

UNIVERSIDAD AUTÓNOMA DE CIUDAD JUÁREZ

Instituto de Ingeniería y Tecnología

Departamento de Ingeniería Eléctrica y Computación



DESARROLLO DE SISTEMA INFORMÁTICO PARA LA  
GESTIÓN Y ADMINISTRACIÓN DE UNA MICROEMPRESA  
TORTILLERA

Reporte Técnico de Investigación presentado por:

José Ángel Doménico Díaz 148791

Requisito para la obtención del título de:

INGENIERO EN SISTEMAS COMPUTACIONALES

M. C. Arnulfo Castro Vásquez

Ciudad Juárez, Chihuahua

16 de noviembre de 2019

Ciudad Juárez, Chihuahua, a 7 de Noviembre de 2019

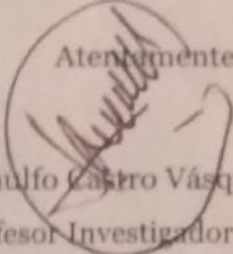
Asunto: Liberación de Asesoría

**Mtro. Ismael Canales Valdiviezo**  
**Jefe del Departamento de Ingeniería**  
**Eléctrica y Computación**  
**Presente.-**

Por medio de la presente me permito comunicarle que, después de haber realizado las asesorías correspondientes al reporte técnico Desarrollo de sistema informático para la gestión y administración de una microempresa tortillera, del alumno José Ángel Doménico Díaz, de la Licenciatura en Ingeniería en Sistemas Computacionales, considero que lo ha concluido satisfactoriamente, por lo que puede continuar con los trámites de titulación intracurricular.

Sin más por el momento, reciba un cordial saludo.

Atentamente

  
Arnulfo Castro Vásquez  
Profesor Investigador IIT

Cep:  
Coordinador del Programa de Sistemas Computacionales  
José Ángel Doménico Díaz  
Archivo



Ciudad Juárez, Chihuahua, a 12 de Noviembre de 2019

Asunto: Autorización de Publicación

C. José Ángel Doménico Díaz

Presente.-

En virtud de que cumple satisfactoriamente los requisitos solicitados, informo a usted que se autoriza la publicación del documento de Desarrollo de sistema informático para la gestión y administración de una microempresa tortillera, para presentar los resultados del proyecto de titulación con el propósito de obtener el título de Licenciado en Ingeniería en Sistemas Computacionales.

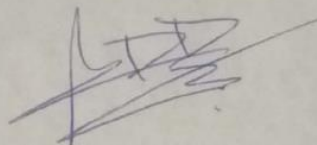
Sin otro particular, reciba un cordial saludo.



Mtra. Ivonne Haydee Robledo Portillo  
Profesor Titular de Seminario de Titulación II

## Declaración de Originalidad

Nosotros/Yo, José Ángel Doménico Díaz, , declaramos que el material contenido en esta publicación fue elaborado con la revisión de los documentos que se mencionan en el capítulo de Bibliografía, y que la solución obtenida es original y no ha sido copiada de ninguna otra fuente, ni ha sido usada para obtener otro título o reconocimiento en otra institución de educación superior.



José Ángel Doménico Díaz

# Agradecimientos

Quiero agradecer al profesor Arnulfo Castro Vázquez por ser mi asesor de proyecto de titulación y haberme brindado toda su atención, tiempo y apoyo durante el desarrollo de este proyecto. De igual forma quiero agradecer al profesor Rene Noriega por haberme guiado y aconsejado al momento de realizar el anteproyecto, a la Dra. Ivonne Robledo por haberme ayudado a concluir este proyecto y aconsejado de una manera atenta y con mucho interés para poder finalizar este trabajo de titulación.

Quiero agradecer a todos mis maestros que me acompañaron y me impartieron sus conocimientos de manera profesional durante estos 4 años y que siempre me apoyaron de manera positiva para llegar hasta este punto.

Quiero agradecer a la Universidad Autónoma de Ciudad Juárez la cual me abrió sus puertas, para poder prepararme para un futuro mejor y competitivo, formándome como una persona responsable y profesionalmente.

# Dedicatoria

La elaboración de este trabajo está dedicado a mis padres que con todo su apoyo me brindaron su atención, me inculcaron valores y educación. A mis hermanos por todo su apoyo y para que también consigan sus metas y objetivos, a mis compañeros de la universidad que juntos logramos llegar a este punto con gran esfuerzo, a mi novia por siempre acompañarme, aconsejarme y apoyarme durante todo este proceso que estuve en la universidad, por último, quiero agradecer principalmente a Dios por permitirme llegar hasta este punto, permitiéndome lograr un objetivo más en mi vida.

# Índice general

<b>1. Planteamiento del Problema</b>	<b>3</b>
1.1. Antecedentes . . . . .	3
1.2. Definición del problema . . . . .	5
1.3. Objetivo general . . . . .	6
1.4. Objetivos específicos . . . . .	6
1.5. Justificación . . . . .	7
<b>2. Marco teórico</b>	<b>9</b>
2.1. Lenguajes de desarrollo web . . . . .	9
2.1.1. HTML5 . . . . .	10
2.1.2. CSS . . . . .	14
2.1.3. JavaScript . . . . .	15
2.1.4. PHP . . . . .	15
2.2. Herramientas de desarrollo de aplicaciones móviles . . . . .	16
2.2.1. Android Studio . . . . .	16
2.2.2. App Inventor . . . . .	17

2.2.3. Ionic . . . . .	18
2.3. Frameworks de desarrollo web . . . . .	19
2.3.1. Bootstrap . . . . .	19
2.3.2. Semantic UI . . . . .	20
2.3.3. Foundation . . . . .	20
2.4. Base de datos . . . . .	21
2.4.1. Bases de datos relacionales . . . . .	21
2.4.2. SGBD (Sistemas gestores de bases de datos) . . . . .	22
2.4.3. MYSQL . . . . .	22
2.5. Seguridad de la información . . . . .	22
<b>3. Desarrollo del Proyecto</b>	<b>24</b>
3.1. Producto propuesto . . . . .	24
3.2. Metodología . . . . .	25
3.2.1. Metodología en cascada . . . . .	25
3.2.2. Fases . . . . .	26
3.3. Análisis de requerimientos . . . . .	27
3.3.1. Definición de requerimientos funcionales y no funcionales del sistema web . . . . .	28
3.3.2. Definición de requerimientos funcionales y no funcionales de la aplicación móvil . . . . .	32
3.3.3. Herramientas . . . . .	33

3.4. Diseño . . . . .	35
3.4.1. Arquitectura del sistema . . . . .	35
3.4.2. Diseño de la base de datos . . . . .	37
3.4.3. Casos de uso . . . . .	43
3.4.4. Diagrama de clases . . . . .	47
3.4.5. Diagrama de secuencia de los pedidos en la aplicación móvil . . . . .	49
3.5. Desarrollo . . . . .	50
3.5.1. Sistema web . . . . .	50
3.5.2. Funcionamiento del sistema web . . . . .	52
3.5.3. Aplicación móvil . . . . .	55
3.6. Pruebas . . . . .	59
3.6.1. Consulta y búsqueda de información . . . . .	60
3.6.2. Registro de información . . . . .	61
3.6.3. Eliminación de un registro . . . . .	62
3.6.4. Edición de información . . . . .	63
3.6.5. Validacion de autenticacion por tipo de usuario . . . . .	65
3.6.6. Validación de campos vacíos . . . . .	66
3.6.7. Validacion al enviar pedido . . . . .	67
<b>4. Resultados y Discusiones</b>	<b>68</b>
4.1. Presentación de resultados . . . . .	68
4.2. Resultados de encuestas . . . . .	69

<i>ÍNDICE GENERAL</i>	x
4.2.1. Usuarios empleados y administradores . . . . .	69
4.2.2. Usuario cliente . . . . .	72
4.3. Análisis de resultados . . . . .	75
4.3.1. Resultados del contenido de la aplicación . . . . .	75
<b>5. Conclusiones</b>	<b>77</b>
5.1. Con respecto al objetivo de la investigación . . . . .	77
5.2. Recomendaciones para futuras investigaciones . . . . .	78
<b>Bibliografía</b>	<b>80</b>
<b>A. Código fuente del sistema web</b>	<b>83</b>
<b>B. Bloques de códigos de la aplicación móvil</b>	<b>95</b>
<b>C. Formatos de encuestas realizadas a los usuarios</b>	<b>109</b>

# Índice de figuras

2.1. Blog normal . . . . .	13
2.2. Blog normal con etiquetas . . . . .	14
3.1. Metodología en cascada. . . . .	26
3.2. Arquitectura lógica del sistema. . . . .	36
3.3. Esquema entidad-relación diseñado para el proyecto. . . . .	42
3.4. Diagrama de caso de uso del sistema web. . . . .	44
3.5. Diagrama de caso de uso de la aplicación móvil. . . . .	46
3.6. Diagrama de clases del sistema web. . . . .	47
3.7. Diagrama de clases de la aplicación móvil. . . . .	48
3.8. Diagrama de secuencia del proceso de pedidos en la aplicación móvil. . . . .	49
3.9. Interfaz del módulo administrativo. . . . .	51
3.10. Interfaz del módulo operativo. . . . .	52
3.11. Interfaz de inicio de sesión del sistema web. . . . .	53
3.12. Estructura general de consulta de información. . . . .	54
3.13. Estructura general de registro de información. . . . .	55

3.14. Capturas de pantalla de las interfaces de la aplicación móvil. . . . .	57
3.15. Capturas de pantalla de las interfaces de la aplicación móvil. . . . .	58
3.16. Código de conexión a la base de datos. . . . .	59
3.17. Datos no encontrados usando el buscador. . . . .	60
3.18. Buscando información usando el buscador. . . . .	61
3.19. Validación de registro de la información. . . . .	62
3.20. Validación de eliminación de un registro. . . . .	63
3.21. Validación de edición de un registro. . . . .	64
3.22. Validación de edición de un registro. . . . .	64
3.23. Validación de autenticación de usuarios. . . . .	65
3.24. Validación de campos vacíos. . . . .	66
3.25. Validación de envió de pedidos. . . . .	67
4.1. Respuesta de la pregunta 1, encuesta de empleados y administradores . . . .	69
4.2. Respuesta de la pregunta 2, encuesta de empleados y administradores . . . .	70
4.3. Respuesta de la pregunta 3, encuesta de empleados y administradores . . . .	70
4.4. Respuesta de la pregunta 4, encuesta de empleados y administradores . . . .	71
4.5. Respuesta de la pregunta 5, encuesta de empleados y administradores . . . .	71
4.6. Respuesta de la pregunta 6, encuesta de empleados y administradores . . . .	72
4.7. Respuesta de la pregunta 7, encuesta de empleados y administradores . . . .	72
4.8. Respuesta de la pregunta 1, encuesta de clientes . . . . .	73
4.9. Respuesta de la pregunta 2, encuesta de clientes . . . . .	73

4.10. Respuesta de la pregunta 3, encuesta de clientes . . . . .	74
4.11. Respuesta de la pregunta 4, encuesta de clientes . . . . .	74
4.12. Respuesta de la pregunta 5, encuesta de clientes . . . . .	75
A.1. Código de consultas para la pantalla principal . . . . .	83
A.2. Librerías utilizadas de Bootstrap . . . . .	84
A.3. Código del layout para el header . . . . .	84
A.4. Código del layout para el sidebar parte 1 . . . . .	85
A.5. Código del layout para el sidebar parte 2 . . . . .	86
A.6. Código del inicio de sesión del sistema web . . . . .	87
A.7. Código de la estructura para la tabla de ventas . . . . .	88
A.8. Código para agregar clientes . . . . .	89
A.9. Código para editar registro de la tabla ventas . . . . .	90
A.10.Código para eliminar registro de la tabla ventas . . . . .	90
A.11.Código para buscador de la tabla ventas parte 1 . . . . .	91
A.12.Código para buscador de la tabla ventas parte 2 . . . . .	92
A.13.Código para buscador de la tabla ventas parte 3 . . . . .	93
A.14.Código para fecha en tiempo real del sistema web . . . . .	94
B.1. Pantalla de carga de la aplicación móvil . . . . .	95
B.2. Inicio de sesión de la aplicación móvil parte 1 . . . . .	96
B.3. Inicio de sesión de la aplicación móvil parte 2 . . . . .	96
B.4. Pantalla principal de la aplicación móvil parte 1 . . . . .	97

B.5. Pantalla principal de la aplicación móvil parte 2 . . . . .	97
B.6. Pantalla de realización de pedidos parte 1 . . . . .	98
B.7. Pantalla de realización de pedidos parte 2 . . . . .	98
B.8. Pantalla de visualización de pedidos parte 1 . . . . .	99
B.9. Pantalla de visualización de pedidos parte 2 . . . . .	100
B.10. Pantalla de visualización de pedidos parte 3 . . . . .	101
B.11. Pantalla de visualización de pedidos parte 4 . . . . .	102
B.12. Pantalla de usuario administrador de pedidos parte 1 . . . . .	103
B.13. Pantalla de usuario administrador de pedidos parte 2 . . . . .	104
B.14. Pantalla de usuario administrador de pedidos parte 3 . . . . .	105
B.15. Pantalla de usuario administrador de pedidos parte 4 . . . . .	106
B.16. Pantalla de usuario administrador de pedidos parte 5 . . . . .	107
B.17. Pantalla de usuario administrador de pedidos parte 6 . . . . .	108
C.1. Formato de encuesta realizada a los usuarios del sistema web . . . . .	109
C.2. Formato de encuesta realizada a los usuarios de la aplicación móvil . . . . .	110

# Índice de tablas

3.1. Requerimientos funcionales por usuario. . . . .	29
3.2. Requerimientos funcionales por el usuario administrador del sistema web. . .	30
3.3. Requerimientos funcionales por el usuario empleado. . . . .	31
3.4. Requerimientos no funcionales del sistema web . . . . .	32
3.5. Requerimientos funcionales de la aplicación móvil. . . . .	33
3.6. Requerimientos no funcionales de la aplicación móvil. . . . .	33
3.7. Entidades de la base de datos de forma conceptual. . . . .	38
3.8. Diccionario de la base de datos respecto a las entidades . . . . .	39

# Resumen

La presente investigación desarrollada se basa en la construcción de un sistema administrativo para la microempresa tortillera Doménico la cual consiste en el control y gestión de su información. En el primer capítulo se menciona los inicios de este negocio, además de los clientes que esta tiene, por otro lado, como es que manejan los ingresos monetarios y como es que los empleados se organizan para la fabricación del producto. En este mismo primer capítulo se habla sobre el objetivo general del proyecto y cuáles son esos objetivos específicos por cumplir.

Posteriormente en el capítulo 2 se habla sobre los conceptos importantes utilizados para este proyecto y algunos que se relacionan con lo que se desarrolló. Algunos conceptos importantes como el desarrollo de aplicaciones móviles que es necesaria para la construcción de nuestra aplicación, así como otros lenguajes como por ejemplo PHP y HTML5 que es utilizado para el desarrollo de sistemas web. También se hablará sobre conceptos de base de datos y como estas se diseñan para almacenar información de una manera eficiente.

En el desarrollo del proyecto se utilizó una metodología en cascada la cual dicha metodología consta de 5 etapas en las que cada una debe de terminar para así poder seguir avanzando en el desarrollo del proyecto estas 5 etapas son: análisis, diseño, desarrollo, pruebas y mantenimiento.

Durante el desarrollo del proyecto en la parte de análisis se mencionan las herramientas que se utilizaron para la construcción de este proyecto y que fueron de gran utilidad para su

realización. En el capítulo 4 se obtuvieron los resultados respecto al sistema administrativo terminado, en donde los usuarios del negocio opinaron por medio de una encuesta sobre si cumplió las expectativas que ellos tenían, y de igual forma a los usuarios clientes del negocio se les realizó una encuesta para ver si están satisfechos con la aplicación móvil y si esta les es útil para la realización de pedidos de una manera más fácil.

**Palabras clave:** Sistema administrativo, Base de datos, Sistema web, Metodología en cascada, Aplicación móvil.

# Abstract

The present research developed is based on the construction of an administrative system for the Dominican tortilla microenterprise which consists of the control and management of your information. In the first chapter, the beginnings of this business are mentioned, in addition to the clients that it has, on the other hand, how they manage monetary income and how employees organize themselves to manufacture the product. In this same first chapter we talk about the general objective of the project and what are those specific objectives to be met.

Later in chapter 2 we talk about the important concepts used for this project and some that relate to what was developed. Some important concepts such as the development of mobile applications that are necessary for the construction of our application, as well as other languages such as PHP and HTML5 that is used for the development of web systems. We will also talk about database concepts and how they are designed to store information in an efficient way.

In the development of the project a cascade methodology was used, which said methodology consists of 5 stages in which each one must finish in order to continue advancing in the development of the project, these 5 stages are: analysis, design, development, testing and maintenance.

During the development of the project in the analysis part, the tools that were used for the construction of this project and that were very useful for its realization are mentioned. In

Chapter 4, the results were obtained regarding the finished administrative system, where the business users expressed their opinion through a survey of whether they met the expectations they had, and in the same way the client customers of the business were conducted a survey. to see if they are satisfied with the mobile application and if it is useful for ordering in an easier way.

**Keywords:** Administrative system, Database, Web system, Cascade methodology, Mobile application.

# Introducción

Muchas ocasiones cuando los negocios se están expandiendo, es decir, están poniendo un nuevo local y/o sucursal de lo que van a ofrecer para vender, muchos inician sin antes tener una tecnología que les facilite sus procesos, o como es en este caso la administración del negocio respecto a la información que se maneja dentro de la microempresa tortillera. Algunos de los negocios llegan a mantenerse en el mercado, pero algunos otros no les dan las cuentas monetarias y tienden a cerrar el negocio a causa de bajas ventas.

Con la ayuda de esta tecnología se busca darles a esas pequeñas empresas que van comenzando con sus negocios, a ofrecerles una mejor experiencia en cuanto a la administración de su información como los son sus ventas, pedidos del día, administración de sus clientes, entre otras funciones que el sistema tiene. Con esta tecnología se les hará más fácil a las personas el control de su negocio, por ejemplos al momento de registrar las ventas y al momento de recibir pedidos por parte de sus clientes.

Además, esta tecnología va dirigida a cualquier tipo de negocio que necesite administrar bien su información y que quiera crecer como empresa.

En términos generales este proyecto describe como se desarrolló un sistema informático para la gestión y administración de una microempresa tortillera, que cumpla con las funciones de administrar la información de los clientes y los productos del negocio, de igual forma que ayude con el registro de las ventas tanto de los clientes consumidores como de los pedidos, por otro lado que tenga la capacidad de comunicarse con una aplicación móvil para

el registro del los pedidos por parte de sus clientes los cuales son tanto tiendas de abarrotes como empresas maquiladoras.

# Capítulo 1

## Planteamiento del Problema

En los siguientes apartados se presentan los antecedentes y la definición de la microempresa desde sus inicios, así como la definición del problema tomando puntos importantes como la manera en que llevan el control de sus productos y clientes. Se verán los proyectos similares que se han desarrollado en otros lugares y ver como solucionaron el problema al que se enfrentaban y cuáles fueron sus resultados.

### 1.1. Antecedentes

La microempresa tortillera “Doménico” tiene sus inicios en el mercado desde 03 de marzo del año 2000 ofreciendo su producto de tortilla, tiene como clientes tiendas de abarrotes en los cuales entrega pedidos todos los días. En la actualidad la microempresa “Doménico” realiza producto de tortilla, tostadas, tortillas para tacos, tortillas de harina entre otros productos. Actualmente la microempresa cuenta con dos sucursales en Ciudad Juárez, Chihuahua, donde la primera está ubicada en la Col. Fronteriza Baja Loreto 2419 y la segunda está en Arroyo de las víboras 111. Actualmente están por abrir una nueva sucursal que sería la tercera para esta microempresa aquí en ciudad Juárez.

Además de tener como clientes tiendas de abarrotes, también tiene empresas maquiladoras a los que les entrega producto para sus cafeterías. Esta empresa brinda una imagen renovada

de lo que es una tortillería, convirtiéndola no solo en una empresa sino también en una industria de la masa y la tortilla y esforzándose por elevar la calidad de este producto de primera necesidad, se trabaja minuciosamente en la textura de la tortilla de maíz dándole el punto exacto y logrando su inigualable sabor.

Existen sistemas y/o proyectos que realizan funciones similares a lo que se pretende hacer en este proyecto que es la administración y gestión de productos, así como de los pedidos de la microempresa.

En la USMP (Universidad de San Martín de Porres) en la Facultad de Ingeniería y Arquitectura se realizó un proyecto En la realización de una plataforma web para la administración de servicios del negocio lavanderías edén. Como mejora en los procesos del negocio, se obtuvieron resultados muy satisfactorios en el registro de clientes y servicios, en la distribución de prendas, entrega de prenda al cliente, además del servicio de entrega y la gestión de stock de recursos utilizados. Los clientes aportaron comentarios muy positivos por el gran y buen servicio brindado, por otro lado, se disminuyó el tiempo que tardaba en realizarse las entregas de prendas a los clientes, y la cantidad de prendas perdida se redujo en un buen porcentaje y los gastos de recursos más usados se pudieron controlar con mayor eficiencia en el negocio, siendo así se aumentó la optimización en el proceso de registro de pedidos y clientes [1].

En el repositorio de la USMP se encuentra otro proyecto similar en donde se desarrolló e implementó un sistema web con aplicación móvil de reservaciones y pedidos en el entorno de negocios de los restaurantes. El sistema antes mencionado permite hacer registros de los clientes y reservas que se realizan de manera telefónica o por medio del correo electrónico, y agregando un stock de las cantidades de reservas que se tendrán durante el día.

En cuanto a la aplicación que ellos realizaron, su función es registrar al cliente y las reservaciones de manera anticipada hechas por el cliente y/o usuario. Estas son visualizadas por el encargado de reservas mediante el sistema web administrativo que ellos tienen [2].

En cuanto a lo que se pretende realizar en este proyecto es algo similar, pero con el

negocio de la microempresa tortillera ya que no cuenta con la tecnología para administrar y gestionar la información de su negocio y buscan con el sistema que se pueda tener un registro de sus clientes, así como mejorar el servicio de entrega del producto mediante la gestión de los pedidos.

## 1.2. Definición del problema

La problemática en esta microempresa radica en que no llevan una buena administración u organización. La manera en que ellos se organizan no es una manera eficiente, ya que los ingresos monetarios que esta microempresa recibe los guardan en hojas de papel, así también como los pedidos que les van llegando los almacenan de igual forma y estos han llegado a perderse.

La comunicación que el negocio tiene con sus clientes se puede decir que no es la apropiada y mucho menos eficiente ya que el cliente al realizar pedidos lo hace de manera telefónica con el encargado del negocio, ha sucedido que el cliente hace su pedido y al momento de que se le lleva el producto y este lo recibe, comenta que había pedido una cantidad menor a la solicitada entonces es ahí cuando el negocio pierde dinero al hacer más cantidad de producto del que ya se había acordado y/o solicitado.

En cuanto a los empleados en la manera que fabrican su producto es un poco desorganizada ya que la manera en que llevan por así decir la cola de pedidos es en una libreta en donde hacen sus apuntes y esta puede llegar a perderse o causar confusión entre los pedidos a realizar. Existe una organización deficiente por parte del negocio ya que no cuentan con un medio donde puedan almacenar la información de sus clientes mucho menos un control de sus productos. El no contar con registro de sus clientes y su historial de pedidos o compras puede causar problemas al momento de realizar algún reporte o facturación que el cliente solicite al negocio y este no se encuentre disponible o en dado caso no se les pueda proporcionar.

Se tiene un lapso durante el día en donde se empiezan a comunicar con el encargado del negocio para realizar pedidos entonces es ahí donde no puede atender simultáneamente todos los pedidos a la vez, es por eso por lo que se quiere implementar sistema donde estén llegando los pedidos en tiempo real y los empleados los puedan visualizar sin que se les interrumpa de la operación que estén realizando.

Resumiendo, el problema está en que no existe una administración en la cual pueda llevar sus pedidos y otras funciones de manera ordenada. La comunicación con sus clientes no es la más adecuada ya que no se cuenta con un medio donde se tenga evidencia que el cliente realiza cierta cantidad de producto. Este problema de organización les ha causado pérdida de dinero y desperdicio de producto durante mucho tiempo. De igual manera el no tener un control de los pedidos y un medio donde se administren les afecta en el tiempo que realizan su producto y el envío a sus tiendas y sucursales maquiladoras. Es por ello por lo que se propone el uso de las tecnologías para desarrollar e implementar un sistema para su gestión de pedidos en el cual se podrán organizar de una manera rápida, segura y eficaz.

### **1.3. Objetivo general**

Desarrollar un sistema de información para una microempresa tortillera que administre los procesos de un negocio, algunas funciones son: registro de ventas, control y gestión de sus pedidos y reportes de ventas. Incluyendo la capacidad de comunicarse con una aplicación móvil para la realización de sus pedidos por parte de sus clientes.

### **1.4. Objetivos específicos**

- Analizar, diseñar e implementar las Interfaces Graficas de los Usuarios del sistema de información.

- Analizar, diseñar e implementar la Base de datos para la empresa de tortillería del sistema de información.
- Analizar, diseñar e implementar la aplicación para dispositivo móvil.

## 1.5. Justificación

Actualmente se cuenta con tecnología donde se puede sacar beneficio para solucionar un problema o facilitar las cosas a una persona o por lo que es en este caso un negocio. La manera en que se puede comunicar una aplicación móvil con un sistema de información toma mucha importancia y una gran ventaja ya que acelera la comunicación entre los usuarios al momento de realizar alguna transacción.

Surge la necesidad de realizar una aplicación que permita coadyuvar en la gestión y manejo de los datos de la microempresa de manera eficiente, y por parte del negocio un sistema capaz de controlar y realizar una buena administración del trabajo que se lleva en la microempresa tortillera.

El administrar los pedidos de una forma más sencilla y eficiente posible le permitirá al encargado del negocio tener un control de su cola de pedidos y de sus productos. Se busca tener una buena comunicación al momento de realizar algún pedido al negocio, con esto para que se tenga la evidencia en donde se concretó dicha operación o transacción para que no haya conflicto alguno, siendo así que el negocio no pierda dinero en hacer mayores cantidades de producto de lo que ya se había solicitado.

El encargado de utilizar el sistema se beneficiará en tiempo y costos, ya que con el sistema que se quiere desarrollar e implementar en la microempresa se busca reducir las tareas que el usuario hace manualmente en hojas de papel o libretas. Se busca que el usuario que son las tiendas de abarrotes al igual que las cafeterías dentro de las empresas maquiladoras realicen

la transacción de sus pedidos de una manera sencilla y rápida para que no pierdan tanto tiempo en comunicarse con el proveedor.

La tecnología a desarrollar para esta microempresa es de gran importancia ya que se busca controlar el proceso de la administración de los pedidos, el control de sus ingresos monetarios, inventarios de productos y las transacciones por parte sus clientes. Esta aplicación no solo va dirigida hacia esta empresa, sino que le puede ayudar a negocios similares a solucionar y automatizar sus procesos de administración y atención de pedidos de sus clientes.

# Capítulo 2

## Marco teórico

Actualmente las tecnologías con las que se tiene a la mano son de gran ayuda para las labores diarias, ya sea en los negocios, en los trabajos, en la escuela o en las actividades cotidianas. Las empresas y organizaciones dan mucho de qué hablar con el uso de las tecnologías ya que gran parte de sus procesos están relacionados con algún sistema y/o tecnología de información. Es por eso por lo que los sistemas de información toman un papel importante en las empresas ayudando y mejorando los procesos dentro de ella y de igual manera en la vida diaria de una persona siendo más eficiente y facilitando el hacer una actividad que se requiera. Los siguientes apartados servirán para definir algunos conceptos teóricos y prácticos, que serán de gran ayuda e indispensables para el desarrollo del sistema de información. Se realizará un análisis para seleccionar los temas para el desarrollo del proyecto.

### 2.1. Lenguajes de desarrollo web

Actualmente se pueden encontrar con diferentes lenguajes de programación para desarrollo web, ya que estos con el tiempo han estado apareciendo debido a las tendencias y necesidades de las plataformas en el Internet. Anteriormente se mencionó que existen otros lenguajes de programación web que se usan o mejor dicho van mejorando el contenido del sitio web, entre ellos esta CSS que mejora la presentación del documento, JavaScript que permite darle

funcionalidad al sitio, PHP que conecta el sitio web con una base de datos entre otros.

En las siguientes secciones se hablará sobre los lenguajes más utilizados para el desarrollo web y como estos funcionan. Los lenguajes necesarios para el proyecto son HTML5, PHP, Javascript donde se abarcará cada uno de ellos, además se hablará sobre lenguajes que se relacionan y ver por qué los anteriores mencionados fueron los seleccionados para este proyecto. Cada lenguaje tiene su sintaxis por eso es importante conocer sus características de cada uno y saber aplicarlo en el documento donde se estará programando.

### 2.1.1. HTML5

Al paso del tiempo han existido distintas versiones de HTML y estas han ido evolucionando de manera que se ha llegado a la actual que es hoy en día la de HTML5. HTML5 es una de la nueva versión de este lenguaje de marcado y es utilizado actualmente, en donde algunos de sus objetivos son [3]:

- Presentar y organizar la información de modo que este lo más dividida posible.
- El código generado debe de ser lo más simple y resumido posible al momento de usarlo en los proyectos.
- Un objetivo más importante es que las páginas de hoy en día se adapten a dispositivos móviles, y además que estas sean compatibles con todos los navegadores que existen.

La organización de los documentos HTML es muy precisa. Dado que cada parte de la estructura esta diferenciada de manera clara y concisa por etiquetas específicas. Para indicar el tipo de documento que se está realizando, se utiliza la siguiente etiqueta:

```
<!DOCTYPE html >
```

La etiqueta DOCTYPE siempre debe ser la primera línea del documento, sin espacio ni líneas que estén antes que esta. Así de esta manera el navegador realiza activa una función donde puede determinar e interpretar las incorporaciones del HTML5 siempre que sea posible o de lo contrario puede ignorarlas. Seguidamente después de declarar el documento, lo que se debe hacer es empezar a construir la estructura HTML. La siguiente estructura en este lenguaje tiene su raíz en el elemento donde encapsula todo el código:

```
<!DOCTYPE html>
<html lang=es>
</html>
```

Después del doctype, el atributo lang en la etiqueta que abre es el único que se necesita para especificar en HTML5. Lo que hace esto es que determina el idioma que se va a utilizar en el documento que estamos creando, dado el caso puede ser español. Las etiquetas <head> definen el título que se visualizara en la página web, además de que se declaran los caracteres correspondientes, se coloca información general sobre el sitio web y se añaden los archivos externos como códigos de javascript, estilos de css o imágenes necesarios para mostrar la página al usuario.

```
<!DOCTYPE html>
<html lang=es>
  <head>
</head>
</html>
```

El contenido que tendrá la parte que será mostrada al usuario y que será mostrada en la página web esta especificado enter un par de etiquetas llamadas *body*.

```
<!DOCTYPE html>

<html lang=es>

  <head>

  </head>

  <body>

  </body>

</html>
```

Hasta aquí viene siendo lo más básico de lo que es HTML5, además de que tiene más etiquetas, esta vendría siendo la estructura más simple de un documento HTML5. La Figura 2.1 es una representación de un blog normal. En este ejemplo se puede claramente identificar cada parte del diseño.

1. Cabecera
2. Barra de Navegación
3. Sección de Información Principal
4. Barra Lateral
5. El pie o la barra Institucional



Figura 2.1: Blog normal

En la Figura 2.2 se puede ver un diseño básico presentado anteriormente, pero en esta parte con las correspondientes etiquetas HTML5 para cada sección.



Figura 2.2: Blog normal con etiquetas

De esta manera es como muchos de los sitios web tienen la estructura de su cuerpo. Y así es como funciona HTML5, esto es solo lo más básico, pero da a entender cómo funciona HTML5.

### 2.1.2. CSS

El concepto de CSS se define como "Hojas de Estilo en Cascada (del inglés *Cascading Style Sheets*)" o CSS es el lenguaje utilizado para describir la presentación de documentos HTML o XML, esto incluye varios lenguajes basados en XML como son XHTML o SVG. CSS

describe como debe ser renderizado el elemento estructurado en pantalla, en papel, hablado o en otros medios”.

Los atributos que tiene el css es una manera definida de sobrescribir los estilos que tiene los navegadores y de esta forma organizar el documento a como se requiere. Estos atributos no siempre tienen que ser de algún modo específicos, pueden estar combinados para formar reglas que después serán utilizados para agrupar modelos de cajas y así tener una disposición correcta en la pantalla que se muestra en el navegador. A todo lo anterior y a la combinación de estas reglas se le llama modelo de caja o sistema de disposición.

### 2.1.3. JavaScript

“JavaScript es un lenguaje de programación que permite realizar actividades complejas en una página web, cada vez más un sitio web hace más cosas que sólo mostrar información estática como mostrar actualizaciones de contenido en el momento, interactuar con mapas, animaciones gráficas 2D/3D etc.”.

Algunas de las características principales que tiene javascript son [4]:

- Almacenar cualquier tipo de datos en variables.
- Operaciones escritas con texto tipo string en lenguajes de programación.
- Hacer la página web más dinámica en cuanto al código y algunos eventos que sucedan en el contenido.

### 2.1.4. PHP

Este lenguaje surge como un lenguaje denominado como código abierto (*open source*) y es un lenguaje que es interpretado del lado del servidor. Al igual que otros lenguajes de

programación web este es integrado directamente dentro de la estructura HTML, además de que se caracteriza por su modularidad, rapidez y robustez.

Comparado con ASP, la principal ventaja de PHP es su característica multiplataforma. De otro modo viendo, los programas en ASP resultan más lentos y pesados, y también menos estables. En los entornos de Microsoft la gran ventaja de ASP es que los servidores web de Microsoft soportan ASP sin la necesidad de ninguna instalación adicional. [5].

## 2.2. Herramientas de desarrollo de aplicaciones móviles

Esta área de programación es donde se realizan aplicaciones móviles para teléfonos inteligentes como smartphones, tablets, etc. Por medio de procesos y procedimientos que involucran la escritura de código de software.

El proceso de creación de aplicaciones para dispositivos móviles es muy similar a los procesos para aplicaciones en la plataforma web y tiene sus inicios en el desarrollo de software más habitual y/o común. La diferencia es que las aplicaciones para teléfonos inteligentes se programan específicamente para las funciones de un teléfono móvil en particular [6].

### 2.2.1. Android Studio

Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones para Android y se basa en IntelliJ IDEA. Además del potente editor de códigos y las herramientas que ofrece para desarrolladores de IntelliJ, Android Studio promete y ofrece más funciones que aumentan la productividad durante la compilación de apps para Android, como las siguientes [7] :

- Gran flexibilidad a la hora de compilar el código.

- Potente emulador con muchas funciones para utilizar
- Un entorno donde se pueden empezar a desarrollar aplicaciones que sean compatibles con todos los dispositivos móviles.
- La importación de plantillas y el trabajo agregado de Github para compilar algunas funciones.
- Herramientas para visualizar problemas de rendimiento, y compatibilidad.
- Gran cantidad de herramientas y frameworks de prueba.
- Además de soportes por parte de Google, lo que facilita la integración de alguna de sus aplicaciones o plugins.

### 2.2.2. App Inventor

El MIT App Inventor es una plataforma de programación para dispositivos inteligentes que permite a cualquier persona crear apps funcionales para facilitar algunas funciones de los teléfonos y tabletas hoy en día. Aquellos que no han utilizado esta plataforma, pueden empezar a desarrollar su primera aplicación de manera simple, esto en no más de 30 minutos.[8]

Ya que la plataforma está basada en programar por medio de bloques de código, hace más fácil la construcción y/o elaboración de aplicaciones más complejas y de alto rendimiento en un tiempo menor que en otros entornos de programación. Esta plataforma es gratuita y está siendo usada por más de 6 millones de usuarios registrados en el mundo, además de ser líderes y fomentar la educación en torno a MIT App Inventor y realizar estudios sobre sus impactos.

Algunas de sus características son [9]:

- No se necesita instalar un IDE.

- Se necesitan mínimos conocimientos de programación.
- Desarrollo rápido de aplicaciones con bajos niveles de error.
- Almacenamiento en la nube para guardar nuestros proyectos.

### 2.2.3. Ionic

“Ionic es una herramienta, gratuita y open source, para el desarrollo de aplicaciones híbridas basadas en HTML5, CSS y JS. Está construido con Sass y optimizado con el framework AngularJS. El MVC (Model-View-Controller o Modelo-Vista-Controlador), es un patrón de diseño que separa los datos, la lógica y las interfaces del usuario. Su nombre lo dice, está separado en tres componentes: Modelo, Vista y Controlador. Está basado en la idea de separación de conceptos y cumple perfectamente con los objetivos de los patrones de diseño” [10]:

- Modelo: es la capa que dirige los datos, es decir, es la que determina como hacer las peticiones a las bases de datos para enviar o recibir información requerida. Las bases de datos pueden estar almacenadas de forma local en la app o de manera remota en un servidor externo.
- Vista: esta parte del código es la que da la posibilidad de visualizar y presentar los datos que el anterior concepto de modelo proporciona, de esta manera determinamos que en una aplicación con código HTML permite mostrar la salida de los datos procesados.
- Controlador: esta última capa es la que proporciona el enlace entre la vista y el modelo. Enviando instrucciones a la capa del modelo para actualizar su estado, y seguidamente mostrar el contenido en la capa de vista

## 2.3. Frameworks de desarrollo web

Los frameworks web son un conjunto de herramientas, estilos y librerías dispuestas a través de una estructura o esqueleto base, para el desarrollo de aplicaciones web más escalables y fáciles de mantener. Algunas de las ventajas de usar un framework web son [11]:

- Documentación y comunidad: La cantidad de documentación que se encuentra sobre un framework web, suele ser enorme y además con una gran comunidad detrás, respondiendo preguntas y desarrollando nuevas funcionalidades.
- Reutilización del código: Uno de los puntos fuertes e importantes de los frameworks web es la modularidad que tienen con su código y la capacidad para poder realizar múltiples proyectos con el mismo código, cambiando fácilmente los textos.
- Arquitectura y metodología: La mayoría de frameworks web del mercado usan arquitecturas y metodologías actuales, como el Modelo-Vista-Controlador.

Se ha decidido trabajar con Bootstrap por su sencillez al momento de realizar un sitio web y la manera en que este se adapta al momento de ser mostrado al usuario final.

### 2.3.1. Bootstrap

“Inicialmente creado como una solución interna para twitter y posteriormente liberado al público en agosto del 2011 como un proyecto Open Source en GitHub, en los meses siguientes a su liberación la comunidad apoyó activamente este proyecto hasta convertirlo en lo que es hoy el más popular HTML, CSS y JS framework para desarrolladores de manera responsiva, para proyectos móviles en la web” [12].

### 2.3.2. Semantic UI

Este es framework para crear el diseño de interfaces de manera responsiva lenguajes de marcado HTML y de estilos como CSS. Su origen da inicio en el año 2013 y actualmente esta en la versión 2.2. en donde viene relacionado e integrado con otros frameworks de diseño o librerías como son Angular, React, entre otros. Utiliza cinco categorías para determinar sus componentes [13]:

- UI Element: es el bloque de construcción básico, puede aparecer solo o agrupado y además puede un botón independiente y/o estar dentro de un grupo de botones.
- UI Collection: puede ser un formulario donde dentro tenga botones, input, checkbox e íconos.
- UI View: puede ser una sección de feeds o comentarios.
- UI Module: este elemento tiene como función la interactividad basada en javascript, como los modales entre otras cosas.
- UI Behaviour: este componente no puede estar de manera independiente, si no que debe de funcionar como ayuda de funcionalidad para otros componentes.

### 2.3.3. Foundation

Primeramente, fue lanzada como un marco de aplicaciones de código abierto en septiembre de 2011, *Foundation* inicio como una guía de estilo interna creada por la agencia de diseño *ZURB* cuyos clientes son *Facebook*, *eBay* y *NYSE*. *Foundation* 6 es la última versión estable que se tiene, donde se lanzó en noviembre de 2015. Como su nombre indica, el objetivo de *Foundation* es proporcionarle al desarrollador la base sobre la que diseñará su sitio web [14].

En la versión 2.0 en adelante se soporta el diseño responsivo. Con la característica de que el diseño gráfico de los sitios web se ajusten de forma dinámica, en función de las características del terminal o dispositivo de uso (PC, Tablet, mobile phone). Luego, la versión 4.0 tuvo un enfoque *mobile-first*, diseñando y desarrollando en primer lugar para terminales móviles más ajustables, de una manera que mejore los sitios web y apps para pantallas más grandes. *Foundation* es open source y está disponible en el repositorio de Github. Las especiales características del proyecto animan a los desarrolladores a colaborar y a contribuir a la mejora continua de la plataforma o este framework [15].

## 2.4. Base de datos

Es una colección de datos, normalmente denominada base de datos que contiene información relevante para una empresa. En si las bases de datos se caracterizan por la representación de un problema en el mundo real, algunos lo llaman banco de datos, estas bases de datos digitales almacenan información que se relaciona con otros datos donde su principal idea es la organización de los datos. Siendo así de gran importancia y utilidad para guardar información y poder tener acceso a la información de manera rápida y eficaz [16].

### 2.4.1. Bases de datos relacionales

Se utiliza para modelar problemas reales, administrar y organizar la información de manera dinámica, es un modelo que su principal función son las relacionar las entidades del diagrama como grupos a los que son llamados tuplas, muchas de las veces se pueden visualizar de una manera fácil de imaginar y hasta de modelar, teniendo en cuenta que cada relación fueran una tabla compuesta por campos y tuplas [17].

### 2.4.2. SGBD (Sistemas gestores de bases de datos)

“Es un conjunto de datos interrelacionados y con herramientas computacionales específicas para acceder a dichos datos. Su lugar de almacenamiento se denomina base de datos, pues es aquella que contiene información relevante de una empresa u organización. El objetivo principal de un SGBD es de almacenar y recuperar la información de la organización o empresa de forma práctica y eficiente” [18].

### 2.4.3. MYSQL

Es una base de datos entre las más usadas en el mundo. Su arquitectura, es rápida, fácil de personalizar y/o configurar su reutilización de código dentro del software ha logrado un sistema de administración con mucha mayor velocidad, compactación, estabilidad y facilidad de despliegue, tiene una compañía comercial llamada MySQL AB, desarrollada por los fundadores de MySQL. Es una base de datos relacional cuya función es de archivar y administrar los datos en tablas separadas en vez de colocarlas en un solo archivo. Todas las mismas tablas están conectadas por relaciones que hacen de una manera posible combinar datos de diferentes tablas; con el fin de obtener mayor velocidad, eficacia y flexibilidad [19].

## 2.5. Seguridad de la información

La seguridad de la información permite garantizar la identificación, valoración y gestión de los activos de información y sus riesgos, en función del impacto que representan para una organización. Es un concepto amplio que no se centra en la protección de las TIC sino de todos los activos de información que son de un alto valor para la institución. La seguridad de la información se debe de entender como un grupo de estrategias y medidas preventivas que se deben de realizar en las empresas y/o instituciones para resguardar la información y

mantener tres conceptos importantes: confidencialidad, disponibilidad e integridad [20].

- Confidencialidad: Propiedad que permite que la información solo esté disponible o sea revelada a personas, entidades o procesos utilizados.
- Confidencialidad: Propiedad de la exactitud e integridad de la información.
- Disponibilidad: Propiedad de la información para estar accesible y utilizable al solicitarlo una entidad autorizada.

# Capítulo 3

## Desarrollo del Proyecto

El siguiente capítulo estará enfocado al desarrollo del proyecto además de su construcción e implementación de una de manera estructurada siguiendo una metodología. Se describe cada una de las fases respecto a la metodología a usar que es la de cascada y en cada una de las fases se detalla lo que se plantea hacer y desarrollar además de sus requerimientos funcionales y no funcionales. Así mismo se verán diferentes diagramas para conocer el funcionamiento del sistema desde la plataforma web hasta con la aplicación móvil y se verá cómo estos dos se comunican con una base de datos a un servidor.

### 3.1. Producto propuesto

Este sistema administrativo se propuso con la finalidad de poder organizar de una manera eficiente toda la información de la microempresa como sus ventas del día, gestión de pedidos, clientes y productos además de brindarle a los clientes de esta microempresa una mejor atención al momento de requerir mas producto para su establecimiento.

De manera general el funcionamiento que este sistema tendrá es:

- Almacenar de manera organizada y eficaz toda la información importante dentro de una base de datos para posteriormente mostrarla al usuario de manera ordenada.

- Proporcionar una aplicación móvil para los clientes del establecimiento para realizar los pedidos de una manera más fácil y rápido.
- Mostrar los reportes de ventas de cualquier día al administrador del negocio cuando él los requiera.

## 3.2. Metodología

Al momento de iniciar y desarrollar un proyecto de software es importante aplicar metodologías, modelos y técnicas que permitan resolver problemas con el fin de satisfacer las necesidades de un cliente en particular o mercado en general.

En este proyecto se decidió usar la metodología tradicional en cascada dado que es la que se ajustó más a este proyecto al momento de analizar diferentes metodologías., acerca de esta metodología se puede decir que es un enfoque clásico para el ciclo de vida del desarrollo de sistemas, donde el inicio de cada etapa debe de esperar la finalización de la etapa anterior.

### 3.2.1. Metodología en cascada

La metodología en cascada es llamada así en la ingeniería de software por la posición de las fases en la que se encuentra estructurada y por la manera en que se lleva a cabo. De tal forma que al inicio de cada etapa debe esperar la finalización de la etapa anterior. El modelo está diseñado de manera que se debe llevar una revisión final y esta se encarga de determinar si el proyecto puede avanzar a la siguiente fase o no [21]. En la Figura 3.1 se muestran las fases de la metodología en cascada.

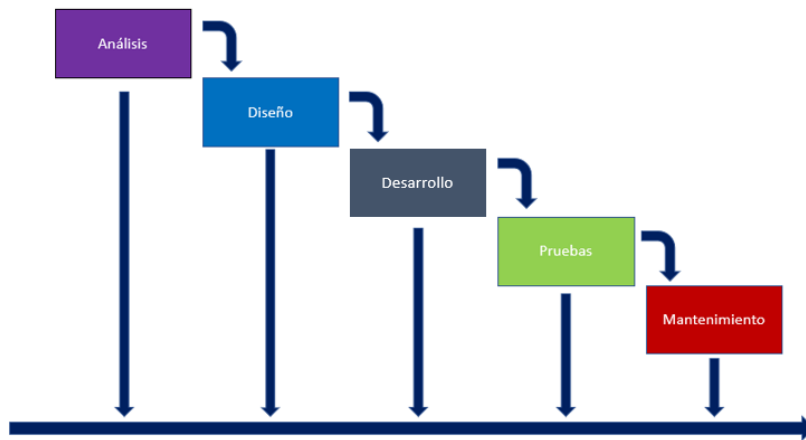


Figura 3.1: Metodología en cascada.

### 3.2.2. Fases

#### Análisis.

En esta fase se inicia tomando los requerimientos del software son los elementos que el sistema lleva y acomodarlos de una manera organizada para posteriormente pasar a la siguiente fase.

#### Diseño.

El diseño del sistema o software se centra en cuatro elementos del programa que son:

- La estructura de los datos.
- La arquitectura del software.
- El detalle procedimental.

- Caracterización de la interfaz.

En esta parte los requisitos capturados en la etapa anterior son traducidos a una representación del software antes de que comience la codificación.

### **Desarrollo**

En esta etapa el diseño debe de traducirse de una forma que la máquina entienda y procese. Es aquí donde se genera el código del software, es decir el desarrollo del proyecto.

### **Pruebas**

Una vez que se ha generado el código comienza la prueba del programa. Aquí es donde se centra en la lógica interna del sistema además de las funciones externas, así asegurando que los resultados que arroje el sistema sean los correctos.

### **Mantenimiento**

En esta etapa el software tiene cambios debido a que son encontrados errores o debido a que el cliente necesite ampliaciones funcionales o de diseño.

## **3.3. Análisis de requerimientos**

A continuación, se describen los requerimientos necesarios que se encontraron para la realización del sistema. Se analizaron los procesos que ellos hacen desde el momento en que se hace el producto que ellos venden hasta que lo entregan y lo venden al cliente para así poder crear las funciones que se necesitan para el sistema de información a desarrollar. A partir

de la investigación de los procesos en la microempresa se pudieron determinar y dividir los requerimientos:

- **Requerimientos funcionales:** Aquí se definen las características deseadas o comportamientos que el usuario espera que tenga el sistema, lo cual vendrán siendo las funciones del sistema.
- **Requerimientos no funcionales:** estos requerimientos definen restricciones y algunos atributos de usabilidad, escalabilidad, etc.

### **3.3.1. Definición de requerimientos funcionales y no funcionales del sistema web**

A continuación, se describen los requerimientos funcionales para el sistema web, aquí se determinan cuales se pueden manejar de manera directa y lo que el sistema debe de cumplir. Además, se describen los requerimientos no funcionales que debe de cumplir el sistema web tales como compatibilidad, comunicación con el servidor y clasificación de las interfaces gráficas del usuario. En este apartado solo se describen los requerimientos del sistema web ya que se describirán por separado el sistema web en comparación de la aplicación móvil ya que son dos programas distintos, pero con el mismo objetivo.

#### **Requerimientos funcionales por usuario**

En la Tabla 3.1 se describen los requerimientos funcionales por parte de un usuario al momento de querer ingresar y/o usar el sistema.

Tabla 3.1: Requerimientos funcionales por usuario.

<b>id</b>	<b>Requerimiento</b>	<b>Descripción</b>
1A	Inicio de sesión	El sistema debe de autenticar a los usuarios registrados mediante un usuario y contraseña.
2A	Identificación del usuario	El sistema debe permitir realizar tareas dependiendo del tipo de usuario que se encuentra registrado.

### **Requerimientos funcionales por el usuario administrador del sistema web**

En la Tabla 3.2 se muestran los requerimientos funcionales del usuario administrador que son los que debe de cumplir el sistema web para poder gestionar la información de una manera adecuada.

Tabla 3.2: Requerimientos funcionales por el usuario administrador del sistema web.

id	Requerimiento	Descripción
1B	Resumen de ventas y pedidos	El sistema debe de mostrar en la página de inicio un resumen estadístico de los movimientos como ventas y pedidos.
2B	Administración de clientes	El sistema debe de permitir al usuario registrar, modificar y eliminar clientes para poder gestionarlos de una manera organizada y eficiente.
3B	Administración de accesos	El sistema debe de dar de alta y darle permisos a usuarios para poder ingresar al sistema web.
4B	Administración de los pedidos	El sistema debe de mostrar los pedidos desde los más recientes y el usuario podrá registrar un pedido de manera manual, así como modificar y eliminar un pedido.
5B	Administración de usuario android	El sistema debe de dar de alta usuarios para la aplicación móvil, así como registrarlos, modificarlos y eliminarlos.
6B	Configuración de perfil de usuario	El sistema debe de permitir al usuario modificar su información como nombre de usuario y contraseña así como su foto de perfil en el sistema.
7B	Administración de productos	El sistema debe de permitir al usuario poder registrar, modificar y eliminar productos de su sistema.
8B	Generación de reportes	El sistema debe permitir al usuario consultar estadísticas y generar reportes bajo distintos parámetros de entrada y distintos parámetros de fecha Se debe permitir: - Reporte de ventas. - Reporte de pedidos.

### Requerimientos funcionales por el usuario empleado

En la Tabla 3.3 se muestran los requerimientos funcionales de un usuario tipo empleado donde se encarga únicamente de la ventas y gestión de pedidos.

Tabla 3.3: Requerimientos funcionales por el usuario empleado.

id	Requerimiento	Descripción
1C	Gestión de ventas localmente	El sistema debe permitir al usuario empleado seleccionar los productos para agregarlos a la lista de compras del cliente, para posteriormente registrarlos y realizar un cálculo del total a pagar por el cliente y el tipo de cambio que debe de dar.
2C	Gestión de pedidos	El sistema debe de mostrar al usuario empleado los pedidos realizados mediante la aplicación móvil, en el orden que fueron recibidas, mostrando información detallada de cuando se envió y a qué hora, así como el tipo de producto y la cantidad requerida por el cliente.

### Requerimientos no funcionales del sistema web

En la Tabla 3.4 se muestran los requerimientos no funcionales del sistema web, como las compatibilidades donde el sistema debiera ejecutarse.

Tabla 3.4: Requerimientos no funcionales del sistema web

id	Requerimiento	Descripcion
1D	Interfaz gráfica por tipo de usuario	El sistema debe mostrar una interfaz dependiendo del usuario que ingrese al sistema web.
2D	Registro de informacion	El sistema debe permitir registrar ventas y pedidos, así como otros datos importantes para la microempresa como clientes, productos, usuarios, reportes.
3D	Comunicacion con una base de datos	El sistema debe comunicarse con una base de datos en un servidor para la gestión y manipulación de la información en el sistema web.
4D	Compatibilidad	El sistema debe ser compatible y con un correcto funcionamiento con sistemas operativos; -Windows y -Linux Y en navegadores de internet como: -Google Chrome y -Mozilla Firefox

### 3.3.2. Definición de requerimientos funcionales y no funcionales de la aplicación móvil

A continuación, se describirán los requerimientos funcionales de lado de la aplicación móvil que es más dirigida hacia los clientes que la microempresa tortillera atiende, además también se describirán los requerimientos no funcionales de la aplicación.

#### Requerimientos funcionales del cliente

En la Tabla 3.5 se muestran los requerimientos funcionales de la aplicación móvil que deberá mostrar al cliente que necesite realizar un pedido con ciertos parámetros o quiera visualizarlo.

Tabla 3.5: Requerimientos funcionales de la aplicación móvil.

id	Requerimiento	Descripcion
1E	Inicio de sesión	La aplicación móvil debe permitir ingresar a los usuarios mediante un usuario y contraseña para autenticarlos.
2E	Realización de pedidos	La aplicación móvil debe permitir a los clientes registrados hacer pedidos enviando parámetros como tipo de producto, cantidad, precio y total.
3E	Visualización de pedidos	La aplicación móvil debe permitir al usuario visualizar su lista de pedidos y el estado del envío en el que se encuentra, el estado toma parámetros como preparando, en camino y entregado, este cambia según su estado de envío.

### Requerimientos no funcionales de la aplicación móvil

En la Tabla 3.6 se muestran los requerimientos no funcionales de la aplicación móvil describiendo requerimientos no funcionales como inicio de sesión, compatibilidad y comunicación con la base de datos.

Tabla 3.6: Requerimientos no funcionales de la aplicación móvil.

id	Requerimiento	Descripcion
1E	Comunicación con la base de datos	La aplicación debe comunicarse con una base de datos en un servidor para la realización y visualización de los pedidos.
2E	Compatibilidad	Compatibilidad con telefono con sistemas operativos android.

### 3.3.3. Herramientas

En esta seccion se describira cada una de las herramientas que se necesitara para la realizacion de este proyecto, tanto para el sistema web como la aplicacion movil algunos son:

editores de texto, servidores locales, servidores web, bases de datos, gestores de bases de datos, etc.

#### **Visual Studio Code**

Este editor se seleccionó por la familiaridad que se tiene con él y además de las funcionalidades tan útiles que este tiene. Además, que ofrece maneras de programar más rápido y eficazmente, así adaptándose a las necesidades que se requieran al momento de realizar un proyecto.

#### **App inventor**

Esta plataforma de la MIT es muy útil ya que te permite crear aplicaciones en poco tiempo, es decir las aplicaciones son mas rápido de hacer y no llevan mucho tiempo, es por ello por lo que se selecciono esta herramienta, por la rapidez en que se crean las aplicaciones. Además, se puede hacer todo tipo de aplicación, de la misma manera como si se desarrollara en Android Studio que es más complejo.

#### **XAMPP**

Se utilizo este paquete de software libre para la realización de las pruebas locales del sistema web y de la aplicación móvil. Este contiene lo necesario ya que hace la función de un servidor, contiene un gestor de base de datos y se puede compilar código de lado del servidor que es php.

### **x10Hosting**

Este se utilizó para las pruebas ya en un servidor web online, es gratuito y ofrece las todo lo que necesitamos para correr nuestro sistema web, así como hacer conexiones a la base de datos, dominio gratis, espacio de almacenamiento entre otras cosas. Además de que es muy flexible también tiene un alto rendimiento al momento de ofrecernos su servicio.

### **MySQL**

Este sistema de gestión de base de datos es que se utiliza para la administración y el almacenamiento de la información de un proyecto de software. En este caso se utilizará en nuestro sistema para almacenar la información del negocio.

### **phpMyAdmin**

Esta herramienta es útil para la administración de MySQL mediante la página web. Esto lo hace un poco más eficaz ya que viene incluido en la paquetería del servidor local XAMPP mencionado anteriormente.

## **3.4. Diseño**

### **3.4.1. Arquitectura del sistema**

A continuación, se explica la arquitectura del sistema en el cual se muestra de manera general una arquitectura de tres capas que es la más usual al momento de desarrollar software. Así mismo esta arquitectura de software consiste en un conjunto de abstracciones que forman el marco del software. Esta arquitectura se diseña después de la primera etapa de análisis de requerimientos.

En la Figura 3.2 se muestra la arquitectura del sistema con sus respectivos componentes.

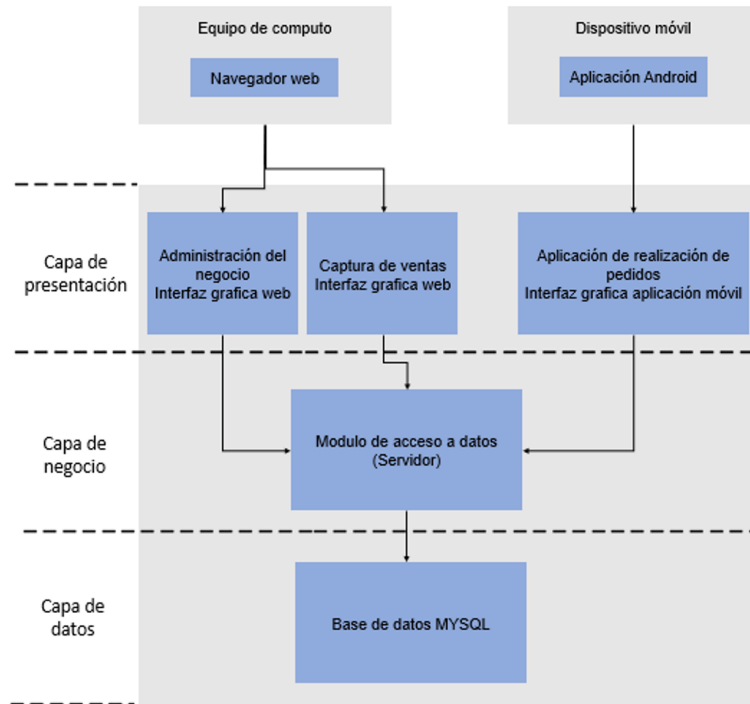


Figura 3.2: Arquitectura lógica del sistema.

En la Figura 3.2 se puede ver que contiene tres capas la cuales cada una de ellas tienen un funcionamiento tanto del sistema web como la aplicación móvil:

- Capa de presentación: aquí es donde se crean las interfaces de usuario, y su única función es pasarle las acciones que realice el usuario a la capa de negocio.
- Capa de negocio: en esta capa se gestiona la lógica de la aplicación es donde se dice que hacer con los datos obtenidos de la otra llamada capa de datos y a partir de esos datos realizar sus funciones.
- Capa de datos: esta es la tercera y última capa que se encarga de guardar y manipular

los datos, se gestiona todo lo relacionado a la base de datos, así como la creación, edición y borrado de los datos que contenga.

Esta arquitectura sirve para visualizar de una manera mas clara y simple de cómo se estructura el sistema, desde como se muestra la información al usuario y como se procesan las funciones dadas las acciones realizadas por el usuario para posteriormente obtener la información de nuestra base de datos.

### **3.4.2. Diseño de la base de datos**

Al momento de diseñar y/o realizar una base de datos es importante analizar las variables que se pueden identificar y que se incluyen dentro del proceso administrativo que debe llevar esta microempresa. A continuación, se muestran los resultados de un análisis para identificar las posibles entidades.

Tabla 3.7: Entidades de la base de datos de forma conceptual.

Entidad	Descripción
Productos	Representa elementos tipo alimento terminados para su venta que la microempresa ofrece a sus clientes.
Pedidos	Representa un elemento que notifica al negocio para abastecer o entregar más producto a un cliente y se relaciona con los productos y posteriormente con un cliente dentro de la base de datos.
Clientes	Representa un actor donde su rol es la compra del producto a la microempresa, existiendo clientes tipos de clientes que son: -Clientes comerciales y -Clientes consumidores
Ventas	Representa la información que registra un pago de un producto vendido o pedido realizado y entregado.
Empleados	Representa un actor donde puede haber diferente tipo de rol como: -Empleado de mostrador -Empleado de servicio -Empleado repartidor -Administrador
Accesos	Representa la entidad donde se almacenan y gestionan los usuarios que podrán ingresar a la interfaz administrativa del sistema web.
Usuarios_Android	Representa la entidad donde se almacenan y gestionan los usuarios que podrán ingresar a la aplicación móvil siempre y cuando estén registrados.
Registros (Bitacora)	Representa un historial de todos los movimientos que se estaran realizando en la base de datos.

Una vez obtenidas las entidades requeridas, se continuó a desglosarlas más a detalles obteniendo los atributos de cada entidad, por ejemplo, para la entidad de producto los atributos que debe de contener son id, nombre, descripción y precio. Cabe mencionar que, para cada entidad esta debe de contener un identificador o clave primaria.

Tabla 3.8: Diccionario de la base de datos respecto a las entidades

Entidad	Atributos	Descripción	Tipo de dato
Productos	id_producto	Identifica el producto	Entero
	nombre	Nombre del producto	Varchar 120
	descripcion	Descripcion textual de producto	Varchar 120
	precio	Precio del producto	Entero
Pedidos	id_pedido	Identifica el pedido	Entero
	id_cliente	Identifica el cliente	Entero
	fecha	Fecha del pedido	Fecha
	estado	Representa el estado en el que se encuentra el pedido	Varchar 10
	id_producto	Identifica el producto	Entero
	cantidad	Representa el total de producto a vender	Entero
	precio	Precio del producto	Entero
	total	Representa el precio final a pagar por el cliente	Entero
Clientes	id_cliente	Identifica el cliente	Entero
	nombre	Nombre del cliente	Varchar 90
	tipo	Tipo de cliente	Varchar 80
	direccion	Ubicación del cliente	Varchar 120
	codigo_postal	Codigo postal del cliente	Varchar 120
	correo	Dirección de correo electrónico del cliente	Varchar 120
	telefono	Numero de teléfono del cliente	Entero

	ciudad	Ciudad donde se ubica el cliente	Varchar 120
Ventas	id_venta	Identifica la venta	Entero
	fecha	Fecha de la venta	Fecha
	id_producto	Identifica el producto	Entero
	cantidad	Cantidad del producto vendido	Entero
	estado	Representa el estado en el que se encuentra la venta	Varchar 100
Empleados	id_empleado	Identifica el empleado	Entero
	nombre	Nombre del empleado	Varchar 110
	calle	Calle donde vive el empleado	Varchar 50
	colonia	Colonia donde vive el empleado	Varchar 50
	num_ext	Numero exterior del domicilio	Entero
	telefono	Numero de contacto del empleado	Entero
	edad	Edad del empleado	Entero
	puesto	Responsabilidad o puesto del empleado	Varchar 80
Accesos	id_acceso	Identifica el acceso	Entero
	id_empleado	Identifica el empleado	Entero
	username	Nombre de usuario	Varchar 50
	password	Contraseña para ingresar	Varchar 80
	tipo	Permisos para ingresar al sistema	Varchar 80
	estado	Representa si el usuario esta activo o no	Varchar 50
Uuarios_Android	id_usuario	Identifica al usuario	Entero
	usuario	Nombre de usuario para ingresar al sistema	Varchar 50
	pwd	Contraseña del usuario para ingresar al sistema	Varchar 80
	estado	Representa si el usuario esta activo o no	Varchar 20

	id_cliente	Identifica al cliente	Entero
Registros	id_registro	Identifica el registro	Entero
	fecha	Fecha del movimiento en la base de datos	Fecha
	motivo	Representa el movimiento que se hizo ya sea unaventa o pedido o modificacion del algun producto, cliente, etc.	Varchar 70
	id_usuario	Identifica el usuario	Entero

En la Tabla 3.7 se obtuvo el diccionario de datos que de manera que nos ayuda a entender con más precisión los datos que se manejan en el sistema administrativo. Así mismo este análisis de datos nos ayuda a complementar el diagrama de entidad-relación que se muestra en la Figura 3.3 y así poder comprender la información en la base de datos de una manera más eficiente. El gestor que se utilizó para la base de datos fue phpmyadmin, utilizando una comunicación a través de un servidor apache haciendo uso del lenguaje más utilizado para trabajar del lado del servidor que es PHP.

A continuación, después de haber analizado y obtenido el diccionario de datos, en la Figura 3.3 se muestra el diseño lógico de la base de datos mediante un diagrama entidad-relación.

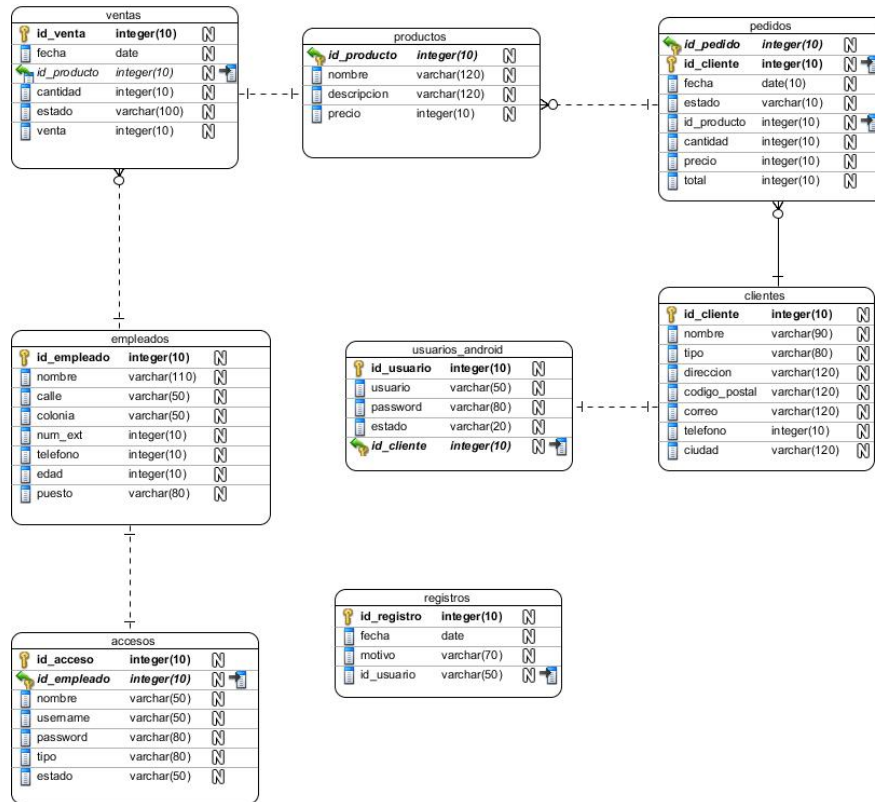


Figura 3.3: Esquema entidad-relación diseñado para el proyecto.

En la Figura 3.3 del diagrama relacional se puede visualizar como la tabla pedidos está hecha para la gestión de los pedidos los cuales serán enviados por los clientes comerciales a través de la aplicación móvil, aquí la aplicación es capaz de enviar el pedido obteniendo la información del cliente y la del producto es por ello por lo que se nota, como la tabla de pedidos está relacionada con la tabla de clientes y productos. La tabla de ventas será la encargada de registrar todas las compras que se realicen por parte de los clientes consumidores y por los pedidos que realicen por medio de la aplicación en Android.

Se puede visualizar que se tienen dos tablas que registran usuarios, una que es la de accesos en donde está definida para permitir a los empleados que estén registrados con un usuario y contraseña entrar al sistema administrativo web. Por otro lado, la tabla de usuarios Android

está diseñada y/o definida para permitir a los clientes registrados entrar en la aplicación móvil de pedidos. Así concluyendo que una es para los usuarios del sistema web y la otra para los usuarios de la aplicación móvil.

En cuanto a la tabla de registros, su funcionamiento es hacer un reporte tipo bitácora de los movimientos que se hagan en la base de datos por ejemplos saber quién agregó, modificó o eliminó un registro de las otras tablas de la base de datos.

### 3.4.3. Casos de uso

La finalidad de un caso de uso es describir la manera en que se usará un sistema: describir sus finalidades esenciales. La finalidad de los diagramas de casos de uso es captar en forma visual las finalidades esenciales [22].

El sistema se compone de cuatro módulos divididos en 2 aplicaciones, en donde los clientes del restaurante son los actores de la aplicación móvil incluyendo un actor que tendrá el rol de consultar los pedidos dentro de la aplicación móvil y el personal del negocio son los empleados y administradores del sistema web, a continuación se muestra dicha estructura:

- Sistema web
  - Módulo administrativo
  - Módulo operativo
- Aplicación móvil
  - Módulo administrativo
  - Módulo de servicio

En la Figura 3.4 se muestra el caso de uso del sistema web con sus respectivos actores y funciones.

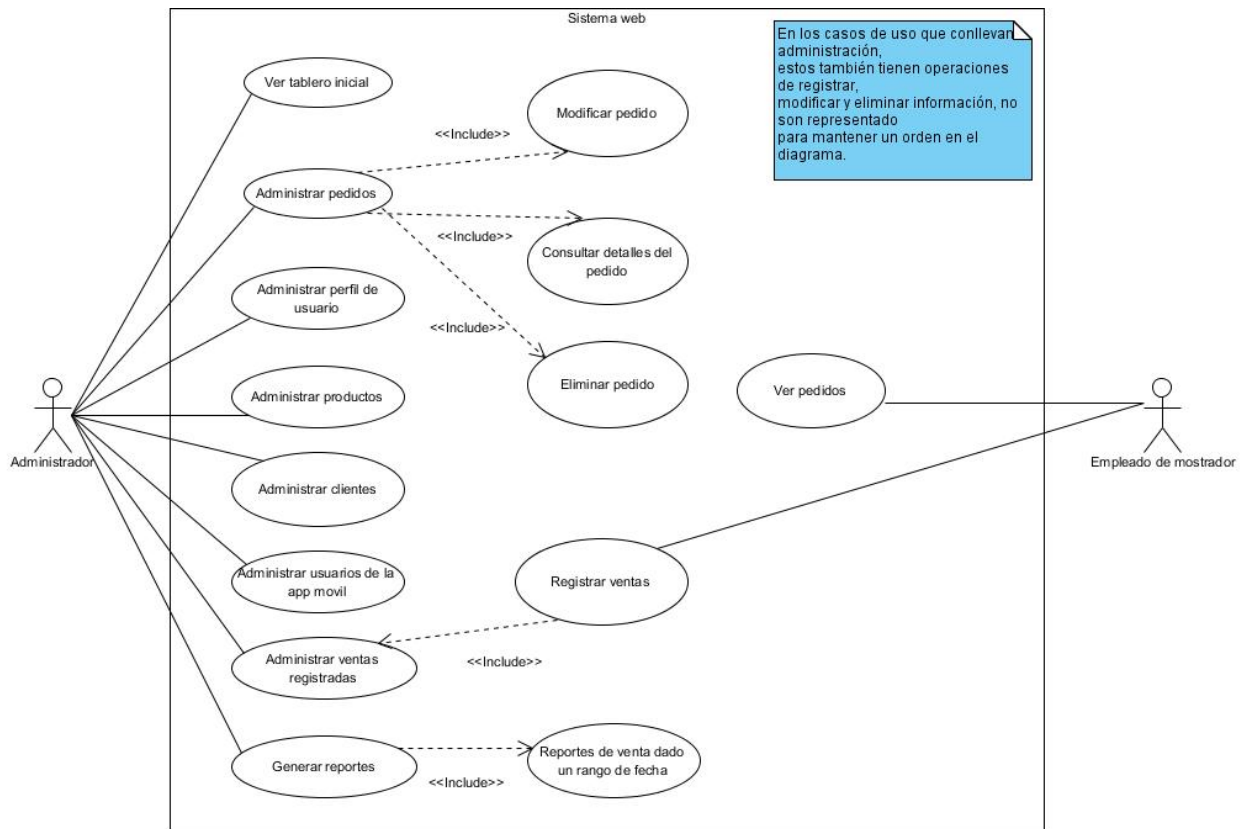


Figura 3.4: Diagrama de caso de uso del sistema web.

### Descripción de caso de uso por parte del actor administrador

- **Ver tablero inicial:** Interfaz gráfica donde su función es la de mostrar un resumen general de las ventas y pedidos que se han realizado en el negocio.
- **Administrar pedidos:** Interfaz gráfica donde permite la creación, modificación y eliminación de pedidos.
- **Administrar perfil de usuario:** Interfaz gráfica que permite la modificación del usuario actual en el sistema.
- **Administrar productos:** Interfaz gráfica que permite la creación, modificación y

eliminación de productos del negocio.

- **Administrar clientes:** Interfaz gráfica que permite la creación, modificación y eliminación de los clientes comerciales del negocio.
- **Administrar usuarios de la app móvil:** Interfaz gráfica que permite la creación, modificación y eliminación de los usuarios que interactúan con la aplicación móvil del negocio.
- **Administrar ventas registradas:** Interfaz gráfica donde su función es la creación, modificación y eliminación de las ventas registradas.
- **Generar reportes:** Interfaz gráfica que permite la generación de reporte como ventas dentro de un determinado rango de fecha.

#### Descripción de caso de uso por parte del actor empleado

- **Ver pedidos:** Interfaz que se encarga únicamente de visualizar la cola de pedidos.
- **Registrar ventas:** Interfaz que se encarga de registrar en la base de datos las ventas que se realicen en el mostrador del local.

Por otro lado, se puede visualizar en la Figura 3.5 el caso de uso de la aplicación móvil cuyos respectivos actores son los clientes y un empleado repartidor encargado de poder visualizar los pedidos y cambiar su estado de entrega.

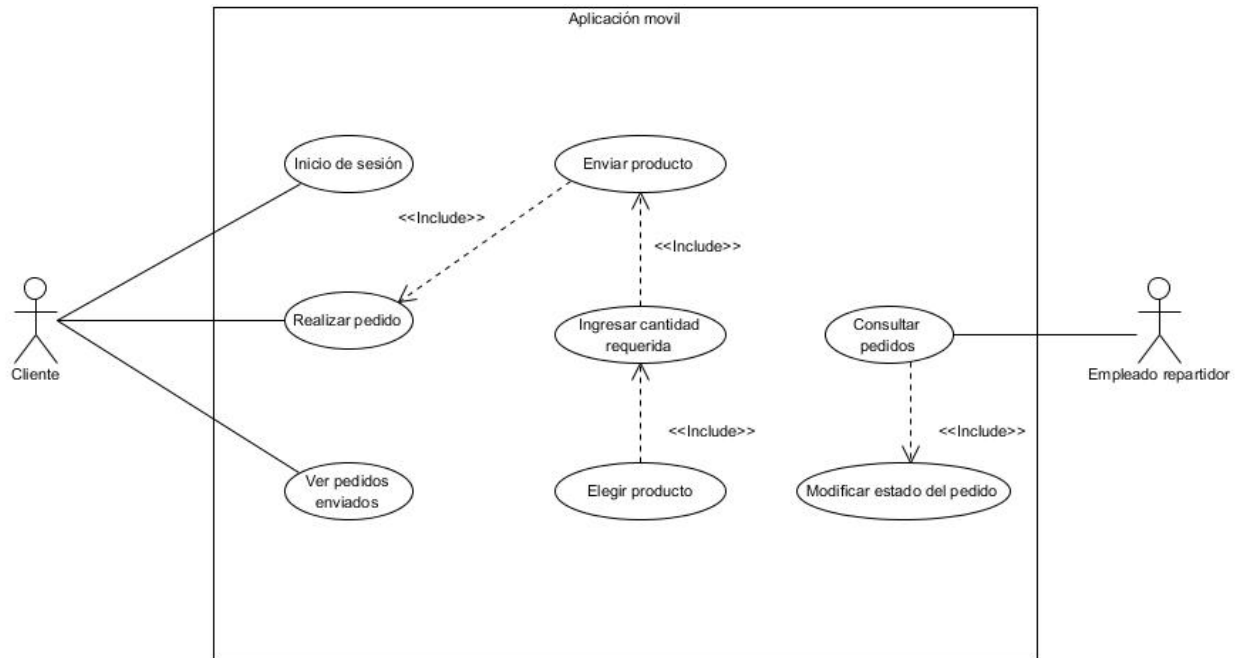


Figura 3.5: Diagrama de caso de uso de la aplicación móvil.

#### Descripción de caso de uso por parte del actor cliente

- **Inicio de sesión:** Interfaz que permite al actor cliente ingresar a la aplicación móvil.
- **Realizar pedido:** Interfaz que permite al usuario comenzar un pedido, así mismo se deberá de elegir un producto con su respectiva cantidad para poder enviar el pedido.
- **Ver pedidos enviados:** Interfaz que permite visualizar la lista de pedidos realizados y enviados.

#### Descripción de caso de uso por parte del actor empleado.

- **Consultar pedidos:** Interfaz que permitirá al actor empleado visualizar la lista de pedidos y a su vez cambiar el estado de entrega del pedido.

### 3.4.4. Diagrama de clases

Los diagramas de clases se diseñan para mostrar todas las piezas de su solución, así mismo cuáles piezas se relacionan con ésta o se usan como partes de totalidades nuevas y deben transmitir un sentido del sistema que se estructurará en reposo [23].

Como se puede observar en los siguientes diagramas de la Figura 3.6 y la Figura 3.7 se muestran las clases requeridas tanto para el sistema web como para la aplicación móvil.

#### Diagrama de clases del sistema web

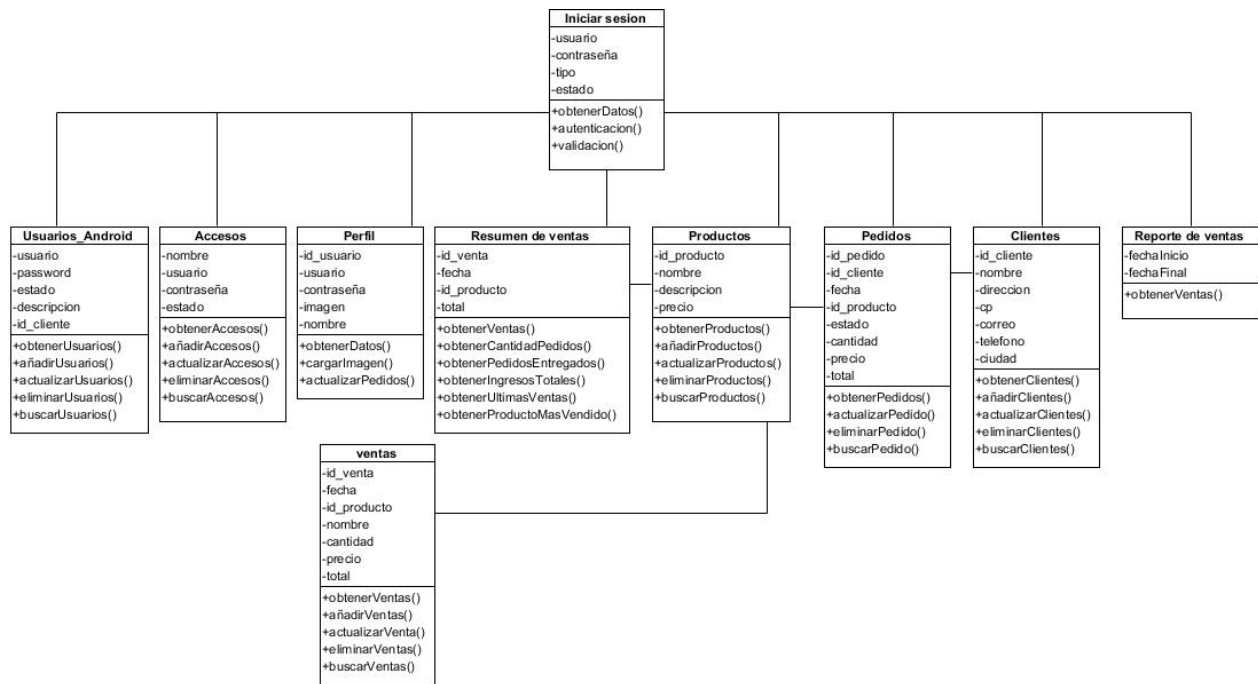


Figura 3.6: Diagrama de clases del sistema web.

## Diagrama de clases de la aplicación móvil

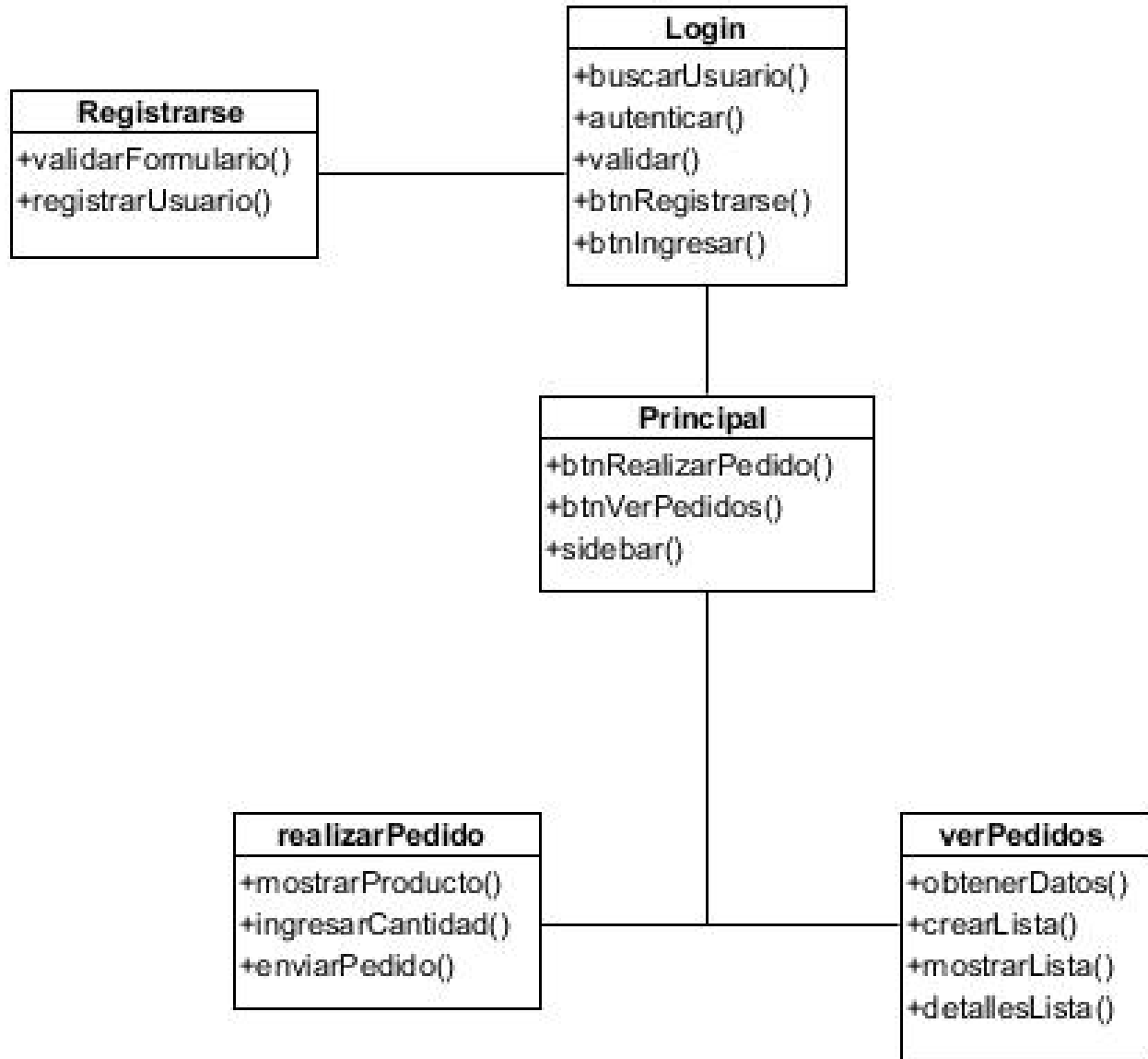


Figura 3.7: Diagrama de clases de la aplicación móvil.

### 3.4.5. Diagrama de secuencia de los pedidos en la aplicación móvil

En la Figura 3.8 se puede visualizar el proceso para poder realizar pedidos en la aplicación Android. Primero el cliente debe de autenticarse de modo que la aplicación manda una solicitud al servidor, después esta consulta en la base de datos si se encuentra registrado o no en la base de datos, entonces devuelve true si se encontraron registros o de lo contrario false si no lo hay, así mismo si es true la petición, el servidor da acceso al cliente a la página principal de lo contrario lo redirecciona al inicio de sesión de la aplicación.

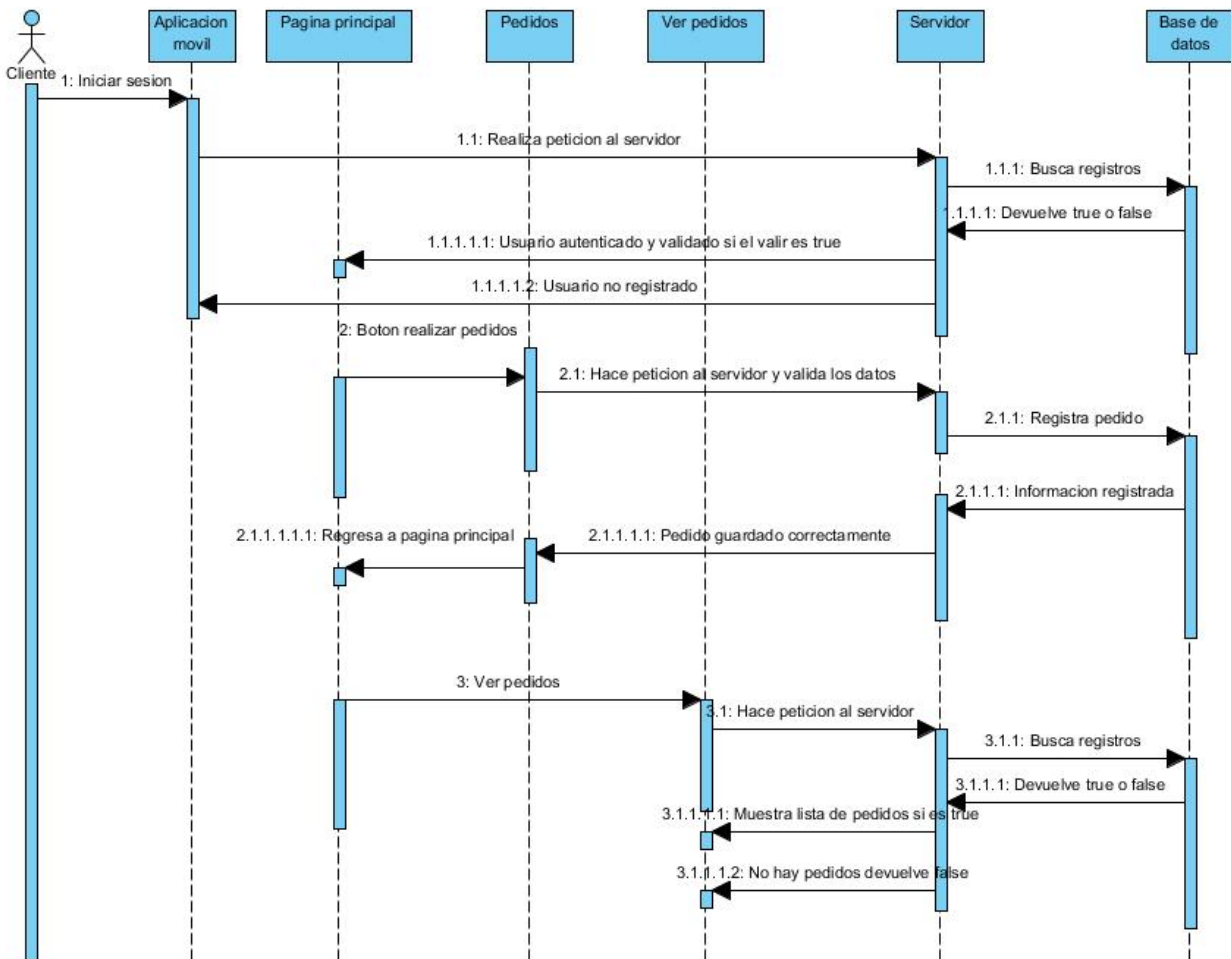


Figura 3.8: Diagrama de secuencia del proceso de pedidos en la aplicación móvil.

## 3.5. Desarrollo

En esta etapa se implementarán y se codificarán las funciones e interfaces para sistema web y la aplicación móvil. Se mostrarán las características que cada una conlleva, además de ver cual es su objetivo y cuál es su funcionamiento. Por parte del sistema web se puede ver cómo funciona cada uno de sus módulos, además de ver como hace las peticiones al servidor y como obtiene la información de la base de datos con código de programación. Por otro lado, la aplicación móvil realizará de igual forma peticiones al servidor ya sea para registrar o consultar datos.(El código fuente del sistema web se encuentra anexo en el apéndice A)

### 3.5.1. Sistema web

En el sistema web se divide en dos módulos uno que es el módulo administrativo y el segundo se le puede llamar módulo operativo. Estos se dividen debido a que llevan diferentes funciones dependiendo del tipo de usuario que lo utiliza por ejemplo el personal o empleado tipo administrador o empleado de mostrador.

#### Módulo administrativo

Este primer módulo está desarrollado para el personal administrativo de la microempresa tortillera, la cual contempla a los dueños y gerentes del negocio. Así que por medio de este módulo es posible realizar y administrar diferentes funciones como agregar, modificar y eliminar productos. De igual manera se permite el acceso y la manipulación de la información de otras interfaces que tiene este módulo administrativo como pueden ser(Ver Figura 3.9 y 3.10):

- Registro de clientes
- Reporte de ventas

- Consulta de pedidos
- Registros de accesos
- Consulta de ventas
- Edición de perfil

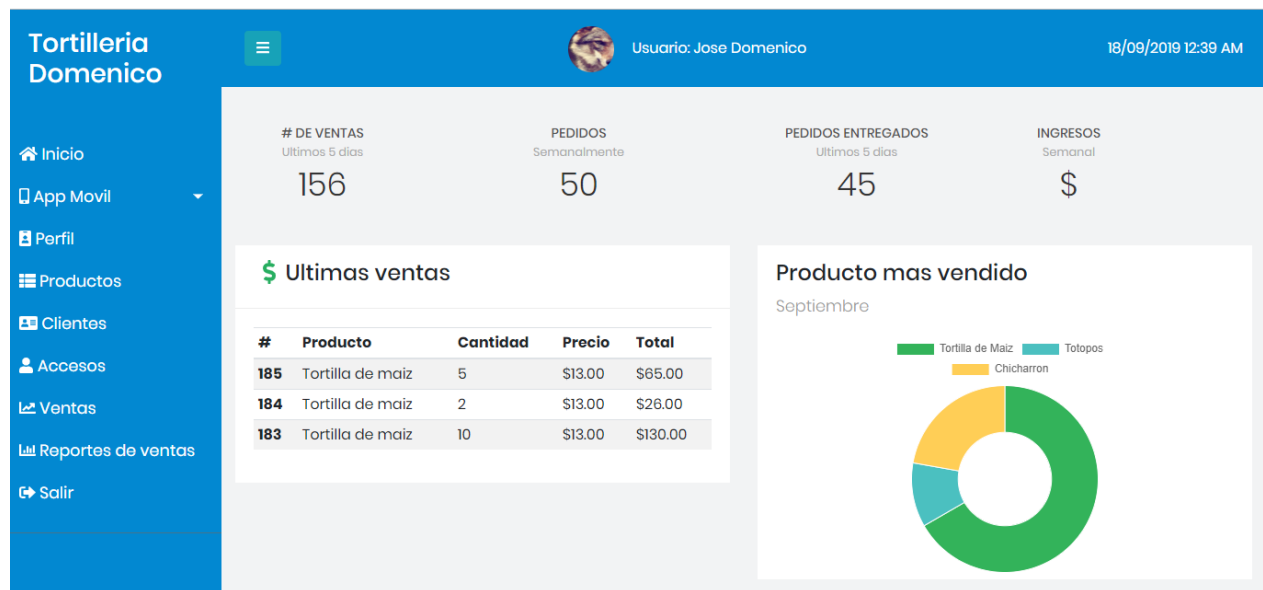


Figura 3.9: Interfaz del módulo administrativo.

### Módulo operativo

Este módulo es el encargado de las tareas con el personal de mostrador, es decir se encarga de ver la lista de pedidos y de gestionar las ventas que se realicen en el mostrador del negocio con los clientes consumidores. Algunas de las funciones de este módulo son:

- Registro de ventas
- Visualización de los pedidos

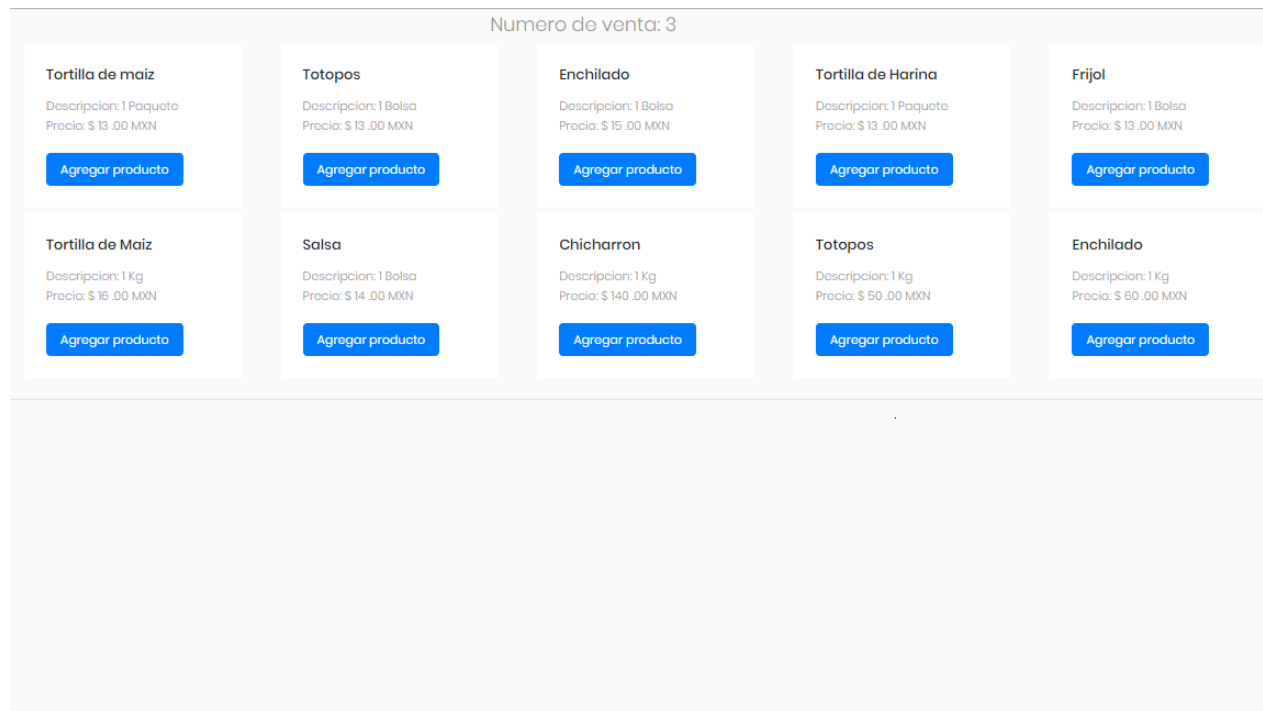


Figura 3.10: Interfaz del módulo operativo.

### 3.5.2. Funcionamiento del sistema web

En la Figura 3.11 se muestra una interfaz de inicio de sesión aquí dependiendo del tipo de usuario que intente ingresar será redireccionado al módulo administrativo o módulo operativo como se menciono anteriormente. Esta interfaz es la que separa los dos módulos mencionados con anterioridad de tal modo que se puede estar administrando por el dueño, gerente o encargado de este negocio y a su vez el empleado de mostrador puede estar registrando las ventas que se realicen en el establecimiento durante el día.

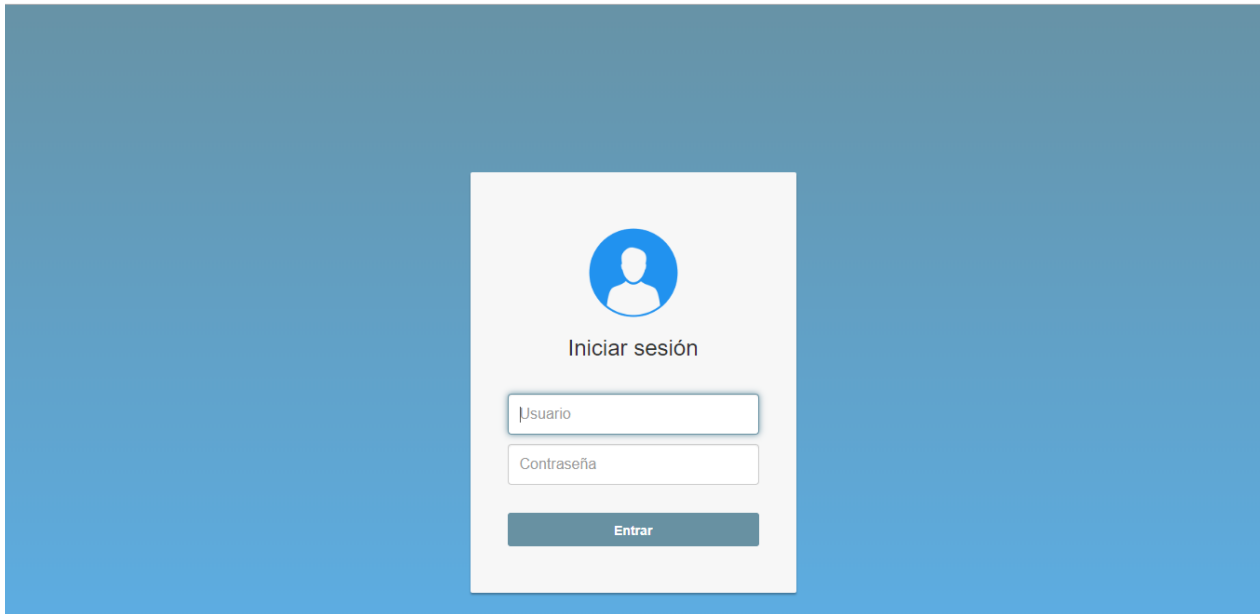
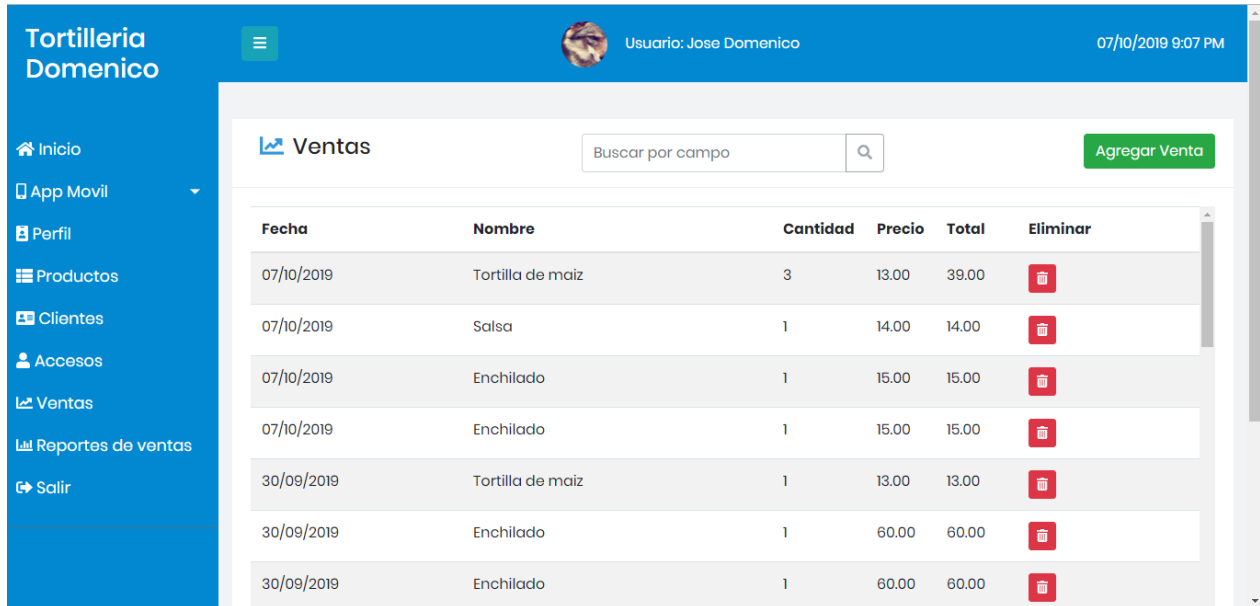


Figura 3.11: Interfaz de inicio de sesión del sistema web.

Una vez que el usuario ingresa al sistema este será redireccionado a la página de inicio como se mostraba anteriormente en la Figura 3.9 En esta parte exceptuando la página de inicio, la de perfil y la página de reportes de ventas, las demás paginas tienen una estructura general, en donde todas se caracterizan por mostrar los registros por medio de una tabla que se puede modificar en tiempo real, además de que cada una tiene un botón para agregar registros dependiendo de la página en que el usuario se encuentre ubicado, seguido de esto se tiene un buscador para encontrar el registro deseado más rápidamente. En la Figura 3.12 podemos visualizar los elementos que se relacionan.










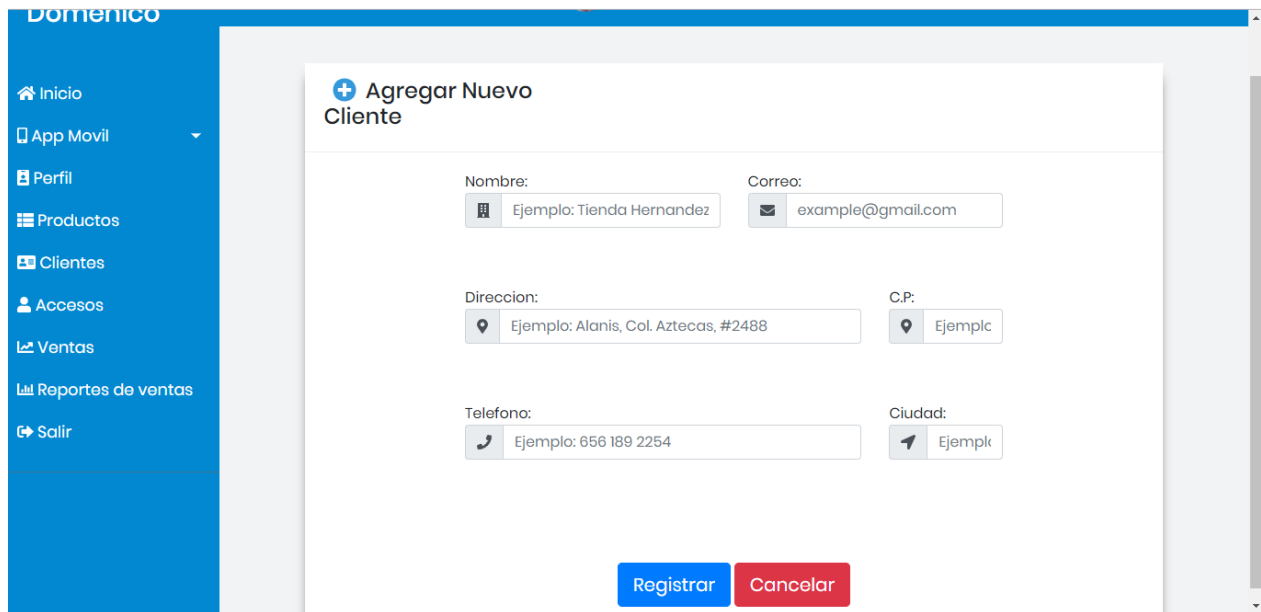
Fecha	Nombre	Cantidad	Precio	Total	Eliminar
07/10/2019	Tortilla de maiz	3	13.00	39.00	
07/10/2019	Salsa	1	14.00	14.00	
07/10/2019	Enchilado	1	15.00	15.00	
07/10/2019	Enchilado	1	15.00	15.00	
30/09/2019	Tortilla de maiz	1	13.00	13.00	
30/09/2019	Enchilado	1	60.00	60.00	
30/09/2019	Enchilado	1	60.00	60.00	

Figura 3.12: Estructura general de consulta de información.

Esta estructura aplica para las siguientes páginas:

- Pedidos
- Usuarios Android
- Productos
- Clientes
- Accesos
- Ventas

De manera que al momento que el usuario interactúe con diferentes módulos y con diferente información no se le complique la usabilidad del sistema.



The image shows a web application interface for adding a new client. On the left, there is a blue sidebar menu with the following items: Inicio, App Movil, Perfil, Productos, Clientes, Accesos, Ventas, Reportes de ventas, and Salir. The main content area is titled 'Agregar Nuevo Cliente' and contains a form with the following fields: Nombre (Ejemplo: Tienda Hernandez), Correo (example@gmail.com), Direccion (Ejemplo: Alanis, Col. Aztecas, #2488), C.P. (Ejemplc), Telefono (Ejemplo: 656 189 2254), and Ciudad (Ejempk). At the bottom of the form are two buttons: Registrar (blue) and Cancelar (red).

Figura 3.13: Estructura general de registro de información.

Al momento de agregar cualquier registro de información se cuenta con un formulario donde se deben llenar los campos solicitados por ejemplo en la Figura 3.13 se muestra los campos requeridos para agregar un registro en la interfaz de clientes. Los campos requeridos son el nombre del cliente, correo, dirección, código postal, teléfono y ciudad donde está ubicado. Todos los campos son obligatorios y en dado caso que el usuario deje un campo en blanco, el sistema le pedirá ingresar el dato faltante y así poder seguir con el registro.

### 3.5.3. Aplicación móvil

A continuación, en la Figura 3.14, se muestran y se explican cada una de las interfaces que se desarrollaron para la aplicación móvil. La aplicación comienza con un inicio de sesión donde el usuario con el rol de cliente deberá autenticarse con un usuario y una contraseña con la que se registró previamente y así de esta manera podrá ingresar a la aplicación. De caso contrario si el usuario no tiene una cuenta activa, deberá registrarse, es por ello por lo

que se tiene una interfaz para registrar usuarios en donde se deberán llenar los campos de manera obligatoria los cuales son:

- Nombre del negocio
- Numero de contacto
- Usuario
- Contraseña

Una vez obtenido una cuenta para ingresar, nos abrirá la página principal mostrándonos un mensaje de bienvenida y dos opciones para seleccionar donde una es para realizar pedidos y la otra para ver los pedidos que el usuario haya realizado. Así mismo estas opciones se encuentran en un menú lateral deslizable. Al momento de seleccionar realizar pedidos la aplicación abre la pantalla para comenzar y enviar un pedido, en donde se mostrará el producto y se tiene que seleccionar la cantidad de unidades a pedir. Una vez que la cantidad haya sido ingresada y se seleccione el botón de enviar, la aplicación aun nos confirmara que, si estamos listos para enviar el pedido, si la cantidad es correcta entonces se selecciona aceptar y el pedido será enviado (Los bloques de código de la aplicación móvil se encuentran anexados en el apéndice B).

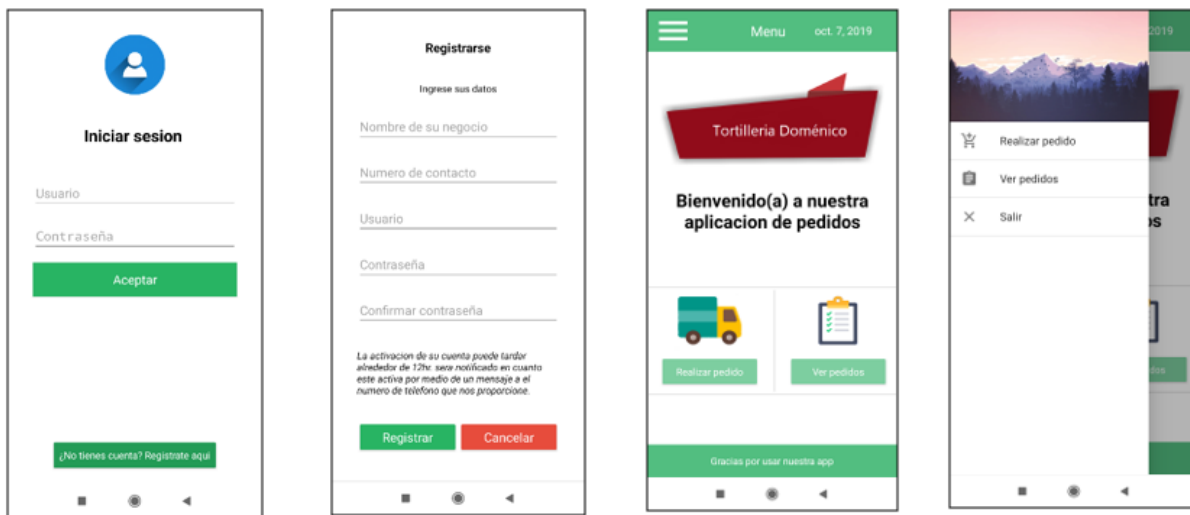


Figura 3.14: Capturas de pantalla de las interfaces de la aplicación móvil.

Por otro lado, en la pantalla de pedidos se muestran cada uno de los pedidos que el usuario a realizado y enviado, a primera vista este nos muestra la imagen del producto, nombre, un estado del proceso de envío del producto, además de la fecha y hora en que se realizó. Cada elemento de la lista de pedidos cuenta con una opción llamada “detalles” que nos permitirá ver todo el pedido completo, mostrándonos así datos como : fecha, hora, nombre del producto, estado, cantidad, precio y total como se muestra en la Figura 3.15.

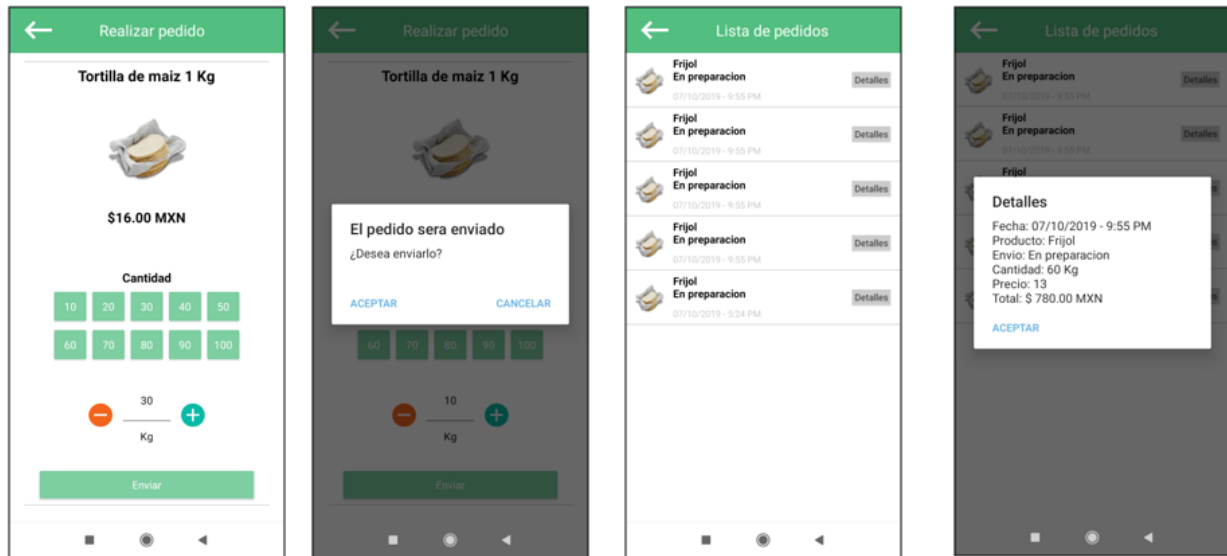


Figura 3.15: Capturas de pantalla de las interfaces de la aplicación móvil.

El siguiente código que se muestra en la Figura 3.16 se muestra la manera en que se hace la conexión a la base de datos local, este se utilizó tanto para el sistema web como para la aplicación móvil.

```
1  <?php
2  //Se establece la zona horario del servidor
3  date_default_timezone_set('America/Chihuahua');
4
5  //Se cre ala variable de conexion con sus parametros
6  $con= new mysqli("localhost","root","","sistemas2");
7
8  //Se valida la conexion
9  if($con->connect_errno){
10 |     echo"Error en la conexion";
11 |     exit;
12 }
```

Figura 3.16: Código de conexión a la base de datos.

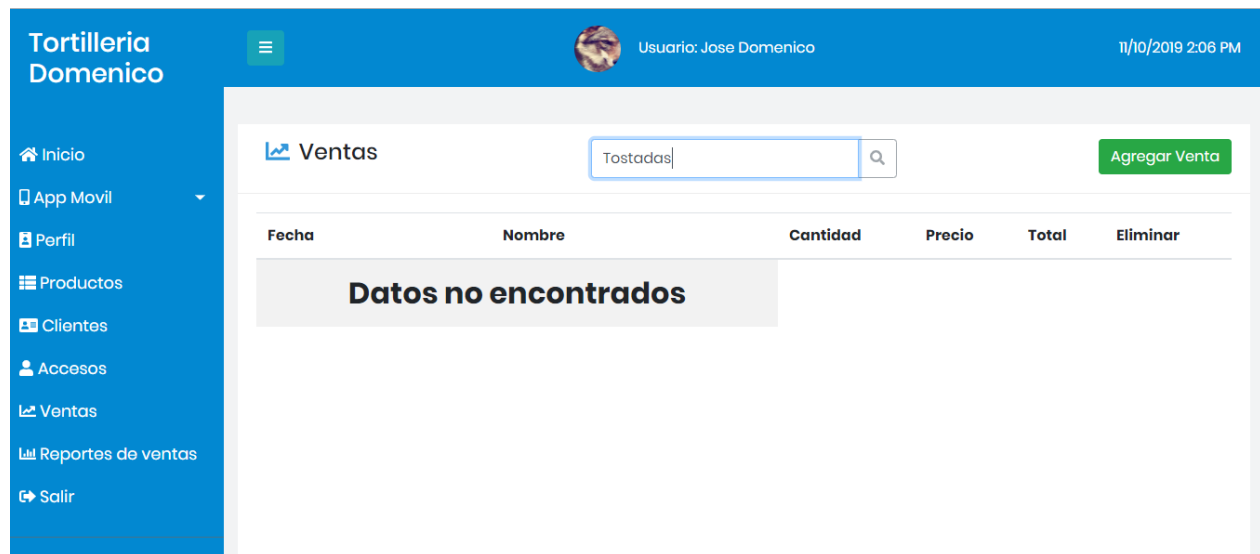
## 3.6. Pruebas

En esta fase del proyecto se realizaron validaciones del proyecto para verificar su funcionamiento tanto del sistema web como de la aplicación móvil. Estas pruebas tienen la finalidad de ver que todo el proyecto esté funcionando correctamente. La estrategia para seguir es la validación de:

- Cuadro de búsqueda
- Registro de información
- Eliminación de información
- Edición de información

### 3.6.1. Consulta y búsqueda de información

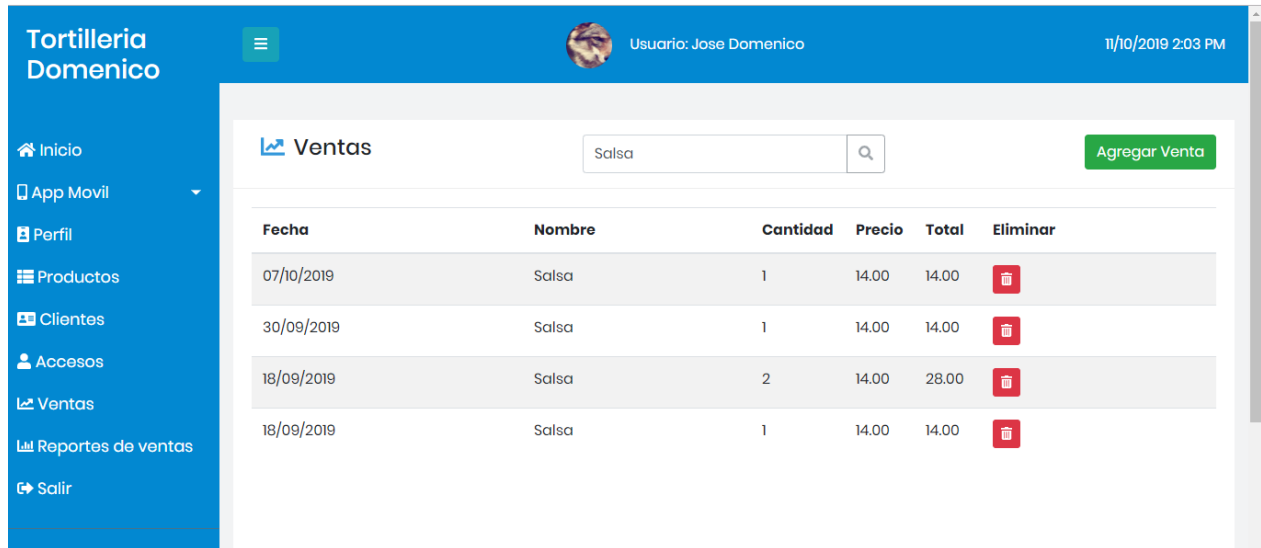
Toda la información que es mostrada en cada pagina del sistema web es obtenida de la base de datos haciendo una consulta cada vez que se ingrese a una pagina en específico. Esta contiene un cuadro de búsqueda que permite encontrar la información mas rápidamente en dado caso de que al ingresar un dato en el buscador y este no este en la base de datos, debe de salir un mensaje de no encontrado como se muestra en la Figura 3.17, de lo contrario al buscar por nombre del producto solo nos muestra los registros que tiene almacenados la base de datos como se muestra en la Figura 3.18 (El código fuente se encuentra en el apéndice A).



The screenshot displays the 'Ventas' (Sales) section of the 'Tortilleria Domenico' web application. The user is logged in as 'Jose Domenico' on 11/10/2019 at 2:06 PM. A search bar contains the text 'Tostadas' and a search icon. To the right of the search bar is a green button labeled 'Agregar Venta'. Below the search bar is a table with columns: 'Fecha', 'Nombre', 'Cantidad', 'Precio', 'Total', and 'Eliminar'. The table content is a grey box with the text 'Datos no encontrados' (Data not found).

Fecha	Nombre	Cantidad	Precio	Total	Eliminar
Datos no encontrados					

Figura 3.17: Datos no encontrados usando el buscador.



The screenshot shows the 'Tortilleria Domenico' web application interface. The top navigation bar is blue and contains the logo, a user profile icon, the text 'Usuario: Jose Domenico', and the date '11/10/2019 2:03 PM'. A left sidebar menu lists various options: Inicio, App Movil, Perfil, Productos, Clientes, Accesos, Ventas, Reportes de ventas, and Salir. The main content area is titled 'Ventas' and features a search bar with the text 'Salsa' and a magnifying glass icon. To the right of the search bar is a green button labeled 'Agregar Venta'. Below the search bar is a table with the following data:





Fecha	Nombre	Cantidad	Precio	Total	Eliminar
07/10/2019	Salsa	1	14.00	14.00	
30/09/2019	Salsa	1	14.00	14.00	
18/09/2019	Salsa	2	14.00	28.00	
18/09/2019	Salsa	1	14.00	14.00	

Figura 3.18: Buscando información usando el buscador.

### 3.6.2. Registro de información

Al momento de agregar información en alguna de las paginas debe de aparecer un mensaje en donde diga que la información fue agregada correctamente como se muestra en la Figura 3.19 en la página de ventas. También se valida que, en los campos de cantidades, precios y todos los relacionados a números solo se aceptara caracteres numéricos para darle más funcionalidad al sistema, además si algún campo se encuentra vacío el sistema debe ser capaz de no mandar la información hasta que todos los campos estén llenados correctamente.

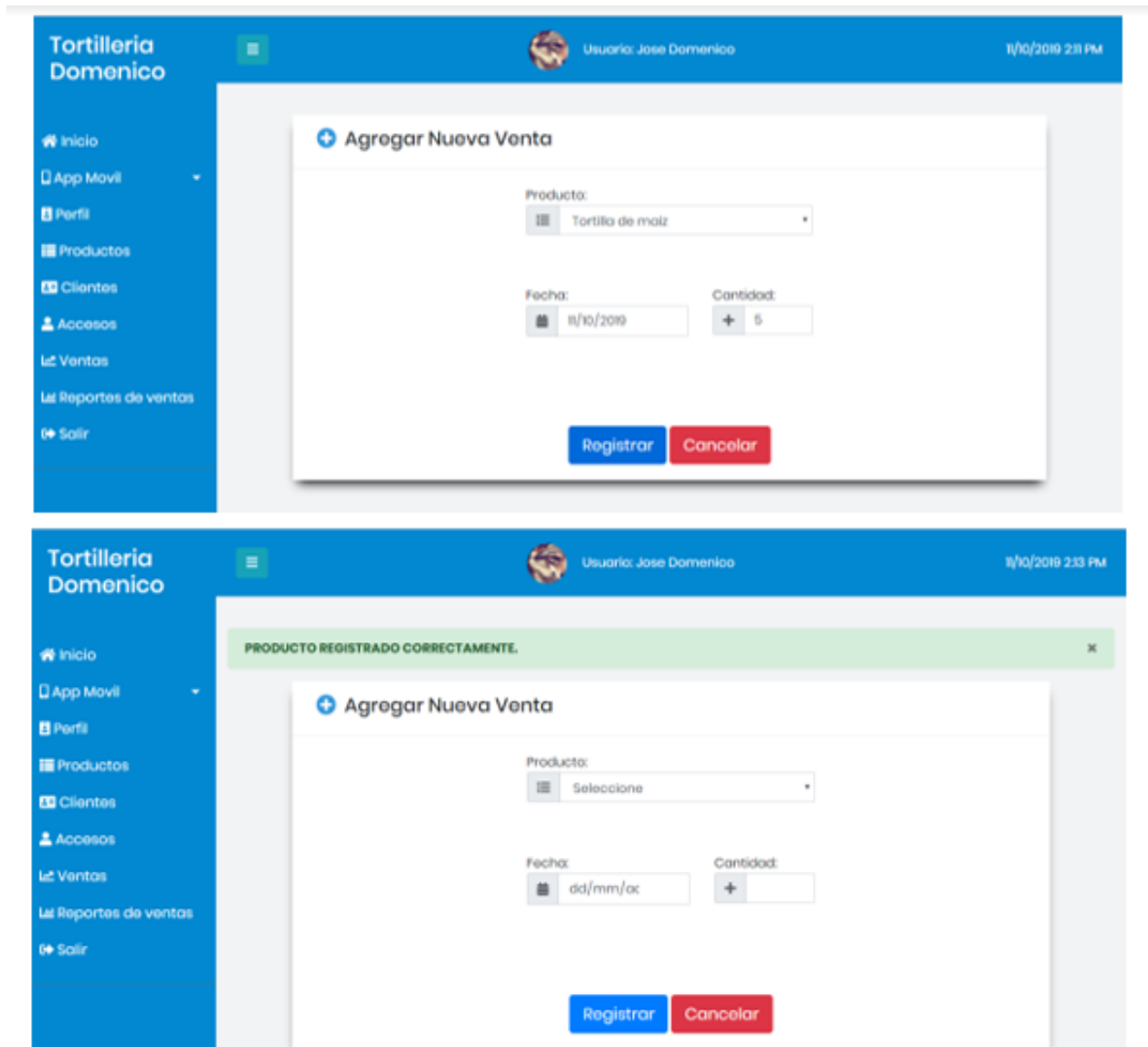


Figura 3.19: Validación de registro de la información.

### 3.6.3. Eliminación de un registro

En esta sección se valida el momento al querer eliminar un registro como se muestra en la Figura 3.20, en la sección de ventas el sistema es capaz de reconocer que un registro será

eliminado, entonces este manda un mensaje de confirmación el cual dice que, si está seguro de eliminar el mensaje, después al dar click en aceptar este se elimina de la base de datos.

The screenshot shows a web application interface for 'Tortilleria Domenico'. A confirmation dialog box is displayed over the main content, asking 'Quieres eliminar este registro?' (Do you want to delete this record?). The dialog has 'Aceptar' (Accept) and 'Cancelar' (Cancel) buttons. Below the dialog, a table titled 'Ventas' (Sales) is visible. The table has columns for 'Fecha' (Date), 'Nombre' (Name), 'Cantidad' (Quantity), 'Precio' (Price), 'Total', and 'Eliminar' (Delete). The table contains several rows of sales data.

Fecha	Nombre	Cantidad	Precio	Total	Eliminar
11/10/2019	Tortilla de maiz	5	13.00	65.00	[X]
07/10/2019	Tortilla de maiz	3	13.00	39.00	[X]
07/10/2019	Salsa	1	14.00	14.00	[X]
07/10/2019	Enchilado	1	15.00	15.00	[X]
07/10/2019	Enchilado	1	15.00	15.00	[X]
30/09/2019	Tortilla de maiz	1	13.00	13.00	[X]
30/09/2019	Enchilado	1	60.00	60.00	[X]

Figura 3.20: Validación de eliminación de un registro.

### 3.6.4. Edición de información

La información que contiene cada página dentro de la tabla se puede editar en tiempo real sin que nos mande a otro formulario. La información se actualiza al instante al momento de estar editándola, en si el sistema está en continua comunicación con el servidor actualizando la base de datos en la Figura 3.21 y 3.22 se muestra lo anteriormente mencionado.

Nombre	Direccion	C.P	Correo	Telefono	Ciudad	Eliminar
Empresa Metalez	CALLE AGUSTIN LARA NO. 69-B	68370	e.meta@gmail.com	6561547896	Cd Juarez	
Empresa Coca-Cola	CARRETERA A LOMA ALTA S/NLOMAS DEL PEDREGAL TUXTEPEC	12345	coca-cola@gmail.com	6561547896	Cd Juarez	
Empresa Jupiter	CALLE MATAMOROS NO.280COL. CENTRO TUXTEPEC	45789	jupitwer.ss@gmail.com	6561547896	Cd Juarez	

Figura 3.21: Validación de edición de un registro.

Nombre	Direccion	C.P	Correo	Telefono	Ciudad	Eliminar
Empresa Metalez	CALLE AGUSTIN LARA NO. 69-B	68370	e.meta@gmail.com	6561547896	Cd Juarez	
Empresa Coca-Cola	CARRETERA A LOMA ALTA S/NLOMAS DEL PEDREGAL TUXTEPEC	12345	coca-col	6561547896	Cd Juarez	
Empresa Jupiter	CALLE MATAMOROS NO.280COL. CENTRO TUXTEPEC	45789	jupitwer.ss@gmail.com	6561547896	Cd Juarez	

Figura 3.22: Validación de edición de un registro.

Las anteriores pruebas fueron para el sistema web y no se encontró ningún inconveniente respecto a fallas, las siguientes pruebas que se realizaron fueron para la aplicación móvil las cuales son:

- Validar autenticación por tipo de usuario
- Validar campos vacíos en el envío de información
- Validar enviar pedido

### 3.6.5. Validacion de autenticacion por tipo de usuario

Al momento de que un usuario necesite ingresar a la aplicación móvil, se debe de autenticar por un inicio de sesión donde dependiendo del tipo de usuario que intente entrar será redirigido a su correspondiente interfaz si el usuario es un cliente será redirigido a la pantalla de clientes y si es un usuario empleado tipo administrador entrara a ver las ordenes que se han hecho por parte de los clientes. En la Figura 3.23 se pueden visualizar ambas pantallas.

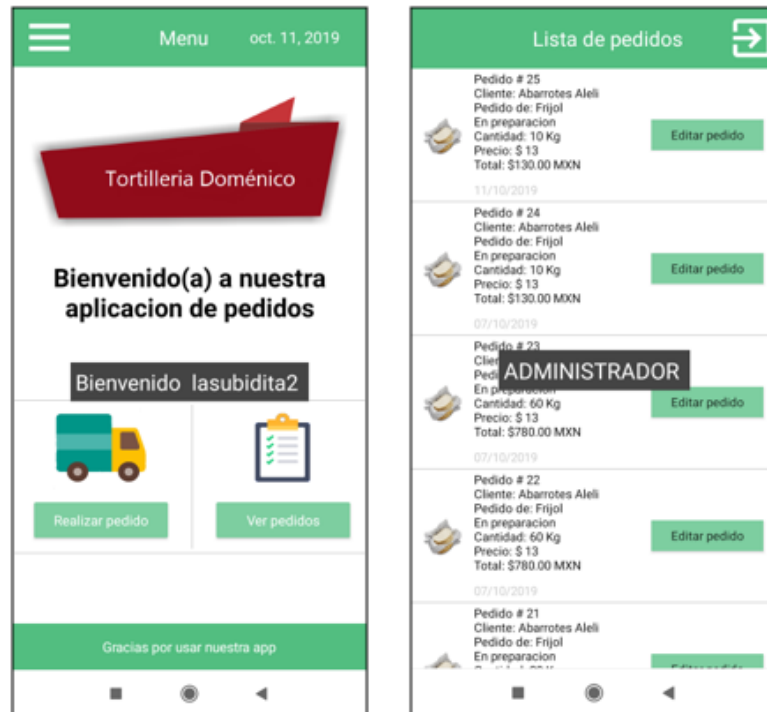


Figura 3.23: Validación de autenticación de usuarios.

### 3.6.6. Validación de campos vacíos

En esta sección se valida que todos los campos sean rellenos con información y esto es en todos los campos de cualquier pagina que necesite completarse, de lo contrario la aplicación mostrara un mensaje donde diga que faltan campos vacíos por llenarse y el formulario no se enviara hasta que estén todos los campos completos. El mensaje de alerta se muestra en la Figura 3.24 (Los bloques de código de la aplicación móvil se encuentran anexados en el apéndice B).

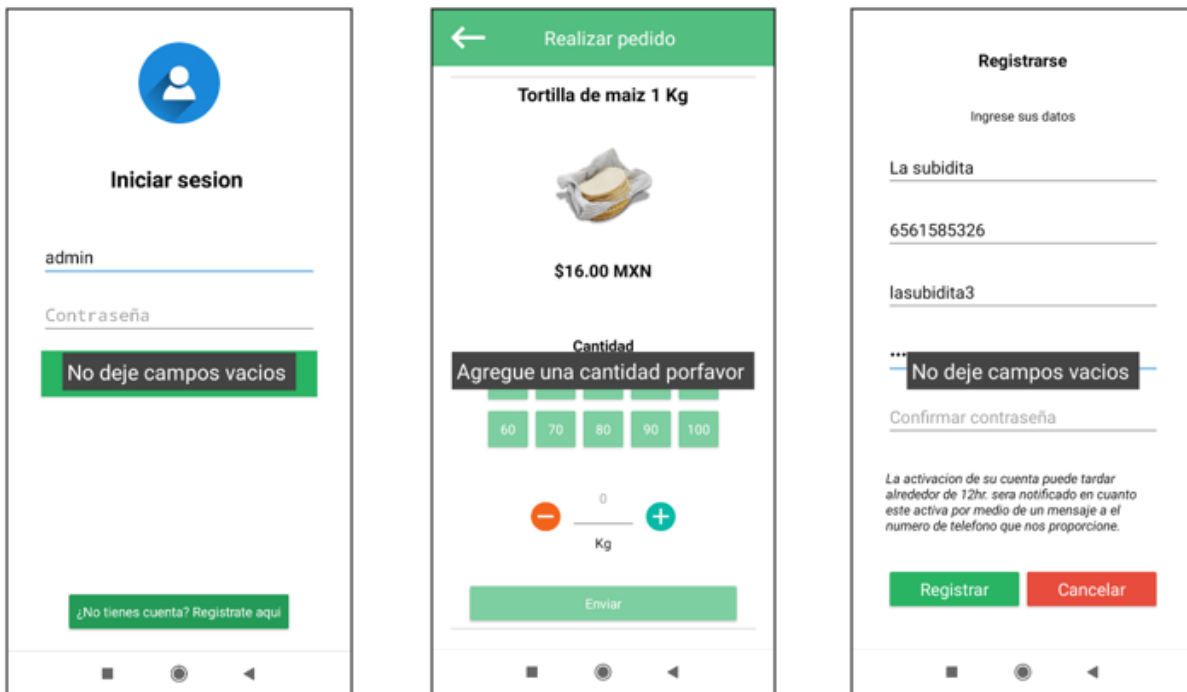


Figura 3.24: Validación de campos vacíos.

### 3.6.7. Validacion al enviar pedido

Al momento en que el usuario este enviando un pedido cuando de clic en el botón de enviar la aplicación mostrará una alerta de confirmación en donde se debe confirmar el envío del pedido, en cuanto el usuario pulse aceptar el pedido será enviado y lo redirigirá a la pantalla principal. En la Figura 3.25 se muestra lo anteriormente mencionado.

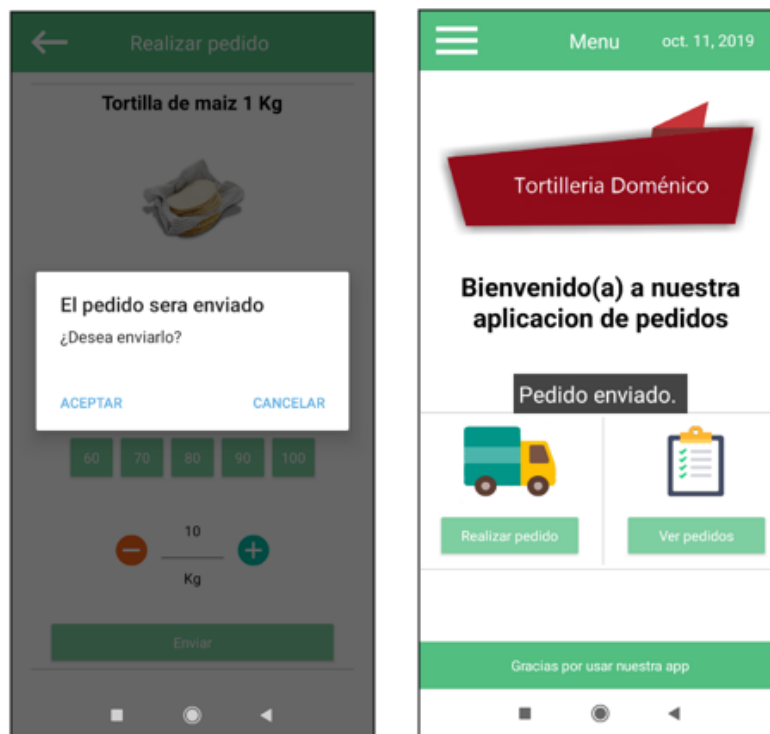


Figura 3.25: Validación de envío de pedidos.

# Capítulo 4

## Resultados y Discusiones

En el contenido de este capítulo se presentan los resultados obtenidos durante el proyecto, además se mostrarán detalladamente las encuestas que se realizaron a los clientes del negocio que están usando la aplicación móvil, también por el lado del negocio se realizó una encuesta a los empleados y al mismo administrador o encargado del negocio para ver la satisfacción del usuario. Estas encuestas se realizaron con la finalidad de obtener resultados del sistema de acuerdo con su diseño y funcionalidad. Algunas encuestas tienen como contenido refiriéndose a si el usuario esta satisfecho con el sistema o si se le hace fácil usar la aplicación móvil. Algunas encuestas tienen como contenido refiriéndose a si el usuario esta satisfecho con el sistema o si se le hace fácil usar la aplicación móvil.

### 4.1. Presentación de resultados

Se realizaron encuestas de satisfaccion (Ver anexo 3) a los clientes del negocio que en este caso son las que utilizan la aplicación móvil para él envió de pedidos. Además de que se realizó una encuesta de satisfaccion (Ver anexo 4) solo para los empleados del negocio de la microempresa tortillera. A continuación, se muestran detalladamente las respuestas a lo que los usuarios contestaron después de haber utilizado las aplicaciones tanto el sistema web como la aplicación móvil.

## 4.2. Resultados de encuestas

Se realizaron un total de 26 encuestas la cuales se dividieron de la siguiente manera entre dos usuarios:

- Clientes: Son los que interactúan con la aplicación móvil y realizan los pedidos de los productos al negocio.
- Empleados: Son los que estarán interactuando con el sistema web y se encargan de registrar las ventas del negocio.
- Administrador: Es el encargado de controlar y organizar la información del negocio realizando cuentas y administrando la información de los clientes.

### 4.2.1. Usuarios empleados y administradores

Primeramente se muestran los resultados obtenidos de las encuestas realizadas a los clientes del negocio los cuales hicieron uso del sistema web.

1.- ¿Considera que los colores utilizados para el desarrollo de la aplicación son agradables al interactuar con ella?

#### 1. Pregunta

[Más detalles](#)

● Si	6
● No	0



Figura 4.1: Respuesta de la pregunta 1, encuesta de empleados y administradores

2.- ¿Considera que mediante el uso de esta aplicación será más fácil el organizar los pedidos de los clientes?

### 2. Pregunta

[Más detalles](#)



Figura 4.2: Respuesta de la pregunta 2, encuesta de empleados y administradores

3.- ¿Considera que el mediante uso de esta aplicación será más fácil el realizar pedidos?

### 3. Pregunta

[Más detalles](#)

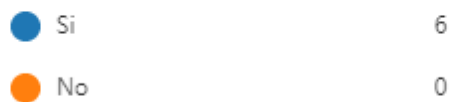


Figura 4.3: Respuesta de la pregunta 3, encuesta de empleados y administradores

4.- ¿La interfaz le parece simple y fácil de usar?

## 4. Pregunta

[Más detalles](#)

● Si	6
● No	0



Figura 4.4: Respuesta de la pregunta 4, encuesta de empleados y administradores

5.- ¿Es útil para usted una aplicación de este tipo?

## 5. Pregunta

[Más detalles](#)

● Si	6
● No	0



Figura 4.5: Respuesta de la pregunta 5, encuesta de empleados y administradores

6.- ¿La aplicación se ejecuta correctamente?

## 6. Pregunta

[Más detalles](#)

● Si	6
● No	0



Figura 4.6: Respuesta de la pregunta 6, encuesta de empleados y administradores

7.- ¿Cuál considera que es el nivel de dificultad para usar esta aplicación?

## 7. Pregunta

[Más detalles](#)

● Muy Facil	3
● Facil	2
● Regular	1
● Dificil	0
● Muy Dificil	0



Figura 4.7: Respuesta de la pregunta 7, encuesta de empleados y administradores

## 4.2.2. Usuario cliente

1.- ¿Considera que el mediante uso de esta aplicación será más fácil el realizar pedidos?

## 1. Pregunta

[Más detalles](#)

Figura 4.8: Respuesta de la pregunta 1, encuesta de clientes

2.- ¿La interfaz le parece simple y fácil de usar?

## 2. Pregunta

[Más detalles](#)

Figura 4.9: Respuesta de la pregunta 2, encuesta de clientes

3.- ¿Es útil para usted una aplicación de este tipo?

### 3. Pregunta

[Más detalles](#)

● Si	20
● No	0



Figura 4.10: Respuesta de la pregunta 3, encuesta de clientes

4.- ¿La aplicación se ejecuta correctamente?

### 4. Pregunta

[Más detalles](#)

● Si	20
● No	0



Figura 4.11: Respuesta de la pregunta 4, encuesta de clientes

5.- ¿Cuál considera que es el nivel de dificultad para usar esta aplicación?

## 5. Pregunta

[Más detalles](#)

<span style="color: blue;">●</span> Muy Facil	16
<span style="color: orange;">●</span> Facil	3
<span style="color: green;">●</span> Regular	1
<span style="color: red;">●</span> Dificil	0
<span style="color: purple;">●</span> Muy Dificil	0



Figura 4.12: Respuesta de la pregunta 5, encuesta de clientes

## 4.3. Análisis de resultados

A continuación, se hace un breve análisis y se toma el modo en que estos van a ser interpretados. Mediante este análisis sabremos si el proyecto esta funcionando correctamente cuando los usuarios interactúan con ella, además de saber si los usuarios quedaron satisfechos.

### 4.3.1. Resultados del contenido de la aplicación

De la primera encuesta que se realizo hacia los empleados y administradores resulto que de la primera pregunta donde el total del 100 % de los usuarios les gustaron los colores que se utilizaron para el proyecto. De la misma manera un 100 % de las personas cree que será más fácil organizar los pedidos, además de que se les hace simple y fácil de usar. Ningún usuario contesto negativamente que el sistema no se ejecutaba correctamente, sino al contrario todo funcionaba correctamente.

En la última pregunta 3 de los encuestados contestaron que es muy fácil usar el sistema, 2 personas contestaron que solamente fácil un nivel mas abajo que el anterior y por último

una persona contestó que el sistema no es ni difícil ni fácil de usar sino regular.

Un total del 100% contestaron positivamente indicando que con la aplicación móvil será más fácil la realización de los pedidos, además de que se les hace fácil el uso de la aplicación. Del mismo modo el 100% de las personas contestaron que la aplicación se ejecuta correctamente además de les será una aplicación muy útil. Por último, de un total de 20 encuestados 16 personas contestaron que es muy fácil usar la aplicación, 3 personas contestaron que es fácil usar la aplicación y solo un contestó que regular. (El formato de las encuestas realizadas se encuentran anexadas en el apéndice C)

# Capítulo 5

## Conclusiones

En este capítulo el objetivo principal es plasmar lo que se obtuvo al terminar el desarrollo del proyecto. Se comentará si se cumplió con el objetivo principal del proyecto, además de cómo se concluyeron los objetivos específicos que derivan del objetivo general. Por otro lado, en este capítulo se presentarán recomendaciones que puedan ayudar o puedan ser útiles en futuras investigaciones que se relacionen con este proyecto.

### 5.1. Con respecto al objetivo de la investigación

Habiendo visto en los anteriores capítulos los resultados y después de haberlos analizado se puede concluir que el objetivo general de este proyecto se cumplió. Gracias a que se logró desarrollar e implementar este sistema que tiene como función administrar los procesos de un negocio relacionado en el área laboral de una microempresa tortillera. En cuanto a los objetivos específicos se logró concluir con cada uno de ellos ya que se logró desarrollar en primer lugar las interfaces gráficas del usuario, así como diseñar y desarrollar una base de datos en la que esta haga la función de almacenar la información, tanto para el sistema web como para la aplicación móvil.

Todo esto se logró cumplir y terminar satisfactoriamente gracias a herramientas que fueron necesarias para el desarrollo del proyecto, por ejemplo, para el alojamiento del sistema web

se utilizó un hosting gratuito que cumplía con las características necesarias para trabajar con la base de datos. Después habiendo realizado la encuesta de satisfacción hacia los usuarios que utilizarían dicho sistema web y aplicación móvil se obtuvieron resultados muy positivos y así cumpliendo con el objetivo del proyecto.

## 5.2. Recomendaciones para futuras investigaciones

Este proyecto se desarrolló cumpliendo los objetivos específicos, sin embargo, aún hay más modificaciones y funciones que se le pueden agregar, de manera que este sistema vaya creciendo aún más para ayudar en la administración de esta microempresa tortillera, además de que este sistema también sea de ayuda para otros tipos de negocios.

Para investigaciones futuras con proyectos similares algunas recomendaciones que se pueden hacer para este proyecto son implementar algunas funciones extras al sistema como la implementación de un ticket al momento de registrar cada venta en el sistema, este para que se le entregue al cliente que acaba de realizar la compra. Por el lado de la aplicación móvil se puede implementar un método de pago para que al momento de realizar el pedido este se pueda pagar en línea con cargo a una cuenta bancaria o tarjeta de crédito.

Otro punto importante del lado de la aplicación móvil es que se puede trabajar para que al momento de realizar el pedido este pueda hacer un seguimiento en tiempo real mediante un mapa de Google utilizando el GPS, algo similar a la aplicación de UBER en donde el usuario puede ir visualizando el trayecto del automóvil.

Una última recomendación en el sistema web, es la de poder centralizar la base de datos para que exista un superusuario donde pueda administrar mas de una sucursal de modo que las sucursales se encuentren en distintos puntos de la ciudad o fuera de la ciudad, así de este modo el superusuario puede ver las estadísticas de todas las sucursales y así el sistema pueda darle una comparación de cada una de las sucursales.

Para concluir, el resultado que se obtiene de esta investigación y la finalización de este proyecto tecnológico deja un gran crecimiento tanto personal como de manera profesional y se espera que se siga mejorando funcionalmente para facilitar el control de información y manejo de procesos de los negocios.

# Bibliografía

- [1] B. W. Gallardo Vidal y B. R. A. Laynes Macazana, “Implementación de un sistema web para la gestión de servicios de la lavandería eden”, <http://www.repositorioacademico.usmp.edu.pe/handle/usmp/3368>, 2017, Accedido: 18-06-2019.
- [2] J. A. Saraza Grande M. A. E. González Macavilca, “Implementación de un sistema vía web con aplicación móvil para la reserva y pedidos en línea de restaurantes”, <http://www.repositorioacademico.usmp.edu.pe/handle/usmp/1202>, 2014, Accedido: 20-03-2019.
- [3] J. D. Gauchat, *El gran libro de HTML5, CSS3, Javascript*, MARCOMBO, 2012.
- [4] MDN Web Docs, “¿qué es javascript?”, <https://developer.mozilla.org/es/docs/Learn/JavaScript>, 2019, Accedido: 22-03-2019.
- [5] Daniel Pérez y Rocío Rocha Ángel Cobo, Patricia Gómez, *PHP y MySQL*, Díaz de Santos, 2005.
- [6] Margaret Rouse, “Desarrollo de aplicaciones móviles”, 2017, Accedido: 26-03-2019.
- [7] Borrego Á. Cordero J. González M. Cruz L. Hernández F. . . . Zapata Á. (s.f.) Báez, M., “Introducción a android”, 1985.

- [8] MIT App Inventor, “Mit app inventor”, <http://appinventor.mit.edu/explore/about-us.html>, 2019, Accedido: 29-03-2019.
- [9] Ramírez Benavides K., “App inventor”, <http://www.kramirez.net/Robotica/Material/>, 2019, Accedido: 29-03-2019.
- [10] J. J. P. Rivas, “Qué es y cómo empezar con ionic framework”, <https://www.phonegapSpain.com/que-es-y-como-empezar-con-ionic-framework/>, 2015, Accedido: 25-03-2019.
- [11] Illusion Studio, “¿qué es un framework web y qué ventajas aportan?”, <https://www.illusionstudio.es/que-es-un-framework-web>, 2018, Accedido: 27-03-2019.
- [12] Guevara Benites A., “¿qué es bootstrap?”, <https://devcode.la/blog/que-es-bootstrap/>, 2018, Accedido: 28-03-2019.
- [13] ITBLOGSOGETI, “Hola semantic ui, ¿adiós bootstrap?”, <https://itblogsogeti.com/2016/09/20/hola-semantic-ui-adios-bootstrap/>, 2018, Accedido: 28-03-2019.
- [14] SHIOTSU YOSHITAKA, “Bootstrap vs. foundation: Which framework is right for you?”, <https://www.upwork.com/hiring/development/bootstrap-vs-foundation-which-framework-is-right-for-you/>, 2017, Accedido: 28-03-2019.
- [15] iwantic, “Qué es el framework foundation”, <https://iwantic.com/que-es-el-framework-foundation/>, 2016, Accedido: 30-03-2019.
- [16] R. F. C. Espinoza y B. E. C. Sarango, “Análisis comparativo entre bases de datos relacionales con bases de datos no relacionales”, 2013.
- [17] Margaret Rouse, “Base de datos relacional”, 2015, Accedido: 31-03-2019.

- [18] “¿qué es un gestor de datos y para qué sirve?”, <https://blog.powerdata.es/el-valor-de-la-gestion-de-datos/que-es-un-gestor-de-datos-y-para-que-sirve>, 2019, Accedido: 31-03-2019.
- [19] José Miguel Rojas Gonzales, “Análisis comparativo de bases de datos relacionales y no relacionales”, Julio 2017.
- [20] Dirección General de Modernización Administrativa, “Metodología de análisis y gestión de riesgos de los sistemas de información”, [urlhttps://www.ccn-cert.cni.es/documentos-publicos/1789-magerit-libro-i-metodo/file.html](https://www.ccn-cert.cni.es/documentos-publicos/1789-magerit-libro-i-metodo/file.html), 2012, Accedido 16-05-2019.
- [21] S. Guardati A. Weitzenfeld, “Capítulo 12 ingeniería de software: El proceso para el desarrollo de software”, [urlhttp://weitzenfeld.robolat.org/wp-content/uploads/2015/01/WeitzenfeldGuardatiComputacion2008.pdf](http://weitzenfeld.robolat.org/wp-content/uploads/2015/01/WeitzenfeldGuardatiComputacion2008.pdf), 2008, Accedido 16-04-2019.
- [22] Paul Kimmel, “Manual de uml”, *Mexico D.F: McGraw Hill*, pp. 17–21, 2011.
- [23] Paul Kimmel, “Manual de uml”, *Mexico D.F: McGraw Hill*, pp. 101–102, 2011.

# Apéndice A

## Código fuente del sistema web

```
1  <?php
2  include ("../includes/conexion.php");
3  date_default_timezone_set('America/Chihuahua');
4  session_start();
5  if (isset($_SESSION['usuario'])) {
6  |   echo "";
7  }else{
8  |   header("Location: ../index.php");
9  }
10
11  $sql="SELECT * FROM accesos WHERE id = '".$_SESSION['id']."' ";
12  $result=mysqli_query($con,$sql);
13  while ($row=mysqli_fetch_array($result)) {
14  |   $ruta_img = $row['imagen'];
15  |   $nombre = $row['nombre'];
16  |   $usuario = $row['username'];
17  }
18  $date = new DateTime("now", new DateTimeZone('America/Chihuahua'));
19  $date= $date->format('Y-m-d');
20  // ----- VENTAS Ultimos 5 dias -----
21  $date_5dias= date('Y-m-d', strtotime('-5 day')) ; // resta 5 días
22  $sql="SELECT COUNT(*) AS resultado FROM ventas WHERE fecha BETWEEN '$date_5dias' AND '$date' ";
23  $result=mysqli_query($con,$sql);
24  $row=mysqli_fetch_array($result);
25  $ultimos5Dias = $row['resultado'];
26  // -----PEDIDOS Semanalmente-----
27  $date_semana= date('Y-m-d', strtotime('-1 week')) ; // resta 1 semana
28  $sql="SELECT COUNT(*) AS resultado FROM pedidos WHERE fecha BETWEEN '$date_semana' AND '$date' ";
29  $result=mysqli_query($con,$sql);
30  $row=mysqli_fetch_array($result);
31  $ultimaSemana = $row['resultado'];
32  // -----PEDIDOS ENTREGADOS Ultimos 5 dias-----
33  $sql="SELECT COUNT(*) AS resultado FROM pedidos WHERE fecha BETWEEN '$date_5dias' AND '$date' AND estado='Entregado' ";
34  $result=mysqli_query($con,$sql);
35  $row=mysqli_fetch_array($result);
36  $entregados = $row['resultado'];
37  ?>
--
```

Figura A.1: Código de consultas para la pantalla principal

```

89 <!-- jQuery CDN - Slim version (=without AJAX) -->
90 <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
91 | integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" crossorigin="anonymous">
92 </script>
93 <!-- Popper.JS -->
94 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.0/umd/popper.min.js"
95 | integrity="sha384-cs/chFZiN24E4KMATLdqvsezGxaxGsi4hLGOz1Xwp5UZB1LY//20VyM2taTB4QvJ" crossorigin="anonymous">
96 </script>
97 <!-- Bootstrap JS -->
98 <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.0/js/bootstrap.min.js"
99 | integrity="sha384-uefMccjF3AIv6A+rW+L4AHf99KvxDjvSu1z9VI8SKNMz4sk7buKt/6v9KI65qnm" crossorigin="anonymous">
100 </script>
101 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
...

```

Figura A.2: Librerías utilizadas de Bootstrap

```

1 <nav class="navbar navbar-expand-lg oculto" style="background-color: #0288D1">
2 <div class="container-fluid">
3 <button type="button" id="sidebarCollapse" class="btn btn-info">
4 | <i class="fas fa-bars"></i>
5 </button>
6 <span>
7 | <?php if (!empty($ruta_img)) { ??>
8 | 
9 | <?php } elseif (empty($ruta_img)) { ??>
10 | 
12 | <?php } ??>
13 | <span style="margin: 15px;color:white;"> Usuario: <?php echo $nombre; ?></span>
14 </span>
15 <span style="color:white;" id="liveclock" style=""></span>
16 </div>
17 </nav>

```

Figura A.3: Código del layout para el header

```
1 <!-- Sidebar -->
2 <nav id="sidebar" class="oculto">
3   <div class="sidebar-header">
4     <h3>Tortilleria Domenico</h3>
5     <strong>TD</strong>
6   </div>
7
8   <ul class="list-unstyled components">
9     <li>
10      <a href="home.php">
11        <i class="fas fa-home"></i>
12        <span id="txt">Inicio</span>
13      </a>
14    </li>
15    <li class="active">
16      <a href="#homeSubmenu" data-toggle="collapse" aria-expanded="false" class="dropdown-toggle">
17        <i class="fas fa-mobile-alt"></i>
18        <span id="txt">App Movil</span>
19      </a>
20      <ul class="collapse list-unstyled" id="homeSubmenu">
21        <li>
22          <a href="pedidos.php">Pedidos</a>
23        </li>
24        <li>
25          <a href="usuariosAndroid.php">Usuarios Android</a>
26        </li>
27      </ul>
28    </li>
29    <li>
30      <a href="editarPerfil.php">
31        <i class="fas fa-id-badge"></i>
32        <span id="txt">Perfil</span>
33      </a>
34    </li>
35    <li>
36      <a href="productos.php">
37        <i class="fas fa-th-list"></i>
38        <span id="txt">Productos</span>
39      </a>
40    </li>
```

Figura A.4: Código del layout para el sidebar parte 1

```
41     <li>
42         <a href="clientes.php">
43             <i class="fas fa-address-card"></i>
44             <span id="txt">Clientes</span>
45         </a>
46     </li>
47     <li>
48         <a href="accesos.php">
49             <i class="fas fa-user"></i>
50             <span id="txt">Accesos</span>
51         </a>
52     </li>
53     <li>
54         <a href="ventas.php">
55             <i class="fas fa-chart-line"></i>
56             <span id="txt">Ventas</span>
57         </a>
58     </li>
59     <li>
60         <a href="reporteVentas.php">
61             <i class="fas fa-chart-bar"></i>
62             <span id="txt">Reportes de ventas</span>
63         </a>
64     </li>
65     <li>
66         <a href="../includes/logout.php">
67             <i class="fas fa-sign-out-alt"></i>
68             Salir
69         </a>
70     </li>
71 </ul>
72 </nav>
```

Figura A.5: Código del layout para el sidebar parte 2

```
1  <?php
2  include ("includes/conexion.php");
3  if(isset($_POST['entrar'])) {
4      $usuario = $_POST['usuario'];
5      $contrasena = $_POST['password'];
6      $query = $con->query("SELECT * FROM accesos WHERE username = '$usuario' AND password = '$contrasena'");
7      $row=mysqli_fetch_array($query);
8      if($usuario == $row['username'] && $contrasena == $row['password']){
9          session_start();
10         $_SESSION['usuario'] = $usuario;
11         $_SESSION['id'] = $row['id'];
12         $_SESSION['nombre'] = $row['nombre'];
13         $_SESSION['tipo'] = $row['tipo'];
14         if ($_SESSION['tipo'] == "Administrador" ) {
15             header("Location: php/home.php");
16         }elseif ($_SESSION['tipo'] == "Usuario_Normal" ) {
17             header("Location: interfazVentas/ventas.php");
18         }
19     }else{
20         echo '<div style="text-align:center;" class="alert alert-danger" id="alert" role="alert"> Usuario y/o Contraseña incorrecto
21             <button type="button" class="close" data-dismiss="alert" aria-label="Close">
22                 <span aria-hidden="true">&times;</span>
23             </button>
24         </div>';
25     }
26 }
27 ?>
```

Figura A.6: Código del inicio de sesión del sistema web

```

1  <?php
2  include ("../includes/conexion.php");
3  mysqli_set_charset($con,'utf8');
4  session_start();
5  ini_set('error_reporting',0);
6  //date_format(Fecha, '%m/%d/%Y')
7  $output = '';
8  $sql = "SELECT ventas.id ,fecha, nombre, cantidad, precio FROM ventas INNER JOIN productos ON ventas.id_producto = productos.id ORDER BY id DESC";
9  if (isset($_POST['consulta'])) {
10     $q = $con->real_escape_string($_POST['consulta']);
11     $sql="SELECT ventas.id ,fecha, nombre, cantidad, precio FROM ventas INNER JOIN productos ON ventas.id_producto = productos.id
12     WHERE CONCAT(fecha,' ',nombre,' ',cantidad,' ',precio) LIKE '%".$q."%' ORDER BY id DESC ";
13 }
14 $result = mysqli_query($con, $sql);
15 $output .= '
16 <table class="table table-striped table-hover">
17 <thead>
18 <tr>
19 <th>Fecha</th>
20 <th>Nombre</th>
21 <th>Cantidad</th>
22 <th>Precio</th>
23 <th>Total</th>
24 <th>Eliminar</th>
25 </tr>
26 </thead>';
27 if(mysqli_num_rows($result) > 0){
28     $output .= <tbody>;
29     while($row = mysqli_fetch_array($result)){
30         $total = $row["precio"] * $row["cantidad"];
31         $fecha = new DateTime($row["fecha"]);
32         $output .= '
33 <tr>
34 <td class="fecha" data-id="'.$row["id"].'" contenteditable>'.$fecha->format('d/m/Y').</td>
35 <td class="nombre" data-id2="'.$row["id"].'">'.$row["nombre"].</td>
36 <td style="width:7%; class="cantidad" data-id3="'.$row["id"].'" contenteditable>'.$row["cantidad"].</td>
37 <td style="width:7%; class="precio" data-id4="'.$row["id"].'">'.number_format($row["precio"],2).</td>
38 <td style="width:7%; class="total" data-id5="'.$row["id"].'">'.number_format($total,2).</td>
39 <td><button type="button" name="delete_btn" data-id20="'.$row["id"].'" class="btn btn-sm btn-danger btn_delete"> <i class="fas fa-trash-alt"></i> </button></td>
40 </tr>
41 <tr>
42 <td colspan="5">
43 </td>
44 }
45 else { $output .= <tr>
46 <td colspan="2" style="font-size: 32px; text-align:center;"><b>Datos no encontrados</b></td>
47 </tr>; }
48 $output .= '
49 </tbody>
</table>

```

Figura A.7: Código de la estructura para la tabla de ventas

```
1  <?php
2  include ("../includes/conexion.php");
3  date_default_timezone_set('America/Chihuahua');
4  session_start();
5  if (isset($_SESSION['usuario'])) {
6  |   echo "";
7  }else{
8  |   header("Location: ../index.php");
9  }
10 $a=0;
11 if(isset($_POST['registrar'])) {
12
13     $nombre = $_POST['nombre'];
14     $correo = $_POST['correo'];
15     $direccion = $_POST['direccion'];
16     $cp = $_POST['cp'];
17     $telefono = $_POST['telefono'];
18     $ciudad = $_POST['ciudad'];
19
20     $sql="SELECT * FROM clientes WHERE nombre = '". $nombre.'" OR correo = '". $correo.'" ";
21     $result=mysqli_query($con,$sql);
22
23     if (mysqli_num_rows($result)>0){
24         $a=1;
25     }else {
26         $a=2;
27         $query = $con->query("INSERT INTO clientes (nombre,direccion,codigo_postal,correo,telefono,ciudad)
28         VALUES(' $nombre',' $direccion',' $cp',' $correo',' $telefono',' $ciudad)");
29     }
30 }
--
```

Figura A.8: Código para agregar clientes

```

1  <?php
2  include("../includes/conexion.php");
3  $id = $_POST["id"];
4  $text = $_POST["text"];
5  $column_name = $_POST["column_name"];
6  $table = $_POST["table"];
7  $id2 = $_POST["id2"];
8
9  if($column_name=='fecha'){
10     $hora = explode("/", $text);
11     $text= $hora[2].'-'. $hora[1].'-'. $hora[0];
12 }
13 $sql = "UPDATE $table SET ".$column_name."='".$text.'" WHERE $id2='".$id.'"";
14 if(mysqli_query($con, $sql))
15 {
16     echo $text;
17 }
18 ?>
--

```

Figura A.9: Código para editar registro de la tabla ventas

```

1  <?php
2  include("../includes/conexion.php");
3  $table = $_POST["table"];
4  $id2 = $_POST["id2"];
5  $sql = "DELETE FROM $table WHERE $id2 = '".$_POST["id"]."'";
6  if(mysqli_query($con, $sql))
7  {
8     echo 'Registro eliminado';
9  }
10 ?>

```

Figura A.10: Código para eliminar registro de la tabla ventas

```
1  ✓ $(document).ready(function() {
2      function fetch_data(consulta) {
3          $.ajax({
4              url: "../php/ventasTabla.php",
5              method: "POST",
6              success: function(data) {
7                  $("#datos").html(data);
8              },
9              dataType: "html",
10             data: {
11                 consulta: consulta
12             }
13         })
14         .done(function(respuesta) {
15             $("#datos").html(respuesta);
16         })
17         .fail(function() {
18             console.log("error");
19         });
20     }
21     $(document).on("keyup", "#busqueda", function() {
22         var valor = $(this).val();
23         if (valor != "") {
24             fetch_data(valor);
25         } else {
26             fetch_data();
27         }
28     });
29     fetch_data();
30
31     function edit_data(id, text, column_name) {
32         $.ajax({
33             url: "../php/edit.php",
34             method: "POST",
35             data: {
36                 id: id,
37                 text: text,
38                 column_name: column_name,
39                 table: "ventas",
40                 id2: "id"

```

Figura A.11: Código para buscador de la tabla ventas parte 1

```
41     },
42     dataType: "text",
43     success: function(data) {
44         function fetch_data(consulta) {
45             $.ajax({
46                 url: "../php/ventasTabla.php",
47                 method: "POST",
48                 success: function(data) {
49                     $("#datos").html(data);
50                 },
51                 dataType: "html",
52                 data: {
53                     consulta: consulta
54                 }
55             });
56         }
57         var valor = document.getElementById("busqueda").value;
58         console.log(valor);
59         if (valor != "") {
60             fetch_data(valor);
61         } else {
62             fetch_data();
63         }
64     }
65 });
66 }
67 $(document).on("blur", ".fecha", function() {
68     var id = $(this).data("id1");
69     var fecha = $(this).text();
70     edit_data(id, fecha, "fecha");
71 });
72 $(document).on("blur", ".nombre", function() {
73     var id = $(this).data("id2");
74     var nombre = $(this).text();
75     edit_data(id, nombre, "nombre");
76 });
77 $(document).on("blur", ".cantidad", function() {
78     var id = $(this).data("id3");
79     var cantidad = $(this).text();
80     edit_data(id, cantidad, "cantidad");
```

Figura A.12: Código para buscador de la tabla ventas parte 2

```
--
81 });
82 $(document).on("blur", ".precio", function() {
83     var id = $(this).data("id4");
84     var precio = $(this).text();
85     edit_data(id, precio, "precio");
86 });
87 $(document).on("click", ".btn_delete", function() {
88     var id = $(this).data("id20");
89     var table = "ventas";
90     var id2 = "id";
91     if (confirm("Quieres eliminar este registro?")) {
92         $.ajax({
93             url: "../php/delete.php",
94             method: "POST",
95             data: {
96                 id: id,
97                 table: table,
98                 id2: id2
99             },
100            dataType: "text",
101            success: function(data) {
102                var valor = document.getElementById("busqueda").value;
103                console.log(valor);
104                if (valor != "") {
105                    fetch_data(valor);
106                } else {
107                    fetch_data();
108                }
109            }
110        });
111    }
112 });
113 });
```

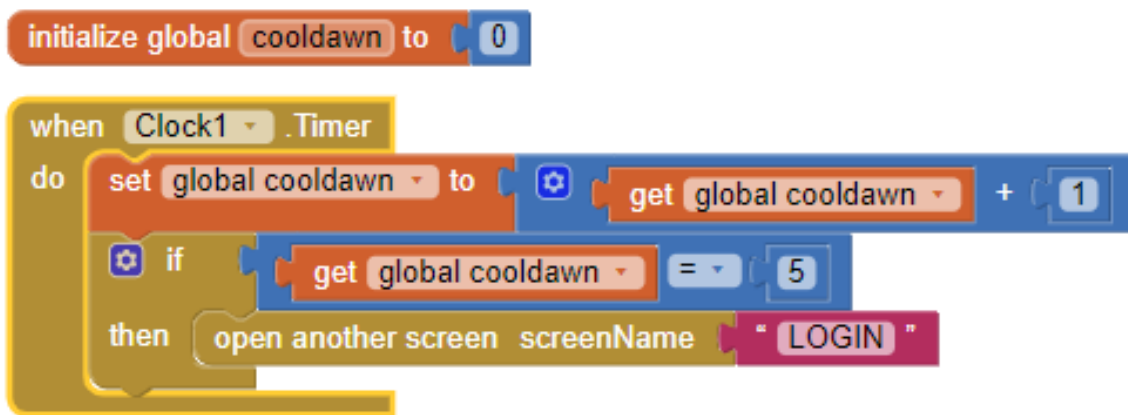
Figura A.13: Código para buscador de la tabla ventas parte 3

```
111 function show5() {
112     if (!document.layers && !document.all && !document.getElementById)
113         return
114
115     var Digital = new Date()
116     var hours = Digital.getHours()
117     var minutes = Digital.getMinutes()
118     var seconds = Digital.getSeconds()
119
120     var dn = "PM"
121     if (hours < 12)
122         dn = "AM"
123     if (hours > 12)
124         hours = hours - 12
125     if (hours == 0)
126         hours = 12
127
128     if (minutes <= 9)
129         minutes = "0" + minutes
130     if (seconds <= 9)
131         seconds = "0" + seconds
132     //change font size here to your desire
133     myclock = "<?php echo date("d/m/Y");?> " + hours + ":" + minutes + " " + " " + dn + ""
134     if (document.layers) {
135         document.layers.liveclock.document.write(myclock)
136         document.layers.liveclock.document.close()
137     } else if (document.all)
138         liveclock.innerHTML = myclock
139     else if (document.getElementById)
140         document.getElementById("liveclock").innerHTML = myclock
141     setTimeout("show5()", 1000)
142 }
143 window.onload = show5
```

Figura A.14: Código para fecha en tiempo real del sistema web

# Apéndice B

## Bloques de códigos de la aplicación móvil



```
initialize global cooldown to 0

when Clock1 .Timer
do
  set global cooldown to get global cooldown + 1
  if get global cooldown = 5
  then open another screen screenName " LOGIN "
```

The image shows a sequence of Scratch code blocks. The first block is an orange 'initialize global' block for a variable named 'cooldown' with a value of 0. The second block is a yellow 'when' block triggered by 'Clock1 .Timer'. Inside the 'do' loop, there are three blocks: an orange 'set global cooldown to' block that takes the value of 'get global cooldown' and adds 1; a blue 'if' block that checks if 'get global cooldown' is equal to 5; and a purple 'then' block that triggers the 'open another screen' block with the screen name 'LOGIN'.

Figura B.1: Pantalla de carga de la aplicación móvil

Figura B.2: Inicio de sesión de la aplicación móvil parte 1

Figura B.3: Inicio de sesión de la aplicación móvil parte 2

```

initialize global (usuario) to (get start value)
initialize global (interpolator) to ( )
when (PRINCIPAL) .Initialize
do (call InicializarScreen)

when (btnMenu) .Click
do (call Sidebar1) .Show
(interpolator) (get global interpolator)

when (btnVerPedidos) .Click
do (open another screen with start value screenName (VERPEDIDOS)
startValue (get global usuario))

when (btnHacerPedido) .Click
do (open another screen with start value screenName (PEDIDOS)
startValue (get global usuario))

when (Clock1) .Timer
do (set fecha) .Text to (call Clock1) .FormatDate
(call Clock1) .Now
instant (call Clock1) .Now
pattern (MMM d, yyyy)
    
```

Figura B.4: Pantalla principal de la aplicación móvil parte 1

```

to (InicializarScreen)
do (set global interpolator) to (Sidebar1) .AccelerateDecelerateInterpolator
(initialize local color_icons) to (Sidebar1) .Coloricons
(initialize local bg_color) to (#285D6B)
in (call Sidebar1) .Start
listItem (make a list) (Realizar pedido)
(Ver pedidos)
(Salir)
icons (make a list) (add_shopping_cart)
(assignment)
(close)
coloricons (make a list) (get color_icons)
(get color_icons)
(get color_icons)
imagen (mountain.jpg)
selectColor (get bg_color)
interpolator (get global interpolator)

when (Sidebar1) .AfterSelecting
itemPosition (itemValue)
do (call Sidebar1) .Hide
(interpolator) (get global interpolator)
if (get itemValue) == (Realizar pedido)
then (open another screen with start value screenName (PEDIDOS)
startValue (get global usuario))
else if (get itemValue) == (Ver pedidos)
then (open another screen with start value screenName (VERPEDIDOS)
startValue (get global usuario))
else if (get itemValue) == (Salir)
then (call Notifier1) .ShowChooseDialog
message (¿Desea cerrar la aplicacion?)
title (La aplicacion se cerrara)
button1Text (Cancelar)
button2Text (Aceptar)
cancelable (false)
    
```

Figura B.5: Pantalla principal de la aplicación móvil parte 2

```

initialize global (usuario) to get start value
initialize global (cont) to 0

when btnSums .Click
do
  set global cont to 0 + get global cont + 1
  set cantidad .Text to get global cont

when btnResta .Click
do
  if get global cont > 0
  then
    set global cont to get global cont - 1
    set cantidad .Text to get global cont

when btnEnviar .Click
do
  if cantidad .Text == 0 or is empty cantidad .Text
  then
    call Notifier1 .ShowAlert
    notice Agregue una cantidad porfavor
  else
    call Notifier1 .ShowChooseDialog
    message ¿Desea enviarlo?
    title El pedido sera enviado
    button1Text Cancelar
    button2Text Aceptar
    cancelable false

when btn10 .Click
do
  set cantidad .Text to 10
  set global cont to 10

when btn20 .Click
do
  set cantidad .Text to 20
  set global cont to 20

when btn30 .Click
do
  set cantidad .Text to 30
  set global cont to 30

when btn40 .Click
do
  set cantidad .Text to 40
  set global cont to 40

when btn50 .Click
do
  set cantidad .Text to 50
  set global cont to 50

when btn60 .Click
do
  set cantidad .Text to 60
  set global cont to 60

when btn70 .Click
do
  set cantidad .Text to 70
  set global cont to 70

when btn80 .Click
do
  set cantidad .Text to 80
  set global cont to 80

when btn90 .Click
do
  set cantidad .Text to 90
  set global cont to 90

when btn100 .Click
do
  set cantidad .Text to 100
  set global cont to 100

when btnRegresar .Click
do
  close screen with value result get global usuario
  
```

Figura B.6: Pantalla de realización de pedidos parte 1

```

when Notifier1 .AfterChoosing
  choice
do
  if compare texts get choice == Aceptar
  then
    set Web1 .Url to http://proyectojad62019.x10host.com/php_android/...
    call Web1 .PostText
    text join id_producto=5&username=
      get global usuario
      &cantidad=
      cantidad .Text

when Web1 .GotText
do
  call Notifier1 .ShowAlert
  notice Pedido enviado
  open another screen with start value screenName PRINCIPAL
  startValue get global usuario
  
```

Figura B.7: Pantalla de realización de pedidos parte 2

```

initialize global (usuario) to (get start value)
initialize global (lista1) to (create empty list)
initialize global (lista2) to (create empty list)
initialize global (lista3) to (create empty list)
initialize global (fecha) to ( )
initialize global (producto) to ( )
initialize global (estado) to ( )

when (VERPEDIDOS) .Initialize
do
  set (Web1) .Uri to (http://proyectojadd2019.x10host.com/php_android/...)
  call (Web1) .PostText
  text (join (username= (get global usuario)
  call (ColinTreeView1) .Initialize_Scroll
  verticalScrollArrangement (VerticalScrollArrangement1)

when (btnBack) .Click
do
  close screen with value result (get global usuario)

initialize global (longitud) to ( )
initialize global (listaTemp) to (create empty list)
initialize global (productoCantidad) to ( )
initialize global (listaFinal) to (create empty list)
initialize global (listaCompleta) to (create empty list)
    
```

Figura B.8: Pantalla de visualización de pedidos parte 1

```

when Web1 .GetText
  uri responseCode responseType responseContent
do
  set global listaCompleta to list from csv table text get responseContent
  set global longitud to length of list list get global listaCompleta
  for each number from 1
    to get global longitud
    by 1
  do
    set global lista1 to select list item list list from csv table text get responseContent
      index get number
    set global lista2 to split text select list item list get global lista1
      index 3
      at " "
    set global fecha to select list item list get global lista2
      index 1
    set global lista2 to split text select list item list get global lista1
      index 4
      at " "
    set global estado to select list item list get global lista2
      index 1
    set global lista2 to split text select list item list get global lista1
      index 8
      at " "
    set global producto to select list item list get global lista2
      index 1
    set global productoCantidad to join get global producto
      " \n "
      get global estado
    set global listaTemp to make a list " p1.png "
      get global productoCantidad
      get global fecha
    add items to list list get global listaFinal
      item get global listaTemp
  call CollinTreeListView1 .Set
    list get global listaFinal
  
```

Figura B.9: Pantalla de visualización de pedidos parte 2

The image shows a visual programming interface with several initialization blocks at the top and a main event handler block. The initialization blocks are:

- initialize global listaExtraButton to create empty list
- initialize global listaExtraButton2 to create empty list
- initialize global precioEx to
- initialize global estadoEx to
- initialize global fechaEx to
- initialize global productoEx to
- initialize global totalEx to
- initialize global detallesFinal to
- initialize global cantidadEx to

The main event handler block is:

```

when ColinTreeView1.ExtraButtonClick
  elementIndex
  do
    set global listaExtraButton to select list item list get global listaCompleta index get elementIndex
    set global listaExtraButton2 to split text select list item list get global listaExtraButton index 3 at " "
    set global fechaEx to select list item list get global listaExtraButton2 index 3
    set global listaExtraButton2 to split text select list item list get global listaExtraButton index 3 at " "
    set global productoEx to select list item list get global listaExtraButton2 index 3
    set global listaExtraButton2 to split text select list item list get global listaExtraButton index 4 at " "
    set global estadoEx to select list item list get global listaExtraButton2 index 4
    set global listaExtraButton2 to split text select list item list get global listaExtraButton index 5 at " "
    set global cantidadEx to select list item list get global listaExtraButton2 index 3
    set global listaExtraButton2 to split text select list item list get global listaExtraButton index 6 at " "
    set global precioEx to select list item list get global listaExtraButton2 index 6
  
```

Figura B.10: Pantalla de visualización de pedidos parte 3

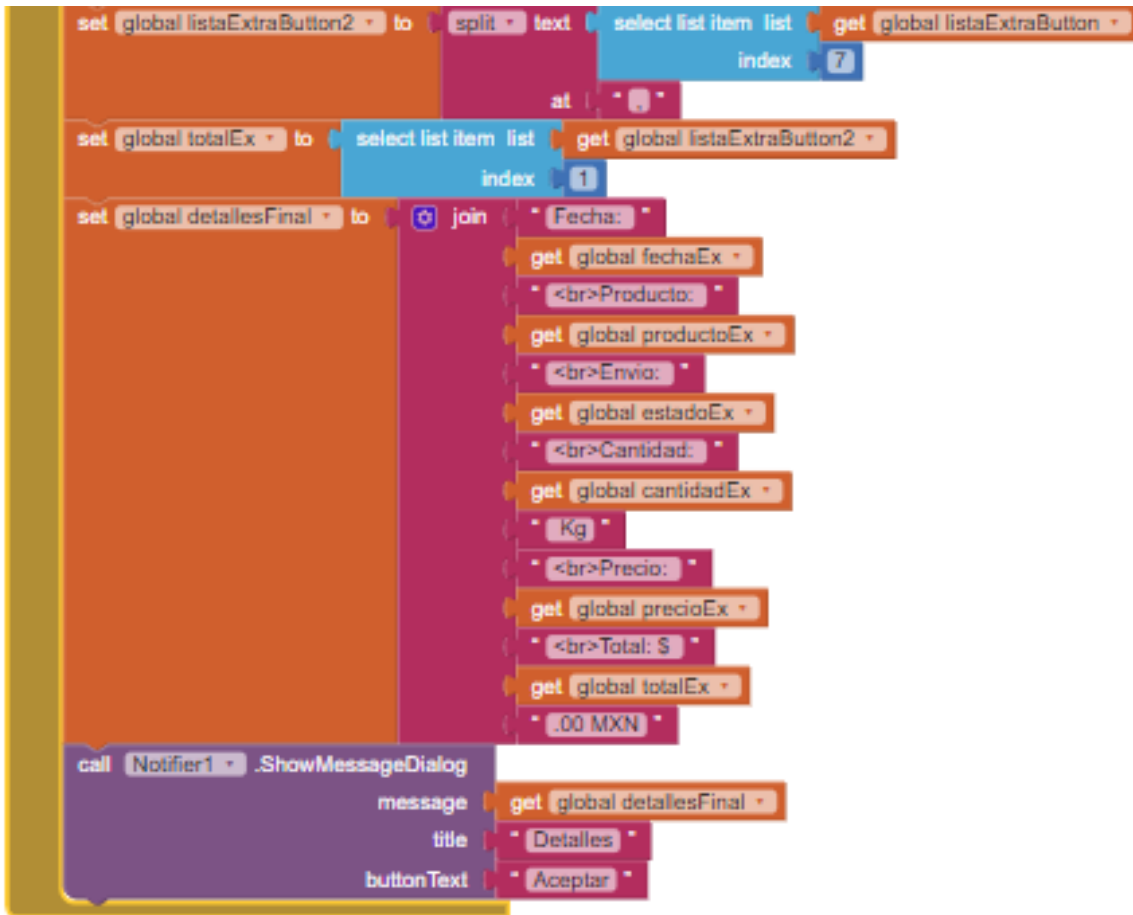


Figura B.11: Pantalla de visualización de pedidos parte 4

```

when ADMIN.Initialize
do
  set Web1.Uri to http://proyectoadd2019.x10host.com/php_android/...
  call Web1.PostText
  text join estado=
  En preparacion
  call ColinTreeView1.Initialize_Scroll
  verticalScrollArrangement VerticalScrollArrangement1
  initialize global (usuario) to get start value
  initialize global (producto) to
  initialize global (cantidad) to
  initialize global (total) to
  initialize global (numPedido) to
  initialize global (listaCompleta) to create empty list
  initialize global (listaTemp) to create empty list
  initialize global (listaFinal) to create empty list
  initialize global (lista1) to create empty list
  initialize global (lista2) to create empty list
  initialize global (longitud) to
  initialize global (cliente) to
  initialize global (fecha) to
  initialize global (precio) to
  initialize global (cadena) to
  initialize global (estado) to
  initialize global (cadena2) to
  
```

Figura B.12: Pantalla de usuario administrador de pedidos parte 1

```

when Web1 .GotText
  url responseCode responseType responseContent
do
  set global listaCompleta to list from csv table text get responseContent
  set global longitud to length of list list get global listaCompleta
  for each number from 1
    to get global longitud
    by 1
  do
    set global lista1 to select list item list list from csv table text get responseContent
      index get number
    set global lista2 to split text select list item list get global lista1
      index 1
      at " "
    set global numPedido to select list item list get global lista2
      index 1
    set global lista2 to split text select list item list get global lista1
      index 2
      at " "
    set global cliente to select list item list get global lista2
      index 1
    set global lista2 to split text select list item list get global lista1
      index 3
      at " "
    set global fecha to select list item list get global lista2
      index 1
    set global lista2 to split text select list item list get global lista1
      index 6
      at " "
    set global cantidad to select list item list get global lista2
      index 1
    set global lista2 to split text select list item list get global lista1
      index 7
      at " "
  
```

Figura B.13: Pantalla de usuario administrador de pedidos parte 2

```

set global lista2 to split text select list item list get global lista1
                                at ,                                index 7
set global precio to select list item list get global lista2
                                index 1
set global lista2 to split text select list item list get global lista1
                                at ,                                index 8
set global total to select list item list get global lista2
                                index 1
set global lista2 to split text select list item list get global lista1
                                at ,                                index 9
set global producto to select list item list get global lista2
                                index 1
set global lista2 to split text select list item list get global lista1
                                at ,                                index 5
set global estado to select list item list get global lista2
                                index 1
    
```

Figura B.14: Pantalla de usuario administrador de pedidos parte 3

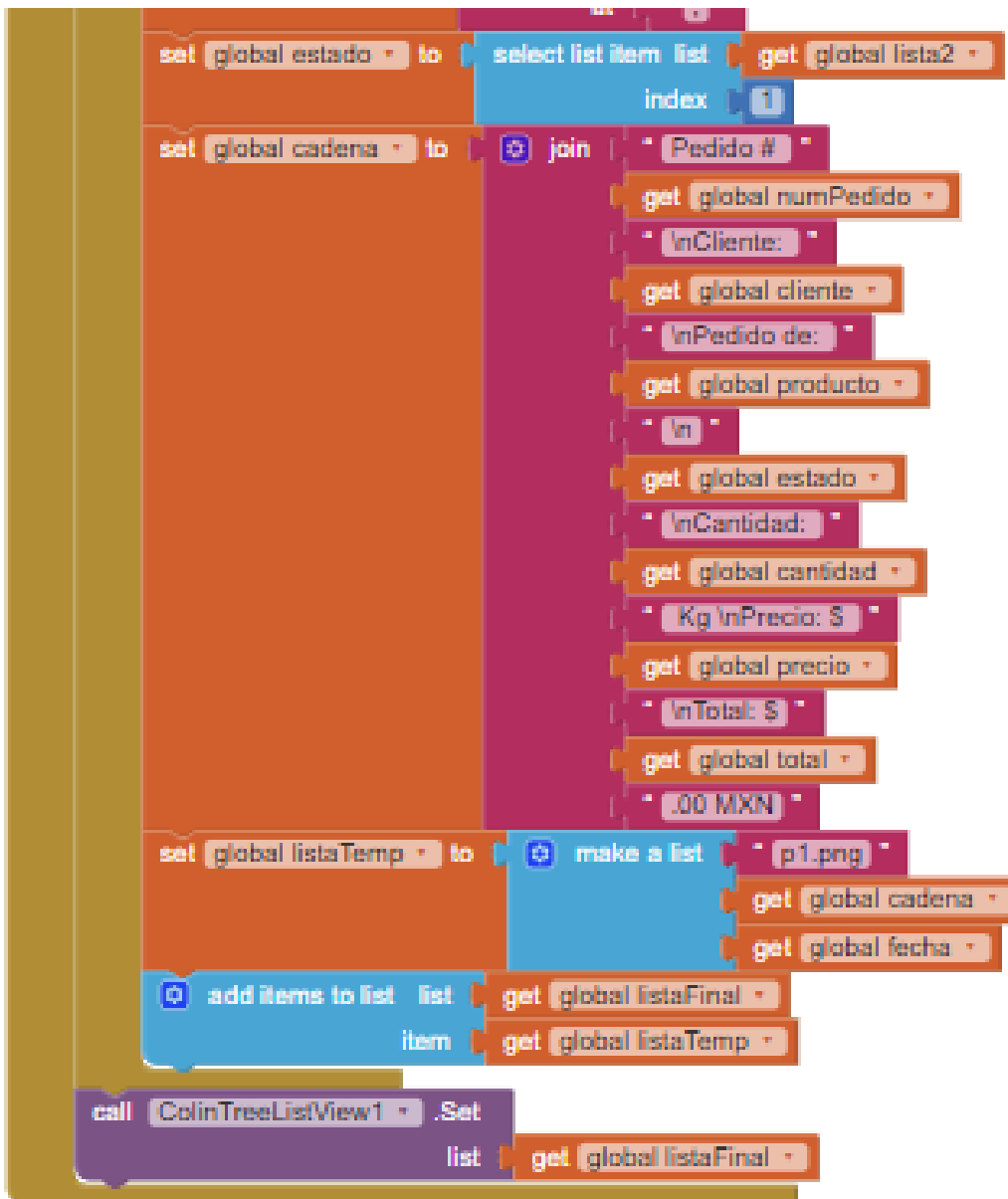


Figura B.15: Pantalla de usuario administrador de pedidos parte 4

```

initialize global listaExtraButton2 to create empty list
initialize global listaExtraButton to create empty list

when ColinTreeView1.ExtraButtonClick
  elementIndex
do
  set global listaExtraButton to select list item list
  index get global listaCompleta
  get elementIndex
  set global listaExtraButton2 to split text
  select list item list get global listaExtraButton
  index 5
  at ','
  set global estado to select list item list
  index 1
  get global listaExtraButton2
  set global listaExtraButton2 to split text
  select list item list get global listaExtraButton
  index 1
  at ','
  set global numPedido to select list item list
  index 1
  get global listaExtraButton2
  if get global estado == 'En preparacion'
  then
    call Notifier1.ShowChooseDialog
    message 'El estado cambiara a 'En camino''
    title 'Editar estado'
    button1Text 'Actualizar'
    button2Text 'Cancelar'
    cancelable false
  else if get global estado == 'En camino'
  then
    call Notifier1.ShowChooseDialog
    message '¿Desea cerrar el pedido?'
    title 'Finalizar pedido'
    button1Text 'Actualizar'
    button2Text 'Cancelar'
    cancelable false
  
```

Figura B.16: Pantalla de usuario administrador de pedidos parte 5

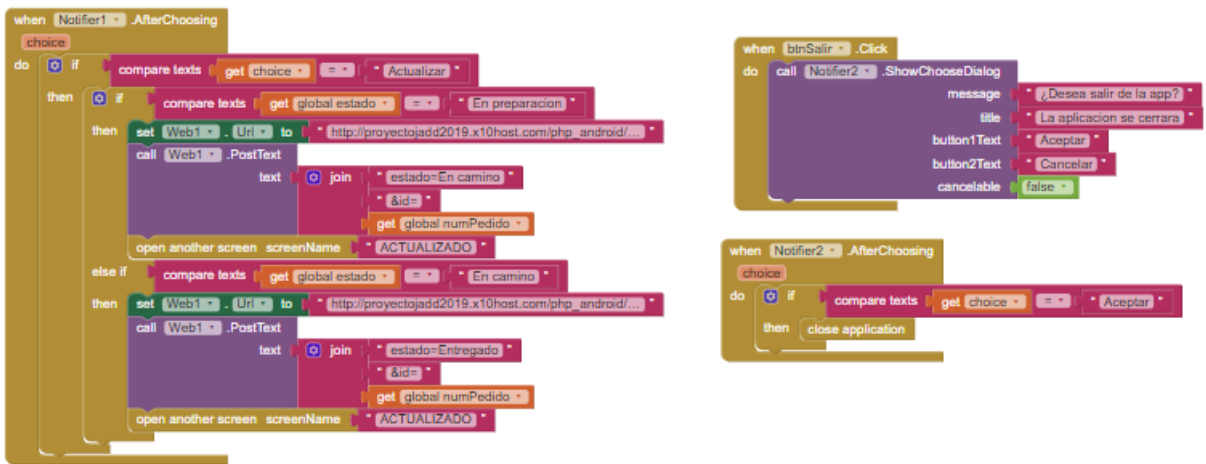


Figura B.17: Pantalla de usuario administrador de pedidos parte 6

# Apéndice C

## Formatos de encuestas realizadas a los usuarios

Encuesta de satisfacción del usuario		
<b>I. Contenido de la aplicación</b>		
1.- ¿Considera que los colores utilizados para el desarrollo de la aplicación son agradables al interactuar con ella?	Si [ ]	No [ ]
2.- ¿Considera que el mediante uso de esta aplicación será más fácil el organizar los pedidos de los clientes?	Si [ ]	No [ ]
3.- ¿Considera que el mediante uso de esta aplicación será más fácil el realizar pedidos?	Si [ ]	No [ ]
4.- ¿La interfaz le parece simple y fácil de usar?	Si [ ]	No [ ]
<b>II. Funcionalidad de la aplicación</b>		
5.- ¿Es útil para usted una aplicación de este tipo?	Si [ ]	No [ ]
6.- ¿La aplicación se ejecuta correctamente?	Si [ ]	No [ ]
7.- ¿Cuál considera que es el nivel de dificultad para usar esta aplicación?	1) Muy fácil 2) Fácil 3) Regular 4) Difícil 5) Muy difícil	

Figura C.1: Formato de encuesta realizada a los usuarios del sistema web

Encuesta de satisfacción del usuario		
<b>I. Contenido de la aplicación</b>		
1.- ¿Considera que el mediante uso de esta aplicación será más fácil el realizar pedidos?	Si [ ]	No [ ]
2.- ¿La interfaz le parece simple y fácil de usar?	Si [ ]	No [ ]
<b>II. Funcionalidad de la aplicación</b>		
3.- ¿Es útil para usted una aplicación de este tipo?	Si [ ]	No [ ]
4.- ¿La aplicación se ejecuta correctamente?	Si [ ]	No [ ]
5.- ¿Cuál considera que es el nivel de dificultad para usar esta aplicación?	1) Muy fácil 2) Fácil 3) Regular 4) Difícil 5) Muy difícil	

Figura C.2: Formato de encuesta realizada a los usuarios de la aplicación móvil