacion-el-empresario>

INEC. Tras las cifras de Quito (2017), Recuperado el 10 de abril de 2019 de <https://www.ecuadorencifras.gob.ec/tras-las-cifras-de-quito/>

Leal, B. (2017). Metodología Scrum en proyectos digitales - Blog IDA Chile | Estrategia para el éxito de tu negocio. Recuperado el 27 de octubre de 2019 de <https://blog.ida.cl/estrategia-digital/metodologia-scrum-en-proyectos-digitales/>

Casillas, L., Gibert, M., Pérez, Ó.(s.f.). Bases de datos en MySQL.

Mell, P., & Grance, T. (2011). The NIST definition of cloud computing.

NeoAttack. (2019). ¿Qué es un Dashboard y para qué sirve? - Neo Wiki | NeoAttack. Recuperado el 6 de febrero de 2020 de <https://neoattack.com/neowiki/dashboard/>

Newer versions of Android Studio and only two drawable directory - drawable and drawable-v21. (2015). Recuperado el 17 de octubre de 2019 de <https://stackoverflow.com/questions/34343611/newer-versions-of-android-studio-and-only-two-drawable-directory-drawable-and>

Openclassrooms. (2017). En qué consiste el modelo en cascada. Recuperado el 27 de octubre de 2019 de <https://openclassrooms.com/en/courses/4309151-gestiona-tu-proyecto-de-desarrollo/4538221-en-que-consiste-el-modelo-en-cascada>

Oracle. (2018). Descarga gratuita de software de Java. Recuperado el 27 de octubre de 2019 de <https://www.java.com/es/download/>

Oracle. (2019). MySQL. Recuperado el 28 de octubre de 2019 de <https://www.mysql.com>

Ordenanza Metropolitana (2017). Recuperado el 10 de abril de 2019 de http://www7.quito.gob.ec/mdmq_ordenanzas/Ordenanzas/ORDENANZAS%

20MUNICIPALES%202017/ORDM%20-%2020170%20%20%20%20%20%20%20%20Sistema%20de%20Estacionamientos%20y%20Terminales%20Terrestres%20del%20DMQ..pdf

Orozco, M. (2017). El parque automotor aumenta y complica más la movilidad. Recuperado el 28 de octubre de 2019 de <https://www.elcomercio.com/actualidad/aumento-parque-automotor-quito-movilidad.html>

Otero, M. F., & Huerga, M. A. N. (2014). Sistemas de gestión de relaciones con clientes en las empresas (CRM). Barcelona, España: Universidad de Alcalá, Servicio de Publicaciones.

Parkcero (2017). Recuperado el 10 de abril de 2019 de <https://www.parkcero.com/>

Recursos de elementos de diseño | Android Developers. (s.f.). Recuperado el 18 de noviembre de 2019 de <https://developer.android.com/guide/topics/resources/drawable-resource?hl=es-419>

Roser, C. (2016). The Key to Lean – Plan, Do, Check, Act! | AllAboutLean.com. Recuperado el 27 de octubre de 2019 de <https://www.allaboutlean.com/pdca/>

Socconini, L., & Reato, C. (2019). Lean Six Sigma. Sistema de gestión para liderar empresas. Barcelona, España: ICG Marge, SL.

Sommerville, I., & Galipienso, M. (2005). Ingeniería del software (7ª ed.). Madrid, España: Pearson Educación.

Tecniseguros. (2019). Estadísticas sorprendentes de autos en Ecuador. Recuperado el 28 de octubre de 2019 de <https://www.tecniseguros.com.ec/blog/vehiculos/estadisticas-de-autos/>

Why do we use drawable v24 in android? (2018). Recuperado el 25 de octubre de 2019 de <https://stackoverflow.com/questions/50849723/why-do-we-use-drawable-v24-in-android>

Yener, M., Dundar, O., & Hellman, E. (2016). Expert Android Studio. Indianapolis, Estados Unidos: Wiley.

ANEXOS

ÍNDICE DE ANEXOS

Anexo 1. Encuesta	1
Anexo 2. Resultados tabulados de la encuesta.....	3
Anexo 3. Diagrama Entidad – Relación.....	7
Anexo 4. Tabla Segunda Historia de Usuario.....	8
Anexo 5. Tabla Tercera Historia de Usuario.....	8
Anexo 6. Tabla Cuarta Historia de Usuario	9
Anexo 7. Tabla Quinta Historia de Usuario.	9
Anexo 8. Tabla Sexta Historia de Usuario.....	10
Anexo 9. Tabla Séptima Historia de Usuario.....	10
Anexo 10. Tabla Octava Historia de Usuario.....	11
Anexo 11. Tabla Novena Historia de Usuario.....	11
Anexo 12. Tabla Décima Historia de Usuario.....	12
Anexo 13. Tabla Décimo - Primera Historia de Usuario	12
Anexo 14. Tabla Décimo - Segunda Historia de Usuario	13
Anexo 15. Tabla Décimo - Tercera Historia de Usuario.	13
Anexo 16. Tabla Décimo - Cuarta Historia de Usuario.....	14
Anexo 17. Tabla Décimo - Quinta Historia de Usuario	14
Anexo 18. Tabla Décimo - Sexta Historia de Usuario	15
Anexo 19. Tabla Décimo - Séptima Historia de Usuario	15
Anexo 20. Diagrama de Flujo y Tablas de Segunda Historia de Usuario.....	16
Anexo 21. Diagrama de Flujo y Tabla de Tercera Historia de Usuario.....	17
Anexo 22. Diagrama de Flujo y Tabla de Cuarta Historia de Usuario.	18
Anexo 23. Diagrama de Flujo y Tablas de Quinta Historia de Usuario.....	19
Anexo 24. Diagrama de Flujo y Tablas de Sexta Historia de Usuario.	20
Anexo 25. Diagrama de Flujo y Tablas de Séptima Historia de Usuario.	21
Anexo 26. Diagrama de Flujo y Tablas de Octava Historia de Usuario.....	22
Anexo 27. Diagrama de Flujo y Tabla de Novena Historia de Usuario.....	23
Anexo 28. Diagrama de Flujo y Tablas de Décima Historia de Usuario.	24

Anexo 29. Diagrama de Flujo y Tablas de Décima-Primera Historia de Usuario. .	25
Anexo 30. Diagrama de Flujo y Tablas de Décima-Segunda Historia de Usuario.	26
Anexo 31. Diagrama de Flujo y Tabla de Décima-Tercera Historia de Usuario. ...	27
Anexo 32. Diagrama de Flujo y Tablas de Decimacuarta Historia de Usuario.	28
Anexo 33. Diagrama de Flujo y Tablas de Décima-Quinta Historia de Usuario. ...	29
Anexo 34. Diagrama de Flujo y Tablas de Décima-Sexta Historia de Usuario.....	30
Anexo 35. Diagrama de Flujo y Tablas de Décima-Séptima Historia de Usuario.	31

Anexo 1. Encuesta

Carpark!

CARPARK, es una aplicación móvil enfocada a usuarios en general con problemas de encontrar un lugar para parquear dentro de la ciudad de Quito, donde además de poseer una interfaz interactiva, rápida y sencilla logra conectar fácilmente con la necesidad del usuario, así mismo de dar oportunidad de que los usuarios moneticen los lugares de su propiedad de los cuales no hacen uso.

¿Con cuál sistema operativo se familiariza más al utilizar un dispositivo inteligente? *

- Android
- iOS
- Windows Phone

¿Estaría dispuesto a utilizar una aplicación desarrollada por estudiantes? *

- Sí
- No
- Tal vez

¿Qué tipo dispositivo utiliza más? *

- Smartphone
- Tablet

¿Estaría dispuesto a descargar una aplicación móvil que le facilita la búsqueda de parqueaderos? *

- Sí
- No
- Tal vez

¿Es difícil para usted encontrar un parqueadero en la ciudad de Quito? *

- Sí
- No
- Tal vez

¿Diariamente por cuánto tiempo utiliza un puesto de parqueadero? *

- 1 a 3 horas
- 4 a 6 horas
- 6 o más horas

¿Qué tipo de parqueadero utiliza usualmente? *

- Público
- Privado (de terceros)
- Propio

¿Cuánto tiempo gasta a diario ubicando un lugar de parqueadero? *

- 5 a 7 minutos
- 10 a 15 minutos
- Más de 15 minutos

¿Le resultaría molesto arrendar su puesto de parqueadero a extraños? *

- Sí
- No
- Tal vez

¿No encontrar un puesto de parqueadero cuanto estrés le genera? *

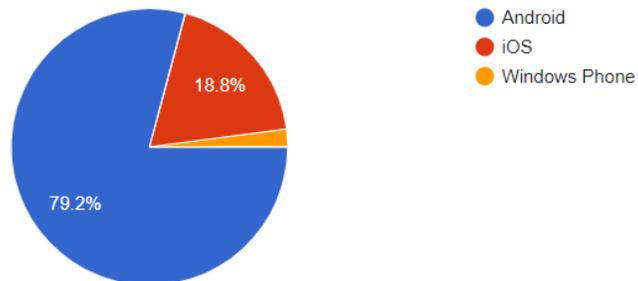
1 2 3 4 5

Nada de estrés Mucho estrés

Anexo 2. Resultados tabulados de la encuesta

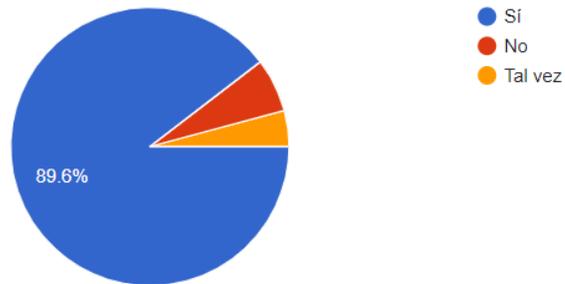
¿Con cuál sistema operativo se familiariza más al utilizar un dispositivo inteligente?

48 responses



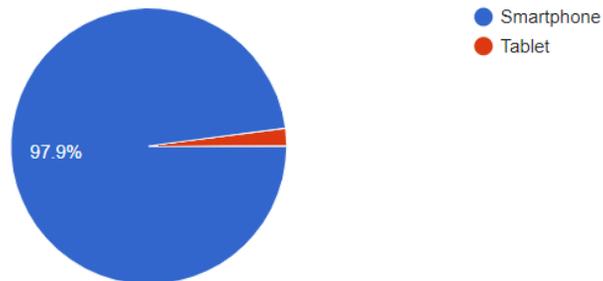
¿Estaría dispuesto a utilizar una aplicación desarrollada por estudiantes?

48 responses



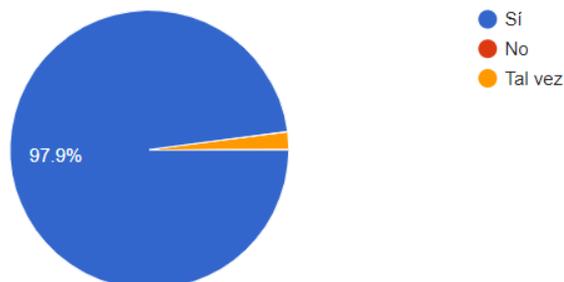
¿Qué tipo dispositivo utiliza más?

48 responses



¿Estaría dispuesto a descargar una aplicación móvil que le facilita la búsqueda de parqueaderos?

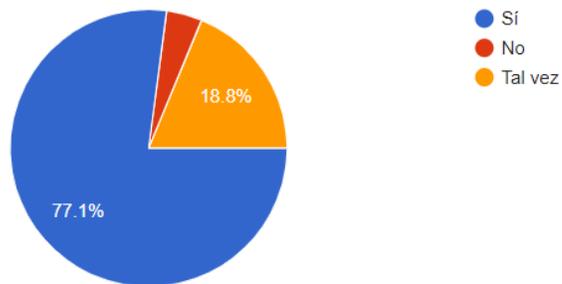
48 responses



¿Es difícil para usted encontrar un parqueadero en la ciudad de Quito?

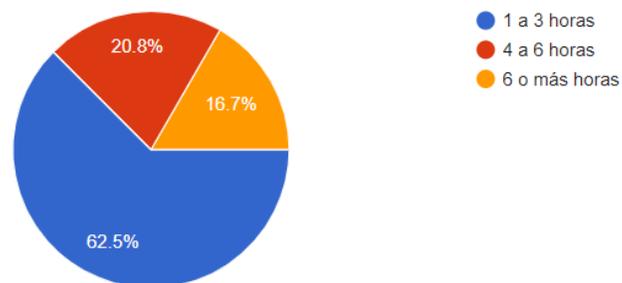


48 responses



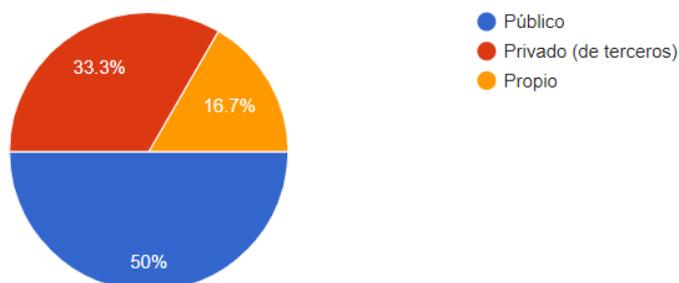
¿Diariamente por cuánto tiempo utiliza un puesto de parqueadero?

48 responses



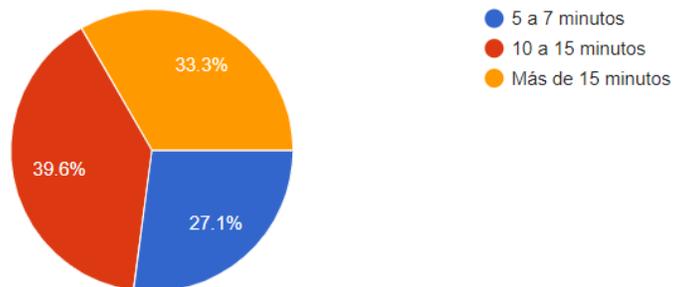
¿Qué tipo de parqueadero utiliza usualmente?

48 responses



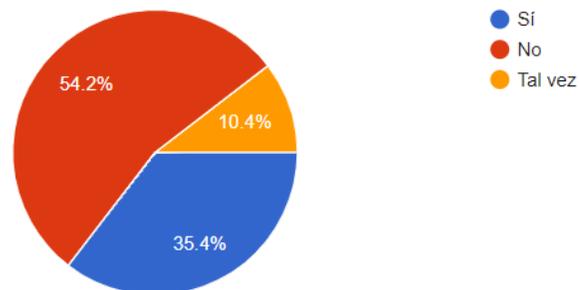
¿Cuánto tiempo gasta a diario ubicando un lugar de parqueadero?

48 responses



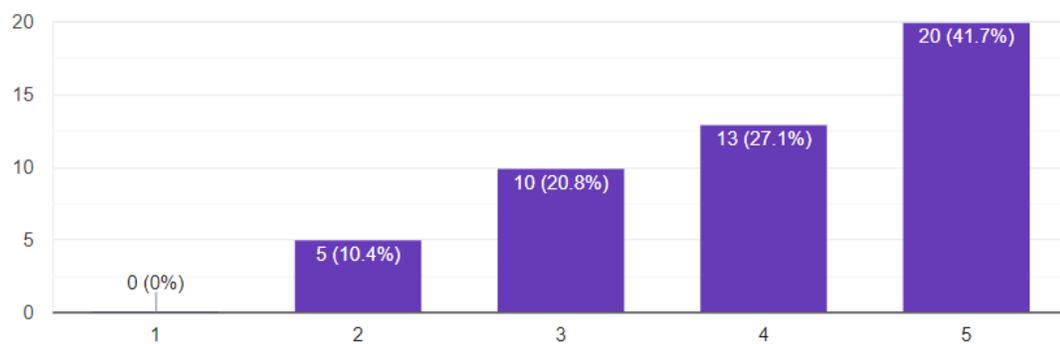
¿Le resultaría molesto arrendar su puesto de parqueadero a extraños?

48 responses

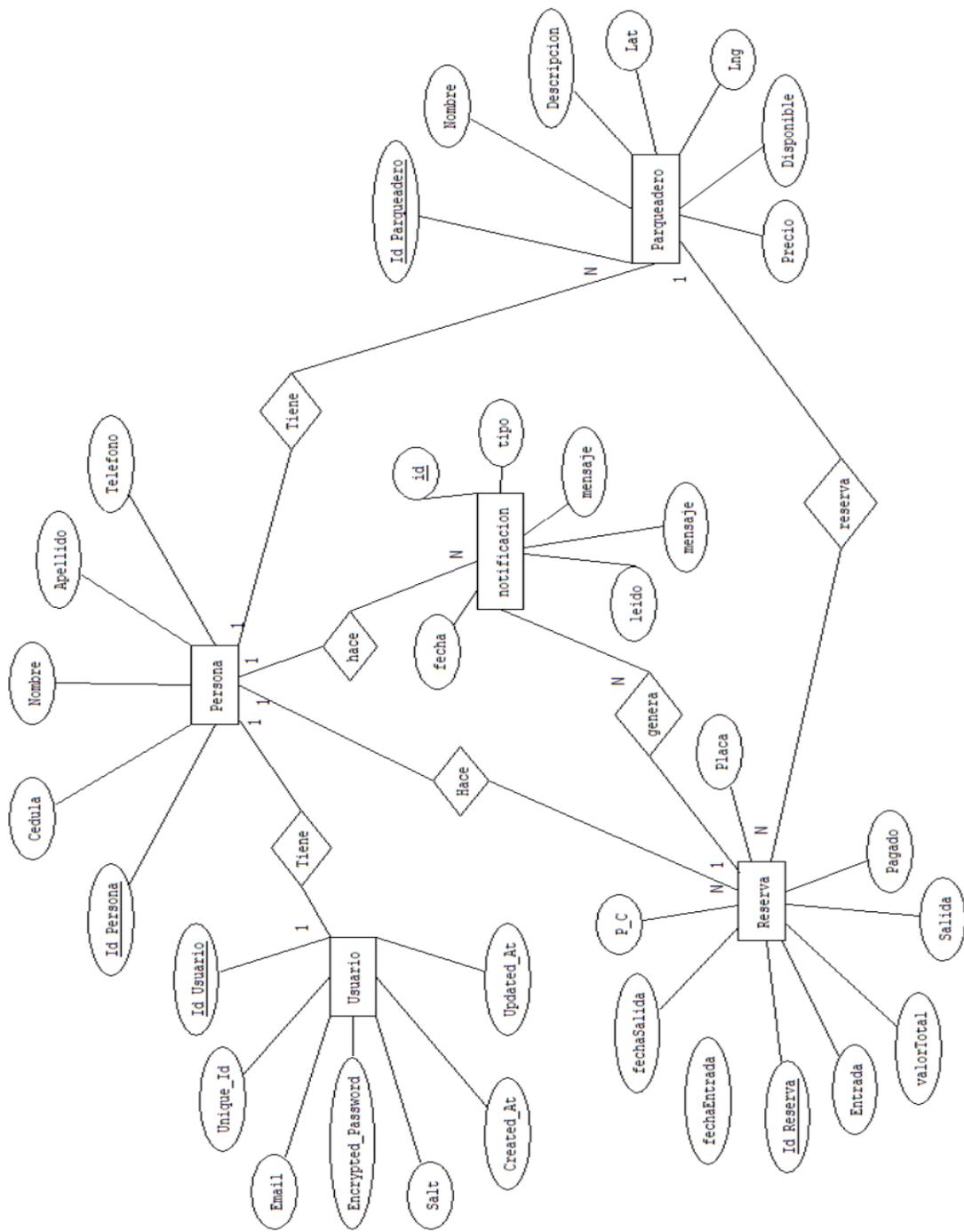


¿No encontrar un puesto de parqueadero cuanto estrés le genera?

48 responses



Anexo 3. Diagrama Entidad – Relación



Anexo 4. Tabla Segunda Historia de Usuario

Historia de Usuario	
Número: 2	Usuario: Usuario
Nombre Historia: Modificar Usuario	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 1
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: Se presenta un formulario en donde el usuario puede modificar sus credenciales para utilizar la aplicación.	
Validación: El usuario modifica sus credenciales llenando los campos del formulario con la información necesaria y al presionar un botón se valida que todos los datos se encuentren correctos y que las credenciales no existan.	

Anexo 5. Tabla Tercera Historia de Usuario

Historia de Usuario	
Número: 3	Usuario: Usuario
Nombre Historia: Login Usuario	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 1
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: El usuario puede loguearse para utilizar la aplicación con las credenciales correspondientes.	
Validación: Se presenta un formulario en donde el usuario escribirá sus credenciales y una vez aplastado el botón se verifica si las credenciales concuerdan con los datos almacenados en la base de datos.	

Anexo 6. Tabla Cuarta Historia de Usuario

Historia de Usuario	
Número: 4	Usuario: Usuario
Nombre Historia: Mostrar Información Usuario	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 1
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: La información ingresada por el usuario puede ser visualizada por el mismo si lo requiere.	
Validación: Se presenta un conjunto de datos que el usuario previamente ingreso en la aplicación de forma informativa.	

Anexo 7. Tabla Quinta Historia de Usuario.

Historia de Usuario	
Número: 5	Usuario: Usuario, dueño de parqueadero
Nombre Historia: Ingresar Parqueadero	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 2
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: Se muestra un formulario en donde el usuario debe detallar la información de su parqueadero y el costo una vez seleccionada la ubicación de este.	
Validación: La información es almacenada y validada en el momento en que se presiona un botón para garantizar que no existan parqueaderos con la misma información y que los datos de este sean coherentes.	

Anexo 8. Tabla Sexta Historia de Usuario.

Historia de Usuario	
Número: 6	Usuario: Usuario, dueño de parqueadero
Nombre Historia: Modificar Parqueadero	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 2
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: El usuario puede modificar la información que ha ingresado anteriormente respecto a su parqueadero en caso de que sea necesario.	
Validación: La información es verificada para que no sea igual a ningún otro parqueadero existente y sea coherente.	

Anexo 9. Tabla Séptima Historia de Usuario

Historia de Usuario	
Número: 7	Usuario: Usuario, dueño de parqueadero
Nombre Historia: Eliminar Parqueadero	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 2
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: El usuario puede eliminar el parqueadero creado en la aplicación móvil cuando lo vea conveniente.	
Validación: Se debe recibir la confirmación del usuario para la eliminación de parqueadero.	

Anexo 10. Tabla Octava Historia de Usuario

Historia de Usuario	
Número: 8	Usuario: Usuario, dueño de parqueadero
Nombre Historia: Mostrar Parqueaderos Usuario	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 2
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: El usuario puede visualizar los parqueaderos que él ha creado en la aplicación junto con su información detallada.	
Validación: La información se muestra solo de manera informativa.	

Anexo 11. Tabla Novena Historia de Usuario

Historia de Usuario	
Número: 9	Usuario: Usuario
Nombre Historia: Mostrar Parqueaderos	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 2
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: El usuario puede visualizar los parqueaderos que se encuentren cerca del destino al cual se va a movilizar.	
Validación: La información se muestra solo de manera informativa.	

Anexo 12. Tabla Décima Historia de Usuario

Historia de Usuario	
Número: 10	Usuario: Usuario
Nombre Historia: Ingresar Reserva	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 3
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: El usuario puede reservar un parqueadero que se encuentre disponible y cerca de su lugar de destino, debe especificar el momento en que llega al parqueadero, la hora de salida y se realiza el pago de la reserva.	
Validación: El usuario debe confirmar que desea realizar la reserva y se verifica que los datos ingresados sean correctos.	

Anexo 13. Tabla Décimo - Primera Historia de Usuario

Historia de Usuario	
Número: 11	Usuario: Usuario
Nombre Historia: Eliminar Reserva	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 3
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: El usuario puede eliminar la reservar de un parqueadero si no le parece conveniente.	
Validación: El usuario debe confirmar que desea eliminar la reserva.	

Anexo 14. Tabla Décimo - Segunda Historia de Usuario

Historia de Usuario	
Número: 12	Usuario: Usuario, dueño de parqueadero
Nombre Historia: Confirmar Reserva	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 3
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: El usuario debe confirmar que se puede realizar una reserva a su parqueadero, debe verificar y aceptar la información ingresada por el usuario que desea realizar la reserva.	
Validación: El usuario debe confirmar la realización de la reserva con los datos ingresados.	

Anexo 15. Tabla Décimo - Tercera Historia de Usuario.

Historia de Usuario	
Número: 13	Usuario: Usuario
Nombre Historia: Contactar Usuario	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 3
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: El usuario puede contactar por vía telefónica al dueño del parqueadero para solventar dudas respecto a la reserva o el parqueadero, el número telefónico al cual se llama fue registrado en la aplicación por el dueño del parqueadero.	
Validación: El usuario debe confirmar la llamada con la aplicación Teléfono, la cual permite realizar llamadas telefónicas.	

Anexo 16. Tabla Décimo - Cuarta Historia de Usuario

Historia de Usuario	
Número: 14	Usuario: Usuario
Nombre Historia: Análisis de reservas realizadas	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 4
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: El usuario puede visualizar la cantidad de reservas que ha realizado en un tiempo determinado, el análisis muestra información de manera estadística.	
Validación: El análisis es mostrado de manera informativa.	

Anexo 17. Tabla Décimo - Quinta Historia de Usuario

Historia de Usuario	
Número: 15	Usuario: Usuario
Nombre Historia: Análisis de reservas	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 4
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: El usuario puede visualizar la cantidad de reservas que se han realizado en sus parqueaderos en un tiempo determinado, el análisis muestra la información de manera estadística.	
Validación: El análisis es mostrado de manera informativa.	

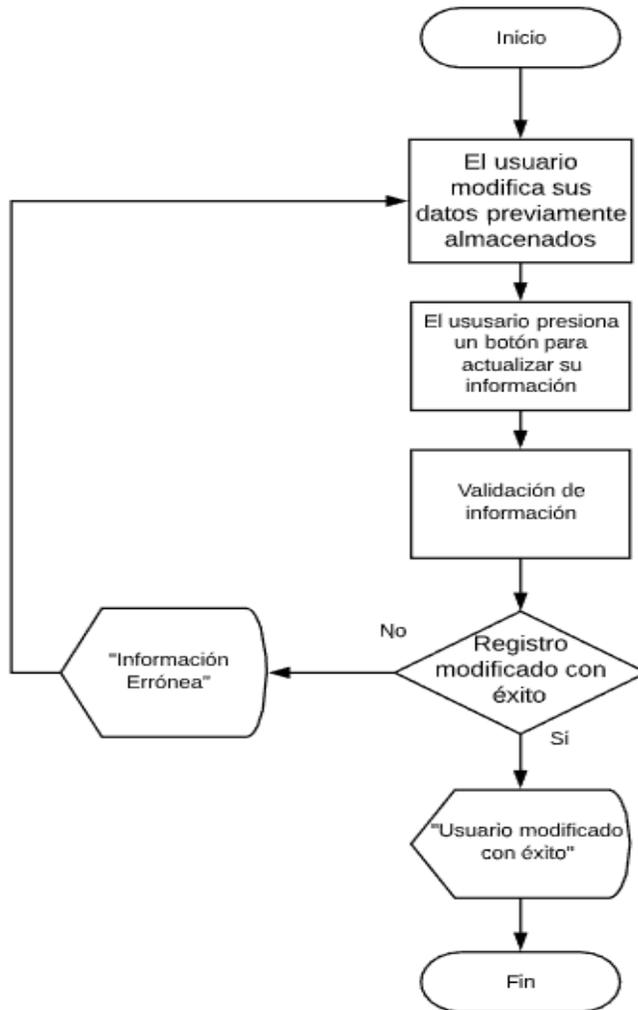
Anexo 18. Tabla Décimo - Sexta Historia de Usuario

Historia de Usuario	
Número: 16	Usuario: Usuario
Nombre Historia: Análisis de Gastos	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 4
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: El usuario puede visualizar los gastos que ha realizado en torno a las reservas que ha realizado, el análisis muestra la información de manera estadística tomando un rango de tiempo determinado.	
Validación: El análisis es mostrado de manera informativa y sirve para conocimiento del usuario.	

Anexo 19. Tabla Décimo - Séptima Historia de Usuario

Historia de Usuario	
Número: 17	Usuario: Usuario
Nombre Historia: Análisis de Ganancias	
Prioridad:	Riesgo de Desarrollo:
Puntos Estimados:	Iteración Asignada: 4
Programador Responsable: Betty Burbano, Tabata Jardim	
Descripción: El usuario puede visualizar las ganancias obtenidas por las reservas que se han realizado en sus parqueaderos, el análisis muestra la información de manera estadística tomando un rango de tiempo determinado.	
Validación: El análisis es mostrado de manera informativa para conocimiento del usuario.	

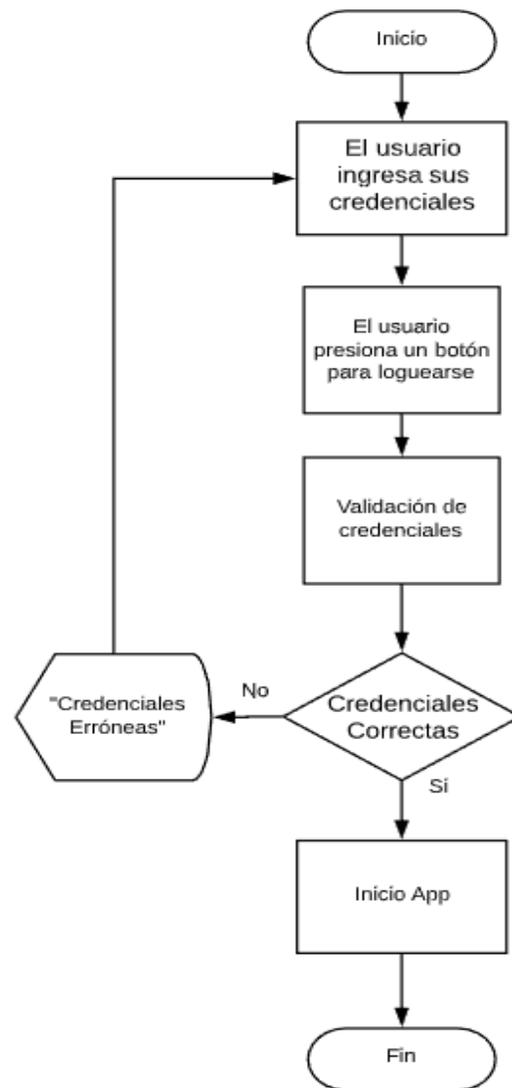
Anexo 20. Diagrama de Flujo y Tablas de Segunda Historia de Usuario.



Usuario
+ Id_Usuario: int + Unique_Id: varchar + Email: varchar + Encrypted_Password: varchar + Salt: varchar + Created_At: Datetime + Updated_At: Datetime

Persona
+ Id_Persona: int + Id_Usuario: int + Cedula: varchar + Nombre: varchar + Apellido: varchar + Telefono: varchar



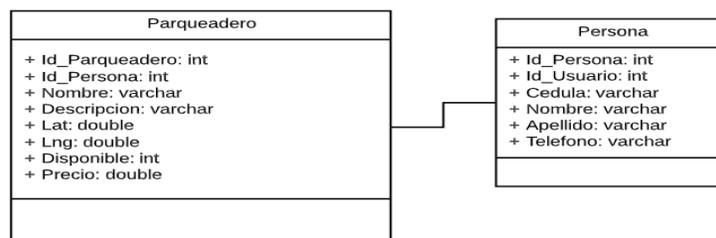
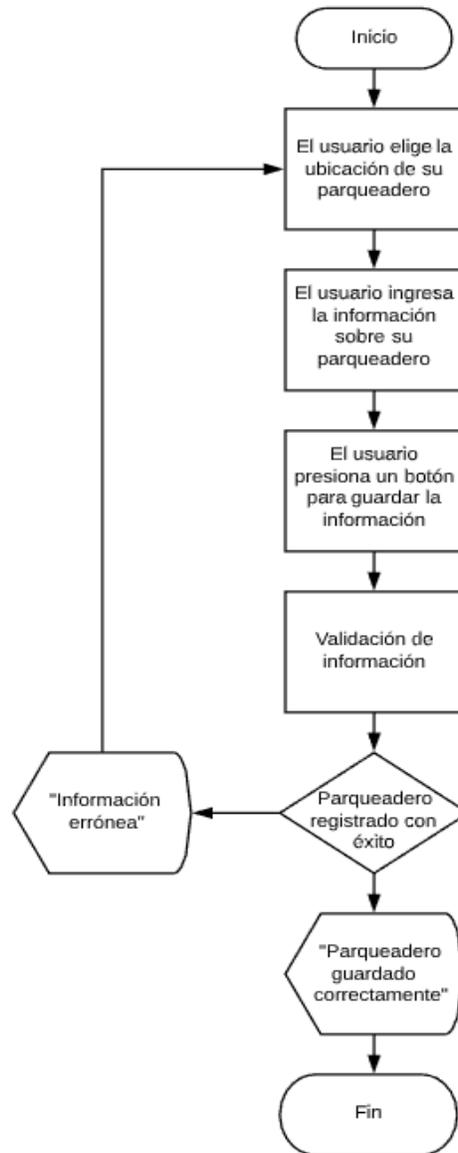
Anexo 21. Diagrama de Flujo y Tabla de Tercera Historia de Usuario.

Usuario
+ Id_Usuario: int + Unique_Id: varchar + Email: varchar + Encrypted_Password: varchar + Salt: varchar + Created_At: Datetime + Updated_At: Datetime

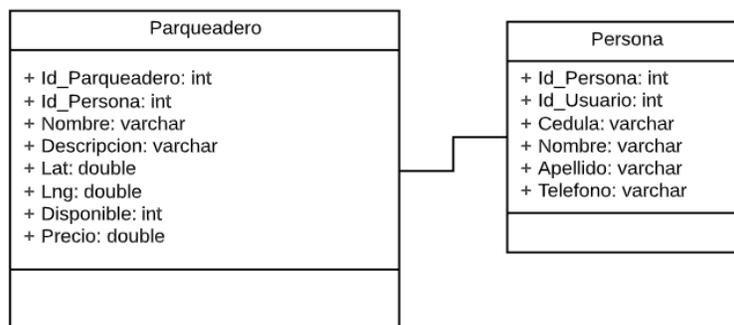
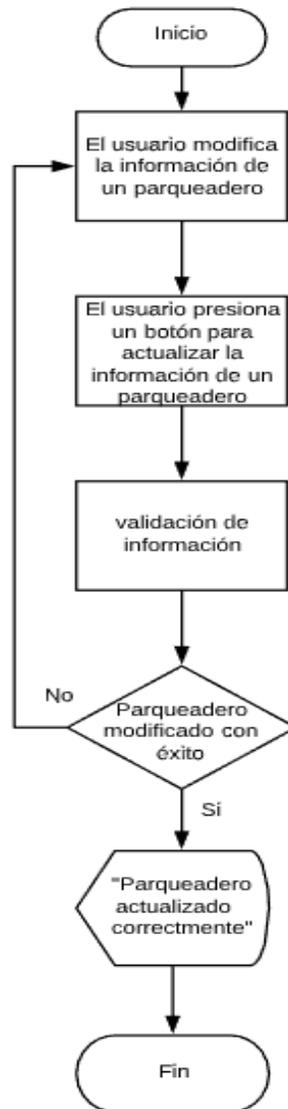
Anexo 22. Diagrama de Flujo y Tabla de Cuarta Historia de Usuario.

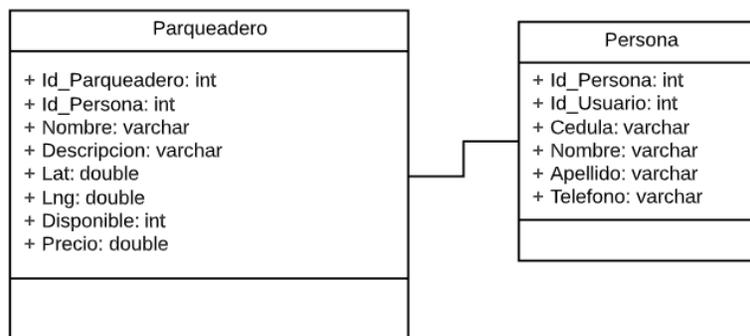
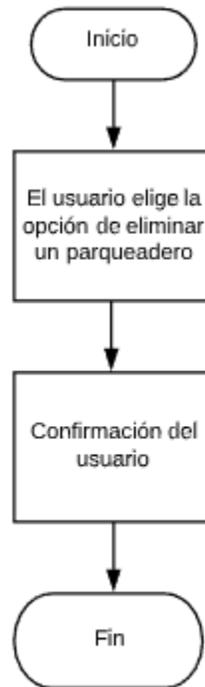
Usuario
+ Id_Usuario: int + Unique_Id: varchar + Email: varchar + Encrypted_Password: varchar + Salt: varchar + Created_At: Datetime + Updated_At: Datetime

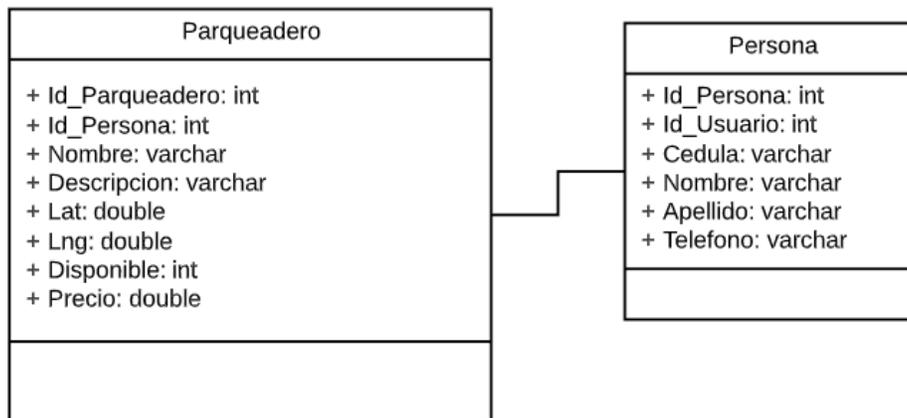
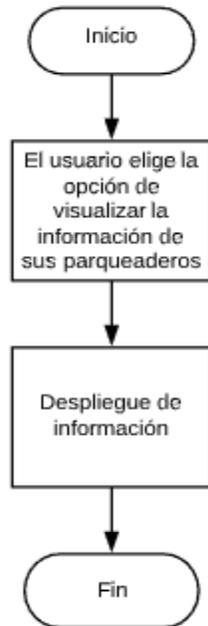
Anexo 23. Diagrama de Flujo y Tablas de Quinta Historia de Usuario.

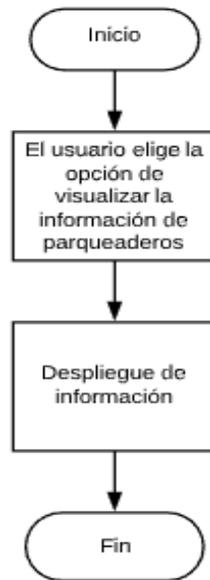


Anexo 24. Diagrama de Flujo y Tablas de Sexta Historia de Usuario.

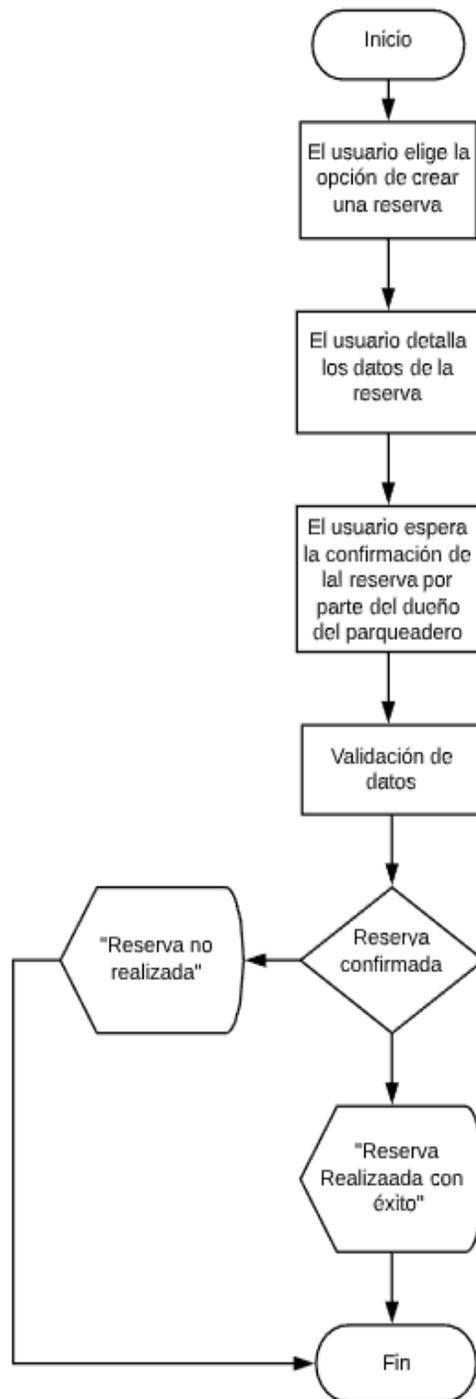


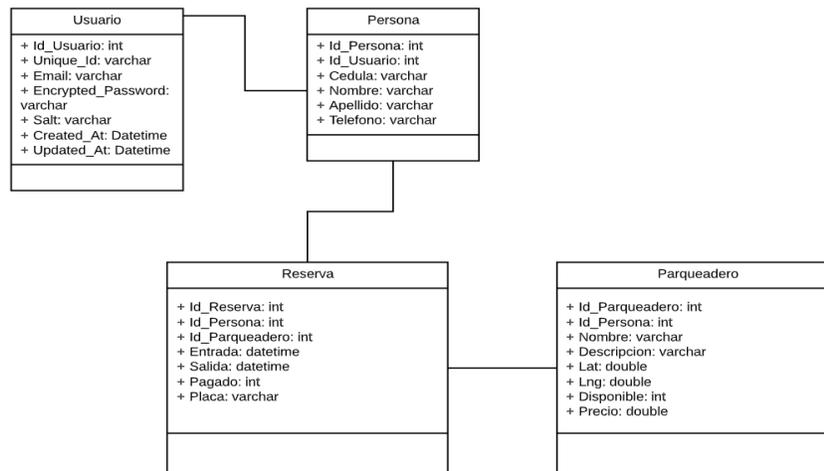
Anexo 25. Diagrama de Flujo y Tablas de Séptima Historia de Usuario.

Anexo 26. Diagrama de Flujo y Tablas de Octava Historia de Usuario.

Anexo 27. Diagrama de Flujo y Tabla de Novena Historia de Usuario.

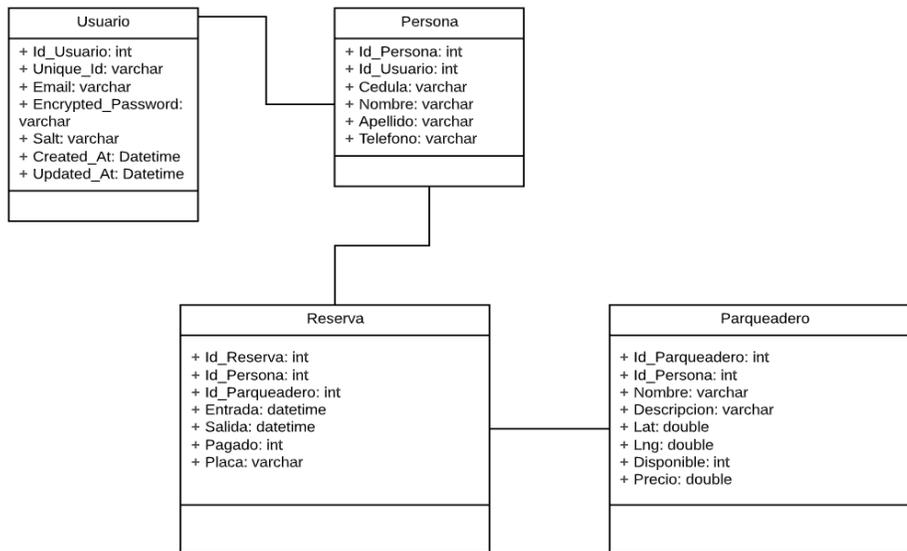
Parqueadero
+ Id_Parqueadero: int + Id_Persona: int + Nombre: varchar + Descripcion: varchar + Lat: double + Lng: double + Disponible: int + Precio: double

Anexo 28. Diagrama de Flujo y Tablas de Décima Historia de Usuario.

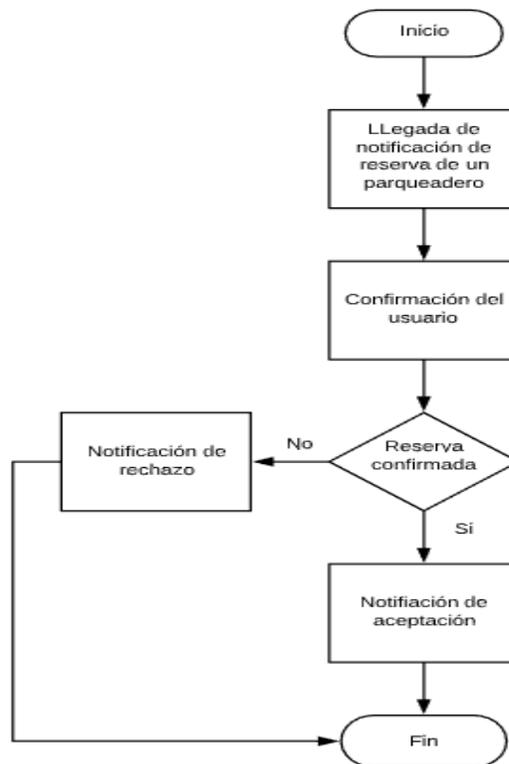


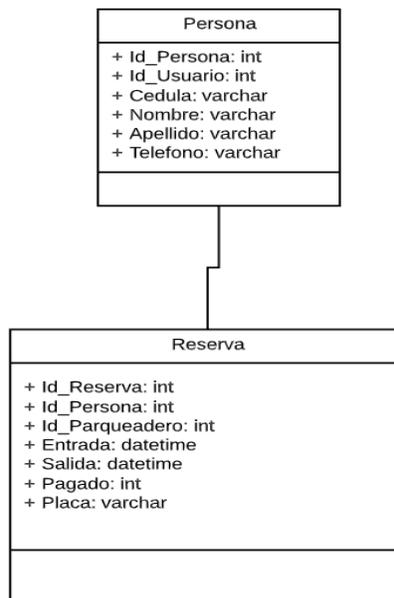
Anexo 29. Diagrama de Flujo y Tablas de Décima-Primera Historia de Usuario.



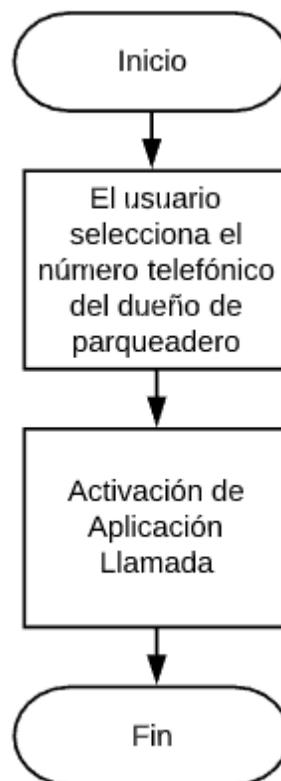


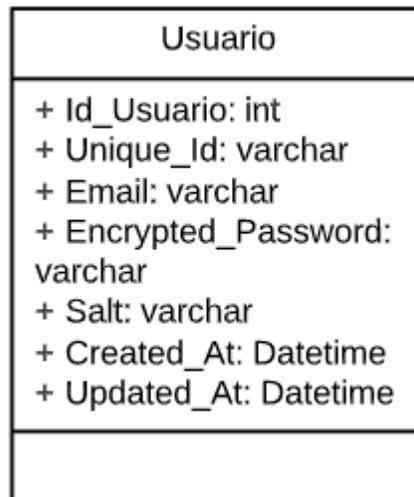
Anexo 30. Diagrama de Flujo y Tablas de Décima-Segunda Historia de Usuario.



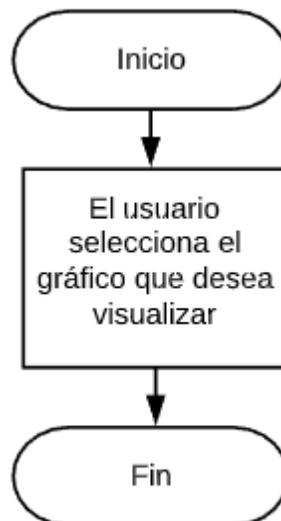


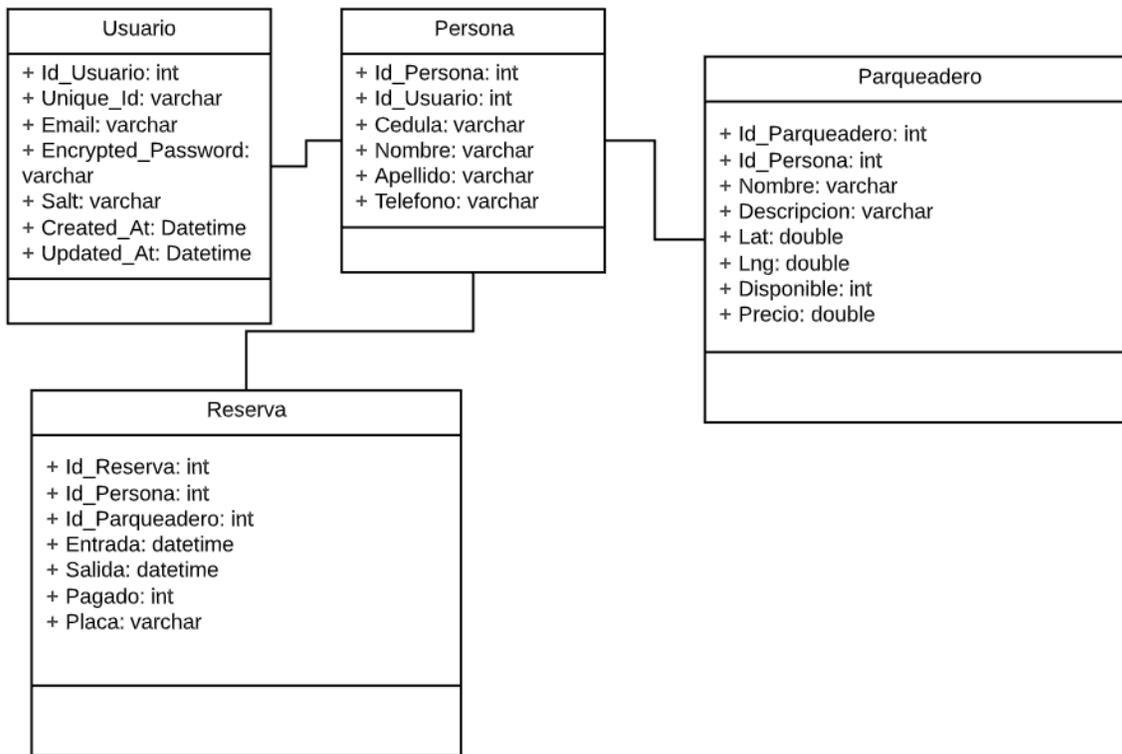
Anexo 31. Diagrama de Flujo y Tabla de Décima-Tercera Historia de Usuario.



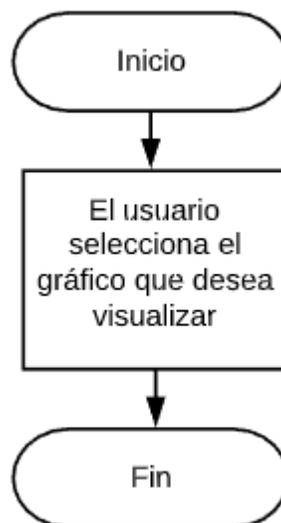


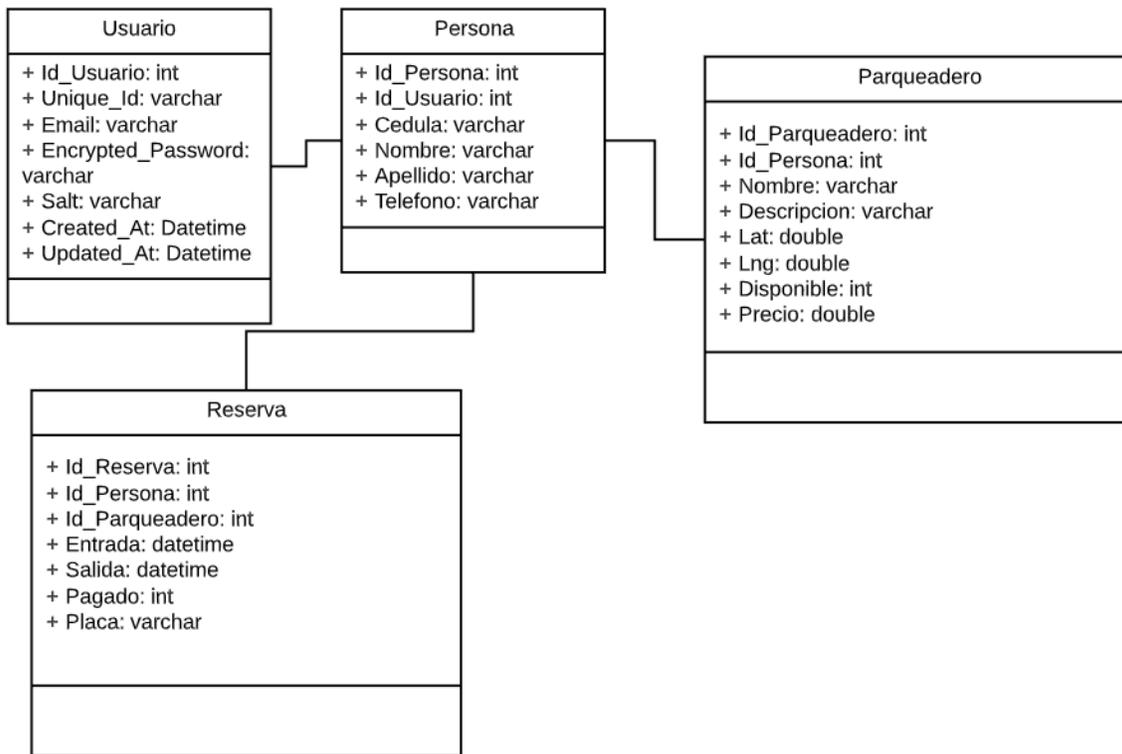
Anexo 32. Diagrama de Flujo y Tablas de Decimacuarta Historia de Usuario.



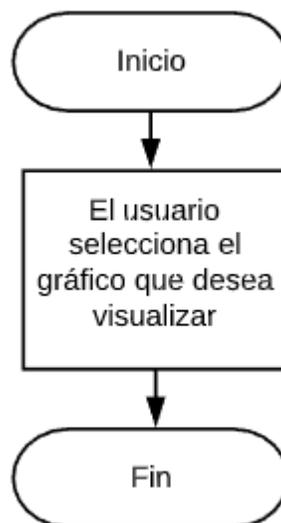


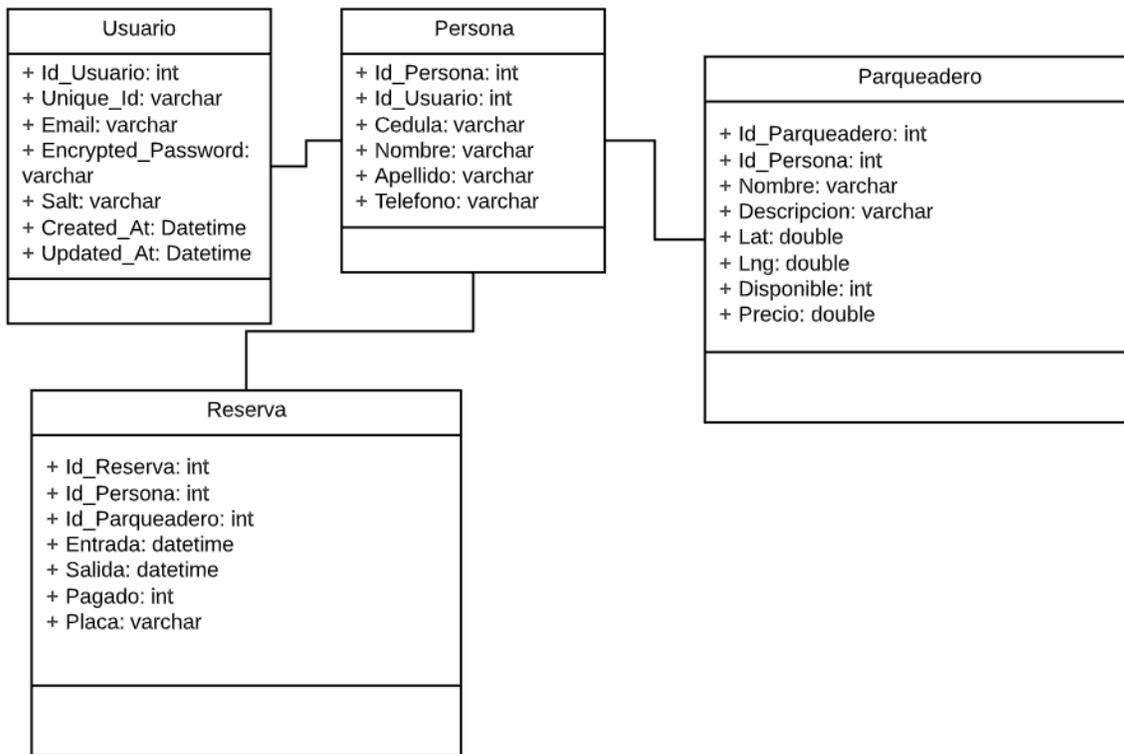
Anexo 33. Diagrama de Flujo y Tablas de Décima-Quinta Historia de Usuario.





Anexo 34. Diagrama de Flujo y Tablas de Décima-Sexta Historia de Usuario.





Anexo 35. Diagrama de Flujo y Tablas de Décima-Séptima Historia de Usuario.

