



UNIVERSIDAD TÉCNICA DE AMBATO

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

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN

Tema:

**APLICACIÓN MÓVIL E-COMMERCE UTILIZANDO FLUTTER PARA LA
GESTIÓN DE VENTAS DE INSUMOS TEXTILES Y PUBLICITARIOS EN
LA EMPRESA MERCANTIL MAKAMBA CIA. LTDA.**

Trabajo de Integración Curricular, 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: Jorge Eduardo Pozo Pozo

TUTOR: Ing. Oscar Fernando Ibarra Torres

Ambato - Ecuador

marzo – 2023

APROBACIÓN DEL TUTOR

En calidad de tutor del Trabajo de Integración Curricular con el tema: APLICACIÓN MÓVIL E-COMMERCE UTILIZANDO FLUTTER PARA LA GESTIÓN DE VENTAS DE INSUMOS TEXTILES Y PUBLICITARIOS EN LA EMPRESA MERCANTIL MAKAMBA CIA. LTDA, desarrollado bajo la modalidad Proyecto de Investigación por el señor Jorge Eduardo Pozo Pozo, 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 de las segundas reformas al Reglamento para la ejecución de la Unidad de Integración Curricular y la obtención del título de tercer nivel, de grado en la Universidad Técnica de Ambato y el numeral 7.4 del respectivo instructivo del reglamento.

Ambato, marzo 2023.

Ing. Oscar Fernando Ibarra Torres
TUTOR

AUTORÍA

El presente Trabajo de Integración Curricular titulado: APLICACIÓN MÓVIL E-COMMERCE UTILIZANDO FLUTTER PARA LA GESTIÓN DE VENTAS DE INSUMOS TEXTILES Y PUBLICITARIOS EN LA EMPRESA MERCANTIL MAKAMBA CIA. LTDA. es absolutamente original, auténtico y personal. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, marzo 2023.



Jorge Eduardo Pozo Pozo

C.C. 1803318813

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Integración Curricular como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Integración Curricular en favor de la Universidad Técnica de Ambato, con fines de difusión pública. Además, autorizo su reproducción total o parcial dentro de las regulaciones de la institución.

Ambato, marzo 2023.



Jorge Eduardo Pozo Pozo

C.C. 1803318813

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del Informe Final del Trabajo de Integración Curricular presentado por el señor Jorge Eduardo Pozo Pozo, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado **APLICACIÓN MÓVIL E-COMMERCE UTILIZANDO FLUTTER PARA LA GESTIÓN DE VENTAS DE INSUMOS TEXTILES Y PUBLICITARIOS EN LA EMPRESA MERCANTIL MAKAMBA CIA. LTDA.** nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 de las segundas reformas al Reglamento para la ejecución de la Unidad de Integración Curricular y la obtención del título de tercer nivel, de grado en la Universidad Técnica de Ambato y al numeral 7.6 del respectivo instructivo del reglamento. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, marzo 2023.

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

Ing. Hernán Fabricio Naranjo Ávalos
PROFESOR CALIFICADOR

Ing. Marlon Antonio Santamaría Villacis
PROFESOR CALIFICADOR

DEDICATORIA

Este trabajo se lo dedico a mis padres y hermanos, quienes nunca perdieron la fe en mí. Esta es la prueba viva de que la disciplina superará a la inteligencia, aunque yo no tenga ninguna de las dos.

AGRADECIMIENTO

En primer lugar, quiero expresar mi más sincero agradecimiento a mi director de tesis, por su paciencia, dedicación y apoyo constante durante todo este proceso. Gracias por guiarme en la dirección correcta y por ser un mentor excepcional.

También quiero agradecer a mis compañeros Amarga2, Memo, Curandero y Fevin por enseñarme que no importa cuánto practiques algo, el que nace en bronce muere en bronce.

Un agradecimiento especial a mi familia y amigos, por su incondicional apoyo, ánimo y comprensión. Gracias por estar siempre a mi lado, apoyándome en cada paso que di.

ÍNDICE GENERAL DE CONTENIDOS

CAPITULO I.- MARCO TEÓRICO	1
1.1 Tema de Investigación	1
1.1.1 Planteamiento del Problema.....	1
1.2 Antecedentes Investigativos.....	2
1.3 Fundamentación Teórica.....	4
1.3.1 Internet.....	4
1.3.2 Plataformas E-commerce.....	4
1.3.3 Comercio Electrónico	4
1.3.4 Sistemas de Información	5
1.3.5 Desarrollo de Software	5
1.3.6 Desarrollo Móvil	5
1.3.7 Aplicación Móvil Multiplataforma.....	6
1.3.8 Flutter	6
1.3.9 Widget	7
1.3.10 Dart.....	8
1.3.11 API.....	8
1.3.12 Node.js.....	8
1.3.13 Almacenamiento en la nube	8
1.4 Objetivos	9
1.4.1 Objetivo General	9
1.4.2 Objetivos Específicos	9
CAPITULO II.- METODOLOGÍA	10
2.1 Materiales.....	10
2.2 Métodos.....	11
2.2.1 Modalidad de la Investigación.....	11
2.2.2 Población y Muestra.....	11
2.2.3 Recolección de Información.....	12
2.2.4 Procesamiento y Análisis de Datos	23
CAPÍTULO III.- RESULTADOS Y DISCUSIÓN	24

3.1 Análisis y discusión de resultados	24
3.1.1 Comparativa de Metodologías de Desarrollo Móvil	24
3.1.2 Metodología seleccionada	25
3.1.3 Arquitectura de la aplicación.....	25
3.2 Desarrollo de la propuesta.....	26
3.2.1 Fase I: Exploración.....	29
3.2.2 Fase II: Inicialización	33
3.2.3 Fase III Producción	43
3.2.4 Fase IV: Estabilización.....	65
3.2.5 Fase V: Pruebas	67
CAPÍTULO IV.- CONCLUSIONES Y RECOMENDACIONES	75
4.1 Conclusiones	75
4.2 Recomendaciones.....	76
5. Bibliografía	76
6. Anexos	78

ÍNDICE DE TABLAS

Tabla 1. Población	12
Tabla 2 Medios de acceso a Internet.....	12
Tabla 3 Compras por E-commerce	13
Tabla 4 Nivel de Seguridad	14
Tabla 5 Métodos de pago.....	15
Tabla 6 Alojamiento E-commerce.....	17
Tabla 7 Tiempo de envío	17
Tabla 8 Fidelización del Cliente	19
Tabla 9 Facturación	19
Tabla 10 Disponibilidad	20
Tabla 11 Máximo Gastado.....	21
Tabla 12 Comparativa entre Metodologías.....	24
Tabla 13 Planificación de Fases.....	29
Tabla 14 Roles de usuarios	30
Tabla 15 Requerimientos no funcionales.....	32
Tabla 16 Requerimientos Funcionales.....	33
Tabla 17 Storycard - Registro de usuario	47
Tabla 18 Storycard - Inicio de Sesión.....	48
Tabla 19 Storycard - Subir archivos multimedia	49
Tabla 20 Storycard – Catálogo de los Productos	51
Tabla 21 Storycard - Agregar productos al carrito de compras.....	52
Tabla 22 Strorycard - Carrito de Compras.....	53
Tabla 23 Storycard - Agregar direcciones de envío	54
Tabla 24 Storycard - Geolocalización del dispositivo	55
Tabla 25 Storycard - Realizar pedidos.....	56
Tabla 26 Storycard - Búsqueda de Productos.....	57
Tabla 27 Storycard - Cerrar Sesión.....	58
Tabla 28 Storycard - Realizar el Pago	59
Tabla 29 Storycard - Crear Productos.....	60
Tabla 30 Storycard - Crear nueva categoría	61
Tabla 31 Storycard - Verificar el pago	62
Tabla 32 Storycard - Despachar Productos.....	63

Tabla 33 Storycard - Seleccionar rol	64
Tabla 34 Storycard - Eliminar tablas de provincia y ciudad.....	65
Tabla 35 Storycard - Ajustar métodos de pago.....	65
Tabla 36 Storycard - Eliminar tablas de fotos	66
Tabla 37 Storycard - Eliminar tabla de líneas.....	66

ÍNDICE DE GRÁFICOS

Gráfico 1 Estructura de una aplicación de Flutter	7
Gráfico 2 Medios de acceso a Internet.....	13
Gráfico 3 Compras por E-commerce	14
Gráfico 4 Nivel de Seguridad	15
Gráfico 5 Métodos de pago.....	16
Gráfico 6 Alojamiento E-commerce.....	17
Gráfico 7 Tiempo de envío	18
Gráfico 8 Fidelización del Cliente	19
Gráfico 9 Facturación	20
Gráfico 10 Disponibilidad	21
Gráfico 11 Máximo Gastado.....	22
Gráfico 12 Proceso de E-commerce	31
Gráfico 13 Arquitectura MVVM	25
Gráfico 14 Modelo Entidad Relación	35
Gráfico 15 Diseño lógico de la base de datos.....	36
Gráfico 16 Prototipo de pantalla de inicio de sesión.....	37
Gráfico 17 Prototipo de pantalla de registro.....	38
Gráfico 18 Prototipo de pantalla de categorías.....	38
Gráfico 19 Prototipo de catálogo	39
Gráfico 20 Prototipo de Pantalla de producto.....	39
Gráfico 21 Prototipo de pantalla de perfil	40
Gráfico 22 Prototipo de carrito de compras.....	41
Gráfico 23 Prototipo de pantalla de pedidos.....	41
Gráfico 24 Casos de uso de clientes	42
Gráfico 25 Casos de uso del administrador	42
Gráfico 26 Casos de uso de bodega	43
Gráfico 27 Backend de registro de usuarios	48
Gráfico 28 Backend de Inicio de Sesión.....	49
Gráfico 29 Backend para subir archivos multimedia.....	50
Gráfico 30 Backend para el catálogo de productos	51
Gráfico 31 Frontend para agregar productos al carrito de compras	52
Gráfico 32 Frontend de carrito de compras	53

Gráfico 33 Backend para agregar direcciones de envío	54
Gráfico 34 Frontend de la geolocalización del envío	55
Gráfico 35 Frontend para crear pedidos	56
Gráfico 36 Backend para buscar productos por nombre.....	57
Gráfico 37 Frontend para Cerrar Sesión	58
Gráfico 38 Frontend para realizar el pago	59
Gráfico 39 Backend para crear productos.....	61
Gráfico 40 Backend para crear nuevas categoría.....	61
Gráfico 41 Backend para verificar el pago	62
Gráfico 42 Backend para despachar productos.....	63
Gráfico 43 Backend para seleccionar rol.....	64
Gráfico 44 Pantalla de Inicio de Sesión.....	67
Gráfico 45 Pantallas de Registro de Usuario	68
Gráfico 46 Pantalla de Catálogo de Productos	68
Gráfico 47 Pantalla de Detalle de Producto.....	69
Gráfico 48 Pantalla de Carrito de Compras	70
Gráfico 49 Pantallas de Direcciones de Envío.....	70
Gráfico 50 Pantallas de Pedido y Pago.....	71
Gráfico 51 Pantalla de Listado de Pedidos	71
Gráfico 52 Pantalla de Agregar nueva categoría	72
Gráfico 53 Pantalla de agregar nuevo producto.....	73
Gráfico 54 Pantalla de Listado de Pedidos	73
Gráfico 55 Pantalla para despachar productos.....	74

ÍNDICE DE ANEXOS

Anexo 1 Ficha Bibliográfica 1.....	78
Anexo 2 Ficha Bibliográfica 2.....	79
Anexo 3 Ficha Bibliográfica 3.....	80

RESUMEN EJECUTIVO

El comercio electrónico en Ecuador ha experimentado un notable aumento en los últimos años impulsado por la pandemia del COVID-19. Sin embargo, todavía hay desafíos a superar, como la falta de confianza de los consumidores con relación a la seguridad de las compras en línea o la escasa infraestructura logística para la gestión de los envíos. A pesar de estos desafíos, el futuro del comercio electrónico en Ecuador es prometedor gracias al aumento de la penetración de internet y la mejora de la infraestructura de pago.

El presente proyecto de investigación tiene como principal objetivo aportar al departamento comercial de la empresa MercantilMakamba Cia. Ltda. mediante la implantación de una aplicación móvil E-commerce con la cual se pretende incrementar las ventas de la empresa y agilizar el proceso comercial y la comunicación con los clientes.

Para el cumplimiento de los objetivos planteados en el proyecto se utilizarán varias herramientas con el fin de desarrollar una aplicación móvil rápida, segura y fácil de usar. Herramientas como el entorno de desarrollo Flutter para desarrollar el frontend, Node.js para el desarrollo del backend, MySQL como motor principal de base de datos y Firebase para almacenar los archivos multimedia utilizados por la aplicación. Gracias al uso eficiente de las herramientas anteriormente mencionadas se logrón desarrollar una aplicación enfocada en el cliente y en el usuario, eficiente, escalable, segura e intuitiva.

Palabras clave: Backend, Frontend, E-commerce, Servicios Web, Flutter, Dart, Widgets.

ABSTRACT

E-commerce in Ecuador has experienced a remarkable increase in recent years, driven by the COVID-19 pandemic. However, there are still challenges to overcome, such as the lack of consumer trust in the security of online shopping or the scarce logistics infrastructure for managing shipments. Despite these challenges, the future of e-commerce in Ecuador is promising thanks to the increased penetration of the internet and the improvement of payment infrastructure.

The main objective of this research project is to contribute to the commercial department of MercantilMakamba Cia. Ltda. by implementing an E-commerce mobile application that aims to increase the company's sales and streamline the commercial process and communication with customers.

To achieve the objectives set out in the project, several tools will be used to develop a fast, secure, and user-friendly mobile application. Tools such as the Flutter development environment for developing the frontend, Node.js for developing the backend, MySQL as the main database engine, and Firebase for storing multimedia files used by the application. Thanks to the efficient use of the aforementioned tools, an application focused on the customer and user, efficient, scalable, secure, and intuitive has been developed.

Keywords: Backend, Frontend, E-commerce, Web Services, Flutter, Dart, Widgets.

CAPITULO I.- MARCO TEÓRICO

1.1 Tema de Investigación

APLICACIÓN MÓVIL E-COMMERCE UTILIZANDO FLUTTER PARA LA GESTIÓN DE VENTAS DE INSUMOS TEXTILES Y PUBLICITARIOS EN LA EMPRESA MERCANTIL MAKAMBA CIA. LTDA.

1.1.1 Planteamiento del Problema

Conforme con el artículo “El e-commerce y las Mipymes en tiempos de Covid-19” [1], el mundo ha cambiado significativamente debido a la emergencia generada por el COVID-19, la cual ha marcado un punto de inflexión y un desafío para la humanidad. La pandemia del COVID-19 ha influido para que se incremente el uso del software en la educación, en el sector laboral, en el entretenimiento y sobre todo en las relaciones interpersonales. El sector laboral adoptó una nueva dinámica, el teletrabajo, el sector del entretenimiento los servicios de streaming y las relaciones interpersonales junto con la educación se virtualizaron. Ecuador el 82,3% de las Mipymes utilizan la internet. La proporción de uso es la siguiente: microempresas 48,6%, medianas 56,9% y pequeñas 52,9%, dando un total general de 52,8%. Por lo tanto, hay indicios de que el internet junto con las redes sociales y páginas de comercio electrónico se convertirán en servicios indispensables para lograr un pleno desenvolvimiento dentro de la sociedad.

Según el artículo “Impact of COVID 19 on E-Commerce” [2], El comercio electrónico ha experimentado un rápido crecimiento a nivel mundial y se proyecta que las ventas crezcan a 599.2 mil millones de dólares para 2024. Como consecuencia de la pandemia del COVID-10, las ventas a través de comercio electrónico aumentaron un 25% solamente en marzo del 2020.

En Ecuador, de acuerdo con el reporte de transacciones electrónicas de la Cámara de Comercio Electrónico las compras por medio de comercio electrónico aumentaron un 22% en el rango de personas de 19 a 26 años en el año 2020. En base a los análisis realizados por la Cámara de Comercio Electrónico referentes a la frecuencia de compras en línea pre-COVID-19 y post-COVID-19, existe una tendencia importante

en la cual las empresas con altas proyecciones de crecimiento requieren contar con algún tipo de plataforma E-commerce como parte indispensable de sus actividades comerciales.[3]

De acuerdo con los datos presentados por el Instituto Nacional de Estadística y Censos (INEC) en 2012, de la población total del Ecuador, el 64% viven en zonas urbanas, 89% poseen líneas telefónicas de los cuales el 69% tiene acceso a internet. Al momento de analizar estos datos se puede deducir que el comercio electrónico ha adquirido una gran ventaja. [4]

Mercantil Makamba Cia. Ltda. Es una empresa ecuatoriana con más de 30 años en el mercado la cual se dedica a la comercialización de maquinaria e insumos en el sector publicitario y textil. Actualmente cuenta con cuatro sucursales en las ciudades de Guayaquil, Quito, Ambato y Cuenca. Debido a la constante expansión de la organización dentro del territorio nacional, existe la necesidad de mejorar el proceso de comercialización de insumos publicitarios y textiles mediante una aplicación E-commerce.

1.2 Antecedentes Investigativos

Para realizar este proyecto de investigación se consultó información de varias fuentes, entre ellas, artículos científicos, tesis, e información de medios oficiales relacionada con el desarrollo de aplicaciones con Flutter y con la importancia del E-commerce en nuestra sociedad actual.

Como indica Wenhao Wu [5] en su artículo titulado “React-native vs Flutter, cross-platform mobile application frameworks”, luego de realizar una comparativa entre React-native y Flutter escribiendo la misma aplicación en los dos frameworks concluye:

- Las aplicaciones escritas en Flutter-Dart son considerablemente más eficiente y requieren menos recursos al momento de su ejecución.
- La comunidad de React-native es significativamente más próspera que la comunidad de Flutter, aunque tomando en cuenta el año de publicación del artículo (2018) se puede concluir que la comunidad de Flutter estaba en una etapa aún muy temprana.

Como señala Altamirano [6] en su tesis “Aplicación móvil multiplataforma de pago en línea Aplicando Flutter y Dart, para la obtención de boletos en la Cooperativa de transporte Cita Express de la Ciudad de Ambato”, trabajo realizado como proyecto de investigación en la Universidad Técnica de Ambato en el año 2021 se puede concluir que:

- Paypal es uno de los métodos de pago más conocidos, el cual surge en 1998 y cuyas ventajas son las transferencias de dinero inmediatas, el cambio de divisas y la seguridad de las transacciones.
- Stripe es otro de los métodos de pago más usados, el cual lleva más de diez años en el mercado y acepta más de cien monedas diferentes. Dentro de las ventajas de Stripe encontramos la gratuidad al momento de crear una cuenta, la compatibilidad con los distintos bancos y la seguridad de las transacciones.

Como señala Felipa Barrientos [7] en su artículo “Marketing + internet = e-commerce: oportunidades y desafíos”, artículo que estudia las ventajas de combinar el marketing con el internet, se puede concluir que:

- Dentro del campo comercial, la tecnología combinada con el intercambio de bienes y servicios puede generar temores, pero esta es solo una cara de la moneda, ya que el rápido avance de la tecnología también puede crear nuevas oportunidades.
- Internet es una herramienta la cual contribuye a la economía de una nación tanto en el ámbito público como en el ámbito privado. En el ámbito público lo utilizan los gobiernos para prestar sus servicios a la comunidad. En el ámbito privado lo utilizan las empresas para promover y vender sus bienes o servicios.

Según Daniel Marroquín [8] en su tesis “Factores que han influenciado la intención de compra de los consumidores quiteños respecto al E Commerce a partir de la pandemia COVID-19 en el año 2021”, trabajo en el que se estudian los factores que han afectado a la intención de compra por internet en Quito a partir de la pandemia, se puede concluir que:

- A raíz de la pandemia del COVID-19, la economía en Latinoamérica y el Caribe causo la disminución del PIB de la región en un -9.4%, llegando a alcanzar valores similares a los obtenidos en 2010.

- Clientes de distintos rangos de edad estarían dispuestos a comprar en tiendas en línea si las empresas hicieran una innovación en sus plataformas virtuales, de tal manera que ofrezcan mayor seguridad, confianza y eficiencia.

1.3 Fundamentación Teórica

1.3.1 Internet

Internet es un producto que nace como herramienta de guerra en 1957 de las manos del departamento de defensa de Estados Unidos. Posteriormente se creó el protocolo IP/TCP (conjunto de normas que define cómo se mueve la información) el cual aún se usa en la actualidad. Actualmente internet es una red mundial de computadoras interconectadas entre sí, esta red tiene forma de tela de araña la cual es utilizada para proveer a los usuarios servicios tales como redes sociales, venta de productos, entre otros. [9]

1.3.2 Plataformas E-commerce

Las Plataformas E-commerce son plataformas que utilizan Internet para hacer factibles las operaciones que tienen como fin la venta o el intercambio de bienes, operaciones tales como la exhibición de productos, servicios, control de stock o promociones. El principal objetivo de una plataforma E-commerce es ofrecer productos y servicios sin tener la necesidad de visitar una tienda física. Una plataforma E-commerce también podría tener la capacidad de facilitar el proceso logístico de la entrega del producto. [10]

1.3.3 Comercio Electrónico

El comercio electrónico se ha convertido en una de las formas de negocio más populares y efectivas en la actualidad. Con el auge de la tecnología y el aumento del acceso a internet, cada vez son más los consumidores que realizan sus compras en línea. El comercio electrónico permite a los negocios llegar a una audiencia global, ofrecer una amplia variedad de productos y servicios, y brindar una experiencia de compra más cómoda y accesible para los clientes. Además, el comercio electrónico ofrece ventajas como la posibilidad de realizar compras en cualquier momento del día, la comparación de precios y productos de manera fácil y rápida, y la entrega a domicilio, lo que se traduce en mayor comodidad para los consumidores.[9]

La necesidad del comercio electrónico se origina de la demanda de las empresas y de la administración, para hacer un mejor uso de la informática y buscar una mejor forma de aplicar las nuevas tecnologías para así mejorar la interrelación entre cliente y proveedor.[11]

1.3.4 Sistemas de Información

Los sistemas de información son herramientas tecnológicas que se utilizan para procesar, almacenar y administrar datos e información. Estos sistemas pueden ser utilizados en una variedad de contextos, incluyendo empresas y organizaciones gubernamentales. Los sistemas de información pueden incluir software, hardware, bases de datos y son fundamentales para la toma de decisiones estratégicas y operativas.[12]

1.3.5 Desarrollo de Software

Desarrollar Software es crear programas de calidad, los cuales atiendan a las necesidades y cumplan con los requisitos que demandan los usuarios. Dentro del desarrollo de software, las interfaces deben ser amigables con el usuario, es decir, que sea fácil de usar. Cabe recalcar que desarrollar software es una actividad muy compleja, sobre todo por los estándares elevados que tienen los usuarios actualmente.[13]

1.3.6 Desarrollo Móvil

El desarrollo móvil se refiere la construcción de aplicaciones para dispositivos móviles como teléfonos inteligentes y tabletas. Estas aplicaciones pueden ser nativas, es decir, desarrolladas específicamente para un sistema operativo en particular, como Android o iOS, o pueden ser híbridas, las cuales se ejecutan en múltiples plataformas. El desarrollo móvil es importante ya que los dispositivos portátiles son cada vez más populares y se han convertido en una herramienta esencial en la vida cotidiana de muchas personas. Las aplicaciones desarrolladas pueden ser utilizadas para una variedad de propósitos, desde juegos y entretenimiento hasta usos empresariales y productivos.[14]

1.3.7 Aplicación Móvil Multiplataforma

Las Aplicaciones Móviles Multiplataforma son aplicaciones escritas en un solo lenguaje con el fin de optimizar la relación costo/beneficio, lo cual conlleva a tener varias ventajas tales como menor tiempo de desarrollo y menor costo. Las aplicaciones multiplataforma pueden clasificarse en: aplicaciones web móviles, híbridas, interpretadas y generadas por compilación cruzada.[14]

El desarrollo móvil multiplataforma procura optimizar la relación costo/beneficio compartiendo la misma codificación entre las versiones del software para las distintas plataformas. Sin embargo, el rendimiento de las aplicaciones y sus interfaces de usuario, pueden afectar la experiencia de usuario.[15]

1.3.8 Flutter

Flutter es un framework creado por Google en 2017. Google afirma que Flutter es una herramienta diseñada para desarrollar aplicaciones para dispositivos móviles compiladas nativamente. Tim Sneath, Director de productos de Google, define Flutter como “Una herramienta poderosa de propósito general”. Así mismo Flutter provee su propio set de objetos para interfaces, renderizados y un motor que contiene sus propias animaciones e imágenes.[16]

Gracias a la arquitectura basada en widgets de Flutter y su motor de renderizado, hacen que la creación de aplicaciones sea más rápida, eficiente y flexible. La comunidad de desarrolladores de Flutter está en constante crecimiento, lo cual asegura poder encontrar una gran cantidad de soporte en línea y una amplia variedad de librerías y paquetes disponibles, por estos motivos Flutter se ha convertido en una de las opciones más populares para el desarrollo de aplicaciones móviles.

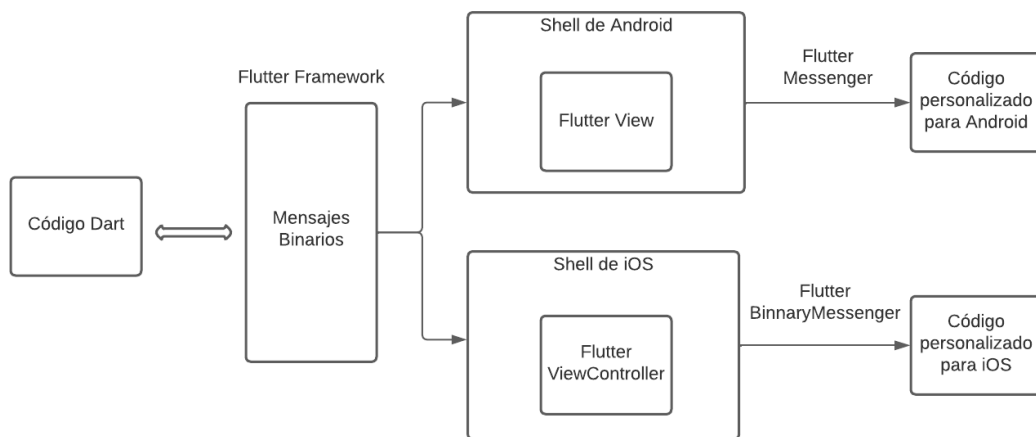


Gráfico 1 Estructura de una aplicación de Flutter

Elaborado por: El Investigador

1.3.9 Widget

Los Widgets son los elementos más importantes dentro de una aplicación hecha en Flutter ya que son los únicos componentes de interfaz del framework, La documentación de Flutter define a los Widgets como “La manera de declarar y construir Interfaces de Usuario”.[16]

Un Widget es todo elemento que aparece en la pantalla dentro de una aplicación de Flutter. Según Google, los Widgets describen como debería verse su vista dada su configuración y estado actual. Existen dos tipos de Widgets: “Stateful Widgets” y “Stateless Widgets”.

Stateless Widgets

Es un tipo de widget cuyas propiedades son invariables, es decir, deberán ser declaradas finales dentro de la codificación. Son muy útiles al momento de desarrollar interfaces ya que ayuda a reducir la cantidad de código repetido.[17]

Statefull Widget

Son lo contrario de los Stateless Widgets, pueden cambiar su comportamiento y sus propiedades dependiendo de las necesidades del contexto en el que se utilizan. Estos widgets se conforman de dos partes, un objeto widget invariable y un objeto de estado variable.[17]

1.3.10 Dart

Todas las aplicaciones de Flutter están escritas con Dart. Dart es un lenguaje de programación creado por Google, quienes actualmente lo siguen desarrollando y manteniendo. Dart es conocido por haber sido desarrollado para reemplazar a JavaScript por lo cual conserva algunas de sus características, sin embargo, su sintaxis es más parecida a Java, esto con el fin de atraer a desarrolladores que no están familiarizados con JavaScript.[5]

1.3.11 API

API, por sus siglas en inglés Application Programming Interface, es una interfaz de un conjunto de librerías o software la cual mediante un conjunto de reglas y protocolos permite que estos software y librerías puedan interactuar con otros sistemas informáticos. En resumen, una API es una herramienta la cual permite que dos sistemas independientes se comuniquen entre sí.[18]

1.3.12 Node.js

Node.js es un Framework basado en JavaScript orientado a eventos asíncronos el cual se utiliza para crear aplicaciones de red por lo cual se ejecuta del lado del servidor. Node.js se basa en una arquitectura non-blocking I/O que le permite manejar grandes cantidades de solicitudes de manera eficiente. Las aplicaciones de Node.js permanecen en estado de espera si no hay una solicitud, en el momento en que se realice una petición, la conexión con el servidor de Node.js se activa.[19]

1.3.13 Almacenamiento en la nube

El almacenamiento en la nube es un modelo de almacenamiento proporcionado por empresas especializadas las cuales tienen una gran cantidad de servidores y discos duros alojados en centros de datos, empresas tales como Google, Dropbox o Mega. Este servicio permite a los usuarios almacenar, acceder y compartir archivos mediante internet.[20]

1.4 Objetivos

1.4.1 Objetivo General

Implantar una aplicación móvil de E-commerce utilizando Flutter en la empresa Mercantil Makamba Cia. Ltda.

1.4.2 Objetivos Específicos

Investigar el proceso de creación de aplicaciones móviles E-commerce.

Analizar el uso de Flutter para la creación de aplicaciones móviles.

Desarrollar una aplicación móvil E-commerce utilizando Flutter y Dart para la gestión de ventas de la empresa Mercantil Makamba Cia. Ltda.

CAPITULO II.- METODOLOGÍA

2.1 Materiales

Para el presente proyecto de investigación se utilizó una encuesta dirigida a los empleados de la empresa MercantilMakamba Cia. Ltda. Con el objetivo de recolectar información sobre experiencia de usuario de aplicaciones E-commerce.

También se utilizaron fichas bibliográficas con el fin de recopilar información sobre la creación de aplicaciones móviles E-commerce.

Encuesta dirigida a los empleados de la empresa MercantilMakamba Cia. Ltda.

1. ¿Desde qué medios tecnológicos ingresa comúnmente a Internet?

Celular() Computadora() Tablet () Televisión ()

2. ¿Alguna vez ha adquirido productos o servicios mediante una aplicación E-commerce?

Si () No ()

3. ¿Se siente seguro al momento de comprar a través de una aplicación E-commerce?

Indique un valor del 1 al 5 siendo 1 totalmente inseguro y 5 totalmente seguro.

1 () 2 () 3 () 4 () 5 ()

4. ¿Si ha adquirido productos o servicios mediante una aplicación E-commerce, cuáles son los métodos de pago que ha utilizado?

Paypal [] Stripe [] Skrill [] Payphone [] No he comprador [] Otro []

5. ¿Cuánto tiempo estaría dispuesto a esperar para recibir el producto o servicio adquirido por medio de la plataforma E-commerce?

1 día () mas de 2 días () entre 3 y 7 días () más de una semana ()

6. ¿Considera usted que las plataformas E-commerce ayudan a fidelizar a los clientes?

Indique un valor del 1 al 5 siendo 1 totalmente inseguro y 5 totalmente seguro.

1 () 2 () 3 () 4 () 5 ()

7. ¿Con qué información le gustaría registrarse en una aplicación E-commerce?

Cédula () Nombre de usuario () Correo Electrónico () Numero de Celular ()

8. ¿Le gustaría que una aplicación E-commerce le entregue una factura al momento de finalizar la compra?

Si () No ()

9. ¿Le gustaría que la aplicación E-commerce esté disponible los fines de semana o feriados?

Si () No ()

10. ¿Cuál es el monto máximo que ha gastado comprando en una aplicación E-commerce?

Menos de 100 () Más de 100 () Menos de 500 () Más de 500 () No he Comprado ()

2.2 Métodos

2.2.1 Modalidad de la Investigación

Las modalidades de investigación fueron Bibliográfica, de Campo y Proyecto factible.

Investigación Bibliográfica

En el presente proyecto se revisaron varias fuentes bibliográficas, las cuales incluyen documentos, artículos, libros, etc., para la redacción y construcción del marco teórico referente a las aplicaciones móviles desarrolladas en Flutter y el E-commerce.

Investigación de Campo

Porque en el proyecto se realizaron las observaciones necesarias en el proceso de venta, en el cual se enfoca este proyecto de investigación. También porque se obtendrán datos directamente del departamento de ventas de la empresa con el fin de encontrar una solución adecuada y eficiente.

2.2.2 Población y Muestra

La población para la presente investigación está construida de la siguiente manera.

Nº	Tipo de población	Descripción	Número
1	Personal de la Empresa MercantilMakamba Cia. Ltda.	Empleados de todos los departamentos	20
Total			20

Tabla 1. Población

Elaborado por: El Investigador

2.2.3 Recolección de Información

Los resultados presentados a continuación se obtuvieron a partir de la aplicación de la encuesta al personal de la empresa; así como también se obtuvieron resultados luego de revisar varias tesis y artículos científicos.

Resultados de la encuesta

Encuesta dirigida a los empleados de la empresa MercantilMakamba Cia. Ltda.

Pregunta 1: ¿Desde qué medios tecnológicos ingresa comúnmente a Internet?

OPCIONES	RESPUESTAS	PORCENTAJE
Celular	17	85%
Computadora	3	15%
Tablet	0	0
Televisión	0	0

Tabla 2 Medios de acceso a Internet

Elaborado por: El investigador

¿Desde qué medios tecnológicos ingresa comúnmente a Internet?

20 respuestas

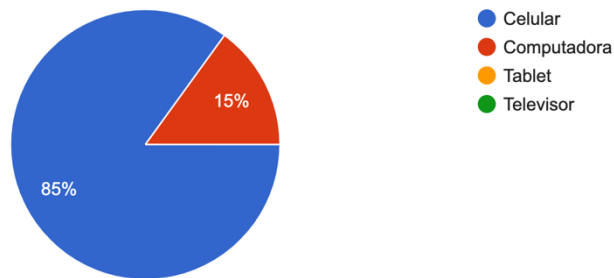


Gráfico 2 Medios de acceso a Internet

Elaborado por: El Investigador

Análisis e Interpretación

El 85% de los encuestados accede a Internet por medio de su celular y el 15% por medio de una computadora, esto significa que mediante el desarrollo de una aplicación móvil la empresa tendría mayor alcance para vender sus productos.

Pregunta 2: ¿Alguna vez ha adquirido productos o servicios mediante una aplicación E-commerce?

OPCIONES	RESPUESTAS	PORCENTAJE
Si	10	50%
No	10	50%

Tabla 3 Compras por E-commerce

Elaborado por: El Investigador

¿Alguna vez ha adquirido productos o servicios mediante una aplicación Ecommerce?

20 respuestas

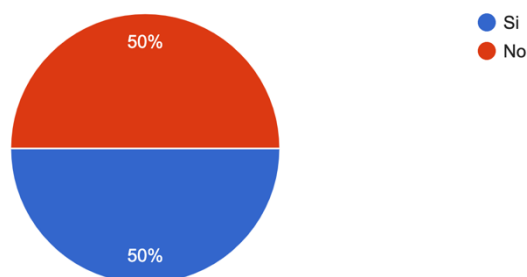


Gráfico 3 Compras por E-commerce

Elaborado por: El Investigador

Análisis e Interpretación

El 50% de los encuestados ha comprado mediante una aplicación E-commerce y el otro 50% no lo ha hecho, esto podría significar que este modelo de negocio aún está inexplorado dentro de nuestro país.

Pregunta 3: ¿Se siente seguro al momento de comprar a través de una aplicación E-commerce?

Indique un valor del 1 al 5 siendo 1 totalmente inseguro y 5 totalmente seguro.

OPCIONES	RESPUESTAS	PORCENTAJE
1	2	10%
2	2	10%
3	10	50%
4	5	25%
5	1	5%

Tabla 4 Nivel de Seguridad

Elaborado por: El Investigador

¿Se siente seguro al momento de comprar a través de una aplicación Ecommerce? Indique un valor del 1 al 5 siendo 1 totalmente inseguro y 5 totalmente seguro.

20 respuestas

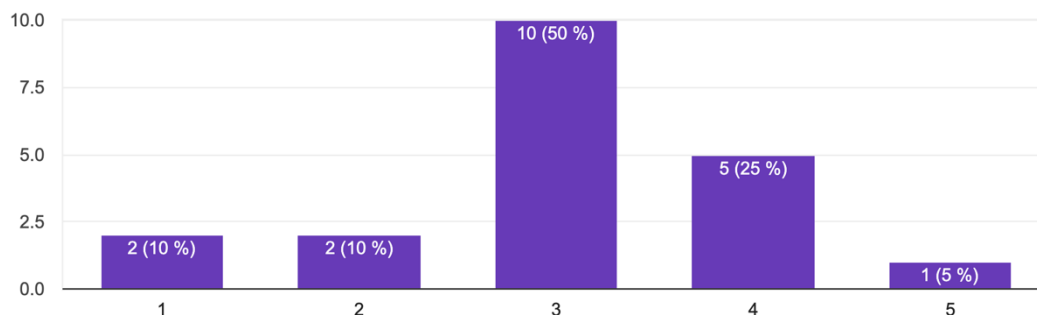


Gráfico 4 Nivel de Seguridad

Elaborado por: El Investigador

Análisis e Interpretación

El 10% de los encuestados considera totalmente inseguro comprar a través de una aplicación E-commerce, el 10% lo considera moderadamente inseguro, el 50% lo considera seguro, el 25% moderadamente seguro y el 5% totalmente seguro. La mayoría de los encuestados se muestran neutrales al momento de realizar compras por medio de una aplicación E-commerce, sin embargo, estos resultados demuestran que las aplicaciones E-commerce deberían generar más seguridad en los clientes.

Pregunta 4: Si ha adquirido productos o servicios mediante una aplicación E-commerce, ¿cuáles son los métodos de pago que ha utilizado?

OPCIONES	RESPUESTAS	PORCENTAJE
Paypal	9	38%
Stripe	0	0%
Skrill	0	0%
Payphone	1	4%
No he Comprado	7	29%
Otro	7	29%

Tabla 5 Métodos de pago

Elaborado por: El Investigador

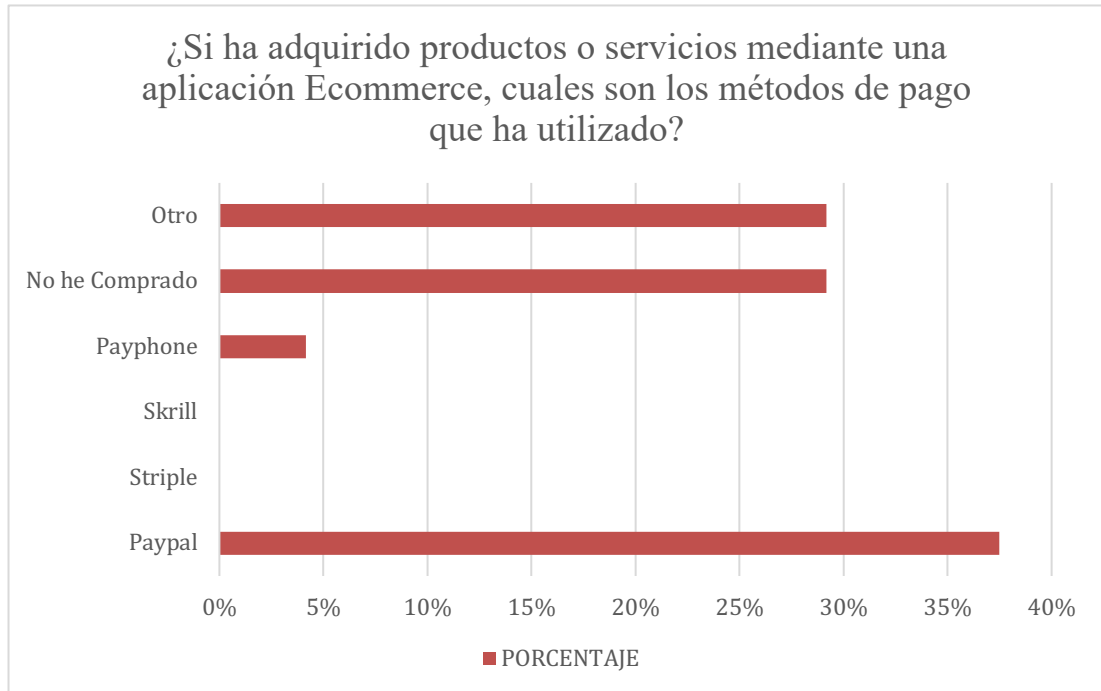


Gráfico 5 Métodos de pago

Elaborado por: El Investigador

Análisis e Interpretación

El 38% de los encuestados han utilizado Paypal, el 4% Payphone, el 29% otros métodos de pago y el 29% restante no ha comprado mediante una aplicación E-commerce. La mayoría de los encuestados ha utilizado Paypal al momento de realizar una compra dentro de una aplicación E-commerce. Esto demuestra la popularidad y seguridad que brinda la plataforma de Paypal. Adicionalmente los encuestados manifestaron, dentro de la opción “Otro” que realizan sus compras mediante transferencia Bancaria.

Pregunta 5: ¿En qué medio están alojadas la mayoría de aplicaciones E-commerce que ha utilizado?

OPCIONES	RESPUESTAS	PORCENTAJE
Aplicación Android	10	50%
Aplicación iOS	1	5%
Aplicación Web	4	20%

Aplicación de Escritorio	0	0%
No he Comprado	5	25%

Tabla 6 Alojamiento E-commerce

Elaborado por: El Investigador

¿En que medio están alojadas la mayoría de aplicaciones Ecommerce que ha utilizado?

20 respuestas

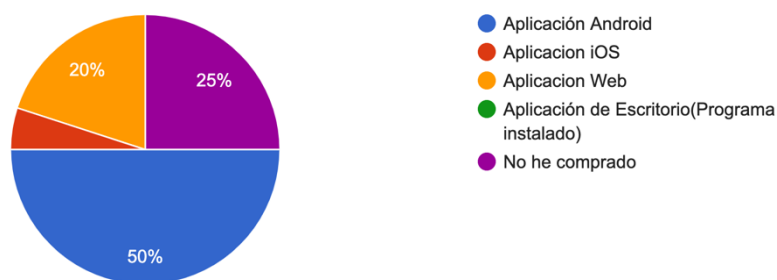


Gráfico 6 Alojamiento E-commerce

Elaborado por: El Investigador

Análisis e Interpretación

El 50% ha comprado en aplicaciones E-commerce en Android, el 20% en web, el 5% en iOS y el 25% restante no ha comprado. Da cuerdo con la Pregunta 1, la mayoría de los usuarios utiliza el sistema operativo Android para realizar las compras por medio de aplicaciones E-commerce, adicionalmente, el sitio web StatCounter señala que en Ecuador, el 90% de los dispositivos móviles utilizan Android y el 10% iOS.

Pregunta 6: ¿Cuánto tiempo estaría dispuesto a esperar para recibir el producto o servicio adquirido por medio de la plataforma E-commerce?

OPCIONES	RESPUESTAS	PORCENTAJE
1 día	4	20%
Más de 2 días	9	45%
Entre 3 y 7 días	7	35%
Más de 7 días	0	0%

Tabla 7 Tiempo de envío

Elaborado por: El Investigador

¿Cuánto tiempo estaría dispuesto a esperar para recibir el producto o servicio adquirido por medio de la plataforma ecommerce?

20 respuestas

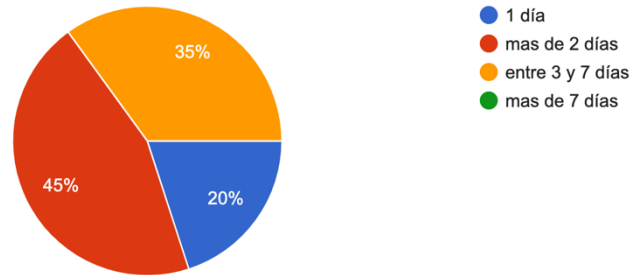


Gráfico 7 Tiempo de envío

Elaborado por: El Investigador

Análisis e Interpretación

El 45% de los encuestados estaría dispuesto a esperar 2 días para recibir su compra, el 35% entre 3 y 7 días y el 20% restante 1 día. Con estos datos se puede concluir que el 100% de los encuestados coincide en que los productos adquiridos mediante una aplicación E-commerce deberían llegar en menos de 7 días. Esto significa que el papel del administrador de la aplicación es de vital importancia para tener éxito con la aplicación E-commerce, ya que una correcta gestión y organización se traduce en menos tiempo de espera para el cliente.

Pregunta 7: ¿Considera usted que las plataformas E-commerce ayudan a fidelizar a los clientes?

Indique un valor del 1 al 5 siendo 1 totalmente en contra y 5 totalmente de acuerdo.

OPCIONES	RESPUESTAS	PORCENTAJE
1	0	0%
2	2	10%
3	3	15%
4	9	45%
5	6	30%

Tabla 8 Fidelización del Cliente

Elaborado por: El Investigador

¿Considera usted que las plataformas Ecommerce ayudan a fidelizar a los clientes? Indique un valor del 1 al 5 siendo 1 totalmente en contra y 5 totalmente de acuerdo.

20 respuestas

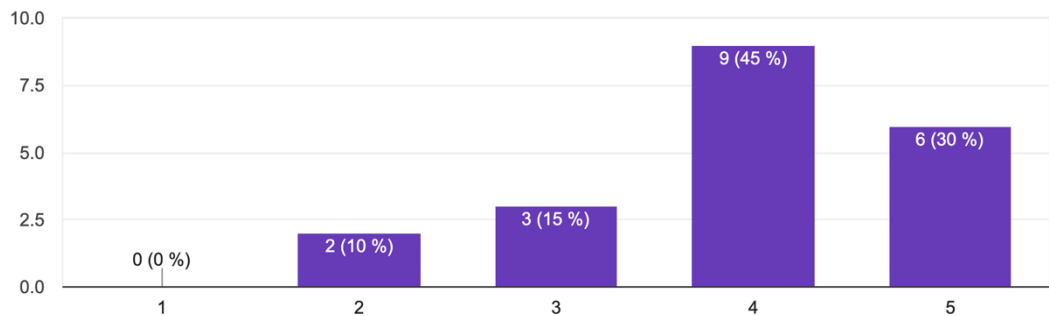


Gráfico 8 Fidelización del Cliente

Elaborado por: El Investigador

Análisis e Interpretación

El 45% de los encuestados considera que una aplicación E-commerce ayuda a fidelizar a los clientes, el 30% está totalmente de acuerdo, el 15% considera que la aplicación podría fidelizar a los clientes y el 10% restante piensa que no necesariamente una aplicación ayudaría a fidelizar a los clientes. Se puede concluir que más del 80% de los encuestados coinciden en que una plataforma E-commerce ayuda a fidelizar a los clientes, esto significa que, luego de una primera compra exitosa, es muy probable que exista una recompra.

Pregunta 8: ¿Le gustaría que una aplicación E-commerce le entregue una factura al momento de finalizar la compra?

OPCIONES	RESPUESTAS	PORCENTAJE
Si	20	100%
No	0	0%

Tabla 9 Facturación

Elaborado por: El Investigador

¿Le gustaría que una aplicación Ecommerce le entregue una factura al momento de finalizar la compra?

20 respuestas

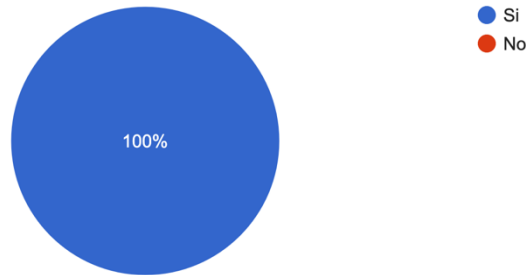


Gráfico 9 Facturación

Elaborado por: El Investigador

Análisis e Interpretación

El 100% de los encuestados coincide en que les gustaría recibir una factura al realizar una compra mediante la aplicación E-commerce, esto está relacionado directamente con que la mayoría de los clientes de MercantilMakamba Cia. Ltda. son compañías las cuales necesitan declarar impuestos de todas las compras que realizan.

Pregunta 9: ¿Le gustaría que la aplicación E-commerce esté disponible los fines de semana o feriados?

OPCIONES	RESPUESTAS	PORCENTAJE
Si	19	95%
No	1	5%

Tabla 10 Disponibilidad

Elaborado por: El Investigador

¿Le gustaría que la aplicación Ecommerce este disponible los fines de semana o feriados?
20 respuestas

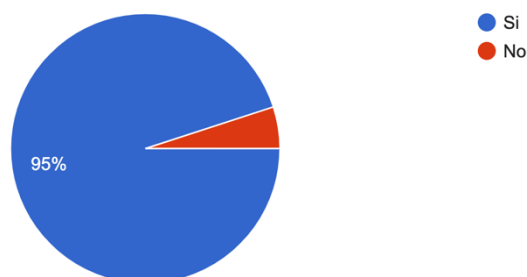


Gráfico 10 Disponibilidad

Elaborado por: El Investigador

Análisis e Interpretación

El 5% de los encuestados considera que la aplicación no debería funcionar en feriados o fines de semana mientras que el 95% de los encuestados coincide en que les gustaría que la aplicación E-commerce esté disponible todo el tiempo. Esto significa que los encuestados están conscientes de las ventajas de las aplicaciones E-commerce sobre el modelo de compraventa tradicional.

Pregunta 10: ¿Cuál es el monto máximo que ha gastado comprando en una aplicación E-commerce?

OPCIONES	RESPUESTAS	PORCENTAJE
Menos de 100	2	10%
Más de 100	3	15%
Menos de 500	4	20%
Más de 500	4	20%
No he comprado	7	35%

Tabla 11 Máximo Gastado

Elaborado por: El Investigador

¿Cuál es el monto máximo que ha gastado comprando en una aplicación Ecommerce?
20 respuestas

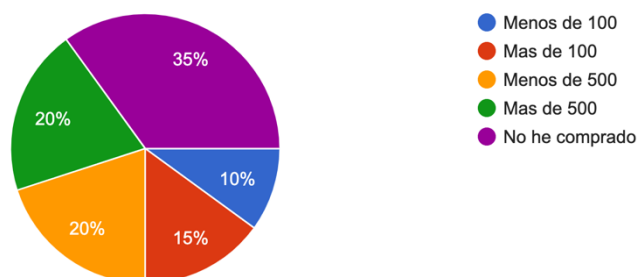


Gráfico 11 Máximo Gastado

Elaborado por: El Investigador

Análisis e Interpretación

El 10% de los encuestados ha gastado menos de 100 dólares comprando en una aplicación E-commerce, el 20% ha gastado más de 500 dólares, el 20% ha gastado menos de 500 dólares, el 15% ha gastado más de 100 dólares y el 35% restante no ha comprado por medio de una aplicación E-commerce. La mayor parte de los encuestados que sí han comprado mediante una aplicación E-commerce han gastado más de 100 dólares por compra, esto significa que, una vez superada la barrera de entrada relacionada con la seguridad de las transacciones, los clientes no presentan problemas en gastar cantidades elevadas de dinero.

Fichas Bibliográficas

Las fichas bibliográficas mostradas en los anexos 1, 2 y 3 señalan conceptos que resultan sumamente importantes para desarrollar aplicaciones móviles E-commerce. Dentro de los conceptos más importantes se encuentran la realización de una matriz para definir los requisitos funcionales y no funcionales de la aplicación, la seguridad de los métodos de pago implementados, la definición de roles de usuarios, la implementación de un carrito de compras y la funcionalidad para que cada producto tenga una pantalla correspondiente con su respectiva información.

Todas las fichas coinciden en que las empresas deberían apuntar hacia un modelo de negocio basado en E-commerce. Así mismo las fichas muestran conceptos que son

ventajas del modelo E-commerce ante el modelo tradicional. También exponen parámetros importantes que deben tener las aplicaciones E-commerce como la alta disponibilidad, seguridad, usabilidad y rapidez.

2.2.4 Procesamiento y Análisis de Datos

De acuerdo con la información recolectada se puede decir que:

- Se debe realizar un diagrama de flujo para definir el proceso de desarrollo de una aplicación E-commerce.
- Se debe definir los módulos que va a tener la aplicación E-commerce, módulos tales como Login, Carrito de compras, Administración.
- La aplicación debe contar con un método de pago seguro y deberá entregar una factura luego de realizado el envío del pedido.
- La aplicación deberá ser desarrollada en Android e iOS para abarcar la mayor cantidad de usuarios posibles.

CAPÍTULO III.- RESULTADOS Y DISCUSIÓN

3.1 Análisis y discusión de resultados

3.1.1 Comparativa de Metodologías de Desarrollo Móvil

XP	Mobile – D	Rational Unified Process
Consta de 5 fases: <ul style="list-style-type: none"> - Planificación - Diseño - Codificación - Pruebas 	Consta de 5 fases: <ul style="list-style-type: none"> - Exploración - Iniciación - Producción - Estabilización - Pruebas de Sistema 	Está compuesta por 4 fases: <ul style="list-style-type: none"> - Comienzo - Elaboración - Construcción - Transición
Es una metodología flexible frente a los contratiempos y cambios en el proyecto.	Se enfoca más en la funcionalidad del software que en la documentación.	Tiene tres perspectivas <ul style="list-style-type: none"> - Dinámica - Estática - Práctica
El cliente participa de forma directa en el proyecto sin la necesidad de firmar contratos.	El cliente participa de forma directa en el proyecto sin la necesidad de firmar contratos.	Es una metodología la cual se basa en asignar roles, tareas y responsabilidades.
Las iteraciones se pueden repetir cada tres a cinco semanas	Es una Metodología flexible ante los contratiempos y cambios del proyecto	La metodología RUP también se basa en la colaboración entre equipos

Tabla 12 Comparativa entre Metodologías

Elaborado por: El Investigador

Como se puede observar en la Tabla 12, la metodología XP y Mobile-D comparten muchas similitudes, ambas entran dentro del grupo de metodologías de desarrollo ágil, ya que ambas cuentan con el cliente final como un miembro más del equipo, también ambas metodologías son flexibles ante los cambios y los contratiempos que se pueden presentar en las fases del desarrollo del software.

3.1.2 Metodología seleccionada

De acuerdo con la Tabla 12, la metodología seleccionada para el desarrollo del presente proyecto es Mobile-D. La razón de que Mobile-D sea la metodología seleccionada es que la duración de las iteraciones se ajusta con el plazo límite del proyecto, así mismo no se requiere de un equipo grande para desarrollar la propuesta de solución.

3.1.3 Arquitectura de la aplicación

La arquitectura seleccionada para el desarrollo de la aplicación fue la arquitectura Model-View-ViewModel mejor conocida como MVVM, la cual es un modelo de diseño de software el cual tiene como principal objetivo separar la interfaz de usuario(vista) de la parte lógica de la aplicación (modelo). Mientras tanto el ViewModel hace las partes de puente entre la vista y el modelo.

Una de las más grandes ventajas de la arquitectura MVVM es que es un modelo recomendado por Google, lo cual significa que existe una alta probabilidad de actualizaciones y mejoras futuras. Otra ventaja es que, al ser recomendado por Google, es compatible con las últimas tecnologías implementadas por Android.

La mayor desventaja del modelo MVVM es que el código es muy difícil de replicar, ya que, incluso cuando dos Fragments comparten similitudes entre sí, siempre existirá la necesidad de realizar algún cambio.

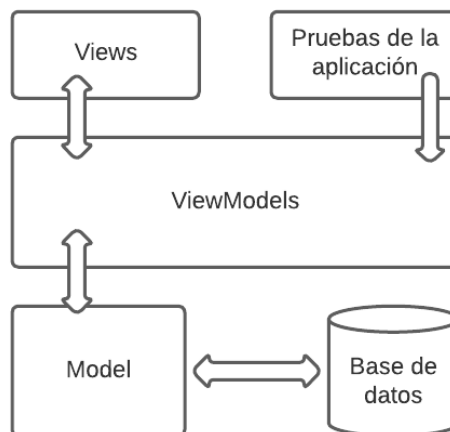


Gráfico 12 Arquitectura MVVM

Elaborado por: El Investigador

3.2 Desarrollo de la propuesta

Planificación de fases

Fase	Iteración	Descripción/Actividades
Exploración	0	<ul style="list-style-type: none"> - Establecer grupos de interés - Determinar el proceso de venta de la empresa - Determinar requisitos funcionales - Determinar requisitos no funcionales - Definición de alcance
Inicialización	1	<ul style="list-style-type: none"> - Preparar el entorno de desarrollo - Diseñar la interfaz - Diseño lógico y físico de la base de datos
Producción	2	<ul style="list-style-type: none"> - Implementar funcionalidad de registro de usuarios - Actualización de storycards - Mejora y pruebas de aceptación
	3	<ul style="list-style-type: none"> - Implementar funcionalidad de inicio de sesión - Actualización de storycards - Mejora y pruebas de aceptación
	4	<ul style="list-style-type: none"> - Implementar funcionalidad de cerrar sesión - Actualización de storycards - Mejora y pruebas de aceptación
	5	<ul style="list-style-type: none"> - Implementar funcionalidad de catálogo - Actualización de storycards - Mejora y pruebas de aceptación
	6	<ul style="list-style-type: none"> - Implementar funcionalidad de carrito de compras - Actualización de storycards - Mejora y pruebas de aceptación
	7	<ul style="list-style-type: none"> - Implementar funcionalidad de agregar productos

		<ul style="list-style-type: none"> - Actualización de storycards - Mejora y pruebas de aceptación
	8	<ul style="list-style-type: none"> - Implementar funcionalidad de quitar productos - Actualización de storycards - Mejora y pruebas de aceptación
	9	<ul style="list-style-type: none"> - Implementar funcionalidad de generar pedidos - Actualización de storycards - Mejora y pruebas de aceptación
	10	<ul style="list-style-type: none"> - Implementar funcionalidad de despachar pedidos - Actualización de storycards - Mejora y pruebas de aceptación
	11	<ul style="list-style-type: none"> - Implementar pasarela de pagos - Actualización de storycards - Mejora y pruebas de aceptación
	12	<ul style="list-style-type: none"> - Implementar funcionalidad de historial de compras - Actualización de storycards - Mejora y pruebas de aceptación
	13	<ul style="list-style-type: none"> - Implementar módulo de promociones - Actualización de storycards - Mejora y pruebas de aceptación
	14	<ul style="list-style-type: none"> - Implementar módulo de gestión de productos - Actualización de storycards - Mejora y pruebas de aceptación
	15	<ul style="list-style-type: none"> - Implementar módulo de gestión de usuarios - Actualización de storycards - Mejora y pruebas de aceptación

Estabilización	16	<ul style="list-style-type: none"> - Retocar funcionalidad de registro de usuarios - Actualización de storycards - Mejora y pruebas de aceptación
	17	<ul style="list-style-type: none"> - Retocar funcionalidad de inicio de sesión - Actualización de storycards - Mejora y pruebas de aceptación
	18	<ul style="list-style-type: none"> - Retocar funcionalidad de cerrar sesión - Actualización de storycards - Mejora y pruebas de aceptación
	19	<ul style="list-style-type: none"> - Retocar funcionalidad de catálogo - Actualización de storycards - Mejora y pruebas de aceptación
	20	<ul style="list-style-type: none"> - Retocar funcionalidad de carrito de compras - Actualización de storycards - Mejora y pruebas de aceptación
	21	<ul style="list-style-type: none"> - Retocar funcionalidad de agregar productos - Actualización de storycards - Mejora y pruebas de aceptación
	22	<ul style="list-style-type: none"> - Retocar funcionalidad de quitar productos - Actualización de storycards - Mejora y pruebas de aceptación
	23	<ul style="list-style-type: none"> - Retocar funcionalidad de generar pedidos - Actualización de storycards - Mejora y pruebas de aceptación
	24	<ul style="list-style-type: none"> - Retocar funcionalidad de despachar pedidos - Actualización de storycards - Mejora y pruebas de aceptación
	25	<ul style="list-style-type: none"> - Retocar pasarela de pagos - Actualización de storycards

		- Mejora y pruebas de aceptación
	26	- Retocar funcionalidad de historial de compras - Actualización de storycards - Mejora y pruebas de aceptación
	27	- Retocar módulo de promociones - Actualización de storycards - Mejora y pruebas de aceptación
	28	- Retocar módulo de gestión de productos - Actualización de storycards - Mejora y pruebas de aceptación
	29	- Retocar módulo de gestión de usuarios - Actualización de storycards - Mejora y pruebas de aceptación
Pruebas de sistema	30	- Se realizan pruebas en la aplicación y se procede con el análisis de resultados

Tabla 13 Planificación de Fases

Elaborado por: El Investigador

3.2.1 Fase I: Exploración

En esta fase se definen los roles de usuarios, se determina el proceso de E-commerce dentro de la empresa, se determinan los requisitos funcionales y no funcionales, se define el alcance y las limitaciones de la solución.

3.2.1.1 Definición de Roles

Rol	Definición
Administrador	- Crea, elimina y modifica categorías de productos - Crea, elimina y modifica productos - Crea, elimina y modifica usuarios
Cliente	- Realiza Pedidos - Cancela Pedidos - Agrega productos al carrito de compras

	- Elimina productos del carrito de compras
Jefe de Bodega	- Recibe Pedidos - Gestiona el envío de pedidos

Tabla 14 Roles de usuarios

Elaborado por: El Investigador

3.2.1.2 Proceso interno de E-commerce

El Gráfico 12 es una representación del proceso de venta interno deseado por la empresa Mercantil Makamba Cia. Ltda. en el cual se pueden observar los siguientes pasos:

Paso 1: Todos los pagos realizados por los clientes serán procesados como pedidos.

Paso 2: El administrador deberá verificar el pago de cada pedido en las cuentas de la empresa.

Paso 3: El administrador realizará la factura en el software de facturación utilizado en la empresa.

Paso 4: El administrador autoriza el despacho de los productos vendidos.

Paso 5: El jefe de bodega recibe la información del pedido que se va a enviar junto con los datos del cliente.

Paso 6: El jefe de bodega revisa la factura para comprobar la lista de productos a despachar.

Paso 7: El jefe de bodega despacha el pedido.

Paso 8: El jefe de bodega envía un correo electrónico al cliente adjuntando la guía del pedido despachado.

Paso 9: El jefe de bodega marca el pedido como despachado dentro de la aplicación.

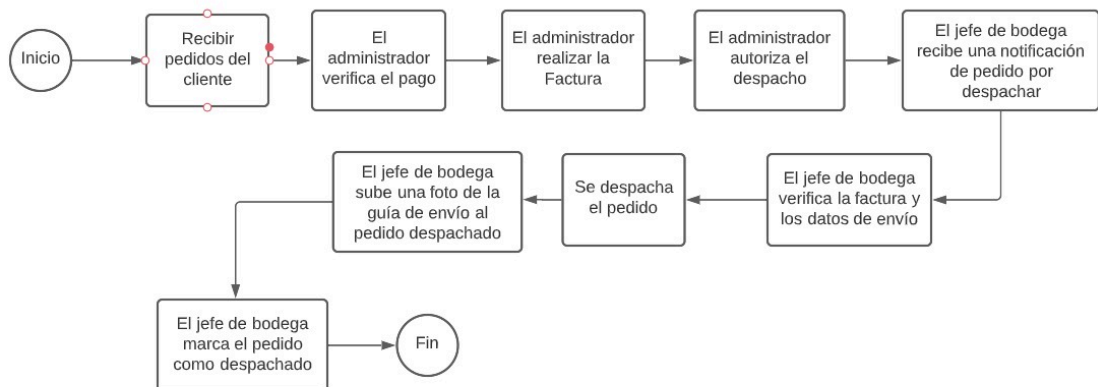


Gráfico 13 Proceso de E-commerce

Elaborado por: El Investigador

3.2.1.3 Requerimientos no funcionales

Requerimiento	Descripción
Usabilidad	<ul style="list-style-type: none"> - Interfaz amigable para el usuario. - Fácil de entender y utilizar.
Eficiencia	<ul style="list-style-type: none"> - La aplicación usa óptimamente los recursos del dispositivo.
Portabilidad	<ul style="list-style-type: none"> - La aplicación está desarrollada para los sistemas operativos móviles más utilizados.
Almacenamiento	<ul style="list-style-type: none"> - La aplicación está desarrollada de tal manera de que no existe código repetido o sin utilizar que utilice mas espacio del necesario.
Estabilidad	<ul style="list-style-type: none"> - La aplicación implementa manejo de errores para aumentar el nivel de estabilidad.
Disponibilidad	<ul style="list-style-type: none"> - La aplicación funciona 24/7. - La base de datos está alojada en un servidor de alta disponibilidad.
Seguridad	<ul style="list-style-type: none"> - La aplicación implementa sistemas de seguridad y autenticación para procesar pagos.

Privacidad	- Los datos de los clientes son almacenados en una base de datos segura alojada en el servidor de la empresa.
Soporte Continuo	- La aplicación contara con un plan actualizaciones y mantenimiento.

Tabla 15 Requerimientos no funcionales

Elaborado por: El Investigador

3.2.1.4 Requerimientos funcionales

Requerimiento	Descripción
Registrar	El cliente podrá crear una cuenta en la aplicación mediante un nombre de usuario o correo electrónico.
Iniciar sesión	El cliente podrá ingresar a la aplicación con una cuenta registrada previamente.
Navegación	El cliente podrá navegar por toda la aplicación.
Agregar productos al carrito	El cliente podrá agregar una lista de productos a su carrito de compras
Eliminar productos del carrito	El cliente podrá eliminar los productos de su carrito de compras
Modificar productos del carrito	El cliente podrá modificar la cantidad de cada producto en su carrito de compras.
Ingresar pedido	El cliente podrá realizar el pedido de los productos de su carrito de compras
Agregar Categorías	El administrador podrá agregar categorías de productos
Agregar productos	El administrador podrá agregar productos
Verificar el pago	El administrador podrá marcar un pedido como pagado.
Despachar producto	El administrador y jefe de Bodega podrán marcar un pedido como despachado.
Agregar Dirección	El cliente podrá agregar una dirección de envío.
Seleccionar método de pago	El cliente podrá seleccionar el método de pago con el que desee pagar.

Pagar Pedido	El cliente podrá pagar su pedido utilizando el método de pago seleccionado.
--------------	---

Tabla 16 Requerimientos Funcionales

Elaborado por: El Investigador

3.2.1.5 Establecimiento del proyecto

En esta etapa de la fase de exploración se describirán los recursos físicos y tecnológicos necesarios para desarrollar la aplicación.

Para el desarrollo de la aplicación móvil se utilizó Android Studio como entorno de desarrollo, El Framework Flutter, el entorno de ejecución para Javascript Node.js y el lenguaje de programación Dart. La base de datos seleccionada fue Mysql y la pasarela de pagos MercadoPago.

Las pruebas del desarrollo se realizaron mediante el módulo de simulación de dispositivos Android incluido en Android Studio y mediante Xcode para simular dispositivos iOS.

3.2.2 Fase II: Inicialización

3.2.2.1 Puesta en Marcha del proyecto

El propósito principal de esta etapa de la metodología es configurar el entorno de desarrollo. Dado que el presente proyecto es llevado a cabo de forma individual se puede afirmar que el equipo cuenta con los conocimientos necesarios para el correcto desarrollo de la solución propuesta.

La configuración necesaria para llevar a cabo el desarrollo está compuesta por:

- Flutter: Framework utilizado para desarrollar la aplicación móvil.
- Android Studio: Entorno de desarrollo integrado.
- Node.js: Framework de Javascript para desarrollar el backend.
- MySQL: Sistema de gestión de base de datos.
- GetX: Gestor de rutas y estados de la aplicación.
- Postman: Plataforma para creación y uso de APIs.
- Xcode: Entorno de desarrollo integrado para iOS.
- Visual Studio Code: Editor de código utilizado para programar backend.

- Dart: Lenguaje de programación de Flutter.

3.2.2.2 Diseño de la base de datos

Modelo Entidad Relación

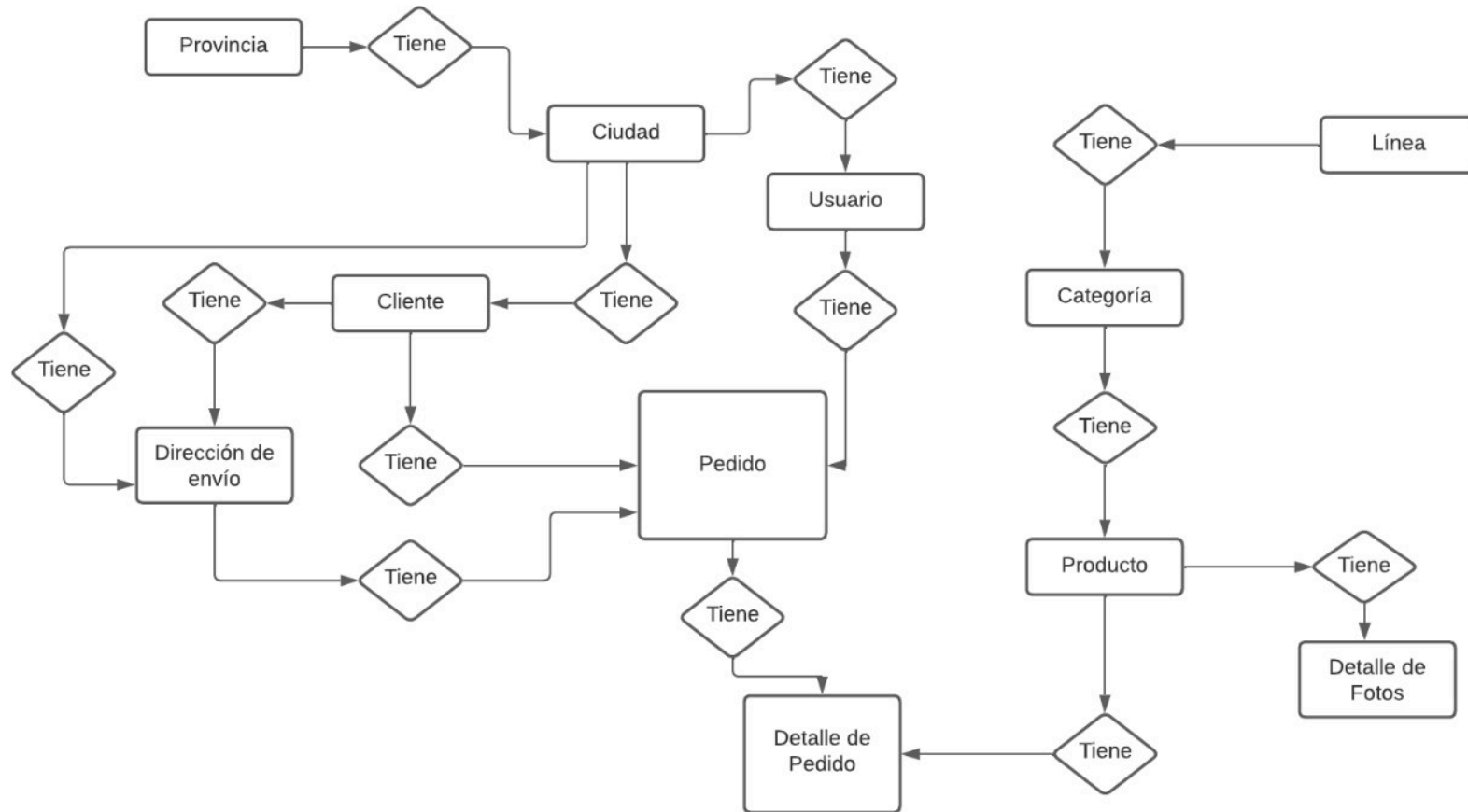


Gráfico 14 Modelo Entidad Relación

Elaborado por: El Investigador

Diseño lógico de la base de datos

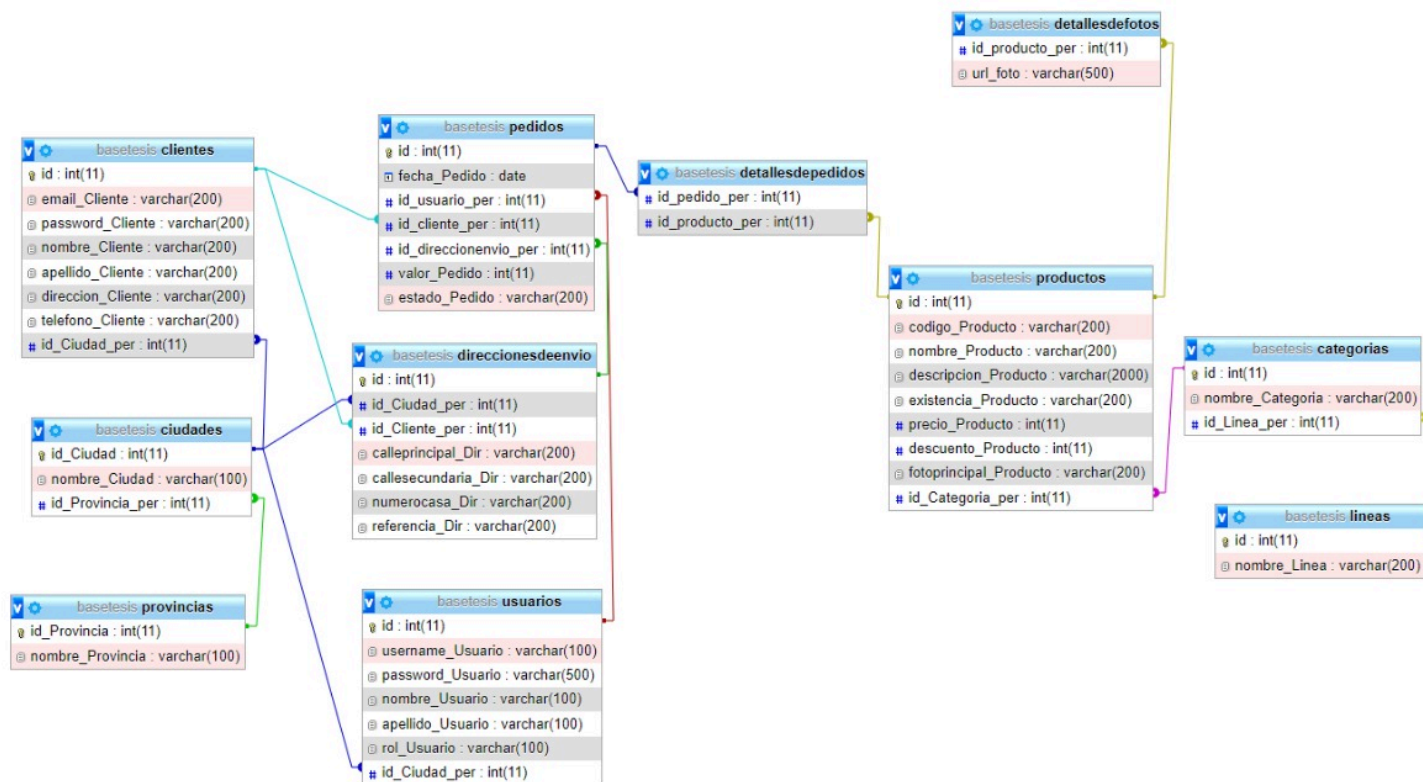


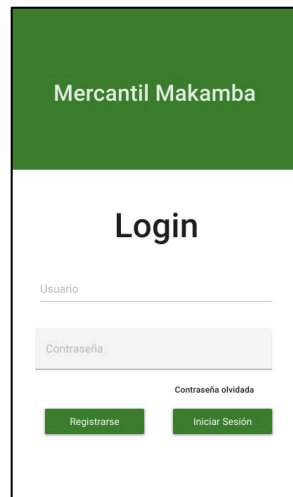
Gráfico 15 Diseño lógico de la base de datos

Elaborado por: El Investigador

3.2.2.3 Diseño de Interfaces

En esta sección se diseñan los prototipos de las interfaces de la aplicación, los cuales pueden variar en la fase de producción de acuerdo con los cambios solicitados por el cliente.

- Prototipo de Pantalla de inicio de sesión



El prototipo de la pantalla de inicio de sesión de Mercantil Makamba presenta un encabezado verde con el nombre de la empresa. El contenido principal es un formulario con un título 'Login', campos de texto para 'Usuario' y 'Contraseña', un enlace 'Contraseña olvidada', y dos botones verdes: 'Registrarse' y 'Iniciar Sesión'.

Gráfico 16 Prototipo de pantalla de inicio de sesión

Elaborado por: El Investigador

En el Gráfico 16 se muestra la pantalla de inicio de sesión, la cual es la primera vista que tiene el usuario al abrir la aplicación. La pantalla está compuesta por un campo de texto para el nombre de usuario, un campo de texto para la contraseña, un botón para iniciar sesión, un botón para registrarse en la aplicación y un enlace para recuperar la contraseña.

- Prototipo de Pantalla de registro



Mercantil Makamba

Registro

Usuario

Contraseña

Re-Contraseña

Correo Electrónico

Cédula/RUC

Nombre Apellido

Celular

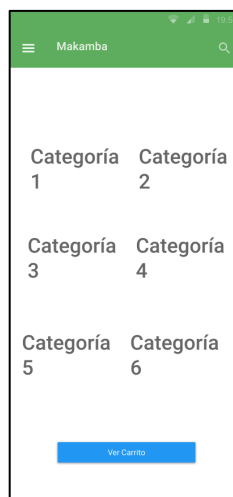
Registrarse

Gráfico 17 Prototipo de pantalla de registro

Elaborado por: El Investigador

En el Gráfico 17 se muestra la pantalla de registro, la cual sirve para crear nuevos clientes que utilizarán la aplicación. La pantalla está compuesta por un campo de texto para el nombre de usuario, un campo de texto para la contraseña, un campo de texto para confirmar la contraseña, un campo para el correo electrónico, un campo para el nombre, un campo para el apellido y un campo para el número de celular.

- Prototipo de Pantalla de categorías



Makamba

Categoría 1	Categoría 2
Categoría 3	Categoría 4
Categoría 5	Categoría 6

Ver Cambio

Gráfico 18 Prototipo de pantalla de categorías

Elaborado por: El Investigador

El Gráfico 18 muestra la vista de categorías de productos, la cual va a desplegar estas categorías a manera de una tabla, cada elemento de la tabla funcionaría como un botón el cual redirigirá al cliente al listado de productos que contiene la categoría pulsada.

- Prototipo de Pantalla de Catálogo de Productos

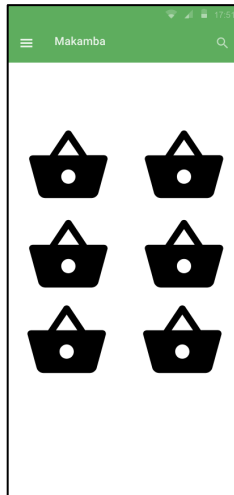


Gráfico 19 Prototipo de catálogo

Elaborado por: El Investigador

El Gráfico 19 muestra la pantalla del catálogo de productos, donde los productos se mostrarán a modo de tabla, los productos serán mostrados a través de su foto principal.

- Prototipo de pantalla de producto



Gráfico 20 Prototipo de Pantalla de producto

Elaborado por: El Investigador

El Gráfico 20 muestra el prototipo de la pantalla de producto, la cual mostrará las fotos del producto en la parte superior. En la parte central se mostrará el nombre del producto, la descripción y el precio unitario. En la parte inferior se mostrarán los botones para aumentar y disminuir la cantidad del producto junto al botón para agregar al carrito de compras.

- Prototipo de Pantalla de Perfil



Gráfico 21 Prototipo de pantalla de perfil

Elaborado por: El Investigador

El gráfico 21 muestra la pantalla del perfil del usuario, en la cual el usuario podrá visualizar las direcciones agregadas a la aplicación, cambiar la contraseña, abrir el historial de compras y adicionalmente tendrá un botón para cerrar sesión.

- Prototipo de carrito de compras



Gráfico 22 Prototipo de carrito de compras

Elaborado por: El Investigador

El gráfico 22 muestra el prototipo del carrito de compras, el cual mostrará un listado de los productos agregados previamente junto a la cantidad. La pantalla tendrá botones para aumentar y disminuir la cantidad de producto y adicional un botón para eliminar el producto del carrito.

- Prototipo de pantalla de pedidos



Gráfico 23 Prototipo de pantalla de pedidos

Elaborado por: El Investigador

El Gráfico 23 muestra el listado de pedidos, los cuales serán clasificados dependiendo de su estado, el cual puede ser "Pagado", "Despachado", "Enviado". Los productos serán mostrados a través de una lista de tarjetas las cuales contendrán el número de pedido y el total.

3.2.2.4 Diagramas de casos de uso

- Casos de uso de clientes

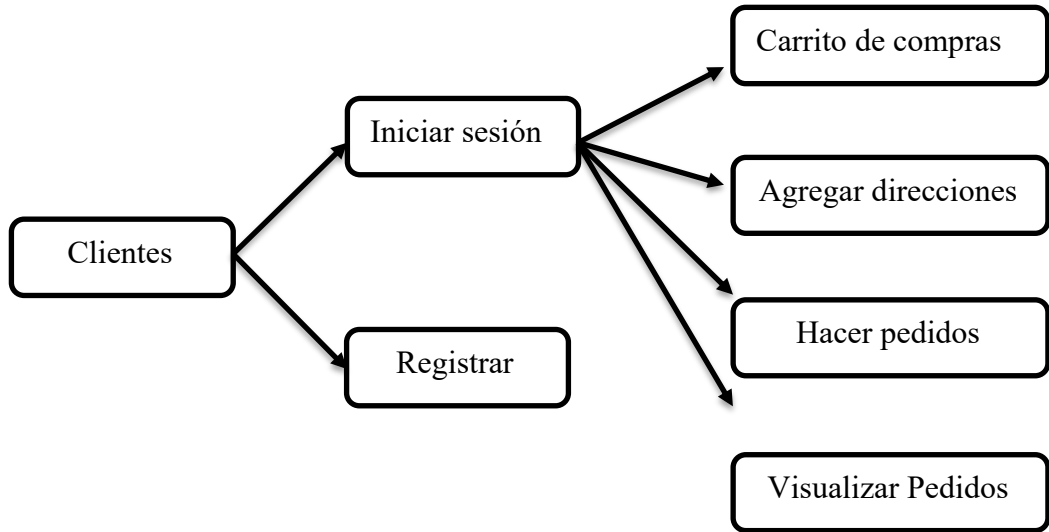


Gráfico 24 Casos de uso de clientes

Elaborado por: El Investigador

- Casos de uso Administrador

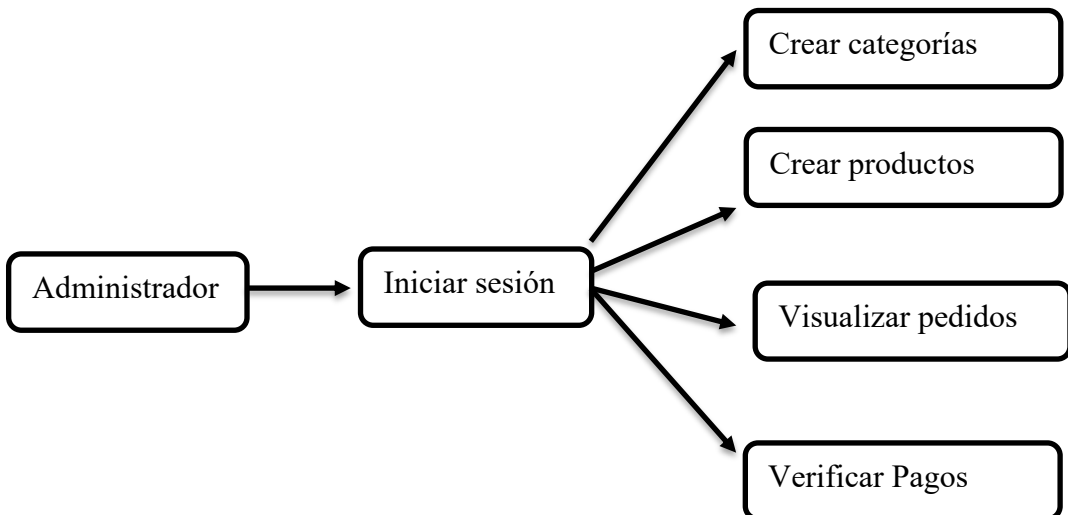


Gráfico 25 Casos de uso del administrador

Elaborado por: El investigador

- Casos de uso Bodega

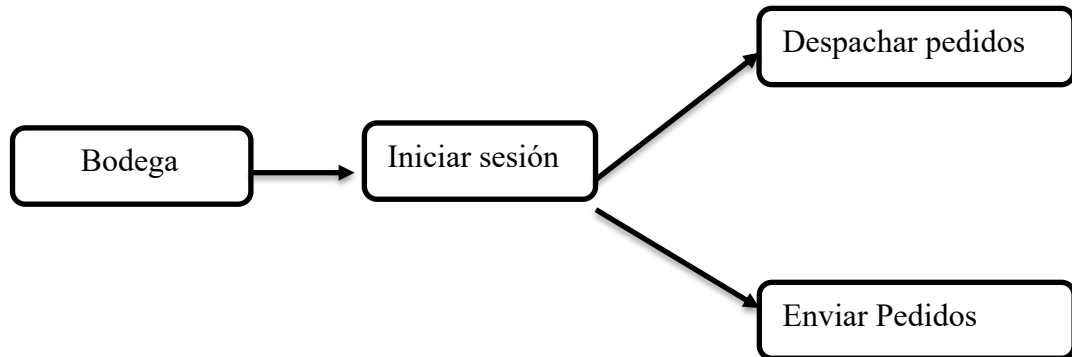


Gráfico 26 Casos de uso de bodega

Elaborado por: El Investigador

3.2.3 Fase III Producción

3.2.3.1 Backend

Para desarrollar el Backend se utilizó el framework Node.js, el cual permite comunicar la base de datos en MySQL con el frontend desarrollado en Flutter.

La estructura del backend está conformada por modelos, controladores y rutas. Esta estructura permite que la aplicación sea escalable y da mucha facilidad de crear nuevos módulos o de realizar correcciones en el código.

- Modelos

Los modelos son las entidades de la aplicación de Node.js, estos contienen las sentencias SQL con las cuales se accederá a la base de datos. Los modelos implementados en la aplicación son los siguientes:

- Categoria.js
- Detalle_pedido.js
- Direccion.js
- Pedido.js

- Usuario.js
- Producto.js
- Rol.js

- Controladores

Los controladores son los encargados de comunicar los modelos con las rutas. Los controladores también ejecutan las sentencias SQL que están en los modelos, estos devuelven un mensaje de éxito o de error. Si la petición resulta en éxito puede devolver un objeto de tipo JSON. Los controladores implementados en la aplicación son los siguientes:

- CategoriasController.js
- UsuariosController.js
- PedidosController.js
- ProductosController.js
- DireccionesController.js

- Rutas

Las rutas son la comunicación entre el frontend y el backend de la aplicación. Las rutas implementadas en la aplicación son las siguientes:

- CategoriaRoutes.js
- UsuarioRoutes.js
- DireccionRoutes.js
- UsuarioRoutes.js
- PedidoRoutes.js

3.2.3.2 Frontend

Para desarrollar el frontend de la aplicación se utilizó Android Studio como entorno de desarrollo y Flutter como framework. La estructura del frontend está conformada por modelos, paginas, controladores y servicios.

- Modelos

Los modelos en el frontend son las entidades de la aplicación, las cuales contienen atributos y métodos pertenecientes a cada entidad. Los métodos comunes de todas las entidades son métodos para convertir un objeto de tipo JSON en un objeto de la entidad y viceversa, esto permite leer y enviar las entidades a la base de datos. La lista de los modelos implementados en la aplicación son los siguientes:

- Categoria.dart
- Direccion.dart
- Pedido.dart
- Usuario.dart
- Rol.dart
- Producto.dart

- Páginas

Las páginas en el frontend son las pantallas que podrán visualizar los usuarios de la aplicación. Están conformadas por Widgets observables y estáticos. La lista de páginas implementadas en la aplicación son las siguientes.

- AlmacenCategoriesCreatePage.dart
- AlmacenHomePage.dart
- AlmacenOrdersPage.dart
- AlmacenProductsCreatePage.dart
- BodegaHomePage.dart
- BodegaOrdersPage.dart
- ClientAddressListPage.dart
- ClientAddressCreatePage.dart
- ClientAddressMapPage.dart
- ClientHomePage.dart
- ClientOrdersCreatePage.dart
- ClientOrdersListPage.dart
- ClientOrdersDetailPage.dart
- ClientPagosCreatePage.dart
- ClientProductsListPage.dart

- ClientProductsDetailPage.dart
- ClientProfileInforPage.dart
- ClientProfileUpdatePage.dart

- Controladores

Los controladores en el frontend contienen los métodos utilizados por las páginas, en resumen, los controladores contienen la lógica de programación que será utilizada por la página que le corresponde. La lista de controladores implementados en la aplicación son los siguientes:

- AlmacenCategoriesCreateController.dart
- AlmacenHomeController.dart
- AlmacenOrdersController.dart
- AlmacenProductsCreateController.dart
- BodegaHomeController.dart
- BodegaOrdersController.dart
- ClientAddressListController.dart
- ClientAddressCreateController.dart
- ClientAddressMapController.dart
- ClientHomeController.dart
- ClientOrdersCreateController.dart
- ClientOrdersListController.dart
- ClientOrdersDetailController.dart
- ClientPagosCreateController.dart
- ClientProductsListController.dart
- ClientProductsDetailController.dart
- ClientProfileInforController.dart
- ClientProfileUpdateController.dart

- Servicios

Los servicios son los encargados de comunicar el frontend con las rutas del backend de la aplicación. La lista de servicios implementados en la aplicación son los siguientes:

- CategoríasProvider.dart
- DireccionesProvider.dart
- PedidosProvider.dart
- ProductosProvider.dart
- UsuariosProvider.dart

3.2.3.3 StoryCards

El principal objetivo de las storycards es construir un registro de todas las actividades con sus características tales como la descripción, la dificultad, el usuario que las ejecuta y las fechas de implementación.

Storycard – Registrar

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
1	Nuevo	3	3	3	3	Alta
DESCRIPCIÓN: Registrar un nuevo usuario						
FECHA	ESTADO		COMENTARIO			
20/12/2022	Definido		Sin Comentarios			
20/12/2022	Implementado		Sin Comentarios			
20/12/2022	Ejecutado		Sin Comentarios			
21/12/2022	Verificado		Sin Comentarios			

Tabla 17 Storycard - Registro de usuario

Elaborado por: El Investigador

Para implementar la funcionalidad de registro se creó un formulario en Flutter el cual contiene los datos del usuario a registrar, en el Backend se utilizó bcrypt.js para encriptar la contraseña. Mediante una petición HTTP de tipo post se accede a la ruta que contiene el método para crear el usuario y luego de crearse el usuario se crea una fila en la tabla de detalle_rols donde se le asigna el rol de cliente por defecto.

```

Usuario.create(usuario, (err,data) => {
  if (err){
    return res.status(501).json({
      success: false,
      message: 'Hubo un error con el registro del usuario',
      error: err
    })
  }

  usuario.id_usu = `${data}`;
  const token = jwt.sign({id_usu:usuario.id_usu,email_usu:usuario.email_usu},keys.secretOrKey, {});
  usuario.session_token = `JWT ${token}`;
  Rol.create(usuario.id_usu,3, (err,data) => {
    if (err){
      return res.status(501).json({
        success: false,
        message: 'Hubo un error con el registro del rol',
        error: err
      })
    }
  })

  return res.status(201).json({
    success: true,
    message: "El registro fue exitoso",
    data: usuario
  })
})

```

Gráfico 27 Backend de registro de usuarios

Elaborado por: El Investigador

Storycard – Inicio de Sesión

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
2	Nuevo	3	3	3	3	Alta
DESCRIPCIÓN: Inicio de sesión						
FECHA	ESTADO		COMENTARIO			
21/12/2022	Definido		Sin Comentarios			
21/12/2022	Implementado		Sin Comentarios			
22/12/2022	Ejecutado		Sin Comentarios			
22/12/2022	Verificado		Sin Comentarios			

Tabla 18 Storycard - Inicio de Sesión

Elaborado por: El Investigador

Para implementar la funcionalidad de inicio de sesión se creó un formulario en la aplicación de Flutter el cual recibe el email y contraseña para luego llamar a la ruta de usuario mediante una petición HTTP de tipo post la cual devuelve un token de sesión para verificar que el usuario está dentro de la aplicación.


```

login(req,res){
  // puede que aqui haya un error
  const email = req.body.email_usu;
  const password = req.body.contra_usu;
  //puede que aqui termine el error

  Usuario.FindByEmail(email, async (err,myUser) => {
    if (err){
      return res.status(501).json({
        success: false,
        message: 'No se encontro al usuario',
        error: err
      })
    }
    if(!myUser){
      return res.status(401).json({
        success: false,
        message: 'El email no fue encontrado',
      })
    }
    const isPasswordValid = await bcrypt.compare(password,myUser.contra_usu);
    if(isPasswordValid){
      const token = jwt.sign({id_usu:myUser.id_usu,email_usu:myUser.email_usu},keys.secret0rKey, {});
      const data = {
        id_usu: `${myUser.id_usu}`,
        nombre_usu: myUser.nombre_usu,
        apellido_usu: myUser.apellido_usu,
        email_usu: myUser.email_usu,
        telefono_usu : myUser.telefono_usu,
        foto_usu : myUser.foto_usu,
        session_token: `JWT ${token}`,
        roles: JSON.parse(myUser.roles)
      }
      return res.status(201).json({
        success: true,
        message: "La autenticación fue exitosa",
        data: data
      })
    }
  })
}

```

Gráfico 28 Backend de Inicio de Sesión

Elaborado por: El Investigador

Storycard – Subir foto al perfil del usuario

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
3	Nuevo	4	4	3	3	Media
DESCRIPCIÓN: Permitir el uso de la cámara y la galería para subir una imagen de usuario						
FECHA	ESTADO		COMENTARIO			
27/12/2022	Definido		Sin Comentarios			
27/12/2022	Implementado		Sin Comentarios			
28/12/2022	Ejecutado		Sin Comentarios			
28/12/2022	Verificado		Sin Comentarios			

Tabla 19 Storycard - Subir archivos multimedia

Elaborado por: El Investigador

Para que la aplicación acceda a la cámara y galería del dispositivo se utilizaron librerías especializadas de Flutter. Para guardar los archivos multimedia se utilizó Firebase el cual se combina con MySQL con el fin de que Firebase almacene el archivo y MySQL almacene el enlace al archivo.

```

async registerWithImage(req,res){

    const usuario = JSON.parse(req.body.usuario); //capturar datos que envia el cliente

    const files = req.files;

    if(files.length > 0){
        const path = `image_${Date.now()}`;
        const url = await storage(files[0],path);

        if(url != undefined && url != null){
            usuario.foto_usu = url;
        }
    }

    Usuario.create(usuario, (err,data) => {
        if (err){
            return res.status(501).json({
                success: false,
                message: 'Hubo un error con el registro del usuario',
                error: err
            })
        }

        usuario.id_usu = `${data}`;
        const token = jwt.sign({id_usu:usuario.id_usu,email_usu:usuario.email_usu},keys.secretOrKey, {});
        usuario.session_token = `JWT ${token}`;
        Rol.create(usuario.id_usu,3, (err,data) => {
            if (err){
                return res.status(501).json({
                    success: false,
                    message: 'Hubo un error con el registro del rol',
                    error: err
                })
            }
        })

        return res.status(201).json({
            success: true,
            message: "El registro fue exitoso",
            data: usuario
        })
    })
}

```

Gráfico 29 Backend para subir archivos multimedia

Elaborado por: El Investigador

Storycard – Catálogo de Productos

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
4	Nuevo	4	4	4	4	Alta
DESCRIPCIÓN: Los usuarios con rol de Cliente pueden visualizar el catálogo de productos						
FECHA	ESTADO			COMENTARIO		

03/01/2023	Definido	Sin Comentarios
03/01/2023	Implementado	Sin Comentarios
04/01/2023	Ejecutado	Sin Comentarios
04/01/2023	Verificado	Sin Comentarios

Tabla 20 Storycard – Catálogo de los Productos

Elaborado por: El Investigador

Los usuarios con el rol de Cliente pueden visualizar el catálogo de productos, los productos llegan luego de hacer una petición HTTP a la ruta del servidor la cual devuelve los productos por categoría. En la aplicación de Flutter los productos se muestran mediante un TabBarView el cual contiene las categorías y cada categoría muestra los productos correspondientes.

```

findByCategory(req, res){
  const id_categoria = req.params.id_categoria;

  Producto.findByCategory(id_categoria, (err, data) => {
    if (err){
      return res.status(501).json({
        success: false,
        message: 'Hubo un error al listar los productos',
        error: err
      })
    }
    return res.status(201).json(data);
  })
},

```

Gráfico 30 Backend para el catálogo de productos

Elaborado por: El Investigador

Storycard – Agregar Producto al Carrito de Compras

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
5	Nuevo	4	4	4	4	Alta

DESCRIPCIÓN: Agregar productos al carrito de compras		
FECHA	ESTADO	COMENTARIO
04/01/2023	Definido	Sin Comentarios
05/01/2023	Implementado	Sin Comentarios
05/01/2023	Ejecutado	Sin Comentarios
05/01/2023	Verificado	Sin Comentarios

Tabla 21 Storycard - Agregar productos al carrito de compras

Elaborado por: El Investigador

Para agregar productos al carrito de compra se utilizaron librerías para acceder al almacenamiento interno del dispositivo, el cual almacena los productos en la sesión del usuario de tal manera que el carrito guarda los productos, aunque la aplicación se cierre. En la aplicación de Flutter se muestran los productos del carrito junto con su respectivo precio y cantidad.

```

void addtoBag(Producto producto, var precio, var counter) {
  if (counter.value > 0) {
    //Validar si el producto ya fue agregado con GetStorage a la sesion del dispositivo
    int index =
      ProductosSeleccionados.indexWhere((p) => p.idPro == producto.idPro);
    if (index == -1) {
      //producto no agregado
      if (producto.cantidadPro == null) {
        if(counter.value > 0){
          producto.cantidadPro = counter.value;
        }else{
          producto.cantidadPro = 1;
        }
      }
      ProductosSeleccionados.add(producto);
    } else {
      // el producto ya ha sido agregado en el storage
      ProductosSeleccionados[index].cantidadPro = counter.value;
    }
    GetStorage().write('shopping_bag', ProductosSeleccionados);
    Fluttertoast.showToast(msg: 'Producto Agregado');
    Get.back();
    ProductosSeleccionados.forEach((p) {
      productsListController.items.value = productsListController.items.value + p.cantidadPro!;
    });
  } else {
    Fluttertoast.showToast(msg: 'Agrega por lo menos un producto');
  }
}

```

Gráfico 31 Frontend para agregar productos al carrito de compras

Elaborado por: El Investigador

Storycard – Carrito de compras

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
6	Nuevo	3	3	4	4	Media
DESCRIPCIÓN: Aumentar, disminuir y eliminar productos del carrito						
FECHA	ESTADO		COMENTARIO			
07/01/2023	Definido		Sin Comentarios			
07/01/2023	Implementado		Sin Comentarios			
08/01/2023	Ejecutado		Sin Comentarios			
08/01/2023	Verificado		Sin Comentarios			

Tabla 22 Storycard - Carrito de Compras

Elaborado por: El Investigador

Para utilizar el carrito de compras la aplicación accede constantemente al almacenamiento interno, esto se hace cada que el usuario hace algún cambio en el carrito. Los productos agregados al carrito también se almacenan en sesión. Dentro del carrito de compras se puede modificar la cantidad de cada producto agregado, así como eliminar al producto del carrito.

```

void checkIfProductWasAdded(Producto producto, var precio, var counter){
    precio.value = producto.precioPro ?? 0.0;
    if (GetStorage().read('shopping_bag') != null) {
        if (GetStorage().read('shopping_bag') is List<Producto>) {
            ProductosSeleccionados = GetStorage().read('shopping_bag');
        } else {
            ProductosSeleccionados = Producto.fromJsonList(GetStorage().read('shopping_bag'));
        }
        int index = ProductosSeleccionados.indexWhere((p) => p.idPro == producto?.idPro);
        if(index != -1){ // el producto ya fue agregado
            counter.value = ProductosSeleccionados[index].cantidadPro ?? 0;
            precio.value = producto.precioPro! * counter.value;
        }
    }
}

void addItem(Producto producto, var precio, var counter) {
    print('${producto.stockPro} hola');
    if (counter.value < double.parse(producto.stockPro.toString())) {
        counter.value++;
        precio.value = producto.precioPro! * counter.value;
    } else {
        Get.snackbar('Stock!!', 'Llegaste al stock máximo');
    }
}

void minItem(Producto producto, var precio, var counter) {
    print('hola');
    if (counter.value > 0) {
        counter.value--;
        precio.value = producto.precioPro! * counter.value;
    } else {}
}

```

Gráfico 32 Frontend de carrito de compras

Elaborado por: El Investigador

Storycard – Direcciones

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
7	Nuevo	4	4	4	4	Media
DESCRIPCIÓN: Agregar direcciones de envío						
FECHA	ESTADO		COMENTARIO			
12/01/2023	Definido		Sin Comentarios			
12/01/2023	Implementado		Sin Comentarios			
13/01/2023	Ejecutado		Sin Comentarios			
13/01/2023	Verificado		Sin Comentarios			

Tabla 23 Storycard - Agregar direcciones de envío

Elaborado por: El Investigador

El cliente debe agregar direcciones antes de realizar un pedido, de tal manera que utilizamos una petición HTTP de tipo post para guardar las direcciones creadas en la base de datos, para agregar una nueva dirección el cliente debe tener activado los servicios de geolocalización de su dispositivo. Las direcciones también guardan un parámetro para latitud y otro para longitud, de tal manera que la dirección es lo más exacta posible. La aplicación también hace una validación para que las direcciones solo pertenezcan a Ecuador.

```

create(req, res){
  const direccion = req.body;
  Direccion.create(direccion, (err, id_dir) => {
    if (err){
      return res.status(501).json({
        success: false,
        message: 'Hubo un error con el registro de la dirección',
        error: err
      })
    }
    return res.status(201).json({
      success: true,
      message: "La dirección se creo correctamente",
      data: `${id_dir}` //id de la nueva categoria
    })
  })
},

findByUser(req, res){
  const id_usu = req.params.id_usu;
  Direccion.findByUser(id_usu, (err, data)=>{
    if (err){
      return res.status(501).json({
        success: false,
        message: 'Hubo un error al obtener las direcciones',
        error: err
      })
    }
    return res.status(201).json(data);
  })
}

```

Gráfico 33 Backend para agregar direcciones de envío

Elaborado por: El Investigador

Storycard – Geolocalización

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
8	Nuevo	4	4	4	4	Alta
DESCRIPCIÓN: Geolocalización del dispositivo						
FECHA	ESTADO		COMENTARIO			
15/01/2023	Definido		Sin Comentarios			
15/01/2023	Implementado		Sin Comentarios			
16/01/2023	Ejecutado		Sin Comentarios			
16/01/2023	Verificado		Sin Comentarios			

Tabla 24 Storycard - Geolocalización del dispositivo

Elaborado por: El Investigador

Para acceder a los servicios de GPS del dispositivo se utilizaron librerías de Flutter, estas librerías brindan una gran cantidad de información de la localización del dispositivo tales como país, provincia, ciudad, calle, latitud y longitud. Para seleccionar un punto de referencia se debe implementar un Widget en el centro de la pantalla con el cual se navega por el mapa.

```

CameraPosition initialPosition = CameraPosition(
  target: LatLng(-0.2943517, -78.383088),
  zoom: 14); // CameraPosition
Completer <GoogleMapController> mapController = Completer();
Position? position;
var Ciudad = '';obs;
var Provincia = '';obs;
String otrocoso2 = '';
String? Pais;
LatLng? addressLatLng;
var addressName = '';obs;
ClientAddressMapController(){
  checkGPS();
}
Future setLocationDraggableInfo() async{
  double lat = initialPosition.target.latitude;
  double lng = initialPosition.target.longitude;
  List<Placemark> address = await placemarkFromCoordinates(lat, lng);
  if(address.isNotEmpty){
    String pais = address[0].country ?? '';
    String direction = address[0].thoroughfare ?? '';
    String street = address[0].subThoroughfare ?? '';
    String city = address[0].locality ?? '';
    String department = address[0].administrativeArea ?? '';
    if(street == ''){
      street = 'S/N';
    }
    Pais = pais;
    Ciudad.value = '$city';
    Provincia.value = '$department';
    addressName.value = '$direction, $street';
    addressLatLng = LatLng(lat, lng);
  }
}

```

Gráfico 34 Frontend de la geolocalización del envío

Elaborado por: El Investigador

Storycard – Realizar pedidos

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
9	Nuevo	4	4	4	4	Alta
DESCRIPCIÓN: Realizar pedidos						
FECHA	ESTADO		COMENTARIO			
27/12/2022	Definido		Sin Comentarios			
27/12/2022	Implementado		Sin Comentarios			
27/12/2022	Ejecutado		Sin Comentarios			
28/12/2022	Verificado		Sin Comentarios			

Tabla 25 Storycard - Realizar pedidos

Elaborado por: El Investigador

Luego de tener productos agregados en el carrito y una dirección de envío seleccionada el usuario puede realizar el pedido. Para que el pedido sea aceptado el usuario tendrá que pagar el valor total mediante la pasarela de pagos implementada en la aplicación.

```

void crearPedido() async{
  Direccion dir = Direccion.fromJson(GetStorage().read('direccion'));
  List<Producto> listaProductos = [];
  if (GetStorage().read('shopping_bag') is List<Producto>) {
    listaProductos = GetStorage().read('shopping_bag');
  }else{
    listaProductos = Producto.fromJsonList(GetStorage().read('shopping_bag'));
  }

  Pedido pedido = Pedido(
    idCliente: usuario.idUsu,
    idDir: dir.idDir,
    productos: listaProductos
  );
  ResponseApi responseApi = await pedidosProvider.create(pedido);
  if(responseApi.success == true){
    GetStorage().remove('shopping_bag');
    Fluttertoast.showToast(msg: '${responseApi.message ?? ''}', toastLength: Toast.LENGTH_LONG);
    Get.offNamedUntil('/client/home', (route) => false);
  }
}

```

Gráfico 35 Frontend para crear pedidos

Elaborado por: El Investigador

Storycard – Búsqueda de Producto

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
10	Nuevo	3	3	3	3	Media
DESCRIPCIÓN: Campo de texto para buscar el producto						
FECHA	ESTADO		COMENTARIO			
18/12/2022	Definido		Sin Comentarios			
18/12/2022	Implementado		Sin Comentarios			
19/12/2022	Ejecutado		Sin Comentarios			
19/12/2022	Verificado		Sin Comentarios			

Tabla 26 Storycard - Búsqueda de Productos

Elaborado por: El Investigador

Para implementar la búsqueda de productos se creó un campo de texto en la pantalla de productos, el cual realiza peticiones HTTP de tipo get al servidor pasado 1 segundo luego de terminar de escribir, esto se hace para que el servidor no se sobrecargue de peticiones por cada letra escrita por el usuario.

```

findByNombre(req, res){
  const id_categoria = req.params.id_categoria;
  const nombre_pro = req.params.nombre_pro;

  Producto.findByNombre(id_categoria, nombre_pro, (err, data) => {

    if (err){
      return res.status(501).json({
        success: false,
        message: 'Hubo un error al listar los productos',
        error: err
      })
    }
    return res.status(201).json(data);
  })
}

```

Gráfico 36 Backend para buscar productos por nombre

Elaborado por: El Investigador

Storycard – Cerrar Sesión

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
11	Nuevo	3	3	3	3	Media
DESCRIPCIÓN: El cierre de sesión del usuario.						
FECHA	ESTADO		COMENTARIO			
26/12/2022	Definido		Sin Comentarios			
26/12/2022	Implementado		Sin Comentarios			
27/12/2022	Ejecutado		Sin Comentarios			
27/12/2022	Verificado		Sin Comentarios			

Tabla 27 Storycard - Cerrar Sesión

Elaborado por: El Investigador

Al momento en que un usuario cierra sesión, se realiza una limpieza del almacenamiento interno del dispositivo, adicionalmente se elimina el token de sesión del usuario, de tal manera que no puede enviar peticiones al servidor, el usuario será redirigido a la pantalla de inicio de sesión y registro.

```

Usuario usuario = Usuario.fromJson(GetStorage().read('usuario'));
void signOut(){
  GetStorage().remove('usuario');
  GetStorage().remove('shopping_bag');
  GetStorage().remove('direccion');
  Get.offNamedUntil('/', (route) => false);
}

```

Gráfico 37 Frontend para Cerrar Sesión

Elaborado por: El Investigador

Storycard – Realizar el Pago

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
12	Nuevo	4	4	4	4	Alta

DESCRIPCIÓN: Pagar el pedido		
FECHA	ESTADO	COMENTARIO
23/12/2022	Definido	Sin Comentarios
23/12/2022	Implementado	Sin Comentarios
24/12/2022	Ejecutado	Sin Comentarios
24/12/2022	Verificado	Sin Comentarios

Tabla 28 Storycard - Realizar el Pago

Elaborado por: El Investigador

Dado que el método de pago seleccionado para le presente proyecto fue PayPal, para que el cliente complete su pedido la aplicación desplegara un formulario para ingresar sus credenciales de PayPal. Luego de seleccionar la tarjeta designada para el pago se envía una petición a los servidores de PayPal y estos retornaran una respuesta. Si el contenido de la respuesta es “success” significa que el pago fue exitoso.

```
Widget paypalProcces(String total, String subtotal, String envio, Usuario usuario){
  return UsePaypal(
    sandboxMode: true,
    clientId:
      "AdYypV5p1fkJqCegN42EBfQBztpnT3q3_vWJNnUvp0I3PmNIBgmsF3yziG0bjkq5bCBgGYNWZGwPJo27",
    secretKey:
      "EInCngidUCwusy2y8R6WzqVp07PIkN931EP-T8I6DmZce-RBpZy6VL7fVN2fz8W4GHEft5kMS-YHnkVa",
    returnUrl: "https://samplesite.com/return",
    //returnURL: "com.example.pruebita://paypalpay",
    cancelURL: "https://samplesite.com/cancel",
    transactions: [
      {
        "amount": {
          "total": '$total',
          "currency": "USD",
          "details": {
            "subtotal": '$subtotal',
            "shipping": '$envio',
            "shipping_discount": 0
          }
        }
      },
      "description":
        "The payment transaction description.",
    ]
  );
}
```

Gráfico 38 Frontend para realizar el pago

Elaborado por: El Investigador

Storycard – Crear Nuevos Productos (Administrador)

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
13	Nuevo	3	3	3	3	Media
DESCRIPCIÓN: Agregar nuevos Productos a la aplicación						
FECHA	ESTADO		COMENTARIO			
28/12/2022	Definido		Sin Comentarios			
28/12/2022	Implementado		Sin Comentarios			
29/12/2022	Ejecutado		Sin Comentarios			
30/12/2022	Verificado		Sin Comentarios			

Tabla 29 Storycard - Crear Productos

Elaborado por: El Investigador

Para agregar nuevos productos al catálogo se necesita tener el rol de “Almacén”, el producto será relacionado a una categoría creada previamente y se enviará una petición HTTP de tipo post al servidor para insertar el nuevo producto en la base de datos. Adicional a tener el rol de “Almacén” el usuario necesita un token de sesión activo para poder realizar la petición, esto ayuda a la seguridad de la aplicación.

```

Producto.create(producto, (err, id_pro) => {
  if (err){
    return res.status(501).json({
      success: false,
      message: 'Hubo un error con el registro del producto',
      error: err
    })
  }
  producto.id_pro = id_pro;
  const start = async () => {
    await asyncForEach(files, async (file) => {
      const path = `image_${Date.now()}`;
      const url = await storage(file, path);

      if(url != undefined && url != null){
        if(inserts === 0){
          producto.foto1_pro = url;
        }else if (inserts === 1){
          producto.foto2_pro = url;
        }
      }
    })
  }
  await Producto.update(producto, (err, data) =>{
    if (err){
      return res.status(501).json({
        success: false,
        message: 'Hubo un error con el registro del producto',
        error: err
      })
    }
    inserts = inserts + 1;
    if(inserts == files.length){//termino de almacenar las imagenes
      return res.status(201).json({
        success: true,

```

Gráfico 39 Backend para crear productos

Elaborado por: El Investigador

Storycard – Crear Nueva Categoría (administrador)

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
14	Nuevo	3	3	3	3	Alta
DESCRIPCIÓN: Agregar nueva categoría						
FECHA	ESTADO		COMENTARIO			
02/01/2023	Definido		Sin Comentarios			
02/01/2023	Implementado		Sin Comentarios			
03/01/2023	Ejecutado		Sin Comentarios			
04/01/2023	Verificado		Sin Comentarios			

Tabla 30 Storycard - Crear nueva categoría

Elaborado por: El Investigador

Para crear una nueva categoría es necesario tener el rol de “Almacén” y llenar el formulario de la aplicación en Flutter. Luego de esto la aplicación enviará una petición HHTP de tipo post al servidor el cual devolverá una respuesta 201 en caso de tener éxito.

```

create(req, res){
  const categoria = req.body;
  Categoria.create(categoria, (err, id_cat) => {
    if (err){
      return res.status(501).json({
        success: false,
        message: 'Hubo un error con el registro del usuario',
        error: err
      })
    }
    return res.status(201).json({
      success: true,
      message: "La categoria se creo correctamente",
      data: `${id_cat}` //id de la nueva categoria
    })
  })
},

```

Gráfico 40 Backend para crear nuevas categorías

Elaborado por: El Investigador

Storycard – Verificación del Pago (Administrador)

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
15	Nuevo	4	4	4	4	Alta
DESCRIPCIÓN: Verificar el pago realizado por el cliente sobre su pedido.						
FECHA	ESTADO		COMENTARIO			
04/01/2023	Definido		Sin Comentarios			
04/01/2023	Implementado		Sin Comentarios			
05/01/2023	Ejecutado		Sin Comentarios			
06/01/2023	Verificado		Sin Comentarios			

Tabla 31 Storycard - Verificar el pago

Elaborado por: El Investigador

Dado que se utilizó PayPal como método de pago, este retorna un mensaje de aprobación del pago si este se realizó correctamente. Luego de esto el administrador de la aplicación tendrá que confirmar que el pago esta realizado y cambiar el estado del pedido a “En despacho”.

```

updateDespachado(req, res){
  const pedido = req.body;
  Pedido.updateDespachado(pedido.id_ped, (err, id_ped) => {
    if (err){
      return res.status(501).json({
        success: false,
        message: 'Hubo un error actualizando la orden',
        error: err
      })
    }
    return res.status(201).json({
      success: true,
      message: "El pedido se actualizo correctamente",
      data: `${id_ped}` //id de la nueva categoria
    })
  })
},

```

Gráfico 41 Backend para verificar el pago

Elaborado por: El Investigador

Storycard – Despachar Producto (Bodeguero)

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
16	Nuevo	3	3	4	4	Alta
DESCRIPCIÓN: Despachar el o los productos adquiridos por los clientes.						
FECHA	ESTADO		COMENTARIO			
08/01/2023	Definido		Sin Comentarios			
09/01/2023	Implementado		Sin Comentarios			
10/01/2023	Ejecutado		Sin Comentarios			
10/01/2023	Verificado		Sin Comentarios			

Tabla 32 Storycard - Despachar Productos

Elaborado por: El Investigador

Posterior a la verificación del pago el encargado de Bodega deberá cambiar el estado del pedido a “Enviado”. Para realizar esto la aplicación hace una modificación en la base de datos mediante una petición HTTP de tipo put.

```

updateEnviado(req, res){
  const pedido = req.body;
  Pedido.updateEnviado(pedido.id_ped, (err, id_ped) => {
    if (err){
      return res.status(501).json({
        success: false,
        message: 'Hubo un error actualizando la orden',
        error: err
      })
    }
    return res.status(201).json({
      success: true,
      message: "El pedido se actualizo correctamente",
      data: `${id_ped}` //id de la nueva categoria
    })
  })
}

```

Gráfico 42 Backend para despachar productos

Elaborado por: El Investigador

Storycard – Seleccionar Rol

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
17	Nuevo	3	3	3	3	Alta
DESCRIPCIÓN: Seleccionar rol luego de iniciar sesión						
FECHA	ESTADO		COMENTARIO			
28/01/2023	Definido		Sin Comentarios			
28/01/2023	Implementado		Sin Comentarios			
28/01/2023	Ejecutado		Sin Comentarios			
28/01/2023	Verificado		Sin Comentarios			

Tabla 33 Storycard - Seleccionar rol

Elaborado por: El Investigador

Para los usuarios que inicien sesión en la aplicación y tengan más de un rol asignado, se les mostrará una pantalla con los roles asignados. Los usuarios deberán seleccionar el rol con el que deseen utilizar la aplicación. Los usuarios que tengan solo un rol no podrán visualizar esta pantalla.

```

const isPasswordValid = await bcrypt.compare(password, myUser.contra_usu);
if(isPasswordValid){
  const token = jwt.sign({id_usu:myUser.id_usu,email_usu:myUser.email_usu},keys.secretOrKey, {});
  const data = {
    id_usu: `${myUser.id_usu}`,
    nombre_usu: myUser.nombre_usu,
    apellido_usu: myUser.apellido_usu,
    email_usu: myUser.email_usu,
    telefono_usu : myUser.telefono_usu,
    foto_usu : myUser.foto_usu,
    session_token: `JWT ${token}`,
    roles: JSON.parse(myUser.roles)
  }
}

return res.status(201).json({
  success: true,
  message: "La autenticación fue exitosa",
  data: data
})

```

Gráfico 43 Backend para seleccionar rol

Elaborado por: El Investigador

3.2.4 Fase IV: Estabilización.

En esta fase se listan los cambios y correcciones necesarias aplicadas en el desarrollo del proyecto para que este sea más eficiente.

- Se suprimieron las tablas de provincia y ciudad y en su lugar se utilizaron las APIs de Google Maps para tener una mayor precisión al momento de ingresar una dirección.

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
18	Nuevo	3	3	3	3	Media
DESCRIPCIÓN: Eliminar tablas de provincia y ciudad						
FECHA	ESTADO		COMENTARIO			
12/01/2023	Definido		Sin Comentarios			
12/01/2023	Implementado		Sin Comentarios			
12/01/2023	Ejecutado		Sin Comentarios			
12/01/2023	Verificado		Sin Comentarios			

Tabla 34 Storycard - Eliminar tablas de provincia y ciudad

Elaborado por: El Investigador

- Se suprimió el método de MercadoPago y se lo reemplazó por PayPal, esto debido a los requerimientos de la empresa luego de comprobar que MercadoPago no está disponible en Ecuador.

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
19	Nuevo	3	3	3	3	Alta
DESCRIPCIÓN: Ajustar método de pago						
FECHA	ESTADO		COMENTARIO			
21/01/2023	Definido		Sin Comentarios			
21/01/2023	Implementado		Sin Comentarios			
21/01/2023	Ejecutado		Sin Comentarios			
21/01/2023	Verificado		Sin Comentarios			

Tabla 35 Storycard - Ajustar métodos de pago

Elaborado por: El Investigador

- Se eliminó la tabla de fotos de productos debido a que, dado que los productos que se venderán en la aplicación son insumos, estos como máximo podrían tener dos fotos, de esta manera el almacenamiento en la nube difícilmente se llenará.

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
20	Nuevo	3	3	3	3	Media
DESCRIPCIÓN: Eliminar tabla de fotos						
FECHA	ESTADO		COMENTARIO			
28/12/2022	Definido		Sin Comentarios			
28/12/2022	Implementado		Sin Comentarios			
28/12/2022	Ejecutado		Sin Comentarios			
28/12/2022	Verificado		Sin Comentarios			

Tabla 36 Storycard - Eliminar tablas de fotos

Elaborado por: El Investigador

- Se elimino la tabla de Líneas debido a los requerimientos de la empresa, dado que la aplicación esta construida en base a vender productos de una sola línea.

N.º	TIPO	DIFICULTAD		ESFUERZO		PRIORIDAD
		ANTES	DESPUES	ANTES	DESPUES	
21	Nuevo	3	3	3	3	Media
DESCRIPCIÓN: Eliminar tabla de líneas						
FECHA	ESTADO		COMENTARIO			
03/01/2023	Definido		Sin Comentarios			
03/01/2023	Implementado		Sin Comentarios			
03/01/2023	Ejecutado		Sin Comentarios			
03/01/2023	Verificado		Sin Comentarios			

Tabla 37 Storycard - Eliminar tabla de líneas

Elaborado por: El Investigador

3.2.5 Fase V: Pruebas

De acuerdo con la metodología implementada para el desarrollo del presente proyecto, la quinta fase permite verificar el correcto funcionamiento de los módulos implementados en la aplicación, así como todas las conexiones con el servidor.

3.2.5.1 Módulo de Usuarios

- Inicio de sesión

Para ingresar en la aplicación el usuario tendrá que ingresar su usuario y su contraseña, la cual será enviada a la base de datos y se encriptará previo a la comparación con la contraseña almacenada. Si las credenciales coinciden el usuario podrá ingresar a la aplicación, en caso de no tener una cuenta deberá presionar en el texto de “Regístrate aquí”.

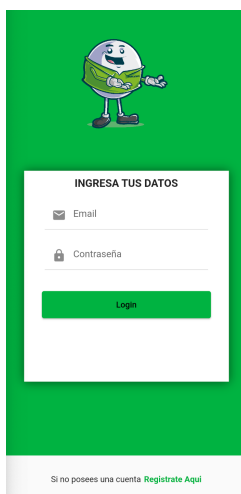


Gráfico 44 Pantalla de Inicio de Sesión

Elaborado por: El Investigador

- Registro de usuarios

En caso de no tener cuenta, los usuarios deberán llenar sus datos personales junto con un correo electrónico y una contraseña, con la posibilidad de subir una foto o imagen para su perfil.

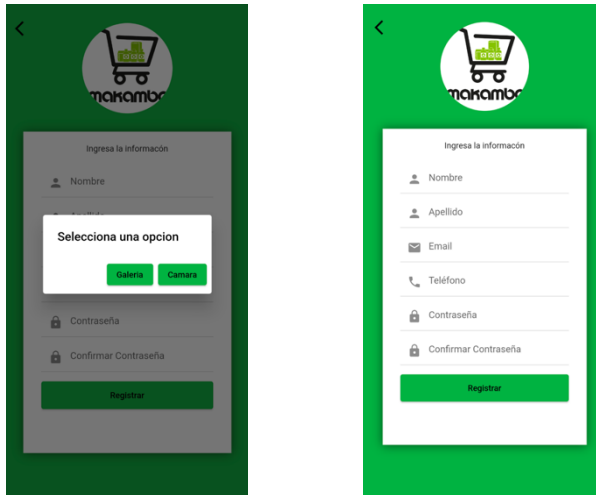


Gráfico 45 Pantallas de Registro de Usuario

Elaborado por: El Investigador

3.4.5.2 Módulo de clientes

- Catálogo de productos

El catálogo de productos muestra los productos filtrados por la categoría a la que pertenecen, los productos son mostrados mediante tarjetas las cuales contienen el precio del producto, el nombre, descripción e imagen.

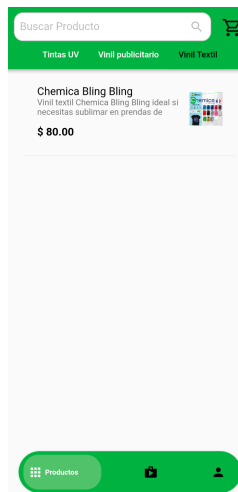


Gráfico 46 Pantalla de Catálogo de Productos

Elaborado por: El Investigador

- **Detalle de producto**

En el detalle del producto el cliente podrá seleccionar la cantidad que va a agregar al carrito de compras, también existen botones para aumentar y disminuir la cantidad, en el botón de agregar productos el usuario tiene una vista previa del subtotal.

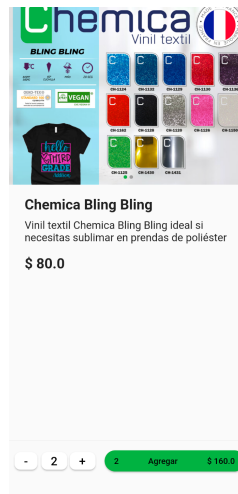


Gráfico 47 Pantalla de Detalle de Producto

Elaborado por: El Investigador

- **Carrito de compras**

La pantalla de carrito de compras muestra la lista de los productos seleccionados provenientes del catálogo de productos, en esta pantalla el cliente puede modificar las cantidades de su orden y podrá visualizar el total de su pedido.

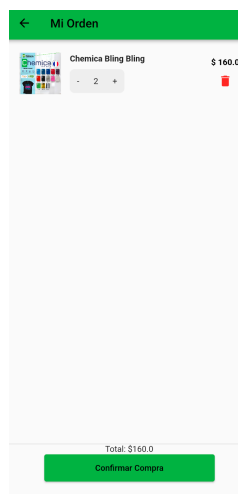


Gráfico 48 Pantalla de Carrito de Compras

Elaborado por: El Investigador

- Direcciones de envío

La sección de direcciones de envío consta de 3 pantallas, la pantalla que muestra la lista de direcciones agregadas, la pantalla para crear una nueva dirección y la pantalla que abre el mapa de Google para seleccionar una nueva dirección.

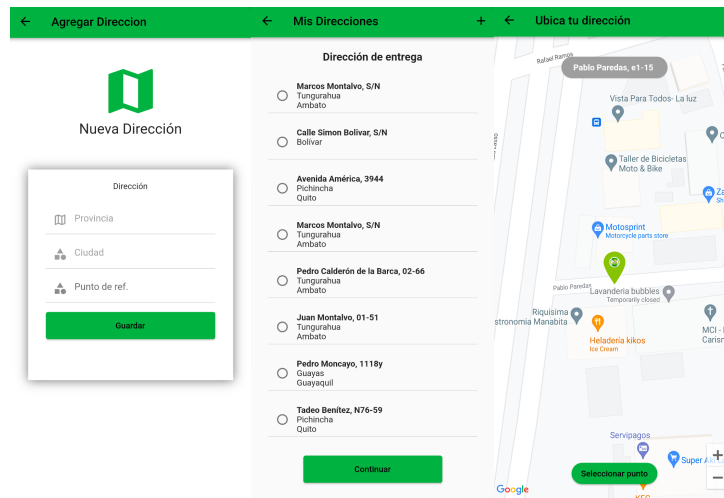


Gráfico 49 Pantallas de Direcciones de Envío

Elaborado por: El Investigador

- Pedido y Pago

Luego de seleccionar la dirección de envío, el cliente procederá a la pantalla de pagos, donde se deberá seleccionar el método de pago. Una vez que el pago ha sido aceptado el cliente volverá a la pantalla de inicio.

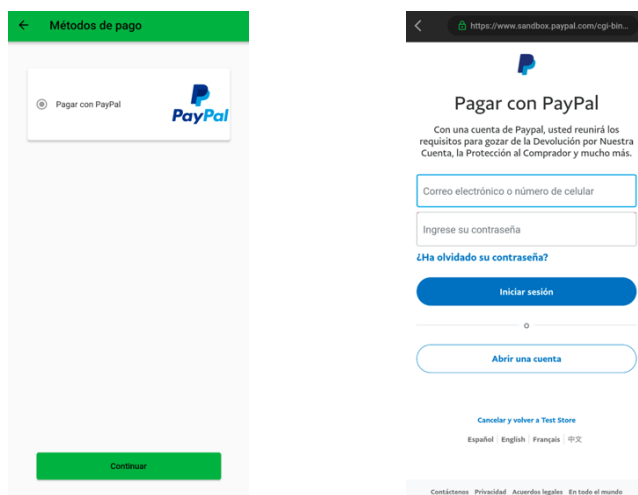


Gráfico 50 Pantallas de Pedido y Pago

Elaborado por: El Investigador

- **Listado de pedidos**

Esta pantalla sirve para que el cliente vea una lista de pedidos, los pedidos están clasificados por estado, los cuales pueden ser “PAGADO”, “EN DESPACHO” Y “ENVIADO”. Además, el cliente puede abrir cada pedido para ver el detalle de los productos y la dirección a la que fue enviado.

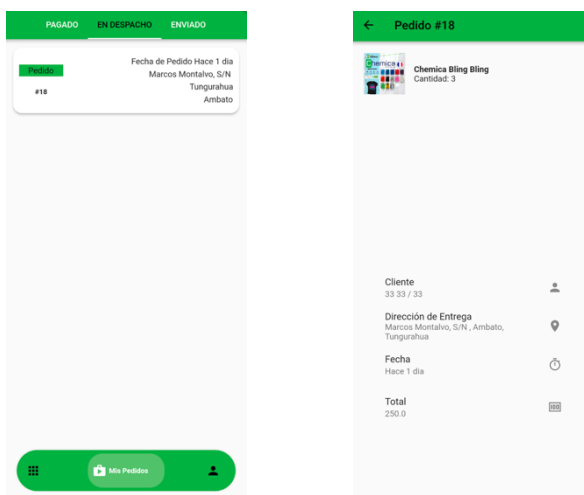


Gráfico 51 Pantalla de Listado de Pedidos

Elaborado por: El Investigador

3.4.5.3 Módulo de Administración

- **Agregar nueva categoría.**

El administrador podrá agregar nuevas categorías de productos, para esto deberá llenar el formulario el cual requiere el nombre y la descripción de la categoría.

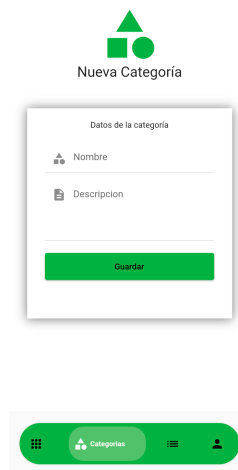


Gráfico 52 Pantalla de Agregar nueva categoría

Elaborado por: El Investigador

- **Agregar nuevo producto**

Para agregar un nuevo producto, el administrador deberá llenar el formulario con la información solicitada, adicionalmente todos los productos tienen 2 fotos para mostrar, el administrador deberá seleccionar las fotos de su galería o capturarlas mediante la cámara. Por último, se deberá seleccionar una categoría a la cual va a pertenecer el producto.

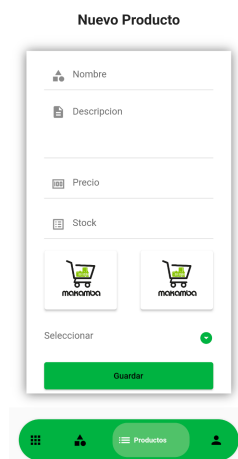


Gráfico 53 Pantalla de agregar nuevo producto

Elaborado por: El Investigador

- **Listado de pedidos**

El administrador es el encargado de verificar el pago del pedido y del despacho del mismo. Esta sección le permitirá al administrador ver el estado de todos los pedidos de la aplicación.



Gráfico 54 Pantalla de Listado de Pedidos

Elaborado por: El Investigador

3.4.5.4 Módulo de Bodega.

- **Despachar productos**

Después de que el administrador haya verificado el pago, el encargado de bodega deberá marcar el pedido como “Despachado”.

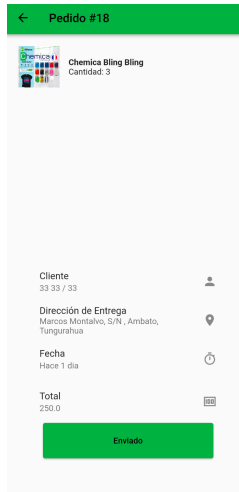


Gráfico 55 Pantalla para despachar productos

Elaborado por: El Investigador

CAPÍTULO IV.- CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- En el desarrollo de aplicaciones móviles E-commerce se pueden combinar bases de datos relacionales tales como MySQL y servicios de Cloud Storage como Firebase para poder almacenar fotos o videos de los productos a la venta. De esta manera se logra obtener una base de datos relacional eficiente y un espacio seguro para almacenar los archivos multimedia.
- El desarrollo del presente proyecto demostró que la arquitectura MVVM proporciona una mayor flexibilidad en la implementación de la interfaz de usuario y la lógica de los procesos, lo cual brinda una mejor experiencia de desarrollo al permitir una mayor velocidad de actualización y representación de datos en la interfaz de la aplicación móvil. Esta flexibilidad se demostró durante la fase de estabilización del proyecto, la cual tuvo como resultado la eliminación de módulos previamente definidos. De igual manera la arquitectura MVVM permite una mayor facilidad de mantenimiento y escalabilidad a la aplicación, las cuales son características esenciales para una eficiente implantación en la empresa MercantilMakamba Cia. Ltda.
- La combinación de Node.js para el desarrollo del backend y Flutter para el desarrollo del frontend es compatible con la gran mayoría de plataformas en el mercado. Node.js está disponible para Windows, macOS y Linux, mientras que Flutter es compatible con iOS, Android, Escritorio y Web. Esto significa que una sola aplicación desarrollada con estas herramientas estará disponible para una amplia variedad de dispositivos.
- La implementación de la aplicación mediante el uso de la metodología Mobile-D resultó exitosa ya que esta metodología está enfocada en la satisfacción del usuario y la rapidez en el desarrollo, cabe recalcar que la metodología Mobile-D brinda una mayor flexibilidad al momento de realizar cambios según los requisitos del cliente.
- La creación de un archivo de tipo JavaScript, el cual almacena las credenciales para acceder a la base de datos, ayudó al alojamiento de la

aplicación de Node.js. Esto se debe a que algunos servicios de hosting como Railway envían, mediante el método `process.env`, información a las credenciales de acceso creadas previamente por el desarrollador.

4.2 Recomendaciones

Implementar técnicas y sistemas de seguridad tales como encriptación de contraseñas, una pasarela de pagos previamente probada, mitigación de ataques de inyección SQL y que la comunicación entre la aplicación y el servidor se realice a través de conexiones seguras.

Mantener un diseño centrado en el usuario con el fin de que la interfaz de la aplicación genere una experiencia de uso agradable e intuitiva. Para lograr esto es necesario un diseño simple y fácil de entender, la aplicación debe ser consistente en cuanto a sus colores, se deben utilizar animaciones y transiciones adecuadas. También se recomienda poner especial atención en que la interfaz de usuario sea adaptable a diferentes tipos de pantallas.

Utilizar las librerías y plugins disponibles para Flutter tales como librerías de geolocalización, acceso a cámara y acceso a galería, de tal manera que estas permitan ahorrar tiempo de desarrollo el cual puede ser utilizado en mejorar las interfaces de usuario o la seguridad del backend.

5. Bibliografía

- [1] K. G. RODRIGUEZ, O. J. ORTIZ, A. I. QUIROZ, and M. L. PARRALES, "El e-commerce y las Mipymes en tiempos de Covid-19," *Espacios*, vol. 41, no. 42, Nov. 2020, doi: 10.48082/espacios-a20v41n42p09.
- [2] S. Kunchaparthi and ----Ms K Susmitha, "Impact of COVID 19 on E-Commerce", doi: 10.13140/RG.2.2.20236.85127.
- [3] Cámara de Comercio Electronico del Ecuador, "Transacciones-electronicas-en-Ecuador-durante-el-Covid19".
- [4] E. Javier, Z. Goveo, M. Fernanda, M. Andrade, J. Fernando, and L. Aguirre, "Ecommerce Business Model in Ecuador," *IAC Edición Número 1*, vol. 1, no. 2, 2020, [Online]. Available: <https://orcid.org/0000-0002-7138-7447>
- [5] Wenhao Wu, "React Native vs Flutter, cross-platform mobile application frameworks."

- [6] A. Altamirano, "APLICACIÓN MÓVIL MULTIPLATAFORMA DE PAGO EN LÍNEA APLICANDO FLUTTER Y DART, PARA LA OBTENCIÓN DE BOLETOS EN LA COOPERATIVA DE TRANSPORTE CITA EXPRESS DE LA CIUDAD DE AMBATO," Universidad Técnica de Ambato, Ambato, 2021.
- [7] P. Barrientos Felipa and P. Barrientos Felipa, "Marketing + internet = e-commerce: oportunidades y desafíos," *Revista Finanzas y Política Económica*, vol. 9, no. 1, pp. 41–56, 2017, doi: 10.14718/revfinanzpolitecon.2017.9.1.3.
- [8] Daniel Napoleón Marroquín Falconí, "Factores que han influenciado la intención de compra de los consumidores quiteños con respecto al E Commerce a partir de la pandemia COVID-19 en el año 2021," 2021.
- [9] Ríos Ruíz and A. de los Angeles, "ANÁLISIS Y PERSPECTIVAS DEL COMERCIO ELECTRÓNICO EN MÉXICO.," 2015.
- [10] E. Monterrubio-Hernández, "Plataformas de comercio electrónico," *Publicación semestral*, vol. 3, no. 6, pp. 25–26, 2021, [Online]. Available: <https://repository.uaeh.edu.mx/revistas/index.php/ixtlahuaco/issue/archive>
- [11] P. Santiago and J. Mesías, "COMERCIO ELECTRÓNICO EN ECUADOR," 2018, [Online]. Available: <https://www.eumed.net/rev/oel/2018/09/comercio-electronico-ecuador.html>
- [12] A. H. Trasobares, "LOS SISTEMAS DE INFORMACIÓN: EVOLUCIÓN Y DESARROLLO."
- [13] C. Ignacio, V. Paola, and J. Fructuoso HERNÁNDEZ, "Metodologías actuales de desarrollo de software," 2015. [Online]. Available: www.ecorfan.org/bolivia
- [14] C. Rodríguez and H. Enríquez, "Características del desarrollo en Frameworks multiplataforma para móviles* Features Frameworks Development Platform for Mobile."
- [15] Lisandro Nahuel Delía, "Desarrollo de Aplicaciones Móviles Multiplataforma," 2017.
- [16] Dagne Lukas, "Flutter for cross-platform App and SDK development," 2019.
- [17] Cristian Zazo Millán, "Migración de aplicaciones Android hacia Flutter, un framework para desarrollo de apps multiplataforma," 2018.
- [18] Cámara de Comercio de Bogotá, "EL MUNDO CONECTADO POR LAS API," 2019.
- [19] E. Haro, T. Guarda, A. O. Zambrano Peñaherrera, and G. Ninahualpa Quiña, "Desarrollo backend para aplicaciones web, Servicios Web Restful: Node.js vs Spring Boot," 2018.
- [20] Á. Gutiérrez, "Almacenamiento en la nube," 2018.

6. Anexos

Ficha Bibliográfica	
TEMA	PROTOTIPO DE APLICACION MOVIL PARA VENTA DE LICORES A DOMICILIO
AUTOR/ES	LEONARDO CUBIDES CUBILLOS OSCAR JAVIER GOMEZ TREJOS
PROPOSITO	El propósito de la ficha es recolectar ideas y buenas prácticas de otros proyectos de investigación relacionados con las aplicaciones móviles E-commerce
IDEAS O CONCEPTOS	Lo que propone el autor es que para crear una aplicación móvil E-commerce es necesaria la definición de requerimientos funcionales, tales como los métodos de pago, los módulos de la aplicación y todas las funcionalidades deseadas por el cliente; y requerimientos no funcionales los cuales se clasifican en: portabilidad, seguridad, rendimiento, escalabilidad, disponibilidad y concurrencia
CONCLUSIONES	La conclusión que se puede obtener es que se debe construir una matriz de requerimientos funcionales y no funcionales antes de la realización del prototipo de una aplicación móvil E-commerce

Anexo 1 Ficha Bibliografica 1

Elaborado por: El Investigador

Ficha Bibliográfica

TEMA	DEVELOPMENT OF ANDROID BASED MOBILE APP FOR PRESTASHOP E-COMMERCE SHOPPING CART
AUTOR/ES	SWAPL JAGTAP
PROPOSITO	El propósito de la ficha es conocer las ventajas del comercio electrónico de tal manera que se puedan tomar ideas y aplicarlas en el desarrollo.
IDEAS O CONCEPTOS	<p>Las ventajas que propone el autor son:</p> <ul style="list-style-type: none"> - Poder comprar en cualquier día y a cualquier hora - Poder comprar desde la comodidad de la casa - No tener que esperar filas para pagar - Poder comprar con máxima privacidad - Ahorrar en gasolina o pasaje <p>Las desventajas que propone el autor son</p> <ul style="list-style-type: none"> - Se visualiza solo un producto a la vez - Puede tomar mucho tiempo si el internet está lento - Costo de envío - Existe mucha inseguridad al momento de pagar por internet - No poder interactuar con el producto personalmente
CONCLUSIONES	El comercio electrónico ofrece muchas ventajas y desventajas, para el desarrollo del presente proyecto se mitigarán la mayoría de desventajas mediante la aplicación de buenas prácticas de desarrollo, tales como la implementación de un método de pago seguro y el diseño de una interfaz de usuario amigable

Anexo 2 Ficha Bibliográfica 2

Elaborado por: El Investigador

Ficha Bibliográfica	
TEMA	ANDROID APP DEVELOPMENT FOR KOKANRAJA
AUTOR/ES	INTERNATIONAL RESEARCH JOURNAL OF ENGINEERING AND TECHNOLOGY(IRJET)
PROPOSITO	El propósito de la ficha es conocer los roles que van a tener los usuarios dentro de una aplicación E-commerce
IDEAS O CONCEPTOS	1) Administrador: Es una persona que utiliza el panel de administrador del lado del vendedor para monitorizar y organizar los pedidos de los clientes 2) Cliente: Es la persona que realiza las compras, es una parte primordial en un modelo de “Business to Customers”(B2C)
CONCLUSIONES	Existen dos tipos de aplicaciones E-commerce, B2C y B2B(Business to Business). Para el desarrollo del presente proyecto se propuso utilizar el modelo de B2C ya que de acuerdo con los requerimientos de la empresa, la aplicación no debe distinguir entre compradores y distribuidores.

Anexo 3 Ficha Bibliográfica 3

Elaborado por: El Investigador