



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E
INDUSTRIAL**

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN

Tema:

**APLICACIÓN MÓVIL MULTIPLATAFORMA APLICANDO FLUTTER
PARA ASIGNACIÓN DE ESPACIOS DE PARQUEO EN LA CIUDAD DE
AMBATO**

Trabajo de titulación modalidad Proyecto de Investigación, presentado previo a la obtención del título de Ingeniero en Tecnologías de la Información.

ÁREA: Software

LÍNEA DE INVESTIGACIÓN: Desarrollo de software

AUTOR: Luis Orlando Salazar Ramos

TUTOR: Ing. Edwin Hernando Buenaño Valencia, Mg.

Ambato, Ecuador

agosto - 2023

APROBACIÓN DEL TUTOR

En calidad de tutor del trabajo de titulación con el tema: APLICACIÓN MÓVIL MULTIPLATAFORMA APLICANDO FLUTTER PARA ASIGNACIÓN DE ESPACIOS DE PARQUEO EN LA CIUDAD DE AMBATO, desarrollado bajo la modalidad Proyecto de Investigación por el señor Luis Orlando Salazar Ramos estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.3 del instructivo reglamento referido.

Ambato, agosto 2023.

Ing. Edwin Hernando Buenaño Valencia, Mg.

TUTOR

AUTORÍA

El presente trabajo de titulación titulado: APLICACIÓN MÓVIL MULTIPLATAFORMA APLICANDO FLUTTER PARA ASIGNACIÓN DE ESPACIOS DE PARQUEO EN LA CIUDAD DE AMBATO es absolutamente original, auténtico y personal y ha observado los preceptos establecidos en la Disposición General Quinta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, agosto 2023.



Luis Orlando Salazar Ramos

C.C. 1804966933

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato para que reproduzca total o parcialmente este trabajo de titulación dentro de las regulaciones legales e institucionales correspondientes. Además, cedo todos mis derechos de autor a favor de la institución con el propósito de su difusión pública, por lo tanto, autorizo su publicación en el repositorio virtual institucional como un documento disponible para la lectura y uso con fines académicos e investigativos de acuerdo con la Disposición General Cuarta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato

Ambato, agosto 2023.



Luis Orlando Salazar Ramos

C.C. 1804966933

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del informe Final del trabajo de titulación presentado por el señor Luis Orlando Salazar Ramos, estudiante de la Carrera de Tecnologías de la Información, bajo la Modalidad Proyecto de Investigación, titulado **APLICACIÓN MÓVIL MULTIPLATAFORMA APLICANDO FLUTTER PARA ASIGNACIÓN DE ESPACIOS DE PARQUEO EN LA CIUDAD DE AMBATO**, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.4 del instructivo del reglamento referido. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, agosto 2023.

Ing. Elsa Pilar Urrutia Urrutia, Mg.
PRESIDENTE DEL TRIBUNAL

Ing. Santiago David Jara Moya, Mg.
PROFESOR CALIFICADOR

Ing. Jaime Bolívar Ruiz Banda, Mg.
PROFESOR CALIFICADOR

DEDICATORIA

Este trabajo representa el resultado de años de mi esfuerzo, pero también representa lo que han hecho mis padres, que, a lo largo de mi vida han sido mi guía y mi inspiración, el presente proyecto, producto del finiquito de la ardua etapa universitaria, va dedicado a ellos, Segundo y Rosa, su apoyo hacia mis metas y anhelos ha sido siempre incondicional, soporte en momentos difíciles con su amor y dedicación, agradecido con la vida por tenerlos como padres, quienes no dejaron que me rinda cuando las cosas se complicaron y las adversidades mentales querían vencer.

En este significativo paso en mi vida, celebro a toda mi familia, a mis hermanos que han sabido ser aliento para seguir, a mis sobrinos, que me hicieron ver que cada acción que tomo puede ser su ejemplo, motivante para cada día ser mejor, para todos ellos, para mí.

Luis Orlando Salazar Ramos

AGRADECIMIENTO

Quiero agradecer a todos quienes sumaron a lo largo de esta etapa universitaria, reflejo del apoyo y contribuciones de quienes me rodean. A mis amigos dentro de la Universidad, muestra de la amistad entre colegas y el apoyo en todo momento para que logremos nuestro objetivo común sin perder rumbo en el proceso.

A mis amigos de toda la vida, que supieron apoyarme y no me dejaron solo en momentos en que los necesité con palabras de aliento y ánimo cuando las cosas se tornaron difíciles y confusas, siendo ejemplo de que con disciplina y constancia las metas se logran, ya que somos reflejo de quienes nos rodean, a todos los mencionados los quiero cerca de mí siempre.

Vale la mención a la gran mayoría de docentes, que supieron transmitir sus conocimientos, años de experiencia y consejos de forma asertiva, logrando formar personas y profesionales que resaltarán donde la vida los coloque.

Gracias a todos ustedes.

ÍNDICE GENERAL DE CONTENIDOS

APROBACIÓN DEL TUTOR.....	ii
AUTORÍA.....	iii
DERECHOS DE AUTOR	iv
.....	iv
APROBACIÓN DEL TRIBUNAL DE GRADO	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
RESUMEN EJECUTIVO	xiv
ABSTRACT.....	xv
CAPÍTULO I.- MARCO TEÓRICO	1
1.1. Tema de investigación.....	1
1.1.1. Planteamiento del problema.....	1
1.2. Antecedentes Investigativos	3
1.3. Fundamentación teórica.....	4
1.3.1. Uso de parqueaderos.	4
1.3.2. Asignación de espacios.	4
1.3.3. Control de espacios.	4
1.3.4. Planificación de áreas.....	5
1.3.5. Ingeniería de software.	5
1.3.6. Desarrollo de software.	5
1.3.7. Aplicaciones móviles.	5
1.3.8. Aplicación móvil Flutter.	6
1.3.9. Metodologías de desarrollo	6
1.3.9.1. Metodologías tradicionales	6
1.3.9.2. Metodologías ágiles.....	6
1.4. Objetivos	7
1.4.1. Objetivo general	7
1.4.2. Objetivos específicos	7
CAPÍTULO II.- METODOLOGÍA	8
2.1. Materiales	8
2.2.1. Modalidad de investigación	9

2.2.2. Población y muestra	9
2.2.3. Recolección de la información	10
2.2.4. Procesamiento y análisis de datos.....	25
CAPÍTULO III.- RESULTADOS Y DISCUSIÓN	27
3.1. Análisis y discusión de los resultados	27
3.1.1. Flutter, que es y su estado actual	27
3.1.2. Dart	31
3.1.3. Ventajas y desventajas de usar Flutter en el desarrollo móvil.....	32
3.1.4. Ejemplos de desarrollos multiplataforma con Flutter.....	36
3.2. Determinación de la metodología de desarrollo	37
3.3. Desarrollo de la propuesta	39
3.3.1. Fase I: Exploración	40
3.3.2. Fase II: Inicialización.....	41
3.3.3. Fase III: Producción.....	61
3.3.4. Fase IV: Estabilización	71
3.3.5. Fase V: Pruebas	72
CAPÍTULO IV.- CONCLUSIONES Y RECOMENDACIONES	91
4.1. Conclusiones	91
4.2. Recomendaciones	91
BIBLIOGRAFÍA	93
ANEXOS	96
A1. Encuesta Usuarios	96
A2. Encuesta personal de control	99
A3. Ficha de observación	101
A4. Manual de Usuario	103

ÍNDICE DE TABLAS

Tabla 1.	Población con la que se realizó la investigación.....	9
Tabla 2.	Varianza de los resultados obtenidos encuesta Personal de control de parqueaderos públicos de la ciudad de Ambato.....	10
Tabla 3.	Varianza de los resultados en la encuesta realizada a los usuarios de parqueadero públicos de la ciudad de Ambato	18
Tabla 4.	Resultados ficha de observación.....	25
Tabla 5.	Tabla resumen Flutter	35
Tabla 6.	Tabla comparativa metodologías ágiles para desarrollo móvil	38
Tabla 7.	Planificación de fases	43
Tabla 8.	Funcionalidades Administrador: Registro de administrador de parqueadero 46	
Tabla 9.	Funcionalidades Administrador. Registro de parqueadero.....	46
Tabla 10.	Funcionalidades Administrador. Registro de pisos.....	47
Tabla 11.	Funcionalidades Administrador. Registro de sitios	47
Tabla 12.	Funcionalidades Usuarios: Registro.....	47
Tabla 13.	Funcionalidades Usuarios: Inicio de sesión	48
Tabla 14.	Funcionalidades Usuarios: Pantalla inicio	48
Tabla 15.	Funcionalidades Usuarios: Listado de parqueaderos	49
Tabla 16.	Funcionalidades Usuarios: Historial de parqueos	49
Tabla 17.	Funcionalidades Usuarios: Pisos disponibles	50
Tabla 18.	Funcionalidades Usuarios: Selección de sitio disponible	50
Tabla 19.	Funcionalidades Usuarios: Parqueo en curso / Finalización.....	51
Tabla 20.	Funcionalidades Usuarios: Publicar reseña.....	51
Tabla 21.	Funcionalidades Usuarios: Editar reseña	51
Tabla 22.	Funcionalidades Usuarios: Ver reseñas	52
Tabla 23.	Caso de uso: Registro.....	66
Tabla 24.	Caso de uso: Inicio de sesión	67
Tabla 25.	Caso de uso: Renta de un sitio de parqueo.....	68
Tabla 26.	Caso de uso: Finalización de renta en curso	69
Tabla 27.	Caso de uso: Reseña a parqueadero	70
Tabla 28.	Caso de uso: Visualizar reseñas de un parqueadero.....	71
Tabla 29.	Prueba de aceptación: 1.....	87
Tabla 30.	Prueba de aceptación: 2.....	87
Tabla 31.	Prueba de aceptación: 3.....	87

Tabla 32.	Prueba de aceptación: 4.....	88
Tabla 33.	Prueba de aceptación: 5.....	88
Tabla 34.	Prueba de aceptación: 6.....	88
Tabla 35.	Prueba de aceptación: 7.....	89
Tabla 36.	Prueba de aceptación: 8.....	89
Tabla 37.	Prueba de aceptación: 9.....	89
Tabla 38.	Prueba de aceptación: 10.....	90
Tabla 39.	Prueba de aceptación: 11.....	90
Tabla 40.	Prueba de aceptación: 12.....	90

ÍNDICE DE FIGURAS

Figura 1.	Tabulación de resultados Preg. 1 Per. Cont.	11
Figura 2.	Tabulación de resultados Preg. 2 Per. Cont.	12
Figura 3.	Tabulación de resultados Preg. 3 Per. Cont.	13
Figura 4.	Tabulación de resultados Preg. 4 Per. Cont.	14
Figura 5.	Tabulación de resultados Preg. 5 Per. Cont.	15
Figura 6.	Tabulación de resultados Preg. 6 Per. Cont.	16
Figura 7.	Tabulación de resultados Preg. 7 Per. Cont.	17
Figura 8.	Tabulación de resultados Preg. 1 Usua.	19
Figura 9.	Tabulación de resultados Preg. 2 Usua.	20
Figura 10.	Tabulación de resultados Preg. 3 Usua.	21
Figura 11.	Tabulación de resultados Preg. 4 Usua.	22
Figura 12.	Tabulación de resultados Preg. 5 Usua.	23
Figura 13.	Proceso de Mobile-D[30].....	39
Figura 14.	Diseño de base de datos no relacional.....	44
Figura 15.	Arquitectura de proyecto.....	45
Figura 16.	Diagrama de caso de uso para desarrollador.....	52
Figura 17.	Diagrama de caso de uso para usuarios.....	53
Figura 18.	Componentes de diseño	54
Figura 19.	Maquetado: Registro	55
Figura 20.	Maquetado: Inicio de sesión	56
Figura 21.	Maquetado: Pantalla de inicio.....	57
Figura 22.	Maquetado: Datos de usuario.....	57
Figura 23.	Maquetado: Historial de uso	58
Figura 24.	Maquetado: Parqueaderos	59
Figura 25.	Maquetado: Esquema y elección de sitio	60
Figura 26.	Maquetado: Finalización de parqueo	61
Figura 27.	Bloque de conexión en archivo index.js	63
Figura 28.	Controlador finalizar renta	64
Figura 29.	Rutas de usuario	65
Figura 30.	Aplicación: Registro de usuario	73
Figura 31.	Aplicación: Inicio de sesión de usuario	74
Figura 32.	Aplicación: Pantalla de inicio	75
Figura 33.	Aplicación: Perfil de usuario.....	76

Figura 34.	Aplicación: Lista de parqueaderos	77
Figura 35.	Aplicación: Lista de pisos	78
Figura 36.	Aplicación: Reseña	79
Figura 37.	Aplicación: Listado de reseñas.....	80
Figura 38.	Aplicación: Esquema de piso	81
Figura 39.	Aplicación: Renta de sitio	82
Figura 40.	Aplicación: Parqueo en curso.....	83
Figura 41.	Aplicación: Historial con finalización pendiente	84
Figura 42.	Aplicación: Finalizar parqueo	85
Figura 43.	Aplicación: Historial de uso.....	86

RESUMEN EJECUTIVO

A través de una investigación enfocada sobre los problemas que enfrentan los usuarios al buscar sitios de estacionamiento en parqueaderos públicos, se identificaron deficiencias en la disponibilidad de información en tiempo real y en la realización de reservas de manera rápida.

Este estudio se enfoca en el desarrollo de una aplicación móvil multiplataforma que se realizó bajo la metodología Mobile-D con el fin de tener rapidez de respuesta ante los cambios con flexibilidad utilizando Flutter, NodeJS y base de datos MongoDB, con Flutter como framework de desarrollo multiplataforma, se logró crear una interfaz fácil de usar que funciona en dispositivos iOS y Android. La eficiencia y rentabilidad de Flutter han sido destacadas, reduciendo costos y tiempo de desarrollo con el propósito de facilitar la asignación de espacios de estacionamiento en la ciudad de Ambato.

La aplicación móvil desarrollada proporciona datos actualizados sobre la disponibilidad de espacios en tiempo real y permite realizar reservas de forma instantánea, logrando cumplir con el objetivo general de la investigación, una aplicación móvil con estas tecnologías y frameworks a la vanguardia para la asignación de espacios de parqueo en la ciudad de Ambato.

Palabras clave: Flutter, NodeJS, MongoDB, parqueo, multiplataforma

ABSTRACT

Through a focused research based on the issues by public parking lots users while searching for parking spaces, we found deficiencies in the availability of real-time information and the process of quick reservations.

This study is focused on the development of a mobile application that is multiplatform which was created under the Mobile-D methodology. The goal is to provide quick response to the changes with flexibility by using Flutter, NodeJS, and MongoDB. With Flutter as a Framework for the cross-platform development, we created a user interface that easy to navigate and compatible with iOS and Android devices. The efficiency and cost-effectiveness of Flutter were highlighted, reducing development costs and time. The purpose of this research was to facilitate parking space allocation in Ambato city.

The developed mobile application provides up-to-date data on real-time space availability and allows for instant reservations, successfully achieving the overall objective of the research a mobile application utilizing cutting-edge technologies and frameworks for parking space allocation in Ambato city.

Keywords: Flutter, NodeJS, MongoDB, parking, cross-platform.

CAPÍTULO I.- MARCO TEÓRICO

1.1. Tema de investigación

APLICACIÓN MÓVIL MULTIPLATAFORMA APLICANDO FLUTTER PARA ASIGNACIÓN DE ESPACIOS DE PARQUEO EN LA CIUDAD DE AMBATO.

1.1.1. Planteamiento del problema

De los 892 millones de autos conformando el patio automotor mundial en 2005 a los 1282 millones alcanzados en 2015, estimando un automóvil por cada seis personas, para tener en cuenta el aumento del patio automotor mundial, desde el último recuento de la Organización Internacional de Fabricantes (OICA), a fecha que se desarrolla esta investigación (primer semestre 2023), son datos para dimensionar la aglomeración de autos en ciudades e instituciones concurridas a nivel del mundo, que han notado una problemática en aumento, la constante falta de encontrar un sitio para dejar el auto mientras las personas llevan a cabo sus acciones cotidianas, siendo inevitable el caos de encontrar un lugar cercano y seguro de aparcamiento, lleva a buscar soluciones tecnológicas para alivianar esta tarea día con día[1].

Las personas que llevan un auto a su sitio de esparcimiento, comercio, trabajo, salud, confían en tener un sitio seguro, cercano y económico para guardar su auto durante el desenvolvimiento de sus actividades cotidianas, pero lamentablemente muchos sitios como: parques, supermercados, hospitales, centros de acopio, Gobiernos descentralizados, instituciones tanto públicas como privadas, suelen no tener un manejo eficiente del parqueo de los vehículos de quienes conforman su comunidad, problemática que envuelve a las ciudades de toda dimensión en toda parte del mundo, involucrando soluciones tecnológicas.

El estacionamiento de los vehículos es un ingreso de recursos económicos para la comunidad involucrada, el ofrecimiento de espacios para el estacionamiento es

fundamental para el crecimiento y prosperidad de la comunidad, pero la mala gestión de estos espacios conlleva contaminación de la mano de obra en la movilidad, afectando la productividad de quienes conforman la sociedad en desarrollo, la implementación de sistemas informáticos para la gestión de los espacios de parqueo permite monitorear y facilitar el acceso a vehículos autorizados a más de restringir el acceso a quienes no reúnen requisitos para el ingreso, en caso de ser un sitio con ingreso restringido[2].

Citar estas soluciones implementadas alrededor del mundo es de suma importancia para conocer sobre las soluciones propuestas por los investigadores involucrados, como es el caso de la Fundación Universitaria Católica Lumen Gentium (Unicatólica), con presencia en Cali, Colombia: donde las personas con vehículo se aglomeran en las entradas en espera de un espacio sin saber la disponibilidad de dichos espacios al interior de las instalaciones de esta universidad, optando por dejar su auto afuera, exponiéndose a daños, robos o retención del vehículo[3].

En el día a día el llevar consigo un vehículo es indispensable cuando se piensa en comodidad y seguridad del entorno, pero esta comodidad se convierte en problema y se torna difícil encontrar un sitio seguro y cercano para dejar el vehículo, a más del tiempo que puede tomar encontrar un sitio con las características de seguridad y confiabilidad necesarias para su funcionamiento y tranquilidad del usuario final.

A nivel de nuestro país, una institución que ha tomado cartas en el asunto mediante la investigación en este campo ha sido la Universidad Católica de Santiago de Guayaquil, en la que buscar un sitio libre para estacionar un vehículo, es un problema que actualmente aqueja al campus universitario, investigadores han desarrollado un sistema de asistencia para búsqueda de disponibilidad de estacionamiento en el campus[4].

La capital del país ecuatoriano, Quito, ha implementado una aplicación móvil para el sistema de estacionamientos municipales, disponible de forma indistinta de Sistema Operativo gracias al desarrollo multiplataforma, mejorando la administración de los espacios y los registros contables, al llevar integrados horarios y precios de hora o fracción si aplica[5].

1.2. Antecedentes Investigativos

El análisis de investigaciones desarrolladas con anterioridad a la investigación presente dentro de repositorios confiables se han encontrado escasos temas similares dentro del rango temporal recomendado para un análisis de antecedentes de calidad notable, sin embargo, a continuación, serán citados los resultados de dicha búsqueda:

Según Jaime A. Páez, Jairo A. Cortes Fredys A. Simanca y Fabián Blanco[6], concluyeron que el desarrollo del software móvil, mejorado bajo el uso adecuado de métricas y requerimientos del software que controla la entrada y salida de bici usuarios por medio de la aplicación móvil, responde a las necesidades de la universidad, mejorando el control sobre el parqueo de bicicletas en sus instalaciones.

En la investigación de Christian Orlando Guachán Cuásquer y Diego Juvenal Sigcho Alvarado en[7], describe el proceso para el desarrollo de una aplicación que se desplegará en dispositivos móviles permitiendo al usuario final identificar con anticipación de forma fácil, rápida y en tiempo real los lugares que se encuentran disponibles en una zona de parqueo, concluyendo una vez realizadas las pruebas de la app que efectivamente se pudo lograr una visualización clara y rápida de las zonas disponibles para el parqueo.

Según Luis Cruz en[4], donde se evidenció que los conductores si están de acuerdo con utilizar una aplicación que les permita buscar y encontrar un estacionamiento disponible en el campus sin desperdicio de su tiempo, en la que se concluyó que la implementación de un sistema de asistencia para la búsqueda de espacios de parqueo libres ha reducido considerablemente los problemas que se generan sobre todo en el horario de mayor afluencia de autos en las horas pico, además sirve de base para que la UCSG extienda este proyecto a los demás parqueaderos para comodidad de los demás usuarios.

Según Lukas Dagne en[8], donde concluyó que las tecnologías revolucionarias internas de Flutter y el ciclo de desarrollo rápido que representa lo convierten en una opción competitiva sobre el desarrollo nativo, independientemente de su aspecto

multiplataforma. Sus técnicas de renderizado únicas y su arquitectura en capas lo convierten en una opción fácil para implementaciones innovadoras de interfaz y animación. La opción de evitar la herencia a favor de la composición en cada capa del marco abre casos de uso emocionante e impredecible de sus componentes.

1.3. Fundamentación teórica

1.3.1. Uso de parqueaderos.

Colocar transitoriamente en un lugar público, señalado al efecto por la autoridad, coches u otros vehículos. Procurando el uso de los espacios designados para este fin transitorio[9]. Fundamental para mantener el orden y la seguridad en la circulación de vehículos, diseñados para ofrecer un espacio seguro y adecuado para que los conductores estacionen sus vehículos temporalmente.

1.3.2. Asignación de espacios.

La asignación de parqueaderos es responsabilidad de la Dirección Administrativa de la ciudad a la que le pertenezcan los predios[10]. La asignación de espacios puede realizarse mediante la creación de zonas designadas con regulación, parqueaderos subterráneos o específicos espacios de la vía pública, debiendo ser normadas y regularizadas para garantizar su uso adecuado.

1.3.3. Control de espacios.

El manejo y control de los parqueaderos es responsabilidad de la Dirección Administrativa, a través de gestión de espacios y personal encargado sí aplica[10]. Puede incluir medidas como la gestión del personal encargado del mantenimiento y vigilancia de los espacios, el monitoreo de disponibilidad y aplicación de sanciones en caso de no acatar las normas establecidas.

1.3.4. Planificación de áreas.

Los estacionamientos son un componente de la ingeniería de transporte que generalmente se aborda de forma tangencial en los procesos de planificación urbana[11]. Una planificación deficiente para áreas de estacionamientos puede tener consecuencias negativas en la movilidad urbana, problemática que debe ser abordada por la planificación urbana de las ciudades.

1.3.5. Ingeniería de software.

La Ingeniería de software es una disciplina que trata sobre el análisis, diseño y construcción de aplicaciones informáticas a través de la utilización de metodologías, herramientas, técnicas de desarrollo y documentación para una correcta gestión del software en una organización[12]. Estas metodologías tienen sus ventajas y desventajas y deben ser elegidas según las necesidades del proyecto y organización.

1.3.6. Desarrollo de software.

El desarrollo de software se refiere a un conjunto de actividades informáticas dedicadas al proceso de creación, diseño, despliegue y compatibilidad de software[13]. Es un proceso que implica actividades informáticas relacionadas con la creación, diseño, despliegue y compatibilidad de software, crucial para la construcción de aplicaciones y sistemas de software de alta calidad que cumplen con los requisitos de los usuarios y sus necesidades.

1.3.7. Aplicaciones móviles.

Aplicación de software que se instala en dispositivos móviles o tabletas para ayudar al usuario en una labor concreta, ya sea de carácter profesional o de ocio y entretenimiento, para facilitarnos la consecución de una tarea determinada o asistirnos en operaciones y gestiones del día a día[14]. Pueden ser diseñadas para sistemas operativos específicos o multiplataforma según las necesidades de uso, el desarrollo

de aplicaciones móviles requiere de técnicas de diseño y programación específicas para optimizar el rendimiento y experiencia de usuario.

1.3.8. Aplicación móvil Flutter.

Flutter es el kit de herramientas de UI de Google para realizar aplicaciones, compiladas nativamente para móvil, web y escritorio desde una única base de código. Los widgets de Flutter incorporan todas las diferencias críticas entre plataformas para proporcionar un rendimiento totalmente nativo tanto en iOS como en Android[15]. Flutter utiliza el lenguaje de programación Dart para construir bloques de interfaces de usuario interactivas y dinámicas.

1.3.9. Metodologías de desarrollo

Las metodologías de desarrollo de software son un conjunto de técnicas y métodos organizativos que se aplican para diseñar soluciones de software informático. El objetivo de las distintas metodologías es el de intentar organizar los equipos de trabajo para que estos desarrollen las funciones de un programa de la mejor manera posible.[16]

1.3.9.1. Metodologías tradicionales

Las metodologías de desarrollo de software tradicionales se caracterizan por definir total y rígidamente los requisitos al inicio de los proyectos de ingeniería de software. Los ciclos de desarrollo son poco flexibles y no permiten realizar cambios.

1.3.9.2. Metodologías ágiles

Las metodologías ágiles de desarrollo de software son las más utilizadas hoy en día debido a su alta flexibilidad y agilidad. Los equipos de trabajo que las utilizan son mucho más productivos y eficientes, ya que saben lo que tienen que hacer en cada momento. Además, la metodología permite adaptar el software a las necesidades que

van surgiendo por el camino, lo que facilita construir aplicaciones más funcionales.[16]

1.4. Objetivos

1.4.1. Objetivo general

Desarrollar una aplicación móvil multiplataforma aplicando Flutter para la asignación de espacios de parqueo en la ciudad de Ambato.

1.4.2. Objetivos específicos

- Analizar los problemas que enfrentan las personas al momento de buscar sitios de asignación en parqueaderos públicos.
- Investigar la tecnología Flutter en el desarrollo de aplicaciones móviles multiplataforma.
- Desarrollar la aplicación móvil para la gestión de los diferentes espacios de parqueo.

CAPÍTULO II.- METODOLOGÍA

2.1. Materiales

En virtud del contexto del problema, se recopiló información mediante el empleo de una ficha de observación, también, a través de encuestas compuestas por preguntas concretas y objetivas. Estas medidas se llevaron a cabo con el propósito de identificar los inconvenientes experimentados por los usuarios al estacionar sus vehículos en los parqueaderos públicos de la ciudad de Ambato. Paralelamente, se buscó obtener una comprensión más profunda de la perspectiva del personal encargado de los parqueaderos en la ciudad.

Encuestas a usuarios

Como se observa en Encuesta Usuarios, el cuestionario de la encuesta dirigida a los usuarios cuenta con un total de 5 preguntas de respuesta única, con el fin de recolectar su percepción a cerca de los parqueaderos públicos de la ciudad de Ambato y cuan afectados se ven su tiempo y actividades a desarrollar por consecuencia de los problemas que enfrentan al parquear sus vehículos, usuarios anónimos para procurar la imparcialidad y sinceridad de sus respuestas al cuestionario.

Encuestas a personal de control

Como se observa en Encuesta para personal de control, “Per. Cont.”, abreviado en figuras próximas, el cuestionario dirigido al Per. Cont. cuenta con un total de 7 preguntas de respuesta única, con el fin de recolectar su percepción de los problemas que enfrentan sus usuarios al momento de utilizar las instalaciones a su cargo, ayudando a contrastar información recolectada a los usuarios.

Ficha de observación

Como se observa en Ficha de observación, consta de un total de 11 apartados divididos en 3 grupos, observaciones del parqueadero, observaciones a los usuarios en búsqueda de un espacio libre y las conclusiones extraídas de la observación, dando lugar a poder detallar con mayor minuciosidad como se percibe los problemas y posibles causas.

2.2.1. Modalidad de investigación

Investigación de modalidades especiales

Investigación de modalidades especiales debido a que el proyecto investigativo se refiere a un contexto específico y responde a la necesidad de tipo socio cultural que engloba al desarrollo de software.

Investigación de campo

Investigación de campo debido a que el proyecto en estudio sirve de fuente de información para el investigador, dando lugar a la recolección de información por observación.

2.2.2. Población y muestra

Al ser un tema extenso y desconocer el total de la población, se tomó una selección aleatoria considerada adecuada por el investigador.

Población	Número	Porcentaje
Usuarios	45	90 %
Personal de control	5	10 %
Total	50	100 %

Tabla 1. Población con la que se realizó la investigación

2.2.3. Recolección de la información

La recolección de la información se dio mediante la observación directa, encuestas a usuarios y personal de control en parqueos en la ciudad de Ambato, de forma presencial a todos los involucrados.

Validación del instrumento dirigido al personal de control de parqueaderos públicos en la ciudad de Ambato

Con el propósito de que la información recolectada se refleje de la forma más cercana a la realidad, se utiliza el método de Alfa de Cronbach para la validación del instrumento de recolección de la información, de tal modo que se pueda calcular un estimado de la confiabilidad de las respuestas obtenidas en el proceso.

Pregunta	Varianza del ítem	Varianza total del instrumento
1	0.56	9.76
2	0.16	
3	0.24	
4	0.24	
5	0.56	
6	0.96	
7	0.64	
Total	3.36	

Tabla 2. Varianza de los resultados obtenidos encuesta al Per. Cont. de parqueaderos públicos de la ciudad de Ambato

Se aplica la fórmula matemática para el cálculo de Alfa de Cronbach que es:

$$\alpha = \frac{K}{K - 1} \left[1 - \frac{\sum s^2}{s^2} \right]$$

Donde:

K: número de ítems del instrumento: 7

$\sum s^2$: sumatoria de la varianza de los ítems: 3.36

s^2 : varianza total del instrumento: 9.76

$$\alpha = 0.77$$

Dado que el coeficiente obtenido del Alfa de Cronbach es de 0.77, se considera que el instrumento es de consistencia interna buena, cercana a muy buena, siendo un instrumento útil para la investigación.

Resultados de la encuesta aplicada a personal de control de parqueaderos de la ciudad de Ambato

La encuesta fue aplicada a un total de 5 personas a cargo del control de parqueaderos públicos de la ciudad de Ambato.

Pregunta 1:

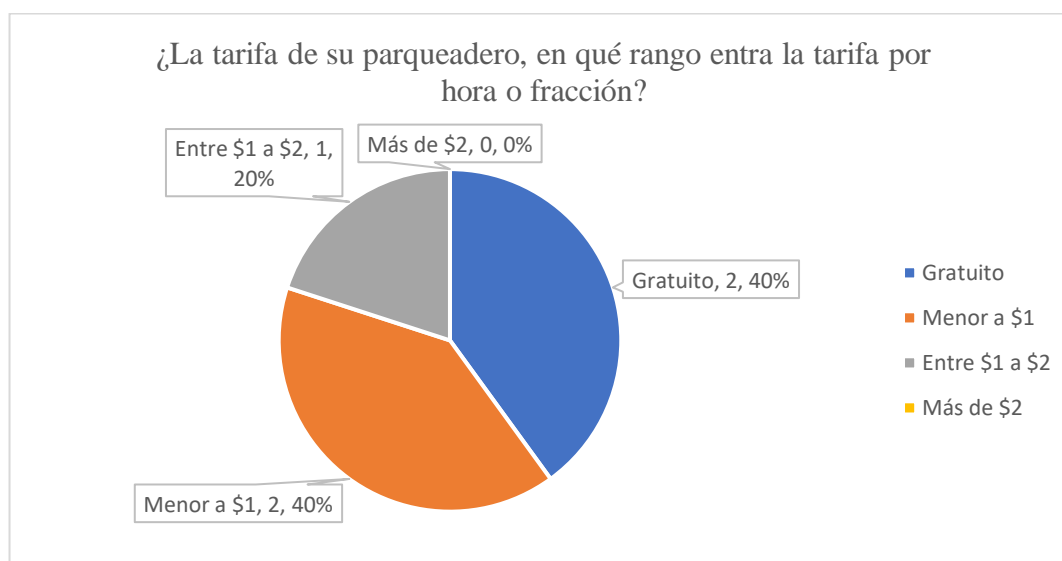


Figura 1. Tabulación de resultados Preg. 1 Per. Cont.

Análisis e interpretación de resultados

De acuerdo con los resultados visualizados en la Figura 1, **el 40% del personal encuestado está a cargo de parqueaderos públicos que no tienen costo alguno para sus usuarios**, el otro 40% tienen a su cargo parqueaderos con una tarifa menor a \$1, el 20% tiene a su cargo un parqueadero con una tarifa de entre \$1 a \$2, mientras que ningún encuestado aseguró tener a su cargo una tarifa mayor a \$2 en su parqueadero a cargo. Resultados que sugieren que la ciudad de Ambato tiene tarifas consideradas bajas o nulas para el uso de sus parqueaderos, al ser así, esperar una mayor afluencia de usuarios.

Pregunta 2:

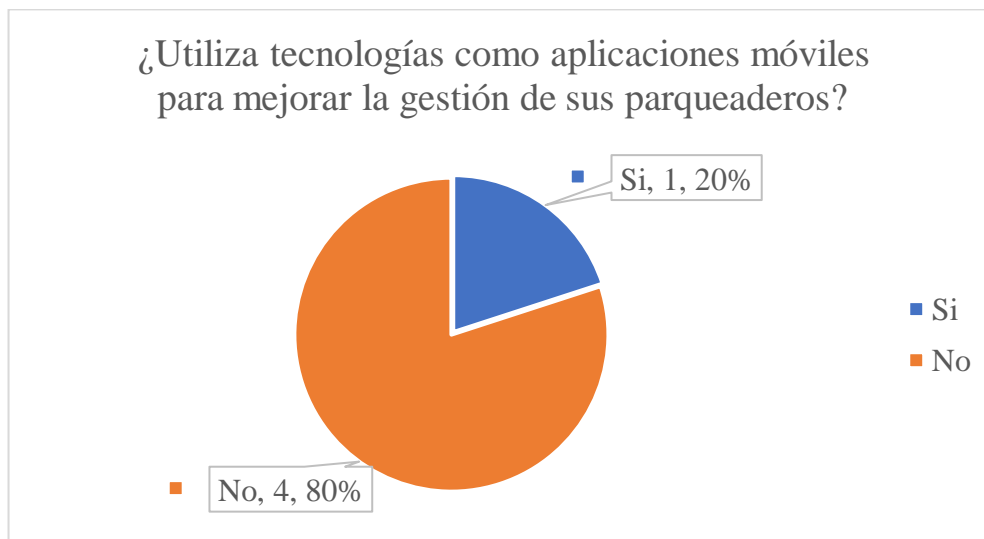


Figura 2. Tabulación de resultados Preg. 2 Per. Cont.

Análisis e interpretación de resultados

De acuerdo con los resultados de la Figura 2, **el 80% del personal encuestado no utiliza tecnologías móviles para mejorar la gestión de los parqueaderos públicos a su cargo**, el restante 20% mencionó que, si las utiliza, dando a entender que hay una gran brecha para mejora de esta gestión con ayuda de tecnologías móviles.

Pregunta 3.

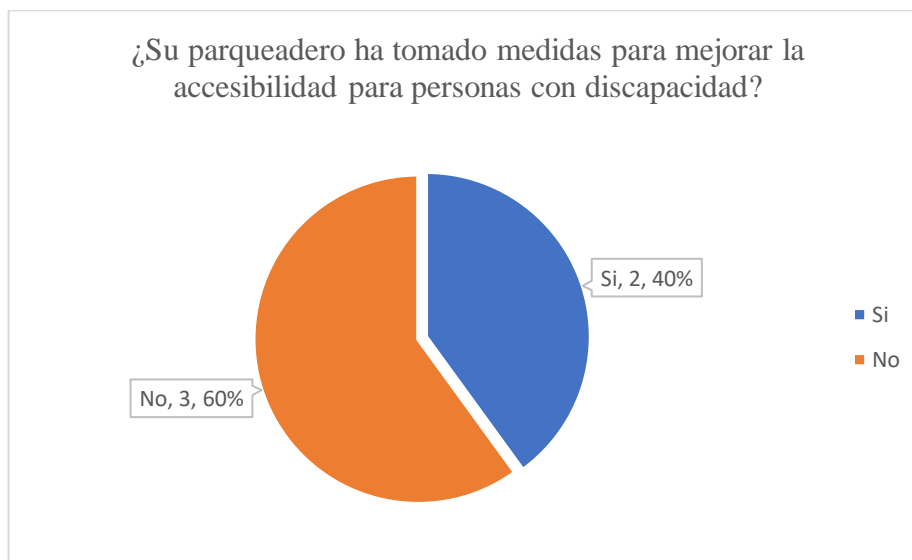


Figura 3. Tabulación de resultados Preg. 3 Per. Cont.

Análisis e interpretación de resultados

De acuerdo con los resultados de la Figura 3, **el 60% del personal encuestado asegura haber tomado medidas para mejorar la accesibilidad para personas con discapacidad**, ampliando la capacidad de su servicio a más usuarios con condiciones especiales, mientras que el restante 40% manifestó que no ha tomado cartas en este asunto, lo que denota un desinterés de parte de los diseñadores y constructores de la mayoría de parqueaderos en la ciudad de Ambato por el bienestar de las personas con capacidades especiales.

Pregunta 4.

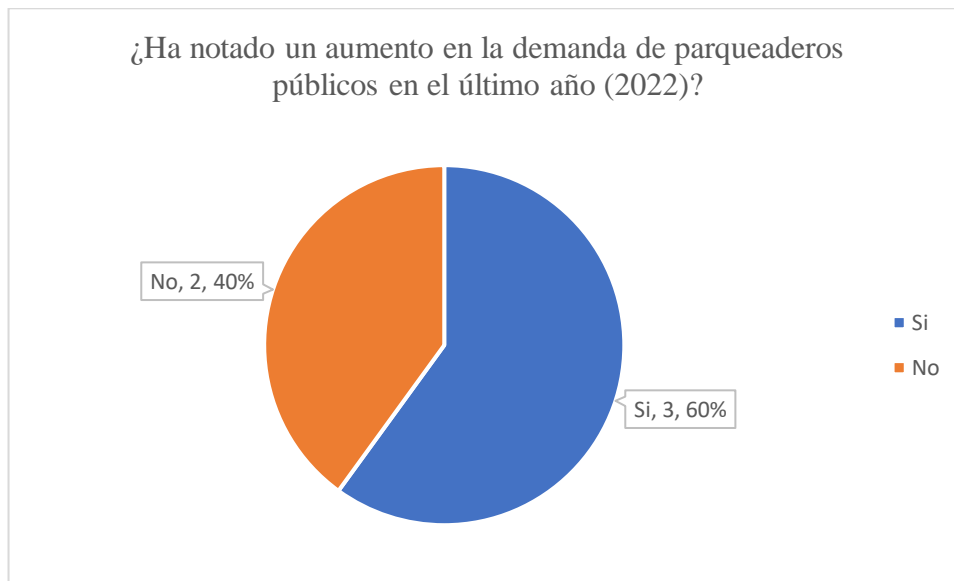


Figura 4. Tabulación de resultados Preg. 4 Per. Cont.

Análisis e interpretación de resultados

De acuerdo con los resultados de la Figura 4, **el 60% del personal encuestado ha notado un aumento en la demanda en el uso de su parqueadero**, mientras que el 40% restante no ha notado un cambio en su demanda, siendo que quienes perciben una mayor afluencia de clientes podrían requerir herramientas de gestión de sus parqueaderos.

Pregunta 5.

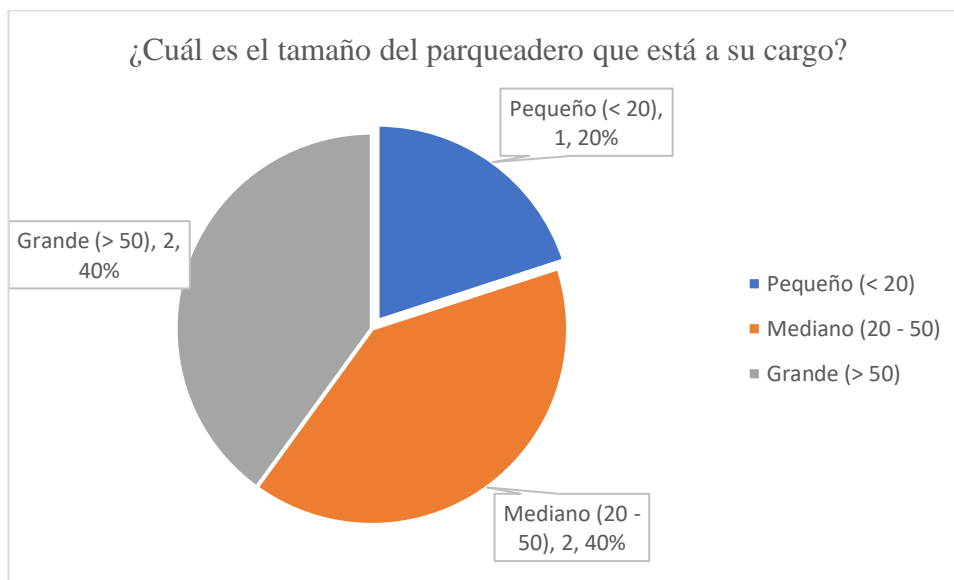


Figura 5. Tabulación de resultados Preg. 5 Per. Cont.

Análisis e interpretación de resultados

Del personal encuestado, según los resultados reflejados en la Figura 5, **el 50% asegura estar a cargo de parqueaderos grandes**, que cuentan con más de 50 espacios a disposición de los usuarios de la ciudad de Ambato, mientras que el 40% encuestado tiene a su cargo un parqueadero considerado mediano ya que consta con menos de 50 plazas a disponibilidad de los usuarios, pero no menos de 20, mientras que el 20% restante tiene a su cargo parqueaderos pequeños, es decir, de menos de 20 plazas para sus usuarios, notando que hay un considerable número de parqueaderos que cuentan con más de 50 plazas para los vehículos, necesitando ayuda humana y tecnológica para su administración y eficiente funcionamiento.

Pregunta 6.

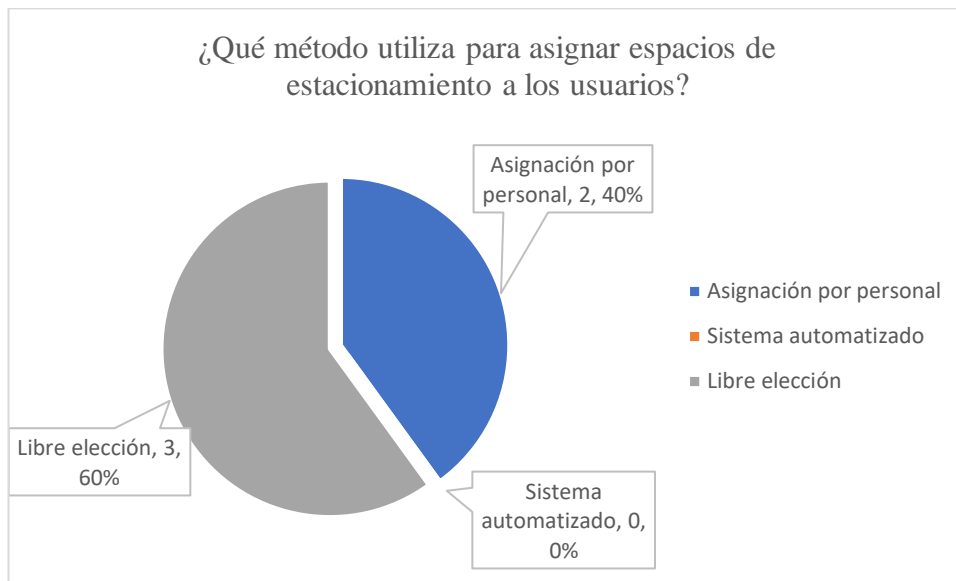


Figura 6. Tabulación de resultados Preg. 6 Per. Cont.

Análisis e interpretación de resultados

Según los resultados que se detallan en la Figura 6, **el 60% del personal aseveró que su parqueadero tiene asignación libre para los sitios disponibles** para sus usuarios, mientras que un 40% aseveró que lo hacen por asignación mediante personal del parqueadero, nadie mencionó contar con un sistema automatizado para esta tarea en sus respectivos parqueaderos, denotando poco cuidado en la organización de espacios, causando gran afluencia en ciertos sectores preferidos por los usuarios y mala gestión de espacios.

Pregunta 7.

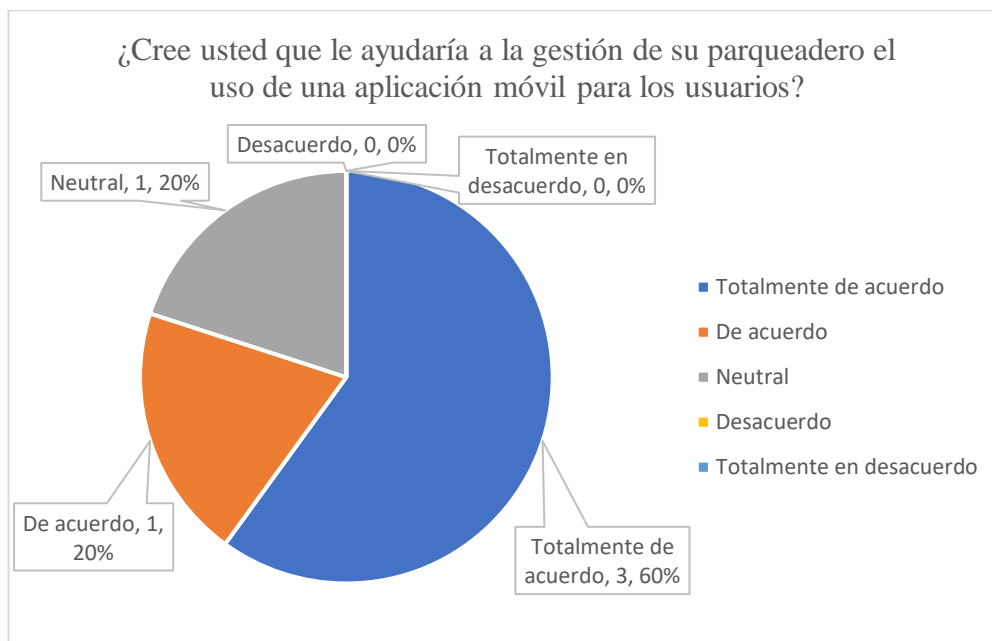


Figura 7. Tabulación de resultados Preg. 7 Per. Cont.

Análisis e interpretación de resultados

De acuerdo con los resultados detallados en la Figura 7, **el 60% del personal encuestado está totalmente de acuerdo que una aplicación móvil para los usuarios les sería de ayuda para la gestión de su parqueadero**, mientras que un 20% estuvo de acuerdo con esta aplicación, un 20% se mantuvo neutral, mientras que ninguna parte del personal se manifestó en desacuerdo o totalmente en desacuerdo con este ítem del instrumento, dando una perspectiva positiva al desarrollo de una app móvil para una mejor gestión de sus parqueaderos, punto muy positivo para el proyecto investigativo.

Validación del instrumento dirigido al personal de control de parqueaderos públicos en la ciudad de Ambato

Con el propósito de que la información recolectada se refleje de la forma más cercana a la realidad, se vuelve a utilizar el coeficiente de Alfa de Cronbach, realizado ya con

la Encuesta dirigida al personal de control, ahora con los resultados del instrumento de recolección de la información dirigido a los usuarios de parqueaderos públicos de la ciudad de Ambato, de tal modo que se pueda calcular un estimado de la confiabilidad de las respuestas obtenidas en el proceso.

Pregunta	Varianza del ítem	Varianza total del instrumento
1	0.47	4.55
2	0.30	
3	0.49	
4	0.24	
5	0.39	
Total	1.89	

Tabla 3. Varianza de los resultados en la encuesta realizada a los usuarios de parqueadero públicos de la ciudad de Ambato

Se aplica la fórmula matemática para el cálculo de Alfa de Cronbach que es:

$$\alpha = \frac{K}{K - 1} \left[1 - \frac{\sum s^2}{s^2} \right]$$

Donde:

K: número de ítems del instrumento: 5

$\sum s^2$: sumatoria de la varianza de los ítems: 1.89

s^2 : varianza total del instrumento: 4.55

$$\alpha = 0.73$$

Dado que el coeficiente obtenido del Alfa de Cronbach es de 0.73, se considera que el instrumento es de consistencia interna buena, útil para la investigación.

Resultados de la encuesta aplicada a los usuarios de parqueaderos de la ciudad de Ambato

La encuesta fue aplicada a un total de 45 usuarios de parqueaderos públicos de la ciudad de Ambato.

Pregunta 1:

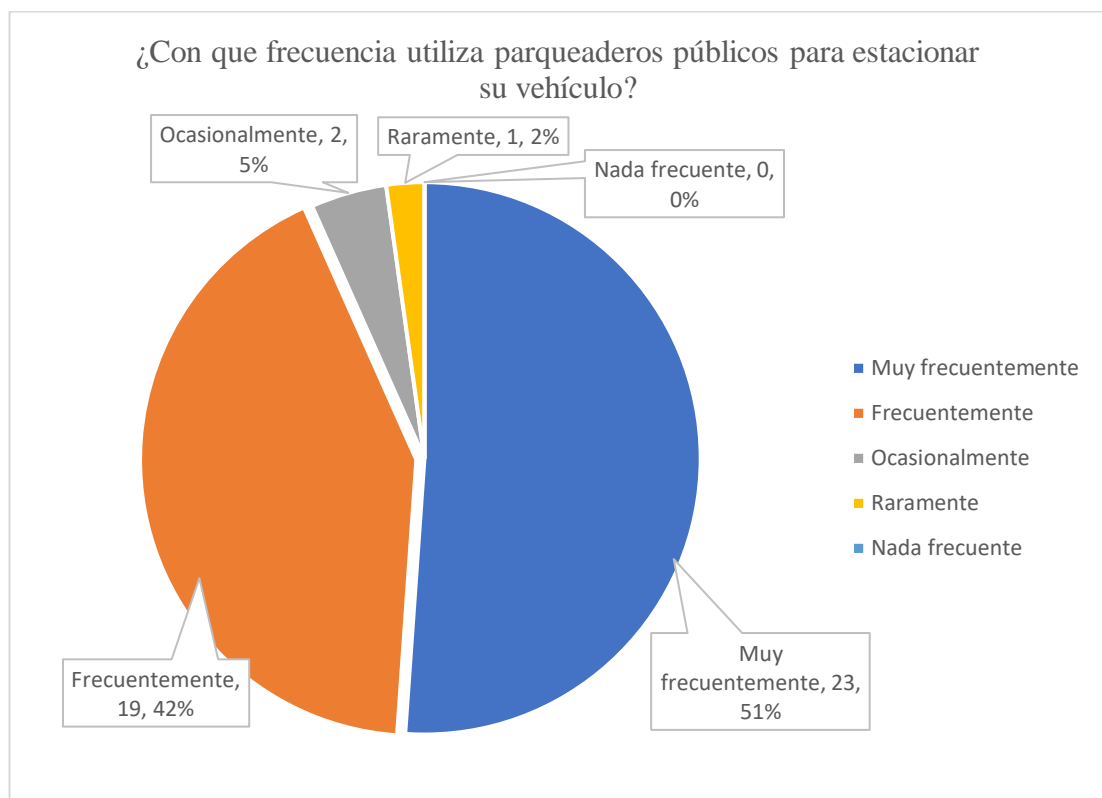


Figura 8. Tabulación de resultados Preg. 1 Usua.

Análisis e interpretación de resultados

De acuerdo con los resultados detallados en la Figura 8, **el 51% de los usuarios encuestados manifestó utilizar parqueaderos públicos con mucha frecuencia**, mientras con frecuencia el 42%, ocasionalmente el 5% y raramente el 2%, ningún encuestado manifestó que sea nada frecuente su uso de parqueaderos en la ciudad de

Ambato, siendo una notable tendencia a la demanda frecuente por usar parqueaderos públicos, existiendo necesidad de optimizar tiempos en tareas cotidianas.

Pregunta 2.

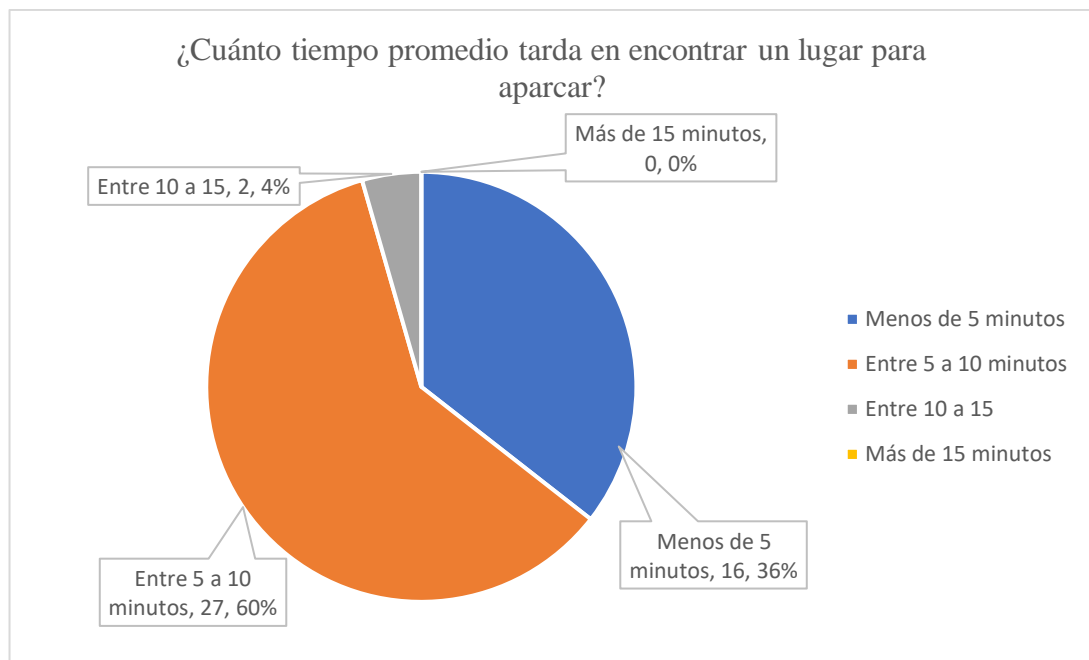


Figura 9. Tabulación de resultados Preg. 2 Usua.

Análisis e interpretación de resultados

Según los resultados de la Figura 9, **el 60% de los usuarios encuestados manifestó que tarda entre 5 a 10 minutos en encontrar un lugar libre para parquear su vehículo**, el 36% indicó que se tarda menos de 5 minutos, el 4% dijo que tarda más, entre 10 a 15 minutos, mientras que ningún encuestado dijo que se tarde más de 15 minutos en aparcar su vehículo, notando ineficiencia en esta tarea sencilla que, bien manejada debe llevar muy pocos minutos, estimando menos de 3 minutos.

Pregunta 3.

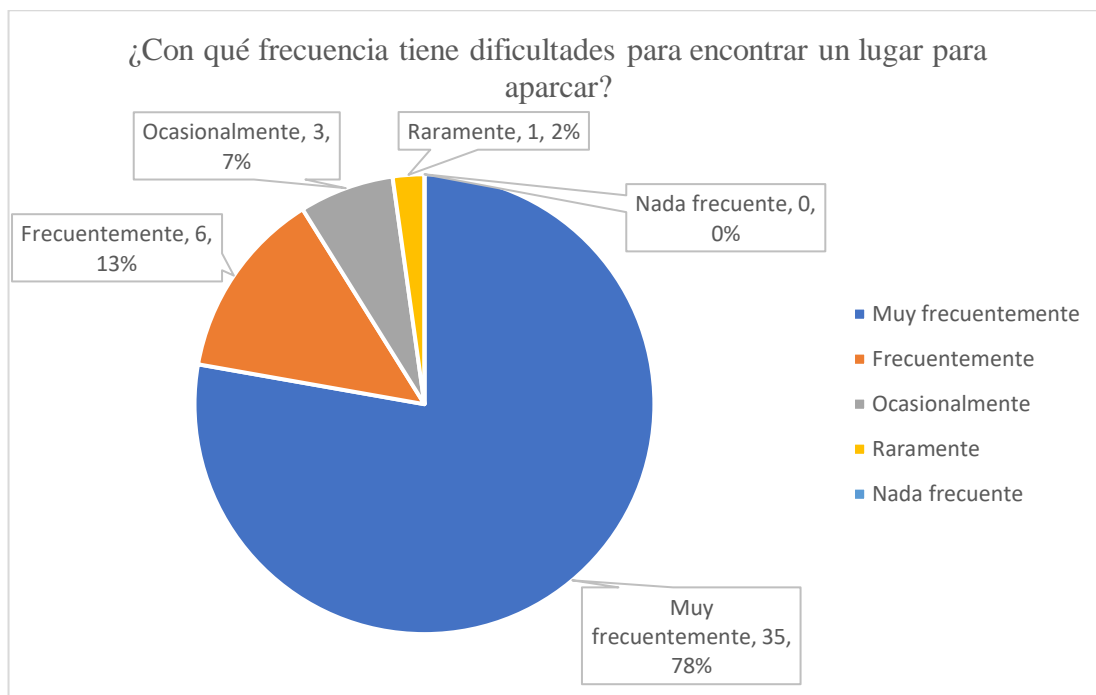


Figura 10. Tabulación de resultados Preg. 3 Usua.

Análisis e interpretación de resultados

Según los resultados mostrados en la Figura 10, **el 78% de los encuestados dijo que es muy frecuente tener problemas para encontrar un lugar para parquear su vehículo**, un 13% dijo que es frecuente tener problemas en esta tarea, mientras que el 7% manifestó que este problema es ocasional, el 2% dijo que raramente tiene problemas en esta tarea y nadie mencionó no tener ningún problema en encontrar un sitio para dejar su auto en un parqueadero público, dando a entender que necesita ayuda para encontrar un sitio adecuado para aparcar su vehículo en un tiempo prudente.

Pregunta 4.

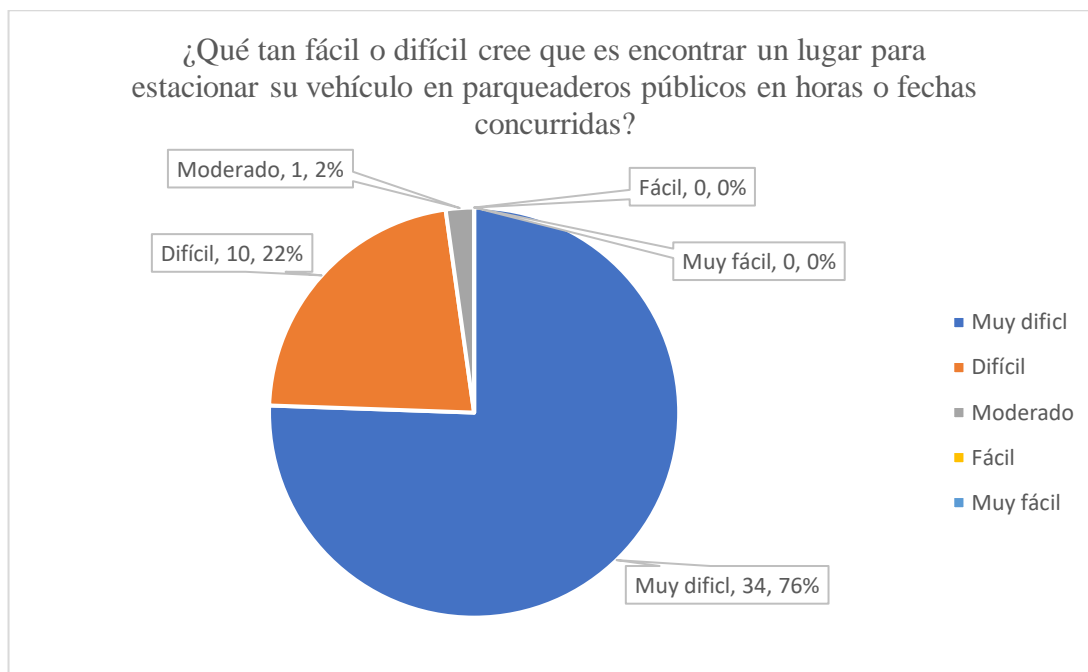


Figura 11. Tabulación de resultados Preg. 4 Usua.

Análisis e interpretación de resultados

Según los resultados abordados en la Figura 11, **al 76% de los usuarios encuestados les resulta muy difícil encontrar un lugar para parquearse en horas pico o concurridas** además de feriados y fechas especiales en parqueaderos de la ciudad, mientras que al 22% le resulta difícil, el 2% lo considera de dificultad moderada y ningún usuario encuestado consideró esta tarea como fácil o muy fácil, el porcentaje mayoritario manifestando que le es muy difícil encontrar un lugar adecuado para dejar su auto, da la pauta a que se busquen soluciones a la asignación de espacios disponibles en parqueaderos públicos en la ciudad.

Pregunta 5.

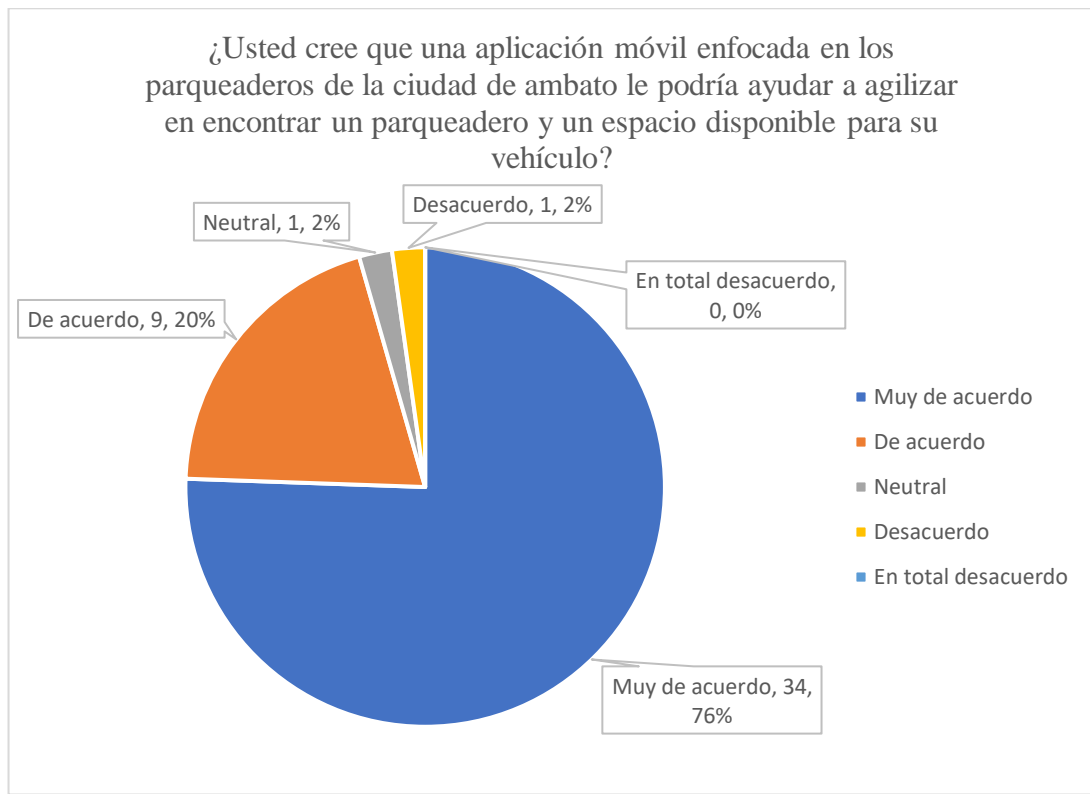


Figura 12. Tabulación de resultados Preg. 5 Usua.

Análisis e interpretación de resultados

Según con los resultados evidenciados en la Figura 12, **el 76% de los usuarios encuestados están muy de acuerdo con que una aplicación para el uso de parqueaderos en la ciudad de Ambato les ayudaría a bajar el tiempo que se gasta en encontrar un sitio disponible para su vehículo**, el 20% está de acuerdo con esto, el 2% se mostró neutral y solo un 2% se mostró en desacuerdo, ningún usuario encuestado se mostró en total desacuerdo con este ítem del instrumento, notando la acogida positiva a una solución móvil para la gestión de los parqueaderos y así verse beneficiados en los tiempos que emplean a diario para aparcar sus respectivos vehículos.

Resultados ficha de observación a un parqueadero.

Nombre o ubicación del parqueadero: Parqueadero Paseo Shopping

Fecha y hora de la observación: 18 de abril 2023, 19H54

N	Descripción	Observación
1	Tamaño del parqueadero	Gran número de espacios designados al parqueo de vehículos
2	Método de asignación de espacios	Libre elección
3	Disponibilidad de espacios	Parcialmente ocupado
4	Cantidad de usuarios buscando espacios disponibles	Ocho dentro del campo visual
5	Tiempo promedio que los usuarios dedican a la búsqueda de un espacio disponible	8 minutos
6	Comportamientos observados de los usuarios al buscar un sitio disponible	Conducen en círculos al menos 3 usuarios, tienen sitios de preferencia
7	Dificultades que enfrentan los usuarios en la búsqueda de espacios de estacionamiento	Falta de señalización para entrar al sitio de forma cómoda
8	Otras observaciones relevantes	Parqueadero de varios pisos, mucho más tiempo en buscar sitio entre pisos
9	Principales problemas identificados en la búsqueda de espacios de parqueo	Favoritismo a los espacios del piso superior, casi siempre está lleno así que se gasta tiempo al

		no saber si hay disponibilidad de espacios libres.
10	Frecuencia y gravedad de problemas observados	Personas conduciendo en círculos al menos uno siempre en búsqueda de su sitio de preferencia

Tabla 4. Resultados ficha de observación

El parqueadero observado tiene gran disponibilidad de espacios, cuenta con un parqueadero pequeño en la planta superior, por su rápido acceso a ciertos locales este suele ser preferido por muchas personas, por dicha razón tiende a no tener espacios disponibles, los usuarios pierden tiempo en subir a este apartado sin tener información previa de si hay o no espacios disponibles.

2.2.4. Procesamiento y análisis de datos

En base a la información recolectada mediante la ficha de observación y encuestas enfocadas a usuarios y personal de control de parqueaderos públicos en la ciudad de Ambato, se determina que:

- La gran mayoría de usuarios reconoce problemas muy frecuentes al momento de buscar y seleccionar un espacio disponible para parquear su vehículo en parqueaderos públicos en la ciudad de Ambato.
- Gran parte del personal de control consultado con las encuestas reconoce problemas en la asignación de espacios por elección libre en sus parqueaderos de gran dimensión.
- Los usuarios identifican un tiempo desperdiciado considerable que convierte en tediosa la sencilla tarea de parquear su vehículo en parqueaderos públicos en la ciudad Ambato.
- La preferencia de los usuarios por espacios para parquear su vehículo provoca congestión en estos espacios, gastando valioso tiempo en la recurrente consulta

de espacios disponibles de su preferencia al estar estos ocupados la mayor parte del día.

- Los usuarios reconocen que un aplicativo móvil que les informe con antelación la ubicación de espacios disponibles para el parqueo de sus vehículos les sería de gran utilidad y así evitar la pérdida de tiempo en búsqueda y asignación de esta sencilla tarea.

CAPÍTULO III.- RESULTADOS Y DISCUSIÓN

3.1. Análisis y discusión de los resultados

Basado en la información recopilada a través de observaciones y encuestas dirigidas a usuarios y personal de control de parqueaderos públicos en Ambato, se revela un patrón claro de desafíos en el proceso de estacionamiento. Los usuarios enfrentan dificultades frecuentes al buscar y elegir espacios disponibles, lo que resulta en la pérdida de tiempo y frustración. Además, la preferencia por ciertos espacios contribuye a la congestión y a la constante necesidad de verificar la disponibilidad.

Se destaca la demanda unánime de los usuarios por una solución que anticipe la disponibilidad de espacios, optimizando así la experiencia de estacionamiento en los parqueaderos públicos de la ciudad, dando apertura a la tecnología con la que se desarrollará una aplicación móvil: Flutter.

3.1.1. Flutter, que es y su estado actual

“Flutter es un marco de código abierto desarrollado y compatible con Google. Los desarrolladores de front-end y pila completa utilizan Flutter para crear una interfaz de usuario (IU) de aplicación para varias plataformas con un único código base.”[17].

Desde su creación en 2018. Flutter, por parte de Google y al ser Open Source, por parte de su comunidad, que día a día es más grande alrededor del mundo, se ha enfocado en brindar un entorno de desarrollo más rápido cuando el propósito del proyecto es la compatibilidad multiplataforma.

En el entonces de su creación, su enfoque multiplataforma era netamente móvil, es decir, su código base tenía como objetivo poder ejecutarse en Android y también en iOS, ahora este poderoso framework es compatible con seis plataformas en total, estas son: Windows, web, MacOS, Android, iOS y Linux.

Al abarcar todas estas plataformas que están presentes en la gran mayoría de dispositivos existentes, también tienen presencia en sistemas integrados como dispositivos de Amazon y Google para el hogar.

En Flutter, los desarrolladores pueden crear interfaces de usuario mediante el uso de widgets, que abarcan desde los elementos visuales más básicos y simples como botones, textos, íconos y demás pequeños elementos visuales hasta ventanas, paneles y pantallas completas, estos widgets son altamente personalizables gracias a la metodología de composición que emplea, se puede decir que gran porcentaje de los componentes de Flutter son widgets, estos se pueden combinar o modificarse según la necesidad del desarrollador o necesidad puntual del usuario.

Flutter utiliza su propio motor gráfico, llamado Skia, derivado del llamado proyecto Sky del cual llegó a los desarrolladores como Flutter en su versión beta en marzo de 2018 y su primera versión oficial en diciembre del mismo año[18], en vez de depender de los widgets integrados en la plataforma, lo que garantiza una apariencia muy similar en todas las plataformas, logrando una mayor flexibilidad de desarrollo, controlando cada píxel de la pantalla para renderizar el contenido programado, inclusive algunos widgets de Flutter pueden hacer funciones que los widgets específicos de la plataforma no logran hacer.

Flutter también facilita y fomenta el uso de widgets creados, actualizados y soportados por la comunidad, se logra al admitir múltiples bibliotecas de widgets y promover a la comunidad a seguir creando nuevos widgets, implementando nuevas funcionalidades, optimizaciones y tolerancia a fallos por parte de la comunidad en constante crecimiento[17].

Pero estos widgets al ser tan variados se pueden clasificar por su tipo, el catálogo contiene 14 categorías de widgets, con componentes Material design de la propia Google, Cupertino (Nombre adquirido por la locación de la sede principal de Apple) de iOS, sumando temas y diseños previos como temas que son de ayuda para que los desarrolladores puedan crear interfaces de usuario de forma aún más rápida.

El soporte de Flutter, de forma más específica, cuenta con apoyo de Google de forma oficial, pero cabe mencionar que su comunidad Open Source activa se encuentra principalmente en Reddit, Slack, Discord, Stack Overflow y Gitter. Estas partes dan cada vez mayor protagonismo al Framework que en 2022 extendió finalmente su soporte para escritorio multiplataforma al soportar Linux y MacOS.

Google, con el propósito de aprender el aprendizaje de su framework, elaboró y mantiene al día una basta documentación originalmente en inglés, traducida a varios idiomas por grupos de su comunidad con idiomas específicos, como la comunidad de ES Flutter que tradujo y mantiene la documentación oficial actualizada en español, disponible en el sitio www.esflutter.dev, además Flutter cuenta con su propio canal de YouTube con tutoriales y actualizaciones del framework, así como secciones interesantes para mantener una comunidad interesada y activa, como la sección de Widget de la semana, donde se habla de un Widget en concreto, sus usos posibles y como implementarla en nuestros proyectos[17].

Google también involucra a Flutter en sus grandes eventos a nivel mundial, tal es el caso del tan conocido evento anual “Google I/O”, su edición 2023, inaugurado el 10 de mayo, se encuentra en desarrollo exactamente en días en la que esta parte de la investigación se llevó a cabo, todas las herramientas y documentación por parte de la comunidad se pueden encontrar en www.pub.dev.

Flutter se puede entender de forma más sencilla al entender sus dos partes fundamentales que lo conforman:

- SDK (Software Development Kit): “Se trata de una colección de herramientas que permite desarrollar aplicaciones, esto incluye elementos para compilar código para diferentes plataformas.”[18]
- Framework (Biblioteca de interfaz de usuario basada en widgets): “Una colección de elementos de interfaz de Usuario (UI) reutilizables como botones, entradas de texto, controles deslizantes, etc.; que pueden personalizarse según requiera el proyecto.”[18]

Una característica muy útil a la hora de desarrollar en Flutter es sin duda, la capacidad de reflejar los cambios en caliente, sin tener que reinstalar toda la aplicación, a esta funcionalidad el equipo de Flutter ha llamado “Hot Reload”. En forma concreta y técnica, “preservan el estado de tu app. Este diseño te permite ver el efecto de los cambios más recientes, sin perder el estado actual.”[19]

A esta funcionalidad le suma el equipo de Flutter la funcionalidad hermana llamada “Hot Restart” que recarga toda la app para retomar nuevamente todos los datos y el contexto de la aplicación de forma rápida, esto es distinto a una reinstalación de la aplicación que tarda mucho más que la funcionalidad de reinicio en caliente.

Estas funcionalidades son de gran ayuda para el equipo de desarrollo y el tiempo que emplean en hacer cambios y ver reflejados en el dispositivo o emulador que usen, ya que, en el desarrollo, gran parte del tiempo es empleado en verificar pequeños cambios en el diseño y respuesta en efectos y transiciones, colores, tamaños, estructura de componentes, dimensiones o sombreados de cómo se renderiza el widget en pantalla.

En el evento Google I/O 2023, se dedicó un momento a novedades a futuro de Flutter, se dijo “presentamos la visión de Flutter para los próximos años. Se puede resumir en cuatro temas: un enfoque continuo en la experiencia del desarrollador, un rendimiento gráfico innovador, una integración sin fisuras para la web, el escritorio y los dispositivos móviles, y una llegada temprana a nuevas arquitecturas emergentes. Veamos dónde estamos en algunas de esas cosas.

En primer lugar, acabamos de anunciar Flutter 3.10, una versión centrada en ayudarte a construir experiencias atractivas, suaves y expresivas de manera más productiva, independientemente de la plataforma de destino. En dispositivos móviles, anteriormente anunciamos el soporte de Impeller en iOS, un nuevo motor de renderizado construido para Flutter para maximizar el rendimiento y la calidad, que ahora está activado por defecto. Impeller debería ayudar con las animaciones complejas que experimentan sacudidas.”[20].

El enfoque al futuro cercano de Flutter es potenciar su motor gráfico, pero mejorar aún más la potencia de sus desarrollos en Android y sobre todo en iOS.

3.1.2. Dart

El SDK de Flutter se basa en el lenguaje de programación Dart, relativamente nuevo, también creado por Google con el fin de convertirse en el “nuevo” JavaScript, así que de igual manera, en la plataforma web, Dart se ejecuta directamente en el navegador, esto no sería posible sin un transpilador llamado Dart2js[18].

En la actualidad, los lenguajes de programación que se desarrollan se ven obligados a enfocarse en el desarrollo de aplicaciones móviles, por lo que el lenguaje de programación que se disponga a ser usado debe dar una buena experiencia con menor y simple sintaxis, además tener en cuenta la memoria del dispositivo en la que se implementa el aplicativo[21].

“La principal diferencia que presenta Dart frente a otros lenguajes de programación es que es más fácil de leer ya que la sintaxis de este se acerca al lenguaje humano, tiene más posibilidades y menos comandos[22]”.

Características de Dart

- Programación estructurada y flexible: “Los ingenieros de Google diseñaron Dart de manera que pueda ser utilizado en proyectos de una sola persona hasta proyectos más desarrollados o complejos.[23]”
- Permite la adaptación de nueva herramienta a cualquier navegador web: “El lenguaje de programación Dart se puede ejecutar de dos maneras, en una máquina virtual o en un motor de JavaScript utilizando un compilador para traducir el código. Esto le permite adaptarse a cualquier navegador.[23]”
- Lenguaje basado en clases e interfaces o POO (Programación orientada a objetos): “Gracias a sus basamentos en clases o en la programación orientada a objetos, se facilita la encapsulación y la reutilización de código.[23]”

Una de las preguntas que pueden surgir es la de, ¿Por qué Flutter usa Dart y no otro lenguaje de programación?

El primer equipo de Flutter examinó una variedad de lenguajes de programación antes de elegir el lenguaje de programación Dart para el desarrollo de la plataforma, Dart encajaba con el enfoque que tenían para la construcción de interfaces de usuario.

Las características de Dart que son particularmente útiles para Flutter incluyen su capacidad para ser compilado AOT (Ahead of time) es decir, compilar un lenguaje de programación de alto nivel o nivel intermedio a un código de máquina nativo, rápido y predecible, lo que permite una alta personalización y velocidad de desarrollo aún más rápida[23].

Dart también se puede compilar JIT (Just in time) que implica, la compilación durante la ejecución de la aplicación o programa, característica que permite una recarga en caliente conllevando ciclos de desarrollo ágiles, también facilita la creación de animaciones y fluidas transiciones al proporcionar una asignación de objetos y un garbage colector (recolector de basura) sin bloqueos.

Dart es útil para Flutter porque permite evitar la necesidad de un lenguaje de diseño declarativo separado como JSX o XML, lo que simplifica el proceso de desarrollo y hace que la creación de interfaces sea más fácil de visualizar. Los desarrolladores también encuentran que Dart es fácil de aprender debido a sus características familiares tanto para los usuarios de lenguaje estáticos como dinámicos[23].

La combinación de características de Dart lo hace particularmente poderoso para la implementación de Flutter, y es difícil imaginar que Flutter sería tan efectivo sin él. Los desarrolladores que utilizan Dart encuentran que es una herramienta valiosa en su arsenal, sumando a esto características que llegan a este poderoso lenguaje mediante el soporte que recibe que lo seguirán haciendo más poderoso con el pasar del tiempo[23].

3.1.3. Ventajas y desventajas de usar Flutter en el desarrollo móvil

Como ventajas del uso de Flutter en el desarrollo de aplicaciones móviles es que permite la creación de aplicaciones compatibles para iOS y Android, con un único

código programado, lo que permite ahorrar tiempo en disponer de aplicaciones y en la mayoría de los casos reducir el equipo de desarrollo tanto en desarrollo como en mantenimiento ya que no se tiene que hacer el cambio o actualización en dos aplicaciones diferentes[24].

Por el lado de Dart, el lenguaje es muy intuitivo y fácil de aprender, recordando que el paradigma de programación de Flutter es de tipo Widget, lo que permite la modularidad y así facilitando el mantenimiento, cabe mencionar que guarda varias similitudes con otros lenguajes de alto nivel como C o Java, como el tipado estricto y que si se viene de estos lenguajes, la adaptación a Dart será cómoda y rápida[24].

Tiene numerosas librerías con cantidad considerable de librerías con muchos elementos gráficos pre creados que propician una rápida creación de interfaces de usuario, cada pequeño elemento requerido para crear la aplicación, como botones, listas, columnas o acceso a recursos del dispositivo móvil como la cámara, ya están predefinidos, acortando tiempo valioso de desarrollo[24].

Otro aspecto que aventaja a Flutter es su performance o rendimiento bastante alto, incluso comparándose con aplicaciones de tipo nativas, con las funciones predefinidas de scroll, navegación, gestión de permisos, que logran dar la impresión de rendimiento nativo con cargas muy rápidos en iOS y Android a diferencia de otros frameworks multiplataforma[24].

La creación de experiencia de usuario con Flutter es sencilla y flexible, con interfaz cuidada a detalle, debido a su composición por capas, la personalización es absoluta, dando una calidad alta de experiencia de uso al usuario con la posibilidad de combinar elementos visuales características tanto de iOS y Android con sus principales fuentes de diseño como son Cupertino y Material Design respectivamente.

El aspecto de la seguridad con Flutter es algo a destacar, ya que con relativamente pocos años de vida ha generado seguridad y confianza en los equipos de desarrollo que lo usan. El hecho de que una gigante tecnológica como Google esté detrás de este poderoso framework genera confianza sabiendo su experiencia en desarrollo de software y tecnología[24].

Por otro lado, Flutter también tiene apartados a considerar como desventajas en su uso. Al tratarse de un framework reciente, la cantidad de recursos aún se puede considerar limitada, por lo que en ocasiones puntuales la investigación sobre como implementar opciones muy específicas será vital para sacar adelante tal proyecto, lo que podría causar frustración en los desarrolladores al no encontrar respuestas rápidas a sus problemas por parte de sus creadores o de su comunidad, frente a otros frameworks que tienen más años en el mercado, caso muy parecido con Dart, que al ser comparado con el tiempo en el mercado de otros lenguajes de programación de alto nivel como Java este se verá en desventaja respecto a las fortalezas que un lenguaje de programación adquiere con el paso del tiempo.

También se puede ver afectado por la dificultad de integración y manutención de bibliotecas de terceros, ya que los módulos se integran de forma estática en la aplicación, siempre que se haga una actualización, el programa se compilará y volverá a instalar estas integraciones, afectando en medida al mantenimiento de este aspecto[24].

Característica	Descripción
Lenguaje de programación	Dart
Plataformas de destino	iOS, Android, web, escritorio y dispositivos integrados
Estilo de desarrollo de UI	Declarativo
Biblioteca de widgets	Conjunto rico de widgets personalizables y adaptables
Hot Reload	Permite ver los cambios realizados en el código en tiempo real
Integración de terceros	Admite la integración con servicios de terceros, como Google Maps, Facebook, Twitter
Rendimiento	Alta velocidad gracias a su motor gráfico de renderizado personalizado

Comunidad	Comunidad activa y creciente de desarrolladores que mejoran y agregan bibliotecas y paquetes adicionales para extender funcionalidades
-----------	--

Tabla 5. Tabla resumen Flutter

3.1.4. Ejemplos de desarrollos multiplataforma con Flutter

Entre las aplicaciones que más renombre pueden tener en el mundo que han sido desarrolladas con Flutter para funcionalidad multiplataforma encontramos:

- Alibaba: “La compañía de comercio online más grande del mundo, ha usado Flutter para crear una hermosa experiencia para iOS y Android en su app Xianyu, que cuenta con más de 50 millones de descargas[25]”.
- Google ADS: “La app de Google ADS te ayuda a mantener tus campañas en funcionamiento – no importa a donde te lleve tu negocio[25]”.
- Reflectly: “Una hermosa aplicación de diario y mindfulness dirigida por inteligencia artificial[25]”.
- Tencent: “Now Live es un servicio de video en streaming de Tencent con decenas de millones de MAU[25]”.

La lista es realmente extensa, sin embargo, hay una lista de aplicaciones con menos renombre, pero con mucha calidad en su desarrollo y despliegue, se puede encontrar estas aplicaciones en el sitio www.itsallwidgets.com que claramente hace referencia a como está construido Flutter, a base de widgets.

En este sitio promovido por la comunidad, para la comunidad y soportado por www.invoiceninja.com procura tener una lista abierta para que se puedan ver las grandiosas aplicaciones desarrolladas tanto para plataformas móviles, como para plataformas de escritorio y entorno web, al momento de esta redacción se evidenciaron un total de 3223 aplicaciones móviles indexadas a este sitio, con muchos recursos para la comunidad de Flutter.

3.2. Determinación de la metodología de desarrollo

Definir una metodología de desarrollo es una tarea crucial en cualquier proyecto de software, ya que proporciona una guía sistemática para llevar a cabo el proceso de desarrollo de manera efectiva. En general, las metodologías de desarrollo establecen un conjunto de reglas y pautas que se deben seguir en todo el ciclo de vida del software, desde la planificación y el análisis hasta la implementación, prueba y mantenimiento.

Metodología	RAD	Mobile-D	XP
Tipo de proyecto	Desarrollo móvil y web.	Desarrollo móvil	Desarrollo de software en general
Fases	<ol style="list-style-type: none"> 1. Planificación inicial y definición de requisitos 2. Prototipado rápido 3. Feedback del usuario 4. Integración y pruebas 5. Despliegue y mantenimiento[26] 	<ol style="list-style-type: none"> 1. Fase de exploración 2. Fase de iniciación 3. Fase de producto 4. Fase de estabilización 5. Fase de pruebas[27] 	<ol style="list-style-type: none"> 1. Planificación 2. Diseño 3. Codificación 4. Pruebas 5. Lanzamiento [28]
Características	Enfocada en el desarrollo rápido y eficiente de aplicaciones, fomenta la colaboración entre equipos multidisciplinarios, proporciona una	Enfocada en el desarrollo de aplicaciones móviles, utiliza un enfoque iterativo e incremental, se centra en la movilidad, la usabilidad y el	Enfocada en la calidad del software y la satisfacción del cliente, Promueve la colaboración estrecha con el cliente, se basa en la programación en

	respuesta rápida a los cambios y requerimientos del cliente.	rendimiento, adaptable a diferentes plataformas móviles	parejas y en pruebas unitarias frecuentes
Descripción general	Es un método de desarrollo de aplicaciones de software basado en creación de prototipos sin una estructuración ni planificación específica. En cambio, se priorizan las tareas de desarrollo, las cuales deben ser desarrolladas en cortos periodos de tiempo[26]	Se basa en el desarrollo de sucesivas fases, por evolución de sistemas más simples a sistemas más complejos. Además, utiliza el modelo de desarrollo en espiral como base, e incorpora procesos de evaluación de usabilidad priorizado a la participación del usuario en todos los procesos de ciclo de vida de diseño[29]	La metodología XP es un conjunto de técnicas que dan agilidad y flexibilidad en la gestión de proyectos. También es conocida como Programación Extrema (Extreme Programming) y se centra crear un producto según los requisitos exactos del cliente[28]

Tabla 6. Tabla comparativa metodologías ágiles para desarrollo móvil

Por el tipo de proyecto y el tiempo disponible para el desarrollo de la propuesta al problema, Mobile-D se considera como la metodología adecuada por su flexibilidad y la rapidez ante cambios durante el desarrollo de la aplicación móvil para cumplir con los planteamientos de funcionalidades y alcance del desarrollo.

“El objetivo de esta metodología es conseguir ciclos de desarrollo muy rápidos en equipos muy pequeños. Se basa en metodologías para el desarrollo de aplicaciones móviles conocidas pero aplicadas de forma estricta como: extreme programming, Crystal Methodologies y Rational Unified Process. Tiene distintas fases: exploración, inicialización, fase de producto, fase de estabilización y la fase de pruebas. Cada una tiene un día de planificación y otro de entrega.”[27]

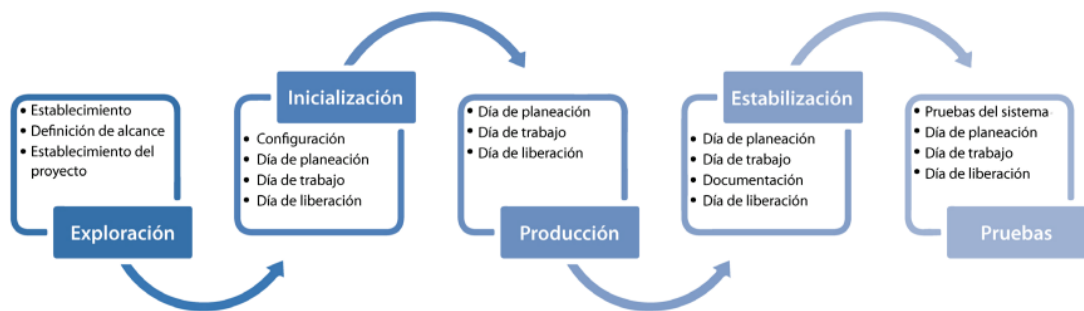


Figura 13. Proceso de Mobile-D[30]

En el caso de la metodología Mobile-D, se enfoca en un equipo integrado recomendable que no sea mayor a 10 personas conviviendo en el mismo espacio de trabajo para desarrollar aplicaciones móviles. El equipo incluye no solo desarrolladores, sino también expertos en diseño de experiencia de usuario, pruebas y calidad, que trabajan juntos para crear aplicaciones móviles eficientes y atractivas, permitiendo cambios breves durante el desarrollo, la metodología es aplicable a equipos unitarios (un solo desarrollador) siendo el caso de este proyecto investigativo que tiene como producto una aplicación móvil multiplataforma.

3.3. Desarrollo de la propuesta

Dada la metodología escogida, se procedió al desarrollo de la propuesta mediante la aplicación de las cinco fases de Mobile-D.

3.3.1. Fase I: Exploración

En esta fase se establecieron los objetivos y el alcance del proyecto, así como los stakeholders del proyecto.

- **Stakeholders**

- Usuarios de la aplicación: Usuarios de parqueaderos públicos de la ciudad de Ambato
- Equipo de desarrollo: Luis Salazar “Autor del proyecto”, como equipo unitario

- **Definición de alcance**

- Las funcionalidades que tiene la aplicación móvil son acordes a los resultados obtenidos en la recolección de información indicados en la sección 2.2.2 más atrás del proyecto.
- La información será manejada en la base de datos en MongoDB y se obtendrán mediante peticiones HTTP por Api Rest.
- La aplicación móvil funciona como un prototipo a implementarse a un entorno real.
- Limitaciones: Conexión a internet necesaria

- **Establecimiento del proyecto**

Las herramientas escogidas para el desarrollo de la propuesta fueron:

- Visual Studio Code: IDE desarrollado por Microsoft, de uso libre y multiplataforma, potente y versátil en gran parte por la cantidad y calidad de plugins a más de atajos que mejoran la experiencia y productividad del desarrollador[31]
- MongoDB: SGBD no relacional basado en documentos BSON, de código abierto y uso gratuito[32]

- NodeJS: Framework de JS que permite ejecutarlo en el servidor, de ejecución multiplataforma utilizado para crear aplicaciones de red escalables y en tiempo real[33]
- Express: Framework de NodeJS que permite estructurar y manejar solicitudes HTTP que simplifica la creación de aplicaciones web y Api Rest[34]
- Flutter: Framework multiplataforma explicado en la sección 3.1.1

3.3.2. Fase II: Inicialización

Esta segunda fase tuvo como resultado el diseño de la aplicación móvil.

- **Configuración del ambiente de desarrollo**

El ambiente de desarrollo se configuró con las herramientas para cumplir con la propuesta y los objetivos correspondientes.

- Tipo de aplicación: Multiplataforma
- Desarrollo front-end:
 - Framework: Flutter
 - Lenguaje de programación: Dart
- Desarrollo back-end:
 - Framework: NodeJS
 - Api Rest: Express
- Sistema de gestión de base de datos (SGBD): MongoDB

- **Preparación del ambiente de desarrollo**

Las herramientas y configuraciones usadas para el desarrollo son:

- MongoDB
- Api Rest con Express
- Flutter

- **Planificación de fases**

Fase	Iteración	Actividades clave	Descripción
Exploración	Iteración 1	- Identificar usuarios y sus necesidades	Definición de alcance y stakeholders
Inicialización	Iteración 2	- Establecer las tecnologías a usar - Desarrollar prototipos de UI	Configuración del ambiente de desarrollo
	Iteración 3	- Diseño de BD -Análisis de requerimientos iniciales	
Producción	Iteración 4	- Desarrollo de funcionalidad registro e inicio de sesión de usuarios. - Visualización y actualización de datos de usuario - Testeo de funcionalidad	Desarrollo de funcionalidades principales
	Iteración 5	- Inserción de parqueaderos con administradores y esquemas correspondientes -Visualización del listado de parqueos -Testeo de la funcionalidad	

	Iteración 6	- Desarrollo de módulo de alquiler de espacio disponible con todas las partes involucradas -Testeo de la funcionalidad	
	Iteración 7	-Visualización de alquiler en marcha y visualización de historial de uso - Implementación de reseñas de usuarios a parqueaderos - Testeo de funcionalidad	
Estabilización	Iteración 8	- Ajustes al diseño y ajustes de funcionalidades según observaciones de tutor	Pruebas y correcciones
Pruebas	Iteración 9	- Pruebas generales y feedback de posibles usuarios, observaciones a posibles optimizaciones	Pruebas exhaustivas, optimización

Tabla 7. Planificación de fases

- **Diseño de BD**

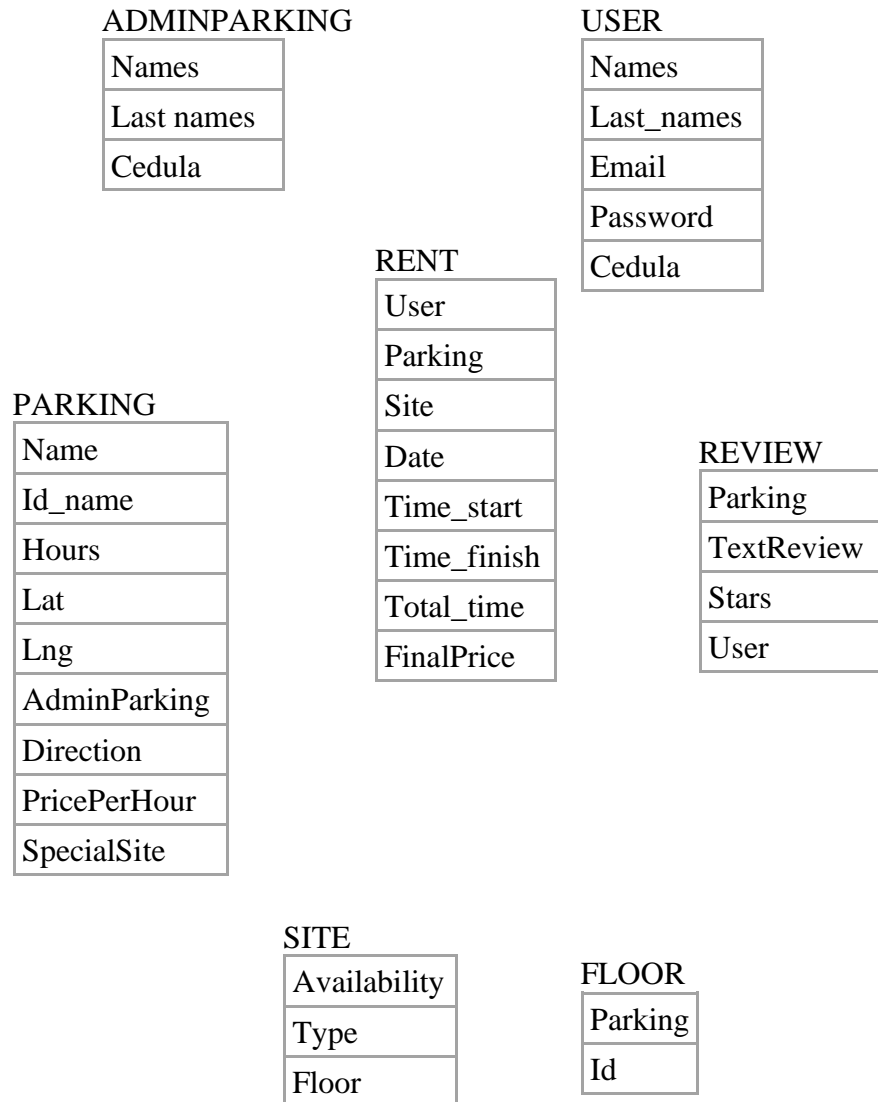


Figura 14. Diseño de base de datos no relacional

- **Arquitectura de la aplicación móvil**

En la arquitectura de desarrollo propuesta, en la Figura 15, se observa la forma en la que funciona el prototipo del proyecto, en la cual solo el Administrador (Investigador) ingresa datos de parqueaderos y administradores de parqueaderos, sin un apartado diseñado a medida de las necesidades de un administrador real ya que el investigador ingresa los datos mínimos necesarios para que la aplicación funcione mediante peticiones HTTP realizadas por Postman.

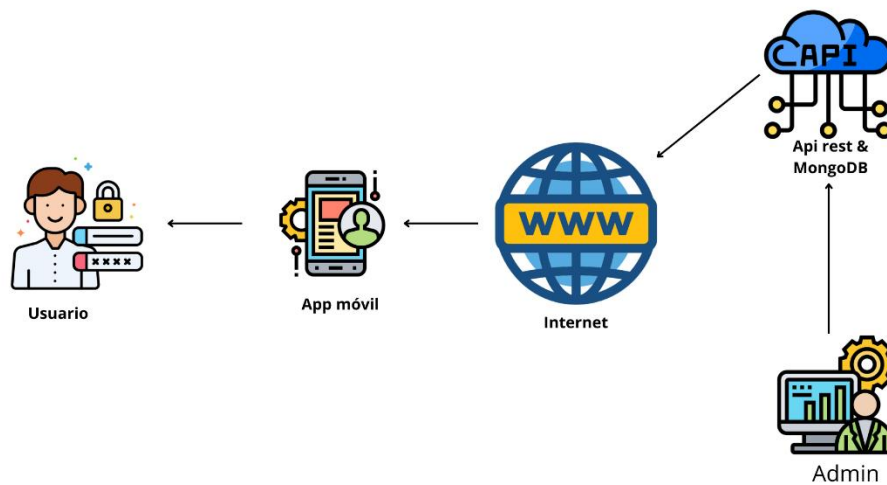


Figura 15. Arquitectura de proyecto

- **Funciones establecidas**

Con los resultados obtenidos en la recolección y procesamiento de datos y acorde a requerimientos establecidos, se estableció las funcionalidades mínimas para lograr un MVP (Producto mínimo viable).

Funcionalidad	Parámetros	Actividad
Registro de Administrador de parqueadero	Nombres	El administrador inserta nombres, apellidos y cédula del administrador de parqueadero afiliado.
	Apellidos	
	Cédula	

Tabla 8. Funcionalidades Administrador: Registro de administrador de parqueadero

Funcionalidad	Parámetros	Actividad
Registro de parqueadero	Nombre	El administrador inserta todos los parámetros necesarios para la identificación y funcionamiento de un parqueadero afiliado
	Id nombre	
	Horario	
	Lat.	
	Lng.	
	Administrador de parqueadero	
	Dirección	
	Precio por hora	
	Sitios especiales	

Tabla 9. Funcionalidades Administrador. Registro de parqueadero

Funcionalidad	Parámetros	Actividad
Registro de pisos pertenecientes a un parqueadero	Id nombre	El administrador inserta los parámetros para identificación y posterior mapeo de la planta
	Parqueadero	

Tabla 10. Funcionalidades Administrador. Registro de pisos

Funcionalidad	Parámetros	Actividad
Registro de cada sitio perteneciente a un piso	Número	El administrador inserta parámetros de disponibilidad, ubicación y tipo de cada sitio
	Disponibilidad	
	Tipo	
	Piso	

Tabla 11. Funcionalidades Administrador. Registro de sitios

Funcionalidades para que el usuario interactúe con la aplicación móvil

Funcionalidad	Parámetros	Actividad
Registro en la aplicación	Nombres	El usuario debe ingresar sus datos personales para ser dado de alta en la aplicación móvil
	Apellidos	
	Correo	
	Contraseña	
	Cédula	

Tabla 12. Funcionalidades Usuarios: Registro

Funcionalidad	Parámetros	Actividad
Inicio de sesión	Correo	El usuario debe ingresar el correo y contraseña proporcionados en el registro
	Contraseña	

Tabla 13. Funcionalidades Usuarios: Inicio de sesión

Funcionalidad	Opción	Actividad
Pantalla de inicio	Mapa general	El usuario puede visualizar marcadores de parqueaderos registrados
	Listado de parqueaderos	Acceso al listado de parqueaderos registrados
	Historial parqueos asignados	Acceso a listado de parqueos escogidos con anterioridad
	Menú inferior	Acceso a parqueo en curso y datos de usuario

Tabla 14. Funcionalidades Usuarios: Pantalla inicio

Funcionalidad	Opción	Actividad
Visualizar parqueos registrados	Lista parqueos	El usuario podrá ver un listado general de parqueaderos registrados en la app en una tarjeta que lleva el nombre, ID, mapa con marcador, dirección, disponibilidad de sitios para personas con discapacidad, tarifa y la calificación promediada por las reseñas que ha recibido cada parqueadero

Tabla 15. Funcionalidades Usuarios: Listado de parqueaderos

Funcionalidad	Opción	Actividad
Visualizar parqueos realizados con anterioridad	Lista de parqueos realizados	El usuario podrá visualizar las reservas de parqueo que ya ha realizado hasta ese momento, si es nuevo en la app esta lista estará vacía

Tabla 16. Funcionalidades Usuarios: Historial de parqueos

Funcionalidad	Opción	Actividad
Visualizar pisos que el parqueadero tiene disponibles	Listado de pisos	El usuario podrá visualizar y elegir uno entre el listado de pisos que el parqueadero elegido tiene disponibles, desde un piso hasta edificios completos dependiendo del caso

Tabla 17. Funcionalidades Usuarios: Pisos disponibles

Funcionalidad	Opción	Actividad
Visualizar esquema de piso y seleccionar sitio disponible	Selección de sitio disponible	El usuario podrá visualizar el esquema de todo el piso y que sitios se encuentran disponibles, podrá elegir uno solo para su uso.

Tabla 18. Funcionalidades Usuarios: Selección de sitio disponible

Funcionalidad	Opción	Actividad
Visualizar detalles de sitio, parqueadero, tarifa y hora si anteriormente ha elegido y confirmado un sitio para su uso, para poder finalizar cuando lo desee	Visualización de parqueo en curso y finalización de uso	El usuario podrá visualizar los detalles del parqueo que ha elegido y podrá finalizar el uso del sitio para ver el total del costo y el tiempo total de su uso

Tabla 19. Funcionalidades Usuarios: Parqueo en curso / Finalización

Funcionalidad	Parámetros	Actividad
Escribir una reseña de parqueadero con una calificación de 1 a 5, un título y texto reseña sobre su experiencia de uso	Puntuación	El usuario podrá calificar al parqueadero, titular y escribir una reseña por su experiencia de uso en el parqueadero
	Título	
	Reseña	

Tabla 20. Funcionalidades Usuarios: Publicar reseña

Funcionalidad	Parámetros	Actividad
Actualizar una reseña hecha con anterioridad, visualizando la reseña actual y teniendo la posibilidad de actualizar los parámetros que el usuario desee	Puntuación	El usuario podrá actualizar su calificación al parqueadero, titular y escribir una nueva reseña por su experiencia de uso en el parqueadero
	Título	
	Reseña	

Tabla 21. Funcionalidades Usuarios: Editar reseña

Funcionalidad	Opción	Actividad
Visualizar las reseñas que ha recibido un parqueadero de todos los usuarios	Listado de reseña	El usuario podrá visualizar todas las reseñas que se han hecho sobre ese parqueadero para recibir un feedback de lo que puede esperar al hacer uso de este espacio

Tabla 22. Funcionalidades Usuarios: Ver reseñas

- **Diagramas de caso de uso**

Los diagramas de caso de uso para el desarrollador y usuarios dan una visualización clara acerca de la relación e interacción con las funcionalidades disponibles en la aplicación móvil.

Diagrama de caso de uso para desarrollador

El desarrollador puede ejecutar las acciones siguientes:



Figura 16. Diagrama de caso de uso para desarrollador

Diagrama de navegación para usuarios

Los usuarios pueden ejecutar las acciones siguientes:

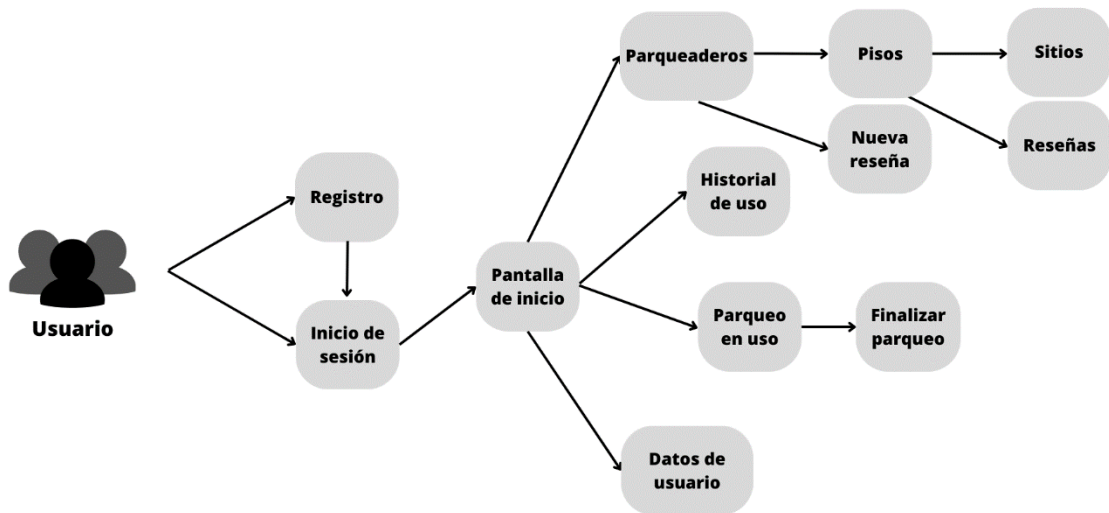


Figura 17. Diagrama de caso de uso para usuarios

- **Prototipo de diseño de la aplicación móvil**

Para el producto de esta investigación se creó un prototipo usando Figma, con el fin de tener una previsualización de lo que se quería conseguir en cuanto a pantallas, funcionalidades e interacción, dado a la flexibilidad de la metodología Mobile-D, hay pantallas que se agregaron o modificaron según necesidades y cambios que surgieron durante el desarrollo, por tanto, hay más pantallas en la aplicación final que en el maquetado Figma.

Componentes

Para el maquetado de la aplicación móvil en general en primer lugar se realizó un maquetado de los componentes a reutilizar, la elección de la paleta de colores y estilos de todos los elementos.

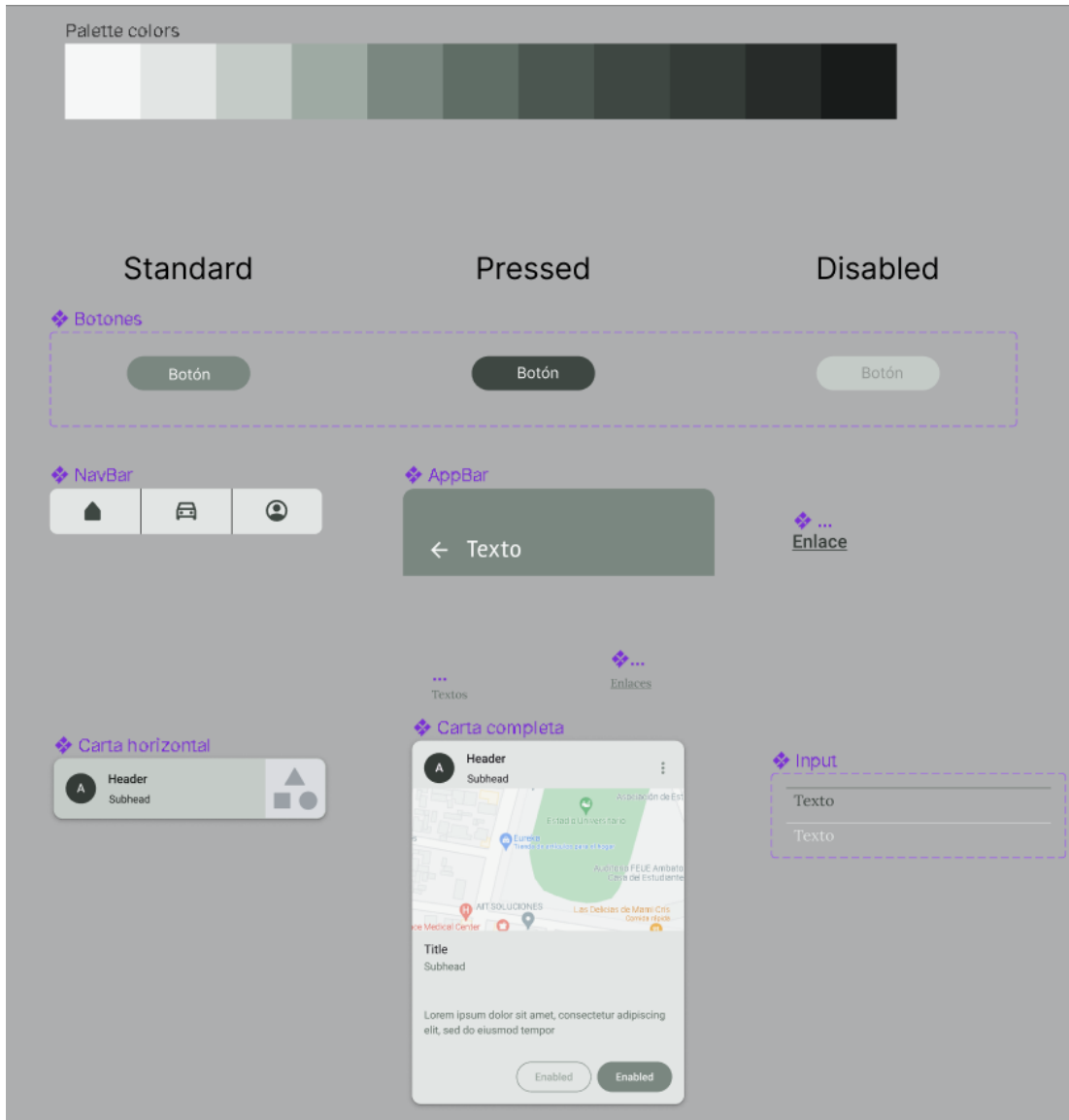


Figura 18. Componentes de diseño

Registro

El registro de usuario tiene campos de texto de tipo específico dentro de un formulario.



9:30

Sign Up

Name:

Last name:

Id:

E-mail:

Password:

Sign Up

Figura 19. Maquetado: Registro

Inicio de sesión

El inicio de sesión de un usuario ya registrado se hace mediante campos de texto.

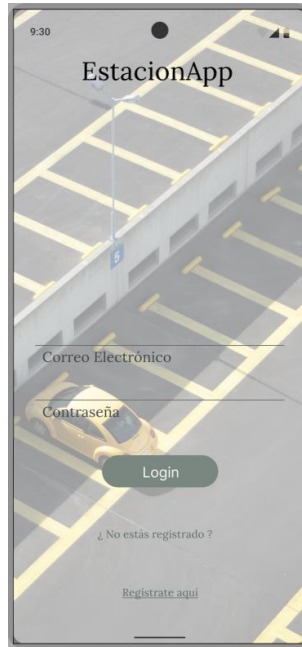


Figura 20. Maquetado: Inicio de sesión

Pantalla de inicio

Se visualiza el diseño que tendrá el inicio de la aplicación, con un mapa de referencia con marcadores de los parqueaderos, el acceso a parqueaderos registrados, un menú de navegación con acceso a los datos del usuario conectado y un parqueo en curso, a más del historial de uso.



Figura 21. Maquetado: Pantalla de inicio

Datos de usuario

Se visualizan los datos del usuario conectado como nombres, apellidos, cédula.

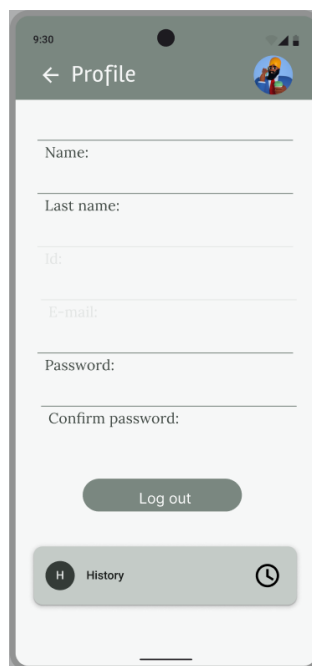


Figura 22. Maquetado: Datos de usuario

Historial de uso

Se visualiza el historial de uso de parqueaderos del usuario conectado

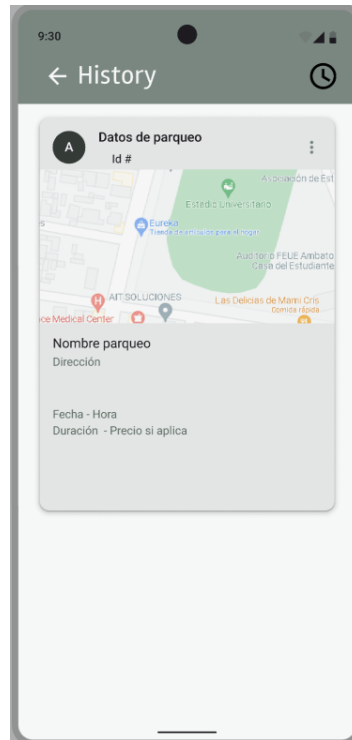


Figura 23. Maquetado: Historial de uso

Visualización de parqueadero

Se visualiza el prospecto de parqueadero para mostrar al usuario

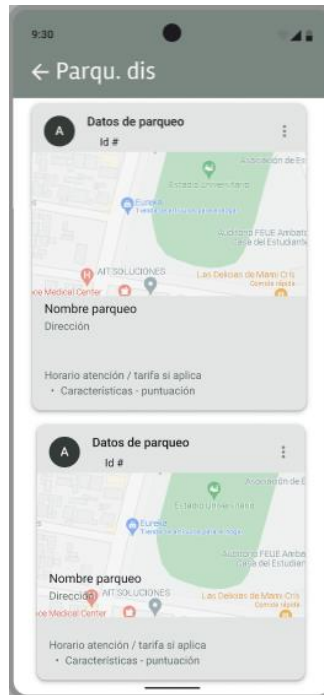


Figura 24. Maquetado: Parqueaderos

Esquema de piso

Se visualiza el diseño previo de los sitios disponibles en el piso de un parqueadero y posterior elección de uso.



Figura 25. Maquetado: Esquema y elección de sitio

Parqueo en uso y finalización de uso

Pantalla de visualización de parqueo en curso y finalización del parqueo.

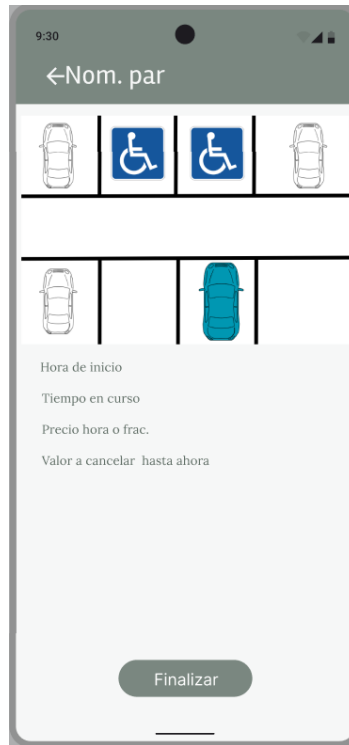


Figura 26. Maquetado: Finalización de parqueo

3.3.3. Fase III: Producción

En esta fase se desarrolló el producto acorde a las funcionalidades establecidas en la anterior fase, la Fase II.

- **Backend**

El servidor que maneja las peticiones HTTP API REST se desarrolló en NodeJS, desarrollado bajo buenas prácticas de programación como la división de código y estructuración en distintos archivos para evitar la acumulación de funcionalidades en pocos archivos, separando modelos, rutas y controladores.

- **Modelos**

Los modelos son las entidades manejadas en la aplicación móvil, equivalentes a los objetos en programación orientada a objetos.

Lista de modelos:

- user.js
- site.js
- review.js
- rent.js
- parking.js
- floor.js
- admin_parking.js

○ **Controladores**

Los controladores son los archivos que se encargan de alojar el código que maneja la información enviada por el Frontend y sabe qué hacer con esta información según las necesidades de la aplicación.

Lista de controladores:

- AdminParkingController.js
- FloorController.js
- ParkingController.js
- RentController.js
- ReviewController.js
- SiteController.js
- UserController.js

○ **Rutas**

Las rutas son las que se encargan de la conexión entre los controladores y el Frontend generando los endpoints correspondientes para cada petición y tipo de petición, ya sea de tipo POST, PUT, GET, etc.

Listado de rutas:

- admin_parking_routes.js
- floor_routes.js
- parking_routes.js
- rent_routes.js
- review_routes.js
- site_routes.js
- user_routes.js

Archivos necesarios de configuración inicial

Para que el backend se ejecute de manera exitosa y logre la convivencia entre modelos, controladores y rutas, también helpers como JasonWebToken para la seguridad de las peticiones y la aplicación misma, puertos que se usan, referencias a la conexión de base de datos MongoDB.

Como ejemplo de esta configuración, en la Figura 27 se puede apreciar el bloque de conexión con la base de datos, cabe mencionar que la url y puertos de conexión fueron manejados bajo buenas prácticas de programación, que son las variables de entorno, a las cuales se hacen referencia en las sentencias process.env.



```
1 // Database Connection & socket conn
2 const connectDB = async () => {
3   try {
4     await mongoose.connect(process.env.DB_URL, { useNewUrlParser: true, useUnifiedTopology: true });
5     console.log('Conexión exitosa a la base de datos');
6   }
7   server.listen(process.env.PORT, (err) => {
8     if (err) throw new Error(err);
9     console.log('Servidor ejecutándose en el puerto:', process.env.PORT);
10  });
11  } catch (error) {
12    console.error('Error al conectar a la base de datos:', error);
13  }
14 };
15
16 connectDB();
```

Figura 27. Bloque de conexión en archivo index.js

En la Figura 28, el controlador que se encarga de finalizar la renta de un sitio en uso con los parámetros a actualizarse de un documento existente, creado al iniciar un parqueo seleccionado.

```
1  const finishRent = async function(req, res = response) {
2    try {
3      const { _id } = req.body;
4      if (_id) {
5        const currentTime = momentZone().tz('America/Guayaquil').format('HH:mm');
6        const finishedRent = await Rent.findOne({ _id });
7        if(!finishedRent){
8          res.status(400).json({ msg: 'No se encontró el documento :/' });
9        }
10       // Obteniendo las horas de time_start y time_finish
11       const start = moment(finishedRent.time_start, 'HH:mm');
12       const finish = moment(currentTime, 'HH:mm');
13       // Calculando la diferencia en minutos
14       const diffMinutes = finish.diff(start, 'minutes');
15       // Obteniendo la cantidad de horas y minutos en el intervalo de tiempo
16       const hoursInterval = Math.floor(diffMinutes / 60);
17       const minutesInterval = diffMinutes % 60;
18       // Formateando el intervalo de tiempo como HH:mm
19       const total_time_calculate = moment({ hours: hoursInterval, minutes: minutesInterval }).format('HH:mm');
20       const parking = await Parking.findOne({ _id: finishedRent.parking });
21       const priceperhour = parking.priceperhour;
22       // Obtener las horas y minutos en total_time_calculate
23       const [hours, minutes] = total_time_calculate.split(':').map(Number);
24       // Calcular el costo total
25       let total_cost = priceperhour * hours;
26       if (minutes > 0) {
27         total_cost += priceperhour;
28       }
29       //cambiar estado de sitio a disponible
30       await Site.findByIdAndUpdate({_id: finishedRent.site},{
31         availability: true
32       });
33       //cambiar estado de usuario en uso
34       await User.findByIdAndUpdate({_id: finishedRent.user},{
35         inUse: false
36       });
37       const rent_final = await Rent.findByIdAndUpdate(_id, {
38         time_finish: currentTime,
39         total_time: total_time_calculate,
40         final_price: total_cost,
41         finished: true
42       });
43       res.json({ rent_final });
44     } else {
45       res.status(400).json({ msg: 'No hay id :/' });
46     }
47   } catch (error) {
48     console.log(error);
49     res.status(500).json({ msg: 'Algo salió mal :/' });
50   }
51 }
```

Figura 28. Controlador finalizar renta

En la Figura 29, las rutas de usuarios con el registro, inicio de sesión y renovación de token para que se mantenga conectado aún si se ha caducado el token, cabe mencionar

que las rutas de registro e inicio de sesión llevan middlewares para realizar validaciones.



```
1 //api request
2 api.post('/user_register', [
3   check('names', 'El nombre es obligatorio').not().isEmpty(),
4   check('email', 'El correo es obligatorio').isEmail(),
5   check('password', 'La contraseña es obligatoria').not().isEmpty(),
6   validateFields
7 ],userRegister);
8
9 api.post('/login',[
10  check('email', 'El correo es obligatorio').isEmail(),
11  check('password', 'La contraseña es obligatoria').not().isEmpty(),
12  validateFields
13 ], userLogin);
14
15 api.get('/renew', validateJWT ,renewToken);
```

Figura 29. Rutas de usuario

Mientras que, en otras metodologías ágiles, el uso de storycards para detallar actividades realizadas es indispensable, Mobile-D se enfoca en utilizar modelos de casos de uso para describir los requisitos y la funcionalidad de la aplicación móvil, aquí se realizaron en forma de tablas para poder sintetizar el proceso.

En la Tabla 23 se constata el caso de uso de un registro de usuario en la aplicación móvil.

Caso de uso	Registro de usuario
Actores:	Usuario
Objetivo:	Permitir que un usuario cree una cuenta en la aplicación
Paso	Acción
1	El usuario inicia la aplicación en su dispositivo
2	La aplicación muestra la página de inicio de sesión
3	El usuario presiona el botón “Regístrate”
4	La aplicación muestra la pantalla de registro
5	El usuario ingresa la información requerida en cada campo de texto existente
6	El usuario presiona el botón “Registrarse”
7	La aplicación verifica que los datos proporcionados sean válidos para cada caso y que no exista una cuenta con estas credenciales
8	Si los datos son válidos y no hay duplicidad, la aplicación registra esta nueva cuenta e inicia sesión de forma automática y la aplicación muestra la pantalla principal
9	Si los datos no son válidos o hay duplicidad, la aplicación mostrará un cuadro de diálogo y solicita al usuario a corregir los problemas existentes

Tabla 23. Caso de uso: Registro

En la Tabla 24 se detalla el caso de uso de un inicio de sesión en la aplicación móvil.

Caso de uso	Inicio de sesión
Actores:	Usuario
Objetivo:	Permitir al usuario acceder a la aplicación mediante la autenticación de sus credenciales registradas
Paso	Acción
1	El usuario inicia la aplicación en su dispositivo
2	La aplicación muestra la página de inicio de sesión
3	El usuario ingresa su correo electrónico y contraseña en los campos correspondientes
4	El usuario presiona el botón “Iniciar sesión”
5	La aplicación verifica las credenciales proporcionadas
6	Si las credenciales son válidas, el usuario es autenticado y redirigido a la pantalla principal
7	Si las credenciales no son válidas, la aplicación mostrará un cuadro de diálogo mostrándole al usuario los problemas existentes

Tabla 24. Caso de uso: Inicio de sesión

En la Tabla 25 se detalla el caso de uso de un usuario rentando un sitio en un parqueadero registrado.

Caso de uso	Renta de un sitio
Actores:	Usuario
Objetivo:	Permitir al usuario rentar un sitio de parqueo de un parqueadero registrado
Paso	Acción
1	El usuario inicia la aplicación en su dispositivo
2	La aplicación verifica si el token es válido
3	Si el token es válido se muestra la pantalla de inicio
4	Si el token es inválido se muestra la pantalla de inicio de sesión
5	En la pantalla de inicio el usuario selecciona la tarjeta “Listado de parqueaderos”
6	La aplicación muestra la pantalla Parqueadero disponibles con el listado de parqueaderos registrados
7	El usuario elige el parqueadero de su preferencia
8	Si el usuario no conoce el parqueadero, selecciona el botón “Como llegar” en el mapa ubicado en la tarjeta del parqueadero y este lo redirige a Google maps con la ruta lista
9	La aplicación le muestra al usuario el listado de pisos que dispone el parqueadero
10	El usuario selecciona el piso de su preferencia
11	La aplicación le muestra el esquema del piso con los sitios disponibles y sitios ya ocupados
12	El usuario selecciona un sitio
13	La aplicación le muestra un dialogo de confirmación
14	Si el usuario confirma la acción será dirigido a la pantalla de “parqueo en curso” que también puede ser accedida desde la pantalla de inicio

Tabla 25. Caso de uso: Renta de un sitio de parqueo

En la Tabla 26 se detalla el caso de uso de un usuario finalizando la renta de un parqueo en curso.

Caso de uso	Finalización de una renta en curso
Actores:	Usuario
Objetivo:	Permitir que el usuario finalice la renta de un parqueo en curso existente
Paso	Acción
1	El usuario ingresa a la pantalla de inicio de la aplicación
2	El usuario selecciona el botón “P” de parqueo en curso
3	Si no tiene un parqueo en curso, la aplicación en este apartado le mostrará una pantalla con información de que no tiene un parqueo en curso
4	La aplicación muestra los datos del parqueo en curso si es que lo tiene
5	El usuario toca el botón “Finalizar parqueo”
6	La aplicación muestra un cuadro de diálogo, preguntando al usuario si desea terminar el parqueo en curso
7	El usuario confirma la acción tocando el botón “Finalizar”
8	La aplicación le muestra al usuario su historial de uso con el parqueo finalizado más reciente como primero de la lista con detalles de fecha, sitio utilizado, precio, tiempo total.

Tabla 26. Caso de uso: Finalización de renta en curso

En la Tabla 27 se detalla el caso de uso de la creación de una reseña de un parqueadero por parte de un usuario.

Caso de uso	Reseñar un parqueadero
Actores:	Usuario
Objetivo:	Permitir al usuario reseñar un parqueadero para dar su opinión sobre el servicio recibido
Paso	Acción
1	El usuario ingresa a la pantalla de inicio de la aplicación
2	En la pantalla de inicio el usuario selecciona la tarjeta “Listado de parqueaderos”
3	La aplicación muestra la pantalla Parqueadero disponibles con el listado de parqueaderos registrados, en la parte inferior de cada tarjeta está la calificación general del parqueadero
4	El usuario toca el botón “Calificación”
5	La aplicación muestra la pantalla de “Reseña”
6	El usuario asigna una calificación entre 1 y 5, un título y una reseña, presiona en botón “Enviar”
7	Si los apartados requeridos fueron llenados de forma correcta la aplicación muestra un cuadro de diálogo exitoso
8	Si los apartados no fueron llenados de forma correcta se muestra un diálogo con los errores de llenado de estos campos

Tabla 27. Caso de uso: Reseña a parqueadero

Para el caso de uso de una actualización de una reseña hecha anteriormente por parte del usuario a un parqueadero, será el mismo caso de uso de la Tabla 27, con la ligera diferencia que en la pantalla “Reseña” encontrará los campos enviados con anterioridad y podrá cambiar el campo que el usuario desee y volver a enviar los datos.

En la Tabla 28 se detalla el caso de uso de un usuario que quiera visualizar todas las reseñas hechas a un parqueadero.

Caso de uso	Visualización de reseñas de un parqueadero
Actores:	Usuario
Objetivo:	Permitir al usuario visualizar todas las reseñas hechas por usuarios a un parqueadero.
Paso	Acción
1	El usuario ingresa a la pantalla de inicio de la aplicación
2	En la pantalla de inicio el usuario selecciona la tarjeta “Listado de parqueaderos”
3	La aplicación muestra la pantalla Parqueadero disponibles con el listado de parqueaderos registrados
4	El usuario elige el parqueadero del que quiere ver las reseñas
5	La aplicación le muestra al usuario el listado de pisos que dispone el parqueadero
6	La aplicación le muestra al usuario el listado de pisos que dispone el parqueadero, al final se visualiza el botón “Reviews”
7	El usuario toca el botón “Reviews”
8	La aplicación muestra un listado de reseñas que ha recibido de los usuarios

Tabla 28. Caso de uso: Visualizar reseñas de un parqueadero

3.3.4. Fase IV: Estabilización

Esta fase tiene como objetivo garantizar el funcionamiento de la aplicación libre de errores críticos, estable y cumpla con los requisitos establecidos en la Fase de Inicialización.

En esta fase se hicieron los siguientes ajustes:

- Ajustó tamaño de textos y botones
- Ajustó tamaño de espacios y tarjetas
- Se limitó funcionalidades de Google maps que cargaban la aplicación de funcionalidades no acordes al proyecto

- Se implementaron cuadros de diálogo en todas las acciones
- Se implementaron precargadores para refrescar información
- Se quitaron rutas innecesarias de la navegación
- Se ajustaron colores para mejor UX

3.3.5. Fase V: Pruebas

Fase que tiene como objetivo comprobar que las funcionalidades de la aplicación se comporten como se espera de acuerdo con el estudio previo al desarrollo en la fase de inicialización, con la finalidad de corregir y mejorar su rendimiento.

Pruebas de funcionalidades

- **Registro de usuario**

Para el registro de usuarios se necesitan llenar los campos, como se ve en la Figura 30, teniendo en los respectivos hintText una guía de relleno de los campos para el usuario.

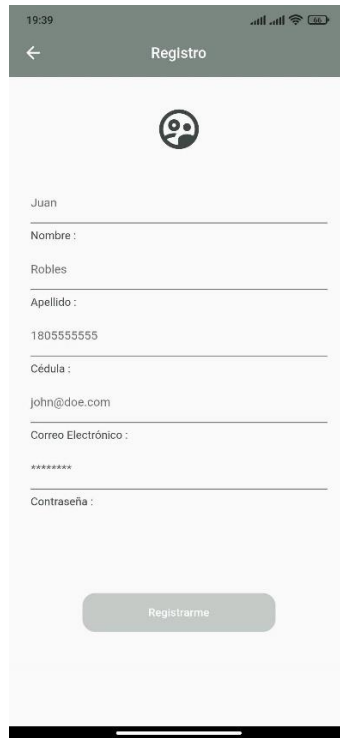


Figura 30. Aplicación: Registro de usuario

- **Inicio de sesión**

Para el inicio de sesión de usuarios se necesita proporcionar las credenciales registradas, requeridas como se visualiza en la Figura 31.

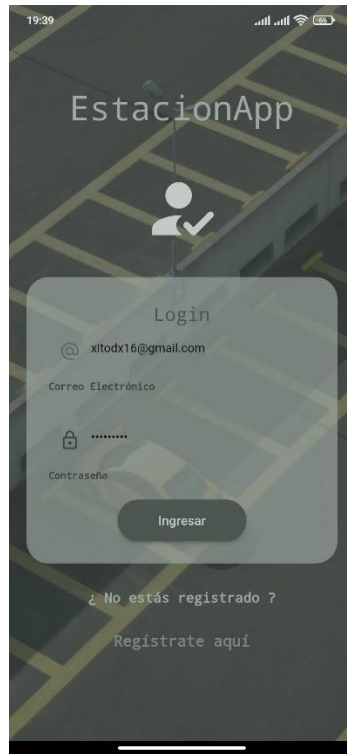


Figura 31. Aplicación: Inicio de sesión de usuario

- **Pantalla de inicio**

En la pantalla de inicio que se aprecia en la Figura 32, se muestran las funcionalidades principales para el usuario, como un Mapa principal con marcadores de los parqueaderos disponibles y su respectiva ayuda para llegar al parqueadero, el acceso al listado de parqueaderos disponibles, el acceso a su historial de uso, el acceso a un parqueadero en curso en el botón “P”, y a la derecha el botón que alberga la navegación hacia la pantalla que muestra los datos del usuario conectado.

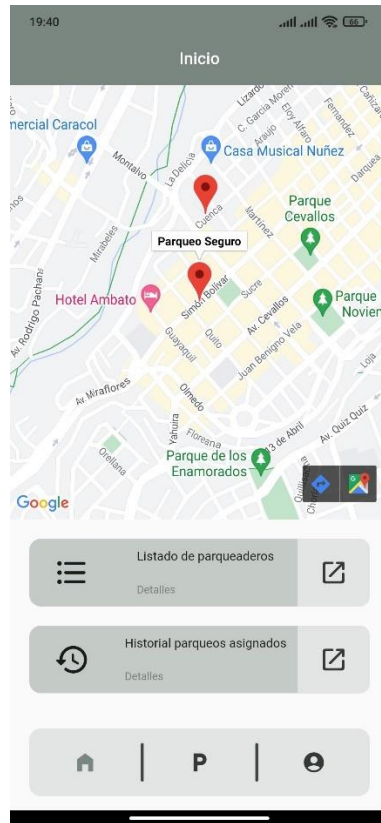


Figura 32. Aplicación: Pantalla de inicio

- **Perfil de usuario**

En la Figura 33, se aprecia la pantalla que muestra el perfil de usuario, sus datos ingresados al momento de registrarse, dando también la opción de cerrar sesión si el usuario lo requiere.

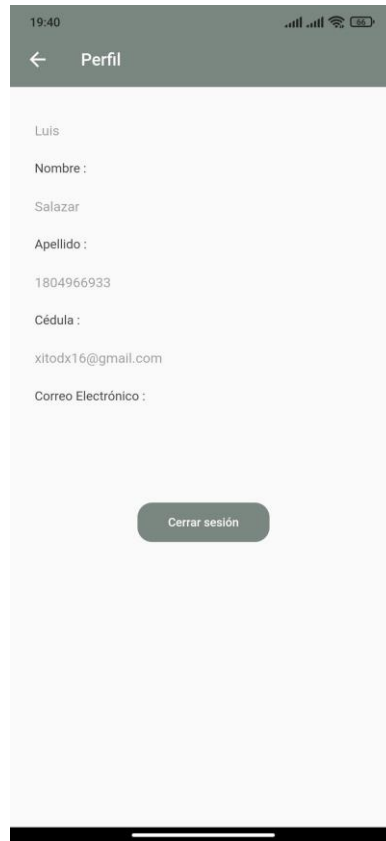


Figura 33. Aplicación: Perfil de usuario

- **Parqueaderos disponibles**

En la Figura 34, se aprecia la pantalla que muestra los parqueaderos disponibles con los atributos que se pueden apreciar.

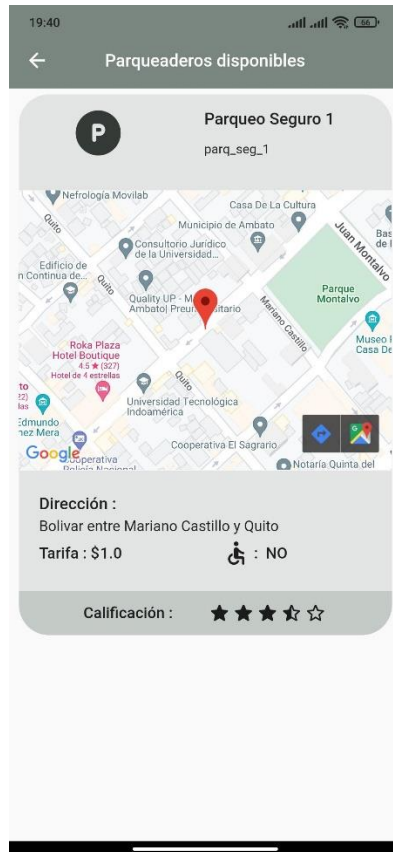


Figura 34. Aplicación: Lista de parqueaderos

- **Pisos disponibles**

En la Figura 35 se muestran los pisos que tiene disponible un parqueadero, a más del botón que llevará al usuario a visualizar las reseñas que ha recibido el parqueadero.



Figura 35. Aplicación: Lista de pisos

- **Reseña**

En la Figura 36 se visualiza la pantalla para enviar una reseña de un parqueadero, la puntuación se hace equivalente a número de estrellas a la experiencia de uso personal.

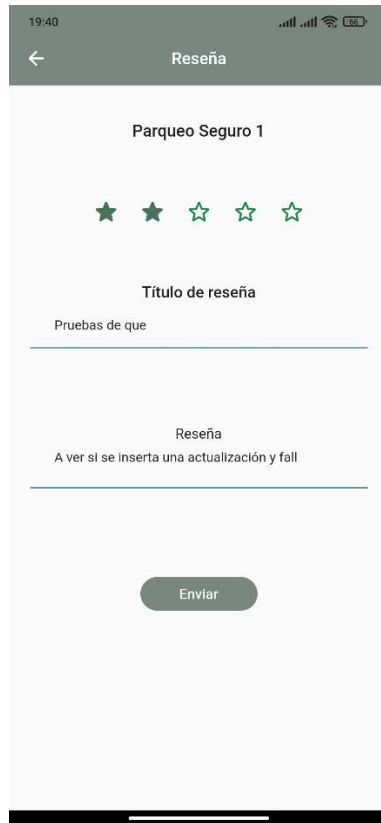


Figura 36. Aplicación: Reseña

- **Lista de reseñas**

En la Figura 37, se aprecia la pantalla que enlista todas las reseñas que el parqueadero ha recibido de los usuarios.

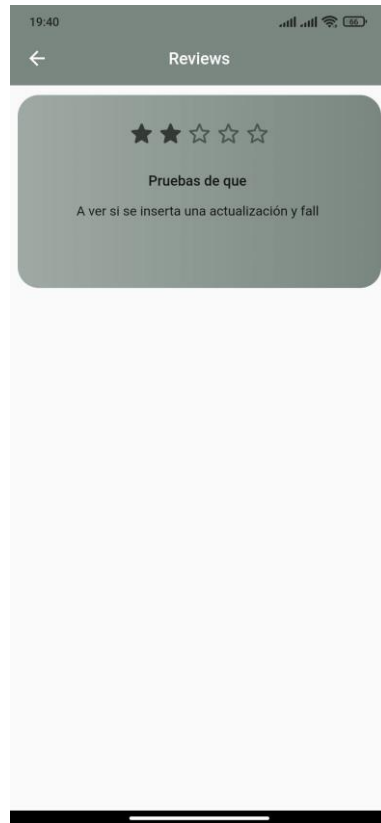


Figura 37. Aplicación: Listado de reseñas

- **Esquema de piso**

En la Figura 38, se aprecia el esquema de un piso con la disponibilidad de los sitios y demás elementos que conforman cada piso.

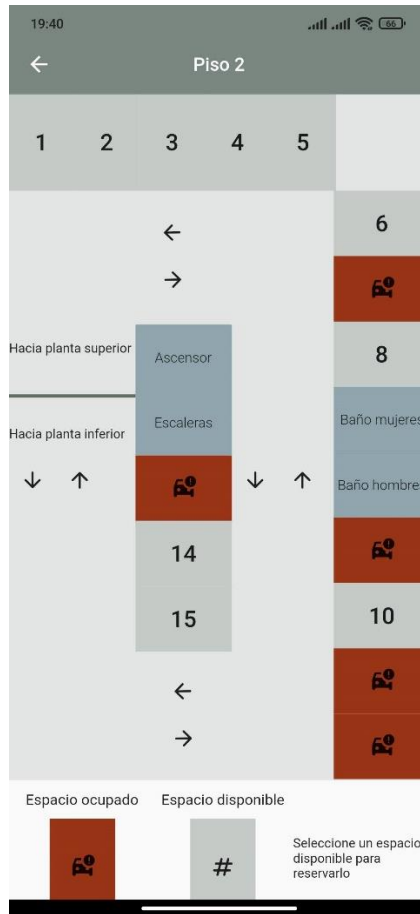


Figura 38. Aplicación: Esquema de piso

- **Renta de un sitio**

En la Figura 39 se aprecia el cuadro de diálogo al reservar un espacio disponible, cabe destacar que los espacios en rojo no están disponibles y no muestran ninguna interacción con el usuario mientras su estado no cambie a disponible.



Figura 39. Aplicación: Renta de sitio

- **Parqueo en curso**

En la Figura 40 se aprecia la pantalla de parqueo en curso cuando efectivamente existe uno.



Figura 40. Aplicación: Parqueo en curso

- **Finalizar un parqueo en curso**

En la Figura 41, se parecía como el historial de uso del usuario refleja que tiene un parqueo en curso pendiente de finalización.



Figura 41. Aplicación: Historial con finalización pendiente

En la Figura 42, se aprecia como se pasa a finalizar un parqueo en curso, insertando los datos faltantes para calcular el tiempo total de uso y el precio final.



Figura 42. Aplicación: Finalizar parqueo

- **Historial de uso**

En la Figura 43 se aprecia el historial de usuario con todos los parqueos finalizados y cada uno con los campos completos.

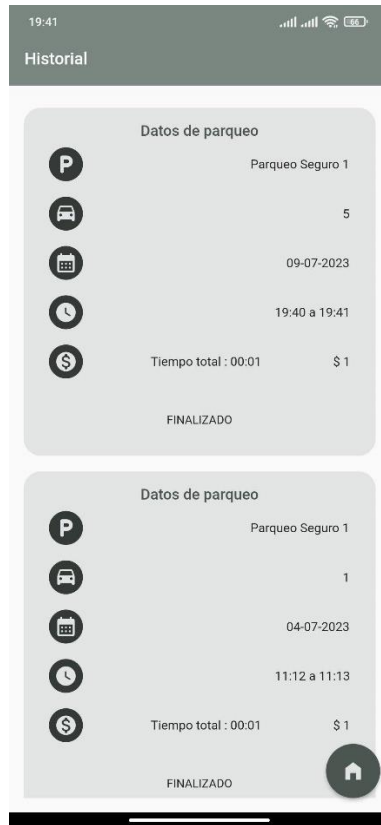


Figura 43. Aplicación: Historial de uso

Pruebas de aceptación

Una vez realizadas las pruebas sobre las funcionalidades de la aplicación móvil comprendidas entre la Figura 30 y la Figura 43, se obtuvo como pruebas de aceptación las siguientes tablas.

Prueba de aceptación	Número: 1
Nombre: Registro para usuarios	
Descripción: Permite registrar un nuevo usuario	
Condiciones de ejecución e interfaz: Se necesitan los datos del nuevo usuario y una contraseña	
Resultado esperado: Si los campos son llenados de forma correcta, el nuevo usuario es registrado y autenticado de forma automática	
Resultado de la prueba: Prueba satisfactoria	

Tabla 29. Prueba de aceptación: 1

Prueba de aceptación	Número: 2
Nombre: Inicio de sesión de usuario	
Descripción: Permite a un usuario existente ingresar a la aplicación	
Condiciones de ejecución e interfaz: Se necesita el correo electrónico y contraseña	
Resultado esperado: Si los campos son llenados de la misma forma que en el registro, el usuario es autenticado.	
Resultado de la prueba: Prueba satisfactoria	

Tabla 30. Prueba de aceptación: 2

Prueba de aceptación	Número: 3
Nombre: Pantalla de inicio	
Descripción: Permite visualizar las principales funciones de la aplicación	
Condiciones de ejecución e interfaz: El usuario debe estar autenticado	
Resultado esperado: El usuario puede ver un mapa principal con marcadores de parqueaderos disponibles, dos tarjetas, una de listado de parqueaderos y otra de historial de uso, un menú inferior con el acceso a la pantalla de inicio, parqueo en curso y perfil de usuario	
Resultado de la prueba: Prueba satisfactoria	

Tabla 31. Prueba de aceptación: 3

Prueba de aceptación	Número: 4
Nombre: Listado de parqueaderos	
Descripción: Permite interactuar con los parqueaderos disponibles	
Condiciones de ejecución e interfaz: Presionar la tarjeta “Listado de parqueaderos”	
Resultado esperado: Si el usuario presiona la tarjeta “Listado de parqueaderos”, podrán visualizar todos los parqueaderos disponibles	
Resultado de la prueba: Prueba satisfactoria	

Tabla 32. Prueba de aceptación: 4

Prueba de aceptación	Número: 5
Nombre: Pisos disponibles	
Descripción: Permite interactuar con los pisos disponibles de un parqueadero	
Condiciones de ejecución e interfaz: Presionar la tarjeta de un parqueadero	
Resultado esperado: Si el usuario presiona una tarjeta de cualquier parqueadero disponible, podrá visualizar todos los pisos que este parqueadero dispone	
Resultado de la prueba: Prueba satisfactoria	

Tabla 33. Prueba de aceptación: 5

Prueba de aceptación	Número: 6
Nombre: Esquema de piso	
Descripción: Permite interactuar con los sitios que dispone un piso	
Condiciones de ejecución e interfaz: Presionar la tarjeta de un piso	
Resultado esperado: Si el usuario presiona una tarjeta de cualquier piso disponible, podrá visualizar todos los sitios que este piso dispone	
Resultado de la prueba: Prueba satisfactoria	

Tabla 34. Prueba de aceptación: 6

Prueba de aceptación	Número: 7
Nombre: Confirmar renta de un sitio	
Descripción: Permite rentar un sitio disponible	
Condiciones de ejecución e interfaz: Presionar un sitio de un piso	
Resultado esperado: Si el usuario presiona un sitio disponible, podrá confirmar el parqueo en este sitio	
Resultado de la prueba: Prueba satisfactoria	

Tabla 35. Prueba de aceptación: 7

Prueba de aceptación	Número: 8
Nombre: Parqueo en curso	
Descripción: Permite visualizar un parqueo en curso con sus datos iniciales	
Condiciones de ejecución e interfaz: Confirmar la renta de un sitio	
Resultado esperado: Si el usuario presiona una tarjeta de cualquier parqueadero disponible, podrá visualizar todos los pisos que este parqueadero dispone	
Resultado de la prueba: Prueba satisfactoria	

Tabla 36. Prueba de aceptación: 8

Prueba de aceptación	Número: 9
Nombre: Finalizar parqueo en curso	
Descripción: Permite finalizar un parqueo en curso	
Condiciones de ejecución e interfaz: Tener un parqueo en curso, ubicarse en pantalla parqueo en curso, accesible en la pantalla inicio	
Resultado esperado: Si el usuario presiona el botón Finalizar, podrá ver los detalles de la renta del sitio seleccionado, tiempo total, precio, el sitio seleccionado volverá a estar en estado “Disponible” y el usuario podrá volver a rentar un sitio.	
Resultado de la prueba: Prueba satisfactoria	

Tabla 37. Prueba de aceptación: 9

Prueba de aceptación	Número: 10
Nombre: Visualizar historial de uso parqueos	
Descripción: Permite visualizar el listado de rentas de sitios de parqueo	
Condiciones de ejecución e interfaz: Presionar la tarjeta “Historial parqueos asignados”	
Resultado esperado: Si el usuario presiona la tarjeta, podrá visualizar todas las rentas de sitios que ha hecho en cualquier parqueadero	
Resultado de la prueba: Prueba satisfactoria	

Tabla 38. Prueba de aceptación: 10

Prueba de aceptación	Número: 11
Nombre: Reseñar	
Descripción: Permite redactar una reseña a un parqueadero	
Condiciones de ejecución e interfaz: Presionar el botón “Calificación”, ubicado en la parte inferior de cada carta de “Listado de parqueaderos”	
Resultado esperado: Si el usuario presiona el botón “Calificación” de una tarjeta de parqueadero, podrá calificar y escribir una reseña al parqueadero seleccionado	
Resultado de la prueba: Prueba satisfactoria	

Tabla 39. Prueba de aceptación: 11

Prueba de aceptación	Número: 12
Nombre: Listado de reseñas	
Descripción: Permite visualizar las reseñas que ha recibido ese parqueadero	
Condiciones de ejecución e interfaz: Presionar botón “Reviews”	
Resultado esperado: Si el usuario presiona una tarjeta de cualquier parqueadero disponible, podrá visualizar al final del listado de pisos disponibles, el botón “Reviews”	
Resultado de la prueba: Prueba satisfactoria	

Tabla 40. Prueba de aceptación: 12

CAPÍTULO IV.- CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

La investigación realizada para analizar los problemas que enfrentan las personas al buscar sitios de asignación en parqueaderos públicos en la ciudad de Ambato reveló desafíos significativos, como la falta de información en tiempo real sobre la disponibilidad de espacios y la dificultad para realizar reservas de forma rápida y conveniente.

La aplicación móvil multiplataforma desarrollada utilizando Flutter llegó a ser una solución viable para la asignación de espacios de parqueo en la ciudad de Ambato. Al proporcionar una interfaz intuitiva y fácil de usar, la aplicación en funcionamiento está enfocada en mejorar la experiencia de los usuarios al encontrar y reservar lugares de estacionamiento en parqueaderos públicos.

El desarrollo de la aplicación móvil resultó en una solución eficiente y rentable en la renta de sitios de parqueo, al ser un desarrollo multiplataforma, ha permitido la creación de una aplicación que funciona tanto en dispositivos iOS como Android, reduciendo así los costos y el tiempo de desarrollo. Además, la interfaz de usuario fluida y atractiva proporcionada por Flutter.

4.2. Recomendaciones

Continuar mejorando la aplicación móvil en función de los comentarios y sugerencias de los usuarios una vez puesta en producción. Es importante recopilar retroalimentación regularmente para identificar áreas de mejora y realizar actualizaciones periódicas que brinden nuevas características y funcionalidades.

Establecer colaboraciones con entidades gubernamentales y empresas de estacionamiento para obtener acceso a datos en tiempo real sobre la disponibilidad de

espacios de estacionamiento, esto permitirá que la aplicación proporcione información precisa, actualizada y automatizada a los usuarios.

Realizar campañas de marketing y promoción para adoptar el uso de la aplicación móvil una vez sea implementada entre los ciudadanos de Ambato. La difusión de los beneficios de la aplicación, como la facilidad de uso, la conveniencia y la capacidad de ahorrar tiempo, ayudará a fomentar su uso generalizado y garantizará su éxito a largo plazo.

BIBLIOGRAFÍA

- [1] «Total_in-use-All-Vehicles.pdf». Accedido: 15 de noviembre de 2022. [En línea]. Disponible en: https://www.oica.net/wp-content/uploads//Total_in-use-All-Vehicles.pdf
- [2] www.gruponw.com, «Gestión de parqueaderos, porqué reduce riesgos y optimiza tiempos». <https://www.sitca.co/blog/nwarticle/74/TODAS/gestion-de-parqueaderos-porque-reduce-riesgos-y-optimiza-tiempos> (accedido 22 de diciembre de 2022).
- [3] C. L. Giraldo Gallego y V. Idarraga Florez, «Diseño y construcción de un sistema de conteo de espacios disponibles en el parqueadero de la Fundación Universitaria Católica Lumen Gentium», may 2020, Accedido: 6 de diciembre de 2022. [En línea]. Disponible en: <http://repository.unicatolica.edu.co/handle/20.500.12237/1808>
- [4] L. A. Cruz Jalón, «Desarrollo de un sistema de asistencia para búsqueda de disponibilidad de estacionamientos en la Universidad Católica de Santiago de Guayaquil UCSG.», mar. 2017, Accedido: 6 de diciembre de 2022. [En línea]. Disponible en: <http://repositorio.ucsg.edu.ec/handle/3317/7653>
- [5] R. D. P. el 28 junio y 2021 a las 11:41 am Permalink, «Aplicación móvil para el Sistema de Estacionamientos Municipales», 27 de agosto de 2019. <http://www.quitoinforma.gob.ec/2019/08/27/aplicacion-movil-para-el-sistema-de-estacionamientos-municipales/> (accedido 1 de febrero de 2023).
- [6] J. A. Páez *et al.*, «Aplicación de UML y SCRUM al desarrollo del software sobre control de acceso», *Inf. Tecnológica*, vol. 32, n.º 5, pp. 57-66, oct. 2021, doi: 10.4067/S0718-07642021000500057.
- [7] C. O. Guachán Cuásquer y D. J. Sigcho Alvarado, «Diseño de una aplicación móvil que permita el registro de vehículos visitantes y consulta de la disponibilidad de espacios de parqueo», bachelorThesis, 2022. Accedido: 24 de noviembre de 2022. [En línea]. Disponible en: <http://dspace.ups.edu.ec/handle/123456789/23374>
- [8] L. Dagne, «Flutter for cross-platform App and SDK development».
- [9] R.- ASALE y RAE, «aparcar | Diccionario de la lengua española», «*Diccionario de la lengua española*» - Edición del Tricentenario. <https://dle.rae.es/aparcar> (accedido 1 de diciembre de 2022).
- [10] «POLITICAS GENERALES PARA LA ASIGNACION DE PARQUEADERO-NOVIEMBRE 10-2011.doc».
- [11] D. A. Escobar, C. A. Moncada, y C. F. Urazán, «Definición de áreas de estacionamiento en una zona urbana. Propuesta metodológica de análisis», p. 19.
- [12] «Ingeniería de Software», *UNEMI*. <https://www.unemi.edu.ec/index.php/carreras-presencial/ingenieria-de-software/> (accedido 1 de diciembre de 2022).
- [13] «¿Qué es el desarrollo de software? | IBM». <https://www.ibm.com/es-es/topics/software-development> (accedido 1 de diciembre de 2022).
- [14] «Aplicaciones móviles: tipos, ventajas e inconvenientes», *Ceupe*. <https://www.ceupe.com/blog/aplicaciones-moviles-tipos-ventajas-e-inconvenientes.html> (accedido 2 de diciembre de 2022).
- [15] «Flutter - Crea hermosas aplicaciones nativas en tiempo récord», 2019. <https://esflutter.dev/> (accedido 1 de febrero de 2023).

- [16] «Metodologías de desarrollo de software: ¿qué son?» <https://www.becas-santander.com/es/blog/metodologias-desarrollo-software.html> (accedido 16 de mayo de 2023).
- [17] «¿Qué es Flutter? - Explicación de la aplicación Flutter - AWS», *Amazon Web Services, Inc.* <https://aws.amazon.com/es/what-is/flutter/> (accedido 8 de mayo de 2023).
- [18] «¿Qué es Flutter y para qué sirve? - Talently», *Talently Blog*, 26 de julio de 2022. <https://talently.tech/blog/que-es-flutter/> (accedido 8 de mayo de 2023).
- [19] «Hot Reload». <https://esflutter.dev/docs/development/tools/hot-reload> (accedido 15 de mayo de 2023).
- [20] *Google Keynote (Google I/O '23)*. Accedido: 15 de mayo de 2023. [En línea Video]. Disponible en: <https://www.youtube.com/watch?v=cNfINi5CNbY>
- [21] «Dart de Google: Una introducción al lenguaje Dart», *IONOS Digital Guide*, 15 de octubre de 2020. <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/lenguaje-de-programacion-dart-de-google/> (accedido 9 de mayo de 2023).
- [22] M. J. M. Cordón, «¿Qué es el lenguaje de programación Dart?», *Blog de Hiberus Tecnología*, 30 de junio de 2021. <https://www.hiberus.com/crecemos-contigo/que-es-el-lenguaje-de-programacion-dart/> (accedido 9 de mayo de 2023).
- [23] «▷ Lenguaje de programación Dart **【también Flutter】** », *Lenguajes de programación*. <https://lenguajesdeprogramacion.net/dart/> (accedido 9 de mayo de 2023).
- [24] «Flutter, SDK para controlar los Sistemas Operativos móviles», *OpenWebinars.net*, 17 de junio de 2022. <https://openwebinars.net/blog/flutter-un-sdk-para-controlar-todos-los-sistemas-operativos-moviles/> (accedido 9 de mayo de 2023).
- [25] «Portafolio». <https://esflutter.dev/showcase> (accedido 9 de mayo de 2023).
- [26] «Metodología RAD: qué es y sus ventajas | Blog | Hosting Plus Perú», *Hosting Plus*, 26 de enero de 2022. <https://www.hostingplus.pe/blog/metodologia-rad-que-es-y-sus-ventajas/> (accedido 22 de mayo de 2023).
- [27] «Metodologías para el desarrollo de aplicaciones móviles», *Syntonize*, 30 de agosto de 2021. <https://www.syntonize.com/metodologias-desarrollo-de-aplicaciones-moviles/> (accedido 16 de mayo de 2023).
- [28] «Metodología XP o Programación Extrema: ¿Qué es y cómo aplicarla?», *Gestor de proyectos online*. <https://www.sinnaps.com/blog-gestion-proyectos/metodologia-xp> (accedido 22 de mayo de 2023).
- [29] «Abel_Trabajo_Bachillerato_2020.pdf». Accedido: 22 de mayo de 2023. [En línea]. Disponible en: https://repositorio.upeu.edu.pe/bitstream/handle/20.500.12840/3906/Abel_Trabajo_Bachillerato_2020.pdf?sequence=1&isAllowed=y
- [30] J. R. Molina Ríos, J. A. Honores Tapia, N. Pedreira-Souto, y H. P. Pardo León, «Estado del arte: metodologías de desarrollo de aplicaciones móviles», *3C Technol. Innov. Apl. Pyme*, vol. 10, n.º 2, pp. 17-45, jun. 2021, doi: 10.17993/3ctecno/2021.v10n2e38.17-45.
- [31] «Qué es Visual Studio Code y qué ventajas ofrece», *OpenWebinars.net*, 22 de julio de 2022. <https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/> (accedido 6 de junio de 2023).
- [32] «¿Qué Es MongoDB?», *MongoDB*. <https://www.mongodb.com/es/what-is-mongodb> (accedido 6 de junio de 2023).

- [33] «Qué es NodeJS y para qué sirve», *OpenWebinars.net*, 4 de septiembre de 2019. <https://openwebinars.net/blog/que-es-nodejs/> (accedido 6 de junio de 2023).
- [34] «Introducción a Express/Node - Aprende desarrollo web | MDN», 9 de marzo de 2023. https://developer.mozilla.org/es/docs/Learn/Server-side/Express_Nodejs/Introduction (accedido 6 de junio de 2023).

ANEXOS

A1. Encuesta Usuarios

Esta encuesta se enfoca en identificar los problemas que enfrentan las personas al momento de parquear su vehículo, sea lo más sincero e imparcial posible, marque la casilla con X.

- ¿Con qué frecuencia utiliza parqueaderos públicos para estacionar su vehículo?

(a) Muy frecuentemente

(b) Frecuentemente

(c) Ocasionalmente

(d) Raramente

(e) Nada frecuente

- ¿Cuánto tiempo promedio tarda en encontrar un lugar para aparcar?

(a) Menos de 5 minutos

(b) Entre 5 a 10 minutos

(c) Entre 10 a 15 minutos

(d) Más de 15 minutos

- ¿Con qué frecuencia tiene dificultades para encontrar un lugar para aparcar?

(a) Muy frecuentemente

(b) Frecuentemente

(c) Ocasionalmente

(d) Raramente

(e) Nada frecuente

- ¿Qué tan fácil o difícil cree que es encontrar un lugar para estacionar su vehículo en parqueaderos públicos en horas o fechas concurridas?

(a) Muy fácil

(b) Fácil

(c) Moderado

(d) Difícil

(e) Muy difícil

- ¿Usted cree que una aplicación móvil enfocada en los parqueaderos de la ciudad de Ambato le podría ayudar a agilizar en encontrar un parqueadero y un espacio disponible para su vehículo?

(a) Muy de acuerdo

(b) De acuerdo

(c) Neutral

(d) Desacuerdo

(e) En total desacuerdo

ENCUESTAS	P1	P2	P3	P4	P5	SUMA
1	5	3	5	4	4	21
2	4	2	4	5	4	19
3	5	3	5	5	5	23
4	5	4	5	5	5	24
5	5	3	5	5	5	23
6	5	3	5	5	5	23
7	5	4	5	5	5	24
8	5	3	5	5	5	23
9	4	3	5	4	5	21
10	5	4	5	4	5	23
11	3	3	5	4	5	20
12	4	4	5	4	5	22
13	5	3	5	5	5	23
14	5	4	5	5	5	24
15	4	3	3	4	4	18
16	4	3	5	4	4	20
17	4	4	5	5	5	23
18	4	4	5	5	5	23
19	5	3	5	5	5	23
20	5	3	5	5	5	23
21	5	4	5	5	4	23
22	4	3	4	4	3	18
23	5	4	4	4	5	22
24	5	3	4	5	5	22
25	4	3	5	5	4	21
26	4	4	5	4	5	22
27	4	3	5	5	5	22
28	4	3	3	5	4	19
29	4	4	5	5	5	23
30	5	3	5	5	5	23
31	4	3	5	5	5	22
32	5	4	5	5	5	24
33	4	3	5	5	4	21
34	4	2	5	5	5	21
35	5	3	5	5	5	23
36	3	3	3	3	2	14
37	4	4	5	5	5	23
38	5	3	5	5	5	23
39	5	4	4	5	5	23
40	2	3	2	5	4	16
41	4	3	4	5	5	21
42	5	3	5	5	5	23
43	5	4	5	5	5	24
44	4	3	5	5	5	22
45	5	4	5	5	5	24
Varianza	0,47	0,30	0,49	0,24	0,39	
Sumatoria de varianza			1,89			
Varianza de la suma de			4,55			

A2. Encuesta personal de control

La presente encuesta tiene como objetivo identificar los problemas que enfrentan las personas al momento de buscar sitios de asignación en parqueaderos, sea lo más sincero e imparcial posible para que este estudio tenga los mejores resultados.

- ¿La tarifa de su parqueadero, en que rango entra la tarifa por hora o fracción?
 - (a) Gratuito
 - (b) Menor a \$1
 - (c) Entre \$1 a \$2
 - (d) Más a \$2

- ¿Utiliza tecnologías como aplicaciones móviles para mejorar la gestión de sus parqueaderos?
 - (a) Si
 - (b) No

- ¿Su parqueadero ha tomado medidas para mejorar la accesibilidad para personas con discapacidad?
 - (a) Si
 - (b) No

- ¿Ha notado un aumento en la demanda de parqueaderos públicos en el último año (2022)?
 - (a) Si
 - (b) No

- ¿Cuál es el tamaño del parqueadero que está a su cargo?
 - (a) Pequeño (hasta 20 espacios)
 - (b) Mediano (20-50 espacios)
 - (c) Grande (más de 50 espacios)

- ¿Qué método utiliza para asignar espacios de estacionamiento a los usuarios?
 - (a) Asignación manual por parte del personal del parqueadero
 - (b) Sistema automatizado de asignación

(c) No se asignan espacios, es de libre elección para los usuarios

• ¿Cree usted que le ayudaría a la gestión de su parqueadero el uso de una aplicación móvil para los usuarios?

(a) Totalmente de acuerdo

(b) De acuerdo

(c) Neutral

(d) Desacuerdo

(e) Totalmente en desacuerdo

ENCUESTAS	PREGUNTA 1	PREGUNTA 2	PREGUNTA 3	PREGUNTA 4	PREGUNTA 5	PREGUNTA 6	PREGUNTA 7	SUMA
1	1	2	2	2	3	3	5	18
2	1	1	2	2	3	3	5	17
3	2	1	1	1	2	1	3	11
4	3	1	1	2	2	3	5	17
5	2	1	1	1	1	1	4	11
Varianza	0,56	0,16	0,24	0,24	0,56	0,96	0,64	
Sumatoria de varianzas				3,36				
Varianza de la suma de l				9,76				

A3. Ficha de observación

Ficha de Observación - Problemas en la Búsqueda de Espacios de Estacionamiento en Parqueaderos

- Ubicación del parqueadero:

Nombre o ubicación del parqueadero:

Fecha y hora de la observación:

- Observaciones del parqueadero:

Tamaño del parqueadero (pequeño, mediano, grande):

.....

Método de asignación de espacios (asignación manual, sistema automatizado, libre elección):

.....

Disponibilidad de espacios (completo, parcialmente ocupado, vacío):

.....

- Observaciones de los usuarios en la búsqueda de espacios de estacionamiento:

Cantidad de usuarios buscando espacios de estacionamiento:

.....

Tiempo promedio que los usuarios dedican en la búsqueda de un espacio de estacionamiento:

.....

Comportamientos observados de los usuarios al buscar un espacio de estacionamiento (por ejemplo, conducir en círculos, esperar a que alguien se vaya, etc.):

.....

Problemas o dificultades observadas que enfrentan los usuarios al buscar espacios de estacionamiento (por ejemplo, falta de espacios disponibles, espacios estrechos o incómodos, falta de señalización clara, etc.):

.....

Otros comentarios u observaciones relevantes:

.....

- Conclusiones:

Principales problemas identificados en la búsqueda de espacios de estacionamiento:

.....

Frecuencia y gravedad de los problemas observados:

.....

Posibles soluciones o mejoras para abordar los problemas identificados:

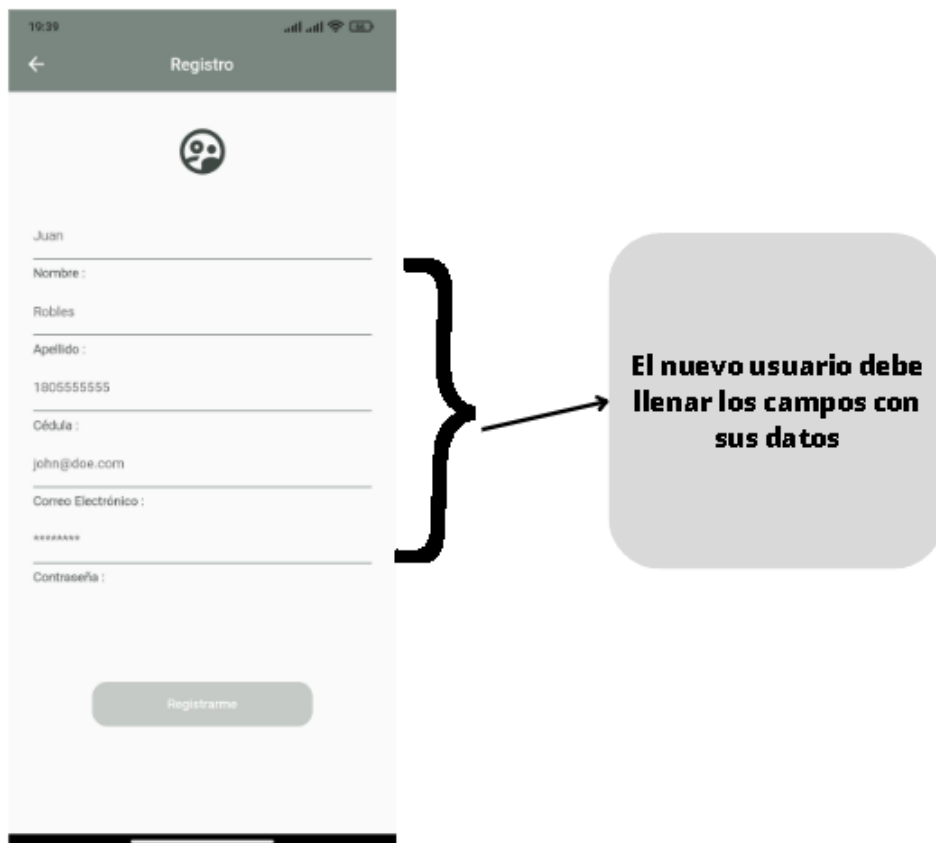
.....

A4. Manual de Usuario

“APLICACIÓN MÓVIL MULTIPLATAFORMA APLICANDO FLUTTER PARA ASIGNACIÓN DE ESPACIOS DE PARQUEO EN LA CIUDAD DE AMBATO”

Este manual de usuario tiene la finalidad de guiar y facilitar al usuario aprovechar las funcionalidades implementadas en la aplicación móvil.

Registro de usuario



Inicio de sesión



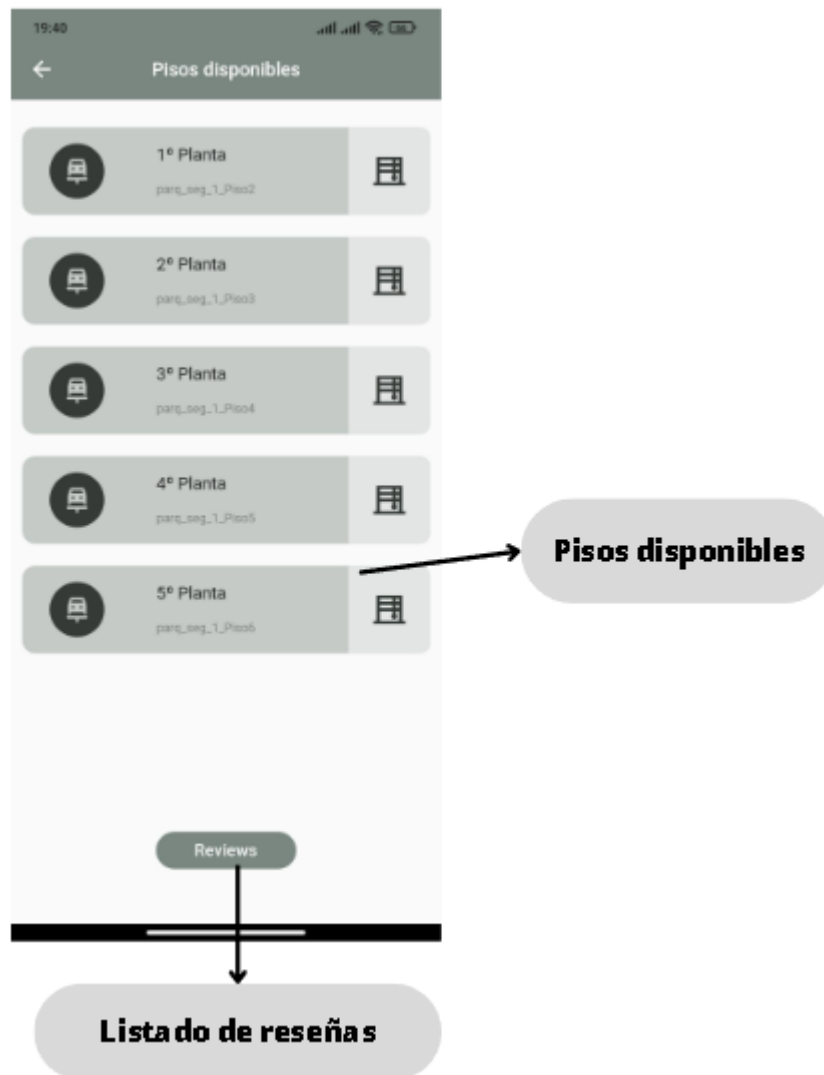
Pantalla de inicio



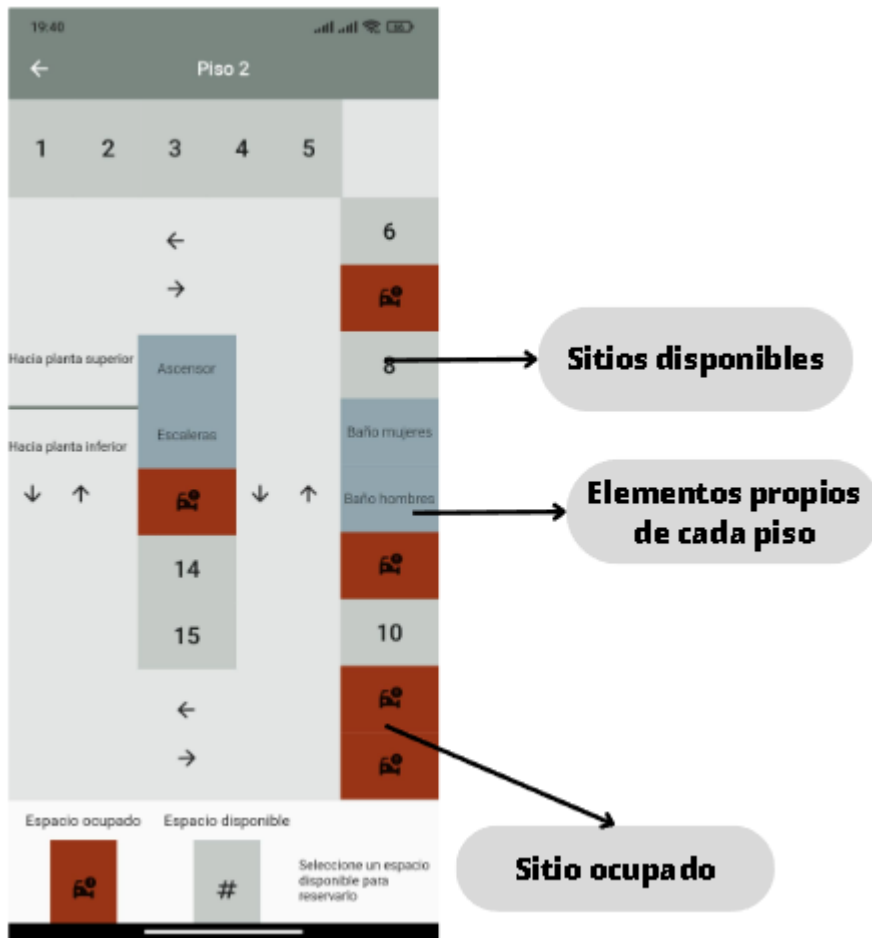
Parqueaderos disponibles



Pisos disponibles



Esquema de piso



Confirmación de renta de sitio



Confirmar renta de sitio

Parqueo en curso y finalizar parqueo



Historial de parqueos



Historial de parqueo

Detalles de un
parqueo finalizado

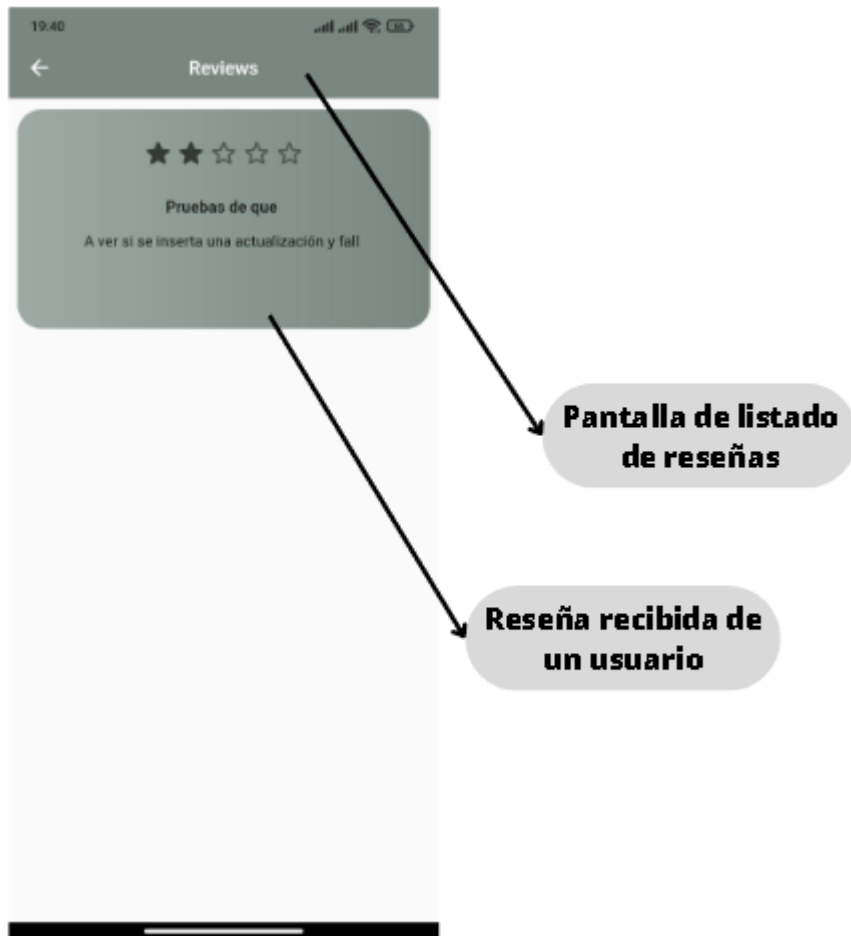
Crear o actualizar reseña

The screenshot shows a mobile application interface for creating or updating a review. At the top, the status bar displays the time 19:40 and signal strength. Below the status bar is a dark header with a back arrow and the title 'Reseña'. The main content area has a light background and contains the following elements: the name of the business 'Parqueo Seguro 1', a star rating of five green stars, a text input field with the placeholder 'Pruebas de que', and another text input field with the placeholder 'Reseña' and a sub-placeholder 'A ver si se inserta una actualización y fall'. At the bottom of the form is a dark button labeled 'Enviar'.

Pantalla de reseña

**Enviar reseña o
actualizar reseña
existente**

Listar reseñas



Perfil de usuario y finalizar sesión

