



Facultad 4

TRABAJO DE DIPLOMA PARA OPTAR POR EL
TÍTULO DE INGENIERO EN CIENCIAS INFORMÁTICAS

Sistema de gestión empresarial para la MIPYME Cubamodela

Autores:

Roxana Turro Aranda
Carlos Daniel Gisbert Morales

Tutores:

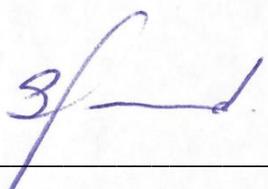
M.Sc. Yadira Ramírez Rodríguez
Ing. David Pérez Falcón

La Habana, diciembre 2023

Año 65 de la Revolución

DECLARACIÓN DE AUTORÍA

Los autores del trabajo de diploma con título “**Sistema para la gestión empresarial de la MIPYME Cubamodela a través de los módulos almacén y economía**” conceden a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la investigación, con carácter exclusivo. De forma similar se declaran únicos autores de su contenido. Para que así conste firma la presente a los 8 días del mes de diciembre del año 2023.



Roxana Turro Aranda

Autor



Carlos Daniel Gisbert Morales

Autor



M.Sc. Yadira Ramírez Rodríguez

Tutor



Ing. David Pérez Falcón

DATOS DE CONTACTO

Yadira Ramírez Rodríguez, graduada de Ingeniería en Ciencias Informáticas en el 2007. Máster en Calidad de Software y profesora auxiliar con más de 15 de años de experiencia en la asignatura de Ingeniería de Software. Jefa del Departamento Docente de Informática de la Facultad 4 de la Universidad de las Ciencias Informáticas. Líneas de Investigación: Ingeniería de Sistemas y Calidad de Software.

David Pérez Falcón, graduado de Ingeniería en Automática en el 2015 en la Universidad Tecnológica de La Habana “José Antonio Echeverría” CUJAE. Especialista en ciencias informáticas en XETID desde 2015 hasta 2016. Especialista en domótica en la Cooperativa Restaura Habana desde 2016 hasta 2018. Ingeniero en automática en ETECSA desde 2018 hasta 2019. Jefe de Grupo de Automática en AICROS desde 2019 hasta 2021. Director de TI de CUBAMODELA S.U.R.L desde 2020 hasta la actualidad. Director y fundador de Lyarte desde abril de 2023 hasta el presente.

AGRADECIMIENTOS

Queremos expresar nuestra profunda gratitud a todas las personas que han desempeñado un papel fundamental en la realización de este trabajo. Sus apoyos, consejos y contribuciones han sido invaluableles. A nuestros padres, quienes nos han brindado su amor incondicional, apoyo emocional y sacrificio constante a lo largo de nuestra vida. Sus palabras de aliento y confianza en nosotros han sido nuestro faro de esperanza. A nuestros respetados tutores Yadira y David, quienes han compartido su sabiduría, paciencia y orientación experta. Sus valiosas sugerencias y comentarios críticos han enriquecido nuestro trabajo. A nuestros profesores agradecemos su dedicación a nuestra formación académica. A nuestros amigos cercanos, quienes nos han apoyado en cada paso de nuestro camino. Sus palabras de ánimo y momentos compartidos han aligerado el peso de esta tarea. Gracias por estar a nuestro lado en los momentos más desafiantes. A nuestros compañeros de clase, quienes han compartido conocimientos, ideas y experiencias a lo largo de estos años académicos. A todas las personas que de una u otra manera contribuyeron a este proyecto, ya sea proporcionando datos, recursos o apoyo logístico, gracias, sus contribuciones han sido esenciales y altamente valoradas.

A ti compañero de tesis, por ser un colaborador leal, un compañero incansable y un amigo fiel. Hemos enfrentado juntos los desafíos y celebrado los éxitos, y esta tesis es un testimonio de nuestra colaboración.

Este fragmento refleja nuestra profunda gratitud a todos los que han sido parte de este viaje. Sin su apoyo y amabilidad, este logro no sería posible. ¡Gracias de todo corazón!

DEDICATORIA

A nuestros familiares,

Por su amor inquebrantable, su apoyo incondicional y su constante aliento a lo largo de nuestra vida y principalmente en este viaje académico. Su sacrificio y dedicación han sido nuestra fuente de inspiración constante. A nuestros hermanos, abuelos y demás familiares, por su amor y comprensión a lo largo de nuestras vidas. A ustedes, dedicamos nuestro trabajo de diploma.

Roxana Furro Aranda:

A mis padres: Yanet y Eglis

A mis abuelos: Irene, Olidia, Ernesto y Urbano

A mi hermano: Miguel Alejandro

Carlos Daniel Gisbert Morales:

A mis padres: Gisela y Juan Carlos

A mis abuelas: Mirta y Hilda

A mi hermano: Adrián Alejandro

RESUMEN

El presente trabajo de diploma se encarga de desarrollar un sistema que permita la gestión del proceso de venta de la MIPYME Cubamodela, realizando un sistema de gestión empresarial. El objetivo de este es facilitar los procesos de ventas que actualmente se tienen en la tienda CBM, permitiendo automatizarlos y garantizando una mejor gestión de venta, además de ofrecer funcionalidades que permiten una mejora en la administración de la tienda, en el flujo de información y gestión de los procesos. Para guiar el proceso que se desprende de la realización de este trabajo de diploma donde se utilizó como metodología de desarrollo de software XP, como lenguaje de modelado UML y Visual Paradigm como herramienta CASE, lo que permitió un correcto entendimiento entre el cliente y el equipo de desarrollo. Para la construcción de la solución se utilizó Django como framework de desarrollo para el backend y React para el frontend, así como otras herramientas y tecnologías de código abierto.

PALABRAS CLAVE

Módulo Almacén, Módulo Economía, Mipyme Cubamodela, gestión empresarial.

Abstract

The present diploma work is in charge of developing a system that allows the management of the sales process of the MSME Cubamodela, creating a business management system. The objective of this is to facilitate the sales processes that currently exist in the CBM store, allowing to automate them and guaranteeing a better sales management, in addition to offering functionalities that allow an improvement in the administration of the store, in the flow of information and management of the processes. To guide the process that resulted from the realization of this diploma work where Scrum was used as software development methodology, UML as modeling language and Visual Paradigm as CASE tool, which allowed a correct understanding between the client and the development team. Django was used as programming language for the construction of the solution for the backend and React for the frontend, as well as other opensource tools and technologies.

KEYWORDS

Warehouse Module, Economy Module, Cubamodela Mipyme, management business.

Índice

INTRODUCCIÓN.....	1
CAPÍTULO I. FUNDAMENTOS TEÓRICOS DE LA INVESTIGACIÓN	6
1.1 SISTEMA DE GESTIÓN DE ALMACENES	6
1.2 GESTIÓN DEL PROCESO DE INVENTARIOS	8
1.3 SISTEMA ECONÓMICO	9
1.4 GESTIÓN EMPRESARIAL	11
1.5 DIAGNÓSTICO DE LOS MÓDULOS ALMACÉN Y ECONOMÍA.....	12
1.6 ESTUDIO DEL ARTE DE SISTEMAS HOMÓLOGOS	13
1.6.1 Sistemas homólogos en el ámbito nacional e internacional.....	14
1.7 TECNOLOGÍAS Y HERRAMIENTAS.....	17
1.7.1 Lenguaje de programación interfaz usuario	18
1.7.2 Framework de desarrollo interfaz usuario	19
1.7.3 Lenguaje de programación para el servidor.....	20
1.7.4 Framework de desarrollo para el servidor	22
1.7.5 Entorno de Desarrollo Integrado (IDE).....	24
1.7.6 Gestores de bases de datos	24
1.8 METODOLOGÍA DE DESARROLLO DE SOFTWARE	25
CONCLUSIONES PARCIALES:	28
CAPÍTULO II. CARACTERÍSTICAS Y DISEÑO DEL SISTEMA	30
2.1 PROPUESTA DE SOLUCIÓN	30
2.1.1 Requisitos funcionales	32
2.2 FASE I: PLANIFICACIÓN.....	37
2.2.1. Historias de usuarios	37
2.2.2 Requisitos no funcionales	55
2.2.3 Estimación de esfuerzo por Historia de Usuario	57
2.2.4 Desarrollo del plan de iteraciones	58
2.2.5 Plan de entregas.....	59
2.3 FASE II: DISEÑO DEL SISTEMA	59
2.4 PATRONES DE DISEÑO	62
2.5 TARJETAS CLASE - RESPONSABILIDAD – COLABORADOR (CRC).....	63
CONCLUSIONES PARCIALES	71
CAPÍTULO III: IMPLEMENTACIÓN Y PRUEBA DEL SISTEMA.....	72
3.1 FASE III: DESARROLLO	72
3.1.1 Tareas de ingeniería	72
3.2 FASE IV: PRUEBAS.....	75
3.2.1 Pruebas de Unitarias.....	75
3.2.2 Pruebas de aceptación.....	78
3.2.3 Pruebas de aceptación para la Iteración I.....	78

3.2.4 Pruebas de aceptación para la Iteración II.....	81
3.2.5 Pruebas de aceptación para la Iteración III.....	82
3.2.6 Análisis de las pruebas de aceptación	83
CONCLUSIONES DEL CAPÍTULO	85
CONCLUSIONES.....	86
RECOMENDACIONES.....	87
REFERENCIA BIBLIOGRÁFICA.....	88
ANEXOS.....	92

Índice de Ilustraciones

Ilustración 1. Propuesta de solución en el módulo almacén. Fuente: elaboración propia.	31
Ilustración 2. Propuesta de solución en el módulo economía. Fuente: elaboración propia.	31
Ilustración 3. Diseño arquitectónico para sistema de gestión empresarial. Fuente: elaboración propia.....	61
Ilustración 4. Pruebas unitarias al gestionar almacenes.....	76
Ilustración 5. Pruebas unitarias al gestionar entrada y salidas de productos del almacén.	77
Ilustración 6. Comparativa de resultados de pruebas unitarias. Fuente: elaboración propia...	78
Ilustración 7. Resultados de las pruebas de aceptación.....	83
Ilustración 8. Aval de satisfacción de cliente. Fuente: dirección de la mipyme cubamodela .	122
Ilustración 9. Pruebas unitarias al gestionar entrada y salidas de productos del almacén. ...	122
Ilustración 11. Pruebas unitarias al gestionar pre despacho	124
Ilustración 12. Pruebas unitarias al gestionar entrada y salida de productos al almacén	125

Índice de Tablas

Tabla 1: sistemas homólogos en el ámbito nacional. Fuente: elaboración propia.....	15
Tabla 2: sistemas homólogos en el ámbito extranjero. Fuente: elaboración propia	16
Tabla 3: características generales de las tecnologías estudiadas. Fuente: elaboración propia	20
Tabla 4: historia de usuario # 1	38
Tabla 5: historia de usuario # 2	39
Tabla 6: historia de usuario # 3	40
Tabla 7: historia de usuario # 4	42
Tabla 8: historia de usuario # 5	43
Tabla 9: historia de usuario # 6	44
Tabla 11: historia de usuario # 7	46
Tabla 12: historia de usuario # 8	47
Tabla 13: historia de usuario # 9	48
Tabla 14: historia de usuario # 10	48
Tabla 15: historia de usuario # 11	49
Tabla 16: historia de usuario # 12	51
Tabla 17: historia de usuario # 13	51
Tabla 18: historia de usuario # 14	53
Tabla 19: historia de usuario # 15	54
Tabla 20: estimación de esfuerzo por historia de usuario.....	57
Tabla 21: plan de duración de las iteraciones	58
Tabla 22: plan de entrega de versiones	59
Tabla 23: tarjeta crc # 1	63
Tabla 24: tarjeta crc # 2.....	64
Tabla 25: tarjeta crc # 3.....	65
Tabla 26: tarjeta crc # 4.....	65
Tabla 27: tarjeta crc # 5.....	66
Tabla 28: tarjeta crc # 6.....	67
Tabla 29: tarjeta crc # 7	68

Tabla 30: tarjeta crc # 8	68
Tabla 31: tarjeta crc # 9	69
Tabla 32: tarjeta crc # 10	70
Tabla 33 tarea de ingeniería # 1	72
Tabla 34 tarea de ingeniería # 2	73
Tabla 35tarea de ingeniería # 3	73
Tabla 36 tarea de ingeniería # 4	74
Tabla 37 tarea de ingeniería # 5	74
Tabla 69 prueba de aceptación # 1	78
Tabla 70 prueba de aceptación # 2	79
Tabla 71 prueba de aceptación # 3	79
Tabla 72 prueba de aceptación # 4	80
Tabla 73 prueba de aceptación # 1	81
Tabla 74 prueba de aceptación # 1	82
Tabla 75 prueba de aceptación # 2	82
Tabla 38 tarea de ingeniería # 6	93
Tabla 39 tarea de ingeniería # 7	93
Tabla 40 tarea de ingeniería # 8	94
Tabla 41 tarea de ingeniería # 9	94
Tabla 42 tarea de ingeniería # 10	94
Tabla 43 tarea de ingeniería # 11	95
Tabla 44 tarea de ingeniería # 12	95
Tabla 45 tarea de ingeniería # 13	95
Tabla 46 tarea de ingeniería # 14	96
Tabla 47 tarea de ingeniería # 15	96
Tabla 48 tarea de ingeniería # 16	97
Tabla 49 tarea de ingeniería # 17	97
Tabla 50 tarea de ingeniería # 18	97
Tabla 51 tarea de ingeniería # 19	98
Tabla 52 tarea de ingeniería # 20	98
Tabla 53 tarea de ingeniería # 21	99

Tabla 54 tarea de ingeniería # 22.....	99
Tabla 55 tarea de ingeniería # 23.....	99
Tabla 56 tarea de ingeniería # 24.....	100
Tabla 57 tarea de ingeniería # 25.....	100
Tabla 58 tarea de ingeniería # 26.....	101
Tabla 59 tarea de ingeniería # 27.....	101
Tabla 60 tarea de ingeniería # 28.....	101
Tabla 61 tarea de ingeniería #29.....	102
Tabla 62 tarea de ingeniería # 30.....	102
Tabla 63 tarea de ingeniería #31.....	103
Tabla 64 tarea de ingeniería #32.....	103
Tabla 65 tarea de ingeniería #33.....	103
Tabla 66 tarea de ingeniería #34.....	104
Tabla 67 tarea de ingeniería #35.....	104
Tabla 68 tarea de ingeniería #36.....	105
Tabla 76 prueba de aceptación # 5.....	105
Tabla 77 prueba de aceptación # 6.....	106
Tabla 78 prueba de aceptación # 7.....	106
Tabla 79 prueba de aceptación # 8.....	107
Tabla 80 prueba de aceptación # 9.....	108
Tabla 81 prueba de aceptación # 10.....	108
Tabla 82 prueba de aceptación # 11.....	109
Tabla 83 prueba de aceptación # 2.....	110
Tabla 84 prueba de aceptación # 3.....	110

Introducción

La informática como herramienta y la información como portador del conocimiento dieron lugar a la información digital. El desarrollo de las nuevas Tecnologías de la Información y las Comunicaciones (TICs) permitió a la sociedad acrecentar como nunca antes el consumo de las mismas. Las TICs se han extendido a nivel social, creando una interdependencia a escala global que se autogenera día a día, ofreciendo nuevas posibilidades de desarrollo en todas las esferas de la vida (Vincent Domenech, 2021).

El Internet, o más precisamente las redes sociales, se han erigido en el espacio por antonomasia donde las personas publican, comentan, debaten y comparten ideas, informaciones, videos, fotografías y demás. El Internet y las redes sociales, haciendo uso de la cita del ex juez de la Corte Suprema de Estados Unidos Anthony M. Kennedy, es la “plaza pública moderna” (Calcaneo Monts, 2021).

A nivel organizacional, este desarrollo ha llevado a la empresa moderna a un contexto donde se produce la necesidad de digitalizar los procesos de la esfera social, económica y política. El uso del comercio electrónico a través de internet permite realizar distintas operaciones (compra-venta, búsqueda de información, comunicación entre personas, etc.) utilizando ordenadores o dispositivos móviles con mayor agilidad, para viabilizar entre otros, las compras y trámites realizados por los usuarios (Vincent Domenech, 2021).

El comercio electrónico se ha convertido en un canal de ventas esencial, sobre todo, desde finales de la década de 1990, la aparición de la tecnología Internet trae consigo una nueva visión estratégica para el comercio detallista, generando nuevas oportunidades en el sector. El comercio electrónico supone la aparición de nuevas formas de comercialización en la cadena de distribución alimentaria, que operan complementando las formas de distribución tradicionales (Revista CEA, 2021).

En el libro “Manual del comercio electrónico”, exponen que el término de comercio electrónico se define como una herramienta instaurada en todo el terreno comercial (Hernández Barrueco & Hernández Ramos, 2021). Para un exitoso desempeño, las empresas y profesionales que han de gestionarlo deben conocer los elementos y procesos clave que intervienen en una actividad comercial. Al mismo tiempo la acogida del comercio web en el ámbito empresarial a

nivel global tiene un aumento continuo, no obstante, en tanto que las organizaciones potencian su tamaño, el comercio electrónico se convierte cada vez más difícil y desafiantes (Fabián et al., 2020).

En los últimos años, la sociedad cubana ha evolucionado hacia la informatización, y en igual medida lo han hecho las empresas y entidades que la conforman. Como parte de este proceso, varias han sido las empresas que han implementado el comercio electrónico, lo cual ayuda a mejorar su productividad y rentabilidad. Actualmente las empresas cuentan con un volumen considerable de datos que pueden ser utilizados en la optimización de estos procesos, pero no son aprovechados en la mayoría de los casos. En el contexto del comercio electrónico con servicio de entrega a domicilio, resulta de vital importancia la optimización de rutas de distribución. Entre las principales ventajas operativas están el ahorro de recursos limitados, como es el caso del combustible y la reducción de los tiempos de entrega, y el agrupamiento de las entregas de forma que más de un vehículo no visite una misma localidad, entre otros (Suárez González & Pérez Pérez, 2023).

Hoy, más que nunca, esta forma de comercialización se elige como la principal para los consumidores, puesto que la perciben como una forma segura para adquirir bienes y servicios. El comercio electrónico ha establecido patrones de actuación en el sector, facilitando las operaciones y ampliando su alcance (López et al., 2021).

La MIPYME Cubamodela, fue una de las primeras miniempresas creadas en Cuba, para mediados del 2021 surgió la misma, desarrollando ventas de diversos productos a la población cubana, en este mismo año Cuba y el mundo atravesaba una situación crítica, debido a la pandemia existente por el virus COVID 19, debido a la misma la empresa opto como una vía de mejora para sus ventas implementar el desarrollo del comercio electrónico en el país. Actualmente la MIPYME cuenta con una tienda online encargada de gestionar ventas de variados productos de proveedores tanto nacionales como internacionales.

La tienda diariamente ofrece gran variedad productos al pueblo cubano, al igual que cada vez más incorpora nuevos artículos de gran interés para la población y de esta forma aumentando sus ventas a disimiles escalas. La empresa tiene como objetivo convertirse en una especie de Amazon cubano, donde el pueblo pueda encontrar todo tipo de productos.

Para que esto sea posible la MIPYME ha optado por la opción de arrendar grandes almacenes donde pueda almacenar sus variados productos y al mismo tiempo contratar nuevo personal para encargarse de estos almacenes, con este arrendamiento ya la entidad no carecerá de espacio y siempre podrá tener un gran stock en la tienda, permitiendo que los clientes no se vean limitados a comprar en pequeñas cantidades, sino que tendrán la opción de comprar las cantidades que deseen.

Al aumentar las ventas en la tienda, aumenta el trabajo económico de fondo que se desarrolla en la entidad, surgiendo como principal necesidad contratar nuevo personal calificado para ayudar a llevar los procesos contables y económicos de la empresa. Con la contratación de nuevas personas la entidad debe pagar a los mismos su salario correspondiente, por lo que aumenta el número de salarios a pagar y con esto disminuye el porcentaje de ganancia de la empresa.

Actualmente la gestión interna de la empresa se realiza de forma manual, tanto el proceso de almacenado de los productos como los procesos económicos de las ventas realizadas a través de la tienda, debido a esto se ve ralentizado el proceso de ventas, ya que la entrada de los productos a almacén, al igual que la salida de los mismos se lleva actualmente a través de planillas, volviendo engorroso este proceso, al igual que dando paso a la pérdida de información valiosa para el negocio. También se ve como los procesos económicos de la empresa se llevan a través de tablas Excel, lo que da paso a una lenta gestión de los procesos económicos, afectando los pagos a través de las ventas realizadas.

Estos aspectos contribuyen en el retraso del proceso de ventas, al igual que exige un numeroso personal calificado para poder llevar a cabo cada uno de estos procesos. Viéndose en esta situación, la empresa vio la necesidad de agilizar la gestión de estos procesos y de esta forma aumentar la funcionalidad y el rendimiento del servicio prestado a la población cubana.

Crear satisfacción en los clientes a día de hoy se ha vuelto una prioridad de la empresa por lo que es necesario el mejoramiento de los servicios prestados al pueblo, del mismo modo el rendimiento de trabajo dentro de la entidad.

En la situación problemática antes descrita, se plantea el siguiente **problema**: ¿Cómo agilizar los procesos asociados a la gestión empresarial de la tienda online Cubamodela?

Se define como **objeto de estudio** de este problema: sistemas de gestión empresarial.

La investigación se enmarca en el siguiente **Campo de acción**: sistema de gestión empresarial para la MIPYME Cubamodela.

Para dar solución al problema se define como **Objetivo general**: desarrollar un sistema de gestión empresarial para la MIPYME Cubamodela.

tareas de investigación:

1. Desarrollo del marco teórico de la investigación a través del estudio del estado del arte para facilitar la comprensión de la investigación
2. Levantamiento de requisitos funcionales y no funcionales para definir la propuesta de solución.
3. Implementación de las funcionalidades del sistema para la gestión del proceso.
4. Validación de las funcionalidades del sistema de gestión empresarial para la tienda online Cubamodela.

Para el desarrollo de la investigación se utilizaron **métodos científicos**, presentados a continuación:

Métodos teóricos:

- **Analítico-Sintético**: permitió la recopilación de información necesaria durante la realización del estudio del estado del arte en el Sistema de gestión empresarial ,siguiendo las reglas lógicas de deducción se llega a obtener nuevos conocimientos del tema y se logra definir una solución informática para gestión del proceso de entrada y salida de productos del almacén y todo el proceso contable de Cubamodela , los cuales una vez implementados se evaluará para conocer su impacto y se podrá corroborar si se cumple la hipótesis descrita en este proyecto (Vincent Domenech, 2021).
- **Comparativo**: permitió comparar diferentes fenómenos o situaciones para identificar similitudes y diferencias. Este método se utilizó para establecer relaciones entre diferentes variables y para identificar patrones en los datos.

Métodos empíricos:

- **Entrevista:** Se le realiza al encargado del almacén, al director de la tienda y a la directora de economía, con el objetivo de tener conocimiento sobre el grado de satisfacción que se desea tener en el producto final y control de la información que se va a manejar ([Ver anexo 1](#)).
- **Observación:** se utiliza para observar cómo se realiza la gestión del proceso de entrada y salida de los productos de la tienda, al igual que los procesos económicos, los cuales se realizan de forma manual ([Ver anexo 2](#)).

El presente documento está compuesto por 3 capítulos:

Capítulo 1: Fundamentos teóricos de la investigación. Se realiza un estudio del estado del arte y se exponen las bases que sustentan la investigación, donde se describen los conceptos esenciales para una mejor comprensión del trabajo a realizar. Así como la selección de las tecnologías a utilizar para la solución del problema y se fundamenta la metodología de desarrollo de software utilizada para guiar el proceso de desarrollo de la propuesta de solución.

Capítulo 2: Características y diseño del sistema. Se elabora una pequeña descripción de la solución propuesta. Se realizan las historias de usuario que se utilizan para especificar los requisitos funcionales y no funcionales, reflejando una breve descripción de cada uno de estos. Se definen los patrones de diseño a utilizar y la arquitectura del sistema. Se elaboran además las tarjetas Clase-Responsabilidades-Colaboradores (CRC), para describir las clases que conforman el sistema. Se especifican los roles del sistema con sus responsabilidades. Concluyendo con el diseño del diagrama entidad relación de la base de datos.

Capítulo 3: Implementación y prueba del sistema. Se realiza una planificación por iteraciones correspondiente a la implementación de la aplicación donde se describen las tareas de ingeniería correspondiente a cada historia de usuario y se realizan un conjunto de pruebas de aceptación a cada una de estas, concluyendo con un gráfico resumen de resultado de las pruebas aplicadas.

Capítulo I. Fundamentos teóricos de la investigación

Este capítulo estará dedicado a referenciar el estado del arte del tema a desarrollar: Módulos Almacén y economía de la Gestión Empresarial de la MIPYME Cubamodela; en el mismo quedarán expuestos de manera general la existencia de otros sistemas homólogos al que se pretende desarrollar como objetivo final de esta investigación, para ello se realizará un análisis del ámbito nacional e internacional con el objetivo de visualizar ventajas y desventajas para la posterior creación del sistema.

1.1 Sistema de gestión de almacenes

Partiendo del término almacén, se conoce que es un lugar donde se guardan y se venden productos, materiales y mercancías. El almacén tiene la función de regular el flujo de existencias entre la producción y el consumo. El almacén también sirve para custodiar, proteger y controlar los bienes de una empresa u organización. También puede ser un espacio físico o virtual. El almacén se puede definir como el espacio físico de una empresa en el que se almacenan productos terminados, materias primas o productos en proceso (Cano Josías Caleb Pág & Cano Josías Caleb, 2019).

Los almacenes pueden ser clasificados de varias maneras, una forma común de clasificar los almacenes es por la naturaleza del producto que se almacena, por ejemplo, hay almacenes especializados en productos inflamables, productos perecederos, recambios, distribución, entre otros. Los almacenes de distribución se organizan para maximizar la eficiencia y la velocidad del proceso de envío. Los productos están en el almacén, se verifican y se colocan en estantes o cajas según su categoría. Los productos se almacenan en función de su tamaño, peso y rotación. Cuando un cliente realiza un pedido, los productos se recogen del almacén y se preparan para el envío o recogida del cliente. Para que este proceso se realice y funcione correctamente debe de existir una buena gestión de almacenes.

Los criterios de complejidad son múltiples, para caracterizar un almacén es frecuente hacer referencia al volumen de stock que alojarán, existen almacenes grandes cuyo diseño y administración resulta muy fácil, porque recogen referencia y dimensiones homogéneas.

Una gestión del almacén es el conjunto de acciones, actividades, planes, política, normas, registros, procedimiento y métodos, con el objetivo de prevenir posibles riesgos que afecte los principios del almacén (De Contabilidad et al., 2020).

Es un medio que permite a las empresas que la integran, gestionar menores niveles de inventarios y efectuar sus actividades en tiempos de respuesta más cortos, intentando mejorar la satisfacción de los clientes y aumentar la eficiencia operacional. Así mismo es necesario indicar que los costos son parte importante en la gestión de almacenes. La gestión de almacenes, representa una porción significativa de los gastos y costos de una compañía, de allí, la presión de estas en diseñarla, administrarla y controlarla de forma eficiente y productiva (Cano Josías Caleb Pág & Cano Josías Caleb, 2019).

Hoy en día, las empresas modernas tienen muchos problemas con el procedimiento para una adecuada gestión de almacén; porque en la medida en que crecen las organizaciones, mayores son los inconvenientes que éstas puedan tener, es por ello que diariamente se requieren de mejores sistemas de planeación y procedimiento, que se apliquen a la gran diversidad de situaciones operativas de las empresas.

La gestión de almacenes, es un proceso clave que busca regular los flujos entre la oferta y la demanda, optimizar los costos de distribución y satisfacer los requerimientos de ciertos procesos productivos. En este sentido, describen que la gestión de almacenes contribuye a una efectiva gestión de la cadena de suministros por estar directamente implicada en el intercambio de información y bienes, entre proveedores y clientes, incluyendo fabricantes, distribuidores y otras empresas que participan en el funcionamiento de la cadena de suministro (Marín, 2018).

La gestión de almacenes, es considerada el eslabón más importante en la cadena logística, dado que permite mantener los productos de la empresa cerca a los distintos mercados objetivo, así como también mantiene y entrega estos productos al cliente, tal como fueron recibidos por el área de producción. La gestión de los inventarios se ha convertido en un elemento estratégico clave, que determina el éxito o fracaso de objetivos importantes.

Adicionalmente, el concepto de rentabilidad se ha vuelto muy importante; por tanto, una adecuada gestión de los almacenes optimiza el uso del espacio de almacenamiento, reduciendo los productos obsoletos y disminuye la cantidad de inventarios que se tiene almacenada. Para

ello, las empresas deben contar con un mínimo de productos almacenados para poder funcionar.

El objetivo de la investigación, se basa en determinar los principios para la gestión de almacenes para la Mipyme Cubamodela, teniendo en cuenta diferentes artículos e investigaciones publicadas relacionados con el tema de estudio y revelando la importancia de la Gestión de Almacén, se observa que, bajo el objeto de estudio del área en mención, la gestión de almacenes al ser una actividad importante en la logística, denota una importancia de optimizar procesos en torno a la manipulación y almacenamiento de la mercadería.

1.2 Gestión del proceso de inventarios

La gestión y el control del inventario son características cruciales para garantizar los segmentos financieros de las pequeñas y medianas empresas. Por tanto, la investigación está orientada a describir el manejo del control interno de los inventarios.

Los inventarios juegan un papel importante en las empresas del sector comercial, industrial y de servicio, dado que la permanencia de estas organizaciones en el mercado, nacional o internacional, está ligada al buen manejo y control de ellos (Restrepo et al., 2020).

Los inventarios son reconocidos como un elemento clave dentro de los factores de competitividad que toda organización debe gestionar, incluyendo las pymes. Sin embargo, la gestión y el control de los inventarios son algunas de las funciones más complejas a nivel empresarial, dado que implican establecer políticas, lineamientos y el diseño de herramientas de control para mantener existencias, con el objetivo de garantizar la fidelidad de los clientes y la protección ante incertidumbres que puedan ser afrontadas a menor costo.

Gestión de inventarios: “La administración de inventarios, es decir, la planificación y control de los inventarios para cumplir las prioridades competitivas de la organización, es un motivo importante de preocupación para todos los gerentes de todo tipo de empresas. La administración eficaz de los inventarios es esencial para realizar el pleno potencial de toda cadena de valor. Para las compañías que operan con márgenes de utilidad relativamente bajos, la mala administración de los inventarios puede perjudicar gravemente sus negocios. El desafío no radica en reducir los inventarios a su mínima expresión para abatir los costos, ni en tener inventario en exceso para satisfacer todas las demandas, sino en mantener la cantidad

adecuada para que la empresa alcance sus prioridades competitivas de la forma más eficiente posible. Este tipo de eficiencia sólo puede darse si la cantidad correcta de inventario fluye a través de la cadena de valor, que abarca a los proveedores, la empresa, los almacenes o centros de distribución y los clientes (Quesquén, 2020).

El correcto orden y administración de inventarios es uno de los principales factores que contribuyen a la gestión operativa y misional de una organización, como a la gestión de servicio al cliente; esto permite obtener un mejor desempeño a nivel de servicio y en su dinámica financiera para el área responsable de garantizar el flujo de productos. De acuerdo con lo anterior es fundamental que las organizaciones cuenten con las herramientas necesarias para lograr mantener un óptimo sistema de inventarios (Andres Hernández et al., 2021).

La aplicación de un Sistema de Gestión de Inventarios es una de las alternativas más influyentes en el esfuerzo por reducir los costos y mejorar la eficiencia económica, ya que incrementa los niveles de servicio al cliente, aumenta la liquidez y permite a las organizaciones estar prevenidas frente a las fluctuaciones de la demanda; manteniendo un óptimo nivel de seguridad y logrando mantener los inventarios necesarios del producto (De Contabilidad et al., 2020).

1.3 Sistema económico

La economía global tiene como característica vital el ser un sistema libre de barreras o fronteras tanto en el tiempo como en el espacio. La economía globalizada contribuye a que las personas dispongan de muchas alternativas de consumo de bienes como de servicios. Asimismo, se destaca que el precio de los productos y servicios son accesibles en muchas circunstancias debido a que las empresas producen bajo la denominada tendencia de la economía a gran escala de manera que se logra gran eficiencia a favor de los consumidores. La economía de cada país está impulsada por un conjunto de empresas tanto públicas como privadas, las cuales generan movimiento y desarrollo económico (Julio Quintana, 2021).

La transformación de la economía y la producción a escala mundial generan un entorno de competencia perfecta añadiéndose a esta situación la volatilidad de los cambios; aspectos que no permiten que las PYMES reaccionen menos que puedan crear ventajas competitivas estables. Ante estas circunstancias, ellas deben establecer estrategias y formas de administrar para lograr por lo menos su permanencia en los mercados. Es en este sentido, que existe la

necesidad de proponer una postura sobre la importancia de los modelos de gestión empresarial para las organizaciones de manera que puedan enfrentar una situación cambiante y superarla como también trabajar con eficiencia, eficacia y por supuesto lograr ser competitivas en el sector en el que participan.

La mercadotecnia ha estado presente desde muchos años atrás. Con el paso del tiempo se ha evidenciado que el buen manejo de esta rama causa un impulso económico en los comercios. Sin embargo, no todos están preparados para resistir un gasto de tal magnitud para competir con las campañas publicitarias de las grandes marcas (Montserrat et al., 2020).

La economía empresarial es un campo de la economía aplicada que utiliza la teoría económica y métodos cuantitativos para planificar e implementar prácticas comerciales. Una teoría económica incluye diferentes técnicas asociadas con el costo, la oferta y la demanda y el consumo para analizar todos los aspectos de una situación comercial. Los economistas de negocios abarcan la investigación de operaciones, la teoría de la toma de decisiones, las matemáticas y el campo de estudio de las estadísticas (affden, 2023).

Gestionar el desarrollo de economías locales de manera exitosa es una tarea que involucra aplicar herramientas administrativas, contables y estratégicas que permitan una mayor competitividad de los actores esenciales de la dinámica socio económica como son las empresas o negocios que generan ingresos a sus dueños y brindan oportunidades de empleo para los pobladores (Flores Dávila et al., 2019).

En todo sistema contable deben existir técnicas de revisión documental de los datos de análisis fundamentado que, reside en examinar e interpretar los datos de los estados financieros para obtener la indagación selecta, así como la estimación de los cálculos de la liquidez y sus respectivos dimensiones e indicadores (Gutiérrez Janampa & Tapia, 2020). De esa forma se manejará la compilación de datos, marcando que los datos recogidos vigorizan e informan el escenario problemático de la investigación. Como instrumento para la medición del análisis de ratios financieros se usarán gráficos de barra en el programa, en el que las variables formarán la organización para la recolección de datos de modo semanal, mostrando en los estados financieros de la empresa.

1.4 Gestión empresarial

La gestión es un término importante, engloba todo el acervo de conocimientos sobre la ciencia administrativa ya que mediante ella se desarrolla la acción no sólo de administrar y cumplir las cuatro fases del proceso administrativo que determinó Henry Fayol, abarca otros elementos para lograr los objetivos propuestos, incorpora la estrategia, la mirada al futuro mediante la visión y misión, debe entender el presente con las influencias del entorno y el contexto, debe proyectar diversos escenarios futuros entre los más importantes aspectos. Este despliegue no sólo de conocimientos sino de la acción holística se la entiende como gerenciamiento o management, término que muestra el desempeño, conocimiento, experticia de los que dirigen la empresa y deben contar con estas principales virtudes para ser un profesional de negocios y de empresas (Julio Quintana, 2021).

El concepto de gestión empresarial, ha evolucionado en la medida que el hombre ha avanzado en la consecución de nuevas tecnologías y relaciones para el mejoramiento de nuevos productos y servicios, buscando satisfacer a los clientes, también busca a través de las personas, mejorar la productividad y la competitividad de las empresas o negocios. El que está de responsable de la gestión se ve influenciado por diversos factores complejos sobre todo de carácter externo, los cuales le fuerzan a que esta acción de gestionar sea dinámica y en constante movimiento.

La gestión empresarial ya no se la considera como una actividad de carácter local ni tampoco nacional, con la evolución de la sociedad de información (TI) y conocimiento se la debe considerar ahora como un asunto de índole mundial, de manera que pueda afrontar el desarrollo de nuevas estrategias para la permanencia en el mercado nacional y proyectar la penetración en mercados internacionales.

La gestión empresarial pone énfasis, no tanto en la organización de los recursos (administración), como en la planificación de los procesos para alcanzar los objetivos de la organización. En este sentido, se la concibe a la gerencia como un cargo que maneja lo estratégico de la organización.

1.5 Diagnóstico de los módulos almacén y economía

Un sistema de módulos de almacenamiento y economía en una empresa se refiere a un enfoque organizado para gestionar el almacenamiento de productos, materias primas, inventarios y otros activos, con el objetivo de optimizar los recursos económicos y mejorar la eficiencia operativa. Este sistema permite a una empresa administrar sus activos de manera más eficaz y reducir costos innecesarios. A continuación, más detalles sobre lo que implica y lo que permite hacer:

1. **Gestión de almacenamiento:** El sistema organiza y supervisa el almacenamiento de productos y activos de la empresa. Puede realizar el seguimiento del inventario y la gestión de productos.
2. **Optimización de espacios:** Ayuda a aprovechar al máximo el espacio disponible en el almacén o instalaciones de almacenamiento. Esto implica asegurarse de que los productos se almacenan de manera eficiente y que no se desperdicia espacio.
3. **Control de inventario:** Facilita el seguimiento preciso del inventario, lo que permite a la empresa saber qué productos están disponibles, cuántos hay y cuándo es necesario reabastecer o retirar productos obsoletos.
4. **Reducción de costos:** Al optimizar el almacenamiento y el inventario, el sistema contribuye a la reducción de costos de almacenamiento, incluyendo costos asociados con espacio, personal, seguros y logística.
5. **Mejora de la eficiencia:** Al automatizar tareas relacionadas con la gestión de almacenamiento, como la recepción de productos, la preparación de pedidos y la gestión de inventarios, el sistema permite una mayor eficiencia operativa, lo que ahorra tiempo y recursos.
6. **Toma de decisiones informadas:** Proporciona información valiosa a los tomadores de decisiones de la empresa a través de análisis de datos, informes y métricas. Esto permite tomar decisiones basadas en datos sólidos para optimizar la cadena de suministro y las operaciones.

7. **Planificación de la demanda:** Al predecir y planificar la demanda de productos con mayor precisión, se puede reducir la necesidad de almacenar grandes cantidades de inventario y, por lo tanto, se disminuyen los costos asociados al almacenamiento.
8. **Negociación con proveedores:** Un sistema de almacenamiento y economía puede facilitar la negociación con proveedores para obtener mejores condiciones en términos de plazos de entrega, descuentos por cantidad y acuerdos de envío que reduzcan los costos de adquisición.
9. **Análisis de costos y rentabilidad:** El sistema permite realizar un seguimiento detallado de los costos relacionados con el almacenamiento y la logística. Esto incluye la capacidad de calcular el costo de almacenamiento por producto y evaluar la rentabilidad de diferentes líneas de productos o SKU (Unidad de Mantenimiento de Inventario).
10. **Minimización de pérdidas y deterioro:** La economía también se relaciona con la reducción de pérdidas debido al deterioro o vencimiento de productos, lo que puede ocurrir si los productos se almacenan inadecuadamente.

En resumen, un sistema de módulos de almacenamiento y economía en una empresa es una solución que permite gestionar de manera más efectiva el almacenamiento de productos y activos, optimizar el espacio, controlar el inventario y reducir costos innecesarios. Al hacerlo, mejora la eficiencia operativa y contribuye a un uso más eficiente de los recursos económicos de la empresa. Esto es especialmente relevante en empresas que manejan grandes cantidades de inventario o que tienen operaciones logísticas complejas.

1.6 Estudio del arte de sistemas homólogos

A raíz del proyecto a desarrollar en la presente investigación se decide conocer acerca del funcionamiento de sistemas similares, de los cuales se puedan tener en cuenta aspectos tanto positivos como negativos para el desarrollo de la gestión empresarial de Cubamodela. Al examinar sistemas similares, se identifican las mejores prácticas y soluciones que se han implementado con éxito en proyectos anteriores. Esto permite aprender de la experiencia de otros y aplicar en el proyecto las soluciones que han demostrado ser eficaces. También ayuda a identificar errores comunes o desafíos que otros han enfrentado en proyectos similares. Se puede tomar medidas proactivas para evitar cometer los mismos errores y superar los

obstáculos de manera más eficiente. Al conocer sistemas similares, se puede obtener inspiración para la innovación. observar cómo otros han abordado problemas similares puede estimular nuevas ideas y enfoques creativos para el proyecto a desarrollar. Al aprender de lo que ya se ha hecho, se puede acortar el proceso de desarrollo al evitar la necesidad de reinventar la rueda. Esto puede ahorrarte tiempo y recursos, lo que es especialmente valioso en proyectos de desarrollo de software. Teniendo en cuenta que, aunque los sistemas similares pueden servir como punto de partida, también se puede identificar cómo adaptar sus características y funcionalidades a los requisitos específicos del presente proyecto. Esto brinda una base sólida sobre la cual construir y al mismo tiempo permitiendo la personalización. En los siguientes subepígrafos se realiza el análisis de sistemas homólogos tanto en el ámbito nacional como internacional.

1.6.1 Sistemas homólogos en el ámbito nacional e internacional

En el ámbito nacional se decide estudiar el **Sistema de Gestión Empresarial (SIGE)** donde se ve que es un sistema modular de gestión empresarial desarrollado por la empresa cubana DESOFT. Este sistema incluye un módulo de gestión de almacenes y un módulo de gestión financiera y contable. El principal aspecto que los autores tuvieron en cuenta es que SIGE es utilizado por varias empresas en Cuba y es compatible con las regulaciones y normas contables y fiscales del país. También se estudió el **Sistema de Control de Almacenes (SCA)** el cual es una solución de gestión de almacenes desarrollada por la empresa cubana DATYS. Se ve que SCA permite a las empresas gestionar el inventario, el movimiento de mercancías y la optimización del espacio de almacenamiento. Además, el sistema incluye herramientas de análisis y reportes para la toma de decisiones, los cuales serían elementos a tomar en cuenta para incluirlo en el desarrollo del proyecto. Por otra parte, se realizó el estudio del **Sistema Integrado de Gestión Empresarial (SIGE) que** es un sistema modular de gestión empresarial desarrollado por la empresa cubana CLAXON. Se conoce que este sistema incluye un módulo de gestión de almacenes y un módulo de gestión financiera y contable y que es utilizado por varias empresas en Cuba y es compatible con las regulaciones y normas contables y fiscales del país, viéndose analizado este elemento los autores deciden tener en cuenta este aspecto. Se considera importante destacar que estos sistemas homólogos cubanos pueden tener características específicas y adaptaciones a la normativa y regulaciones locales, por lo que se

recomienda analizar detalladamente las funcionalidades y requerimientos específicos de la empresa antes de tomar una decisión de implementación.

Tabla 1: Sistemas homólogos en el ámbito Nacional. Fuente: Elaboración propia

Parámetros	Sistema de Gestión Empresarial (SIGE)	Sistema de Control de Almacenes (SCA)	Sistema Integrado de Gestión Empresarial (SIGE)
Tipo de Software Libre	No	No	No
Licencia de Pago	Si	Si	Si
Impacto en el Negocio	Alto	Medio	Alto
Escalabilidad	Si	Si	Si
Flexibilidad	Si	Si	Si

En el ámbito extranjero, existen varios sistemas homólogos de gestión de almacén y economía en el mercado. Se realizó un estudio para analizar algunos de estos sistemas para tener en cuenta sus funcionalidades a la hora del desarrollo del sistema a implementar. Para iniciar se realizó el estudio del sistema **SAP Warehouse Management (WM)** el cual es un tipo de software enfocado a las empresas e instituciones para la gestión de proyectos y procesos, se pudo analizar que actualmente el software ERP está enfocado a las medianas y grandes empresas pero, desde hace unos años, se ha estado implantando en el sector público, tanto en Europa como en España, viéndose como esta gestión de las Administraciones Públicas, es un software que en base a las debilidades de las empresas se adapta a la situación para darle solución a los problemas existentes, de ahí un aspecto a tener en cuenta ya que el presente sistema a desarrollar tendría ese principal aspecto ya que en base a los problemas presentados por la Mypime a través de la automatización dará una vía de solución (Capó Terrer, 2022). Se realizó el estudio del sistema **Oracle Warehouse Management Cloud (WMS)** el cual es una solución de gestión de almacén basada en la nube que permite controlar y optimizar la gestión de inventarios y operaciones de almacén. Se pudo analizar que con Oracle WMS, se puede gestionar el inventario en tiempo real, el movimiento de mercancías y la optimización del espacio de almacenamiento, al realizar el estudio de este sistema y ver estas funcionalidades los autores se percataron que era una de las principales funcionalidades que también existirían en el

proyecto a desarrollar. La solución de gestión de almacenes de Oracle combina las ventajas de la nube con la funcionalidad de gestión de almacenes de nivel empresarial (Oracle, 2023). También se realizó el estudio del sistema **Microsoft Dynamics 365 Supply Chain Management** el cual es una solución de administración empresarial para pequeñas y medianas empresas que automatiza y optimiza los procesos empresariales y le ayuda a administrar su empresa. Se analizó que gracias a su gran adaptabilidad y variedad de funciones, Business Central permite a las empresas administrar su negocio, incluidas las áreas de finanzas, fabricación, ventas, envíos, administración de proyectos, servicios, entre otras funcionalidades (Microsoft, 2023), viendo analizado todas estas funcionalidades se pudo determinar que varias de ellas guardan relación con las definidas en el actual proyecto a desarrollar, por lo que se tendrá en cuenta este sistema para la implementación del actual proyecto.

Tabla 2: Sistemas homólogos en el ámbito Extranjero. Fuente: Elaboración propia

Parámetros	SAP Warehouse Management (WM)	Oracle Warehouse Management Cloud (WMS)	Microsoft Dynamics 365 Supply Chain Management
Tipo de Software Libre	No	No	No
Licencia de Pago	Si	Si	Si
Impacto en el Negocio	Alto	Alto	Alto
Escalabilidad	No	No	-
Flexibilidad	Si	Si	Si

Luego del estudio realizado a estos sistemas

homólogos tanto internacionales como nacionales se pudo concluir que son sistemas los cuales cuentan con funcionalidades muy similares a las que se desean desarrollar en el actual proyecto, por lo que brindan una guía de desarrollo a tener en cuenta en la implementación y aportan otra visión de desarrollo la cual contribuye en el desarrollo del mismo. También se pudo analizar la diferencia en la forma de desarrollo, ya que las empresas internacionales, como SAP, Oracle y Microsoft, suelen estar diseñadas para operaciones a gran escala y empresas con una

presencia global. Tienen características y funcionalidades específicas para manejar operaciones en múltiples ubicaciones, países y con volúmenes significativamente grandes de datos y transacciones. Los sistemas nacionales pueden tener un enfoque más regional o local y, por lo tanto, no requieren la misma complejidad. Los sistemas internacionales deben cumplir con regulaciones de múltiples jurisdicciones, lo que influye en la flexibilidad de sus sistemas. Los sistemas nacionales pueden tener objetivos y estrategias empresariales diferentes de las empresas internacionales. Esto puede influir en la elección de sistemas que se adapten mejor a sus necesidades específicas, como el control de costos, la optimización de procesos locales o la personalización de soluciones.

1.7 Tecnologías y herramientas

Las tecnologías y herramientas para el desarrollo de aplicaciones web son muy importantes, ya que las mismas poseen diversas características y ventajas que ayudan en el proceso de desarrollo. Por eso a continuación se hace una breve descripción de cada una de estas que son las que se utilizarán en la presente investigación

El desarrollo de aplicaciones web se conforma de la capa de presentación (FrontEnd) y una capa de acceso a datos (BackEnd). El FrontEnd trabaja la interfaz visual, y hace que el usuario pueda interactuar con nuestro sitio o sistema. Está orientado al lenguaje de marcas y al lenguaje de programación web de ejecución en equipos clientes. El Backend se encarga de la manipulación de los datos, un Backend no sirve de mucho si no existe un FrontEnd de por medio, el desarrollador Backend debe de conocer de bases de datos, frameworks y aspectos de seguridad. Él debe encargarse de que la información que llega desde el FrontEnd, sea almacenada en una base de datos. Así mismo se encarga de crear API's para que sus datos puedan consumirse de manera cómoda y pueda mejorar la experiencia del usuario. El avance tecnológico obliga al profesional de la informática, dedicado al desarrollo web, a conocer estos términos y las tecnologías relacionadas. El presente trabajo pretende centrarse en el estudio y análisis de estas tecnologías, con lo cual se pretende determinar cuáles son las más usadas y relevantes en el ámbito laboral, y en razón de ello, fomentar o motivar la incorporación de sus conocimientos en la formación de los ingenieros que están cursando los últimos años de las carreras de informática (Graciela et al., 2021).

1.7.1 Lenguaje de programación interfaz usuario

JavaScript: JavaScript es un lenguaje de programación o de secuencias de comandos que permite implementar funciones complejas en páginas web. JavaScript (abreviado comúnmente JS) es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico. Se utiliza principalmente del lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas y JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo, en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo. JavaScript es un lenguaje creado por Netscape, la compañía propietaria de un navegador con el mismo nombre hoy desaparecido, que fue precursor del actual Firefox (Germain Ramírez & Cristal Esmeralda, 2020).

JavaScript ofrece ventajas como:

- Es un lenguaje muy sencillo y fácil de aprender e implementar.
- Es rápido, por lo que ejecuta las funciones inmediatamente y favorece la detección de errores.
- Es versátil, ya que es compatible con otros lenguajes, con los navegadores más populares y con los dispositivos más modernos.
- Cuenta con múltiples opciones de efectos visuales y permite desarrollar páginas dinámicas y aplicaciones web.
- Se puede usar para crear relojes, contadores, calculadoras, validaciones, chats, cajas de texto y mucho más.

JavaScript es un lenguaje de programación muy popular y ampliamente utilizado en el desarrollo web, el cual permite crear páginas web interactivas y dinámicas. Permitirá agregar efectos visuales, animaciones y elementos interactivos a tus páginas para mejorar la experiencia del usuario. También es compatible con todos los navegadores web modernos, por lo que permitirá funcionar en la mayoría de los dispositivos y plataformas sin problemas. También tiene una gran comunidad de desarrolladores en línea que comparten recursos, bibliotecas y frameworks, permitiendo encontrar fácilmente ayuda, ejemplos de código y soluciones a problemas comunes mientras se codifica. El uso de JavaScript permitirá aprender y aplicar buenas prácticas de

desarrollo web moderno, como el uso de patrones de diseño, modularidad y optimización del rendimiento.

1.7.2 Framework de desarrollo interfaz usuario

Un framework de desarrollo es un software que proporciona una estructura y unas herramientas para facilitar el desarrollo de aplicaciones web. Un framework permite reutilizar código, seguir buenas prácticas, mejorar la seguridad y la productividad, y crear interfaces de usuario más atractivas. Existen diferentes tipos de frameworks según el lenguaje de programación que utilizan, como Python, Java, PHP, Ruby, entre otros. A la hora de desarrollar una aplicación o una página web hay que tener en cuenta algunos elementos que forman parte del mismo desarrollo, en este caso de los frameworks que pueden ser utilizados en la parte “interfaz” a la que se dedica el frontend (Vázquez del Cerro, 2020).

React es una biblioteca de JavaScript para construir interfaces de usuario. React te permite construir interfaces de usuario a partir de piezas individuales llamadas componentes. Crea tus propios componentes de React como Thumbnail, LikeButton, y Video. Luego combinarlos para formar pantallas, páginas y aplicaciones. Los componentes de React son funciones de JavaScript (react, 2023). Por tanto, React representa una base sólida sobre la cual se puede construir casi cualquier cosa con JavaScript. Además, facilita mucho el desarrollo, ya que nos ofrece muchas cosas ya listas, en las que no necesitamos invertir tiempo de trabajo (react, 2023).

También **Angular** es un framework para aplicaciones web desarrollado en TypeScript, de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web de una sola página, proporciona una potente interfaz de línea de comandos para realizar tareas comunes de desarrollo de aplicaciones angulares. La CLI angular implementa la guía de estilo Angular para admitir la coherencia entre los componentes. El comando `ng generate` y los esquemas angulares mantienen el código fuente de tu aplicación coherente y organizado, al tiempo que generan código repetitivo para ayudarte a iniciar tu aplicación. También incluye soporte para formularios y rutas. Estas características se demuestran diariamente en miles de aplicaciones dentro de Google (ANGULAR, 2023). Cuenta también con una productividad de plantillas, angular CLI e IDEs.

Por otro lado, está **Vue.js** es un framework de JavaScript de código abierto para la construcción de interfaces de usuario y aplicaciones de una sola página. Es un marco accesible, eficiente y versátil para crear interfaces de usuario web, se basa en HTML estándar, CSS y JavaScript con API intuitiva y documentación de clase mundial, cuenta con un sistema de renderizado verdaderamente reactivo y optimizado para compiladores que rara vez requiere optimización manual, también con un ecosistema rico y gradualmente adoptable que escala entre una biblioteca y un marco con todas las funciones (Vue.js, 2023).

Tabla 3: Características generales de las tecnologías estudiadas. Fuente: Elaboración propia

Criterios Tecnologías	Curva de aprendizaje	Flexibilidad	Tipo	Lenguaje	Rendimiento
React.js	Media	Alto	Biblioteca	JavaScript	Alto
ANGULAR	Alta	Medio	Framework	JavaScript	Medio
Vue.js	Baja	Medio	Framework	JavaScript	Alto

Para el desarrollo de los módulos Almacén y Economía se utilizará React ya que cuenta con variadas facilidades como es permitir crear páginas web interactivas y dinámicas, permite agregar efectos visuales, animaciones y elementos interactivos. También por su compatibilidad con todos los navegadores web modernos lo que significa que el proyecto funcionará en la mayoría de los dispositivos y plataformas sin problemas, al igual que con su gran comunidad de desarrolladores en línea que comparten recursos, bibliotecas y frameworks permitirá encontrar fácilmente ayuda, ejemplos de código y soluciones a problemas comunes mientras se trabaja en el proyecto.

1.7.3 Lenguaje de programación para el servidor

PHP: es un lenguaje de programación que permite el desarrollo web o aplicaciones web dinámicas, el cual es apto para incrustar el lenguaje HTML. PHP puede emplearse en todos los sistemas operativos principales, incluyendo Linux, muchas variantes de Unix (incluyendo HP-UX, Solaris y OpenBSD), Microsoft Windows, macOS, RISC OS y probablemente otros más.

PHP admite la mayoría de servidores web de hoy en día, incluyendo Apache, IIS, y muchos otros. Una de las características más potentes y destacables de PHP es su soporte para un amplio abanico de bases de datos. Escribir una página web con acceso a una base de datos es increíblemente simple utilizando una de las extensiones específicas de bases de datos (p.ej., para MySQL), o utilizar una capa de abstracción como PDO, o conectarse a cualquier base de datos que admita el estándar de Conexión Abierta a Bases de Datos por medio de la extensión (Vincent Domenech, 2021).

Java: Java es una plataforma informática de lenguaje de programación. Aunque la mayoría de aplicaciones Java modernas combinan el tiempo de ejecución y la aplicación de Java, todavía existen algunas aplicaciones e incluso sitios web que no funcionan sin instalar Java para escritorio. El sitio web Java.com está pensado para consumidores que todavía necesitan Java en sus aplicaciones de escritorio, sobre todo las aplicaciones que tienen como destino Java 8. Los desarrolladores y los usuarios que quieran aprender sobre programación Java, deben consultar el sitio web dev.java y los usuarios profesionales, oracle.com/java para obtener más información (Java, 2023).

Python: es un lenguaje de programación interpretado cuya filosofía hace hincapié en la legibilidad de su código. Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, dinámico y multiplataforma. Es administrado por la Python Software Foundation. Posee una licencia de código abierto (Germain Ramírez & Cristal Esmeralda, 2020). Python es un lenguaje de scripting independiente de plataforma y orientado a objetos, preparado para realizar cualquier tipo de programa, desde aplicaciones Windows a servidores de red o incluso, páginas web. Es un lenguaje interpretado, lo que significa que no se necesita compilar el código fuente para poder ejecutarlo, lo que ofrece ventajas como la rapidez de desarrollo. En los últimos años el lenguaje se ha hecho muy popular,⁹ gracias a varias razones como: (Germain Ramírez & Cristal Esmeralda, 2020).

Selección del lenguaje de programación para el servidor:

Después de haber expuesto las características de los lenguajes mencionados se ha decidido utilizar Python, ya que es el lenguaje que, por su sintaxis clara y legible, facilita la lectura y el mantenimiento del código. También por su amplia variedad de aplicaciones, desde desarrollo

web hasta análisis de datos, aprendizaje automático. Al igual que tiene una biblioteca estándar muy completa que incluye módulos para tareas comunes, como manipulación de archivos, acceso a bases de datos, procesamiento de texto y más, lo que permite ahorrar tiempo y esfuerzo al no tener que escribir código desde cero para muchas tareas comunes. Debido a estos aspectos se decidió emplear el lenguaje Python.

1.7.4 Framework de desarrollo para el servidor

Django: Django es un marco web Python de alto nivel que fomenta el desarrollo rápido y el diseño limpio y pragmático. Es gratis y de código abierto, fue diseñado para ayudar a los desarrolladores a llevar las aplicaciones desde el concepto hasta su finalización lo más rápido posible, se toma en serio la seguridad y ayuda a los desarrolladores a evitar muchos errores de seguridad comunes, Algunos de los sitios más concurridos de la web aprovechan la capacidad de Django para escalar de manera rápida y flexible (Django, 2023). Django hace que sea más fácil crear mejores aplicaciones web más rápidamente y con menos código. La seguridad es un tema de suma importancia en el desarrollo de la web aplicaciones y Django proporciona múltiples herramientas y mecanismos de protección (Django, 2023).

Hay una variedad de técnicas y herramientas que pueden ayudar a que su código se ejecute. Más eficientemente, más rápido y utilizando menos recursos del sistema, ofrece un sólido marco de internacionalización y localización para ayudarle en el desarrollo de aplicaciones para múltiples idiomas y mundo Regiones, proporciona un rico marco para facilitar la creación de formularios y Manipulación de datos de formularios (Django, 2023).

Flask: Flask es un framework web escrito en Python que te permite crear aplicaciones web de forma rápida y sencilla. Algunas de sus principales ventajas son:

- Es un micro framework, lo que significa que sólo incluye las herramientas básicas para el desarrollo web, pero se puede ampliar con muchas extensiones disponibles.
- Tiene un servidor web integrado que te permite probar tus aplicaciones sin necesidad de instalar nada más.
- Tiene un depurador integrado que te ayuda a encontrar y corregir los errores en tu código.
- Es compatible con Python 3, uno de los lenguajes de programación más populares y potentes del mundo.

- Soporta cookies seguras, lo que te permite almacenar información en el navegador del usuario de forma cifrada (Flask, 2023).

FastAPI: FastAPI es un web framework moderno y rápido (de alto rendimiento) para construir APIs con Python 3.6+ basado en las anotaciones de tipos estándar de Python, es un marco web de Python para crear API Rest y aplicaciones web rápidas. Fue creado por Sebastian Ramirez, es de código abierto y se encuentra en Github. Es usado por empresas como Uber, Windows, Netflix y Office

Sus características principales son:

- **Rapidez:** Alto rendimiento, a la par con **NodeJS** y **Go** (gracias a Starlette y Pydantic). Uno de los frameworks de Python más rápidos.
- **Rápido de programar:** Incrementa la velocidad de desarrollo entre 200% y 300%. *
- **Menos errores:** Reduce los errores humanos (de programador) aproximadamente un 40%. *
- **Intuitivo:** Gran soporte en los editores con auto completado en todas partes. Gasta menos tiempo debugging.
- **Fácil:** Está diseñado para ser fácil de usar y aprender. Gastando menos tiempo leyendo documentación.
- **Corto:** Minimiza la duplicación de código. Múltiples funcionalidades con cada declaración de parámetros. Menos errores.
- **Robusto:** Crea código listo para producción con documentación automática interactiva.
- **Basado en estándares:** Basado y totalmente compatible con los estándares abiertos para APIs: OpenAPI (conocido previamente como Swagger) y JSON Schema.

Selección del framework de programación para el servidor:

Basándonos en el estudio de estos framework de desarrollos para el lenguaje de programación Python se utilizará Django ya que por sus características es el que mejor se adecua a las tecnologías a utilizar. Los autores se basaron en su rapidez en el desarrollo, ya que proporciona

una estructura y una serie de herramientas que facilitan el desarrollo web. Al igual que fomenta un diseño limpio y pragmático para tus aplicaciones web, lo que facilita la organización y el mantenimiento del código. Y por otro lado teniendo en cuenta su escalabilidad, por proporcionar herramientas y técnicas para optimizar el rendimiento y manejar grandes volúmenes de tráfico. Teniendo en cuenta estos aspectos se decide trabajar con Django.

1.7.5 Entorno de Desarrollo Integrado (IDE)

Visual Studio Code: es un editor de código fuente desarrollado por Microsoft para Windows, Linux y macOS. Incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código. También es personalizable, por lo que los usuarios pueden cambiar el tema del editor, los atajos de teclado y las preferencias. Es gratuito y de código abierto, aunque la descarga oficial está bajo software privativo e incluye características personalizadas por Microsoft.

Entre las características que podremos agregar en Visual Studio Code, se encuentran desde herramientas de Debug hasta opciones para actualización en tiempo real de nuestro código en la vista del navegador y compilación en vivo de los lenguajes que lo requieran (por ejemplo, en el caso de SASS a CSS) (Damián De Luca, 2023).

Un aspecto fundamental de un editor de código es que podamos utilizarlo con los lenguajes de programación que trabajamos a diario. Visual Studio Code es una herramienta que tiene soporte nativo para gran variedad de lenguajes, entre ellos podemos destacar los principales del desarrollo Web: HTML, CSS, y JavaScript, entre otros.

Otra ventaja interesante es la posibilidad de configurar la vista a nuestro gusto. De esta forma, podremos tener más de un código visible al mismo tiempo, las carpetas de nuestro proyecto y también acceso a la terminal o un detalle de problemas, entre otras posibilidades (Damián De Luca, 2023).

1.7.6 Gestores de bases de datos

MySQL: MySQL es un sistema gestor de bases de datos (SGBD, DBMS por sus siglas en inglés) muy conocido y ampliamente usado por su simplicidad y notable rendimiento. Aunque carece de algunas características avanzadas disponibles en otros SGBD del mercado, es una opción atractiva tanto para aplicaciones comerciales, como de entretenimiento precisamente

por su facilidad de uso y tiempo reducido de puesta en marcha. Esto y su libre distribución en Internet bajo licencia GPL le otorgan como beneficios adicionales (no menos importantes) contar con un alto grado de estabilidad y un rápido desarrollo.

MySQL está disponible para múltiples plataformas, la seleccionada para los ejemplos de este libro es GNU/Linux. Sin embargo, las diferencias con cualquier otra plataforma son prácticamente nulas, ya que la herramienta utilizada en este caso es el cliente mysql-client, que permite interactuar con un servidor MySQL (local o remoto) en modo texto. De este modo es posible realizar todos los ejercicios sobre un servidor instalado localmente o, a través de Internet, sobre un servidor remoto.

Para la realización de todas las actividades, es imprescindible que dispongamos de los datos de acceso del usuario administrador de la base de datos. Aunque en algunos de ellos los privilegios necesarios serán menores, para los capítulos que tratan la administración del SGBD será imprescindible disponer de las credenciales de administrador (Alberto et al., 2014).

1.8 Metodología de Desarrollo de Software

Scrum: Es un marco de trabajo ágil que incluye entregas rápidas y un alto poder de decisión del equipo. Es ideal para proyectos de desarrollo de software, marketing y de otras áreas donde se requiere llegar a objetivos en el menor tiempo posible, con una gran flexibilidad para satisfacer las necesidades del cliente (Ramírez Ramírez et al., 2018).

El uso de metodología SCRUM en cualquier proyecto, ofrece ventajas como la adaptabilidad, esto permite que en el proyecto se incorpore el cambio, la transparencia, es posible utilizar herramientas que muestran los avances del proyecto como el Scrumboard y Sprint Burndown chart, esto propicia un ambiente abierto, retroalimentación continua, la cual es posible mediante Conduct Daily StandUp y Demonstrate and validate Sprint, el proyecto se divide en entregables, los cuales es posible evaluar y mejorar de manera progresiva (Ramírez Ramírez et al., 2018). Vemos que un Equipo Scrum (Scrum Teams) está conformados de seis a diez miembros. Se desarrolló un proyecto en un ambiente de introspección, aprendizaje y capacidad de adaptación que lleva a un entorno de trabajo innovador y creativo (Ramírez Ramírez et al., 2018).

OUP: OpenUP es un proceso de desarrollo de software mínimamente suficiente, esto quiere decir que incluye solo el contenido fundamental, esto es que no provee orientación sobre temas

en los que el proyecto tiene que lidiar, como son: el tamaño del equipo, el cumplimiento, seguridad, orientación tecnológica entre otras. Sin embargo, OpenUP es completa en el sentido de que manifiesta por completo el proceso de construir un sistema. Para atender las necesidades que no están cubiertas en su contenido OpenUp es extensible a ser utilizado como base sobre la cual se pueden añadir o adaptarse a contenido de otro proceso que sea necesario.

Open UP adopta un enfoque pragmático, con una filosofía ágil que se centra en la naturaleza colaborativa de desarrollo de software. Es una herramienta agnóstica, procesos de baja formalidad que puede ser usado tal cual o ampliarse para hacer frente a una amplia variedad de proyectos.

Características de Open UP

- Desarrollo incremental.
- Uso de casos de uso y escenarios.
- Manejo de riesgos.
- Diseño basado en la arquitectura

Metodología XP: La metodología extreme programming, es una metodología ágil. Los valores y principios que sigue este modelo son sus principales características como es la comunicación tal como se mencionó anteriormente, uno de los fundamentos principales de esta metodología es la comunicación asertiva entre los involucrados del proyecto y, por ende, el trabajo en equipo generando un buen ambiente laboral. La simplicidad como es característico de las metodologías ágiles lo más importante son los deseos del cliente y sus prioridades, desarrollándolos de la manera más sencilla posible. La retroalimentación la comunicación permite la construcción conjunta del proyecto y de todo el equipo en las dos direcciones de cliente hacia los desarrolladores del proyecto y viceversa (Salazar et al., 2018). Esta metodología realiza el proceso denominado Planning game, que define la fecha de cumplimiento y el alcance de una entrega funcional, el cliente define las historias de usuario y el desarrollador con base en ellas establece las características de la entrega, costos de implementación y número de interacciones para terminarla (Vite Cevallos et al., n.d.).

A diferencia de Scrum o Kanban, el XP no es un método generalista y tiene unas características muy concretas. A continuación, detallamos cómo funciona esta metodología:

- No hay un Project manager ni jerarquías verticales tradicionales.
- Habilita un entorno con reglas sencillas, buscando la simplicidad para conseguir ser más efectivo.
- Se basa en la autoorganización para resolver problemas específicos de la manera más eficiente.
- Se trabaja en períodos semanales, siguiendo un círculo de tareas a realizar.

Selección de la metodología de desarrollo:

Para la elección de la forma de trabajo se analizaron alternativas y se determinó trabajar en la metodología ágil XP, ya que esta metodología permitirá un mejor trabajo en pareja, al igual que desarrollar un buen proyecto en corto periodo de tiempo, debido a que permite la presencia del cliente en la definición de las historias de usuarios, permitirá entregar un producto con mayor aceptación. Al trazar períodos de trabajos semanales permitirá tener mayor organización y desempeño en la implementación, facilitando y haciendo más eficiente el trabajo en equipo, basándose en estos principales aspectos los autores deciden trabajar con la metodología XP.

1.9 Visual Paradigm for UML 8.0

Visual Paradigm Visual Paradigm es una herramienta CASE: Ingeniería de Software Asistida por Computación. La misma propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación.

Además de las características mencionadas anteriormente, Visual Paradigm también ofrece herramientas para el modelado de procesos de negocios, la gestión de requisitos, la ingeniería de sistemas y la gestión de proyectos. La herramienta también ofrece la posibilidad de integrarse con otras herramientas de desarrollo de software, como IDEs, control de versiones y herramientas de gestión de pruebas (Vincent Domenech, 2021).

Estaremos utilizando Visual Paradigm UML 8.0

Las principales características de Visual Paradigm UML 8.0 incluyen:

1. Soporte completo para el lenguaje UML 2.5: Visual Paradigm UML 8.0 soporta todas las características de UML 2.5, incluyendo diagramas de casos de uso, diagramas de clases, diagramas de secuencia, diagramas de actividades, diagramas de componentes y diagramas de despliegue.
2. Herramientas de modelado intuitivas: la herramienta proporciona un entorno de modelado intuitivo y fácil de usar que facilita la creación de diagramas UML.
3. Generación de código automático: Visual Paradigm UML 8.0 permite generar código automáticamente a partir de los modelos UML creados, ahorrando tiempo y reduciendo errores.
4. Integración con otras herramientas: la herramienta se integra con otras herramientas de desarrollo de software, como IDEs, control de versiones y herramientas de gestión de proyectos.
5. Colaboración en equipo: la herramienta permite la colaboración en equipo en tiempo real, lo que facilita la comunicación y el trabajo en equipo.
6. Personalización: Visual Paradigm UML 8.0 permite personalizar los modelos UML, los informes y los diagramas según las necesidades del proyecto.

En resumen, Visual Paradigm UML 8.0 es una herramienta completa y poderosa para el modelado de sistemas de software utilizando el lenguaje UML, que ofrece una variedad de características para facilitar el trabajo de los desarrolladores y analistas de sistemas.

Conclusiones parciales:

En base a la información recopilada en este capítulo, se definen las siguientes conclusiones:

- Se decide realizar un sistema de gestión eficiente de almacenes y de procesos económicos para optimizar las operaciones logísticas y reducir costos en la empresa.
- El análisis de sistemas homólogos proporciona una base sólida para el diseño de un nuevo sistema de gestión de almacenes y procesos económicos, identificando características esenciales y oportunidades de mejora.
- La elección de tecnologías como JavaScript con React y Python con Django se alinea con los requisitos del proyecto y la creación de una aplicación de gestión de almacenes eficiente y escalable. La metodología XP se presenta como un enfoque prometedor para el desarrollo de software, permitiendo la adaptabilidad y la entrega rápida de soluciones.

En resumen, el primer capítulo de la presente investigación sienta las bases para el desarrollo de un sistema de gestión de almacenes y economía eficiente y adaptable, aprovechando las mejores prácticas en la gestión de procesos y la ingeniería de software. Los conceptos y decisiones presentados en este capítulo proporcionan una sólida dirección para el desarrollo futuro de la aplicación.

Capítulo II. Características y diseño del sistema

En este capítulo, se presenta en detalle una propuesta de solución, con un enfoque en la optimización de la gestión de empresarial para la Mipyme Cubamodela. La metodología de desarrollo ágil Extreme Programming (XP) será el marco de trabajo que guiará nuestra planificación y diseño. Siguiendo las fases propuestas por XP, estableceremos un proceso estructurado y eficiente para la creación del sistema de gestión empresarial. Esto asegurará que las etapas de diseño, desarrollo y entrega se lleven a cabo de manera coordinada y con un enfoque en la mejora continua. Las Historias de Usuario (HU). Este capítulo establece las bases para el diseño, la planificación y la colaboración interdisciplinaria que guiarán el desarrollo del sistema de gestión de almacén y economía en Cubamodela, con un enfoque en la agilidad y la mejora continua.

2.1 Propuesta de solución

Después de saber las necesidades del sistema y seleccionar las herramientas a trabajar, se determinan los módulos almacén y economía a desarrollar para darle solución al actual problema en la MIPYME Cubamodela. Como parte del sistema de gestión empresarial el módulo almacén como se muestra en la ilustración 1 debe contar con un predespacho, inventario, registro de almacenes, gráficos de ventas por productos, productos con salidas especiales y seguridad. De conjunto el módulo economía como se muestra en la ilustración 2 debe contar con las ventas de la tienda, gráficos de ventas, administrar facturas internas, pagos y buscar facturas. En las siguientes imágenes se describen las configuraciones que deben de tener los módulos.

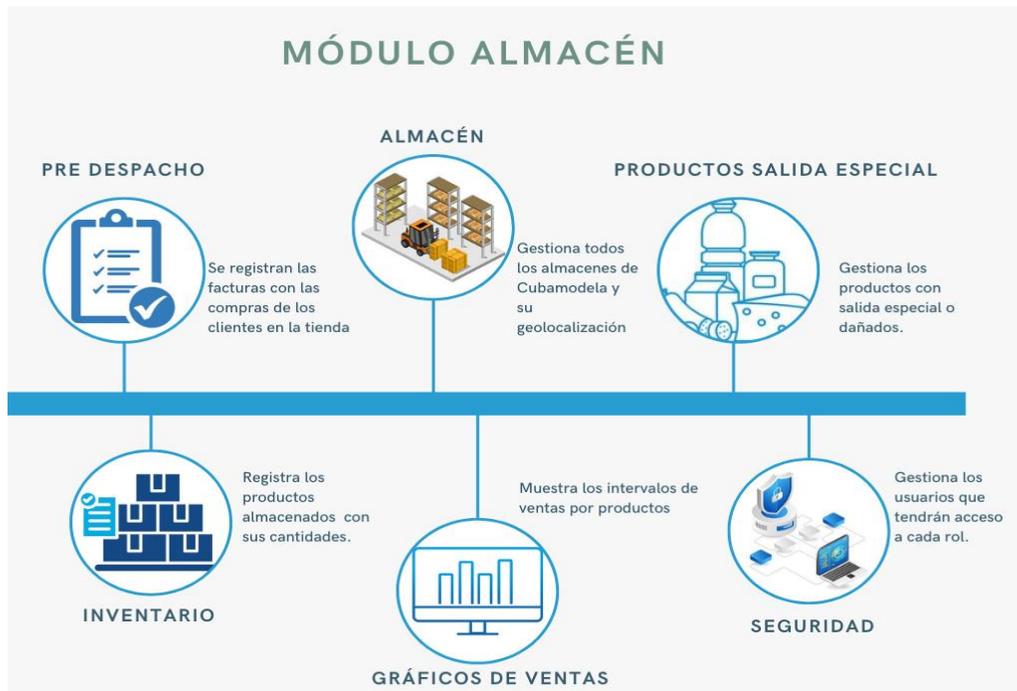


Ilustración 1. Propuesta de solución en el módulo almacén. Fuente: Elaboración propia.

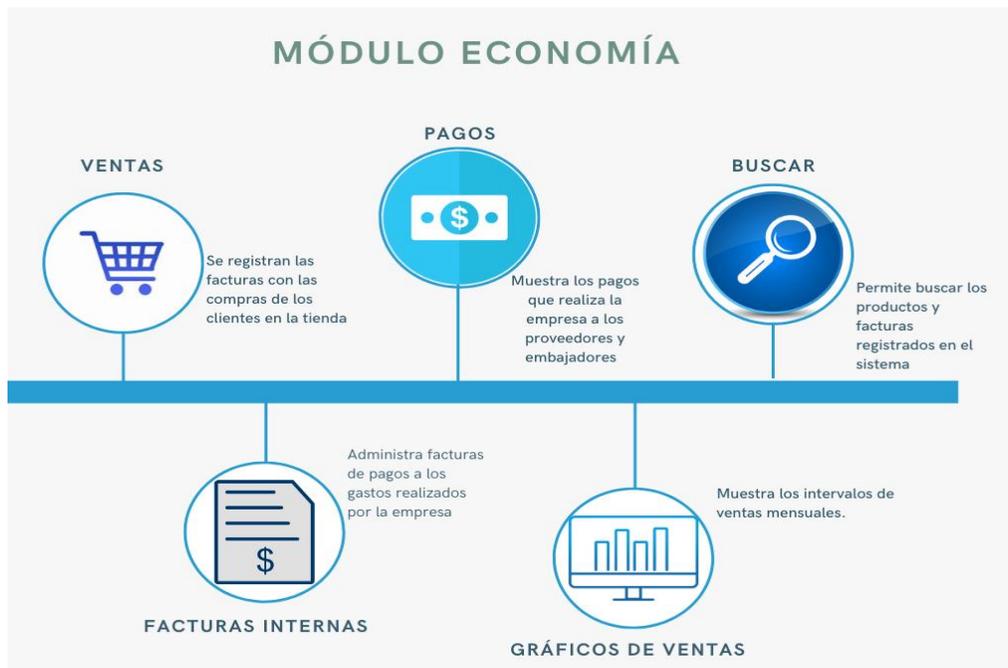


Ilustración 2. Propuesta de solución en el módulo economía. Fuente: Elaboración propia.

- **Módulo Almacén:** permitirá darle entrada a los productos que entren nuevos al almacén, al igual que darles salida en dependencia de la vía, por venta, dañados o por causa especial. De igual forma llevar la gestión del estado de cada producto después y antes de una venta, permitirá a través de un predespacho ver las facturas que se van procesando por ventas en la tienda. También permitirá gestionar los distintos almacenes de la entidad, además mostrar a través de gráficos los niveles de ventas por productos. Al igual que gestionará los usuarios que tendrán acceso a cada rol que interviene en el sistema.
- **Módulo Economía** por su parte debe gestionar todo el proceso económico de la tienda, de todos los productos vendidos por facturas, los pagos a los proveedores por producto, al igual que el pago a los embajadores. También mostrará a través de gráficos los niveles de ventas mensuales, al igual que permitirá administrar las facturas internas que realizará la directora de economía para justificar el pago de algún gasto realizado para cubrir las necesidades de la entidad.

Para el desarrollo de estos módulos se siguen las fases que establece la metodología ágil XP, Fase I implementación, Fase II Diseño del sistema, Fase III Codificación y Fase IV Pruebas.

2.1.1 Requisitos funcionales

No.	Nombre	Descripción
[RF1]	[Permitir dar salida de inventario productos dañados o con salidas especiales]	[El sistema debe darle salidas de inventario a aquellos productos dañados o con salidas especiales]
[RF2]	[Permitir añadir productos dañados o con salidas especiales a las tablas correspondientes]	[El sistema debe añadir aquellos productos dañados o con salidas especiales a las tablas correspondientes]

[RF3]	[Listar productos dañados o con salidas especiales]	[El sistema debe crear un historial con todos los productos dañados o con salidas especiales]
[RF4]	[Eliminar productos dañados o con salidas especiales]	[El sistema debe permitir eliminar productos dañados o con salidas especiales]
[RF5]	Enviar correo	[El sistema debe enviar un correo al director de la tienda, cada vez que el almacenero añada un nuevo producto con la información: Se ha añadido un nuevo producto a almacén]
[RF6]	Añadir nuevos productos.	[El sistema debe permitir al almacenero añadir un nuevo producto a almacén]
[RF7]	Modificar producto	[El sistema debe permitir al almacenero modificar los productos del almacén]
[RF8]	Listar producto	[El sistema debe realizar un listado con todos los productos de cada almacén]
[RF9]	Eliminar producto	[El sistema debe permitir al almacenero eliminar los productos del almacén]
[RF10]	[Visualizar Productos]	[El sistema debe permitir al almacenero visualizar el informe del producto con toda la información]
[RF11]	[Generar gráficos con márgenes de ventas por productos]	[El sistema debe permitir al director de la tienda y al almacenero visualizar a través de gráficos de barra ver los productos más vendidos por días, semanas, quincenas y años]

[RF12]	Añadir factura a pre despacho.	[El sistema debe permitir al almacenero añadir una nueva factura al pre despacho]
[RF13]	Modificar facturas del pre despacho	[El sistema debe permitir al almacenero modificar una factura al pre despacho]
[RF14]	Listar facturas del pre despacho	[El sistema debe permitir al almacenero visualizar una lista con las facturas del pre despacho]
[RF15]	Eliminar facturas del pre despacho	[El sistema debe permitir al almacenero eliminar una o varias facturas del pre despacho]
[RF16]	Mostrar detalles	El sistema debe permitir al almacenero y al director de la tienda visualizar los detalles específicos de entrada de los productos, con una breve descripción del estado de entrada de los productos y la fecha en que se creó el producto.
[RF17]	Generar historial de productos dañados o con salida especial	El sistema debe permitir al director de la tienda visualizar un historial con todos los productos que han salido de almacén por estar dañados y por otro lado todos los productos con una salida por causa especial.
[RF18]	Añadir nuevo usuario	El sistema debe permitir al director de la tienda añadir nuevos usuarios al sistema.
[RF19]	Modificar usuario	El sistema debe permitir al director de la tienda modificar los usuarios del sistema.
[RF20]	Eliminar usuario	El sistema debe permitir al director de la tienda eliminar usuarios del sistema.

[RF21]	Listar usuario	El sistema debe permitir al director de la tienda visualizar los usuarios del sistema.
[RF22]	Añadir nuevo almacén	El sistema debe permitir al director de la tienda añadir nuevos almacenes al sistema.
[RF23]	Modificar almacén	El sistema debe permitir al director de la tienda modificar los almacenes del sistema.
[RF24]	Eliminar almacén	El sistema debe permitir al director de la tienda eliminar un almacén del sistema.
[RF25]	Listar almacén	El sistema debe permitir al director de la tienda visualizar los almacenes del sistema.
[RF26]	Visualizar Ventas	El sistema debe permitir al director de economía visualizar todas las ventas realizadas por la tienda
[RF27]	Generar gráficos de ventas	El sistema debe permitir al director de la tienda realizar análisis a través de gráficos sobre las ventas de la tienda, permitiendo prever problemas económicos dentro de la entidad.
[RF28]	Añadir pago	El sistema debe permitir al director de economía añadir el pago quincenal de los embajadores por las compras realizada en la tienda por sus clientes y el pago a los proveedores por cada una de las ventas realizadas de sus productos
[RF29]	Modificar pago	El sistema debe permitir al director de economía modificar el pago quincenal de los embajadores por las compras realizada en la tienda por sus clientes y el pago a los

		proveedores por cada una de las ventas realizadas de sus productos
[RF30]	Eliminar pago	El sistema debe permitir al director de economía eliminar un pago quincenal de algún embajadore, al igual que el pago a los proveedores.
[RF31]	Visualizar pagos	El sistema debe permitir al director de economía visualizar los pagos quincenales de los embajadores por las compras realizada en la tienda por sus clientes y los pagos a los proveedores por cada una de las ventas realizadas de sus productos.
[RF32]	Buscar productos o facturas	El sistema debe permitir al director de economía buscar los productos o facturas de la tienda por negocio, material, tipo de producto, color, precio, medida, número de orden, número de factura y peso, permitiéndole acceder a la información de algún producto con mayor rapidez.
[RF33]	Añadir facturas internas	El sistema debe permitir al director de economía realizar facturas internas de costo con monto, área, motivo, nombre del trabajador encargado, para justificar el pago a alguna acción realizada por la misma empresa.
[RF34]	Eliminar facturas internas	El sistema debe permitir al director de economía eliminar las facturas internas del sistema.
[RF35]	Listar facturas internas	El sistema debe permitir al director de economía visualizar las facturas internas del sistema.

2.2 Fase I: Planificación

En esta fase se define el alcance general del proyecto. En esta el cliente define lo que necesita mediante la redacción de HU y establece la prioridad de cada una. Esta es la primera etapa de la metodología XP que inicia en la identificación de las historias de usuario las cuales luego se priorizan y al final de esta fase se tendrá muy claro hacia donde estará orientado el proyecto con un tiempo establecido (Library, 2023).

2.2.1. Historias de usuarios

Las historias de usuario se usan, en el contexto de la ingeniería de requisitos ágil, como una herramienta de comunicación que combina las fortalezas de ambos medios: escrito y verbal. Describen, en una o dos frases, una funcionalidad de software desde el punto de vista del usuario, con el lenguaje que éste emplearía.¹ El foco está puesto en qué necesidades o problemas soluciona lo que se va a construir. Su origen viene de la metodología eXtremeProgramming (programación extrema, abreviado normalmente como XP), donde las historias de usuario deben ser escritas por los clientes. XP fue creada por Kent Beck y descrita por primera vez en 1999 (Angel et al., 2022).

Es fundamental resaltar que, en el proceso de desarrollo de este proyecto, se ha tomado una decisión estratégica orientada a mejorar la eficiencia y garantizar la coherencia del código. Con este propósito, se ha optado por agrupar las Historias de Usuario (HU) que se relacionan con las operaciones CRUD. Las operaciones CRUD (Create, Read, Update, Delete, por sus siglas en Ingles) hace referencia a las operaciones básicas de Crear, Leer, Actualizar y Eliminar (Muñiz del Valle et al., 2019). A esta agrupación de operaciones, en esta investigación se les denomina como gestionar. Esta elección se basa en varios factores clave. En primer lugar, al abordar las operaciones CRUD como una entidad cohesiva, se habilita la capacidad de implementar y desplegar la funcionalidad completa de manera más eficiente. En vez de dividir el trabajo en partes individuales, esta estrategia también permite aprovechar las funcionalidades y herramientas provistas por el Framework de Django Rest para simplificar el desarrollo de las operaciones CRUD. Como resultado, se logra un ahorro significativo de tiempo y esfuerzo en comparación con la creación de historias de usuario separadas para cada operación. En el contexto de esta investigación, nos referimos a estas operaciones CRUD como "gestionar". De igual forma con el propósito de agrupar las Historias de Usuario se decide que aquellos CRUD

de los cuales no se realice una operación en el sistema se les llame " Administrar", esto ocurre en caso de que falte una operación de añadir, eliminar o modificar. Esto es posible debido a las facilidades de las herramientas y tecnologías sobre las que se está desarrollando el sistema.

Tabla 4: Historia de Usuario # 1

Historia de Usuario	
Número: 1	Nombre: Administrar productos dañados o con salidas por causas especiales
Usuario: director de la tienda	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 1.4	Iteración asignada: 1
Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda	
<p>Descripción: Ofrece al director de la tienda una vista para poder darle salida de almacén a los productos dañados o con salida por causas especiales, al mismo tiempo permitiéndole añadir nuevos productos dañados y con salida especial a la tabla salida y descontando del almacén los productos, también elimina los productos con salida especial. Al igual que permite visualizar todos los productos con estos tipos de salidas de almacén en las diferentes fechas. Además, se ofrecen funcionalidades para que pueda visualizar las existencias de los productos en almacén.</p>	
<p>Observaciones: El director de la tienda debe estar registrado en el sistema para poder tener acceso.</p>	

Prototipo de Interfaz:

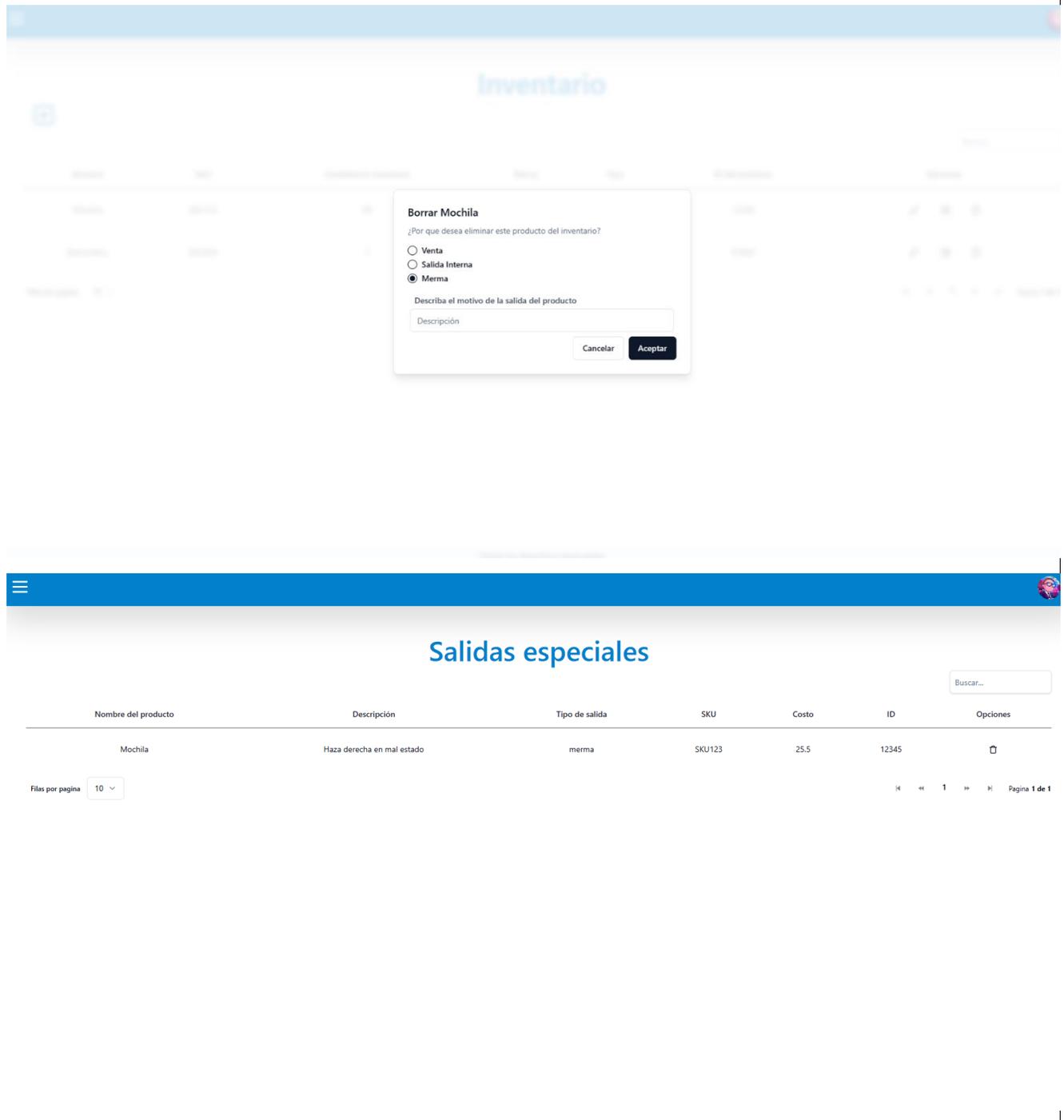


Tabla 5: Historia de Usuario # 2

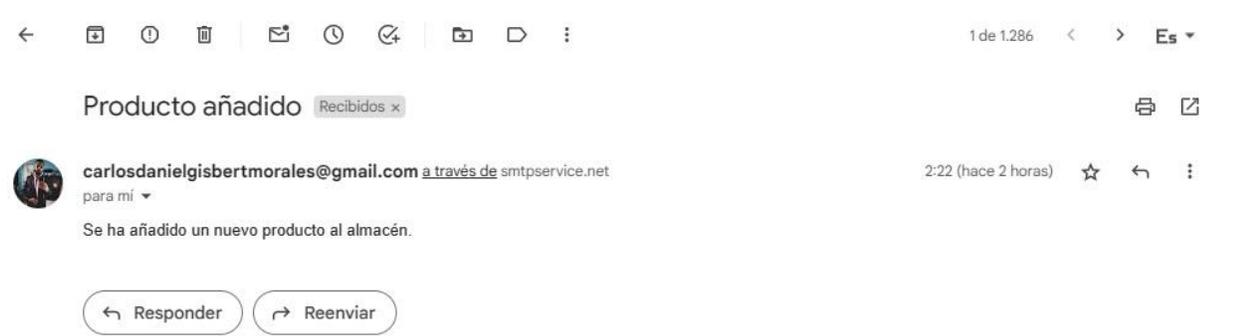
Historia de Usuario	
Número: 2	Nombre: Enviar correo
Usuario: almacenero	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 0.5	Iteración asignada: 1
Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda	
<p>Descripción: Permitirá al almacenero enviar una notificación al director de la tienda notificando "Se ha añadido un nuevo producto al almacén", y de esta forma este al tanto de los nuevos productos entrantes.</p>	
Observaciones: Se enviarán la notificación cada 15 días para mantener un control.	
<p>Prototipo de Interfaz:</p>  <p>The screenshot shows an email client interface. At the top, there's a navigation bar with icons for back, home, search, trash, mail, clock, refresh, folder, and a menu. The main content area shows an email titled "Producto añadido" with a "Recibidos x" label. The sender is "carlosdanielgisbertmorales@gmail.com a través de smtpservice.net" and the recipient is "para mí". The email body contains the text "Se ha añadido un nuevo producto al almacén." Below the email, there are two buttons: "Responder" and "Reenviar".</p>	

Tabla 6: Historia de Usuario # 3

Historia de Usuario	
Número: 3	Nombre: Gestionar la entrada y salida de productos en almacén
Usuario: Almacenero	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 2.3	Iteración asignada: 1
Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda	
<p>Descripción: ofrecerá al almacenero la vista para que pueda añadir nuevos productos al almacén y de igual forma modificar las cantidades de los productos ya existentes, al igual el sistema permitirá rebajar las cantidades de almacén de los productos vendidos a través de la tienda, ofrecerá al almacenero poder realizar conteos de supervisión en el inventario y en caso de no coincidir las cantidades poder aumentar o disminuir las cantidades en correspondencia con las existencias reales del almacén. También añadir la descripción a cada uno de los artículos.</p>	

Observaciones: El almacenero deberá estar autenticado en el sistema para poder realizar estas acciones.

Prototipo de interfaz:

Nuevo Producto
Introduzca la información relacionada al producto por agregar

Almacen

Proveedor

Nombre

SKU

Foto

Marca

SLUG

Fecha de creación

Fecha de modificación

Tipo

Tabla 7: Historia de Usuario # 4

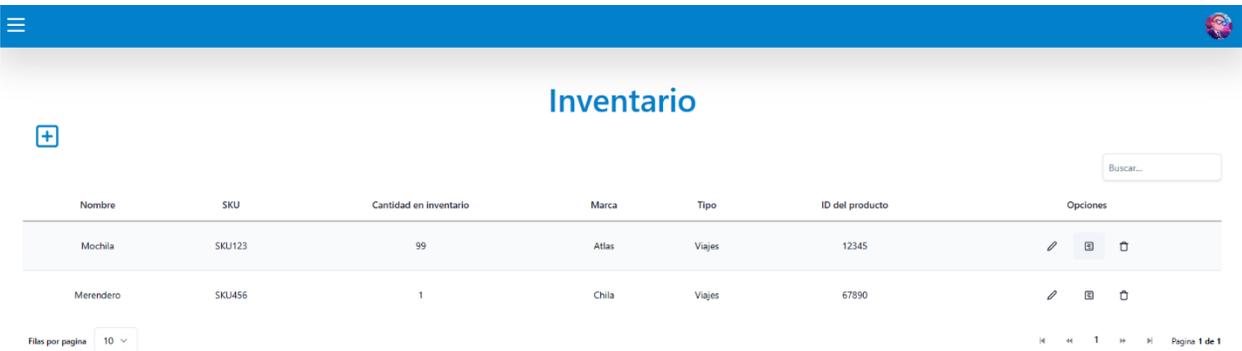
Historia de Usuario	
Número: 4	Nombre: Visualizar productos
Usuario: Gestor de negocios	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 0.4	Iteración asignada: 1
Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda	
<p>Descripción: De forma dinámica, permitirá al gestor de negocios visualizar las cantidades en almacén de cada uno de los productos existentes. A través de un acceso de seguridad solamente podrá ver esta información.</p>	
Observaciones: Debe estar autenticado en el sistema para poder realizar la acción	
Prototipo de interfaz:	
 <p>The screenshot shows a web interface for an inventory system. At the top, there is a blue navigation bar with a menu icon on the left and a profile picture on the right. Below the navigation bar, the word 'Inventario' is displayed in a large blue font. To the left of the title is a plus sign icon, and to the right is a search bar labeled 'Buscar...'. Below these elements is a table with the following columns: 'Nombre', 'SKU', 'Cantidad en inventario', 'Marca', 'Tipo', 'ID del producto', and 'Opciones'. The table contains two rows of data: one for 'Mochila' (SKU: SKU123, Quantity: 99, Brand: Atlas, Type: Viajes, ID: 12345) and one for 'Merendero' (SKU: SKU456, Quantity: 1, Brand: Chila, Type: Viajes, ID: 67890). Each row has three icons in the 'Opciones' column: a pencil (edit), a square (toggle), and a trash can (delete). At the bottom left, there is a 'Files por pagina' dropdown menu set to '10'. At the bottom right, there are pagination controls showing 'Página 1 de 1'.</p>	

Tabla 8: Historia de Usuario # 5

Historia de Usuario																											
Número: 5	Nombre: Generar gráficos con márgenes de ventas por productos																										
Usuario: almacenero, director de la tienda																											
Prioridad en negocio: Alta	Riesgo en desarrollo: Media																										
Puntos estimados: 0.5	Iteración asignada: 1																										
Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda																											
Descripción: Permitirá al director de la tienda y al almacenero visualizar a través de gráficos de barra ver los productos más vendidos por días, semanas, quincenas y años.																											
Observaciones:																											
Prototipo de interfaz:																											
<table border="1"> <caption>Ventas por productos</caption> <thead> <tr> <th>Categoría</th> <th>Cantidad de ventas</th> </tr> </thead> <tbody> <tr> <td>Alimentos y Bebidas</td> <td>10</td> </tr> <tr> <td>Aseo</td> <td>17</td> </tr> <tr> <td>Combos Familiares</td> <td>12</td> </tr> <tr> <td>Infantil</td> <td>3</td> </tr> <tr> <td>Joyería y Bisutería</td> <td>6</td> </tr> <tr> <td>Mascotas</td> <td>5</td> </tr> <tr> <td>Muebles y Decoración</td> <td>10</td> </tr> <tr> <td>Paletaria</td> <td>11</td> </tr> <tr> <td>Piezas de Camos</td> <td>15</td> </tr> <tr> <td>Ropa</td> <td>12</td> </tr> <tr> <td>Utiles del Hogar</td> <td>19</td> </tr> <tr> <td>Utiles y Herramientas</td> <td>7</td> </tr> </tbody> </table>		Categoría	Cantidad de ventas	Alimentos y Bebidas	10	Aseo	17	Combos Familiares	12	Infantil	3	Joyería y Bisutería	6	Mascotas	5	Muebles y Decoración	10	Paletaria	11	Piezas de Camos	15	Ropa	12	Utiles del Hogar	19	Utiles y Herramientas	7
Categoría	Cantidad de ventas																										
Alimentos y Bebidas	10																										
Aseo	17																										
Combos Familiares	12																										
Infantil	3																										
Joyería y Bisutería	6																										
Mascotas	5																										
Muebles y Decoración	10																										
Paletaria	11																										
Piezas de Camos	15																										
Ropa	12																										
Utiles del Hogar	19																										
Utiles y Herramientas	7																										

Tabla 9: Historia de Usuario # 6

Historia de Usuario	
Número: 6	Nombre: Gestionar pre despacho
Usuario: almacenero	
Prioridad en negocio: Alta	Riesgo en desarrollo: alta
Puntos estimados: 1.2	Iteración asignada: 1
Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda	
<p>Descripción: Permitirá al almacenero visualizar todas las compras de los clientes en la tienda por facturas, permitiéndole apartar los productos del almacén para una vez el cliente pase por la tienda a recoger su pedido ya esté listo, de igual forma a través de esa vista le permitirá marcar que el producto está listo para recoger por el cliente, una vez se haya realizado la acción. También permitirá añadir nuevas facturas, modificarlas y eliminarlas de forma permanente.</p>	
Observaciones: Debe estar autenticado en el sistema para poder realizar la acción	
Prototipo de interfaz:	

Pre despacho

Buscar...

Numero de la orden	Productos	Pago total	Pagado	Opciones
1	Mochila	45.74	Pagado	
2	Mochila	65.99	No Pagado	

Filas por pagina 10

« 1 » Pagina 1 de 1

Tabla 11: Historia de Usuario # 7

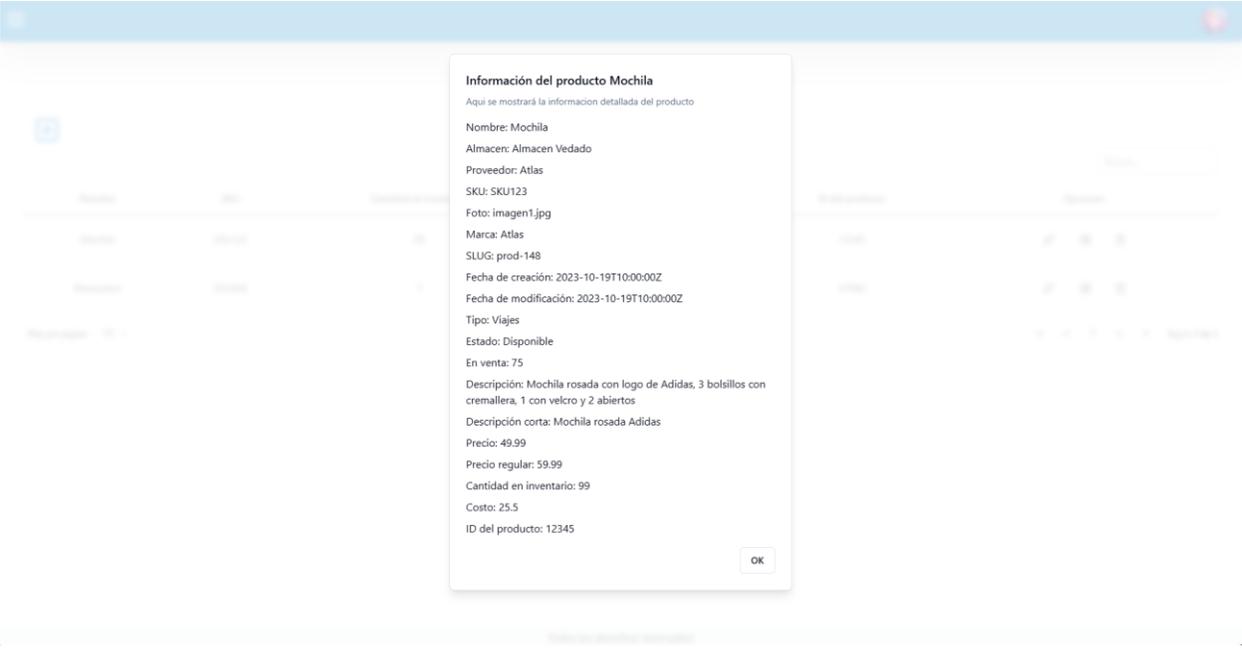
Historia de Usuario	
Número: 7	Nombre: Mostrar detalles
Usuario: almacenero, director de la tienda	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 0,4	Iteración asignada: 2
Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda	
Descripción: Permitirá al almacenero y al director de la tienda visualizar los detalles específicos de entrada de los productos, con una breve descripción del estado de entrada de los productos y la fecha en que se creó el producto.	
Observaciones: Debe estar autenticado en el sistema para poder realizar la acción	
Prototipo de interfaz:	
 <p>The screenshot shows a web application interface with a modal window titled 'Información del producto Mochila'. The modal contains the following text: 'Aquí se mostrará la información detallada del producto', 'Nombre: Mochila', 'Almacén: Almacén Vedado', 'Proveedor: Atlas', 'SKU: SKU123', 'Foto: imagen1.jpg', 'Marca: Atlas', 'SLUG: prod-148', 'Fecha de creación: 2023-10-19T10:00:00Z', 'Fecha de modificación: 2023-10-19T10:00:00Z', 'Tipo: Viajes', 'Estado: Disponible', 'En venta: 75', 'Descripción: Mochila rosada con logo de Adidas, 3 bolsillos con cremallera, 1 con velcro y 2 abiertos', 'Descripción corta: Mochila rosada Adidas', 'Precio: 49.99', 'Precio regular: 59.99', 'Cantidad en inventario: 99', 'Costo: 25.5', and 'ID del producto: 12345'. There is an 'OK' button at the bottom right of the modal.</p>	

Tabla 12: Historia de Usuario # 8

Historia de Usuario	
Número: 8	Nombre: Generar historial de productos dañados o con salida especial
Usuario: director de la tienda	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 0,5	Iteración asignada: 2
Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda	
Descripción: Permitirá al director de la tienda visualizar un historial con todos los productos que han salido de almacén por estar dañados y por otro lado todos los productos con una salida por causa especial.	
Observaciones: Debe estar autenticado en el sistema para poder realizar la acción	
Prototipo de interfaz:	

Tabla 13: Historia de Usuario # 9

Historia de Usuario	
Número: 9	Nombre: Gestionar usuarios
Usuario: director de la tienda	
Prioridad en negocio: Alta	Riesgo en desarrollo: alta
Puntos estimados: 1.2	Iteración asignada: 2
Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda	
<p>Descripción: Permitirá al director de la tienda añadir nuevos usuarios al sistema, modificarlos, mostrar todos los usuarios en forma de lista y eliminar un usuario. Permitiendo que solamente este usuario sea el encargado de la gestión de los usuarios autorizados que podrán acceder a cada módulo.</p>	
Observaciones: Debe estar autenticado en el sistema para poder realizar la acción	
<p>Prototipo de interfaz:</p> 	

Tabla 14: Historia de Usuario # 10

Historia de Usuario	
Número: 10	Nombre: Gestionar almacenes
Usuario: director de la tienda	
Prioridad en negocio: Alta	Riesgo en desarrollo: alta
Puntos estimados: 1.3	Iteración asignada: 2
Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda	
<p>Descripción: Permitirá al director de la tienda añadir un nuevo almacén, al mismo tiempo modificarlo y eliminar un almacén, al igual que mostrar un listado con todos los almacenes. Permitiéndole llevar el control de todos los almacenes de la entidad. La gestión de los almacenes tiene como objetivo aumentar el nivel de organización de los productos de la tienda.</p>	
Observaciones: Debe estar autenticado en el sistema para poder realizar la acción	
Prototipo de interfaz:	

Tabla 15: Historia de Usuario # 11

Historia de Usuario

Número: 11

Nombre: Visualizar ventas

Usuario: director de economía

Prioridad en negocio: Alta

Riesgo en desarrollo: Alta

Puntos estimados: 0.6

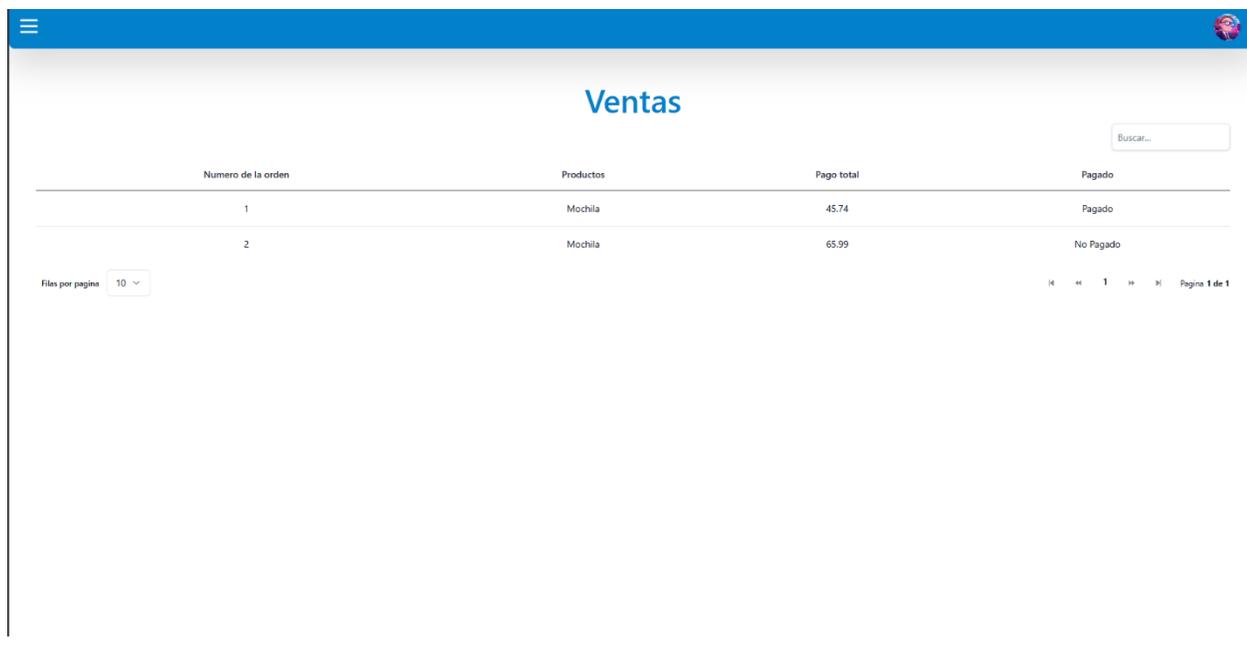
Iteración asignada: 3

Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda

Descripción: Permitirá al director de economía visualizar todas las ventas realizadas por la tienda, también observar los balances de ventas diarios, semanales, quincenales y anuales. Permitiéndole llevar una supervisión general de los márgenes de ganancias y pérdidas en la tienda.

Observaciones: Debe de estar autenticado en el sistema para poder realizar esta acción.

Prototipo de interfaz:



The screenshot shows a web interface with a blue header bar containing a menu icon and a profile picture. The main content area is titled 'Ventas' and features a search bar labeled 'Buscar...'. Below the title is a table with the following columns: 'Numero de la orden', 'Productos', 'Pago total', and 'Pagado'. The table contains two rows of data:

Numero de la orden	Productos	Pago total	Pagado
1	Mochila	45.74	Pagado
2	Mochila	65.99	No Pagado

At the bottom left of the table area, there is a 'Filas por pagina' dropdown menu set to '10'. At the bottom right, there are navigation icons and the text 'Página 1 de 1'.

Historia de Usuario

Número: 12

Nombre: Generar gráficos de ventas

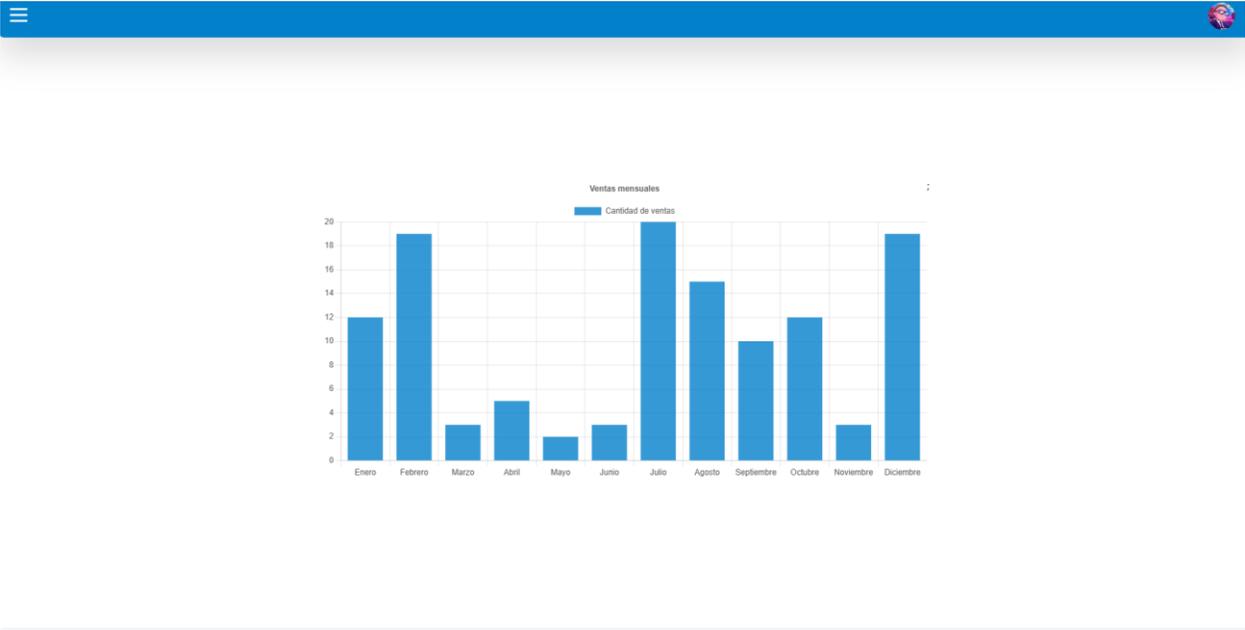
Usuario: director de economía	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 3
Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda	
Descripción: Permitirá realizar análisis a través de gráficos sobre las ventas de la tienda, permitiendo prever problemas económicos dentro de la entidad. Para tener mejores márgenes de rentabilidad, se podrán observar los análisis de ventas de forma mensuales.	
Observaciones: Debe de existir registro de venta para que se generen los gráficos.	
Prototipo de interfaz:	
	

Tabla 16: Historia de Usuario # 12

Tabla 17: Historia de Usuario # 13

Historia de Usuario	
Número: 13	Nombre: Gestionar pagos

Usuario: director de economía	
Prioridad en negocio: Alta	Riesgo en desarrollo: medio
Puntos estimados: 1.1	Iteración asignada: 3
Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda	
<p>Descripción: Permitirá al director de economía añadir, modificar, eliminar y visualizar los pagos quincenales de los embajadores por las compras realizada en la tienda por sus cliente y pagos a los proveedores por cada una de las ventas realizadas de sus productos.</p>	
Observaciones: Debe de estar autenticado en el sistema para poder realizar esta acción.	
Prototipo de interfaz:	

Nombre	Código	Pago por ventas	Pagado	Opciones
Roxana Turro Aranda	5	1000.5	Pagado	
Carlos Daniel Gisbert Morales	3	750.75	No Pagado	
Enrique Ferreiro Avila	10	400.75	No Pagado	

Tabla 18: Historia de Usuario # 14

Historia de Usuario	
Número: 14	Nombre: Buscar productos o facturas
Usuario: director de economía	
Prioridad en negocio: Medio	Riesgo en desarrollo: medio
Puntos estimados: 0.5	Iteración asignada: 3
Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda	
Descripción: Permitirá al director de economía buscar los productos o facturas de la tienda por negocio, material, tipo de producto, color, precio, medida, número de orden, número de factura y peso, permitiéndole acceder a la información de algún producto con mayor rapidez.	
Observaciones: Debe de existir el producto en el almacén o la factura se debió de realizar por la tienda.	

Prototipo de interfaz:

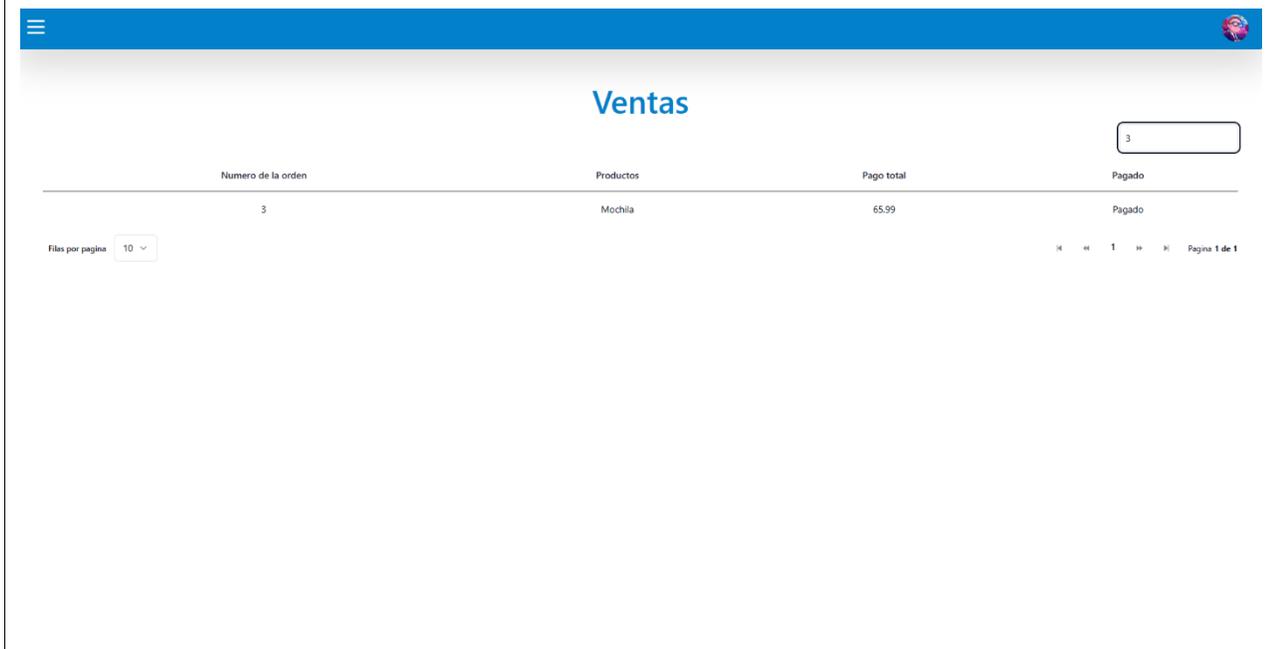
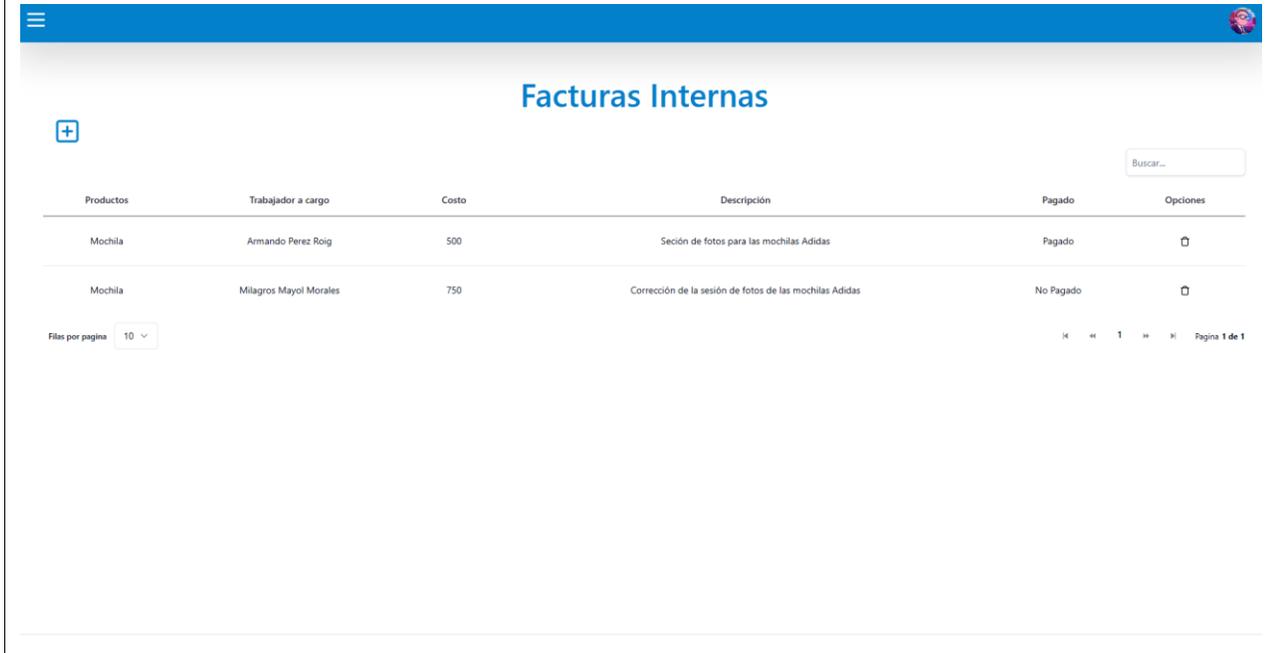


Tabla 19: Historia de Usuario # 15

Historia de Usuario	
Número: 15	Nombre: Administrar facturas internas
Usuario: director de economía	
Prioridad en negocio: Medio	Riesgo en desarrollo: medio
Puntos estimados: 1	Iteración asignada: 3
Programador responsable: Carlos Daniel Gisbert Morales, Roxana Turro Aranda	
Descripción: Permitirá al director de economía realizar facturas internas de costo con monto, área, motivo, nombre del trabajador encargado, para justificar el pago a alguna acción realizada por la misma empresa. También podrá eliminar facturas siempre y cuando sea necesario.	

Observaciones: Debe de estar autenticado en el sistema para realizar la acción

Prototipo de interfaz:



2.2.2 Requisitos no funcionales

Los Requisitos No Funcionales (RNF) de software forman una parte significativa de la especificación de requisitos y en algunos casos estos son críticos para el éxito del producto. Con frecuencia estos requisitos son ignorados o subestimados debido a que para muchos proyectos estos implican una cantidad considerable de trabajo y esfuerzo; resultan ser más complejos y requieren un mayor nivel de conocimiento. En el desarrollo de software para la salud estos problemas se incrementan dada la complejidad del mismo (Yenisel Molina Hernández et al., 2019).

RNF1 - Software

RNF1.1 – El sistema deberá de funcionar sobre cualquier distribución de sistema operativo Linux y/o superior a Windows 7.

RNF1.2 - El sistema deberá para su ejecución utilizarse como navegador web Mozilla Firefox, Chrome; Internet Explorer o cualquier otro.

RNF2 - Hardware

RNF2.1 - El sistema deberá requerir de 2gb de memoria RAM.

RNF2.2 - El sistema deberá requerir de 120gb de disco duro.

RNF3 - Restricciones de Diseño e Implementación

RNF3.1 - El sistema deberá utilizar un gestor de base de datos para el almacenamiento persistente de información MySQL.

RNF3.2 - El sistema deberá de utilizarse como lenguaje de programación Python 3.11.0 para el backend.

RNF 3.3- El sistema deberá utilizarse como framework Django 4.2.5 para el backend.

RNF3.3 - El sistema deberá de utilizarse como herramienta de modelado para los artefactos ingenieriles Visual Paradigm 8.0.

RNF4 - Interfaz o Apariencia Externa

RNF4.1 - El diseño gráfico no debe ser complejo, con colores en la gama de blanco, amarillo, negro y azul.

RNF4.2 - El sistema deberá tener una interfaz de usuario adaptable a diferentes dispositivos y tamaños

RNF5 - Eficiencia

RNF5.1 - El tiempo de respuesta a la hora de generar los distintos tipos de reportes no debe excederse de los 3ms y las restante funciones deben estar en el rango de 1 o 3 ms.

RNF6.1 - El sistema deberá permitir el acceso de hasta 10 personas de manera simultánea.

2.2.3 Estimación de esfuerzo por Historia de Usuario

La estimación de esfuerzos es uno de los principales retos de cualquier organización a la hora de considerar los recursos que dispondrá para la planeación y ejecución de un proyecto (Ortega-Hernandez, 2019). La estimación de esfuerzo que arroja cada historia de usuario, tiene como objetivo obtener un desarrollo del sistema correcto. Para una mejor organización se decide, asignar a cada iteración el conjunto de historias agrupadas en correspondencia con el módulo que represente.

Tabla 20: Estimación de esfuerzo por Historia de Usuario

Historias de Usuario		Puntos estimados(semanas)
1	Gestionar productos dañados o con salidas por causas especiales	1.4
2	Enviar correo	0.5
3	Gestionar la entrada y salida de productos en almacén	2.3
4	Visualizar productos	0.4
5	Generar gráficos con márgenes de ventas por productos	0.5
6	Gestionar pre despacho	1.5
7	Mostrar detalles	0.4
8	Generar historial de productos dañados o con salida especial	0.5
9	Gestionar usuarios	1.2
10	Gestionar almacenes	1.3
11	Visualizar ventas	0.6
12	Generar gráficos de ventas	0.5
13	Gestionar pagos	1.5

14	Buscar productos o facturas	0.5
15	Administrar facturas internas	1

2.2.4 Desarrollo del plan de iteraciones

Ya definidas las historias de usuario y realizada una previa estimación de esfuerzos, se pasa a la planificación de la etapa de implementación del sistema. Se crea un plan de iteraciones, donde se especifica la prioridad con que se implementarán las historias de usuario organizadas por iteraciones. Teniendo en cuenta el esfuerzo asociado a las historias de usuario y a las prioridades del cliente.

La fase de plan de iteraciones, en la cual se divide el proyecto en partes funcionales que permitan mejor control en su desarrollo. En la etapa de desarrollo se tienen: la fase de diseño computacional, en la que se realizará un diseño computacional detallado y específico del software (Gauthier & Budán, 2023). Se presenta a continuación el plan de duración de las iteraciones.

Tabla 21: Plan de duración de las iteraciones

Iteración	Historias de Usuario		Duración (semanas)
1	1	Gestionar productos dañados o con salidas por causas especiales	6
	2	Enviar correo	
	3	Gestionar la entrada y salida de productos en almacén	
	4	Visualizar productos	
	5	Generar gráficos con márgenes de ventas por productos	
2	1	Gestionar pre despacho	5.5
	2	Mostrar detalles	

	3	Generar historial de productos dañados o con salida especial	
	4	Gestionar usuarios	
	5	Gestionar almacenes	
3	1	Visualizar ventas	5
	2	Generar gráficos de ventas	
	3	Gestionar pagos	
	4	Buscar productos o facturas	
	5	Administrar facturas internas	
Total			16.5

2.2.5 Plan de entregas

A continuación, se presenta una tabla con las iteraciones con su fecha de entrega, el plan de entrega que se plantea a continuación, se hace una propuesta de las versiones del sistema.

Tabla 22: Plan de entrega de versiones

Entregable	Iteración 1	Iteración 2	Iteración 3
Fecha Inicio	19/6/23	1/8/23	8/9/23
Fecha de entrega	31/7/23	7/9/23	7/10/23

2.3 Fase II: Diseño del sistema

Un patrón de diseño es una técnica para resolver problemas simples y comunes que se encuentran en la vida diaria del desarrollo de software, sobre todo constituye una solución respecto al desarrollo de interacciones o interfaces. Se puede definir un patrón como un modelo que explica cómo resolver un determinado problema en el desarrollo de software, no consiste en un conjunto de herramientas que se implementan en el código de una aplicación

para resolver el problema. Un patrón es un concepto, es el razonamiento lógico que existe para resolver determinado problema. Aprender a implementar patrones de diseño permite tener el conocimiento para resolver todo tipo de problemas utilizando principios del diseño orientado a objetos (Danilo Gavilánez Alvarez et al., 2022).

Durante la fase de diseño y programación se hace una revisión de los estándares y los fundamentos teóricos recibidos durante el periodo de formación académica. La Fase de Diseño es la siguiente etapa definida en la metodología de desarrollo aplicada, el objetivo es en base a los datos recopilados con las necesidades iniciales del usuario, crear las bases y estructuras que servirán de guía para el proceso de programación del sistema (De et al., 2021).

La arquitectura de software son patrones o lineamientos que ayudan a la construcción de un programa. Estos patrones permiten tener una guía para los desarrolladores, analistas y todos los cargos relacionados para lograr cumplir con los requerimientos de la aplicación. La arquitectura del software por tanto define la estructura que debe de tener un software, las piezas que se debe construir y el modo en el que se deben de juntar y trabajar entre ellas. Se define a alto nivel mediante una serie de patrones y abstracciones que seguir para el desarrollo del software y para la interacción entre sus diversas piezas (Flores et al., 2021).

En la presente investigación se eligió la Arquitectura Cliente servidor, se ve que el dispositivo que solicita información se denomina cliente y el dispositivo que responde a la solicitud se denomina servidor. Los procesos de cliente y servidor se consideran una parte de la capa de Aplicación, ver ilustración 3. El cliente comienza el intercambio solicitando los datos al servidor, que responde enviando uno o más streams de datos al cliente. Los protocolos de capa de Aplicación describen el formato de las solicitudes y respuestas entre clientes y servidores. Además de la transferencia real de datos, este intercambio puede requerir de información adicional, como la autenticación del usuario y la identificación de un archivo de datos a transferir.

Aunque los datos generalmente se describen como un flujo del servidor al cliente, algunos datos siempre fluyen del cliente al servidor. El flujo de datos puede ser el mismo en ambas direcciones o inclusive ser mayor en la dirección que va del cliente al servidor. Por ejemplo, un cliente puede transferir un archivo al servidor con fines de almacenamiento. La transferencia de datos de un cliente a un servidor se conoce como subida y la de los datos de un servidor a un cliente. Las

características básicas de una arquitectura Cliente-Servidor son: Combinación de un cliente que interactúa con el usuario, y un servidor que interactúa con los recursos compartidos. El proceso del cliente proporciona la interfaz entre el usuario y el resto del sistema. El proceso del servidor actúa como un motor de software que maneja recursos compartidos tales como bases de datos, impresoras, módems, entre otras (Ovallos Ovallos & Medina Cardenas, 2020).

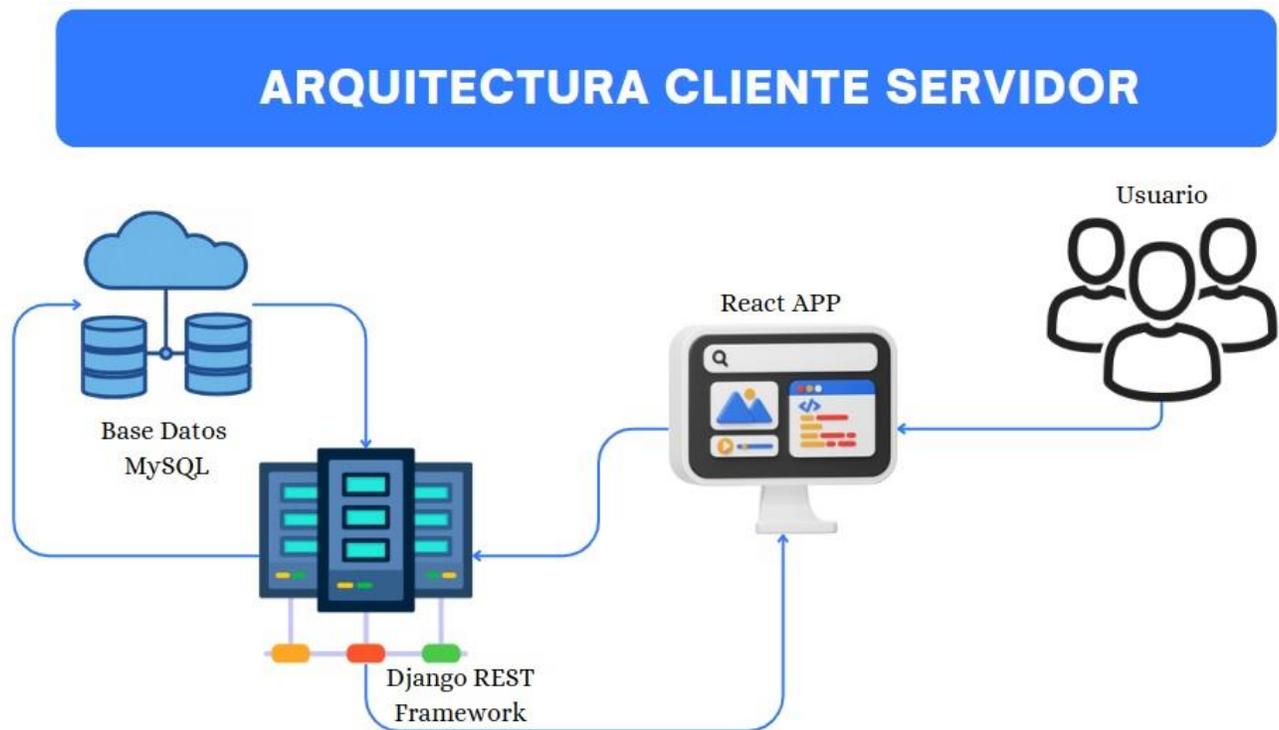


Ilustración 3. Diseño arquitectónico para Sistema de gestión empresarial. Fuente: Elaboración propia.

- **Django** utiliza la librería Django Rest Framework que se utiliza para manejar las peticiones al backend vía API. Es una aplicación Django que permite construir proyectos software bajo la arquitectura REST. Incluye gran cantidad de código para reutilizar Views, Resources, entre otros y una interfaz administrativa desde la cual es posible realizar pruebas sobre las operaciones HTTP como lo son: POST y GET. Django REST Framework es un framework que nos permite el fácil desarrollo de una API REST en Python. Es una herramienta poderosa y flexible para construir Web APIs.
- **React** es responsable de enviar solicitudes a la API RESTful proporcionada por Django. React utiliza la biblioteca axios para enviar solicitudes HTTP a la API RESTful. La biblioteca axios se utiliza para enviar solicitudes GET, POST, PUT y DELETE.

- **PostgreSQL** es la base de datos utilizada para almacenar los datos en la aplicación. Django utiliza el ORM (Object-Relational Mapping) para interactuar con PostgreSQL. El ORM se utiliza para realizar operaciones CRUD en la base de datos.

React envía solicitudes HTTP a la API RESTful proporcionada por Django, que a su vez interactúa con PostgreSQL utilizando el ORM. Juntos, estos componentes forman una arquitectura cliente-servidor completa que puede manejar aplicaciones web complejas.

2.4 Patrones de diseño

En el contexto de XP (Programación Extrema), los patrones de diseño se utilizan para mejorar la calidad del código y la eficiencia del equipo de desarrollo.

Los patrones de diseño son importantes en XP porque ayudan a los desarrolladores a crear software más robusto y fácil de mantener. Los patrones de diseño proporcionan soluciones probadas y comprobadas para problemas comunes de diseño, lo que permite a los desarrolladores evitar errores costosos y optimizar el tiempo de desarrollo.

Al comprender y aplicar estos patrones, los profesionales del desarrollo pueden mejorar significativamente la eficiencia y mantenibilidad del código. Los patrones de diseño también ayudan a los desarrolladores a comunicarse y compartir soluciones eficientes y efectivas que han demostrado ser exitosas en el pasado.

Los patrones GRASP son un conjunto de patrones de diseño orientado a objetos que se utilizan para asignar responsabilidades a los objetos y mejorar la calidad del código, a continuación, los patrones GRASP a utilizar en el desarrollo del proyecto:

- **Bajo acoplamiento:** Este patrón se utiliza para reducir la dependencia entre los objetos. El acoplamiento mide el grado en que una clase está conectada a otra, tiene conocimiento de otra o, de alguna manera, depende de otra. Al reducir el acoplamiento, mejora la reutilización del código y la facilidad de mantenimiento. Se utilizó el patrón de Inyección de Dependencias, este se utiliza para reducir el acoplamiento entre los componentes de la aplicación. Este patrón permite que los componentes dependan de abstracciones en lugar de implementaciones concretas, lo que facilita la modificación y el mantenimiento del código. Se aplicó en plantillas de React reutilizables para que puedan utilizarse en diferentes partes del proyecto sin depender de un contexto específico, facilitando la reutilización de componentes de interfaz.

- **Experto:** Este patrón se utiliza para asignar responsabilidades a los objetos que tienen la información necesaria para realizar una tarea. Por ejemplo, la responsabilidad de crear un objeto debe recaer sobre la clase que conoce toda la información necesaria para crearlo. De este modo, permite obtener un diseño con mayor cohesión y se mantiene encapsulada la información. Este patrón se aplicó en varias partes del código para lograr una arquitectura de software más modular y escalable. Se aplicó en al crear hooks personalizados, que encapsulan la lógica relacionada con un aspecto específico de la aplicación, como es la gestión de formularios.
- **Alta cohesión:** Este patrón se utiliza para asegurar que cada objeto tenga una única responsabilidad y que todas las partes de ese objeto estén relacionadas con esa responsabilidad. Un objeto tiene alta cohesión si todo lo que hace está bien delimitado dentro del mismo objeto. Además, para tener alta cohesión, debe ser un objeto cuanto más pequeño mejor. Este concepto está relacionado inversamente con el acoplamiento, es decir, si aumenta la cohesión, suele disminuir el acoplamiento, y viceversa. Se aplica este patrón por todas partes del código desde modelo, vistas y formularios para organizar el código de forma que las partes relacionadas estén agrupadas y sean fáciles de entender y mantener, permitiendo mejorar la claridad y mantenibilidad de la aplicación.

2.5 Tarjetas Clase - Responsabilidad – Colaborador (CRC)

Las HU son evaluadas para dividirlos en tareas, estas tareas se representan por medio de las tarjetas Clase-Responsabilidad-Colaborador (CRC). A continuación, se describen las tarjetas definidas para la implementación de la solución.

Tabla 23: Tarjeta CRC # 1

Tarjeta CRC	
Clase: ProductoSalida	
Responsabilidad	Colaboración
create() Permite darle entrada a la tabla ProductoSalida	Producto FacturasInternas

<p>retrieve() se utiliza para recuperar un objeto específico de la base de datos</p> <p>update() Actualiza los valores de la tabla</p> <p>partial_update() se utiliza para actualizar parcialmente un objeto específico de la base de datos.</p> <p>destroy()se utiliza para eliminar un objeto específico de la base de datos.</p> <p>list() Permite realizar una lista de objetos</p>	
---	--

Tabla 24: Tarjeta CRC # 2

Tarjeta CRC	
Clase: Almacen	
Responsabilidad	Colaboración
<p>create() Permite darle entrada a la tabla almacén</p> <p>retrieve() se utiliza para recuperar un objeto específico de la base de datos</p> <p>update() Actualiza los valores de la tabla</p> <p>partial_update() se utiliza para actualizar parcialmente un objeto específico de la base de datos.</p>	<p>Producto</p>

<p>destroy()se utiliza para eliminar un objeto específico de la base de datos.</p> <p>list() Permite realizar una lista de objetos</p>	
--	--

Tabla 25: Tarjeta CRC # 3

Tarjeta CRC	
Clase: Producto	
Responsabilidad	Colaboración
<p>create() Permite darle entrada a la tabla Producto</p> <p>retrieve() se utiliza para recuperar un objeto específico de la base de datos</p> <p>update() Actualiza los valores de la tabla</p> <p>partial_update() se utiliza para actualizar parcialmente un objeto específico de la base de datos.</p> <p>destroy()se utiliza para eliminar un objeto específico de la base de datos.</p> <p>list() Permite realizar una lista de objetos</p>	<p>Almacen</p> <p>ProductoVariacion</p> <p>Proveedor</p> <p>Orden</p>

Tabla 26: Tarjeta CRC # 4

Tarjeta CRC

Clase: Orden	
Responsabilidad	Colaboración
<p>create() Permite darle entrada a la tabla orden</p> <p>retrieve() se utiliza para recuperar un objeto específico de la base de datos</p> <p>update() Actualiza los valores de la tabla</p> <p>partial_update() se utiliza para actualizar parcialmente un objeto específico de la base de datos.</p> <p>destroy()se utiliza para eliminar un objeto específico de la base de datos.</p> <p>list() Permite realizar una lista de objetos</p>	<p>Producto</p> <p>Embajador</p>

Tabla 27: Tarjeta CRC # 5

Tarjeta CRC	
Clase: Embajador	
Responsabilidad	Colaboración
<p>create() Permite darle entrada a la tabla embajador</p> <p>retrieve() se utiliza para recuperar un objeto específico de la base de datos</p> <p>update() Actualiza los valores de la tabla</p>	<p>Orden</p>

<p>partial_update() se utiliza para actualizar parcialmente un objeto específico de la base de datos.</p> <p>destroy()se utiliza para eliminar un objeto específico de la base de datos.</p> <p>list() Permite realizar una lista de objetos</p>	
--	--

Tabla 28: Tarjeta CRC # 6

Tarjeta CRC	
Clase: ProductoVariacion	
Responsabilidad	Colaboración
<p>create() Permite darle entrada a la tabla ProductoVariacion</p> <p>retrieve() se utiliza para recuperar un objeto específico de la base de datos</p> <p>update() Actualiza los valores de la tabla</p> <p>partial_update() se utiliza para actualizar parcialmente un objeto específico de la base de datos.</p> <p>destroy()se utiliza para eliminar un objeto específico de la base de datos.</p> <p>list() Permite realizar una lista de objetos</p>	<p>Producto</p>

Tabla 29: Tarjeta CRC # 7

Tarjeta CRC	
Clase: Proveedor	
Responsabilidad	Colaboración
<p>create() Permite darle entrada a la tabla Proveedor</p> <p>retrieve() se utiliza para recuperar un objeto específico de la base de datos</p> <p>update() Actualiza los valores de la tabla</p> <p>partial_update() se utiliza para actualizar parcialmente un objeto específico de la base de datos.</p> <p>destroy()se utiliza para eliminar un objeto específico de la base de datos.</p> <p>list() Permite realizar una lista de objetos</p>	<p>Producto</p>

Tabla 30: Tarjeta CRC # 8

Tarjeta CRC	
Clase: FacturasInternas	
Responsabilidad	Colaboración
<p>create() Permite darle entrada a la tabla FacturasInternas</p>	<p>Producto</p>

<p>retrieve() se utiliza para recuperar un objeto específico de la base de datos</p> <p>update() Actualiza los valores de la tabla</p> <p>partial_update() se utiliza para actualizar parcialmente un objeto específico de la base de datos.</p> <p>destroy()se utiliza para eliminar un objeto específico de la base de datos.</p> <p>list() Permite realizar una lista de objetos</p>	
---	--

Tabla 31: Tarjeta CRC # 9

Tarjeta CRC	
Clase: User	
Responsabilidad	Colaboración
<p>create() Permite darle entrada a la tabla user</p> <p>retrieve() se utiliza para recuperar un objeto específico de la base de datos</p> <p>update() Actualiza los valores de la tabla</p> <p>partial_update() se utiliza para actualizar parcialmente un objeto específico de la base de datos.</p>	<p>Group</p> <p>Permission</p>

<p>destroy()se utiliza para eliminar un objeto específico de la base de datos.</p> <p>list() Permite realizar una lista de objetos</p>	
--	--

Tabla 32: Tarjeta CRC # 10

Tarjeta CRC	
Clase: Reporte	
Responsabilidad	Colaboración
<p>create() Permite darle entrada a la tabla Reporte</p> <p>retrieve() se utiliza para recuperar un objeto específico de la base de datos</p> <p>update() Actualiza los valores de la tabla</p> <p>partial_update() se utiliza para actualizar parcialmente un objeto específico de la base de datos.</p> <p>destroy()se utiliza para eliminar un objeto específico de la base de datos.</p> <p>list() Permite realizar una lista de objetos</p>	<p>Producto</p> <p>Orden</p>

Conclusiones parciales

En este capítulo se profundizó en varios aspectos fundamentales relacionados con el desarrollo del sistema a desarrollar. Esto permitió arribar a las siguientes conclusiones:

- Con la definición de las Historias de Usuario (HU) se estableció una base sólida para desarrollar las funcionalidades requeridas en el sistema. Permitiendo el enfoque en las necesidades del cliente.
- La estimación del tiempo para la Implementación de las HU, se ve que es un componente crítico en el proceso de desarrollo, donde se define un tiempo estimado de 16.5 semanas. Al analizar y cuantificar el tiempo necesario para cada HU, se logró una planificación más precisa y un control eficiente del proyecto, permitiendo calcular el tiempo para la entrega final al cliente.
- La identificación de Patrones GRASP, en el diseño del sistema proporciona una estructura y organización más sólidas. Estos patrones ayudan a distribuir las responsabilidades de manera efectiva entre las clases y objetos, lo que contribuye a un diseño más coherente y mantenible.
- La adopción de la Arquitectura Cliente-Servidor, permitió una clara separación de responsabilidades entre el cliente y el servidor, facilitando la escalabilidad y la interacción eficiente con el sistema desde múltiples ubicaciones, facilitando la implementación de las funcionalidades del módulo almacén y economía.

En conjunto, los temas abordados en este capítulo nos proporcionan un conjunto de herramientas y procesos sólidos para avanzar con confianza en el desarrollo de nuestro sistema de gestión de almacenes. La combinación de HU bien definidas, estimaciones precisas, patrones GRASP y la arquitectura cliente-servidor allana el camino para un desarrollo exitoso y un producto final que cumple con las expectativas del cliente. Estos fundamentos nos permiten abordar la siguiente fase de implementación con un enfoque claro y una base robusta en la metodología XP, respaldados por una arquitectura sólida y eficiente.

Capítulo III: Implementación y prueba del sistema

En este capítulo se detallan las iteraciones realizadas en la etapa de construcción de los módulos a desarrollar, también se exponen las tareas de ingeniería generadas para cada HU definida, al igual las pruebas de aceptación para el sistema. De igual forma es obtenido un producto funcional probado y listo para entregar al cliente al final de cada iteración como propone la metodología de XP.

3.1 Fase III: Desarrollo

En esta fase los desarrolladores deben diseñar las pruebas de unidad que ejercitarán cada historia de usuario. Después de tener las pruebas, los desarrolladores trabajarán en parejas para concentrarse en lo que debe implementarse para pasar la prueba de unidad (Pérez A, 2023).

3.1.1 Tareas de ingeniería

Tareas de Ingeniería, en ellas se almacena suficiente información para identificar y detallar la historia, se define como un conjunto de actividades en los cuales, utilizando técnicas y herramientas, se analiza un problema y se concluye con la especificación de una solución (Instituto de Estudios Superiores del Valle de Orizaba, 2019).

Para la primera iteración, se definieron un total de 12 tareas de ingeniería. Todas, desglosadas a partir de las HU correspondientes al módulo almacén. Se describen algunas de las tareas realizadas, el resto pueden ser consultadas en los anexos de la investigación.

Tabla 33 Tarea de ingeniería # 1

Tarea	
Número de tarea: 1	Número de Historia de Usuario: 1
Nombre de la tarea: Implementar la salida de productos dañados o con salida especial del almacén	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2

Fecha de inicio: 19 de junio del 2023	Fecha de fin: 20 de junio de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad que permite darle salida de almacén a los productos de forma manual, o sea que solamente no salgan por las ventas de la tienda, que es un proceso que el programa deberá realizar de forma automática, sino que siempre que sea necesario el director de la tienda podrá descontar del inventario del almacén productos dañados o con salidas especiales.	

Tabla 34 Tarea de ingeniería # 2

Tarea	
Número de tarea: 2	Número de Historia de Usuario: 1
Nombre de la tarea: Implementar la eliminación de los reportes de salida de productos dañados o con salidas especiales	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 23 de junio de 2023	Fecha de fin: 25 de junio de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad, que permite eliminar la información del reporte de salida para productos dañados o con salidas de forma especiales.	

Tabla 35 Tarea de ingeniería # 3

Tarea	
Número de tarea: 3	Número de Historia de Usuario: 1
Nombre de la tarea: Implementar la función de añadir los reportes de salida de productos dañados o con salidas especiales	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 26 de junio de 2023	Fecha de fin: 28 de junio de 2023

Programador responsable: Carlos Daniel Gisbert Morales
Descripción: Se implementa una funcionalidad, que permite añadir la información del reporte de salida par a productos dañados o con salidas de forma especiales.

Para una segunda iteración, se elaboraron un total de 14 tareas de ingeniería. Desglosadas a partir de las HU correspondientes al módulo Economía y parte del módulo almacén. Se describen algunas de las tareas realizadas, el resto pueden ser consultadas en los anexos de la investigación.

Tabla 36 Tarea de ingeniería # 4

Tarea	
Número de tarea: 4	Número de Historia de Usuario:1
Nombre de la tarea: Implementar la función de generar un pre despacho	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 1 de agosto de 2023	Fecha de fin: 3 de agosto de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad, que permita que generar un pre despacho, su función principal es una vez que el cliente realiza una compra a través de la tienda, todos los pedidos se vallan organizando en el predespacho, ordenado por su número de pedido y al mismo tiempo descontando de almacén las cantidades de cada producto por cada factura.	

Para una tercera iteración, se elaboraron un total de 10 tareas de ingeniería. Desglosadas a partir de las HU correspondientes al módulo Economía y parte del módulo almacén. Se describen algunas de las tareas realizadas, el resto pueden ser consultadas en los anexos de la investigación.

Tabla 37 Tarea de ingeniería # 5

Tarea

Número de tarea: 5	Número de Historia de Usuario: 5
Nombre de la tarea: Implementar el añadir facturas internas al sistema	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 26 de septiembre de 2023	Fecha de inicio: 28 de septiembre de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa esta funcionalidad, que permite realizar y guardar facturas internas de la empresa para justificar un pago sobre algún servicio que la empresa haya necesitado realizar.	

3.2 Fase IV: Pruebas

Las pruebas de unidad deben implementarse con un marco de trabajo que permita automatizarlas, con la finalidad de realizar pruebas de Integración y validación diarias, esto proporcionará al equipo un indicador del progreso y revelarán a tiempo si existe alguna falla en el sistema. Las pruebas tienen las siguientes características, desarrollo previamente aprobado, lo cual significa que primero se escriben las pruebas y luego el código. También las pruebas deben simular el envío de la entrada a probar y deben verificar que el resultado cumpla con las especificaciones de salida. El desarrollo de pruebas incremental y los requerimientos del usuario se expresan como historias, el equipo de desarrollo evalúa cada historia y la divide en tareas. Cada una representa una característica distinta del sistema y se pueden diseñar pruebas de unidad para esa tarea. La participación del usuario en el desarrollo de las pruebas también es una de las principales características, ya que el usuario ayuda a desarrollar las pruebas de aceptación, las cuales son pruebas que se implementan con los datos reales del cliente para verificar el cumplimiento real de sus necesidades. Al igual que el uso de bancos de pruebas automatizados, donde se ve que se debe usar un sistema que envíe a ejecución las pruebas automatizadas y de esta forma probar constantemente el sistema software (Pérez A, 2023).

3.2.1 Pruebas de Unitarias

Las pruebas unitarias, también conocidas como "unit testing", son una práctica de desarrollo de software que implica probar de manera exhaustiva y aislada las unidades más pequeñas y

básicas de código en un programa o sistema. Una "unidad" en este contexto se refiere a la parte más pequeña y autónoma de código que puede ser evaluada por sí misma, como una función, un método o incluso una clase.

El propósito principal de las pruebas unitarias es asegurarse de que cada unidad de código funcione correctamente y produzca los resultados esperados. Al llevar a cabo pruebas unitarias, los desarrolladores pueden detectar y corregir errores y fallos de manera temprana en el proceso de desarrollo, lo que hace que el software sea más robusto y confiable.

A continuación, se muestra la realización de algunas pruebas unitarias al modelo relacionado con la Historia de Usuario Gestionar Almacenes. El resto pueden ser consultadas en los anexos de la investigación.

```
class AlmacenTestCase(TestCase):
    def setUp(self):
        # Crear un objeto Almacen para su uso en las pruebas
        self.almacen = Almacen.objects.create(
            geolocalizacion="Latitud 123, Longitud 456",
            name="Mi Almacén"
        )

    def test_crear_almacen(self):
        # Comprobar que se puede crear un Almacen
        self.assertEqual(self.almacen.name, "Mi Almacén")

    def test_actualizar_almacen(self):
        # Actualizar el nombre del Almacen y comprobar el cambio
        self.almacen.name = "Nuevo Nombre de Almacén"
        self.almacen.save()
        self.assertEqual(self.almacen.name, "Nuevo Nombre de Almacén")

    def test_leer_almacen(self):
        # Leer el Almacen desde la base de datos y comprobar si los datos
        # son correctos
        almacen_leido = Almacen.objects.get(pk=self.almacen.pk)
        self.assertEqual(almacen_leido.geolocalizacion, "Latitud 123,
        Longitud 456")

    def test_eliminar_almacen(self):
        # Eliminar el Almacen y comprobar que ya no exista en la base de
        # datos
        self.almacen.delete()
        with self.assertRaises(Almacen.DoesNotExist):
            Almacen.objects.get(pk=self.almacen.pk)
```

Ilustración 4. Pruebas unitarias al Gestionar Almacenes.

Pruebas unitarias al modelo relacionado con la Historia de Usuario Gestionar entrada y salidas de productos del almacén.

```
# Comprobar que se puede crear un Producto
self.assertEqual(self.producto.name, "Producto 1")

def test_actualizar_producto(self):
    # Actualizar el nombre del Producto y comprobar el cambio
    self.producto.name = "Producto Actualizado"
    self.producto.save()
    self.assertEqual(self.producto.name, "Producto Actualizado")

def test_leer_producto(self):
    # Leer el Producto desde la base de datos y comprobar si los
    # datos son correctos
    producto_leido = Producto.objects.get(pk=self.producto.pk)
    self.assertEqual(producto_leido.sku, "SKU001")

def test_eliminar_producto(self):
    # Eliminar el Producto y comprobar que ya no exista en la base de
    # datos
    self.producto.delete()
    with self.assertRaises(Producto.DoesNotExist):
        Producto.objects.get(pk=self.producto.pk)
```

Ilustración 5. Pruebas unitarias al Gestionar entrada y salidas de productos del almacén.

Se realizaron tres iteraciones de prueba, en la primera no se alcanzó un 100% de satisfacción, alcanzando solamente el 80% de satisfacción. En la segunda se obtuvo un 90% de satisfacción en las pruebas realizadas. Y en la tercera iteración se obtuvieron resultados satisfactorios. A continuación, se presenta un gráfico resumen con los resultados obtenidos:

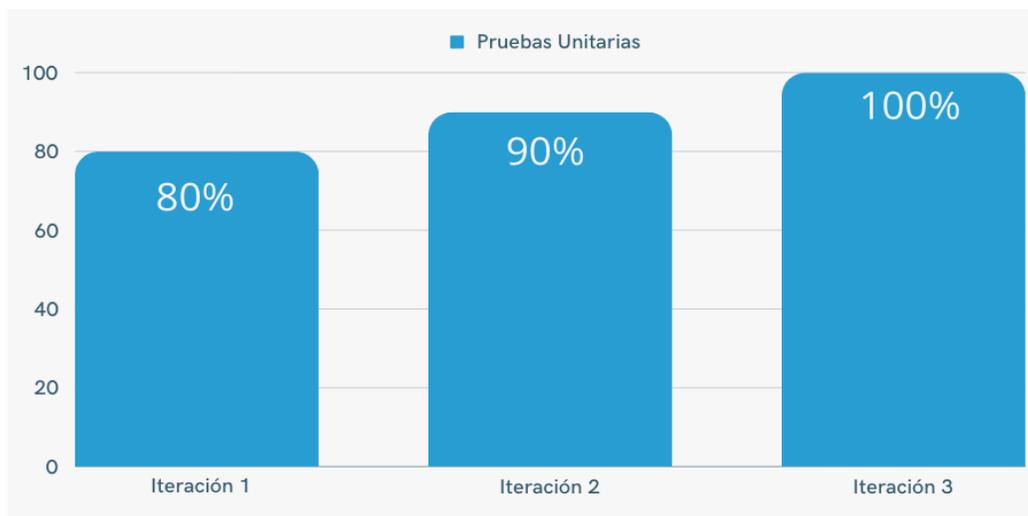


Ilustración 6. Comparativa de resultados de pruebas unitarias. Fuente: Elaboración propia

3.2.2 Pruebas de aceptación

Las pruebas de aceptación son pruebas de caja negra que se realizan a partir de las historias de usuario, una historia de usuario puede tener varias pruebas de aceptación que requiera para asegurar el correcto funcionamiento. El principal objetivo es garantizar que los requerimientos han sido cumplidos (De Ingeniería et al., 2020).

3.2.3 Pruebas de aceptación para la Iteración I

Para la primera iteración, se definieron un total de 11 casos de pruebas de aceptación. Todas enfocadas a evaluar la implementación del módulo Almacén. Se describen algunas de las pruebas realizadas, el resto pueden ser consultadas en los anexos de la investigación.

Tabla 69 Prueba de aceptación # 1

Caso de prueba de aceptación	
Código: HU1_P1	Historia de Usuario: 1
Nombre: gestión de salida de productos dañados o por salidas especiales de almacén	
Descripción: Prueba para la funcionalidad: gestión de salida de productos dañados o por salidas especiales de almacén.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la tienda. 	
Pasos de ejecución:	
El director de la tienda entra a la vista del inventario del almacén.	
Se selecciona el o los productos a descontar de inventario por alguno de estos dos motivos.	
Se marca la opción descontar seleccionados.	
Se muestra una notificación indicando que se descontó correctamente los productos seleccionados.	

Resultado: Satisfactorio

Tabla 70 Prueba de aceptación # 2

Caso de prueba de aceptación	
Código: HU1_P2	Historia de Usuario: 1
Nombre: Modificación de productos dañados o por salidas especiales.	
Descripción: Prueba para la funcionalidad: modificar reporte de salida de productos dañados o por salidas especiales de almacén.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la tienda. 	
Pasos de ejecución:	
El director de la tienda entra a la vista de los productos dañados o por salida especial.	
Se selecciona el reporte a modificar	
Selecciona el campo a modificar y lo modifica	
Se muestra una alerta indicando que se modificó correctamente el reporte.	
Resultado: Satisfactorio	

Tabla 71 Prueba de aceptación # 3

Caso de prueba de aceptación	
Código: HU1_P3	Historia de Usuario: 1
Nombre: Eliminar productos dañados o por salidas especiales de almacén	
Descripción: Prueba para la funcionalidad: eliminar reporte de salida de productos dañados o por salidas especiales de almacén.	

<p>Condiciones de ejecución:</p> <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la tienda.
<p>Pasos de ejecución:</p> <p>El director de la tienda entra a la vista de los productos dañados o por salida especial.</p> <p>Se selecciona el reporte a eliminar</p> <p>Selecciona el campo eliminar</p> <p>Se muestra una alerta indicando que se eliminó correctamente el reporte.</p>
<p>Resultado: Satisfactorio</p>

Tabla 72 Prueba de aceptación # 4

Caso de prueba de aceptación	
Código: HU1_P4	Historia de Usuario: 1
Nombre: Añadir productos dañados o por salidas especiales de almacén	
Descripción: Prueba para la funcionalidad: Añadir reporte de salida de productos dañados o por salidas especiales de almacén.	
<p>Condiciones de ejecución:</p> <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la tienda. 	
<p>Pasos de ejecución:</p> <p>El director de la tienda entra a la vista del inventario del almacén.</p> <p>Luego de descontar el o los productos de almacén</p> <p>Selecciona el campo reporte</p> <p>Realiza un reporte de salida con los motivos de la salida del producto</p>	

<p>Selecciona el campo añadir.</p> <p>Se muestra una alerta indicando que se añadió correctamente el reporte.</p>
<p>Resultado: Satisfactorio</p>

3.2.4 Pruebas de aceptación para la Iteración II

Para una segunda iteración, se diseñaron un total de 14 casos de pruebas de aceptación. Todas enfocadas a evaluar la implementación del módulo economía. Se describen algunas de las pruebas realizadas, el resto pueden ser consultadas en los anexos de la investigación.

Tabla 73 Prueba de aceptación # 1

Caso de prueba de aceptación	
Código: HU6_P1	Historia de Usuario: 6
Nombre: Gestionar pre despacho	
Descripción: Prueba para la funcionalidad: añadir factura al pre despacho	
<p>Condiciones de ejecución:</p> <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol almacenero. 	
<p>Pasos de ejecución:</p> <p>El almacenero entra a la vista del almacén.</p> <p>Selecciona la opción pre despacho.</p> <p>Selecciona la opción añadir factura.</p> <p>Llena el formulario.</p> <p>Marca la opción aceptar</p>	
Resultado: Satisfactorio	

3.2.5 Pruebas de aceptación para la Iteración III

Para una tercera iteración, se diseñaron un total de 6 casos de pruebas de aceptación. Todas enfocadas a evaluar la implementación del módulo economía. Se describen algunas de las pruebas realizadas, el resto pueden ser consultadas en los anexos de la investigación.

Tabla 74 Prueba de aceptación # 1

Caso de prueba de aceptación	
Código: HU15_P1	Historia de Usuario: 15
Nombre: Administrar facturas internas	
Descripción: Prueba para la funcionalidad: el añadir facturas internas al sistema	
Condiciones de ejecución: <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de economía. 	
Pasos de ejecución: El director de la economía entra a la vista principal del sistema Se selecciona el campo Facturas internas Selecciona el campo Nueva Realiza el llenado de todos los campos Selecciona el campo añadir Se muestra una alerta indicando que se añadió correctamente la factura.	
Resultado: Satisfactorio	

Tabla 75 Prueba de aceptación # 2

Caso de prueba de aceptación	
Código: HU15_P2	Historia de Usuario: 15
Nombre: Administrar facturas internas	

Descripción: Prueba para la funcionalidad: eliminar facturas internas
Condiciones de ejecución: <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la economía.
Pasos de ejecución: El director de economía entra a la vista del módulo economía. Se selecciona la opción facturas internas. Selecciona la opción eliminar. Marca la opción aceptar.
Resultado: Satisfactorio

3.2.6 Análisis de las pruebas de aceptación

Hasta el momento se han desarrollado un total de 31 casos de pruebas de aceptación. Estas pruebas fueron realizadas de forma organizada, por cada iteración definida. A continuación, se muestra en gráficas, los porcentos de satisfacción alcanzados en cada iteración.

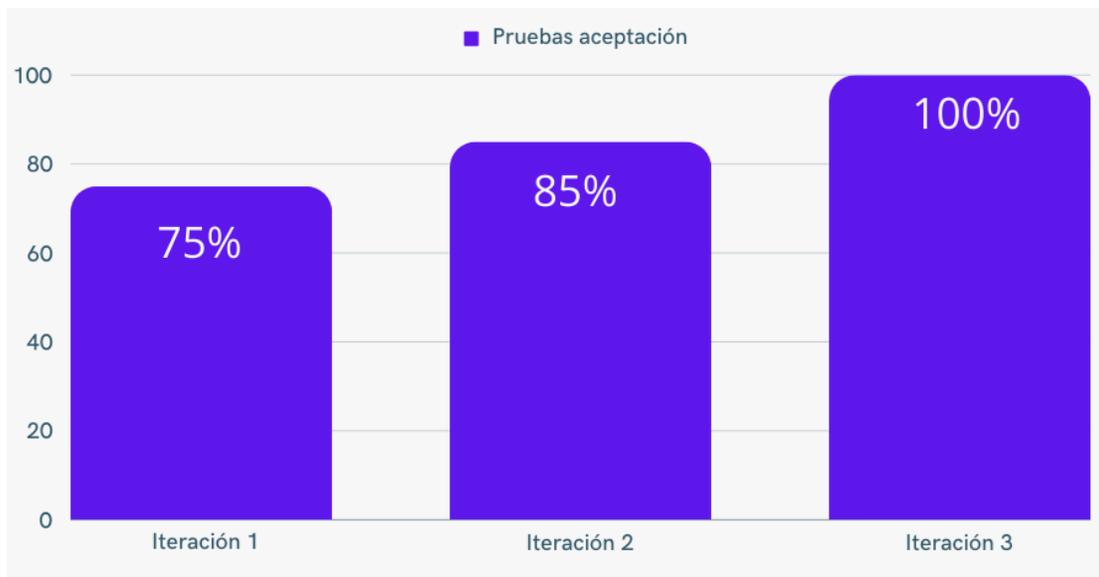


Ilustración 7. Resultados de las pruebas de aceptación

En la imagen se puede observar que en la primera iteración se alcanzó un 75% de satisfacción. En tanto, en la iteración II, las pruebas evaluadas alcanzaron un 85% de resultado satisfactorio. Mientras que en la Iteración III, se alcanzó un 100% de satisfacción. Con estas comprobaciones, se obtiene un 100% de satisfacción en la iteración final del producto, comprobando el correcto funcionamiento de las funcionalidades implementadas.

El correcto resultado de las mismas facilitó la entrega por parte del cliente de un aval de satisfacción, el cual puede ser consultado en los Anexo: Ilustración 7 Aval de satisfacción de cliente. Fuente: Dirección de la MIPYME Cubamodela. Este documento tiene como propósito expresar su conformidad con el software entregado.

Conclusiones del capítulo

Este capítulo ha sido fundamental en la materialización de nuestro sistema de gestión de almacenes y economía.

- Las tareas de ingeniería permitieron una ejecución eficiente de las funcionalidades requeridas, proporcionando una estructura organizada y orientada a resultados.
- Un componente esencial de este capítulo fue la definición y aplicación de pruebas de aceptación a los módulos implementados. Estas pruebas aseguraron que todo funcionara de acuerdo a las especificaciones de las HU. Este proceso de validación fue esencial para garantizar la calidad y la integridad del sistema en desarrollo.
- Las pruebas unitarias proporcionaron una oportunidad temprana para identificar problemas o inconsistencias en el software, lo que permitió realizar correcciones oportunas y evitó que afectaran negativamente el progreso del proyecto.

En resumen, el Capítulo 3 se centró en la fase de implementación, desglosando las HU en tareas de ingeniería, aplicando pruebas de aceptación y garantizando el cumplimiento de las HU, las pruebas unitarias desempeñaron un papel esencial en la identificación y corrección temprana de posibles fallos, contribuyendo a la calidad del sistema. Este proceso sentó las bases para la fase final de la investigación, que se enfocará en la evaluación y validación del sistema desarrollado.

Conclusiones

Con el desarrollo de la presente investigación se arriba a las siguientes conclusiones:

- Se desarrollo un sistema que sea capaz de mejorar los procesos de gestión empresarial en la MIPYME Cubamodela.
- Las funcionalidades que fueron añadidas en la solución propuesta demostraron su utilidad y calidad.
- La arquitectura y tecnologías utilizadas fueron elementos clave en el éxito de la solución desarrollada.
- Con la aplicación de las pruebas se aseguró la calidad del software. Con las pruebas unitarias se logró validar el correcto funcionamiento de unidades individuales de código, mientras que las pruebas de aceptación se consiguieron evaluar si el sistema en su conjunto cumple con los requisitos y expectativas del usuario, donde se obtuvo un aval de satisfacción de cliente que demuestra el éxito de la solución propuesta.

Recomendaciones

A partir de los resultados obtenidos se recomienda para próximas versiones:

- Desarrollar una aplicación móvil para el sistema desarrollado en la presente investigación.

Referencia Bibliográfica

- affden. (2023). Economía empresarial: definición, características y tipos - Affde Marketing. <https://www.affde.com/es/business-economics.html>
- Alberto, L., Santillán, C., Gibert, M., Óscar, G., & Mora, P. (2014). Bases de datos en MySQL. LAC Santillán, 5–5.
- Andres Hernández, H., Yeimi, ;, Cruz-Gil, L., María, ;, Puentes-Saavedra, D., Diego, ;, & Mendoza-Patiño, E. (2021). Diseño de un sistema de gestión de inventarios para el almacén TÉCNITALLER S.A.S de la ciudad Neiva-Huila, Colombia. *Revista de Investigaciones Universidad Del Quindío*, 33(2), 143–152. <https://doi.org/10.33975/RIUQ.VOL33N2.562>
- Angel, M., Autores, S., Menzinsky, A., López, G., Palacio, J., Sobrino, M. Á., Álvarez, R., & Rivas, V. (2022). Historias de Usuario Ingeniería de Requisitos Ágil Imagen de cubierta.
- ANGULAR. (2023). Angular. 2023. <https://angular.io/>
- Calcaneo Monts, M. A. (2021). Internet, redes sociales y libertad de expresión. *Cuestiones Constitucionales*, 1(44), 35–54. <https://doi.org/10.22201/IJ.24484881E.2021.44.16157>
- Cano Josías Caleb Pág, B., & Cano Josías Caleb, B. (2019). Importancia de la gestión de almacenes en las empresas: revisión de la literatura. *Universidad Privada Del Norte*. <https://repositorio.upn.edu.pe/handle/11537/15287>
- Capó Terror, F. (2022). La integración de SAP en la Administración Pública. <https://riunet.upv.es/handle/10251/188877>
- Damián De Luca. (2023, April 18). Visual Studio Code: características principales. <https://damiandeluca.com.ar/visual-studio-code-caracteristicas-principales>
- Danilo Gavilánez Alvarez, O. I., Layedra, N. I., & Ramos III, V. (2022). Análisis comparativo de Patrones de Diseño de Software. 7(7), 2145–2565. <https://doi.org/10.23857/pc.v7i7>
- De, C., Vallejo, E. X., Revisor, B., & Lozada, L. H. (2021). REPÚBLICA DEL ECUADOR.
- De Contabilidad, C., Finanzas, Y., Del, D., Montenegro, R., Sandra, S., Tanta, M., Asesor, A., Emma, M., & Blas, G. T. (2020). Principios para una adecuada gestión del almacén: una revisión de la literatura científica en los últimos diez años. *Universidad Privada Del Norte*. <https://repositorio.upn.edu.pe/handle/11537/24958>
- De Ingeniería, F., Arquitectura, Y., Hilasaca, S., Jhonny, A., Peralta, Y., Asesor, M., & Gutierrez Quispe, I. E. (2020). Desarrollo de un sistema de control de inventarios para pymes comercializadoras aplicando la metodología personalizada de XP. <https://repositorio.upeu.edu.pe/handle/20.500.12840/3930>
- Django. (2023). El framework web para perfeccionistas con plazos | Django. <https://www.djangoproject.com/>

- Fabián, C., Rojas, R., Fernanda, M., Ortiz, C., & Becerra Márquez, N. A. (2020). Estrategias para la Implementación del Comercio Electrónico en las Mipymes del Municipio de Cúcuta. *Visión Internacional (Cúcuta)*, 4(1), 40–50. <https://doi.org/10.22463/27111121.3401>
- Flask. (2023). Bienvenido a Flask — Documentación de Flask (2.3.x). <https://flask.palletsprojects.com/en/2.3.x/>
- Flores, D., Hugo, V., & Rojas, P. (2021). Sistema web de correspondencia para la gestión de documentación electrónica aplicando firma digital en los trámites del Gobierno Autónomo Municipal de Entre Ríos, Bolivia. *Revista Boliviana de Ingeniería*, 3(1), 91–117. <https://doi.org/10.33996/REBI.V3I1.7>
- Flores Dávila, J., Lara, M. C., Superior, I., Stanford -Ecuador, T., Ortega, T. J., Melendres Medina, G. E., & Acosta Velarde, R. (2019). PROCESO ADMINISTRATIVO, CONTABLE EN LOS NEGOCIOS Y LA COMPETITIVIDAD PARA LA GESTIÓN ECONÓMICA LOCAL. *Revista MktDescubre*.
- Gauthier, M., & Budán, P. D. (2023). SETIC: un Software Educativo sobre el Funcionamiento de las Partes de un Computador.
- Germain Ramírez, & Cristal Esmeralda. (2020). UNIVERSIDAD POLITÉCNICA DE SINALOA PROGRAMA ACADÉMICO DE INGENIERÍA EN INFORMÁTICA.
- Graciela, S., Ibarra, P., Quispe, R., Mullicundo, F. F., Lamas, D. A., & Presente, L. (2021). Herramientas y tecnologías para el desarrollo web desde el FrontEnd al BackEnd. XXIII Workshop de Investigadores En Ciencias de La Computación (WICC 2021, Chilecito, La Rioja), August 2021, 963–968. <http://sedici.unlp.edu.ar/handle/10915/120476>
- Gutiérrez Janampa, J. A., & Tapia, J. (2020). Liquidez y rentabilidad. Una revisión conceptual y sus dimensiones. *Revista de Investigación Valor Contable*, 3(1), 9–30. <https://doi.org/10.17162/RIVC.V3I1.1229>
- Hernández Barrueco, L. C., & Hernández Ramos, E. M. (2021). Manual del comercio electrónico. <https://books.google.com.mx/books?hl=es&lr=&id=ybPTEAAAQBAJ&oi=fnd&pg=PA5&dq=libro+Manual+del+comercio+electr%C3%B3nico&ots=67aqoylAKD&sig=jpvkKWYThfCsX5-tVHeA9LI0dO4#v=onepage&q=libro%20Manual%20del%20comercio%20electr%C3%B3nico&f=false>
- Instituto de Estudios Superiores del Valle de Orizaba. (2019). COMPRENDER EL ENTORNO. CAMINOS HACIA EL CONOCIMIENTO. www.univo.edu.mx
- Java. (2023). Que es Java. https://www.java.com/es/download/help/whatis_java.html
- Julio Quintana, P. del C. (2021). IMPORTANCIA DEL MODELO DE GESTIÓN EMPRESARIAL PARA LAS ORGANIZACIONES MODERNAS. *Revista Enfoques*, 4(16), 272–283. <https://doi.org/10.33996/revistaenfoques.v4i16.99>
- Library. (2023). Fase de planificación - Programación extrema (Extreme programming). <https://1library.co/article/fase-de-planificaci%C3%B3n-programaci%C3%B3n-extrema-extreme-programming.yj718e2p>

- López, A. V., Carballo, M. M., & Hechavarría, L. T. (2021). Acciones para la gestión del comercio electrónico en la agencia de viajes Cubana Ecotur. *REVISTA CIENTÍFICA ECOCIENCIA*, 8(1), 18–48. <https://doi.org/10.21855/ECOCIENCIA.81.408>
- Marín, L. E. (2018). Gestión de almacenes para el fortalecimiento de la administración. <https://www.eumed.net/rev/oel/2018/11/almacenes-inventarios.html>
- Microsoft. (2023). Bienvenido a Microsoft Dynamics 365 Business Central - Business Central | Microsoft Learn. <https://learn.microsoft.com/es-es/dynamics365/business-central/>
- Montserrat, Y., Juárez, L., Alejandro, L., Tirado, M., & Beltrán López, R. (2020). Online marketing: rentabilidad al alcance de las Mipymes. *Revista Visión Contable*, 21, 57–76. <https://doi.org/10.24142/RVC.N21A3>
- Muñiz del Valle, V. M., Feliciano Morales, S., Solís Carmona, E., Hernández-Hernández, J. L., Hernández Hernández, M., Álvarez Hilario, V., FELICIANO MORALES, S. 870008, Solís Carmona, E.-0002-5346-0428, Hernández-Hernández, J. L.-0003-0231-2019, Hernández Hernández, M.-0001-8330-4779, & Álvarez Hilario, V.-0002-5853-4246. (2019). Operaciones crud en datos semiestructurados. *Exploraciones, Intercambios y Relaciones Entre El Diseño y La Tecnología*, 11(9), 57–79. <https://doi.org/10.16/CSS/JQUERY.DATATABLES.MIN.CSS>
- Oracle. (2023). Gestión de almacenes | Oráculo. <https://www.oracle.com/scm/logistics/warehouse-management/>
- Ortega-Hernandez, V. de J. (2019). Diseño de una metodología para la estimación de esfuerzos en proyectos de consultoría en innovación. <https://reunir.unir.net/handle/123456789/8280>
- Ovallos Ovallos, J. A., & Medina Cardenas, Y. (2020). A practical guide to analyzing vulnerabilities in a GNU/Linux client-server environment using a pentesting methodology. <http://www.aisti.eu>
- Pérez A, O. A. (2023). Vista de Cuatro enfoques metodológicos para el desarrollo de Software RUP – MSF – XP – SCRUM. <https://revistas.uniminuto.edu/index.php/Inventum/article/view/9/9>
- Quesquén, A. A. C. (2020). Gestión de almacén e inventarios para reducir los costos de inventarios en un almacén de productos terminados. *Revista de Investigación Multidisciplinaria CTSCAFE*, 4(12), 14–14. <http://www.ctscafe.pe/index.php/ctscafe/article/view/133>
- Ramírez Ramírez, M., Del Consuelo, M., Soto, S., Beatriz, H., Moreno, R., Rojas, E. M., Del Carmen, N., Millán, O., Fernando, R., & Cisneros, R. (2018). Metodología SCRUM y desarrollo de Repositorio Digital. *Risti*, 7.
- react. (2023). React. <https://es.react.dev/>
- Restrepo, M. C., González, C. C., & Serna, D. C. (2020). El control interno de los inventarios: su incidencia en la gestión financiera de “Due Amici Pizzería.” *Revista Activos*, 18(2), 137–163. <https://doi.org/10.15332/25005278/6264>

- Revista CEA. (2021, November 18). Estrategias de marketing digital utilizadas por empresas del retail deportivo (Digital Marketing Strategies Used by Sports Retail Companies) by Sergio Luque-Ortiz :: SSRN. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3788414
- Salazar, J. C., Casallas, Á. T., Linares, J. C., Lozano, A., & Valbuena, Y. L. (2018). Scrum versus XP: similitudes y diferencias. *Tecnología Investigación y Academia*, 6(2), 29–37. <https://revistas.udistrital.edu.co/index.php/tia/article/view/10496>
- Suárez González, L., & Pérez Pérez, A. C. (2023, July). Transformación digital en las empresas de comercio electrónico: Una mirada desde la optimización de rutas | *Revista Cubana de Transformación Digital*. *Revista Cubana Transformación Digital*. <https://rctd.uic.cu/rctd/article/view/162>
- Vázquez del Cerro, P. (2020). Frameworks y lenguajes de desarrollo para Frontend. *MoleQla: Revista de Ciencias de La Universidad Pablo de Olavide*, ISSN-e 2173-0903, N°. 36, 2020, 36, 6. <https://dialnet.unirioja.es/servlet/articulo?codigo=7220422&info=resumen&idioma=SPA>
- Vinent Domenech, J. E. (2021). Módulo para la gestión del proceso de solicitud de certificación de sentencias en el Sistema de Tribunales [Universidad de Ciencias Informática]. <https://www.oracle.com/scm/logistics/warehouse-management/>
- Vite Cevallos, H., Molina Montero, K., & Dávila Cuesta, J. (n.d.). (PDF) Metodologías ágiles frente a las tradicionales en el proceso de desarrollo de software. Retrieved September 11, 2023, from https://www.researchgate.net/publication/327537074_Metodologias_agiles_frente_a_las_tradicionales_en_el_proceso_de_desarrollo_de_software
- Vue.js. (2023). Vue.js - El marco progresivo de JavaScript | Vue.js. <https://vuejs.org/>
- Yenisel Molina Hernández, I., Ailec Granda Dihigo, D., & Velázquez Cintra, A. (2019). Los requisitos no funcionales de software. Una estrategia para su desarrollo en el Centro de Informática Médica. *Revista Cubana de Ciencias Informáticas*, 13(2), 77–90. http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2227-18992019000200077&lng=es&nrm=iso&tlng=es

Anexos

1. Entrevista

Nombre de los entrevistadores: Estudiantes Roxana Turro Aranda y Carlos Daniel Gisbert Morales.

Centro de estudio: Universidad de las ciencias informáticas (UCi).

Nombre del entrevistado: Ing. Alejandro Peñalver Mauri

Centro de trabajo: Dirección General de la MIPYME Cubamodela.

Objetivo:

Obtener información sobre el proceso de almacenado de productos y gestión económica en la Mipyme Cubamodela, para que el sistema de gestión empresarial cumpla con los requerimientos necesarios para su puesta en funcionamiento.

Preguntas:

1. Inicialmente es necesario saber ¿Cómo se les da entrada y salida a los productos del almacén?
2. ¿De qué forma se almacenan los productos en el almacén?
3. ¿Quiénes son los actores que actúan en este proceso?
4. ¿Qué información se recopila para dar entrada a un producto en almacén?
5. ¿Cómo se realiza el proceso contable de la empresa?
6. ¿Cuáles son los procesos contables que se realizan para realizar la economía de la empresa?
7. ¿A qué información debe de tener acceso la directora de economía para poder gestionar los procesos contables de la empresa?
8. ¿Quiénes son las personas autorizadas a obtener esta información?

2. Observación

Nombre del trabajador: Alejandro Peñalver Mauri	
Cargo: director general de la empresa	
1. Organización y estructura de la empresa	Relevancia
Tienda online	Alta
Recepción	Media
Almacén	Alta
Economía	Alta
2. Claridad de las normas:	
Los trabajadores saben cómo realizar su trabajo y las responsabilidades que tienen	
Los trabajadores saben a q atenerse en cada momento	
3. Relación de departamentos	
Todos los departamentos se relacionan entre ellos	

Los departamentos dependen unos de otros
4. Relación con el actor
Cada departamento tiene una persona a cargo
Cada actor responde por cualquier problema que ocurra en su departamento

3. Tareas de ingeniería asociadas a Historias de Usuarios

3.2 Tareas de ingeniería para la Iteración I

Tabla 38 Tarea de ingeniería # 6

Tarea	
Número de tarea: 6	Número de Historia de Usuario: 2
Nombre de la tarea: Implementar el envío de correos	
Tipo de tarea: Desarrollo	Puntos estimados: 0.5
Fecha de inicio: 29 de julio de 2023	Fecha de fin: 2 de julio de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que le permite al almacenero enviar una notificación al director de la tienda notificando que hay un nuevo producto añadido.	

Tabla 39 Tarea de ingeniería # 7

Tarea	
Número de tarea: 7	Número de Historia de Usuario: 3
Nombre de la tarea: Implementar la función añadir nuevos productos al almacén	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 3 de julio de 2023	Fecha de fin: 6 de julio de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que le permite al almacenero añadir nuevos productos al almacén	

Tabla 40 Tarea de ingeniería # 8

Tarea	
Número de tarea: 8	Número de Historia de Usuario:3
Nombre de la tarea: Implementar la función modificar productos al almacén	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 7 de julio de 2023	Fecha de fin: 9 de julio de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que le permite al almacenero modificar los productos del almacén	

Tabla 41 Tarea de ingeniería # 9

Tarea	
Número de tarea: 9	Número de Historia de Usuario:3
Nombre de la tarea: Implementar la función eliminar productos al almacén	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 10 de julio de 2023	Fecha de fin: 12 de julio de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que le permite al almacenero eliminar productos del almacén	

Tabla 42 Tarea de ingeniería # 10

Tarea	
Número de tarea: 10	Número de Historia de Usuario:3
Nombre de la tarea: Implementar la función listar productos al almacén	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 13 de julio de 2023	Fecha de fin: 15 de julio de 2023

Programador responsable: Roxana Turro Aranda
Descripción: Se implementa una funcionalidad, que permita listar los productos del almacén

Tabla 43 Tarea de ingeniería # 11

Tarea	
Número de tarea: 11	Número de Historia de Usuario:3
Nombre de la tarea: Implementar la función descontar productos del almacén	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 16 de julio de 2023	Fecha de fin: 19 de julio de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que permita descontar los productos del almacén por petición de la tienda.	

Tabla 44 Tarea de ingeniería # 12

Tarea	
Número de tarea: 12	Número de Historia de Usuario:4
Nombre de la tarea: Implementar la función visualizar historial de inventario	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 20 de julio de 2023	Fecha de fin: 22 de julio de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad, que permita que a través de una interfaz el gestor de negocios pueda visualizar los productos del almacén	

Tabla 45 Tarea de ingeniería # 13

Tarea	
Número de tarea: 13	Número de Historia de Usuario:4

Nombre de la tarea: Implementar la función modificar la descripción a los productos	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 23 de julio de 2023	Fecha de fin: 25 de julio de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad, que permita que a través de una interfaz el almacenero pueda modificar la descripción a los productos	

Tabla 46 Tarea de ingeniería # 14

Tarea	
Número de tarea: 14	Número de Historia de Usuario:5
Nombre de la tarea: Implementar la conversión para generar gráficos de línea	
Tipo de tarea: Desarrollo	Puntos estimados: 0.5
Fecha de inicio: 26 de julio de 2023	Fecha de fin: 31 de julio de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad, que permita que generar gráficos de línea que permita ver los análisis de ventas por días, semanas, quincenas, meses y años	

3.3 Tareas de ingeniería para la Iteración II

Tabla 47 Tarea de ingeniería # 15

Tarea	
Número de tarea: 15	Número de Historia de Usuario:1
Nombre de la tarea: Implementar la función de añadir una factura al pre despacho	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 4 de agosto de 2023	Fecha de fin: 6 de agosto de 2023

Programador responsable: Carlos Daniel Gisbert Morales
Descripción: Se implementa una funcionalidad, que permita que añadir una nueva factura al pre despacho.

Tabla 48 Tarea de ingeniería # 16

Tarea	
Número de tarea: 16	Número de Historia de Usuario:1
Nombre de la tarea: Implementar la función de modificar una orden en el pre despacho	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 7 de agosto de 2023	Fecha de fin: 9 de agosto de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad, que permita que modificar una nueva factura del pre despacho.	

Tabla 49 Tarea de ingeniería # 17

Tarea	
Número de tarea: 17	Número de Historia de Usuario:1
Nombre de la tarea: Implementar la función de eliminar una orden en el pre despacho	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 10 de agosto de 2023	Fecha de fin: 12 de agosto de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad, que permita que eliminar una nueva factura del pre despacho.	

Tabla 50 Tarea de ingeniería # 18

Tarea	
Número de tarea: 18	Número de Historia de Usuario:2
Nombre de la tarea: Implementar la función de mostrar detalles de entrada.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 13 de agosto de 2023	Fecha de fin: 16 de agosto de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que permita visualizar los detalles cada vez que, se marque la opción ver detalle de entrada con la descripción del producto junto con la fecha y descripción de entrada.	

Tabla 51 Tarea de ingeniería # 19

Tarea	
Número de tarea: 19	Número de Historia de Usuario:3
Nombre de la tarea: Implementar la función de generar historial de productos dañados o con salida especial	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 17 de agosto de 2023	Fecha de fin: 18 de agosto de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que permita realizar un historial con todos los productos dañados que hayan existido en el almacén, al igual que con los productos que han salido del almacén por causas especiales.	

Tabla 52 Tarea de ingeniería # 20

Tarea	
Número de tarea: 20	Número de Historia de Usuario:4
Nombre de la tarea: Implementar la función de añadir usuario	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3

Fecha de fin: 19 de agosto de 2023	Fecha de fin: 21 de agosto de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad, que permita al director de la tienda añadir los usuarios que desempeñaran los roles de almacenero, gestor de negocio y director de economía	

Tabla 53 Tarea de ingeniería # 21

Tarea	
Número de tarea: 21	Número de Historia de Usuario:4
Nombre de la tarea: Implementar la función de modificar usuario	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de fin: 22 de agosto de 2023	Fecha de fin: 24 de agosto de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad, que permita modificar los usuarios que interactúen con la tienda	

Tabla 54 Tarea de ingeniería # 22

Tarea	
Número de tarea: 22	Número de Historia de Usuario:4
Nombre de la tarea: Implementar la función de eliminar usuario	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de fin: 25 de agosto de 2023	Fecha de fin: 26 de agosto de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad, que permita eliminar los usuarios que interactúen con la tienda	

Tabla 55 Tarea de ingeniería # 23

Tarea

Número de tarea: 23	Número de Historia de Usuario:4
Nombre de la tarea: Implementar la función de listar usuario	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de fin: 27 de agosto de 2023	Fecha de fin: 28 de agosto de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad, que permita listar los usuarios que interactúen con la tienda	

Tabla 56 Tarea de ingeniería # 24

Tarea	
Número de tarea: 24	Número de Historia de Usuario:5
Nombre de la tarea: Implementar la función de añadir almacén	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de fin: 29 de agosto de 2023	Fecha de fin: 1 de septiembre de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que permita añadir un almacén nuevo	

Tabla 57 Tarea de ingeniería # 25

Tarea	
Número de tarea: 25	Número de Historia de Usuario:5
Nombre de la tarea: Implementar la función de modificar almacén	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 2 de septiembre de 2023	Fecha de fin: 3 de septiembre de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que permita modificar un almacén	

Tabla 58 Tarea de ingeniería # 26

Tarea	
Número de tarea: 26	Número de Historia de Usuario:5
Nombre de la tarea: Implementar la función de eliminar almacén	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 4 de septiembre de 2023	Fecha de fin: 6 de septiembre de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que permita eliminar un almacén	

Tabla 59 Tarea de ingeniería # 27

Tarea	
Número de tarea: 27	Número de Historia de Usuario:5
Nombre de la tarea: Implementar la función de listar almacenes	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 6 de septiembre de 2023	Fecha de fin: 7 de septiembre de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que permita listar los almacenes.	

3.4 Tareas de ingeniería para la Iteración III

Tabla 60 Tarea de ingeniería # 28

Tarea

Número de tarea: 28	Número de Historia de Usuario:1
Nombre de la tarea: Implementar la función de mostrar ventas	
Tipo de tarea: Desarrollo	Puntos estimados: 0.6
Fecha de inicio: 8 de septiembre de 2023	Fecha de fin: 13 de septiembre de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad, que permita visualizar todas las ventas diarias, semanales, quincenales, mensuales y anuales de la tienda, y supervisando los márgenes de ganancia por ellas.	

Tabla 61 Tarea de ingeniería #29

Tarea	
Número de tarea: 29	Número de Historia de Usuario:2
Nombre de la tarea: Implementar la función de visualización de rentabilidad a través de gráficos.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.5
Fecha de inicio: 14 de septiembre de 2023	Fecha de fin: 18 de septiembre de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad, que permita visualizar a través de gráficos todo el análisis de rentabilidad de la tienda.	

Tabla 62 Tarea de ingeniería # 30

Tarea	
Número de tarea: 30	Número de Historia de Usuario:3
Nombre de la tarea: Implementar la función de añadir pago.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 19 de septiembre de 2023	Fecha de inicio: 22 de septiembre de 2023
Programador responsable: Roxana Turro Aranda	

Descripción: Se implementa una funcionalidad, que permita añadir un pago a embajadores y a los proveedores de la tienda.

Tabla 63 Tarea de ingeniería #31

Tarea	
Número de tarea: 31	Número de Historia de Usuario:3
Nombre de la tarea: Implementar la función de modificar pago.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 22 de septiembre de 2023	Fecha de inicio: 24 de septiembre de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que permita modificar los pagos a los embajadores y a los proveedores de la tienda.	

Tabla 64 Tarea de ingeniería #32

Tarea	
Número de tarea: 32	Número de Historia de Usuario:3
Nombre de la tarea: Implementar la función de eliminar pago.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 25 de septiembre de 2023	Fecha de inicio: 27 de septiembre de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que permita modificar los pagos a los embajadores y a los proveedores de la tienda.	

Tabla 65 Tarea de ingeniería #33

Tarea	
Número de tarea: 33	Número de Historia de Usuario:3
Nombre de la tarea: Implementar la función de listar pagos.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 28 de septiembre de 2023	Fecha de inicio: 30 de septiembre de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que permita visualizar los pagos a los embajadores y a los proveedores de la tienda.	

Tabla 66 Tarea de ingeniería #34

Tarea	
Número de tarea: 34	Número de Historia de Usuario:4
Nombre de la tarea: Implementar la función de buscar productos o facturas	
Tipo de tarea: Desarrollo	Puntos estimados: 0.3
Fecha de inicio: 1 de octubre de 2023	Fecha de inicio: 3 de septiembre de 2023
Programador responsable: Roxana Turro Aranda	
Descripción: Se implementa una funcionalidad, que permita filtrar los productos y facturas	

Tabla 67 Tarea de ingeniería #35

Tarea	
Número de tarea: 35	Número de Historia de Usuario:5
Nombre de la tarea: Implementar la función de eliminar facturas internas	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 4 de octubre de 2023	Fecha de inicio: 5 de octubre de 2023

Programador responsable: Carlos Daniel Gisbert Morales
Descripción: Se implementa una funcionalidad, que permita eliminar las facturas internas.

Tabla 68 Tarea de ingeniería #36

Tarea	
Número de tarea: 36	Número de Historia de Usuario:5
Nombre de la tarea: Implementar la función de listar facturas internas	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 6 de octubre de 2023	Fecha de inicio: 7 de octubre de 2023
Programador responsable: Carlos Daniel Gisbert Morales	
Descripción: Se implementa una funcionalidad, que permita listar las facturas internas.	

4. Pruebas de Aceptación

4.1 Pruebas de aceptación para la Iteración I

Tabla 76 Prueba de aceptación # 5

Caso de prueba de aceptación	
Código: HU2_P1	Historia de Usuario: 2
Nombre: Enviar correo	
Descripción: Prueba para la funcionalidad: enviar correo	
Condiciones de ejecución:	

<ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol almacenero.
<p>Pasos de ejecución:</p> <p>El almacenero entra a la vista del inventario del almacén.</p> <p>Selecciona la opción añadir producto.</p> <p>Al dar en el botón aceptar se enviará automáticamente el correo al director de la tienda.</p>
Resultado: Satisfactorio

Tabla 77 Prueba de aceptación # 6

Caso de prueba de aceptación	
Código: HU3_P1	Historia de Usuario: 3
Nombre: Gestionar la entrada y salida de productos en almacén	
Descripción: Prueba para la funcionalidad: añadir nuevo producto al almacén	
<p>Condiciones de ejecución:</p> <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol almacenero. 	
<p>Pasos de ejecución:</p> <p>El almacenero entra a la vista del inventario del almacén.</p> <p>Selecciona la opción añadir nuevo producto.</p> <p>Llena la descripción del producto</p> <p>Marca la opción aceptar</p>	
Resultado: Satisfactorio	

Tabla 78 Prueba de aceptación # 7

Caso de prueba de aceptación

Código: HU3_P2	Historia de Usuario: 3
Nombre: Gestionar la entrada y salida de productos en almacén	
Descripción: Prueba para la funcionalidad: modificar producto en el almacén	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol almacenero. 	
Pasos de ejecución:	
El almacenero entra a la vista del inventario del almacén.	
Selecciona la opción modificar producto.	
Modifica la descripción del producto	
Marca la opción aceptar	
Resultado: Satisfactorio	

Tabla 79 Prueba de aceptación # 8

Caso de prueba de aceptación	
Código: HU3_P3	Historia de Usuario: 3
Nombre: Gestionar la entrada y salida de productos en almacén	
Descripción: Prueba para la funcionalidad: eliminar producto en el almacén	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol almacenero o director de la tienda. 	

<p>Pasos de ejecución:</p> <p>El almacenero o director de la tienda entra a la vista del inventario del almacén.</p> <p>Selecciona la opción eliminar producto.</p> <p>Selecciona la opción por merma, salida especial, por venta</p> <p>Marca la opción aceptar</p>
Resultado: Satisfactorio

Tabla 80 Prueba de aceptación # 9

Caso de prueba de aceptación	
Código: HU3_P4	Historia de Usuario: 3
Nombre: Gestionar la entrada y salida de productos en almacén	
Descripción: Prueba para la funcionalidad: listar producto en el almacén	
<p>Condiciones de ejecución:</p> <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol almacenero o director de la tienda. 	
<p>Pasos de ejecución:</p> <p>El almacenero o director de la tienda entra a la vista del almacén.</p> <p>Selecciona la opción inventario.</p>	
Resultado: Satisfactorio	

Tabla 81 Prueba de aceptación # 10

Caso de prueba de aceptación	
Código: HU4_P1	Historia de Usuario: 4

Nombre: Visualizar Productos
Descripción: Prueba para la funcionalidad: Visualizar Productos del almacén.
Condiciones de ejecución: <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol gestor de negocio.
Pasos de ejecución: El gestor de negocio entra a la vista del almacén. Selecciona la opción inventario.
Resultado: Satisfactorio

Tabla 82 Prueba de aceptación # 11

Caso de prueba de aceptación	
Código: HU5_P1	Historia de Usuario: 5
Nombre: Generar gráficos con márgenes de ventas por productos	
Descripción: Prueba para la funcionalidad: Generar gráficos con márgenes de ventas por productos	
Condiciones de ejecución: <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol almacenero y director de la tienda. 	
Pasos de ejecución: El almacenero o director de la tienda entra a la vista del almacén. Selecciona la opción gráficos con ventas por productos.	
Resultado: Satisfactorio	

4.2 Pruebas de aceptación para la Iteración II

Tabla 83 Prueba de aceptación # 2

Caso de prueba de aceptación	
Código: HU6_P2	Historia de Usuario: 6
Nombre: Gestionar pre despacho	
Descripción: Prueba para la funcionalidad: modificar factura en el pre despacho	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol almacenero. 	
Pasos de ejecución:	
El almacenero entra a la vista del almacén.	
Selecciona la opción pre despacho.	
Selecciona la opción modificar factura.	
Modifica el formulario.	
Marca la opción aceptar	
Resultado: Satisfactorio	

Tabla 84 Prueba de aceptación # 3

Caso de prueba de aceptación	
Código: HU6_P3	Historia de Usuario: 6
Nombre: Gestionar pre despacho	
Descripción: Prueba para la funcionalidad: eliminar factura en el pre despacho	
Condiciones de ejecución:	

<ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol almacenero.
<p>Pasos de ejecución:</p> <p>El almacenero entra a la vista del almacén.</p> <p>Selecciona la opción pre despacho.</p> <p>Selecciona la opción eliminar factura.</p> <p>Marca la opción aceptar</p>
Resultado: Satisfactorio

Tabla 85 Prueba de aceptación # 4

Caso de prueba de aceptación	
Código: HU6_P4	Historia de Usuario: 6
Nombre: Gestionar pre despacho	
Descripción: Prueba para la funcionalidad: listar facturas en el pre despacho	
<p>Condiciones de ejecución:</p> <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol almacenero o director de la tienda. 	
<p>Pasos de ejecución:</p> <p>El almacenero entra a la vista del almacén.</p> <p>Selecciona la opción pre despacho.</p>	
Resultado: Satisfactorio	

Tabla 86 Prueba de aceptación # 5

Caso de prueba de aceptación	
Código: HU7_P1	Historia de Usuario: 7

Nombre: Mostrar detalles
Descripción: Prueba para la funcionalidad: mostrar detalle
Condiciones de ejecución: <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol almacenero o director de la tienda.
Pasos de ejecución: El almacenero entra a la vista del almacén. Selecciona la opción inventario. Selecciona la opción mostrar detalle. Marca la opción ok.
Resultado: Satisfactorio

Tabla 87 Prueba de aceptación # 6

Caso de prueba de aceptación	
Código: HU8_P1	Historia de Usuario: 8
Nombre: Generar historial de productos dañados o con salida especial	
Descripción: Prueba para la funcionalidad: Generar historial de productos dañados o con salida especial	
Condiciones de ejecución: <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol almacenero o director de la tienda. 	
Pasos de ejecución: El almacenero entra a la vista del almacén. Selecciona la opción salidas especiales.	
Resultado: Satisfactorio	

Tabla 88 Prueba de aceptación # 7

Caso de prueba de aceptación	
Código: HU9_P1	Historia de Usuario: 9
Nombre: Gestionar usuarios	
Descripción: Prueba para la funcionalidad: añadir nuevo usuario	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la tienda. 	
Pasos de ejecución:	
El director de la tienda entra a la vista de usuarios.	
Selecciona la opción añadir.	
Llena la información del nuevo usuario.	
Marca la opción aceptar.	
Resultado: Satisfactorio	

Tabla 89 Prueba de aceptación # 8

Caso de prueba de aceptación	
Código: HU9_P2	Historia de Usuario: 9
Nombre: Gestionar usuarios	
Descripción: Prueba para la funcionalidad: modificar usuario	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la tienda. 	

<p>Pasos de ejecución:</p> <p>El director de la tienda entra a la vista de usuarios.</p> <p>Selecciona la opción modificar.</p> <p>Modifica la información del nuevo usuario.</p> <p>Marca la opción aceptar.</p>
Resultado: Satisfactorio

Tabla 90 Prueba de aceptación # 9

Caso de prueba de aceptación	
Código: HU9_P3	Historia de Usuario: 9
Nombre: Gestionar usuarios	
Descripción: Prueba para la funcionalidad: eliminar usuario	
<p>Condiciones de ejecución:</p> <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la tienda. 	
<p>Pasos de ejecución:</p> <p>El director de la tienda entra a la vista de usuarios.</p> <p>Selecciona la opción eliminar.</p> <p>Marca la opción aceptar.</p>	
Resultado: Satisfactorio	

Tabla 91 Prueba de aceptación # 10

Caso de prueba de aceptación	
Código: HU9_P4	Historia de Usuario: 9

Nombre: Gestionar usuarios
Descripción: Prueba para la funcionalidad: listar usuario
Condiciones de ejecución: <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la tienda.
Pasos de ejecución: El director de la tienda entra a la vista de usuarios.
Resultado: Satisfactorio

Tabla 92 Prueba de aceptación # 11

Caso de prueba de aceptación	
Código: HU10_P1	Historia de Usuario: 10
Nombre: Gestionar almacenes	
Descripción: Prueba para la funcionalidad: añadir nuevo almacén	
Condiciones de ejecución: <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la tienda. 	
Pasos de ejecución: El director de la tienda entra a la vista de almacén. Selecciona la opción almacenes. Selecciona la opción añadir. Llena el formulario. Marca la opción aceptar.	
Resultado: Satisfactorio	

Tabla 93 Prueba de aceptación # 12

Caso de prueba de aceptación	
Código: HU10_P2	Historia de Usuario: 10
Nombre: Gestionar almacenes	
Descripción: Prueba para la funcionalidad: modificar almacén	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la tienda. 	
Pasos de ejecución:	
El director de la tienda entra a la vista de almacén.	
Selecciona la opción almacenes.	
Selecciona la opción modificar.	
modifica el formulario.	
Marca la opción aceptar.	
Resultado: Satisfactorio	

Tabla 94 Prueba de aceptación # 13

Caso de prueba de aceptación	
Código: HU10_P3	Historia de Usuario: 10
Nombre: Gestionar almacenes	
Descripción: Prueba para la funcionalidad: eliminar almacén	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la tienda. 	

Pasos de ejecución: El director de la tienda entra a la vista de almacén. Selecciona la opción almacenes. Selecciona la opción eliminar. Marca la opción aceptar.
Resultado: Satisfactorio

Tabla 95 Prueba de aceptación # 14

Caso de prueba de aceptación	
Código: HU10_P4	Historia de Usuario: 10
Nombre: Gestionar almacenes	
Descripción: Prueba para la funcionalidad: listar almacenes	
Condiciones de ejecución: • El usuario debe estar previamente autenticado y contar con rol director de la tienda.	
Pasos de ejecución: El director de la tienda entra a la vista de almacén. Selecciona la opción almacenes.	
Resultado: Satisfactorio	

4.3 Pruebas de aceptación para la Iteración III

Tabla 96 Prueba de aceptación # 3

Caso de prueba de aceptación	
Código: HU11_P1	Historia de Usuario: 11
Nombre: Visualizar Ventas	

Descripción: Prueba para la funcionalidad: visualizar Ventas
Condiciones de ejecución: <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de economía.
Pasos de ejecución: <p>El director de economía entra a la vista de economía.</p> <p>Selecciona la opción ventas.</p>
Resultado: Satisfactorio

Tabla 97 Prueba de aceptación # 4

Caso de prueba de aceptación	
Código: HU12_P1	Historia de Usuario: 12
Nombre: Generar gráficos de ventas	
Descripción: Prueba para la funcionalidad: generar gráficos de ventas.	
Condiciones de ejecución: <ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la economía. 	
Pasos de ejecución: <p>El director de economía entra a la vista del módulo economía.</p> <p>Se selecciona la opción que aparece arriba a la izquierda en la vista.</p> <p>Se selecciona la opción gráfica de ventas.</p> <p>Se muestra el grafico con las ventas por meses.</p>	
Resultado: Satisfactorio	

Tabla 98 Prueba de aceptación # 4

Caso de prueba de aceptación	
Código: HU13_P1	Historia de Usuario: 13
Nombre: Gestionar pagos	
Descripción: Prueba para la funcionalidad: gestionar pagos.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la economía. 	
Pasos de ejecución:	
El director de economía entra a la vista del módulo economía.	
Se selecciona la opción pago embajadores.	
Se selecciona la opción pago proveedores.	
Resultado: Satisfactorio	

Tabla 99 Prueba de aceptación # 5

Caso de prueba de aceptación	
Código: HU14_P1	Historia de Usuario: 14
Nombre: Buscar productos o facturas	
Descripción: Prueba para la funcionalidad: buscar productos o facturas	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario debe estar previamente autenticado y contar con rol director de la economía. 	
Pasos de ejecución:	
El director de economía entra a la vista del módulo economía.	
Se selecciona la opción buscar.	
Resultado: Satisfactorio	

Tabla 100 Prueba de aceptación # 6

Caso de prueba de aceptación	
Código: HU15_P3	Historia de Usuario: 15
Nombre: Administrar facturas internas	
Descripción: Prueba para la funcionalidad: listar facturas internas	
Condiciones de ejecución: <ul style="list-style-type: none">• El usuario debe estar previamente autenticado y contar con rol director de la economía.	
Pasos de ejecución: El director de economía entra a la vista del módulo economía. Se selecciona las facturas internas.	
Resultado: Satisfactorio	

4.4 Aval de satisfacción del cliente:



La Habana, 15 de noviembre de 2023

"Año 65 de la Revolución"

MIPYME Cubamodela.

ACTA DE CONFORMIDAD

Por este medio comunicamos nuestra satisfacción con el Sistema de Gestión Empresarial entregado por Carlos Daniel Gisbert Morales y Roxana Turro Aranda. Certificamos que el producto entregado cumplió con la totalidad de los requisitos especificados y fue entregado en los plazos establecidos. El sistema fue probado y cumple con los estándares requeridos. La MIPYME Cubamodela acepta el software en su estado actual y se compromete con la utilización del mismo en los términos y condiciones acordados.

Y PARA QUE CONSTE la aceptación del trabajo realizado, se firma la presente ACTA, cordialmente

Alejandro Peñalver Mauri

Director General de la

MIPYME Cubamodela

Ilustración 8. Aval de satisfacción de cliente. Fuente: Dirección de la MIPYME Cubamodela

5. Pruebas de Unitarias

```
# Crear un objeto SalidaProducto para su uso en las pruebas
self.salida_producto = SalidaProducto.objects.create(
    producto=self.producto,
    description="Salida 1",
)

def test_crear_salida_producto(self):
    # Comprobar que se puede crear una SalidaProducto
    self.assertEqual(self.salida_producto.description, "Salida 1")

def test_actualizar_salida_producto(self):
    # Actualizar la descripción de SalidaProducto y comprobar el
cambio
    self.salida_producto.description = "Salida Actualizada"
    self.salida_producto.save()
    self.assertEqual(self.salida_producto.description, "Salida
Actualizada")

def test_leer_salida_producto(self):
```

Ilustración 9. Pruebas unitarias al Gestionar entrada y salidas de productos del almacén.

```
class OrdenTestCase(TestCase):
    def setUp(self):
        # Crear objetos relacionados (Embajador y Producto) para su uso
        # en las pruebas
        self.embajador = Embajador.objects.create(
            name="Juan Pérez",
            referred_number=5,
            retribution_for_sales=100.0,
            retribution_for_fortnight=50.0,
            payment_date="2023-09-30 12:00:00",
            payment_status="Pendiente",
            payment_method="Transferencia"
        )
        self.producto = Producto.objects.create(
            name="Producto 1",
            sku="SKU001",
            picture="imagen.jpg",
            brand="Marca A",
            slug="slug-1",
            date_created="2023-09-30 12:00:00",
            date_modified="2023-09-30 12:00:00",
            type="Tipo A",
            status="Activo",
            on_sale=1,
            description="Descripción del producto",
            short_description="Descripción corta",
            price=99.99,
            regular_price=129.99,
            stock_quantity=100,
            cost=50.0,
            product_id=12345,
        )
```

Ilustración 10. Pruebas unitarias al Mostrar tablas de pagos

```

# Comprobar que se puede crear una Orden
self.assertEqual(self.orden.order_id, 1)

def test_actualizar_orden(self):
# Actualizar el número de la Orden y comprobar el cambio
self.orden.number = 456
self.orden.save()
self.assertEqual(self.orden.number, 456)

def test_leer_orden(self):
# Leer la Orden desde la base de datos y comprobar si los datos
son correctos
orden_leida = Orden.objects.get(pk=self.orden.pk)
self.assertEqual(orden_leida.order_key, "clave-123")

def test_eliminar_orden(self):
# Eliminar la Orden y comprobar que ya no exista en la base de
datos
self.orden.delete()
with self.assertRaises(Orden.DoesNotExist):
    Orden.objects.get(pk=self.orden.pk)

```

Ilustración 11. Pruebas unitarias al Gestionar pre despacho

```

# Crear un objeto ProductoVariacion para su uso en las pruebas
self.producto_variacion = ProductoVariacion.objects.create(
    producto=self.producto,
    date_created="2023-09-30 12:00:00",
    description="Variación 1",
    sku="VAR001",
    price=89.99,
    regular_price=119.99,
    status="Activo",
    sale_price=79.99,
)

def test_crear_producto_variacion(self):
# Comprobar que se puede crear una ProductoVariacion
self.assertEqual(self.producto_variacion.sku, "VAR001")

def test_actualizar_producto_variacion(self):
# Actualizar la descripción de ProductoVariacion y comprobar el
cambio
self.producto_variacion.description = "Variación Actualizada"
self.producto_variacion.save()
self.assertEqual(self.producto_variacion.description, "Variación
Actualizada")

```

```
        # Leer la ProductoVariacion desde la base de datos y comprobar si
los datos son correctos
        producto_variacion_leida =
ProductoVariacion.objects.get(pk=self.producto_variacion.pk)
        self.assertEqual(producto_variacion_leida.sku, "VAR001")

    def test_eliminar_producto_variacion(self):
        # Eliminar la ProductoVariacion y comprobar que ya no exista en
la base de datos
        self.producto_variacion.delete()
        with self.assertRaises(ProductoVariacion.DoesNotExist):
            ProductoVariacion.objects.get(pk=self.producto_variacion.pk)
```

Ilustración 12. Pruebas unitarias al Gestionar entrada y salida de productos al almacén