



Universidad Autónoma del Estado de México

Centro Universitario UAEM Valle de Chalco

DESARROLLO DE UN SISTEMA PARA LA ADMINISTRACIÓN EFICIENTE DE UNA DULCERÍA IMPLEMENTANDO VISUAL STUDIO CODE, XAMPP, BOOTSTRAP Y MARIADB

T E S I S

QUE PARA OBTENER EL TÍTULO DE

INGENIERO EN COMPUTACIÓN

P R E S E N T A

AXEL FABIAN CLARA FLORES

ASESOR:

MTRO. EN C.C. RICARDO JAVIER BUCIO LÓPEZ
Revisora: DRA. EN C. MAGALLY MARTÍNEZ REYES
Revisor: MTRO. EN ING. LUIS RENÉ SANTOS RUBIO

VALLE DE CHALCO SOLIDARIDAD, MÉXICO AGOSTO 2025.



CUVCH

**DESARROLLO DE UN SISTEMA PARA LA
ADMINISTRACIÓN EFICIENTE DE UNA DULCERÍA
IMPLEMENTANDO VISUAL STUDIO CODE, XAMPP,
BOOTSTRAP Y MARIADB**

ÍNDICE

I. RESUMEN	6
II. ANTECEDENTES	9
2.1 Justificación	15
III. IMPORTANCIA DEL PROBLEMA	18
IV. OBJETIVOS	19
General	19
Específicos	19
V. PLANTEAMIENTO DEL PROBLEMA O PREGUNTA DE INVESTIGACIÓN	20
VI. MARCO TEÓRICO	22
6.1 Historia y evolución de la computación	22
Orígenes de la computación	22
Primeras computadoras y su impacto en el mundo.....	23
6.2 Ingeniería en Computación	24
6.2.1 Evolución de la carrera y su impacto en la tecnología	24
6.3 Programación	26
6.4 Paradigmas de programación	27
Programación imperativa	28
Programación estructurada.....	28
Programación procedimental	28
Programación modular.....	29
Programación declarativa	29
Programación lógica	29
Programación funcional	29
Programación orientada a objetos (POO)	30
6.5 Clasificación de los lenguajes de programación	30
Lenguajes de bajo y alto nivel.....	31
Lenguajes de bajo nivel	31
Lenguajes de alto nivel	31
Lenguajes compilados e interpretados	32
6.6 Ejemplos de lenguajes de programación	32
Lenguaje C.....	32
Lenguaje C++.....	33
Lenguaje Java.....	33
Lenguaje Python	33
JavaScript	34
Lenguaje PHP	34

Lenguaje Swift.....	34
Lenguaje Go (Golang)	34
Lenguaje Kotlin	35
6.7 La importancia de la programación en el desarrollo de sistemas	35
Rol de la programación en la creación del software	35
Diferencia entre software a medida y software comercial	36
Programación en los negocios.....	36
6.8 Ciclo de vida del desarrollo de software (SDLC).....	37
Modelos de desarrollo de software	37
Modelo en Cascada	37
Modelo Espiral	37
Modelo Ágil	38
6.9 Metodologías ágiles en el desarrollo de software	38
Metodología Scrum	38
Metodología Kanban	39
Beneficios de una metodología ágil en proyectos de software	39
6.10 Frameworks	40
Ventajas de usar frameworks	41
6.11 Frameworks más utilizados	41
6.12 Base de datos (BD)	42
Base de datos Estáticas	43
Base de datos Dinámicas	43
6.13 Modelos de base de datos	43
Modelo relacional.....	43
Modelo no relacional	44
Jerárquicas.....	44
De red	44
Orientadas a objetos	45
6.14 Visual Studio Code como entorno de desarrollo	45
Principales ventajas	45
6.15 Bootstrap como herramienta de diseño responsivo	46
6.16 XAMPP como servidor local para desarrollo web	48
6.17 MariaDB y su importancia en la gestión de base de datos	48
Diferencias entre MariaDB y MySQL	49
6.18 Lenguaje de programación PHP	49
6.19 Interfaz de usuario y experiencia del usuario (UI/UX).....	50
Importancia de UI/UX en el desarrollo de software	50
6.20 Modelo cliente-servidor	50

6.21 Front-end y Back-end	51
6.22 Lenguaje unificado de modelado (UML)	51
Diagrama de caso de uso	51
Diagrama de secuencia	52
Modelo entidad-relación.....	53
Diagrama de actividades	54
Diagrama de flujo de datos	55
VII. MÉTODOS Y TÉCNICAS DE INVESTIGACIÓN.....	57
VIII. PRESENTACIÓN Y DISCUSIÓN DE RESULTADOS	60
8.1 Análisis de requerimientos	60
Requerimientos funcionales.....	60
Requerimientos no funcionales.....	61
8.2 Herramientas a utilizar.....	61
8.3 UML del desarrollo del sistema	62
8.3.1 Diagrama caso de uso	62
8.3.2 Diagrama de secuencia	64
8.3.3 Diagrama de Clases	69
8.3.4 Diagrama Entidad – Relación (ER).....	70
8.4 Implementación (Desarrollo del sistema)	71
Modelo	71
8.4.1 Diseño del sistema.....	74
8.5 Verificación	87
8.7 Lanzamiento del sistema.....	89
IX. CONCLUSIONES Y SUGERENCIAS	95
Conclusiones	95
Sugerencias.....	96
X. REFERENCIAS	97

I. RESUMEN

En este trabajo se desarrollará un sistema que apoye a una dulcería en la organización y control de sus tareas diarias, como el registro de ventas, productos, clientes, empleados entre otras más, así permitiendo una administración más clara, rápida y segura de su información.

Actualmente la dulcería no cuenta con un sistema de gestión para que pueda llevar un buen control de su información personal y de los movimientos que genera, así como la de sus empleados, clientes y de los movimientos que se hacen como las ventas de cada día, las entradas y salidas de productos o incluso la llegada de producto nuevo. Causando un conflicto al querer adquirir cualquier tipo de información, ya que todo este tipo de trabajo se hace de forma manual, tanto en los cálculos de las ventas del día, como los nuevos registros que se genera cada día, requiriendo de mucho tiempo y esfuerzo por el simple hecho de no llevar un buen control y no tener una forma más fácil para adquirir esa información con pocas probabilidades de errores humanos.

En el presente trabajo se diseñó un sistema que permite tener un mejor manejo de información, así como también una forma segura y fácil de obtener la información requerida con poca probabilidad de errores. El sistema cuenta con una autenticación para que solamente el personal autorizado pueda entrar al programa mediante un correo con una contraseña y pueda realizar ciertas actividades dentro como registrar a un nuevo usuario, empleado, cliente, producto o venta con las especificaciones de cada uno, también puede modificar los registros que necesiten actualizaciones, así como también pueda eliminarlos o que solo tenga la necesidad de buscar cierta información de cualquier registro puede hacerlo de una manera más sencilla sin necesidad de buscarla personalmente y de forma manual. Con la finalidad de facilitar el acceso a la información, reducir el tiempo que un empleado pierde en ello y que la dulcería tenga un mejor manejo de su información.

Los métodos utilizados para llevar a cabo la investigación de este trabajo fue la investigación experimental y documental. Tomando en cuenta que la documental se basó en la recopilación de información de diferentes tipos de fuente y la experimental para la implementación y pruebas en tiempo real durante la creación del programa.

El sistema se desarrolló haciendo uso de la herramienta Visual Studio Code, un editor de código fuente con la versión 8.2, con el lenguaje de programación para el desarrollo del Back-end se utilizó PHP (Hypertext Preprocessor) en español (Preprocesador de Hipertexto) y para el Front-end se utilizó el framework de multiplataforma Bootstrap, para que el diseño del sistema logre una vista amigable y confiable para los usuarios.

El sistema gestor de base de datos utilizado para esta creación es MariaDB en su versión 10.6.4, utilizando el lenguaje SQL para la creación de la base de datos y la herramienta de XAMPP para soportar la BD en PHP para que el sistema lo reconozca.

También se utilizó la herramienta de Freepik, para la creación de algunos iconos y mensajes de entrada y salida de cualquier movimiento para que el sistema tenga una presentación más profesional y amigable a vista de los usuarios.

Como resultado de este proyecto, se logró implementar un sistema funcional que permite a la dulcería realizar todas sus tareas de manera eficiente. Dado que proporciona un acceso controlado mediante la autenticación, lo que brinda una mayor seguridad, donde solo el personal puede realizar operaciones dentro de él. Gracias a su interfaz y diseño, se facilitó la navegación y el uso por parte de los trabajadores del negocio, reduciendo significativamente la posibilidad de errores y mejorando el control de la información.

Finalmente se concluyó que el uso de las herramientas tecnológicas como Visual Studio Code, XAMPP, Bootstrap, MariaDB y el lenguaje PHP facilitó el desarrollo del programa y permitió resolver necesidades específicas del negocio. Por lo que, la implementación de esta herramienta digital, contribuyó a optimizar

procesos internos que antes se realizaban de forma manual, reduciendo problemas y pérdida de tiempo esperando una respuesta. A lo largo de este proyecto, se comprobó que la metodología SCRUM permitió mantener un desarrollo estructurado y centrado en los requerimientos del cliente, logrando una solución funcional que facilita el trabajo diario del personal.

II. ANTECEDENTES

A continuación, se presentan los antecedentes más relevantes relacionados con el tema del proyecto de investigación. Los cuales han sido seleccionados debido a su relevancia teórica, práctica y metodológica en el diseño y la creación del sistema para la dulcería. Cada uno de ellos contribuye a ampliar el conocimiento sobre el tema, además de ofrecer algunas sugerencias que este proyecto busca abordar. Por lo cual se revisará algunas definiciones.

Una computadora es un dispositivo electrónico capaz de procesar y almacenar datos de manera eficiente, permitiendo la realización de diversas tareas. Por otra parte, un sistema es un conjunto de elementos interconectados que trabajan en conjunto para cumplir un objetivo específico, como gestionar información, optimizar procesos o resolver distintos problemas.

Un framework es un marco de trabajo cuyo propósito es facilitar la solución de problemas que pueden surgir durante la programación. Los frameworks aceleran el proceso de desarrollo al simplificar tareas como la organización del código o el trabajo colaborativo dentro de un proyecto determinado (Juárez, 2022).

Una página web responsiva es aquella que es capaz de adaptarse a cualquier dispositivo donde se visualice. En pocas palabras, es aquella página que se adapta automáticamente al tamaño de la pantalla de cualquier dispositivo como un teléfono, Tablet o una computadora.

Bootstrap es un framework que permite a los desarrolladores construir páginas web responsivas de manera más rápida y sencilla. En este sentido, proporciona un conjunto de componentes y plantillas CSS, HTML y JavaScript que pueden ser utilizados o modificados de forma gratuita. Además, ofrece una amplia variedad de componentes que pueden incorporarse y personalizarse en

cualquier proyecto, como botones, menús de navegación, galerías de imágenes, formularios y páginas de productos (Flores, 2023).

En seguida se mostrarán algunos ejemplos de soluciones a las necesidades específicas de un negocio que hacen uso de la tecnología que se utilizó en este proyecto en sus diseños.

Un sistema de gestión para puntos de venta enfocado en pequeñas tiendas de abarrotes fue desarrollado con la tecnología de Visual Studio Code como entorno de desarrollo, junto con el framework Bootstrap para el diseño de vistas. La lógica fue implementada con PHP y conectada a una base de datos relacional en MySQL. Los autores concluyeron que el uso de Bootstrap y Visual Studio Code permitió acelerar el proceso de diseño y desarrollo, mientras que PHP y MySQL proporcionaron eficiencia y robustez en la gestión de datos (López y Pérez, 2020).

Un sistema automatizado para la gestión de inventarios en pequeñas empresas fue desarrollado empleando el framework Bootstrap para garantizar vistas responsivas y JavaScript para funcionalidades dinámicas. El back-end fue desarrollado con Node.js y Express.js, y la base de datos fue gestionada con MongoDB. Los desarrolladores utilizaron Visual Studio Code como entorno de desarrollo, destacando que la combinación de estas tecnologías permitió un desarrollo ágil y eficiente (Martínez, 2021).

Asimismo, se han creado aplicaciones web con el apoyo de estas tecnologías. La creación de estos programas es para la gestión de ventas e inventarios en tiendas de conveniencia se realizó utilizando el framework Bootstrap para el diseño de la interfaz de usuario y Java con Spring Boot para el back-end. La base de datos fue gestionada con MySQL y el sistema se implementó en un servidor web Apache Tomcat. Según los autores, Bootstrap facilitó el diseño de vistas, mientras que Spring Boot ofreció flexibilidad para funcionalidades avanzadas (Torres, 2022).

Un sistema de punto de venta con autenticación de usuarios fue desarrollado utilizando Bootstrap para el diseño de las vistas y Django como framework principal en el back-end. La base de datos fue gestionada con PostgreSQL, lo que resultó en un sistema robusto y una interfaz más atractiva (Gómez, 2021).

De igual manera, el desarrollo de una plataforma web para la gestión de pedidos en línea en restaurantes utilizó Visual Studio Code como entorno de desarrollo, el framework Bootstrap para la creación de vistas responsivas y jQuery para funcionalidades interactivas en el front-end. En el back-end, se empleó PHP con el framework Laravel, y la base de datos fue gestionada con MySQL Workbench. Laravel simplificó la implementación de funcionalidades avanzadas, mientras que Bootstrap y Visual Studio Code facilitaron el diseño y desarrollo del sistema (Vásquez, 2020).

Un sistema de facturación electrónica para pequeñas y medianas empresas fue desarrollado utilizando Visual Studio Code como entorno de desarrollo, Bootstrap para el diseño responsivo de la interfaz de usuario y PHP con MySQL para la gestión de la base de datos. El objetivo principal era mejorar la precisión y eficiencia en la emisión de facturas digitales, reduciendo los errores manuales y agilizando los procesos administrativos. Los desarrolladores concluyeron que la combinación de estas tecnologías permitió una rápida implementación del programa, facilitando su escalabilidad y mantenimiento (Rodríguez y Sánchez, 2021).

Por otro lado, un sistema de administración de reservas para hoteles de pequeño y mediano tamaño fue desarrollado con Bootstrap para la interfaz de usuario, PHP en el back-end y MySQL como base de datos. Se utilizó Visual Studio Code como entorno de desarrollo principal, lo que facilitó la integración de herramientas de control de versiones y pruebas de código. Según los autores, el uso de Bootstrap permitió un diseño flexible y adaptable a dispositivos móviles, mientras que MySQL garantizó un almacenamiento eficiente y seguro de los

datos de las reservas, contribuyendo a la automatización de procesos en la gestión hotelera (Fernández, 2020).

Asimismo, el desarrollo de un sistema de control de inventarios para una empresa distribuidora de productos de consumo masivo utilizó Bootstrap para el diseño de la interfaz gráfica, Laravel como framework back-end en PHP y MySQL para la administración de datos. Visual Studio Code fue empleado para la codificación y pruebas del sistema. La solución permitió mejorar la gestión del inventario en tiempo real, minimizando pérdidas y optimizando la rotación de productos. Los autores destacaron que la elección de Laravel permitió una mayor seguridad y estructura en el desarrollo del sistema, mientras que Bootstrap ofreció una experiencia de usuario intuitiva y moderna (Jiménez, 2022).

De igual manera, se ha desarrollado un sistema de gestión de citas médicas en clínicas privadas que fue implementado utilizando Bootstrap para el diseño responsivo de la interfaz, PHP con CodeIgniter en el back-end y MySQL para el almacenamiento de datos. Visual Studio Code fue empleado como entorno de desarrollo debido a su compatibilidad con múltiples extensiones que facilitaron la depuración del código. Permitted a los pacientes programar, modificar y cancelar citas en línea, reduciendo el tiempo de espera y optimizando la organización interna de las clínicas. Los desarrolladores concluyeron que la combinación de estas tecnologías permitió un desarrollo ágil y eficiente del sistema, mejorando la experiencia del usuario final (Hernández, 2021).

Además, un software para la gestión de pedidos y entregas en una empresa de comida rápida fue desarrollado utilizando Bootstrap para la interfaz de usuario, Node.js en el back-end y MySQL como base de datos. El programa permitió automatizar la toma de pedidos, el seguimiento de entregas y la administración de inventarios en tiempo real. Visual Studio Code se empleó como entorno de desarrollo debido a su facilidad para integrar herramientas de depuración y control de versiones. Los autores resaltaron que la utilización de Bootstrap mejoró la experiencia del usuario al ofrecer una interfaz atractiva y

adaptada a dispositivos móviles, mientras que MySQL proporcionó un almacenamiento de datos seguro y escalable (Ortega, 2023).

Finalmente, el desarrollo de un sistema de administración para bibliotecas escolares utilizó Visual Studio Code como entorno de desarrollo, Bootstrap para la interfaz, PHP en el back-end y MySQL como base de datos. Dado que, permitió a los estudiantes y personal docente realizar préstamos y devoluciones de libros de manera automatizada, optimizando la gestión del inventario bibliográfico. Los autores concluyeron que el uso de Bootstrap facilitó la forma responsiva hacia la accesibilidad y visibilidad del sistema en diversos dispositivos, mientras que la integración con MySQL permitió un control eficiente del historial de préstamos y devoluciones, mejorando la organización y reduciendo pérdidas de material bibliográfico (Ramírez, 2022).

De todos ejemplos se observa que utilizar este tipo de tecnologías en un proyecto es una buena opción porque permiten optimizar procesos, mejorar el control de la información y brindar una solución tecnológica adaptada a distintas peticiones específicas. Por lo que, de estos ejemplos se retoma la importancia de desarrollar sistemas personalizados que respondan a problemas reales y concretos. Finalmente, la propuesta para la dulcería es diferente a los sistemas desarrollados que fueron mencionados anteriormente, donde se basa específicamente en las necesidades del cliente considerando su entorno, recursos y procesos particulares, con el objetivo de ofrecer una herramienta digital útil.

El sistema desarrollado para la dulcería permitirá optimizar los procesos administrativos y operativos del negocio mediante el registro de usuarios para acceder al sistema, garantizando la seguridad y el control de la información. Además, se implementarán módulos específicos para gestionar las distintas áreas del negocio, lo que facilitará un seguimiento más preciso de las tareas diarias de la dulcería. Este sistema brindará un servicio más rápido, eficiente y organizado, reduciendo errores manuales y mejorando la experiencia del

personal. De esta manera, se logrará un acceso a la información más detallada y específica, lo que permitirá tomar decisiones informadas y satisfacer las necesidades del negocio de manera más efectiva.

2.1 Justificación

La mayoría de los pequeños negocios, como las micro y las pequeñas empresas, presentan la necesidad de gestionar y administración de manera eficiente. Sin embargo, adquirir licencias de software comercial suele resultar costo y muchas veces requiere una inversión adicional en la capacitación para su adecuado uso, debido a su complejidad. Por ello, desarrollar un programa de administración específico para el negocio, que se adapte a sus necesidades particulares, es una propuesta hecha a la medida y de bajo costo.

La razón de llevar a cabo esta implementación de un sistema para la administración de una dulcería es para ofrecer una solución que mejore todo su movimiento interno, llevando a cabo un buen uso y manejo de la información que se encuentra dentro del negocio como lo que es, información personal de los empleados, datos de los clientes, las cuentas de las ventas del día, datos de los productos y movimientos de los mismos. Este trabajo es el encargado de administrar los datos de una manera mucho más segura y eficiente, implementando los registros de cada dato o movimiento de forma digital.

Se espera obtener varios beneficios a favor del comercio, empezando por la seguridad de su información interna, tomando en cuenta que al pasar su contenido de forma digital reduce significativamente el riesgo de la pérdida total del mismo, tanto delicada como lo que son las ventas del día, aunque cualquier registro es delicado, existe una más que la otra como la documentación personal de sus mismos empleados, donde todo está archivado en papel, donde en un futuro podría llegar a perderse por cualquier motivo imprevisto, pérdida de esos documentos, algún incidente que los dañe físicamente como por ejemplo, que se lleguen a mojar, ensuciarse por polvo u otra cosa, o por el simple hecho de que el papel se deteriore con el tiempo, así perdiendo la claridad y eficiencia, por lo que esto no solo podría pasarle a los datos de los empleados sino a todo el movimiento interno de la dulcería, perdiendo poco a poco su eficiencia y

seguridad, dando una mala imagen e impresión hacia sus clientes y empleados. Otro beneficio al implementar este sistema es que sea muy eficiente, tomando en cuenta que gracias a esto se obtiene la reducción de carga de trabajo hacia el personal, debido a que esto evita el mal manejo de la administración que se realiza de forma manual, lo que ocasiona malos entendidos entre los pedidos como productos de más o de menos, cuentas erróneas, muy mala perspectiva, malas interpretaciones entre los empleados y el cliente, el uso excesivo de papel y tiempo de los mismos trabajadores y clientes, donde esto es muy valioso para todos, considerando que entre más rápido y efectivo sea el trabajo y la atención a sus compradores es mejor, dando una buena atención y evitar hacerle perder bastante tiempo y así poder dejar una buena imagen y buen manejo de sus servicios.

Al implementar este sistema es conveniente si se observa desde distintos puntos de vista, por lo que es conveniente porque mejora la seguridad, es más eficiente, eficaz, mejora la experiencia del o los usuarios, facilita el manejo de información de una manera digital y moderna, así haciendo que el negocio entre en competencia de otras micro empresas, facilita el trabajo y la reducción de mucho tiempo que los trabajadores utilizan realizando esos registros y movimientos de forma manual, tomando en cuenta que beneficia mucho, tanto a los empleados como a los clientes, obteniendo cuentas claras y el buen manejo de la administración que se realiza al momento de hacer cualquier movimiento con los productos, reduciendo las pérdidas de los mismos, así mostrando el avance tecnológico en el comercio.

El sistema será muy eficiente, empezando por la reducción del tiempo, como ya se mencionó, que es lo más valioso que tenemos las personas, y abusar del de los compradores para poder realizar sus registros manualmente como se ha estado haciendo últimamente es bastante tedioso y fastidioso para ellos, lo que provoca que la clientela disminuya poco a poco, entonces, implementando este programa hace que se eviten ese tipo de problemas y fastidios, así

reduciendo el tiempo que los trabajadores se llevan. Otro punto sería la reducción de fallas, ya que el uso del papel genera problemas humanos como pérdidas de información, cálculos incorrectos o registros inconscientes, también es eficiente en la optimización de inventario del negocio, donde permite un seguimiento preciso de los productos disponibles, así evitando desabastos o pérdidas de producto por caducidad y para poder tener un acceso rápido, fácil y práctico a los registros que se desea buscar de inmediato sin necesidad de buscarlos entre archivos físicos y la pérdida de tiempo que uno se lleva buscando de forma física.

Así mismo es muy eficaz por el hecho de que tiene un mejor control administrativo, esto quiere decir que facilita la gestión de la información de los empleados, los clientes y sus productos, asegurando una mejor organización de los recursos de la dulcería. Considerando también que la administración interna será mucho más segura a diferencia de los registros en papel, donde los datos se pasan de forma digital y con restricciones de acceso para proteger esos mismos, considerando también la pérdida de la misma por cualquier motivo, tomando en cuenta también la facilidad para tomar decisiones, por ejemplo, identificar patrones de compras y tomar decisiones estratégicas para el crecimiento del comercio y una atención mejorada, lo que esto permite agilizar más las ventas y que los empleados se dediquen a brindar un mejor servicio, en lugar de gestionar registro de forma manual.

En vista de que este sistema no solamente va a hacer que la dulcería funcione de una manera más rápida y organizada, sino que también mejora los resultados y la calidad del servicio, así contribuyendo al crecimiento que pueda tener.

III. IMPORTANCIA DEL PROBLEMA

La dulcería no cuenta con ningún sistema que le permita llevar un buen manejo de toda su documentación interna, como lo que puede ser la información personal de sus empleados, sus clientes, también de sus ventas de día a día y movimientos que llegan a tener sus bienes, como cuando llega nuevo producto, cuantos dulces se vendieron y cuantos quedan, su caducidad, entre otro tipo de registro que necesitan saber de esos artículos, por lo que ese tipo de actividades de estar revisando lo requerido de forma física y manual lleva mucho tiempo, lo cual no es muy recomendable. Ya que al crear un programa que pueda realizar todo este tipo de acciones sin la necesidad de revisarlo personalmente, sin perder mucho tiempo y de hacerlo físicamente sería de mucha utilidad y a su vez facilitaría la consulta de su contenido.

IV. OBJETIVOS

General

Diseñar una solución eficiente y segura para una mejor administración interna de una dulcería, donde el propósito es reducir la pérdida de información, tiempo y evitar una mala experiencia tanto a los empleados como a los clientes, desarrollando un sistema de gestión para poder administrar todos esos registros y esos movimientos que se realizan dentro, utilizando la tecnología como Visual Studio Code, XAMPP, Bootstrap y MariaDB, haciendo que el programa pueda administrar el contenido de forma digital de una manera más segura, eficiente y eficaz.

Específicos

Los siguientes objetivos específicos que se esperan al diseñar e implementar este sistema son los siguientes:

Investigar y analizar los procesos operativos actuales de la dulcería para identificar áreas de mejora y definir los requerimientos del sistema de gestión.

Desarrollar la estructura y lógica del programa utilizando Visual Studio Code, la base de datos con MariaDB y XAMPP y una interfaz de usuario utilizando Bootstrap y creación de contraseñas encriptadas de los usuarios.

Implementar módulos específicos para gestionar el control de inventario, el manejo de información del personal y clientes, así como los registros de los mismos.

Evaluar el desempeño del sistema mediante pruebas funcionales para garantizar su correcto funcionamiento, minimizando errores operativos y mejorando la eficiencia del negocio.

V. PLANTEAMIENTO DEL PROBLEMA O PREGUNTA DE INVESTIGACIÓN

En la actualidad, la dulcería enfrenta diversos desafíos relacionados con la gestión de sus operaciones diarias, como el control del inventario, la atención eficiente a los clientes y el manejo adecuado de la información. Todavía, los procesos administrativos y operativos se realizan de manera manual, lo que genera ineficiencias que impactan negativamente tanto en la experiencia del cliente como en la productividad del negocio. La falta de un adecuado manejo de los registros limita la capacidad de realizar un seguimiento preciso del inventario de los productos, gestionar las ventas y analizar el rendimiento de manera más efectiva y clara.

Uno de los principales problemas que enfrenta la dulcería es la dificultad para mantener un control adecuado del inventario, en varias ocasiones el desperdicio de mercancía valiosa o la acumulación excesiva de dulces caducados. Esto no solo afecta las ventas, sino que también disminuye la satisfacción de los compradores al no encontrar lo que buscan en buen estado. Asimismo, el registro de ventas, clientes, productos y empleados se realiza manualmente, lo que aumenta el riesgo de errores por parte del personal, la pérdida de información importante y la dificultad para acceder a datos actualizados o históricos. En momentos de alta demanda, estos problemas se agravan, generando largas filas de espera, retrasos en las transacciones y una experiencia de compra menos satisfactoria para los clientes, lo que puede ocasionar la pérdida de ventas.

Por otro lado, el registro y la gestión de los empleados también presentan múltiples inconvenientes, ya que no se cuenta con un sistema adecuado para llevar un control preciso de la información, lo que incrementa la posibilidad de errores administrativos. Esta falta de organización interna dificulta la toma de

decisiones y la identificación de áreas de mejora en las operaciones de la dulcería.

Ante este panorama, surge la necesidad de diseñar e implementar un sistema que permita a la dulcería registrar y controlar de manera más eficiente la información de productos, empleados, clientes y ventas diarias. Este sistema facilitará la organización interna y reducirá los errores operativos generados por los procesos manuales, lo que contribuirá a mejorar la competitividad del negocio y la satisfacción de sus clientes.

El problema central radica en que, sin un sistema adecuado, la dulcería continuará enfrentando estos desafíos, lo que podría impactar negativamente en su desempeño y sostenibilidad en el mercado, incluso hasta el punto de poner en riesgo su continuidad. Por ello, este proyecto busca desarrollar una solución tecnológica que aborde estas necesidades específicas y permita el negocio pueda operar con mayor eficiencia y éxito, evitando así su posible cierre y fomentando su crecimiento.

Pregunta de investigación

¿Cómo el desarrollo e implementación de un sistema de gestión basado en Visual Studio Code, XAMPP, Bootstrap y MariaDB puede optimizar los procesos internos de una dulcería, mejorando la administración del inventario, la gestión de información y la eficiencia operativa?

VI. MARCO TEÓRICO

En el presente trabajo se realiza el diseño e implementación de un sistema para la administración de una dulcería, como lo que es el manejo de información personal de empleados y clientes, así como los movimientos de sus productos y ventas. Por lo tanto, es importante mencionar algunos conceptos desde los más básicos hasta más avanzados que hablen acerca de la ingeniería en computación, desde como inicia hasta como se llega a relacionar con este trabajo.

6.1 Historia y evolución de la computación

En este apartado se desarrollará un resumen acerca de la historia de la computación, destacando los principales hitos desde sus orígenes hasta su impacto en el mundo moderno. A través del tiempo, la computación ha evolucionado desde simples maquinas mecánicas hasta complejos sistemas capaces de realizar millones de operaciones por segundo.

Orígenes de la computación

Ceruzzi (2012) menciona que uno de los primeros intentos de mecanizar los procesos de cálculo fue la Maquina de Pascal, desarrollada en el siglo XVII por Blaise Pascal. Donde este dispositivo permitía realizar sumas y restas automáticamente, facilitando los cálculos matemáticos de esa época. Después, en el siglo XIX Charles Babbage diseño la Maquina Analítica, considerada como un antecedente de las computadoras modernas, ya que estaba más completa y contenía conceptos como memoria, una unidad aritmética y tenía la capacidad de programación mediante tarjetas perforadas.

Primeras computadoras y su impacto en el mundo

El desarrollo de la computación tuvo un gran avance en el siglo XX con la creación de los primeros ordenadores electrónicos. La computadora ENIAC (Electronic Numerical Integrator and Computer), fue desarrollada en el año 1946, es considerada una de las primeras computadoras electrónicas de propósito general. Esta máquina fue construida en la Universidad de Pensilvania, donde utilizaba tubos de vacío para procesar información y era capaz de realizar miles de operaciones por segundo.

Sebesta (2020) menciona que durante las décadas de 1950 a 1960 la evolución de la computación estuvo marcada por la invención del transistor y los circuitos integrados los cuales fueron el reemplazo de los tubos de vacío y permitieron reducir el tamaño de los equipos, mejorar su eficiencia y aumentar su capacidad de procesamiento. Por lo que, en ese mismo periodo surgieron los primeros lenguajes de programación como FORTRAN y COBOL los cuales facilitaron la escritura de programas para distintas tareas y propósitos.

Tomando en cuenta que el impacto de las primeras computadoras en el mundo fue algo significativo en diversos ámbitos, especialmente en la ciencia, la industria y en el área militar. Donde la computación permitió mejorar la capacidad de cálculo en las investigaciones científicas, optimizar la producción en fábricas y en la creación de sistemas de defensa avanzados. Hoy en día, la computación sigue avanzando con tendencias como lo que es la IA, la computación en la nube y la computación cuántica, tecnologías como asistentes virtuales, aprendizaje automático y la automatización de procesos ha ido evolucionando distintos sectores como en el área de la medicina hasta la economía digital. Así también en el ámbito de los negocios, como el desarrollo de software y sistemas de información que ha permitido la optimización de procesos, lo que demuestra el avance tecnológico en la actualidad.

6.2 Ingeniería en Computación

La Ingeniería en Computación es una disciplina que combina principios de la ingeniería y la informática para el diseño, desarrollo, implementación y mantenimiento de sistemas computacionales y software. Según Tanenbaum (2019) menciona que la ingeniería en computación abarca tanto el hardware como el software, así permitiendo la creación de soluciones tecnológicas eficientes y escalables, donde su principal objetivo es mejorar la automatización de procesos en diversas áreas industriales, comerciales y científicas.

6.2.1 Evolución de la carrera y su impacto en la tecnología

Ceruzzi (2018) menciona que la ingeniería en computación tiene sus raíces en el desarrollo de las primeras computadoras en la década de 1940, donde la tecnología estaba muy limitada a laboratorios militares y a universidades, dado que utilizaban la tecnología para el cálculo de trayectorias balísticas y para simulaciones matemáticas complejas. A medida que la tecnología fue evolucionando también incremento la necesidad de profesionales especializados para su desarrollo y su mantenimiento.

En la década de 1960, la ingeniería en computación comenzó a consolidarse como una carrera formal en diversas universidades del mundo. Ya que el crecimiento de la industria microelectrónica y el desarrollo de los primeros procesadores fue durante las décadas de los 70 y 80, esto impulsó la demanda de ingenieros en computación. Como menciona Levy (2021) que, con la aparición del internet y la globalización de la información, la ingeniería en computación expandió muchos campos como la IA, la ciberseguridad y la computación en la nube.

Schneider (2022) describe como la ingeniería en computación ha ido tomando un papel muy importante últimamente, donde menciona que hoy en día, la ingeniería en computación desempeña un papel fundamental en la

transformación digital de las empresas, así facilitando la automatización de procesos, el análisis de grandes volúmenes de datos (Big Data) y el desarrollo de tecnologías emergentes como la realidad aumentada y la computación cuántica.

Por otro lado, Brookshear & Brylow (2020) también mencionan que, en el desarrollo de sistemas, la ingeniería en computación interviene en el diseño y mantenimiento de redes, base de datos, sistemas operativos y plataformas en la nube. Que, gracias a esto, se ha logrado la automatización de procesos en sectores como la salud, la educación, la industria manufacturera y el comercio electrónico.

Y por último Fowler (2021) menciona que, en el ámbito del desarrollo de software, los ingenieros en computación se encargan del diseño de arquitecturas de software escalables, el análisis de requisitos, la programación en distintos lenguajes y la implementación de metodologías ágiles para la gestión de proyectos. Además, contribuyen a la creación de sistemas de información seguros, aplicando estrategias de ciberseguridad y encriptación de datos.

Por lo que, yo estoy de acuerdo con los tres autores que mencionan como la ingeniería en computación ha conseguido un papel muy importante hoy en día. Donde tiene una amplia aplicación en el desarrollo de software y sistemas de información, donde, unos de sus principales enfoques es la creación de software que optimice el desempeño de los sistemas computacionales así mejorando la interacción entre el humano y la tecnología, influyendo en diversas áreas tecnológicas y facilitando el desarrollo de soluciones innovadoras ya que permite la automatización y optimización de procesos, mejorando la eficiencia y la productividad en distintos sectores.

6.3 Programación

La programación es el proceso de crear conjunto de instrucciones, también conocido como código, para que una computadora pueda entender y ejecutar diferentes tareas específicas. Es como darle a la computadora una receta con instrucciones detalladas para que pueda preparar un platillo (ejecutar una tarea). Dado que se utiliza una variedad de lenguajes de programación para escribir estas instrucciones como Python, Java, C++, etc.

Markoff (2020) redacta que la programación de las computadoras tiene sus raíces en el siglo XIX, cuando Ada Lovelace diseñó el primer algoritmo pensado para ser ejecutado en una máquina, como, por ejemplo, la Máquina Analítica de Charles Babbage, aunque esta nunca se construyó, su trabajo sentó las bases de la programación moderna.

Donde Pressman (2022) menciona que el desarrollo de lenguajes de programación de alto nivel comenzó en la década de los 50 con la creación de FORTRAN (Formula Translation) diseñado por IBM para la computación científica y matemática. Donde al mismo tiempo se lanzó COBOL (Common Business-Oriented Language) enfocado en aplicaciones comerciales, estos lenguajes marcaron el inicio de la programación estructurada y facilitaron el desarrollo de software más complejo.

A medida que la computación avanzó, también lo hicieron los lenguajes de programación. En las décadas de los 60 y 70 surgieron lenguajes como por ejemplo "C", el cual influyó en muchos de los lenguajes modernos debido a su flexibilidad y eficiencia.

La programación es el motor de la computación moderna, permitiendo la creación de software y de sistemas que facilitan la vida cotidiana. Desde aplicaciones móviles hasta sistemas de IA. Ya que la programación está presente en diferentes áreas como la salud, la economía, la educación y la investigación científica, donde su importancia radica en la capacidad de poder transformar ideas en soluciones tecnológicas funcionales.

En la actualidad Lutz (2023) menciona que, la tendencia apunta hacia el uso de lenguajes más intuitivos, con sintaxis simplificada y herramientas que facilitan el desarrollo de software. Lenguajes de programación como Python han ganado la popularidad en IA y en la ciencia de datos debido a su facilidad de uso y bibliotecas especializadas. Asimismo, el paradigma de la programación funcional con lenguajes como Scala y Rust, está cobrando relevancia en aplicaciones que requieren alta eficiencia y concurrencia.

6.4 Paradigmas de programación

Es un enfoque fundamental para la resolución de problemas mediante la programación. Por lo que, es una manera de organizar y estructurar el código para lograr un objetivo específico. Cada paradigma define cómo se ve el código, cómo se organiza y cómo interactúa con los datos. Piensa en ellos como diferentes "filosofías" o "modelos" para abordar la programación.

Sebesta (2021) en su análisis menciona que, un paradigma de programación es un estilo fundamental para la resolución de problemas mediante esta área. Ya que representan un conjunto de principios, conceptos y metodologías que guían la forma en que los desarrolladores estructuran y escriben el código de un programa. Estos han evolucionado con el tiempo, pudiéndose adaptar a las necesidades de la computación y la tecnología.

A lo largo de la historia, han surgido diversos paradigmas de programación, cada uno con sus ventajas y desventajas, algunos más adecuados para ciertos tipos de problemas o aplicaciones y otros no. Es común que los lenguajes modernos integren elementos de más de un paradigma para ofrecer mayor flexibilidad y eficiencia en el desarrollo de software.

A continuación, en la **Figura 1** se muestran algunos tipos de paradigmas de programación.



Figura 1. Paradigmas de programación

Fuente: IONOS (2020)

Programación imperativa

Digital Guide (2020) describió que la programación imperativa como el paradigma clásico, así como los primeros programas informáticos se basan en este paradigma, ya que proporciona una secuencia regularizada de ordenes determinadas. Por lo que, el código resultante es fácil de comprender y muy abarcable.

Programación estructurada

Es un paradigma basado en el uso de estructuras de control de flujo, como secuencias, selecciones e interacciones. Dado que su objetivo principal es mejorar la claridad y mantenibilidad del código al eliminar el uso excesivo de instrucciones de salto como GOTO, así promoviendo en su lugar una organización jerárquica del código.

Programación procedimental

El sentido y el propósito de la programación procedimental es hacer que el código del programa sea más claro y evitar las repeticiones innecesarias de código.

Mediante la abstracción de los algoritmos, el paradigma de software procedimental representa un paso decisivo desde los lenguajes ensambladores simples hasta los lenguajes estándar complejos.

Programación modular

Con relación a la programación modular es una adaptación del enfoque procedimental para requerimientos de proyectos de softwares más amplios. El código fuente se divide específicamente en bloques parciales lógicos independientes los unos de los otros para proporcionar más transparencia y facilitar el proceso de debugging (resolución de errores).

Programación declarativa

Se centra en el resultado que se busca. Por lo tanto, en primera línea se encuentra el “qué” del resultado y no el “cómo” de los pasos que llevan a la solución, para poder comprender el código es complicado ya que se requiere un alto grado de abstracción, aunque resulta muy corto y preciso.

Programación lógica

Es conocida como programación predictiva, así mismo se basa en la lógica matemática. Un software programado según este inicio contiene un conjunto de principios que se pueden entender como una recopilación de hechos y suposiciones, en lugar de una sucesión de instrucciones. Todas las solicitudes al programa se procesan de forma que el intérprete recurre a estos comienzos para posteriormente aplicar reglas definidas previamente para alcanzar el resultado deseado.

Programación funcional

Un programa de programación funcional consta principalmente de llamadas de función concatenadas en las que cada parte del programa se interpreta como una función.

Programación orientada a objetos (POO)

Booch (2022) menciona que es un paradigma basado en la representación de los datos y sus operaciones a través de objetos. Un objeto es una entidad que combina datos (atributos) y comportamientos (métodos) dentro de una misma estructura. La POO promueve la reutilización del código y la modularidad mediante conceptos fundamentales como la encapsulación, herencia y el polimorfismo.

Para terminar con los paradigmas de programación, se toma en cuenta que, cada uno de ellos ofrece un enfoque distinto para la resolución de problemas computacionales. Donde la elección del paradigma adecuado depende del tipo de tarea o aplicación que se desea desarrollar, así como los requisitos de escalabilidad, mantenibilidad y rendimiento. En la actualidad, muchos lenguajes de programación pueden combinar elementos de distintos paradigmas para ofrecer una mayor flexibilidad para los desarrolladores.

6.5 Clasificación de los lenguajes de programación

Los lenguajes de programación son un conjunto de instrucciones utilizadas para que el hardware y el software puedan tener comunicación entre sí y así poder realizar las tareas que se le asignen. Estos lenguajes han evolucionado a lo largo del tiempo, pudiéndose adaptar a las necesidades de los desarrolladores y a los avances tecnológicos de hoy en día. Por lo que, existen diversas formas de poder clasificarlos dependiendo sus características y su nivel de abstracción. En seguida, se presentan las principales categorizaciones de los lenguajes de programación.

Lenguajes de bajo y alto nivel

Los lenguajes de programación se pueden clasificar en dos categorías según su nivel de abstracción respecto al hardware, donde la primera categoría se encuentran los lenguajes de bajo nivel y en la segunda categoría se encuentran los lenguajes de alto nivel. Ahora se presentan las especificaciones de cada categoría.

Lenguajes de bajo nivel

Los lenguajes de bajo nivel son aquellos que están más cerca del hardware y requieren un conocimiento profundo de la arquitectura de la máquina para poder ser utilizados. Ya que son más eficientes en términos de rendimiento, pero son más difíciles de programar y mantener, donde estos lenguajes incluyen lo siguiente:

- Lenguaje de máquina: Es la que está compuesta por instrucciones binarias, en este caso 0 y 1 que la CPU interpreta directamente.
- Lenguaje ensamblador: Es la que utiliza representaciones simbólicas de las instrucciones de máquina, lo que facilita ligeramente su comprensión en comparación con el lenguaje de máquina.

Lenguajes de alto nivel

Sebesta (2020) menciona que los lenguajes de alto nivel son más comprensibles para los humanos y permiten escribir código de una manera más intuitiva. Donde utilizan estructuras y sintaxis más cercanas al lenguaje natural y requieren de un proceso de traducción a código de máquina a través de compiladores o intérpretes como por ejemplo estos lenguajes incluyen: Python, Java, C++, y JavaScript.

Lenguajes compilados e interpretados

Otra manera de clasificar estos lenguajes de programación es según la forma en que su código fuente es procesado para ser ejecutado por la computadora. Donde se dividen en dos, lenguajes compilados y lenguajes interpretados.

- **Lenguajes compilados:** Son los que requieren un programa denominado compilador, el cual traduce el código fuente a un archivo ejecutable antes de que pueda ser ejecutado por la computadora, por ejemplo, estos lenguajes incluyen: C, C++, y Go.
- **Lenguajes interpretados:** Son los que ejecutan el código fuente línea por línea a través de un intérprete, sin la necesidad de generar un archivo ejecutable. Ya que son más flexibles y portátiles, pero lentos en rendimiento como, por ejemplo: Python, JavaScript y PHP.

6.6 Ejemplos de lenguajes de programación

Existen varios lenguajes de programación, donde cada uno fue creado y diseñado para resolver diferentes tipos de tareas y problemas específicos, donde a continuación, se muestran los lenguajes de programación más populares en la actualidad.

Lenguaje C

El lenguaje C fue creado en el año 1972 por Dennis Ritchie en los laboratorios de Bell de AT&T. Fue desarrollado como una evolución del lenguaje B, con el propósito de poder mejorar la portabilidad y eficiencia de los programas. Por lo que C es un lenguaje de programación de propósito general que permite manipulación de memoria de bajo nivel y programación estructurada, donde es más utilizado en el desarrollo de sistemas operativos, software embebido y aplicaciones de alto rendimiento, considerado uno de los lenguajes más influyentes.

Lenguaje C++

Martínez (2020) indica que el lenguaje C++ fue creado en el año 1983 por Bjarne Stroustrup creado como una extensión del lenguaje C, pero incorporando el paradigma de programación Orientada a objetos. Dado que, su objetivo es combinar la eficiencia y control del lenguaje C con nuevas características como lo que es la encapsulación, herencia y polimorfismo. Actualmente este lenguaje se utiliza en sistemas de alto rendimiento, motores de videojuegos, software financiero y aplicaciones que requieren acceso diferente a los recursos de hardware.

Lenguaje Java

Camacho (2023) menciona que este lenguaje Java fue creado en el año 1995 por James Gosling mientras trabajaba en Sun Microsystems, que hoy en día esta empresa pertenece a Oracle. Menciona que es un lenguaje orientado a objetos, diseñado específicamente para ser portátil, seguro y altamente escalable. Donde su lema es "Write Once, Run Anywhere" lo que significa (Escribe una vez, ejecuta en cualquier lugar) esto se debe a que sus programas se ejecuten en JVM (Java Virtual Machine), lo que esto permite que sean independientes del sistema operativo. Hoy en día, se utiliza en aplicaciones empresariales, desarrollo de aplicaciones móviles en especial Android, para sistemas de servidores, entre otras más.

Lenguaje Python

El lenguaje Python fue lanzado en el año 1991 por Guido van Rossum. Como un lenguaje de alto nivel, con una sintaxis sencilla y legible. El cual, su objetivo era proporcionar un lenguaje fácil de aprender y de poder usar. Por lo que, actualmente es muy popular en la IA, análisis de datos, desarrollo web, automatización de tareas y ciberseguridad.

JavaScript

Este lenguaje fue creado en el año 1995 por Brendan Eich mientras trabajaba en la empresa Netscape Communications. Ya que se diseñó originalmente para poder agregar dinamismo a las páginas web y así permitiendo la manipulación de HTML y CSS en el navegador. Por lo que con el tiempo ha ido evolucionando para poder ser utilizado en el desarrollo web tanto en front-end como en back-end, donde hoy en día es uno de los lenguajes más utilizados en esta área de desarrollo web.

Lenguaje PHP

El lenguaje PHP fue creado en el año 1994 por Rasmus Lerdorf como un conjunto de scripts para la gestión de páginas digitales. Con el tiempo fue muy utilizado en el desarrollo web dinámico. Ya que su uso principal es la creación de sitios y aplicaciones web, destacándose en la programación de servidores y gestión de base de datos utilizando herramientas como MySQL.

Lenguaje Swift

Un lenguaje de programación moderno creado por la empresa de Apple. Es un lenguaje creado en el año 2014 para la creación de aplicaciones en su ecosistema como lo que viene siendo (iOS, macOS, WatchOS y tvOS). Lo que lo hace un lenguaje seguro, rápido y fácil de entender, asimismo fue diseñado para reemplazar a Objective-C en el desarrollo de aplicaciones para sus dispositivos.

Lenguaje Go (Golang)

Fue creado en el año 2009 por Google, con el fin de ofrecer un lenguaje eficiente y fácil de compilar. Es utilizado en desarrollo de infraestructura, servidores web, sistemas en la nube y aplicaciones escalables gracias a su capacidad de manejar múltiples tareas de una manera más eficiente

Lenguaje Kotlin

Este lenguaje fue lanzado en el año 2011 y creado por JetBrains. Lanzado como una alternativa moderna para Java para el desarrollo de aplicaciones Android. En el año 2017 Google lo adoptó como el lenguaje oficial, ya que este lenguaje ofrece una sintaxis más concisa, segura y eficiente que Java, lo que esto lo ha llevado a su creciente popularidad hoy en día en el desarrollo móvil.

6.7 La importancia de la programación en el desarrollo de sistemas

Tomando en cuenta que la programación es una disciplina fundamental para el desarrollo de software y en los sistemas informáticos. Se puede definir como el proceso mediante el cual se diseñan y escriben instrucciones que una computadora puede interpretar y ejecutar para que pueda realizar ciertas tareas específicas. Su importancia radica es que la programación es la base donde se construyen soluciones tecnológicas, como aplicaciones móviles hasta sistemas empresariales más complejos y así permitiendo la automatización de procesos, de recursos y el mejoramiento en la toma de decisiones.

Rol de la programación en la creación del software

El desarrollo de software se sujeta en la programación, ya que, sin la implementación de código, los sistemas no servirían de nada, porque no tendrían una función o tarea específica para realizar. En vista de que la programación es la que permite transformar los requerimientos de los usuarios para que sus aplicaciones sean funcionales, siguiendo metodologías y modelos de desarrollo como por ejemplo el ciclo de vida del software. Como indica Sommerville (2021) que, dentro de este proceso, los programadores emplean diversos lenguajes de programación y paradigmas para estructurar el código de una manera más eficiente y escalable y así asegurando el correcto funcionamiento del software en distintos entornos y dispositivos.

Diferencia entre software a medida y software comercial

Considerando que se puede clasificar en dos categorías llamadas: software a medida y comercial. Cada categoría cumple ciertas funciones específicas cada una. A continuación, se especifica cada una de ellas, la cual será detallada posteriormente.

Pressman (2020) menciona que el software a medida es aquel que se desarrolla específicamente para cubrir las necesidades de una empresa o usuario particular. Se diseña con funcionalidades personalizadas y se adapta a los requerimientos específicos del negocio, permitiendo una mayor flexibilidad y optimización de los procesos internos, donde un ejemplo de este tipo de software es un sistema de gestión empresarial que es diseñado para una compañía en particular. Por otro lado, Sommerville (2020) redacta que el software comercial es el que está diseñado para ser utilizado por un amplio grupo de usuarios y suele ofrecer soluciones estándar sin modificaciones personalizadas. Estos programas se comercializan en el mercado y pueden ser adquiridos por cualquier usuario o empresa. Ejemplos de estos sistemas son: Microsoft Office o sistemas operativos como Windows y macOS. En resumen, es que la diferencia es que el software de medida ofrece soluciones específicas y flexibles para una empresa en particular, mientras que el comercial es una solución estándar utilizada por muchas organizaciones sin modificaciones sustanciales.

Programación en los negocios

El impacto que tiene la programación en los negocios es muy significativo, ya que en los estudios de McKinsey Company (2022) mencionan que la programación permite la digitalización y la automatización de procesos empresariales, donde las empresas pueden optimizar la gestión de recursos, mejorar la comunicación interna y externa, y aumentar la eficiencia operativa. Hoy en día, muchas empresas han adoptado soluciones tecnológicas basadas en programación para

mejorar su competitividad en el mercado implementando sistemas de gestión, comercio electrónico y aplicaciones móviles.

6.8 Ciclo de vida del desarrollo de software (SDLC)

El ciclo de vida del desarrollo de software o también conocido por sus siglas en inglés SDLC (Software Development Life Cycle) es un proceso estructurado la cual define las etapas a seguir para poder desarrollar un sistema de una manera eficiente y organizada. Dado que este ciclo permite garantizar la calidad del producto final y optimizar los recursos involucrados en su desarrollo.

Modelos de desarrollo de software

Son una representación abstracta de un proceso de software, que se utiliza para describir sistemas, ya que se crean para facilitar la comunicación entre las partes interesadas y ayudar en el análisis del diseño. Tomando en cuenta que existen varios tipos de modelos que estructuran la vida de estos sistemas, donde cada uno con características particulares en distintos contextos. Mas adelante, se muestran los diferentes tipos de modelos detallados posteriormente:

Modelo en Cascada

Es uno de los modelos más tradicionales y se basa en un enfoque secuencial, conformada por las siguientes fases: Requerimientos, diseño, implementación, pruebas, despliegue y mantenimiento. Donde cada fase del desarrollo se ejecuta en orden sin la posibilidad de retroceder a la fase anterior.

Modelo Espiral

Es el modelo que combina características del modelo en cascada con un enfoque iterativo, permitiendo varias revisiones constantes en cada fase del desarrollo.

Donde esto es útil para proyectos grandes con riesgos elevados y su principal desafío es la complejidad y el costo asociado.

Modelo Ágil

Es el modelo que se enfoca en el desarrollo iterativo e incremental, permitiendo una entrega continua de software funcional en pequeños ciclos de trabajo denominados “Sprints”, lo que lo hace altamente flexible y adaptable. Los métodos como Scrum y Kanban se basan en este modelo.

6.9 Metodologías ágiles en el desarrollo de software

Las metodologías ágiles en el desarrollo de software han surgido como una respuesta a la rigidez de los métodos tradicionales antes vistos, así, promoviendo la flexibilidad, la colaboración y la entrega continua del valor. Estas metodologías permiten adaptarse a los cambios en los requerimientos de los clientes y optimizar el proceso de desarrollo de software. Ya que existen varias metodologías y claramente cada una con diferentes características y formas de llevar un buen control del desarrollo de software, donde a continuación se explicará con más detalle acerca de cada una.

Metodología Scrum

La metodología Scrum es una de las más utilizadas en el desarrollo de software, como mencionan Schwaber y Sutherland (2020) esta metodología se basa en ciclos de trabajo llamados “Sprints”, que suelen durar entre dos o cuatro semanas, ya que, durante ese tiempo, un equipo de personas trabaja en la entrega de incrementos funcionales del software. Por lo que, en este enfoque los roles principales incluyen el Product Owner, el cual, es el que gestiona los requisitos y las prioridades, el Scrum Master que es el que facilita el proceso y elimina los obstáculos y por último, el equipo de desarrollo, que son los que

implementan las funcionalidades requeridas, Tomando en consideración, que sus eventos claves incluyen la planificación del sprint, donde se define el trabajo que se va a realizar, las reuniones diarias de seguimiento en las que el mismo equipo revisa durante su progreso, la revisión del sprint, demostración de avances al cliente y la retrospectiva del cliente en caso de requerir nuevas mejoras o sugerencias.

Metodología Kanban

Como menciona Anderson (2021) Kanban es otra metodología ágil que se centra en la visualización del flujo de trabajo a través de un tablero dividido en columnas para distintas tareas las cuales son las siguientes: (Pendiente, En progreso y Completado) donde este enfoque permite a los equipos identificar cuellos de botella y así poder mejorar la eficiencia del proceso para el desarrollo de software. Por lo tanto, a diferencia de Scrum, Kanban no requiere de Sprints fijos ni de roles específicos, sino que solo se enfoca en la mejora continua mediante la limitación del trabajo que hay en curso y la optimización del flujo de entrega. Esta metodología es fácil de usar para proyectos donde se requiera de una alta flexibilidad y en equipos que trabajan con entregas muy constantes.

Beneficios de una metodología ágil en proyectos de software

Fowler (2021) menciona algunos beneficios al usar una metodología ágil en un proyecto de software donde se muestran a continuación.

- **Mayor flexibilidad:** Una mayor flexibilidad permite que se pueda adaptar más a los cambios en los requerimientos sin afectar el desarrollo.
- **Entrega continua de valor:** Aquí es donde los clientes reciben funcionalidades utilizables en intervalos de poco tiempo.
- **Mayor colaboración:** Es la comunicación que se da entre los miembros del equipo y los interesados en el proyecto.

- **Reducción de riesgos:** Es donde se identifican y se corrigen rápidamente los problemas que pueden llegar a tener.
- **Mejora la calidad del software:** Es donde gracias a las pruebas frecuentes y la retroalimentación aseguran que el software cumpla con los requisitos requeridos.

6.10 Frameworks

Según Pérez (2022) un framework es una estructura de trabajo que proporciona un conjunto de herramientas, bibliotecas y mejores prácticas para poder facilitar el desarrollo de software. Su principal objetivo es simplificar la implementación de soluciones al poder ofrecer una base estandarizada sobre la cual construir aplicaciones o páginas web. Puesto que, los frameworks permiten que los desarrolladores se centren más en la funcionalidad de la aplicación o página sin la necesidad de preocuparse por aspectos técnicos de bajo nivel, como lo que es la gestión de memoria o la comunicación con base de datos. Ya que suelen incluir varios componentes esenciales para el desarrollo de software como:

- **Bibliotecas de funciones:** Son conjuntos de códigos reutilizables que facilitan tareas comunes.
- **Gestores de dependencias:** Son las herramientas que permiten administrar paquetes externos.
- **Estructuras predefinidas:** Es la organización de archivos y código para mantener buenas prácticas de programación.
- **Herramientas de prueba y depuración:** Son los que facilitan la identificación y corrección de errores en el mismo código.

Ventajas de usar frameworks

López (2021) menciona que el uso de frameworks en el desarrollo de software ofrece diversas ventajas y menciona unas de ellas con la especificación de cada una como se muestra a continuación.

1. **Eficiencia y rapidez en el desarrollo:** Al momento de proporcionar estructuras y funciones predefinidas, los mismos frameworks reducen el tiempo necesario para escribir el código desde cero.
2. **Mantenimiento y escalabilidad:** Aquí es cuando al seguir estándares bien definidos, los frameworks facilitan la ampliación de proyectos y la colaboración entre desarrolladores.

6.11 Frameworks más utilizados

Debido a que existen diversos frameworks especializados para distintos propósitos cada uno en el desarrollo de software. Mas adelante, se muestran los frameworks más utilizados en el desarrollo web y de aplicaciones.

1. Frameworks para desarrollo web
 - **Bootstrap:** Este framework facilita el diseño de interfaces responsivas mediante componentes CSS y JavaScript predefinidos.
 - **Angular:** Fue desarrollado por Google, el cual, permite la creación de aplicaciones web escalables con una arquitectura basada en componentes.
 - **React:** Es una librería desarrollada por Facebook, el cual, optimiza la construcción de interfaces de usuario con un enfoque declarativo.
 - **Vue.js:** Es un framework progresivo que combina con facilidad de uso con potentes funcionalidades para aplicaciones interactivas.

2. Framework para desarrollo de aplicaciones

- **Django:** En un framework de Python que permite el desarrollo rápido de aplicaciones web seguras y escalables.
- **Spring Boot:** Es un framework basado en Java, lo cual, facilita la creación de aplicaciones empresariales con arquitecturas robustas.
- **Laravel:** Un framework de PHP que simplifica el desarrollo web con una sintaxis elegante y herramientas de autorización.
- **Flask:** Es otro framework de Python, más ligero que Django, ya que es ideal para aplicaciones web pequeñas y medianas.

6.12 Base de datos (BD)

Son conjuntos organizados que permiten el almacenamiento, la gestión y la recuperación eficiente de la información. Considerando que la BD, es un sistema estructurado que facilita la administración de grandes volúmenes de datos y así asegurando su seguridad y su disponibilidad. Contienen entidades y estas a su vez atributos en donde se guardan diferentes tipos de datos, esto con la finalidad de evitar la mayor redundancia posible. Para posteriormente esos datos convertirlos a información que se puede consultar en algún momento determinado. Existen diferentes tipos de bases de datos, cada una con características diferentes, de acuerdo con las necesidades de los usuarios. A continuación, se muestran los diferentes tipos de ellos.

La importancia de estas bases como menciona Connolly (2021) radica en su capacidad para manejar datos estructurados de una manera eficiente, facilitando la organización de grandes cantidades de información y permitiendo el acceso a los datos de forma rápida y segura. Además, ofrecen soporte para diversas aplicaciones empresariales, desde sistemas de gestión empresarial hasta plataformas de comercio electrónico y redes sociales.

Base de datos Estáticas

Juárez (2021) menciona que las bases de datos estáticas son aquellas cuyo diseño está enfocado a la lectura de datos, donde su principal función es almacenar y registrar datos con el objetivo de realizar estudios históricos y comparativos para mejorar la toma de decisiones.

Base de datos Dinámicas

En las bases de datos dinámicas no solo se almacena la información, sino que se va modificando a lo largo del tiempo cuando es necesario.

6.13 Modelos de base de datos

Existen diversos modelos de base de datos que se utilizan, dependiendo las necesidades de cada aplicación. Los modelos principales es el modelo relacional y el modelo no relacional (NoSQL). Por lo que, a continuación de especifican cada uno de ellos.

Modelo relacional

Es un modelo propuesto por Edgar F. Codd en 1970, es el que organiza los datos en las tablas estructuradas que establecen relaciones entre sí. Por lo que, este modelo se basa en algebra relacional y permite la manipulación de datos mediante lenguajes de consulta estructurados como SQL (Structured Query Language). Se utilizan ampliamente en aplicaciones empresariales, financieras y de gestión, ya que ofrecen integridad referencial, consistencia de datos y la facilidad para realizar consultas complejas. Existen varios ejemplos de estas bases de datos relacionales incluyen MySQL, PostgreSQL, Oracle Database, y Microsoft SQL Server.

Modelo no relacional

Surgió como una alternativa al modelo relacional, especialmente para gestionar grandes volúmenes de datos no estructurados o semi-estructurados. Por lo que, este modelo no se basa en tablas y ofrece mayor flexibilidad en la organización y almacenamiento de la información.

Jerárquicas

Estas bases de datos es donde la información se clasifica siguiendo una estructura jerárquica como se muestra en la **Figura 2**. Debido a que la estructura jerárquica es en forma de árbol, donde comienza por una raíz de la que surgen distintos nodos llamados “padres” y desde cada uno de ellos surgen nuevos nodos llamados “hijos” y así sucesivamente.

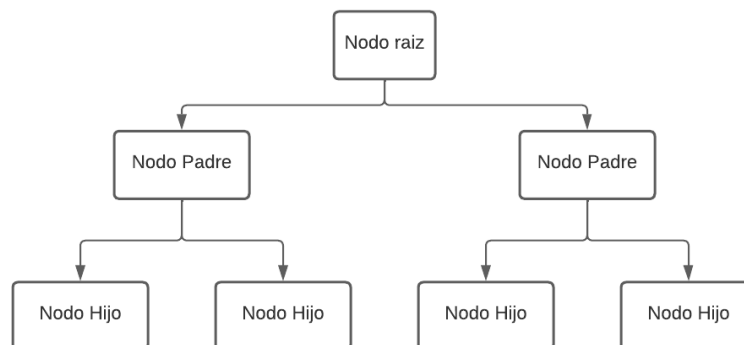


Figura 2 Estructura jerárquica.

Fuente: Ortiz (2020)

De red

Se trata de un tipo de BD especial de la base de datos jerárquica, donde cada nodo puede tener distintos “padres”. Por lo que se tiene acceso a las entidades por diferentes caminos y esto hace que la búsqueda sea más rápida. Ya que es utilizada por programadores, aunque eso implique una dificultad a la hora de administrar la base de datos como se muestra en la **Figura 3**.

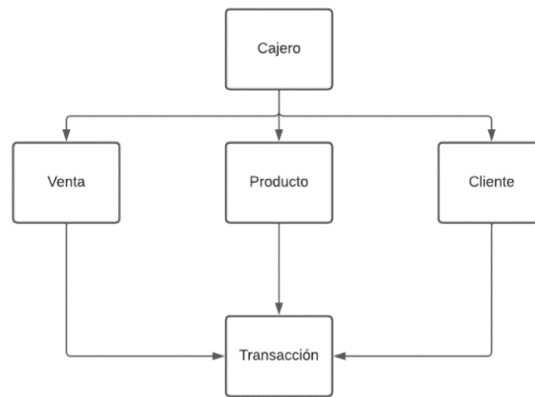


Figura 3 Base de datos - De red

Fuente: Sánchez (2023)

Orientadas a objetos

Por último, estas no almacenan datos sobre un objeto, sino que guardan el objeto en sí mismo. La ventaja de utilizar este tipo de BD, es que admite mayor contenido que otros tipos de base de datos, facilitando al usuario el acceso a información actualizada y completa.

6.14 Visual Studio Code como entorno de desarrollo

Visual Studio Code (VS Code) es un entorno de desarrollo integrado (IDE), donde es ligero y multiplataforma desarrollado por Microsoft. Ya que se caracteriza por ser altamente personalizable y ofrecer soporte para múltiples lenguajes de programación, lo que lo convierte en una herramienta versátil para desarrolladores de software.

Principales ventajas

Existen varias ventajas de usar esta herramienta para el desarrollo de software las cuales Microsoft (2023) las describe de la siguiente manera.

- **Interfaz intuitiva y personalizable:** VS Code permite modificar su apariencia, cambiar temas y ajustar la disposición de los paneles según las preferencias del usuario.

- **Soporte para múltiples lenguajes:** Es compatible con lenguajes como: JavaScript, Python, C++, Java, PHP, entre otros lenguajes. Así facilitando el desarrollo en distintos entornos.
- **Integración con Git:** Incluye soporte integrado para Git, permitiendo el control de versiones sin necesidad de herramientas externas.
- **Depuración eficiente:** Ya que cuenta con herramientas avanzadas de depuración que permiten detectar errores y poder mejorar el rendimiento del código.
- **Ligereza y rendimiento:** Ya que, a diferencia de otros entornos de desarrollo más pesados, VS Code es rápido y consume menos recursos del sistema.
- **Comunidad activa y actualizaciones constantes:** Microsoft y la comunidad de desarrolladores trabajan continuamente en la mejora de esta herramienta, así proporcionando actualizaciones frecuentes con nuevas funciones y correcciones de errores.

6.15 Bootstrap como herramienta de diseño responsivo

Bootstrap es un framework de desarrollo web de código abierto. El cual facilita la creación de interfaces responsivas y modernas. Considerando que fue diseñado por Twitter y fue lanzado en el año 2011, con el objetivo de ofrecer un conjunto de herramientas predefinidas que permitieran que los trabajadores puedan construir sitios digitales con diseños atractivos y adaptables a diferentes dispositivos.

Se compone de una serie de archivos CSS, JavaScript y una amplia biblioteca de componentes reutilizables que simplifican la maquetación y el diseño de páginas web. A continuación, se especifican los componentes y beneficios que este framework tiene al ser utilizado como una herramienta de diseño, Bootstrap (2023).

Componentes principales.

- **Sistema de rejilla (Grid System):** Permite organizar el contenido en filas y columnas, facilitando el diseño responsivo mediante una estructura flexible.
- **Componentes UI:** Aquí en donde incluye, botones, formularios, alertas, tarjetas, barras de navegación y otros elementos que pueden integrarse de manera sencilla en cualquier diseño.
- **Utilidades CSS:** Ofrece clases predefinidas para el manejo de espaciados, colores, alineaciones y tipografía, así permitiendo una personalización rápida.
- **Compatibilidad con JavaScript:** Incorpora componentes interactivos como modales, carruseles, tooltips y dropdowns sin necesidad de escribir código JavaScript desde cero.

Beneficios principales.

- **Diseño responsivo:** Los sitios creados con Bootstrap se ajustan automáticamente a diferentes tamaños de pantalla.
- **Facilidad de uso:** Su estructura modular y la amplia documentación disponible facilitan la integración y la personalización en cualquier proyecto.
- **Consistencia:** Al utilizar un framework estandarizado, se mantiene una coherencia visual en el diseño de aplicaciones y sitios web.
- **Ahorro de tiempo:** Los desarrolladores pueden reducir el tiempo de codificación y concentrarse más en la funcionalidad del sitio.

6.16 XAMPP como servidor local para desarrollo web

XAMPP es un paquete de software libre que proporciona un entorno de servidor local para el desarrollo web. Su nombre representa cada componente los cuales son: **X** (cross-platform), **A** (Apache), **M** (MariaDB), **P** (PHP) y **P** (Perl). Por lo que, este tipo de herramientas permite a los desarrolladores de software crear, probar y depurar aplicaciones web sin la necesidad de un servidor en línea. Después, se describe que significa cada componente de los antes mencionados GoDaddy (2024).

- **Apache:** Es el servidor web que permite la ejecución de páginas y aplicaciones desarrolladas en lenguajes como PHP.
- **MariaDB:** Es un sistema de gestión de base de datos relacional que sustituye a MySQL en las versiones más recientes de XAMPP.
- **PHP:** Es un lenguaje de programación ampliamente utilizado para el desarrollo de aplicaciones web dinámicas.
- **Perl:** Es un lenguaje de programación versátil, pero hoy en día ya no es tan utilizado.
- **phpMyAdmin:** Es una herramienta de administración de base de datos MySQL y MariaDB con una interfaz gráfica fácil de usar.

6.17 MariaDB y su importancia en la gestión de base de datos

MariaDB es un sistema de gestión de base de datos relacional, el cual se desarrolló como una bifurcación de MySQL. Debido que su creación fue impulsada por la comunidad de desarrolladores con el objetivo de garantizar la comunidad y accesibilidad del software después de la adquisición de MySQL por parte de Oracle Corporación.

Diferencias entre MariaDB y MySQL

Johnson (2022) menciona que estas dos comparten muchas similitudes, pero, aun así, existen diferencias clave entre ellos. Una de las principales es que MariaDB ofrece un mejor rendimiento en consultas complejas gracias a su optimización en la ejecución de subconsultas y mejoras en la replicación de datos. Además, también menciona que incorpora motores de almacenamiento adicionales que brindan una mayor flexibilidad y escalabilidad en el manejo de grandes volúmenes de información.

Mientras tanto, MySQL es gestionado por Oracle y algunas de sus versiones incluyen características propietarias. Por lo que, MariaDB mantiene un desarrollo completamente abierto con actualizaciones impulsadas por la comunidad. Ya que, estas actualizaciones aseguran que sean accesibles para todos los usuarios sin restricciones de licencias.

6.18 Lenguaje de programación PHP

El lenguaje de programación PHP (Hypertext Preprocessor) por sus siglas en inglés, como menciona Thomson (2021) es un lenguaje de código abierto ampliamente utilizado en el desarrollo web. Donde fue creado por Rasmus Lerdorf en 1994, lo que ha evolucionado bastante hoy en día. Su sintaxis es muy sencilla y similar a la de otros lenguajes como C y JavaScript, lo que facilita su lenguaje y uso. Una de las principales aplicaciones de PHP es la creación de sitios web dinámicos, donde a diferencia de los estáticos, estos permiten la interacción con el usuario y la personalización del contenido según las necesidades específicas. Permite realizar operaciones como el almacenamiento de información en base de datos, autenticación de usuarios y generación de contenido en tiempo real.

Además, es muy compatible con múltiples frameworks, como lo que es Laravel, Symfony y CodeIgniter, los cuales facilitan la estructuración de código y

agilizan el proceso de desarrollo. PHP es ampliamente utilizado en sitios web populares como Facebook, Wikipedia y WordPress.

6.19 Interfaz de usuario y experiencia del usuario (UI/UX)

La interfaz de usuario (UI) y la experiencia del usuario (UX) son conceptos fundamentales en el diseño de software y aplicaciones. Norman (2021) indica que la UI hace referencia a los elementos gráficos con los que interactúa el usuario como botones, menús y formularios, mientras que UX abarca la percepción general que tiene el usuario a utilizar un sistema, incluyendo factores como por ejemplo la facilidad de uso, la accesibilidad y satisfacción.

Importancia de UI/UX en el desarrollo de software

El diseño de UI/UX influye directamente en la adopción y en el éxito de un software. Donde, Norman (2021) menciona que, en el ámbito empresarial, un buen diseño de UX puede traducirse en mayores conversiones y satisfacciones del cliente. Asimismo, empresas tecnológicas invierten en pruebas de usabilidad y análisis de comportamiento de usuarios para poder optimizar sus productos digitales y así poder garantizar experiencias fluidas y agradables.

6.20 Modelo cliente-servidor

Es un paradigma de arquitectura de redes que divide las responsabilidades entre el cliente y el servidor. Donde el cliente es un dispositivo o programa que realiza solicitudes de servicios o recursos, mientras que el servidor es una máquina o software que procesa estas solicitudes y proporciona las respuestas correspondientes. Por ejemplo, en una aplicación web, el usuario accede al sitio web a través de un navegador (cliente), que envía una solicitud HTTP al servidor web, el cual este servidor responde con el contenido solicitado.

6.21 Front-end y Back-end

Empezando de que el front-end y el back-end trabajan en un conjunto para ofrecer una experiencia fluida y funcional a los usuarios. Por lo que, front-end maneja la interacción visual y la usabilidad, mientras que back-end se encarga de la seguridad, el almacenamiento y el procesamiento de los datos.

- **Front-end:** Se refiere a la parte de la aplicación con la que el usuario interactúa directamente.
- **Back-end:** Es la lógica del servidor, la gestión de base de datos y el procesamiento de las solicitudes enviadas por el cliente.

Estos dos componentes son muy esenciales para el desarrollo de aplicaciones web modernas, así, garantizando una interfaz amigable para el usuario y una gestión eficiente de los datos y procesos internos.

6.22 Lenguaje unificado de modelado (UML)

El lenguaje unificado de modelado (UML por sus siglas en inglés) es un estándar para la visualización, especificación, construcción y la documentación de los artefactos de un sistema de software. Lo que proporciona un conjunto de diagramas que permiten representar distintos aspectos de un sistema, así, facilitando la comunicación entre los desarrolladores en el proyecto. Mas adelante se especifica cada diagrama con más detalle.

Diagrama de caso de uso

Es una representación gráfica de las interacciones entre los usuarios (actores) y el sistema. Su propósito es definir los requisitos funcionales del software y describir las funcionalidades que deben implementarse, ya que, sus principales elementos son los siguientes:

- **Actores:** Representas a los usuarios que interactúan con el software.

- **Casos de uso:** Describen las funciones específicas que el sistema debe realizar.
- **Relaciones:** Indican la interacción entre los actores y los casos de uso.

A continuación, en la **Figura 4** se muestra un ejemplo de un diagrama de un caso de uso de un sistema de riego, especificando cada elemento.

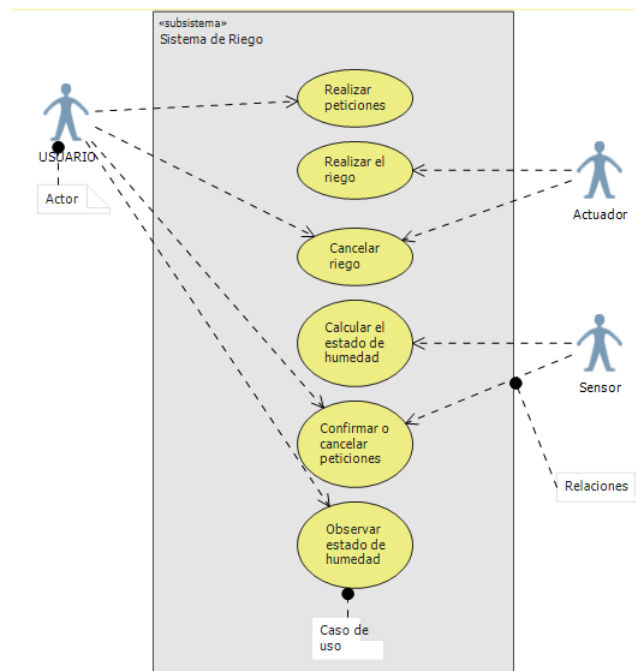


Figura 4. Diagrama de caso de uso - Sistema de riego.

Fuente: Elaboración propia

Diagrama de secuencia

Es el que modela la interacción entre los objetos de un sistema a lo largo del tiempo. Debido a que, es muy útil para entender la lógica del flujo de control y la comunicación entre componentes, por lo que, sus principales componentes son los siguientes:

- **Objetos:** Son representados por rectángulos con una línea de vida.

- **Mensajes:** Indicados por flechas, ya que muestran la comunicación entre los objetos.
- **Tiempo:** Es representado en el eje vertical, donde los eventos ocurren en secuencia.

Como se muestra en la **Figura 5** un ejemplo de un diagrama de secuencia al ingresar a un sistema de riego y realizar una acción.

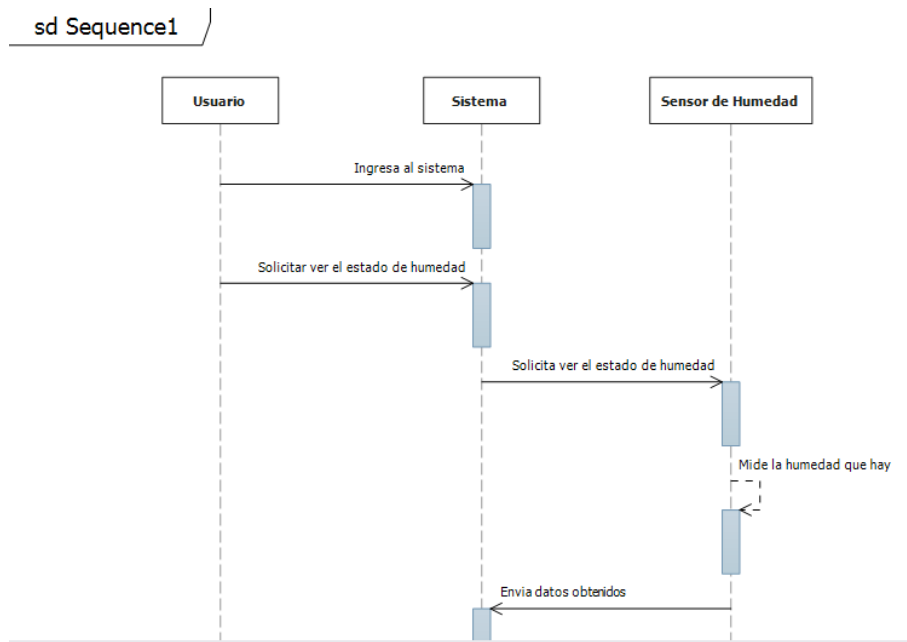


Figura 5. Diagrama de secuencia - Ingreso a un sistema de riego.

Fuente: Elaboración propia

Modelo entidad-relación

Es una representación gráfica de la estructura de una base de datos, se utiliza para diseñar la organización de los datos antes de su implementación en un sistema de gestión de base de datos. Sus principales componentes son los siguientes:

- **Entidades:** Representan objetos o conceptos con información relevante como, por ejemplo: Clientes, productos, pedidos, etc.
- **Relaciones:** Muestran como las entidades están conectadas entre sí.

- **Atributos:** Describen las propiedades de cada entidad como por ejemplo su nombre, dirección, fecha, etc.

Mas adelante, en la **Figura 6** se muestra un ejemplo de un modelo entidad-relación de cómo se relaciona un sistema de riego.

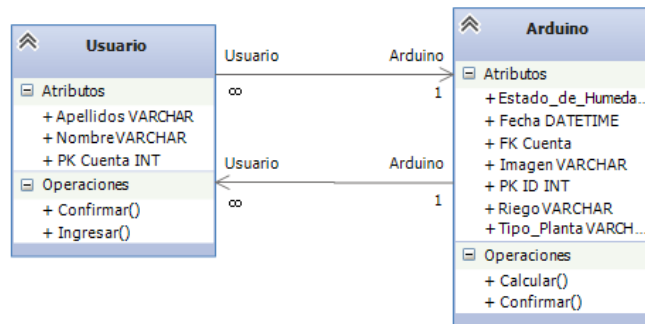


Figura 6. Modelo entidad-relación - Sistema de riego.

Fuente: Elaboración propia

Diagrama de actividades

Es una representación visual de flujo de trabajo dentro de un sistema. El cual se asemeja a un diagrama de flujo y permite modelar procesos empresariales o algoritmos. Sus principales componentes son los siguientes:

- **Acciones:** Representan tareas o actividades realizadas en el proceso.
- **Flujos de control:** Indican la secuencia de ejecución de las actividades.
- **Decisiones:** Representadas por rombos, ya que permiten adaptar el flujo de trabajo según condiciones específicas.

En la **Figura 7** se muestra un ejemplo de un diagrama de actividades de la configuración de una aplicación.

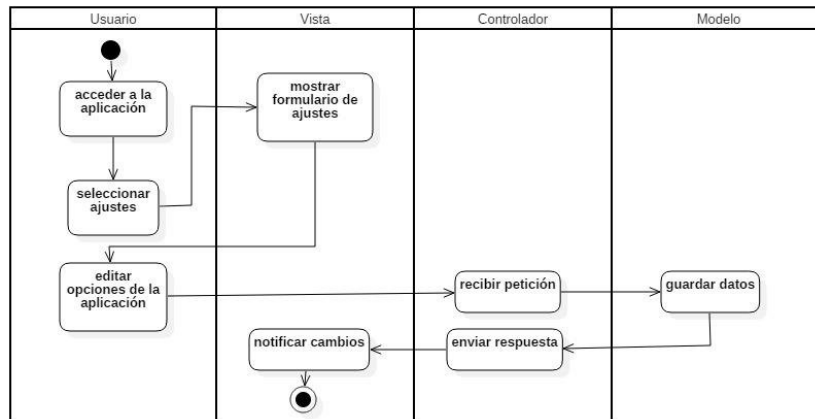


Figura 7. Diagrama de actividades - Configuración de una aplicación.

Fuente: ResearchGate (2025)

Diagrama de flujo de datos

Es el que modela el flujo de información dentro de un sistema. Es utilizado para describir como los datos entran, son procesados y almacenados en un sistema.

Sus principales componentes son los siguientes:

- **Procesos:** Representan transformaciones de datos.
- **Flujo de datos:** Muestran el movimiento de la información.
- **Almacenes de datos:** Indican donde se guardan los datos en el sistema.
- **Entidades externas:** Representan fuentes o destinos de datos fuera del sistema.

En la **Figura 8** se muestra un ejemplo de un diagrama de flujo de datos acerca de cómo hacer un huevo frito.

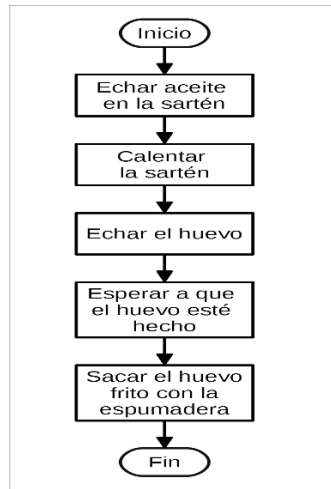


Figura 8. Diagrama de flujo de datos - Como hacer un huevo frito.

Fuente: Picuino (2025)

VII. MÉTODOS Y TÉCNICAS DE INVESTIGACIÓN

En este trabajo se aplicará el método de investigación experimental y documental para el diseño del sistema para la administración de la dulcería. Los cuales permitieron obtener información relevante sobre la problemática a abordar y para comprobar la efectividad de la implementación tecnológica propuesta en el negocio. Por lo que más adelante se describirá en breve en que se basan específicamente estos métodos y como se implementan en el proyecto.

- **Investigación documental:** Es un proceso basado principalmente en la búsqueda, recuperación, análisis, crítica e interpretación de los datos secundarios o tomados de otras fuentes documentales como: impresas, audiovisuales o electrónicas.
- **Investigación experimental:** Es un proceso que consiste en someter a un objeto o grupo de individuos, a determinadas condiciones para observar los defectos o reacciones que se producen.

Por lo tanto, se explica, como estas dos investigaciones se relacionan con este proyecto para llevarlo a cabo al objetivo deseado.

- **Investigación documental:** Consistió en la recopilación, análisis y organización de fuentes bibliográficas, como en este caso artículos científicos y documentos digitales, los cuales son relacionados con el desarrollo de software, bases de datos, el uso de Bootstrap, MySQL y como se utiliza Visual Studio Code en entornos de programación. Ya que, este pequeño método permitió establecer un marco teórico sólido para poder comprender el avance tecnológico y la importancia de la implementación de tecnologías en la optimización de procesos internos. Ya que, a través de analizar casos con éxito en la implementación de sistemas a negocios pequeños resulto bastante útil, lo que permitió

identificar mejores prácticas para la aplicación de tecnologías en una dulcería.

- **Investigación experimental:** Se llevo a cabo mediante la implementación y prueba del sistema en un entorno real. Permitiendo así evaluar su funcionamiento con cada prueba realizada, así evitando errores y medir su impacto en la optimización de los procesos internos de la dulcería.

Así, este enfoque experimental permitió validar la eficiencia y eficacia del sistema y poder realizar ajustes en función de los resultados obtenidos, asegurando que la solución tecnológica desarrollada respondiera de manera efectiva a los requerimientos de la dulcería.

Una vez mencionados los métodos que se implementarán en este proyecto. Para el desarrollo de este proyecto se empleará la metodología ágil Scrum, la cual permite una evolución continua del sistema mediante ciclos iterativos e incrementales. Esta metodología facilita la adaptación a cambios y mejoras durante el desarrollo, asegurando que el producto final cumpla con los requisitos del negocio. A continuación, se detallan las fases del proceso de desarrollo Scrum:

Inicio y Análisis: En esta fase, se definirán los objetivos del sistema y se realizará un análisis detallado de los requerimientos específicos:

- Identificación de necesidades clave como gestión de usuarios, empleados, clientes, productos y registros de compra.
- Creación del Product Backlog, el cual contendrá todas las funcionalidades requeridas para el sistema.

Diseño: Se establecerá la estructura inicial del programa, enfocándose en su arquitectura y diseño de interfaces:

- Definición de la arquitectura del sistema mediante el uso de Visual Studio Code como entorno de desarrollo.
- Creación de las vistas principales utilizando Bootstrap, garantizando una interfaz responsiva y atractiva.
- Definición de la estructura de la base de datos en MariaDB, asegurando una organización eficiente de la información.

Implementación Iterativa (Sprints): El desarrollo se llevará a cabo en ciclos iterativos denominados Sprints, en los cuales se implementarán funcionalidades de manera incremental:

- Programación de las funcionalidades definidas en el Sprint Backlog utilizando PHP y conexión con MariaDB.
- Ejecución de pruebas continuas para validar el correcto funcionamiento de cada módulo.
- Identificación y resolución de problemas unitarios durante el desarrollo.

Verificación: Después de cada iteración, se evaluará el avance del sistema con el propósito de realizar los ajustes necesarios:

- Aplicación de pruebas de integración, garantizando la correcta interacción entre los diferentes módulos del sistema.
- Recopilación de comentarios y sugerencias de los usuarios clave, con el fin de incorporar mejoras en el Product Backlog en caso de ser necesario.

Mantenimiento: Debido a las características y peticiones del cliente, no se requiere de esta fase que está considerada en esta metodología.

VIII. PRESENTACIÓN Y DISCUSIÓN DE RESULTADOS

A continuación, se explica todos los procedimientos que se llevan a cabo, así como las necesidades de la dulcería, las herramientas que se utilizan para el diseño y el desarrollo del sistema que llevará a cabo para su administración interna. Por lo que, se comenzará explicando los requerimientos que se necesitan para la elaboración del programa.

8.1 Análisis de requerimientos

En base a una pequeña entrevista con el dueño de la dulcería el Sr. Daniel Barajas Flores, se planteó como debe funcionar el sistema para que pueda ayudar a dar solución a los problemas que han tenido últimamente. Considerando que al mismo tiempo se obtuvieron los siguientes requerimientos funcionales que debe llevar este sistema.

Requerimientos funcionales

- Tener una base de datos relacional.
- El sistema debe ser entendible y amigable para los usuarios.
- Que cuente con una autenticación.
- Pueda registrar nuevos usuarios con contraseñas encriptadas de seguridad.
- Tener acciones como: registrar, eliminar, actualizar y se puedan observar los nuevos o viejos registros como los que son los clientes, empleados, usuarios, productos y las ventas que se realizan por día.
- Mande un mensaje cuando se elimine o registre un nuevo usuario.
- Aparezca el nombre del usuario quien está activo o dentro del programa.
- Que en la esquina superior derecha aparezca el logo de la dulcería para representar que es su sistema.

- Gestión de inventario de sus productos.
- Tener permisos limitados a algunos usuarios.
- Contar con una barra de búsqueda mediante palabras clave.
- Exportar los documentos en PDF, Excel y otros sugeridos.
- Tenga la capacidad de mostrar la información específica sin necesidad de mostrar toda.

Requerimientos no funcionales

- No es necesario cifrar la información.
- No hace falta cargar el sistema a un servidor de internet, debido a que su funcionamiento será de forma local.
- No requiere un tiempo específico de respuesta.
- No es necesario exportar imágenes ya que no es necesario en la dulcería.
- No se necesita un navegador específico para correr el sistema.
- No hay necesidad de la impresión de un ticket para los clientes.

8.2 Herramientas a utilizar

Tomando en cuenta los requerimientos funcionales para el desarrollo del sistema para la administración de la dulcería, Posteriormente, se especifican las herramientas a ocupar para poder desarrollar el sistema:

- Editor de código fuente Visual Studio Code versión 8.2
- XAMPP versión 8.2.12
- Gestor de base de datos MariaDB versión 10.6.4
- Framework Bootstrap.
- Lenguaje de programación PHP.
- Aplicación web phpMyAdmin.

8.3 UML del desarrollo del sistema

En esta sección se presentan los diagramas desarrollados para la presentación estructural y funcional del sistema, utilizando el lenguaje unificado de modelado (UML). Estos diagramas permiten visualizar la interacción de los usuarios con el sistema.

8.3.1 Diagrama caso de uso

El diagrama de caso de uso se utiliza para poder visualizar de manera grafica el proceso que se realiza mediante la interacción de los usuarios con el sistema para realizar cualquier tarea. Seguidamente se especifican los componentes de este diagrama con el sistema de forma general y en la **Figura 9** se muestra un ejemplo:

Actores:

- **Administrador:** Es el que puede realizar todas las acciones, ya que es el usuario con todos los permisos hacia el sistema, desde el más mínimo hasta la mayor tarea.
- **Cajero:** Realiza la mayoría de las tareas dentro del sistema, ya que es el encargado de vender los productos dentro de la dulcería y realizar algunas modificaciones y cuenta con un usuario y contraseña para ingresar al sistema.
- **Bodeguero:** El que no cuenta con un usuario o una contraseña, pero es el que está al pendiente de los movimientos de los productos, ya sea como su propia información y puede vender productos en caso de que sea necesario con la cuenta de otro usuario.
- **Cliente:** Es el que nada más compra cualquier producto dentro de la dulcería.

Casos de uso:

- Ingresar al sistema.
- Registrar productos vendidos.
- Registrar nuevos registros.
- Actualizar y eliminar registros.
- Observar registros.
- Buscar registros.
- Comprar productos.
- Imprimir información.

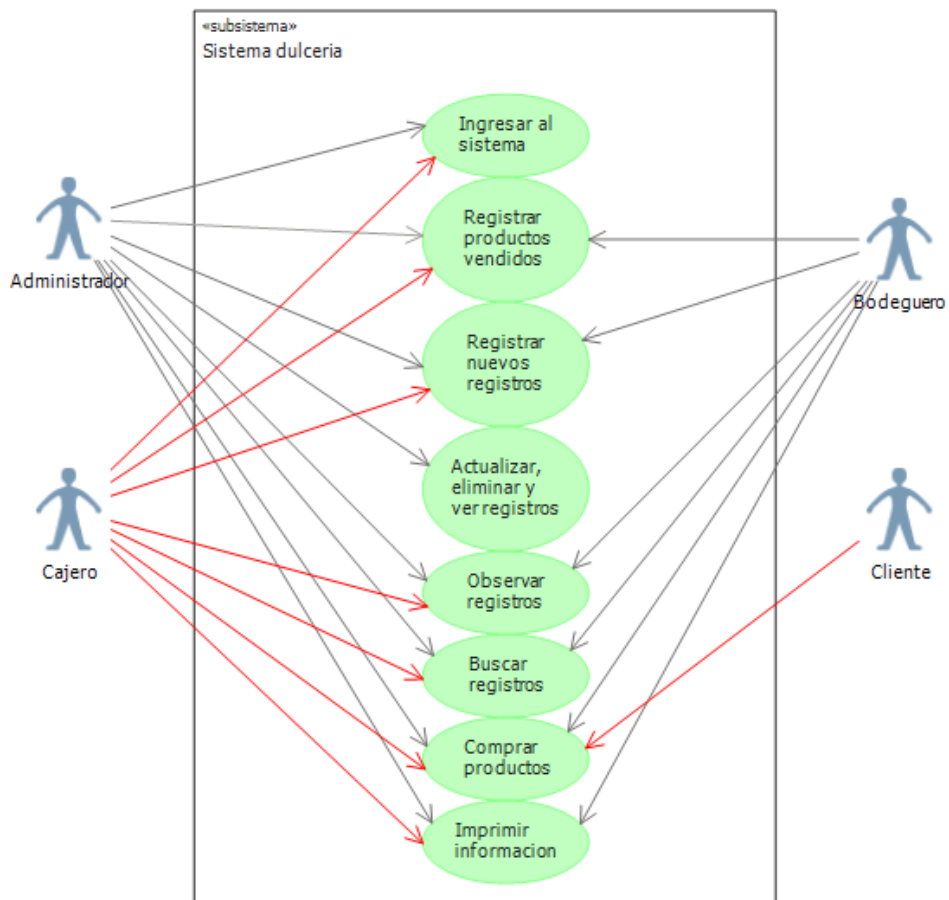


Figura 9. Diagrama caso de uso - Sistema de una dulcería

Fuente: Elaboración propia

8.3.2 Diagrama de secuencia

El diagrama de secuencia permite representar de forma detallada la interacción temporal entre los distintos objetos del sistema y los actores que intervienen en un proceso específico. Donde, a diferencia del diagrama anterior que ofrece un ejemplo de forma general, ya que este diagrama se centra en el orden cronológico de los mensajes intercambiados durante una tarea determinada. Por lo que, se ha elaborado un diagrama de secuencia para cada uno de los actores principales del sistema, con el objetivo de describir con mayor claridad las tareas específicas que realiza cada usuario y como interactúan con los distintos módulos del sistema.

Actor: Administrador

En este diagrama se representa las diferentes actividades que el administrador puede hacer dentro del sistema y como es que el sistema responde mediante las diferentes tareas que el administrador hace desde que entra al sistema hasta donde cierra sesión de la cuenta como se muestra más adelante en la **Figura 10**.



Figura 10. Diagrama de secuencia – Administrador

Fuente: Elaboración propia

Actor: Cajero

Aquí se muestran las actividades que puede realizar un cajero dentro del sistema, donde tiene ciertas limitaciones, como que un cajero no pueda registrar un nuevo usuario o empleado o pueda eliminarlos o actualizarlos y el sistema no le permita realizar ese tipo de tareas a los cajeros como se muestra en la **Figura 11**.

☒

Figura 11. Diagrama de secuencia – Cajero

Fuente: Elaboración propia

Actor: Bodeguero

Por lo que aquí se muestra las actividades que puede realizar un bodeguero dentro del sistema, donde solo puede realizar algunos registros con el usuario de otro compañero de trabajo, ya que en la dulcería no es necesario que cuente con un usuario porque el solo está en el almacén, pero cuenta con el conocimiento necesario para poder realizar algunas tareas dentro del sistema como se muestra en la **Figura 12**.

✖

Figura 12. Diagrama de secuencia - Bodeguero

Fuente: Elaboración propia

Actor: Cliente

En este caso se indican las actividades que puede realizar un cliente desde que llega a la tienda pero que no interactúa con el sistema como se muestra en la **Figura 13**.

☞

Figura 13. Diagrama de secuencia - Cliente

Fuente: Elaboración propia

8.3.3 Diagrama de Clases

Este diagrama representa la estructura estática del sistema, mostrando las clases que componen la base de datos, así como sus atributos, llaves primarias, etc. Donde, en la **Figura 14** se muestra el diagrama de clases de la base de datos para el sistema.

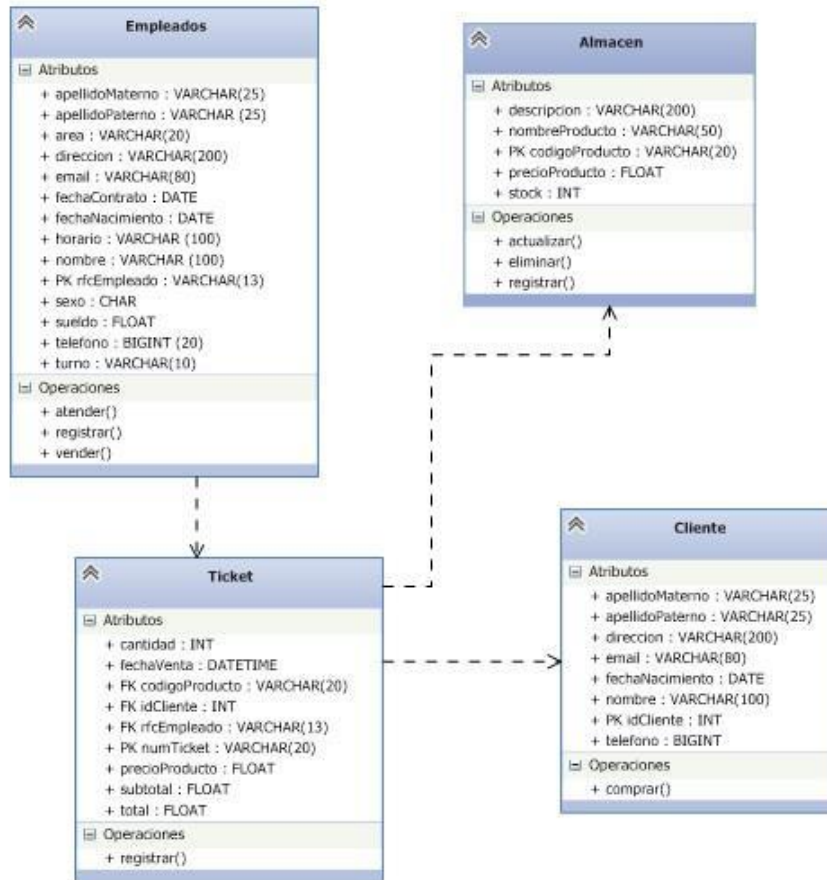


Figura 14. Diagrama de clases - BD del sistema

Fuente: Elaboración propia

Nota: La clase llamada “Almacén” es donde están registrados todos los productos de la dulcería dentro del sistema y la clase llamada “Ticket” ahí es donde se encuentran las ventas del día dentro del sistema sin la necesidad de arrojar un ticket físico para el cliente. Ya que se hizo el uso de nombres cortos con el fin de no utilizar nombres largos dentro de la base de datos.

8.3.4 Diagrama Entidad – Relación (ER)

El diagrama ER permite representar gráficamente la estructura lógica de la base de datos para este sistema, mostrando las entidades principales, sus atributos y las relaciones que hay entre ellas. Este diagrama es fundamental para la fase de análisis y de diseño, ya que ayuda a comprender como se va a organizar y a relacionar la información del sistema, así como se muestra en la **Figura 15**, el cual detalla las entidades más relevantes del sistema.

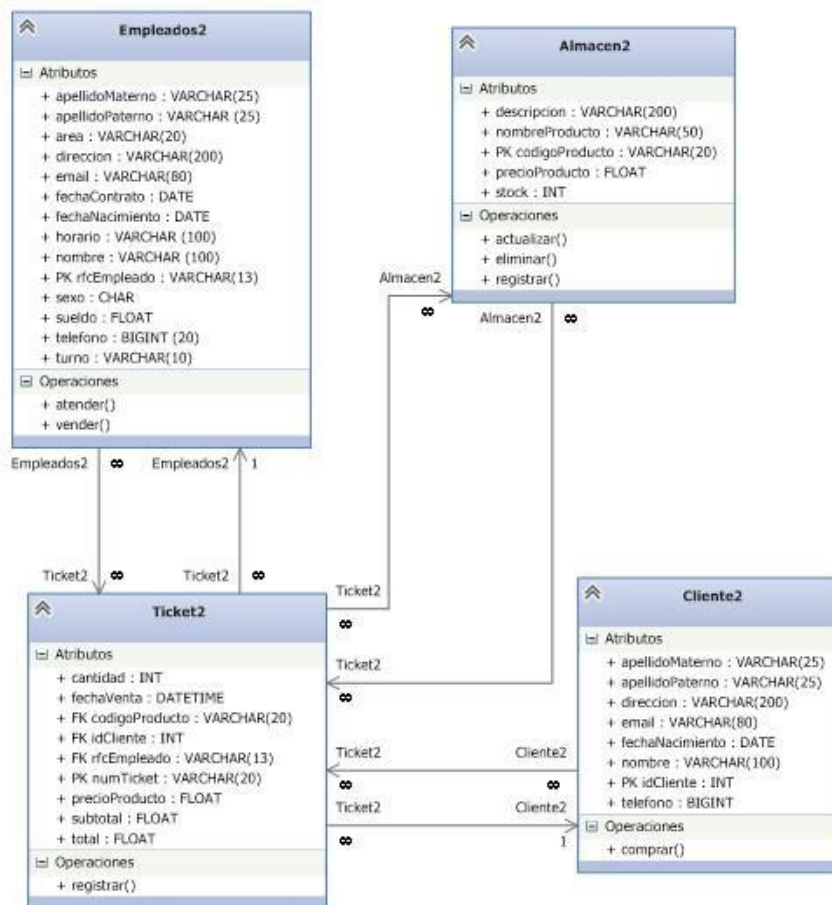


Figura 15 Diagrama ER - BD del Sistema

Fuente: Elaboración propia

8.4 Implementación (Desarrollo del sistema)

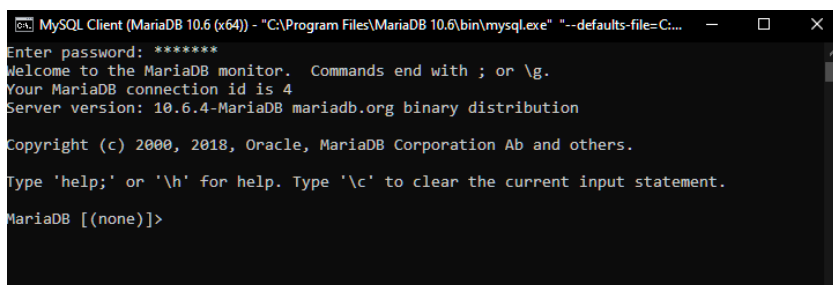
Para el desarrollo de este sistema se utilizó Visual Studio Code como entorno de desarrollo principal, debido a su ligereza, compatibilidad con múltiples extensiones y la facilidad de integración con otras tecnologías. En cuanto al servidor local se utilizó XAMPP permitiendo simular un entorno de producción para ejecutar y poder probar el sistema en tiempo real, asimismo para el diseño de su interfaz gráfica se utilizó el framework Bootstrap para que permitiera que el sistema fuera más atractivo y adaptable para los usuarios y por último para la gestión de base de datos se llevó a cabo MariaDB por su facilidad de administración.

Modelo

El modelo es el encargado de encapsular los datos del sistema, por lo que, dentro del modelo se realiza la creación de la base de datos.

8.4.1 Base de datos

En la **Figura 16** se muestra la línea de comando de MariaDB versión 10.6.4, que junto con el lenguaje SQL se utilizan para la implementación del diseño lógico de la base de datos para este sistema.



```
MySQL Client (MariaDB 10.6 (x64)) - "C:\Program Files\MariaDB 10.6\bin\mysql.exe" "--defaults-file=C:...
Enter password: *****
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 4
Server version: 10.6.4-MariaDB mariadb.org binary distribution
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
MariaDB [(none)]>
```

Figura 16. Línea de comando MariaDB versión 10.6.4

Fuente: Elaboración propia

El código en lenguaje SQL se utiliza para la creación de la base de datos **Figura 17**. Donde, se muestra el código utilizado para la creación de tablas dentro de la base de datos **Figura 18**.

```
MaríaDB [(none)]> create database dulceriaKiosko;
```

Figura 17. Creación de base de datos

Fuente: Elaboración propia

```
MaríaDB [dulceriakiosko]> create table empleado(
-> rfcEmpleado varchar(13) primary key,
-> apellidoPaterno varchar(25),
-> apellidoMaterno varchar(25),
-> nombre varchar(100),
-> sexo char,
-> fechaNacimiento date,
-> telefono bigint,
-> email varchar(80),
-> direccion varchar(200),
-> fechaContrato date,
-> area varchar(20),
-> turno varchar(10),
-> horario varchar(100),
-> sueldo float);
Query OK, 0 rows affected (4.456 sec)

MaríaDB [dulceriakiosko]> create table cliente(
-> idCliente int primary key,
-> apellidoPaterno varchar(25),
-> apellidoMaterno varchar(25),
-> nombre varchar(100),
-> direccion varchar(200),
-> telefono bigint,
-> fechaNacimiento date,
-> email varchar(80));
Query OK, 0 rows affected (1.598 sec)
MaríaDB [dulceriakiosko]>

MaríaDB [dulceriakiosko]> create table ticket(
-> numTicket varchar(20) primary key,
-> fechaVenta datetime,
-> idCliente int,
-> rfcEmpleado varchar(13),
-> codigoProducto varchar(13),
-> precioProducto float,
-> cantidad int,
-> subtotal float,
-> total float);
Query OK, 0 rows affected (0.795 sec)

MaríaDB [dulceriakiosko]> create table almacen(
-> codigoProducto varchar(20) primary key,
-> nombreProducto varchar(50),
-> precioProducto float,
-> stock int,
-> descripcion varchar(200));
Query OK, 0 rows affected (0.710 sec)
```

Figura 18. Creación de tablas

Fuente: Elaboración propia

Luego, en la **Figura 19** se muestra la pantalla principal de XAMPP el cual vamos a ocupar para exportar nuestra base de datos a phpMyAdmin.

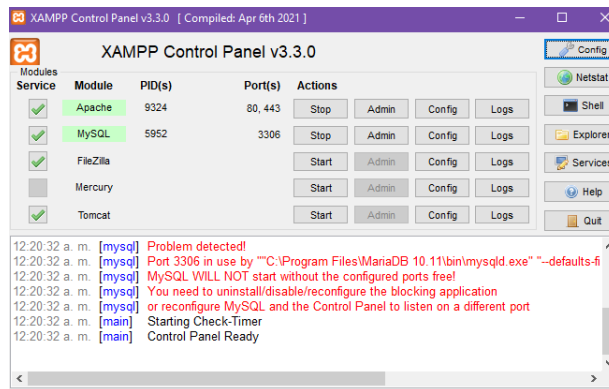


Figura 19. Pantalla principal de XAMPP

Fuente: Elaboración propia

Una vez dentro en la **Figura 20** se muestra como se visualiza la base de datos creada en phpMyAdmin mismo que tiene incluido XAMPP, para que así pueda tener conexión con el sistema.

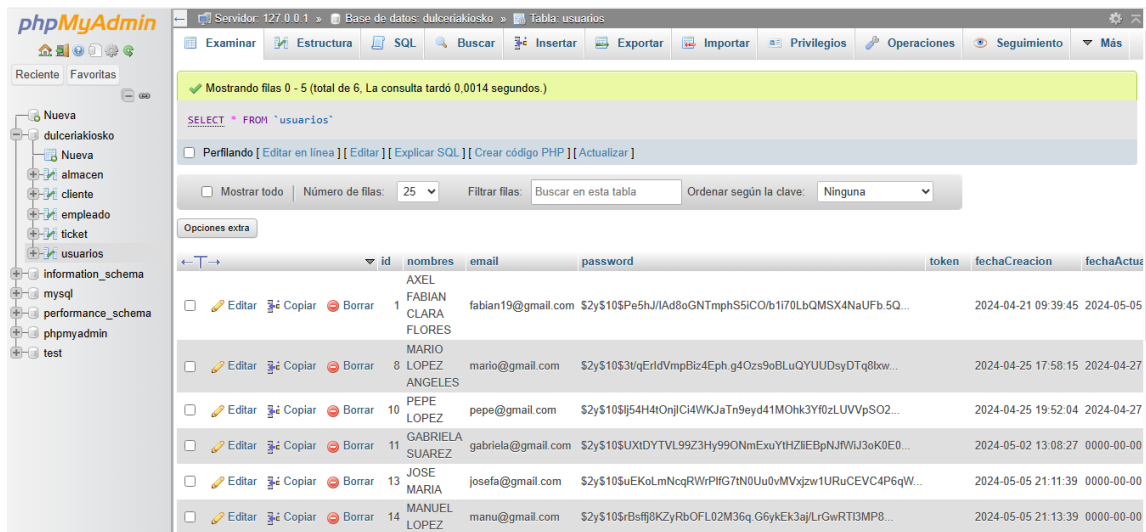


Figura 20. Base de datos en phpMyAdmin

Fuente: Elaboración propia

Conexión de Base de datos con el sistema

Antes de crear el diseño del sistema, primero debemos de enlazar la base de datos con el sistema, como se muestra en la **Figura 21** el código para enlazar el sistema con la base de datos haciendo uso de la herramienta Visual Studio Code.

```
1 <?php
2
3 define ('SERVIDOR','localhost');
4 define ('USUARIO','root');
5 define ('PASSWORD','');
6 define ('BD','dulceriakiosko');
7
8 $servidor = "mysql:dbname=".BD.";host=".SERVIDOR;
9
10 try {
11     $pdo = new PDO($servidor, USUARIO, PASSWORD,array(PDO::MYSQL_ATTR_INIT_COMMAND->"SET NAMES utf8"));
12     //echo "ahuevo";
13 } catch (PDOException $e) {
14     //print_r($e);
15     echo "Error al conectar a la base de datos";
16 }
17
18 $URL = "http://localhost/www.dulceria.com";
19
20 date_default_timezone_set("America/Mexico_City");
21 $fechaHora = date('Y-m-d H:i:s');
22
```

Figura 21. Conexión de BD con el sistema

Fuente: Elaboración propia

8.4.2 Diseño del sistema

Después, se presenta el diseño inicial del sistema, el cual constituye una etapa fundamental en el proceso del desarrollo. Tomando en cuenta que, se documenta paso a paso la construcción visual del sistema.

Autenticación

En la **Figura 22** se muestra el código correspondiente al módulo de inicio de sesión del sistema y en la **Figura 23** el código correspondiente al cierre de sesión, por lo que, en ambas partes se implementa la lógica para autenticar a los usuarios mediante la validación de las credenciales ingresadas.

```

1 <?php
2 include('.././config.php');
3
4 $email = $_POST['email'];
5 $password = $_POST['password'];
6
7
8 $contador = 0;
9 $sql = "SELECT * FROM usuarios WHERE email = '$email'";
10 $query = $pdo->prepare($sql);
11 $query->execute();
12 $usuarios = $query->fetchAll( fetch_style: PDO::FETCH_ASSOC);
13 foreach ($usuarios as $usuario){
14     $contador = $contador + 1;
15     $email = $usuario['email'];
16     $nombres = $usuario['nombres'];
17     $password1 = $usuario['password'];
18 }
19 }
20
21
22 if( ($contador > 0) && (password_verify($password, $password1))){
23     echo "Datos correctos";
24     session_start();
25     $_SESSION['sesion email'] = $email;
26     header('Location: ' . $URL . '/index.php');
27 }else{
28     echo "Acceso denegado, vuelva a intentarlo";
29     session_start();
30     $_SESSION['mensaje'] = "Acceso denegado";
31     header('Location: ' . $URL . '/login');
32 }

```

Figura 22. Código de autenticación - Inicio de sesión

Fuente: Elaboración propia

```

1 <?php
2
3 include ('.././config.php');
4
5 session_start();
6 if (isset($_SESSION['sesion email'])) {
7     session_destroy();
8     header('Location: ' . $URL . '/');
9 }

```

Figura 23. Código de autenticación - Cierre de sesión

Fuente: Elaboración propia

Usuarios

En esta sección se presenta los códigos correspondientes a las operaciones de crear, eliminar, visualizar y actualizar un usuario, con la especificación principal para las cuentas de usuarios, en este caso es que sean seguros, por lo que también se presenta el código para que las contraseñas que se crean por otros usuarios sean encriptadas una vez guardadas en el sistema, así como se muestran en las siguientes Figuras.

```

1 <?php
2 include('../././config.php');
3
4 $nombres = $_POST['nombres'];
5 $email = $_POST['email'];
6 $password = $_POST['password'];
7 $password_R = $_POST['password_R'];
8
9 if ($password == $password_R) {
10     $password = password_hash($password, PASSWORD_DEFAULT);
11     $sentencia = $pdo->prepare("INSERT INTO usuarios
12         (nombres, email, password, fechaCreacion)
13         VALUES (:nombres,:email,:password,:fechaCreacion)");
14     $sentencia->bindParam(':nombres', $nombres);
15     $sentencia->bindParam(':email', $email);
16     $sentencia->bindParam(':password', $password);
17     $sentencia->bindParam(':fechaCreacion', $fechaHora);
18     $sentencia->execute();
19     session_start();
20     $_SESSION['mensaje'] = "Se registro al nuevo usuario";
21     $_SESSION['icono'] = "success";
22     header('location:'. $URL .'/usuarios');
23 }else{
24     //echo "Las contraseñas no coinciden";
25     session_start();
26     $_SESSION['mensaje'] = "Las contraseñas no coinciden";
27     $_SESSION['icono'] = "error";
28     header('location:'. $URL .'/usuarios/create.php');
29 }
30
31 >>

```

Figura 24. Código de función Agregar - Usuarios

Fuente: Elaboración propia

```

1 <?php
2 include('../././config.php');
3
4 $id_usuario = $_POST['id_usuario'];
5
6 $sentencia = $pdo->prepare("DELETE FROM usuarios WHERE id=:id_usuario");
7 $sentencia->bindParam(':id_usuario', $id_usuario);
8 $sentencia->execute();
9 session_start();
10 $_SESSION['mensaje'] = "Se elimino al usuario";
11 $_SESSION['icono'] = "success";
12 header('location:'. $URL .'/usuarios');
13
14 >>

```

Figura 25. Código de función Eliminar - Usuarios

Fuente: Elaboración propia

```

1 <?php
2
3 $id_usuario_get = $_GET['id'];
4
5 $sql_usuarios = "SELECT * FROM usuarios where id = '$id_usuario_get'";
6 $query_usuarios = $pdo->prepare($sql_usuarios);
7 $query_usuarios->execute();
8 $usuarios_datos = $query_usuarios->fetchAll( fetch_style: PDO::FETCH_ASSOC);
9
10 foreach ($usuarios_datos as $usuarios_datos){
11     $nombres = $usuarios_datos['nombres'];
12     $email = $usuarios_datos['email'];
13

```

Figura 26. Código de función Visualizar - Usuarios

Fuente: Elaboración propia

```

1  <?php
2
3  $id_usuario_get = $_GET['id'];
4
5  $sql_usuarios = "SELECT * FROM usuarios where id = '$id_usuario_get'";
6  $query_usuarios = $pdo->prepare($sql_usuarios);
7  $query_usuarios->execute();
8  $usuarios_datos = $query_usuarios->fetchAll( fetch_style: PDO::FETCH_ASSOC);
9
10 foreach ($usuarios_datos as $usuarios_dato){
11     $nombres = $usuarios_dato['nombres'];
12     $email = $usuarios_dato['email'];
13 }
14 ?>

```

Figura 27. Código de función Actualizar - Usuarios

Fuente: Elaboración propia

```

if ($password == "") {
    if ($password == $password_R) {
        $password = password_hash($password, PASSWORD_DEFAULT);
    }
}

```

Figura 28. Código para contraseñas encriptadas - Usuarios

Fuente: Elaboración propia

password

\$2y\$10\$Pe5hJ/IA8oGNTmphS5iCO/b1i70LbQMSX4NaUFb.5Q...

\$2y\$10\$3t/qErdVmpBiz4Eph.g4Ozs9oBLuQYUUDsyDTq8lxw...

Figura 29. Contraseñas encriptadas en phpMyAdmin - Usuarios

Fuente: Elaboración propia

Diseño de pantalla principal

Después de establecer el sistema de autenticación, se presenta el código del diseño de la pantalla principal del sistema una vez dentro, el cual representa la interfaz que visualizará el usuario. Así como se muestra en la **Figura 30** y en la **Figura 31** se muestra el diseño final de la pantalla principal.

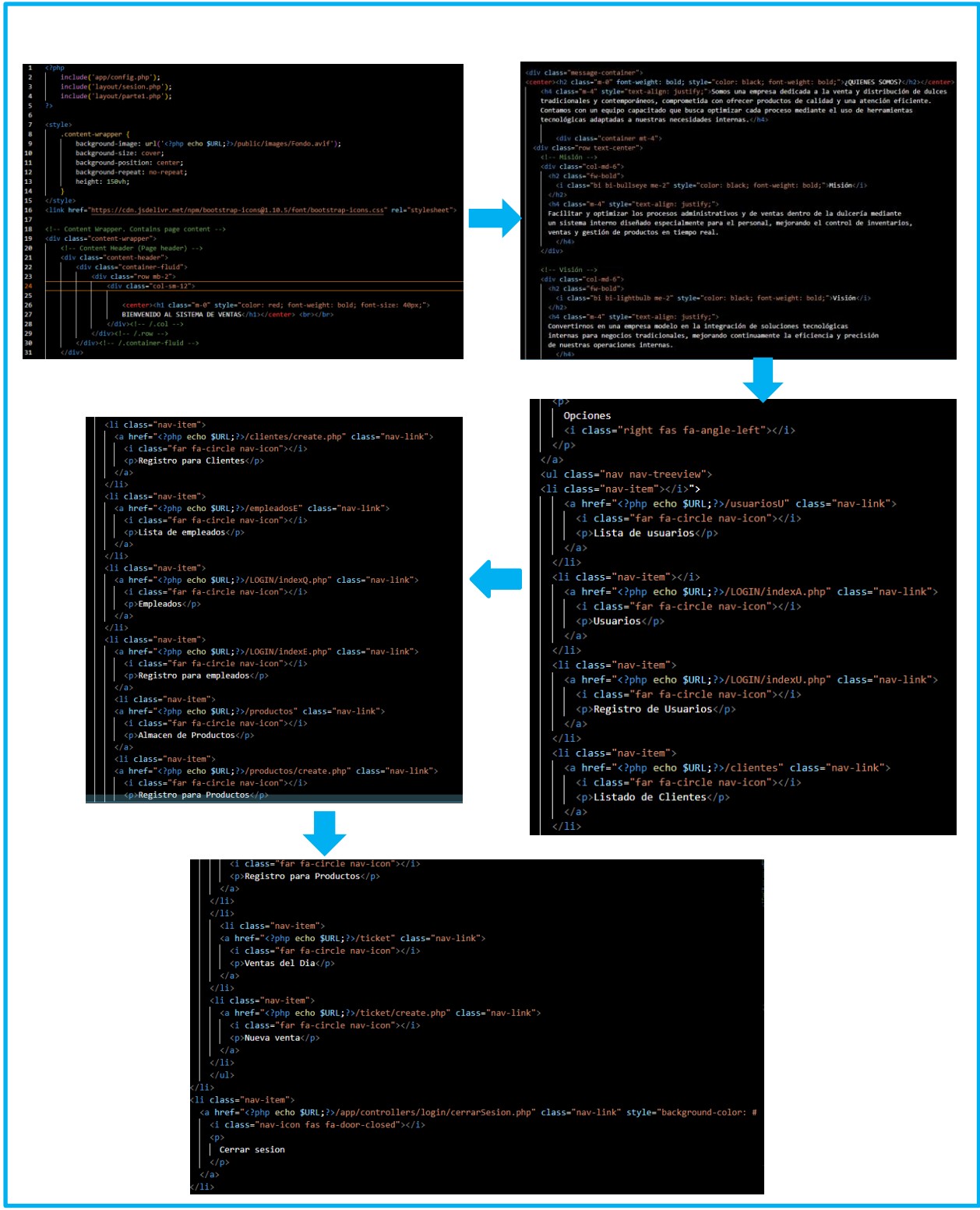


Figura 30. Vistas del código de la pantalla principal.

Fuente: Elaboración propia

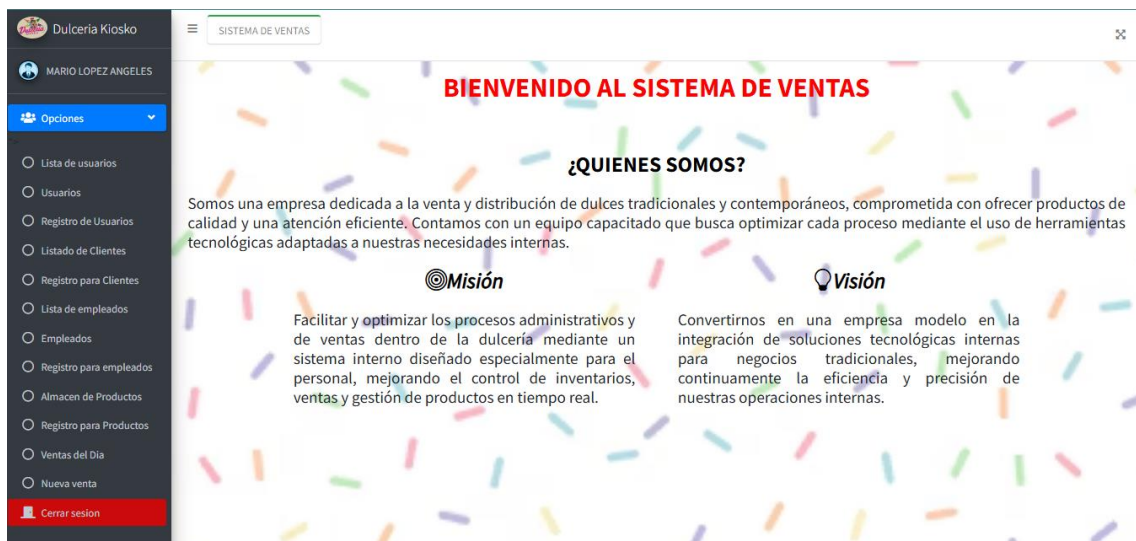


Figura 11. Diseño final de la pantalla principal.

Fuente: Elaboración propia

Operaciones funcionales

En esta sección se presentan los fragmentos de código utilizados para realizar las operaciones funcionales como las opciones para poder agregar, actualizar, eliminar y visualizar los registros dentro de la base de datos. Estas funciones son esenciales para una correcta manipulación de la información en el sistema.

Mas adelante se muestra un ejemplo de cada opción con el código correspondiente a cada una de estas funciones como se muestra en la **Figura 32**, tomando como ejemplo una de las tablas fuente de la base de datos, en este caso la tabla cliente, donde, todas las tablas representan el código similar, pero con los diferentes atributos de cada una.

```

1 <?php
2 include('../config.php');
3
4 $idCliente = $_POST['idCliente'];
5 $apellidoPaterno = $_POST['apellidoPaterno'];
6 $apellidoMaterno = $_POST['apellidoMaterno'];
7 $nombre = $_POST['nombre'];
8 $direccion = $_POST['direccion'];
9 $telefono = $_POST['telefono'];
10 $fechaNacimiento = $_POST['fechaNacimiento'];
11 $email = $_POST['email'];
12
13
14
15
16 if (isset($_POST['idCliente']) && isset($_POST['apellidoPaterno']) && isset($_POST['apellidoMaterno']) && isset($_POST['nom
17     $sentencia = $pdo->prepare("INSERT INTO cliente
18     (idCliente, apellidoPaterno, apellidoMaterno, nombre, direccion, telefono, fechaNacimiento, email)
19     VALUES (:idCliente, :apellidoPaterno, :apellidoMaterno, :nombre, :direccion, :telefono, :fechaNacimiento, :email)");
20 $sentencia->bindParam(':idCliente', $idCliente);
21 $sentencia->bindParam(':apellidoPaterno', $apellidoPaterno);
22 $sentencia->bindParam(':apellidoMaterno', $apellidoMaterno);
23 $sentencia->bindParam(':nombre', $nombre);
24 $sentencia->bindParam(':direccion', $direccion);
25 $sentencia->bindParam(':telefono', $telefono);
26 $sentencia->bindParam(':fechaNacimiento', $fechaNacimiento);
27 $sentencia->bindParam(':email', $email);
28 $sentencia->execute();
29 session_start();
30 $_SESSION['mensaje'] = "Se registro al nuevo cliente";
31 $_SESSION['icono'] = "success";
32 header('location:'.$_URL.'/clientes');
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
```

Código del diseño de la tabla fuente para agregar un nuevo cliente.

En la **Figura 33** se mostrará el código del diseño de la tabla fuente, para poder agregar a un cliente y como se muestra el diseño total en el sistema en la **Figura 34**.

```
</style>
<!-- Content Wrapper. Contains page content -->
<div class="content-wrapper">
  <!-- Content Header (Page header) -->
  <div class="content-header">
    <div class="container-fluid">
      <div class="row mb-2">
        <div class="col-sm-12">
          <h1 class="h0" style="color: green; font-weight: bold;">REGISTRO PARA UN NUEVO CLIENTE</h1>
        </div><!-- /.col -->
      </div><!-- /.row -->
    </div><!-- /.container-fluid -->
  </div>
  <!-- /.content-header -->
  <center>
    <div class="content">
      <div class="container-fluid">
        <div class="row">
          <div class="col-md-5">
            <div class="card card-outline card-success">
              <div class="card-header">
                <h3 class="card-title">Ingresar datos</h3>
                <div class="card-tools">
                  <button type="button" class="btn btn-tool" data-card-widget="collapse"><i class="fas fa-minus"></i>
                </div>
              </div>
              <div class="card-body" style="display: block;">
                <div class="row">
                  <div class="col-md-12">
                    <input type="text" name="id" class="form-control" placeholder="Escriba el ID" req
                  </div>
                </div>
                <div class="form-group">
                  <label for="">ID:</label>
                  <input type="number" name="idCliente" class="form-control" placeholder="Escriba el ID" req
                </div>
                <div class="form-group">
                  <label for="">Apellido Paterno:</label>
                  <input type="text" name="apellidoPaterno" class="form-control" placeholder="Escriba el Ap
                </div>
                <div class="form-group">
                  <label for="">Apellido Materno:</label>
                  <input type="text" name="apellidoMaterno" class="form-control" placeholder="Escriba el Ap
                </div>
                <div class="form-group">
                  <label for="">Nombre:</label>
                  <input type="text" name="nombre" class="form-control" placeholder="Escriba el Nombre" req
                </div>
                <div class="form-group">
                  <label for="">Direccion:</label>
                  <input type="text" name="direccion" class="form-control" placeholder="Escriba la Direccio
                </div>
                <div class="form-group">
                  <label for="">Telefono:</label>
                  <input type="text" name="telefono" class="form-control" placeholder="Escriba el telefono"
                </div>
                <div class="form-group">
                  <label for="">Fecha Nacimiento:</label>
                  <input type="date" name="fechaNacimiento" class="form-control" placeholder="Escriba la fe
                </div>
                <div class="form-group">
                  <label for="">Correo:</label>
                  <input type="email" name="email" class="form-control" placeholder="ejemplo@gmail.com" req
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </center>
  </div>
  <div class="row">
    <div class="col-md-12">
      <div class="form-group">
        <label for="">Fecha Nacimiento:</label>
        <input type="date" name="fechaNacimiento" class="form-control" placeholder="Escriba la fe
      </div>
      <div class="form-group">
        <label for="">Correo:</label>
        <input type="email" name="email" class="form-control" placeholder="ejemplo@gmail.com" req
      </div>
      <hr>
      <div class="form-group">
        <a href="index.php" class="btn btn-secondary">Cancelar</a>
        <button type="submit" class="btn btn-success">Guardar</button>
      </div>
    </div>
  </div>
</div>
</div>
```

Figura 33. Vistas del código para el diseño de la tabla Agregar.

Fuente: Elaboración propia

The image shows a web interface for a sales system. At the top, there is a navigation bar with a hamburger menu icon and the text "SISTEMA DE VENTAS". Below this, a green header reads "REGISTRO PARA UN NUEVO CLIENTE". The main content area is titled "Ingresar datos" and contains a form with the following fields: "ID:" with a placeholder "Escriba el ID"; "Apellido Paterno:" with a placeholder "Escriba el Apellido Paterno"; "Apellido Materno:" with a placeholder "Escriba el Apellido Materno"; "Nombre:" with a placeholder "Escriba el Nombre"; "Direccion:" with a placeholder "Escriba la Direccion"; "Telefono:" with a placeholder "Escriba el Telefono"; "Fecha Nacimiento:" with a placeholder "dd/mm/aaaa" and a calendar icon; and "Correo:" with a placeholder "ejemplo@gmail.com". At the bottom of the form are two buttons: "Cancelar" (grey) and "Guardar" (green).

Figura 34. Diseño final de la tabla Agregar cliente.

Fuente: Elaboración propia

Código del diseño de la tabla fuente para actualizar un cliente.

En la **Figura 35** se mostrará el código del diseño de la tabla actualizar y como se muestra el diseño total en el sistema en la **Figura 36**.



Figura 35. Vistas del código para el diseño de la tabla Actualizar.

Fuente: Elaboración propia

Figura 36. Diseño final de la tabla actualizar cliente.

Fuente: Elaboración propia

The screenshot shows a web application interface for a sales system. At the top, there is a navigation bar with a hamburger menu icon and the text 'SISTEMA DE VENTAS'. Below this is a header section titled 'Vista de Clientes'. The main content area is titled 'Datos del Cliente' and contains a form with the following fields:

- ID:** 123
- Apellido Paterno:** Lopez
- Apellido Materno:** Angolez
- Nombre:** Mario Bucio
- Direccion:** Valle de Chalco
- Telefono:** 5576873890
- Fecha Nac:** 2002-03-10
- Correo:** mario19@gmail.com

At the bottom of the form, there is a yellow button labeled 'Regresar'.

Figura 38. Diseño final de la tabla visualizar cliente.

Fuente: Elaboración propia

Código del diseño de la tabla fuente para eliminar un cliente.

Después, en la **Figura 39** se mostrará el código del diseño de la tabla para poder eliminar un cliente y como se muestra el diseño total en el sistema en la **Figura 40**.

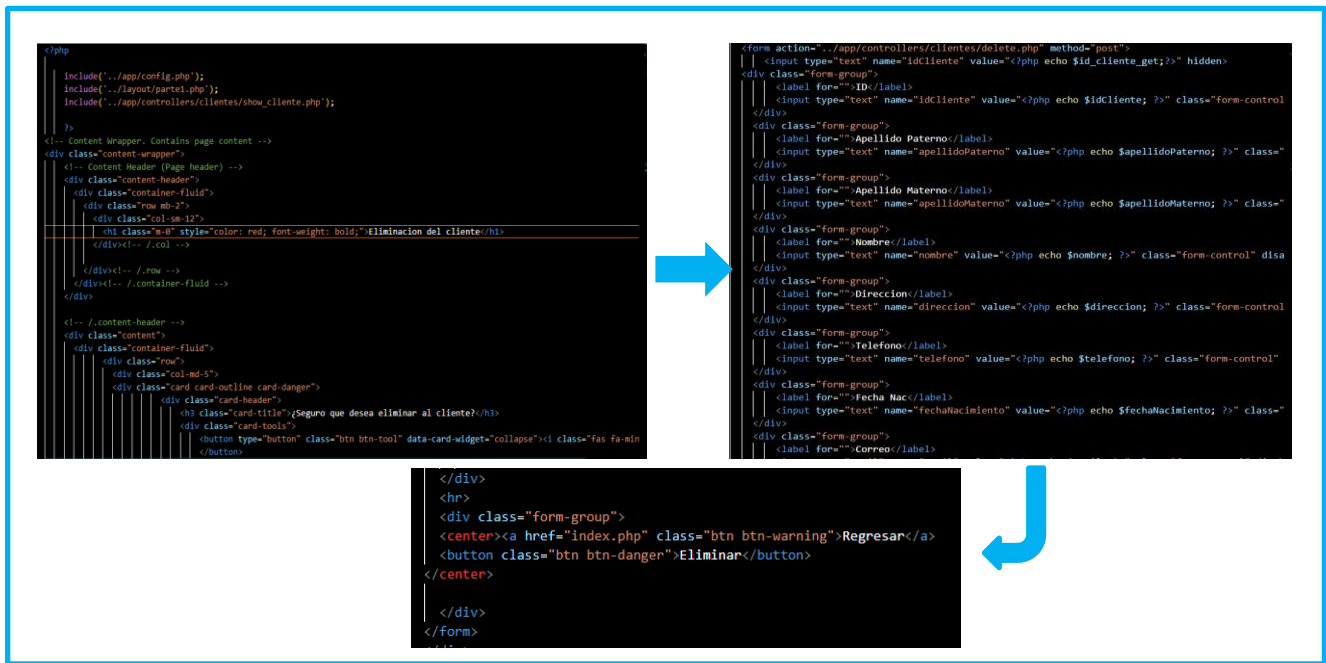


Figura 39. Vistas del código para el diseño de la tabla eliminar.

Fuente: Elaboración propia

Figura 40. Diseño final de la tabla eliminar cliente.

Fuente: Elaboración propia

8.5 Verificación

El día 13 de enero del 2025 se tuvo una reunión con el dueño de la dulcería, el señor Daniel Ricardo Barajas Flores, con el propósito de presentarle el sistema desarrollado específicamente para su dulcería. Durante la reunión se le mostró el sistema en su versión final, tomando en cuenta que fue construido cumpliendo cada una de las especificaciones y requerimientos que él mismo mencionó en las fases iniciales del proyecto.

Asimismo, se solicitó que realizara una revisión detallada del sistema para verificar que cumpliera con los requisitos establecidos durante la fase de análisis, enmarcados dentro de la metodología que se definió para este proyecto. Tras una evaluación funcional y visual del sistema. Por lo que, el propietario expresó su conformidad respecto a su funcionamiento general para poder realizar las tareas del negocio, donde el sistema estuvo a prueba una semana, del día 6 de enero al día 12 de enero del presente año.

Como parte del cierre de este proyecto, el señor Daniel Ricardo Barajas Flores firmó un documento donde manifiesta estar satisfecho con el programa desarrollado, reconociendo que cumple con los requerimientos propuestos. Ya que cabe destacar que no se presentaron observaciones técnicas o funcionalidades que implican realizar algunos cambios, lo cual valida que el sistema fue correctamente implementado y adaptado a las necesidades reales de la dulcería.

A continuación, se muestra evidencia fotográfica de la validación del sistema por parte del propietario, donde se muestra la calificación del sistema, si tiene algunas observaciones, su nombre y firma de que está satisfecho con el sistema como se muestra en la **Figura 41**.



DOCUMENTO DE ACEPTACIÓN DEL SISTEMA

Nombre del proyecto: Sistema para la administración de la dulcería.

Responsable del proyecto: Axel Fabian Clara Flores.

Desarrollado para: Dulcería Kiosko.

Nombre del propietario: Daniel Ricardo Barajas Flores.

Fecha de entrega: 13 de enero del 2025.

El presente documento tiene como objetivo dejar constancia de la entrega y aceptación del sistema desarrollado para la administración dentro de la dulcería, el cual fue diseñado con base a los requerimientos establecidos por el propietario. Por lo que, el sistema ha sido revisado y calificado por el Sr. Daniel Ricardo Barajas Flores quien manifiesta que:

EVALUACIÓN DEL SISTEMA	SI	NO
El sistema cumple con los requisitos planteados durante la fase de análisis.	✓	
Las funcionalidades implementadas permiten una correcta administración interna para la dulcería.	✓	
Se encuentra satisfecho con el diseño y funcionamiento del sistema.	✓	
La interfaz es clara, intuitiva y accesible para los usuarios autorizados.	✓	
La seguridad implementada protege adecuadamente la información y el acceso al sistema.	✓	
El sistema queda formalmente aprobado para su uso dentro del entorno operativo de la dulcería.	✓	

Observaciones: Porque, el sistema cumple con los requisitos establecidos, con lo que queda satisfecho con el sistema.

FIRMAS

Nombre del Propietario: Daniel Ricardo Barajas Flores
Firma del propietario: 
Fecha: 13 de enero del 2025

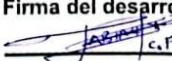
Nombre del desarrollador: Axel Fabian Clara Flores
Firma del desarrollador: 
Fecha: 13 de enero del 2025

Figura 41. Documento de aceptación firmado por el dueño de la dulcería.

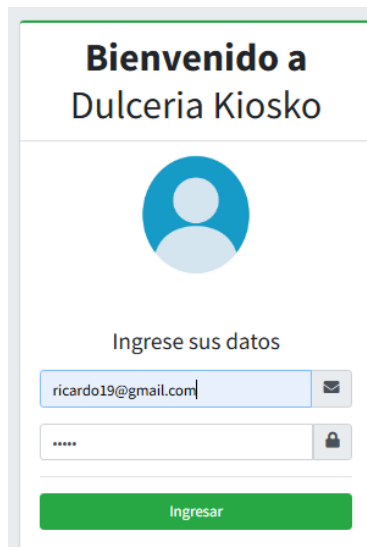
Fuente: Elaboración propia

8.7 Lanzamiento del sistema

El día 13 de enero del 2025, se llevó a cabo el lanzamiento del sistema dentro de las instalaciones de la dulcería, tomando en cuenta su uso en el entorno real de operación para las tareas de la dulcería. Esta implementación se llevó a cabo después de haber completado todas las fases del desarrollo, incluyendo el diseño, funcionamiento y la revisión por el mismo cliente.

Por lo que, tras el periodo de prueba, el propietario de la dulcería, confirmó su conformidad con el desempeño del sistema, así expresando una satisfacción con el cumplimiento de los requisitos establecidos. Con esta aprobación, el sistema quedó oficialmente implementado como herramienta de trabajo diario en la dulcería, consolidando así el objetivo de optimizar las tareas internas de la dulcería mediante una solución tecnológica.

En la **Figura 42** se muestra como el propietario inicia sesión con su correo electrónico y contraseña para el sistema y en la **Figura 43** se muestra el nombre del usuario que está dentro del sistema.



The image shows a login interface for 'Dulceria Kiosko'. At the top, it says 'Bienvenido a Dulceria Kiosko'. Below that is a blue circular profile icon. The text 'Ingrese sus datos' is centered above two input fields. The first field contains the email address 'ricardo19@gmail.com' and has an envelope icon on the right. The second field contains six asterisks '*****' and has a lock icon on the right. At the bottom, there is a green button with the text 'Ingresar'.

Figura 42. Inicio de sesión con la cuenta del propietario de la dulcería.

Fuente: Elaboración propia

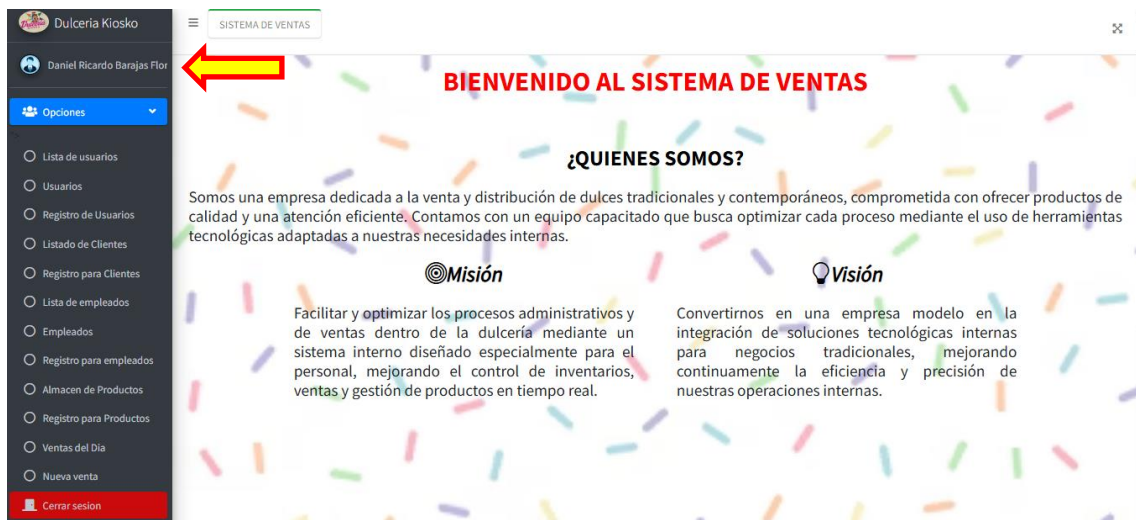


Figura 43. Nombre del usuario dentro del sistema.

Fuente: Elaboración propia


Tomando en cuenta que, con el paso del tiempo, el sistema ha demostrado ser una herramienta eficaz, ya que se ha registrado un crecimiento en el número de usuarios, donde ahora no solo el propietario podía hacer uso del sistema, sino que también empleados que han sido dados de alta con diferentes niveles de acceso como se muestra en las Figuras 44 y 45.

LISTADO DE USUARIOS			
Usuarios Registrados			
Numero	Nombres	Correo	Acciones
1	Juan Armando Suarez Rodriguez	jarmando19@gmail.com	Ver Editar Eliminar
2	Daniel Ricardo Barajas Flores	ricardo19@gmail.com	Ver Editar Eliminar
3	Mario Antonio Vargas Pérez	mario@gmail.com	Ver Editar Eliminar
4	José López López	josell18@gmail.com	Ver Editar Eliminar
5	Gabriela Suarez Flores	gabriela23@gmail.com	Ver Editar Eliminar

Figura 44. Lista de usuarios creados por el propietario.

Fuente: Elaboración propia

PERSONAL NO AUTORIZADO



Ingrese datos de supervisor

Email

Password

[Ingresar](#)

[Regresar](#)

Figura 45. Vistas donde algunos usuarios no tienen suficiente acceso.

Fuente: Elaboración propia

Asimismo, se han agregado nuevos registros en el sistema como por ejemplo nuevos empleados, clientes, nuevas ventas del día y nuevos productos como se muestran a continuación en las siguientes capturas.
















Numero	RFC	Apellido Paterno	Apellido Materno	Nombre	Genero	Fecha Nac	Telefono	Correo	Direccion	Fecha Contrato	Area	Turno	Horario	Sueldo	Sucursal	Acciones
1	MJYHW2363MD	DOLORES	SANDOVAL	DIEGO	M	1992 01 12	553222234	diego@gmail.com	CHALCO	2024 05 12	Surtidor	Matutino	De Lunes a Sabado de 07:00 am a 13:00 pm	2500	CHALCO	  
2	CAFE02043HLL	MIENDEZ	LOPEZ	ADRIAN	M	1997 05 20	5543234566	adrian@gmail.com	CHALCO	2023 05 02	Surtidor	Matutino	De Martes a Sabado de 07:00 am a 13:00 pm	2000	CHALCO	  
3	CAFE230419HLL	LOPEZ	DOMINGUEZ	ARIADNE	F	2001 05 22	5567878888	ari@gmail.com	CHALCO	2024 05 05	Cajera	Vespertino	De Lunes a Sabado de 13:00 pm a 20:00 pm	2000	CHALCO	  
4	DADE120302HLL	Oregon	Hernandez	Daniel Abraham	M	1999 01 14	5565433356	abraham@gmail.com	Valle de Chalco	2015 03 10	Bodeguero	Vespertino	De Lunes a Sabado de 13:00 pm a 19:30 pm	2300	CHALCO	  
5	JJML910212HMC	Juarez	Martinez	Jose Luis	M	1991 02 12	5543346798	joseluis34@gmail.com	Ixtapaluca	2022 01 20	Pasillero	Matutino	De Martes a Sabado de 07:00 am a 13:00 pm	1900	CHALCO	  

Figura 46. Tabla de la lista de empleados registrados.

Fuente: Elaboración propia

LISTADO DE CLIENTES									
Clientes Registrados									
Mostrar <input type="text" value="Clientes"/> Buscador: <input type="text"/>									
Reportes <input type="text"/> Visor de columnas <input type="text"/>									
Numero	ID	Apellido Paterno	Apellido Materno	Nombre	Direccion	Telefono	Fecha Nac	Correo	Acciones
1	1234	Garcia	Ramirez	Itzel Airam	Valle de Chalco	5576879822	2002-05-12	itzel2423@gmail.com	
2	2414	Garcia	Rodriguez	Diana	Chalco	5585835462	1996-11-06	diana_rodriguez@gmail.com	
3	3463	Vasquez	Torres	Tania Sofia	Chalco	5543345676	2000-09-19	chofis2356@gmail.com	
4	12545	Dolores	Romero	Valeria	Valle de Chalco	5576890098	1990-01-12	vale12@outlook.com	
5	36569	LOPEZ	SOL	MARIA	CHALCO	5567879999	1990-05-12	maria@gmail.com	
Numero	ID	Apellido Paterno	Apellido Materno	Nombre	Direccion	Telefono	Fecha Nac	Correo	Acciones

Figura 47. Tabla de la lista de clientes registrados.

Fuente: Elaboración propia

VENTAS									
Ventas del Dia									
Mostrar <input type="text" value="Ventas"/> Buscador: <input type="text"/>									
Reportes <input type="text"/> Visor de columnas <input type="text"/>									
Numero	Fecha de Venta	ID del Cliente	RFC de Empleado	Codigo del Producto	Precio del Producto	Cantidad de productos	Subtotal	Total	Acciones
6	2025-04-19 01:59:30	2234	CAFE020434HLL	3298567	77	1	77	70	
7	2025-04-19 02:00:34	3563	CAFE230419HLL	2355	60	3	180	180	
8	2025-04-19 02:01:44	6433	MJYHNF2365MFS	3298567	24	5	120	120	
Numero	Fecha de Venta	ID del Cliente	RFC de Empleado	Codigo del Producto	Precio del Producto	Cantidad de productos	Subtotal	Total	Acciones

Mostrando 6 a 8 de 8 Ventas Anterior 1 2 Siguiente

Figura 48. Tabla de la lista de algunas de las ventas del día 07/01/2025.

Fuente: Elaboración propia

LISTADO DE PRODUCTOS									
Productos Registrados									
Mostrar <input type="text" value="Productos"/> Buscador: <input type="text"/>									
Reportes <input type="text"/> Visor de columnas <input type="text"/>									
Numero	Codigo	Nombre	Precio	Stock	Stock Mínimo	Stock Máximo	Descripcion		Acciones
1	1235258	TOP-TOPS	60	20	30	500	Papas TOP-TOP sabor: Nachos, 10 cajas de 10 paquetes con 10 paquetes de papas c/u, caducidad: 23/03/2025		
2	1345	Panditas	77	20	40	500	Gomitas de diferentes sabores, contenido por caja: 8 bolsas de kilo con 12 bolsitas c/u, caducidad: 20/11/2024		
3	2355	Coca-Cola	25	14	30	200	Refresco Coca-Cola, contenido por paquete: 20 refrescos de 500 ml, caducidad: 12:10:2025		
4	23990	Coca-Cola	25	45	30	300	Refresco Coca-Cola, contenido por paquete: 20 refrescos de 500 ml, caducidad: 12:10:2025		
5	3298567	polo	244	12	26	233	xcb		
Numero	Codigo	Nombre	Precio	Stock	Stock Mínimo	Stock Máximo	Descripcion		Acciones

Figura 49. Tabla de la lista de productos agregados.

Fuente: Elaboración propia

De igual manera, se muestra a continuación algunos detalles que el cliente quería en su sistema, uno de ellos es una barra de búsqueda que le permitiera encontrar cualquier dato sin la necesidad de estar buscando uno por uno hasta encontrarlo, como se muestra en la **Figura 50**.

The screenshot shows a web interface titled "LISTADO DE PRODUCTOS". At the top, there is a search bar labeled "Buscador:" with the text "Panditas" entered. Below the search bar, there are buttons for "Reportes" and "Visor de columnas". The main content is a table with the following columns: "Numero", "Codigo", "Nombre", "Precio", "Stock", "Stock Minimo", "Stock Maximo", "Descripcion", and "Acciones". The first row of data shows a product with "Numero" 2, "Codigo" 1345, "Nombre" Panditas, "Precio" 77, "Stock" 20, "Stock Minimo" 40, "Stock Maximo" 500, and "Descripcion" "Gomitas de diferentes sabores, contenido por caja: 8 bolsas de kilo con 12 bolsitas c/u, caducidad: 20/11/2024". The "Acciones" column for this row contains three icons: a yellow eye, a green checkmark, and a red square. Below the table, there is a pagination bar showing "Mostrando 1 a 1 de 1 Productos (Filtrado de 6 total Productos)" and navigation buttons for "Anterior", "1", and "Siguiente".

Numero	Codigo	Nombre	Precio	Stock	Stock Minimo	Stock Maximo	Descripcion	Acciones
2	1345	Panditas	77	20	40	500	Gomitas de diferentes sabores, contenido por caja: 8 bolsas de kilo con 12 bolsitas c/u, caducidad: 20/11/2024	[Icons]

Figura 50. Buscador de registros en la lista de productos.

Fuente: Elaboración propia

Otro detalle que el cliente quería, es que pudiera imprimir su información de diferentes maneras donde en la **Figura 51** se muestra el diferente tipo de impresión para su información.

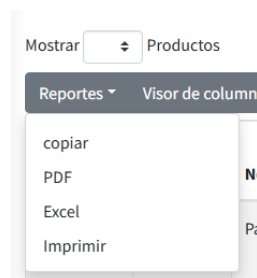


Figura 51. Tipos de impresión para la información.

Fuente: Elaboración propia

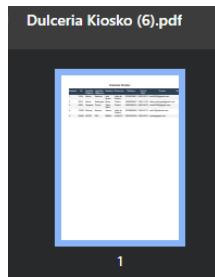
Tomando en cuenta los diferentes tipos de impresión, las siguientes capturas muestran un ejemplo de la impresión de la información de cada uno de ellos.

Dulceria Kiosko

Numero	ID	Apellido Paterno	Apellido Materno	Nombre	Dirección	Teléfono	Fecha Nac	Correo	Acciones
1	1234	García	Ramirez	Itzel Airam	Valle de Chalco	5576879822	2002-05-12	itzel2423@gmail.com	
2	2414	García	Rodriguez	Diana Chalco	Chalco	5585835462	1996-11-06	diana_rodriguez@gmail.com	
3	3463	Vásquez	Torres	Tania Sofia	Chalco	5543345676	2000-09-19	chofis2356@gmail.com	
4	12545	Dolores	Romero	Valeria	Valle de Chalco	5576890098	1990-01-12	vale12@outlook.com	
5	36569	LÓPEZ	SOL	MARÍA	CHALCO	5567879999	1990-05-12	maria@gmail.com	

Figura 52. Tipo de impresión - Copiar.

Fuente: Elaboración propia



Dulceria Kiosko

Numero	ID	Apellido Paterno	Apellido Materno	Nombre	Dirección	Telefono	Fecha Nac	Correo
1	1234	Garcia	Ramirez	Itzel Airam	Valle de Chalco	5576879822	2002-05-12	itzel2423@gmail.com
2	2414	Garcia	Rodriguez	Diana Chalco	Chalco	5585835462	1996-11-06	diana_rodriguez@gmail.com
3	3463	Vasquez	Torres	Tania Sofia	Chalco	5543345676	2000-09-19	chofis2356@gmail.com
4	12545	Dolores	Romero	Valeria	Valle de Chalco	5576890098	1990-01-12	vale12@outlook.com
5	36569	LOPEZ	SOL	MARIA	CHALCO	5567879999	1990-05-12	maria@gmail.com

Figura 53. Tipo de impresión - PDF.

Fuente: Elaboración propia

Numero	ID	Apellido Paterno	Apellido Materno	Nombre	Dirección	Telefono	Fecha Nac	Correo
1	1234	Garcia	Ramirez	Itzel Airam	Valle de Chalco	5576879822	12/05/2002	itzel2423@gmail.com
2	2414	Garcia	Rodriguez	Diana Chalco	Chalco	5585835462	06/11/1996	diana_rodriguez@gmail.com
3	3463	Vasquez	Torres	Tania Sofia	Chalco	5543345676	19/09/2000	chofis2356@gmail.com
4	12545	Dolores	Romero	Valeria	Valle de Chalco	5576890098	12/01/1990	vale12@outlook.com
5	36569	LOPEZ	SOL	MARIA	CHALCO	5567879999	12/05/1990	maria@gmail.com

Figura 54. Tipo de impresión - Excel.

Fuente: Elaboración propia



Figura 55. Tipo de impresión - Impresión.

Fuente: Elaboración propia

IX. CONCLUSIONES Y SUGERENCIAS

En esta sección se elaboran las principales conclusiones derivadas de la implementación del trabajo, las cuales reflejan los beneficios alcanzados a través del desarrollo del sistema para la dulcería. Por otro lado, también se presentan algunas sugerencias con base en la experiencia obtenida durante el desarrollo y prueba del sistema.

Conclusiones

El uso de las herramientas tecnológicas como Visual Studio Code, XAMPP, MariaDB, el framework Bootstrap, el lenguaje de programación PHP, facilitan el desarrollo de sistemas y permiten dar solución a problemas específicos dentro de un negocio como una dulcería.

La implementación del sistema contribuyó a optimizar procesos internos, como el registro de usuarios, empleados, clientes, productos o ventas.

Cumplió con su objetivo de poder permitir tener una administración más rápida, precisa y ordenada de los datos internos de la dulcería, así logrando que se realicen de una forma más eficiente.

Ofrece una interfaz agradable para el usuario, lo cual facilita su uso por cualquier usuario sin la necesidad de requerir conocimientos técnicos avanzados.

A lo largo del desarrollo del programa, se validó que la metodología ocupada SCRUM permitió mantener un avance ordenado y enfocado a las necesidades reales del cliente.

Su uso facilitó a los trabajadores del negocio a realizar sus tareas de una manera más fácil sin la necesidad de hacerlo manualmente, de cometer algún error o simplemente de perder demasiado tiempo en busca de cualquier registro cuando el mismo sistema lo resuelve en segundos.

Sugerencias

No es necesario el uso de un navegador en específico o un navegador con la actualización más reciente, ya que el sistema funciona por igual.

Para fines futuros, si la dulcería quisiera expandirse, se sugiere que el sistema se podría implementar a un servidor privado con un dominio específico.

Se puede considerar a futuro la implementación de usuarios con diferentes niveles de permisos (por ejemplo: administrador, cajero, supervisor), para mejorar el control y seguridad del sistema.

En caso de que el negocio crezca o se abran nuevas sucursales, sería conveniente adaptar el sistema para que pueda trabajar con múltiples sucursales, centralizando la información y permitiendo una administración más integral.

X. REFERENCIAS

- Admin. (2021, febrero 5). ¿Qué es un sistema de gestión y para qué sirve? ISBL. Recuperado el 10 de julio de 2021, de <https://isbl.eu/2021/01/que-es-un-sistema-de-gestion-y-para-que-sirve/>
- Admin. (2023, agosto 16). Metodología Scrum: ¿Qué es y cómo se aplica? PMBC. Recuperado de <https://pmbc.es/metodologia-waterfall-que-es-y-como-se-aplica/>
- Anderson, D. J. (2021). Kanban: Successful Evolutionary Change for Your Technology Google. Recuperado de https://books.google.com.mx/books/about/Kanban.html?id=RJOVUkfUWZkC&redir_esc=y
- Apd, R. (2024, abril 9). Cómo aplicar la metodología Scrum y qué es el método Scrum. APD España. Recuperado de <https://www.apd.es/metodologia-scrum-que-es/>
- Aroca, E. D. (2022, julio 20). El futuro de la tecnología biométrica. Dynatec. Recuperado de <https://dynatec.es/2022/07/20/el-futuro-de-la-tecnologia-biometrica/>
- Azanza, E. (2024, febrero 12). Identidad, Biometría e Inteligencia Artificial: y proyectos de gestión para abarroses. Veridas. Recuperado de <https://veridas.com/es/biometria-etica-mitos-realidades/>
- Blog. (2023, noviembre 18). Top 9 mejores frameworks para desarrollo web. Kiwop. Recuperado de <https://www.kiwop.com/blog/mejores-frameworks-para-desarrollo-web>

- Bootstrap. (2023). Introducción a Bootstrap. Recuperado de <https://getbootstrap.com/docs/4.0/getting-started/introduction/>
- Brookshear, J. G., & Brylow, D. (2020). Computer Science: An Overview. Pearson. Recuperado de <https://www.pearson.com/en-us/subject-catalog/p/computer-science-aoverview/P200000003351/9780137503216>
- Camacho (2023) Lenguaje Java. Recuperado de <https://docencia.tic.unam.mx/presenciales/Lenguaje-de-programacion-java.html>
- Centro de Aprendizaje de Google. (s. f.). Los 10 mejores consejos para el desarrollo web. Recuperado de <https://support.google.com/a/users/answer/9991078?hl=es>
- Ceruzzi, P. (2012). Computing Concise History. The MIT Press. Recuperado de: <https://direct.mit.edu/books/book/3732/ComputingA-Concise-History>
- Ceruzzi, P. (2018). Una historia de la informática moderna miPress Recuperado de <https://mitpress.mit.edu/9780262531696/a-history-of-modern-computing/>
- Connolly, T., & Begg, C. (2021). ¿Qué es una base de datos? Oracle. Recuperado de. <https://www.oracle.com/mx/database/what-is-database/>
- Digital Guide. (mayo, 2020). Paradigmas de programación: definición y tipos más importantes. Recuperado de <https://www.ionos.mx/digitalguide/paginas-web/desarrollo-web/el-modelo-en-cascada>
- Fernández, P. (2020). Desarrollo de un sistema de administración de reservas para hoteles de pequeño y mediano tamaño. Recuperado de <https://bibdigital.epn.edu.ec/bitstream/15000/19649/1/CD-9055.pdf>

- Flores. (2023, 30 de mayo). Qué es Bootstrap, para qué sirve y cómo funciona. Hubspot. Recuperado de <https://blog.hubspot.es/website/que-es-bootstrap>
- Fowler, M. (2021). La nueva metodología. Recuperado de <https://www.martinfowler.com/articles/newMethodology.html>
- GoDaddy (2024). Guía sobre XAMPP. Recuperado de <https://www.godaddy.com/resources/es/crearweb/xampp-que-es>
- Gómez, F. (2021). Desarrollo de un sistema de punto de venta con autenticación de usuarios. Universidad Tecnológica de UNAD. Recuperado de <https://repository.unad.edu.co/bitstream/handle/10596/41806/jhvillamilr.pdf?sequence=1>
- Hernández, L. (2021). Implementación de un sistema de gestión de citas médicas en clínicas privadas. Recuperado de <https://uvadoc.uva.es/bitstream/handle/10324/36492/TFG-B.1225.pdf>
- Hubspot. (2023, mayo 30). Qué es Bootstrap, para qué sirve y cómo funciona. Recuperado de <https://blog.hubspot.es/website/que-es-bootstrap>
- IONOS. (mayo, 2020). Paradigmas de programación: definición y tipos más importantes. Recuperado de <https://www.ionos.mx/digitalguide/paginas-web/desarrollo-web/paradigmas-de-programacion/>
- Jiménez, R. (2022). Sistema de control de inventarios para empresas distribuidoras de productos de consumo masivo. Recuperado de <https://www.redalyc.org/journal/6219/621968032003/html/>
- Juárez (2021). Base de datos. Recuperado de <https://es.scribd.com/document/581173818/Bases-de-Datos>

- Juárez, P. (2022). Introducción a los frameworks y su impacto en el desarrollo de software. *Revista de Tecnología y Desarrollo*, 10(2), 45-60. Recuperado de <https://www.redalyc.org/journal/6139/613964507005/html/>
- Levy, S. (2021). *Hackers: Héroes de la revolución informática*. O'Reilly Media. Recuperado de <https://www.oreilly.com/pub/pr/2566>
- López, M., & Pérez, R. (2020). Implementación de sistemas de gestión en puntos de venta. Instituto Tecnológico de DGBSDI. Recuperado de <https://ru.dgb.unam.mx>
- Lutz, M. (2023). *Learning Python*. Universidad Galileo. Recuperado de <https://www.galileo.edu/noticias/la-importancia-de-la-programacion-en-la-actualidad-un-pilar-en-la-revolucion-industrial-digital/>
- Markoff, J. (2020). *Machines of Loving Grace: The Quest for Common Ground Between Humans and Robots*. HarperCollins. Recuperado de <https://latam.casadellibro.com/libro-machines-of-loving-grace-the-quest-for-common-ground-between-humans-arobots/9780062266699/10065169>
- Martínez. (marzo, 2020). Historia de los lenguajes de programación. Recuperado de <https://computerhoy.20minutos.es/reportajes/tecnologia/historia-lenguajes-programacion-428041>
- Martínez, J. (2021). Sistema automatizado para la gestión de inventarios en pequeñas empresas. *Revista de Innovación Tecnológica*, 15(3), 120-135. Recuperado de <https://www.redalyc.org/pdf/4778/477847107005.pdf>
- McKinsey & Company. (2022). *The State*. Recuperado de <https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/the-state-of-ai-in-2022>

- Microsoft. (2023). Visual Studio Code - Code Editing. Redefined. Recuperado de <https://code.visualstudio.com/>
- Norman, S. (2021). Diseño UI/UX. Rockcontent. Recuperado de <https://rockcontent.com/es/blog/ui-ux/>
- Ortega, M. (2023). Software para la gestión de pedidos y entregas en una empresa de comida rápida. Recuperado de <http://repopib.ubiobio.cl/jspui/bitstream/123456789/2760/1/Parra%20Olave%2C%20Fernando%20Felipe.pdf>
- Ortiz, M. (2020). Tablas jerárquicas. Recuperado de <http://elies.rediris.es/elies9/4-2-1.htm>
- Picuino (2025). Diagramas de flujo. Recuperado de <https://www.picuino.com/es/prog-flowchart.html>
- Pressman, R. (2020). Ingeniería del software: Un enfoque práctico. McGraw-Hill. Recuperado de <https://www.mheducation.com.mx/connect-ingenieria-de-software-un-enfoque-practico-3-meses-9781456290047-latam>
- Ramírez, J. (2022). Desarrollo de un sistema de administración para bibliotecas escolares. Recuperado de <https://www.itca.edu.sv/wp-content/themes/elaniin-itca/docs/2022-Diseno-de-sistema-de-control-bibliotecario-escolar.pdf>
- ResearchGate (2025). Diagrama de actividades. Recuperado de https://www.researchgate.net/figure/Figura-311-Diagrama-de-actividades-para-la-configuracion-de-la-aplicacion_fig3_333703737
- Rodríguez, C., & Sánchez, D. (2021). Sistema de facturación electrónica para pequeñas y medianas empresas. Recuperado de <https://www.empresas.mx/startupguide/sistemasdefacturacion/proyecto>

- Sánchez, B (2023) Modelos de datos. Recuperado de <https://cursos.aiu.edu/base%20de%20datos%20SOG/Sesi%C3%B3n%2004.pdf>
- Schneider, S. (2022). IA en el futuro de la informática. Springer Recuperado de https://link.springer.com/chapter/10.1007/978-3-031-47324-1_1
- Schwaber, K., & Sutherland, J. (2020). La guía de Scrum. Scrum.org. Recuperado de <https://www.scrum.org/resources/scrum-guide>
- Sebesta, R. W. (2020). Conceptos de lenguaje de programación. concepto Recuperado de <https://concepto.de/lenguaje-de-programación/>
- Sebesta, R. W. (2020). Concepts of Programming Languages. Pearson. Recuperado de <https://www.pearson.com/en-us/subject-catalog/p/conceptsofprogrammilinguages/P200000003361/90135102268>
- Sommerville, I. (2021). Software Engineering. Pearson. Recuperado de <https://www.pearson.com/en-us/subject-catalog/p/software-engineering>
- Tanenbaum, A. S. (2019). Structured Computer Organization Recuperado de <https://cscknu.github.io/sysprog/books/Andrew%20S.%20Tanenbaum%20%20Structured%20Computer%20Organization.pdf>
- Torres, A. (2022). Aplicación web para la gestión de ventas e inventarios en tiendas de conveniencia. Universidad Politécnica de Aguascalientes. Recuperado de <https://revistapolitecnicaags.upa.edu.mx/wp-content/uploads/2024/07/V3124.pdf>
- Vásquez, L. (2020). Plataforma web para la gestión de pedidos en línea en restaurantes. Centro de Investigación en Tecnología y Negocios. Recuperado de <https://bibdigital.epn.edu.ec/bitstream/15000/10337/3/CD-6157.pdf>

Welling, L., & Thomson, L. (2021). ¿Qué es PHP? PHP. Recuperado de <https://www.php.net/manual/es/intro-what-is.php>