

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE INGENIERÍA

CARRERA DE: SISTEMAS DE INFORMACIÓN



Trabajo de Titulación

Tema: Desarrollo de una aplicación web para el control de desechos, aplicado a la empresa  
GERBI S.A.S

AUTOR:

JOSÉ ALFREDO PRADO HERRERA

QUITO DM, 10 DE MAYO 2024

## Tabla de contenido

<b>CAPÍTULO I</b> .....	4
<b>INTRODUCCIÓN</b> .....	4
<b>1 TEMA</b> .....	4
<b>2 JUSTIFICACIÓN</b> .....	4
<b>3 PLANTEAMIENTO DEL PROBLEMA</b> .....	4
<b>4 OBJETIVOS</b> .....	6
<b>4.1. GENERAL:</b> .....	6
<b>4.2 ESPECÍFICOS:</b> .....	6
<b>5 ALCANCE</b> .....	6
<b>CAPÍTULO II</b> .....	8
<b>MARCO TEÓRICO</b> .....	8
<b>1 APLICACIONES WEB</b> .....	8
<b>2 FRAMEWORK</b> .....	9
2.1 Framework Angular .....	9
<b>3 SOFTWARE DE DESARROLLO</b> .....	10
<b>4 VISUAL STUDIO CODE</b> .....	11
<b>5 LENGUAJE DE PROGRAMACIÓN</b> .....	12
<b>7 ARQUITECTURA DE LAS APLICACIONES WEB</b> .....	13
<b>8 POWER DESIGNER</b> .....	15
<b>9 BASES DE DATOS</b> .....	16
9.1 Normalizaciones en las bases de datos.....	17
9.2 Formas Normales .....	18
9.2.1 Primera Forma Normal (1NF).....	18
9.2.2 Segunda Forma Normal (2NF).....	18
9.2.3 Tercera Forma Normal (3NF) .....	18
9.2.4 Cuarta Forma Normal (4NF).....	18
9.2.5 Quinta Forma Normal (5NF).....	18
<b>10 MySQL</b> .....	19
10.1 Desventajas de MySQL.....	20
<b>CAPÍTULO III</b> .....	21
<b>ANÁLISIS Y LEVANTAMIENTO DE REQUERIMIENTOS</b> .....	21

<b>1 TOMA Y ANÁLISIS DE REQUERIMIENTOS CON LA METODOLOGÍA ÁNCORA</b>	<b>21</b>
<b>2 ELABORACIÓN DE REPORTE DE REQUERIMIENTOS</b>	<b>22</b>
<b>3 DOCUMENTO DE FACTIBILIDAD</b>	<b>24</b>
<b>CAPÍTULO IV</b>	<b>26</b>
<b>ANÁLISIS DEL SISTEMA</b>	<b>26</b>
<b>1 DESCRIPCIÓN DEL SISTEMA</b>	<b>26</b>
<b>2 FUNCIONALIDADES GENERALES DEL SISTEMA</b>	<b>26</b>
<b>3 FUNCIONALIDADES ENFOCADAS AL ADMINISTRADOR</b>	<b>27</b>
<b>4 FUNCIONALIDADES ENFOCADAS AL CLIENTE</b>	<b>27</b>
<b>CAPÍTULO V</b>	<b>28</b>
<b>DISEÑO DEL SISTEMA</b>	<b>28</b>
<b>1 MODELO CONCEPTUAL</b>	<b>28</b>
<b>2 MODELO FÍSICO</b>	<b>30</b>
<b>3 DIAGRAMAS DE CASOS DE USO</b>	<b>32</b>
F1 Administración de Usuario	34
<b>GLOSARIO</b>	<b>45</b>
<b>BIBLIOGRAFÍA</b>	<b>47</b>

# **CAPÍTULO I**

## **INTRODUCCIÓN**

### **1 TEMA**

Desarrollo de una aplicación web para el control de desechos, aplicado a la empresa GERBI S.A.S

### **2 JUSTIFICACIÓN**

Hoy en día un gran porcentaje de la población mundial tiene acceso a internet, esto significa que cualquier persona puede ingresar a una aplicación o página web para enterarse qué productos, servicios o noticias ofrece dicha página (Hootsuite, 2022). En la actualidad es muy importante que una empresa cuente con una aplicación web, debido al alcance mundial que tiene y aprovechar una ventaja competitiva respecto a otras empresas que no cuentan con una. Además de tener un espacio personalizado para ofrecer sus servicios a los clientes (Santiago Bedoya Monsalve, 2020).

GERBI S.A.S., es una empresa que está ubicada en la ciudad de Tulcán, desde el 2021 ha participado en la gestión y recuperación de una variedad de residuos, incluyendo orgánicos, inorgánicos, plásticos, aceites y materiales tecnológicos, empleando procesos logísticos y biotecnológicos. Además, se ha encargado de establecer comunicación con los usuarios a través de llamadas, visitas puerta a puerta, ferias y medios de comunicación. Esta empresa tiene todos sus registros de datos en hojas de Excel, por ello se le ha propuesto automatizar el proceso de control de desechos, donde se desarrollará una aplicación web que apoye a la gestión del reciclaje de la empresa GERBI S.A.S.

### **3 PLANTEAMIENTO DEL PROBLEMA**

Debido al rápido aumento de la población en la ciudad de Tulcán, la cantidad de residuos que se generan es inmensa, esto conlleva a la práctica de estrategias de gestión de desechos como: recolección, transporte, procesamiento y disposición final de residuos de todo tipo. En el Relleno Sanitario de la ciudad de Tulcán se almacenan aproximadamente 56,6 toneladas/día de residuos sólidos, de los cuales el 69% corresponde a residuos

orgánicos, el 21% a material reciclable y 10% se divide entre desechos bio-peligrosos y material que no puede ser aprovechado fuente (Méndez Bravo, 2015).

El problema de la contaminación en la ciudad de Tulcán surge de la acumulación y el inadecuado manejo de residuos. Las personas carecen de conciencia sobre el impacto ambiental negativo que esto provoca, debido a la falta de prácticas de reciclaje y de conocimiento sobre cómo gestionar adecuadamente los materiales reciclables. (Eugenia, 2014). GERBI S.A.S es una empresa que brinda un servicio privado de reciclaje para diversos tipos de desechos. Esta iniciativa contribuye a reducir la cantidad de residuos enviados a vertederos, alentando a los habitantes de Tulcán a clasificar sus desechos por categorías. Este enfoque resulta en una distribución más eficaz de los materiales reciclables, lo que a su vez facilita un procesamiento más rápido y de mayor calidad, permitiendo obtener materia prima de manera más eficiente.

Ante esta preocupante situación, y con el propósito de respaldar el servicio privado de gestión de residuos, se propone a GERBI S.A.S la creación de una aplicación web. Este software permitirá a los usuarios seleccionar la categoría y cantidad de residuos que desean reciclar, además de facilitar el registro de sus datos para llevar un control preciso del volumen reciclado diariamente. Así mismo, agilizará el proceso de reciclaje que la empresa proporciona.

## **4 OBJETIVOS**

### **4.1. GENERAL:**

1.4.1.1 Desarrollar una aplicación web para control de la información relacionada a desechos de la empresa GERBI S.A.S

### **4.2 ESPECÍFICOS:**

4.2.1 Realizar el levantamiento de la situación actual del control de desechos de la empresa GERBI S.A.S.

4.2.2 Identificar los requerimientos funcionales y no funcionales de la aplicación.

4.2.3 Realizar el diseño de la base de datos para el almacenamiento de la información y los demás elementos necesarios para el desarrollo de la aplicación.

4.2.4 Codificar la aplicación que cumpla el diseño realizado.

4.2.5 Realizar las pruebas necesarias para validar la categoría del residuo a reciclar, cantidad, datos del usuario y confirmación de la fecha cuando se realizará la entrega de los residuos.

## **5 ALCANCE**

Esta aplicación web tiene como finalidad ayudar a difundir las exigencias del proceso de reciclaje, y compartir los detalles de la manufacturación de los productos reciclados. Además, tendrá un apartado para fomentar e informar acerca de la reutilización de los desechos que se producen en el hogar, con esto se obtendrá que la ciudadanía tenga mayor conciencia de todos los residuos que se generan en el hogar.

El presente proyecto se considerará finalizado cuando la empresa GERBI S.A.S emita una carta de cumplimiento de todos los objetivos del proyecto, una vez probado y entregado el software, con su respectiva documentación a satisfacción de la entidad y que se cumplan con los siguientes módulos:

- Administración de inventario: Manejo de productos y stock.

- Administración de clientes: Control de clientes, se podrán realizar consultas a través del nombre del cliente.
- Administración de pedidos: Creación, modificación y visualización de pedidos.
- Pagina completa: La pagina web debe estar completa con todos los componentes que se acordaron en la toma de requerimientos.
- Roles de usuario:
  - Administrador: Acceso total al sistema, podrá acceder a los pedidos y a la administración de inventario, clientes, pedidos.
  - Empleado: Acceso a los pedidos e inventario.

## CAPÍTULO II MARCO TEÓRICO

### 1 APLICACIONES WEB

Las aplicaciones web, también conocidos como sistemas en línea, son desarrolladas y desplegadas en servidores en la nube o en una red local en lugar de ser instaladas en un sistema operativo específico, como Windows o Linux. Aunque comparten similitudes con las páginas web convencionales, estos sistemas poseen una funcionalidad potente diseñada para adaptarse a casos de uso particulares. (Service, 2023)

La facilidad de uso, un criterio esencial en el diseño de aplicaciones web, se traduce en interfaces de usuario intuitivas y una organización clara de la información. La funcionalidad, adaptada al propósito de la empresa, se convierte en un componente clave para garantizar que la aplicación cumpla con los objetivos a deseados. La velocidad de carga rápida y la adaptabilidad a diferentes dispositivos aseguran una experiencia de usuario consistente y positiva. (Arquitectura, 2020)

La seguridad, otro aspecto crítico, se aborda mediante medidas como la implementación de certificados SSL para proteger los datos de los usuarios. Al elegir desarrollar esta aplicación web se tomó consideraciones técnicas para garantizar la integridad y confidencialidad de la información transmitida entre el usuario y el servidor. (IONOS, 2020)

#### Características de una aplicación web

- Una aplicación web se caracteriza por tener un diseño atractivo ya que esto genera una mayor retención de usuarios y aumentar el tiempo de permanencia en la página.
- La velocidad, es una característica importante de una página web, debe cargar rápidamente para evitar que los usuarios se aburran y abandonen la página.
- La aplicación web debe ser accesible desde diferentes dispositivos como computadoras, tabletas y smartphones y debe adaptarse al tamaño de pantalla de cada uno de ellos.

- La aplicación web debe tener medidas de seguridad para proteger los datos de los usuarios, como, por ejemplo, un certificado SSL.

## 2 FRAMEWORK

El propósito fundamental de un framework es aliviar la carga de tareas del programador mediante la provisión de clases y funciones predefinidas. Este enfoque no solo promueve buenas prácticas de codificación, sino que también establece estándares coherentes para el proyecto. La eficiencia de un framework se ilustra en la rápida implementación de funcionalidades comunes, como el sistema de login en un proyecto web. (Munte, 2020)

Existen varios frameworks, se encuentran opciones destacadas como ReactJS, Angular, Vue.js, Svelte y Ember.js. Cada uno tiene sus propias características y ventajas. ReactJS, impulsado por Facebook, destaca por su enfoque en componentes reutilizables y gestión eficiente del estado. Angular, respaldado por Google, ofrece una estructura completa para aplicaciones web complejas, basándose en un enfoque basado en componentes. Vue.js se destaca por su simplicidad y facilidad de integración, mientras que Svelte, aunque clasificado como un compilador de componentes, ofrece eficiencia y velocidad en el desarrollo de aplicaciones web. (ADM, 2023)

Comparado con otros frameworks, Angular se destaca por su capacidad para crear diferentes tipos de aplicaciones web. Esta versatilidad asegura que el framework sea apto para una variedad de proyectos y requisitos, lo que contribuye a su adaptabilidad.

### 2.1 Framework Angular

Angular es un potente framework de JavaScript, especialmente adecuado para desarrollar aplicaciones frontend modernas de complejidad media o alta. Con Angular, se pueden crear aplicaciones del estilo de página única o aplicaciones web progresivas en JavaScript. (Gonçalves, 2021)

Angular se destaca por ofrecer una arquitectura completa que facilita el desarrollo de aplicaciones web. El uso de TypeScript como lenguaje predeterminado proporciona ventajas significativas, como la detección temprana de errores y un mantenimiento más sencillo del

código. El enlace de datos bidireccional simplifica la manipulación de datos y la actualización de la interfaz de usuario, mejorando la eficiencia del desarrollo. (A, 2023)

Sin embargo, Angular presenta desventajas, como una curva de aprendizaje inicial pronunciada, un mayor tamaño de descarga inicial y una complejidad excesiva para proyectos más simples. Su estructura completa puede parecer menos flexible en comparación con algunos frameworks más ligeros. Además, la obligatoriedad de TypeScript se la puede considerar como una limitación en la elección de lenguajes de programación. (Cruz, 2020)

Considerando todos estos aspectos, la elección de Angular como framework para el desarrollo frontend de esta aplicación web se justifica por su solidez técnica y versatilidad en la creación de aplicaciones. Además, las ventajas que nos da su arquitectura y el respaldo de la comunidad de desarrollo de software lo posicionan como un buen framework para este tipo de proyectos.

### **3 SOFTWARE DE DESARROLLO**

Un software de desarrollo se refiere a un conjunto de herramientas y programas informáticos diseñados para facilitar el proceso de creación, prueba y mantenimiento de software y aplicaciones. Estos programas proporcionan un entorno de trabajo donde los desarrolladores pueden escribir, depurar, compilar y administrar el código fuente.

Los softwares de desarrollo más conocidos presentan características diversas. En términos generales, herramientas como Visual Studio, Eclipse y NetBeans ofrecen versatilidad en el soporte de múltiples lenguajes, cada una con sus particularidades en términos de interfaz y rendimiento. Editores de texto como Atom y Sublime Text destacan por su flexibilidad, aunque carecen de algunas funcionalidades propias de entornos más completos. (Gwen, 2023)

Todas estas ventajas asociadas a los softwares de desarrollo nos proporcionan el entorno esencial para ser eficientes y productivos a la hora de crear aplicaciones web. Además, contribuyen significativamente en la calidad de las aplicaciones creadas, al

proporcionar ambientes integrados, herramientas especializadas y capacidades multiplataforma.

El uso de software de desarrollo, si bien es esencial para la creación de aplicaciones, también, presenta desventajas que deben ser consideradas, una de la más comunes, es la integración de sistemas, el cual puede plantear desafíos. La falta de interoperabilidad puede impactar negativamente en la eficiencia y el flujo de trabajo.

#### **4 VISUAL STUDIO CODE**

Visual Studio Code es un editor de código fuente creado por Microsoft, que está disponible de forma gratuita y es compatible con múltiples plataformas, incluyendo Windows, GNU/Linux y macOS. Este editor de código cuenta con una excelente integración con Git, así como con herramientas para depurar código. Además, dispone de numerosas extensiones que permiten a los usuarios escribir y ejecutar código en cualquier lenguaje de programación. (Flores, 2022)

La eficiente gestión de proyectos es otro factor técnico significativo. La capacidad de organizar archivos y carpetas de manera intuitiva contribuye a proyectos bien estructurados y promueve las mejores prácticas de desarrollo. Las potentes capacidades de depuración ofrecen una detección eficiente y rápida de errores, ahorrando tiempo al desarrollador y mejorando la calidad del código.

Visual Studio Code destaca por su versatilidad, integración sólida, extensibilidad y potentes capacidades de depuración. Estas características técnicas no solo respaldan su eficiencia, sino que también lo posicionan como una opción natural para un entorno de desarrollo adaptable y potente, capaz de satisfacer las demandas de una variedad de lenguajes y tecnologías. La preferencia por Visual Studio Code se fundamenta en su capacidad para mejorar la productividad y ofrecer una experiencia de desarrollo flexible y eficaz. (IT, 2021)

Visual Studio Code ofrece ventajas significativas en el desarrollo de software. Su amplia biblioteca de extensiones permite una personalización modular y adaptación precisa

a las necesidades de proyectos. Además, es compatible con una amplia gama de lenguajes de programación, donde cada extensión está diseñada para brindar soporte y herramientas específicas para cada lenguaje en particular. Dichas extensiones, son actualizadas regularmente para mantenerse al día con las últimas tendencias y tecnologías, esto asegura una mejora en la calidad del código que se escribe.

## **5 LENGUAJE DE PROGRAMACIÓN**

En el ámbito de la informática, se hace referencia al término lenguaje de programación para describir un tipo de software cuya función es permitir la creación de otros programas informáticos. El lenguaje de programación se caracteriza por ser un lenguaje formal diseñado específicamente para estructurar algoritmos y procesos lógicos, que serán posteriormente ejecutados por un sistema informático, y así controlar su comportamiento físico, lógico y su interacción con el usuario. (Mendoza, 2020)

El lenguaje de programación se compone de símbolos, reglas sintácticas y semánticas que se organizan en forma de instrucciones y relaciones lógicas para crear el código fuente de una aplicación o software. Los lenguajes de alto nivel como Golang y JavaScript simplifican la codificación y mejoran la productividad. La diversidad de lenguajes permite elegir según los necesitemos. La comunidad activa en torno a lenguajes como Golang y Python facilita el soporte y la resolución de problemas. (Etece, 2021)

La amplia variedad de opciones puede generar indecisión en la elección del lenguaje más adecuado. Lenguajes de bajo nivel, aunque potentes, pueden ser propensos a errores y requerir más esfuerzo de codificación. La rapidez con la que ciertos lenguajes evolucionan puede resultar en desafíos de mantenimiento. Lenguajes más complejos, como C++, pueden tener curvas de aprendizaje pronunciadas, especialmente para personas que no están familiarizadas con esta herramienta.

## **6 GOLANG**

Go es un lenguaje de programación concurrente y compilado, que, como C y C++, también admite canales de comunicación basados en el lenguaje CSP. A diferencia de los criterios de programación basados en bloqueos, Go utiliza un enfoque de concurrencia único. Los creadores de Go tomaron inspiración de la versatilidad y las cualidades de otros

lenguajes como Python, C++ y Java, entre otros, para lograr un lenguaje con características compartidas con otros lenguajes compilados, así como algunas únicas. (EDteam, 2019)

En términos de rendimiento y eficiencia de ejecución, Go se compara favorablemente con lenguajes compilados como C y C++. Su recolección de basura está optimizada para minimizar interrupciones, asegurando tiempos de ejecución predecibles, lo que es fundamental para este tipo de aplicaciones web. Una característica distintiva de Go es su modelo de concurrencia incorporado, basado en goroutines y canales. Esto simplifica la creación de programas concurrentes, permitiendo a los desarrolladores aprovechar eficientemente sistemas multiproceso y multicore. (Workana, 2021)

La gestión eficiente de dependencias es otra fortaleza de Go. Su sistema de vendoring permite controlar explícitamente las versiones de las dependencias, mitigando problemas comunes asociados con la gestión de dependencias en otros lenguajes. El ecosistema integrado de Go, que incluye herramientas de prueba, perfilado y documentación, contribuye a la eficiencia en la creación y mantenimiento de proyectos web. (Mytaskpanel, 2022)

El enfoque de Go en la simplicidad y minimalismo, eliminando características innecesarias, facilita el aprendizaje y reduce la complejidad del código. La elección de Go sobre otros lenguajes se fundamenta en estas características, para el desarrollo de esta aplicación web.

## **7 ARQUITECTURA DE LAS APLICACIONES WEB**

La arquitectura de aplicaciones web se refiere a la estructura organizativa y el diseño de aplicaciones destinadas a funcionar a través de la web. Se compone de tres capas principales: la capa de presentación o frontend, que maneja la interfaz de usuario y la interacción directa con los usuarios; la capa lógica, de negocio o backend, que gestiona la lógica central de la aplicación y procesa las solicitudes del cliente; y la capa de datos que también es parte del backend, encargada del almacenamiento y recuperación de datos. En esta arquitectura, se emplean tecnologías como HTML, CSS y diversos frameworks para el frontend, mientras que en el backend se utilizan lenguajes de programación y frameworks específicos, junto con bases de datos para gestionar la información. La arquitectura moderna

también incorpora conceptos como microservicios y contenedores para lograr mayor flexibilidad y escalabilidad. (Harsh, 2022)

La capa de presentación se encarga de toda la parte visible y la interacción con los usuarios. Las tecnologías web modernas, como React, Angular o Vue.js, facilitan el desarrollo de interfaces interactivas y dinámicas. El diseño responsivo se ha vuelto esencial para garantizar una experiencia consistente en diferentes dispositivos.

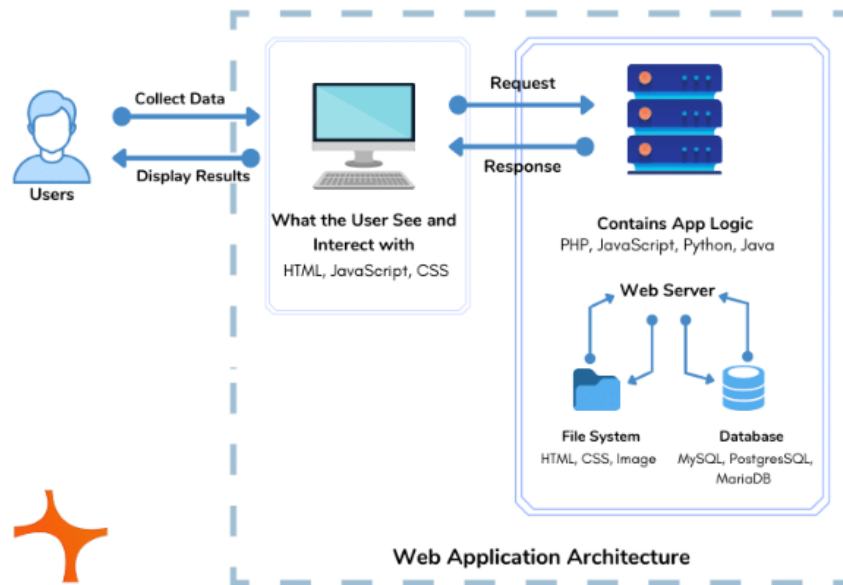
La capa lógica o de negocio contiene la lógica central de la aplicación. Aquí, los frameworks backend, como Django, Flask, Express, o Spring, son comunes. Estos frameworks permiten gestionar la lógica de la aplicación, manejar las solicitudes del cliente y gestionar la comunicación con la capa de datos. (hat, 2023)

La capa de datos aborda el almacenamiento y la recuperación de datos. Las bases de datos relacionales, como MySQL, son comunes para estructurar datos, mientras que las bases de datos NoSQL, como MongoDB, se utilizan para aplicaciones que requieren flexibilidad en el esquema de datos. Las API RESTful facilitan la comunicación entre la capa de lógica y la capa de presentación.

La seguridad es un aspecto crítico en la arquitectura de aplicaciones web. La implementación de prácticas de seguridad, como el uso de HTTPS, la validación de entrada, y la gestión segura de credenciales, es esencial. Los sistemas de autenticación, como OAuth, juegan un papel vital para proteger los datos del usuario.

**Figura 1**

*Arquitectura de las aplicaciones web*



Arquitectura de aplicaciones web

*Nota.* Imagen que representa los componentes de la arquitectura de aplicaciones web, Pooja Sharma, 2021 (<https://cynoteck.com/es/blog-post/reasons-to-treat-web-application-architecture-seriously/>),

## 8 POWER DESIGNER

Power Designer, desarrollado por SAP, es una herramienta de modelado y diseño que abarca una amplia gama de funciones en el ámbito de la ingeniería de software. Esta herramienta es reconocida por su capacidad para crear modelos completos de bases de datos, facilitando la visualización de la arquitectura de sistemas complejos. A través de sus funciones, como el modelado de entidad-relación y de clases, Power Designer permite diseñar de manera integral la estructura de sistemas de información. (Designer, 2020)

Una de las ventajas clave de Power Designer es su integración con diversos sistemas de gestión de bases de datos y entornos de desarrollo. Esto proporciona flexibilidad para trabajar con diferentes tecnologías y facilita la adaptación, en este caso en un proyecto de desarrollo web. La generación automática de código SQL y otros lenguajes de programación a partir de los modelos diseñados es otra característica destacada. Este proceso acelera la implementación y reduce posibles errores asociados con la codificación manual, contribuyendo a un desarrollo más eficiente. (Guaman, 2012)

Power Designer no solo se centra en el diseño, sino que también facilita la documentación clara y detallada de los modelos. Además, ofrece capacidades de análisis de impacto, permitiendo evaluar cómo los cambios propuestos afectarán a otras partes del sistema, lo que contribuye a una planificación más precisa. Sin embargo, PowerDesigner presenta desafíos. Su curva de aprendizaje puede ser pronunciada, ya que ofrece una amplia gama de características avanzadas. Además, el costo asociado con las licencias puede limitar su accesibilidad, pero para proyectos pequeños funciona bien.

## **9 BASES DE DATOS**

Las bases de datos son elementos fundamentales en la gestión y organización de la información en entornos digitales. Su función principal es almacenar, gestionar y proporcionar acceso eficiente a datos estructurados, permitiendo manejar grandes volúmenes de información de manera sistemática. Un aspecto clave de las bases de datos es su capacidad para garantizar la integridad y consistencia de los datos. A través de relaciones y restricciones, se establecen reglas que aseguran que la información almacenada cumpla con ciertos estándares, evitando inconsistencias y errores. (Etece E. , 2021)

La evolución de las bases de datos ha llevado a la aparición de sistemas de gestión de bases de datos (SGBD) que ofrecen interfaces y herramientas para administrar de manera eficiente la creación, modificación y consulta de datos. SGBD populares como MySQL, Oracle, y Microsoft SQL Server han contribuido significativamente al desarrollo y escalabilidad de sistemas de información. (Oracle, 2023)

La flexibilidad de las bases de datos se refleja en su capacidad para adaptarse a diferentes modelos de datos, como el modelo relacional, el modelo de documentos y el

modelo de grafos. Cada modelo tiene sus propias ventajas y desventajas, lo que permite a elegir la estructura que mejor se adapte. No obstante, el diseño y mantenimiento efectivo de bases de datos requiere consideraciones cuidadosas, como la normalización para evitar redundancias y la optimización de consultas para mejorar el rendimiento. Además, la seguridad y privacidad de los datos son aspectos críticos, especialmente en un entorno donde la ciberseguridad es una preocupación creciente.

## 9.1 Normalizaciones en las bases de datos

La normalización de bases de datos persigue diversos objetivos clave para optimizar la eficiencia y la integridad del sistema. En primer lugar, busca reducir la redundancia de datos, eliminando la repetición innecesaria de información y optimizando así el espacio de almacenamiento. Al mismo tiempo, la normalización tiene como objetivo principal mejorar la integridad de los datos, eliminando dependencias no deseadas entre tablas y minimizando posibles anomalías, como la pérdida de información o inconsistencia de datos. (Diaz, 2017)

Otro objetivo es facilitar las operaciones de mantenimiento, ya que la estructura organizada resultante simplifica las actualizaciones y gestiones de datos. Al tener tablas más pequeñas y específicas, se logra una mayor eficiencia en las operaciones cotidianas de la base de datos. Asimismo, la normalización persigue optimizar operaciones de consulta, aunque puede introducir complejidades adicionales. El objetivo es encontrar un equilibrio adecuado entre la estructura normalizada y la eficiencia en la recuperación de información.

La normalización también promueve la flexibilidad y escalabilidad del sistema, permitiendo que la base de datos evolucione y se adapte a cambios en los requisitos o a un aumento en el volumen de datos. Evitar anomalías en operaciones críticas como la actualización, inserción y eliminación de datos es otro objetivo crucial. Al tener datos organizados y relacionados adecuadamente, se reducen los riesgos de errores y problemas en estas operaciones esenciales. (Torrejón, 2022)

## 9.2 Formas Normales

Las tres primeras formas normales son suficientes para cubrir las necesidades de la mayoría de las bases de datos.

### 9.2.1 Primera Forma Normal (1NF)

Una tabla está en 1NF si todos sus atributos contienen valores atómicos, es decir, valores indivisibles. No debe haber conjuntos de valores ni atributos multivaluados.

### 9.2.2 Segunda Forma Normal (2NF)

Una tabla está en 2NF si cumple con la 1NF y todos sus atributos no clave están completamente dependientes de la clave primaria. En otras palabras, no debe haber dependencias parciales de la clave primaria.

### 9.2.3 Tercera Forma Normal (3NF)

Una tabla está en 3NF si cumple con la 2NF y no hay dependencias transitivas entre los atributos no clave y la clave primaria. Esto significa que no debe haber atributos no clave que dependan de otros atributos no clave.

### 9.2.4 Cuarta Forma Normal (4NF)

Una tabla está en 4NF si cumple con la 3NF y no tiene dependencias multivaluadas. En otras palabras, no debe haber conjuntos de atributos que dependan de la clave primaria y que, a su vez, tengan múltiples valores para otra clave no relacionada.

### 9.2.5 Quinta Forma Normal (5NF)

Una tabla está en 5NF si cumple con la 4NF y elimina la redundancia derivada de las dependencias de unión. Esto significa que no debe haber redundancia basada en la unión de otras tablas.

## 10 MySQL

MySQL destaca como una opción sólida de base de datos, respaldada por varios motivos técnicos. Su estatus como sistema de gestión de bases de datos de código abierto y la presencia de una comunidad activa de desarrolladores son factores clave. La accesibilidad y personalización inherentes al código abierto, junto con el constante apoyo de la comunidad, contribuyen a mejoras continuas y ofrecen recursos valiosos. (Robledano, 2019)

En términos de rendimiento, MySQL ha sido optimizado para proporcionar accesos rápidos a datos, especialmente en operaciones de lectura. Su capacidad de almacenamiento en memoria caché y técnicas de indexación lo convierten en una elección eficiente para entornos donde la velocidad es esencial. Además, la base de datos destaca por su capacidad de escalabilidad, lo que la hace apta para manejar grandes volúmenes de datos y adaptarse a crecimientos futuros en la carga de trabajo.

La compatibilidad y versatilidad de MySQL también son destacables. Compatible con varios sistemas operativos y lenguajes de programación, ofrece soporte para procedimientos almacenados, disparadores y funciones, permitiendo la implementación de lógica de negocio directamente en la base de datos. Además, sus herramientas y utilidades, como MySQL Workbench, facilitan el desarrollo, la administración y el monitoreo de bases de datos. (Techtarget, 2021)

En términos de seguridad, MySQL ofrece funciones robustas, incluyendo la gestión de permisos a nivel de usuario y capacidades de encriptación. La garantía de cumplir con propiedades ACID en transacciones respalda la integridad de los datos y la confiabilidad de las operaciones en entornos críticos. Además, MySQL es conocido por su costo atractivo al ser de código abierto y generalmente distribuido bajo la licencia GNU, lo que lo convierte en una opción rentable en comparación con soluciones comerciales.

## 10.1 Desventajas de MySQL

- Aunque MySQL es escalable, puede no ser tan eficiente en la optimización para grandes conjuntos de datos en comparación con sistemas diseñados específicamente para manejar grandes volúmenes de información. (Aldeahost, 2020)
- En comparación con otros sistemas de gestión de bases de datos, MySQL puede enfrentar desafíos en operaciones de escritura intensiva. La complejidad de estas operaciones podría resultar en un rendimiento subóptimo en entornos donde la escritura de datos es frecuente y sustancial.

## CAPÍTULO III ANÁLISIS Y LEVANTAMIENTO DE REQUERIMIENTOS

### 1 TOMA Y ANÁLISIS DE REQUERIMIENTOS CON LA METODOLOGÍA ÁNCORA

En este capítulo, se empleará la Metodología Áncora juntamente con sus herramientas. Es importante destacar que en el desarrollo de este sistema se integrarán diversas características de la metodología XP en conjunto con Áncora.

En esta etapa inicial, se llevará a cabo la recopilación, análisis y diseño de acuerdo con los requisitos proporcionados por el cliente. Para llevar a cabo esta actividad, se seguirá meticulosamente cada una de las características de Áncora, incluyendo el uso de guiones, diálogos, pistas, escenas y utensilios.

El método seleccionado para la recolección de requisitos consistió en una reunión a través de la plataforma de reuniones ZOOM con el cliente. Esta reunión proporcionó información valiosa para la evaluación de la viabilidad del sistema propuesto. La información recopilada abarca los antecedentes actuales sobre la forma en que la empresa y el cliente ejecutan los procesos que serán automatizados. Asimismo, ofreció detalles sobre las necesidades del cliente y cómo estas pueden ser satisfechas a través del sistema que se va a desarrollar.

Siguiendo los pasos delineados por la metodología Áncora y su ciclo de vida, se iniciará definiendo los roles que se describen en la tabla siguiente:

**Tabla 1**

*Roles que contendrá la aplicación web*

Roles	Actividades
Analista de requerimientos	Recolección de requerimientos, tiene comunicación con el cliente.
Usuario (Cliente)	Provee información sobre la situación actual de la empresa al analista de requerimientos

Una vez que se han establecido los Roles y las Actividades asociadas a cada uno, es importante especificar los productos que resultan del análisis de requerimientos, los cuales incluyen guiones y diálogos. Siguiendo el ciclo de vida de Áncora, se procederá a organizar y clasificar de manera jerárquica cada uno de los requisitos, además de validarlos. Por último, se determinarán qué funcionalidades tienen el potencial de ser reutilizadas en el futuro.

En la siguiente tabla se detalla cada uno de los artefactos que se obtendrán al aplicar el ciclo de vida de Ancora.

**Tabla 2**

*Pasos para realizar la toma y análisis de los requerimientos*

Fase	Artefacto
Entendimiento del dominio y contexto de la aplicación.	Contexto y situación actual
Recolección y clasificación de requerimientos.	Propuesta computacional
Resolución de conflictos y validación de requerimientos.	Propuesta computacional validada, observaciones y acuerdos

## 2 ELABORACIÓN DE REPORTE DE REQUERIMIENTOS

Para la creación del informe de requisitos, se emplearán guiones. A continuación, se presenta el guion que describe la situación actual de la empresa.

**Tabla 3**

*Reporte de requerimientos*

Guion:	Escena: Gestión de información en hojas de Excel
Papeles:	
Encargado de registro de datos Encargado de supervisión Gerente	El Encargado de Supervisión verifica la consistencia y exactitud de los datos. Si se identifican errores, se notifica al Encargado de Registro de Datos para su corrección.
Utensilios	Escena: Registro de información de materiales reciclables
Servicio de reciclaje	El Encargado de Registro de Datos ingresa la cantidad y tipo de materiales reciclables en la hoja de Excel designada. El Encargado de Supervisión revisa y verifica la precisión de los datos ingresados.
Condiciones de entrada	Escena: Actualización del Inventario y Reporte al Gerente
El encargado de registro de datos recopila la información de los materiales reciclables recibidos. El gerente proporciona orientación sobre los tipos de materiales a registrar.	El Encargado de Registro de Datos actualiza la cantidad y tipo de materiales reciclables en la hoja de Excel designada El Gerente utiliza la información para realizar un seguimiento del inventario y tomar decisiones sobre la gestión de los materiales reciclables.
Condiciones de salida	
La hoja de Excel se actualiza con la información precisa sobre los materiales reciclables recibidos	

Al revisar el guion y tras llevar a cabo la entrevista con el cliente, se genera el siguiente informe que enumera las posibles capacidades que debe poseer el sistema a ser creado, las cuales se describen a continuación:

1. Ingreso de clientes deberá cumplir con lo siguiente

- Datos personales
- Contraseña

2. Registro de materiales a reciclar

- Tipo de material
- Cantidad
- Fecha

3. Elaboración de orden de servicio de reciclaje

4. Manejo de inventario

- Cliente
- Categoría del material reciclado
- Material reciclado

### 3 DOCUMENTO DE FACTIBILIDAD

Siguiendo las pautas establecidas por el ciclo de vida de Áncora y la Metodología XP, así como considerando el análisis de requisitos previamente mencionado, en esta sección, se procederá a organizar los requisitos agrupándolos en posibles capacidades del sistema. Estas se detallan en la tabla a continuación:

**Tabla 4**

*Capacidades del sistema*

Funcionalidad	Actividades
Gestión de usuarios	CRUD de Clientes
Gestión de materiales reciclados	CRUD de Productos reciclados
Gestión de materiales que se recicla	Estado de stock de productos.
Control de materiales reciclados	Reporte sobre los materiales que más se recicla

Se deben establecer las prioridades para las funcionalidades mencionadas anteriormente, utilizando dos categorías: Prioridad Alta y Prioridad Baja. La determinación de estas prioridades se basará en factores como la importancia de cada función en el sistema,

la complejidad para su codificación y su relevancia para el cliente. La tabla siguiente detalla las funcionalidades junto con su respectiva prioridad. Las funcionalidades principales serán etiquetadas con Prioridad Alta, mientras que las funciones menos críticas, como los informes de datos estadísticos, recibirán la etiqueta de Prioridad Baja.

**Tabla 5**

*Prioridades de las funcionalidades de la aplicación web*

Funcionalidad	Prioridad
Gestión de usuarios	ALTA
Gestión de materiales reciclados	ALTA
Gestión de materiales que se recicla	ALTA
Control de materiales que se recicla	BAJA

## **CAPÍTULO IV**

### **ANÁLISIS DEL SISTEMA**

#### **1 DESCRIPCIÓN DEL SISTEMA**

Basándonos en los requisitos de los usuarios y las capacidades que debe tener este sistema, se han definido dos perfiles con distintos niveles de acceso. Esto se determina según la función que cada usuario desempeña dentro de la empresa GERBI S.A.S

**Administrador:** El usuario con roles de administrador en la página web de reciclaje es responsable de llevar a cabo las principales funciones del sistema. Esto incluye la gestión de categorías, unidades, productos, clientes, así como la supervisión de la entrada y salida de materiales que se reciclan. Por lo tanto, este usuario cuenta con todos los permisos para acceder y modificar los datos.

**Cliente:** Los usuarios con perfiles de cliente en la página web de reciclaje tienen la responsabilidad de administrar sus entregas de materiales a reciclar. Tienen la capacidad de ver el estado del servicio prestado por la empresa. Además, poseen ciertos permisos para actualizar su información como es la contraseña u otro dato relevante que se quiera actualizar, pero no cuentan con la autorización para eliminarlos. Estos usuarios están vinculados como colaboradores de la empresa.

#### **2 FUNCIONALIDADES GENERALES DEL SISTEMA**

Una vez finalizado el desarrollo del sistema actual, la empresa GERBI S.A.S dispondrá de una plataforma web, accesible desde cualquier dispositivo con conexión a Internet y equipado con el software necesario para la navegación. Esto se logra a través de su asociación con un dominio. El enfoque principal de esta aplicación es automatizar los procesos de la empresa, agilizando las operaciones y asegurando la integridad de los datos. Asimismo, se espera una mejora en la eficiencia en la gestión de las órdenes de servicio de la empresa y los materiales que recicla la empresa, actualmente gestionados manualmente.

La plataforma web permitirá el registro de usuarios y de materiales a reciclar mediante el servicio de reciclaje que ofrece la empresa. Del mismo modo, se podrá realizar un seguimiento detallado de cada entrega de materiales que realice un usuario proporcionando cierta información como la cantidad o categoría de material reciclable. Esto contribuirá con datos valiosos para la toma de decisiones empresariales.

### **3 FUNCIONALIDADES ENFOCADAS AL ADMINISTRADOR**

El administrador tendrá acceso a un panel completo a través del sistema, donde se mostrarán todas las funciones principales. En la barra de navegación se ubicarán botones que facilitarán la interacción hacia diferentes funciones, incluyendo las siguientes:

- Ingreso al sistema
- Gestión de usuario
- Gestión del material reciclado: Ingresar, actualizar, consultar y eliminar
- Gestión del material que recicla la empresa: Ingresar, actualizar, consular y eliminar.

### **4 FUNCIONALIDADES ENFOCADAS AL CLIENTE**

El usuario, como miembro colaborador de la empresa, contará con un perfil que le confiere ciertos privilegios para llevar a cabo ciertas funciones. Entre las capacidades dirigidas al operador se incluyen:

- Registro en el sistema
- Ingreso al sistema
- Actualizar contraseña
- Orden de reciclaje: Consultar, ingresar
- Recuperar contraseña

## **CAPÍTULO V**

### **DISEÑO DEL SISTEMA**

En este capítulo, se llevará a cabo la creación del modelo conceptual y físico del sistema. Esto implica la construcción del diseño de la base de datos del sistema, incluyendo todas las entidades necesarias y sus relaciones correspondientes. Para determinar las entidades correctas, nos basamos en la información recopilada en capítulos anteriores.

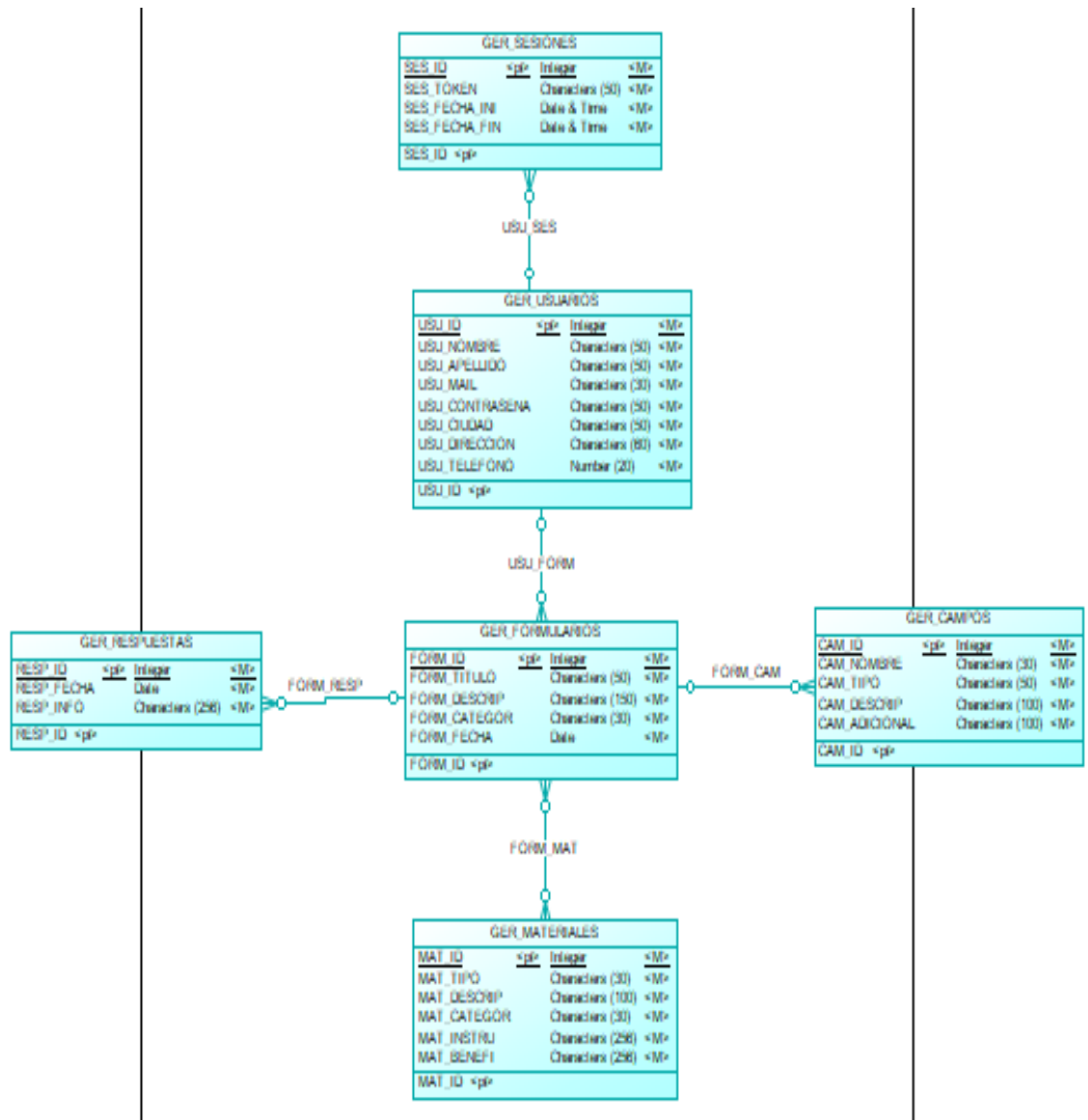
#### **1 MODELO CONCEPTUAL**

A través del diseño del modelo conceptual, se genera la representación abstracta de la base de datos a partir de una enumeración descriptiva de los objetos y relaciones identificados durante el análisis organizacional.

El modelo conceptual puede ser definido como un conjunto de conceptos y principios destinados a representar de manera comprensiva los aspectos lógicos de los diversos tipos de datos que están siendo analizados.

**Figura 2**

*Modelo conceptual de la base de datos*



## **2 MODELO FÍSICO**

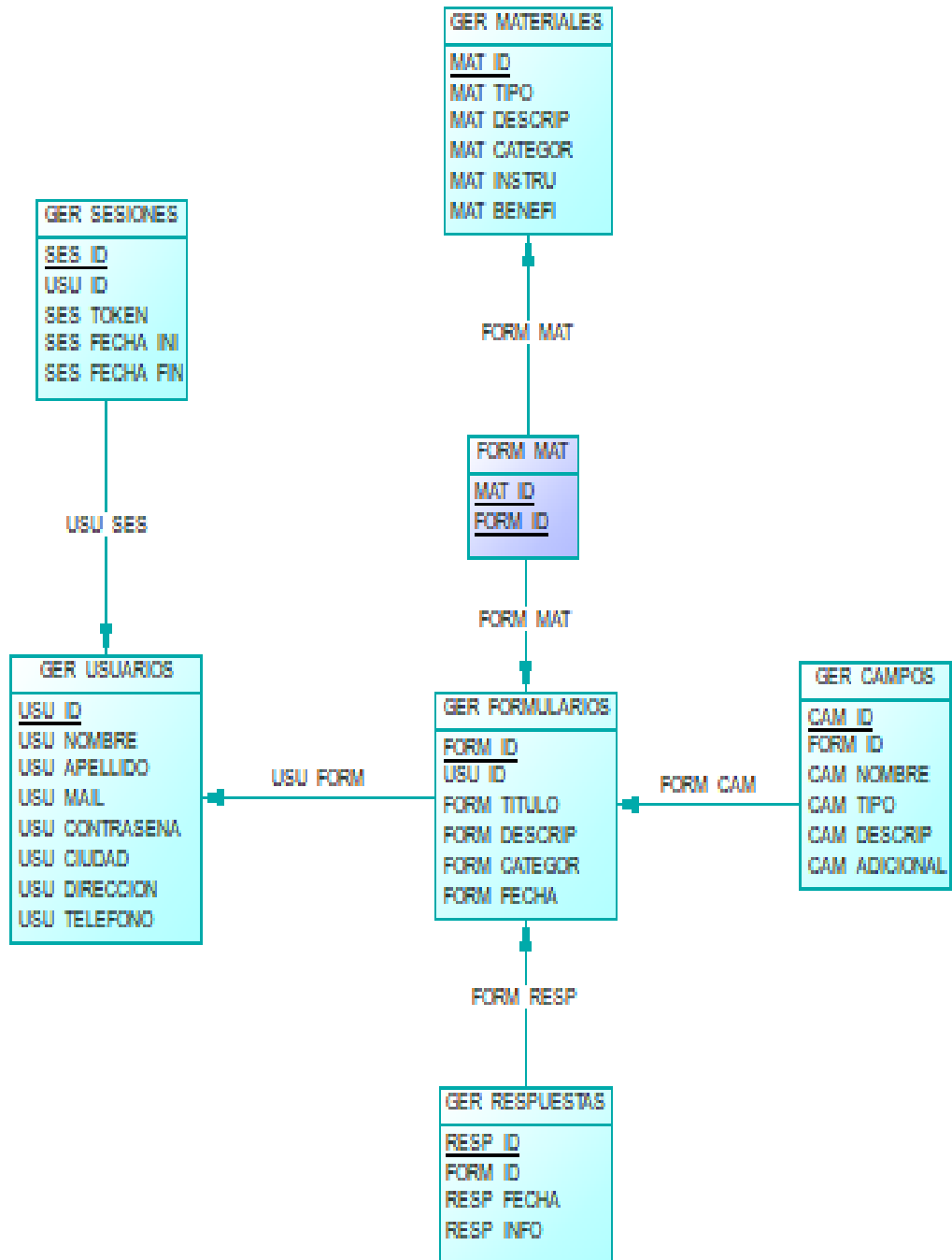
Después de haber definido el modelo conceptual de la base de datos, el diseño lógico posibilita la representación eficiente de ésta utilizando recursos disponibles para estructurar los datos y aplicar las restricciones especificadas en el modelo lógico.

El propósito del modelo físico es transformar el esquema conceptual de datos en un esquema lógico que sea compatible con el sistema de gestión de bases de datos que se utilizará. Además, el modelo físico es un esquema que conceptualmente representa la estructura de una base de datos.

Es un esquema que depende del gestor de base de datos que se vaya a utilizar.

**Figura 3**

*Modelo físico de la base de datos*



### 3 DIAGRAMAS DE CASOS DE USO

Los diagramas de casos de uso son una herramienta fundamental en el análisis de sistemas de software, ofreciendo una representación visual de las interacciones entre los usuarios y el sistema. Esta representación facilita la comprensión de los requisitos funcionales del sistema, identificando las diversas funciones que los usuarios desean realizar y cómo estas interactúan con el sistema.

Estos diagramas actúan como una base crucial para el diseño y la implementación del sistema. Permiten derivar escenarios detallados, diagramas de secuencia y otros artefactos de diseño, orientando la creación efectiva del software

UML, como estándar de modelado, desempeña un papel vital al ofrecer una notación común para la creación de diagramas de casos de uso. Esta estandarización facilita la comprensión, independientemente de la plataforma o tecnología utilizada.

#### **Figura 4**

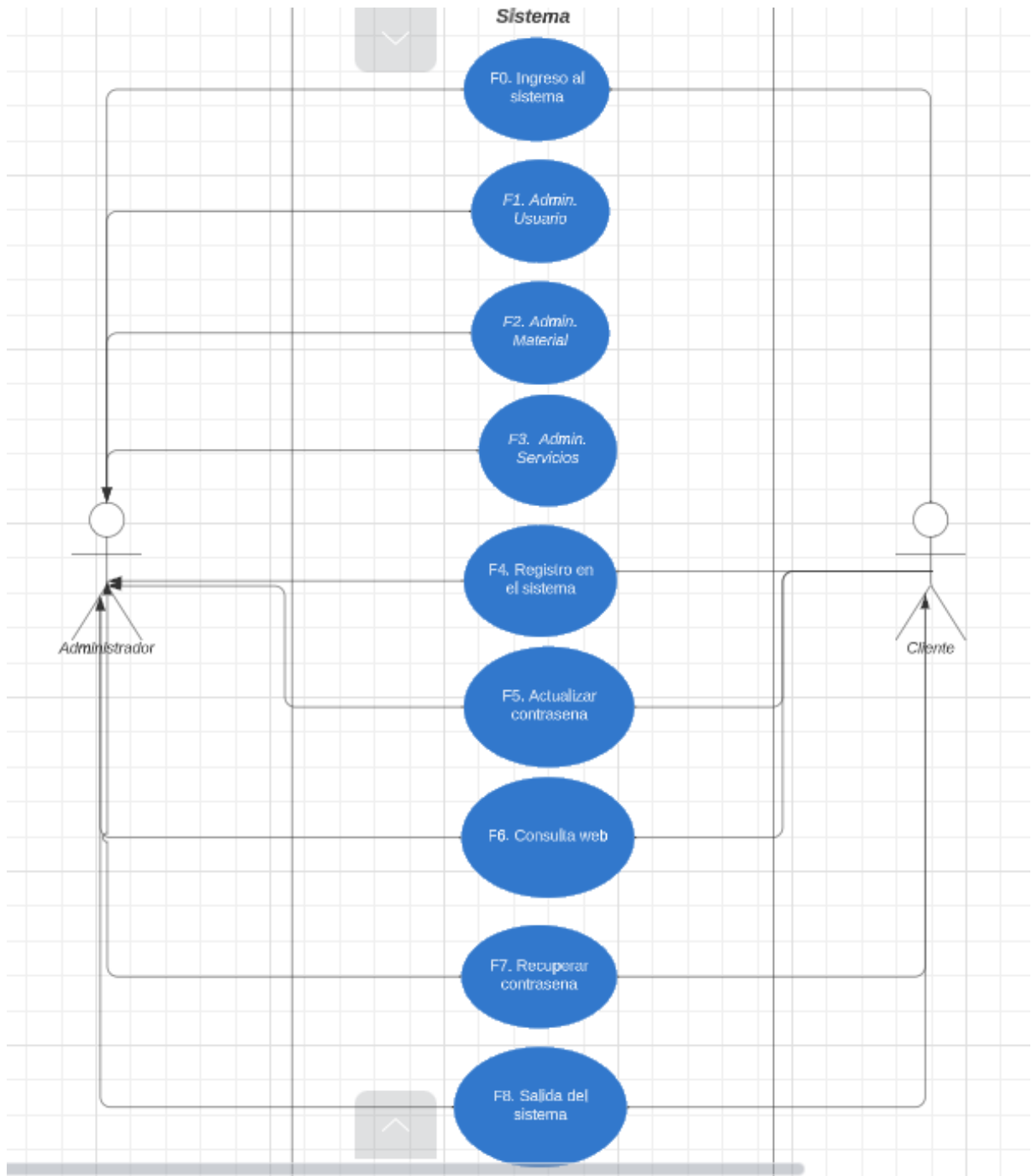
*Representación de un caso de uso y un actor*



A continuación, se expondrá el diagrama de casos de uso general del sistema y los casos de uso a detalle

**Figura 5**

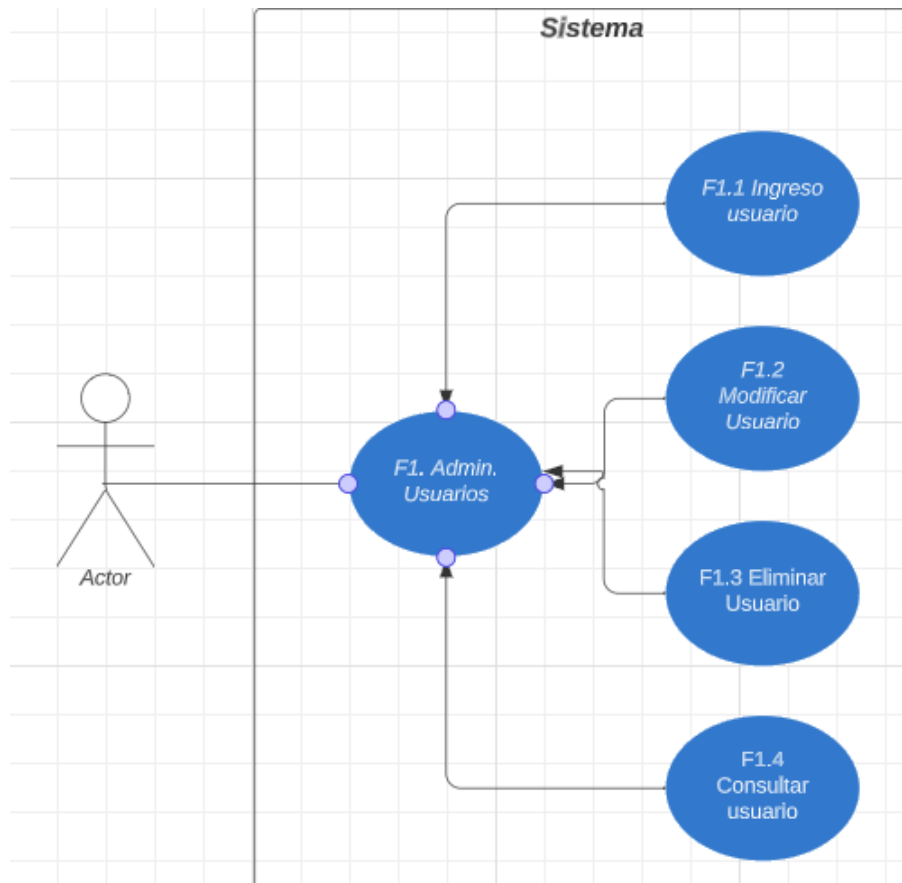
*Diagrama general de casos de uso de la aplicación web*



## F1 Administración de Usuario

**Figura 6**

*Diagrama de caso de uso de la Administración de usuarios.*



## CAPÍTULO VI DESARROLLO DEL SISTEMA

### ITERACIÓN 1

#### HISTORIA DE USUARIO

Historia de Usuario	
<b>ID:</b> HU1	<b>Usuario:</b> Administrador y empleado
<b>Iteración asignada:</b> 1	<b>Prioridad:</b> Alta
<b>Nombre:</b> Administrar a los clientes	
<b>Descripción:</b>  Quiero que el administrador pueda agregar, modificar y quitar viejos y nuevos clientes, también poder consultarlos mediante su numero de cedula y observar el historial de compras.	

Pruebas de aceptación
<ul style="list-style-type: none"> <li>• Introducir una cédula incorrecta y comprobar que se indique el error.</li> <li>• Introducir un usuario ya existente al agregar, y comprobar que el sistema no permita agregarlo nuevamente.</li> <li>• Dejar todos los campos vacíos y comprobar que existan validaciones.</li> <li>• Introducir todos los datos correctos y comprobar que el sistema registre la información.</li> <li>• Modificar un cliente y comprobar que se actualice.</li> <li>• Eliminar un cliente.</li> </ul>

Caso de Prueba de Aceptación	
<b>ID:</b> HU1_P01	<b>Historia de Usuario:</b> HU1
<b>Nombre</b>	Ingresar una cedula incorrecta y comprobar que se indique el error
<b>Descripción</b>	Se verificará que se validen las cedulas de manera correcta
<b>Condiciones de ejecución</b>	Usuario con permiso de administrador o empleado
<b>Pasos de ejecución</b>	<ul style="list-style-type: none"> <li>• Se ingresará a la interfaz de Ingreso a clientes</li> <li>• Se escribirá una cedula invalida</li> </ul>

<b>Resultado esperado</b>	Se muestra el error.
<b>Evaluación de la prueba</b>	Prueba satisfactoria

Caso de Prueba de Aceptación	
<b>ID:</b> HU1_P02	<b>Historia de Usuario:</b> HU1
<b>Nombre</b>	Ingresar un usuario ya existente, y verificar que el sistema no permita agregarlo nuevamente.
<b>Descripción</b>	Se verificará que no se puede agregar mas de un cliente con el mismo número de cedula.
<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>• Usuario con permiso de administrador o empleado</li> <li>• Que un cliente este registrado</li> </ul>
<b>Pasos de ejecución</b>	<ul style="list-style-type: none"> <li>• Se ingresará a la interfaz de Ingreso a clientes</li> <li>• Se seleccionará el tipo de documento cedula</li> <li>• Se introducirá el numero de cedula de un cliente ya existente.</li> <li>• Se completará los demás datos del formulario.</li> <li>• Se presionará el botón registrar</li> </ul>
<b>Resultado esperado</b>	Se muestra mensaje de error.
<b>Evaluación de la prueba</b>	Prueba satisfactoria

Caso de Prueba de Aceptación	
<b>ID:</b> HU1_P03	<b>Historia de Usuario:</b> HU1
<b>Nombre</b>	No llenar ningún campo del formulario y comprobar que existan validaciones.
<b>Descripción</b>	Se probará que las validaciones estén funcionando
<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>• Usuario con permiso de administrador o empleado.</li> </ul>
<b>Pasos de ejecución</b>	<ul style="list-style-type: none"> <li>• Se ingresará a la interfaz de Ingreso a clientes</li> <li>• Se presionará el botón registrar</li> </ul>
<b>Resultado esperado</b>	Se muestra el mensaje de error y se indican los errores.
<b>Evaluación de la prueba</b>	Prueba satisfactoria

Caso de Prueba de Aceptación	
<b>ID:</b> HU1_P04	<b>Historia de Usuario:</b> HU1
<b>Nombre</b>	Ingresar todos los datos correctos y comprobar que el sistema los guarde.
<b>Descripción</b>	Se verificará que se puedan agregar clientes de manera correcta
<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>• Usuario con permiso de administrador o empleado</li> </ul>
<b>Pasos de ejecución</b>	<ul style="list-style-type: none"> <li>• Se ingresará a la interfaz de Ingreso a clientes</li> <li>• Se completará todos los recuadros del formulario con la información correcta</li> <li>• Se presionará el botón registrar</li> </ul>
<b>Resultado esperado</b>	Se registraran los datos del cliente en la base de datos y se presentara un mensaje de guardado.
<b>Evaluación de la prueba</b>	Prueba satisfactoria

Caso de Prueba de Aceptación	
<b>ID:</b> HU1_P05	<b>Historia de Usuario:</b> HU1
<b>Nombre</b>	Modificar un cliente y comprobar que se actualice.
<b>Descripción</b>	Se verificará que se pueda modificar la información de los clientes satisfactoriamente.
<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>• Usuario con permiso de administrador o empleado</li> <li>• Que un cliente este registrado</li> </ul>
<b>Pasos de ejecución</b>	<ul style="list-style-type: none"> <li>• Se ingresará a la interfaz de Ingreso a clientes</li> <li>• Se seleccionará un usuario.</li> <li>• Se aplastará en el botón modificar.</li> <li>• Se modificará cualquier dato del usuario</li> <li>• Se presionará el botón actualizar</li> </ul>
<b>Resultado esperado</b>	Se actualizara la información del cliente en la base de datos y se mostrara un mensaje de actualizado.
<b>Evaluación de la prueba</b>	Prueba satisfactoria

Caso de Prueba de Aceptación	
<b>ID:</b> HU1_P06	<b>Historia de Usuario:</b> HU1
<b>Nombre</b>	Eliminar un cliente
<b>Descripción</b>	Se verificará que se pueda eliminar clientes del sistema

<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>• Usuario con permiso de administrador o empleado</li> <li>• Que un cliente este registrado</li> </ul>
<b>Pasos de ejecución</b>	<ul style="list-style-type: none"> <li>• Se ingresará a la interfaz de Ingreso a clientes</li> <li>• Se seleccionará un cliente</li> <li>• Se debe aplastar en el botón eliminar</li> <li>• Se presionará el botón confirmar</li> </ul>
<b>Resultado esperado</b>	Se eliminará de la base de datos la información del cliente y se mostrara el mensaje de eliminado
<b>Evaluación de la prueba</b>	Prueba satisfactoria

## Historia de Usuario 2

Historia de Usuario	
<b>ID:</b> HU2	<b>Usuario:</b> Administrador
<b>Iteración asignada:</b> 2	<b>Prioridad:</b> Media
<b>Nombre:</b> Administrar categorías de los clientes	
<b>Descripción:</b> Quiero que el administrador pueda agregar, modificar y eliminar categorías de clientes, para que puedan ser utilizadas en la parte de manejar clientes. Solo es necesario el nombre de la categoría.	

Pruebas de aceptación
<ul style="list-style-type: none"> <li>• Dejar todos los campos vacíos y comprobar que existan validaciones.</li> <li>• Introducir todos los datos correctos y comprobar que el sistema registre la información.</li> <li>• Modificar una categoría y comprobar que se actualice.</li> <li>• Eliminar una categoría.</li> </ul>

Caso de Prueba de Aceptación	
<b>ID:</b> HU2_P01	<b>Historia de Usuario:</b> HU2
<b>Nombre</b>	No llenar ningún campo del formulario y comprobar que existan validaciones.
<b>Descripción</b>	Se probará que las validaciones estén funcionando
<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>• Usuario con permiso de administrador.</li> </ul>

<b>Pasos de ejecución</b>	<ul style="list-style-type: none"> <li>• Se ingresará a la interfaz de Ingresar Categoría.</li> <li>• Se presionará el botón registrar</li> </ul>
<b>Resultado esperado</b>	Se muestra el mensaje de error y se indican los errores.
<b>Evaluación de la prueba</b>	Prueba satisfactoria

Caso de Prueba de Aceptación	
<b>ID:</b> HU2_P02	<b>Historia de Usuario:</b> HU2
<b>Nombre</b>	Introducir todos los datos correctos y comprobar que el sistema registre los datos.
<b>Descripción</b>	Se probará que se puedan ingresar nuevas categorías satisfactoriamente.
<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>• Usuario con permiso de administrador.</li> </ul>
<b>Pasos de ejecución</b>	<ul style="list-style-type: none"> <li>• Se ingresará a la interfaz de Ingresar Categoría</li> <li>• Se ingresará una nueva categoría</li> <li>• Se presionará el botón registrar</li> </ul>
<b>Resultado esperado</b>	Se guardará la información de la nueva categoría en la base de datos y se presentará un mensaje de guardado
<b>Evaluación de la prueba</b>	Prueba satisfactoria

Caso de Prueba de Aceptación	
<b>ID:</b> HU2_P03	<b>Historia de Usuario:</b> HU2
<b>Nombre</b>	Modificar una categoría y comprobar que se actualice
<b>Descripción</b>	Se probará que se pueda actualizar una categoría correctamente
<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>• Usuario con permiso de administrador.</li> <li>• Categoría existente.</li> </ul>
<b>Pasos de ejecución</b>	<ul style="list-style-type: none"> <li>• Se ingresará a la interfaz de Administrar categorías</li> <li>• Se seleccionará una categoría</li> <li>• Se presionará en el botón modificar</li> <li>• Se modificará el nombre de la categoría</li> <li>• Se presionará el botón actualizar</li> </ul>
<b>Resultado esperado</b>	Se actualizara la información de la categoría en la base de datos y mostrara el mensaje de actualizado.
<b>Evaluación de la prueba</b>	Prueba satisfactoria

Caso de Prueba de Aceptación	
<b>ID:</b> HU2_P04	<b>Historia de Usuario:</b> HU2
<b>Nombre</b>	Eliminar una categoría
<b>Descripción</b>	Se probará que se pueda eliminar una categoría del sistema
<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>• Usuario con permiso de administrador.</li> <li>• Categoría existente</li> </ul>
<b>Pasos de ejecución</b>	<ul style="list-style-type: none"> <li>• Se ingresará a la interfaz de Administrar Categorías.</li> <li>• Se seleccionará una categoría.</li> <li>• Se presionará el botón eliminar.</li> <li>• Se aplastará el botón confirmar.</li> </ul>
<b>Resultado esperado</b>	Se eliminará la información de la categoría y se mostrara el mensaje de eliminada
<b>Evaluación de la prueba</b>	Prueba satisfactoria

### Historia de Usuario 3

Historia de Usuario	
<b>ID:</b> HU3	<b>Usuario:</b> Administrador
<b>Iteración asignada:</b> 2	<b>Prioridad:</b> Alta
<b>Nombre:</b> Administrar pedidos	
<p><b>Descripción:</b></p> <p>Quiero que el administrador pueda agregar nuevos pedidos, en estos podrán ingresar el documento del cliente y si existe, se cargará la información, caso contrario se tendrá que agregar uno nuevo. Ellos podrán modificar el estado de los pedidos pendientes, y podrán borrar cualquier pedido sin importar su estado. En los pedidos se podrá seleccionar los productos y luego indicar la cantidad para que carguen los subtotales y los totales. El stock de los productos se deberá actualizar cuando el pedido se encuentre en estado "Completado".</p>	

Pruebas de aceptación
<ul style="list-style-type: none"> <li>• Introducir la identificación de un cliente existente y comprobar que el sistema cargué la información.</li> <li>• Introducir la identificación de un nuevo cliente y verificar que se pueda agregarlo.</li> <li>• Dejar los campos vacíos y comprobar que existan validaciones.</li> <li>• Introducir todos los datos correctos y comprobar que el sistema registre la información.</li> </ul>

- Validar que el stock de productos se actualice si el pedido se encuentra completado.
- Eliminar un pedido.

Caso de Prueba de Aceptación	
<b>ID:</b> HU3_P01	<b>Historia de Usuario:</b> HU3
<b>Nombre</b>	Introducir la cedula de un cliente existente y comprobar que el sistema cargue la información.
<b>Descripción</b>	Se probará que se pueda indicar la información de un cliente a través de una búsqueda
<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>• Usuario con permiso de administrador o empleado.</li> <li>• Cliente registrado.</li> </ul>
<b>Pasos de ejecución</b>	<ul style="list-style-type: none"> <li>• Se ingresará a la interfaz de Ingresar Pedido</li> <li>• Se seleccionará el tipo de documento cedula</li> <li>• Se ingresará el documento de un cliente existente</li> <li>• Se presionará el botón buscar</li> </ul>
<b>Resultado esperado</b>	Se presentará los datos del cliente encontrado y se cargara la información de los pedidos que ha realizado en el caso de haber hecho alguna compra.
<b>Evaluación de la prueba</b>	Prueba satisfactoria

Caso de Prueba de Aceptación	
<b>ID:</b> HU3_P02	<b>Historia de Usuario:</b> HU3
<b>Nombre</b>	Introducir la identificación de un nuevo cliente y verificar que se pueda agregarlo.
<b>Descripción</b>	Se probará que se pueda agregar nuevos clientes desde la interfaz de pedidos
<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>• Usuario con permiso de administrador o empleado.</li> </ul>
<b>Pasos de ejecución</b>	<ul style="list-style-type: none"> <li>• Se ingresará a la interfaz de Pedido.</li> <li>• Se seleccionará el tipo de documento cedula.</li> <li>• Se ingresara una cedula que no exista en el sistema</li> <li>• Se presionara el botón buscar</li> <li>• El sistema le presentara el mensaje de cliente no encontrado.</li> <li>• Se presiona sobre el botón de registrarse.</li> </ul>

	<ul style="list-style-type: none"> <li>• Se completara todos los recuadros del formulario satisfactoriamente.</li> <li>• Se presionara el botón registrar</li> </ul>
<b>Resultado esperado</b>	Se registrara la información del nuevo cliente en la base de datos y se cargara la información en el pedido.
<b>Evaluación de la prueba</b>	Prueba satisfactoria

Caso de Prueba de Aceptación	
<b>ID:</b> HU3_P03	<b>Historia de Usuario:</b> HU3
<b>Nombre</b>	Dejar todos los campos vacíos y comprobar que existan validaciones
<b>Descripción</b>	Se probará que las validaciones estén funcionando
<b>Condiciones de ejecución</b>	<ul style="list-style-type: none"> <li>• Usuario con permiso de administrador o empleado.</li> </ul>
<b>Pasos de ejecución</b>	<ul style="list-style-type: none"> <li>• Se ingresará a la interfaz de Ingreso a Pedido.</li> <li>• Se presionará el botón agregar pedido</li> </ul>
<b>Resultado esperado</b>	Se muestra el mensaje de error y se indican los errores.
<b>Evaluación de la prueba</b>	Prueba satisfactoria

## Historia de Usuario 4

Historia de Usuario	
<b>ID:</b> HU4	<b>Usuario:</b> Administrador
<b>Iteración asignada:</b> 3	<b>Prioridad:</b> Alta
<b>Nombre:</b> Administrar el inventario	
<b>Descripción:</b> Quiero que el administrador pueda agregar, modificar y quitar productos. Podrán modificar el stock del producto, ya sea añadiendo más unidades o quitando y podrán ingresar el costo total o una breve descripción para tener un mejor manejo de inventario. En cada producto será necesario subir una imagen de este para tener la referencia. Además, se deberá incluir un stock mínimo y medio para que se pinten los productos dependiendo del stock actual.	

Pruebas de aceptación
<ul style="list-style-type: none"><li>• Dejar todos los campos vacíos y comprobar que existan validaciones.</li><li>• Introducir todos los datos correctos y comprobar que el sistema registre la información.</li><li>• Actualizar un producto.</li><li>• Buscar un producto mediante su clave.</li></ul>

## Historia de Usuario 5

Historia de Usuario	
<b>ID:</b> HU5	<b>Usuario:</b> Administrador
<b>Iteración asignada:</b> 3	<b>Prioridad:</b> Alta
<b>Nombre:</b> Administrar a los empleados	
<b>Descripción:</b> Quiero que el administrador pueda registrar nuevos empleados, pero también modificar y eliminar a los que ya se encuentren registrados. Se deberá agregar una forma de generar una contraseña y también se incluirá la selección del rango del empleado en su registro y actualización. También quiero que se puedan consultar a los empleados mediante su identificación, nombre o apellido.	

Pruebas de aceptación
-----------------------

<ul style="list-style-type: none"> <li>• Introducir una cédula incorrecta y comprobar que se indique el error.</li> </ul>
<ul style="list-style-type: none"> <li>• Introducir un empleado ya existente al agregar, y comprobar que el sistema no permita agregarlo nuevamente.</li> <li>• Dejar todos los campos vacíos y comprobar que existan validaciones.</li> <li>• Introducir todos los datos correctos y comprobar que el sistema registre la información.</li> <li>• Eliminar un empleado.</li> </ul>

### Historia de Usuario 6

Historia de Usuario	
<b>ID:</b> HU6	<b>Usuario:</b> Administrador
<b>Iteración asignada:</b> 4	<b>Prioridad:</b> Baja
<b>Nombre:</b> Notificaciones mediante WhatsApp	
<b>Descripción:</b> Quiero que me lleguen alertas a mi WhatsApp en función al stock del producto, siempre y cuando sea igual o menor al stock mínimo.	

Pruebas de aceptación
<ul style="list-style-type: none"> <li>• Actualizar el stock de un producto (bajo el mínimo) y comprobar que funcionen las alertas.</li> </ul>

### Historia de Usuario 7

Historia de Usuario	
<b>ID:</b> HU7	<b>Usuario:</b> Administrador y empleado
<b>Iteración asignada:</b> 4	<b>Prioridad:</b> Media
<b>Nombre:</b> Emisión de notas de venta.	
<b>Descripción:</b> Quiero que el administrador pueda tener una ayuda mediante un documento que tenga la información del pedido (simulando una nota de venta), esto servirá únicamente para completar las notas de venta de forma manual.	

Pruebas de aceptación
<ul style="list-style-type: none"> <li>• Obtener la nota de venta de un pedido.</li> </ul>

### Historia de Usuario 8

Historia de Usuario	
<b>ID:</b> HU8	<b>Usuario:</b> Administrador
<b>Iteración asignada:</b> 4	<b>Prioridad:</b> Alta
<b>Nombre:</b> Dashboard	
<b>Descripción:</b> Quiero que el administrador pueda ver un conteo de los productos con stock bajo, medio y normal a través de una grafica.	

Pruebas de aceptación
<ul style="list-style-type: none"> <li>• Comprobar que se presente una gráfica que muestre el número de pedidos con respecto al tiempo.</li> </ul>

## GLOSARIO

### A

**Aplicación web:** Una aplicación web es un programa o software que se ejecuta en un navegador web y permite realizar tareas específicas o acceder a funciones a través de una interfaz gráfica.

**Adaptabilidad (responsive):** Se refiere a la capacidad de un sitio web para adaptarse y verse correctamente en diferentes dispositivos y tamaños de pantalla, como computadoras de escritorio, tabletas y teléfonos móviles.

### B

**Base de datos:** Una base de datos es un conjunto organizado de información almacenada de forma estructurada en un sistema informático. Esta información puede ser de diversa

naturaleza, como datos numéricos, textos, imágenes, entre otros. Las bases de datos permiten almacenar, organizar, recuperar y manipular esta información de manera eficiente.

## D

**Desarrollo web:** El desarrollo web se refiere al proceso de crear y construir sitios web y aplicaciones web. Involucra una variedad de tareas que van desde la planificación y diseño hasta la programación y mantenimiento de un sitio web.

## F

**Frontend:** El término frontend se refiere a la parte de un sistema informático, aplicación o página web que interactúa directamente con el usuario. Es la interfaz visible y la experiencia que experimenta un usuario al utilizar un programa o navegar en un sitio web.

## G

**GIT:** Es un sistema de control de versiones distribuido de código abierto que se utiliza para llevar un registro de los cambios en el código fuente durante el desarrollo de software.

## I

**IDE (Entorno de desarrollo integrado):** Es una aplicación informática que proporciona a los programadores todas las herramientas necesarias para escribir, editar, compilar y depurar código de programación en un solo lugar.

**Internet:** Internet es una red global de computadoras interconectadas que permite la comunicación y el intercambio de información entre usuarios de todo el mundo. Esta red se compone de una amplia variedad de dispositivos, servidores y sistemas de comunicación que están interconectados a través de diferentes tecnologías y protocolos.

## S

**Software:** El software se refiere al conjunto de programas, instrucciones y datos que permiten a un sistema informático realizar tareas específicas. Es la parte lógica y no tangible de un sistema de cómputo, que comprende los programas y algoritmos necesarios para que el hardware funcione de manera adecuada.

**SPA (single page application):** Es un tipo de aplicación web que carga una única página inicial desde el servidor y luego actualiza dinámicamente su contenido, mediante las interacciones del usuario, sin necesidad de cargar páginas adicionales del servidor.

## BIBLIOGRAFÍA

- Santiago Bedoya Monsalve, K. A. (diciembre de 2020). *researchgate.net*. Obtenido de researchgate.net:  
[https://www.researchgate.net/publication/348350110\\_Identificacion\\_del\\_desarrollo\\_de\\_aplicaciones\\_web\\_en\\_la\\_actualidad](https://www.researchgate.net/publication/348350110_Identificacion_del_desarrollo_de_aplicaciones_web_en_la_actualidad)
- A, D. (11 de enero de 2023). *Hostinger Tutoriales*. Obtenido de Hostinger Tutoriales:  
<https://www.hostinger.es/tutoriales/que-es-angular>
- ADM. (10 de noviembre de 2023). *ADM Cloud and Services*. Obtenido de ADM Cloud and Services:  
<https://admcloidservices.com/blog/que-es-un-framework-tipos-de-frameworks.html>
- Aldeahost, S. (6 de julio de 2020). *AldeaHost*. Obtenido de AldeaHost:  
<https://aldeahost.com.mx/que-es-mysql-ventajas-y-desventajas/>
- Arquitectura, E. d. (24 de noviembre de 2020). *Ingeniería y Arquitectura*. Obtenido de <https://postgradoingenieria.com/que-son-aplicaciones-web/>
- AZUL, I. T. (10 de noviembre de 2023). *INSTITUTO TECNOLÓGICO DE CERRO AZUL*. Obtenido de INSTITUTO TECNOLÓGICO DE CERRO AZUL: <https://nagaaralive0012rh.wixsite.com/mi-pagina-web/12-arquitectura-de-las-aplicaciones-web>
- Cruz, J. C. (4 de mayo de 2020). *JC on Software*. Obtenido de JC on Software:  
<https://jconsoftwares.com/blog/article/11>
- Designer, P. (2020). *Power Designer*. Obtenido de Power Designer:  
<https://www.powerdesigner.biz/ES/powerdesigner/powerdesigner-features.html>
- Diaz, J. (2017). *EDteam*. Obtenido de Edteam: <https://ed.team/blog/normalizacion-de-bases-de-datos>
- EDteam. (2019). *EDteam*. Obtenido de EDteam: <https://ed.team/blog/que-es-go-golang>
- Etece. (5 de agosto de 2021). *Concepto*. Obtenido de Concepto: <https://concepto.de/lenguaje-de-programacion/>
- Etece, E. (5 de agosto de 2021). *Concepto*. Obtenido de Concepto: <https://concepto.de/base-de-datos/>
- Eugenia, B. R. (mayo de 2014). *Repositorio Institucional de la UTPL*. Obtenido de Repositorio Institucional de la UTPL:  
[https://dspace.utpl.edu.ec/bitstream/123456789/20351/1/Bolanos\\_Rosero\\_Sandra\\_Eugenia.pdf](https://dspace.utpl.edu.ec/bitstream/123456789/20351/1/Bolanos_Rosero_Sandra_Eugenia.pdf)
- Flores, F. (22 de julio de 2022). *Open Webinars*. Obtenido de Open Webinars:  
<https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/>
- Gonçalves, M. J. (13 de octubre de 2021). *Hiberus Blog*. Obtenido de Hiberus Blog :  
<https://www.hiberus.com/crecemos-contigo/que-es-angular-y-para-que-sirve/>

Guaman, C. (12 de noviembre de 2012). *Prezi*. Obtenido de Prezi: <https://prezi.com/urpqaap--bbw/powerdesigner/>

Gwen. (7 de noviembre de 2023). *Abrirarchivos*. Obtenido de Abrirarchivos: <https://abrirarchivos.info/tema/software-de-desarrollo-tipos-ejemplos-y-tendencias/>

Harsh, K. (14 de octubre de 2022). *Kinsta*. Obtenido de Kinsta: <https://kinsta.com/es/blog/arquitectura-aplicaciones-web/#:~:text=La%20arquitectura%20de%20las%20aplicaciones%20web%20define%20c%C3%B3mo%20se%20estructuran,causar%20estragos%20en%20tu%20negocio.>

hat, R. (15 de marzo de 2023). *Red hat*. Obtenido de Red Hat: <https://www.redhat.com/es/topics/cloud-native-apps/what-is-an-application-architecture>

Hootsuite. (julio de 2022). *Hootsuite*. Obtenido de [https://hootsuite.widen.net/s/kd6qgn9rwx/digital2022globaloverview\\_report\\_en](https://hootsuite.widen.net/s/kd6qgn9rwx/digital2022globaloverview_report_en)

IONOS, D. G. (24 de marzo de 2020). *Digital Guide IONOS*. Obtenido de Digital Guide IONOS: <https://www.ionos.es/digitalguide/paginas-web/creacion-de-paginas-web/certificados-ssl-y-https-maxima-seguridad-para-tu-web/>

IT, R. (12 de abril de 2021). *Reclu IT*. Obtenido de Reclu IT: <https://recluit.com/que-es-visual-studio-code/>

Mendez Bravo, D. A. (2015). *app.ute.edu.ec*. Obtenido de [app.ute.edu.ec/](https://repositorio.flacsoandes.edu.ec/bitstream/10469/18778/2/TFLACSO-2022GCPL.pdf): <https://repositorio.flacsoandes.edu.ec/bitstream/10469/18778/2/TFLACSO-2022GCPL.pdf>

Mendoza, M. L. (16 de julio de 2020). *Open Webinars*. Obtenido de Open Webinars: <https://openwebinars.net/blog/que-es-un-lenguaje-de-programacion/>

Muente, G. (8 de enero de 2020). *Rockcontent*. Obtenido de Rockcontent: <https://rockcontent.com/es/blog/framework/>

Mytaskpanel. (2022). *Mytaskpanel*. Obtenido de Mytaskpanel: <https://www.mytaskpanel.com/lenguaje-programacion-go/>

Oracle. (8 de noviembre de 2023). *Oracle*. Obtenido de Oracle: <https://www.oracle.com/mx/database/what-is-database/>

Robledano, A. (24 de septiembre de 2019). *Open Webinars*. Obtenido de Open Webinars: <https://openwebinars.net/blog/que-es-mysql/>

Santiago Bedoya Monsalve, K. A. (diciembre de 2020). *researchgate.net*. Obtenido de [researchgate.net](https://www.researchgate.net/publication/348350110_Identificacion_del_desarrollo_de_aplicaciones_web_en_la_actualidad): [https://www.researchgate.net/publication/348350110\\_Identificacion\\_del\\_desarrollo\\_de\\_aplicaciones\\_web\\_en\\_la\\_actualidad](https://www.researchgate.net/publication/348350110_Identificacion_del_desarrollo_de_aplicaciones_web_en_la_actualidad)

Service, A. W. (10 de 11 de 2023). *AWS*. Obtenido de AWS: <https://aws.amazon.com/es/what-is/web-application/>

Techtarget. (abril de 2021). *Computer weckly*. Obtenido de Computer weckly: <https://www.computerweekly.com/es/definicion/MySQL>

Torrejón, H. C. (24 de noviembre de 2022). *Open Webinars*. Obtenido de Open Webinars: <https://openwebinars.net/blog/como-realizar-la-normalizacion-de-bases-de-datos-y-porque/>

Workana. (2021). *Workana*. Obtenido de Workana: <https://i.workana.com/glosario/que-es-golang/>