



UNIVERSIDAD TECNOLÓGICA CENTROAMERICANA

FACULTAD DE INGENIERÍA

PROYECTO DE GRADUACIÓN

**APLICACIÓN MÓVIL PARA SERVICIOS DE VENTA Y ENTREGA A DOMICILIO CON
USO DE INTELIGENCIA ARTIFICIAL, MERCADITO C.H.W**

PREVIO A LA OBTENCIÓN DEL TÍTULO

INGENIERO EN SISTEMAS COMPUTACIONALES

PRESENTADO POR:

11611003 JEAN CARLOS WAH LUNG VALERIANO

ASESOR METODOLÓGICO: LIC. TANIA LUCILA MEZA AMADOR

CAMPUS TEGUCIGALPA; ABRIL, 2021

RESUMEN EJECUTIVO

El presente informe se entrega como requisito para optar el título de Ingeniero en Sistemas Computacionales, en él se detalla el proceso y metodología utilizados para desarrollar una aplicación móvil para servicios de venta y entrega a domicilio con uso de inteligencia artificial y una plataforma administrativa que ayuda a gestionar la aplicación móvil, para el mercadito C.H.W.

Esta aplicación móvil presenta distintas funcionalidades además de ofrecer el servicio de entrega y venta de productos, contiene un chatbot inteligente capaz de ayudar o despejar ciertas dudas que al usuario le surjan y un modelo inteligente de recomendación de productos que ofrece al usuario la posibilidad de agregar a su carrito de compras productos relacionados con los que ya se encuentran en la lista de compras.

El proyecto tuvo un periodo de duración de 6 meses, entre octubre del 2020 y marzo del 2021, durante la creación de este proyecto se utilizaron distintas tecnologías, entre ellas están React Native para la creación de la aplicación móvil, React JS se utilizó para hacer el portal administrativo, Appium para las pruebas automatizadas, ASP.NET Core para la creación de las web APIs y se hizo uso de diferentes servicios de la nube de Microsoft Azure para poder ofrecer distintas funcionalidades, para el chatbot se utilizaron Botframework y LUIS que permite al mismo comprender lenguaje natural, Microsoft Azure Machine Learning Studio para crear el modelo de recomendación de productos, el cual toma como base los productos que se llevan en el carrito de compras para ofrecer al usuario productos relacionados con los mismos y otros servicios que la amplia nube de Microsoft Azure proporciona como el servicio de hosting que se utilizó para la plataforma administrativa, base de datos SQL y contenedores de datos.

Este proyecto también cuenta con una plataforma administrativa, la cual solo puede ser accedida por los administradores de la aplicación móvil, esta permite realizar distintas funciones como crear, eliminar y editar categorías y productos, permite también enviar anuncios publicitarios a los usuarios de la aplicación y generar recordatorios para no olvidar fechas de pago u otras tareas que tenga pendiente el administrador.

EXECUTIVE SUMMARY

This report is delivered as a requirement to obtain the degree of Computer Systems Engineer, it details the process and methodology used to develop a mobile application for sales and home delivery services with the use of artificial intelligence and an administrative platform that helps manage the mobile application, for the C.H.W. market.

This mobile application has different functionalities in addition to offering the service of delivery and sale of products, it contains an intelligent chatbot capable of helping or clearing certain doubts that the user may have and an intelligent model of product recommendation that offers the user the possibility of adding to his shopping cart products related to those already in the shopping list.

The project had a duration period of 6 months, between October 2020 and March 2021, during the creation of this project different technologies were used, among them are React Native for the creation of the mobile application, React JS was used to make the administrative portal, Appium for automated testing, ASP. NET Core for the creation of the web APIs and different services of the Microsoft Azure cloud were used to offer different functionalities, Botframework and LUIS were used for the chatbot that allows it to understand natural language, Microsoft Azure Machine Learning Studio to create the product recommendation model, which is based on the products that are carried in the shopping cart to offer the user products related to them and other services that the extensive Microsoft Azure cloud provides as the hosting service that was used for the administrative platform, SQL database and data containers.

This project also has an administrative platform, which can only be accessed by the administrators of the mobile application, this allows to perform various functions such as creating, deleting and editing categories and products, also allows to send advertisements to users of the application and generate reminders to not forget payment dates or other tasks that the administrator has pending.

ÍNDICE DE CONTENIDO

I.	Introducción.....	1
II.	Planteamiento del Problema.....	3
2.1	Precedentes del problema.....	3
2.2	Definición del problema.....	3
2.3	Justificación.....	3
2.4	Preguntas de Investigación.....	4
2.5	Objetivos.....	4
2.5.1	Objetivo General.....	4
2.5.2	Objetivos Específicos.....	4
III.	Marco Teórico.....	5
IV.	Metodología.....	14
4.1	Enfoque.....	14
4.2	Variables de Investigación.....	14
4.3	Técnicas e Instrumentos Aplicados.....	14
4.3.1	ReactJS.....	14
4.3.2	React Native.....	15
4.3.3	C# y .Net.....	15
4.3.4	Microsoft Azure.....	16
4.3.5	Microsoft Visual Studio.....	16
4.3.6	Appium.....	17
4.3.7	Diseño de Base de Datos.....	18
4.3.8	Casos de Uso.....	19
4.3.9	Diseño de la experiencia de usuario.....	21

4.3.10	Escenarios de Prueba	28
4.4	Metodología de Estudio.....	31
4.4.1	Modelo en Cascada	31
4.5	Cronograma de Actividades	32
V.	Resultados y Análisis.....	33
5.1	Interfaz de Programación de Aplicaciones	33
5.2	Capturas de Pantalla de Plataforma Administrativa.....	33
5.3	Recomendación de Productos.....	43
5.4	Capturas de Pantalla Aplicación Móvil	45
5.5	Pruebas Generales.....	54
VI.	Conclusiones.....	60
VII.	Recomendaciones.....	61
	Bibliografía.....	62
	Anexos.....	65

ÍNDICE DE ILUSTRACIONES

Ilustración 1 - Capas de la nube.....	7
Ilustración 2 - Arquitectura de un sistema de procesamiento de lenguaje natural	9
Ilustración 3 – Funcionamiento Microsoft Azure	12
Ilustración 4 – Funcionamiento servicio de bot en escenario informativo.	13
Ilustración 5 - Diseño de base de datos de la aplicación móvil y plataforma administrativa..	18
Ilustración 6 – Diagrama de casos de uso administrador en la plataforma administrativa.....	19
Ilustración 7 - Diagrama de casos de uso de clientes sobre aplicación móvil.	20
Ilustración 8 - Diagrama de casos de uso de motorista sobre la aplicación móvil.....	21
Ilustración 9 - Pantalla inicio de sesión	22
Ilustración 10 - Pantalla de registro.....	22
Ilustración 11 - Pantalla chat	23
Ilustración 12 - Pantalla menú principal.....	23
Ilustración 13 - Pantalla información.....	24
Ilustración 14 - Pantalla editar información.....	24
Ilustración 15 - Pantalla de pedidos.	25
Ilustración 16 - Pantalla resumen de pedido.....	25
Ilustración 17 - Pantalla rastreo de pedido.....	26
Ilustración 18 - Pantalla de productos.....	26
Ilustración 19 - Pantalla de compra	27
Ilustración 20 - Pantalla de carrito de compras.....	27
Ilustración 21 - Cronograma de actividades.....	32
Ilustración 22 - Pantalla inicio de sesión.....	33
Ilustración 23 - Pantalla gestion de usuarios.....	34
Ilustración 24 - Pantalla creación de usuario.....	34

Ilustración 25 - Pantalla edición de usuario.....	35
Ilustración 26 - Pantalla eliminación de usuario	35
Ilustración 27 - Pantalla gestión de productos.....	36
Ilustración 28 - Pantalla creación de producto.....	36
Ilustración 29 - Pantalla edición de producto	37
Ilustración 30 - Pantalla gestión de roles.....	37
Ilustración 31 - Pantalla creación de rol.....	38
Ilustración 32 - Pantalla edición de rol.....	38
Ilustración 33 - Pantalla gestión de categorías.....	39
Ilustración 34 - Pantalla creación de categoría.....	39
Ilustración 35 - Pantalla edición de categoría.....	40
Ilustración 36 - Pantalla gestión de anuncios	40
Ilustración 37 - Pantalla creación de anuncio	41
Ilustración 38 - Pantalla edición de anuncio.....	41
Ilustración 39 - Pantalla gestión de recordatorios	42
Ilustración 40 - Pantalla creación de recordatorio	42
Ilustración 41 - Pantalla edición de recordatorio.....	43
Ilustración 42 - Modelo de recomendación de productos	44
Ilustración 43 - Icono aplicación móvil.....	45
Ilustración 44 - Pantalla inicio de sesión.....	45
Ilustración 45 - Pantalla de registro	46
Ilustración 46 - Pantalla confirmación de registro	46
Ilustración 47 - Pantalla información personal.....	47
Ilustración 48 - Pantalla edición información de usuario	47
Ilustración 49 - Pantalla de pedidos	48

Ilustración 50 - Pantalla pedido entregado.....	48
Ilustración 51 - Pantalla de pedido activo.....	49
Ilustración 52 - Pantalla de rastreo de pedido.....	49
Ilustración 53 - Pantalla de categorías.....	50
Ilustración 54 - Pantalla de productos.....	50
Ilustración 55 - Pantalla carrito de compras.....	51
Ilustración 56 - Pantalla de compra.....	51
Ilustración 57 - Pantalla del chatbot.....	52
Ilustración 58 - Ejemplo consulta sobre un producto.....	52
Ilustración 59 - Ejemplo reinicio de contraseña.....	53
Ilustración 60 - Ejemplo ver información.....	53
Ilustración 61 - Ejemplo calificar chatbot.....	54
Ilustración 62 - Prueba de comunicación con el chatbot.....	55
Ilustración 63 - Prueba de creación de pedido.....	56
Ilustración 64 - Prueba de agregar y visualizar productos recomendados.....	56
Ilustración 65 - Prueba de visualización de pedidos activos e inactivos.....	57
Ilustración 66 - Prueba de rastreo de pedido.....	57
Ilustración 67 - Pruebas de estrés con 20 hilos.....	58
Ilustración 68 - Pruebas de estrés con 100 hilos.....	59

ÍNDICE DE TABLAS

Tabla 1 - Rutas del API.....	65
------------------------------	----

ÍNDICE DE ANEXOS

Anexo 1 - Tabla de rutas del API.....	65
---------------------------------------	----

LISTADO DE SIGLAS Y GLOSARIO

API	Application Programming Interface
IA	Inteligencia Artificial
IoT	Internet of Things
IaaS	Infrastructure as a Service
REST	Representational State Transfer
PaaS	Platform as a Service
SaaS	Software as a Service
PLN	Procesamiento de Lenguaje Natural
UAT	User Acceptance Testing
LUIS	Language Understanding Intelligent Service
DOM	Document Object Model
CLR	Common Language Runtime
JSON	Notación de Objeto de Javascript

Microsoft Azure: es la plataforma de computación de Microsoft, esta permite obtener distintos tipos de servicios en la nube enfocándose en ofrecer una plataforma flexible y abierta, esta nube pública ofrece seguridad, privacidad y completa transparencia (Meza, E. R., & Imbachi, 2016).

Computación en la nube: la computación en la nube consiste en los servicios ofrecidos a través de la red como: correo electrónico, almacenamiento, aplicaciones, entre otras., estos normalmente se pueden usar a través de un navegador web. Estos servicios son procesados y ejecutados por un servidor en internet (HernandezN. L., & Florez-FuentesA. S., 2014).

I. INTRODUCCIÓN

“Una de las únicas formas de salir de una caja cerrada es inventar la salida” (Jeff Bezos).

En la actualidad, el mundo está en constante evolución, y nadie quiere quedarse atrás, para ello múltiples instituciones, empresas y negocios de distintos tipos de rubro, están usando herramientas tecnológicas para poder avanzar y competir dentro del mercado.

Con la gran evolución tecnológica que se ha dado en los últimos años, se pueden realizar cosas anteriormente inimaginables. Esto le está cambiando a las personas la forma de percibir el mundo. Un claro ejemplo de esto es la inteligencia artificial que es una rama de las ciencias computacionales encargada de estudiar modelos de cómputo capaces de realizar actividades propias de los seres humanos en base a dos de sus características primordiales que son el razonamiento y la conducta (Takeyas, B. L., 2007), esta puede ser utilizada para desempeñar múltiples tareas. Algunas de sus áreas son el procesamiento de lenguaje natural y el aprendizaje automático.

El procesamiento de lenguaje natural facilita la comunicación entre el humano y la máquina, ya que permite usar palabras o frases que se usan de manera cotidiana para comunicarse con una máquina. Por su lado el aprendizaje automático utiliza técnicas estadísticas y algoritmos computacionales para dar la capacidad a los ordenadores de aprender, de esta forma mejorar sus resultados en una tarea específica después de procesar los datos (Álvarez Vega, M., Quirós Mora, L. M., & Cortés Badilla, M. V. 2020).

Existen herramientas en el internet que permiten utilizar servicios de Inteligencia artificial en la nube, como Microsoft Azure, la mezcla entre estas herramientas y otras tecnologías dan la posibilidad de desarrollar distintas cosas entre ellas plataformas web y aplicaciones móviles poderosas.

En Mercadito C.H.W ubicado en el departamento de Francisco Morazán, se tiene una enorme oportunidad de crecimiento en el proceso de venta de productos varios, actualmente hay limitaciones tanto en espacio físico, ya que es un lugar que tiene un espacio reducido, como de expansión ya que solo puede realizar ventas en el área donde está ubicado.

Actualmente se cuenta con servicio de venta en caja, este lugar podría crecer permitiendo a sus clientes adquirir los productos mediante una aplicación móvil la cual pueda ayudar al mismo con recomendaciones de algunos productos o despejando consultas que surjan antes o durante el proceso de compra.

Lo que se quiere lograr con el proyecto es facilitar la experiencia del usuario y ayudarlo durante el proceso de compra. De esta forma se mitigaría el problema de espacio físico ya que la clientela tendría la facilidad de poder realizar pedidos a sus domicilios con los productos de su preferencia y también se podría alcanzar a más clientes ya que se eliminaría la limitación de área.

En este informe se encuentra detallado el proceso e información relacionada con el desarrollo de una aplicación móvil para brindar los servicios de venta y entrega a domicilio con uso de inteligencia artificial que presentará a los clientes recomendaciones para sus compras mediante análisis predictivo y también logrará ayudar resolviendo consultas y gestiones mediante un chatbot para así brindar una mejor experiencia al usuario, dividido en distintas secciones. En el planteamiento del problema se expone la problemática que se quiere resolver, justificando porque es importante resolver dicho problema y los objetivos que se quieren alcanzar. En el marco teórico se presenta información de mucha importancia que le servirá al lector para aprender un poco sobre las técnicas y tecnologías que se usarán en el desarrollo del proyecto. En la metodología se exponen las actividades, herramientas y estrategias que se utilizarán para alcanzar los objetivos planteados. Se concluirá con los resultados y análisis obtenidos al final del desarrollo del proyecto.

II. PLANTEAMIENTO DEL PROBLEMA

2.1 PRECEDENTES DEL PROBLEMA

Durante el caótico 2020 con el tema de la pandemia se han visto afectados muchos negocios de distintos tipos de rubros, varios de ellos han optado por utilizar herramientas tecnológicas para lograr avanzar y seguir compitiendo en el mercado, el Mercadito C.H.W es uno de los afectados, ya que sus ventas han disminuido porque las personas optan por salir la menor cantidad de veces posibles a realizar compras y con el surgimiento de muchas aplicaciones de entregas a domicilio las personas optan por ordenar productos o comida a sus domicilios con el afán de no salir.

2.2 DEFINICIÓN DEL PROBLEMA

El solo realizar ventas en su local físico trae ciertas desventajas : se tiene que hacer mucha fila en ciertas ocasiones cuando el local está lleno, al no tener un amplio parqueo alguna clientela muchas veces opta por comprarle a la competencia, está limitado a solo vender a personas dentro de su ubicación geográfica, al haber aglomeraciones de gente comprando se suele tener un menor control por parte del personal muchas veces descuidando la atención al cliente , no percibiendo ciertos casos de hurto de productos por parte de algunos clientes o incluso olvidando hacer labores de vital importancia en su día a día.

2.3 JUSTIFICACIÓN

La creación de una aplicación móvil aplica perfectamente al problema, ya que mejoraría la experiencia al usuario en su compra mediante las recomendaciones personalizadas y generales que obtendrá gracias al análisis predictivo al igual que con el chatbot que podrá resolver ciertas dudas o consultas que surjan por parte del usuario, reduciría las filas y flujo de gente dentro de su local permitiendo al personal prestar un mejor servicio y tener un mayor control de sus labores, así mismo lograría expandirse ya que podría llegar a generar ventas fuera de la zona.

2.4 PREGUNTAS DE INVESTIGACIÓN

- ¿Cómo se define, entrena e implementa el modelo de aprendizaje automático con Microsoft Azure?
- ¿Cómo se implementa un modelo de Procesamiento de lenguaje con Microsoft Azure?
- ¿Qué beneficios obtendrá el Mercadito con la creación de la aplicación móvil?

2.5 OBJETIVOS

2.5.1 OBJETIVO GENERAL

Crear una aplicación disruptiva que facilite la experiencia del usuario y lo acompañe en el proceso de precompra, compra y postcompra, mediante el uso de inteligencia artificial que permitirá al usuario despejar consultas y obtener recomendaciones sobre productos de su interés.

2.5.2 OBJETIVOS ESPECÍFICOS

- Entrenar e implementar un modelo de aprendizaje automático utilizando Microsoft Azure.
- Implementar un modelo de procesamiento de lenguaje natural utilizando Microsoft Azure.
- Desarrollar una plataforma administrativa para facilitar el manejo de los productos por parte del personal y envío de promociones.
- Desarrollar una aplicación móvil que haga uso de inteligencia artificial para facilitar a los usuarios el proceso de compra, precompra y post compra.
- Realizar pruebas unitarias, de integración, de estrés y de aceptación del usuario para asegurar la calidad del software.

III. MARCO TEÓRICO

Para poder comprender el proceso de desarrollo, implementación y análisis de una aplicación de ventas y entrega a domicilio con uso de inteligencia artificial, a continuación, se podrán observar varios términos y herramientas que son de vital importancia para el desarrollo de este proyecto.

Para entrar un poco en contexto se comenzará hablando sobre computación en la nube, la computación en la nube consiste en los servicios ofrecidos a través de la red como: correo electrónico, almacenamiento, aplicaciones, entre otras., estos normalmente se pueden usar a través de un navegador web. Estos servicios son procesados y ejecutados por un servidor en internet (HernandezN. L., & Florez-FuentesA. S., 2014).

Según Hernández y Flórez (2014) algunas características de la computación en la nube pueden ser:

- Autoservicio: el consumidor puede obtener de manera unilateralmente capacidades de computación, como tiempo de servidor y almacenamiento en la red, según lo que requiera de manera automática sin necesitar la interacción humana con cada proveedor de servicios.
- Reservas de un recurso en común: Los recursos computacionales se pueden reservar, de esta forma cuando múltiples usuarios utilicen estos recursos no habrá ningún problema, ya que los recursos físicos y virtuales son asignados de manera dinámica y reasignados en base a la demanda de los usuarios.
- Rapidez y elasticidad: Las capacidades pueden suministrarse de manera rápida y elástica. Para los consumidores las capacidades disponibles casi siempre aparecen como ilimitadas y pueden adquirirse distintas cantidades y en el momento que se desee.
- Servicio Supervisado: los servicios en la nube controlan y optimizan el uso de los recursos de manera automática, además se pueden controlar, esto permite transparencia tanto para el proveedor como para el consumidor de dicho servicio.
- Auto reparable: En caso de algún fallo, el último respaldo se convierte en la copia primaria y a partir de esta se genera uno nuevo.

- Escalabilidad: este tipo de sistema suele ser predecible y eficiente ya que crea nuevas instancias de acuerdo con la demanda que se produzca de tal forma reduce el tiempo de espera y los cuellos de botella.
- Virtualización: las aplicaciones son independientes del hardware en que corran.
- Seguridad: El sistema permite a diferentes usuarios compartir infraestructura sin comprometer su seguridad y privacidad, ya que el proveedor se encarga de cifrar los datos.
- Disponibilidad: la información puede ser accedida desde cualquier dispositivo al estar en internet.

Según Mejía, O (2011) la arquitectura en la nube se basa en realizar una separación entre hardware, plataforma y aplicaciones, quedando de esta forma en distintas capas las cuales son:

- Software como Servicio (SaaS) : esta consiste en entregar aplicaciones completas como un servicio, estas aplicaciones las pueden usar múltiples usuarios a través de la red, sin que estos tengan que instalar ningún tipo de software adicional.

Los proveedores de SaaS son responsables de la disponibilidad y funcionalidad de sus servicios sin dejar de lado las necesidades de los clientes los cuales finalmente consumirán el software.

- Plataforma como Servicio (PaaS): esta proporciona un servicio de plataforma con todo lo necesario para dar soporte al ciclo de planteamiento, desarrollo y despliegue de aplicaciones y servicios web.

El proveedor de PaaS se encarga de escalar los recursos en caso de que la aplicación lo requiera, que esta tenga un rendimiento óptimo y de la seguridad de acceso etc.

- Infraestructura como Servicio (IaaS): esta consiste en hacer el uso externo de servidores para espacio en disco, base de datos, switches, routers, así como también tiempo computacional, de esta forma se evita tener un servidor local y toda la infraestructura necesaria.

Con la IaaS solo se tienen que pagar los recursos que son utilizados.

En la Ilustración 1 se muestran las distintas capas de la nube.

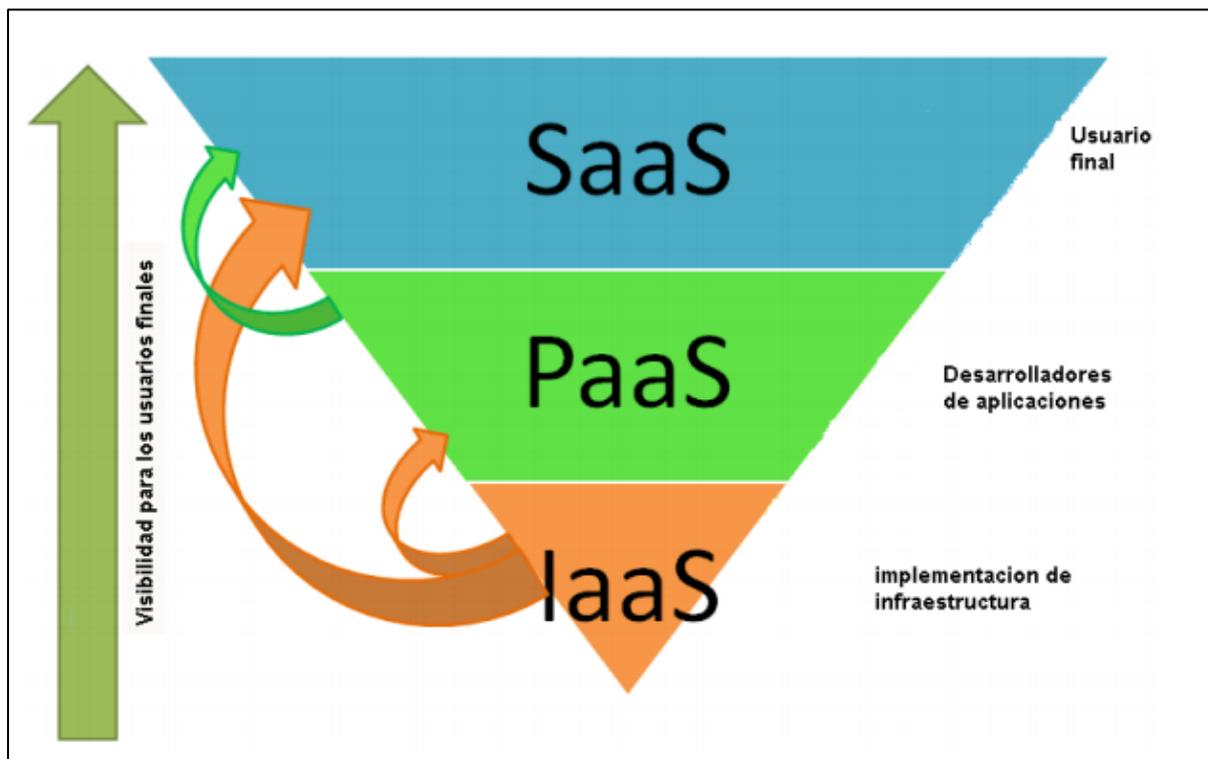


Ilustración 1 - Capas de la nube

Fuente: (Mejía, O., 2011)

La computación en la nube facilita múltiples tareas, algunas plataformas de computación en la nube permiten utilizar servicios de inteligencia artificial, la inteligencia artificial es la rama de la ciencia que se encarga del estudio de la inteligencia en elementos artificiales, desde el punto de vista de ingeniería propone la creación de elementos que posean un comportamiento inteligente. La Inteligencia Artificial (IA) pretende construir sistemas y máquinas que presenten un comportamiento que si fuera ejecutado por una persona, se diría que es inteligente (Romero, J. J., Dafonte, C. A. R. L. O. S., Gómez, Á. N. G. E. L., & Penousal, F. J., 2007).

Una de las ramas de la IA es el Procesamiento de Lenguaje Natural (PLN), para comenzar a hablar sobre PLN primero se entenderá qué es un lenguaje, este se puede definir de forma lingüística como una función que expresa pensamientos y comunicaciones entre la gente. Esto se puede realizar de manera escrita, vocal o con señales (Cortez Vásquez, A., Vega huerta, H., Pariona Quispe, J., & Huayna, A. M. , 2014).

Cuando se habla de Lenguaje Natural (LN) se refiere al medio que se utiliza de manera cotidiana para la comunicación con las demás personas.

Ahora que ya se está un poco más en contexto se hablará un poco de qué es el procesamiento de lenguaje natural, una de las tareas principales de la inteligencia artificial es la manipulación de lenguajes naturales usando herramientas computacionales (Cortez Vásquez, A., Vega huerta, H., Pariona Quispe, J., & Huayna, A. M. , 2014).

La idea principal del PLN es utilizar lenguaje natural para permitir la comunicación con la computadora.

Una ventaja del PLN es que el locutor no tiene que esforzarse para aprender el medio de comunicación a diferencia de otros medios de interacción como los lenguajes de comando o las interfaces gráficas (Cortez Vásquez, A., Vega huerta, H., Pariona Quispe, J., & Huayna, A. M. , 2014).

Una desventaja es que su uso también tiene ciertas limitaciones ya que la computadora tiene una comprensión limitada del lenguaje.

Algunas cosas en las que se puede usar el PLN son (Cortez Vásquez, A., Vega huerta, H., Pariona Quispe, J., & Huayna, A. M. , 2014).

- Traducción automática
- Recuperación de la información
- Extracción de información y resúmenes
- Reconocimiento de voz

Dentro de la arquitectura de un sistema PLN se encuentran distintos niveles estos son (Cortez Vásquez, A., Vega huerta, H., Pariona Quispe, J., & Huayna, A. M. , 2014).

- Nivel Fonológico: consiste en cómo las palabras se relacionan con los sonidos que representan.
- Nivel Morfológico: consiste en cómo las palabras se construyen a partir de unas unidades de significado más pequeñas.
- Nivel Sintáctico: consiste en cómo las palabras pueden unirse formando de esta forma oraciones.
- Nivel Semántico: consiste en el significado de las palabras y en cómo estos se unen para dar sentido a una oración.

- Nivel Pragmático: consiste en cómo las oraciones se usan en distintas situaciones y de cómo el uso afecta al significado de una oración.

En la Ilustración 2 se muestra la arquitectura de un sistema de procesamiento de un lenguaje natural.

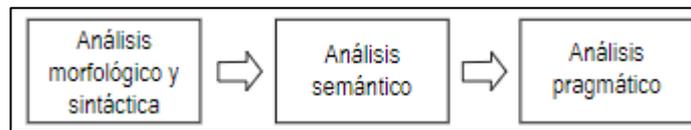


Ilustración 2 - Arquitectura de un sistema de procesamiento de lenguaje natural

Fuente: (Cortez Vásquez, A., Vega huerta, H., Pariona Quispe, J., & Huayna, A. M. , 2014).

Otra rama de la IA que va de la mano con el PLN es el aprendizaje automático, este se define como una tecnología que utiliza técnicas estadísticas y algoritmos computacionales para dar la capacidad a los ordenadores de aprender, de esta forma mejorar sus resultados en una tarea específica después de procesar los datos (Álvarez Vega, M., Quirós Mora, L. M., & Cortés Badilla, M. V. 2020).

El aprendizaje automático o machine learning es una rama de la inteligencia artificial debido a que aprende analizando datos y luego toma su propia decisión mediante el uso de algoritmos. Este es capaz de reconocer ciertos tipos de patrones permitiéndole aprender de errores y predecir resultados (Álvarez Vega, M., Quirós Mora, L. M., & Cortés Badilla, M. V. 2020).

Algunas categorías del aprendizaje automático son (Álvarez Vega, M., Quirós Mora, L. M., & Cortés Badilla, M. V. 2020):

- Aprendizaje supervisado: consiste en inicialmente entrenar la máquina proporcionando entradas que estén asociadas a un resultado conocido determinado por expertos humanos. La idea de estos algoritmos es aprender reglas generales que mapean entradas con salidas, de esta forma la máquina gana experiencia con los grandes números de datos proporcionados y aprende distintos patrones, para luego obtener resultados acertados la próxima vez que se le dé una entrada.
- Aprendizaje no supervisado: En este tipo de aprendizaje no se proporciona información previa, sino que se introducen grandes cantidades de datos no etiquetados y el sistema encuentra distintos patrones y automáticamente la separa en grupos.

- Aprendizaje por refuerzo: en este tipo de aprendizaje se le proporcionan datos etiquetados y no etiquetados a la máquina, este interactúa con el entorno y recibe respuestas negativas o positivas y estas permiten que vaya aprendiendo para lograr desarrollar clasificaciones más acertadas.

Como se puede observar el aprendizaje automatizado es una herramienta de vital importancia y se puede aplicar a muchos casos de uso como, por ejemplo: detección de software malicioso, comprensión de textos, análisis de imágenes, detección de fraudes, predicción de fallos, reconocimiento de voz, etc.

La aplicación del PLN y el aprendizaje automático pueden usarse en la creación de un chatbot o asistente virtual. Los chatbots son programas informáticos que tienen la habilidad de interactuar con personas utilizando interfaces basadas en el lenguaje. Su propósito es simular una conversación humana inteligente, de esta forma se busca que el usuario tenga una experiencia de conversación lo más parecida a una conversación humana (García Brustenga, Fuertes Alpiste, y Molas Castells, 2018).

El funcionamiento de los chatbots parte del uso de lenguaje natural, pero puede estar basado en conversaciones de flujo definido basadas en interacciones estructuradas que aunque son limitadas, generan pocas ambigüedades de significado (García Brustenga, Fuertes Alpiste, y Molas Castells, 2018).

Según García Reina, L. F (2018) existen distintos tipos de chatbots de acuerdo con el servicio que prestan:

- Chatbot de ventas, los cuales están orientados a la comercialización de productos o servicios.
- Chatbot de servicio al cliente, se han orientado a la resolución de dudas que los clientes tienen en cuanto a un determinado servicio.
- Chatbot de noticias y contenido, estos se han logrado implementar mediante canales de mensajería instantánea, su principal objetivo se ha centrado en el envío de contenido masivo mediante estos diferentes canales.

Los alcances de los chatbots pueden ser positivos como negativos, los primeros se basan en aspectos como reducción de los tiempos de respuesta, autoservicio, optimización de los costos e integración omnicanal, y los segundos, son aspectos sociales, como la falta de humanidad

que hay en ellos, esto puede ocasionar el riesgo de perder clientes sino se configuran de manera correcta (Weeks, 2018).

Para la utilización de servicios de IA existen distintas plataformas en la nube, como por ejemplo, la plataforma en la nube de Microsoft Azure, a continuación se hablará un poco sobre Microsoft Azure.

Microsoft Azure es la plataforma de computación de Microsoft, esta permite obtener distintos tipos de servicios en la nube enfocándose en ofrecer una plataforma flexible y abierta, esta nube pública ofrece seguridad, privacidad y completa transparencia (Meza, E. R., & Imbachi, D. R. V, 2016).

En Azure existen roles enfocados a sitios web, procesamiento, conocido como trabajadores o workers estos son roles dentro de la modalidad PaaS y máquinas virtuales bajo IaaS, que manejan diferentes tarifas, estas únicamente se pagan por los servicios consumidos (Meza, E. R., & Imbachi, D. R. V, 2016).

Microsoft Azure ofrece más de 50 servicios y características al mercado que están divididos en grupos según la necesidad de los consumidores, enmarcando en PaaS e IaaS, Azure oferta estos dos modelos a un nivel empresarial donde los escenarios pueden ser muy complejos, para temas de seguridad, cómputo, analítica de datos, IOT, almacenamiento, entre otros. (Ramos Pérez, L. G, 2019)

En la Ilustración 3 se muestra de manera muy general cómo trabaja Microsoft Azure.

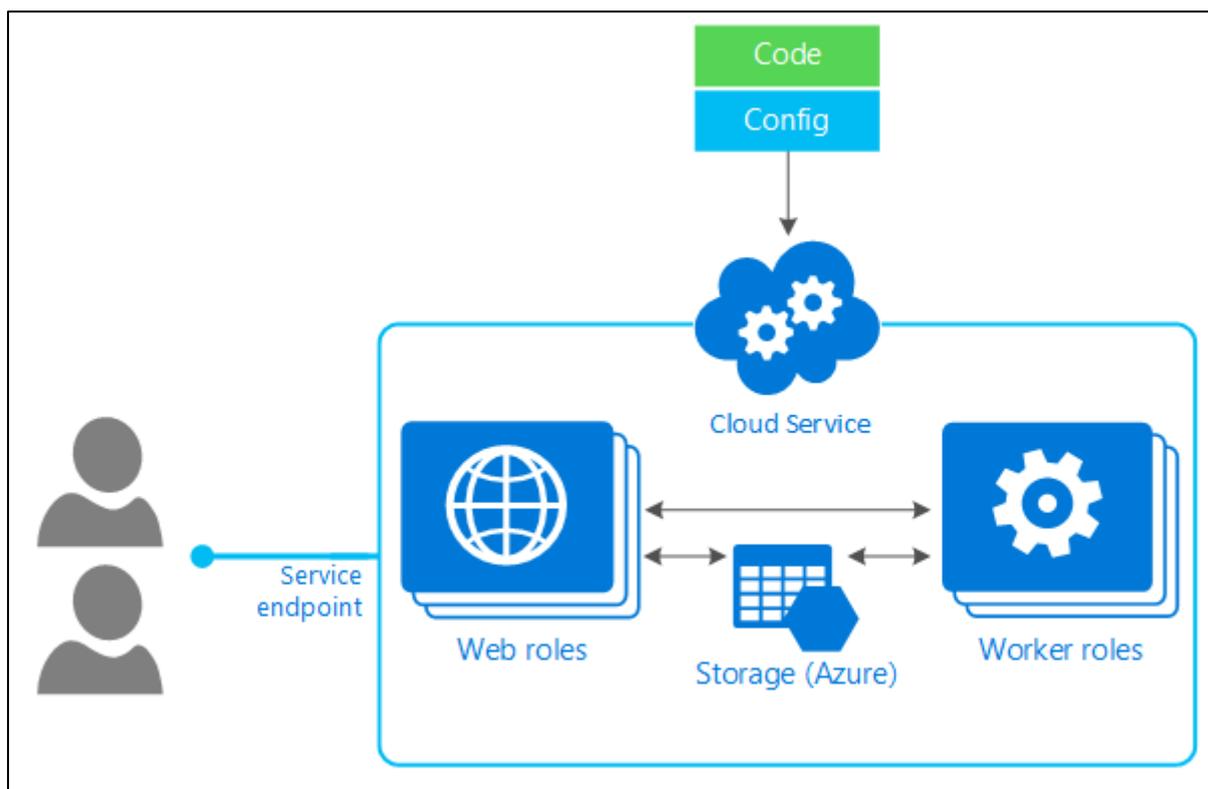


Ilustración 3 – Funcionamiento Microsoft Azure

Fuente: (Microsoft Docs, 2020).

Uno de los servicios de IA pertenecientes a la plataforma de Microsoft Azure que se utilizan para la implementación de chatbots es el de Language Understanding Intelligent Service (LUIS) es un servicio basado en el aprendizaje automático, el cual busca desarrollar la comprensión del lenguaje en aplicaciones, bots y dispositivos de Internet de las Cosas (IoT) (Microsoft Azure, 2020), este permite a los desarrolladores sin experiencia en aprendizaje automático que puedan crear y utilizar modelos de comprensión del lenguaje específicos a sus tareas (Williams, J. D., Kamal, E., Ashour, M., Amr, H., Miller, J., & Zweig, G., 2015).

LUIS utiliza el aprendizaje automático para permitir a los desarrolladores crear aplicaciones que puedan entender el lenguaje natural de los usuarios para extraer su significado y comprender lo que el usuario quiere (Microsoft Azure, 2020).

El potencial y facilidad de uso que ofrece LUIS es fantástico para el desarrollo de un chatbot, este junto con Azure Bot Service, proporcionan herramientas para crear, probar, implementar

y administrar bots inteligentes, todo en un solo lugar. El uso de estas herramientas permite crear bots que usen el habla, comprendan el lenguaje natural, manejen preguntas y respuestas y otras cosas más (Microsoft Docs, 2019).

En la Ilustración 4 se muestra el funcionamiento de un servicio de bot en un escenario informativo.

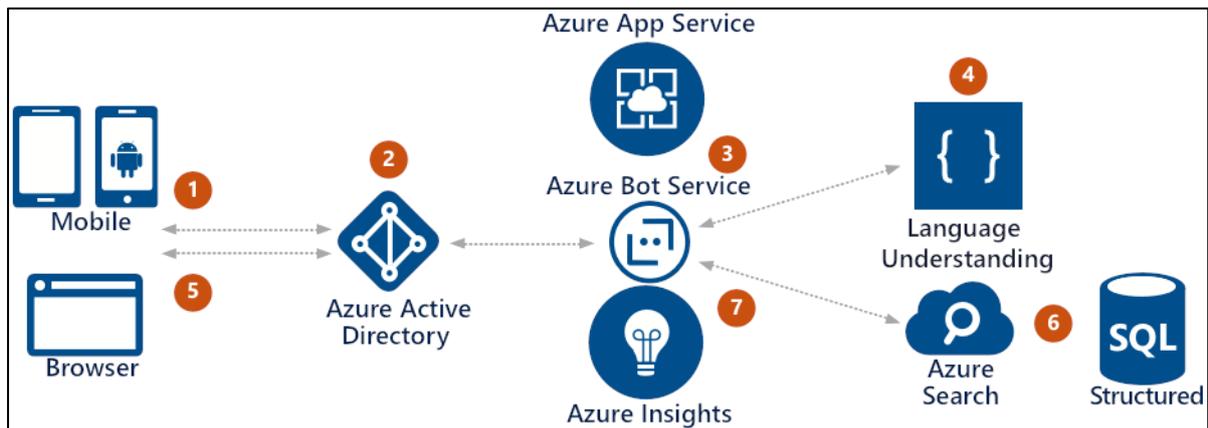


Ilustración 4 – Funcionamiento servicio de bot en escenario informativo.

Fuente: (Microsoft Docs, 2017).

Otra tecnología que viene de la mano con las aplicaciones móviles en la actualidad son las notificaciones push, esta es una forma de comunicación en la que una aplicación web envía un mensaje a una aplicación móvil, siempre que exista nueva información. En esta tecnología el servidor es el que inicia este tipo de comunicación, y no es indispensable que el cliente tenga interés en saber el instante en que existe nueva información. El servidor siempre lo notificará (Soldado Zuña, A. F., 2018).

Según Soldado Zuña, A. F (2018), la principal ventaja de esta tecnología es la inmediatez con la que se reciben las notificaciones push. Esto es debido a la posibilidad de enviar y recibir notificaciones de un sistema a otro, tanto si las aplicaciones están siendo usadas en ese mismo instante, si están ejecutándose en segundo plano, si todavía no han sido iniciadas, o incluso si el dispositivo está en modo reposo. Por lo tanto, las notificaciones push despiertan al dispositivo móvil se esté ejecutando o no una aplicación.

Todas estas herramientas y conceptos que se observaron ayudarán en el proceso de desarrollo de este proyecto.

IV. METODOLOGÍA

4.1 ENFOQUE

Este proyecto utiliza un enfoque cualitativo, ya que se enmarca en el paradigma científico naturalista, el cual, también es denominado naturalista-humanista o interpretativo, y cuyo interés “se centra en el estudio de los significados de las acciones humanas y de la vida social” (Barrantes, 2014), ya que luego de desarrollar el proyecto se busca hacer un análisis por medio de cuestionarios y encuestas de satisfacción, de esta forma se observará la aceptación hacia la aplicación móvil y su inteligencia artificial, por parte de los usuarios al añadir la nueva forma de comprar los productos de su preferencia.

Este enfoque permitirá observar la manera en la que los usuarios se sienten con respecto a la aplicación, permitirá ver cuáles son los posibles problemas o dificultades que se presenten y permitirá observar si los usuarios conviven de buena forma con la inteligencia artificial que esta aplicación provee.

4.2 VARIABLES DE INVESTIGACIÓN

- Tiempo de respuesta de la aplicación.
- Nivel de aceptación de la aplicación por parte de los clientes y el personal.
- Nivel de satisfacción en cuanto a la nueva forma de atención al cliente, en comparación con la forma tradicional.

4.3 TÉCNICAS E INSTRUMENTOS APLICADOS

Tras realizar una investigación previa sobre las distintas tecnologías disponibles para el desarrollo de este proyecto, se seleccionaron las siguientes, ya que se consideran las más adecuadas para la creación tanto de la plataforma administrativa como para la aplicación móvil.

4.3.1 REACTJS

React JS es una librería que funciona bajo JavaScript, por lo que es de código abierto. Así, tiene la ventaja de que no sólo se utiliza del lado del cliente, sino también puede estar presente del lado del servidor y trabajar en conjunto (Calixto, González, Díaz & Guzmán, 2019). Además, tiene

como objetivo el desarrollo de interfaces de usuario. Esta tecnología ayuda en el desarrollo de todos los componentes de frontend, se decidió utilizar esta librería para la creación de la plataforma administrativa que será de tipo web, ya que permite muchas facilidades como:

- Poder generar el Modelo de Objetos del Documento (DOM) de forma dinámica. Esto permite visualizar los cambios de manera inmediata sin necesidad de renderizar toda la página de nuevo, sino que solamente el componente que haya sido modificado, esto ayuda a desarrollar de manera más cómoda y rápida.
- Cuenta con una gran comunidad por ende existen muchas librerías externas que facilitan las diversas tareas que se quieran realizar
- Cada componente tiene su propio estado, este puede ser modificado durante su ciclo de vida.

4.3.2 REACT NATIVE

En el tema de desarrollo del frontend de la aplicación móvil se utilizará React Native este es un framework para el desarrollo de aplicaciones nativas multiplataforma que se basa en Javascript y ReactJS, este en lugar de ser ejecutado en el navegador, corre directamente sobre las plataformas móviles nativas, además permite ciertas facilidades como:

- Es multiplataforma por ende se pueden crear aplicaciones que pueden ser ejecutadas tanto en IOS como Android con el mismo código base.
- Permite refrescar en tiempo de ejecución, lo que permite desarrollar aplicaciones móviles de manera más cómoda y rápida.

4.3.3 C# Y .NET

Para la creación de las APIs de Transferencia de Estado Representacional (REST) se utilizarán estas dos tecnologías ya que junto con Microsoft Azure permitirán desplegar los servicios para luego consumirlos desde los frontend.

C# es un lenguaje orientado a objetos elegante y con seguridad de tipos, permite a los desarrolladores crear muchos tipos de aplicaciones seguras y sólidas que se ejecutan en el sistema .NET proporciona funciones poderosas, aceptación de valores vacíos, delegados,

expresiones lambda, coincidencia de patrones, acceso directo y seguro a la memoria, polimorfismo, encapsulación, herencia, entre otras (Microsoft Docs, 2020).

.NET es un entorno de ejecución runtime que administra aplicaciones cuyo destino es .NET Framework, este incorpora Common Language Runtime (CLR), que proporciona la administración de la memoria y otros servicios del sistema, y una biblioteca de clases completa, que permite a los desarrolladores aprovechar el código estable y fiable de todas las áreas principales del desarrollo de aplicaciones (Microsoft Docs, 2020).

4.3.4 MICROSOFT AZURE

El proyecto hace gran uso del ecosistema proporcionado por Microsoft Azure, este proyecto integrará distintos servicios de IA Microsoft Azure como Language Understanding y Azure Bot Framework para la integración de un chat Bot y Machine Learning para el análisis predictivo que permitirá a los usuarios obtener recomendaciones al momento de realizar sus compras, Además ayudará a desplegar los distintos servicios que se desarrollen en C# y NET CORE para el backend y proporcionará el servicio de base de datos en la nube. Este poderoso ecosistema facilitará el desarrollo del proyecto, ya que cuenta con una amplia gama de servicios en la nube.

4.3.5 MICROSOFT VISUAL STUDIO

Microsoft Visual Studio es un entorno de desarrollo integrado para Windows y macOS. Es compatible con múltiples lenguajes de programación y entornos de desarrollo web, este permite a los desarrolladores crear sitios y aplicaciones web así como servicios web en cualquier entorno compatible con la plataforma .NET.

Este cuenta con algunas características como (Visual Studio, 2020):

- Escritura de código con menos errores, ya que cuenta con IntelliSense que proporciona sugerencias de código.
- Desarrollo local con gran número de emuladores comunes.
- Acceso simplificado a las pruebas de explorador de soluciones.
- Creación de repositorios y administración GIT en IDE

- Compatibilidad con Kubernetes incluida ahora en la carga de trabajo de Microsoft Azure.
- Conexión a la nube
- Facilidad en el desarrollo con Azure.

En este proyecto se utilizará este entorno, ya que este tiene una facilidad de integración con el desarrollo de Azure, esto permitirá el desarrollo de forma más cómoda y rápida.

4.3.6 APPIUM

Para la realización de algunas pruebas tanto unitarias como de integración se utilizará una herramienta llamada Appium.

Appium es una herramienta de código abierto para automatizar aplicaciones nativas, web móviles e híbridas en plataformas móviles Android o iOS y Windows.

Las aplicaciones nativas son aquellas que están escritas con los SDK de IOS, Android o Windows.

Las aplicaciones web móviles son aplicaciones web a las que se accede mediante un navegador móvil.

Las aplicaciones híbridas son aquellas que tienen un envoltorio alrededor de una vista web, un control nativo que permite la interacción con el contenido web.

Appium fue diseñado para satisfacer las necesidades de automatización móvil de acuerdo con una filosofía delineada por los siguientes cuatro principios (Hans, 2015):

- No debería tener que volver a compilar su aplicación o modificarla de ninguna manera para automatizarla.
- No debe estar encerrado en un lenguaje o marco específico para escribir y ejecutar las pruebas.
- Un marco de automatización móvil no debería reinventar la rueda cuando se trata de API de automatización.
- Un marco de automatización móvil debe de ser código abierto.

Appium tiene una arquitectura cliente-servidor, es un servidor HTTP escrito en Node.js que crea y maneja sesiones de webdriver. El servidor web de Appium recibe solicitudes HTTP de bibliotecas cliente a través de Notación de Objeto de Javascript (JSON) y luego maneja estas solicitudes de diferentes maneras dependiendo de la plataforma en la que se esté ejecutando (Appium, 2020).

4.3.7 DISEÑO DE BASE DE DATOS

Para el diseño de la base de datos se decidió utilizar una base de datos relacional, en este caso se utilizará una base de datos SQL que estará en el portal de Azure. Este tipo de base de datos es beneficioso para este proyecto, ya que garantiza la integridad de los datos y cuenta con una estructura bien definida.

En la Ilustración 5 se presenta el diseño de la base de datos del sistema.

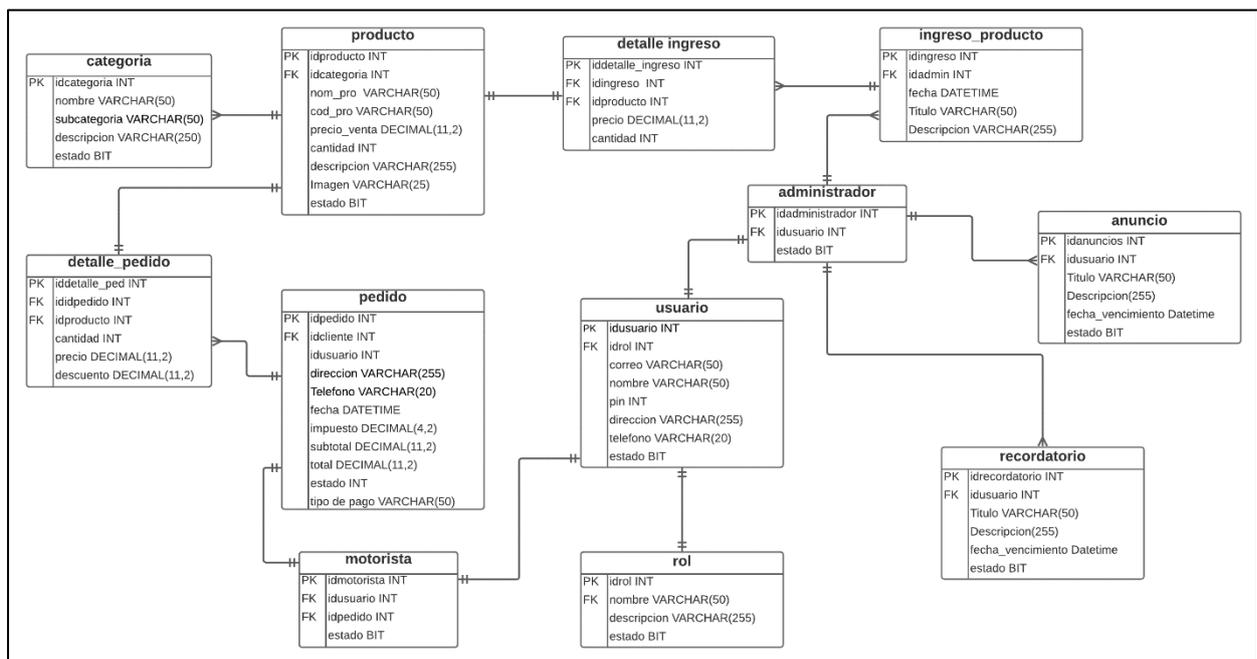


Ilustración 5 - Diseño de base de datos de la aplicación móvil y plataforma administrativa

Fuente: Elaboración propia

4.3.8 CASOS DE USO

El diagrama de casos de uso para el personal administrativo se muestra en la Ilustración 6, el administrador será capaz de realizar operaciones de creación, modificación, lectura y eliminación sobre usuarios, productos, anuncios y recordatorios, además podrán visualizar las ventas y generar un reporte de ventas.

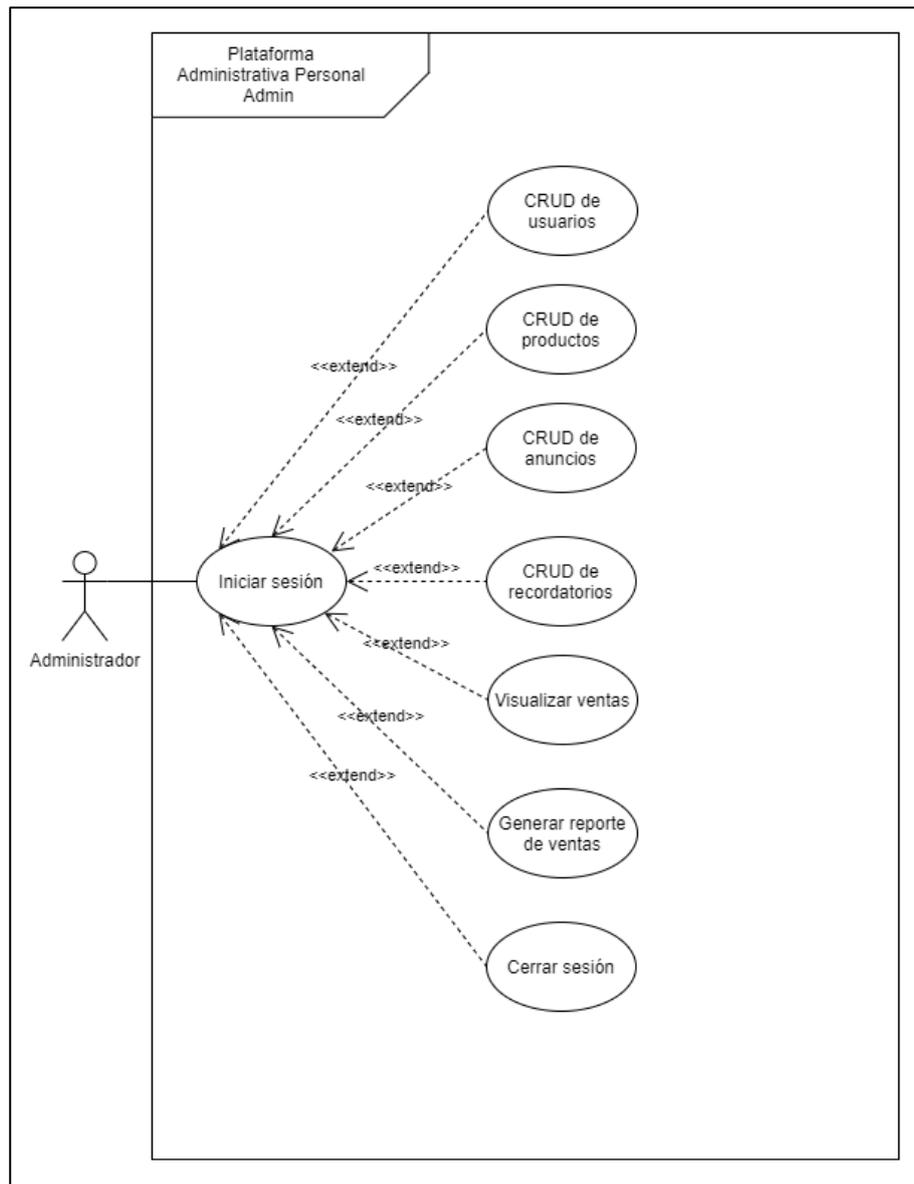


Ilustración 6 – Diagrama de casos de uso administrador en la plataforma administrativa

Fuente: Elaboración propia

Como se observa en la Ilustración 7, los clientes son capaces de utilizar diversas funcionalidades como realizar consultas al chatbot, registrarse, iniciar sesión, editar su información personal, ver el historial de pedidos realizados, ver pedidos activos, rastrear los pedidos activos, comunicarse con el repartidor, seleccionar productos a comprar, seleccionar recomendaciones y realizar pedidos.

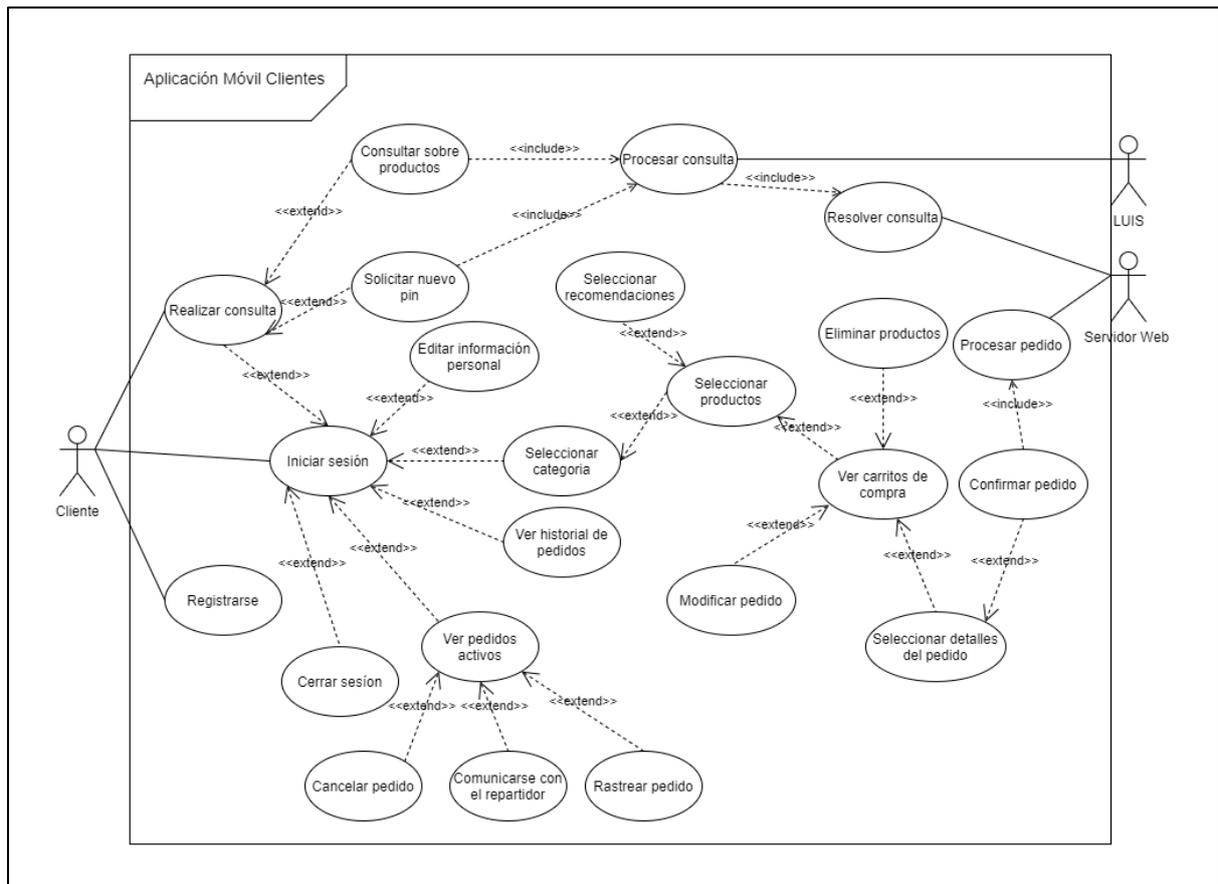


Ilustración 7 - Diagrama de casos de uso de clientes sobre aplicación móvil.

Fuente: Elaboración propia

En la Ilustración 8 se observarán las distintas funcionalidades que tendrán los motoristas o repartidores sobre la aplicación móvil, como iniciar sesión, modificar su información personal, ver el historial de pedidos entregados, visualizar los pedidos activos, tomar un pedido, modificar el estado del pedido y ver la dirección de entrega en el mapa.

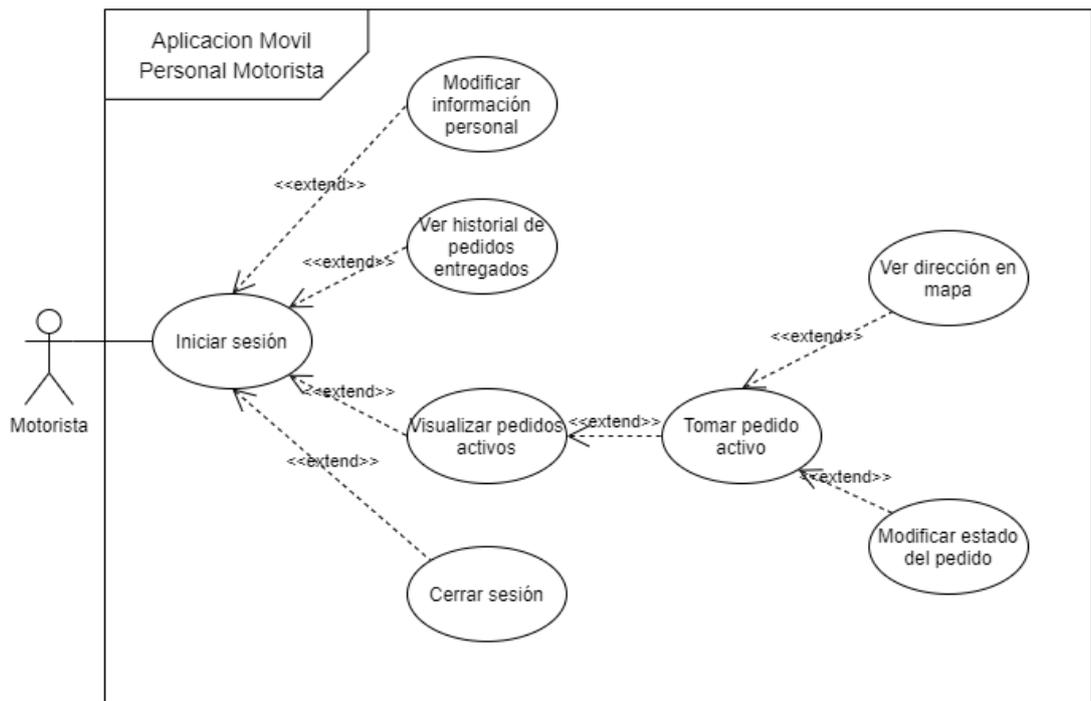


Ilustración 8 - Diagrama de casos de uso de motorista sobre la aplicación móvil.

Fuente: Elaboración propia

4.3.9 DISEÑO DE LA EXPERIENCIA DE USUARIO

Se decidió utilizar los colores naranja, verde y blanco, para el diseño de la experiencia de usuario de la aplicación, el naranja al ser llamativo proporciona alegría y energía, el verde es un color que brinda frescura, tranquilidad y armonía y el color blanco simboliza la paz y pureza. El uso de estos colores en conjunto proporciona un estilo alegre, limpio y elegante, para la aplicación móvil.

En la Ilustración 9 se observará el diseño de la pantalla de inicio de sesión.



Ilustración 9 - Pantalla inicio de sesión

Fuente: Elaboración propia

En la Ilustración 10 se observa el diseño de la pantalla de registro.



Ilustración 10 - Pantalla de registro

Fuente: Elaboración propia

En la Ilustración 11 se observa el diseño de la pantalla de chat.



Ilustración 11 - Pantalla chat

Fuente: Elaboración propia

En la Ilustración 12 se observa el diseño del menú principal.



Ilustración 12 - Pantalla menú principal

Fuente: Elaboración propia

En la Ilustración 13 se observa el diseño de la pantalla de información.



Ilustración 13 - Pantalla información

Fuente: Elaboración propia

En la Ilustración 14 se observa el diseño de la pantalla de editar información.



Ilustración 14 - Pantalla editar información.

Fuente: Elaboración propia

En la Ilustración 15 se observa el diseño de la pantalla de pedidos.



Ilustración 15 - Pantalla de pedidos.

Fuente: Elaboración propia

En la Ilustración 16 se muestra la pantalla de resumen de pedido.



Ilustración 16 - Pantalla resumen de pedido

Fuente: Elaboración propia

En la Ilustración 17 se muestra la pantalla de rastreo de pedido.

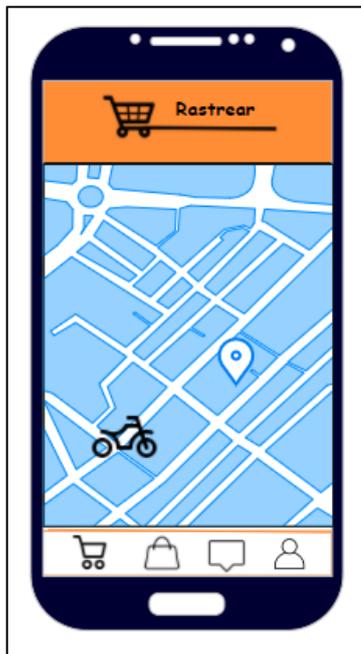


Ilustración 17 - Pantalla rastreo de pedido

Fuente: Elaboración propia

En la Ilustración 18 se muestra la pantalla de productos.



Ilustración 18 - Pantalla de productos

Fuente: Elaboración propia

En la Ilustración 19 se muestra la pantalla de compra.



Ilustración 19 - Pantalla de compra

Fuente: Elaboración propia

En la Ilustración 20 se muestra la pantalla del carrito de compras.



Ilustración 20 - Pantalla de carrito de compras

Fuente: Elaboración propia

4.3.10 ESCENARIOS DE PRUEBA

Escenarios de prueba pantalla inicio de sesión:

- Ingreso de caracteres especiales en la entrada de correo.
- Ingreso de caracteres especiales en la entrada de PIN.
- Ingreso de caracteres alfanuméricos en la entrada de PIN.
- Presionar el botón ingresar con las entradas vacías.
- Ingreso de entradas válidas y presionar el botón ingresar múltiples ocasiones.
- Ingreso de entradas válidas y presionar el botón de registrarse.
- Ingreso de entradas válidas, presionar el botón de inicio de sesión y luego el de registrarse.
- Ingreso de entrada de correo válida y PIN vacío.
- Ingreso de entrada de PIN válido y correo vacío.
- Ingreso de entradas inválidas en múltiples ocasiones.
- Ingreso de correo válido y PIN inválido en múltiples ocasiones.

Escenarios de prueba pantalla registrarse:

- Ingreso de caracteres especiales en la entrada de correo.
- Ingreso de caracteres especiales en la entrada de nombre.
- Ingreso de caracteres especiales en la entrada de dirección.
- Ingreso de caracteres alfanuméricos en la entrada de teléfono.
- Ingreso de caracteres especiales en la entrada de teléfono.
- Presionar el botón registrar con las entradas vacías.
- Presionar el botón registrar con el campo correo vacío.
- Presionar el botón registrar con el campo nombre vacío.
- Presionar el botón registrar con la dirección vacía.
- Presionar el botón registrar con el teléfono vacío.

- Ingreso de entradas válidas y presionar el botón registrar múltiples ocasiones.
- Presionar el botón registrar con correo inválido y demás entradas válidas.

Escenarios de prueba pantalla menú principal:

- Presionar un botón y presionar otro rápidamente.
- Subir y bajar el scroll de manera abrupta.
- Presionar dos botones al mismo tiempo.
- Intercambiar menús con la barra inferior rápidamente.

Escenarios de prueba pantalla información:

- Presionar los campos para ver si se pueden editar.
- Presionar iconos de los campos.
- Presionar el botón de editar múltiples ocasiones.
- Presionar el botón de editar de manera prolongada.

Escenarios de prueba pantalla de editar información:

- Ingreso de caracteres especiales en la entrada de nombre.
- Ingreso de caracteres especiales en la entrada de dirección.
- Ingreso de caracteres alfanuméricos en la entrada de teléfono.
- Ingreso de caracteres especiales en la entrada de teléfono.
- Presionar el botón confirmar con las entradas vacías.
- Presionar el botón confirmar con el campo nombre vacío.
- Presionar el botón confirmar con la dirección vacía.
- Presionar el botón confirmar con el teléfono vacío.
- Ingreso de entradas válidas y presionar el botón confirmar múltiples ocasiones.
- Presionar el botón confirmar con teléfono inválido y demás entradas válidas.

Escenarios de prueba pantalla de chat:

- Presionar botón enviar sin escribir mensaje.
- Ingresar y enviar caracteres especiales.

- Enviar mensajes sin contexto.

Escenarios de prueba pantalla de pedidos:

- Mover el scroll de arriba para abajo.
- Presionar botones de ver detalle en dos pedidos al mismo tiempo.
- Presionar botón ver detalle múltiples ocasiones.

Escenarios de prueba pantalla de resumen de pedido:

- Presionar botón múltiples ocasiones.
- Presionar botón ver dirección.
- Cambiar la dirección.

Escenarios de prueba pantalla de rastrear:

- Probar gesto de acercar.
- Probar gesto de alejar.
- Probar actualización de ubicación motorista.

Escenarios de prueba pantalla productos:

- Agregar productos al carrito.
- Eliminar productos del carrito.
- Agregar múltiples productos.
- Agregar producto varias veces.

Escenarios de prueba pantalla compra:

- Presionar botón pagar múltiples veces.
- Presionar botón cancelar compra.

Escenarios de prueba pantalla carrito:

- Agregar productos al carrito.
- Eliminar productos del carrito.
- Presionar botón de comprar múltiples veces.

4.4 METODOLOGÍA DE ESTUDIO

4.4.1 MODELO EN CASCADA

En este modelo las actividades fundamentales del proceso de desarrollo de software se llevan a cabo como fases separadas y consecutivas.

Según Gómez y Fuentes (2012) las 5 fases del modelo en cascada son:

- Análisis y definición de requerimientos. Se trabaja con los clientes y los usuarios finales del sistema para determinar el dominio de la aplicación y los servicios que debe proporcionar el sistema, así como sus restricciones.
- Diseño del sistema y del software. Durante el proceso de diseño del sistema se distinguen cuáles son los requerimientos de software y cuáles los de hardware. Después se establece una arquitectura completa del sistema. Durante el diseño de software se identifican los subsistemas que componen el sistema y se describe cómo funciona cada uno y las relaciones entre estos.
- Implementación y validación de unidades. Consiste en codificar y probar los diferentes subsistemas por separado.
- Integración y validación del sistema. Una vez que se probó individualmente cada una de las unidades, estas se integran para formar un sistema completo que debe cumplir con todos los requerimientos del software.
- Funcionamiento y mantenimiento. El sistema se instala y se pone en funcionamiento práctico. EL mantenimiento implica corregir errores no descubiertos en etapas anteriores del ciclo y mejorar la implantación de las unidades del sistema para darles mayor robustez.

4.5 CRONOGRAMA DE ACTIVIDADES

En la Ilustración 21 se muestra el cronograma de actividades para el desarrollo del proyecto.

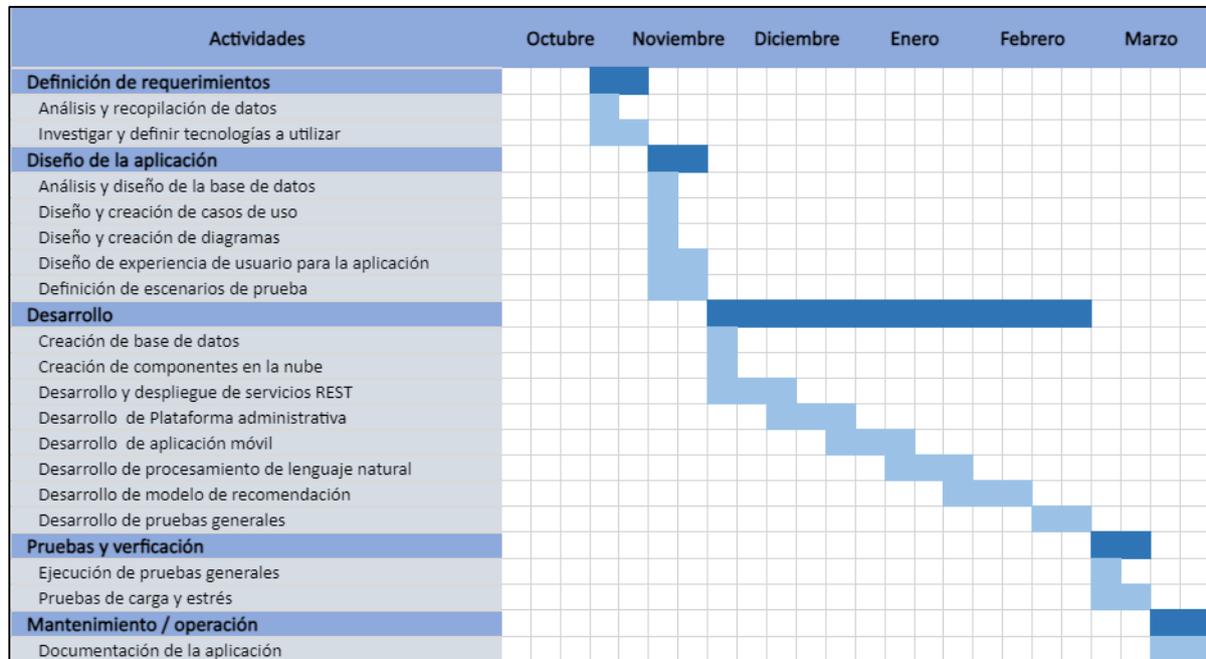


Ilustración 21 - Cronograma de actividades

Fuente: Elaboración propia

V. RESULTADOS Y ANÁLISIS

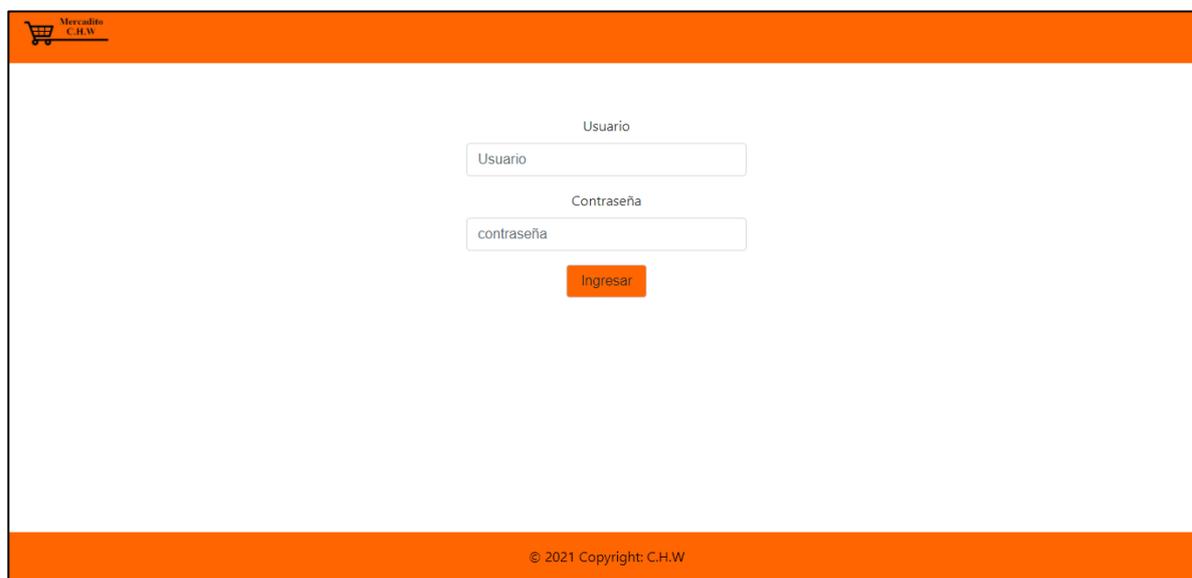
5.1 INTERFAZ DE PROGRAMACIÓN DE APLICACIONES

Esta API está diseñada para satisfacer las necesidades que podrían tener los usuarios de la aplicación móvil y de la plataforma administrativa. La API fue desarrollada en ASP.NET Core 3.1, utiliza métodos HTTP como POST, GET, PUT y DELETE para realizar distintas acciones. De igual manera se utilizan los parámetros dentro de la ruta y el cuerpo para enviar información. Véase la tabla de rutas del API en el Anexo 1.

5.2 CAPTURAS DE PANTALLA DE PLATAFORMA ADMINISTRATIVA

La plataforma administrativa facilita al administrador distintas tareas como crear, editar y eliminar distintos ítems que componen la aplicación móvil, de esta forma se gestionan de mejor manera los parámetros que contiene la misma.

En la Ilustración 22 se muestra la pantalla de inicio de sesión a la plataforma administrativa. El administrador de la aplicación puede ingresar sus credenciales y acceder al sistema.



Usuario

Usuario

Contraseña

contraseña

Ingresar

© 2021 Copyright: C.H.W.

Ilustración 22 - Pantalla inicio de sesión

Fuente: Elaboración propia

La plataforma administrativa permite gestionar la aplicación móvil para ello tiene distintas acciones una de ellas es la pantalla de gestión de usuarios.

En la Ilustración 23 se muestra la pantalla de gestión de usuarios desde ella se pueden realizar las operaciones de crear, editar y eliminar los usuarios.

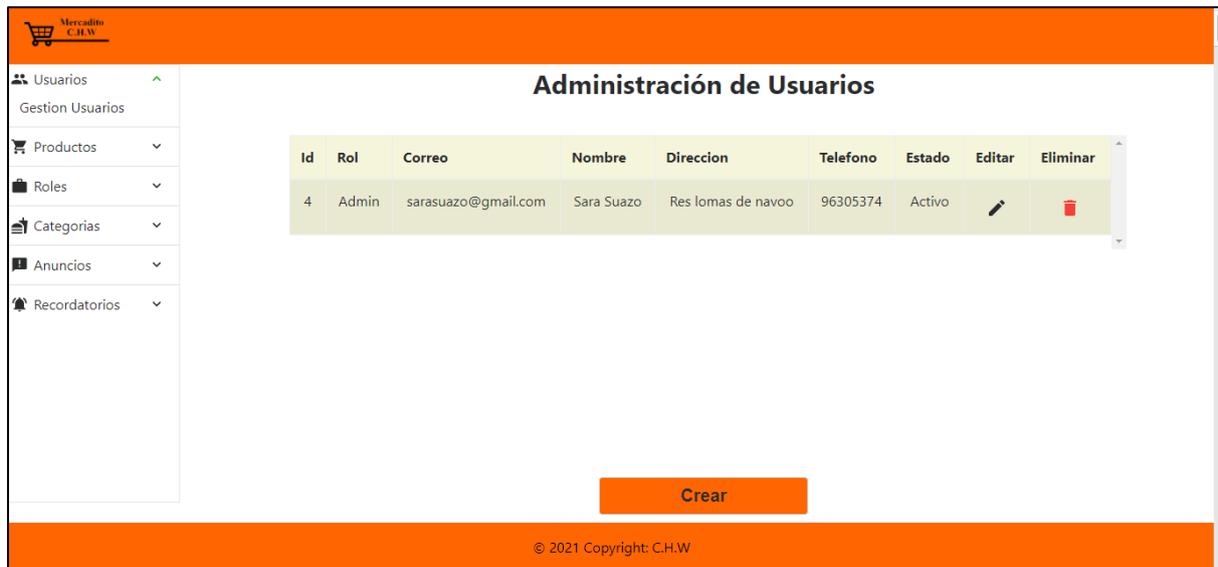


Ilustración 23 - Pantalla gestion de usuarios

Fuente: Elaboración propia

En la Ilustración 24 se muestra la pantalla de creación de usuario.



Ilustración 24 - Pantalla creación de usuario

Fuente: Elaboración propia

En la Ilustración 25 se muestra la pantalla de edición de usuario, como se puede observar se carga de manera automática la información del usuario que se desea editar.

© 2021 Copyright: C.H.W.

Ilustración 25 - Pantalla edición de usuario

Fuente: Elaboración propia

En la Ilustración 26 se muestra la alerta que se despliega al querer eliminar un usuario, si esta se confirma el usuario es eliminado.

© 2021 Copyright: C.H.W.

Ilustración 26 - Pantalla eliminación de usuario

Fuente: Elaboración propia

En la Ilustración 27 se muestra la pantalla de gestión de productos.



Ilustración 27 - Pantalla gestión de productos

Fuente: Elaboración propia

En la Ilustración 28 se muestra la pantalla de creación de producto.

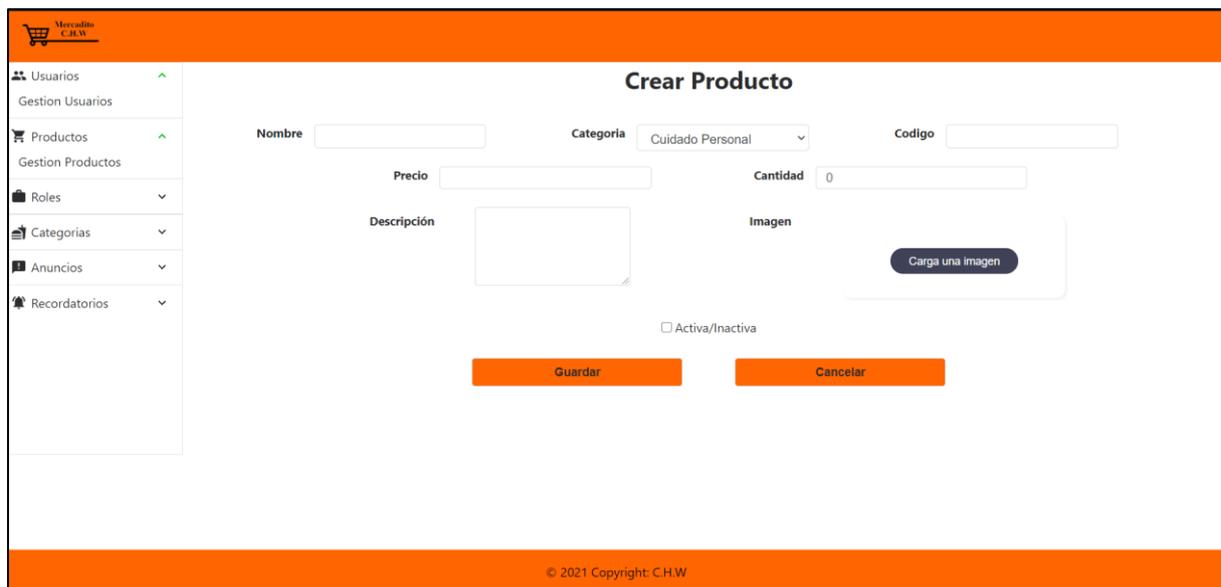


Ilustración 28 - Pantalla creación de producto

Fuente: Elaboración propia

En la Ilustración 29 se muestra la pantalla de edición de producto.

Ilustración 29 - Pantalla edición de producto

Fuente: Elaboración propia

En la Ilustración 30 se muestra la pantalla de gestión de roles.

Id	Nombre	Descripción	Estado	Editar	Eliminar
3	Admin	superusuario	Activo		
4	Motorista	hace mandados	Activo		

Ilustración 30 - Pantalla gestión de roles

Fuente: Elaboración propia

En la Ilustración 31 se muestra la pantalla de creación de rol.

Mercedito
C.H.W.

Usuarios
Gestion Usuarios

Productos
Gestion Productos

Roles
Gestion Roles

Categorías

Anuncios

Recordatorios

Crear Rol

Nombre

Nombre

Descripción

Activa/Inactiva

Crear Cancelar

© 2021 Copyright: C.H.W.

Ilustración 31 - Pantalla creación de rol

Fuente: Elaboración propia

En la Ilustración 32 se muestra la pantalla de edición de rol.

Mercedito
C.H.W.

Usuarios
Gestion Usuarios

Productos
Gestion Productos

Roles
Gestion Roles

Categorías

Anuncios

Recordatorios

Editar Rol

Nombre

Admin

Descripción

superusuario

Activa/Inactiva

Guardar Cancelar

© 2021 Copyright: C.H.W.

Ilustración 32 - Pantalla edición de rol

Fuente: Elaboración propia

En la Ilustración 33 se muestra la pantalla de gestión de categorías.



Ilustración 33 - Pantalla gestión de categorías

Fuente: Elaboración propia

En la Ilustración 34 se muestra la pantalla de creación de categoría.

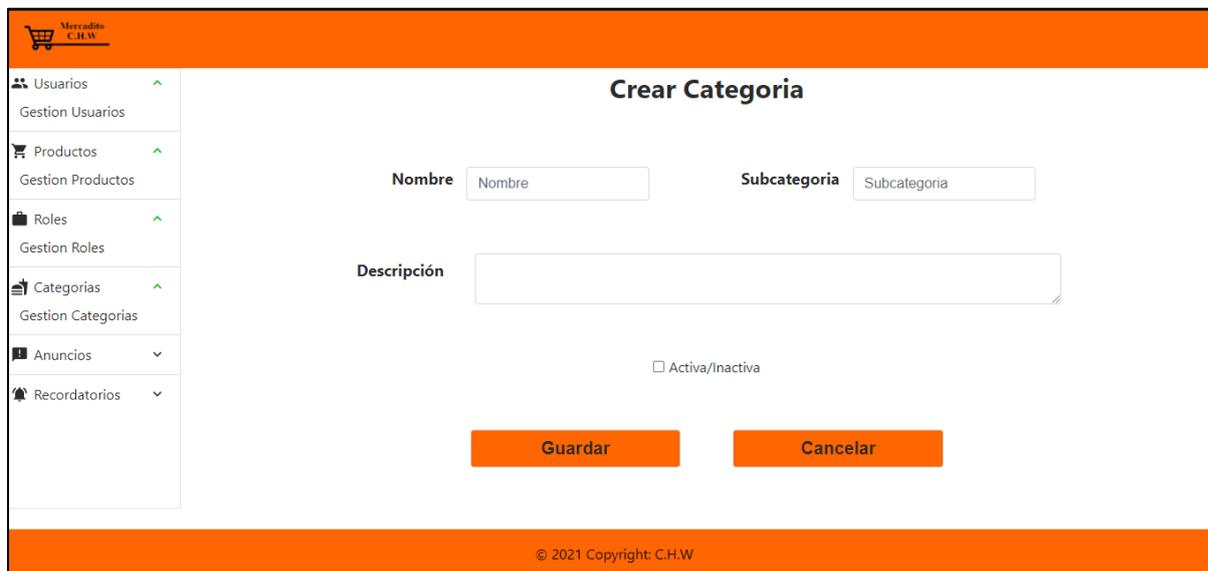


Ilustración 34 - Pantalla creación de categoría

Fuente: Elaboración propia

En la Ilustración 35 se muestra la pantalla de edición de categoría.

© 2021 Copyright: C.H.W.

Ilustración 35 - Pantalla edición de categoría

Fuente: Elaboración propia

En la Ilustración 36 se muestra la pantalla de gestión de anuncios.

Id	Usuario	Título	Descripción	Fecha Vencimiento	Estado	Editar	Eliminar
17	0	Pago cervecería	Pago 50 mil lempiras	2021-01-20T14:10:00	Activo		

© 2021 Copyright: C.H.W.

Ilustración 36 - Pantalla gestión de anuncios

Fuente: Elaboración propia

En la Ilustración 37 se muestra la pantalla de creación de anuncio.

Mercedito C.H.W.

Usuarios ^
Gestion Usuarios

Productos ^
Gestion Productos

Roles ^
Gestion Roles

Categorías ^
Gestion Categorías

Anuncios ^
Gestion Anuncios

Recordatorios ^

Crear Anuncios

Titulo

Fecha Vencimiento
mm/dd/yyyy --:-- --

Descripción

Activa/Inactiva

Guardar Cancelar

© 2021 Copyright: C.H.W.

Ilustración 37 - Pantalla creación de anuncio

Fuente: Elaboración propia

En la Ilustración 38 se muestra la pantalla de edición de anuncio.

Mercedito C.H.W.

Usuarios ^
Gestion Usuarios

Productos ^
Gestion Productos

Roles ^
Gestion Roles

Categorías ^
Gestion Categorías

Anuncios ^
Gestion Anuncios

Recordatorios ^

Editar anuncios

Titulo
Pago cervecería

Fecha Vencimiento
01/20/2021 02:10 PM

Descripción
Pago 50 mil lempiras

Activa/Inactiva

Guardar Cancelar

© 2021 Copyright: C.H.W.

Ilustración 38 - Pantalla edición de anuncio

Fuente: Elaboración propia

En la Ilustración 39 se muestra la pantalla de gestión de recordatorios.

Id	Usuario	Título	Descripción	Fecha Vencimiento	Estado	Editar	Eliminar
13	sarasuazo@gmail.com	Pago cervecería	Pago de 40 mil lempiras	2021-01-13T09:29:00	Activo		

Ilustración 39 - Pantalla gestión de recordatorios

Fuente: Elaboración propia

En la Ilustración 40 se muestra la pantalla de creación de recordatorio.

Titulo Usuario

Fecha Vencimiento

Descripción

Activa/Inactiva

Ilustración 40 - Pantalla creación de recordatorio

Fuente: Elaboración propia

En la Ilustración 41 se muestra la pantalla de edición de recordatorio.

Ilustración 41 - Pantalla edición de recordatorio

Fuente: Elaboración propia

5.3 RECOMENDACIÓN DE PRODUCTOS

En este proyecto se implementó un modelo de recomendación de productos, el cual permite a los usuarios ver un apartado de productos sugeridos dentro de la aplicación móvil, estos se muestran en base a la lista de productos que están agregados al carrito de compras. Para la implementación de este modelo se utilizó la herramienta de Microsoft Azure Machine Learning Studio.

Para generar el modelo de recomendación de productos, los pasos a seguir fueron:

1. Recolectar los datos para entrenar el modelo.
2. Preprocesar los datos que se obtuvieron.
3. Llenar la base de datos con los datos procesados.
4. Importar los datos para la creación del modelo de recomendación de productos.
5. Editar la metadata para que el modelo funcione de manera correcta.
6. Seleccionar las columnas con los datos que entrenará al modelo.
7. Entrenar el modelo de recomendación de productos.
8. Generar una entrada de datos para probar el modelo que se entrenó previamente.

9. Probar el modelo de recomendación de productos con los datos de entrada.
10. Generar una web API que proporcione los datos de entrada.
11. Generar una web API que retorne los resultados de los productos recomendados en base a la entrada de datos que se proporcionó.

En la Ilustración 42 se muestra el modelo de recomendación de productos.

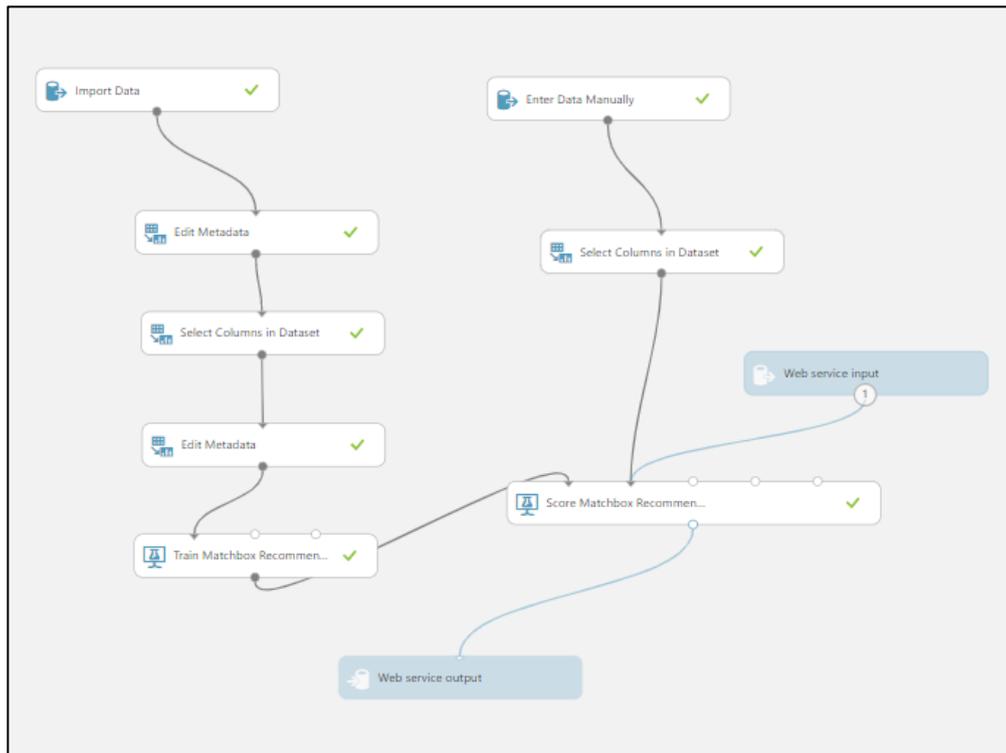


Ilustración 42 - Modelo de recomendación de productos

Fuente: Elaboración propia

5.4 CAPTURAS DE PANTALLA APLICACIÓN MÓVIL

Este proyecto consta de una aplicación móvil, esta será usada por los clientes de dicho local, tiene la capacidad de realizar múltiples funciones como creación de perfil, inicio de sesión, compras, rastreo de envíos, editar información personal, entre otras.

En la Ilustración 43 se muestra el icono de la aplicación móvil.



Ilustración 43 - Icono aplicación móvil

Fuente: Elaboración propia

En la Ilustración 44 se muestra la pantalla de inicio de sesión.



Ilustración 44 - Pantalla inicio de sesión

Fuente: Elaboración propia

En la Ilustración 45 se muestra la pantalla de registro, la dirección se obtiene mediante GPS al presionar el botón obtener, una vez lleno el formulario se presiona el botón registrarse y la aplicación envía un correo con un código de verificación, este se utiliza para activar el usuario.

The screenshot shows a mobile application interface for registration. At the top, there is an orange header with a back arrow, the text 'Inicio Sesión', and the application logo 'Mercadito C.H.W.' featuring a shopping cart icon. Below the header, the registration form consists of several fields: 'Correo' (Email) with the value 'jeanwahlung@gmail.com', 'Nombre' (Name) with 'Jean Carlos Wah Lung', 'Dirección' (Address) with 'Colonia Miraflores, Tegucigalpa, Honduras' and an orange 'Obtener' button, and 'Teléfono' (Phone) with '31768607'. A large green 'Registrarse' button is positioned at the bottom of the form. The bottom of the screen shows the standard Android navigation bar.

Ilustración 45 - Pantalla de registro

Fuente: Elaboración propia

En la Ilustración 46 se muestra la pantalla de confirmación de registro, una vez lleno el formulario el usuario asociado al correo ingresado se activa.

The screenshot shows the confirmation screen of the mobile application. The header is orange and contains a back arrow, the text 'Registro', and the application logo 'PIN' with a shopping cart icon. The form includes: 'Correo' (Email) with 'jeanwahlung@gmail.com', 'Codigo Verificación' (Verification Code) with an empty input field, and 'PIN' with an empty input field. A large green 'Confirmar' button is located at the bottom of the form. The bottom of the screen shows the standard Android navigation bar.

Ilustración 46 - Pantalla confirmación de registro

Fuente: Elaboración propia

En la Ilustración 47 se muestra la pantalla de información personal, en esta pantalla se observa la información del usuario.



Ilustración 47 - Pantalla información personal

Fuente: Elaboración propia

En la Ilustración 48 se muestra la pantalla edición información personal, en esta pantalla se puede actualizar la información del usuario y su dirección.



Ilustración 48 - Pantalla edición información de usuario

Fuente: Elaboración propia

En la Ilustración 49 se muestra la pantalla de pedidos, los pedidos con el botón de color verde están activos y los demás están entregados.



Ilustración 49 - Pantalla de pedidos

Fuente: Elaboración propia

En la Ilustración 50 se observa la pantalla de pedido entregado, en esta se muestra la respectiva información del pedido.

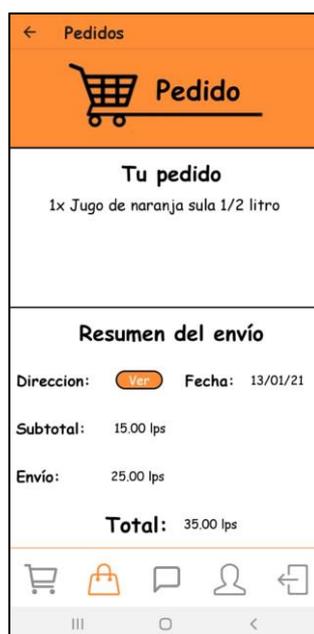


Ilustración 50 - Pantalla pedido entregado.

Fuente: Elaboración propia

En la Ilustración 51 se muestra la pantalla de pedido activo, esta muestra la información del pedido activo además permite cancelar o rastrear el pedido.



Ilustración 51 - Pantalla de pedido activo

Fuente: Elaboración propia

En la Ilustración 52 se muestra la pantalla de rastreo de pedido.



Ilustración 52 - Pantalla de rastreo de pedido

Fuente: Elaboración propia

En la Ilustración 53 se muestra la pantalla de categorías, en esta se encuentran todas las categorías existentes.



Ilustración 53 - Pantalla de categorías

Fuente: Elaboración propia

En la Ilustración 54 se muestra la pantalla de productos, en esta se encuentran todos los productos de la categoría seleccionada.



Ilustración 54 - Pantalla de productos

Fuente: Elaboración propia

En la Ilustración 55 se muestra la pantalla del carrito de compras, en esta se pueden agregar productos o eliminarlos del carrito de compras.



Ilustración 55 - Pantalla carrito de compras.

Fuente: Elaboración propia

En la Ilustración 56 se muestra la pantalla de compra, en esta se procede a la confirmación de la compra.

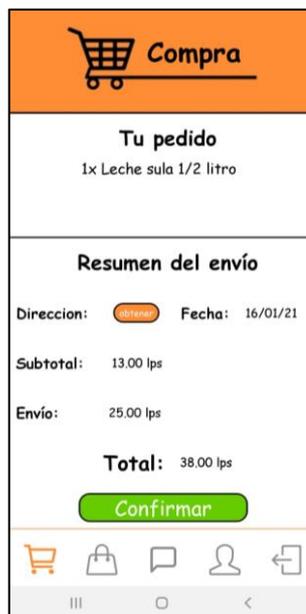


Ilustración 56 - Pantalla de compra

Fuente: Elaboración propia

En la Ilustración 57 se muestra la pantalla del chatbot, en esta se pueden realizar varias funciones como consultar sobre algún producto, realizar un reinicio de contraseña, ver ubicación o número de teléfono del local y calificar al chatbot.

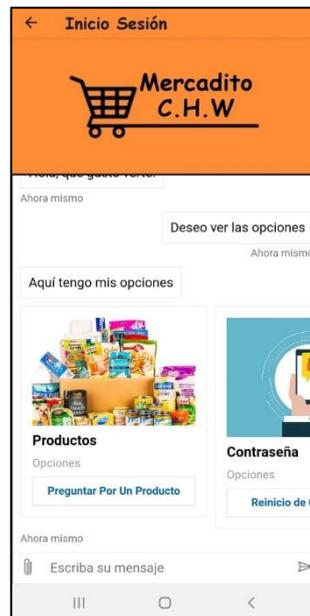


Ilustración 57 - Pantalla del chatbot

Fuente: Elaboración propia

En la Ilustración 58 se muestra el ejemplo de la función consultar sobre un producto.

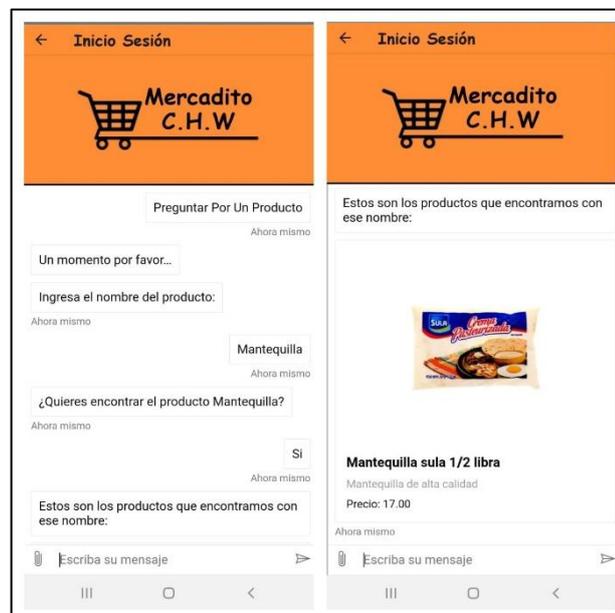


Ilustración 58 - Ejemplo consulta sobre un producto

Fuente: Elaboración propia

En la Ilustración 59 se muestra el ejemplo de la función reinicio de contraseña.

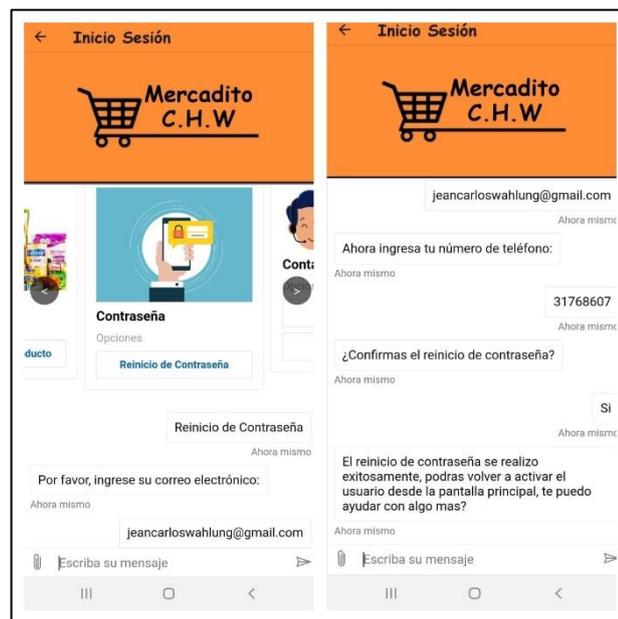


Ilustración 59 - Ejemplo reinicio de contraseña

Fuente: Elaboración propia

En la Ilustración 60 se muestra el ejemplo de la función ver información sobre el local.

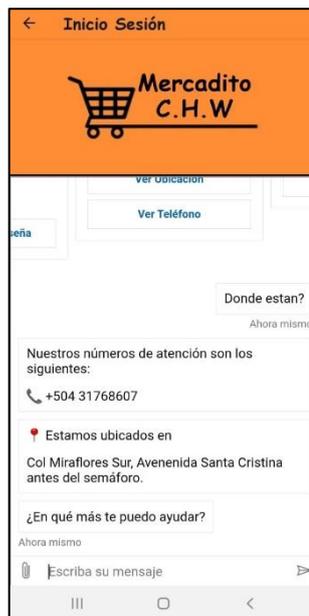


Ilustración 60 - Ejemplo ver información

Fuente: Elaboración propia

En la Ilustración 61 se muestra el ejemplo de la función calificar chatbot.

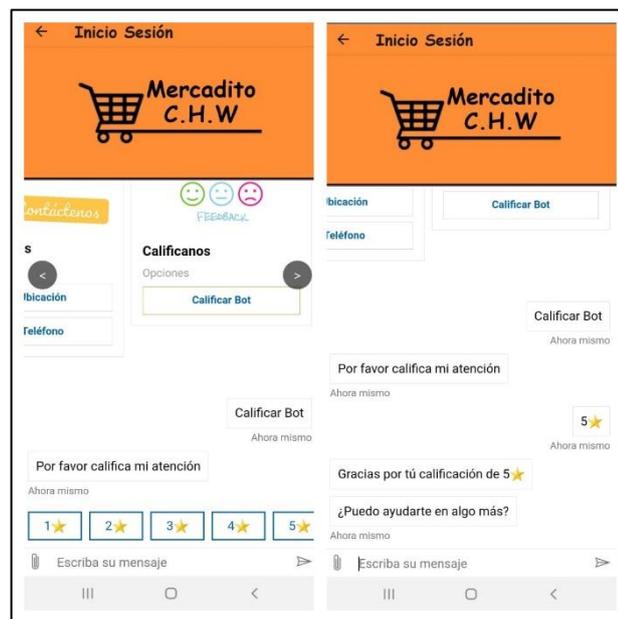


Ilustración 61 - Ejemplo calificar chatbot

Fuente: Elaboración propia

5.5 PRUEBAS GENERALES

Para garantizar la calidad del producto final se realizaron distintas pruebas entre ellas los escenarios planteados en la sección 4.3.10, pruebas funcionales y pruebas de carga y estrés.

Para el desarrollo de estas pruebas se usaron dos herramientas de vital importancia Appium y Apache JMeter, las pruebas Appium en conjunto con la plataforma de App Center que proporciona una granja de dispositivos reales permite observar el comportamiento de la aplicación móvil corriendo sobre distintos modelos ya existentes en el mercado.

Los casos de pruebas funcionales que se realizaron para la app fueron los siguientes:

- Registro de usuario.
- Confirmación de registro de usuario.
- Inicio de sesión.
- Creación de un pedido.
- Visualización de pedidos activos.

- Visualización de pedidos inactivos.
- Rastreo de pedidos.
- Comunicación con el chatbot.
- Edición de información de usuario.
- Comunicación con chatbot.
- Agregar y visualizar productos recomendados.

A continuación, un muestreo seleccionado de manera arbitraria de pruebas que se ejecutaron sobre la aplicación móvil.

En la Ilustración 62 se muestra la prueba de comunicación con el chatbot al cual se le solicita mostrar las opciones disponibles, la cual en este caso se ejecutó de manera exitosa en distintos dispositivos.



Ilustración 62 – Prueba de comunicación con el chatbot

Fuente: Elaboración propia

En la Ilustración 63 se muestra la prueba de creación de pedido la cual en este caso se ejecutó de manera exitosa en distintos dispositivos.



Ilustración 63 – Prueba de creación de pedido

Fuente: Elaboración propia

En la Ilustración 64 se muestra la prueba de agregar y visualizar los productos recomendados la cual en este caso se ejecutó de manera exitosa en distintos dispositivos.



Ilustración 64 – Prueba de agregar y visualizar productos recomendados

Fuente: Elaboración propia

En la Ilustración 65 se muestra la prueba visualización de pedidos activos e inactivos la cual en este caso se ejecutó de manera exitosa en distintos dispositivos.



Ilustración 65 – Prueba de visualización de pedidos activos e inactivos

Fuente: Elaboración propia

En la Ilustración 66 se muestra la prueba de rastreo de pedido la cual en este caso se ejecutó de manera exitosa en distintos dispositivos.

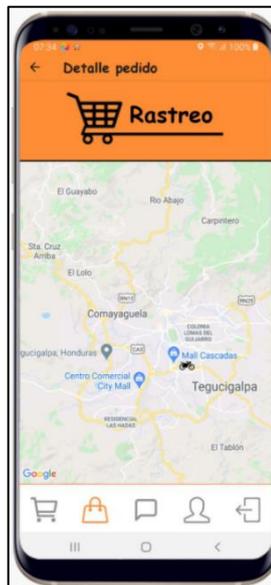


Ilustración 66 – Prueba de rastreo de pedido

Fuente: Elaboración propia

Para poder medir la capacidad que tiene el servidor web al momento de soportar cargas, se hace uso de la herramienta Apache JMeter.

Se decidió realizar pruebas con carga moderada, haciendo 6 tipos de consultas distintas al servidor, estas se seleccionaron arbitrariamente:

- Inicio de sesión.
- Editar información de usuario.
- Crear pedido.
- Obtener categorías.
- Obtener productos por categoría.
- Obtener pedidos.

Primero se realizó una prueba con carga teniendo 20 hilos de procesamiento, cada uno realizando 5 consultas. En total se realizaron 600 consultas al servidor. En la Ilustración 67 se puede observar un gráfico de los resultados obtenidos, donde las mediciones se realizan en milisegundos.

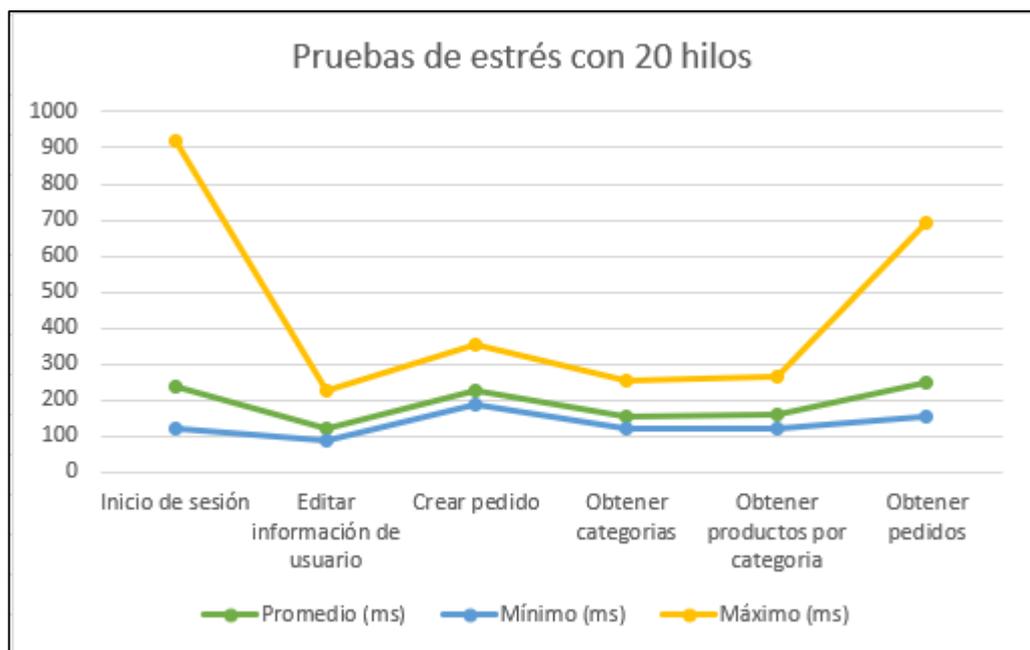


Ilustración 67 - Pruebas de estrés con 20 hilos

Fuente: Elaboración propia

Luego, se volvió a correr la misma prueba, solo que esta vez con 100 hilos de procesamiento. En total se realizaron 3000 consultas al servidor. En la Ilustración 68 se puede observar un gráfico de los resultados obtenidos, donde las mediciones se realizan en milisegundos.

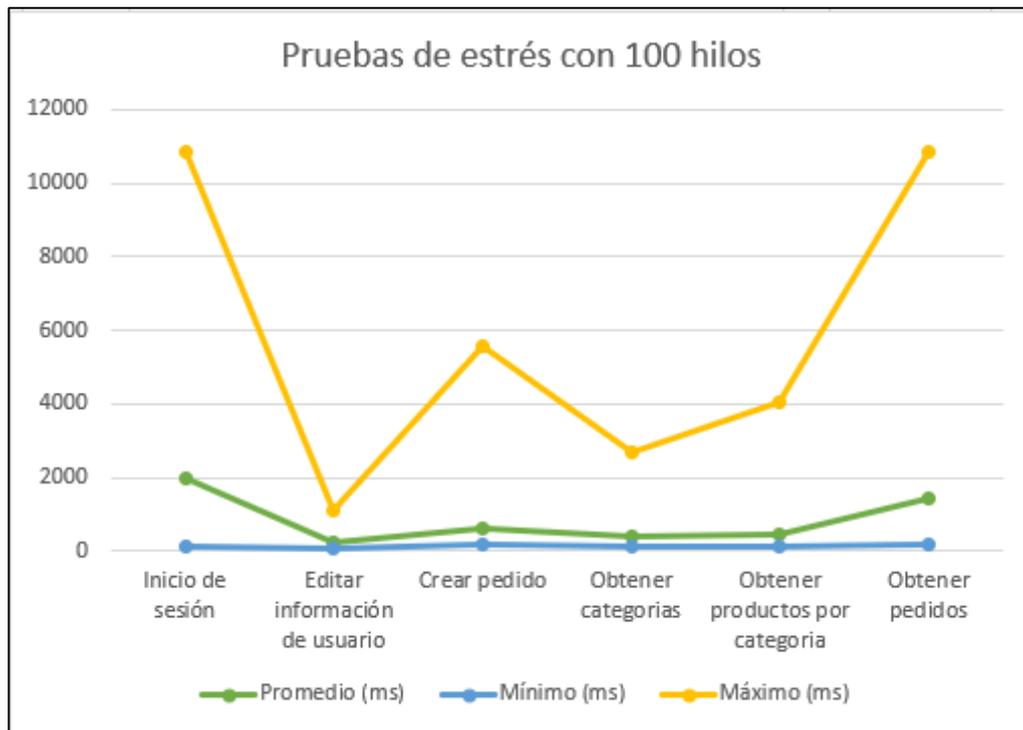


Ilustración 68 - Pruebas de estrés con 100 hilos

Fuente: Elaboración propia

Como se puede observar en el gráfico, los tiempos de ejecución del inicio de sesión se elevan llegando a tardar aproximadamente 11 segundos.

Aunque en un ambiente de producción es poco probable que 100 usuarios intenten acceder a los mismos recursos al mismo tiempo de manera continua, aun así, el sistema puede lidiar moderadamente con una carga de tan alta magnitud.

Cabe destacar que en ambas pruebas realizadas el 100% de las solicitudes al servidor fueron procesadas de manera exitosa.

VI. CONCLUSIONES

Luego de finalizar con el desarrollo de este proyecto, se ha podido llegar a las siguientes conclusiones:

- Se logró entrenar e implementar un modelo de aprendizaje automático utilizando los servicios que proporciona el portal de Microsoft Azure, debido a ello la aplicación móvil es capaz de recomendar ciertos productos en base a los que están agregados en el carrito de compras.
- Se logró implementar un modelo de procesamiento de lenguaje natural utilizando el servicio de LUIS que proporciona el portal Microsoft Azure, debido a ello el chatbot es capaz de reconocer distintas frases utilizadas por los humanos y proporcionar al usuario sus distintas funciones.
- Se desarrolló una plataforma administrativa que facilita la gestión de productos, categorías, usuarios y publicidad, sobre la aplicación móvil.
- Se desarrolló una aplicación móvil que hace uso de la inteligencia artificial y facilita a los usuarios el proceso de compra, precompra y postcompra, mediante el modelo de recomendación de productos y asistencia por parte del chatbot, lo cual permite a los usuarios tener una mejor experiencia de compra haciendo uso de la aplicación móvil.
- Se desarrollaron y ejecutaron distintos tipos de pruebas, entre ellas pruebas unitarias, de integración, de estrés y de aceptación del usuario, de esta forma se garantizó la calidad del software.

VII. RECOMENDACIONES

A continuación, se presentan algunas recomendaciones que deberán de ser consideradas para los encargados de dar mantenimiento a la aplicación móvil y plataforma administrativa que se desarrollaron en este informe.

- Se recomienda a los encargados de la aplicación móvil estar al pendiente y tomar en cuenta las recomendaciones y sugerencias que los usuarios puedan tener sobre la aplicación móvil para poder mejorar la experiencia del usuario. Los usuarios son quienes finalmente utilizan la aplicación móvil y pueden tener muchas ideas o sugerencias sobre cosas que podrían facilitar o mejorar el uso de dicha aplicación. A través de un proceso de mejora continuo se logra mantener los usuarios ya existentes y atraer nuevos usuarios.
- Se recomienda a los encargados de la aplicación móvil seguir alimentando el vocabulario del chatbot, de esta forma será capaz de comprender las distintas frases que puedan utilizar los usuarios en la actualidad, así se logrará una comunicación más fluida entre usuario y chatbot.
- Se recomienda a los encargados de la plataforma administrativa estar al pendiente y tomar en cuenta las observaciones y nuevas funciones que desee implementar el personal administrativo sobre dicha plataforma, ya que esto le facilitará el proceso de gestión de la aplicación móvil al personal.
- Se recomienda a los encargados de la plataforma administrativa implementar gráficas dentro de la plataforma, de esta forma el personal administrativo podrá observar de manera visual los movimientos que ocurren en la aplicación móvil por parte de los usuarios y logrará tomar mejores decisiones para impulsar el uso de la misma.

BIBLIOGRAFÍA

1. Aguilar, L. J. (2012). Computación en la nube: Notas para una estrategia española en cloud computing. *Revista del Instituto Español de Estudios Estratégicos*, (00).
2. Álvarez Vega, M., Quirós Mora, L., & Cortés Badilla, M. (01 de Agosto de 2020). Inteligencia artificial y aprendizaje automático en medicina. *Revista Médica Sinergia*, 5(8), e557-e557. Recuperado el 15 de octubre de 2020, de Inteligencia artificial y aprendizaje automático en medicina.: <https://revistamedicasinergia.com/index.php/rms/article/view/557>
3. Appium. (2020). Introduction - Appium. Recuperado el 7 de noviembre de 2020, de Introduction - Appium: <http://appium.io/docs/en/about-appium/intro/?lang=es>
4. Barrantes, R. (2014). Investigación: Un camino al conocimiento, Un enfoque Cualitativo, cuantitativo y mixto. San José, Costa Rica: EUNED.
5. Wagner, B. (2020). Get Started - an introduction to the C# language and .NET'. Recuperado el 9 de noviembre de 2020, de Microsoft Docs: <https://docs.microsoft.com/en-us/dotnet/csharp/getting-started/>
6. Randell, B. (2020). Information bot scenario - Bot Service. Recuperado el 11 de noviembre de 2020, de Microsoft Docs: <https://docs.microsoft.com/en-us/previous-versions/azure/bot-service/bot-service-scenario-informational>
7. Calixto, R. N. L., González, L. Á. V., Díaz, D. E. B., & Guzmán, R. V. (2019). React Native: acortando las distancias entre desarrollo y diseño móvil multiplataforma. *Revista Digital Universitaria*, 20(5).
8. Cortez Vásquez, A., Vega Huerta, H., Pariona Quispe, J., & Huayna, A. M. (2009). Procesamiento de lenguaje natural. *Revista de investigación de Sistemas e Informática*, 6(2), 45-54. Recuperado el 16 de agosto de 2020, de <https://revistasinvestigacion.unmsm.edu.pe/index.php/sistem/article/view/592>
9. Garcia Brustenga, G., Fuertes Alpiste, M., & Molas Castells, N. (2018). Briefing paper: los chatbots en educación.
10. García Reina, L. F. (2018). Asistente virtual de tipo ChatBot.

11. Gómez, J. C., & Fuentes, M. D. C. G. (2012). Taxonomía de los modelos y metodologías de desarrollo de software más utilizados. *Universidades*, 62(52), 37-47.
12. Hans, M. (2015). *Appium essentials*. Packt Publishing Ltd.
13. Hernandez, N. L., & Florez Fuentes, A. S. (15 de Diciembre de 2014). COMPUTACIÓN EN LA NUBE. *Mundo FESC*, 4(8), 46-51. Recuperado el 16 de agosto de 2020, de <https://www.fesc.edu.co/Revistas/OJS/index.php/mundofesc/article/view/48>
14. Kaiqb. (2020). What is the Bot Framework SDK? - Bot Service. Recuperado el 11 de noviembre de 2020, de Microsoft Docs: <https://docs.microsoft.com/en-us/azure/bot-service/bot-service-overview-introduction>
15. *Marketing predictivo con Machine Learning | Microsoft Azure*. (2020). Recuperado el 12 de noviembre de 2020, de Microsoft Azure: <https://azure.microsoft.com/es-es/solutions/architecture/predictive-marketing-campaigns-with-machine-learning-and-spark/>
16. Mejia, O. (2011). Computación en la nube. *ContactoS*, 80, 45-52.
17. Meza, E. R., & Imbachi, D. R. V (2016). Las ventajas de la utilización de Microsoft Azure en proyectos Open Source. *Algunas estrategias para mejorar la competitividad*, 6.
18. Ramos Pérez, L. G (2019). Computación en la nube—Análisis de proveedor Microsoft Azure para migración de aplicaciones y servicios, caso de estudio, Unidad para las Víctimas (UARIV). *Ingeniería de Sistemas*.
19. Romero, J. J., Dafonte, C. A. R. L. O. S., Gómez, Á. N. G. E. L., & Penousal, F. J. (2007). *Inteligencia artificial y computación avanzada*. Santiago de Compostela: Fundación Alfredo Brañas, 10-15.
20. T. (2020). *What is Azure Cloud Services*. Recuperado el 20 de agosto de 2020, de What is Azure Cloud Services: <https://docs.microsoft.com/en-us/azure/cloud-services/cloud-services-choose-me>
21. Takeyas, B. L. (2007). *Introducción a la inteligencia artificial*. Instituto Tecnológico de Nuevo Laredo.
22. *IDE de Visual Studio 2019: software de programación para Windows*. (2020). Recuperado el 13 de noviembre de 2020, de <https://visualstudio.microsoft.com/es/vs/>

23. Williams, J. D., Kamal, E., Ashour, M., Amr, H., Miller, J., & Zweig, G. (2015, Septiembre). Fast and easy language understanding for dialog systems with Microsoft Language Understanding Intelligent Service (LUIS). In Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue (pp. 159-161).
24. Weeks, C. (2018). Chatbots are ruining the customer experience. Here's how to fix it. Enterprise Innovation.
25. Soldado Zuña, A. F. (2018). Análisis, diseño e implementación de un sistema con notificaciones push para plataforma Android, Bachelor's thesis, Quito: UCE.

ANEXOS

Anexo 1 - Tabla de rutas del API

En la Tabla1 se muestra un resumen de todas las rutas que se encuentran en el sistema.

Tabla 1 - Rutas del API

Campo	Descripción
Ruta	/api/roles/{id}
Función	Obtener lista de roles existentes.
Tipo	GET
Permisos	Administrador
Ruta	/api/roles
Función	Obtener rol específico usando el identificador único.
Tipo	GET
Permisos	Administrador
Ruta	/api/roles
Función	Crear un nuevo rol de usuario.
Tipo	POST
Permisos	Administrador
Cuerpo	Nombre (nombre): cadena Descripción (descripción): cadena Estado (estado): booleano

Campo	Descripción
Ruta	/api/roles/{id}
Función	Editar los campos de un rol mediante su identificador único.
Tipo	PUT
Permisos	Administrador
Cuerpo	Nombre (nombre): cadena Descripción (descripcion): cadena Estado (estado): booleano
Ruta	/api/roles/{id}
Función	Eliminar un rol mediante su identificador único.
Tipo	DELETE
Permisos	Administrador
Ruta	/api/categorias
Función	Obtener lista de categorías existentes.
Tipo	GET
Permisos	Administrador o cliente
Ruta	/api/categorias/{id}
Función	Obtener una categoría mediante su identificador único.
Tipo	GET
Permisos	Administrador

Campo	Descripción
Ruta	/api/categorias
Función	Crear una nueva categoría.
Tipo	POST
Permisos	Administrador
Cuerpo	Nombre (nombre): cadena Subcategoría (subcategoria): cadena Descripción (descripción): cadena Estado (estado): booleano
Ruta	/api/categorias/{id}
Función	Editar una categoría mediante su identificador único.
Tipo	PUT
Permisos	Administrador
Cuerpo	Nombre (nombre): cadena Subcategoría (subcategoria): cadena Descripción (descripción): cadena Estado (estado): booleano
Ruta	/api/categorias/{id}
Función	Eliminar una categoría mediante su identificador único.
Tipo	DELETE
Permisos	Administrador

Campo	Descripción
Ruta	/api/productos
Función	Obtener una lista de productos existentes.
Tipo	GET
Permisos	Administrador o cliente
Ruta	/api/productos/{id}
Función	Obtener producto mediante su identificador único.
Tipo	GET
Permisos	Administrador
Ruta	/api/productos
Función	Crear un nuevo producto.
Tipo	POST
Permisos	Administrador
Cuerpo	<p>Identidad Categoría (idcategoria): entero</p> <p>Nombre (nom_pro): cadena</p> <p>Código (cod_pro): cadena</p> <p>Precio (precio_venta): real</p> <p>Cantidad (cantidad): entero</p> <p>Descripción (descripcion): cadena</p> <p>Imagen (img): Imagen</p> <p>Estado (estado): booleano</p>

Campo	Descripción
Ruta	/api/productos/{id}
Función	Editar un producto mediante su identificador único.
Tipo	PUT
Permisos	Administrador
Cuerpo	Identidad Categoría (idcategoria): entero Nombre (nom_pro): cadena Código (cod_pro): cadena Precio (precio_venta): real Cantidad (cantidad): entero Descripción (descripcion): cadena Imagen (img): Imagen Estado (estado): booleano
Ruta	/api/productos/{id}
Función	Eliminar un producto mediante su identificador único.
Tipo	DELETE
Permisos	Administrador
Ruta	/api/recordatorios
Función	Obtener una lista de recordatorios existentes.
Tipo	GET
Permisos	Administrador

Campo	Descripción
Ruta	/api/recordarios/{id}
Función	Obtener recordatorio mediante su identificador único.
Tipo	GET
Permisos	Administrador
Ruta	/api/recordatorios
Función	Crear un nuevo recordatorio.
Tipo	POST
Permisos	Administrador
Cuerpo	Identidad usuario (idusuario): entero Título (titulo): cadena Descripción (descripcion): cadena Fecha de Expiración (fecha_vencimiento): date Estado (estado): booleano
Ruta	/api/recordatorios/{id}
Función	Editar un recordatorio mediante su identificador único.
Tipo	PUT
Permisos	Administrador
Cuerpo	Identidad usuario (idusuario): entero Título (titulo): cadena Descripción (descripcion): cadena Fecha de Expiración (fecha_vencimiento): date Estado (estado): booleano

Campo	Descripción
Ruta	/api/recordatorios/{id}
Función	Eliminar un recordatorio mediante su identificador único.
Tipo	DELETE
Permisos	Administrador
Ruta	/api/anuncios
Función	Obtener una lista de anuncios existentes.
Tipo	GET
Permisos	Administrador
Ruta	/api/anuncios/{id}
Función	Obtener anuncio mediante su identificador único.
Tipo	GET
Permisos	Administrador
Ruta	/api/anuncios
Función	Crear un nuevo anuncio.
Tipo	POST
Permisos	Administrador
Cuerpo	<p>Identidad usuario (idusuario): entero</p> <p>Título (titulo): cadena</p> <p>Descripción (descripcion): cadena</p> <p>Fecha de Expiración (fecha_vencimiento): date</p> <p>Estado (estado): booleano</p>

Campo	Descripción
Ruta	/api/anuncios/{id}
Función	Editar un anuncio mediante su identificador único.
Tipo	PUT
Permisos	Administrador
Cuerpo	Identidad usuario (idusuario): entero Titulo (titulo): cadena Descripción (descripcion): cadena Fecha de Expiración (fecha_vencimiento): date Estado (estado): booleano
Ruta	/api/anuncios/{id}
Función	Eliminar un anuncio mediante su identificador único.
Tipo	DELETE
Permisos	Administrador
Ruta	/api/usuarios/Obtener
Función	Obtener una lista de usuarios existentes.
Tipo	GET
Permisos	Administrador
Ruta	/api/usuarios/obtenerUsuario/{id}
Función	Obtener un usuario mediante su identificador único.
Tipo	GET
Permisos	Administrador

Campo	Descripción
Ruta	/api/usuarios/Crear
Función	Crear un nuevo usuario.
Tipo	POST
Permisos	Administrador o cliente
Cuerpo	Identidad Rol (idrol): entero Correo (correo): cadena Nombre (nombre): cadena PIN (pin): entero Dirección (direccion): cadena Teléfono (telefono): cadena Estado (estado): booleano
Ruta	/api/usuarios/Editar/{id}
Función	Editar un usuario mediante su identificador único.
Tipo	PUT
Permisos	Administrador o cliente
Cuerpo	Identidad Rol (idrol): entero Correo (correo): cadena Nombre (nombre): cadena PIN (pin): entero Dirección (direccion): cadena Teléfono (telefono): cadena Estado (estado): booleano

Campo	Descripción
Ruta	/api/usuarios/Delete/{id}
Función	Eliminar un usuario mediante su identificador único.
Tipo	DELETE
Permisos	Administrador
Ruta	/api/usuarios/Login
Función	Iniciar sesión.
Tipo	POST
Permisos	Administrador, Cliente
Cuerpo	Correo (correo): cadena PIN (pin): entero Estado (estado): booleano
Ruta	/api/usuarios/OtpConfirm
Función	Confirmar el registro del usuario.
Tipo	POST
Permisos	Administrador o cliente
Cuerpo	Correo (correo): cadena PIN (pin): entero Código de verificación (otp): entero
Ruta	/api/usuarios/OtpResend
Función	Reenviar código de verificación.
Tipo	POST
Permisos	Administrador o cliente
Cuerpo	Correo (correo): cadena

Campo	Descripción
Ruta	/api/pedidos/Get
Función	Obtener una lista de pedidos existentes.
Tipo	GET
Permisos	Administrador
Ruta	/api/pedidos/Detallesid/{id}
Función	Obtener una lista de detalles de pedido mediante el identificador único del usuario.
Tipo	GET
Permisos	Administrador o cliente
Ruta	/api/pedidos/Detalles
Función	Obtener una lista de detalles de pedidos existentes.
Tipo	GET
Permisos	Administrador
Ruta	/api/pedidos/Pedidosid/{id}
Función	Obtener una lista de pedidos mediante el identificador único del usuario.
Tipo	GET
Permisos	Administrador o cliente
Ruta	/api/pedidos/Getid/{id}
Función	Obtener un pedido mediante su identificador único.
Tipo	GET
Permisos	Administrador o cliente

Campo	Descripción
Ruta	/api/pedidos/CrearPedido
Función	Crea un nuevo pedido.
Tipo	POST
Permisos	Administrador o cliente
Cuerpo	<p>Identificador del cliente (idcliente): entero</p> <p>Identificador del motorista (idusuario): entero</p> <p>Dirección (direccion): cadena</p> <p>Teléfono (telefono): cadena</p> <p>Fecha (fecha): date</p> <p>Impuesto (impuesto): decimal</p> <p>Subtotal (subtotal): decimal</p> <p>Total (total): decimal</p> <p>Tipo de pago (tipo_pago): cadena</p> <p>Estado (estado): booleano</p>
Ruta	/api/pedidos/Delete/{id}
Función	Eliminar un pedido mediante su identificador único.
Tipo	DELETE
Permisos	Administrador o cliente