



# UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

DESARROLLO DE UN SISTEMA WEB Y MOVIL PARA LA GESTIÓN EN  
EL PROCESO DE VENTAS EN LA EMPRESA SOLO PANAS

CHALCO MOROCHO DAVIS SALVATORE  
INGENIERO DE SISTEMAS

MACHALA  
2023



# UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

DESARROLLO DE UN SISTEMA WEB Y MOVIL PARA LA  
GESTIÓN EN EL PROCESO DE VENTAS EN LA EMPRESA SOLO  
PANAS

CHALCO MOROCHO DAVIS SALVATORE  
INGENIERO DE SISTEMAS

MACHALA  
2023



# UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

TRABAJO TITULACIÓN  
PROPUESTAS TECNOLÓGICAS

DESARROLLO DE UN SISTEMA WEB Y MOVIL PARA LA GESTIÓN EN EL  
PROCESO DE VENTAS EN LA EMPRESA SOLO PANAS

CHALCO MOROCHO DAVIS SALVATORE  
INGENIERO DE SISTEMAS

VALAREZO PARDO MILTON RAFAEL

MACHALA, 17 DE OCTUBRE DE 2023

MACHALA  
2023

# TITULACION

---

## INFORME DE ORIGINALIDAD

---

9%

INDICE DE SIMILITUD

9%

FUENTES DE INTERNET

0%

PUBLICACIONES

9%

TRABAJOS DEL  
ESTUDIANTE

---

## FUENTES PRIMARIAS

---

1	Submitted to Universidad Técnica de Machala Trabajo del estudiante	2%
2	repository.unicatolica.edu.co Fuente de Internet	2%
3	tauja.ujaen.es Fuente de Internet	1%
4	revistas.pascualbravo.edu.co Fuente de Internet	1%
5	www.coursehero.com Fuente de Internet	1%
6	repositorio.uta.edu.ec Fuente de Internet	1%
7	hdl.handle.net Fuente de Internet	1%
8	Submitted to Universidad del Istmo de Panamá Trabajo del estudiante	1%

---

## CLÁUSULA DE CESIÓN DE DERECHO DE PUBLICACIÓN EN EL REPOSITORIO DIGITAL INSTITUCIONAL

El que suscribe, CHALCO MOROCHO DAVIS SALVATORE, en calidad de autor del siguiente trabajo escrito titulado DESARROLLO DE UN SISTEMA WEB Y MOVIL PARA LA GESTIÓN EN EL PROCESO DE VENTAS EN LA EMPRESA SOLO PANAS, otorga a la Universidad Técnica de Machala, de forma gratuita y no exclusiva, los derechos de reproducción, distribución y comunicación pública de la obra, que constituye un trabajo de autoría propia, sobre la cual tiene potestad para otorgar los derechos contenidos en esta licencia.

El autor declara que el contenido que se publicará es de carácter académico y se enmarca en las disposiciones definidas por la Universidad Técnica de Machala.

Se autoriza a transformar la obra, únicamente cuando sea necesario, y a realizar las adaptaciones pertinentes para permitir su preservación, distribución y publicación en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

El autor como garante de la autoría de la obra y en relación a la misma, declara que la universidad se encuentra libre de todo tipo de responsabilidad sobre el contenido de la obra y que asume la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

Aceptando esta licencia, se cede a la Universidad Técnica de Machala el derecho exclusivo de archivar, reproducir, convertir, comunicar y/o distribuir la obra mundialmente en formato electrónico y digital a través de su Repositorio Digital Institucional, siempre y cuando no se lo haga para obtener beneficio económico.

Machala, 17 de octubre de 2023



CHALCO MOROCHO DAVIS SALVATORE  
0706247533

## **DEDICATORIA**

Dedico el presente trabajo a mi familia, en particular, a mis padres que han sido de grande apoyo en mi vida y carrera universitaria, fundamentales para el éxito en esta titulación.

A mis compañeros y amigos, que han sido parte del camino y que me han ayudado cuando he necesitado.

A mis docentes que, gracias a ellos, me ha sido posible completar la culminación de mi carrera para la obtención de mi título universitario.

*Sr. Davis Salvatore Chalco Morocho*

## **AGRADECIMIENTO**

El primer agradecimiento es a Dios por haberme permitido llegar a esta etapa de mi vida, con salud y fortaleza; a mis padres por el esfuerzo que han realizado a lo largo de mi carrera, apoyándome en los momentos duros con buenos ánimos y sanos consejos.

A mis amigos y compañeros, que me han enseñado el trabajo en equipo para mi formación profesional.

A mis docentes, tutor y universidad, por ser una segunda casa donde he vivido bien mi etapa universitaria, conociendo siempre personas de calidad y claves para la culminación de mi carrera, en particular, al Ing. Milton Valarezo Pardo, por ser mi tutor de tesis que me brindo su valioso conocimiento.

*Sr. Davis Salvatore Chalco Morocho*

## RESUMEN

Actualmente, la gestión de las ventas en una empresa es fundamental para el éxito y crecimiento de esta, sin embargo, con el avance de la tecnología, el desarrollo de sistemas web y móviles se ha convertido en una herramienta esencial para optimizar y agilizar estos procesos con la finalidad del crecimiento empresarial.

Frente a esta problemática se propone el desarrollo de una aplicación móvil y web para la toma de pedidos de una empresa, facilitando al administrador ver las necesidades y recomendaciones del negocio, para ofrecer a sus clientes un mejor servicio de calidad.

Para el desarrollo del proyecto de investigación se utilizó la metodología SWIRL que se caracteriza por ofrecer una mejor flexibilidad con respecto a los cambios y que permite integrarse ante los diferentes procesos inmerso en el desarrollo de software, además con el apoyo de Node JS, MySQL y React Native el cual es un Framework JavaScript el cual es para el desarrollo de aplicaciones nativas, se obtuvo una aplicación con una interfaz amigables para facilitar el ingreso y consulta de los productos que requiere el cliente. Además, que la aplicación móvil cuenta con una sección Web específica para el administrador de la tienda donde podrá visualizar mediante un Dashboard métricas importantes para la toma de decisiones del negocio.

La aplicación móvil ofrece diversas funcionalidades que facilitaran la gestión de ventas entre ellas se incluyen el registro de los clientes, el sistema permitirá almacenar y gestionar la información de sus clientes como nombres, apellidos, correo electrónico, direcciones y teléfono de contacto lo que facilitara la comunicación con ellos.

Dentro de la gestión de productos la aplicación muestra los productos que ofrece la empresa, incluyendo información detallada del productos, precio y disponibilidad, esto facilitara la sección y venta de productos.

La aplicación permitirá realizar un seguimiento de los pedidos realizadas por el cliente, incluyendo el estado de estos, la fecha de entrega y el pago, además cuenta con sistema de pago en efectivo y con tarjeta.

El sistema web proporcionará al administrador tener un análisis de ventas, como ventas por producto, por cliente, esto permitirá a la empresa tomar decisiones informadas y mejorar su estrategia de ventas.

Una vez finalizada la aplicación se realizó una prueba de calidad aplicando la norma de calidad ISO/IEC 9126, que proporciona métricas que califican la funcionalidad,



confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad, todas estas características permiten determinar si la aplicación es de calidad o no, con la aplicación de esta norma se puede determinar que el software posee un alto grado de cumplimiento de los requisitos del usuario.

Como resultado, se lograron excelentes niveles de calidad, según la escala de Likert; en funciones sobre funciones para 4 criterios, una calificación de 4/4 es excelente; también en carácter confiabilidad, 2/3 excelente y 1/3 buena 3 criterios; facilidad de uso continua, puntuado 4/4 excelente en los 4 criterios; la características de desempeño, 2 criterios evaluados excelente 1/2, bueno 1/2; con funciones de mantenimiento, los resultados fueron 1/2 excelente y 1/2 tan bueno en sus 2 criterios, y finalmente obtiene 2 criterios de portabilidad, 1/2 excelente tan malo como 1/2, se puede decir que el software cumple en gran medida con los requisitos y detalles del usuario, planificación y diseño.

**Palabras claves:** Aplicación Móvil, Node Js, React Native, Dashboard, MySql

## **ABSTRACT**

Currently, sales management in a company is essential for its success and growth, however, with the advancement of technology, the development of web and mobile systems has become an essential tool to optimize and streamline these processes. for the purpose of business growth.

Faced with this problem, the development of a mobile and web application is proposed for taking orders from a company, making it easier for the administrator to see the needs and recommendations of the business, to offer its customers a better quality service.

For the development of the research project, the SWIRL methodology was used, which is characterized by offering better flexibility with respect to changes and that allows integration with the different processes immersed in software development, also with the support of Node JS, MySql and React Native which is a JavaScript Framework which is for the development of native applications, an application with a friendly interface was obtained to facilitate the entry and consultation of the products that the client requires. In addition, the mobile application has a specific Web section for the store administrator where you can view important metrics for business decision-making through a Dashboard.

The mobile application offers various functionalities that will facilitate sales management, including customer registration, the system will allow you to store and manage customer information such as names, surnames, email, addresses and contact telephone numbers, which will facilitate the communication with them.

Within product management, the application shows the products offered by the company, including detailed product information, price and availability, this will facilitate the section and sale of products.

The application will allow you to track the orders placed by the customer, including their status, delivery date and payment, it also has a cash and card payment system.

The web system will provide the administrator with sales analysis, such as sales by product, by customer, this will allow the company to make informed decisions and improve its sales strategy.

Once the application was finished, a quality test was carried out applying the ISO/IEC 9126 quality standard, which provides metrics that qualify the functionality, reliability, usability, efficiency, maintainability and portability, all these characteristics allow us to determine if the application is of quality. or not, with the application of this standard it can be determined that the software has a high degree of compliance with the user's requirements.

As a result, excellent quality levels were achieved, according to the Likert scale; in features on features for 4 criteria, a rating of 4/4 is excellent; also in reliability, 2/3 excellent and 1/3 good 3 criteria; continued ease of use, rated 4/4 excellent on all 4 criteria; the performance characteristics, 2 criteria evaluated excellent 1/2, good 1/2; with maintenance features, the results were 1/2 excellent and 1/2 as good on your 2 criteria, and finally you get 2 portability criteria, 1/2 excellent as bad as 1/2, you can say that the software complies on largely with user requirements and details, planning and design.

**Keywords:** Mobile Application, Node Js, React Native, Dashboard, MySql

## ÍNDICE DE CONTENIDO

DEDICATORIA.....	1
AGRADECIMIENTO.....	2
RESUMEN.....	3
ABSTRACT .....	5
INTRODUCCION.....	11
1  CAPÍTULO I. DIAGNOSTICO DE NECESIDADES Y REQUERIMIENTOS .....	13
1.1  Ámbito de aplicación: descripción del contexto y hecho de intereses .....	13
1.2  Establecimiento de requerimientos.....	14
1.3  Justificación del requerimiento a satisfacer .....	15
2  CAPÍTULO II. DESARROLLO DEL PROTOTIPO .....	16
2.1  Definición del prototipo.....	16
2.2  Fundamentación teórica del prototipo .....	16
2.2.1  Arquitectura de software .....	17
2.2.2  MySql.....	19
2.2.3  Herramientas CASE.....	20
2.2.4  Metodología de desarrollo de Software .....	24
2.3  Objetivos del prototipo.....	28
2.3.1  Objetivo general.....	28
2.3.2  Objetivos específicos .....	28
2.4  Diseño del prototipo.....	28
2.4.1  FASE DE ANÁLISIS .....	28
2.4.2  FASE DE PLANIFICACIÓN .....	34
2.4.3  FASE DE MODELADO .....	39
2.5  FASE DE EJECUCIÓN Y/O ENSAMBLAJE DEL PROTOTIPO.....	60
2.5.1  Módulo principal.....	60
2.5.2  Módulo Registro de Usuario.....	61
2.5.3  Módulo Iniciar Sesión.....	61
2.5.4  Módulo Ventana Principal .....	62

2.5.5	Módulo Detalle de los Productos .....	63
2.5.6	Módulo Carrito de Compras.....	64
2.5.7	Módulo Datos de Envió.....	65
2.5.8	Módulo Métodos de Pago .....	66
2.5.9	Módulo Pago en Línea.....	67
2.5.10	Módulo Órdenes.....	68
2.5.11	Módulo Detalle de las Órdenes .....	69
2.5.12	Dashboard Web.....	71
3	CAPÍTULO III. EVALUACIÓN DEL PROTOTIPO .....	72
3.1	PLAN DE EVALUACIÓN .....	72
3.2	RESULTADOS DE LA EVALUACIÓN.....	72
3.3	CONCLUSIONES.....	76
3.4	RECOMENDACIONES .....	77
	REFERENCIAS BIBLIOGRAFICAS .....	78
4	ANEXOS .....	82

## **INDICE DE ILUSTRACIONES**

Ilustración 1:	Ventana Principal de la Aplicacion .....	60
Ilustración 2:	Ventana del Registro de Clientes .....	61
Ilustración 3:	Ventana de Inicio de Sesion.....	62
Ilustración 4:	Ventana Principal Aplicación.....	63
Ilustración 5:	Ver productos disponibles .....	64
Ilustración 6:	Carrito de compras.....	65
Ilustración 7:	Datos de envió del pedido .....	66
Ilustración 8:	Método de pago del pedido.....	67
Ilustración 9:	Método Pago en línea del pedido.....	68
Ilustración 10:	Ver todas las ordenes realizadas .....	69
Ilustración 11:	Ver detalle de cada pedido.....	70
Ilustración 12:	Dashboard Web .....	71

## INDICE DE TABLAS

Tabla 1: Matriz de Stakeholders .....	29
Tabla 2: Funcionalidades del sistema .....	29
Tabla 3: Limitaciones del sistema.....	30
Tabla 4: Tabla de factibilidad Operativa .....	31
Tabla 5: Tabla de Factibilidad Técnica .....	31
Tabla 6: Tabla de Factibilidad Economica.....	32
Tabla 7: Requerimientos Funcionales .....	33
Tabla 8: Requerimientos no funcionales .....	34
Tabla 9: Historia de usuario (Inicio de Sesión) .....	34
Tabla 10: Historia de usuario (Registro de usuario) .....	35
Tabla 11: Historia de usuario (Administración de datos de Facturación).....	35
Tabla 12: Historia de usuario (Perfil del Negocio) .....	36
Tabla 13: Historia de usuario (Visualización de los productos) .....	36
Tabla 14: Historia de usuario (Gestión de productos) .....	36
Tabla 15: Historia de usuario (Gestión de Pedidos) .....	37

## INDICE DE FIGURAS

Figura 1 : Fundamentación teórica del prototipo .....	17
Figura 2: Arquitectura Cliente-Servidor [11].....	18
Figura 3: Modelado de tres capas .....	19
Figura 4: Representación React Native [31].....	23
Figura 5: Flujo de datos Redux [33].....	24
Figura 6: Ciclo de desarrollo del software [36].....	24
Figura 7: Ciclo de vida de SWIRL [42].....	27
Figura 8: Gestión del Cronograma .....	38
Figura 9: Diseño de la Base de datos.....	39
Figura 10: Diseño Navegacional.....	39
Figura 11: Modelado Menú Principal .....	40
Figura 12: Modelado Registro de Usuario .....	41
Figura 13: Modelado Inicio de Sesión .....	42
Figura 14: Modelado Ventana Principal.....	43
Figura 15: Modelado Productos disponibles.....	44

Figura 16: Modelado Detalle del producto .....	45
Figura 17: Modelado carrito de compras .....	46
Figura 18: Modelado Dirección de Entrega .....	47
Figura 19: Modelado Método de Pago .....	48
Figura 20: Modelado Pago con Tarjeta .....	49
Figura 21: Modelado del listado de todos los pedidos realizados.....	50
Figura 22: Modelado Detalles del pedido .....	51
Figura 23: Caso de Uso (Inicio de Sesión) .....	52
Figura 24: Caso de Uso (Registro de Clientes) .....	53
Figura 25: Caso de Uso (Búsqueda de productos).....	53
Figura 26: Caso de Uso (Realizar Pedido) .....	54
Figura 27: Diagrama de Actividades (Registro Cliente).....	55
Figura 28: Diagrama de Actividades (Iniciar Sesión).....	55
Figura 29: Diagrama de Actividades (Selección de Productos).....	56
Figura 30: Diagrama de Actividades (Ver Productos).....	56
Figura 31: Diagrama de Secuencia (Registro de Cliente) .....	57
Figura 32: Diagrama de Secuencia (Inicio de Sesión).....	58
Figura 33: Diagrama de Secuencia (Realizar Pedido).....	58

## INTRODUCCION

A consecuencia del inicio del Covid-19 los negocios han tenido que adaptarse a las nuevas tecnologías para poder ofertar sus productos, como son las aplicaciones de pedido de comida a domicilio; Francesca Ferrero señala que en tiempos pre pandemia el servicio de delivery solo constituía hasta un 15% de las ventas, pero esta tasa alcanzó el 50% en mayo del 2021 [1].

A lo largo de los últimos años la tecnología para dispositivos móviles ha revolucionado de una manera muy significativa, tal que permite realizar actividades de la vida cotidiana a través de dispositivos móviles como teléfonos inteligentes. A pesar del sombrío panorama económico, algunas empresas se han aventurado en las herramientas digitales para sobrevivir y continuar generando ingresos, incluidas las aplicaciones digitales, o "Apps", programas informáticos diseñados para realizar tareas complejas y mejorar la experiencia informática [2].

Actualmente la automatización de los negocios es la clave para alcanzar el éxito, la gestión de ventas es un proceso fundamental de cualquier empresa, ya que de ello depende la generación de ingresos y el crecimiento del negocio. En la actualidad, el desarrollo de sistemas web y móviles se ha convertido en una herramienta clave para optimizar y agilizar este proceso. El objetivo de esta tesis es desarrollar una aplicación móvil para la gestión en el proceso de ventas en una empresa. Este sistema permitirá a la empresa gestionar de manera efectiva todas las etapas del proceso de ventas, desde la captación de clientes hasta la generación de facturas y seguimiento postventa.

Para lograr esto, se utilizarán tecnologías web y móviles como Node Js, React Native, Redux y otras herramientas de desarrollo. El sistema móvil ofrecerá diversas funcionalidades que facilitarán la gestión de ventas, como registro de clientes, gestión de productos, proceso de ventas. Con el sistema web el administrador podrá realizar un análisis de todas las ventas de la empresa con la finalidad de mejorar su eficiencia y aumentar sus ventas.

El documento se estructura en tres capítulos:

1er capítulo. – Dentro de este capítulo se analizarán los requerimientos del prototipo, además de conocer la problemática y justificación del proyecto.

2do capítulo. - detalla el desarrollo de proyecto, esta fase consta de 2 partes, en la primera parte del capítulo se describe el diseño del prototipo, describe conceptos



utilizados en el desarrollo del proyecto y en la última parte se procede a los diseños de la aplicación y su correspondiente desarrollo.

3er capítulo. - es el último capítulo en el que, se procede a conocer los resultados del proyecto implementado, así como también, se detallan las conclusiones y recomendaciones según los objetivos planteados acorde a su solución.

# **1 CAPÍTULO I. DIAGNOSTICO DE NECESIDADES Y REQUERIMIENTOS**

## **1.1 Ámbito de aplicación: descripción del contexto y hecho de intereses**

El internet ha tenido un crecimiento y una expansión global desde los 90, año de su primera aparición [3]. En Ecuador este incremento se ha visto reflejado desde el año 2010, en la zona urbana con un 16.70%, con un incremento del 50% en el año 2020, y, en la zona rural en el año 2010 con un porcentaje del 1.3% incrementando en el tiempo al año 2020 con un 66.70% [4].

Hoy en día, existen todo tipo de aplicaciones que permiten satisfacer las necesidades de las personas. Los juegos, escuchar música, leer libros, ver videos, redes sociales, etc. son cada vez más populares y la demanda también aumenta.

Actualmente existen múltiples plataformas y aplicaciones móviles que brindan diferentes servicios para permitir cubrir las necesidades de las personas como es pedir transporte, pago de servicios básicos, pedido de comidas rápidas en locales comerciales, hasta algo simple que es pedir productos de un negocio.

Dentro del negocio las tareas administrativas son llevadas de manera manual, por lo que ha ocasionado problemas como es no tener a la mano de manera sistematizada la información de los clientes o de las ventas realizadas, lo cual genera pérdida de tiempo entre los empleados y cuando el cliente realiza un pedido de algún producto del negocio.

Desde el 2020 la creación de aplicaciones móviles para la gestión de ventas se incrementó para satisfacer las necesidades de los negocios, tanto así que cada ciudad se adapta a las necesidades de sus clientes para crear aplicaciones amigables para el usuario y fácil de usar, pero en Ecuador el sector comercial se ve afectado por el proceso de compra, estamos pasando de comprar en tiendas físicas tradicionales a comprar en línea a través de redes sociales, páginas web, App y otras herramientas de compra virtual, aunque esta forma de comprar parece más fácil, es también un proceso complejo, ya que se han encontrado problemas como retrasos en los plazos de entrega, problemas de pago, falta de selección de productos, problemas de ubicación, etc.; como resultado, los clientes no procesan las compras ni evitan las aplicaciones, lo que genera una experiencia de usuario más deficiente.

Al poder ver estos tipos de inconvenientes que se presentan dentro de la sociedad, se puede conocer que para lograr el éxito y la experiencia sea satisfactoria se debe desarrollar servicios rápidos y que garanticen la entrega a tiempo del servicio a ofrecer.

## 1.2 Establecimiento de requerimientos

La presente aplicación móvil se denomina SOLOPANAS la cual tiene como finalidad brindar a las personas que puedan comprar los productos con el servicio de entrega a domicilio.

Por lo tanto, se plantea desarrollar una aplicación móvil donde se agrupe todo lo antes mencionado y para eso se obtienen los siguientes requerimientos:

### Requerimientos funcionales

- La aplicación permitirá el registro de los clientes por medio de correo electrónico.
- El registro del cliente debe realizarse desde la aplicación donde deberá registrar el usuario información relevante del mismo.
- Los productos registrados tendrán un perfil donde se mostrará su información básica: código, nombre, categoría, precio, stock disponible.
- La orden de pedido podrá ser visualizada y eliminada en cualquier momento por el dueño del negocio que tenga acceso a la cuenta.
- El dueño del negocio y el cliente tendrán un perfil donde se mostrará su información: Nombre del propietario, correo, nombre de la empresa, ciudad, dirección y teléfono.
- La aplicación contará con un panel general donde se visualizará todos los pedidos realizados.
- La aplicación poseerá un buscador y un filtro para los usuarios puedan encontrar los productos con mayor facilidad.
- Los usuarios podrán observar todas sus facturas de los pedidos realizados.
- En cuanto al método de pago, se manejará con pago en efectivo y en línea.

### Requerimientos no funcionales

- **Requisitos de rendimiento**

El tiempo de respuesta para las consultas o toma de pedidos deben ser rápidas, garantizando que no afecte el tiempo de búsqueda dentro de la base de datos.

- **Seguridad**

Garantizar la seguridad del sistema con respecto a la información y datos de todos sus clientes, manteniendo la confiabilidad de estos.

- **Fiabilidad**

La interfaz de la aplicación debe ajustarse a las diferentes dimensiones de los distintos dispositivos móviles.

- **Disponibilidad**

Para realizar los pedidos solo lo podrá hacer en el horario establecido por el negocio.

- **Mantenibilidad**

La aplicación debe ser fácil de utilizar para que sea administrada por los usuarios así no tengan experiencia en el manejo de aplicaciones móviles.

- **Portabilidad**

La aplicación será implantada bajo la plataforma Android e IOS

### **Requisitos comunes de las interfaces**

- **Interfaces de hardware**

Es necesario poseer dispositivos móviles funcionales

- **Interfaces de software**

En dispositivos Android debe ser superior a la versión 9.0

En dispositivos IOS debe ser superior a la versión 14.0

- **Interfaces de comunicación**

El servidor, cliente y aplicación se comunicarán entre si mediante protocolos y estándares en internet.

### **1.3 Justificación del requerimiento a satisfacer**

Desde la llegada de las actuales tecnologías y la creación de las diferentes plataformas, las grandes, medianas , medianas y pequeñas empresas e incluso pequeños emprendimientos han optado por una estrategia muy eficaz y útil como son las aplicaciones móviles de servicio de delivery, el uso de estas tecnologías han permitido que los clientes interactúen en tiempo real con el negocio [5] , el motivo del desarrollo de esta aplicación móvil es para mejorar la interacción entre el negocio con las personas que deseen adquirir un producto que se ofrece en el negocio La aplicación lo podrán utilizar los usuarios desde un Smartphone con el sistema operativo Android o iOS.

Para facilitar agilizar y facilitar los procesos de venta del negocio se considera la creación de la aplicación móvil denominada SOLOPANAS, para que los clientes tengan la facilidad de poder comprar sus productos preferidos desde la comodidad de su domicilio y así

eviten trasladarse hasta el negocio y evitar alguno inconveniente al momento de realizar un pedido en el negocio.

El sistema web contará con un panel general (Dashboard) donde se mostrarán todas las ventas realizadas y poder realizar filtros de búsqueda con la finalidad que se pueda obtener información relevante para la toma de decisiones y el crecimiento de la empresa.

## **2 CAPÍTULO II. DESARROLLO DEL PROTOTIPO**

### **2.1 Definición del prototipo**

El enfoque de este proyecto tiene como objetivo desarrollar una aplicación móvil que permita realizar las compras del negocio con un diseño intuitivo y fácil de usar para los usuarios finales, para su desarrollo se usará Node.js y el IDE Android Studio, con un backend basado en la nube a través de los servicios que ofrece MySQL. El proyecto se desarrolló con el objetivo de aumentar las ventas del negocio y facilitar al cliente realizar pedidos desde la comodidad de su hogar.

### **2.2 Fundamentación teórica del prototipo**

Para llevar a cabo un software de calidad se debe implementar herramientas significativas para su desarrollo, por lo tanto, para este proyecto se ha implementado tecnologías modernas, centradas al desarrollo de aplicaciones multiplataforma y elaboración de proyectos. Como se aprecia en la figura 1 se detalla cada una de estas.

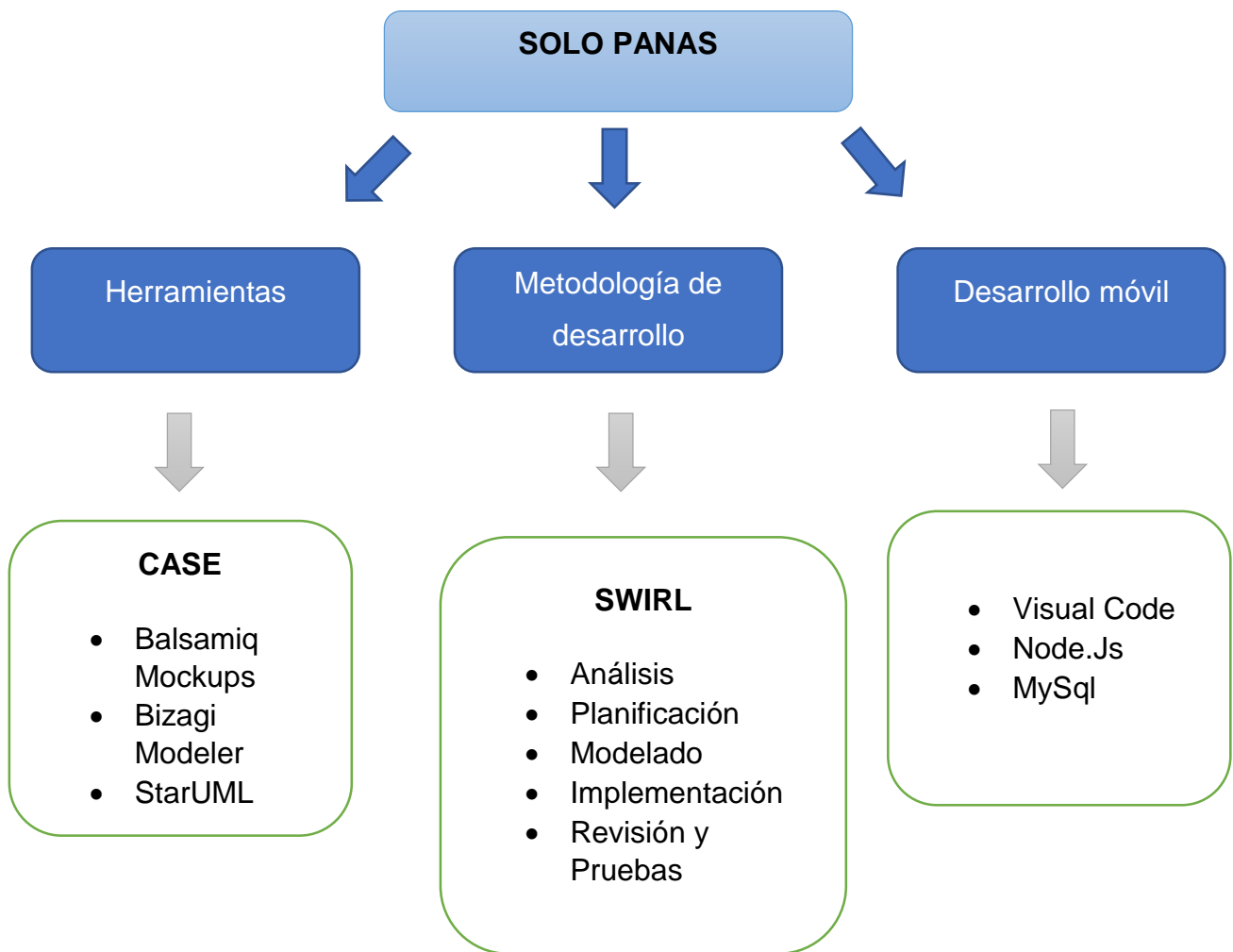


Figura 1 : Fundamentación teórica del prototipo

### 2.2.1 Arquitectura de software

Un modelo de arquitectura de software representa la estructura o las estructuras del sistema, consta de componentes de software, las propiedades visibles externamente y las relaciones entre ellas [6].

Los modelos de arquitectura de software tienen como finalidad soportar múltiples tareas de desempeño como son las operaciones del sistema, el tiempo de respuesta de las peticiones y el rendimiento [7] [8]. La arquitectura de software brinda una visión abstracta de alto nivel de sus componentes, y la relación entre ellos permitiendo plantear la reutilización y la evolución del código [9].

### 2.2.1.1 Arquitectura Cliente/servidor

La arquitectura cliente/servidor es un modelo para el desarrollo de sistemas de información en el que las transacciones se dividen en procesos independientes que cooperan entre sí para intercambiar información, servicios o recursos. Se denomina cliente al proceso que inicia el diálogo o solicita los recursos y servidor al proceso que responde a las solicitudes [10].

Dentro del modelo cliente servidor, los procesos de los clientes interactúan con los procesos de servidor individuales en equipos anfitriones que se encuentran separados con el objetivo que se pueda acceder a los recursos que estos brindan.



Figura 2: Arquitectura Cliente-Servidor [11]

### 2.2.1.2 Arquitectura tres capas

Para garantizar la escalabilidad del sistema es importante implementar una arquitectura basada en capas, de esta forma permite de manera eficaz el buen funcionamiento de las tareas del sistema de forma independiente sin implicar otras capas y que el rendimiento sea el más óptimo [12],

En la figura 2 se detalla el modelado tres capas.

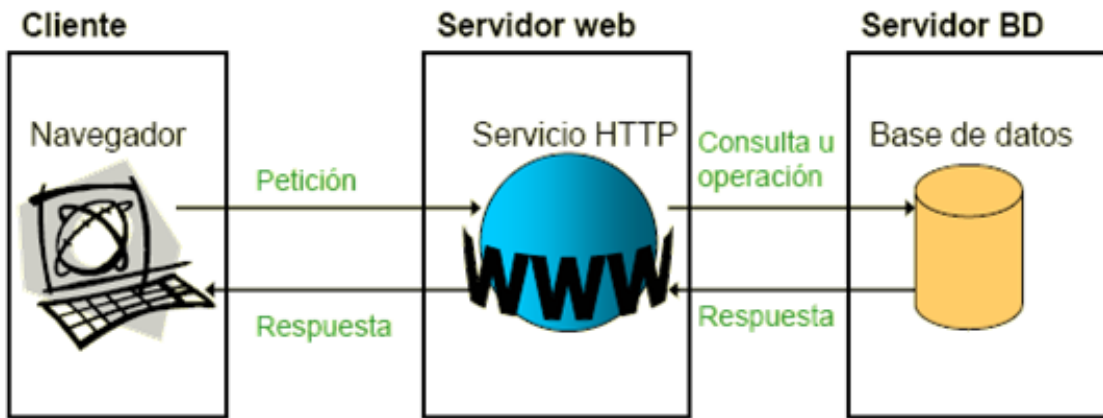


Figura 3: Modelado de tres capas

### Capa de datos

La capa de datos es la que se encarga en el almacenamiento y gestión que posea el sistema que puede ser tanto relacional como no relacional y que mantiene una conexión constante con la capa lógica [13].

### Capa de presentación

Conocida como capa de diseño es donde se visualizan todas las interfaces que el usuario podrá hacer uso del sistema. En esta capa se obtiene todas las peticiones que el usuario hará y luego enviar los datos adquiridos a la siguiente capa lógica [14].

### Capa lógica

Esta capa se encarga de abarcar toda la lógica en cada fase del sistema, la cual permite una interacción entre la capa de datos y presentación para que ambas se puedan comunicar internamente cuando el usuario vaya a realizar una petición [15].

### 2.2.2 MySql

MySQL es uno de los sistemas de gestión de bases de datos más populares para aplicaciones basadas en la web. Es gratuito y además está constantemente actualizado con nuevas características y medidas de seguridad. Existen una variedad de ediciones de pago destinadas al uso comercial pero la versión gratuita ofrece más velocidad y seguridad al no llevar incluidas una gran cantidad de características, esto puede ser positivo o negativo depende de las necesidades del administrador [16].

Este sistema de bases de datos permite seleccionar y manejar datos de una gran cantidad diferente de tipos de tablas, con una seguridad excelente y una gran fama de no desperdiciar los recursos del servidor. Sus principales ventajas son:



- Es gratuito.
- Ofrece una gran cantidad de características incluso en su versión gratuita.
- Existen multitud de interfaces de usuario y una gran comunidad detrás.
- Es compatible con otras bases de datos como Oracle.

Sus principales desventajas son:

- No existe soporte gratuito oficial para la versión gratuita.
- Puedes necesitar tiempo para automatizar funciones que otros sistemas llevan ya incluidos [17]

### **2.2.3 Herramientas CASE**

Las herramientas CASE son un conjunto de aplicaciones informáticas que se utilizan para automatizar las actividades del ciclo de vida del desarrollo del sistema (SDLC). Los administradores, analistas e ingenieros de proyectos de software utilizan las herramientas CASE para desarrollar sistemas de software.

Existen muchas herramientas CASE diferentes que se pueden utilizar para simplificar las diferentes etapas del ciclo de vida del desarrollo de software, como herramientas de análisis, herramientas de diseño, herramientas de gestión de proyectos, herramientas de gestión de proyectos, bases de datos, gestión de bases de datos, etc. Herramientas de base de datos, también se deben mencionar algunas herramientas de documentación [18].

#### ***2.2.3.1 Enterprise Architect***

Permite a las empresas evaluar y ajustar de manera proactiva las políticas y los sistemas para lograr los objetivos comerciales que monetizan las interrupciones comerciales relevantes. La noción y las tecnologías de modelado de EA provienen originalmente de la década de 1980. Con las crecientes necesidades de transformación digital, hoy EA se usa ampliamente en la industria como un proceso de cambio continuo impulsado por la tecnología para las empresas y toda nuestra sociedad [19].

Permite a las empresas modelar la TI y, por lo tanto, evaluar las necesidades de cambio, incluida la TI tradicional, servicios en la nube y sistemas integrados distribuidos [20].

#### ***2.2.3.2 Balsamiq Mockups***

Es de uso fácil y rápido para trabajar con visualizaciones, demostraciones y pruebas la cual permite la modificación de diseños de cómo se vería el software que se está

trabajando para poder llegar al producto del proyecto final. Tiene una interfaz accesible que se puede usar en Linux también en Mac OS X [21].

Una vez mencionada la metodología del proyecto, las herramientas Se introducirán Balsamiq Mockups, que nos beneficiarán para la creación de las vistas de interfaz que nos mostrarán los sistemas, como botones, enlaces, tablas, etc. En otras palabras, será la herramienta con la que podremos diseñar el modelo de lo que planeamos implementar en el sistema terminado [22].

### **2.2.3.3 Android Studio**

Este entorno de desarrollo fue presentado en el año 2013 en Google, desarrollar aplicaciones para Android no siempre fue tan conveniente como hoy. Cuando se lanzó Android 1.0 en algún momento de 2008, lo que los desarrolladores obtuvieron a través de un kit de desarrollo no era más que un montón de herramientas de línea de comandos y scripts de compilación de hormigas. Crear aplicaciones con vim, ant y algunas herramientas de línea de comandos no era tan malo si estabas acostumbrado a ese tipo de cosas, pero muchos desarrolladores no lo estaban. La falta de capacidades del entorno de desarrollo integrado (IDE), como la sugerencia de código, la configuración de proyectos y la depuración integrada, fue en cierto modo una barrera de entrada [23].

Creación de aplicaciones de Android usando Android Studio que admite dos lenguajes de programación, Android utilizar el lenguaje XML para el diseño de secuencias de comandos. Varios términos populares en la programación Android como Actividad, Fragmento es un contenedor para usuarios Interfaz (UI) en programación Android. Una aplicación de Android despertado de uno o varias actividades, puede constar de varios Frams. El sistema principal de mensajes en ejecución Android se llama Intents. Las intenciones consisten de la acción a ejecutar (Ver, Cambiar, Marcar, etc.) y datos Las intenciones se utilizan para iniciar actividades y comunicaciones entre partes del sistema Android [24].

### **2.2.3.4 JavaScript**

JavaScript es el lenguaje que proporcionó un sitio web dinámico que se comunica activamente con los usuarios. JavaScript se utiliza en las aplicaciones web actuales como lenguaje de secuencias de comandos del cliente y del lado del servidor [25].

JavaScript es relativamente fácil de utilizar y se la utiliza para realizar:

- Juegos basados en el navegador web, incluidos los juegos 2D y 3D
- Desarrollo de aplicaciones para móviles

- Presentaciones web [26]

### **2.2.3.5 Node JS**

Node es un entorno de ejecución de JavaScript asíncrono basado en eventos diseñado para crear aplicaciones de red escalables, como la característica principal que tenemos Node.js, le permite crear una aplicación web asíncrona y escalable basada en eventos. Node.js permanece inactivo hasta que una conexión activa una llamada de servicio, todo lo posible gracias al uso de JavaScript del lado del servidor [27].

Node.js (más específicamente, JavaScript) no es un lenguaje complicado. Sigue la base principios que siguen lenguajes de secuencias de comandos similares (como Ruby, Python y PHP), pero con un toque (como todos lo hacen). Node.js es lo suficientemente simple para que cualquier desarrollador lo recoja y comience a codificar en poco tiempo y, sin embargo, es lo suficientemente potente como para lograr casi cualquier cosa los desarrolladores pueden decidir [28]

Node.js es una opción particularmente sólida para desarrollar API, como está a punto de ver.

### **2.2.3.6 React Native**

React Native, un marco popular de código abierto, ha ganado una tracción significativa en la comunidad de desarrollo para crear aplicaciones móviles multiplataforma. React Native permite desarrolladores para escribir código en JavaScript y aprovechar una única base de código para crear aplicaciones compatibles con las plataformas iOS y Android. Numerosos estudios han exploró las ventajas de usar React Native para el desarrollo multiplataforma, incluida la reducción tiempo de desarrollo, rentabilidad y reutilización del código [29].

La identificación de React está diseñada de tal manera que se permite y admite el flujo de datos unidireccional que está aguas abajo. Identificación se requiere un flujo de datos bidireccional, es necesario implementar características adicionales. Esto se hizo como los componentes deben ser inmutables y los datos dentro de ellos no deben cambiar bajo ninguna circunstancia. Por lo tanto, escuchar los datos que vienen en una dirección solo se hace, no en la otra. De ahora en adelante, React.js es bien conocido por la generación de fuentes de datos canónicos que permanece sincronizado entre los componentes que le prestan atención. Como un Como resultado, demuestra ser uno de los mejores marcos para desarrollar aplicaciones interactivas basadas en la web [30].



*Figura 4: Representación React Native [31]*

### **2.2.3.7 Redux**

Redux es un contenedor de estado predecible para aplicaciones de JavaScript. La gestión estatal es una de los aspectos más difíciles del desarrollo de software. La mala gestión estatal es la fuente de la mayoría de los errores. Redux es una implementación simplificada de la arquitectura Flux de Facebook que es un marco Modelo-Vista-Controlador. Disminuye la complejidad mediante el uso de reductores. Los reductores de Redux son funciones sin efectos secundarios que calculan el estado de la aplicación [32].

Redux se basa en tres principios:

- El estado de la aplicación se almacena en un solo objeto. Redux almacena el estado en un solo JavaScript objeto para que sea más fácil mapear y pasar datos a través de toda la aplicación. La centralización del estado en un solo objeto también hace que el proceso de prueba y depuración más rápido.
- El estado de la aplicación es inmutable. En Redux, los estados no se pueden modificar. La única manera de cambiar el estado es proporcionar una acción. Las acciones son objetos JavaScript inmutables que describe los cambios de estado. Las acciones se ejecutan en un orden para evitar las condiciones.
- Los reductores especifican cómo la acción transforma el estado. Los reductores son funciones de JavaScript que crean un nuevo estado con el estado y la acción actuales dados. ellos centralizan mutaciones de datos y puede actuar sobre todo o parte del estado. Los reductores también se pueden combinar y reutilizado.

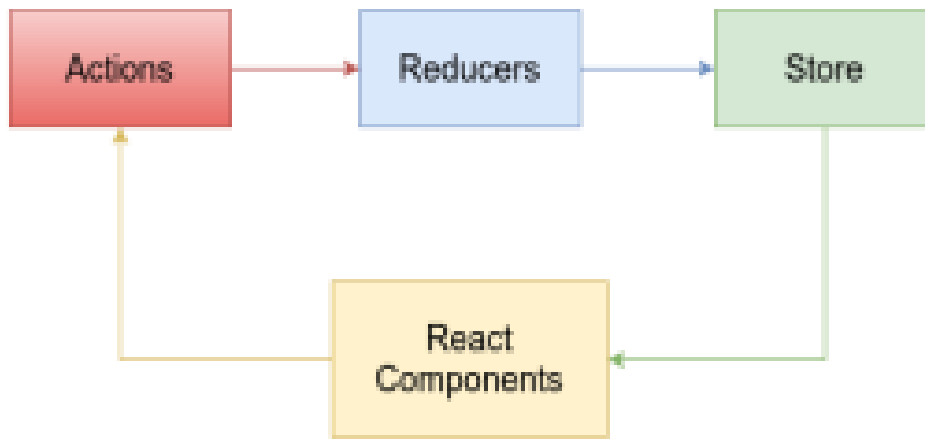


Figura 5: Flujo de datos Redux [33]

## 2.2.4 Metodología de desarrollo de Software

Las metodologías de desarrollo de software se clasifican en tradicionales y modernas [4]. En este trabajo se describen las características, ventajas y desventajas más relevantes de ambos tipos de metodologías según los reportes de la literatura. Inicialmente se presenta el método empleado para realizar la revisión, luego se presentan los resultados obtenidos y su respectivo análisis, y finalmente se exponen las conclusiones, incluyendo recomendaciones para trabajos futuro [34].

Una metodología de desarrollo de software es similar a una receta de cocina. De la misma manera que una receta le enseña a uno cómo cocinar una comida, un método de desarrollo de software enseña cómo construir un producto de software. Las metodologías aumentan los esfuerzos para mejorar la calidad de los productos al mejorar los procesos que producen los productos. Se requiere un estándar o método formalmente definido para controlar los procesos de desarrollo. Las diferentes metodologías de desarrollo de software tienen fortalezas y debilidades. La elección de qué método usar realmente depende de los objetivos que una empresa quiera alcanzar [35].



Figura 6: Ciclo de desarrollo del software [36]

### **2.2.4.1 Metodologías ágiles**

Las metodologías ágiles presentan como principal particularidad la flexibilidad, los proyectos en desarrollo son subdivididos en proyectos más pequeños, incluye una comunicación constante con el usuario, son altamente colaborativos y es mucho más adaptable a los cambios [37]. De hecho, el cambio de requerimientos por parte del cliente es una característica especial, así como también las entregas, revisión y retroalimentación constante. Entre las más notables metodologías de desarrollo ágil, se encuentran:

- Scrum (Muy popular en emprendimientos)
- Programación extrema (XP)
- Crystal Clear
- Mobile-D (ágil y extrema para móviles)
- Adaptive Software Development (ASD)
- Lean Development

El objetivo de la metodología ágil es dar solución a problemas que surgen después de la implementación y uso de un software o producto, teniendo en cuenta que las expectativas y necesidades de los consumidores son hoy en día más urgentes que frecuentes.

Una de las diferencias que posee con la metodología tradicional es que la metodología ágil está abierta a que sucedan errores debido a que durante su desarrollo permite que los errores que sucedan se vayan encontrando y pueda ser solucionado en el transcurso que se avanza el proyecto [38].

### **2.2.4.2 Metodologías híbridas**

Las metodologías en la gestión de los proyectos no se reducen a unas pocas, debido a la necesidad constante de optimizar los procesos dentro de los proyectos, se han desarrollado metodologías híbridas que permiten aprovechar los beneficios de los enfoques tradicionales en Cascada y otros más modernos como los ágiles. Sin embargo, las distintas combinaciones que se pueden generar a partir de estas metodologías híbridas están determinadas por las necesidades que cada proyecto o empresa puedan requerir. Para esto, la aplicación de las metodologías está determinada bien por las políticas de la organización o a su vez por la planificación de la Dirección de Proyectos, los cuales deben analizar aspectos internos y externos que puedan favorecer o no, a la puesta en marcha de este tipo de metodología de trabajo [39].

Tiene como objetivo realizar una gestión e implementación más flexible orientada a los resultados del proyecto y necesidades del cliente, buscando a su vez reducir tiempos de ejecución. Igualmente, el éxito de este enfoque está en mejorar continuamente los procesos, innovación que permitan crear ambientes colaborativos y dinámicos, y la flexibilidad de adaptación al cambio. Algunos de los beneficios que ofrece esta metodología son: fácil adaptación a las necesidades del proyecto permite reducir incertidumbres debido a la planeación previa de recursos y actividades, mayor control de los resultados además minimiza los riesgos debido a la constante interacción con los usuarios, entre otros [40].

#### **2.2.4.3 Metodología SWIRL**

En este proyecto la metodología que se escogió para la elaboración de la ampliación móvil es la SWIRL debido a que posee un marco de desarrollo con una guía que facilitará en el desarrollo de aplicaciones móviles.

La metodología SWIRL es un modelo que permite la iteración considerando cinco criterios tales como: costo, tiempo, calidad, volumen y comunicación que El modelo se hace de acuerdo a las necesidades de cada cliente, es decir Adaptar siempre los cambios necesarios para cada proyecto y cuando se considera el alcance.

Cada actividad en sus fases aporta agilidad al proceso de desarrollo, además de mejorar la comunicación del equipo de desarrollo con los clientes, que está incluida en la mayoría de los procesos. Este último beneficia tanto al producto como al proceso en el proyecto. En términos de producción, se garantiza que una aplicación web, incluido el cliente, cumplirá con todos los requisitos que ha establecido al inicio, así como también cumplirá con los requisitos de los proyectos orientados a la web [41].

Esta metodología posee cinco fases para el ciclo de vida sobre el proyecto de software, a continuación, en la figura 7 se detalla las fases de la metodología:

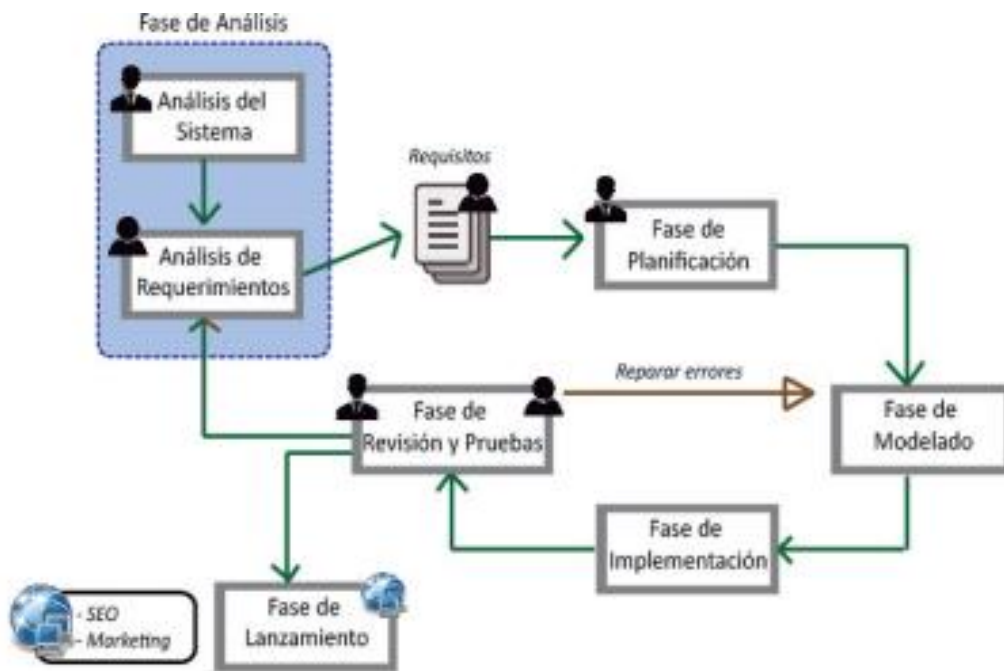


Figura 7: Ciclo de vida de SWIRL [42]



## **2.3 Objetivos del prototipo**

### **2.3.1 Objetivo general**

- Desarrollar una aplicación para la gestión en el proceso de ventas de una empresa utilizando NodeJS y React Native, para optimizar la eficiencia y la precisión en la gestión de pedidos.

### **2.3.2 Objetivos específicos**

- Recolectar los requerimientos de la empresa para el desarrollo de la aplicación mediante la implementación de historias de usuario.
- Investigar mediante fuentes bibliográficas las tecnologías a implementarse en el desarrollo de la aplicación.
- Implementar un Dashboard que permite al administrador visualizar los productos, realizar búsquedas inteligentes, monitorear las ventas para hacer un seguimiento del negocio y permita realizar la toma de decisiones.
- Validar el correcto funcionamiento de la aplicación a través de la ISO/IEC 9126 para garantizar el funcionamiento y desarrollo de un sistema de calidad

## **2.4 Diseño del prototipo**

### **2.4.1 FASE DE ANÁLISIS**

#### ***2.4.1.1 Reconocimiento general del sistema***

Como es de conocimiento de la problemática surgida desde la pandemia se vieron afectados los negocios en torno a sus ventas por diferentes factores por cual razón los negocios tuvieron la necesidad por optar por nuevas modalidades en base a las compras, siendo una solución la compra en línea la opción más utilizada en estos tiempos.

El motivo del desarrollo de esta aplicación móvil es para mejorar la interacción entre el negocio con las personas que deseen adquirir un producto que se ofrece en el negocio, y el cliente pueda realizar su pedido desde la comodidad de su hogar sin exponerse a riesgos, además que podrá pagar en efectivo al momento de la entrega o en línea con su tarjeta de preferencia.

#### ***2.4.1.2 Stakeholders***

Al mencionar a las partes interesadas o las personas que intervienen dentro de la aplicación hacemos énfasis a los stakeholders los mismos que durante el proceso ayudan a recoger los requerimientos de software, estos son analizados tanto para el éxito o fracaso de proyectos de software, de la misma forma se busca ir de la mano de una

metodología la que ayudará a la búsqueda de datos y así caracterizar y obtener los mejores stakeholders [43], [44].

Para el desarrollo de este proyecto, se detalla en la tabla 1 los stakeholders.

Tabla 1: Matriz de Stakeholders

MATRIZ GENERAL DE STAKEHOLDERS			
Nombre	Roles	Profesión	Responsabilidad
<b>Ing. Milton Valarezo Pardo</b>	Docente/ Gerente de proyecto	Ingeniero de Sistemas	Limitaciones del sistema y revisión de entregables.
<b>Davis Chalco Morocho</b>	Desarrollador	Estudiante	Planificación. Análisis. Diseño. Desarrollo. Implementación. Pruebas y Lanzamiento.
<b>Rafael Lopez Jimenez</b>	Usuario/Cliente	Administrador	Requerimientos

### 2.4.1.3 Alcance del proyecto

En base a la creación de la aplicación se busca permitir al cliente ahorrar tiempo al momento de realizar los pedidos de sus productos dentro del negocio, mediante la visualización de la información detallada de los productos a adquirir, además de poder observar todos sus pedidos realizados de tal manera que el proceso de compra sea eficiente y seguro.

De acuerdo con la tabla 2 se detalla las funcionalidades del sistema.

Tabla 2: Funcionalidades del sistema

Funcionalidades de SOLOPANAS
Permite la autenticación mediante email para que el usuario pueda registrar e iniciar sesión.
El usuario podrá agregar los datos de facturación en caso de solicitar la factura en su pedido.

El usuario podrá conocer el perfil del negocio como su dirección, ubicación, productos, teléfono, horario de atención, etc.
El usuario tendrá la opción de buscar el producto que necesite mediante el buscador que se encuentra en la pantalla principal.
El usuario podrá editar su perfil en caso lo requiera.
El usuario podrá visualizar sus pedidos realizados.
El usuario podrá conocer el total de la compra.
El usuario podrá observar el estado de su pedido.
El usuario podrá cancelar el valor total en efectivo o en `pago en línea con tarjeta

Las limitaciones del sistema se detallan en la tabla 3.

*Tabla 3: Limitaciones del sistema*

Limitaciones de SOLOPANAS
El usuario no podrá escribir al repartidos desde la aplicación
El usuario no puede realizar pagos en efectivo con billetes de denominaciones de 50 y 100 dólares.
El usuario no podrá realizar pedidos para otra ciudad.

#### **2.4.1.4 Análisis de la factibilidad del sistema**

La factibilidad del sistema son todos los recursos que se tiene a disposición para poder llevar a cabo los objetivos establecidos, por tal razón se deben analizar tanto la factibilidad operativa, técnica y económica [45].

##### **2.4.1.4.1 Factibilidad Operativa**

La factibilidad operativa abarca todos los recursos que el proyecto necesita tales como recursos humanos o tecnológicos, la cual se debe garantizar la viabilidad de estos para poder llevar a cabo el software [46]

De acuerdo con la tabla 4 se detalla la factibilidad operativa del proyecto.

Tabla 4: Tabla de factibilidad Operativa

Factibilidad Operativa		
No.	Actividad	Prioridad
1	Se tiene los componentes informáticos precisos para el desarrollo de la aplicación	Alta
2	Poseer conexión a internet para su buen funcionamiento	Alta
3	Tener conocimientos técnicos sobre el desarrollo de aplicaciones móviles	Alta
4	Compra de las cuentas de desarrollador de las principales tiendas de Play Store y App Store	Alta

#### 2.4.1.4.2 Factibilidad Técnica

En esta fase de este proyecto se usarán herramientas tecnológicas que permitirá el desarrollo de la aplicación que funcionará en base a placativos como Android, espacio en la nube etc. [47].

La factibilidad técnica del proyecto se encuentra detallada en la tabla 5.

Tabla 5: Tabla de Factibilidad Técnica

Hardware		
Cantidad	Descripción	Costo (\$)
1	Laptop Intel Core i7 6ta generación, 8 Gb memoria RAM, 512 SSD, 1 TB HDD	\$1100.00
1	Teléfono móvil iPhone XS	\$400
Software		
Cantidad	Descripción	Costo (\$)
1	Node.js	\$0.00
1	Android Studio	\$0.00
1	Suite Microsoft Office	\$0.00
1	Bizagi Modeler	\$0.00

1	StartUML	\$0.00
1	Balsamiq Mockups	\$0.00
1	React Native	\$0.00
1	Appetize.io	\$0.00
<b>Total de Costos</b>		<b>\$1500.00</b>

#### 2.4.1.4.3 Factibilidad Económica

En esta fase se enfoca todos los recursos económicos con los que posee el equipo de trabajo para lograr el desarrollo de la aplicación y que pueda funcionar.

Según la tabla 6 se detalla la factibilidad económica del proyecto [48]

Tabla 6: Tabla de Factibilidad Económica

Tipo Recurso	Tipo Elemento	Descripción	Unidad Medida	Cantidad	P. Unitario	Costo
HUMANO	Ing. Sistemas	Davis Chalco	Horas	560	7.50	4,200.00
TECNOLOGICO	Software	Node.Js	Unidad	1	0	0
TECNOLOGICO	Software	React Native	Unidad	1	0	0
TECNOLOGICO	Software	Bizagi Modeler	Unidad	1	0	0
TECNOLOGICO	Software	StartUML	Unidad	1	0	0
TECNOLOGICO	Software	Balsamiq Mockups	Unidad	1	0	0
TECNOLOGICO	Software	Android Studio	Unidad	1	0	0
TECNOLOGICO	Software	Appetize.io	Unidad	1	0	0
TECNOLOGICO	Hardware	Laptop Core i7, 512 SSD, 8 RAM	Unidad	1	1,100.00	1,100.00
TECNOLOGICO	Hardware	Dispositivo IOS	Unidad	1	400.00	400.00
TECNOLOGICO	Servicios	App Store	Unidad	1	100.00	100.00

TECNOLOGICO	Servicios	Play Store	Unidad	1	25.00	25.00
TECNOLOGICO	Servicios	Internet	Meses	3	30.00	90.00
MATERIALES	Bienes	Escritorio	Unidad	1	80.00	80.00
MATERIALES	Bienes	Sillas	Unidad	1	60.00	60.00
MATERIALES	Bienes	Oficina	Meses	3	100.00	300.00
<b>COSTO</b>					\$	6,355.00
<b>IMPREVISTOS</b>					10%	635.50
<b>PRESUPUESTO TOTAL</b>					\$	6,990.50

#### 2.4.1.5 Requerimientos funcionales

Estos requerimientos son los que permiten el funcionamiento del software por lo tanto esto influye en si la solución es aceptada o rechazada por parte de los usuarios finales, se corre los riesgos de que el proyecto falle en caso de que estos requerimientos sean deficientes.

De acuerdo con la tabla 7 se encuentra detallado los requerimientos funcionales del sistema.

Tabla 7: Requerimientos Funcionales

REQUERIMIENTOS FUNCIONALES	
Código	Descripción
RF – 01	Permite el ingreso al sistema mediante autenticación a los usuarios
RF – 02	Permite al usuario registrarse a través de su correo electrónico
RF – 03	Administración de datos de facturación
RF – 04	El usuario podrá visualizar el perfil del negocio
RF – 05	El usuario podrá observar los productos disponibles para poder seleccionarnos.
RF – 06	Permite al dueño del negocio la gestión de los productos (Editar, agregar y eliminar)
RF – 07	El usuario podrá realizar su pedido seleccionando los productos y la cantidad deseada.

### 2.4.1.6 Requerimientos no funcionales

Estos requerimientos son todos aquellos requisitos indirectos del sistema pero que no son importantes ni indispensables y que son utilizados para juzgar las funciones del sistema [49].

Tabla 8: Requerimientos no funcionales

REQUERIMIENTOS NO FUNCIONALES	
Código	Descripción
RNF – 01	La aplicación debe ser intuitiva para el usuario.
RNF – 02	La aplicación de ajustar a las diversas dimensiones de los dispositivos móviles
RNF – 03	El tiempo de carga y búsqueda deben ser rápidos

## 2.4.2 FASE DE PLANIFICACIÓN

### 2.4.2.1 Historias de usuario

Las historias de usuario sirven para mostrar la presentación de las funciones del software de forma más explícita y general e informal, tiene como objetivo proporcionar valor al cliente de acuerdo a las funciones [50].

Tabla 9: Historia de usuario (Inicio de Sesión)

Historia de usuario	
<b>Numero:</b> 1	<b>Usuario:</b> Usuario Final
<b>Nombre de la historia:</b> Inicio de sesión en la aplicación	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Baja
<b>Requerimiento funcional:</b> RF- 01	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Davis Chalco	
<b>Descripción:</b> La aplicación móvil debe contar con un inicio de sesión a través de su correo y contraseña para los clientes, además de un face id para que pueda ingresar y poder realizar sus pedidos	
<b>Observaciones:</b> Mostrar un mensaje de error cuando los datos ingresados al inicio de sesión no se encuentren en la base de datos. La aplicación permitirá el ingreso a los usuarios que quieran realizar un pedido del negocio y podrá visualizar todos los productos disponibles.	

Tabla 10: Historia de usuario (Registro de usuario)

Historia de usuario	
<b>Numero:</b> 2	<b>Usuario:</b> Usuario Final
<b>Nombre de la historia:</b> Registro de Usuario	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Baja
<b>Requerimiento funcional:</b> RF- 02	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Davis Chalco	
<b>Descripción:</b> En lo que respecta al registro de usuario tiene que ingresar los Nombres y Apellidos de la persona, además del correo electrónico su respectiva contraseña.	
<b>Observaciones:</b> Los usuarios que ingresen datos erróneos se presentaran un mensaje para que pueda corregir los datos. La aplicación permitirá recuperar la contraseña a través del correo electrónico registrado.	

Tabla 11: Historia de usuario (Administración de datos de Facturación)

Historia de usuario	
<b>Numero:</b> 3	<b>Usuario:</b> Usuario Final
<b>Nombre de la historia:</b> Administración de datos de facturación	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Baja
<b>Requerimiento funcional:</b> RF- 03	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Davis Chalco	
<b>Descripción:</b> El usuario tiene la opción de solicitar una factura al momento de solicitar el pedido para ello debe registrar sus datos como cedula o RUC, nombres y apellidos completos, dirección, correo electrónico y teléfono.	
<b>Observaciones:</b> La aplicación permitirá realizar cualquier cambio una vez que haya registrado los datos	



Tabla 12: Historia de usuario (Perfil del Negocio)

Historia de usuario	
<b>Numero:</b> 4	<b>Usuario:</b> Usuario Final
<b>Nombre de la historia:</b> Perfil de Negocio	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Baja
<b>Requerimiento funcional:</b> RF- 04	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Davis Chalco	
<b>Descripción:</b> El empleador podrá visualizar los datos relevantes del negocio tales como: nombre, dirección, horario de atención, teléfono, servicio que oferta.	
<b>Observaciones:</b> La aplicación permitirá visualizar los datos generales del negocio.	

Tabla 13: Historia de usuario (Visualización de los productos)

Historia de usuario	
<b>Numero:</b> 5	<b>Usuario:</b> Usuario Final
<b>Nombre de la historia:</b> Visualización de los productos	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Baja
<b>Requerimiento funcional:</b> RF- 05	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Davis Chalco	
<b>Descripción:</b> En el apartado de productos el cliente podrá visualizar toda la variedad de productos donde se mostrará una descripción, la cantidad y el precio disponible de los productos.	
<b>Observaciones:</b> Se visualizará toda la información del producto seleccionado, además existe un buscador de productos.	

Tabla 14: Historia de usuario (Gestión de productos)

Historia de usuario	
<b>Numero:</b> 6	<b>Usuario:</b> Usuario Final
<b>Nombre de la historia:</b> Gestión de los productos	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Baja
<b>Requerimiento funcional:</b> RF- 06	<b>Iteración asignada:</b> 1

<b>Programador responsable:</b> Davis Chalco
<b>Descripción:</b> Como administrador tendrá habilitado todas las funciones como agregar, editar y eliminar los productos del negocio
<b>Observaciones:</b> La aplicación permitirá la edición de información y visualizar los productos.

Tabla 15: Historia de usuario (Gestión de Pedidos)

Historia de usuario	
<b>Numero:</b> 7	<b>Usuario:</b> Usuario Final
<b>Nombre de la historia:</b> Gestión pedidos	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Baja
<b>Requerimiento funcional:</b> RF- 07	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Davis Chalco	
<p><b>Descripción:</b> En esta sección el usuario podrá agregar productos que necesita y conocerá el subtotal a pagar, así como el listado de los productos agregados.</p> <p>Luego deberá seleccionar la dirección de la entrega, una vez registrada la dirección el usuario podrá conocer el valor total a pagar.</p>	
<p><b>Observaciones:</b> El usuario deberá seleccionar el método de pago, estará disponible la opción de efectivo y pago en línea con tarjeta; en caso de que el usuario necesite pagar con un billete puede escribir la denominación del mismo para que el repartidor lleve el cambio necesario y así evitar contratiempos.</p>	

#### 2.4.2.2 Gestión de cronograma

La gestión de cronograma del proyecto es importante para conocer las actividades que se sigue en la gestión del proyecto, para lo cual se detallan las actividades y el tiempo estimado para su ejecución además de definir el tiempo para la presentación del proyecto, lo cual se la visualiza en la figura 8

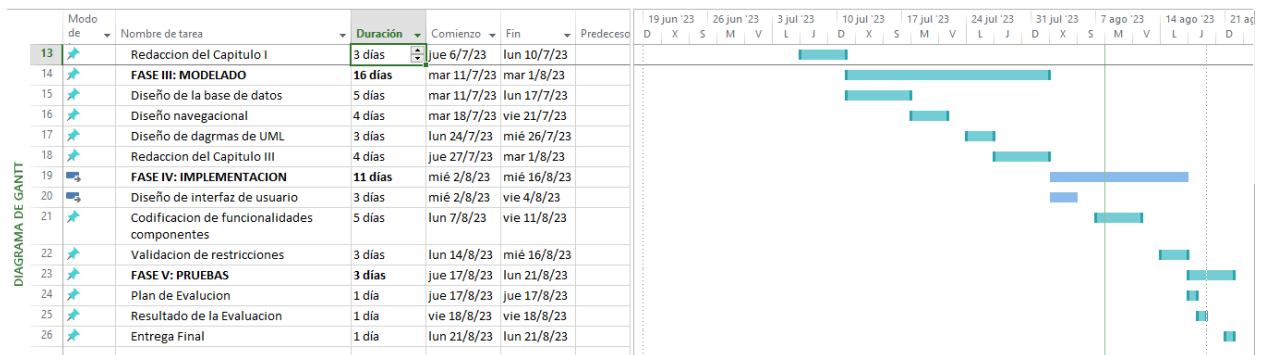
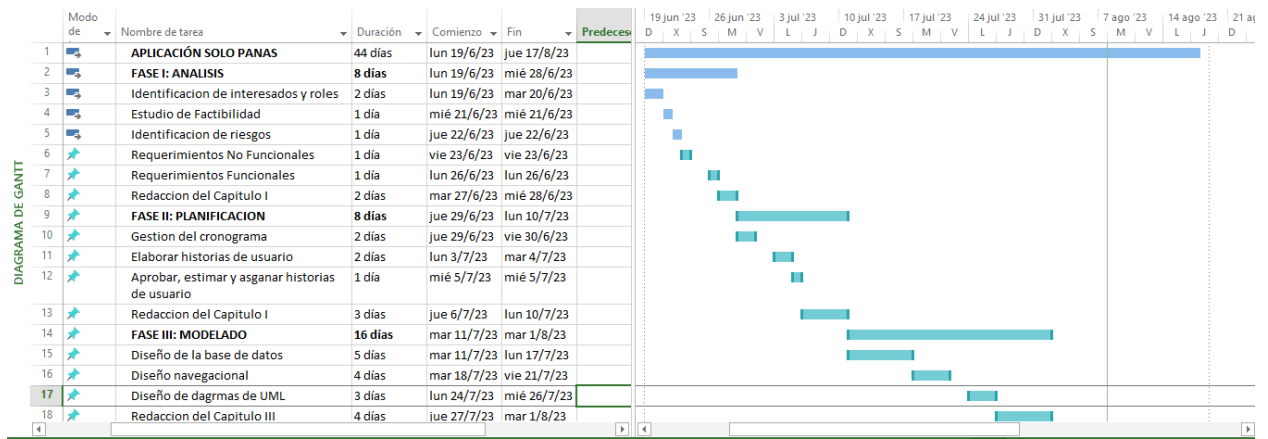


Figura 8: Gestión del Cronograma

### 2.4.3 FASE DE MODELADO

Dentro de esta fase de la metodología SWIRL

#### 2.4.3.1 Diseño de base de datos

De acuerdo con la figura 6 se detalla el diseño de base de datos.

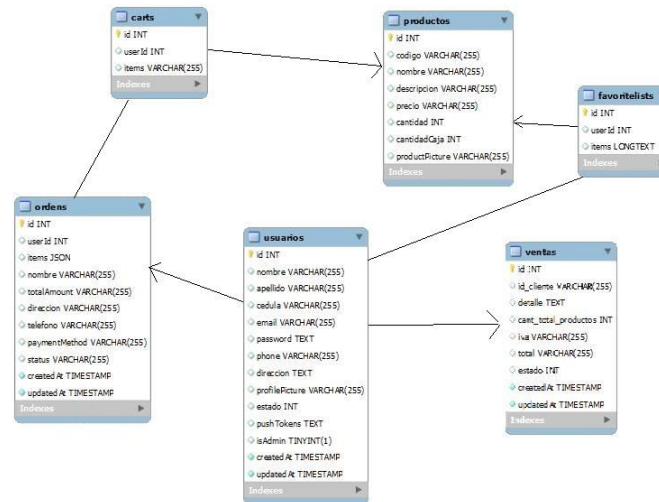


Figura 9: Diseño de la Base de datos

#### 2.4.3.2 Diseño navegacional

Dentro de la metodología SWIRL el diseño navegacional permite desarrollar una topología que detalla todas las funcionalidades que el usuario podrá acceder con las tareas unificadas que contará la aplicación, en la figura 10 se detalla la estructura del diseño.

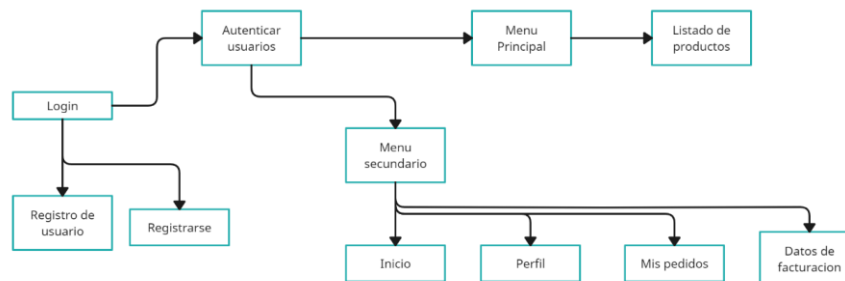


Figura 10: Diseño Navegacional

En la figura 10 se puede observar las funcionalidades de las aplicaciones a las cuales el cliente final tendrá acceso, para lo cual en primer lugar deberá autenticarse o registrarse para poder disfrutar de todas las opciones de la aplicación.

### 2.4.3.3 Diseño de interfaces

Para el desarrollo de interfaces se utilizó la herramienta Balsamiq Mockups la cual nos permite crear modelos similares a cómo será la representación final de la aplicación móvil en los dispositivos móviles, a continuación, se mostrarán los prototipos diseñados.

#### Módulo 1 – Pantalla principal

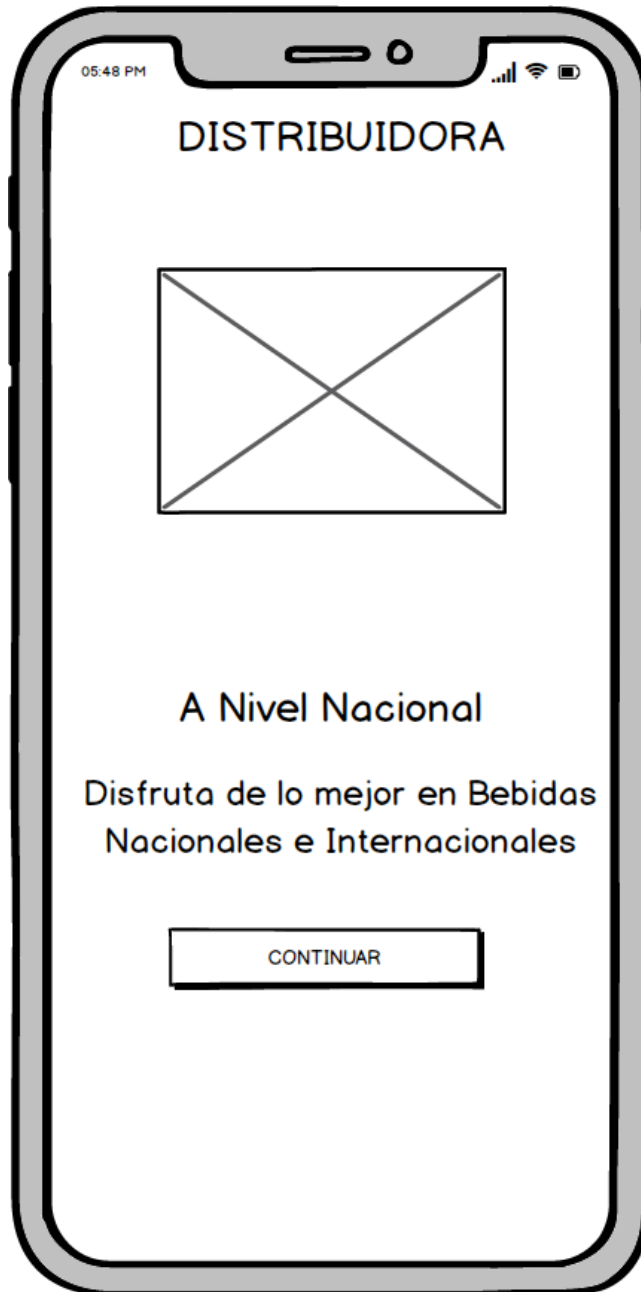


Figura 11: Modelado Menú Principal

La figura 11 nos muestra el nombre e información del negocio donde se debe hacer click para poder iniciar sesión o registrarse.

## Módulo 2 – Registro de usuarios

The image shows a mobile application wireframe for a registration screen. At the top, the status bar displays the time '05:48 PM' and icons for signal strength, Wi-Fi, and battery. The main title 'REGISTRO' is centered at the top of the screen. Below the title, there are six input fields, each represented by a rounded rectangle with a label inside: 'Usuario', 'Nombres completos', 'Email', 'Contraseña', and 'Confirme su contraseña'. At the bottom of the form is a rectangular button labeled 'REGISTRO'.

*Figura 12: Modelado Registro de Usuario*

La figura 12 nos indica el proceso para registrar un nuevo usuario por cual se le solicitará información básica de la misma que servirá para que pueda ingresar y hacer uso a la aplicación.

## Módulo 3 – Inicio de sesión

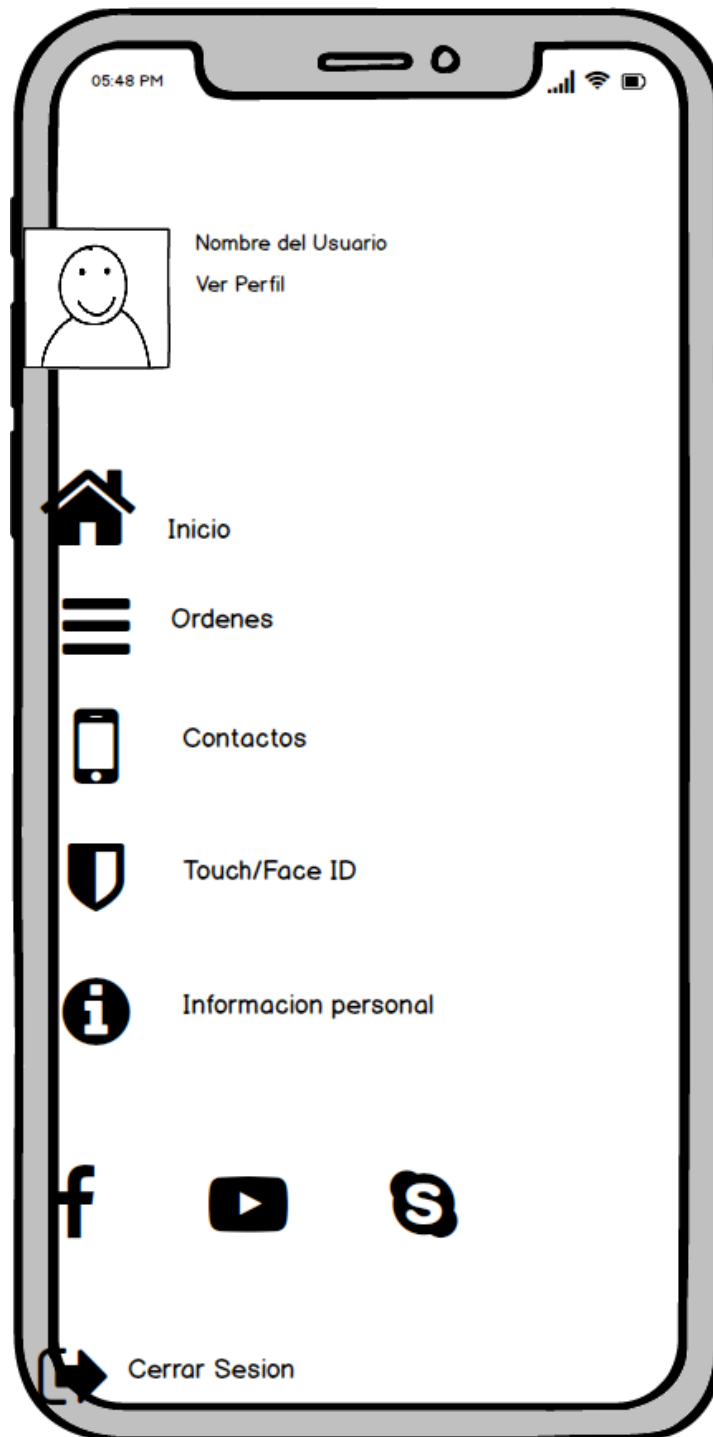


*Figura 13: Modelado Inicio de Sesión*

La figura 13 nos indica el proceso para iniciar sesión donde se debe ingresar el correo y contraseña anteriormente registrados, además que se puede iniciar sesión con el face id o huella digital, además que se puede reestablecer la contraseña en caso de haberse olvidado.

## Módulo 4 – Ventana Principal del Usuario

Figura 14: Modelado Ventana Principal

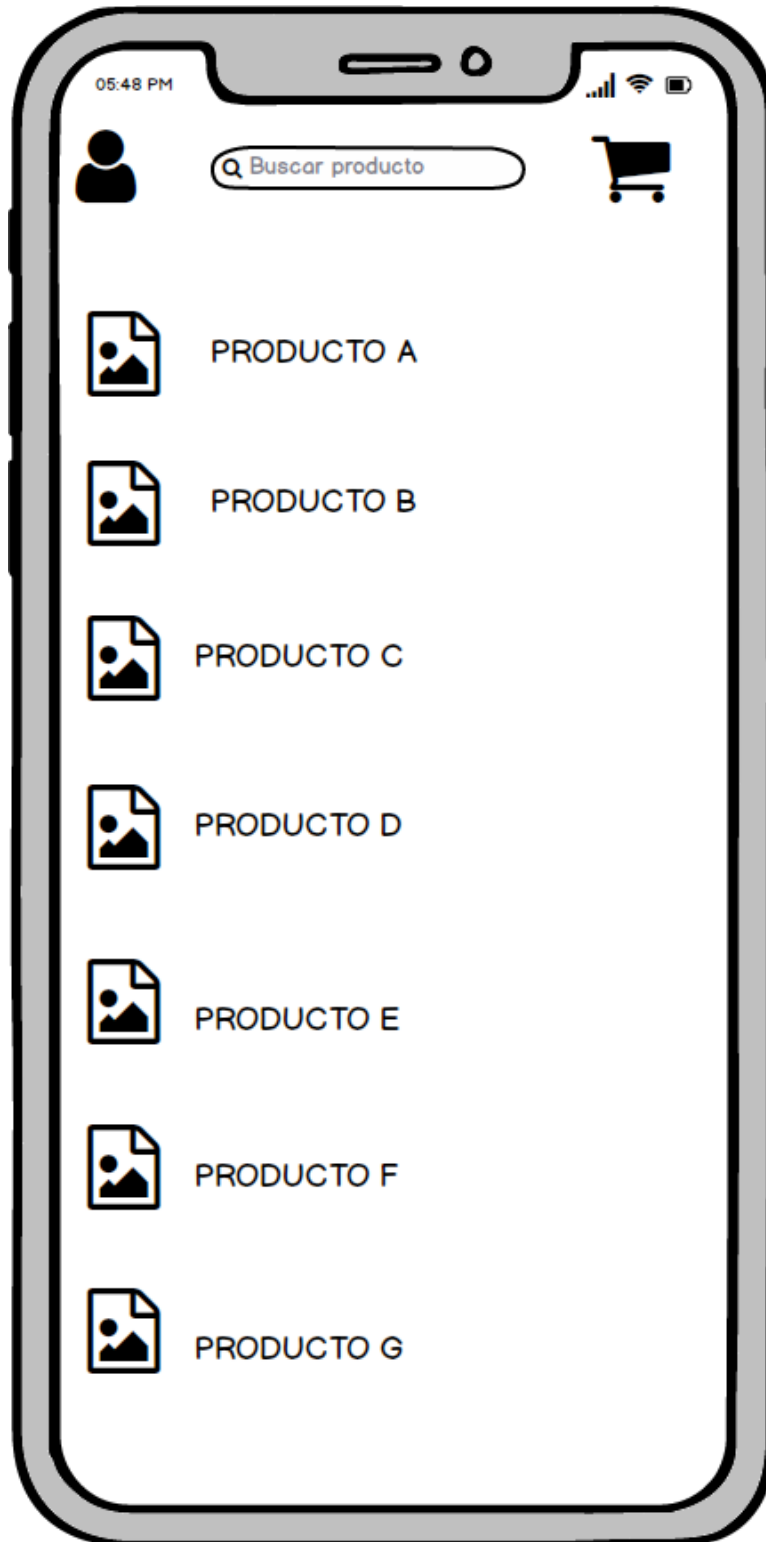


La figura 14 nos presenta la ventana principal que se muestra al usuario, donde se muestra la información del usuario, además de las opciones de ver sus órdenes, contactos, además que se puede cerrar sesión e iniciar con otra cuenta.



## Módulo 5 – Buscar Productos

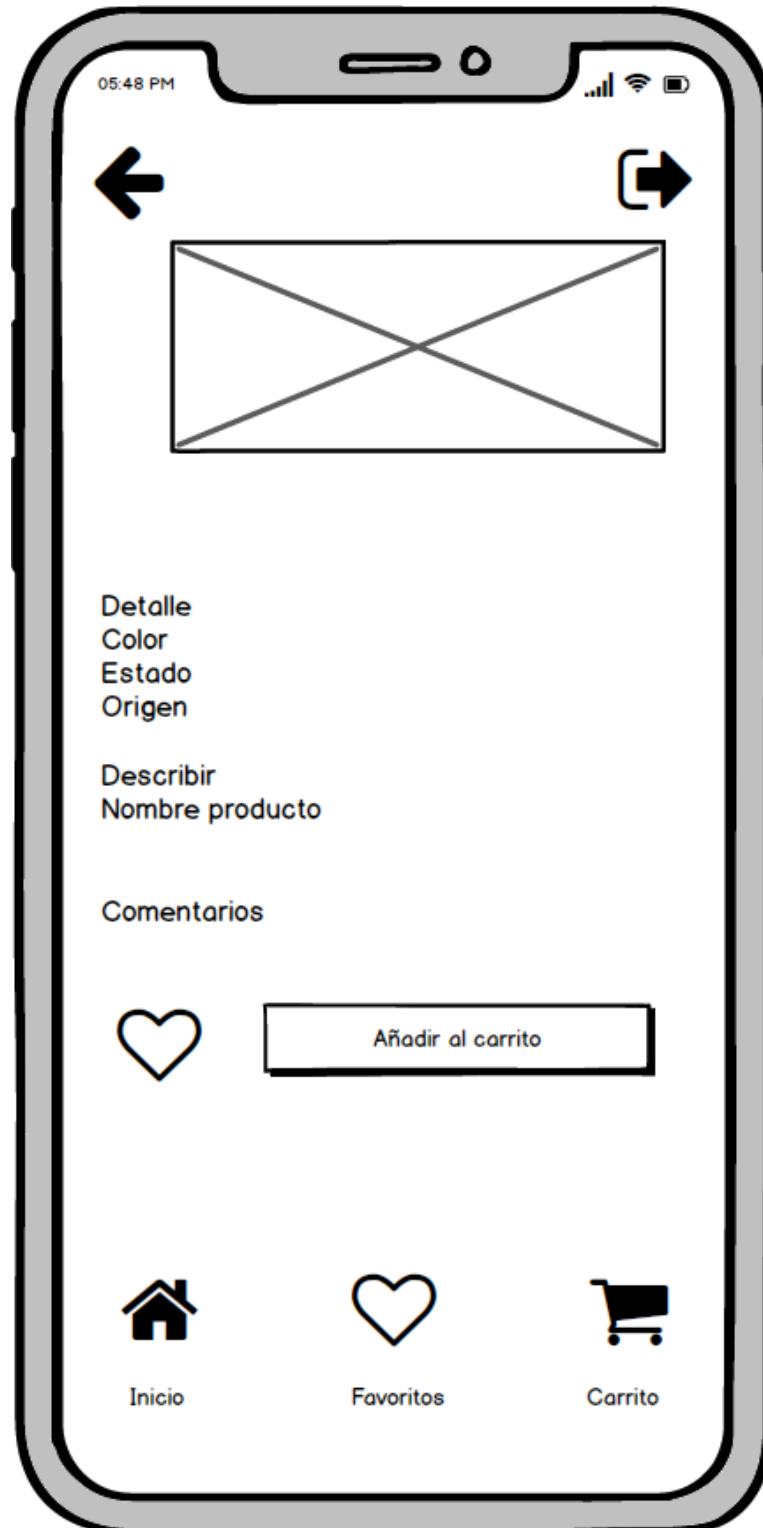
Figura 15: Modelado Productos disponibles



La figura 15 muestra la interfaz que el usuario visualizará al momento de ver los productos disponibles del negocio, cuenta con un buscador que permitirá encontrar un producto más rápido.

## Módulo 6 – Detalle Productos

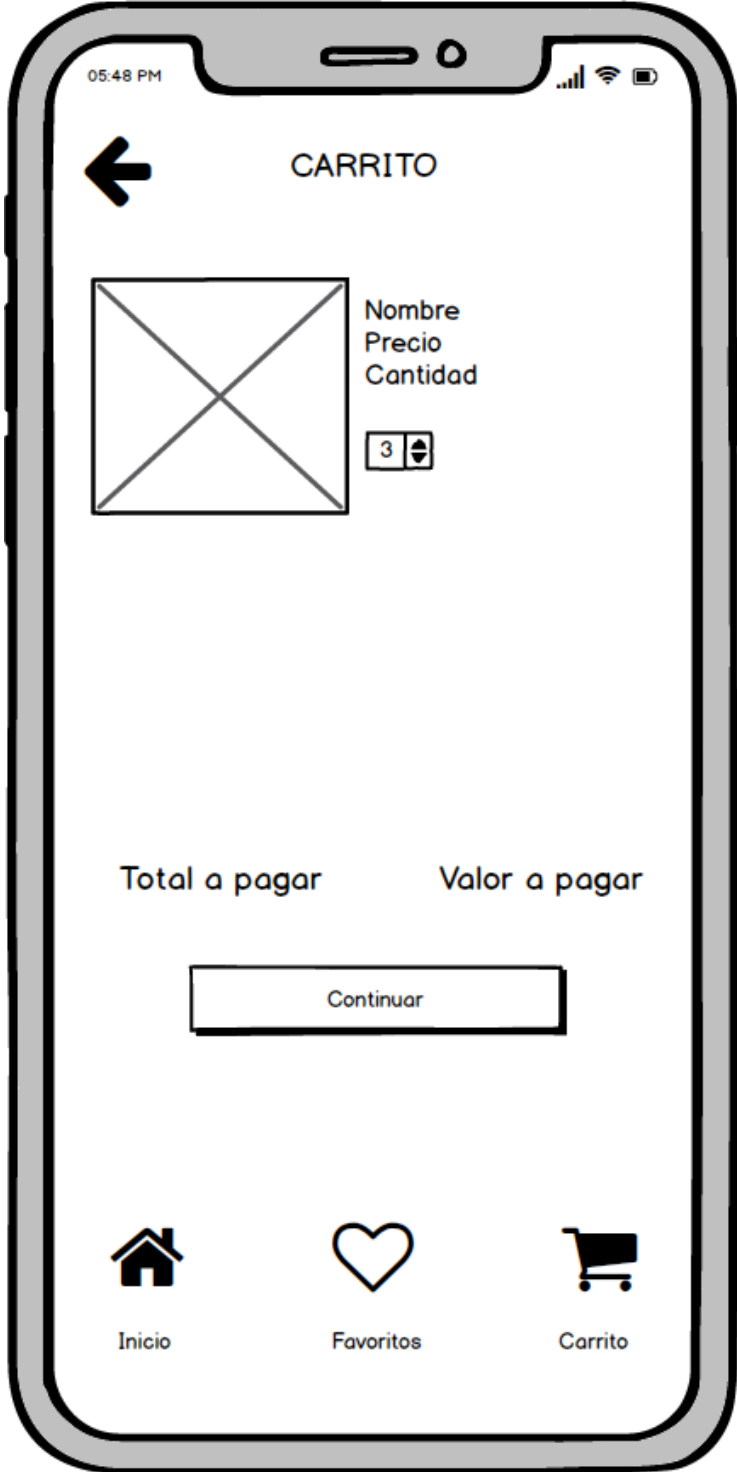
Figura 16: Modelado Detalle del producto



La figura 16 muestra la interfaz que el usuario visualizará el producto donde constará de una imagen de este, su nombre, color, origen, estado del producto. Donde el cliente selecciona la cantidad y podrá añadir al carrito de compras para proceder con el pago.

# Módulo 7 – Carrito de Compras

Figura 17: Modelado carrito de compras



La figura 17 muestra la interfaz que el usuario visualizará los productos seleccionados para la compra con el nombre y descripción de cada uno, además del valor total a cancelar.

## Módulo 8 – Dirección de entrega

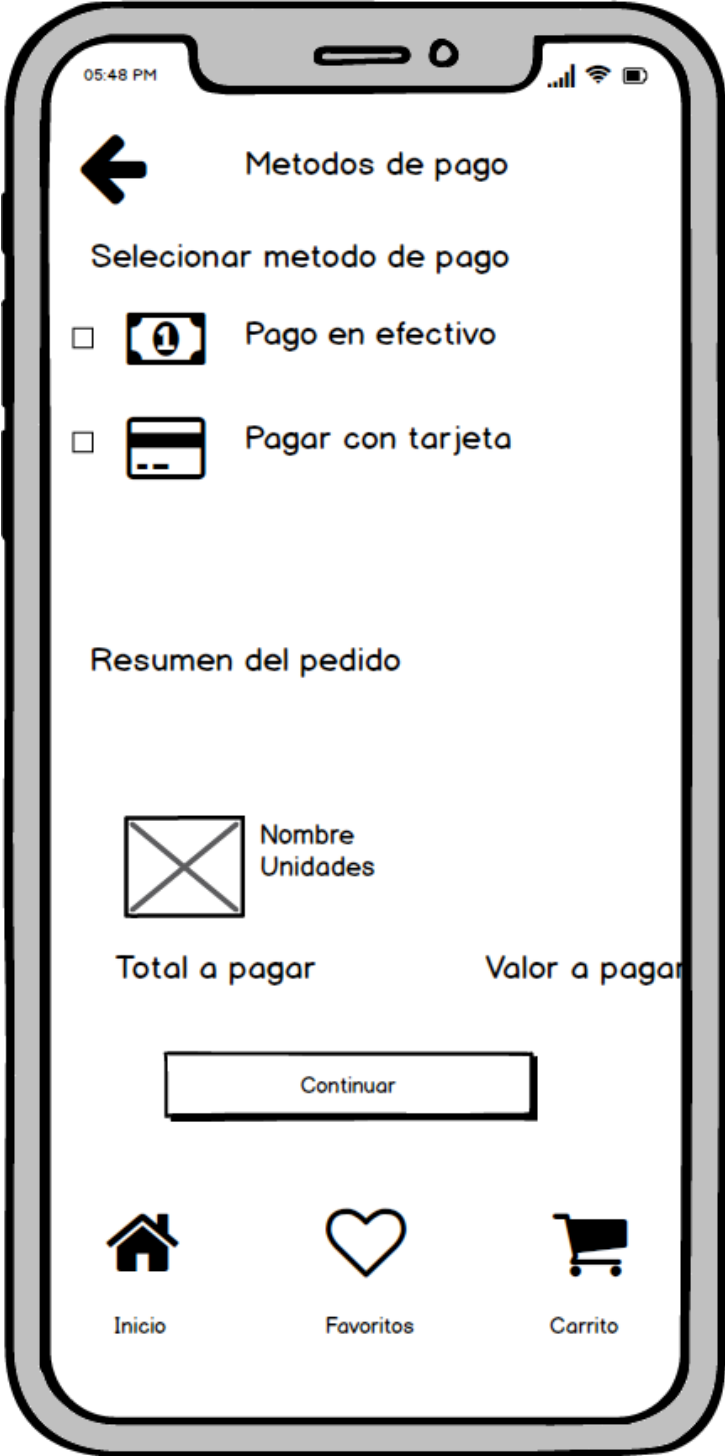
Figura 18: Modelado Dirección de Entrega

The image shows a mobile application interface for entering delivery details. At the top, the status bar displays '05:48 PM', signal strength, Wi-Fi, and battery icons. Below the status bar is a back arrow and the title 'Dirección de entrega'. The main section is titled 'Detalles de envío' and contains three rounded rectangular input fields labeled 'Nombre Completo', 'Numero de telefono', and 'Direccion'. Below these is a square icon with an 'X' and the text 'Nombre Unidades'. At the bottom of the form area, there are two labels: 'Total a pagar' and 'Valor a pagar'. A large rectangular button labeled 'Continuar' is positioned below the labels. At the very bottom of the screen is a navigation bar with three icons: a house for 'Inicio', a heart for 'Favoritos', and a shopping cart for 'Carrito'.

La figura 18 muestra la interfaz que el usuario deberá agregar sus datos personales para él envió, los cuales servirán para la facturación de la orden.

**Módulo 9 – Método de pago**

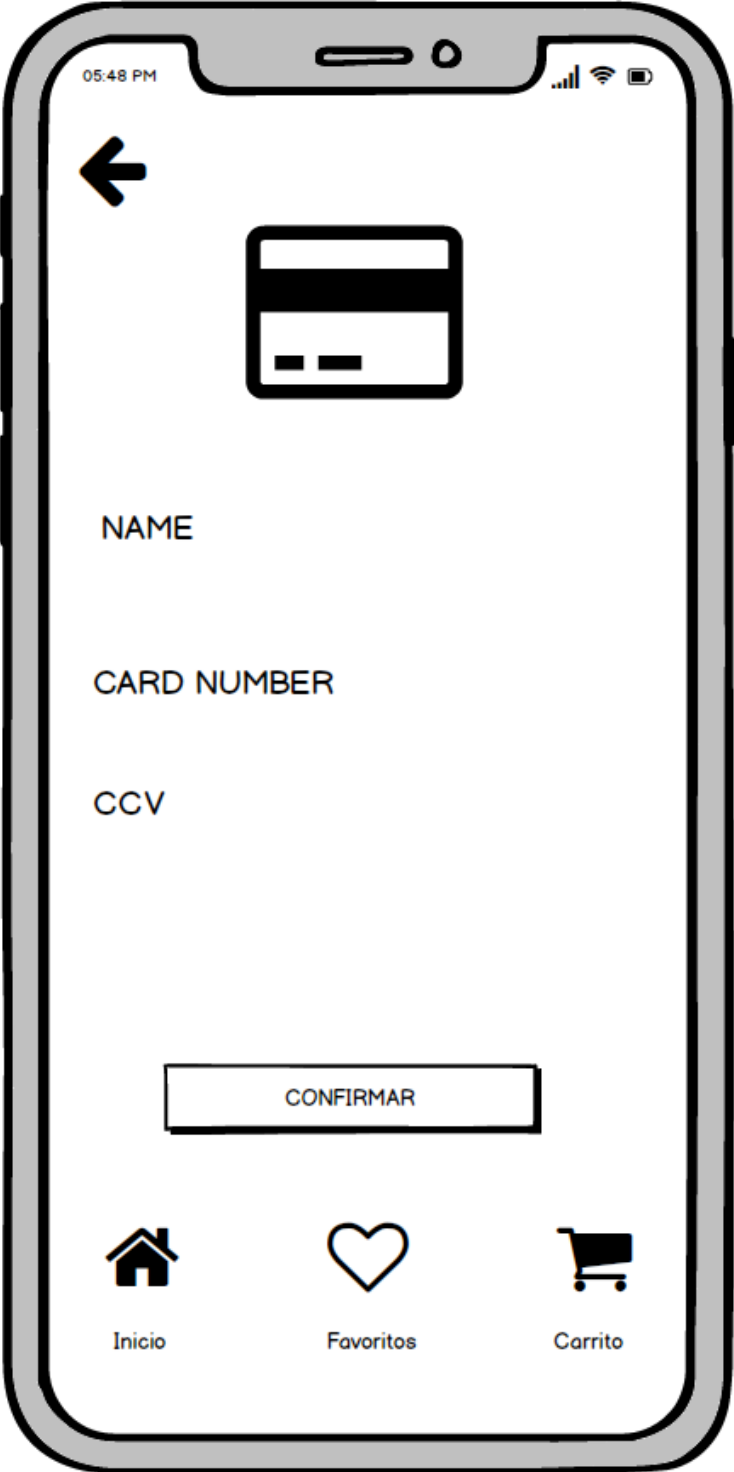
*Figura 19: Modelado Método de Pago*



La figura 19 muestra la interfaz que el usuario deberá seleccionar el método de pago, el cual será en efectivo o en línea con tarjeta de crédito o débito.

Módulo 10 – Pago en Línea

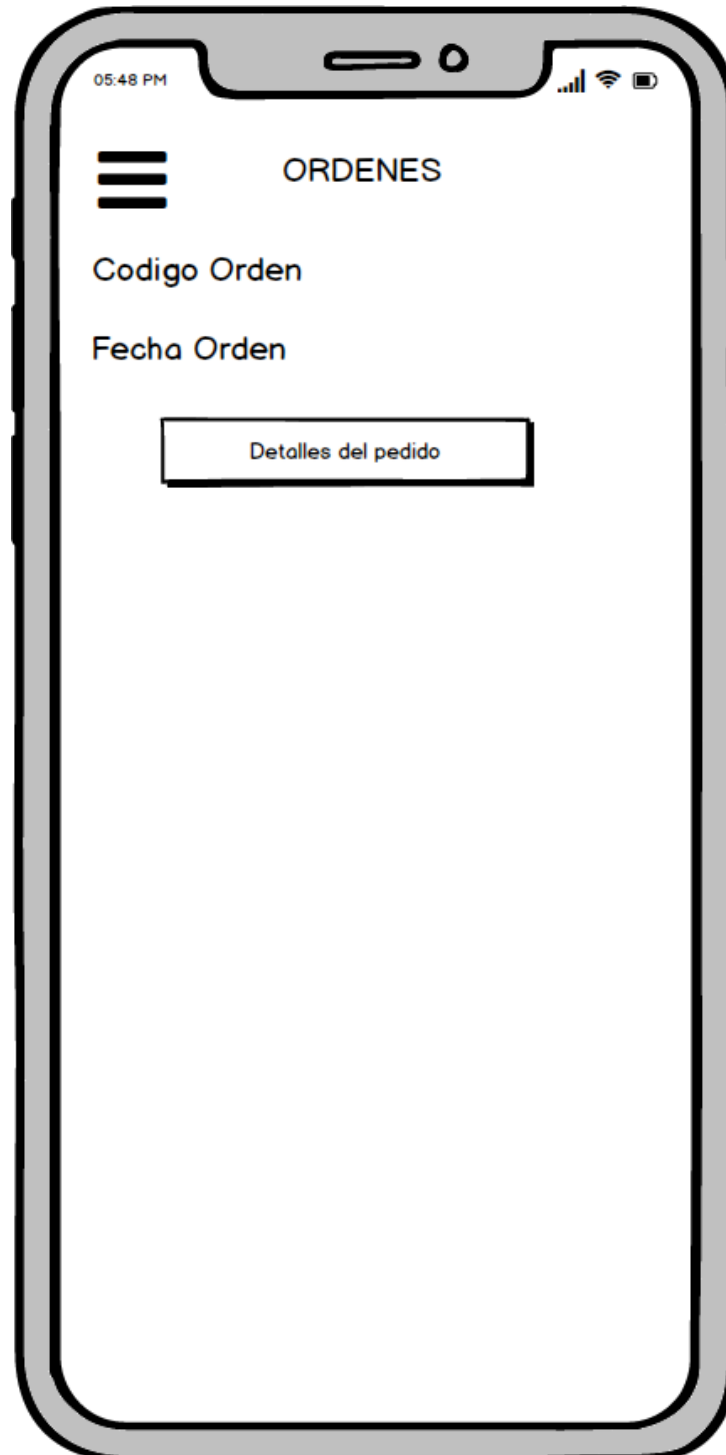
Figura 20: Modelado Pago con Tarjeta



La figura 20 muestra la interfaz donde el cliente debe ingresar los datos correspondientes de la tarjeta con la que desea pagar.

## Módulo 11 – Ordenes

Figura 21: Modelado del listado de todos los pedidos realizados



La figura 21 muestra la interfaz donde el cliente podrá observar todos los pedidos realizados por este, donde se puede mostrar el número de pedido y la fecha en que este se generó.

**Módulo 12 – Detalle de Ordenes**

*Figura 22: Modelado Detalles del pedido*



La figura 22 muestra la interfaz donde el cliente podrá observar toda la información del pedido realizado.

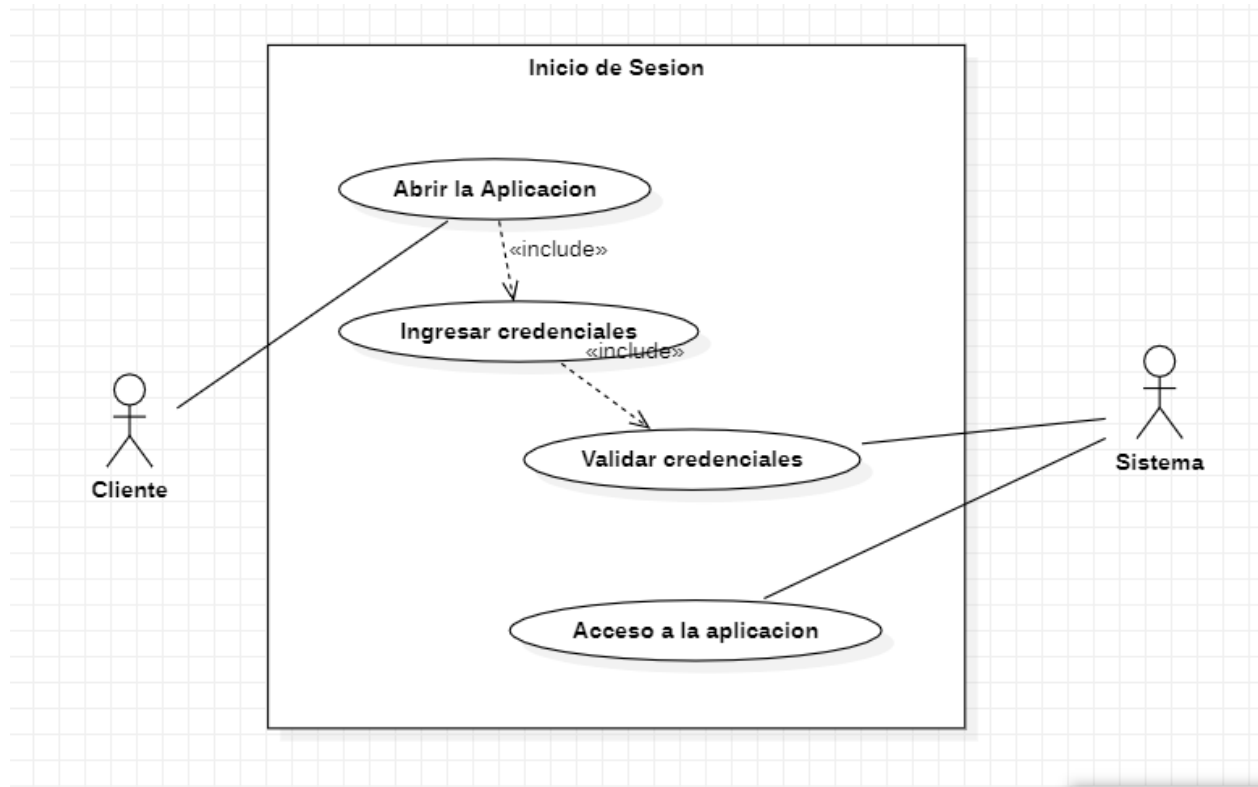


#### 2.4.3.4 Diseño de diagramas UML

Los diagramas UML también llamado lenguaje de modelado unificado permite diseñar un sistema de software y documentarlo, son de gran utilidad debido a que permiten poseer una mejor vista de cómo será el funcionamiento de la aplicación.

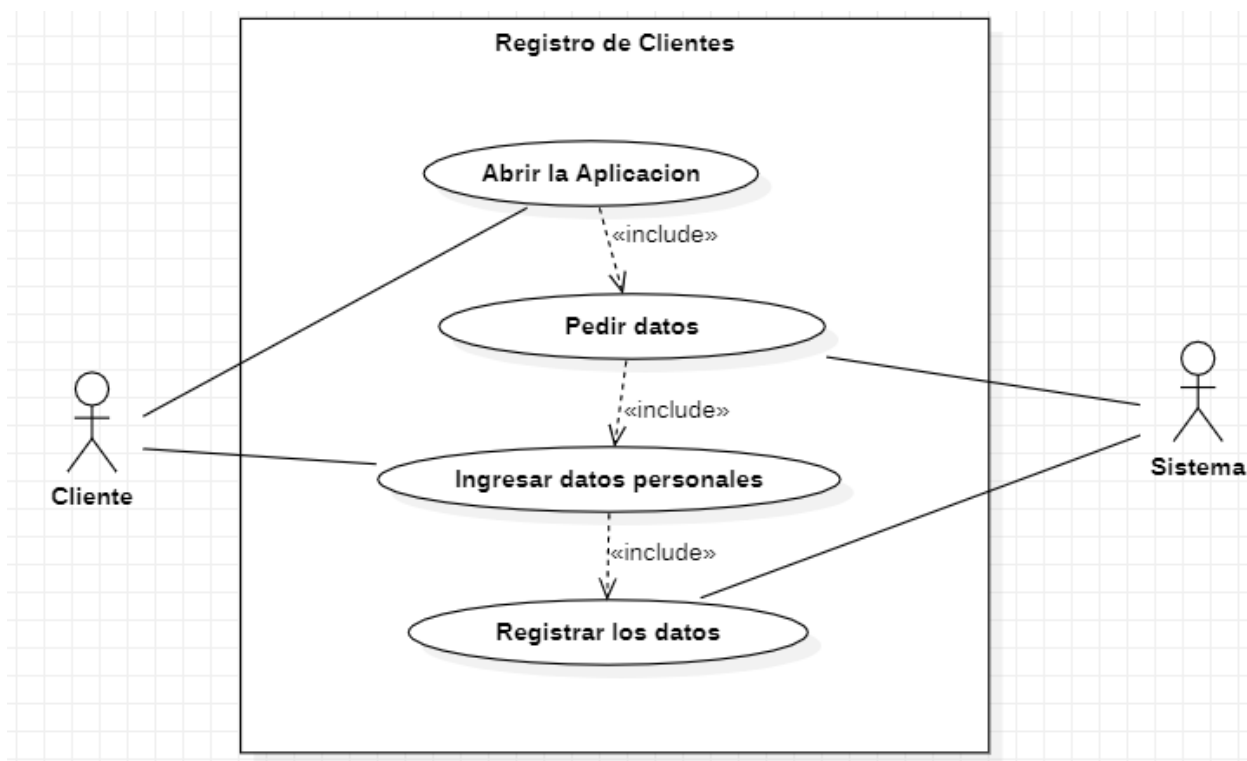
##### 2.4.3.4.1.1 DIAGRAMAS DE CASOS DE USO

Figura 23: Caso de Uso (Inicio de Sesión)



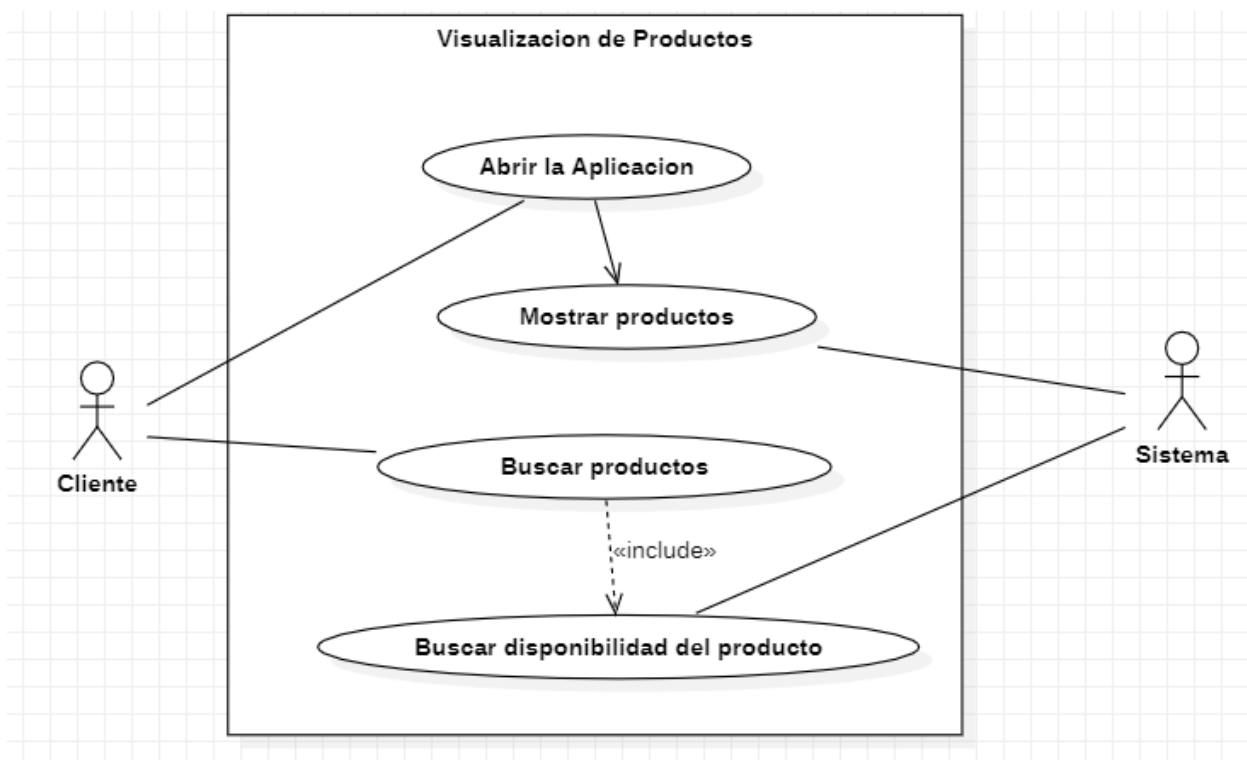
En la figura 23 se observa el proceso que el cliente debe realizar para ingresar a la aplicación mediante la comprobación de sus datos como es el correo y la contraseña, las mismas que el sistema valida de forma interna, por lo tanto, permitirá de forma correcta ingresar para tener acceso a todas las funcionalidades de la aplicación.

Figura 24: Caso de Uso (Registro de Clientes)



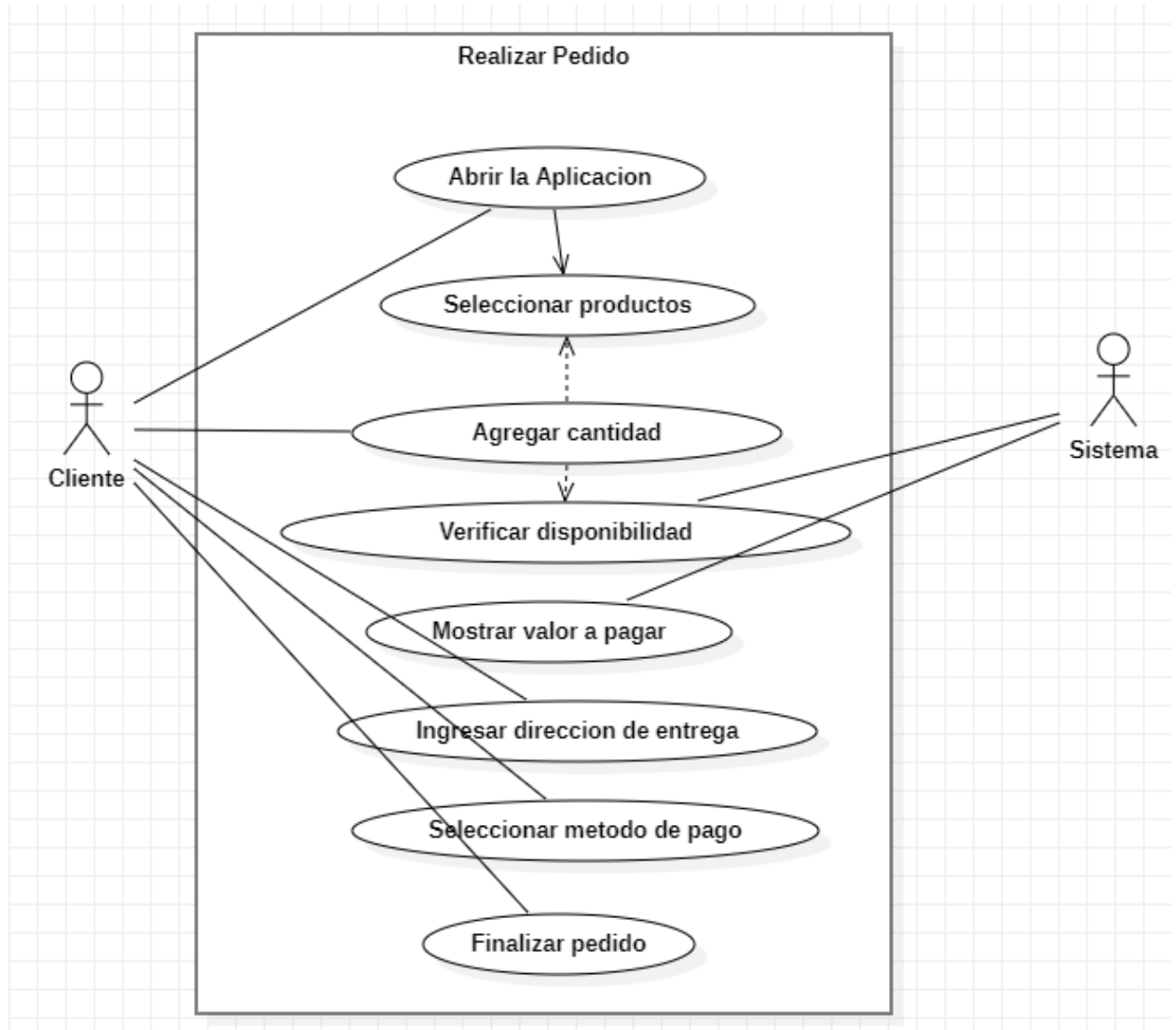
En la figura 24 se detalla el módulo de registrarse el cual solo los clientes podrán registrarse para poder iniciar sesión en la aplicación y poder tener acceso a todas las funcionalidades de la aplicación.

Figura 25: Caso de Uso (Búsqueda de productos)



En la figura 25 nos muestra de forma general todos los productos que se encuentran disponibles que el usuario podrá observar, también contará con una opción de búsqueda la misma que será verificada por el sistema antes de visualizarla al usuario.

Figura 26: Caso de Uso (Realizar Pedido)

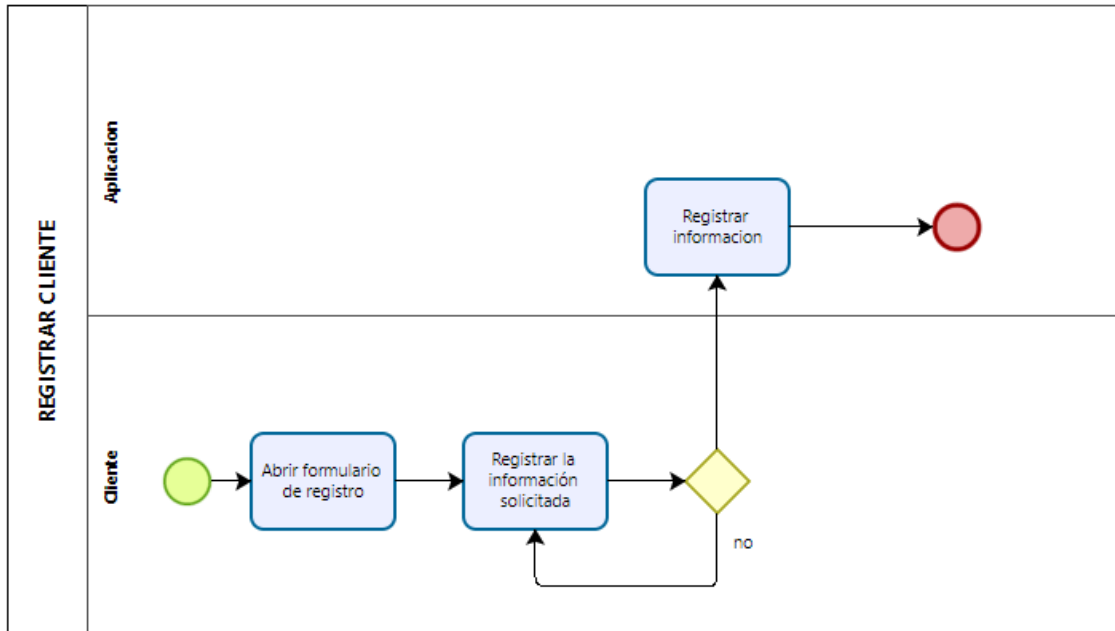


En la figura 26 se visualiza las opciones que tendrá el cliente al momento de agregar productos para realizar su pedido, para lo cual es debe ingresar la dirección y método de pago a realizarse para que el repartidor haga la entrega.

#### 2.4.3.4.1.2 DIAGRAMAS DE ACTIVIDADES

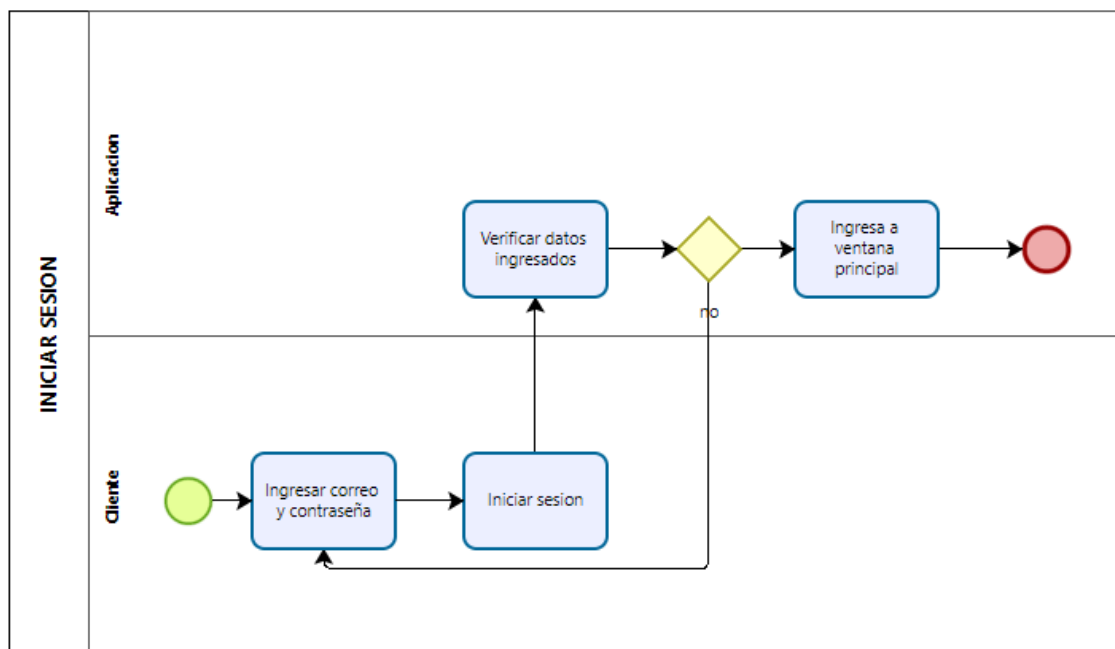
Estos diagramas permiten representar todos los flujos que poseen los procesos dentro del software con el objetivo de analizar cómo se comportan dentro del sistema.

Figura 27: Diagrama de Actividades (Registro Cliente)



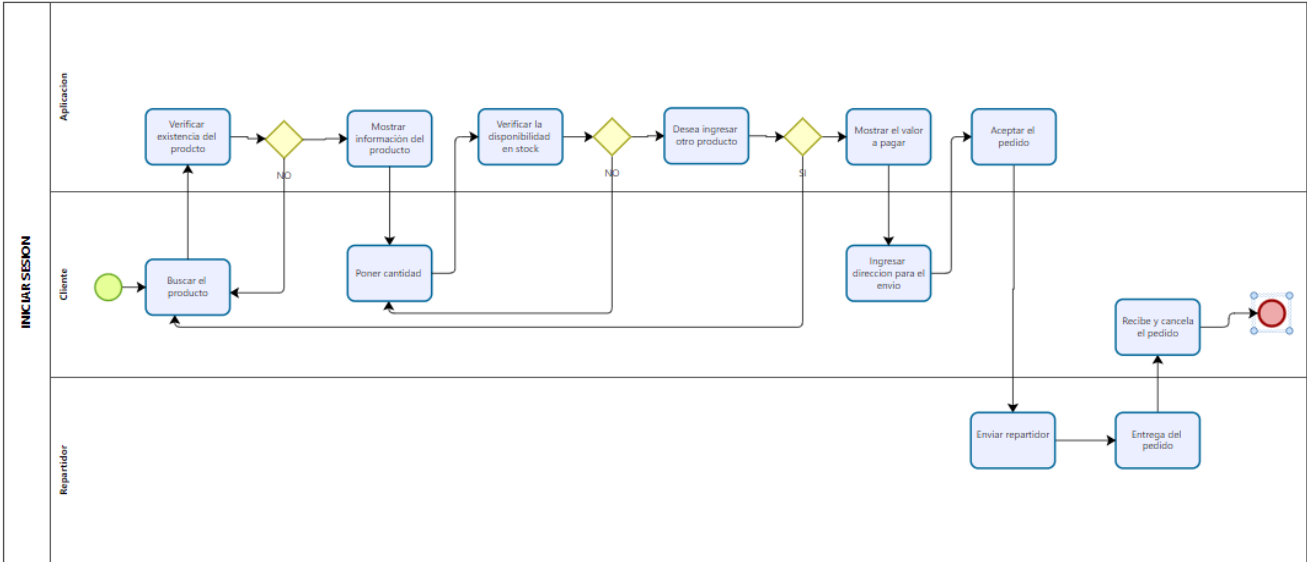
La figura 27 representa el desarrollo que se realizará el software al momento que el cliente se registra, la cual abrirá un formulario donde el cliente deberá registrar la información que se pide para poder ser registrado.

Figura 28: Diagrama de Actividades (Iniciar Sesión)



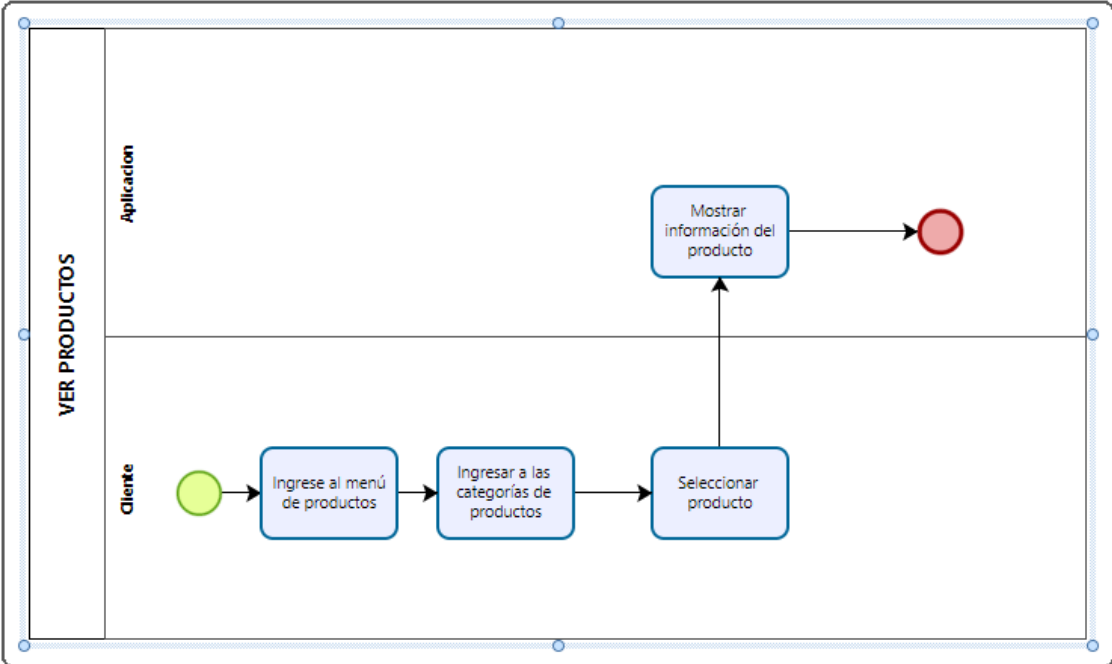
La figura 28 nos detalla el proceso que se realiza para ingresar a la aplicación por medio del inicio de sesión, donde el sistema pedirá el correo y la contraseña, las cuales estas serán autenticadas y en caso que el sistema no las valide el cliente no podrá acceder a la aplicación.

Figura 29: Diagrama de Actividades (Selección de Productos)



La figura 29 nos muestra el proceso para llevar a cabo el registro de los productos para comprar por parte del cliente, donde en primer lugar deberá iniciar sesión y escoger los productos que desea adquirir para luego proceder al pago correspondiente.

Figura 30: Diagrama de Actividades (Ver Productos)

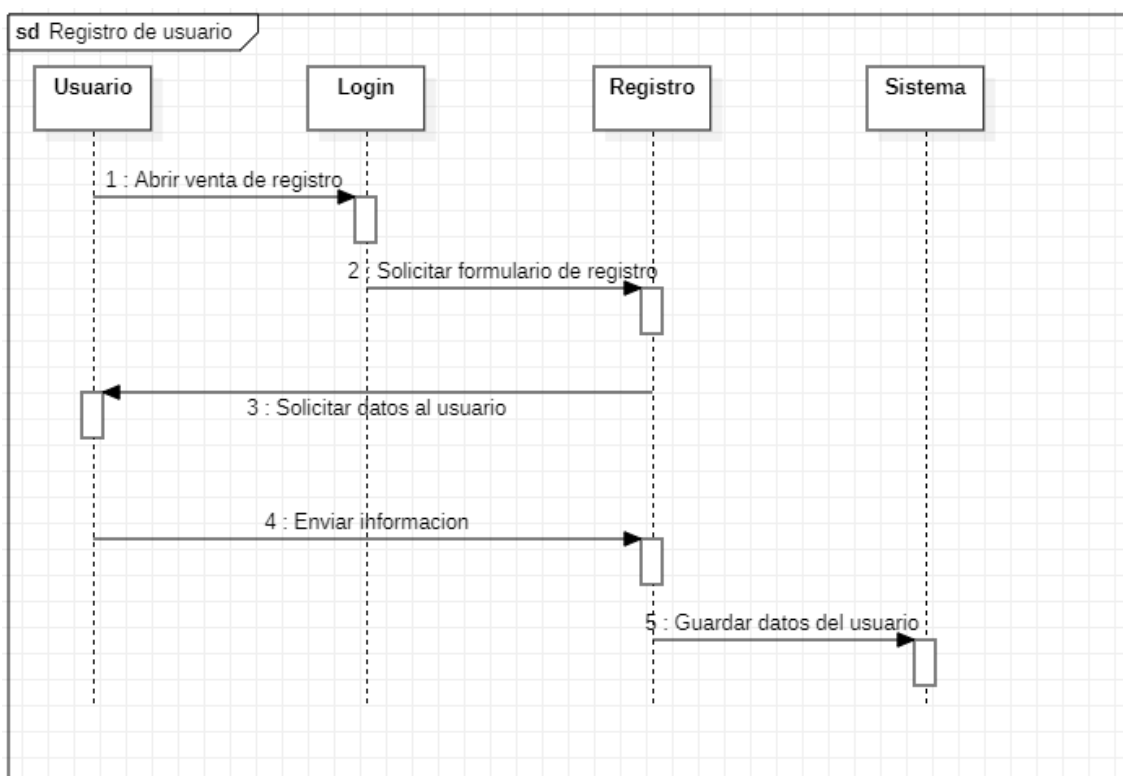


La figura 30 nos detalla el proceso para buscar los productos disponibles, el usuario deberá acceder a la aplicación y se procederá a visualizar los productos disponibles y una vez que el usuario haya seleccionado el producto en específico elegirá la cantidad y se mostrara el valor a cancelar.

#### 2.4.3.4.1.3 DIAGRAMAS DE SECUENCIA

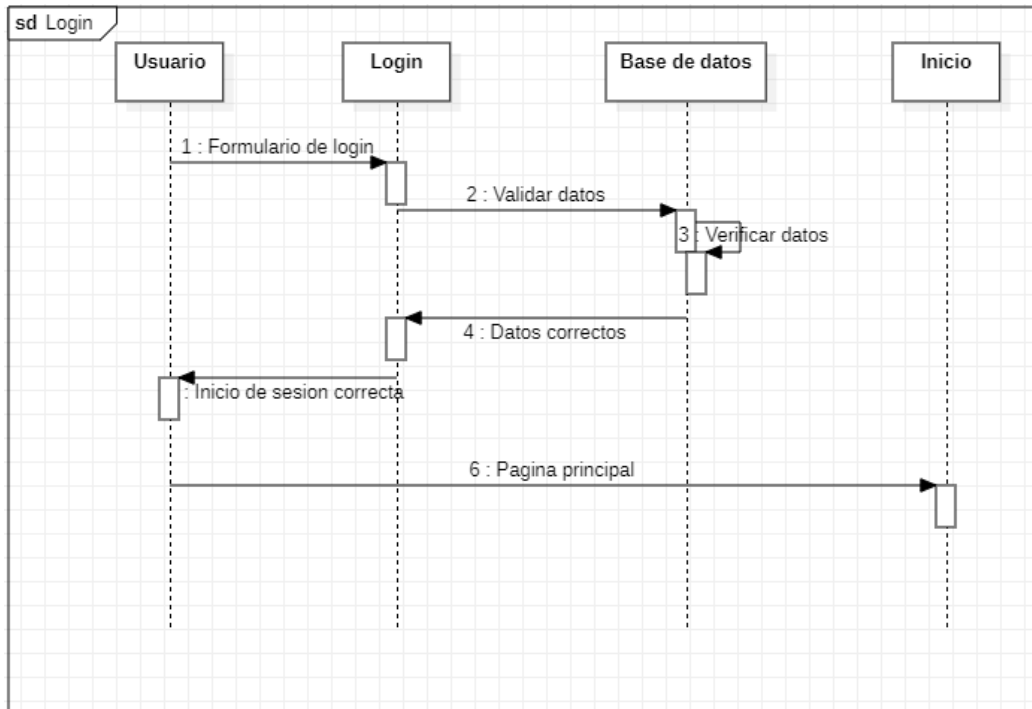
Estos diagramas ayudan a verificar los diferentes roles de los objetos dentro del software mediante una secuencia de mensajes que describen la interacción entre ellos.

Figura 31: Diagrama de Secuencia (Registro de Cliente)



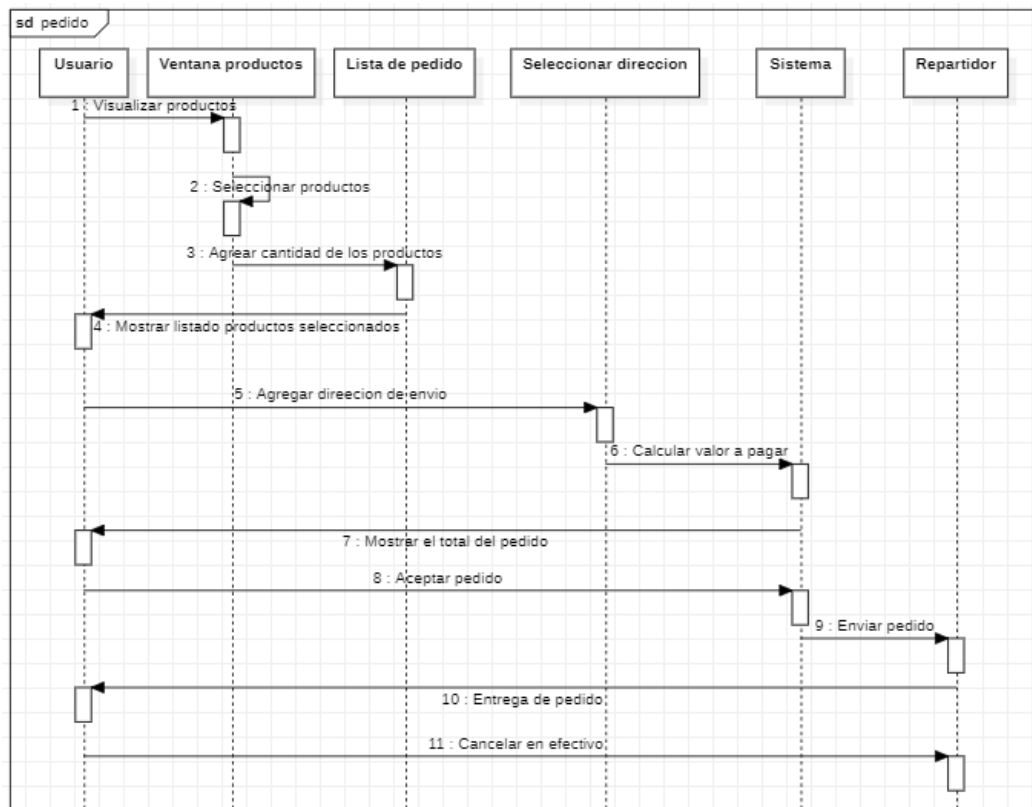
La figura 31 permite conocer el proceso para un cliente nuevo, para lo cual se mostrará en primer lugar la interfaz del login, donde podrá acceder al formulario para ingresar sus datos y luego proceder a guardar la información en la base de datos para poder ingresar al sistema.

Figura 32: Diagrama de Secuencia (Inicio de Sesión)



La figura 32 nos permite conocer cómo será el proceso para el inicio de sesión, mediante el ingreso del correo y contraseña se procede a buscar dichos datos dentro de la base de datos, si el usuario existe el usuario podrá navegar por toda la aplicación y hacer uso de sus funciones.

Figura 33: Diagrama de Secuencia (Realizar Pedido)



La figura 26 nos permite conocer cómo será el procedimiento la selección de productos para realizar el pedido, la cual el cliente deberá seleccionar los productos y las cantidades, una vez seleccionados el sistema calculara el total a pagar.



## 2.5 FASE DE EJECUCIÓN Y/O ENSAMBLAJE DEL PROTOTIPO

Para el desarrollo de la presente aplicación para realizar pedidos con entrega a domicilio del negocio se utilizó la metodología Swirl con el objetivo de realizar una aplicación de forma correcta y ágil.

A continuación, se detallarán los módulos funcionales más destacados en la aplicación.

### 2.5.1 Módulo principal

Una vez que el usuario descargue la aplicación y ejecute se mostrará una ventana principal donde se muestra información del negocio.



Ilustración 1: Ventana Principal de la Aplicación

## 2.5.2 Módulo Registro de Usuario

Para el registro del cliente este debe poner los datos solicitados por el sistema, para poder ingresar.

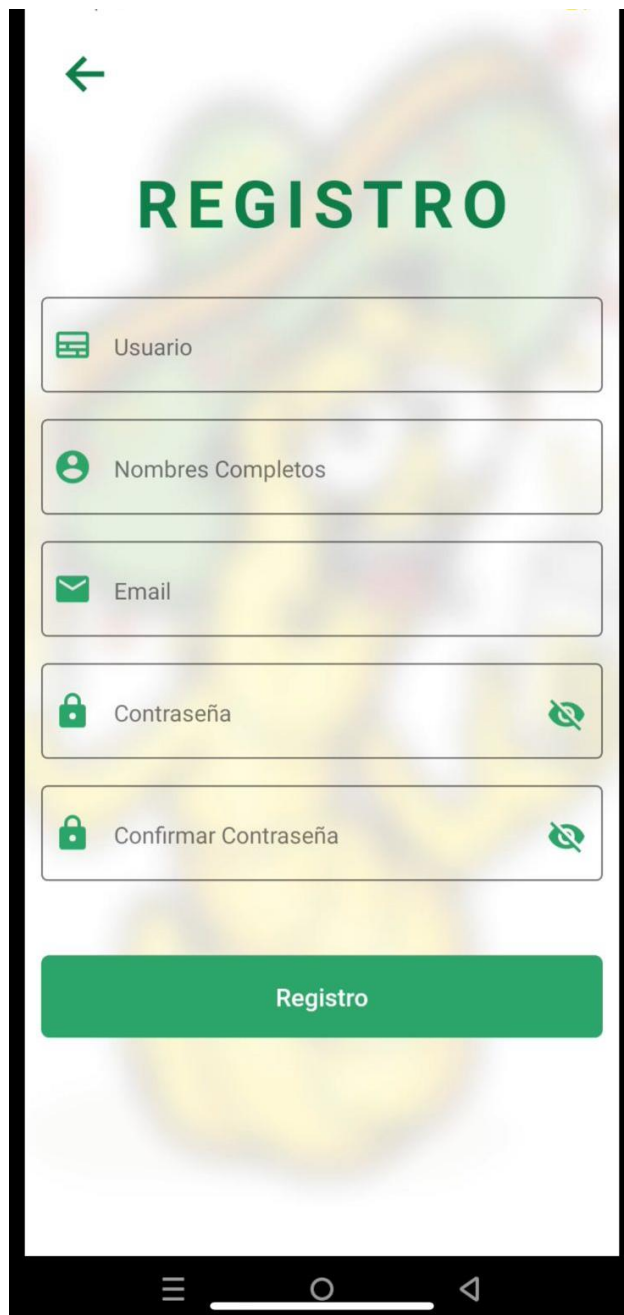
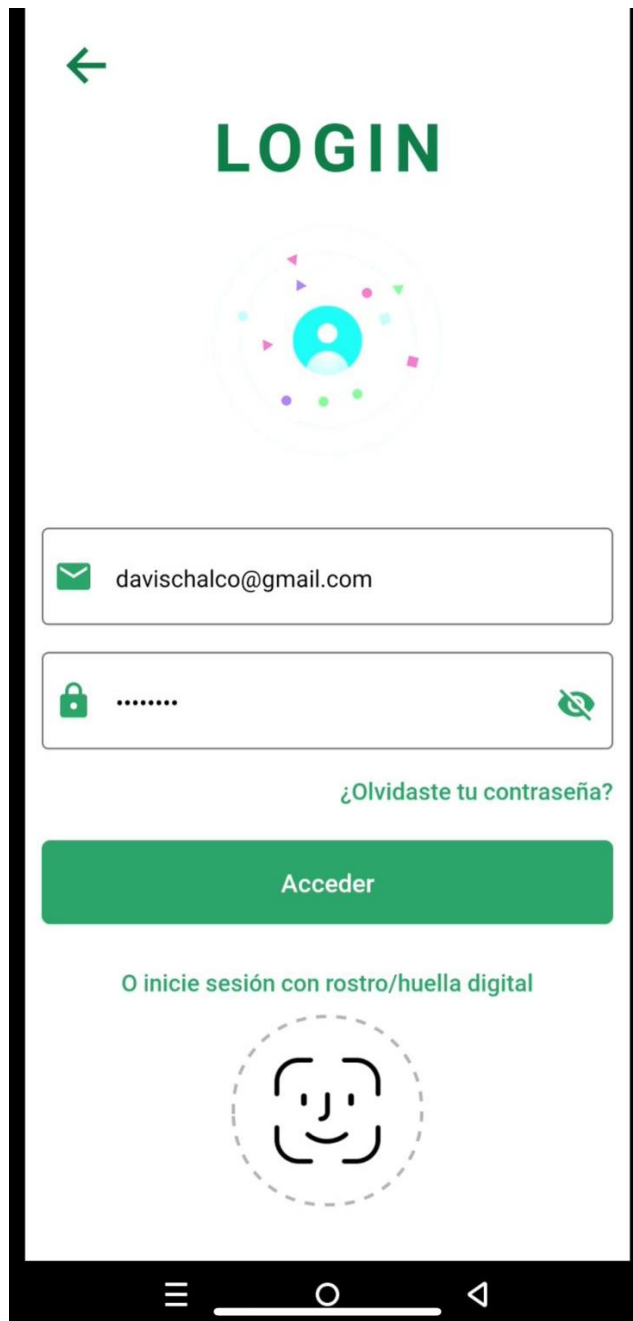


Ilustración 2: Ventana del Registro de Clientes

## 2.5.3 Módulo Iniciar Sesión

El usuario debe ingresar su correo electrónico y su contraseña para poder iniciar sesión, en caso que no se acuerde la clave podrá recuperarla a través del correo electrónico.



*Ilustración 3: Ventana de Inicio de Sesión*

#### **2.5.4 Módulo Ventana Principal**

Después de haber iniciado sesión se mostrará todas las opciones de la aplicación donde el cliente puede observar todas las funcionalidades de esta.

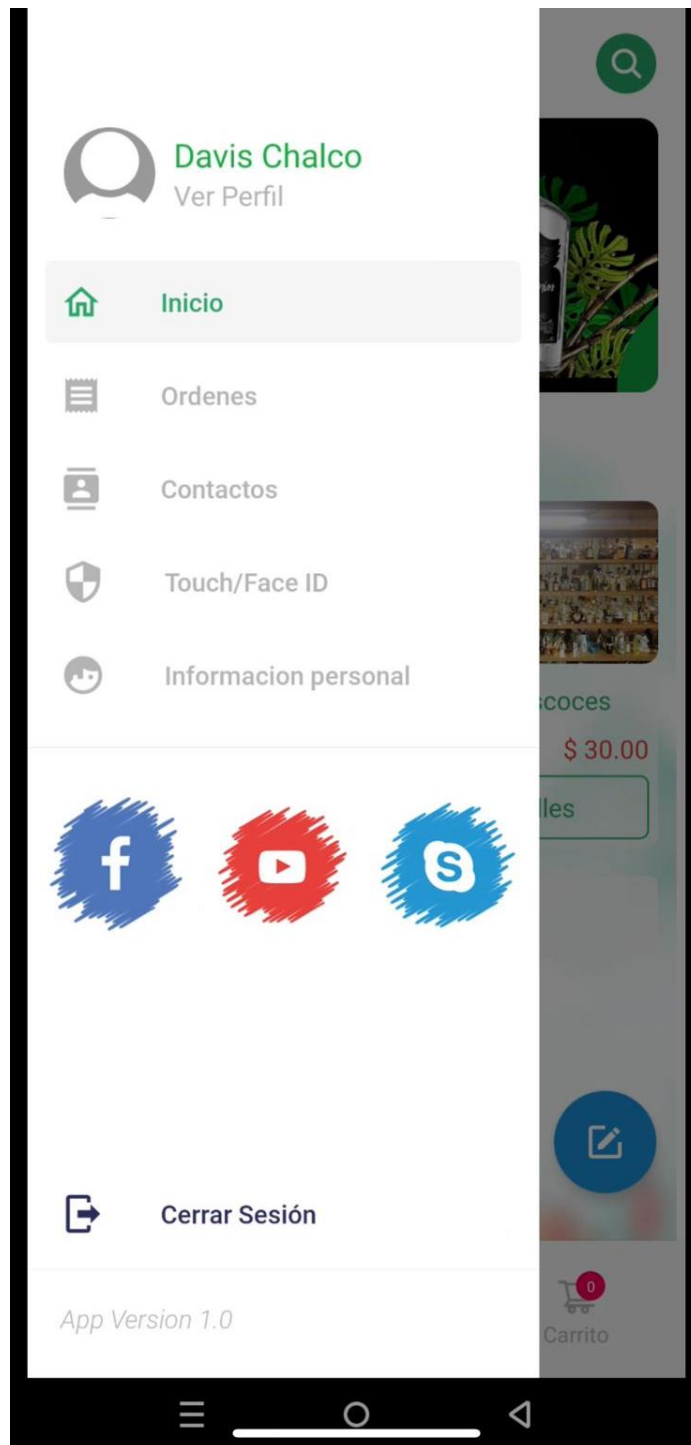


Ilustración 4: Ventana Principal Aplicación

### 2.5.5 Módulo Detalle de los Productos

Se muestra la información detallada de cada uno de los productos. Una vez que el usuario haya ingresado a la aplicación, debe dirigirse al feed de navegación de productos donde el usuario podrá conocer los diversos productos que se encuentran disponibles, esta ventana también incluye un buscador para ayudar al usuario a encontrar de una manera más rápida los productos que necesite.

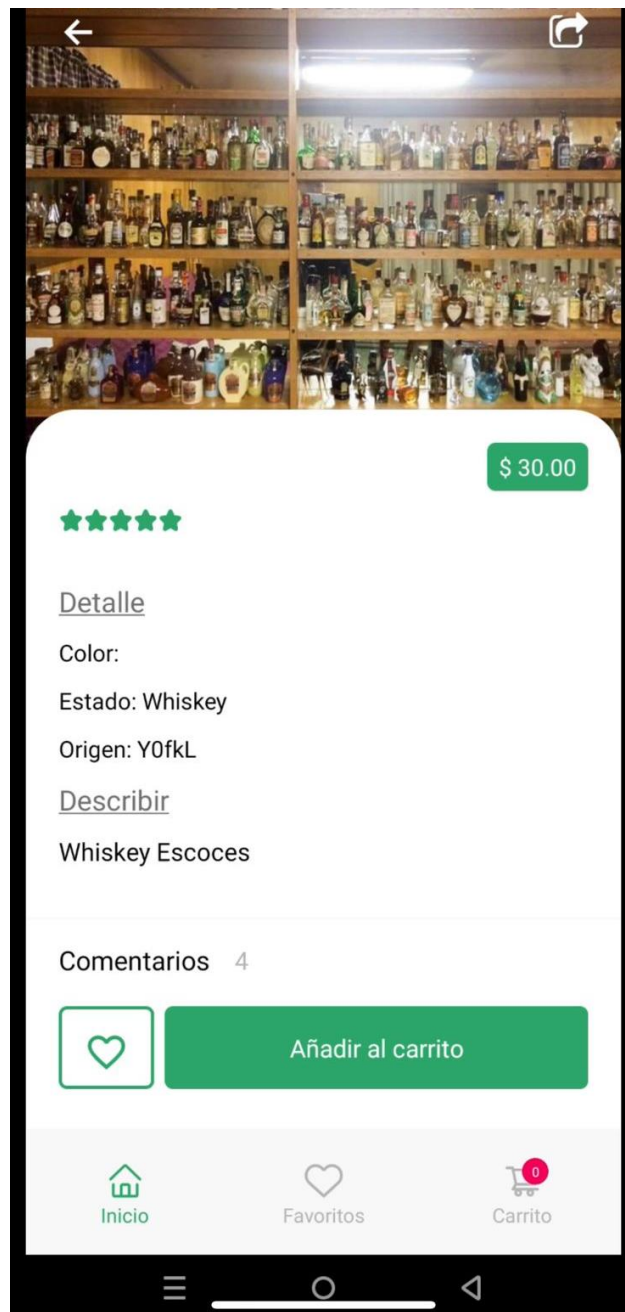


Ilustración 5: Ver productos disponibles

### 2.5.6 Módulo Carrito de Compras

Una vez que el usuario que esté listos los productos que desea comprar se le mostrará una lista con todos los productos agregados con los respectivos precios y cantidades solicitadas, de esta manera el cliente podrá conocer el valor a cancelar.



Ilustración 6: Carrito de compras

### 2.5.7 Módulo Datos de Envió

Como siguiente paso el usuario deberá poner la dirección de envió. Este módulo permite al cliente proporcionar datos para que se le pueda emitir una factura, estos datos el usuario lo podrá modificar en cualquier momento.

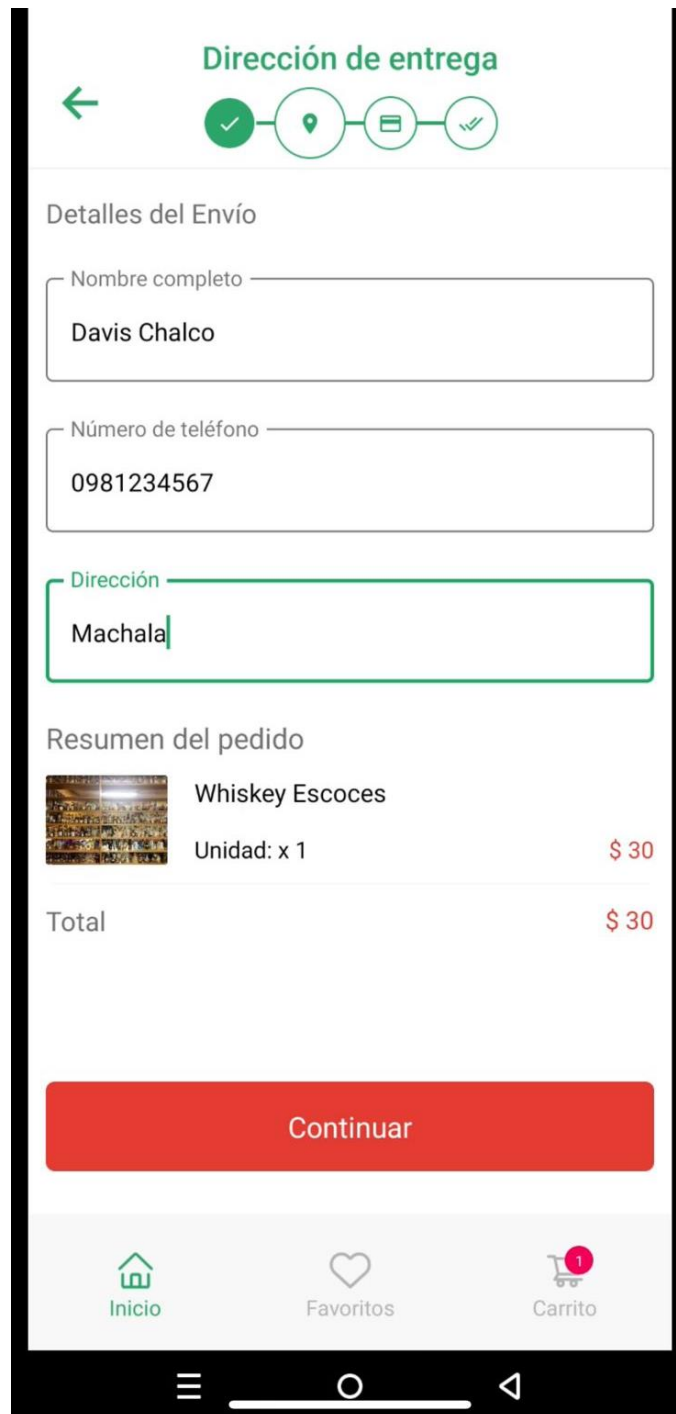


Ilustración 7: Datos de envío del pedido

### 2.5.8 Módulo Métodos de Pago

El cliente deberá seleccionar su método de pago ya sea en efectivo cuando se le entregue su pedido o en línea con su tarjeta preferida.

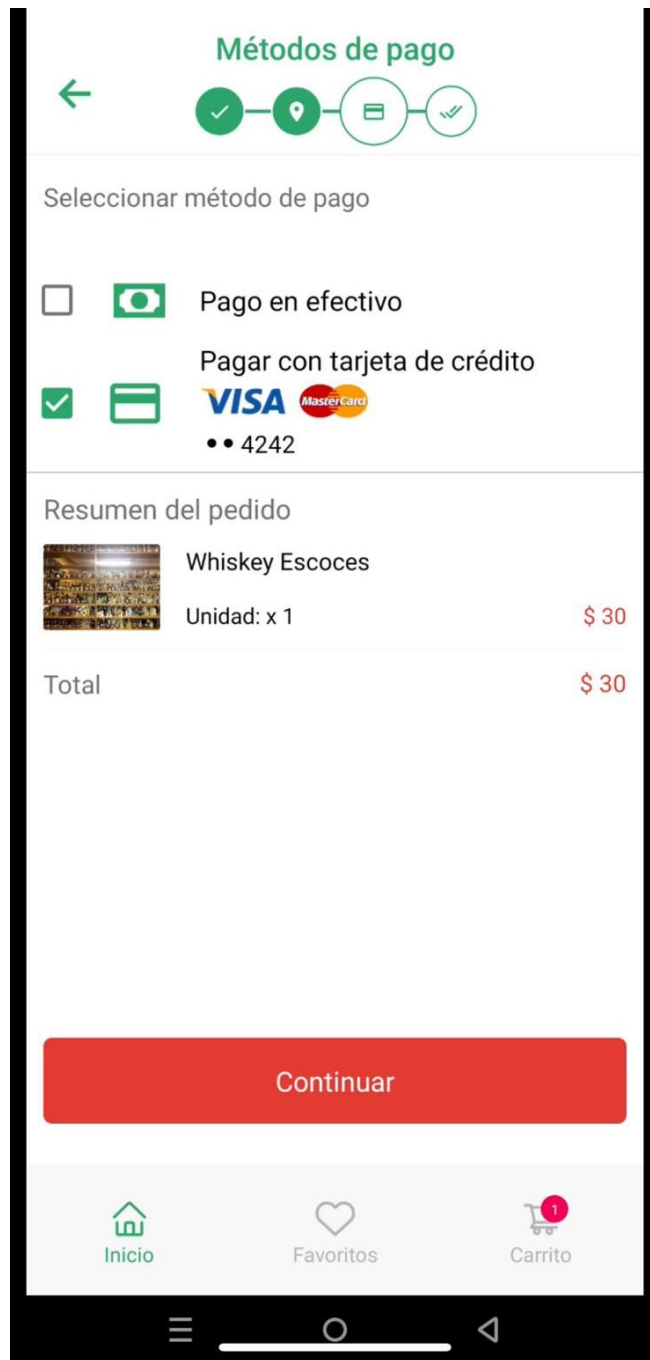


Ilustración 8: Método de pago del pedido

### 2.5.9 Módulo Pago en Línea

El cliente debe ingresar los datos de su tarjeta para que sea verificada y aprobada.



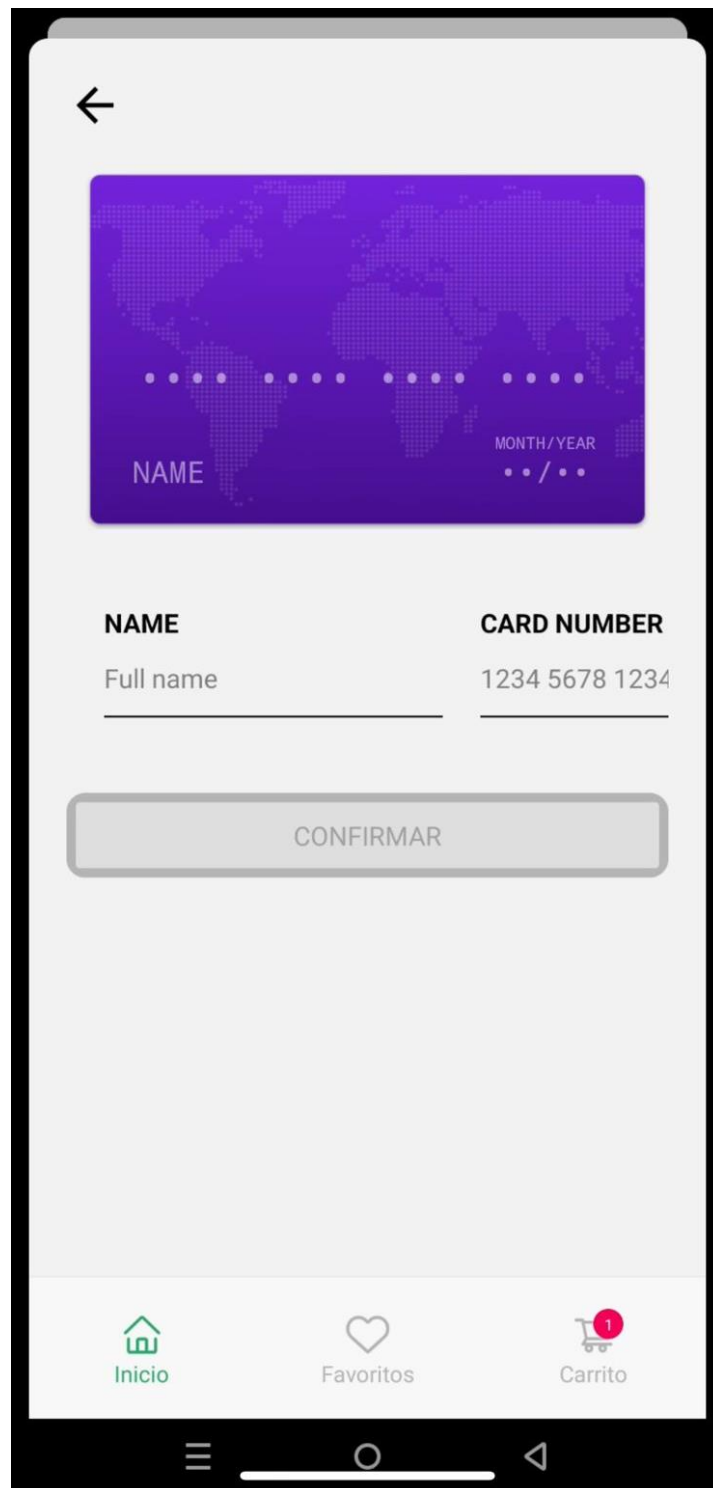
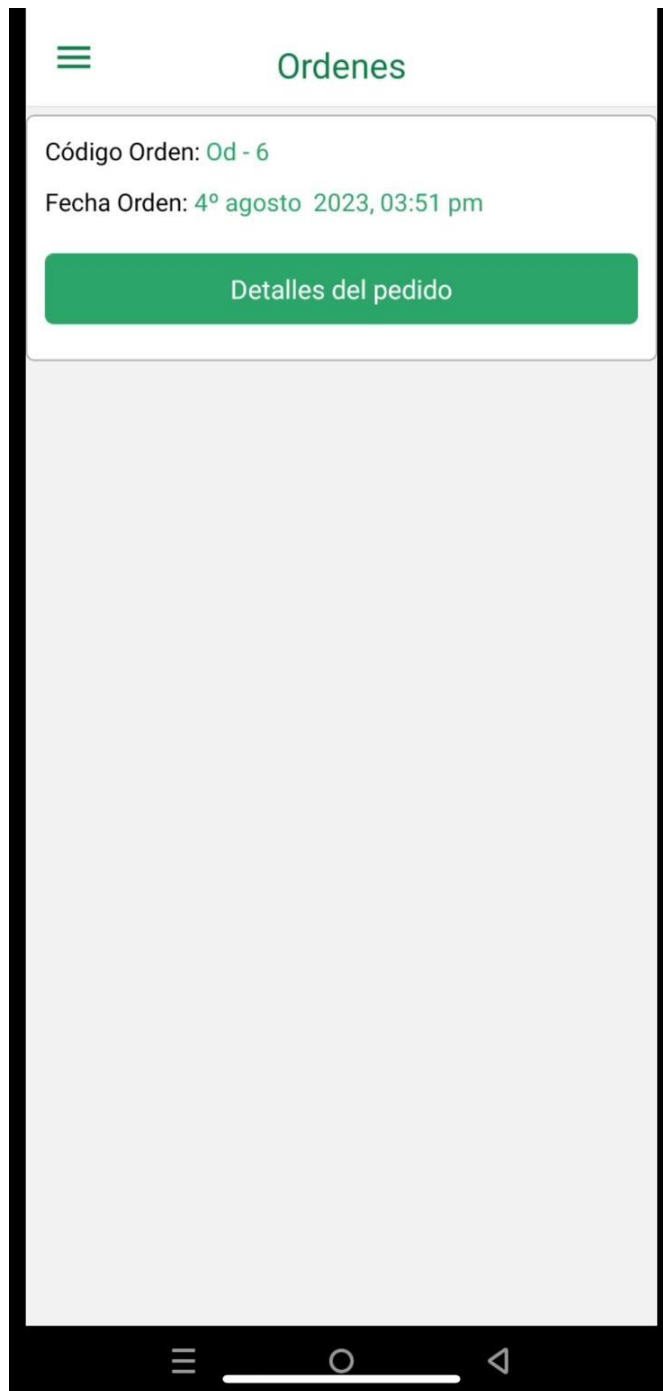


Ilustración 9: Método Pago en línea del pedido

### 2.5.10 Módulo Órdenes

El cliente podrá observar todo el listado de las órdenes pedidas con el número y fecha emitidos.



*Ilustración 10: Ver todas las ordenes realizadas*

### **2.5.11 Módulo Detalle de las Órdenes**

Se presenta todos los detalles de la orden, además del seguimiento del estado de este.

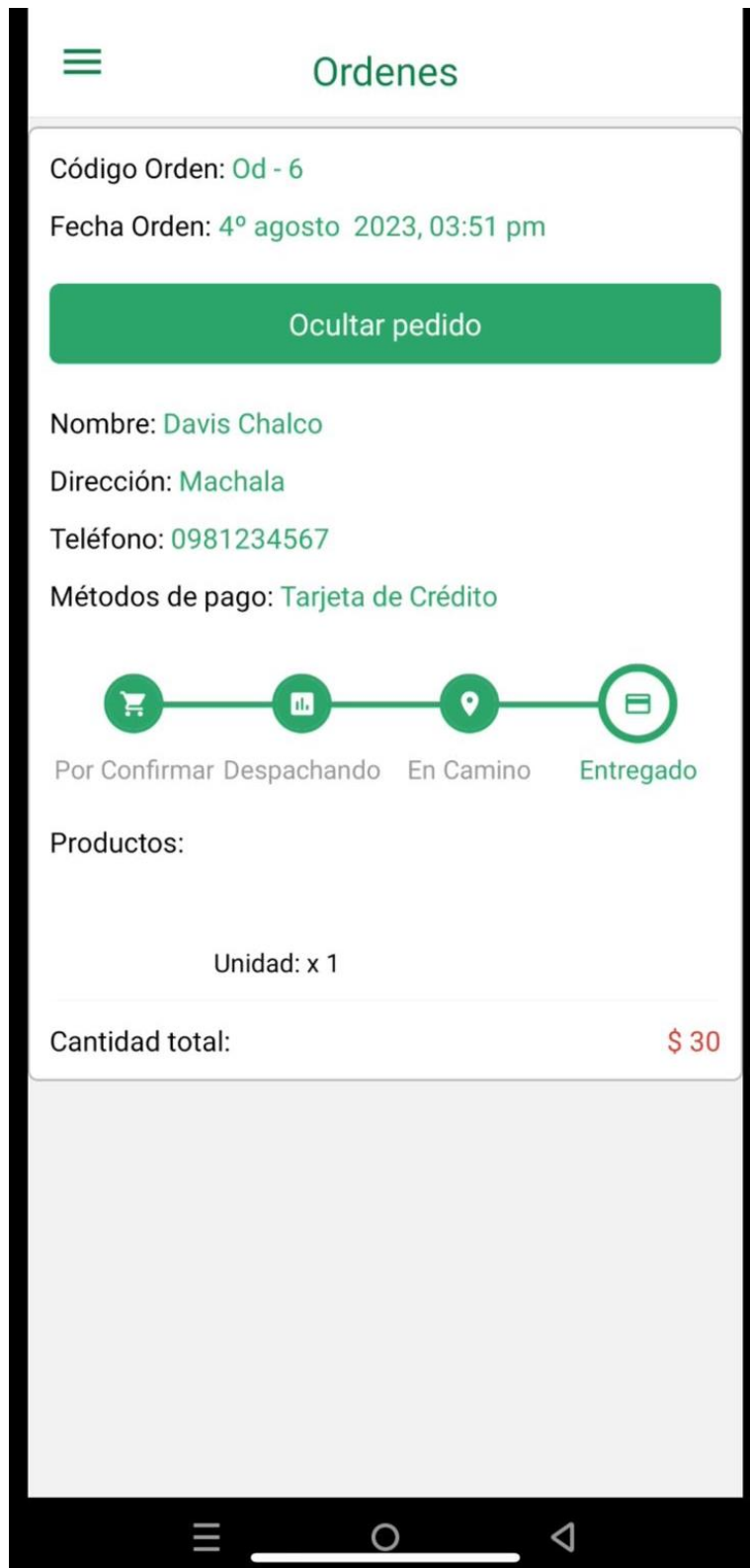


Ilustración 11: Ver detalle de cada pedido

### **2.5.12 Dashboard Web**

En esta sección el administrador del negocio podrá observar todos los pedidos y además de mostrar datos métricos para conocer cuáles son los productos con más demanda y poder tomar decisiones para el crecimiento de esta, ayudando a reducir costos y tiempo.

*Ilustración 12: Dashboard Web*

### 3 CAPÍTULO III. EVALUACIÓN DEL PROTOTIPO

#### 3.1 PLAN DE EVALUACIÓN

Los criterios y objetivos del estándar ISO/IEC9126 se utilizan para la evaluación de aplicaciones, con el objetivo de proporcionar herramientas de evaluación de la calidad del software con el objetivo de probar la funcionalidad de los sistemas para cumplir con los requisitos de los usuarios.

El estándar evaluará los siguientes indicadores: funcionalidad, confiabilidad, facilidad de uso, eficiencia, mantenibilidad y portabilidad, los cuales serán divididos en subcategorías y lineamientos necesarios para determinar la calidad de la aplicación.

#### 3.2 RESULTADOS DE LA EVALUACIÓN

Mediante los resultados que se mostraron al aplicar la norma IOS/IEC 9126 en la aplicación ha dado a conocer el nivel de calidad del sistema en base a las métricas de funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad, las cuales los resultados fueron evaluados mediante la escala de Likert que cuenta con una puntuación del 1 al 5.

Característica	Sub-características	Criterio	Escala de Likert				
			Excelent	Bueno	Regular	Malo	Muy
Funcionalidad	Conveniencia	El software desempeña las tareas requeridas por el usuario.	x				
	Exactitud	El software presenta los resultados sin errores.	x				
	Interoperabilidad	Las funciones del software van	x				

		acorde al rol del usuario.					
	Seguridad	El software prohíbe el acceso no autorizado.	x				
Confiabilidad	Tolerancia a las fallas	El software es tolerante antes las fallas.		x			
	Madurez	El software muestra mensajes de advertencia antes fallos.	x				
	Recuperación	El software puede seguir funcionando antes fallos.		x			
Usabilidad	Comprensión	El software detecta y entiende el rol de cada usuario.	x				
	Capacidad de aprender	El software es fácil a utilizar.	x				
	Operatividad	El software permite adaptarse velozmente a él.	x				
	Atractividad	El software tiene una interfaz amigable.	x				
Eficiencia	Comportamiento en el tiempo	El software tiene tiempos de espera agradables.	x				

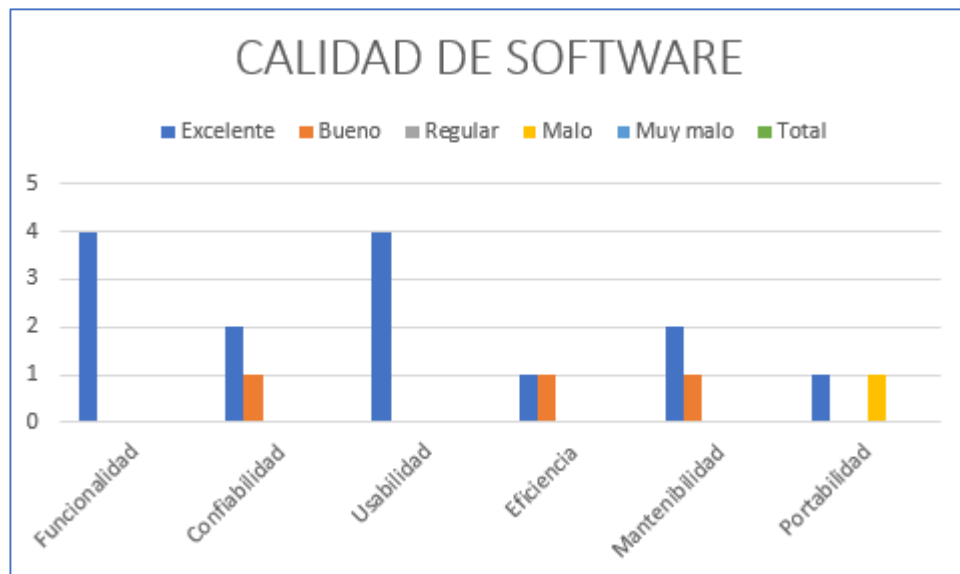
	Gasto de recursos	El software utiliza de forma eficiente los recursos del dispositivo.		x			
Mantenibilidad	Facilidad de análisis	El software permite detectar fácilmente los errores.	x				
	Facilidad de pruebas	El software puede implementar un proceso de pruebas.		x			
	Facilidad de cambio	El software puede implementar cambios dentro del sistema rápidamente.	x				
Portabilidad	Capacidad de instalación	El software es fácil de instalar.	x				
	Capacidad de reemplazamiento	El software puede ser sustituido por otro.				x	

Para llevar a cabo los valores obtenidos de la tabla anterior, se presenta un resumen en una tabla para cada sub característica, dándonos a conocer de forma más exacta los resultados obtenidos.

Característica	Excelente	Bueno	Regular	Malo	Muy malo	Total
Funcionalidad	4					
Confiabilidad	2	1				
Usabilidad	4					
Eficiencia	1	1				
Mantenibilidad	2	1				
Portabilidad	1			1		

Después de la evaluación de calidad utilizando el estándar ISO/IEC 1926, según la tabla, se puede encontrar que la mayoría de las funciones tienen niveles excelentes y buenos, lo que demuestra que el software es de alta calidad.

Para un mejor entendimiento de los resultados se presenta un gráfico que represente la evaluación de calidad bajo la norma ISO/IEC 1926 a la aplicación móvil para la búsqueda de trabajo, visualice figura





### 3.3 CONCLUSIONES

- Se obtuvo los requisitos funcionales y no funcionales del software mediante el desarrollo de historias de usuario para definir más claramente el proceso de aplicación.
- Las tecnologías utilizadas en el desarrollo de este proyecto, tales como: MySQL, Node.js, React Native, permiten lograr una aplicación multiplataforma con un sistema back-end totalmente basado en la nube, lo que facilita el acceso rápido a los datos.
- El prototipo de la aplicación se desarrolló utilizando la herramienta Balsamiq Mockups, ya que brindó una vista previa de la aplicación y facilitó al usuario navegar y navegar por la aplicación. Varios procesos y funciones de la aplicación se pueden modelar correctamente utilizando herramientas de modelado, así como Bizagi y StarUML, lo que ayuda a optimizar y reducir la frustración durante la fase de implementación.
- Un conjunto de indicadores y lineamientos en el estándar ISO/IEC9126 pueden evaluar si el software cumple con sus requisitos en un nivel excelente y bueno, por lo que, debido a esta evaluación, se hace posible un software de buena calidad en la aplicación, satisfaciendo así las necesidades del cliente.

### **3.4 RECOMENDACIONES**

- Para garantizar el cumplimiento de entregables en los tiempos establecidos es recomendable planificar actividades en relación al proyecto utilizando una metodología de software.
- Es indispensable realizar una investigación sobre las tecnologías que se van a implementar para el desarrollo del software con la finalidad de poder cumplir con todos los objetivos planteados y desarrollar una aplicación eficiente.
- Utilizar herramientas de evaluación de aplicaciones para detectar los posibles errores presentados en las diferentes etapas del desarrollo de la aplicación y poder mejorar estos errores sin afectar el funcionamiento.
- Se recomienda para proyectos de software que manejen grandes volúmenes de información el uso de una base de datos “NoSQL” ya que permiten almacenar grandes cantidades de información y evitar una afectación en la integridad de los datos.

## REFERENCIAS BIBLIOGRAFICAS

- [1] S. Oñate, «El confinamiento impulsa otra vez al “delivery” en Ecuador», *El Comercio*, 6 de mayo de 2021. <https://www.elcomercio.com/actualidad/negocios/confinamiento-impulsa-delivery-ecuador-pandemia.html> (accedido 23 de julio de 2023).
- [2] A. V. R. Campuzano, «Aplicaciones digitales de delivery: Incidencia en el consumo de comidas rápidas durante el primer semestre del año 2020 en Guayaquil».
- [3] R. Cañedo Andalia, «Aproximaciones para una historia de Internet», *ACIMED*, vol. 12, n.º 1, pp. 0-0, feb. 2004.
- [4] «Boletín Técnico ESPAC\_2019.pdf». Accedido: 23 de julio de 2023. [En línea]. Disponible en: [https://www.ecuadorencifras.gob.ec/documentos/web-inec/Estadisticas\\_agropecuarias/espac/espac-2019/Boletin%20Tecnico%20ESPAC\\_2019.pdf](https://www.ecuadorencifras.gob.ec/documentos/web-inec/Estadisticas_agropecuarias/espac/espac-2019/Boletin%20Tecnico%20ESPAC_2019.pdf)
- [5] L. Cobo y R. Perez Uribe, «Proyecto Anamnesis – Desarrollo de una aplicación web y móvil para la gestión de una historia clínica unificada de los colombianos», *Rev. EAN*, p. 91, jun. 2016, doi: 10.21158/01208160.n80.2016.1459.
- [6] A. Hernández Paez, J. A. Domínguez Falcón, y A. A. Pi Cruz, «Arquitectura de software para el desarrollo de videojuegos sobre el motor de juego Unity 3D», *ID Tecnológico*, vol. 14, n.º 1, pp. 54-65, jun. 2018, doi: 10.33412/idt.v14.1.1803.
- [7] J. M. García, Á. E. Gil, E. A. Sánchez, J. M. García, Á. E. Gil, y E. A. Sánchez, «Desarrollo de una arquitectura de software para el robot móvil Lázaro», *Ingeniare Rev. Chil. Ing.*, vol. 26, n.º 3, pp. 376-390, 2018, doi: 10.4067/S0718-33052018000300376.
- [8] Instituto de Ingeniería, Facultad de Ingeniería *et al.*, «Identificación de áreas de aplicación de arquitecturas de software basadas en modelos, técnicas y herramientas de social media», *RISTI - Rev. Ibérica Sist. E Tecnol. Informação*, n.º 42, pp. 12-29, jun. 2021, doi: 10.17013/risti.42.12-29.
- [9] M. G. Cambarieri, F. Difabio, y N. G. Martínez, «Implementación de una Arquitectura de Software guiada por el Dominio».
- [10] L. M. Nancy Magaly, M. R. Jimmy Rolando, M. R. Rodrigo Fernando, y P. S. Celmira Marcela, «Mensajería cliente-servidor aplicando sockets en las herramientas GEANY IDE 1.31, PHYTON 3.7 y POSTGRESQL 9.5 en el sistema operativo CENTOS 7: Client-server messaging applying sockets in GEANY IDE 1.31, PHYTON 3.7 and POSTGRESQL 9.5 tools in CENTOS 7 operating system», *Conf. Proc. Machala*, vol. 4, n.º 1, pp. 177-185, ago. 2020, doi: 10.48190/cp.v4n1a11.
- [11] C. F. V. Pérez, J. E. P. Cleves, y L. Pallares, «Computación en la nube: Un nuevo paradigma en las tecnologías de la información y la comunicación», *Redes Ing.*, pp. 138-146, sep. 2017, doi: 10.14483/2248762X.12485.
- [12] C. G. Chanchí Golondrino, M. A. Ospina Alarcón, y M. E. Monroy Ríos, «Arquitectura IoT para el desarrollo de sistemas de monitorización y análisis de variables fisiológicas en el área de asistencia médica», *Investig. E Innov. En Ing.*, vol. 8, n.º 3, pp. 1-13, 2020.
- [13] E. Salinas, N. Cerpa, y P. Rojas, «Arquitectura orientada a servicios para software de apoyo para el proceso personal de software», *Ingeniare Rev. Chil. Ing.*, vol. 19, n.º 1, pp. 40-52, jun. 2011, doi: 10.4067/S0718-33052011000100005.
- [14] «(PDF) Nuevo paradigma en la tecnología: computación en la nube». [https://www.researchgate.net/publication/258884808\\_Nuevo\\_paradigma\\_en\\_la\\_tecnologia\\_computacion\\_en\\_la\\_nube](https://www.researchgate.net/publication/258884808_Nuevo_paradigma_en_la_tecnologia_computacion_en_la_nube) (accedido 7 de agosto de 2023).
- [15] D. Inupakutika, S. Kaghyan, D. Akopian, P. Chalela, y A. G. Ramirez, «Facilitating the development of cross-platform mHealth applications for chronic supportive care and a case study», *J. Biomed. Inform.*, vol. 105, p. 103420, may 2020, doi: 10.1016/j.jbi.2020.103420.

- [16] S. Combaudon, *MySQL 5.7: administración y optimización*. Ediciones ENI, 2018.
- [17] J. J. León Soberón, «Análisis comparativo de sistemas gestores de bases de datos postgresql y mysql en procesos crud», *Repos. Inst. - USS*, 2020, Accedido: 7 de agosto de 2023. [En línea]. Disponible en: <http://repositorio.uss.edu.pe//handle/20.500.12802/7012>
- [18] N. Battaglia, C. Neil, M. D. Vincenzi, y J. P. Beltramino, «UAI CASE: DESARROLLO Y EVALUACIÓN DE COMPETENCIAS EN LA INGENIERÍA DE SOFTWARE EN UN ENTORNO VIRTUAL DE APRENDIZAJE COLABORATIVO».
- [19] R. Perez-Castillo, F. Ruiz, M. Piattini, y C. Ebert, «Enterprise Architecture», *IEEE Softw.*, vol. 36, n.º 4, pp. 12-19, jul. 2019, doi: 10.1109/MS.2019.2909329.
- [20] «A Systematic Review of Business-IT Alignment Research With Enterprise Architecture». <https://ieeexplore.ieee.org/abstract/document/8325275/> (accedido 7 de agosto de 2023).
- [21] F. N. Khasanah, S. Rofiah, y D. Setiyadi, «Metode User Centered Design Dalam Merancang Tampilan Antarmuka Ecommerce Penjualan Pupuk Berbasis Website Menggunakan Aplikasi Balsamiq Mockups», *JAST J. Apl. Sains Dan Teknol.*, vol. 3, n.º 2, p. 14, ago. 2019, doi: 10.33366/jast.v3i2.1443.
- [22] Z. Munawar, «Perancangan Interface Aplikasi Pencatatan Persediaan Barang Di Kios Buku Palasari Bandung Dengan Metode User Centered Design Menggunakan Balsamiq Mockups», *Comput. J. Inform.*, vol. 6, n.º 2, Art. n.º 2, dic. 2019.
- [23] J. D. L. Castillo, *Desarrollo de aplicaciones Android con Android Studio: Conoce Android Studio*. José Dimas Luján Castillo, 2019.
- [24] A. Nasution, B. Efendi, y I. K. Siregar, «PELATIHAN MEMBUAT APLIKASI ANDROID DENGAN ANDROID STUDIO PADA SMP NEGERI 1 TINGGI RAJA», *Jurdimas J. Pengabd. Kpd. Masy. R.*, vol. 2, n.º 1, Art. n.º 1, feb. 2019, doi: 10.33330/jurdimas.v2i1.321.
- [25] «Modern JavaScript frameworks: A Survey Study | Request PDF». [https://www.researchgate.net/publication/327332343\\_Modern\\_JavaScript\\_frameworks\\_A\\_Survey\\_Study](https://www.researchgate.net/publication/327332343_Modern_JavaScript_frameworks_A_Survey_Study) (accedido 7 de agosto de 2023).
- [26] R. Malik, J. Patra, y M. Pradel, «NL2Type: Inferring JavaScript Function Types from Natural Language Information», may 2019, pp. 304-315. doi: 10.1109/ICSE.2019.00045.
- [27] E. Haro, T. Guarda, A. O. Z. Peñaherrera, y G. N. Quiña, «Backend development for web applications, restful web services: Node.js vs spring boot», *RISTI - Rev. Iber. Sist. E Tecnol. Inf.*, pp. 309-321, ene. 2019.
- [28] «REST API Development with Node.js: Manage and Understand the Full Capabilities of Successful REST Development - Doglio, Fernando: 9781484237144 - AbeBooks». <https://www.abebooks.com/9781484237144/REST-API-Development-Node.js-Manage-1484237145/plp> (accedido 7 de agosto de 2023).
- [29] A. Gupta, H. Sevak, H. Gupta, y R. K. Solanki, «Swiggy Genie Clone Application», vol. 14, n.º 01, 2023.
- [30] S. Aggarwal, «Modern Web-Development using ReactJS», vol. 5, n.º 1, 2018.
- [31] F. F. Nursaid, A. H. Brata, y A. P. Kharisma, «Pengembangan Sistem Informasi Pengelolaan Persediaan Barang Dengan ReactJS Dan React Native Menggunakan Prototype (Studi Kasus: Toko Uda Fajri)», *J. Pengemb. Teknol. Inf. Dan Ilmu Komput.*, vol. 4, n.º 1, Art. n.º 1, feb. 2020.
- [32] T. KhuatKhuat, «Developing a frontend application using ReactJS and ReduxDeveloping a frontend application using ReactJS and Redux».
- [33] «Bioorthogonal chemistry | Nature Reviews Methods Primers». <https://www.nature.com/articles/s43586-021-00028-z> (accedido 7 de agosto de 2023).

- [34] D. Carrizo, A. Alfaro, D. Carrizo, y A. Alfaro, «Método de aseguramiento de la calidad en una metodología de desarrollo de software: un enfoque práctico», *Ingeniare Rev. Chil. Ing.*, vol. 26, n.º 1, pp. 114-129, mar. 2018, doi: 10.4067/S0718-33052018000100114.
- [35] S. M. Velásquez, J. D. V. Montoya, M. E. G. Adasme, E. J. R. Zapata, A. A. Pino, y S. L. Marín, «Una revisión comparativa de la literatura acerca de metodologías tradicionales y modernas de desarrollo de software», *Rev. CINTEX*, vol. 24, n.º 2, Art. n.º 2, dic. 2019, doi: 10.33131/24222208.334.
- [36] J. Zumba Gamboa, «Evolución de las Metodologías y Modelos utilizados en el Desarrollo de Software», *INNOVA Res. J.*, vol. 3, n.º 10, pp. 20-33, 2018.
- [37] B. M. Montero, H. V. Cevallos, y J. D. Cuesta, «Agile methodologies against traditional methods in the software development process.».
- [38] C. Tam, E. J. da C. Moura, T. Oliveira, y J. Varajão, «The factors influencing the success of on-going agile software development projects», *Int. J. Proj. Manag.*, vol. 38, n.º 3, pp. 165-176, abr. 2020, doi: 10.1016/j.ijproman.2020.02.001.
- [39] J. A. Barrera y S. A. Barrera, «Metodologías para el desarrollo de Proyectos», 2020.
- [40] M. K. C. Gonzaga, W. J. O. Pazos, L. J. U. Meneses, y J. A. Esteban, «METODOLOGÍA HÍBRIDA DE DESARROLLO DE SOFTWARE COMBINANDO XP Y SCRUM», *Mikarimin Rev. Científica Multidiscip.*, vol. 5, n.º 2, Art. n.º 2, ago. 2019.
- [41] S. Sánchez Riera, «Método flamelet para la simulación de llamas turbulentas difusivas con swirl», Proyecto/Trabajo fin de carrera/grado, Universitat Politècnica de València, 2019. Accedido: 7 de agosto de 2023. [En línea]. Disponible en: <https://riunet.upv.es/handle/10251/125870>
- [42] J. R. Molina Ríos y M. de las N. Pedreira-Souto, «SWIRL», *metodología para el diseño y desarrollo de aplicaciones web*, 1.ª ed. Editorial Científica 3Ciencias, 2019. doi: 10.17993/IngyTec.2019.55.
- [43] «Identificación de los Stakeholders y conflictos de interés en las Cooperativas de Ahorro y Crédito del Ecuador». [http://www.scielo.org.ar/scielo.php?script=sci\\_arttext&pid=S1668-87082018000200003](http://www.scielo.org.ar/scielo.php?script=sci_arttext&pid=S1668-87082018000200003) (accedido 7 de agosto de 2023).
- [44] «La responsabilidad social empresarial y la sostenibilidad, criterios habilitantes en la gerencia de proyectos». [http://www.scielo.org.co/scielo.php?pid=S1900-38032018000100052&script=sci\\_arttext](http://www.scielo.org.co/scielo.php?pid=S1900-38032018000100052&script=sci_arttext) (accedido 7 de agosto de 2023).
- [45] M. P. Abreu, C. R. R. Rodríguez, y P. Y. P. Pérez, «Computación con palabras para el análisis de factibilidad de proyectos de software», vol. 20, n.º 50, 2016.
- [46] «CONJUNTO HABITACIONAL DE INTERÉS SOCIAL PARA EL BARRIO SAN ANTONIO EN RIOBAMBA. ANÁLISIS DE FACTIBILIDAD.» <https://www.redalyc.org/journal/5717/571763649004/> (accedido 7 de agosto de 2023).
- [47] A. Montenegro, R. Zamora Sánchez, y V. Arias-Villavicencio, «Análisis de factibilidad en la producción de calzado ortopédico para niños en la provincia de Tungurahua», *Retos*, vol. 7, n.º 14, p. 131, sep. 2017, doi: 10.17163/ret.n14.2017.07.
- [48] A. Altamirano, «Identificación de los Stakeholders y conflictos de interés en las Cooperativas de Ahorro y Crédito del Ecuador», *Visión Futuro*, vol. 22, n.º 2, pp. 00-00, jul. 2018.
- [49] K. I. Gómez Sotelo, C. Baron, P. Esteban, C. Y. A. G. Estrada, y L. de J. Laredo Velázquez, «How to find non-functional requirements in system developments», *IFAC-Pap.*, vol. 51, n.º 11, pp. 1573-1578, 2018, doi: 10.1016/j.ifacol.2018.08.272.

[50] A. J. Gregory, J. P. Atkins, G. Midgley, y A. M. Hodgson, «Stakeholder identification and engagement in problem structuring interventions», *Eur. J. Oper. Res.*, vol. 283, n.º 1, pp. 321-340, may 2020, doi: 10.1016/j.ejor.2019.10.044.

## 4 ANEXOS