

**TRABAJO DE GRADO**  
**Proyecto de Grado**

**Diseño de un sistema web para la automatización del proceso de gestión de transferencias en una empresa comercializadora de autopartes y accesorios para vehículos automotores.**

Corporación Universitaria Remington.  
Facultad de ingeniería.  
Ingeniería de sistemas.

Miguel Antonio Caballero Rivera  
Tutor Nixon Alonso Duarte Acosta.  
Proyecto de grado.  
2024.

## **Dedicatoria**

A mi amada esposa.

Este trabajo de grado es para ti, porque has sido parte fundamental en este proceso y un apoyo constante en cada paso de esta carrera. Tu comprensión, tu amor incondicional y tu fe en mi me han ayudado a llegar hasta este gran paso.

Gracias por estar a mi lado, por tu paciencia en los momentos difíciles y por celebrar cada pequeño logro conmigo. Tu amor y dedicación me han ayudado a mantener el rumbo a la meta, y este logro es tan tuyo como mío.

Con todo mi amor,

Miguel Antonio Caballero Rivera.



## Tabla de Contenidos

Resumen.....	8
Palabras clave.....	8
Introducción .....	9
Marco de referencia .....	10
Estado del arte.....	11
Marco conceptual.....	13
Marco teórico .....	17
Programación Web.....	17
Programación Frontend.....	18
Lenguaje HTML (HyperText Markup Language).....	18
CSS (Cascading Style Sheets U Hojas De Estilo En Cascada). .....	19
Frameworks CSS. ....	19
JavaScript.....	19
Programación Backend .....	20
PHP (Hypertext Preprocessor).....	20
Sistema Web. ....	20
Programación Orientada A Objetos. ....	21
Algoritmo .....	21
Laravel .....	22
Vue.Js.....	22
MySQL .....	22
MariaDB .....	23
Arquitectura Web.....	23
Modelo – Vista – Controlador (MVC).....	24
Casos De Uso.....	25
Diagrama De Casos De Uso. ....	26
Automatización.....	27
Logística.....	28
Automatización En La Logística .....	28
Gestión De Transferencias De Productos. ....	28
Optimización De Procesos Logísticos. ....	29
Metodología SCRUM .....	29
Planteamiento del problema.....	32
Justificación .....	34
Objetivos .....	36
Objetivo General.....	36
Objetivos específicos .....	36
Metodología .....	37
Tipo De Estudio .....	37
Diseño metodológico .....	37
Fase 1: Análisis Situacional (Blacklog).....	37
Análisis Situacional .....	38
Procesos internos del negocio.....	40
Casos de uso del negocio .....	40

Identificación De Requerimientos .....	50
Product Backlog.....	53
Fase 2: Planificación (Sprint Blacklog) .....	54
Fase 3: Desarrollo .....	59
Arquitectura web del sistema.....	60
Diseño de interfaz del sistema .....	70
Formulario de Notas. ....	74
Formulario de modificación de datos. ....	75
Fase 4: Finalización .....	76
Pruebas.....	78
Resultados Y Discusión .....	80
Cumplimiento de objetivos .....	80
Conclusiones .....	82
Referencias.....	84

### Lista de tablas

Tabla 1. Análisis DOFA del área de logística.....	38
Tabla 2. Tiempos promedio de gestión de transferencia. ....	45
Tabla 3. Reglas del negocio. ....	49
Tabla 4. Stakeholders.....	50
Tabla 5. Requisitos funcionales. ....	50
Tabla 6. Requisitos no funcionales. ....	51
Tabla 7. Restricciones. ....	52
Tabla 8. Product Backlog.....	53
Tabla 9. Sprint Backlog. ....	54
Tabla 10. Project feature requirements. ....	56
Tabla 11. Tablero de logística.....	71
Tabla 12. Orden y eventos. ....	76
Tabla 13. Pruebas.....	78
Tabla 14. Pruebas del sistema en distintos navegadores.....	80

## Lista de figuras

Ilustración 1. Ejemplo de MVC (Ovando, 2019).....	25
Ilustración 2. Ejemplo de diagrama de caso de uso (Departamento Nacional de planeación - DNP, 2020).....	27
Ilustración 3. Sistema manual de logística.....	42
Ilustración 4. Diagrama de flujo de Procesamiento manual de transferencias. ....	44
Ilustración 5. Diagrama Solicitud de seguimiento de salidas a bodegas. ....	47
Ilustración 6. Diagrama Gantt Tareas PFR-1 - PFR-9.....	60
Ilustración 7. Diagrama Gantt Tareas PFR-10 - PFR-16.....	60
Ilustración 8. Diagrama Gantt Tareas PFR-17 - PFR-23.....	60
Ilustración 9. Arquitectura web del sistema.....	62
Ilustración 10. Diagrama de componentes del sistema por capas.....	64
Ilustración 11. Diagrama de nuevo flujo de procesamiento de transferencias. ....	66
Ilustración 12. Consultar tabla de transferencias. ....	67
Ilustración 13. Modelo de base de datos. ....	70
Ilustración 14. Login del sistema web propuesto.....	70
Ilustración 15. Página de transferencia primera sección.....	73
Ilustración 16. Página de transferencia segunda sección.....	73
Ilustración 17. Modal de listado de notas ....	74
Ilustración 18. Formulario de nota.....	74
Ilustración 19. Formulario de modificación de datos de la transferencia. ....	75
Ilustración 20. Colores de los eventos. ....	76

## **Resumen**

El presente proyecto de grado se centra en el desarrollo de un sistema web para la gestión eficiente de transferencias de productos en una empresa dedicada al comercio de partes piezas y accesorios para vehículos automotores. En esta las transferencias realizadas en el área de logística se procesaban manualmente, lo que conllevaba muchos errores humanos, problemas en la lógica del proceso, afectando la precisión y eficiencia en la distribución de productos, asimismo, aumentando los tiempos. Lo anterior, hizo necesario la automatización del proceso de gestión de transferencias con el fin de disminuir los errores y los tiempos de respuesta. Para lograr este objetivo, se utilizó la metodología SCRUM, se identificaron las partes del proceso que requerían ser automatizadas y con base a este análisis, se diseñó y desarrolló un sistema web, utilizando Laravel como framework para el Backend, Vue.js para el Frontend, y el gestor de base de datos MariaDB, que facilitara el procesamiento y seguimiento de la transferencia de productos, y que a su vez, entregara información clave para la toma de decisiones, finalmente se realizaron pruebas funcionales para validar el desempeño del sistema web. La implementación de este sistema web permitió a la empresa superar las limitaciones del proceso manual, optimizando los tiempos y reduciendo los errores asociados. Además, proporcionó herramientas de monitoreo en tiempo real que facilitan una mejor toma de decisiones y una mayor transparencia en las operaciones.

## **Palabras clave**

Automatización; SCRUM; logística; sistema web; empresa de autopartes.

## Introducción

En el ámbito de la logística y la distribución, la automatización de procesos ha demostrado ser una herramienta clave para mejorar la eficiencia y reducir los errores. En empresas comercializadoras, donde la gestión adecuada de las transferencias de productos entre bodegas y clientes es fundamental, la implementación de sistemas automatizados puede transformar significativamente la operación aumentando su rendimiento.

La automatización consiste en usar la tecnología para realizar tareas con muy poca intervención humana, y aunque se puede implementar en cualquier área, es más común en la fabricación, la robótica y los automóviles, así como en los sistemas de TI (Red Hat, 2023).

Para el contexto del presente proyecto, la automatización se proyecta como la solución al problema presentado en el proceso de gestión de transferencias de una empresa dedicada al comercio de partes piezas y accesorios para vehículos automotores, en la cual las transferencias se procesaban manualmente, lo que conllevaba muchos errores humanos y el aumento del tiempo de respuesta dentro del proceso.

La importancia de la realización de proyectos como este radica en la necesidad de las empresas de optimizar sus procesos, especialmente aquellos que presentan más errores y que son vitales para su funcionamiento, en un mercado que cada vez es más competitivo y orientado hacia el avance tecnológico. Es por esto que el objetivo de este proyecto se fundamentó en la automatización del proceso de gestión de transferencias en el área de logística en una empresa comercializadora de autopartes y accesorios para vehículos

automotores a través del desarrollo de un sistema web que permitiera disminuir los tiempos de respuesta dentro de este proceso.

El alcance del presente proyecto abarca el diseño e implementación del sistema web, empleando tecnologías como Laravel y Vue.js, el gestor de bases de datos MariaDB. No contempla los costos asociados. Asimismo, se desarrolló empleando la metodología SCRUM de desde la etapa de planeación.

Finalmente, el presente proyecto se encuentra estructurado de la siguiente forma:

Marco referencial: contiene la consulta de los principales conceptos y teorías de programación, base de datos y logística relacionados con el fin de plantear el diseño del software.

Planteamiento del problema: se explica la problemática identificada en la empresa hasta llegar a la pregunta problema del proyecto.

Justificación: se presenta la importancia de la realización del presente proyecto para la empresa.

Objetivos: se presenta el propósito del proyecto.

Diseño metodológico: se describe el tipo de investigación empleada y las fases de la metodología SCRUM.

Resultados y discusión: se muestran los resultados obtenidos del desarrollo e implementación del sitio web.

Conclusiones: se presentan los principales logros, hallazgos, aprendizajes y limitaciones del presente proyecto.

Marco de referencia

El presente marco de referencia tiene como propósito proporcionar el contexto histórico y conceptual necesario para entender la propuesta del sitio web para la empresa de autopartes. Comprende los antecedentes históricos, teorías y conceptos clave relacionados con la automatización, gestión de transferencias y optimización de procesos logísticos, los cuales guiarán el análisis y la interpretación de los resultados, creando una base sólida que orienta y justifica el desarrollo de la propuesta.

### **Estado del arte**

El desarrollo de sistemas web para la automatización de procesos logísticos y administrativos ha ganado relevancia en diferentes sectores industriales, debido a la necesidad de optimizar tiempos, reducir errores y mejorar la eficiencia operativa. A continuación, se presentan algunos antecedentes relevantes que fundamentan el presente proyecto:

#### **Implementación de un ERP para la automatización del proceso logístico en una empresa de servicios técnicos**

Esta investigación se centró en la implementación del sistema ERP SAP Business One para automatizar los procesos logísticos de una empresa de servicios técnicos. Entre los procesos optimizados se encuentran la compra y venta de repuestos, servicio al cliente y la gestión de inventario. El uso de SAP Business One permitió a la empresa integrar la información de manera accesible para sus empleados, mejorando el flujo de datos y reduciendo la dependencia en personas específicas para acceder a la información. La metodología ASAP fue clave en la implementación, permitiendo ejecutar el proyecto en un menor tiempo y logrando optimizar el control del inventario y la atención al cliente.

Este estudio destaca cómo la automatización mediante un ERP puede reducir tiempos de respuesta y mejorar la eficiencia en la gestión de servicios (Bardales & Galarza, 2018).

### **Desarrollo de un sistema web basado en Laravel y VueJs para la gestión por procesos: Un estudio de caso**

En este estudio de caso, se desarrolló un sistema web para mejorar la gestión por procesos en la compañía de seguridad UNICEPRI, utilizando los frameworks Laravel y Vue.js, y la base de datos MariaDB. El enfoque Modelo Vista-Controlador permitió reducir el consumo de recursos y mejorar la eficiencia en la transferencia de datos, mientras que la metodología SCRUM facilitó la comunicación entre el cliente y el desarrollador, garantizando la entrega de resultados en tiempos óptimos. La evaluación del sistema bajo la norma ISO/IEC 25010 demostró una mejora significativa en la gestión administrativa de la compañía. Este antecedente es particularmente relevante para el presente proyecto, ya que utiliza la misma combinación de tecnologías (Laravel y Vue.js) y la metodología SCRUM, demostrando su efectividad en la mejora de procesos internos (Ávila, 2020).

### **Desarrollo de una aplicación web progresiva para la gestión de pedidos aplicada a la empresa de ropa Hoxton**

El proyecto desarrollado en la empresa textil Hoxton consistió en una aplicación web progresiva (PWA) para la gestión de pedidos. La PWA, desarrollada con React JS y siguiendo la metodología XP, permitió que los usuarios gestionaran los pedidos desde cualquier dispositivo, mejorando la accesibilidad y la experiencia del usuario. Además, el enfoque progresivo garantizó que la aplicación pudiera ejecutarse de manera nativa en teléfonos móviles sin perder funcionalidades clave. Este antecedente resalta la

importancia de usar tecnologías modernas y multiplataforma para la gestión eficiente de procesos, mejorando la interacción entre la empresa y sus clientes (Guerra, 2021).

### **Desarrollo e implementación de un sistema web para la gestión logística en la botica Robles**

La implementación de un sistema web en la botica "Robles" transformó la manera en que la empresa gestionaba sus productos, pasando de un sistema manual a uno automatizado. El sistema, desarrollado utilizando herramientas de código libre como PHP, MySQL y Bootstrap, permitió optimizar el control de inventario y mejorar los procesos logísticos. Antes de la implementación, la botica enfrentaba problemas en la verificación de stock, lo que generaba retrasos en las ventas. Con el nuevo sistema, se logró un manejo más eficiente de las transferencias de productos entre sucursales, incrementando la velocidad de atención al cliente. Este caso demuestra cómo la implementación de sistemas web puede impactar positivamente en pequeñas y medianas empresas del sector farmacéutico (Pachas, 2021).

### **Marco conceptual**

#### **Albarán**

Es un documento que contiene los datos de quien hace la entrega y quien la recibe, los productos y las cantidades. Inicia la operación de flujo físico con productos materiales que implican una transferencia (Lozano Rojo, 2003, pág. 87).

#### **Gestión**

Es un conjunto de procedimientos y acciones que se llevan a cabo para lograr un determinado objetivo (Westreicher, 2024).

#### **Logística**

Conjunto de medios y métodos necesarios para llevar a cabo la organización de una empresa o de un servicio, especialmente de distribución (Real Academia Española, s.f.).

**Transferencias**

Actividad de movimiento interno donde determinados bienes en el almacén cambian de ubicación. Parte también del control de la recepción y salida de mercancía, esta operación demanda un registro exhaustivo de los movimientos (Acacia Technologies, 2024).

**Monitoreo**

Es la observación del curso de uno o más parámetros para detectar eventuales anomalías (Pérez, 2022).

**Optimización**

Buscar la mejor manera de realizar una actividad (Real Academia Española, s.f.).

**Algoritmo**

Es un conjunto de reglas que hay que seguir para realizar una tarea o resolver un problema (Datascientest, 2024).

**Integración**

Proceso que consiste en reunir datos de diferentes fuentes para obtener una vista unificada y más valiosa de ellos, puede consolidar todo tipo de datos para realizar cualquier tipo de tareas (Google Cloud, s.f.).

**Flete**

Es el costo por pagar por el desplazamiento de una carga en un medio de transporte (Páez, 2020).

**Stock**

Es el inventario de productos o materias primas que una empresa almacena para la venta o comercialización (aprende-logistica.com, s.f.).

**Frontend**

Es la parte de una aplicación que interactúa con los usuarios, es conocida como el lado del cliente (Celi, Boné, & Mora, Programación Web del Frontend al Backend., 2023, pág. 6).

**Backend**

Se refiere al interior de las aplicaciones que viven en el servidor y al que a menudo se le denomina “el lado del servidor” (Celi, Boné, & Mora, Programación Web del Frontend al Backend., 2023, pág. 6).

**Sistema Web O Aplicación Web**

Es aquel que está creado e instalado en un servidor en Internet o sobre una intranet (red local) (Pachas, 2021).

**Programación Web**

Es una disciplina que se enfoca en el desarrollo de aplicaciones y sitios web para ser utilizados en internet (Celi, Boné, & Mora, Programación Web del Frontend al Backend., 2023).

**Base De Datos**

Es una recopilación organizada de información o datos estructurados, que normalmente se almacena de forma electrónica en un sistema informático (Oracle, 2020).

**PHP**

Es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML (Pachas, 2021).

**MySQL**

Es un sistema de gestión de base de datos relacional (RDBMS) de código abierto, basado en lenguaje de consulta estructurado (SQL) (Pachas, 2021).

**Programación Orientada A Objetos (POO)**

Es un modelo de programación en el que el diseño de software se organiza alrededor de datos u objetos en lugar de usar funciones y lógica (Universidad Europea, 2024).

**JSON (Javascript Object Notation - Notación De Objetos De Javascript)**

Es un formato ligero de intercambio de datos (Pachas, 2021).

**Javascript**

Es un lenguaje de programación del tipo interpretado, basado en el estándar ECMAScript y se define como un lenguaje orientado a objetos (Luna, 2019, pág. 4).

**Datatable**

Es un complemento para la biblioteca jQuery Javascript (Pachas, 2021).

**Laragon**

Es una herramienta para equipos técnicos que permite crear diferentes entornos de desarrollo, facilitando el trabajo con las aplicaciones. Con ella, se pueden trabajar con lenguajes de programación del lado del servidor, como NodeJS, Ruby, Python y, por supuesto, PHP (García, 2024).

**Framework**

Es un conjunto de código que se puede utilizar para hacer un sistema personalizado muy simple o complejo (Ovando, 2019).

**Browser**

Aplicación que permite al usuario navegar en la World Wide Web (Celi, Boné, & Mora, Programación Web del Frontend al Backend., 2023, pág. 4).

**Servidor Web**

Son ordenadores especializados que vuelven posible el Web hosting; en otras palabras, el servicio de hosting o alojamiento web es el alquiler de un espacio en un servidor para almacenar los archivos de nuestro sitio (Pachas, 2021).

**CSS3**

Es la última evolución del lenguaje de las Hojas de Estilo en Cascada (Cascading Style Sheets) (Pachas, 2021).

**HTML5**

Lenguaje de Marcado de Hipertexto, es un lenguaje que pertenece a la familia de los "lenguajes de marcado" y es utilizado para la elaboración de páginas web (Pachas, 2021).

**HTTP (Protocolo De Transferencia De Hipertexto)**

Es el protocolo usado para el intercambio de información entre los clientes web y los servidores (Cardador, 2014).

**Intranet**

Es una red interna, diseñada y desarrollada para trabajar dentro de los límites de una determinada compañía. Normalmente es privada (Cardador, 2014).

**Transformación Digital**

Es el proceso mediante el cual una organización integra tecnología digital a todas las áreas empresariales (Fuente, 2023).

**Stakeholder**

Personas, grupos o entidades que constituyen una parte interesada en una organización, proyecto o empresa y que pueden verse afectados por las acciones o decisiones tomadas a su respecto (Martins J. , ¿Quiénes son los stakeholders de un proyecto?, 2024).

**Base De Datos Relacional**

Es una base de datos en donde todos los datos visibles al usuario están organizados estrictamente como tablas de valores y todas las operaciones de esta se operan sobre esas tablas (Equipo editorial de IONOS, 2024).

**Código Abierto**

Modelo de producción descentralizada que permite que cualquier persona modifique y comparta tecnología porque su diseño es accesible de manera pública (Basañes, 2019).

**Análisis DOFA**

Es una técnica de análisis que se utiliza para identificar los factores internos y externos que pueden afectar una situación o decisión (Universidad de los Andes, 2023).

**Marco teórico**

**Programación Web.** Este título va sin espacio entre el título y el texto que viene delante, pero debe llevar un punto intermedio. La programación web se puede utilizar para la creación de muchos proyectos, desde las redes sociales, hasta la gestión de tiendas en línea, incrementándose en los últimos años el uso de estas en la vida de las personas. En esta área se usan diferentes lenguajes de programación para crear páginas web

dinámicas y atractivas de tal manera que proporcionen una experiencia al usuario (Celi, Boné, & Mora, 2023).

La programación web se divide en Frontend y Backend. El primero se refiere a la parte visible de una aplicación o sistema web y consta de lenguajes de programación como HTML, CSS y JavaScript, los cuales permiten la creación de páginas web dinámicas. El segundo, describe toda la lógica de negocio de la aplicación, la gestión de bases de datos y la seguridad, usando lenguajes como PHP, Python, Java, entre otros, con el objetivo de crear aplicaciones escalables y robustas que manejen grandes cantidades de tráfico y datos (Celi, Boné, & Mora, 2023).

**Programación Frontend.** Se conoce como la parte de una aplicación con la que los usuarios pueden interactuar, llamada “el lado del cliente”, es decir, todo lo que se ve en la pantalla cuando se accede a un sitio web: letras, colores, adaptaciones, efectos visuales, etc. Todo lo anterior crea la experiencia del usuario, esto implica que cuando un usuario acceda a una página web, este debe ser capaz de navegar en ella a través de una interfaz sencilla, atractiva y funcional. Algunos de los lenguajes más usados en este tipo de programación son HTML5, CSS3, JavaScript, JQuery, Ajax (Celi, Boné, & Mora, 2023).

**Lenguaje HTML (HyperText Markup Language).** Según Celi, Boné y Mora (2023) HTML es el lenguaje de marcado predominante para la construcción de páginas web y se emplea para describir la estructura y contenido en forma de texto, además de complementarlo con objetos (Celi, Boné, & Mora, 2023).

Este lenguaje se escribe como etiquetas o marcas, rodeadas por corchetes angulares (< >). Fue desarrollado por Tim Berners-Lee y se popularizó por el navegador Mosaic. Dentro de las características de HTML está el hecho de que se basa en la utilización de

un sistema de etiquetas restringido, aplicado a un documento de texto. No necesita ser compilado, sino que se ejecuta a medida que se avanza por el documento HTML y su intérprete es el navegador Web (Celi, Boné, & Mora, 2023).

**CSS (Cascading Style Sheets U Hojas De Estilo En Cascada).** Es un lenguaje de programación que permite aplicar estilos a los diferentes elementos de las páginas web, de manera que los títulos, listas y párrafos pueden verse igual en todas y cada una de las páginas (Celi, Boné, & Mora, 2023).

**Frameworks CSS.** Se trata de una biblioteca de estilos visuales genéricos empleados para desarrollar una página web. Además, añaden una serie de utilidades, como componentes para hacer cuadros de diálogo, tablas, carrusel de imágenes, entre otros. Dentro de los frameworks CSS más usados se encuentran Tailwind, Bootstrap, Materialize CSS, Fpundation y Bulma (Celi, Boné, & Mora, 2023).

**JavaScript.** Después de HTML y CSS es la tercera tecnología fundamental para el desarrollo web Frontend. Esta posibilita la interacción con el usuario. Su estandarización con HTML5 permite la creación de diferentes tipos de aplicaciones para ejecutar en cualquier plataforma (Celi, Boné, & Mora, 2023).

Al programar en JavaScript se debe tener en cuenta que la secuencia de sentenciar u órdenes se separan por punto y coma (;). Asimismo, este programa distingue entre mayúsculas y minúsculas, por lo que se debe saber que while no será interpretado igual que WHILE. Además, ignora espacios en blanco, tabuladores y saltos de línea entre palabras reservadas. También, se pueden colocar comentarios que ocupan una o varias líneas (Celi, Boné, & Mora, 2023).

**Programación Backend.** Backend hace referencia al interior de una aplicación, es decir, aquello que vive en el servidor y que comúnmente se llama “el lado del servidor”. Este consiste en un servidor, una aplicación y una base de datos, en palabras de Celi, Boné y Mora (2023) “se toman los datos, se procesa la información y se envía al usuario”. Algunos de los lenguajes de programación más usados en Backend son: ASP.NET, PHP, Python, Ruby, Node.js, Java, MySQL, SQL Server, PostgreSQL, Oracle, MongoDB (pág. 6).

**PHP (Hypertext Preprocessor).** Es un lenguaje de programación que se usa para el desarrollo de aplicaciones web y para crear páginas web porque favorece la conexión entre servidores e interfaz de usuario (Celi, Boné, & Mora, 2023).

PHP es de código abierto, por lo que cualquier persona puede hacer cambios en su estructura, además que puede ser usado sin problemas y está continuamente en perfeccionamiento. Asimismo, es bastante simple y proporciona características avanzadas (Celi, Boné, & Mora, 2023).

**Sistema Web.** Los sistemas web también son conocidos como aplicaciones web y pueden definirse como herramientas en las que los usuarios acceden a un servidor web a través de una intranet o de internet, es decir, programas informáticos que se ejecutan a través de un navegador, programados en lenguaje HTML y que no es necesario instalar en el computador. Estos permiten acceder a la información de una manera rápida y sencilla (Valarezo, Honores, & Gómez, 2018).

Es interesante analizar el origen de los sistemas webs, los cuales surgieron porque en un principio la Web se trataba de páginas estáticas que sólo podían descargarse y consultarse a la vez, lo que implicó la necesidad de crear sitios dinámicos, lo que al

mismo tiempo llevó a la construcción de sistemas de ejecución de módulos integrados al servidor y a la utilización de un lenguaje de programación que pudiera ser interpretado por el servidor. De esto se deriva el hecho de que el funcionamiento de la Web se fundamentara en el protocolo HTTP y el lenguaje HTML (Valarezo, Honores, & Gómez, 2018).

Por otro lado, los sistemas webs se relacionan con el almacenamiento en la nube, en grandes servidores de internet (Ovando, 2019). Estos son populares por la practicidad del navegador Web, su facilidad de actualización y mantenimiento de las aplicaciones.

**Programación Orientada A Objetos.** Es un tipo de programación que se puso en auge a finales de 1980. Está caracterizada por conceptos como Objetos, Clases, Encapsulación, Herencia y polimorfismo. En palabras de “permite manejar la complejidad de los programas y la reutilización de código porque permite una mayor pulverización o segmentación de los programas a través de los objetos de una forma más eficiente (López, 2013).

**Algoritmo.** En términos de programación, el algoritmo es el diseño del programa. En términos de la vida diaria, son todos los pasos o acciones que se llevan a cabo para resolver un problema, de esta manera se hace evidente que los algoritmos están presentes en todo lo que se hace.

Un algoritmo se define como una secuencia ordenada y cronológica de pasos que solucionan un problema o ejecutan una tarea. Es necesario que, al momento de programar, los algoritmos sean simples y precisos, tengan un orden lógico y un principio y fin (López, 2013).

Para documentar un algoritmo, los lenguajes algorítmicos más usados son los diagramas, pseudocódigo y los lenguajes de programación (Sznajdieder, 2023).

**Laravel.** Es un framework de PHP, desarrollado por Taylor Otwell y empleado para crear sitios web y aplicaciones. Es un lenguaje de Backend que permite simplificar la sintaxis de un lenguaje de programación base para que sea más fácil de usar (Educa Open, 2024).

Laravel ofrece una estructura organizada simplificando el proceso de desarrollo, a través de herramientas y funciones predeterminadas para tareas comunes. Asimismo, promueve el Modelo-Vista-Controlador, un patrón de arquitectura que hace que sea más fácil el mantenimiento, la refactorización y la programación. Dentro de las herramientas de Laravel están el Artisan, Vite, Eloquent, Sanctum, Middleware, entre otros (Educa Open, 2024).

**Vue.js.** Es un framework progresivo de JavaScript empleado para la construcción de interfaces de usuario. Está diseñado desde cero para ser utilizado incrementalmente. Su librería central está enfocada solo en la capa de visualización, y es fácil de utilizar e integrar con otras librerías o proyectos existentes. Por otro lado, Vue también es perfectamente capaz de impulsar sofisticadas Single-Page Applications cuando se utiliza en combinación con herramientas modernas y librerías de apoyo (Vue, s.f.).

**MySQL.** Es un sistema de gestión de base de datos relacional, multihilo y multiusuario. Es muy empleado en aplicaciones web como phpBB, en plataformas y por herramientas de seguimiento de errores como Bugzilla (Natsys, 2014).

La parte "SQL" de "MySQL" significa "Structured Query Language" (lenguaje de consulta estructurado). SQL es el lenguaje estandarizado más común utilizado para

acceder a las bases de datos. MySQL es rápido, confiable y fácil de utilizar. Sus principales ventajas son (Oracle, s.f.):

- Fácil de usar.
- Es una de las bases de datos más maduras y utilizadas.
- Es más rápido y menos costoso que otros servicios de datos.
- Proporciona un conjunto completo de tecnologías de replicación nativas y totalmente integradas para una alta disponibilidad y recuperación ante desastres.
- Seguridad de los datos.
- Su almacén de documentos ofrece flexibilidad para desarrollar aplicaciones de base de datos SQL tradicionales y sin esquema NoSQL.

**MariaDB.** Es un motor de base de datos relacional de código abierto. Está escrito en C y C++ y es compatible con varios lenguajes de programación y sistemas operativos. Ofrece las mismas características de MySQL y puede usarse en su lugar. Está diseñado para brindar velocidad, confiabilidad y facilidad de uso (Purestorage, s.f.).

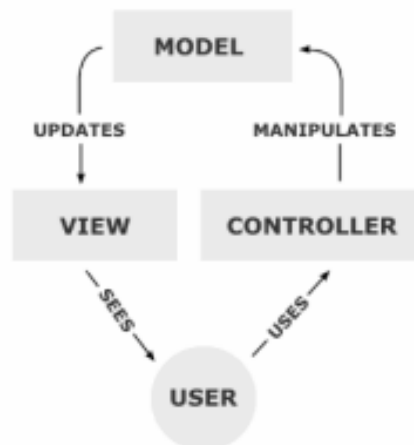
MariaDB es ideal para aplicaciones transaccionales empresariales que requieren soporte para consultas frecuentes, tiempos de respuesta rápidos y capacidad para procesar pequeñas cantidades de datos. Asimismo, es muy útil para el desarrollo de aplicaciones web y plataformas de comercio electrónico (Purestorage, s.f.).

**Arquitectura Web.** Es la planificación y el diseño de los componentes técnicos, funcionales y visuales de un sitio web, antes de que sea diseñado, desarrollado e implementado (De Souza, 2021).

Es la forma en la que se estructura el sitio web, sus contenidos, URL, enlaces, entre otros. Esta mejora la usabilidad, el rastreo y la indexación (Pulido, 2024). Hay varios tipos, pero las principales son:

- **Arquitectura web vertical:** en esta todas las urls están enlazadas desde el dominio. Se emplea especialmente para sitios especializados en temáticas muy específicas. Se denomina vertical o plana porque desde el home saldrían enlaces a todas las urls existentes y de estos enlaces hacia el home, por lo que la autoridad que se reciba se distribuirá de manera lineal en todas las páginas. No es recomendable cuando la web tenga un contenido extenso, ya que se vuelve tediosa para el usuario (Pulido, 2024).
- **Arquitectura web tipo SILO:** es la más recomendable, esta estructura tiene todos los beneficios de los que antes hemos hablado. Está bien jerarquizada, de manera que se categoriza y subcategoriza por tipo de productos o servicios nuestra web, dejando así una estructura bien definida fácil de entender (Manzano, 2021).

**Modelo – Vista – Controlador (MVC).** Es un patrón de arquitectura de software que, utilizando 3 componentes (Vistas, Models y Controladores) separa la lógica de la aplicación de la lógica de la vista en una aplicación. Es una arquitectura importante puesto que se utiliza tanto en componentes gráficos básicos hasta sistemas empresariales (Hernández, 2015). La [¡Error! No se encuentra el origen de la referencia.](#) muestra un ejemplo de este patrón.



*Ilustración 1. Ejemplo de MVC (Ovando, 2019).*

A continuación, se desglosan cada uno de los componentes (Hernández, 2015):

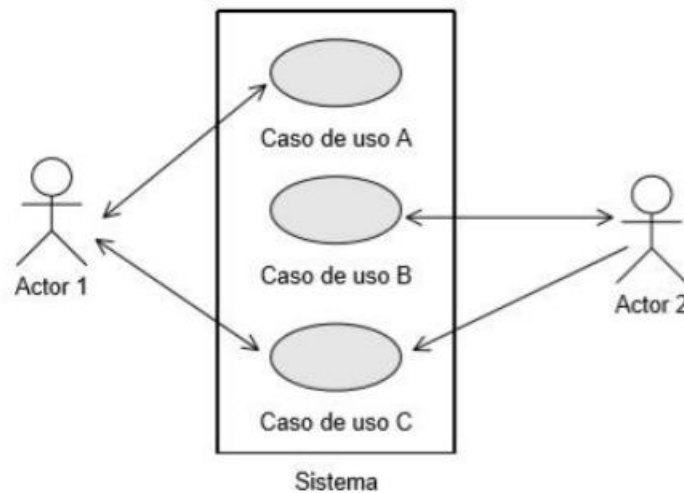
- El Modelo se encarga de los datos y la lógica.
- El Controlador recibe las órdenes del usuario y se encarga de solicitar los datos al modelo y de comunicárselos a la vista.
- Las Vistas Son la representación visual de los datos, es decir, el diseño y la presentación.

**Casos De Uso.** Un caso de uso es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema. Los diagramas de casos de uso sirven para especificar la comunicación y el comportamiento de un sistema mediante su interacción con los usuarios y/u otros sistemas (Departamento Nacional de planeación - DNP, 2020).

Los casos de uso fueron propuestos como un método para documentar las funcionalidades de un sistema existente o planeado a partir de cómo éste será usado (Departamento Nacional de planeación - DNP, 2020).

La combinación de los casos de uso y actores de un sistema forman el modelo de casos de uso el cual ayuda al cliente, a los usuarios, y a los desarrolladores a llegar a un acuerdo sobre cómo utilizar el sistema. Cada tipo de usuario del sistema se representa mediante un actor que define un rol de utilización del sistema. Los actores modelan el entorno del sistema, y los casos de uso especifican el sistema (Departamento Nacional de planeación - DNP, 2020).

**Diagrama De Casos De Uso.** Los casos de uso tienen una representación gráfica en los denominados diagramas de casos de uso. En estos diagramas, los actores se representan en forma de pequeños monigotes o stick, si el actor es un sistema se puede representar ya sea por medio de un rectángulo o una figura con el estereotipo “sistema” y los casos de uso se representan por elipses contenidas dentro de un rectángulo que representa al sistema, el nombre del caso de uso se registra al interior de la elipse. La participación de los actores en los casos de uso se indica por una flecha entre el actor y el caso de uso que apunta en la dirección en la que fluye la información y corresponde a una relación de asociación. Los diagramas de casos de uso sirven para proporcionar una visión global del conjunto de casos de uso de un sistema, así como de los actores y los casos de uso en los que éstos intervienen. Adicionalmente los diagramas de casos de uso también reflejan las relaciones entre los mismos casos de uso (Departamento Nacional de planeación - DNP, 2020). La **¡Error! No se encuentra el origen de la referencia.** muestra un ejemplo de diagrama.



*Ilustración 2. Ejemplo de diagrama de caso de uso (Departamento Nacional de planeación - DNP, 2020).*

**Automatización.** Es la aplicación de tecnología, programas, robótica, procesos o métodos que reducen los tiempos de ejecución de las tareas.

Según la IBM la automatización está cada vez más presente en todo el mundo y tiene numerosas aplicaciones desde los procesos empresariales y finanzas hasta atención médica, servicios públicos, etcétera (IBM, s.f.).

Asimismo, las empresas usan la automatización para aumentar la productividad lo que se traduce en una mayor rentabilidad, para la mejora de la prestación de los servicios y para la satisfacción del cliente, reducción de costos y errores en los procesos, cumplimiento de normas, entre otros (IBM, s.f.).

Existen varios tipos de automatización, a saber:

- Básica: automatiza tareas simples y rudimentarias. Consiste en digitalizar el trabajo mediante el uso de herramientas para agilizar y centralizar las tareas rutinarias. Ejemplo

de esto son la gestión de procesos empresariales (BPM) y la automatización de procesos robóticos (RPA) (IBM, s.f.).

- De procesos: gestiona los procesos empresariales para lograr uniformidad y transparencia. Se administra mediante software dedicado y aplicaciones empresariales. Esto puede aumentar la productividad y la eficiencia. Ejemplo de esta es la minería de procesos y la automatización de flujos de trabajo (IBM, s.f.).
- Inteligente: En esta las máquinas pueden imitar las tareas humanas y repetir las acciones una vez que los humanos definen las reglas para la máquina (IBM, s.f.).

**Logística.** Según el diccionario de logística y SCM, la logística es el arte y la ciencia de obtener, producir y distribuir materiales y productos en el lugar apropiado y en cantidades requeridas. Asimismo, según la Real Academia Española son los métodos o procesos necesarios para llevar a cabo la organización de una empresa, o de un servicio, especialmente de distribución (Bardales & Galarza, 2018).

**Automatización En La Logística.** La automatización en la logística se refiere al uso de tecnologías para realizar tareas que tradicionalmente eran ejecutadas manualmente. Según Solística (2022) la automatización en la logística no solo mejora la eficiencia operativa, sino que también reduce el riesgo de errores humanos y mejora la precisión en la gestión de inventarios y transferencias. Las tecnologías modernas, como los sistemas de gestión de transporte (TMS) y los algoritmos de optimización, permiten una integración fluida entre diferentes funciones logísticas, facilitando una mejor toma de decisiones y una mayor visibilidad en tiempo real.

**Gestión De Transferencias De Productos.** La gestión de transferencias de productos es un componente crítico en la cadena de suministro. Un estudio realizado indica que una

gestión efectiva de las transferencias puede reducir costos y tiempos de entrega, además de mejorar el servicio al cliente. La implementación de sistemas de gestión que automatizan el proceso de transferencia puede llevar a una reducción significativa en los errores asociados con el procesamiento manual. Además, la integración de algoritmos avanzados para la validación y distribución de transferencias puede optimizar la logística y mejorar la precisión en la ejecución de las órdenes (Cohpra, 2020).

**Optimización De Procesos Logísticos.** La optimización de procesos logísticos es esencial para mantener la competitividad en el mercado actual. La optimización implica la mejora continua de procesos para reducir costos, mejorar la calidad y aumentar la satisfacción del cliente. La automatización y el uso de algoritmos avanzados juegan un papel crucial en este proceso, permitiendo una gestión más eficiente y una mayor capacidad para adaptarse a cambios en la demanda y en las condiciones del mercado. La implementación de plataformas digitales para la gestión de transferencias puede ser una solución efectiva para alcanzar estos objetivos de optimización (Del Hoyo, 2017).

**Metodología SCRUM.** Scrum es un marco de trabajo ágil que organiza el desarrollo del producto en ciclos cortos llamados sprints, que suelen durar entre 2 y 4 semanas. Cada sprint produce un incremento del producto, permitiendo revisiones y ajustes continuos basados en el feedback del cliente y los resultados obtenidos. Scrum está diseñado para mejorar la colaboración y la adaptabilidad en el desarrollo de proyectos (Martins J. , ¿Quiénes son los stakeholders de un proyecto?, 2024).

#### ***Roles Scrum.***

Dado el tamaño del equipo, cada miembro desempeñará un rol clave:

- Scrum Máster: Responsable de facilitar el proceso Scrum, asegurando que el equipo siga las prácticas establecidas y eliminando obstáculos que puedan surgir. El Scrum

Máster también actúa como el principal facilitador de las reuniones Scrum y asegura que el equipo tenga un entorno productivo y libre de impedimentos (Palacio, 2024).

- **Product Owner:** Encargado de definir y priorizar los requisitos del producto en el Product Backlog. El Product Owner actúa como el principal punto de contacto entre el equipo y los stakeholders, asegurando que los requisitos reflejen las necesidades y expectativas del cliente

- **Desarrollador (Equipo de Desarrollo):** responsable de la implementación de las funcionalidades del producto durante los sprints (Palacio, 2024).

#### *Fases De La Metodología (Avila Pesantez, 2020).*

- **Análisis Situacional (Blacklog):** describe el análisis del contexto de la organización.

- **Planificación (Sprint Blacklog):** corresponde a la recopilación de información sobre el proceso y la determinación de los requerimientos.

- **Desarrollo:** comprende la elaboración de la arquitectura del sistema, elaboración de casos de uso, la creación de bases de datos.

- **Finalización:** comprende las pruebas al sistema para verificar el cumplimiento de las funcionalidades.

#### *Artefactos Scrum.*

Los artefactos principales en Scrum son:

- **Product Backlog:** Lista priorizada de requisitos y funcionalidades a desarrollar. El Product Owner gestionará el Product Backlog, actualizándolo en función de las prioridades y el feedback del cliente (Palacio, 2024).

- **Sprint Backlog:** Conjunto de tareas seleccionadas del Product Backlog que se completarán durante el sprint. El Sprint Backlog será gestionado conjuntamente por el Product Owner y tú, el desarrollador (Palacio, 2024).

- **Incremento:** Resultado del trabajo completado durante el sprint, junto con los incrementos de los sprints anteriores. Representa el estado actual del producto, el cual será revisado al final de cada sprint (Palacio, 2024).

## Planteamiento del problema

La automatización de los procesos se hace cada vez más necesaria dentro de las organizaciones a nivel nacional e internacional. No es un secreto que, además de agilizar la realización de las tareas, aumentando así la productividad, también permite la disminución de los errores, facilita la gestión de grandes cantidades de datos y el manejo de tiempos de respuesta más cortos, en un mundo que no da espera. Es así como, según la firma de análisis e investigación Gartner, se estima que para el 2026 el 30% de las empresas automatizarán más de la mitad de sus actividades de red (Iglesias Álvarez, 2024), lo cual no es algo de extrañar teniendo en cuenta el auge del desarrollo de software y las aplicaciones de la inteligencia artificial (Rootstack, 2024).

Son muchas las áreas en las organizaciones que se suman a la automatización de procesos en estos últimos tiempos, desde la gestión del talento humano para el reclutamiento de los posibles empleados, hasta la realización de pedidos por medio de sistemas web o aplicaciones que permiten que un usuario pueda realizar una compra a cualquiera hora desde cualquier parte del mundo. En Colombia, esto no es una excepción, son más las empresas que automatizan sus procesos para prestar mejores servicios y estar a la vanguardia.

El presente proyecto de grado muestra un ejemplo de todo lo anterior, de cómo una empresa dedicada a la comercialización de autopartes y accesorios para vehículos automotores que manejaba las transferencias de sus productos de forma manual, lo que generaba tiempos de respuestas altos, errores y reprocesos, vio la necesidad de automatizar el proceso de gestión de transferencias en el área de logística. Esto significa que la automatización de los subprocesos identificados como cuellos de botella dentro de

este proceso permitirá que mejoren los tiempos de respuesta y disminuyan los errores dentro del mismo. Por el contrario, de no abordarse, y conforme al crecimiento de la empresa, esto redundará en pérdidas para la compañía.

Es por todo lo anterior, que el presente proyecto de grado, cuyo alcance es el diseño e implementación de un sitio web para automatizar el proceso de la gestión de transferencias, pretendió responder el siguiente interrogante:

¿Cuánto la automatización del proceso de gestión de transferencias permite la disminución de los tiempos de respuesta y de errores en la empresa comercializadora de autopartes y accesorios para vehículos automotores?

## **Justificación**

La automatización se define como el uso de tecnología para realizar tareas automáticamente, minimizando la intervención humana y mejorando la precisión y eficiencia. Por su parte, el desarrollo de software es la creación de programas y aplicaciones informáticas para abordar necesidades específicas de una empresa (Creante Lab SAS, 2023).

El presente proyecto de grado relaciona los dos conceptos anteriores porque estos aportan grandes beneficios para las organizaciones en todos los ámbitos, convirtiéndose en elementos importantes para impulsar la eficiencia, la innovación, la productividad, el ahorro de tiempo, recursos y mejorar la satisfacción de los usuarios. Es así que con el objetivo de disminuir los tiempos de respuesta y los errores, se requirió la automatización del proceso de gestión de las transferencias de la empresa comercializadora de autopartes diseñando y desarrollando un sitio web, empleando Laravel como framework para el Backend y Vue.js para el Frontend.

El estudio de problemáticas como estas desde lo académico y empresarial permite el diseño de soluciones que van desde la teoría a la práctica, empleando tecnologías ya existentes e integrándolas con las últimas tendencias del desarrollo. De esta manera, se sientan precedentes para que pequeñas y medianas empresas en Colombia sigan el ejemplo y estén cada vez más a la vanguardia.

Por su parte, emplear Laravel y Vue.js como dos tecnologías que aunque no son novedosas se complementan y se integran muy bien, permite el desarrollo de sitios web modernos, interactivos y de alto rendimiento que sin duda serán muy aplicables a las organizaciones nacionales. Asimismo, facilitan la creación de forma rápida y eficiente, la

adaptación a las necesidades y características de cada proyecto y garantizan un alto rendimiento debido a la optimización del uso de los recursos y la velocidad de carga de la aplicación web (Capurro, 2024).

Sin dudas el desarrollo del sistema web del presente proyecto es viable y de vital importancia porque permite mejorar el proceso logístico, reduciendo los tiempos de respuesta, lo cual beneficiará la prestación del servicio de la empresa en mención.

## Objetivos

### **Objetivo General**

Automatizar el proceso de gestión de transferencias en el área de logística en una empresa comercializadora de autopartes y accesorios para vehículos automotores a través del desarrollo de un sistema web que permita disminuir los tiempos de respuesta dentro de este proceso.

### **Objetivos específicos**

- Analizar los procesos actuales de gestión de transferencias en la empresa para identificar cuellos de botella y áreas susceptibles de automatización.
- Diseñar un sitio web que facilite procesamiento y el seguimiento de transferencias de productos, proporcionando una interfaz intuitiva para los usuarios y garantizando la accesibilidad desde diversos navegadores e integrando las plataformas tecnológicas existentes en la empresa.
- Establecer un tablero de reportes y análisis dentro de la plataforma que permita a los usuarios visualizar información sobre las transferencias realizadas y sus eventos, facilitando la toma de decisiones basada en datos y la identificación de áreas de mejora.
- Realizar pruebas funcionales y de usabilidad para validar el desempeño del sistema web, asegurando que las transferencias se ejecuten de manera eficiente y sin errores.

## Metodología

### **Tipo De Estudio**

El presente proyecto de grado se enmarca en un estudio descriptivo, puesto que se centra en el diseño, desarrollo y evaluación de un sistema web para mejorar el proceso de transferencias de productos en una empresa de autopartes, lo cual implica describir el problema existente (errores y tiempos elevados en el proceso manual) y la solución desarrollada para optimizar dicho proceso.

### **Diseño metodológico**

Para el desarrollo del sistema web para la empresa dedicada a la comercialización de autopartes, se utilizó la metodología Scrum, un marco ágil que se adapta bien a equipos pequeños y permite una entrega iterativa y continua del producto. Dado que el equipo está compuesto por tres participantes, cada uno asumirá un rol específico, a saber:

- Scrum Máster: lo asume el líder técnico del área de sistemas.
- Product Owner: Líder administrativo general de la organización.
- Desarrollador (Equipo de Desarrollo): Este Rol lo asume el desarrollador asignado al proyecto.

A continuación, se describen cada una de las fases en las que se desarrolló el presente proyecto de grado:

#### **Fase 1: Análisis Situacional (Blacklog)**

Se partió de un análisis del contexto del área de logística de la empresa empleando la herramienta DOFA identificando sus necesidades puntuales. Asimismo, se realizó una revisión bibliográfica de las principales teorías y conceptos que soportan el desarrollo del sistema web del presente proyecto.

### Análisis Situacional

Para el análisis del contexto del área de logística de la organización se identificaron las Debilidades, Oportunidades, Fortalezas y amenazas (DOFA) en el proceso de gestión de transferencias, el cual se muestra en La *Tabla 1*.

*Tabla 1.* Análisis DOFA del área de logística.

<b>Debilidades</b>	<b>Oportunidades</b>
<p><b>Visibilidad Limitada del Flujo de Transferencias</b></p> <p>Existe una falta de visibilidad en tiempo real sobre el estado de las transferencias, sumado a solicitudes manuales a las bodegas lo que puede dificultar la toma de decisiones rápidas y la resolución de problemas logísticos.</p> <p><b>Falta de Automatización</b></p> <p>Los procesos logísticos actuales dependen en gran medida de la intervención manual, lo que aumenta el riesgo de errores humanos y reduce la eficiencia operativa.</p> <p><b>Sistemas Desconectados</b></p> <p>La logística utiliza herramientas de gestión que no están completamente integradas, lo que genera retrasos en la</p>	<p><b>Transformación Digital</b></p> <p>La implementación de una plataforma de gestión de transferencias basada en tecnología avanzada, como la que se está desarrollando, permitirá a La organización optimizar sus operaciones logísticas y reducir errores manuales.</p> <p><b>Reducción de costos y tiempo</b></p> <p>Al implementar mejoras en la toma de decisiones respecto al tiempo y distancia de los fletes, se mejora el cumplimiento de las entregas, y se obtienen menores costos de envío.</p> <p><b>Automatización de Procesos Logísticos</b></p> <p>La automatización de tareas como la asignación de rutas y la validación de transferencias mejorará la precisión y</p>

<p>obtención de información y afecta la fluidez del proceso.</p> <p><b>Procesos Lentos de Validación</b></p> <p>La validación de las transferencias de productos suele ser un proceso lento debido a la necesidad de revisar manualmente cada una, lo que puede generar cuellos de botella en la operación.</p>	<p>reducirá los tiempos de procesamiento, aumentando la productividad del área de logística.</p>
<p><b>Fortalezas</b></p>	<p><b>Amenazas</b></p>
<p><b>Relaciones sólidas con clientes y proveedores</b></p> <p>La organización mantiene una buena relación con sus proveedores y lealtad de muchos de sus clientes.</p> <p><b>Infraestructura logística bien establecida</b></p> <p>La organización está establecido a nivel nacional e internacional y por lo tanto tiene bodegas en capitales y ciudades estratégicas del país para poder dar una amplia cobertura a todos sus clientes de forma óptima.</p> <p><b>Adaptación al cambio</b></p>	<p><b>Cambios regulatorios</b></p> <p>Al ser la organización una empresa que no solo envía productos, sino que también recibe de sus proveedores en el exterior, estos cambios pueden impactar no solo en los costos sino en la disponibilidad de los productos.</p> <p><b>Errores humanos en los procesos</b></p> <p>Un error humano de percepción puede llevar un producto a un cliente incorrecto y La organización no está exento de dichos errores.</p> <p><b>Retrasos en las entregas por parte de las transportadoras</b></p>

<p>La organización ha demostrado un gran capacidad para adaptarse a los cambios en el mercado, la transformación digital y al crecimiento de la competencia, mejorando sus procesos, optimizando ruta y actualizándose constantemente.</p>	<p>Los retrasos o pérdidas de mercancía por parte de la transportadora que lleva los pedidos pueden ser un riesgo latente.</p>
--	--

Nota. Esta tabla presenta un análisis DOFA del área de logística. Identifica debilidades como la visibilidad limitada del flujo de transferencias y la falta de automatización. Las oportunidades incluyen la implementación de plataformas digitales para optimizar procesos. Las fortalezas abarcan relaciones sólidas con clientes y una infraestructura bien establecida, mientras que las amenazas se relacionan con cambios regulatorios y posibles errores humanos. Este análisis permite enfocar esfuerzos en mejorar la eficiencia y reducir riesgos en logística.

### **Procesos internos del negocio**

Se identificó solo un proceso clave en la gestión de logística en la organización:  
Gestión de transferencias.

### **Casos de uso del negocio**

Para el proceso de gestión de transferencias se desarrollaron los diagramas de caso de uso mostrados en la **¡Error! No se encuentra el origen de la referencia.**, **¡Error! No se encuentra el origen de la referencia.** y la **¡Error! No se encuentra el origen de la referencia.**.

### **Caso: Procesamiento manual de una transferencia.**

**Código:** CU-1

**Actores:** Asistente logístico (Usuario), Bodegas.

**Descripción:** Definir la bodega de origen y las cantidades de productos para la salida del pedido.

**Condiciones previas:** Los datos del cliente deben haber sido comprobados, así como las existencias de bodega.

**Flujo básico:**

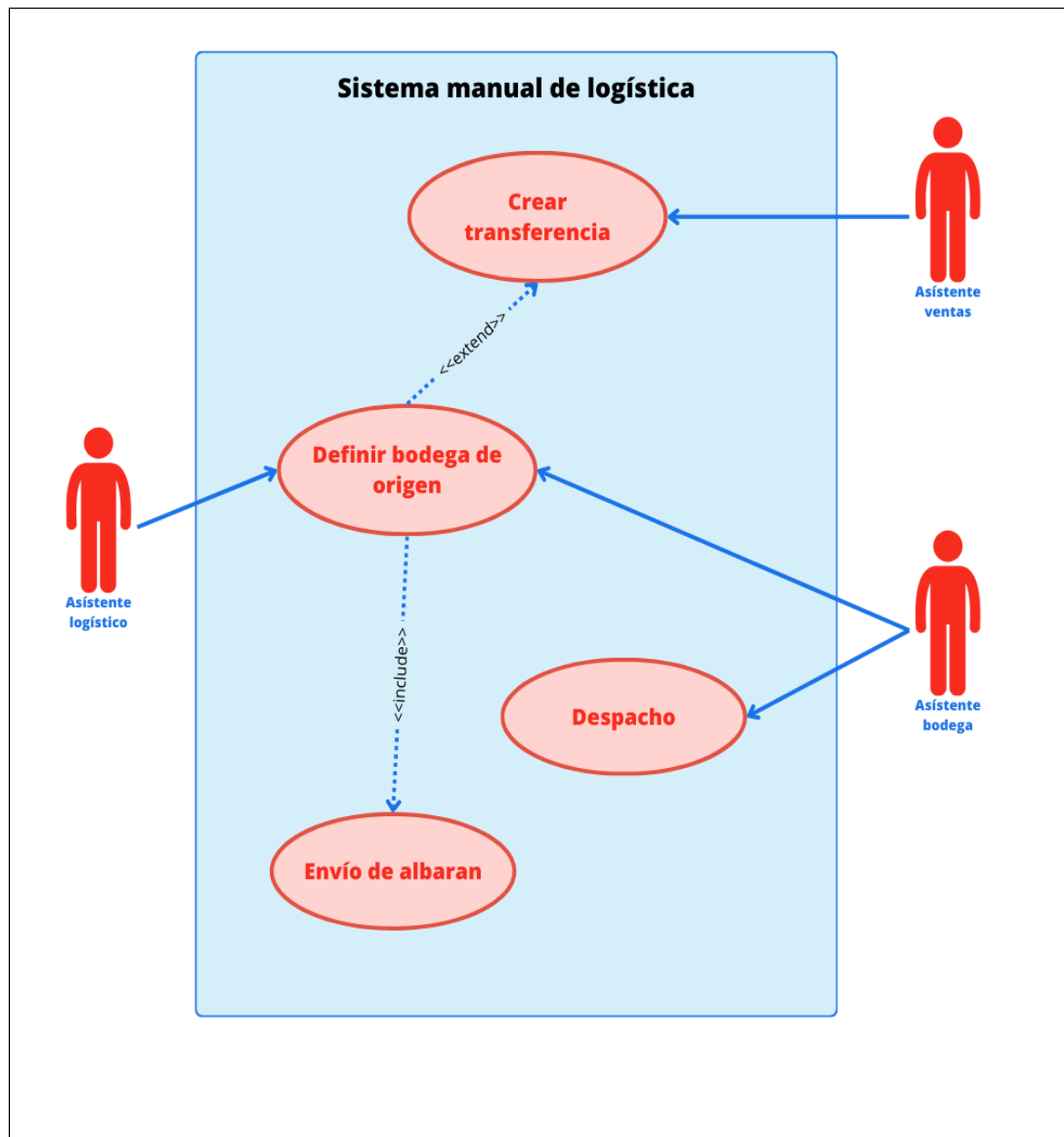
1. Validar cuales bodegas tienen stock del producto.
2. Realizar consultas con la transportadora para ver desde que bodega es más económico el flete.
3. Crear un albarán con las especificaciones del destinatario, productos y cantidades.
4. Enviar correo a la bodega seleccionada con el Albarán.

**Flujos alternativos:**

1. Si el stock no está disponible por completo en una bodega, crea transferencias para las siguientes hasta completar el pedido.
2. Si no hay stock disponible o falta algún dato del cliente, notificar al vendedor.

**Dependencias:** Ninguna.

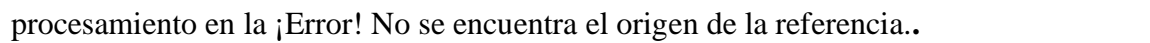
**Diagramas:**

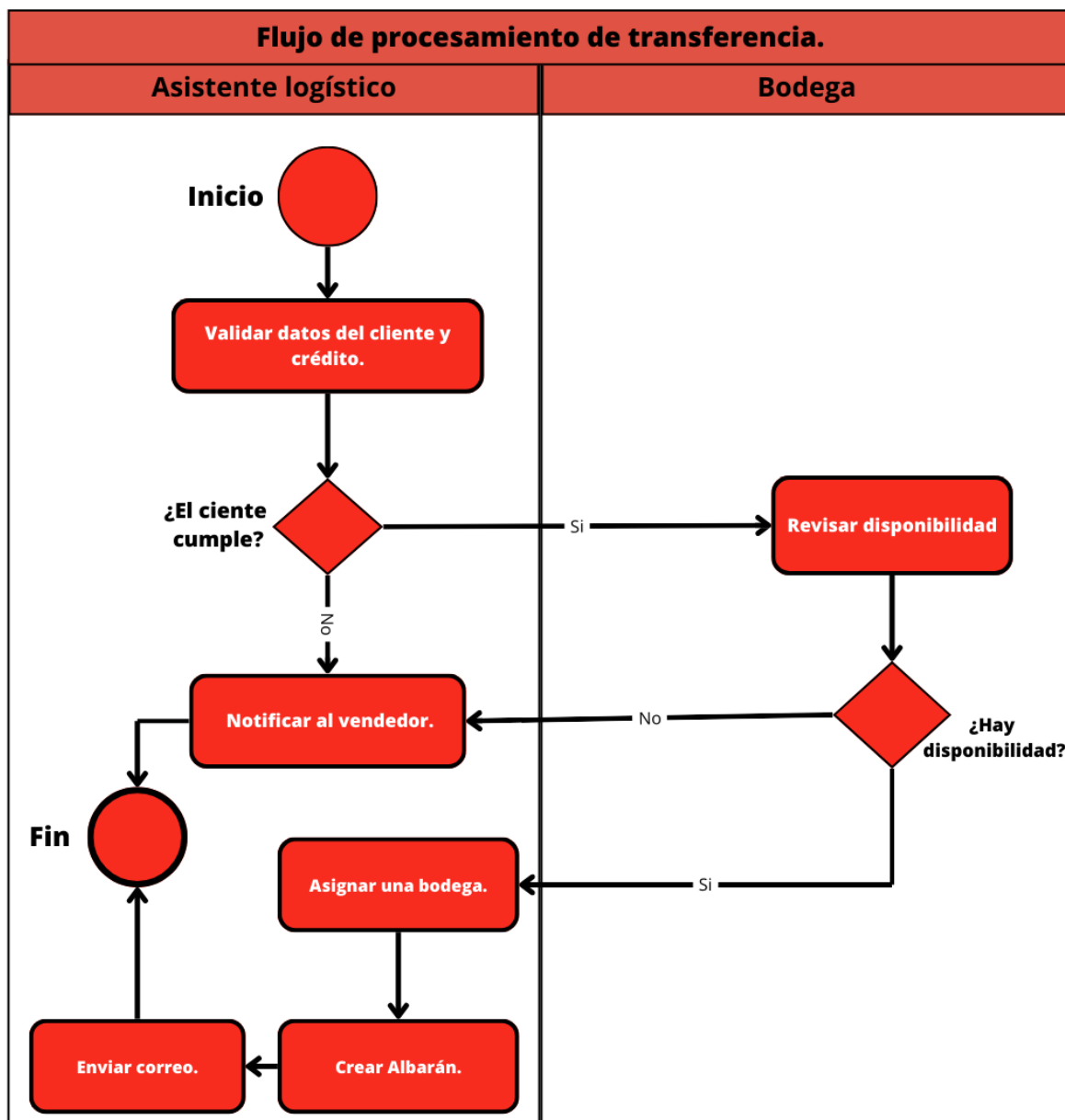


*Ilustración 3.* Sistema manual de logística.

Nota. Este diagrama muestra un caso de uso del sistema manual de logística, en el cual se representan las principales funciones y roles involucrados en el proceso. Los actores, como el asistente de ventas, asistente de bodega, y asistente logístico, interactúan con distintas funcionalidades del sistema, como la creación de transferencias, definición de

bodegas de origen, despacho, actualización de datos de transferencias, envío de albaranes, autenticación, y visualización del tablero de transferencias. Las relaciones de <<include>> y <<extend>> indican dependencias y extensiones de funcionalidades entre los casos de uso principales.

El flujo de procesamiento de las transferencias es realizado por asistente logístico y su interacción y acciones de para los casos de uso suelen tomar tiempos determinados, lo que a veces causa cuellos de botella en el área logística, podemos ver el flujo de dicho procesamiento en la  ¡Error! No se encuentra el origen de la referencia..



*Ilustración 4.* Diagrama de flujo de Procesamiento manual de transferencias.

*Nota.* Este diagrama muestra el flujo del procesamiento manual de una transferencia. Al sumar los tiempos promedio de las actividades realizadas por el asistente de logística para procesar una transferencia manualmente, dependiendo del caso, este podía demorar entre 8 y 37 minutos aproximadamente por cada transferencia.

A partir de la **¡Error! No se encuentra el origen de la referencia.** se muestra el promedio de los tiempos que toman los diferentes casos de uso para la gestión de transferencias en la **Tabla 2.**

*Tabla 2. Tiempos promedio de gestión de transferencia.*

<b>Caso</b>	<b>Tiempo promedio total</b>
El cliente no cumple con el límite de crédito o los datos son inválidos.	8 minutos promedio.
El cliente cumple, pero no hay suficiente Stock del producto en las bodegas.	18 minutos promedio.
La transferencia pudo ser procesada.	37 minutos promedio.

*Nota.* Las actividades del usuario por cada acción del proceso añaden tiempo dependiendo el caso.

#### **Historias de usuario:**

HU-1	Automatización de transferencias.	Como asistente de logística, deseo poder reducir el tiempo que me toma procesar las transferencias para optimizar mis actividades.
Criterios de aceptación:		
<ol style="list-style-type: none"> <li>1. El sistema debe asignar una o varias bodegas de salida para los productos de la transferencia.</li> <li>2. El sistema de poder detener y generar un evento según el error correspondiente.</li> <li>3. El sistema debe correr el procesamiento de transferencias cada 10 minutos desde las 7:00am hasta las 6:00pm de lunes a viernes y sábados de 7:00am a 12:00pm.</li> </ol>		
Casos de uso	CU-1	

Tareas	PFR-1, PFR-2, PFR-3, PFR-4, PFR-5, PFR-6, PFR-7, PFR-8
--------	--

HU-2	Envío de albarán por email automático.	Como asistente de logística, deseo poder automatizar la creación y envío de los albaranes de las transferencias para optimizar mis actividades y evitar errores humanos.
Criterios de aceptación:		
<ol style="list-style-type: none"> <li>1. Al procesar una transferencia el sistema debe generar un albarán de entrega en PDF con los datos de la transferencia.</li> <li>2. Los datos deben coincidir con los productos, cantidades y direcciones especificados.</li> <li>3. El sistema debe enviar este albarán por correo a la bodega de origen asignada a cada transferencia.</li> </ol>		
Casos de uso	CU-1	
Tareas	PFR-10	

**Caso: Seguimiento de una transferencia.**

**Código:** CU-1

**Actores:** Asistente logístico (Usuario), Bodegas.

**Descripción:** Solicitar información para el seguimiento de las salidas de las transferencias hasta su destino.

**Condiciones previas:** Las transferencias deben haber sido procesadas correctamente y los albaranes enviados a las bodegas.

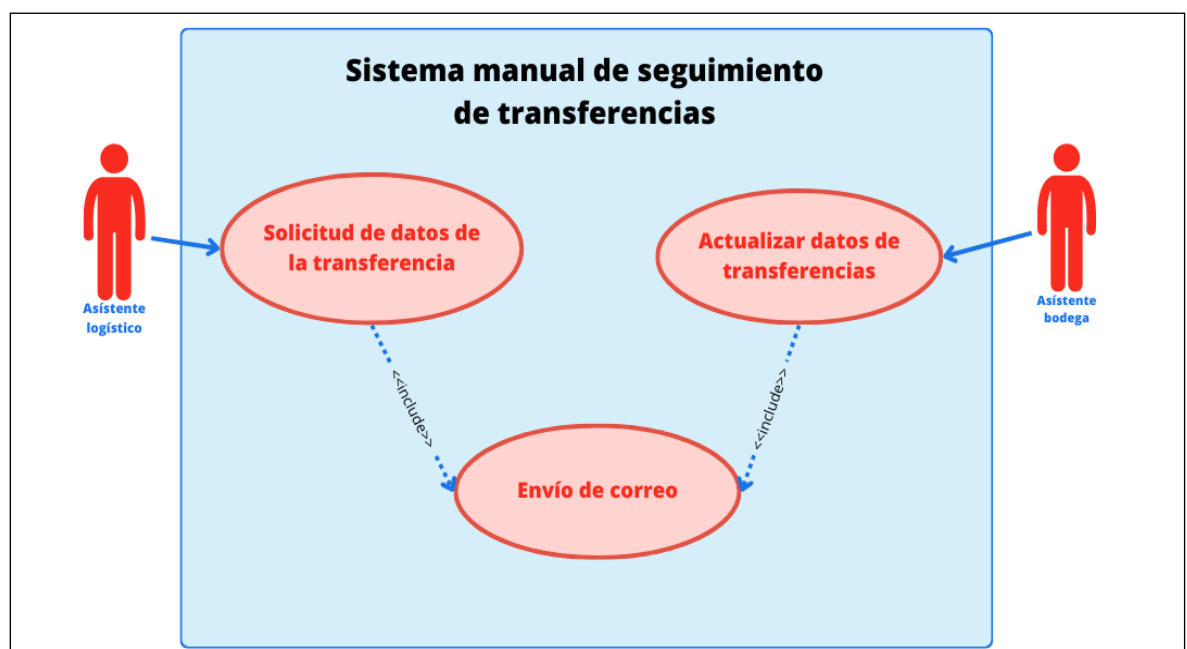
**Flujo básico:**

1. Solicitar información sobre la fecha de empaquetado y fecha de despacho del producto.
2. Solicitar información de las guías para el seguimiento con las transportadoras.

**Flujos alternativos:** Ninguno.

**Dependencias:** Ninguna.

**Diagramas:**



*Ilustración 5. Diagrama Solicitud de seguimiento de salidas a bodegas.*

*Nota.* Este diagrama muestra el proceso de seguimiento por parte del asistente de logística a una transferencia. Este debe solicitar la información a las bodegas o transportadoras, en cuyo caso los tiempos de espera para obtener dicha información o en sistemas externos toman entre 5 y 15 minutos.

**Historias de usuario:**

HU-3	Sitio web de logística.	Como asistente de logística, deseo poder ingresar a un espacio donde pueda hacer seguimiento de las salidas de las transferencias sin tener que hacer solicitudes a las bodegas por información.
Criterios de aceptación:		
<ol style="list-style-type: none"> <li>1. El sistema debe permitir al usuario autenticarse e ingresar al sitio web con su correo electrónico y contraseña.</li> <li>2. El sistema debe contener una interfaz dinámica y amigable con el usuario.</li> <li>3. El sistema debe cargar y funcionar correctamente en los distintos navegadores web.</li> </ol>		
Casos de uso	CU-2	
Tareas	PFR-9, PFR-11, PFR-12	
HU-4	Formulario de edición de salida.	Como asistente de logística, deseo poder ingresar a un espacio donde pueda hacer seguimiento de las salidas de las transferencias sin tener que hacer solicitudes a las bodegas por información.
Criterios de aceptación:		
<ol style="list-style-type: none"> <li>1. El sistema debe permitir al usuario autenticarse e ingresar al sitio web con su correo electrónico y contraseña.</li> <li>2. El sistema debe contener una interfaz dinámica y amigable con el usuario.</li> </ol>		

3. El sistema debe cargar y funcionar correctamente en los distintos navegadores web.	
Casos de uso	CU-2
Tareas	PFR-13, PFR-14

Los tiempos mostrados anteriormente, confirmaron la necesidad de desarrollar el sistema web que automatizara la gestión de las transferencias. Por su parte, una vez establecidos los tiempos del proceso, se procedió a identificar las reglas del negocio con los roles directamente relacionados, las cuales se muestran en la **Tabla 3**.

*Tabla 3.* Reglas del negocio.

<b>Actores</b>	<b>Asistente de venta</b>	<b>Asistente de logística</b>
<b>Descripción</b>	Al hacer una venta debe llegar al área de logística para hacer las transferencias desde la bodega óptimas hacia el cliente.	Al ser creada una orden de venta el sistema de ventas crea una transferencia para que sea procesada por el área de logística.
<b>Actividades</b>	Crear la orden de venta con los datos y dirección del cliente y especificaciones para la transferencia	Validar la disponibilidad del producto en las bodegas.  Asignar la entrega de los productos, y notificar a la bodega de la salida, a que transportadora o mensajero será entregado el producto.  Hacer seguimiento del envío hasta que llegue al cliente.
<b>Reglas</b>	Debe Crear la orden de venta por la plataforma de ventas. Debe completar toda la información en caso de que el cliente pague el flete o lo incluya en la compra.	Solo se va a despachar productos si hay suficiente stock.  Si ninguna bodega tiene el stock suficiente para entregar todo el pedido, se debe crear una nueva transferencia para el producto restante

		<p>desde la siguiente bodega más cercana.</p> <p>Si la dirección de entrega está dentro del área de cobertura de los mensajeros de la bodega, debe notificar a la bodega para que no se pague flete.</p>
--	--	--

*Nota.* Las restricciones direccionan la toma de decisiones dentro del área de logística y sus procesos.

Por su parte, la Tabla 4 muestra la influencia de los Stakeholder en el proceso, esto con el fin de determinar sus niveles de influencia.

*Tabla 4.* Stakeholders.

<b>Stakeholder</b>	<b>Rol / Interés Principal</b>	<b>Nivel de Influencia</b>	<b>Participación Directa en el Proyecto</b>
Product Owner	Definir prioridades y backlog	Alta	Alta
Equipo de Logística	Usuario objetivo de la solución	Alta	Media
Clientes	Recepción y experiencia del cliente	Media	Baja
Equipo de desarrollo	Implementación técnica	Alta	Alta
Scrum Master	Facilitar el proceso de Scrum	Alta	Alta
Equipo de ventas	Proveer información	Alta	Baja

*Nota.* Estos son todos los interesados de un modo u otro en el proyecto.

### **Identificación De Requerimientos**

A partir de toda la información obtenida, se obtuvieron los requerimientos funcionales y no funcionales para el diseño y desarrollo del sistema web, los cuales se muestran en las Tabla 5 y la **¡Error! No se encuentra el origen de la referencia.** :

Tabla 5. Requisitos funcionales.

<b>Código</b>	<b>Descripción</b>	<b>Prioridad</b>	<b>Casos de uso</b>
RF-1	El sistema debe permitir la gestión automática de transferencias de inventario, procesando las transferencias creadas cada 10 minutos.	Alta	CU-1
RF-2	El sistema debe contar con un módulo que permita a los usuarios autorizados realizar el seguimiento en tiempo real de las transferencias y sus salidas.	Alta	CU-2
RF-3	El sistema debe enviar automáticamente correos electrónicos a las bodegas con los Albaranes de las entregas en formato PDF, una vez que se haya procesado la transferencia.	Alta	CU-1
RF-4	El sistema debe ser accesible desde cualquier navegador web, garantizando la compatibilidad con los principales navegadores en el mercado.	Alta	CU-2
RF-5	El sistema debe integrar y acceder a la base de datos general de la organización, asegurando la consistencia y disponibilidad de la información de inventario.	Alta	CU-1
RF-6	El sistema debe permitir la asignación automática de transferencias a la bodega más cercana a la dirección del cliente, optimizando la logística de entrega.	Alta	CU-1
RF-7	El sistema debe permitir a los usuarios autorizados agregar manualmente datos adicionales al seguimiento de las transferencias, facilitando la actualización de información.	Alta	CU-2

*Nota.* Estos son los requisitos que definen el funcionamiento y los componentes que debe tener el sistema.

Tabla 6. Requisitos no funcionales.

<b>Código</b>	<b>Descripción</b>	<b>Prioridad</b>
RNF-1	El sistema debe estar disponible el 99,9% del tiempo, garantizando alta disponibilidad.	Alta

RNF-2	El sistema debe ser escalable para soportar un aumento de hasta un 50% en la cantidad de transferencias sin degradar su rendimiento.	Media
RNF-3	La interfaz de usuario debe ser intuitiva y fácil de usar, con un nivel de aprendizaje bajo para los usuarios.	Alta
RNF-4	La plataforma debe integrarse con sistemas externos mediante APIs estándar (REST/JSON).	Alta
RNF-5	La plataforma debe soportar múltiples idiomas, permitiendo futuras expansiones internacionales.	Media

*Nota.* Estos requisitos representan el atributo de Calidad del sistema.

De igual manera, se identificaron las restricciones para el desarrollo del sistema web teniendo en cuenta las condiciones de la empresa. Estos se muestran en la **¡Error! No se encuentra el origen de la referencia.**, mostrada a continuación:

*Tabla 7. Restricciones.*

<b>Código</b>	<b>Descripción</b>	<b>Prioridad</b>
R-1	El sistema debe ser compatible con los principales navegadores web, incluyendo Chrome, Firefox, Safari y Edge.	Alta
R-2	Solo un desarrollador estará encargado de la implementación técnica del sistema.	Alta
R-3	El cliente requiere que la planificación y revisión del sprint se realicen juntas, limitando los tiempos de reuniones.	Media
R-4	El sistema utilizará <b>Laravel 10</b> para el Backend.	Alta
R-5	El <b>Frontend</b> estará desarrollado en <b>Vue.js 3</b> para garantizar una interfaz dinámica.	Alta
R-6	La base de datos será <b>MariaDB</b> versión 10.6, compatible con el sistema para gestionar la persistencia de datos.	Alta
R-7	El lenguaje de programación principal para el Backend será <b>PHP 8</b> .	Alta
R-8	Se empleará <b>Laragon</b> como entorno de desarrollo principal para facilitar la codificación y el manejo de extensiones útiles.	Media

R-9	<b>Node.js</b> y <b>npm</b> se utilizarán para la gestión de dependencias del proyecto y compilación de recursos FrontEnd.	Media
R-10	Se debe realizar un despliegue compatible con <b>SCRUM</b> , empleando herramientas que faciliten la entrega continua, como Git	Alta
R-11	Se empleará <b>Visual Studio Code</b> como entorno de desarrollo principal para facilitar la codificación y el manejo de extensiones útiles.	Baja

*Nota.* Estos son Requerimientos u obstáculos que pueden dificultar la salida del proyecto.

### Product Backlog

A partir de todo lo anterior se tiene el siguiente Product Backlog donde se relacionan todas las Historias de usuario (HU) con los casos de uso y los (PFR) Project Feature Requirement Necesarios para Cumplir con dicha historia:

*Tabla 8. Product Backlog.*

ID de HU	Historia de Usuario (HU)	Casos de uso (CU)	Tareas (PFR)	Prioridad	Estado
HU-1	Automatización de transferencias.	CU-1	PFR-1, PFR-2, PFR-3, PFR-4, PFR-5, PFR-6, PFR-7, PFR-8	Alta	Hecho
HU-2	Envío de albarán por email automático.	CU-1	PFR-10	Alta	Hecho

HU-3	Sitio web de logística.	CU-2	PFR-9, PFR-11, PFR-12	Alta	Hecho
HU-4	Formulario de edición de salida.	CU-2	PFR-13, PFR-14	Alta	Hecho

### **Fase 2: Planificación (Sprint Backlog)**

En esta fase se definió la estructura del proyecto a través de los sprints, cada uno con tareas y entregables específicos, estableciendo reuniones diarias para seguimiento y revisiones con el Product Owner, incluyendo la planificación y retrospectiva al final de cada sprint. Además, se asignaron las tareas técnicas con prioridades claras, determinándose un cronograma de trabajo detallado para cumplir con los objetivos en tiempos específicos. Este enfoque permitió organizar y distribuir el trabajo de manera eficiente, asegurando que cada fase del proyecto tuviera un propósito y entregables definidos.

En esta fase se llevaron a cabo reuniones con el cliente y de acuerdo con el análisis previo y las negociaciones, se definieron 8 sprints y las tareas relacionadas a cada uno de ellos. Al ser sólo un desarrollador, un Scrum Máster y un Product Owner se definieron reuniones diarias a las 10:30am, planificaciones y retrospectivas. La review se hacía junto con el planning por decisión del cliente o dueño del producto (Product Owner). La Tabla 9 muestra el cronograma establecido para el diseño y desarrollo del sistema web según cada iteración de sprint.

Tabla 9. Sprint Backlog.

<b>Sprint</b>	<b>Actividad</b>	<b>Descripción</b>	<b>Entregable</b>	<b>Tareas</b>	<b>Duración</b>
1	Planificación inicial	Definición de objetivos, alcance y metodología del proyecto.	Documento de Planificación	PFR-1, PFR-2, PFR-3	10 puntos
2	Investigación preliminar	Investigación de Conceptos, Extracción de información.	Informe de Investigación Preliminar	PFR-4, PFR-5, PFR-6, PFR-7	14 puntos
3	Diseño Backend del proyecto	Definición de diseño y estructura del proyecto y base de datos.	Documento de Diseño Back del Proyecto	PFR-8, PFR-9, PFR-10	14 puntos
4	Diseño Frontend del proyecto	Definición de diseño de interfaz de usuario del proyecto.	Documento de Diseño Front del Proyecto	PFR-11, PFR-12, PFR-13	14 puntos
5	Implementación inicial	Comienzo de la implementación del proyecto.	Informe de Avance de Implementación	PFR-14, PFR-15, PFR-16	14 puntos
6	Revisión y ajuste de implementación	Evaluación de los resultados iniciales y ajustes necesarios.	Informe de Revisión y Ajuste	PFR-17, PFR-18, PFR-19	28 puntos
7	Preparación del documento final	Elaboración de los resultados, discusión y conclusiones.	Documento Final	PFR-20, PFR-21	10 puntos
8	Entrega final	Entrega de proyecto final con ajustes.	Entrega del Proyecto	PFR-22, PFR-23	10 puntos

*Nota.* Esta tabla muestra el cronograma de actividades obtenido en el planning de cada iteración.

Además, se desarrollaron los *Project Feature Requirements* (PFR), detallados en la

Tabla

Tabla 10, que abarcan todas las tareas ejecutadas durante el desarrollo del proyecto. Estos requisitos incluyen desde tareas iniciales, como la creación del cronograma y la

preparación del documento de proyecto, hasta historias técnicas específicas relacionadas con la implementación de funcionalidades críticas, pruebas de integración y ambiente de pruebas, así como la preparación para la salida a producción. Cada *PFR* fue planificado y priorizado según su impacto y contribución a los objetivos del proyecto, permitiendo una gestión estructurada y eficiente de los recursos disponibles para el desarrollo del sistema web.

*Tabla 10.* Project feature requirements.

<b>Código</b>	<b>Unidad</b>	<b>Título</b>	<b>Descripción</b>	<b>Puntos</b>	<b>Prioridad</b>	<b>HU</b>
PFR-1	Tarea	Definición de Cronograma y entregables.	Crear cronograma de acuerdo con la metodología Scrum.	2	Alta	HU-1
PFR-2	Tarea	Preparación inicial de documento de proyecto.	Introducción, marco teórico, Planteamiento del problema, Justificación, Objetivos General y específico, Metodología.	4	Alta	HU-1
PFR-3	Spike	Spike de validación de lógica	Investigar y validar la Factibilidad de un algoritmo para la separación y designación de transferencias por bodega.	4	Alta	HU-1
PFR-4	Historia técnica	Montaje inicial de proyecto.	Iniciar desarrollo del proyecto Backend según las especificaciones.	2	Alta	HU-1

PFR-5	Historia técnica	Extracción inicial.	Diseño de base de datos y Extracción de modelos iniciales desde la integración.	2	Alta	HU-1
PFR-6	Historia técnica	Validar Modelo Orden de Venta	Validar que la información de la orden de venta es la correcta: - Tipo de flete, Transportadora, Datos del cliente.	5	Alta	HU-1
PFR-7	Historia técnica	Validar Modelo Transferencias.	Validar que la información de la Transferencia es correcta: - Dirección de entrega, Productos, Cantidades.	5	Alta	HU-1
PFR-8	Historia técnica	Procesamiento de Transferencias.	Desarrollar algoritmo de procesamiento de transferencias según las restricciones y modelos lógicos definidos.	5	Alta	HU-1
PFR-9	Historia técnica	Crear Endpoint de Autenticación y listado de transferencias.	Crear un EndPoint para el inicio y cierre de sesión de los usuarios al sistema de logística. Crear un EndPoint de muestra de las transferencias y sus estados y comentarios.	5	Alta	HU-3
PFR-10	Historia de técnica	Aplicar lógica de envío de Emails y PDF Adjunto.	Crear Servicio de creación de PDF con los datos de la transferencia para su respectivo envío por Email.	4	Alta	HU-2

PFR-11	Historia técnica	Creación de Formulario de Autenticación y Layout.	Crear formulario de autenticación con sus validaciones y Layout de entrada con opción de Cerrar sesión.	5	Alta	HU-3
PFR-12	Historia técnica	Creación de Tablero de logística.	Creación de tablero que muestre los datos de las transferencias, estado, notas. Consumiendo las APIs del Backend.	5	Alta	HU-3
PFR-13	Historia técnica	Formulario de creación de notas y eventos de la transferencia.	Creación de formulario de notas, y formulario de datos de salida con los respectivos eventos.	4	Alta	HU-4
PFR-14	Historia técnica	Actualización de datos de la salida.	Crear Endpoint de Actualización de datos de la salida	4	Alta	HU-4
PFR-15	Historia técnica	Ambiente de pruebas para Back	Actualizaciones de documentos y escalada de proyecto a ambientes de pruebas de Backend.	5	Alta	
PFR-16	Historia técnica	Ambiente de pruebas Front	Actualizaciones de documentos y escalada de proyecto a ambientes de pruebas de Frontend.	5	Alta	
PFR-17	Historia técnica	Pruebas de Funcionamiento Back	Prueba de funcionamiento de lógica procesamiento de datos y envío de Email.	10	Alta	

PFR-18	Historia técnica	Pruebas de funcionamiento Front	Prueba de funcionamiento en el tablero de transferencias, estados y formulario de creación.	10	Alta	
PFR-19	Tarea	Presentación de informe.	Informe de novedades y ajustes de las pruebas realizadas.	8	Alta	
PFR-20	Tarea	Actualización de documento.	Agrega estado del cronograma, informes de pruebas, discusión, resultados y sugerencias del líder técnico.	5	Alta	
PFR-21	Tarea	Validación de pruebas integrales y de extremo a extremo.	Validar correcciones de las pruebas y avances del documento, y establecer las conclusiones para entrega final.	5	Alta	
PFR-22	Historia técnica	Preparación para salida a producción	Validación de avances finales del documento y código.	5	Alta	
PFR-23	Tarea	Entrega final del proyecto	Documento final del proyecto.	5	Alta	

*Nota.* En esta tabla podemos ver todas las tareas ejecutadas en el desarrollo del proyecto.

### **Fase 3: Desarrollo**

En esta fase se ejecutaron las tareas programadas según su prioridad y orden, comenzando con las investigaciones y posteriormente la ejecución. Se desarrolló el mapa de procesos para automatizar la asignación de transferencias a las bodegas. Y se implementó tanto el Frontend usando Vue.js como el Backend con Laravel, integrando MariaDB como base de datos para poder crear un comando que procesara las

transferencias. Asimismo, se creó un tablero para poder hacer seguimiento según las especificaciones y necesidades planteadas. Por último, se ejecutaron pruebas para garantizar el funcionamiento, optimización y rendimiento del sistema.

En el siguiente Diagrama de Gantt toda la ejecución de las actividades mostrando una visión clara del avance en las ilustraciones **¡Error! No se encuentra el origen de la referencia., ¡Error! No se encuentra el origen de la referencia. e ¡Error! No se encuentra el origen de la referencia..**

Nombre de la tarea	Fecha de inicio	Fecha de finalización	Asignado	Estado	01.01.2024	02.01.2024	03.01.2024	04.01.2024	05.01.2024	06.01.2024	07.01.2024	08.01.2024	09.01.2024	10.01.2024	11.01.2024	12.01.2024	13.01.2024	14.01.2024	15.01.2024	16.01.2024	17.01.2024	
<b>PROYECTO WEB</b>				Terminado																		
PFR-1	02.01.2024	02.01.2024	Miguel	Terminado		█																
PFR-2	03.01.2024	04.01.2024	Miguel	Terminado			█	█														
PFR-3	05.01.2024	06.01.2024	Miguel	Terminado					█	█												
PFR-4	07.01.2024	07.01.2024	Miguel	Terminado							█											
PFR-5	08.01.2024	08.01.2024	Miguel	Terminado								█										
PFR-6	09.01.2024	11.01.2024	Miguel	Terminado									█	█								
PFR-7	11.01.2024	13.01.2024	Miguel	Terminado											█	█						
PFR-8	13.01.2024	15.01.2024	Miguel	Terminado													█	█				
PFR-9	15.01.2024	17.01.2024	Miguel	Terminado															█	█		

Ilustración 6. Diagrama Gantt Tareas PFR-1 - PFR-9.

Nombre de la tarea	Fecha de inicio	Fecha de finalización	Asignado	Estado	18.01.2024	19.01.2024	20.01.2024	21.01.2024	22.01.2024	23.01.2024	24.01.2024	25.01.2024	26.01.2024	27.01.2024	28.01.2024	29.01.2024	30.01.2024	31.01.2024	01.02.2024	02.02.2024	
<b>PROYECTO WEB</b>				Terminado																	
PFR-10	18.01.2024	19.01.2024	Miguel	Terminado	█	█															
PFR-11	20.01.2024	22.01.2024	Miguel	Terminado			█	█													
PFR-12	22.01.2024	24.01.2024	Miguel	Terminado					█	█											
PFR-13	25.01.2024	26.01.2024	Miguel	Terminado								█	█								
PFR-14	27.01.2024	28.01.2024	Miguel	Terminado										█	█						
PFR-15	29.01.2024	31.01.2024	Miguel	Terminado												█	█				
PFR-16	31.01.2024	02.02.2024	Miguel	Terminado														█	█		

Ilustración 7. Diagrama Gantt Tareas PFR-10 - PFR-16.

Nombre de la tarea	Fecha de inicio	Fecha de finalización	Asignado	Estado	03.02.2024	04.02.2024	05.02.2024	06.02.2024	07.02.2024	08.02.2024	09.02.2024	10.02.2024	11.02.2024	12.02.2024	13.02.2024	14.02.2024	15.02.2024	16.02.2024	17.02.2024	18.02.2024	19.02.2024	20.02.2024	21.02.2024	22.02.2024	23.02.2024	24.02.2024	25.02.2024	26.02.2024
					PROYECTO WEB				Terminado																			
PFR-17	03.02.2024	07.02.2024	Miguel	Terminado																								
PFR-18	08.02.2024	12.02.2024	Miguel	Terminado																								
PFR-19	13.02.2024	16.02.2024	Miguel	Terminado																								
PFR-20	17.02.2024	19.02.2024	Miguel	Terminado																								
PFR-21	19.02.2024	21.02.2024	Miguel	Terminado																								
PFR-22	21.02.2024	23.02.2024	Miguel	Terminado																								
PFR-23	24.02.2024	26.02.2024	Miguel	Terminado																								

Ilustración 8. Diagrama Gantt Tareas PFR-17 - PFR-23.

## Arquitectura web del sistema

Para la implementación del sistema se utilizó Para el Frontend El framework Vue.js por sus características de modularidad y libertad para añadir herramientas según la necesidad, además de sus propiedades reactivas y componentes.

Para el Backend Se utilizó el framework Laravel en su versión 10 principalmente por ser una herramienta potente y sencilla de utilizar, gracias a su arquitectura MVC (Modelo – Vista - Controlador) y la facilidad para integrar con otras arquitecturas como los servicios, pero sobre todo por la capacidad de facilitar al programador el desarrollo de API's para su consumo e integración.

Como base de datos el sistema utiliza MariaDB y se optó por un modelo de base de datos relacional por su precisión en lo que se desea implementar y el manejo que permite las librerías de Laravel.

El proyecto estará alojado en el servidor de DigitalOcean, a petición del cliente por su familiaridad con dicha plataforma de servicios en la nube, su simplicidad y seguridad.

La **¡Error! No se encuentra el origen de la referencia.** muestra la arquitectura del sistema web propuesto.

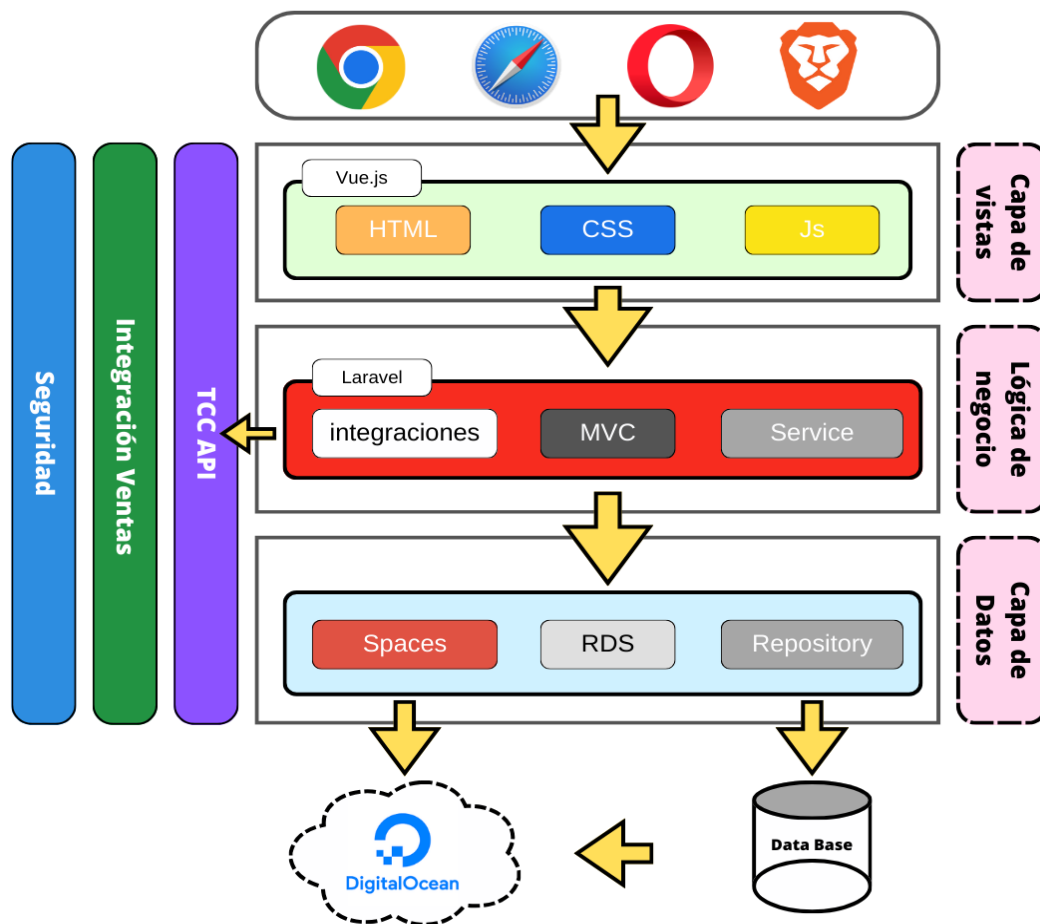


Ilustración 9. Arquitectura web del sistema.

*Nota.* Podemos ver la arquitectura del sistema desde su base de datos hasta su alojamiento.

De mejor manera podemos ver en la Ilustración 10 la arquitectura del sistema organizado en capas funcionales.

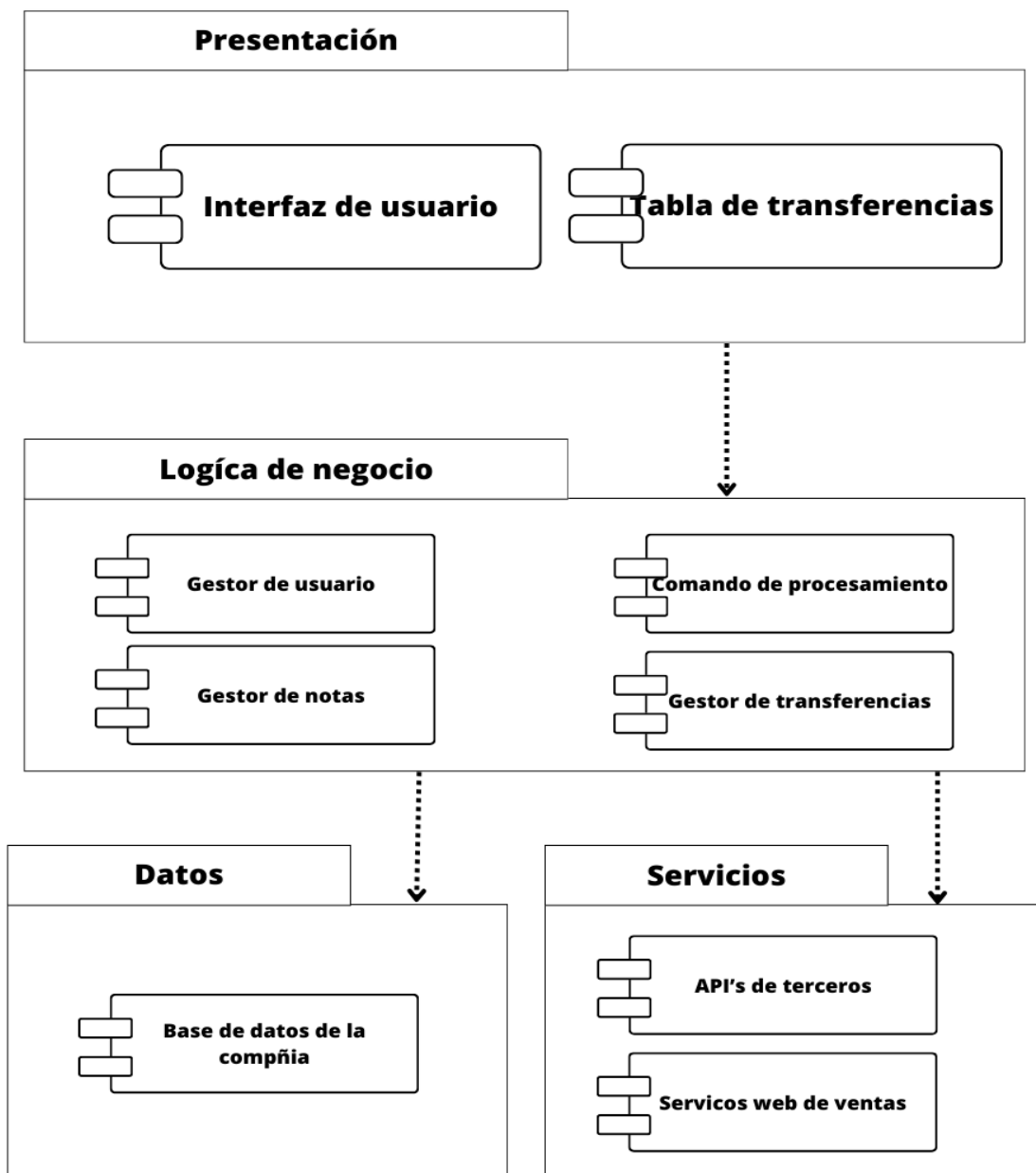
**La presentación** que incluye la **Interfaz de usuario** y la **Tabla de transferencias**, que son las partes visibles y manipulables para el usuario. Esta capa permite la interacción directa con el sistema.

**Lógica de Negocio** que contiene módulos que gestionan la lógica y reglas del sistema:

- **Gestor de usuario:** Administra la información y acciones relacionadas con los usuarios.
- **Gestor de notas:** Maneja la creación y modificación de notas o comentarios asociados a las transferencias.
- **Comando de procesamiento:** Ejecuta los algoritmos que procesan las transferencias.
- **Gestor de transferencias:** Organiza y gestiona el flujo y estado de las transferencias.

**Datos** que incluye la **Base de datos de la compañía**, que almacena la información estructurada necesaria para el funcionamiento del sistema.

**Servicios** Abarca las **API's de terceros** y **Servicios web de ventas**, permitiendo al sistema integrarse con servicios externos y acceder a datos adicionales para complementar sus funcionalidades.



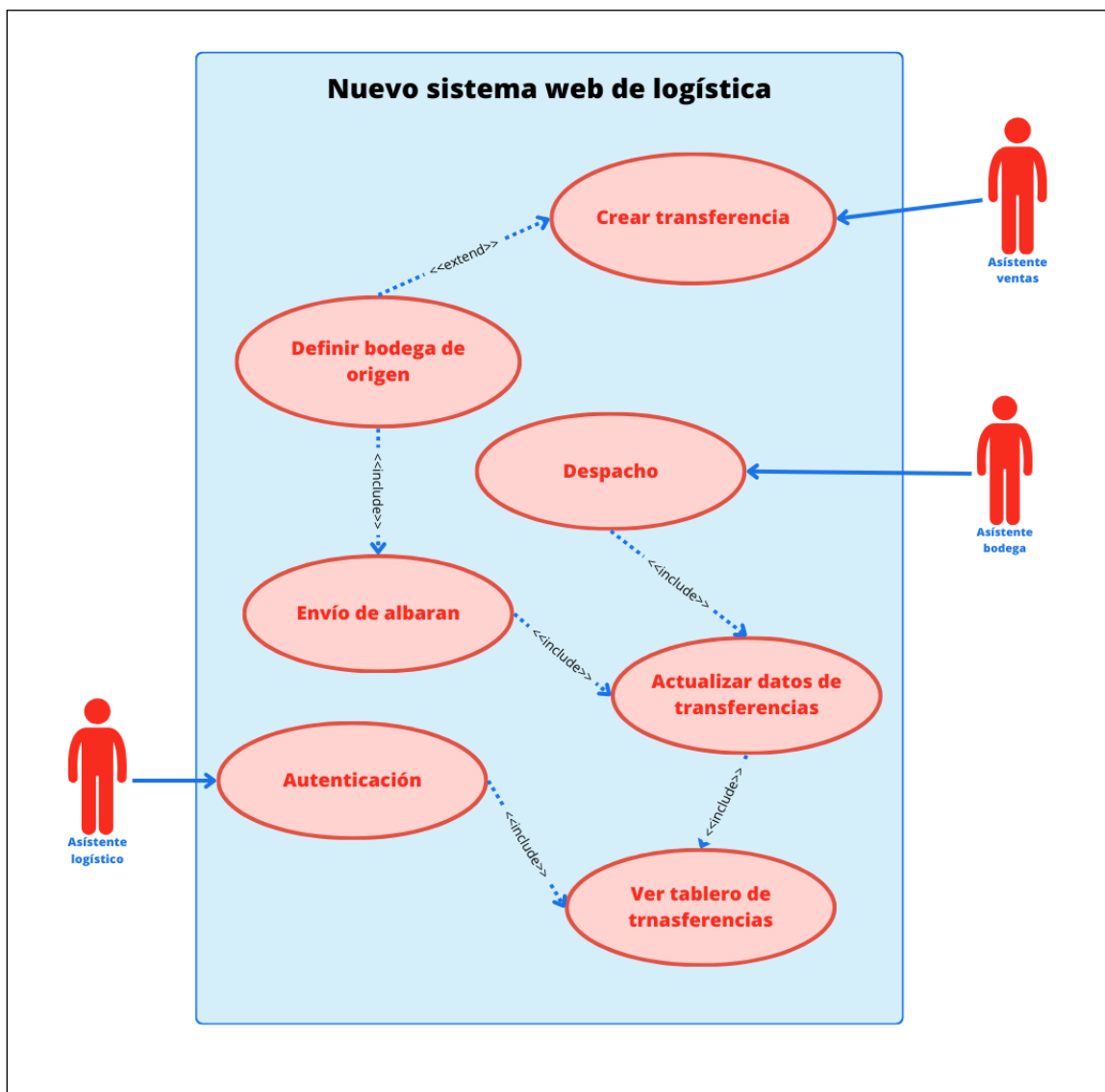
*Ilustración 10.* Diagrama de componentes del sistema por capas.

*Nota.* Arquitectura del sistema que muestra las capas de Presentación, Lógica de Negocio, Datos y Servicios, con módulos especializados para gestionar usuarios, transferencias y comunicación con servicios externos.

Por otra parte, para el desarrollo del mapa de procesos, al cual se aplicará para automatizar la asignación de las transferencias a las bodegas. A continuación, en la **¡Error! No se encuentra el origen de la referencia.** y en la **¡Error! No se encuentra el origen de la referencia.** se presentan los diagramas de casos de uso según el sistema web propuesto.



*Nota.* Con la automatización del procesamiento, este algoritmo logra procesar desde las validaciones hasta el envió o del albarán por cada transferencia entre 0.5 y 4 segundos y este comando se ejecuta entre 7:00am y 5:00pm cada 10 minutos.

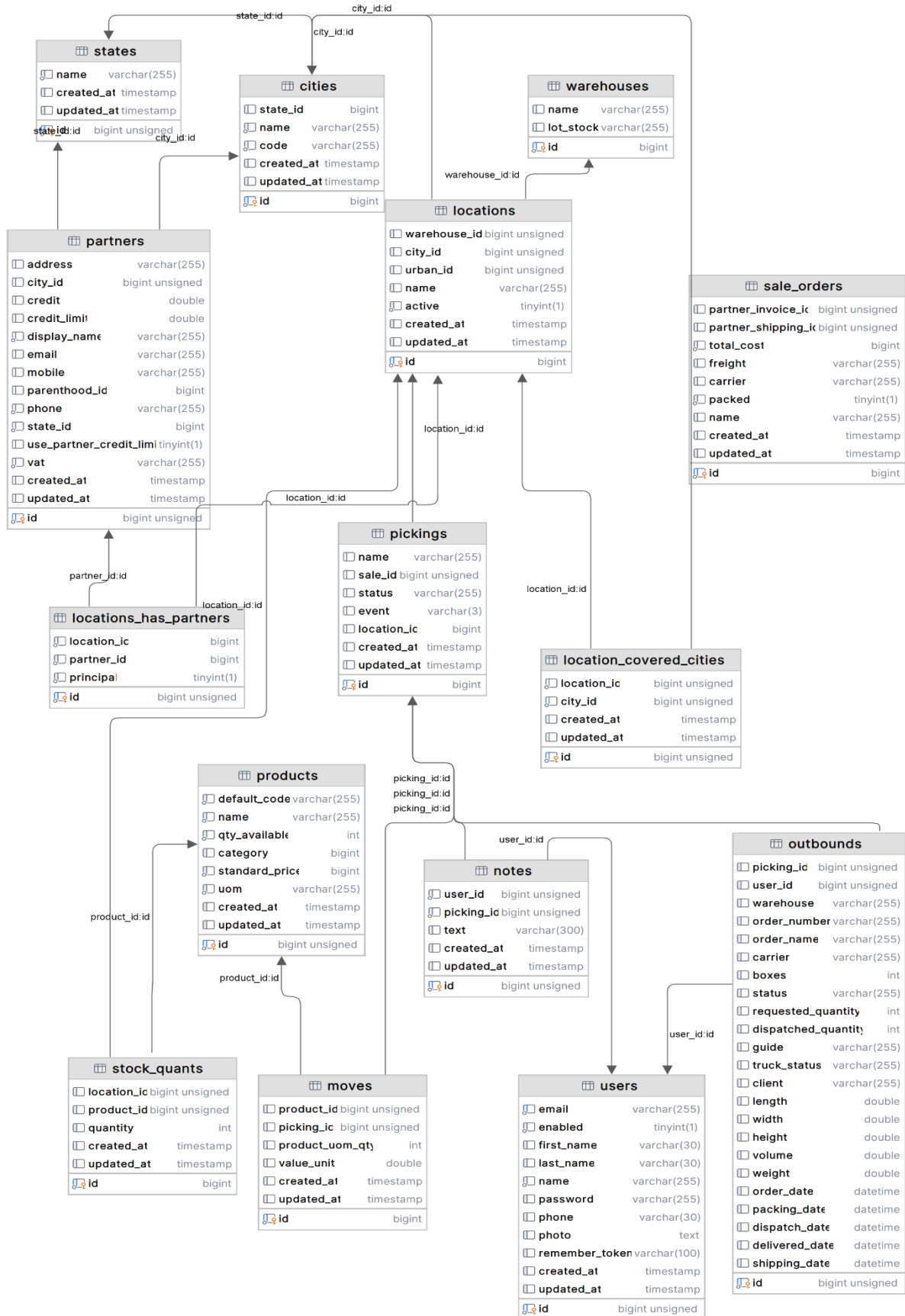


*Ilustración 12.* Consultar tabla de transferencias.

*Nota.* Con el nuevo tablero de seguimiento de transferencias el usuario no tendrá que hacer solicitudes a las bodegas o terceros para obtener información de las salidas de las transferencias pues estos cargarán los avances directamente a la base de datos.

Por otra parte, a partir de los diagramas de casos de uso se estructuraron los nuevos modelos para la base de datos, los cuales se muestran en la

, a continuación:



### *Ilustración 13.* Modelo de base de datos.

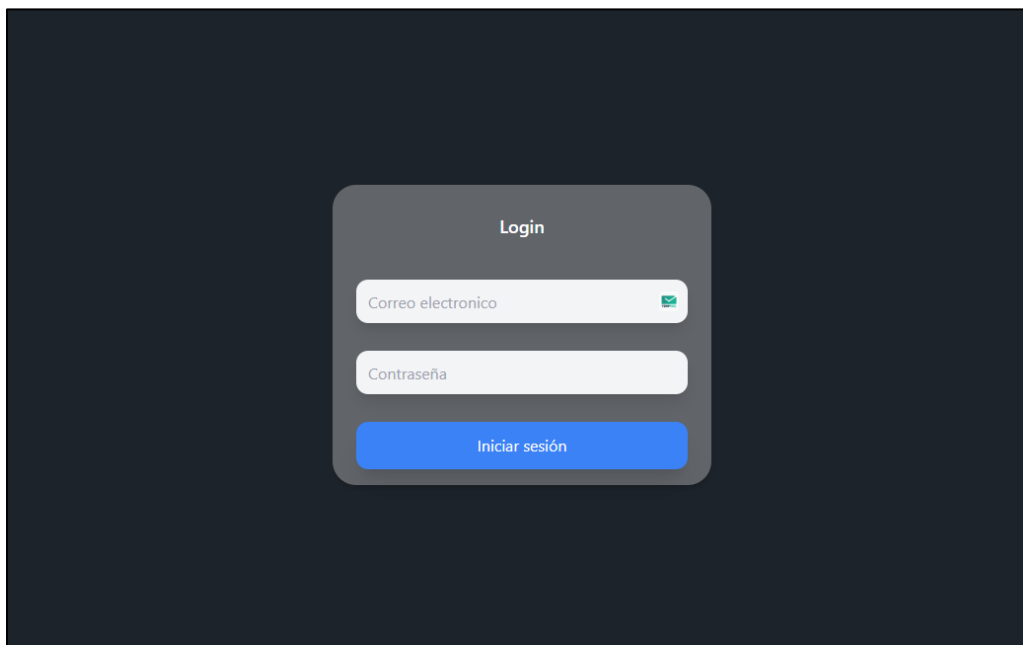
*Nota. Estos son los modelos de tablas que utiliza el sistema para su funcionamiento.*

### **Diseño de interfaz del sistema**

Para la parte visual se tuvieron en cuenta todos los aspectos necesarios para una óptima experiencia de usuario. A continuación, se muestran los principales elementos.

#### ***Login.***

El login cuenta con el formulario de inicio de sesión con sus respectivas validaciones y datos requeridos, la **¡Error! No se encuentra el origen de la referencia.** muestra la interfaz de inicio de sesión. Dado que los usuarios serán creados directamente en la base de datos no se precisó un formulario de registro.

The image shows a dark-themed login interface. At the top center, the word "Login" is displayed in white. Below it, there are two input fields: the first is labeled "Correo electronico" and has a small green envelope icon on the right; the second is labeled "Contraseña". At the bottom of the form is a prominent blue button with the text "Iniciar sesión" in white.

*Ilustración 14.* Login del sistema web propuesto

*Nota.* Podemos ver el formulario de inicio de sesión con su botón.

### ***Tablero de logística.***

La pantalla del tablero de logística del sistema web propuesto muestra una tabla con todas las transferencias procesadas y eventos, además de una serie de datos para su seguimiento. La **¡Error! No se encuentra el origen de la referencia.** muestra los campos que componen el tablero.

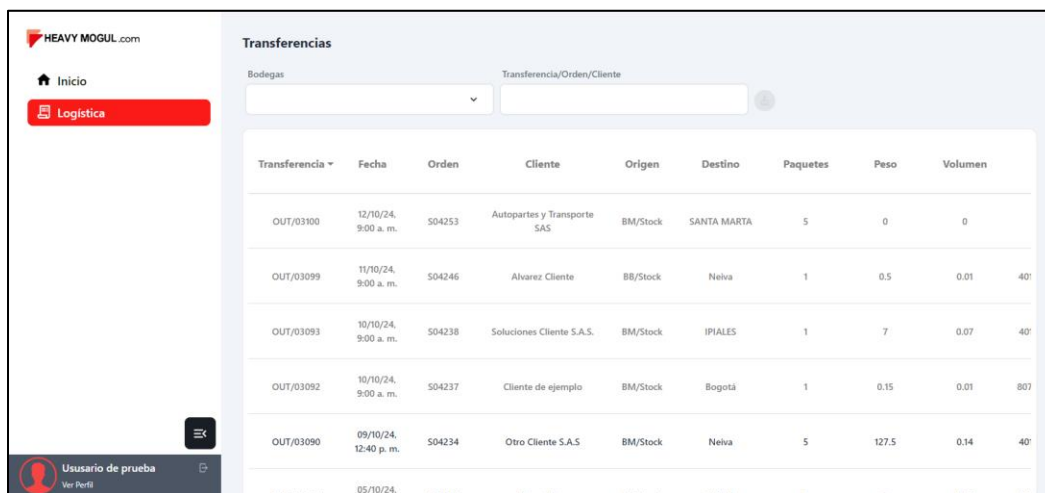
*Tabla 11.* Tablero de logística

<b>Campo</b>	<b>Descripción</b>	<b>Características</b>
Transferencia	Código de identificación de la transferencia.	Es texto y no puede ser modificado y es único.
Fecha	Fecha prevista en la que se empiece su proceso de salida.	Es un dato guardado en la base de datos posterior al procesamiento de la transferencia. Si la transferencia se procesa muy tarde este campo mostrará la fecha del día siguiente a las 9:00am Sino debe ser 3 horas después.
Orden	Código de la orden asignada a esta transferencia.	Si la transferencia se divide, este campo debe mostrar un botón que permita filtrar las transferencias de esta orden.
Cliente	Es el contacto a quien se le factura esta orden.	Es texto y no puede ser modificado.
Origen	Es el nombre de la bodega de donde saldrá el producto.	Es texto y no puede ser modificado.
Destino	Es la ciudad del contacto/dirección donde será entregado el pedido.	Es texto y no puede ser modificado.
Paquetes	Numero de paquetes después del empaque en la bodega.	Es un dato reportado por el sistema de bodega.

Peso	Es el peso total de los paquetes.	Es un dato reportado por el sistema de bodega.
Volumen	Es el volumen total de los paquetes.	Es un dato reportado por el sistema de bodega.
Guía	Es el número de guía que genera la transportadora.	Puede ser modificado por el usuario.
Transportadora	Es el dato de transportadora elegido por el Vendedor en la orden.	Es texto y no puede ser modificado.
Tiempo de empaque	Es el tiempo que le toma a la bodega empaclar los productos desde que se les envía el Albarán.	Es un contador de tiempo continuo y solo se detiene cuando la salida recibe una fecha de empaque. El usuario puede añadir la fecha de empaque o modificarla.
Tiempo de entrega	Es el tiempo que toma la entrega desde que se despacha el pedido hasta que llega al cliente.	Es un contador de tiempo continuo y solo se detiene cuando la salida recibe una fecha de entrega. El usuario puede añadir la fecha de entrega o modificarla.
Evento	Es el estado o evento en el que se encuentra la transferencia.	Los estados están categorizados por color y ordenados por importancia, donde los eventos de entregado y cancelado están de últimos y las fallas o transito están de primeros.
Notas	Es un espacio para agregar notas a la transferencia según la necesidad.	Este es un botón que muestra un listado y un formulario para agregar una nota relacionada a dicha transferencia.
Acciones	Es un espacio para agregar o modificar datos de la transferencia.	Este es un botón que muestra un formulario donde se puede cancelar, Finalizar o modificar y agregar datos de la transferencia como guía, fecha de empaque, fecha de entrega y evento.

*Nota.* Estas son todas las columnas que debe tener el sistema según los requerimientos.

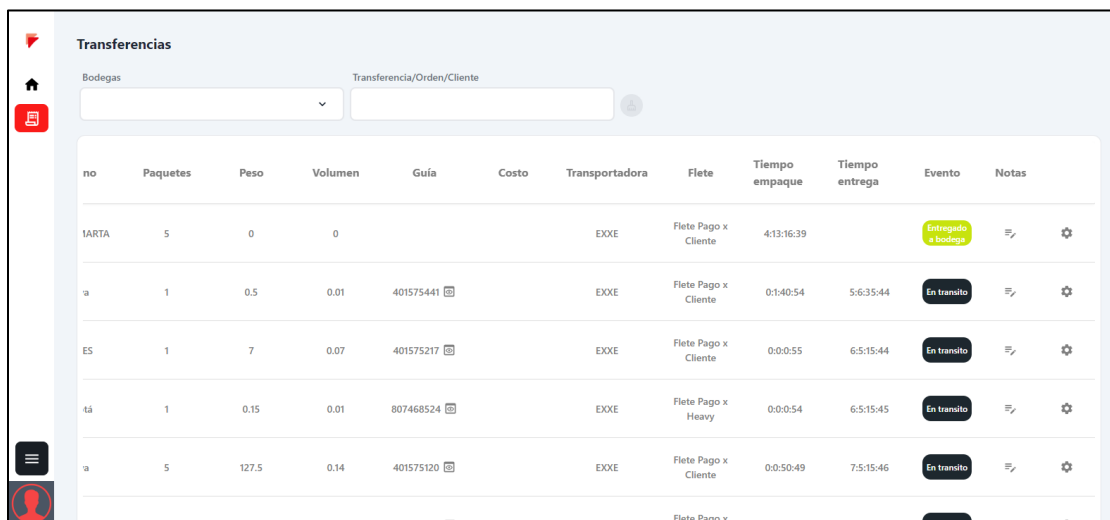
Por su parte, la **¡Error! No se encuentra el origen de la referencia.** y la **¡Error! No se encuentra el origen de la referencia.** muestran la interfaz del tablero de seguimiento de transferencias con los campos especificados en la **¡Error! No se encuentra el origen de la referencia.**



Transferencia	Fecha	Orden	Cliente	Origen	Destino	Paquetes	Peso	Volumen
OUT/03100	12/10/24, 9:00 a. m.	S04253	Autopartes y Transporte SAS	BM/Stock	SANTA MARTA	5	0	0
OUT/03099	11/10/24, 9:00 a. m.	S04246	Alvarez Cliente	BB/Stock	Neiva	1	0.5	0.01
OUT/03093	10/10/24, 9:00 a. m.	S04238	Soluciones Cliente S.A.S.	BM/Stock	IPIALES	1	7	0.07
OUT/03092	10/10/24, 9:00 a. m.	S04237	Cliente de ejemplo	BM/Stock	Bogotá	1	0.15	0.01
OUT/03090	09/10/24, 12:40 p. m.	S04234	Otro Cliente S.A.S.	BM/Stock	Neiva	5	127.5	0.14

*Ilustración 15.* Página de transferencia primera sección.

*Nota.* La primera parte del tablero con el menú de navegación abierto.



no	Paquetes	Peso	Volumen	Guía	Costo	Transportadora	Flete	Tiempo empaque	Tiempo entrega	Evento	Notas
MARTA	5	0	0			EXXE	Flete Pago x Cliente	4:13:16:39		Entregado a Bodega	
va	1	0.5	0.01	401575441		EXXE	Flete Pago x Cliente	0:1:40:54	5:6:35:44	En tránsito	
ES	1	7	0.07	401575217		EXXE	Flete Pago x Cliente	0:0:0:55	6:5:15:44	En tránsito	
da	1	0.15	0.01	807468524		EXXE	Flete Pago x Heavy	0:0:0:54	6:5:15:45	En tránsito	
va	5	127.5	0.14	401575120		EXXE	Flete Pago x Cliente	0:0:50:49	7:5:15:46	En tránsito	

*Ilustración 16.* Página de transferencia segunda sección.

*Nota.* La segunda parte del tablero con sus columnas y eventos, con el menú de navegación cerrado.

### Formulario de Notas.

*Dentro del sistema web se incluyó un formulario de notas el cual se muestra en la*

*¡Error! No se encuentra el origen de la referencia. y en la*

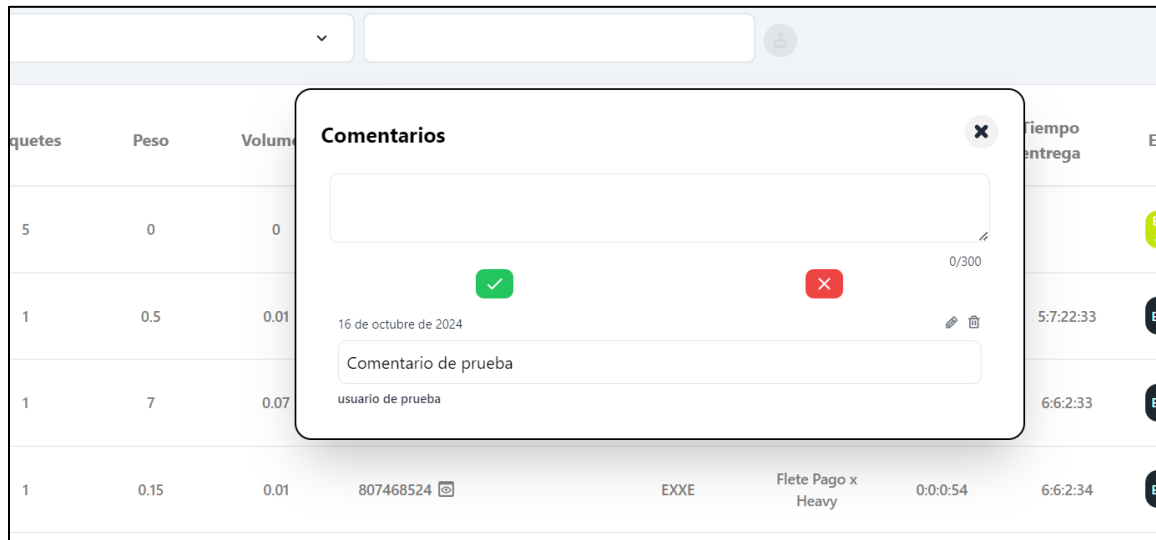
*Nota.* Podemos ver el listado de notas con la fecha, y usuario que la creó.

. Con este modal el usuario podrá agregar, modificar y eliminar, comentarios relacionados a la transferencia.

Peso	Volumen	Guía	Costo	Transportadora	Flete	Tiempo	Tiempo entrega	Evento
0	0							Entregado a bodega
0.5	0.01						5:7:20:57	En transito
7	0.07	401575217		EXXE	Flete Pago x Cliente	0:0:0:55	6:6:0:57	En transito
0.15	0.01	807468524		EXXE	Flete Pago x Heavy	0:0:0:54	6:6:0:58	En transito

*Ilustración 17.* Modal de listado de notas

*Nota.* Podemos ver el listado de notas con la fecha, y usuario que la creó.



*Ilustración 18.* Formulario de nota

*Nota.* Podemos ver que el formulario de creación de nota solo contiene un campo y puede ser cancelado.

### **Formulario de modificación de datos.**

Dentro del sistema web también se incluyó un formulario que permitiera modificar información de las transferencias este se muestra en la **¡Error! No se encuentra el origen de la referencia.** El modal incluye solo los datos pertinentes a la salida que necesitan ser modificados por el usuario en caso de que la bodega no cargue dichos datos o no sean correctos.

*Ilustración 19.* Formulario de modificación de datos de la transferencia.

*Nota.* Vemos el formulario para modificar campos como guía, número de paquetes, Las fechas de los eventos, y los mismos.

### ***Eventos.***

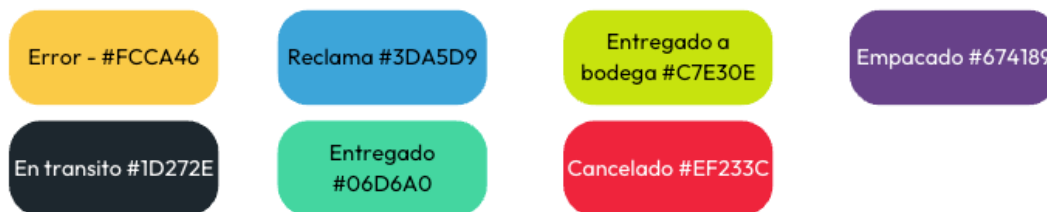
En el sistema web se incluyeron unos eventos que muestran los estados actuales de las transferencias e influyen en el orden del tablero, teniendo en cuenta el orden y los eventos de la **Tabla 12**. Asimismo, la **¡Error! No se encuentra el origen de la referencia.**

muestra los códigos de los colores

*Tabla 12.* Orden y eventos.

Amarillo	Azul	Verde amarillento	Morado	Negro	Verde	Roja
Novedad error	Reclama	Entregado a bodega	Empacado	En tránsito	Entregado al cliente	Cancelado

*Nota.* Esta tabla también está ordenada por orden de importancia de los eventos.



*Ilustración 20.* Colores de los eventos.

*Nota.* Podemos ver los colores y códigos de Color de cada evento.

#### **Fase 4: Finalización**

Para esta fase se hicieron las reuniones finales y pruebas con los interesados en el proyecto y se validó que los objetivos y requerimientos estuvieran cubiertos. Además de esto, se capacitó al personal que le daría uso al sistema.

Después de haber completado todas las pruebas funcionales, se hizo una reunión con los stakeholders necesarios para una última prueba de valoración donde se pudo mostrar el producto final y la valoración por parte de los interesados, asegurándonos que todas las características del proyecto fueron cubiertas.

Posterior a esto, se preparó la migración del sistema a producción validando que la base de datos estuviera correctamente integrada, las API's funcionales y el funcionamiento en el entorno de DigitalOcean fuera óptimo.

- El personal fue capacitado en el uso del sistema.
- Se recogió feedback para futuras actualizaciones y mejoras.
- Se entregó la documentación del proyecto y del sistema.

### **Entrega del producto**

El sistema web fue entregado al cliente cumpliendo con los requisitos funcionales y no funcionales establecidos al inicio del proyecto. Esta solución permite gestionar de manera eficiente las transferencias de inventario, optimizar el control de bodega, y mejorar la comunicación entre áreas mediante un flujo automatizado y trazable. La interfaz amigable y el rendimiento robusto del sistema aseguran una experiencia de usuario fluida, mientras que la integración con servicios externos fortalece la capacidad del sistema para adaptarse a las necesidades de la empresa.

### Pruebas.

la Tabla 13 muestra los resultados de las pruebas funcionales, las cuales comprueban la capacidad del sistema para procesar transferencias, gestionar la autenticación, organizar la información, enviar notificaciones automáticas y permitir la gestión de notas y comentarios.

Tabla 13. Pruebas.

<b>Código</b>	<b>Descripción</b>	<b>Condición</b>	<b>Entrada</b>	<b>Resultado</b>	<b>Pasa</b>
P1	El algoritmo procesa y asigna los movimientos de los productos a la bodega correcta o crea transferencias nuevas para repartirlos.	Debe haber la cantidad del producto en las tres bodegas y los datos deben ser correctos.	Transferencias sin procesar.	El algoritmo asigna las transferencias a la bodega más óptima según la disponibilidad.	Si
P2	Iniciar y cerrar sesión en el sistema web llevando al usuario a la pantalla principal o devolviéndolo al formulario de Login.	Debe existir un usuario con correo y contraseña creados en la base de datos.	Ingresar usuario y contraseña al formulario de Login.	El sistema recibe las credenciales y cuando fueron correctas envió al usuario a la pantalla principal.	Si
P3	La información de las transferencias debe estar correctamente Tabulada y ordenada según los parámetros de los eventos y los filtros funcionales.	- Deben existir registros de transferencias en la Base de datos. - El usuario debe haber iniciado sesión en el sistema.	El usuario entra a la sección de logística del sistema.	El sistema muestra las transferencias con los datos según la historia y ajustados según los filtros.	Si

P4	Envío de albaranes por correo automático.	La transferencia debe haber sido procesada satisfactoriamente.	La transferencia es procesada.	El sistema crea un PDF con los detalles de la transferencia y posteriormente adjuntado se envía un correo a la bodega.	Si
P5	Visualización, creación modificación y eliminación de notas/comentarios para la transferencia.	Debe haber una transferencia en la base de datos para asociar el comentario.	El usuario presiona el botón de notas en la fila de la transacción, escribe un comentario y presiona el botón crear.	El sistema crea el comentario asociado a la transferencia y le permite al usuario visualizarlo, modificarlo o eliminarlo.	Si
P6	Formulario para editar los datos de la transferencia y completarla o cancelarla.	La transferencia debe estar en un evento distinto al entregado o cancelado.	El usuario presiona el botón de editar e ingresa los datos del formulario que desea modificar y así mismo el evento.	El sistema muestra el formulario con los datos de la transferencia y permite al usuario modificarla guardando los datos y actualizando el evento.	Si

*Nota.* El sistema logró pasar todas las pruebas cumpliendo con los requisitos funcionales y no funcionales.

Adicionalmente, la **Tabla 14** detalla las pruebas de compatibilidad realizadas en diversos navegadores, garantizando una experiencia de usuario consistente y sin errores en Microsoft Edge, Chrome, Safari, Opera, Firefox y Brave.

Tabla 14. Pruebas del sistema en distintos navegadores

	<b>Microsoft Edge</b>	<b>Chrome</b>	<b>Safari</b>	<b>Opera</b>	<b>Firefox</b>	<b>Brave</b>
P1	Pasó	Pasó	Pasó	Pasó	Pasó	Pasó
P2	Pasó	Pasó	Pasó	Pasó	Pasó	Pasó
P3	Pasó	Pasó	Pasó	Pasó	Pasó	Pasó
P4	Pasó	Pasó	Pasó	Pasó	Pasó	Pasó
P5	Pasó	Pasó	Pasó	Pasó	Pasó	Pasó
P6	Pasó	Pasó	Pasó	Pasó	Pasó	Pasó

*Nota.* El sistema logró operar correctamente y de forma óptima en todos los navegadores.

## Resultados Y Discusión

A continuación, se presentan los resultados obtenidos dentro de este proyecto de grado, estructurado según las diferentes fases de la metodología scrum, como se planteó en el diseño metodológico.

### Cumplimiento de objetivos

Se desarrolló con éxito un sistema para la gestión y seguimiento de las transferencias de los productos con una interfaz intuitiva y accesible desde diversos navegadores.

Dentro del sistema el usuario tiene un tablero para hacer seguimiento a las transferencias con toda la información requerida por el usuario para la evaluación y toma de decisiones.

Se desarrolló con éxito el comando que se ejecuta cada 10 minutos procesando las transferencias creadas en ese intervalo automáticamente.

Al implementar el proyecto desde el primer día se pudo evidenciar una mejora en los tiempos de respuesta en el procesamiento de las transferencias, pasando de casi 37

minutos por transferencia a un máximo de 4 segundos por transferencia. Así mismo un mejor direccionamiento de las bodegas según la cercanía y costo de fletes.

Una vez desplegado el proyecto y finalizadas las pruebas en concordancia con las reuniones finales, el cliente manifestó el cumplimiento de las expectativas y se pudo dar el cierre formal del proyecto.

## Conclusiones

A continuación, se presentan las conclusiones del proyecto:

**Reducción de Tiempos de Procesamiento:** El sistema automatizado para la gestión y procesamiento de transferencias permitió reducir significativamente el tiempo de procesamiento de cada transferencia, pasando de un promedio de 37 minutos a un máximo de 4 segundos. Esto representa una mejora del 99.82% aproximadamente en los tiempos, impactando directamente en la eficiencia de las operaciones logísticas de la empresa.

**Optimización del Control de Inventario:** Con la implementación de algoritmos que verifican la disponibilidad de inventario y optimizan el envío desde la bodega más adecuada, se garantizó que las transferencias no excedan las cantidades disponibles, minimizando los errores y mejorando la precisión en la gestión de existencias.

**Automatización y Reducción de Errores:** La automatización del proceso de transferencias eliminó la necesidad de intervención humana en actividades repetitivas, lo cual redujo los errores asociados al proceso manual y mejoró la trazabilidad de cada transferencia desde su creación hasta su envío.

**Escalabilidad y Flexibilidad del Sistema:** La arquitectura modular del sistema, basada en Laravel y MariaDB, y alojada en DigitalOcean, asegura que la plataforma es escalable y adaptable, permitiendo la inclusión de nuevas funcionalidades en el futuro sin comprometer el rendimiento o la estabilidad.

**Mejora en la Comunicación entre Bodegas:** La integración de un módulo de comunicación para la notificación automática entre bodegas optimizó la coordinación

logística, lo que contribuye a una mejor gestión del proceso de transferencias y un control logístico más eficiente.

Capacitación y Feedback del Equipo: El equipo de trabajo recibió la capacitación necesaria para operar el sistema de manera efectiva, facilitando la adopción del software. Además, el feedback recibido permitirá realizar futuras mejoras y adaptaciones, asegurando que el sistema siga respondiendo a las necesidades cambiantes de la empresa.

## Referencias

- Acacia Technologies. (13 de Marzo de 2024). *Acacia Technologies*. Recuperado el 2 de Octubre de 2024, de <https://acaciatec.com/transferecia-mercancia-productos-dentro-almacen-sin-perder-rastro/aprende-logistica.com>. (s.f.). *aprende-logistica.com*. Recuperado el 2 de Octubre de 2024, de <https://aprende-logistica.com/almacen/inventario/stock/>
- Avila Pesantez, D. F. (Diciembre de 2020). *Researchgate*. (R. P. Sistemas, Ed.) doi:10.15381/rpcs.v3i2.19256
- Ávila, D. (Diciembre de 2020). Desarrollo de sistema Web basado en los frameworks de Laravel y VueJs, para la gestión por procesos: Un estudio de caso. *Revista peruana de computación y sistemas*, 1 - 10. doi:10.15381/rpcs.v3i2.19256
- Bardales, A., & Galarza, M. (2018). *Universidad San Ignacio de Loyola*. (U. S. Loyola, Ed.) Recuperado el 2 de Octubre de 2024, de <https://repositorio.usil.edu.pe/entities/publication/f19beb80-eec4-4213-8782-9bb9ae30afae>
- Basañes, F. (2 de Enero de 2019). *Banco Interamericano de Desarrollo*. Recuperado el 2 de Octubre de 2024, de <https://blogs.iadb.org/conocimiento-abierto/es/codigo-abierto/>
- Capurro, L. (25 de Marzo de 2024). *Nuthost*. Recuperado el Octubre de 2024, de <https://blog.nuthost.com/laravel-y-vue-js-que-son-y-beneficios-en-la-creacion-de-una-aplicacion-web/>
- Cardador, A. (2014). *Implantación de aplicaciones web en entornos internet, intranet y extranet. ifcd0210 - desarrollo de aplicaciones con tecnologías web*. IC Editorial. Recuperado el 2 de Octubre de 2024
- Celi, R., Boné, M., & Mora, A. (2023). *Programación Web del Frontend al Backend* (Vol. 1). Santo Domingo, Ecuador: Grupo AEA. Recuperado el 2 de Octubre de 2024, de <https://www.editorialgrupo-aea.com/index.php/EditorialGrupoAEA/catalog/book/18>
- Celi, R., Boné, M., & Mora, A. (2023). *Programación Web del Frontend al Backend*. Santo Domingo, Ecuador: Grupo AEA. Recuperado el 2 de Octubre de 2024
- Cohpra, S. (2020). *Administración de la cadena de suministro: estrategia, planeación y operación*. Ciudad de Mexico: Pearson Education.
- Creante Lab SAS. (15 de Junio de 2023). *Linkedin*. Recuperado el 2 de Octubre de 2024, de <https://www.linkedin.com/pulse/automatizaci%C3%B3n-y-desarrollo-de-software-impulsando-la/>
- Datascientest. (9 de Junio de 2024). *Datascientest*. Recuperado el 2 de Octubre de 2024, de <https://datascientest.com/es/que-es-un-algoritmo>
- De Souza, I. (16 de Julio de 2021). *Rock content*. Recuperado el 2 de Octubre de 2024, de <https://rockcontent.com/es/blog/arquitectura-web/#:~:text=La%20arquitectura%20web%20es%20la,medio%20para%20ejecutar%20su%20trabajo.>
- Del Hoyo, O. (Junio de 2017). *oa.upm.es*. (Á. G. Beltrán, Ed.) Recuperado el 2 de Octubre de 2024, de Universidad Politécnica de Madrid: [https://oa.upm.es/47419/1/TFG\\_OSCAR\\_DEL\\_HOYO\\_GUTIERREZ.pdf](https://oa.upm.es/47419/1/TFG_OSCAR_DEL_HOYO_GUTIERREZ.pdf)

- Departamento Nacional de planeación - DNP. (2020). *Departamento Nacional de planeación*. Recuperado el 2 de Octubre de 2024, de <https://colaboracion.dnp.gov.co/CDTI/Oficina%20Informatica/Sistemas%20de%20Informaci%C3%B3n/Gu%C3%ADas%20Formatos%20Plantillas/Gu%C3%ADa%20para%20la%20Elaboraci%C3%B3n%20y%20Presentaci%C3%B3n%20de%20Casos%20de%20Uso.pdf>
- Educa Open. (14 de Febrero de 2024). *Educa Open*. Recuperado el 2 de Octubre de 2024, de <https://www.educaopen.com/digital-lab/blog/software/que-es-laravel>
- Equipo editorial de IONOS. (3 de Junio de 2024). *IONOS*. Recuperado el 2 de Octubre de 2024, de <https://www.ionos.com/es-us/digitalguide/hosting/cuestiones-tecnicas/bases-de-datos-relacionales/>
- Fuente, O. (22 de Noviembre de 2023). *IEBS Digital School*. Recuperado el 2 de Octubre de 2024, de <https://www.iebschool.com/blog/que-es-transformacion-digital-business/>
- García, F. (18 de Abril de 2024). *Arsys*. Recuperado el 2 de Octubre de 2024, de <https://www.arsys.es/blog/introduccion-a-laragon>
- Google Cloud. (s.f.). *Google Cloud*. Recuperado el 2 de Octubre de 2024, de <https://cloud.google.com/learn/what-is-data-integration?hl=es-419>
- Guerra, D. (Marzo de 2021). *Universidad técnica de Ambato*. Recuperado el 26 de Octubre de 2024, de <https://repositorio.uta.edu.ec/bitstream/123456789/32673/1/t1791si.pdf>
- Hernández, U. (22 de Febrero de 2015). *Código facilito*. Recuperado el 2 de Octubre de 2024, de <https://codigofacilito.com/articulos/mvc-model-view-controller-explicado>
- IBM. (s.f.). *IBM*. Recuperado el 2 de Octubre de 2024, de <https://www.ibm.com/mx-es/topics/automation>
- Iglesias Álvarez, I. (18 de Septiembre de 2024). *CIO*. Recuperado el 2 de Octubre de 2024, de <https://www.cio.com/article/3529002/el-30-de-las-empresas-automatizaran-mas-de-la-mitad-de-sus-actividades-de-red-para-2026.html>
- López, L. (2013). *Metodología de la programación Orientada a Objetos - 2da. Ed.* México: Alfaomega Grupo Editó. Recuperado el 2 de Octubre de 2024
- Lozano Rojo, J. (2003). *Cómo y Dónde Optimizar Los Costes Logísticos: En el Sistema Integral de Operaciones y en Las Diferentes áreas de Actividad Logística*. Madrid, España: Fundación Cofemetal. Recuperado el 2 de Octubre de 2024
- Luna, F. O. (2019). *JavaScript - Aprende a programar en el lenguaje de la web*. SIX Ediciones. Recuperado el 2 de Octubre de 2024
- Manzano, A. (28 de Septiembre de 2021). *Dos Setenta*. Recuperado el 2 de Octubre de 2024, de <https://dossetenta.com/arquitectura-web/>
- Martins, J. (18 de Febrero de 2024). *Asana*. Recuperado el 2 de Octubre de 2024, de <https://asana.com/es/resources/project-stakeholder>
- Natsys. (2014). *Todo sobre MySQL*. Natsys. Recuperado el 2 de Octubre de 2024, de [https://books.google.com.co/books/about/Todo\\_sobre\\_MySQL.html?id=GS3kAgAAQBAJ&redir\\_esc=y](https://books.google.com.co/books/about/Todo_sobre_MySQL.html?id=GS3kAgAAQBAJ&redir_esc=y)
- Oracle. (Noviembre de 2020). *oracle.com*. Recuperado el 2 de Octubre de 2024, de <https://www.oracle.com/co/database/what-is-database/>

- Oracle. (s.f.). *Oracle*. Recuperado el 2 de Octubre de 2024, de <https://www.oracle.com/co/mysql/what-is-mysql/>
- Ovando, D. (15 de Diciembre de 2019). *UNIVERSIDAD POLITÉCNICA DE SINALOA*. Recuperado el 2 de Octubre de 2024, de <http://repositorio.upsin.edu.mx/Fragmentos/tesinas/142016030030OvandoOrtegaDenzelJavier10843.pdf>
- Pachas, J. (2021). *Universidad nacional santiago Antúnez de Mayolo*. Recuperado el 2 de Octubre de 2024, de [https://alicia.concytec.gob.pe/vufind/Record/RUNM\\_18a5a47ccc75899cc362830aef504237](https://alicia.concytec.gob.pe/vufind/Record/RUNM_18a5a47ccc75899cc362830aef504237)
- Páez, G. (5 de Mayo de 2020). *Economipedia*. Recuperado el 2 de Octubre de 2024, de <https://economipedia.com/definiciones/flete.html>
- Palacio, M. (2024). *Scrum Master* (Vol. 4). Uncovering Better Ways SLU. Recuperado el 2 de Octubre de 2024, de [https://www.scrummanager.com/files/scrum\\_master.pdf](https://www.scrummanager.com/files/scrum_master.pdf)
- Pérez, J. (18 de Febrero de 2022). *Definición.de*. Recuperado el 2 de Octubre de 2024, de [https://definicion.de/monitoreo/#google\\_vignette](https://definicion.de/monitoreo/#google_vignette)
- Pulido, A. (16 de Enero de 2024). *Online Zebra*. Recuperado el 2 de Octubre de 2024, de <https://onlinezebra.com/arquitectura-web-definicion-y-ejemplos/>
- Purestorage. (s.f.). *Purrestorage*. Recuperado el 2 de Octubre de 2024, de <https://www.purestorage.com/es/knowledge/what-is-mariadb.html>
- Real Academia Española. (s.f.). *Diccionario de la lengua española*. Recuperado el 2 de Octubre de 2024, de <https://dle.rae.es/log%C3%ADstico?m=form>
- Real Academia Española. (s.f.). *Diccionario de la lengua española*. Recuperado el 2 de Octubre de 2024, de <https://dle.rae.es/optimizar?m=form>
- Red Hat. (20 de Septiembre de 2023). *Red hat*. Recuperado el 2 de Octubre de 2024, de <https://www.redhat.com/es/topics/automation>
- Rootstack. (5 de Julio de 2024). *Rootstack*. Recuperado el 2 de Octubre de 2024, de <https://rootstack.com/es/blog/desarrollo-de-software-estadisticas-2024>
- Sznajdieder, P. A. (2023). *Curso de algoritmos y programación a fondo* (Segunda ed.). Argentina: Alfaomega. Recuperado el 2 de Febrero de 2024
- Universidad de los Andes. (21 de Septiembre de 2023). Recuperado el 2 de Octubre de 2024, de Universidad de los Andes: <https://programas.uniandes.edu.co/blog/que-es-una-matriz-dofa-descubre-como-usar-esta-herramienta-para-potenciar-tus-fortalezas>
- Universidad Europea. (15 de Abril de 2024). *universidadeuropea.com*. Recuperado el 2 de Octubre de 2024, de <https://colombia.universidadeuropea.com/blog/programacion-orientada-objetos/>
- Valarezo, M., Honores, J., & Gómez, A. V. (14 de Septiembre de 2018). *COMPARACIÓN DE TENDENCIAS TECNOLÓGICAS*. 7(3), 49. doi:<https://3ciencias.com/articulos/articulo/comparacion-de-tendencias-tecnologicas-en-aplicaciones-web/>
- Vue. (s.f.). *Vue.js*. (E. You, Productor) Recuperado el 2 de Octubre de 2024, de <https://es.vuejs.org/v2/guide/>
- Westreicher, G. (19 de Febrero de 2024). *economipedia.com*. Recuperado el 2 de Octubre de 2024, de <https://economipedia.com/definiciones/gestion.html>

