



INSTITUTO TECNOLÓGICO DEL VALLE DE OAXACA

SISTEMA WEB PARA LA EMPRESA “SANDY’S PIZZA”

PROYECTO QUE PRESENTA:

Karen Magali Ruiz López
Mareli Jollet Aragón Antonio
Radai Manzano Ramos
Demetrio Montaña Perez

Como requisito para acreditar la materia de Desarrollo e implementación de Sistemas de Información en:

INGENIERIA INFORMÁTICA



AGRADECIMIENTOS

En primer lugar, queremos expresar nuestro más sincero agradecimiento a nuestras familias, por su apoyo incondicional, comprensión y motivación durante el desarrollo de este proyecto. Su respaldo constante ha sido fundamental para mantenernos firmes en cada etapa, brindándonos la confianza y el ánimo necesarios para superar los retos y alcanzar esta meta académica y profesional. Asimismo, agradecemos profundamente al Instituto Tecnológico por el acompañamiento académico y por las herramientas proporcionadas a lo largo de nuestra formación. En especial, queremos reconocer y agradecer a nuestros docentes, quienes con su dedicación, paciencia y orientación nos guiaron en el proceso de aprendizaje, permitiéndonos aplicar los conocimientos adquiridos en la implementación de un sistema web real y funcional. De igual manera, expresamos nuestro agradecimiento a la empresa SANDY'S PIZZA, por brindarnos la oportunidad de llevar a cabo este proyecto en un entorno práctico, permitiéndonos integrar la teoría con la práctica y adquirir experiencias valiosas que fortalecen nuestro desarrollo profesional. Finalmente, queremos agradecer a cada integrante del equipo por su esfuerzo, responsabilidad y compromiso.

RESUMEN

El presente proyecto consiste en el desarrollo e implementación de un sitio web para la pizzería SanDy's Pizza, con el objetivo de resolver la falta de una plataforma digital que permita a los clientes consultar el menú, realizar pedidos en línea y contactar fácilmente con la empresa. Actualmente, la pizzería carece de herramientas tecnológicas que apoyen la gestión de pedidos y la promoción de sus productos, lo que limita su alcance y competitividad en el mercado. Para atender esta problemática se aplicó una metodología basada en el ciclo de desarrollo de software, que incluyó las fases de planeación, análisis de requisitos, diseño, implementación y pruebas. En el desarrollo se emplearon tecnologías como HTML, CSS, JavaScript y PHP, así como un sistema gestor de bases de datos (MySQL/MariaDB) para el almacenamiento y administración de la información relacionada con clientes, productos y ventas. El resultado fue la creación de un sistema web funcional que integra un catálogo digital del menú, un módulo de pedidos en línea, un apartado de contacto y un panel de gestión para la administración interna de pedidos y ventas. Se concluye que la implementación de este sitio web contribuye significativamente a mejorar la eficiencia operativa de la pizzería, incrementar su visibilidad digital y ofrecer un servicio más accesible y práctico a los clientes, fortaleciendo así su posición competitiva en el mercado local.

DEDICATORIA

Dedicamos este proyecto a nuestras familias, por su apoyo incondicional, comprensión y paciencia en cada etapa de nuestra formación académica y personal. Su confianza y aliento nos dieron la fuerza necesaria para superar los retos y llegar hasta aquí. También dedicamos este logro a nuestros docentes y compañeros, quienes con sus enseñanzas, consejos y colaboración han sido parte fundamental de nuestro crecimiento profesional. Finalmente, lo dedicamos a todas aquellas personas que creyeron en nosotros y nos motivaron a continuar, ya que este resultado no solo representa nuestro esfuerzo, sino también el respaldo y cariño de quienes nos acompañaron en este camino.

ÍNDICE GENERAL

Página

I. GENERALIDADES DEL PROYECTO.....	10
1.1 Introducción	11
1.2 Descripción de la empresa	13
1.3 Misión	14
1.4 Visión	14
1.5 Valores.....	14
1.6 Problemáticas a resolver, priorizándolos	16
1.7 Objetivos.....	17
1.7.1 General.....	16
1.7.2 Específicos.....	16
1.8 Justificación	17
1.8.1 Social:.....	17
1.8.2 Académica:.....	17
 II. MARCO TEÓRICO.....	 20
2.1 Software.....	22
2.2 Aplicaciones web	22
2.2.1 Tipos de aplicaciones web.....	23
2.3 Base de datos	22
2.4 Dominio web	22
2.5 Hosting web	22
2.6 Metodología de desarrollo	27
2.7 Requisitos del sistema	27
2.8 Diseño de la interfaz de usuario	28
2.9 Seguridad en sistemas de software	28
2.10 Experiencia de usuario (UX)	29
2.11 Algoritmo.....	30
2.12 Lenguajes de Programación.....	30
2.13 Frameworks y Librerías	31

2.14 Diseño responsivo	31
2.15 Rendimiento.....	32
2.16 Accesibilidad web.....	33
2.17 Escalabilidad de sistemas web	32
2.18 Pruebas de usabilidad.....	34
2.19. Adaptabilidad de contenidos.....	35
 III.PROCEDIMIENTO DE PRIMER SPRINT.....	36
3.1 Historias de usuario	36
3.1.1 Priorización de historias de usuario.....	37
3.2 Desglosamiento de historias de usuario.....	38
3.3 Caso de uso.....	49
3.3.1 Caso de uso HU-01: Login.....	42
3.3.2 Caso de uso HU-02: Cliente	43
3.3.3 Caso de uso HU-03: Administrador... ..	44
3.4 Modelo relacional... ..	45
3.5 Wireframes.....	46
3.6 Diagrama de Clases.....	50
3.7 Diagrama de secuencia.....	51
3.7.1 Diagrama de secuencia de HU-01: Login.....	51
3.7.2 Diagrama de secuencia de HU-02: Cliente.....	52
3.7.3 Diagrama de secuencia de registrar empleado.....	53
3.7.4 Diagrama de secuencia de visualizar empleado.....	54
3.7.5 Diagrama de secuencia de actualizar empleado.....	55
3.7.6 Diagrama de secuencia de eliminar empleado.....	56
3.8 Sprint backlog.....	57

INDICE DE CUADROS

Cuadro	Página
CUADRO 1. DESCRIPCIÓN DE HISTORIAS DE USUARIO DEL SISTEMA.	37
CUADRO 2. PRIORIZACIÓN DE HISTORIAS DE USUARIO.....	38
CUADRO 3. HU-01: ACCEDER AL SISTEMA (LOGIN DE USUARIOS).	39
CUADRO 4. HU-02: CAMBIAR CONTRASEÑA.....	40
CUADRO 5. HU-03: GESTIÓN DE EMPLEADOS.	41

INDICE DE FIGURAS

Figura	Pagina
FIGURA 1. MICROLOCALIZACIÓN DE DONDE SE ENCUENTRA EL ESTABLECIMIENTO.	9
FIGURA 2. REQUERIMIENTOS DEL SISTEMA DE UN PROYECTO..	27
FIGURA 3. PROCESO DE MEJORA CONTINUA MEDIANTE PRUEBAS DE USABILIDAD. ...	31
FIGURA 4. DIAGRAMA DE CASO DE USO “LOGIN”	42
FIGURA 5. CASO DE USO HU-02: CLIENTE.....	43
FIGURA 6. CASO DE USO DE GESTIÓN DE EMPLEADOS.....	44
FIGURA 7. MODELO RELACIONAL.....	45
FIGURA 8. WIREFRAME DE LOGIN PARA INICIAR SESIÓN	46
FIGURA 9. ACCESO CLIENTE	46
FIGURA 10. ACCESO ADMINISTRADOR.....	47
FIGURA 11. RECUPERAR CONTRASEÑA CLIENTE-ADMINISTRADOR..	47
FIGURA 12. NUEVA CONTRASEÑA CLIENTE-ADMINISTRADOR.	48
FIGURA 13. PANTALLA DE BIENVENIDA CLIENTE.....	48
FIGURA 14. CRUD DE ADMINISTRADOR DEL SISTEMA.....	49
FIGURA 15. DIAGRAMA DE CLASES.....	50
FIGURA 16 DIAGRAMA DE SECUENCIA DE HU-01: LOGIN.....	51
FIGURA 17. DIAGRAMA DE SECUENCIA DE HU-02: CLIENTE.	52
FIGURA 18. DIAGRAMA DE SECUENCIA ADMINISTRADOR (REGISTRAR EMPLEADO).	53

FIGURA 19. DIAGRAMA DE SECUENCIA VISUALIZAR EMPLEADO.....	54
FIGURA 20. DIAGRAMA DE SECUENCIA ACTUALIZAR EMPLEADO.....	55
FIGURA 21. DIAGRAMA DE SECUENCIA ELIMINAR EMPLEADO	56
FIGURA 22. SPRINT BACKLOG.....	57

I.GENERALIDADES DEL PROYECTO

El proyecto tiene como finalidad diseñar e implementar un sitio web para la pizzería SanDy's Pizza, con el propósito de integrar soluciones tecnológicas que optimicen su funcionamiento interno y fortalezcan su presencia digital. Para ello, se contempla el uso de herramientas de desarrollo web, lenguajes de programación, bases de datos y metodologías de ingeniería de software que garanticen la creación de una plataforma funcional, segura y accesible. El sistema permitirá gestionar pedidos en línea, ofrecer información clara a los clientes y mejorar la organización interna del negocio mediante reportes y módulos administrativos. Asimismo, este proyecto se desarrolla con un enfoque académico y práctico, ya que fomenta la aplicación de conocimientos adquiridos en el área de programación, bases de datos, diseño de interfaces y gestión de proyectos, además de impulsar el trabajo colaborativo entre los integrantes del equipo.

1.1 Introducción

El presente proyecto consiste en el desarrollo e implementación de un sistema web para la empresa SanDy's Pizza, con el propósito de modernizar sus procesos internos, optimizar la gestión de pedidos y fortalecer su presencia digital. Hasta el momento, el trabajo ha abarcado distintas etapas del ciclo de vida del software, iniciando con la planeación y análisis de los requerimientos, seguido por el diseño conceptual y la documentación técnica del sistema. En el capítulo I, se presentan las generalidades del proyecto, incluyendo la descripción de la empresa, su misión, visión y valores, así como las problemáticas identificadas, los objetivos planteados y la justificación tanto social como académica de la propuesta. El capítulo II aborda el marco teórico, donde se explican los conceptos fundamentales relacionados con el software, las aplicaciones web, las bases de datos, los lenguajes de programación, la metodología de desarrollo, la seguridad, la usabilidad, el diseño responsivo y otros elementos técnicos que sustentan el desarrollo del sistema. Posteriormente, en el capítulo III, se documenta el procedimiento del primer sprint, el cual incluye la elaboración de historias de usuario, su priorización y desglosamiento, así como la representación de los casos de uso que describen las funcionalidades principales del sistema. También se integran los modelos relacionales, wireframes, el diagrama de clases y los diagramas de secuencia, que muestran el flujo de interacción entre los usuarios y el sistema en los

procesos de inicio de sesión, recuperación de contraseña y gestión de empleados. En conjunto, este proyecto combina la teoría con la práctica, aplicando los conocimientos adquiridos en el área de desarrollo de software para ofrecer una solución tecnológica funcional, segura y adaptable a las necesidades reales de SanDy's Pizza. De esta forma, se sientan las bases para continuar con las siguientes fases de implementación y pruebas del sistema.

1.2 Descripción de la empresa

La empresa denominada SanDy's Pizza se encuentra ubicada en Morelos S/N, Cabecera Municipal, Santa Cruz Xoxocotlán, C.P. 71230, Oaxaca, como se muestra en la (Figura 1).



Figura 1. Microlocalización de donde se encuentra el establecimiento.
Fuente: <https://www.google.com/maps/place/Sandys+pizza>

Se dedica principalmente a la preparación y venta de pizzas artesanales, ofreciendo además complementos como bebidas, postres y entradas. El negocio busca destacar por la calidad de sus ingredientes, el sabor de sus productos y la atención personalizada a sus clientes. Se trata de un establecimiento de tamaño pequeño a mediano, con un mercado en crecimiento, ya que en la zona existen otros negocios de comida rápida y pizzerías en un rango de distancia corta, lo que representa una competencia directa.

1.3 Misión

Nuestra misión es ofrecer a la comunidad pizzas artesanales y productos complementarios de la más alta calidad, elaborados con ingredientes frescos y a precios accesibles. Buscamos brindar un servicio rápido, eficiente y personalizado que satisfaga el paladar y las expectativas de nuestros clientes, creando un ambiente acogedor y familiar tanto en nuestro local como a través de nuestra plataforma en línea.

1.4 Visión

Aspiramos a convertirnos en la pizzería preferida de la región, reconocida por la calidad de nuestros productos, la excelencia en la atención al cliente y la innovación tecnológica que facilite la experiencia de compra. Nos proyectamos como un negocio en constante crecimiento, integrando herramientas digitales que nos permitan ampliar nuestro alcance y consolidar nuestra presencia en el mercado.

1.5 Valores

Nos comprometemos a ofrecer productos frescos y de calidad, garantizando satisfacción en cada pedido, y actuamos con honestidad.

1.6 Problemáticas a resolver, priorizándolos

Falta de presencia y visibilidad en línea: La pizzería no cuenta con sitio web ni plataformas digitales que muestren su menú, promociones o servicios, limitando su competitividad y alcance a nuevos clientes.

Limitación en las ventas y pedidos: Los clientes solo pueden hacer pedidos de manera presencial o por teléfono, lo que reduce las oportunidades de ingresos y dificulta la expansión del negocio.

Procesos de entrega y gestión de pedidos ineficientes: La ausencia de un sistema automatizado provoca errores, retrasos y problemas de organización en la recepción y entrega de pedidos

1.7 Objetivos

1.7.1 General

Desarrollar e implementar un sitio web para la pizzería SanDy's Pizza que permita a los clientes consultar el menú, realizar pedidos en línea y establecer contacto con la empresa, con el fin de mejorar la gestión de pedidos, aumentar la visibilidad digital y optimizar la relación con los consumidores.

1.7.2 Específicos

- Diseñar una interfaz web interactiva y fácil de usar que muestre de manera clara el menú, promociones y productos disponibles de la pizzería.
- Implementar un sistema de pedidos en línea que permita a los clientes seleccionar productos, personalizar su orden y enviarla de manera digital.

- Desarrollar un módulo de gestión de pedidos y ventas para que la empresa administre de forma organizada los pedidos recibidos y reduzca errores en la atención.
- Incorporar un apartado de contacto en línea que facilite la comunicación directa entre los clientes y la empresa (consultas, quejas o sugerencias).

1.8 Justificación

La realización de este proyecto resulta de gran importancia ya que busca atender una de las principales limitaciones de la pizzería SanDy's Pizza: la ausencia de una plataforma digital que facilite la consulta del menú, la gestión de pedidos y la comunicación con los clientes. En un contexto en el que la tecnología se ha convertido en un factor determinante para la competitividad de los negocios, contar con un sitio web moderno y funcional representa una necesidad estratégica.

1.8.1 Social:

Este estudio es conveniente porque permitirá a la empresa ampliar su alcance y mejorar su presencia en el mercado local, al ofrecer a los clientes

una herramienta práctica y accesible que simplifique la realización de pedidos y fortalezca la experiencia de compra. Además, la digitalización de procesos contribuirá a optimizar la gestión interna, reduciendo errores en el control de pedidos y facilitando el análisis de ventas mediante reportes automatizados. Los principales beneficiados con los resultados serán tanto los clientes, quienes obtendrán un servicio más rápido y eficiente, como la empresa, que logrará incrementar sus ventas, fidelizar a sus consumidores y mejorar su posicionamiento frente a la competencia. Asimismo, el proyecto tiene un valor académico, ya que permite aplicar los conocimientos adquiridos en desarrollo web, bases de datos y diseño de sistemas, aportando una experiencia práctica que refuerza la formación profesional de los estudiantes involucrados

Los resultados de este proyecto pueden tener implicaciones positivas para futuros trabajos, al servir como referencia para la implementación de sistemas similares en negocios de comida rápida u otros giros comerciales que enfrenten problemáticas relacionadas con la gestión de pedidos y la visibilidad en línea. De esta forma, se demuestra que la aplicación de soluciones tecnológicas es una vía efectiva para resolver problemas prácticos y proponer nuevas estrategias de mejora en la administración de pequeñas y medianas empresas.

1.8.2 Académica:

Este proyecto brinda a los estudiantes la oportunidad de aplicar de manera práctica los conocimientos adquiridos en áreas como diseño y desarrollo web, bases de datos, programación y gestión de sistemas. La experiencia obtenida fortalece la formación profesional, fomenta el trabajo en equipo y contribuye a desarrollar competencias tecnológicas que serán útiles en futuros entornos laborales.

II. MARCO TEÓRICO

En la actualidad, la mayoría de las empresas buscan implementar herramientas tecnológicas que les permitan ser más competitivas y responder a las necesidades de sus clientes. El uso de las Tecnologías de la Información y Comunicación (TIC) se ha convertido en una estrategia esencial para mejorar la calidad de los productos y servicios, optimizar los procesos internos y ampliar el alcance de los negocios en el mercado digital. Particularmente, las pequeñas y medianas empresas de comida rápida, como las pizzerías, enfrentan la necesidad de adaptarse a un entorno en el que los clientes valoran la rapidez, la facilidad de acceso a la información y la posibilidad de realizar pedidos en línea. La implementación de un sitio web que integre estas funcionalidades representa una solución tecnológica que no solo mejora la atención al cliente, sino que también contribuye al crecimiento del negocio.

2.1 Software

Según Sommerville (2011), el software es el conjunto de programas, instrucciones y datos que permiten a una computadora o dispositivo electrónico ejecutar tareas específicas, ya sean simples o complejas. Se divide principalmente en software de sistema y software de aplicación. El primero incluye los sistemas operativos y controladores que gestionan y coordinan el funcionamiento del hardware, asegurando que los distintos componentes del equipo puedan comunicarse y operar correctamente. El segundo, el software de aplicación, está diseñado para realizar funciones concretas para el usuario, como procesadores de texto, navegadores web, programas de diseño gráfico o aplicaciones móviles. Además, el software puede ser propietario, desarrollado y controlado por una empresa o persona que restringe su uso y modificación, o de código abierto, disponible para que cualquier usuario lo utilice, modifique y distribuya libremente. Su desarrollo implica planificación, programación, pruebas y mantenimiento continuo para garantizar un funcionamiento eficiente, seguro y actualizado conforme a las necesidades de los usuarios.

2.2 Aplicaciones web

Una aplicación web es un programa de software que se ejecuta en servidores remotos y se accede a través de un navegador web, sin necesidad de instalar nada en el dispositivo del usuario. Estas aplicaciones permiten que múltiples usuarios interactúen en tiempo real con la información almacenada en el servidor, realizando tareas como enviar y recibir correos electrónicos, gestionar cuentas bancarias, realizar compras en línea, acceder a plataformas educativas o colaborar en documentos compartidos. La arquitectura cliente-servidor de las aplicaciones web permite centralizar la lógica del negocio y los datos, lo que facilita su actualización, mantenimiento y escalabilidad, además de garantizar la seguridad de la información. Las aplicaciones web también son multiplataforma, ya que se pueden usar desde computadoras, tabletas o teléfonos inteligentes, adaptándose a distintos tamaños de pantalla mediante diseño responsivo. Su funcionamiento depende de tecnologías como HTML, CSS, JavaScript, PHP, AJAX y bases de datos, que permiten la interacción dinámica entre el cliente y el servidor. Además, estas aplicaciones pueden ofrecer servicios personalizados, funciones avanzadas de seguridad como autenticación, autorización y cifrado de datos mediante HTTPS. (Pressman, 2010; W3C, 2023).

2.2.1 Tipos de aplicaciones web

- Aplicaciones web estáticas: Se caracterizan por tener un contenido fijo y no interactivo, por lo que son comunes para mostrar información básica en blogs, portfolios o landing pages informativas. La ventaja es que son rápidas y de fácil lectura, aunque no tienen mucha interactividad y su contenido es poco personalizable.
- Aplicaciones web dinámicas: Su contenido cambia en respuesta a las interacciones del usuario y se basan en bases de datos y scripts del lado del servidor para generar ese dinamismo. Usan tecnologías como AJAX para actualizar el contenido sin recargar la página. Algunos ejemplos son los foros o las redes sociales, sitios altamente interactivos y con una personalización del contenido en tiempo real.
- Aplicaciones web de múltiples páginas (MPA): Su estructura es la tradicional, con múltiples páginas HTML para que cada acción del usuario cargue una nueva página del servidor. Se puede apreciar en e-commerce, portales de noticias o plataformas educativas, lugares con un SEO sencillo, aunque con una experiencia de flujo menos fluida.¹

¹ <https://www.esic.edu/rethink/tecnologia/que-son-las-aplicaciones-web-c>

2.3 Base de datos

Según Sommerville (2011), una base de datos es un conjunto organizado de información o datos que se almacenan de manera estructurada y se gestionan mediante un Sistema de Gestión de Bases de Datos (SGBD). Estas bases de datos permiten que los usuarios y las aplicaciones accedan, consulten, modifiquen y administren la información de manera eficiente, evitando redundancias y asegurando la integridad y consistencia de los datos. Además, facilitan la recuperación rápida de información, el manejo de grandes volúmenes de datos y la implementación de reglas de seguridad y control de acceso. Existen distintos modelos de bases de datos, entre los más comunes se encuentran el modelo relacional, donde la información se organiza en tablas relacionadas mediante claves primarias y foráneas; el modelo jerárquico, que organiza los datos en forma de árbol; el modelo de red, que permite relaciones más complejas entre registros; y las bases de datos NoSQL, diseñadas para manejar grandes volúmenes de datos no estructurados, como documentos, grafos o pares clave-valor, siendo especialmente útiles en aplicaciones web modernas. El diseño de una base de datos implica planificar cuidadosamente la estructura de las tablas, relaciones, índices y restricciones, con el objetivo de optimizar el rendimiento y garantizar que los datos sean precisos y confiables. En el contexto de aplicaciones empresariales, como un sistema de gestión para una pizzería, las bases de datos son fundamentales para registrar clientes, pedidos, productos, inventarios y ventas, proporcionando información estratégica para la toma d

decisiones y el crecimiento del negocio. Además, las bases de datos permiten automatizar procesos, generar reportes y análisis en tiempo real, incrementando la eficiencia y reduciendo la posibilidad de errores humanos.

2.4 Dominio web

Un dominio web es un nombre único que identifica a un sitio en Internet y actúa como una dirección legible para los usuarios, en lugar de tener que recordar direcciones IP numéricas. Está conformado por un nombre y una extensión (.com, .org, .edu, .mx, entre otros) y se administra mediante entidades registradoras autorizadas. Los dominios forman parte del Sistema de Nombres de Dominio (DNS), el cual traduce estos nombres en direcciones IP que permiten la comunicación entre computadoras en la red (Tanenbaum & Wetherall, 2011).

2.5 Hosting web

El hosting web es un servicio que permite almacenar y publicar los archivos, bases de datos y aplicaciones de un sitio en un servidor conectado permanentemente a Internet, de manera que los usuarios puedan acceder a él en cualquier momento y lugar. Este servicio incluye espacio en disco, ancho

de banda, cuentas de correo electrónico y soporte técnico, entre otros recursos. Según Stallings (2017), el hosting es un componente esencial de la infraestructura de Internet, ya que proporciona los medios para que las páginas web estén disponibles de forma continua para los usuarios (Stallings, 2017).

2.6 Metodología de desarrollo

Una metodología de desarrollo es un marco estructurado que guía todo el proceso de creación de software, desde la concepción de la idea hasta su despliegue y mantenimiento. Incluye fases como análisis de requisitos, diseño, codificación, pruebas, implementación y mantenimiento, y define roles, tareas, entregables y criterios de calidad. Sirve para coordinar al equipo, optimizar recursos y tiempos, disminuir riesgos, y asegurar que el producto final cumpla con las expectativas del cliente y los estándares técnicos. Según la explicación de PegasusOne, estas metodologías combinan principios de planificación, control de calidad, flexibilidad ante cambios, iteraciones y comunicación continua entre stakeholders.²

² <https://www.pegasusone.com/what-is-software-development-methodology/>

2.7 Requisitos del sistema

Pressman (2010) señala que los requisitos constituyen el contrato entre el cliente y el desarrollador, pues establecen con claridad qué se espera del producto final. Los requisitos del sistema representan la base sobre la cual se construye un software, ya que describen lo que este debe realizar y las condiciones bajo las cuales debe operar. Se clasifican en funcionales, que determinan los servicios o funciones específicas del sistema, y no funcionales, que incluyen restricciones de desempeño, seguridad, usabilidad o confiabilidad. Una correcta definición de requisitos es esencial para reducir ambigüedades, guiar el diseño y evitar fallos en fases posteriores del desarrollo, como se observa en la (Figura 2).



Figura 2. Requerimientos del sistema de un proyecto. Fuente: <https://www.google.com/maps/place/Sandys+pizza>.

2.8 Diseño de la interfaz de usuario

El diseño de la interfaz de usuario (UI) se refiere a la creación de la parte visual e interactiva de un sistema con la cual el usuario establece contacto directo. Su objetivo es facilitar la comunicación entre el usuario y el sistema a través de elementos gráficos como botones, menús, tipografías, colores e iconos, garantizando que la interacción sea intuitiva, atractiva y eficiente. De acuerdo con Shneiderman y Plaisant (2010), un buen diseño de interfaz no solo debe ser estéticamente agradable, sino también coherente, predecible y accesible, de modo que contribuya a mejorar la experiencia general del usuario.

2.9 Seguridad en sistemas de software

Según Stallings (2012), la seguridad informática debe integrarse en todas las fases del desarrollo de software, ya que prevenir vulnerabilidades desde el diseño es más eficiente y menos costoso que corregirlas en etapas posteriores. La seguridad en sistemas de software consiste en aplicar principios, técnicas y mecanismos destinados a proteger la información y los recursos frente a accesos no autorizados, uso indebido, fallos o ataques maliciosos.

2.10 Experiencia de usuario (UX)

La experiencia de usuario (UX) se refiere a la percepción, emociones y actitudes de una persona al interactuar con un sistema, producto o servicio digital. Incluye factores como la facilidad de uso, eficiencia, accesibilidad, satisfacción, y percepción estética, así como la forma en que el usuario se siente al completar tareas o alcanzar sus objetivos. El diseño de UX busca que la interacción sea no solo funcional, sino también agradable, intuitiva y memorable, fomentando la confianza y fidelización del usuario. Además, la UX implica analizar y comprender el comportamiento de los usuarios mediante pruebas de usabilidad, entrevistas, encuestas y análisis de métricas, permitiendo iterar y mejorar continuamente el sistema.³

³ <https://www.interaction-design.org/literature/topics/ux-design>

2.11 Algoritmo

Knuth (1997) enfatiza que un buen algoritmo debe ser correcto, comprensible, eficiente y finito, ya que su diseño adecuado impacta directamente en el rendimiento y la confiabilidad del sistema. Los algoritmos se aplican en operaciones como búsquedas, ordenamientos, cálculos y procesos lógicos, siendo fundamentales en cualquier sistema web o aplicación digital. Un algoritmo es un conjunto finito de instrucciones o pasos claramente definidos que permiten resolver un problema específico o realizar una tarea determinada. En programación, los algoritmos constituyen la base para el desarrollo de funciones y procedimientos dentro de un software, asegurando que los procesos se ejecuten de manera eficiente, precisa y predecible.

2.12 Lenguajes de Programación

Según Sebesta (2012), estos lenguajes proporcionan una sintaxis y semántica específicas, que determinan cómo se estructuran los programas y cómo se comunican con el hardware. Los lenguajes de alto nivel, como JavaScript, Python y PHP, son más cercanos al lenguaje humano y permiten

desarrollar aplicaciones complejas rápidamente, mientras que los de bajo nivel, como ensamblador, ofrecen control detallado del hardware, pero requieren más precisión y conocimiento técnico. Los lenguajes de programación son herramientas formales que permiten escribir instrucciones comprensibles para una computadora, facilitando la creación, control y manipulación de software.

2.13 Frameworks y Librerías

Gamma, Helm, Johnson y Vlissides (1995) destacan que el uso de frameworks y librerías reduce errores, acelera el desarrollo y permite aplicar patrones de diseño comprobados, siendo esenciales para la creación de aplicaciones web modernas y eficientes. Un framework es un conjunto organizado de herramientas, componentes y estructuras que proporciona un esqueleto sobre el cual se puede construir software, facilitando la reutilización de código y la estandarización del desarrollo. Por otro lado, una librería es un conjunto de funciones o rutinas predefinidas que simplifican tareas específicas.

2.14 Diseño responsivo

Marcotte (2011) señala que el diseño responsivo no solo mejora la legibilidad y usabilidad, sino que también optimiza la navegación, reduce la necesidad de desplazamiento horizontal y aumenta la satisfacción del usuario. Este enfoque es fundamental en la era de múltiples dispositivos, donde usuarios acceden a sitios web desde computadoras, tabletas y teléfonos inteligentes. El diseño responsivo es una técnica de diseño web que permite que una aplicación o sitio web se adapte automáticamente al tamaño y resolución de la pantalla del dispositivo del usuario, garantizando una experiencia uniforme y accesible.

2.15 Rendimiento

Pressman (2010) explica que un buen rendimiento en aplicaciones web incluye tiempos de respuesta rápidos, gestión eficiente de la memoria y estabilidad bajo carga, lo que mejora la experiencia del usuario y la confiabilidad del sistema. El rendimiento es la capacidad de un sistema para realizar sus funciones de manera eficiente, utilizando recursos de hardware y software de forma óptima.

2.16 Accesibilidad web

Según Caldwell, Cooper, Reid y Vanderheiden (2008), la accesibilidad web no solo garantiza igualdad de acceso a la información, sino que también mejora la usabilidad general del sitio, fomentando la inclusión y cumpliendo con estándares internacionales como las Pautas de Accesibilidad para el Contenido Web (WCAG). La accesibilidad web se refiere al diseño y desarrollo de sitios web y aplicaciones digitales de manera que todas las personas, incluidas aquellas con discapacidades visuales, auditivas, motrices o cognitivas, puedan percibir, entender, navegar e interactuar con la información de manera efectiva.

2.17 Escalabilidad de sistemas web

Según Bass, Clements y Kazman (2012), la escalabilidad es un atributo crítico de calidad en el software, ya que permite a las empresas ampliar sus servicios de manera eficiente, mantener la satisfacción del usuario y optimizar el uso de recursos tecnológicos. La escalabilidad se refiere a la capacidad de un sistema para adaptarse a incrementos en la carga de trabajo, el número de usuarios o el volumen de datos sin que se degrade su rendimiento o funcionalidad.

2.18 Pruebas de usabilidad

Nielsen (2012) destaca que la usabilidad no solo influye en la satisfacción del usuario, sino también en la productividad, la reducción de errores y la adopción del sistema, siendo un factor crítico para el éxito de cualquier aplicación web. La retroalimentación obtenida durante estas pruebas es utilizada para realizar ajustes iterativos en el diseño y la interfaz, promoviendo la mejora continua del sistema y la fidelización de los usuarios. Son evaluaciones sistemáticas diseñadas para medir qué tan fácil, eficiente y satisfactorio es para los usuarios interactuar con un sistema o aplicación web, permiten identificar problemas en la navegación, la disposición de los elementos, el flujo de tareas y la funcionalidad general del sistema, asegurando que la experiencia del usuario sea intuitiva y agradable, como se aprecia en la (Figura 4).

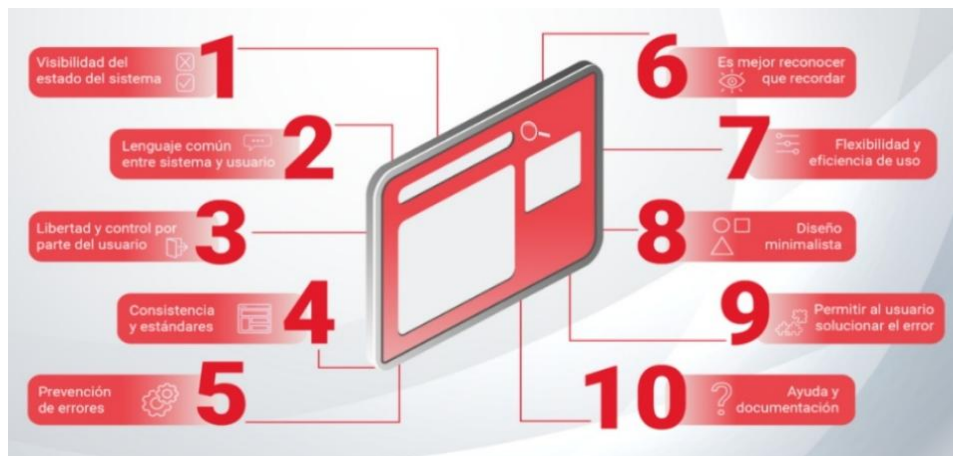


Figura 3. Proceso de Mejora Continua mediante Pruebas de Usabilidad. Fuente: <https://www.google.com/url?sa=i&url=https%3A%2F%2Fenriqueelite>

2.19. Adaptabilidad de contenidos

Según Ethan Marcotte en *Responsive Web Design* (2010), la adaptabilidad de contenidos se refiere a la capacidad de un sitio web para ajustar automáticamente el diseño, las imágenes y la disposición de los elementos según las características del dispositivo y la resolución de pantalla del usuario. Esta práctica no solo mejora la experiencia de usuario, sino que también optimiza la accesibilidad y asegura que la información se presente de manera clara y funcional en cualquier entorno digital.

III.PROCEDIMIENTO DEL PRIMER SPRINT

3.1 Historias de usuario

En este apartado se presentan las historias de usuario que conforman el backlog funcional del sistema, las cuales describen de forma clara y sencilla las funcionalidades requeridas desde la perspectiva de los distintos actores que interactúan con la aplicación. Su propósito es expresar qué necesita el usuario, con qué finalidad y qué beneficio obtiene, siguiendo la estructura: Como [tipo de usuario], quiero [acción o funcionalidad] para [obtener un resultado o beneficio]. Este apartado tiene como objetivo proporcionar una visión funcional del sistema que sirva como base para la planificación de los sprints de desarrollo, la elaboración de casos de uso, diagramas de secuencia y pruebas funcionales. A continuación, en el Cuadro 1, se muestran en el las historias de usuario identificadas, clasificadas según el rol correspondiente y priorizadas conforme a su relevancia dentro del proyecto.

Cuadro 1. Descripción de historias de usuario del sistema.

Historia de usuario	Descripción	Prioridad
HU-01 — Iniciar sesión	Como usuario del sistema, quiero iniciar sesión utilizando mi nombre de usuario y contraseña, para poder acceder de forma segura a las funcionalidades permitidas según mi rol.	Alta
HU-02 — Cambiar contraseña	Como cliente, quiero cambiar y actualizar mi contraseña mediante una función de actualización (<i>update</i>), para poder acceder nuevamente al sistema en caso de olvido o pérdida de la contraseña.	Alta
HU-03 — Gestión de datos de empleados	Como administrador del sistema, quiero actualizar, visualizar, registrar, eliminar los datos de los empleados registrados, para que la información del personal se mantenga correcta, actualizada y confiable dentro del sistema de gestión.	Alta

Fuente: Elaboración propia (Sprint 1 – Historias de Usuario).

3.1.1 Priorización de historias de usuario

De acuerdo con el Cuadro 2, se realizó la priorización de las historias de usuario con el objetivo de definir el orden de desarrollo de las funcionalidades principales del sistema. En este Sprint se desarrollaron tres historias de usuario clasificadas con prioridad alta, correspondientes a funciones esenciales del sistema: el acceso de los usuarios al sistema (HU-01), el cambio de contraseña (HU-02) y la gestión de datos de empleados por parte del administrador (HU-03). Estas funcionalidades representan la base operativa inicial del proyecto,

permitiendo asegurar el control de acceso, la administración de credenciales y el manejo adecuado de la información del personal.

Cuadro 2. Priorización de historias de usuario.

ID	Área	Historia de Usuario	Prioridad	Sprint
HU-01	Usuario del sistema	Acceder al sistema	Alta	Sprint 1
HU-02	Cliente	Cambiar contraseña	Alta	Sprint 1
HU-03	Administrador	Gestión de datos de empleados	Alta	Sprint 1

Fuente: Elaboración propia a partir de las Historias de Usuario

3.2 Desglosamiento de historias de usuario

Como se observa en el Cuadro 3, el módulo de inicio de sesión (LOGIN) fue desarrollado con el objetivo de permitir que tanto los administradores como los clientes puedan acceder de forma segura al sistema mediante la validación de su nombre de usuario y contraseña. Este módulo garantiza que solo los usuarios con credenciales correctas ingresen, mostrando mensajes de error cuando los datos son incorrectos y redirigiendo al usuario a la interfaz correspondiente según su rol. Además, se priorizó que el acceso sea rápido, eficiente y sin complicaciones, cumpliendo con los criterios de aceptación establecidos y asegurando la protección de la información dentro del módulo de seguridad.

Cuadro 3. HU-01: Acceder al sistema (Login de usuarios).

HISTORIA DE USUARIO 1	
NÚMERO: HU-01	USUARIO: Admin y clientes
NOMBRE DE LA HISTORIA: Iniciar sesión	
PRIORIDAD: ALTA	IMPORTANCIA DE DESARROLLO: 100
TIEMPO ESTIMADO: 2 días	MÓDULO ASIGNADO: Seguridad
PROGRAMADORES RESPONSABLES: Radai Manzano Ramos	
DESCRIPCIÓN: Como usuario del sistema, quiero iniciar sesión utilizando mi nombre de usuario y contraseña, para poder acceder de forma segura a las funcionalidades permitidas según mi rol.	
CRITERIOS DE ACEPTACIÓN: <ul style="list-style-type: none"> - El sistema debe validar que el usuario y contraseña coincidan con los registros - Se debe mostrar un mensaje de error si las credenciales son incorrectas - El usuario debe ser redirigido a la interfaz correspondiente según su rol 	
OBSERVACIONES: <ul style="list-style-type: none"> - El login debe permitir acceso rápido y sin complicaciones para el usuario. 	

Fuente: Elaboración propia

Como se observa en el Cuadro 4, el módulo de cambio de contraseña fue diseñado para permitir que los clientes puedan restablecer su acceso al sistema en caso de olvido de su contraseña. Este proceso incluye la validación del usuario mediante su nombre o correo electrónico, la verificación de la existencia de la cuenta y la posibilidad de registrar una nueva contraseña que cumpla con los requisitos mínimos de seguridad, como tener al menos ocho caracteres. Además, el sistema notifica al usuario si el cambio fue exitoso o si ocurrió algún error, asegurando así una experiencia clara y segura dentro del módulo de seguridad.

Cuadro 4. HU-02: Cambiar contraseña.

HISTORIA DE USUARIO 2	
NÚMERO: HU-02	USUARIO: Clientes
NOMBRE DE LA HISTORIA: Cambiar contraseña	
PRIORIDAD: ALTA	IMPORTANCIA DE DESARROLLO: 100
TIEMPO ESTIMADO: 3 días	MÓDULO ASIGNADO: Seguridad
PROGRAMADORES RESPONSABLES: Radai Manzano Ramos	
DESCRIPCIÓN: Como cliente, quiero restablecer mi contraseña en caso de olvido, para poder acceder nuevamente al sistema.	
CRITERIOS DE ACEPTACIÓN: <ul style="list-style-type: none"> • El cliente puede ingresar su usuario o correo para identificar su cuenta. • El sistema valida que el usuario exista. • El cliente puede escribir su nueva contraseña. • La contraseña debe tener al menos 8 caracteres para ser aceptada. • El sistema muestra un mensaje indicando si el cambio fue exitoso o si ocurrió un error. 	
OBSERVACIONES: - El sistema debe permitir cambiar la contraseña de usuarios existentes, validando que tenga al menos 8 caracteres y mostrando mensajes claros de éxito o error.	

Fuente: Elaboración propia.

Como se observa en el Cuadro 5, el módulo de gestión de datos de empleados fue desarrollado para que el administrador pueda visualizar, registrar, actualizar y eliminar la información del personal de forma eficiente y controlada. Este módulo permite mantener una base de datos confiable y actualizada mediante funciones que facilitan la creación de nuevos registros, la edición de datos existentes, la eliminación con confirmación previa y la visualización completa del personal registrado. Además, se garantiza la integridad, seguridad y disponibilidad de los datos de empleados en todo momento, asegurando una correcta administración de la información dentro del sistema.

Cuadro 5. HU-03: Gestión de empleados.

HISTORIA DE USUARIO 3	
NÚMERO: HU-03	USUARIO: Admin
NOMBRE DE LA HISTORIA: Gestión de datos de empleados	
PRIORIDAD: ALTA	IMPORTANCIA DE DESARROLLO: 100
TIEMPO ESTIMADO: 5 días	MÓDULO ASIGNADO: Administración
PROGRAMADORES RESPONSABLES: Radai Manzano Ramos	
DESCRIPCIÓN: Como administrador del sistema, quiero actualizar, visualizar, registrar, eliminar los datos de los empleados registrados, para mantener la información del personal correcta, actualizada y confiable dentro del sistema de gestión.	
CRITERIOS DE ACEPTACIÓN: <ul style="list-style-type: none"> - Visualizar listado completo de empleados registrados - Crear nuevos registros de empleados - Editar y actualizar datos de empleados existentes - Eliminar empleados con confirmación previa 	
OBSERVACIONES: <ul style="list-style-type: none"> - Garantizar la integridad, seguridad y disponibilidad de los datos de empleados en todo momento. 	

Fuente: Elaboración propia.

3.3 Casos de uso

3.3.1 Caso de uso login

En este diagrama de casos de uso del "Sistema de Gestión" como se muestra en la Figura 4, se representa cómo interactúa el usuario con las funcionalidades principales del sistema, enfocándose en los procesos de inicio de sesión, recuperación de contraseña y registro. El actor "usuario" puede iniciar sesión, lo cual incluye la validación de credenciales, y en caso de fallo, se muestra un mensaje de error. Si el usuario no tiene cuenta, puede registrarse, lo que implica validar los datos ingresados; si esta validación falla, se genera un error de registro. Además, se contempla la opción de recuperar contraseña cuando el usuario lo solicita.

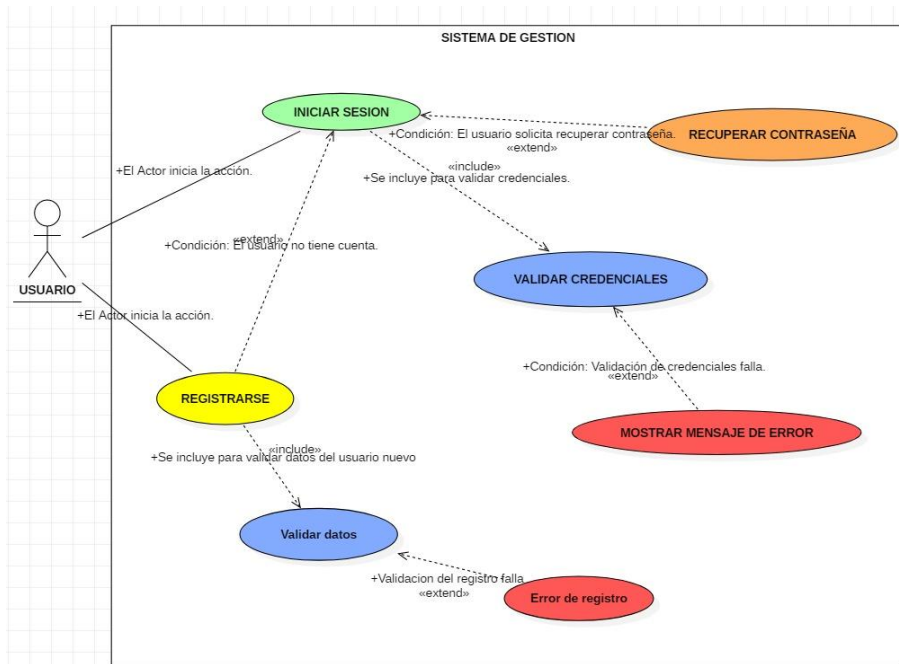


Figura 4. Diagrama de caso de uso "Login" Fuente: Elaboración propia.

3.3.2 Caso de uso HU-02: Cliente

Como se muestra en la figura 5, el proceso de actualización de contraseña dentro del Sistema de Gestión inicia con la acción del usuario, quien quiere modificar su contraseña. El sistema valida el formato de la nueva contraseña y, si cumple con los criterios establecidos, procede a guardarla en la base de datos. En caso de que la validación falle, se activa una ruta de excepción que muestra un mensaje de error al usuario. Este diagrama permite visualizar claramente el flujo de acciones, validaciones y manejo de errores involucrados en esta funcionalidad.

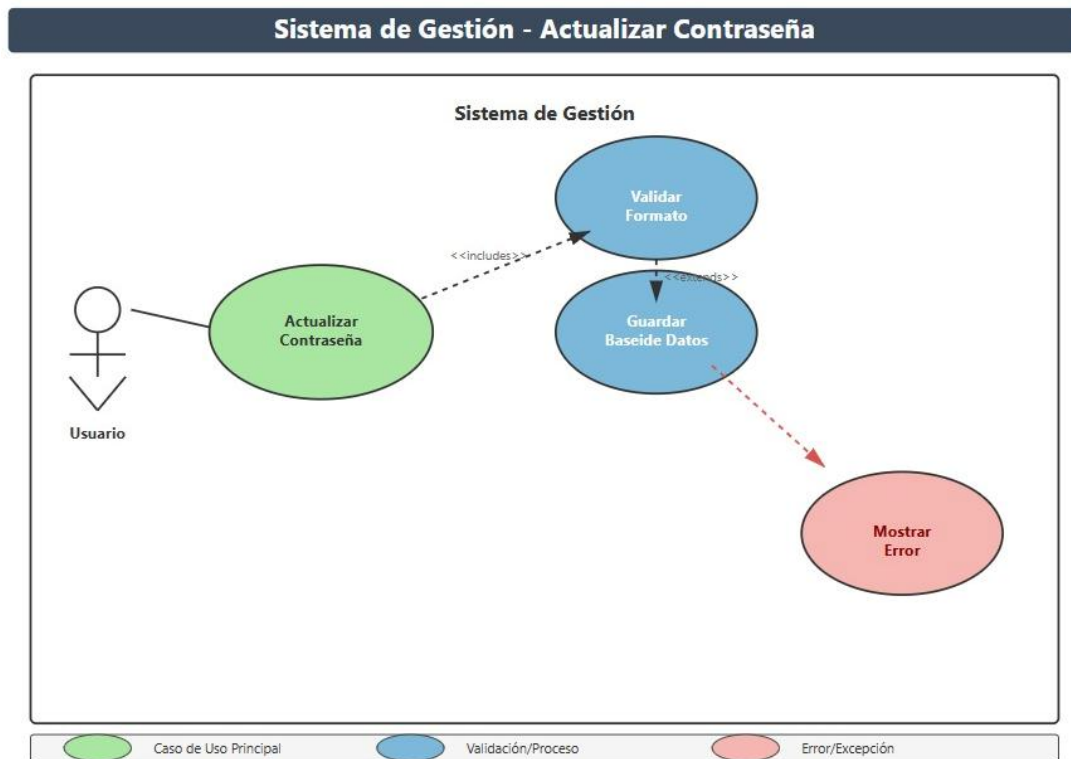


Figura 5. Caso de uso HU-02: Cliente Fuente: Elaboración propia.

3.3.3 Caso de uso HU-03: Administrador

En la a Figura 6 se representa el caso de uso del módulo "Administrar Usuarios" dentro del Sistema de Gestión. En él se muestra cómo el actor "Administrador" interactúa con las funciones clave del sistema, tales como registrar, actualizar, visualizar y eliminar usuarios. Cada acción está vinculada a procesos de validación de datos y manejo de errores, lo que permite asegurar la integridad de la información y una respuesta adecuada ante fallos. Este diagrama proporciona una visión estructurada de las operaciones administrativas disponibles, facilitando el análisis funcional del sistema.

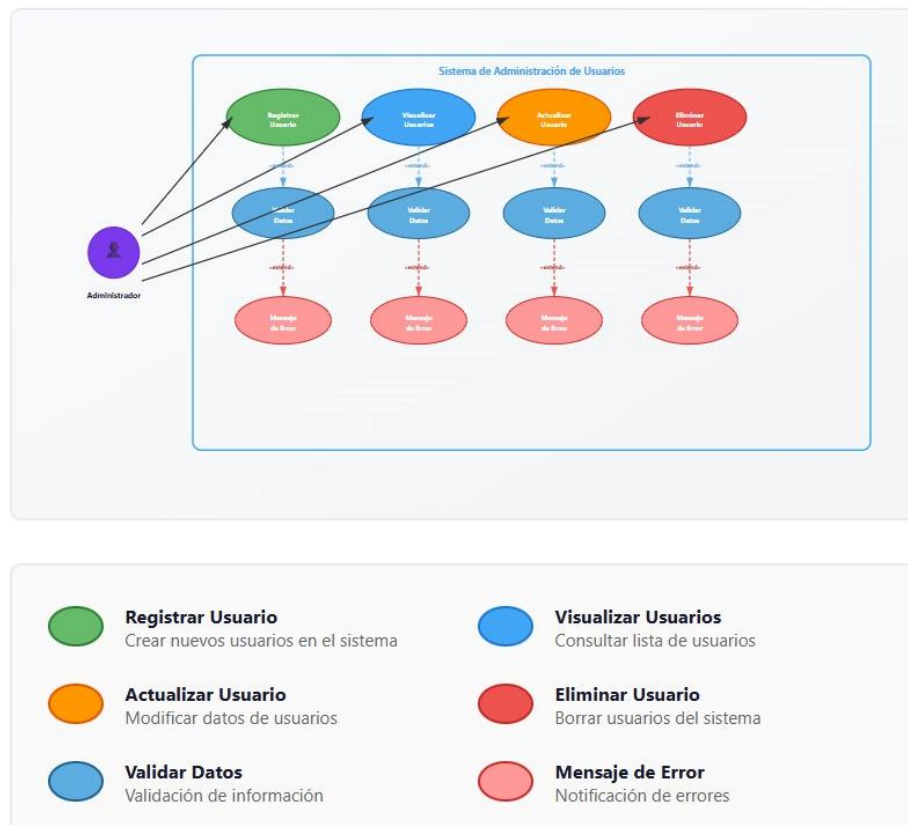


Figura 6. Caso de uso de gestión de empleados. Fuente: Elaboración propia.

3.4 Modelo relacional

Como se muestra en la figura 7, se representa el modelo entidad-relación del sistema propuesto, el cual permite visualizar las principales entidades involucradas, sus atributos y las relaciones entre ellas. Esta estructura facilita la organización de la información y respalda el diseño lógico de la base de datos.

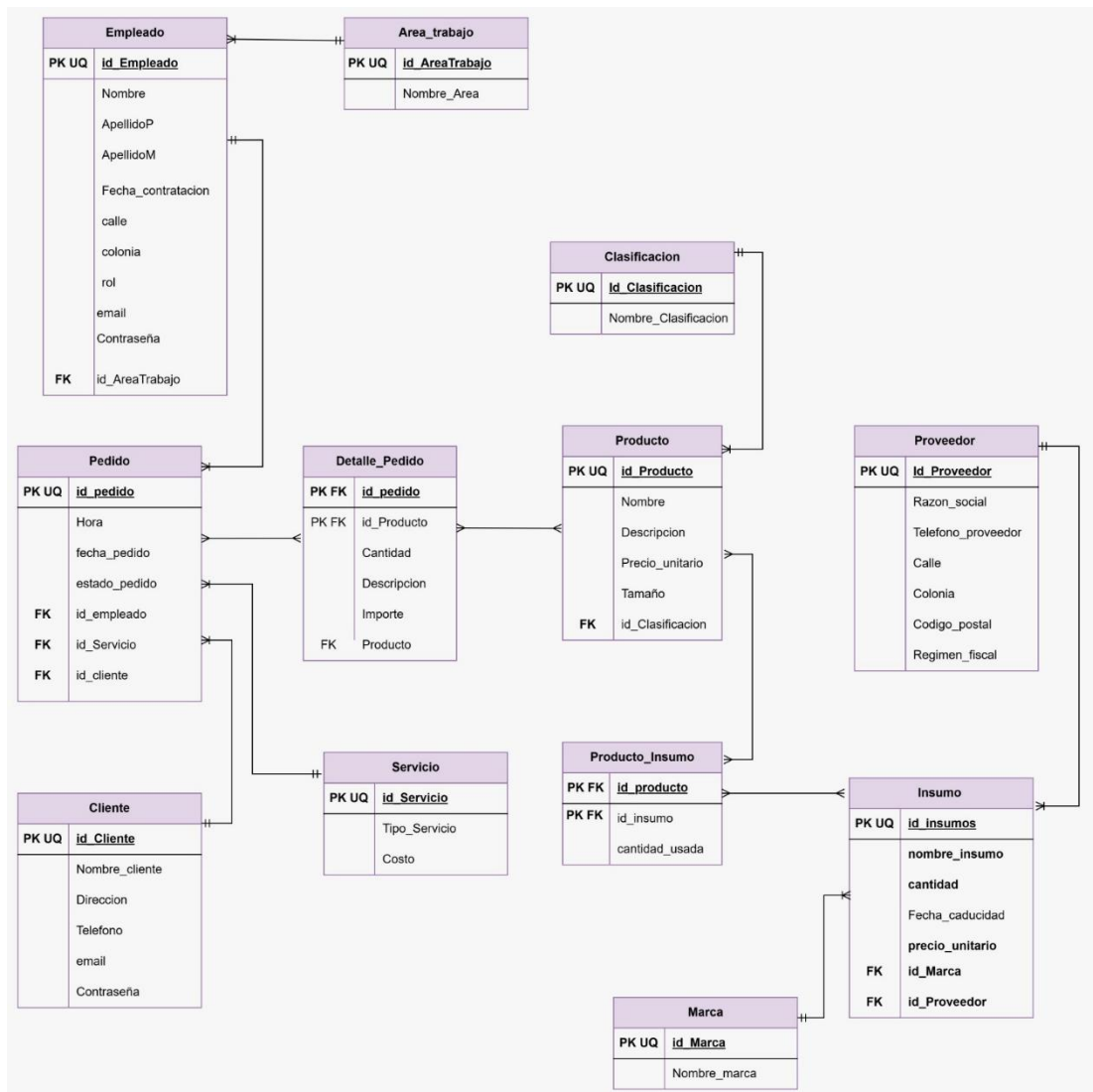
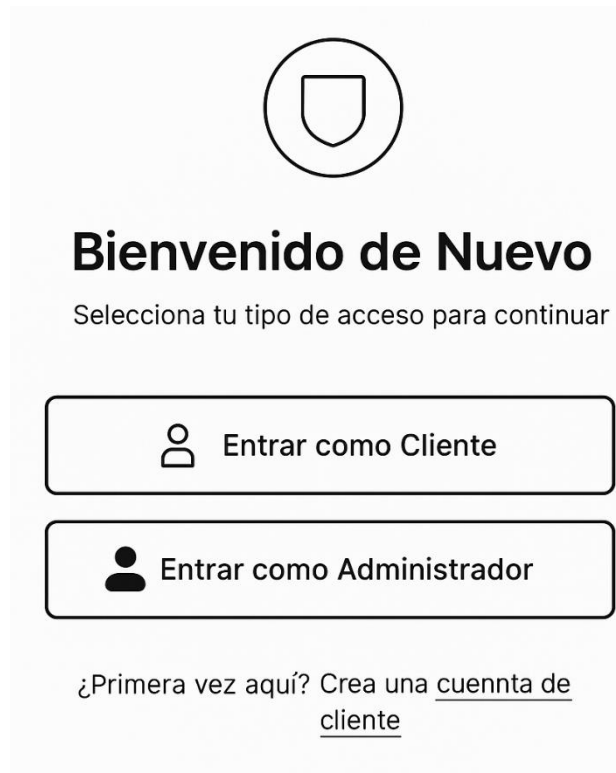


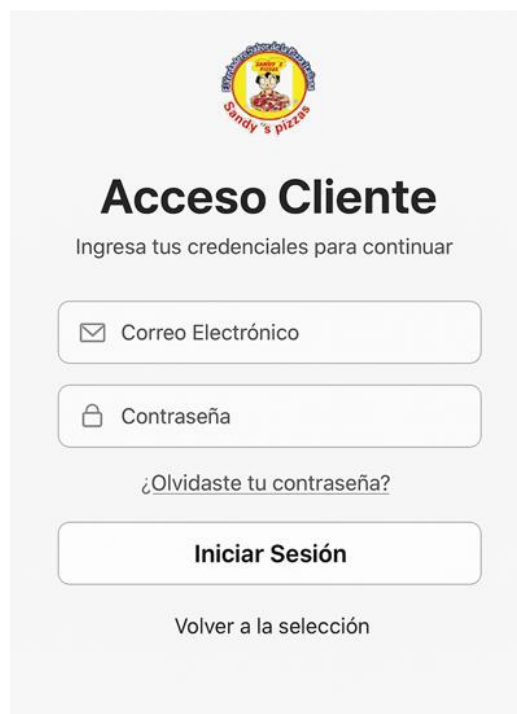
Figura 7. Modelo relacional. Fuente: Elaboración propia.

3.5 Wireframes




The wireframe shows a login page with a light gray background. At the top center is a circular logo containing a shield. Below the logo is the heading "Bienvenido de Nuevo" in a large, bold, black font. Underneath the heading is the instruction "Selecciona tu tipo de acceso para continuar" in a smaller, regular black font. There are two rounded rectangular buttons stacked vertically. The top button contains a person icon and the text "Entrar como Cliente". The bottom button contains a person icon and the text "Entrar como Administrador". At the bottom of the page, there is a link that says "¿Primera vez aquí? Crea una cuenta de cliente".


Figura 8. Wireframe de login para iniciar sesión Fuente: Figma



The wireframe shows a customer access page with a light gray background. At the top center is a circular logo for "Sandy's pizzas" featuring a cartoon character. Below the logo is the heading "Acceso Cliente" in a large, bold, black font. Underneath the heading is the instruction "Ingresa tus credenciales para continuar" in a smaller, regular black font. There are two input fields: the first is labeled "Correo Electrónico" with an envelope icon, and the second is labeled "Contraseña" with a lock icon. Below these fields is a link that says "¿Olvidaste tu contraseña?". At the bottom, there is a button labeled "Iniciar Sesión" and a link below it that says "Volver a la selección".

Figura 9. Acceso Cliente Fuente: Figma.



 **Panel de Administración**

Correo Electrónico

Contraseña

Ingresar

[¿Olvidaste tu contraseña?](#)
[Volver a la selección](#)

Figura 10. Acceso Administrador Fuente:Figma

**Recuperar
Contraseña**

Ingresar tu correo para continuar

Correo Electrónico

Verificar

[Volver a Iniciar Sesión](#)

Figura 11. Recuperar contraseña cliente-administrador. Fuente: Figma.

Nueva Contraseña

Establece una nueva contraseña para
0202manzano@gmail.com

Actualizar

[Volver a Iniciar Sesión](#)

Figura 12. Nueva Contraseña cliente-administrador Fuente: Figma.

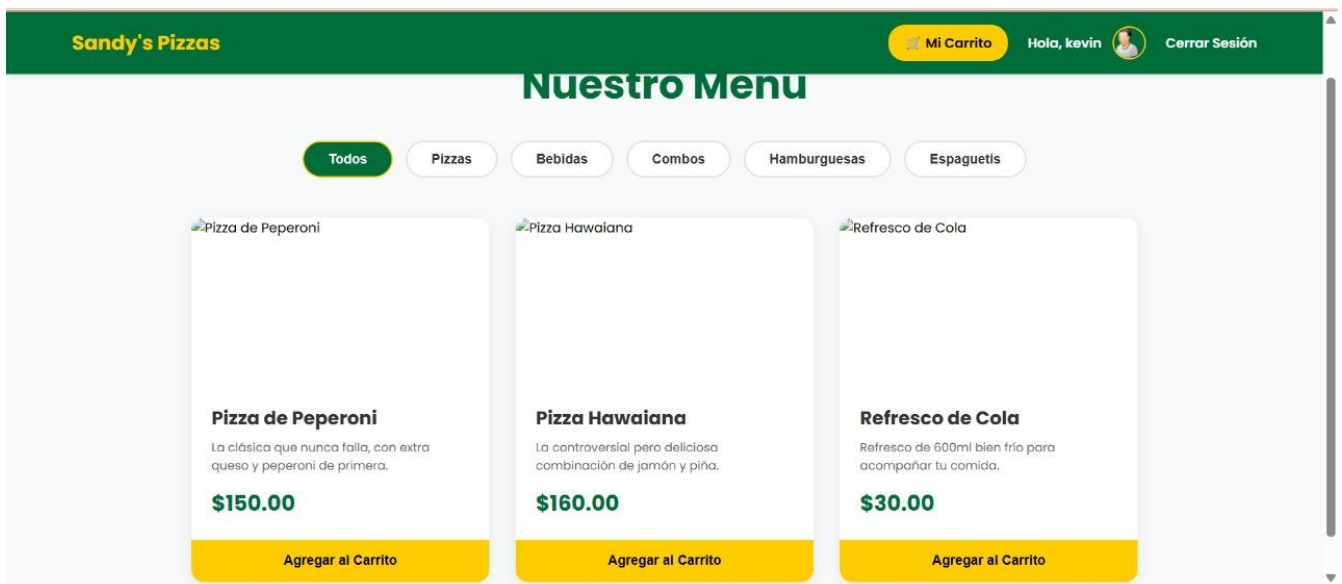


Figura 13. Pantalla de bienvenida cliente Fuente: Figma.

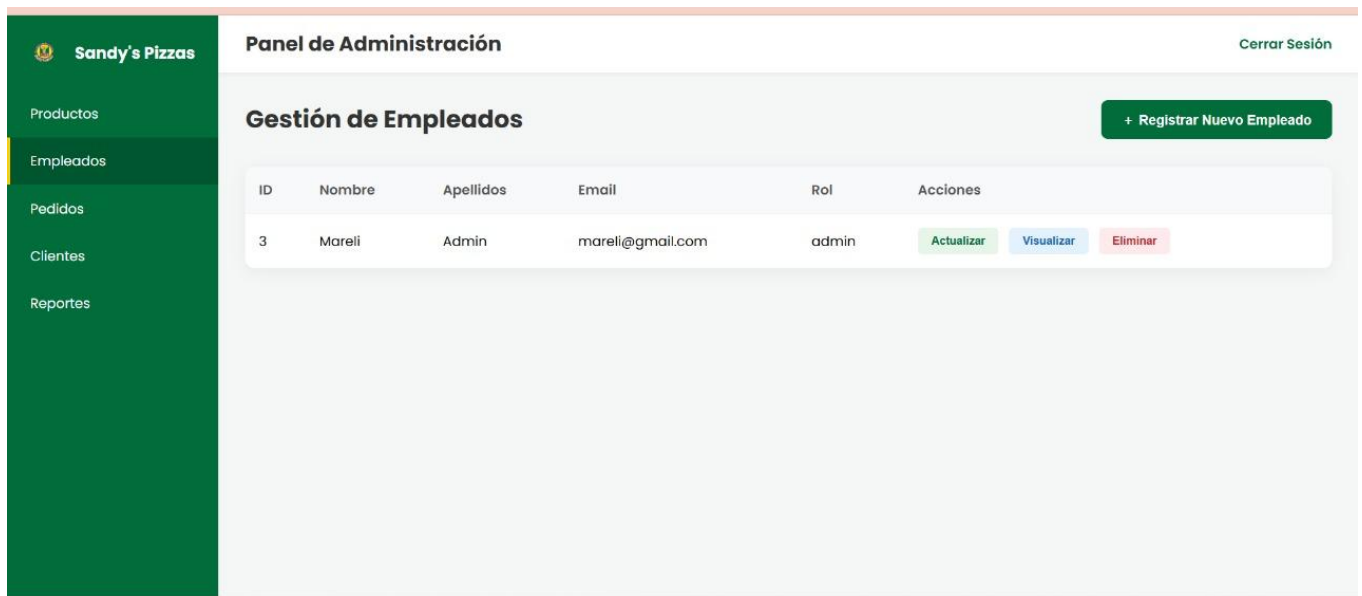


Figura 14. Crud de administrador del sistema. Fuente: Figma.

3.6 Diagrama de Clases

La Figura 16 muestra el diagrama entidad-relación del sistema, donde se representan las tablas principales como Usuario, Empleado, Cliente, Producto, Pedido, entre otras, junto con sus atributos y relaciones. Este modelo permite visualizar cómo se organiza la información, cómo se vinculan los datos entre entidades y cómo se gestionan aspectos como preferencias, insumos y servicios dentro de la base de datos.

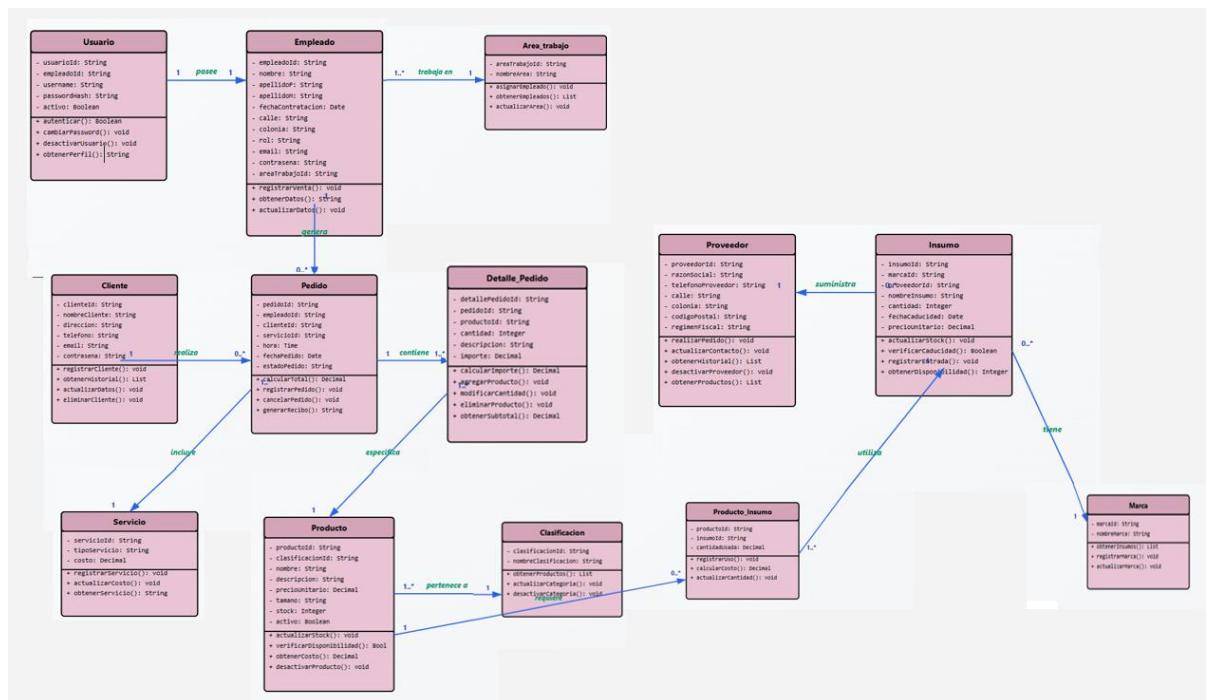


Figura 15. Diagrama de clases. Fuente: Elaboración propia.

3.7 Diagrama de secuencia

3.7.1 Diagrama de secuencia de HU-01: Login

De acuerdo con lo que se ilustra en la figura 17, el diagrama de secuencia del sistema de login muestra el flujo de interacción entre el usuario, la interfaz de inicio de sesión, el controlador y la base de datos. El proceso comienza cuando el usuario envía sus credenciales; estas son procesadas por el controlador, que consulta la base de datos para validar la información. Si la autenticación es exitosa, se genera la sesión y se asignan los permisos correspondientes; en caso contrario, se muestra un mensaje de error.

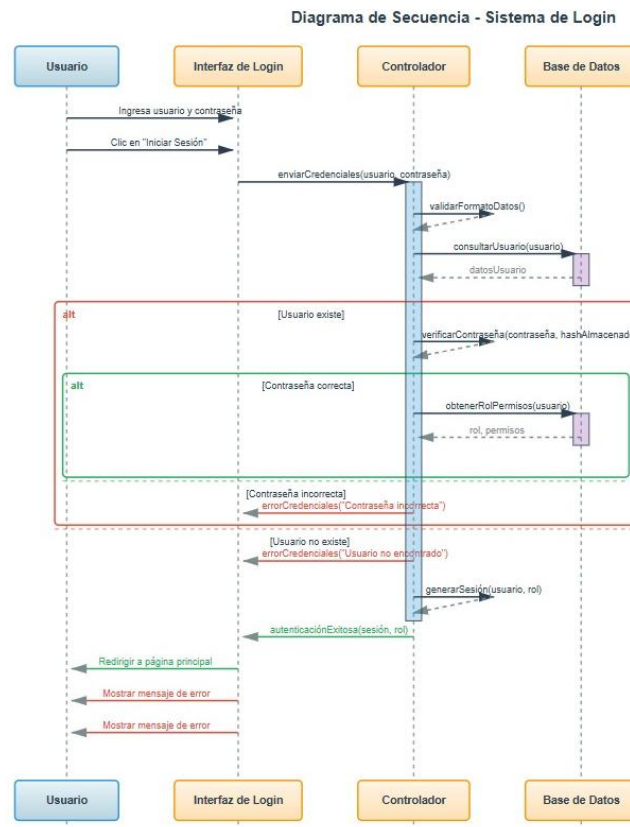


Figura 16 Diagrama de secuencia de HU-01: Login. Fuente: Elaboración propia.

3.7.2 Diagrama de secuencia de HU-02: Cliente

De acuerdo con lo que se ilustra en la Figura 18, el diagrama de secuencia representa el proceso de actualización de contraseña dentro del sistema. El usuario ingresa una nueva contraseña a través de la interfaz, la cual valida el formato antes de enviarla a la base de datos. Si la validación es exitosa, la contraseña se guarda y se muestra una confirmación al usuario; en caso contrario, se despliega un mensaje de error. Este flujo permite visualizar claramente las condiciones y respuestas del sistema ante una solicitud de cambio de contraseña.

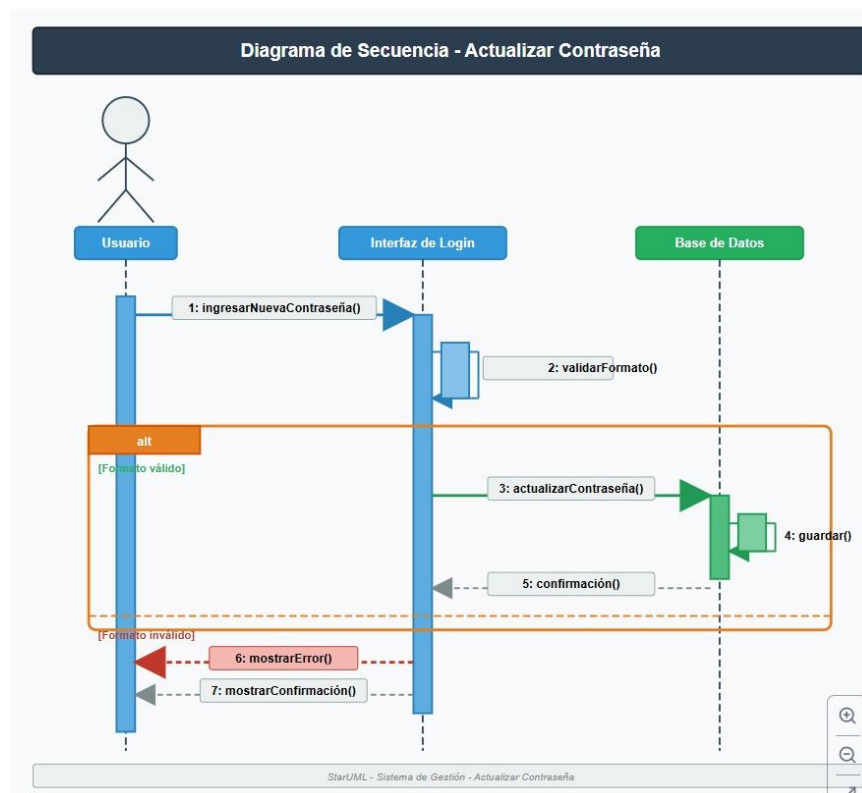


Figura 17. Diagrama de secuencia de HU-02: Cliente. Fuente: Elaboración propia

3.7.3 Diagrama de secuencia registrar empleado

De acuerdo con lo que se ilustra en la figura 19, el diagrama de secuencia describe el proceso de registro de un empleado en el sistema. El administrador solicita el registro, la interfaz muestra el formulario y recibe los datos, que luego son enviados a la base de datos para su validación. Si los datos son válidos y no hay duplicados, se inserta el registro y se confirma el alta; en caso contrario, se muestran errores y se solicita la corrección. Este flujo permite entender cómo se gestionan las validaciones y excepciones durante el registro de personal.



Figura 18. Diagrama de secuencia administrador (registrar empleado) Fuente: Elaboración propia.

3.7.4 Diagrama de secuencia visualizar empleado

De acuerdo con lo que se ilustra en la figura 20, el diagrama de secuencia describe el proceso de visualización de empleados por parte del administrador. La interfaz solicita los datos a la base de datos, los formatea y los presenta al usuario. El flujo contempla condiciones como la existencia de registros, fallos en la base de datos o errores de conexión, mostrando mensajes adecuados en cada caso. Este diagrama permite entender cómo se gestionan las consultas y el manejo de errores en la visualización de información.

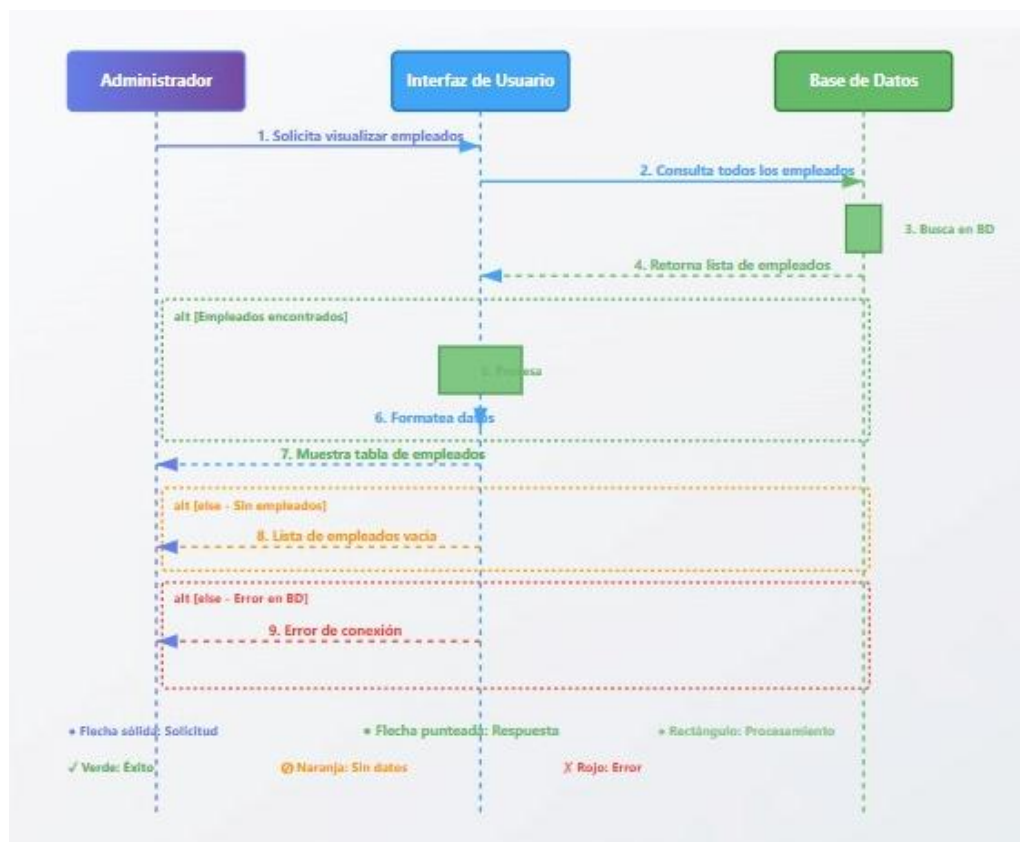


Figura 19. Diagrama de secuencia visualizar empleado. Fuente: Elaboración propia.

3.7.5 Diagrama de secuencia actualizar empleado

De acuerdo con lo que se ilustra en la figura 21, el diagrama de secuencia representa el proceso de visualización de empleados por parte del administrador. La interacción inicia con la solicitud desde la interfaz, que consulta la base de datos y recibe los registros encontrados. Luego, los datos se formatean y se muestran al usuario. El flujo contempla condiciones como la existencia de empleados, errores de conexión o fallos en la base de datos, mostrando mensajes adecuados según cada caso.

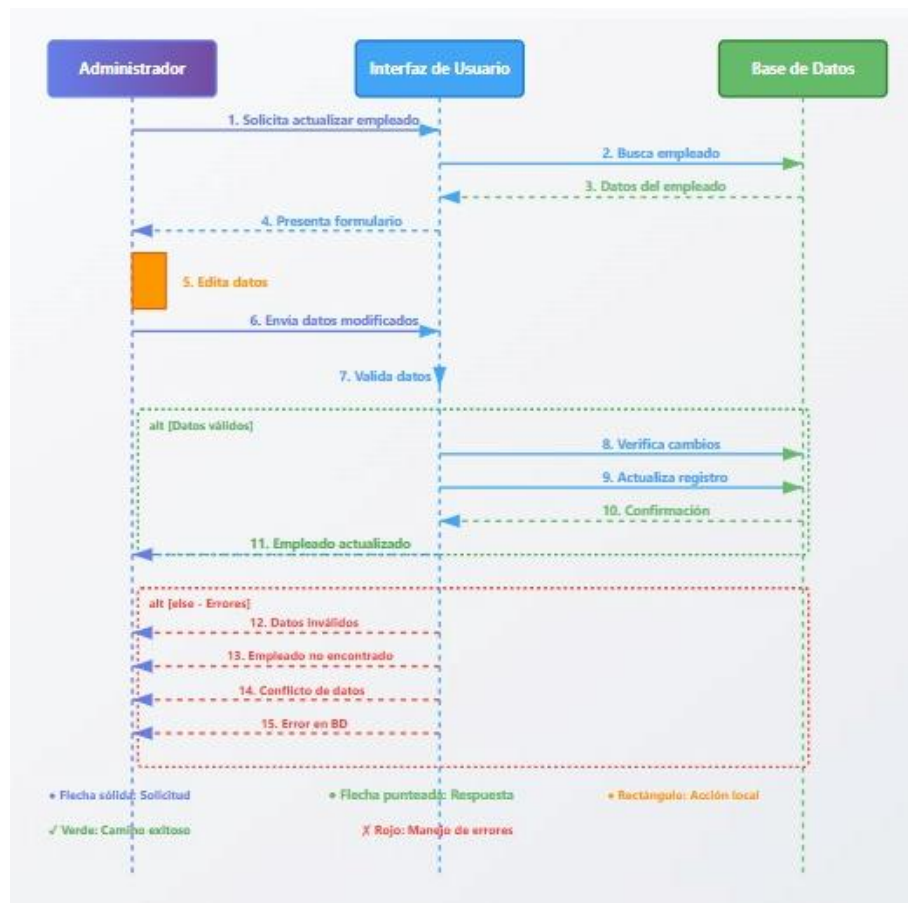


Figura 20. Diagrama de secuencia actualizar empleado Fuente: Elaboración propia.

3.7.6 Diagrama de secuencia eliminar empleado

De acuerdo con lo que se ilustra en la figura 22, el diagrama de secuencia describe el proceso de eliminación de un empleado en el sistema. El administrador inicia la solicitud, proporciona el ID del empleado y confirma la acción. La interfaz valida los datos y envía la orden a la base de datos, que elimina el registro y devuelve una confirmación.

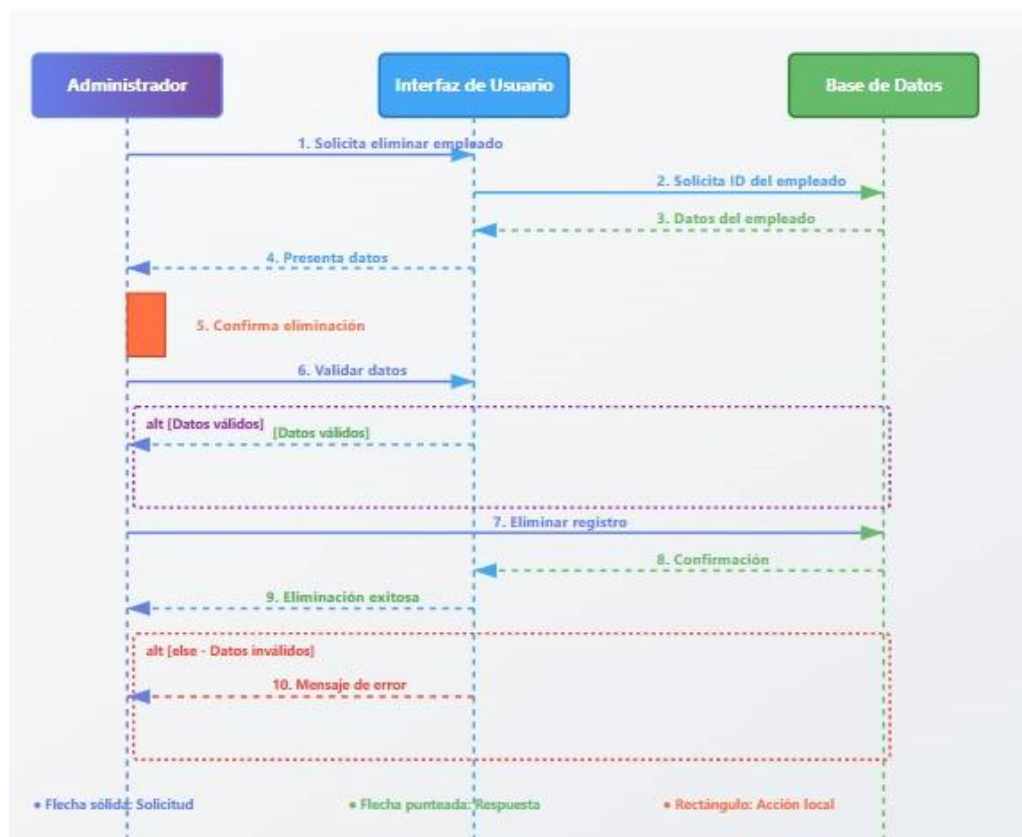


Figura 21. Diagrama de secuencia eliminar empleado Fuente: Elaboración propia.

3.8 Sprint backlog

Así como se muestra en la Figura 22, se presenta una tabla de planificación correspondiente al Sprint 1 del desarrollo del sistema, donde se detallan las tareas asociadas a cada historia de usuario. Cada fila especifica el identificador de la tarea, su descripción, el responsable asignado, el tiempo estimado de ejecución y el nivel de prioridad. Las actividades incluyen el diseño e implementación de interfaces de autenticación, validación de credenciales, gestión de contraseñas y actualización de módulos administrativos, todas con prioridad alta y asignadas al programador principal.

ID DEL SPRINT	HISTORIA DE USUARIO	ID TAREA	TAREA	RESPONSABLE	TIEMPO ESTIMADO	PRIORIDAD
Sprint 1	HU-01	T1	Diseñar la interfaz de inicio de sesión (usuario y contraseña).	Diseñador	2 días	ALTA
Sprint 1	HU-01	T2	Implementar la validación de credenciales contra la base de datos.	Programador	2 días	ALTA
Sprint 1	HU-01	T3	Conectar la interfaz con la base de datos para autenticar usuarios según su rol.	Programador	2 días	ALTA
Sprint 1	HU-02	T4	Diseñar la interfaz de cambio de contraseña (sin recuperación por correo).	Programador	3 días	ALTA
Sprint 1	HU-02	T5	Actualizar la nueva contraseña directamente en la base de datos.	Programador	1 día	ALTA
Sprint 1	HU-03	T6	Diseñar la interfaz de gestión de empleados (Visualizar, eliminar, actualizar, registrar).	Programador	2 días	ALTA

Figura 22. Sprint backlog. Fuente: Elaboración propia.