

**SOFTWARE DE GESTIÓN, INVENTARIO Y FACTURACIÓN, PARA EL
MICRO MERCADO PROVISIONES DYLAN DE LA CIUDAD DE
VALLEDUPAR, CESAR**

**SANTIAGO PUPO COGOLLO
JHON JAIDER MANOSALVA RANGEL**

**UNIVERSIDAD POPULAR DEL CESAR
FACULTAD DE INGENIERÍAS Y TECNOLÓGICAS
PROGRAMA INGENIERÍA DE SISTEMAS
LÍNEA DE INVESTIGACIÓN INGENIERÍA DEL SOFTWARE
SECCIONAL AGUACHICA, CESAR
2025**

**SOFTWARE DE GESTIÓN E INVENTARIOS Y FACTURACIÓN, PARA
EL MICRO MERCADO PROVISIONES DYLAN DE LA CIUDAD DE
VALLEDUPAR, CESAR**

**SANTIAGO PUPO COGOLLO
JHON JAIDER MANOSALVA RANGEL**

**PROPUESTA PROYECTO DE GRADO PARA OPTAR EL TÍTULO DE
INGENIERO DE SISTEMAS, MODALIDAD PRESENCIAL.**

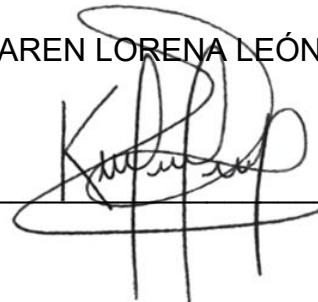
**DIRECTORA
KAREN LORENA LEÓN PÉREZ
CODIRECTOR
FRANCISCO CHINCHILLA TORRES**

**UNIVERSIDAD POPULAR DEL CESAR
FACULTAD DE INGENIERÍAS Y TECNOLÓGICAS
PROGRAMA INGENIERIA DE SISTEMAS
LÍNEA DE INVESTIGACIÓN INGENIERÍA DEL SOFTWARE
SECCIONAL AGUACHICA, CESAR
2025**

Nota de aceptación:

KAREN LORENA LEÓN PÉREZ

Director



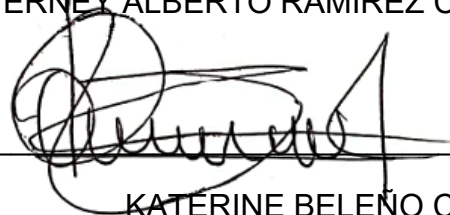
FRANCISCO CHINCHILLA TORRES

Codirector



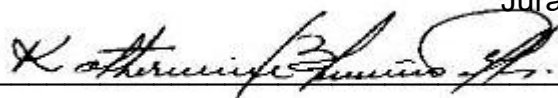
ERNEY ALBERTO RAMÍREZ CAMARGO

Jurado



KATERINE BELEÑO CASELLES

Jurado



Aguachica, Día _____ Mes _____ Año _____

AGRADECIMIENTOS

Deseamos transmitir nuestro más profundo agradecimiento, en especial a Dios por otorgarnos sabiduría y salud a lo largo de este proceso de formación, a todas las personas que hicieron viable este proyecto. Extendemos nuestro agradecimiento a nuestras familias, cuya ayuda incondicional y motivación constante nos han acompañado durante todo este trayecto académico.

Deseamos expresar nuestro agradecimiento a nuestra directora de proyecto, Karen Lorena León Pérez, y codirector Francisco Chinchilla Torres, por su dirección y dedicación en cada fase del desarrollo de esta investigación. Su experiencia y asesoría han sido claves para alcanzar este trabajo.

Igualmente, queremos expresar nuestro agradecimiento al micro mercado Provisiones Dylan por darnos la oportunidad de diseñar y poner en práctica nuestro sistema en su local, proveyéndonos del espacio y los datos requeridos para la instalación del software.

Para terminar, queremos dar un agradecimiento especial a nuestros colegas y amigos, quienes con su apoyo y orientación han sido fundamentales en este proceso.

CONTENIDO

TÍTULO.....	14
1. PROBLEMA	14
1.1. FORMULACIÓN DEL PROBLEMA	16
1.2. JUSTIFICACIÓN	16
1.3. OBJETIVOS	17
1.3.1. OBJETIVO GENERAL:	17
1.3.2. OBJETIVOS ESPECÍFICOS:.....	18
1.4. DELIMITACIONES	18
1.4.1. TEMPORAL	18
1.4.2. ESPACIAL	18
1.4.3. CONTEXTUAL.....	19
2. MARCO REFERENCIAL	21
2.1. MARCO HISTÓRICO	21
2.1.1. Nivel Internacional	21
2.1.2. Nivel Nacional.....	24
2.1.3. Nivel Regional.....	27
2.2. MARCO TEÓRICO.....	29
2.2.1. Software de Escritorio	29
2.2.2. Diccionario de datos	30
2.2.3. Clean Architecture.....	30
2.2.4. Principio SOLID	31
2.2.5. Patrones de diseño.....	32
2.2.6. Casos de uso.....	32
2.3. MARCO LEGAL	33
2.3.1. Propiedad Intelectual y Derechos de Autor	33
2.3.2. Comercio Electrónico y Transacciones Digitales.....	33
2.3.3. Protección de Datos Personales y Seguridad de la Información	34
2.3.4. Normativa, Contable y Fiscal para MiPymes.....	34
2.3.5. Metodología de Desarrollo de Software.....	40
2.4. MARCO CONCEPTUAL	34
Software:.....	34
Base de datos:.....	35
Micromercado:	35
Modelo ROP (Modelo de punto de pedido):	35
Gestión:.....	36
Control de inventario.....	36

Facturación:	36
Inventario:	36
FEFO (First Expires, First Out) primero en expirar, primero en salir:	37
• FIFO (First In, First Out): Primero en entrar, primero en salir:.....	37
Información:	37
Proceso:.....	38
Requerimientos funcionales:	38
Sistema:	38
C#:.....	39
.NET 8:.....	39
SQL Server:	39
3. DISEÑO METODOLÓGICO PRELIMINAR.....	40
3.1. DISEÑO METODOLÓGICO	41
3.2. TIPO Y ENFOQUE DE INVESTIGACIÓN	42
3.4. POBLACIÓN.....	43
3.5. MUESTRA	43
3.6. TÉCNICAS PARA LA RECOLECCIÓN DE DATOS	43
3.3.1. Entrevistas:	44
3.3.2. Observación directa:.....	44
4. ESQUEMA TEMÁTICO	45
4.1. OBJETIVO 1	45
4.1.1. FASE 1: PLANIFICACIÓN	45
4.1.1.2. Visitar el micromercado "Provisiones Dylan" para observar cómo se gestionan los procesos de inventario y facturaciones.	46
4.1.1.3. Realizar la entrevista con el representante legal del micromercado como técnica de recolección de necesidades y expectativas del cliente respecto al software.	46
4.1.1.4. Establecer el alcance del proyecto	47
4.1.1.5. Requisitos funcionales	47
4.1.1.6. Requerimientos No Funcionales	47
4.1.1.7. Elaborar las historias de usuario que describan las funcionalidades con las que va a interactuar el usuario.	47
4.1.1.8. Recursos necesarios para el proyecto.....	48
4.1.1.9. Recursos humanos.....	48
4.1.1.10. Recursos Materiales, Servicios y Financieros.	49
4.2. OBJETIVO 2	50
4.2.1. DISEÑO	50
4.2.1.2. Diseñar las vistas del aplicativo de escritorio para determinar los colores, fuentes y que el diseño sea agradable.....	52
4.2.1.3. Paleta de Colores y Estilos	54

4.2.1.4. Diseño de la Pantalla de Selección de Perfil	55
4.2.1.5. Pantalla de Inicio de Sesión con Teclado Numérico	56
4.2.1.6. Pantalla de Menú Principal.....	57
4.2.1.7. Pantalla de Gestión de Proveedores	58
4.2.1.8. Pantalla de Registro de Productos	59
4.2.1.9. Pantalla de Gestión de Inventario	60
4.2.1.10. Pantalla de Registro de Venta	61
4.2.1.11. Diseño de la base de datos.....	62
4.2.1.12. Diccionario de Datos	63
4.3. OBJETIVO 3.....	69
4.3.1. CODIFICACIÓN	69
4.3.1.2. Desarrollar el código de programación que cumpla con las historias de usuario.....	69
4.4. OBJETIVO 4.....	71
4.4.1. PRUEBAS Y LANZAMIENTO	71
4.4.1.1. Realizar las pruebas unitarias continuas para detectar y corregir errores durante el desarrollo de cada historia	71
4.4.1.2. Colocar en funcionamiento el software en el micromercado Provisiones Dylan para la prueba piloto.....	72
4.4.1.3. Ejecución de pruebas de aceptación: permitir que el cliente valide funcionalidades clave.	73
4.5. PRINCIPIO SOLID	74
5. CONCLUSIONES	75
6. RECOMENDACIONES.....	76
7. REFERENCIAS.....	77
8. ANEXOS	83
Anexo A: Entrevista a la representante legal del Micromercado provisiones Dylan, de Valledupar, Cesar, Mayra Alejandra chinchilla Orozco.....	83
Anexo B: Cronograma de actividades.....	86
Anexo C: Acta de aceptación de historias de usuario	87

LISTA DE TABLAS

Tabla 1. Población del micromercado.....	43
Tabla 2. Participantes del proyecto.....	49
Tabla 3. Recursos disponibles del proyecto.....	49
Tabla 4. Empresa	63
Tabla 5. Proveedores	63
Tabla 6. Productos.....	63
Tabla 7. Usuarios.....	64
Tabla 8. Clientes.....	65
Tabla 9. Ventas.....	66
Tabla 10. Detalle de ventas.....	66
Tabla 11. Pagos.....	66
Tabla 12. Compras	67
Tabla 13. Detalle de compra	68
Tabla 14. Caja.....	68

LISTA DE FIGURAS

Figura 1. Ubicación de la Ciudad de Valledupar en Colombia y el departamento del Cesar.....	19
Figura 2. Capas de Clean Architecture.....	31
Figura 4. Historias de Usuarios.....	48
Figura 5. Estructura de la arquitectura del proyecto	51
Figura 6. Diagrama del patrón de diseño MVC	52
Figura 7. Repositorio github del sistema	53
Figura 8. Manejo de errores Try Catch.....	54
Figura 9. Diseño de la Pantalla de Selección de Perfil	55
Figura 10. Pantalla de Inicio de Sesión con Teclado Numérico	56
Figura 11. Pantalla de menú Principal	57
Figura 12. Pantalla de gestión de proveedores.....	58
Figura 13. Pantalla de registro de productos.....	59
Figura 14. Pantalla de gestión de inventario	60
Figura 15. Pantalla de registro de venta.....	61
Figura 16. Diagrama Entidad Relación.....	62
Figura 17. Fragmento del código del método CrearProductoAsync.....	69
Figura 18. Fragmento del código del método RegistrarVentaAsync.....	70
Figura 19. Registro de Producto	71
Figura 20. Crear cliente	72
Figura 21. Evidencia del funcionamiento del software en el micromercado provisiones Dylan.	72
Figura 22. Evidencia de las pruebas de aceptación.....	73
Nota. Los autores	73
Figura 23. Cronograma de actividades.....	86

RESUMEN

La ineficiencia en la gestión del inventario y la facturación constituye un reto importante para los pequeños negocios, como el micro mercado provisiones Dylan, situado en Valledupar, Cesar. La carencia de un sistema automatizado provoca errores en la administración, pérdidas financieras y demoras en los procesos operativos. Para resolver esta situación, se sugiere la creación de un software que maneje la gestión, el inventario y la facturación, con el objetivo de optimizar estos procedimientos y ofrecer un mayor control y precisión en las operaciones del comercio.

El sistema se desarrollará con el uso de tecnologías como .NET 8, C#, y SQL Server, asegurando una solución sólida, escalable y efectiva. La metodología que se utilizará es Extreme Programming (XP), un enfoque ágil que favorece la retroalimentación continua y la mejora constante del software.

La investigación tiene un enfoque cualitativo con una metodología aplicada, la cual busca optimizar tecnológicamente los procesos comerciales. La obtención de datos se llevará a cabo a través de observación directa y entrevistas con los administradores del micro mercado, con el fin de identificar los requerimientos específicos y analizar el impacto de la solución implementada. Se anticipa que este software apoye la gestión del negocio, disminuyendo errores en la facturación, mejorando el control del inventario y favoreciendo la toma de decisiones estratégicas.

ABSTRACT

Inefficient inventory and billing management is a significant challenge for small businesses, such as the provisions Dylan micro market, located in Valledupar, Cesar. The lack of an automated system leads to administrative errors, financial losses, and delays in operational processes. To address this situation, the creation of software to manage management, inventory, and billing is suggested, with the goal of optimizing these procedures and offering greater control and precision in business operations.

The system will be developed using technologies such as .NET 8, C#, and SQL Server, ensuring a robust, scalable, and effective solution. The methodology used is Extreme Programming (XP), an agile approach that favors continuous feedback and ongoing software improvement.

The research has a qualitative approach with an applied methodology that seeks to technologically optimize business processes. Data will be collected through direct observation and interviews with micro market managers to identify specific requirements and analyze the impact of the implemented solution. This software is expected to support business management by reducing invoicing errors, improving inventory control, and facilitating strategic decision-making.

INTRODUCCIÓN

El propósito fundamental de este proyecto es elaborar un software para la gestión, inventario y facturación del micro mercado provisiones Dylan, localizado en Valledupar, Cesar. La adopción de esta herramienta tecnológica pretende optimizar el control de inventarios, acelerar el proceso de facturación y mejorar la administración del negocio, disminuyendo los errores operativos y aumentando la eficacia en la toma de decisiones.

Hoy en día, numerosos pequeños negocios siguen llevando a cabo sus operaciones de forma manual, lo que provoca inconsistencias en el control de inventarios, pérdidas financieras y una gestión ineficiente. De acuerdo con diversos estudios sobre la transformación digital en microempresas, la implementación de sistemas automatizados puede aumentar considerablemente la productividad y competitividad de los comercios al por menor, favoreciendo la optimización de recursos y la lealtad de los clientes.

Este proyecto no solo apoyará al micro mercado provisiones Dylan, sino que también podría establecerse como un modelo de referencia para otros pequeños negocios que necesiten actualizar sus procesos administrativos. A través de la implementación de esta solución tecnológica, se espera ayudar a la digitalización del sector minorista, mejorando la gestión empresarial y fomentando la competitividad en el mercado local.

TÍTULO

SOFTWARE DE GESTIÓN, INVENTARIO Y FACTURACIÓN, PARA EL MICRO MERCADO PROVISIONES DYLAN DE LA CIUDAD DE VALLEDUPAR, CESAR

1. PROBLEMA

En un mundo globalizado, la falta de un software adecuado puede tener repercusiones negativas para los negocios. Según Trejo (2021), en la sociedad moderna se ha desarrollado una fuerte dependencia de las tecnologías, ya que su integración permite un mejor rendimiento con menor esfuerzo. La tecnología facilita la realización de tareas en diversos campos y mejora los resultados.

La empresa colombiana (Loggro,2024), señala que entre las consecuencias de no contar con un sistema tecnológico está la pérdida de clientes y ventas por falta de stock. Cuando un cliente solicita un producto que no está disponible, se deteriora la imagen del negocio, lo que afecta la fidelización. Un software de inventarios permite conocer la cantidad de productos disponibles y establecer un stock óptimo para satisfacer la demanda.

Por otro lado, (Álvarez,2022), advierte que los comercios sin sistemas automatizados almacenan sus registros de forma manual, usando libretas para conservar facturas y controlar las compras. Este método retrasa los procesos, aumenta los errores humanos y puede provocar pérdidas económicas, dificultando el crecimiento empresarial.

Las pequeñas empresas, especialmente las dedicadas a la venta de alimentos y productos básicos, son particularmente vulnerables a una gestión ineficiente. La Cámara de Comercio de (Cali,2023), destaca que la transformación digital ha mejorado la productividad y reducido pérdidas en negocios que antes dependían de procesos manuales. A su vez, la (Cámara de Comercio de Casanare,2023), informa que el comercio al por mayor y al por menor representa el 39% de las actividades económicas registradas, lo que resalta la necesidad de modernización para mantener la competitividad.

En Valledupar, la mayoría de los micro mercados carecen de sistemas de gestión automatizados. A través de una entrevista al propietario del micro mercado Provisiones Dylan, se identificó la necesidad urgente de implementar un software que optimice el control de inventario y facturación, facilitando las ventas y mejorando el rendimiento administrativo.

Provisiones Dylan enfrenta serios problemas operativos por la ausencia de un sistema automatizado. Actualmente, el inventario se gestiona manualmente, lo que genera desabastecimientos, compras innecesarias y una planificación

deficiente. Esta situación retrasa la atención al cliente, ya que se debe verificar físicamente la disponibilidad de productos, afectando la satisfacción del consumidor. La facturación también se realiza a mano, lo que incrementa los tiempos de espera y afecta la fidelización.

Además, el cierre de caja y la conciliación del inventario son procesos manuales que consumen tiempo y son propensos a errores, dificultando el análisis financiero y la toma de decisiones. La falta de registros digitales impide hacer un seguimiento preciso de las ventas y el comportamiento del inventario, limitando la capacidad de anticiparse a las necesidades del negocio. Esto se traduce en pérdidas por errores en los cobros, ventas no concretadas y mala gestión del stock.

Provisiones Dylan, ubicado en las Torres Nando Marín, Manzana 15, Bloque R de (Valledupar, Cesar), se dedica a la venta de productos de la canasta familiar. Su gestión se basa en bitácoras escritas, lo que genera pérdidas de tiempo y errores. Las inconsistencias entre las anotaciones y el inventario real provocan duplicaciones o pérdidas de productos, afectando la rentabilidad y la experiencia del cliente.

El manejo manual del inventario y las ventas demanda largas jornadas en tareas administrativas como el registro de productos y la elaboración de facturas. Esto reduce el tiempo disponible para mejorar la atención al cliente o explorar oportunidades de negocio, además de aumentar el margen de error en los registros. Las discrepancias en el inventario generan pérdidas económicas y gastos adicionales para reponer productos. La facturación manual también puede provocar errores que afectan la confianza del cliente y la reputación del negocio. En un entorno competitivo, estas fallas reducen las posibilidades de crecimiento y sostenibilidad, evidenciando la necesidad de implementar herramientas tecnológicas.

Las anotaciones hechas a mano están expuestas a errores como olvidos o duplicaciones, generando desabastecimientos o acumulaciones innecesarias. En facturación, los errores perjudican la claridad financiera y la relación con los clientes. En contraste, un sistema automatizado permite registrar y actualizar en tiempo real los movimientos de inventario y las transacciones, asegurando información confiable para la toma de decisiones. Además, reduce el tiempo en tareas administrativas, permitiendo al personal enfocarse en actividades que generen valor, como la atención al cliente o la planificación estratégica.

La ausencia de herramientas tecnológicas limita la capacidad del micro mercado para tomar decisiones informadas, reduciendo su competitividad. El enfoque manual actual no permite identificar tendencias de compra ni ajustar la oferta de productos, lo que impide mejorar las ventas y alcanzar los objetivos del negocio.

1.1. FORMULACIÓN DEL PROBLEMA

¿A través del desarrollo de un software de gestión, inventarios y facturación se podrán resolver los problemas que se presentan en estos procesos del micro mercado provisiones Dylan en la Ciudad de Valledupar?

1.2. JUSTIFICACIÓN

En el contexto actual, la competencia en el mercado minorista es cada vez más intensa, lo que exige a los negocios pequeños adoptar herramientas tecnológicas que garanticen eficiencia y precisión en sus procesos operativos. Los micro mercados, como provisiones Dylan, enfrentan el desafío de optimizar sus operaciones para mantenerse competitivos y satisfacer las expectativas de sus clientes. Según un informe de la (Cámara de Comercio de Bogotá, 2023), la incorporación de tecnología en las pequeñas empresas aumenta significativamente su capacidad para reducir errores, mejorar la gestión de inventarios y agilizar los procesos de facturación, aspectos clave para asegurar su sostenibilidad en el mercado.

Los micro mercados, como provisiones Dylan, desempeñan un papel fundamental en la distribución de productos de consumo masivo y en satisfacer las necesidades de las comunidades locales. Sin embargo, muchos de estos establecimientos aún dependen de métodos manuales para la gestión de inventarios y facturación, lo que puede provocar errores, confusiones y pérdidas económicas significativas. En este sentido, la presente investigación cobra especial relevancia al abordar el desarrollo e implementación de un software de gestión, inventario y facturación específicamente diseñado para provisiones Dylan. Este sistema tiene el potencial de revolucionar las operaciones diarias del micro mercado, minimizando los errores asociados con el manejo manual de datos, brindando una experiencia de compra más fluida y satisfactoria para los clientes, y, en última instancia, impulsando la rentabilidad y competitividad del negocio en el mercado local.

Según (Rodríguez, 2022), el avance de la tecnología ha causado gran impacto a nivel mundial, debido a este gran desarrollo nos vemos en la obligación de evolucionar acorde a las necesidades presentadas, ya que este tipo de software permitiría al micro mercado provisiones Dylan llevar un registro detallado de sus productos y mantener un control adecuado de su inventario. Además, esta herramienta simplificará el proceso de registro y facturación de las ventas, lo que a su vez permitiría al micro mercado llevar un seguimiento más preciso de sus ingresos y gastos. Igualmente, (Lidefer, 2023) indica que es común que algunos micro mercados o tiendas utilicen sistemas manuales de inventario y facturación, lo que genera largas filas para pagar los productos y una facturación ineficiente. Estas razones dan a notar la importancia y la necesidad de un método de gestión

de inventario que ayude a las empresas a registrar diariamente, para que puedan lograr un control aproximado sin frustración.

El impacto de la innovación tecnológica en la gestión de inventarios es un tema ampliamente discutido en la literatura académica. Autores como (Ramírez, 2024), destacan la transformación profunda que la tecnología ha traído a las MiPymes, reformulando la gestión de inventarios y permitiendo una coordinación más eficiente del flujo y almacenamiento de productos. Por otro lado, (González, Barreto, Wah, & Lombardo, 2020), enfatizan la importancia de la implementación de las TICs en la gestión de inventario dentro de la cadena de suministro, lo que se alinea con la propuesta de integrar tecnologías avanzadas para mejorar la precisión y eficiencia. De esta manera, cada uno de estos autores nos proporciona una visión integral de cómo los sistemas de gestión de inventarios y facturación tienen gran magnitud y pueden incorporar innovación para mejorar la eficiencia y la competitividad en el mercado.

Un software de gestión, inventario y facturación automatizará y facilitará todos los procesos que se llevan a cabo en el micro mercado Dylan, logrará que sean más eficientes, seguros y proporcionará información oportuna y exacta, permitirá disminuir los tiempos de revisión de inventarios y lograr mantener la existencia de los productos en los niveles deseados para evitar desabastecimiento a través de la implementación del modelo ROP (Modelo de punto de pedido), ayudando así a alcanzar los objetivos y metas empresariales e involucrándolos en las nuevas tendencias tecnológicas.

Este software representa una opción práctica y puntual para solucionar el problema actual del micro mercado Dylan, generando un impacto a nivel de productividad y rentabilidad, lo que a su vez se refleja en el mejoramiento financiero.

1.3. OBJETIVOS

1.3.1. OBJETIVO GENERAL:

Desarrollar un software de gestión, inventario y facturación, para el micro mercado provisiones Dylan de la Ciudad de Valledupar, Cesar.

1.3.2. OBJETIVOS ESPECÍFICOS:

- Establecer los requisitos funcionales y no funcionales para diseñar los módulos de gestión, inventario y facturación para el micro mercado provisiones Dylan.
- Diseñar el software que integre las funcionalidades que son requeridas para la gestión, inventario y facturación del micro mercado.
- Construir el código del software aplicando prácticas de programación y refactorización continua para garantizar un diseño simple y eficiente.
- Realizar una prueba de aceptación de usuario que permita verificar la funcionalidad del software con los requisitos y funcionalidades establecidas en el diseño.

1.4. DELIMITACIONES

Con referencia a las delimitaciones del alcance del software de gestión y facturación de inventarios, se tiene que este proyecto no contempla: facturación electrónica, módulo de nómina y módulo contable.

1.4.1. TEMPORAL

Durante siete (7) meses a partir de la fecha de aprobación del anteproyecto se desarrollarán las actividades, las cuales permitirán lograr el objetivo de la presente investigación.

1.4.2. ESPACIAL

El micro mercado provisiones Dylan está ubicado en torres Nando Marín, Manzana 15 bloque R apartamento 104 en la Ciudad de Valledupar, Cesar.

1.4.3 CONTEXTUAL



Figura 1. Ubicación de la Ciudad de Valledupar en Colombia y el departamento del Cesar.
Nota: Google fotos.

El alcance de este proyecto se desarrolló en la ciudad de Valledupar, capital del departamento del Cesar, Colombia. Estuvo enfocado específicamente en optimizar los procesos internos del micro mercado Provisiones Dylan, así como en beneficiar directamente a sus clientes habituales y al entorno comercial inmediato. Este establecimiento se encuentra ubicado en la zona urbana de Valledupar, en la dirección registrada por el propietario, con coordenadas aproximadas: Latitud 10.474245, longitud -73.243631.

El micro mercado provisiones Dylan se encuentra ubicado en la Ciudad de Valledupar, Cesar y según la página web de la alcaldía: (Valledupar, 2024), Valledupar es la capital del departamento del Cesar, Colombia. Está ubicada al nororiente de la costa caribe colombiana, a orillas del río Guatapurí, en el valle del río Cesar, formado por la Sierra Nevada de Santa Marta y la serranía del Perijá. La ciudad es un importante centro para la producción agrícola, agroindustrial y ganadera en la región comprendida entre el norte del departamento del Cesar y el sur del departamento de La Guajira. También es uno de los principales epicentros musicales, culturales y folclóricos de Colombia por ser la cuna del vallenato, género musical de mayor popularidad en el país y actualmente símbolo de la música colombiana. Anualmente, atrae a miles de visitantes de Colombia y del exterior durante el Festival de la Leyenda Vallenata, máximo evento del vallenato.

(Valledupar, 2024) Se encuentra ubicada al norte del Valle del Cesar, entre la Sierra Nevada de Santa Marta y la Serranía del Perijá, al margen de los ríos Cesar y Guatapurí, en la Costa Caribe colombiana. Su territorio es llano y basculado hacia el sureste mediante una leve pendiente. La ciudad se encuentra a una altitud que oscila entre los 220 m al norte y 150 m a sur, siendo la altitud media de 168 m. Además de las enormes estructuras montañosas que la rodean (Pico Bolívar 5.775 m), sobresalen en inmediaciones de la ciudad dos cerros, al nororiente el "Cicolac" con 330 m.s.n.m. y el de "la Popa" con 310 m.s.n.m. (Valledupar, 2024), Tiene una Extensión total de 4.493 Km², en la extensión área urbana, el casco urbano tiene una longitud norte-sur de 8.3 km y este-oeste de 6.2 km, su altitud de la cabecera municipal (metros sobre el nivel del mar): la ciudad se encuentra a una altitud que oscila entre los 220 m. al norte y 150 m. a sur, siendo la altitud media de 168 m. La temperatura media Anual es de 28,4 °C, con máximas y mínimas de 22°C y 34°C respectivamente, la temperatura máxima histórica registrada es de 41.5°C y la mínima de 16°C. El mes más caluroso es abril, con un promedio de 30°C y el más fresco octubre. Su distancia de referencia es de 132 km, Santa Marta.

2. MARCO REFERENCIAL

2.1. MARCO HISTÓRICO

Un sistema de control orientado a inventarios, según señalan (Vivas Riaño, Rodríguez Alonso, & Barbosa Moreno, 2023), no es considerado una prioridad para las pequeñas empresas, donde a menudo no existe documentación, sistemas o responsabilidad real que respalde el proceso. Es crucial resaltar la importancia de un adecuado control de inventarios, ya que una gestión deficiente puede conducir a una toma de decisiones errónea e incluso promover actividades que impacten negativamente las finanzas de la empresa, como robos constantes, desperdicios, etc. Estas acciones afectan fuertemente el desempeño financiero de la empresa o pequeños negocios.

Estos proyectos en diferentes contextos, así como los resultados obtenidos, se realizó una investigación de proyectos similares a la solución propuesta, abarcando niveles internacional, nacional y local o regional, para comprender los procesos implementados en las soluciones referenciadas. Además, se busca ofrecer una perspectiva más amplia sobre cómo se abordaron al aplicarlos.

2.1.1. A nivel internacional

En Jipijapa – Manabí – Ecuador, (Delgado A. L., 2022), se presentó un trabajo titulado **“aplicación informática para la gestión de procesos de facturación e inventario para el hostel el café de jc de la ciudad jipijapa”**. En la Universidad Estatal del Sur de Manabí. El objetivo principal de esta investigación es crear un software que gestione los procesos de facturación y los inventarios del Hostel El Café de JC en Jipijapa. La aplicación de escritorio busca solucionar los problemas existentes en la gestión manual de estos procesos, que actualmente presentan deficiencias y falta de automatización. El hostel cuenta con 21 habitaciones individuales, dobles y triples, con capacidad para 60 personas.

La metodología empleada por (Delgado, 2022), fue la RUT, que es un enfoque de ingeniería de software destinado a asignar tareas y responsabilidades a los miembros del equipo de desarrollo. Esta metodología se divide en cuatro fases: Inicio, donde se establecen los objetivos y alcances del proyecto; Elaboración, donde se determina la arquitectura del sistema; Construcción, donde se finaliza la funcionalidad del sistema; y Transición, que es la etapa final en la que se realiza la depuración y se entrega al usuario. También se utilizó la investigación cualitativa para recopilar y analizar datos en el proceso de interpretación. Las herramientas tecnológicas empleadas fueron: PHP, MySQL, CSS y JavaScript.

El autor (Delgado, 2022), llegó a la conclusión de que, en la investigación, se reconocieron las necesidades del establecimiento, lo que permitió evidenciar la relevancia de conseguir un software que facilite la automatización de las tareas del negocio. Se logró implementar la aplicación de acuerdo con los requisitos establecidos, lo que permite gestionar automáticamente el proceso de facturación e inventario en el hostel de café de JC de Jipijapa. Las pruebas realizadas confirmaron que funciona correctamente y sin errores durante su ejecución.

En Milagro-Ecuador (Victoria, 2020), presentó una propuesta tecnológica titulada “implementación de un software de gestión de las operaciones internas y externas de la ferretería “Cárdenas” de la ciudad Milagro”. En la Universidad Agraria del Ecuador, el propósito principal de esta investigación fue: desarrollar un software de gestión utilizando un lenguaje y gestor de base de datos de código abierto que facilite el control de las operaciones internas y externas de la Ferretería 'Cárdenas' en Milagro.

La metodología empleada por (Victoria, 2022), fue la RAD (Desarrollo Rápido de aplicaciones). Esta metodología se usó para crear un software de gestión que se implementó directamente en la ferretería 'Cárdenas'. Este enfoque es característico del desarrollo en cascada, ya que se centra en la planificación y el diseño secuencial, según (Power Morse, 2016), el desarrollo del software de gestión y su posterior implementación en la ferretería permitió mejorar tanto el trabajo interno como externo, facilitando un progreso rápido e interactivo. Las herramientas tecnológicas utilizadas incluyeron Python, PyCharm, JavaScript y PostgreSQL.

La conclusión de la autora (Victoria, 2022), logró implementar el software de gestión en la ferretería “cárdenas”, lo que permitió tener un inventario de forma sistemática, facilitando las operaciones sin inconvenientes. Se realizaron pruebas utilizando casos de caja blanca y caja negra, que ayudaron a determinar si el software funcionaba adecuadamente en la ferretería, asegurando que no hubiera problemas en la lógica de programación, ya que es fundamental que todo software opere correctamente para evitar futuros inconvenientes. Además, consiguió implementar un visualizador que permite buscar informes filtrando por fecha, mes y año, y ofrece la opción de descarga en formatos Excel o PDF.

En el Alto-Bolivia (Cruz, 2020), se llevó a cabo un proyecto de grado titulado “**Sistema de control de inventarios y facturación**”, caso: “**empresa sharpoint**”, en la Universidad pública del Alto. El objetivo de esta investigación es desarrollar un sistema que gestione los inventarios y la facturación de Sharpoint, con el propósito de centralizar la información y la administración de inventarios, facilitando así las ventas. Además, se identificaron las actividades que debían ser automatizadas.

Según (Cruz, 2020), la metodología empleada para el desarrollo fue RUP junto con UML. Para evaluar la calidad del sistema se utilizó el modelo McCall y el modelo COCOMO II se aplicó para estimar los costos. RUP, o Proceso Unificado Racional, es un grupo de metodologías que se ajustan al contexto y necesidades de las organizaciones, desarrollado por Rational Software, que ahora es parte de IBM. Junto con RUP, el Lenguaje Unificado de Modelado (UML) es la metodología más común para el análisis, diseño, implementación y documentación de sistemas orientados a objetos. Las herramientas utilizadas fueron MariaDB, Java y NetBeans.

El autor (Cruz, 2020) concluyó que se desarrollaron módulos para gestionar ventas, clientes, productos, materiales, pedidos y reportes de facturación, lo cual redujo el tiempo necesario para buscar y acceder a información sobre productos, clientes y proveedores. Esto llevó a una mejora en la generación de ventas y facturas, así como a un registro detallado de las transferencias, lo que incrementó el control y disminuyó las pérdidas de productos. Se elaboraron informes periódicos que ayudaron en la toma de decisiones. Se resolvieron todos los problemas específicos y, de esta manera, se abordó la situación general, permitiendo así entregar a la empresa "sharpshoot" un sistema de control de inventarios y facturación que satisface sus necesidades.

En Tena, Ecuador (Alvarado, 2021) llevó a cabo un estudio titulado **"Implementar un software para el control de inventario en el negocio 'adelita' utilizando la metodología ágil xp"**, en el Instituto Superior Tecnológico Tena. El objetivo principal fue crear un software que ayude a controlar el inventario del negocio 'Adelita', ya que actualmente no cuentan con un sistema informático para organizar sus procesos contables debido a la falta de recursos o desconocimiento en el área, por lo que se decidió desarrollar este software.

La metodología utilizada por (Alvarado, 2021), fue la ágil, enfocada en el desarrollo de software conforme a principios, valores y prácticas esenciales. Se empleó XP, programado en PHP y utilizando MySQL como gestor de base de datos, además de JS y AJAX. Esto permite un mejor control del inventario, logrando resultados más eficientes y efectivos, y facilita que el usuario complete facturas y gestione el inventario en menos tiempo, evitando así la pérdida de información. La metodología XP se centra en buenas prácticas de desarrollo de software, fundamentada en cuatro valores clave: comunicación, simplicidad, retroalimentación y valentía. Las herramientas tecnológicas utilizadas incluyeron PHP, MySQL, XP, JS y Ajax.

El autor (Alvarado, 2021), concluyó en esta investigación que se logró crear una propuesta para un sistema automatizado que controla el inventario en las operaciones de la empresa 'Adelita', y que fue exitosa. Crearon una estructura simple de usar que se ajusta a las necesidades para construir el software, lo que permitió reducir el tiempo de creación de facturas y gestionar las cuentas,

aumentando la productividad del negocio. Durante las pruebas de funcionalidad, pudieron verificar que la información se ingresaba correctamente en la base de datos, lo que mostró que el sistema funcionaba bien. También destacaron que el modelo Ajax permitió al servidor hacer solicitudes en segundo plano usando JavaScript, proporcionando una respuesta inmediata al usuario.

2.1.2. A nivel nacional

En Bogotá, (Riaño, Alonso, & Sinvesol, 2023) expusieron un estudio titulado: **“aplicación web para el control y gestión de inventario de mercancías y herramientas de solincorp s.a.s. - sinvesol”**. En la fundación Universitaria Los Libertadores. El objetivo de la investigación es crear una aplicación web que facilite el control del inventario de mercancías y herramientas de la empresa Solincorp S.A.S., en su almacén de Bogotá, y que permita acceder a ella desde diferentes lugares del país.

La metodología que utilizaron (Riaño, Alonso, & Sinvesol, 2023), se aplicó a los tres modelos para identificar el tipo de investigación llevada a cabo. Se utilizaron tres enfoques de investigación: cualitativa, cuantitativa y mixta, cada uno con distintos métodos para recoger y analizar datos. La elección depende del problema que se evalúe, así como de las características de la población, la información y el tipo de estudio. Las herramientas tecnológicas utilizadas incluyeron: Sublime Text, JavaScript, Python, C++, Java, XAMPP, MySQL, PHP y Perl.

Los autores (Riaño, Alonso, & Sinvesol, 2023), concluyeron que era importante identificar las fallas en la gestión de inventarios en la bodega de almacenamiento de solincorp S.A.S. Al hacerlo, pudieron detectar debilidades que integraron en una matriz DOFA para diseñar una aplicación web. Esta matriz les permitió determinar que era necesario mejorar la toma de decisiones actual para facilitar el acceso a los productos en stock y así liberar espacio para la investigación y futuras oportunidades de expansión del mercado. Como resultado, crearon una herramienta que promueve el crecimiento de la empresa, aumentando la demanda de los clientes, la seguridad y la confianza en la innovación de sus productos y la eficiencia del servicio.

En Tunja (Sánchez & Galeano, 2021) llevaron a cabo un estudio llamado: **“desarrollo de un sistema de gestión de inventario para el control de productos en las fases de solicitud, envío, recepción y retorno”**, en la Universidad Santo Tomás. El objetivo de esta investigación es crear un sistema de control de inventario que gestione los productos en las etapas de solicitud, envío, recepción y retorno.

La metodología que utilizaron los autores (Sánchez & Galeano, 2021), fue Scrum, que asigna tareas y tiempos de desarrollo a cada equipo. Se tomaron en cuenta las etapas del ciclo de vida del software, que son: análisis de requerimientos; diseño y arquitectura; implementación; pruebas y validación; y documentación. Se definieron los siguientes roles: Product Owner (PO), quien actúa como gerente de la empresa y es responsable de la comunicación con el cliente, maneja los presupuestos, establece prioridades según las actividades y determina el valor del producto. Scrum Máster: quien lidera el equipo de desarrollo para asegurar el cumplimiento de la metodología y la entrega del producto. El equipo de desarrollo se encarga de crear el backend del producto, siguiendo las historias de usuario para cada sprint. Las herramientas tecnológicas que se utilizaron incluyen: Api GraphQL, el framework Django, Python, PostgreSQL y Power Designer.

Los autores (Sánchez & Galeano, 2021), concluyeron que lograron un desarrollo exitoso que cumplía con las historias de usuario, aplicando buenas prácticas de programación orientada a objetos y utilizando el framework Django. Destacan que la metodología SCRUM fue fundamental en su proceso, ya que les ayudó a priorizar tareas y mejorar la comunicación y retroalimentación entre las partes involucradas. También pudieron definir el alcance del sistema para comprender las necesidades de las historias de usuario, lo cual fue crucial para identificar y analizar tanto los requisitos funcionales como no funcionales del software, así como las limitaciones, facilitando así una clara comprensión de las necesidades del cliente a resolver.

En Bucaramanga, (Dávila & abril, 2020) presentaron una investigación titulada: “desarrollo de un prototipo de sistema de facturación e inventarios para tiendas minoristas de ropa que mediante redes neuronales mejore el control de inventarios”, en la universidad autónoma de Bucaramanga. El objetivo de este proyecto es crear un software de facturación POS que incluya una red neuronal artificial para predecir la demanda de productos, lo que ayuda a reducir la incertidumbre y disminuye el riesgo de tomar decisiones incorrectas en la gestión de inventarios. También se utilizará un modelo matemático EOQ que aplicará la demanda pronosticada para determinar la cantidad óptima de productos a solicitar, con el fin de minimizar los costos asociados a los pedidos y al mantenimiento de productos.

La metodología empleada por (Dávila & abril, 2020), fue Scrum, un modelo predictivo que utiliza una media móvil para aumentar la precisión de las previsiones y que tiende a distorsionar los datos de manera lineal, facilitando así el proceso de predicción. Esto se debió al tiempo limitado para el desarrollo y al alto control que este modelo exige. Al final, decidieron utilizar el modelo de regresión ARIMA, que también es un modelo predictivo con un promedio móvil que mejora la precisión de las predicciones y analiza los datos, haciendo más sencillo el proceso de predicción. Además, implementaron una red neuronal para estimar la demanda de productos basándose en datos históricos. Las

herramientas tecnológicas que utilizaron incluyeron machine learning, Visual Studio Code, SQL Server, Entity Framework, ASP.NET y la librería Brain.js.

Según la investigación de (Dávila & abril, 2020), los autores concluyeron que integrar una red neuronal de forma nativa en un software POS mejora la manera de introducir datos para el entrenamiento, ya que permite tener un formato estandarizado. Sin embargo, señalaron que las redes neuronales con arquitecturas pequeñas tienden a ser poco fiables al hacer predicciones en los escenarios estudiados. Durante las pruebas, emplearon diferentes métodos para normalizar datos grandes y pequeños con el objetivo de evaluar el entrenamiento de la red neuronal, lo cual demostró ser útil para gestionar inventarios. Además, concluyeron que las pruebas indicaron que las arquitecturas con nodos que disminuyen son más efectivas para este tipo de problemas, al igual que las que tienen nodos en un crecimiento lineal, lo que facilita el proceso de predicción.

En Bogotá (Pereira, 2019) llevó a cabo un estudio titulado: **“desarrollo de software web para la gestión de inventario, con aplicación móvil de consulta para la empresa cosdy sas”**, en la Universidad Distrital Francisco José de Caldas. En este proyecto se buscó crear un software que facilite la gestión de inventarios de las materias primas y productos terminados de la empresa Laboratorio de cosméticos cosdy.sas.

La metodología utilizada por (Pereira, 2019), fue la ágil SCRUM, ya que es un método que incorpora buenas prácticas para la colaboración en equipos, logrando así resultados positivos en el proyecto. Usar SCRUM para el desarrollo de software permitirá administrar el inventario de la empresa Laboratorio de cosméticos cosdy s.a.s., teniendo en cuenta que la empresa tiene poca experiencia en desarrollo de software, por lo que es más sencillo entregarles una versión funcional en cada sprint. Esto ayudará a que los responsables puedan proporcionar requisitos de forma más clara, asegurando una mejor calidad y mantenibilidad del software. También se usarán algunos diseños UML para apoyar la implementación, cumpliendo así con el principio del manifiesto ágil. Las herramientas tecnológicas que se utilizaron fueron Java EE, PostgreSQL, BPMN, Apache Tomcat, CSS, Bootstrap, Eclipse y Python Dash.

Según la conclusión del autor (Pereira, 2019), se logró crear un software beta para la gestión de inventarios, desarrollado en Java, utilizando JSP, y se emplearon otros lenguajes de apoyo como Python y JavaScript. Este software mejoró el proceso de inventario de materias primas y productos en la empresa Laboratorio de cosméticos cosdy s.a.s., que era el objetivo principal de la investigación. También decidieron establecer un proceso de inventarios porque la empresa carecía de uno bien definido y documentado. Por lo tanto, la definición del proceso de inventario fue aprobada por la dirección y los departamentos de calidad y producción.

2.1.3. A nivel regional

Respecto a la disponibilidad de software para la gestión de inventarios y facturación a nivel local, nuestras investigaciones han revelado que hay pocos proyectos sobre este tema, debido a que la demanda en esta área ha sido baja. A continuación, se describirán estos proyectos.

En la Universidad Abierta y a Distancia UNAD el proyecto de investigación realizado por Picón, (2019) se centró en el "Diseño de un Plan de Contingencia Informático basado en las normas ISO/IEC 22301 e ISO/IEC 27031 para la Ferretería Cesar S.A.S. en la ciudad de Valledupar", El objetivo principal de esta investigación fue garantizar la continuidad de los servicios de la Ferretería Cesar, ante la posible ocurrencia de un incidente de seguridad que pudiera afectar total o parcialmente la prestación de los servicios informáticos.

En la sección metodológica de la investigación se utilizó un enfoque cuantitativo. A lo largo del proyecto, se realizó una evaluación detallada de los riesgos y amenazas que podrían interferir con el funcionamiento habitual de los procesos informáticos de la empresa, lo que permitió identificar claramente los posibles escenarios de riesgo y medir cuantitativamente su impacto en las operaciones de la Ferretería Cesar. La metodología que se utilizó para el desarrollo del proyecto sigue las recomendaciones del Instituto Nacional de Estadística e Informática del Perú (INEI) y del Instituto Nacional de Ciberseguridad de España, y se ajusta a los estándares internacionales ISO/IEC 27031:2011 e ISO/IEC 22301:2012.

Finalmente, esta investigación llegó a la conclusión de que el diseño e implementación del plan de contingencia para el área informática en la Ferretería Cesar tuvo resultados favorables. La dirección pudo identificar los riesgos que enfrentaban y promover medidas preventivas para reducirlos. Además, se fomentó la concienciación y el compromiso de los empleados respecto a la seguridad de la información. Hoy en día, la Ferretería Cesar dispone de un protocolo bien estructurado para la recuperación rápida y ordenada de su tecnología, que respalda los procesos clave del negocio, mejorando así su capacidad de respuesta ante posibles problemas informáticos.

En la Universidad Popular del Cesar, sede Aguachica (UPCSA), se llevó a cabo un proyecto llamado "Desarrollo de una solución de Software que permita la gestión de pedidos, ventas y control de inventario en la empresa Distribuciones Oliva Sur S.A.S. de Aguachica, Cesar". Según Caicedo, (2018), este proyecto se clasificó como investigación aplicada y descriptiva, ya que estaba enfocado en resolver problemas concretos de la empresa y en describir las diversas situaciones que se presentaban. Para llevar a cabo el proyecto, se utilizaron tecnologías como bases de datos SQL y framework de JavaScript, incluyendo Node.js y Electron.js. Por esta razón, se adoptó la metodología de programación

extrema (XP), que permite un ahorro significativo de tiempo en cada fase del proceso.

En la conclusión del proyecto de grado sobre la creación de un software para Distribuciones Oliva Sur S.A.S., se resalta que se alcanzaron todos los objetivos y requisitos solicitados por la empresa, además el producto demostró ser confiable gracias a las pruebas realizadas. Se utilizaron herramientas de desarrollo junto con protocolos de seguridad e integridad de la información, lo que condujo a un software de escritorio efectivo y fácil de usar para los usuarios, a pesar de la complejidad que tuvo su desarrollo.

Otro proyecto desarrollado en la UPCSA fue el “Diseño de un aplicativo web que permita gestionar y controlar el sistema de facturación, clientes y reservas que se llevan a cabo en el hotel Monterrey Plus de la Ciudad de Aguachica (Cesar)”. En este proyecto, los autores, Solano y Jiménez (2019), proponen aumentar la eficiencia y la gestión del Hotel Monterrey Plus a través de la creación e implementación de una aplicación web.

Para llevar a cabo el proyecto, se eligió la metodología IWEB (Ingeniería Web), que es conocida por su enfoque en la creación, implementación y mantenimiento de aplicaciones y sistemas web. Esta metodología se basa en principios científicos, de ingeniería y gestión, asegurando un desarrollo ordenado y disciplinado para lograr la calidad en los sistemas y aplicaciones web. En cuanto a las tecnologías empleadas, se utilizó el gestor de base de datos MySQL y se programó el sistema con HTML, PHP, JavaScript, CSS, jQuery y AJAX. También era necesario tener acceso a internet para facilitar la transmisión de información.

En resumen, el proyecto logró establecer un proceso organizado y controlado para llevar a cabo todas las actividades necesarias para cumplir con los objetivos individuales y generales del desarrollo de la aplicación web. La arquitectura cliente-servidor, basada en la tecnología PHP, facilitó la centralización de la información y mejoró los procesos tanto para los usuarios como para el servidor, lo que resultó en una experiencia de navegación mucho mejor y una respuesta más rápida a las solicitudes de los usuarios. La implementación de la aplicación en el Hotel Monterrey Plus de Aguachica aportó beneficios importantes, como la reducción del tiempo en los procesos, mayor orden y control en la realización de tareas, eficiencia en la búsqueda de información, seguridad y protección de los datos, la capacidad de manejar registros ilimitados y la posibilidad de añadir nuevas funcionalidades en el futuro.

En la UPCSA se llevó a cabo un proyecto titulado "Implementación de un sistema de información en entorno web para la gestión de inventario, facturación y procesos de proveedores de la empresa Dentales del Oriente en Aguachica, Cesar". En este trabajo, los autores Manzano y Duarte (2018) tienen como objetivo principal crear una solución web que satisfaga las necesidades

específicas de facturación, gestión de proveedores e inventario, así como desarrollar e implementar el sistema de información.

2.2. MARCO TEÓRICO

2.2.1. Software de escritorio

Según lo define (TecnoEmpresa, 2025), una aplicación de escritorio es un software que requiere del sistema operativo de la computadora para operar. Se instala en el sistema mediante un instalador específico y emplea recursos del ordenador para su funcionamiento. La característica más destacada de estos programas es la capacidad de funcionar sin conexión a Internet. Las aplicaciones actuales continúan permitiendo el uso con conexión, pero esta se vuelve indispensable para manejar diferentes dispositivos, integrar a otros usuarios en el proyecto o actualizar el software sin requerir una reinstalación manual. Numerosas compañías están atentas a la seguridad, por lo que prefieren que las aplicaciones no se conecten a Internet. En este escenario, el desarrollo y las pruebas de escritorio pueden realizarse de manera llave en mano, y la conectividad resulta viable en una red de área local restringida.

De esta misma manera (Junquera, 2021), nos expresa que es un tipo de aplicaciones que se instalan directamente en la computadora, dependiendo del sistema operativo que tenga, se ejecutan de manera local en lo que se conoce como el lado del cliente, aunque también pueden conectarse de forma remota a un servidor. Por esta razón, se les llaman aplicaciones cliente-servidor.

2.2.2. Diccionario de datos

Según (sunrise, 2025), un diccionario de datos es un repositorio centralizado de información relacionada con los datos de una organización. Incluye metadatos que abarcan la definición, el nombre y los atributos de los elementos de datos dentro de una base de datos o pipeline de datos. Los diccionarios de datos son útiles para evitar errores y desacuerdos, ya que ofrecen un sitio confiable para toda la información sobre datos. Esto ayuda a prevenir la confusión y los errores que pueden surgir cuando las personas utilizan distintas formas de referirse a los datos.

2.2.3. Clean Architecture

Según lo define (domainlogi, 2024), Clean Architecture es un término que alude a un método de diseño de software sugerido por el autor y consultor Robert C. Martin; esto se detalla en su obra "Clean Architecture: A Craftsman 's Guide to Software Structure and Design" ("Arquitectura Limpia: Guía del artesano para la estructura y diseño de software").

El principio esencial de Clean Architecture consiste en que la estructura y el diseño del software deben centrarse en la separación entre las preocupaciones del negocio y las preocupaciones técnicas. La Arquitectura Limpia, dicho de otra manera, tiene como objetivo desarrollar un software de excelente calidad, que sea fácil de modificar y que, por otra parte, se ajuste sin complicaciones a las cambiantes demandas del negocio.

La Arquitectura Limpia se fundamenta en los principios del diseño SOLID y emplea patrones de diseño para conseguir una clara separación de responsabilidades y una estructura modular. Esto se consigue a través de la conformación de capas de abstracción, en las que las capas internas albergan la lógica de negocio, mientras que las capas externas comprenden la infraestructura técnica.

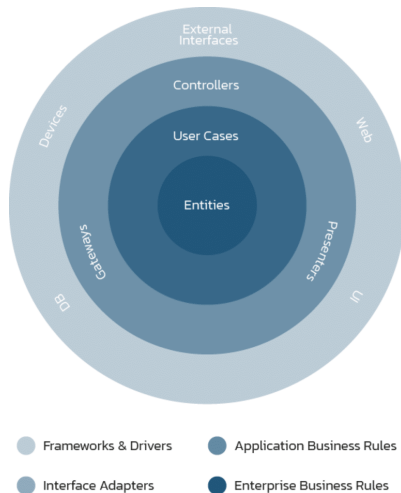


Figura 2. Capas de Clean Architecture
 Nota. Tomada de domainlogi

2.2.4. Principio SOLID

El principio solid según lo define (TechEdu, 2024), los principios SOLID constituyen un grupo de cinco pautas que se pueden aplicar en el diseño de software para que sea más fácil de entender, mantener y ampliar. El acrónimo representa El Principio de responsabilidad única (**S**ingle Responsibility Principle), Principio Abierto-Cerrado (**O**pen-Closed Principle), Principio de sustitución de Liskov (**L**iskov Substitution Principle), Principio de segregación de interfaz (**I**nterface Segregation Principle) y Principio de inversión de dependencia (**D**ependency Inversion Principle).

- **El Principio de responsabilidad única (Single Responsibility Principle):** Una clase debería tener solamente una responsabilidad, es decir, una causa para modificarse. Esto implica que una clase debe contar con un único objetivo, y todos sus métodos y atributos deben estar vinculados a ese objetivo.
- **Principio Abierto-Cerrado (Open-Closed Principle):** Una clase debe ser expansible pero inalterable. Esto implica que se puede incorporar una nueva funcionalidad a una clase sin necesidad de modificar el código ya existente.
- **Principio de sustitución de Liskov (Liskov Substitution Principle):** Los objetos dentro de un programa tienen que ser reemplazables por instancias de sus subtipos sin que se afecte la corrección de este. Esto implica que las subclasses deben estar diseñadas de forma que puedan ser empleadas indistintamente con sus superclases.
- **Principio de segregación de interfaz (Interface Segregation Principle):** Una clase no debería verse obligada a implementar interfaces que no requiere. Es decir, una clase debe implementar

únicamente las interfaces que realmente necesita, sin cargarlas con aquellas que no le son útiles.

- **El Principio de inversión de dependencia (Dependency Inversion Principle):** Las relaciones de dependencia entre los módulos tienen que ser invertidas. Esto implica que los módulos deben depender de abstracciones y no de implementaciones específicas.

2.2.5. Patrones de diseño

Un patrón de diseño, según lo define (Refactoring_Guru, 2024), son respuestas comunes a inconvenientes que se presentan a menudo en el desarrollo de software. Funcionan como esquemas preelaborados que se pueden adaptar para abordar un problema de diseño que aparece de manera frecuente en tu código.

No es posible seleccionar un patrón y replicarlo en el programa de la misma manera que se haría con funciones o bibliotecas predefinidas. El patrón no representa un fragmento concreto de código, sino que es una idea general para abordar un problema específico. Puedes aplicar los detalles del patrón e idear una solución que se ajuste a las particularidades de tu propio programa.

Frecuentemente, los patrones son confundidos con algoritmos, ya que ambos términos reflejan respuestas comunes a problemas reconocidos. Un algoritmo establece un conjunto específico de pasos para alcanzar un objetivo, mientras que un patrón proporciona una descripción más general de una solución. El código que utiliza el mismo patrón en dos programas diferentes puede variar.

Una forma de entender un algoritmo es compararlo con una receta de cocina: ambos tienen instrucciones precisas para lograr un objetivo. En cambio, un patrón se asemeja más a un plano, puesto que puedes ver el resultado y las funciones que tiene, pero el orden específico en el que lo implementas depende de ti.

2.2.6. Casos de uso

Un caso de uso lo define (DNP, 2020). Un caso de uso es una serie de interacciones que ocurren entre un sistema y sus actores como respuesta a un evento que inicia un actor principal en el sistema. Los diagramas de casos de uso se utilizan para detallar la comunicación y el comportamiento de un sistema a través de su interacción con los usuarios y/o con otros sistemas.

De esta misma manera (Historiadelaempresa, 2024), lo define como los casos de uso son una representación textual de la forma en que una persona se relaciona con un sistema con el fin de alcanzar un objetivo específico. Generalmente, los casos de uso se componen de una serie de pasos que detallan este procedimiento desde la perspectiva del usuario. Existen diferentes motivos para crear casos de uso, tales como verificar las funcionalidades del software o redactar guías de uso para los clientes.

Los expertos en tecnología suelen redactar casos de uso que ilustran las distintas maneras en las que una persona puede interactuar con el software. Por ejemplo, un gerente de proyecto podría desarrollar dos casos de uso para un nuevo software de cámara. El primero describe cómo una persona toma una fotografía y utiliza el nuevo software de cámara, mientras que el segundo expone un procedimiento para generar secuencias de vídeo con dicho software. Algunos casos de uso pueden incluir una descripción de los requisitos del sistema que resuma las principales características del software.

2.3. MARCO LEGAL

2.3.1. Propiedad Intelectual y Derechos de Autor

El desarrollo y uso del software de gestión e inventarios para el micro mercado Provisiones Dylan está protegido bajo la legislación colombiana de propiedad intelectual. (La Ley 23 de 1982), modificada por la (Ley 1915 de 2018), establece el régimen de derechos de autor aplicable al software. Según estas normas, el código fuente y el programa ejecutable se consideran obras protegidas, otorgando a sus creadores derechos morales y patrimoniales (Congreso de Colombia, 1982; 2018). Adicionalmente, la Decisión 486 de 2000 de la Comunidad Andina proporciona un marco regional para la protección de la propiedad industrial, que puede ser relevante si el software incorpora elementos patentables o marcas registradas (Comunidad Andina, 2000). Es crucial que el desarrollo del software respete estos derechos y, si se utilizan bibliotecas o componentes de terceros, se asegure el cumplimiento de sus respectivas licencias.

2.3.2. Comercio Electrónico y Transacciones Digitales

Aunque el enfoque principal no es la facturación electrónica, es importante considerar las implicaciones legales de las transacciones digitales y el comercio electrónico que el software pueda facilitar. (La Ley 527 de 1999), regula el comercio electrónico en Colombia, definiendo y normalizando el acceso y uso de los mensajes de datos, las firmas digitales y las entidades de certificación (Congreso de Colombia, 1999). Esta ley proporciona un marco legal para la validez de los registros electrónicos y las transacciones realizadas a través de

medios digitales. El software de gestión e inventarios debe asegurar que cualquier registro de transacción o documento generado cumpla con los requisitos de esta ley para garantizar su validez legal.

2.3.3. Protección de Datos Personales y Seguridad de la Información

La gestión de datos de clientes, proveedores y transacciones en el software debe cumplir con la Ley (1581 de 2012) de Protección de Datos Personales y su decreto reglamentario 1377 de 2013 (Congreso de Colombia, 2012), Esta normativa establece los principios para el tratamiento de datos personales, los derechos de los titulares de la información, y las obligaciones de los responsables y encargados del tratamiento de datos.

2.3.4. Normativa, Contable y Fiscal para MiPymes

El software de gestión e inventarios debe alinearse con la normativa contable y fiscal aplicable a las micro, pequeñas y medianas empresas (MiPymes) en Colombia. (La Ley 1314 de 2009) regula los principios y normas de contabilidad e información financiera y de aseguramiento de información aceptados en Colombia. Él (Decreto 2420 de 2015), conocido como el Decreto Único Reglamentario de las Normas de Contabilidad, de Información Financiera y de Aseguramiento de la Información, establece el marco técnico normativo para los preparadores de información financiera que conforman el Grupo 2, que incluye a las Pymes. Adicionalmente, la (Ley 590 de 2000), modificada por la (Ley 905 de 2004), promueve el desarrollo de las micro, pequeñas y medianas empresas, estableciendo un marco regulatorio específico para este sector.

2.4. MARCO CONCEPTUAL

El marco conceptual del presente proyecto se fundamenta en la definición clara de diversos conceptos clave, lo cual resulta fundamental para comprender la aplicabilidad y el alcance de este.

Software: El software es un grupo de programas utilizados para operar celulares, computadoras, tabletas, entre otros dispositivos, y ejecutar tareas específicas en cada una de ellas. Es completamente opuesto al hardware, que hace referencia a los componentes físicos de los dispositivos, pero se complementan bastante bien. Los programas de software sirven para diversas tareas, desde el almacenamiento, consulta, envío y organización de información, creación de otros programas o aplicaciones, hasta simple entretenimiento (Beltran, 2023).

Base de datos: Una base de datos es una recopilación organizada de información o datos estructurados que se almacenan electrónicamente en un sistema informático. Normalmente, una base de datos está controlada por un sistema de gestión de bases de datos (DBMS). En conjunto, los datos y el DBMS, junto con las aplicaciones asociadas, forman un sistema de bases de datos, comúnmente denominado simplemente base de datos. Los tipos más comunes de bases de datos utilizan estructuras de filas y columnas en tablas para mejorar la eficiencia del procesamiento y la consulta de datos. Esto permite acceder, gestionar, modificar, actualizar, controlar y organizar fácilmente la información. La mayoría de las bases de datos emplean un lenguaje de consulta estructurada (SQL) para escribir y consultar datos (Oracle, 2024).

Micro mercado: Definido por el autor (Pérez,2022), como tiendas locales o pequeñas bodegas comercializadoras de productos primarios, que cubren las necesidades básicas familiares.

Según (DANE, 2023), la define como una unidad económica, es aquella que cuenta con un máximo de 9 empleados y se dedica a producir bienes o servicios con el fin de generar ingresos, ya sea como propietario o arrendatario de los recursos necesarios para la producción.

Modelo ROP (Modelo de punto de pedido): Es un parámetro que indica el instante preciso en que una empresa necesita hacer un nuevo pedido para no quedarse sin existencias. Este se fundamenta en elementos como la demanda diaria promedio, el tiempo de entrega de los proveedores y un inventario de seguridad pensado para afrontar eventualidades. Este término se emplea principalmente en la organización y administración de inventarios para garantizar la continuidad de las operaciones y cumplir con las demandas del mercado, evitando así el exceso de stock y costos innecesarios (Imperia, 2025).

Fórmula del Punto de Reorden (ROP): La fórmula para el modelo ROP es la siguiente:

ROP = (Demanda diaria promedio x Lead Time) + Stock de seguridad.

Esta ecuación incorpora los elementos clave para identificar el momento preciso en el que se debe hacer un pedido.

Ejemplo: el micro mercado provisiones Dylan tiene una demanda diaria promedio de 70 unidades, un lead time de 8 días y un stock de seguridad de 200 unidades.

Aplicando la fórmula:

$$\text{ROP} = (70 \times 8) + 200$$

$$\text{ROP} = 560 + 200 = 760 \text{ unidades}$$

Esto indica que cuando el inventario llegue a las 760 unidades, será necesario hacer un nuevo pedido para no quedarse sin existencias.

Gestión: La gestión es un conjunto de procedimientos y acciones que se llevan a cabo para lograr un determinado objetivo. El objetivo de la gestión es alcanzar un objetivo optimizando al máximo posible los recursos disponibles, (Westreicher, 2024).

Control de inventario: Los diferentes aspectos de la responsabilidad en relación con los inventarios impactan en muchos departamentos, y cada uno de estos ejerce un cierto nivel de control sobre los productos mientras se van moviendo a través de los diversos procesos de inventario. Todos estos controles, que van desde el procedimiento para elaborar presupuestos y pronósticos de ventas hasta la operación de un sistema de costos en el departamento de contabilidad para la determinación de los costos de los inventarios, conforman el sistema de control interno de los inventarios. Las funciones generales son: planificación, compra, recepción, almacenamiento, embarques y contabilidad. (Peña, Machado, & Dávila, 2020)

Facturación: La facturación se refiere al conjunto de facturas que engloba todas las ventas de bienes o servicios de una empresa. Las facturas son documentos que reflejan el valor y los datos de una transacción comercial. La facturación ayuda a conocer el nivel de liquidez del negocio y a identificar los productos o servicios estrella según su volumen de ventas (Dobaño, 2024).

Inventario: Según el autor (Palacio, 1988) “Se denominan existencias o inventarios a la variedad de materiales que se utilizan en la empresa y se guardan en sus almacenes a la espera de ser utilizados, vendidos o consumidos, permitiendo a los usuarios desarrollar su trabajo sin que se vean afectados por la falta de continuidad en la fabricación o por la demora en la entrega por parte del proveedor.” Como lo define también el autor (Gasbarrino, 2023) Un inventario es un documento donde se registran todos los bienes tangibles y en existencia de una empresa, que pueden utilizarse para distintos fines. El inventario permite a una empresa conocer la fluctuación de artículos, lo cual influye en las decisiones sobre proveedores y la cantidad de existencias necesarias para cumplir con la demanda de los clientes. Además, garantiza que no haya mermas, robos o excedentes de productos, evitando así gastos innecesarios de almacenaje.

FEFO (First Expires, First Out) primero en expirar, primero en salir: El método FEFO implica estructurar la operación del almacén de tal manera que se priorice la venta de los productos que tengan una fecha de caducidad más inminente.

Un método FEFO que se aplique correctamente previene gastos adicionales por inventario caducado, permite mover los productos considerando su obsolescencia y minimiza el riesgo de que la mercancía se deteriore, un riesgo que se incrementa a medida que transcurre más tiempo en el almacén. (SCM & Arilla, 2021)

- **FIFO (First In, First Out): Primero en entrar, primero en salir:** Como lo define (CFOremoto & Philippe, 2021), conocido también como método PEPS (primero en entrar, primero en salir), es un sistema que te facilita la valoración del inventario no solo al final del ejercicio fiscal, sino que también te ayuda a establecer a qué precio deberías vender las mercancías durante el año.

El método FIFO implica que la venta de los productos se efectúe en el mismo orden en que fueron adquiridos por la empresa. De este modo, los lotes más antiguos en el inventario son los que primero se ofrecen al punto de venta.

- **LIFO (Last in, First Out): Último en entrar, primero en salir.** Según lo define (Ribeter, 2024), es un término empleado en la contabilidad y la administración de inventarios. Se fundamenta en la noción de que los últimos productos que entran a un inventario son los primeros en ser comercializados o utilizados. Este método se usa de manera extensa en varios sectores, incluyendo el comercio al por menor, la industria de fabricación y la contabilidad financiera.

En pocas palabras, el método LIFO plantea que los artículos más recientemente comprados son los primeros en ser vendidos. Esto implica que el coste de los productos vendidos se determina con el precio de compra más reciente, mientras que los artículos que quedan en el inventario se estiman a los precios de compra anteriores.

El método LIFO se emplea principalmente en sectores donde los precios de los productos suelen ascender con el tiempo. Esto ocurre porque al adoptar el costo más reciente para determinar el costo de los productos vendidos, el método LIFO posibilita a las empresas presentar un costo de venta más bajo y, por ende, un margen de beneficio bruto más elevado en sus estados financieros.

Información: es un conjunto organizado de datos que son relevantes para uno o más sujetos, permitiendo extraer conocimiento de ellos. En otras palabras, es

una serie de conocimientos comunicados, compartidos o transmitidos que forman algún tipo de mensaje. La información se puede clasificar de diferentes maneras, dependiendo de varios criterios. Uno de los criterios más comunes tiene que ver con la relación entre los emisores de la información y sus posibles receptores. Se puede clasificar de la siguiente manera:

- **Información confidencial o clasificada:** solo accesible a un pequeño grupo de personas debido a la naturaleza secreta, peligrosa, delicada o privada de los datos.
- **Información pública:** disponible para cualquier persona sin necesidad de permisos especiales y sin ninguna restricción de privacidad.
- **Información personal:** pertenece a una persona en particular y puede decidir con quién compartirla o a quién ofrecérsela.
- **Información externa:** proviene de un organismo, institución o empresa y está destinada a personas o entidades externas.
- **Información interna:** generada por un organismo, institución o empresa, destinada a ser utilizada solo dentro de la organización (Equipo editoria, 2020).

Proceso: Un proceso se define como un conjunto secuencial de acciones ejecutadas para alcanzar un objetivo específico. En el ámbito microeconómico, un proceso implica una secuencia lógica y planificada de etapas que se cumplen con el fin de lograr ciertos propósitos. Por lo tanto, la estructuración de procesos es fundamental para que las organizaciones puedan realizar sus actividades y obtener resultados óptimos (DocuSign, 2022).

Requerimientos funcionales: Los requerimientos funcionales de software son esenciales en el proceso de desarrollo, ya que proporcionan una descripción detallada de las funcionalidades que el programa debe cumplir para satisfacer las necesidades de los usuarios. Estos requisitos actúan como guías para los desarrolladores, delineando claramente qué acciones debe ser capaz de realizar el software. Al establecer claramente las expectativas de funcionalidad, los requerimientos funcionales ayudan a garantizar que el producto final cumpla con las necesidades y expectativas de los usuarios finales (Northware, 2022).

Sistema: Un sistema se define básicamente como un conjunto de partes que están conectadas y trabajan juntas como un todo. Aunque cada parte puede funcionar sola, en conjunto forman una unidad organizada y funcional. Además, un sistema puede ser parte de otro más grande o incluir sistemas más pequeños dentro de él. Por ejemplo, uno muy común es el sistema de información, que se compone de diferentes elementos organizados para manejar y procesar datos de manera eficiente para cumplir con un objetivo específico en un área determinada (Graus, 2023).

C#: Es un lenguaje de programación creado por Microsoft para la plataforma .NET. Es orientado a objetos, con un tipado fuerte y soporta múltiples paradigmas de programación, lo que permite desarrollar aplicaciones de escritorio, web, móviles y videojuegos. Se diferencia por su seguridad, la gestión automática de la memoria, la programación asincrónica y su compatibilidad con .NET. Su sintaxis clara y ordenada facilita la elaboración de software escalable, robusto y eficiente en diversos entornos (Cabanés, 2022).

.NET 8: ES una actualización más reciente del marco unificado y multiplataforma de Microsoft, que fue lanzado en noviembre de 2023. Esta herramienta permite el desarrollo de aplicaciones para la web, el escritorio, dispositivos móviles, la nube, microservicios e inteligencia artificial. Sobresale por su alto rendimiento y ofrece mejoras en Native AOT, Dynamic PGO y Blazor Full Stack. Es compatible con sistemas operativos como Windows, Linux y macOS, ofreciendo mayor seguridad y eficacia, además de optimizar el desarrollo en la nube con una mejor integración para contenedores y microservicios (Varela, 2023).

SQL Server: Es un sistema de manejo de bases de datos relacional (RDBMS) creado por Microsoft. Brinda alta seguridad, escalabilidad y rendimiento, lo que lo hace perfecto para aplicaciones corporativas. Admite transacciones ACID, consultas SQL sofisticadas y almacenamiento estructurado. Es compatible tanto con Windows como con Linux, lo que permite una gestión eficiente de datos en entornos locales, en la nube y con grandes volúmenes de información (Digitalgrow, 2023).

3. DISEÑO METODOLÓGICO PRELIMINAR

En este capítulo se detalla la metodología que se empleó para llevar a cabo el presente proyecto de investigación. En la sección 3.1, se presenta la metodología de desarrollo de software, donde se explica la metodología aplicada en el proyecto, en el 3.2 el diseño metodológico, especificando los procedimientos utilizados para organizar la investigación de manera eficiente. La sección 3.3 describe el tipo y enfoque de investigación aplicados, aclarando su relevancia en el estudio. En la sección 3.4 aborda la población objeto del estudio, identificando sus características más significativas. En la sección 3.5, se especifica la muestra seleccionada, detallando los criterios de selección y su representatividad en la investigación. En la sección 3.6, se explican las técnicas de recolección de datos implementadas, que incluyen entrevistas (3.6.1) y finalmente la observación directa en la sección 3.6.2, lo cual facilitará la recopilación de información importante para el análisis. A partir de estos componentes metodológicos, se estructurará el análisis de los datos recogidos, lo cual permitirá interpretar los resultados en función de los objetivos planteados en el proyecto.

3.1. METODOLOGÍA DE DESARROLLO DE SOFTWARE

La metodología de desarrollo de software comprende un conjunto de técnicas y métodos empleados en la creación de soluciones informáticas. La adopción de una metodología garantiza una organización efectiva, definiendo la disposición y utilización de los recursos, lo que facilita un control más riguroso del proceso de trabajo. Es relevante destacar que existen diversas metodologías entre las cuales elegir, dependiendo de las necesidades y preferencias de cada equipo de trabajo. (Universitat Carlemany, 2024).

Para la realización del proyecto, se optó por emplear la metodología XP Programación Extrema, la cual surgió a finales de los años 90 y se clasifica como una metodología ágil de desarrollo. Esta metodología se destaca por la creación de sistemas de alta calidad; además, mantiene al mismo tiempo la flexibilidad inherente a las metodologías ágiles. Su filosofía se centra en la simplicidad, priorizando lo esencial, fomenta una comunicación fluida entre los miembros del equipo y promueve la constante revisión y mejora del código (Raeburn, 2025).

Los valores principales de este modelo de desarrollo son:

Comunicación: Fomenta el diálogo directo entre desarrolladores y clientes para entender las necesidades del sistema.

Simplicidad: Prioriza lo esencial, reduciendo costos y manteniendo el diseño simple.

Feedback: Valora la retroalimentación rápida y constante para hacer ajustes precisos.

Coraje: Promueve la apertura al cambio, acepta errores y confiar en el proceso.

Respeto: Impulsa la colaboración y el buen trato entre los miembros del equipo. (Sydle, 2023)

Las fases que componen esta metodología son cinco:

Fases de la metodología de desarrollo XP (Programación extrema)

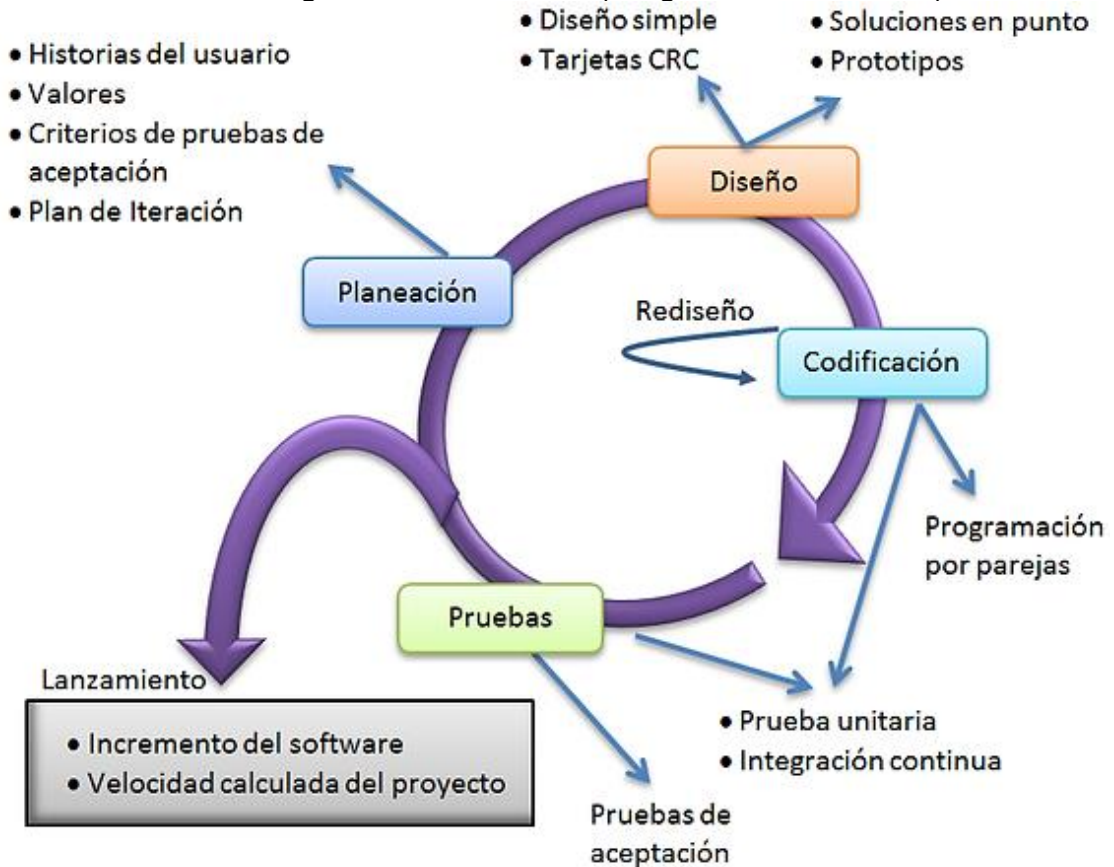


Figura 3. Fases de la metodología de desarrollo XP (Programación extrema)
Nota. Universitat Carlemany

Nota. El diagrama describe el flujo iterativo entre las fases de planeación, diseño, codificación y pruebas, lo que ilustra la naturaleza cíclica y evolutiva de la metodología XP. Tomado de (Pinto, 2020)

3.2. DISEÑO METODOLÓGICO

El proyecto se encuentra orientado hacia un enfoque cualitativo como lo expresa (Corona Lisboa, José Luis, 2018) en su investigación titulada “**Investigación cualitativa: fundamentos epistemológicos, teóricos y metodológicos**”, ya que comprende las necesidades y expectativas de los usuarios como lo son: explorar el contexto organizativo, evaluar la usabilidad del software y analizar las implicaciones más amplias de su implementación. Esto proporcionaría una base sólida para desarrollar el software de gestión e inventarios y facturación que se

ajuste a las necesidades del micro mercado “Provisiones Dylan” y que sea bien recibido por sus usuarios. Así mismo, la investigación en su metodología lo enfocaremos en la aplicada, ya que es esencial para el desarrollo del software las siguientes características: la evaluación de viabilidad, implementación y pruebas, así como para proporcionar capacitación y soporte. De esta manera, garantiza que el proyecto se lleve a cabo de manera estructurada y eficiente, maximizando las posibilidades de éxito en su implementación.

Para ejecutar este proyecto, se adoptó la metodología ágil de Extreme Programming (XP), seleccionada por su enfoque en la calidad del software, un diseño simple y la continua retroalimentación del cliente. Esta metodología es particularmente idónea para proyectos que, al igual que este, demandan una interacción estrecha con el usuario final y un elevado nivel de adaptabilidad durante las distintas fases del desarrollo

La metodología XP se fundamenta en cinco valores esenciales: comunicación, simplicidad, retroalimentación, coraje y respeto. Estos principios se evidenciaron en la dinámica del proyecto desde sus inicios, fomentando una colaboración continua con el micro mercado Provisiones Dylan, lo que facilitó una identificación exacta de los requerimientos tanto funcionales como no funcionales del sistema. Del mismo modo, permitió la entrega de versiones parciales y operativas del software a lo largo del proceso, recopilando observaciones clave para su mejora continua.

En el ámbito operativo, la metodología XP se implementó a través de ciclos iterativos y frecuentes, donde se abordaron distintas fases: planificación, diseño, codificación, pruebas y lanzamiento. Durante la planificación, se desarrollaron las historias de usuario y se ordenaron según su impacto en el negocio. En la fase de diseño, se creó una arquitectura fundamentada en principios de diseño limpio (Clean Architecture) y se establecieron las estructuras internas del software. Luego, en la etapa de codificación, se llevó a cabo la implementación de cada historia de usuario de manera incremental, manteniendo buenas prácticas como la refactorización continua y el desarrollo modular. Finalmente, en la fase de pruebas, se realizaron pruebas unitarias, funcionales y de aceptación por parte del usuario, con el objetivo de asegurar la estabilidad y fiabilidad del sistema antes de su despliegue final.

Optar por XP como metodología no solo mejoró la calidad técnica del producto que se entregó, sino que también aseguró un desarrollo que era flexible, organizado y conforme a las verdaderas necesidades del micro mercado. Esta metodología fue determinante para alcanzar los objetivos del proyecto, facilitar la participación activa del cliente y desarrollar un software sólido y operativo

3.3. TIPO Y ENFOQUE DE INVESTIGACIÓN

Para el desarrollo del software de gestión, inventario y facturación para el micro mercado “Provisiones Dylan” de la Ciudad de Valledupar, Cesar, se aplicó una investigación aplicada. Este tipo de investigación se enfoca en desarrollar nuevas soluciones tecnológicas para resolver problemas específicos o mejorar la eficiencia y el rendimiento de los procesos existentes. Además, se utilizó un enfoque cualitativo, recolectando datos a través de entrevistas con el propietario y empleados del micro mercado, así como la observación directa de los procesos actuales de gestión e inventario (Macias, 2023).

3.4. POBLACIÓN

La población total del proyecto de investigación está conformada por el personal del micro mercado

PERSONAL	CANTIDAD	PORCENTAJE
PROPIETARIO	1	50%
ADMINISTRADOR	1	50%
TOTAL	2	100%

Tabla 1. Población del micro mercado

Nota. Los autores.

3.5. MUESTRA

Al ser la población tan reducida, la población se convierte en la misma muestra.

3.6. TÉCNICAS PARA LA RECOLECCIÓN DE DATOS

Con el fin de recolectar la información esencial para desarrollar el software de gestión, inventario y facturación del micro mercado Provisiones Dylan, se utilizaron dos técnicas cualitativas clave: la entrevista y la observación directa. Estas herramientas hicieron posible recopilar información relevante, veraz y contextualizada acerca del funcionamiento del negocio, lo que facilitó la clara definición de los requisitos tanto funcionales como no funcionales del sistema

La entrevista se llevó a cabo directamente con el representante legal del micro mercado, con la finalidad de entender de manera directa sus necesidades, problemas y expectativas en relación con el sistema a implementar. Mediante la formulación de preguntas estructuradas y semiestructuradas, se pudieron identificar los procesos más críticos, como la gestión manual del inventario, los

errores recurrentes en la facturación y la ausencia de informes para la toma de decisiones. Esta metodología facilitó la recopilación de información detallada, fundamentada en la experiencia del usuario, que posteriormente se convirtió en historias de usuario y características clave del sistema.

Por otro lado, la observación directa fue realizada durante visitas al lugar, donde se pudo examinar el flujo efectivo de las actividades diarias. Se observaron las labores de registro de productos, ventas, manejo de inventario y cierre de caja, destacando cuellos de botella, errores por duplicidad de datos y tiempos extendidos en la atención al cliente. Esta técnica otorgó una perspectiva objetiva y contextual del ambiente, facilitando la validación de la información recogida en la entrevista y la identificación de necesidades que el cliente no había expresado.

Las dos técnicas se complementaron de manera eficaz en la recolección de información, formando la base para el desarrollo del sistema. Gracias a la entrevista y a la observación directa, se pudo crear un software que se adaptara a la realidad del negocio, asegurando que su funcionamiento abordara los problemas reales identificados y ofreciera mejoras significativas en la gestión operativa del micro mercado Provisiones Dylan.

3.6.1. Entrevistas:

A través de esta técnica se busca conocer el nivel de aceptación del proyecto a nivel administrativo, mediante una serie de preguntas dirigidas al propietario del micro mercado Provisiones Dylan.

3.6.2. Observación directa:

Esta técnica se utiliza para identificar, observar y analizar las fallas, evaluando el estado actual de los procesos de gestión e inventario del micro mercado, verificando los problemas mencionados en la formulación del problema de investigación.

4. ESQUEMA TEMÁTICO

Cumpliendo con los objetivos específicos para alcanzar el general de manera correspondiente. En este índice se presentarán los capítulos que forman parte del análisis, diseño y desarrollo del software de gestión, inventario y facturación, destinado al micro mercado "Provisiones Dylan" ubicado en la Ciudad de Valledupar, Cesar. De la misma forma, la metodología y las técnicas empleadas en la documentación para llevar a cabo el informe que comprende la nomenclatura de cada capítulo de la investigación.

4.1. OBJETIVO 1

Establecer los requisitos funcionales y no funcionales para diseñar los módulos de gestión, inventario y facturación para el micro mercado "Provisiones Dylan".

4.1.1. FASE 1: PLANIFICACIÓN

En esta etapa de planificación se identificaron las historias de usuario, que describen las funcionalidades específicas del software. Cada historia se detalla en una tarjeta con un número, un título, y se asigna a las personas responsables de su desarrollo. Estas historias se priorizan según las necesidades del cliente y se descomponen en versiones. La planificación se revisa aproximadamente cada dos semanas para asegurar las entregas que serán evaluadas por el cliente.

En el transcurso del desarrollo de este proyecto se diseñó un calendario de actividades, donde se definen las etapas del modelo XP, junto con los objetivos del proyecto. Además, se lleva a cabo una estimación de los recursos requeridos, abarcando tanto materiales tangibles como intangibles, costos financieros y la distribución de roles y funciones concretas para el equipo de trabajo.

En el contexto del desarrollo de software para la gestión, el inventario y la facturación del micro mercado "Provisiones Dylan", se llevaron a cabo iteraciones utilizando la metodología ágil Extreme Programming (XP). Dichas iteraciones se realizaron semanales, lo que permitió que cada ciclo de trabajo se centrara en el diseño, desarrollo y validación de funcionalidades específicas priorizadas por el cliente. Este enfoque iterativo promovió un desarrollo ágil, flexible y enfocado en la mejora continua.

En cada iteración se escogió un grupo de historias de usuario, que reflejaban funcionalidades esenciales como la administración de inventarios, el registro de ventas, el manejo de usuarios y la generación de informes. Cada historia se desarrolló a través de programación en equipo, refactorización constante y

pruebas unitarias automatizadas, garantizando la calidad del producto parcial que se entregaba al final de cada ciclo. Al término de cada iteración, se entregaba una versión funcional del software al cliente, quien participaba activamente en la evaluación del progreso. Esto permitió obtener retroalimentación directa, ajustar funcionalidades, y confirmar que el sistema avanzaba en concordancia con los objetivos del negocio. De este modo, se estableció un proceso de desarrollo centrado en el cliente, altamente adaptativo y con entregas constantes de valor.

las iteraciones XP en este proyecto fueron clave para organizar el desarrollo, centrarse en las verdaderas necesidades del cliente y asegurar que cada progreso se validó de manera adecuada. La naturaleza breve y cíclica de las iteraciones facilitó un control exhaustivo del avance y promovió la identificación temprana de mejoras requeridas, garantizando de esta manera un producto robusto y acorde con las demandas del micro mercado.

4.1.1.2. Visitar el micro mercado "Provisiones Dylan" para observar cómo se gestionan los procesos de inventario y facturaciones.

En esta fase se realizó una visita al micro mercado "Provisiones Dylan" con el fin de analizar y entender cómo se están gestionando actualmente los procesos de inventario y facturación. La observación directa permitirá detectar los métodos utilizados para el control de stock, la facturación de productos y la administración de las ventas diarias. En el transcurso de la visita, se identificaron eventuales fallas en los procedimientos vigentes, tales como la ineficiencia en los registros manuales, desajustes en el control de productos o demoras excesivas en la facturación. Esta información será fundamental para establecer los requisitos funcionales del nuevo software, asegurando que la solución tecnológica sugerida satisfaga las necesidades auténticas del negocio y mejore su operatividad.

4.1.1.3. Realizar la entrevista con el representante legal del micro mercado como técnica de recolección de necesidades y expectativas del cliente respecto al software.

Se realizó una entrevista con el representante legal del micro mercado "Provisiones Dylan", (**ANEXO A**), con el objetivo de obtener información fundamental sobre sus expectativas y necesidades con relación al software. Se trataron temas como la gestión del inventario y la facturación que tienen actualmente, las principales dificultades que enfrentan, las funcionalidades que requieren y el grado de conocimiento tecnológico que posee el usuario. Esta información será clave para establecer los requerimientos básicos del sistema y asegurar que la solución que se presente se ajuste a las necesidades del negocio. De esa entrevista se consiguió la siguiente declaración:

4.1.1.4. Establecer el alcance del proyecto

La meta fundamental de este proyecto es la creación de un software de escritorio que se encargue de la gestión de inventario y facturación en el micro mercado "Provisiones Dylan", con el fin de optimizar los procesos administrativos y fortalecer el control de productos y ventas en tiempo real. Este software de escritorio facilitará la automatización de tareas, reducirá los errores en el registro de información y aumentará la eficiencia operativa del negocio.

Con el fin de alcanzar estos objetivos, el software de escritorio contará con las siguientes funcionalidades

4.1.1.5. Requisitos funcionales

- Administración de inventarios, facilitando el registro, la actualización y el seguimiento del inventario en tiempo real, búsqueda y/o filtrado.
- Creación de facturas digitales, con la posibilidad de enviarlas a los clientes a través de correo electrónico.
- Gestión de usuarios y roles, garantizando accesos diferenciados para el propietario y el personal.
- Elaboración de informes sobre ventas y existencias, lo que permite una mejor toma de decisiones estratégicas.
- Verificaciones de funcionamiento y estabilidad, garantizando el adecuado rendimiento del sistema.
- Capacitación para el dueño y los empleados, con el fin de lograr un uso eficiente de la plataforma.

4.1.1.6. Requerimientos No Funcionales

- Rendimiento y Estabilidad
- El sistema debe contar con verificaciones que garanticen su correcto funcionamiento.
- El sistema debe mantener la estabilidad durante su operación continua.
- Usabilidad y Capacitación
- El sistema debe contar con procesos de capacitación para el dueño y los empleados.
- El sistema debe facilitar el uso eficiente por parte de todos los usuarios capacitados.

4.1.1.7. Elaborar las historias de usuario que describan las funcionalidades con las que va a interactuar el usuario.

Se crearon las historias de usuario para especificar las funcionalidades que utilizan los usuarios del software. Estas historias explicarán de forma clara y precisa las necesidades del propietario y del personal del micro mercado "Provisiones Dylan", estableciendo los requisitos fundamentales del sistema. Se incluirán acciones como la administración de inventario, facturación, elaboración de reportes y control de usuarios, garantizando que el software sea fácil de usar y eficiente.



Figura 4. Historias de Usuarios
Nota. Los autores

4.1.1.8. Recursos necesarios para el proyecto

A continuación, se detallan los recursos que se tienen a disposición para llevar a cabo el proyecto:

4.1.1.9. Recursos humanos

Para este proyecto se proponen los siguientes participantes; quienes son las personas que están directa o indirectamente relacionados con la ejecución del proyecto, como se observa en la Tabla 2.

Rol	Participantes
Directora:	Karen Lorena León Pérez
Codirector:	Francisco Chinchilla Torres
Evaluador:	Erney Alberto Ramírez Camargo
Evaluadora:	Katherine Beleño Caselles
Población objetivo:	Propietario del negocio y trabajadores del micro mercado Provisiones Dylan
Intermediario:	Micro mercado las Provisiones Dylan
Estudiante de Ingeniería de Sistemas y autor del proyecto	Santiago Pupo Cogollo
Estudiante de Ingeniería de Sistemas y autor del proyecto	Jhon Jaider Manosalva Rangel

Tabla 2. Participantes del proyecto.

Nota. los autores.

4.1.1.10. Recursos Materiales, Servicios y Financieros.

A continuación, se pueden evidenciar los recursos disponibles para la realización del proyecto, como se observa en la tabla 3.

ELEMENTOS	CANTIDAD	VALOR UNITARIO	VALOR TOTAL
Computador portátil	1	\$ 2.500.000	\$ \$ 2.500.000
Internet	--	\$ 65.000	\$ 65.000
Impresiones	150	\$ 200	\$ 30.000
Cuaderno	2	\$ 3.500	\$ 7.000
Lapicero	2	\$ 2.700	\$ 5.400
TOTAL			\$ 2.607.400

Tabla 3. Recursos disponibles del proyecto.

Nota: Los autores.

4.2. OBJETIVO 2

Diseñar el software que integre las funcionalidades que son requeridas para la gestión, inventario y facturación del micro mercado.

4.2.1. FASE 2: DISEÑO

En la fase de **diseño**, se desarrolla un código simple y funcional que cubra la historia del usuario, siempre teniendo en cuenta su experiencia. Se planifica una programación colaborativa y flexible, preparando el software para su evaluación en la siguiente etapa. En el caso de diseño orientado a objetos, se utilizan tarjetas CRC (Clase-Responsabilidad-Colaboración) para describir las clases involucradas en la programación de la historia.

En el proceso de desarrollo del sistema de gestión, inventario y facturación para el micro mercado Provisiones Dylan, se implementaron criterios de ingeniería de software que buscan asegurar la sostenibilidad, escalabilidad y solidez de la solución brindada. Un aspecto esencial del diseño técnico fue la elección de una arquitectura multicapa, basada en los principios de Clean Architecture propuestos por Robert C. Martin (Uncle Bob), junto con los principios de diseño SOLID. Esta elección responde a la necesidad de crear un sistema con una clara división de responsabilidades, lo que facilita su mantenimiento y evolución a lo largo del tiempo.

Arquitectura multicapa aplicada

El sistema está organizado en cuatro capas fundamentales, cada una con responsabilidades claras y definidas, lo que asegura una disposición lógica del código y facilita el desacoplamiento de las funcionalidades del sistema.

Capa de presentación (Presentation Layer): Esta capa constituye la interfaz gráfica que utiliza el usuario final para interactuar. Se desarrolló utilizando Windows Forms en el entorno .NET 8, teniendo en cuenta la facilidad de uso y las características de los usuarios del micro mercado. En esta capa se crearon formularios que son accesibles e intuitivos, facilitando la gestión de clientes, productos, ventas, proveedores e inventario. Los eventos de la interfaz, tales como clics o validaciones, están conectados a controladores que dirigen las solicitudes hacia la capa de aplicación, sin incluir lógica de negocio.

Capa de aplicación (Application Layer): Actúa como intermediaria entre la interfaz y el dominio. Se encarga de orquestar los casos de uso y coordinar el flujo de datos entre las demás capas. Aquí se encapsulan los servicios de aplicación y se definen las interfaces para los repositorios, lo que permite

mantener aislada la lógica del negocio de las tecnologías concretas de acceso a datos. Esta capa facilita la reutilización de lógica en diferentes interfaces si a futuro se desea extender el sistema a web o móvil.

Capa de dominio (Domain Layer): Representa el núcleo del sistema y contiene las entidades principales del negocio, como Producto, Cliente, Proveedor, Venta, Usuario, entre otras. Estas entidades definen sus atributos, comportamientos y reglas de negocio, garantizando que las operaciones que se realicen estén alineadas con los objetivos comerciales del micro mercado. La independencia de esta capa es total; no depende de ninguna tecnología externa ni de frameworks, lo que permite que pueda ser fácilmente testeada y reutilizada.

Capa de infraestructura (Infrastructure Layer): Esta capa implementa los contratos definidos en la capa de aplicación, particularmente las interfaces de los repositorios. Se encarga de realizar el acceso físico a la base de datos SQL Server, implementar la persistencia, enviar correos (en caso de ser requerido en el futuro), y gestionar otros recursos del sistema. Aquí también se manejan los detalles técnicos como conexiones, consultas SQL y almacenamiento de datos.

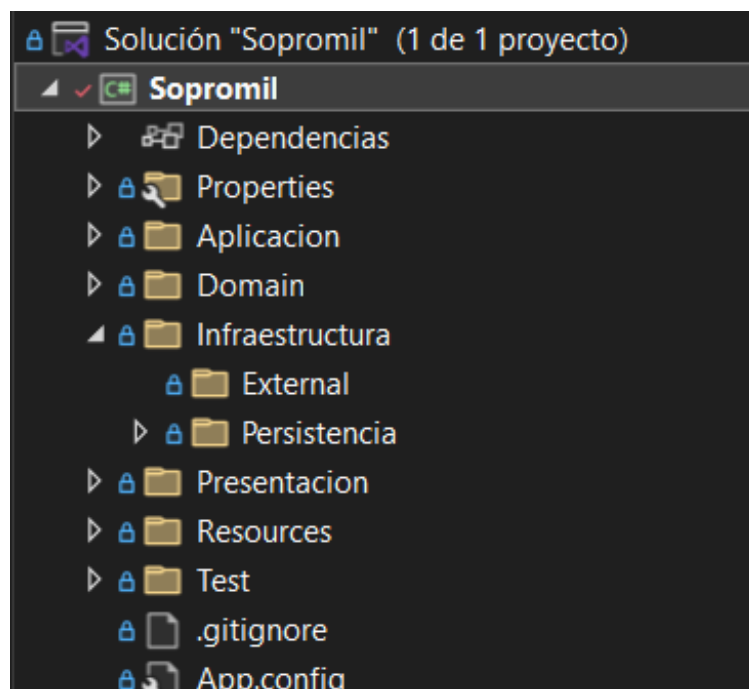


Figura 5. Estructura de la arquitectura del proyecto.
Nota. los Autores

Patrón de diseño modelo vista controlador (MVC)

Aunque el software es una aplicación de escritorio, conceptualmente se organiza siguiendo la lógica del patrón MVC.

Durante la creación del sistema de gestión, inventario y facturación para el micro mercado Provisiones Dylan, se utilizó el patrón arquitectónico Modelo Vista Controlador (MVC), el cual es bien conocido en la ingeniería de software por facilitar una estructura modular, escalable y de sencillo mantenimiento. La adopción de este modelo respondió a la necesidad de separar las distintas capas de la aplicación, lo que permite una clara división de responsabilidades entre la lógica de presentación, la lógica de negocio y el manejo del flujo de la aplicación.

En este marco, el Modelo se conforma por las entidades del dominio que agrupan los datos y las acciones del negocio, como productos, ventas, clientes y proveedores. Estas clases no dependen de la interfaz gráfica ni del almacenamiento, lo que garantiza su reusabilidad y facilita la aplicación de pruebas unitarias. La Vista, en este contexto, se presenta a través de las interfaces gráficas elaboradas con Windows Forms en .NET 8, que tienen la función de exhibir la información al usuario final y recoger sus interacciones, adhiriéndose a los principios de diseño enfocados en el usuario para asegurar una experiencia agradable y eficaz. Por último, el Controlador funciona como un intermediario lógico entre la Vista y el Modelo, manejando los eventos de la interfaz, validando la información ingresada, organizando los casos de uso establecidos en la capa de aplicación y coordinando con los servicios requeridos para realizar las acciones pedidas por el usuario.

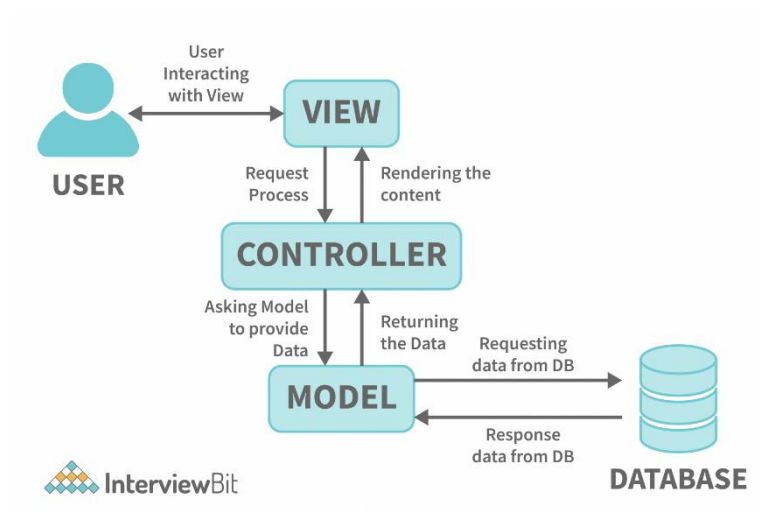


Figura 6. Diagrama del patrón de diseño MVC

Nota. www.interviewbit.com/blog/mvc-architecture

Estructura y organización del código de programación

Control de versiones

Para llevar un control organizado y seguro del código fuente durante el desarrollo del proyecto, se utilizó **Git** como sistema de control de versiones, junto con la plataforma **GitHub** para el almacenamiento remoto y la colaboración.

Git permitió mantener un historial completo de los cambios realizados en el proyecto, facilitando la recuperación de versiones anteriores y la solución de conflictos durante la edición del código. GitHub sirvió como repositorio central para respaldar el trabajo, compartir avances, y coordinar el desarrollo entre los integrantes del equipo, asegurando así una mayor trazabilidad, integridad del proyecto y continuidad en el proceso de desarrollo.

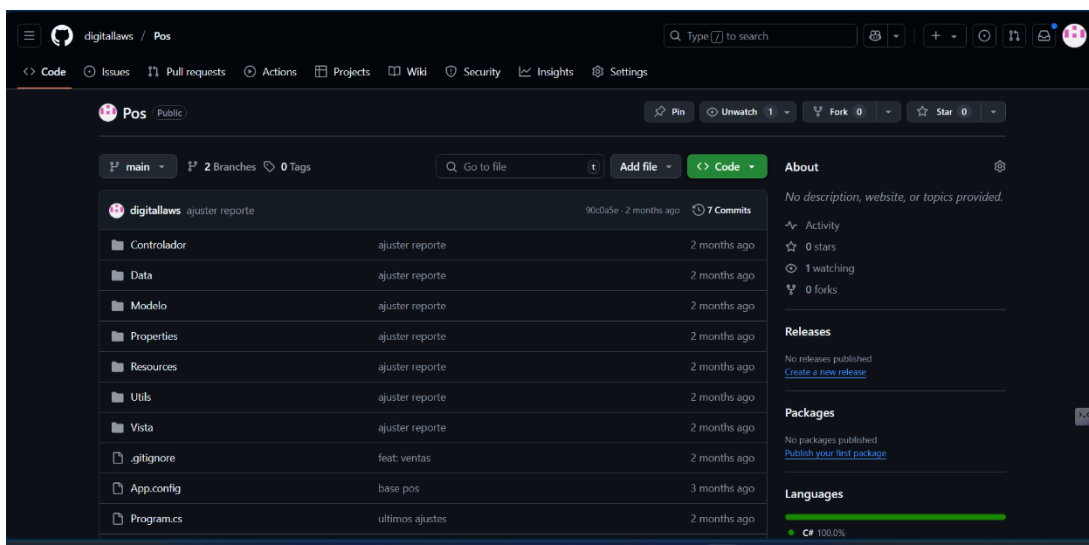


Figura 7. Repositorio github del sistema
Nota. los autores

Manejo de Errores y Fallos Inesperados

Durante el desarrollo del sistema de gestión, inventario y facturación para el micro mercado Provisiones Dylan, se incorporaron mecanismos de control de errores con el fin de mejorar la estabilidad y evitar fallos inesperados durante la ejecución del software.

Para ello, se implementaron bloques try-catch en operaciones críticas, especialmente en aquellas relacionadas con el acceso a datos, validaciones y ejecución de acciones del negocio, como el registro de ventas, gestión de inventario y actualización de clientes o proveedores. Estos bloques permiten capturar excepciones de manera controlada y evitar que el sistema se detenga bruscamente ante un error.

Cuando ocurre una excepción, el sistema responde mostrando mensajes claros al usuario final, indicando que ha ocurrido un error sin exponer detalles técnicos. Esto contribuye a una experiencia de usuario más segura y estable.

```
1 referencia
public async Task<bool> InsertarDescuentoAsync(Descuento descuento)
{
    try
    {
        using var connection = new SqlConnection(_connectionString);
        using var command = new SqlCommand("InsertarDescuento", connection);
        command.CommandType = CommandType.StoredProcedure;

        command.Parameters.AddWithValue("@IDProducto", descuento.IDProducto);
        command.Parameters.AddWithValue("@ValorOriginal", descuento.ValorOriginal);
        command.Parameters.AddWithValue("@Cantidad", descuento.Cantidad);
        command.Parameters.AddWithValue("@ValorDescuentoUnitario", descuento.ValorDescuentoUnitario);
        command.Parameters.AddWithValue("@ValorDescuentoTotal", descuento.ValorDescuentoTotal);

        await connection.OpenAsync();
        await command.ExecuteNonQueryAsync();
        return true;
    }
    catch (Exception ex)
    {
        MessageBox.Show($"Error al insertar descuento: {ex.Message}", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
        return false;
    }
}
```

Figura 8. Manejo de errores Try Catch
Nota. los autores

4.2.1.2. Diseñar las vistas del aplicativo de escritorio para determinar los colores, fuentes y que el diseño sea agradable.

El diseño del aplicativo de escritorio se enfocó en garantizar una interfaz **intuitiva, limpia y profesional**, utilizando una paleta de colores sobria y elementos visuales bien distribuidos. Se priorizó la usabilidad y accesibilidad para mejorar la experiencia del usuario.

4.2.1.3. Paleta de Colores y Estilos

- **Color principal:** Azul (#0078D7) – Representa confiabilidad y es utilizado en iconos y elementos destacados.
- **Color secundario:** Blanco (#FFFFFF) – Para fondos y mantener un diseño limpio.
- **Color de texto:** Negro (#000000) – Para asegurar buena legibilidad.
- **Tipografía:** Se utilizó **Segoe UI**, una fuente moderna y fácil de leer, con los siguientes tamaños:
- **Títulos:** 16-20 px, en negrita.
- **Texto principal:** 12-14 px

4.2.1.4. Diseño de la Pantalla de Selección de Perfil



Figura 9. Diseño de la Pantalla de Selección de Perfil.
Nota. los Autores

Esta vista permite al usuario elegir un perfil antes de iniciar sesión. Presenta un diseño limpio y minimalista, con un **icono representativo** y el **nombre del usuario** debajo. El diseño busca facilitar la identificación rápida y mejorar la experiencia de navegación.

4.2.1.5. Pantalla de Inicio de Sesión con Teclado Numérico



Figura 10. Pantalla de Inicio de Sesión con Teclado Numérico
Nota. Los Autores

Esta interfaz permite el ingreso de la contraseña mediante un teclado numérico. Incluye un botón de retroceso para corregir errores y un botón de limpieza rápida. El diseño es intuitivo, con colores diferenciados para las acciones clave.

4.2.1.6. Pantalla de Menú Principal



Figura 11. Pantalla de menú Principal
Nota. Autores

Esta interfaz presenta un diseño **intuitivo y organizado**, permitiendo el acceso rápido a los módulos clave del sistema. Cada opción está representada con un **icono visual** y una etiqueta descriptiva, facilitando la navegación del usuario. En la parte superior se encuentra un botón de **"Cerrar Sesión"**, asegurando un acceso seguro al sistema.

4.2.1.7. Pantalla de Gestión de Proveedores

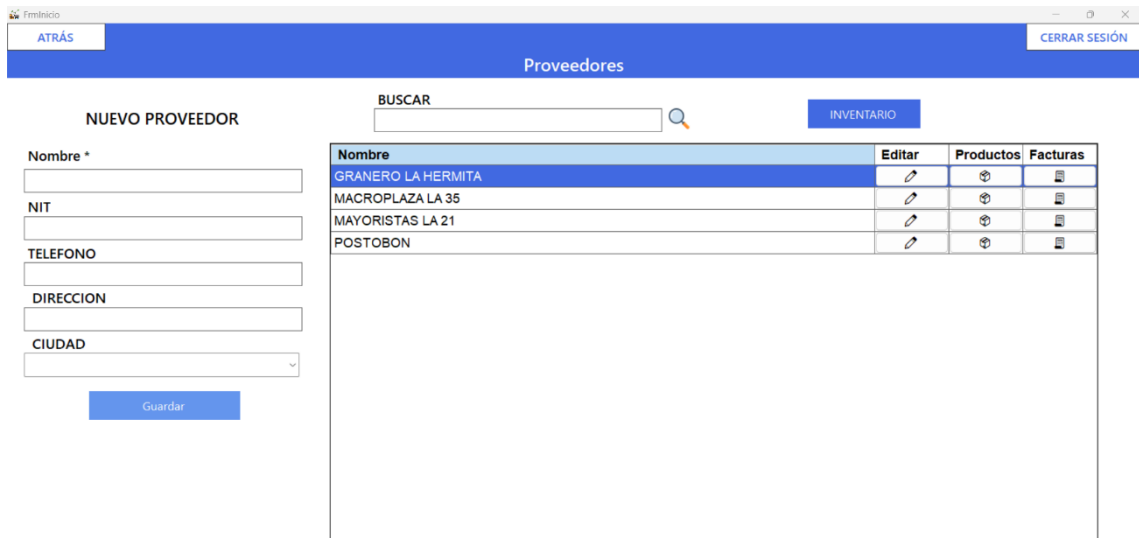


Figura 12. Pantalla de gestión de proveedores
Nota. Autores

Esta interfaz permite **registrar, buscar, editar y eliminar proveedores** en el sistema. Presenta un **formulario de ingreso** en el lado izquierdo y una **tabla de proveedores** en el lado derecho con opciones de edición y eliminación. Incluye una barra de búsqueda para una localización rápida y un botón de acceso al inventario asociado. El diseño mantiene una estructura clara y accesible para una gestión eficiente.

4.2.1.8. Pantalla de Registro de Productos

Productos - POSTOBON

Atras REGISTRAR PRODUCTO

Proveedor POSTOBON Nombre * Marca Unidad de medida Codigo (Opcional) Limpiar

Fecha de Vencimiento Cantidad Valor Compra Flete Utilidad Precio Venta

Porcentaje %

Nombre	Marca	Unidad	Código	Cantidad	Compra	Venta	Editar	Eliminar
GASEOSA	POSTOBON	UNIDAD	43645756876...	20	20.000	28.000	✎	✕
GASEOSA 1.5...	POSTOBON	UNIDAD	43546757	20	3.000	5.000	✎	✕

Agregar

Finalizar Registro

Flete Total: \$ 120.000
 Total Compra: \$ 460.000
 Total Venta: \$ 660.000
 Utilidad Neta: \$ 80.000

Figura 13. Pantalla de registro de productos
 Nota. Autores.

Esta interfaz permite registrar nuevos productos en el sistema, asociándolos a un proveedor. Incluye un **formulario de ingreso** con datos clave como cantidad, costos y utilidad. Los productos agregados se muestran en una tabla con opciones de **edición y eliminación**. Además, se presentan **cálculos automáticos** de costos y utilidades para facilitar la gestión de inventarios.

4.2.1.9. Pantalla de Gestión de Inventario



Buscar Proveedor

Código	Nombre	Marca	Cantidad	Unidad	Valor Compra	Venta	Historial	Ubicación
1472583690236	Aceite de Gira.	Naturelle	31	MILILITRO	7.783	7.783		?
7256984310459	Ajo 100g	Natural	53	GRAMO	15.190	15.190		?
1234567890123	Arroz 1kg	La Libertad	36	KILOGRAMO	7.519	7.519		?
8462751930641	Atún en Lata 1...	Bumble Bee	82	GRAMO	6.729	6.729		?
9876543210987	Azúcar 1kg	Diana	19	KILOGRAMO	7.044	7.044		?
6741983563208	Came Molida ...	Carnes La Vict...	15	KILOGRAMO	5.198	5.198		?
3651478590247	Cebolla 1kg	La Huerta	48	KILOGRAMO	14.740	14.740		?
5487691234567	Cereal Cornfla...	Kellogg	79	GRAMO	16.804	16.804		?
1324657890245	Frijoles 1kg	La Fama	14	KILOGRAMO	13.545	13.545		?
7531862049853	Galletas Maria ...	Galletera	54	GRAMO	5.638	5.638		?
7418529630842	Harina de Trig...	Fama	20	KILOGRAMO	18.899	18.899		?
8527419630874	Huevos x12	Pueblo	89	UNIDAD	10.256	10.256		?
4738290567910	Jugo Naranja 1L	Tropicana	99	LITRO	14.776	14.776		?
6258741903742	Ketchup 500g	Heinz	81	GRAMO	7.510	7.510		?
2589631470253	Leche 1L	Alpina	60	LITRO	14.983	14.983		?
9645328710234	Lentejas 1kg	La Fama	54	KILOGRAMO	15.336	15.336		?
9853746102305	Margarina 250g	Regina	11	GRAMO	6.018	6.018		?
5279836410298	Mayonesa 400g	Hellmann	13	GRAMO	5.993	5.993		?
6382751093846	Mostaza 200g	French	73	GRAMO	19.535	19.535		?
7462938195036	Pan de Molde ...	Bimbo	65	GRAMO	16.476	16.476		?

Activo Total: 19.495.691

AGREGAR PRODUCTO
REPORTES

Figura 14. Pantalla de gestión de inventario
Nota. Autores.

Esta interfaz permite visualizar y administrar el inventario del sistema. Incluye una **barra de búsqueda y filtro por proveedor**, mostrando información detallada de los productos como cantidad, costos y precios de venta. También ofrece opciones para **ver el historial de movimientos, ubicación del producto y agregar nuevos artículos**. Se muestra el **total del inventario activo** y botones de acceso rápido a la gestión de productos y reportes.

4.2.1.10. Pantalla de Registro de Venta

REGISTRAR VENTA

CLIENTE * ADIN DOCUMENTO 1012543678 TELEFONO 222

BUSCAR PRODUCTO Stock: 31

Producto	Cantidad	Precio Unitario	Total	+	-
Papel Higiénico x12	1	11.066	11.066	+	-
Pan de Molde 450g	1	16.476	16.476	+	-
Frijoles 1kg	1	13.545	13.545	+	-
Aceite de Girasol 900ml	1	7.783	7.783	+	-

TOTAL: Total: \$ 48.870

Figura 15. Pantalla de registro de venta
Nota. Autores.

Esta interfaz permite registrar una nueva venta, seleccionando un cliente y agregando productos al detalle de compra. Incluye opciones para **buscar productos, ajustar cantidades y calcular el total automáticamente**. Los botones de **"Agregar"** y **"Facturar"** agilizan la finalización de la transacción. El diseño es limpio y organizado para facilitar su uso en entornos de atención al cliente.

4.2.1.11. Diseño de la base de datos

Diagrama Entidad Relación

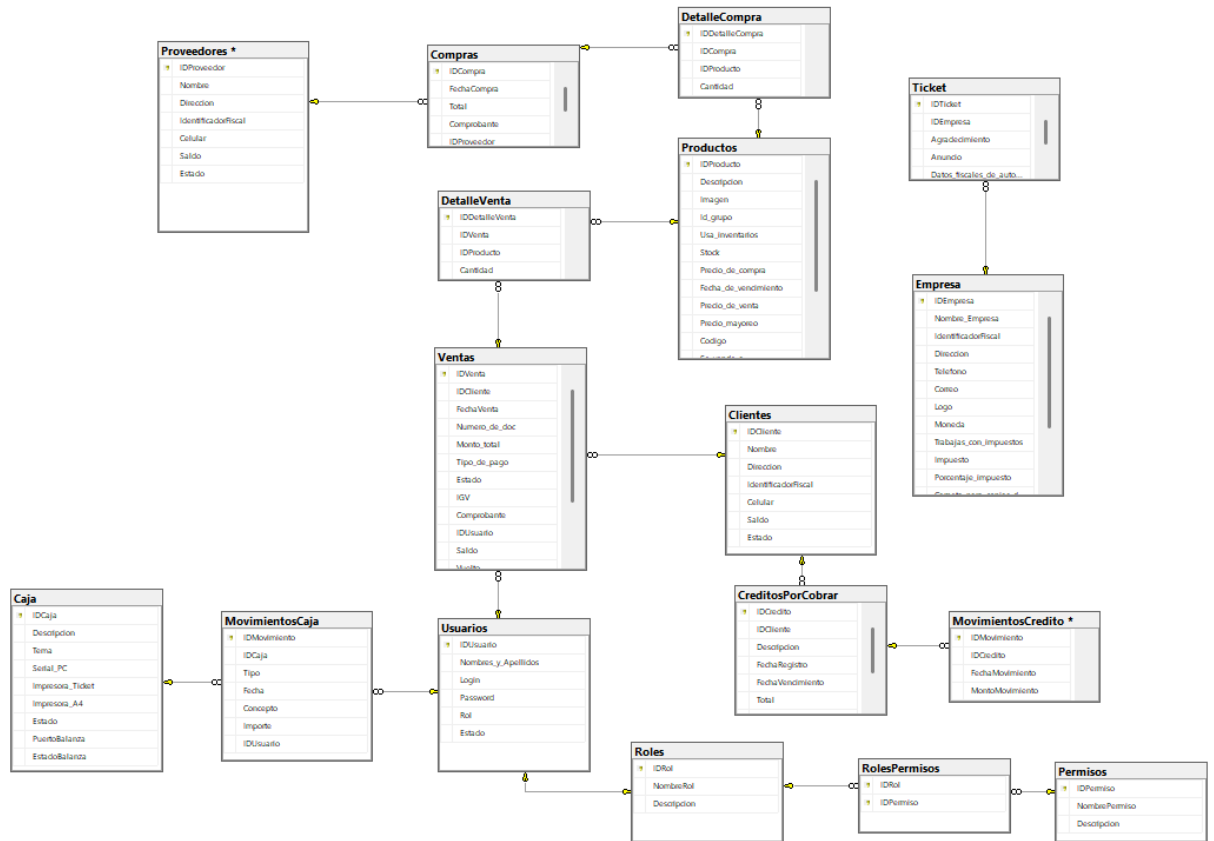


Figura 16. Diagrama Entidad Relación
Nota. Los autores

4.2.1.12. Diccionario de Datos

Tabla 4. Empresa

Nombre del Campo	Tipo de Dato	Descripción	Restricciones
IDEmpresa	INT	Identificador único de la empresa	PRIMARY KEY, DEFAULT 1
Nombre	VARCHAR (100)	Nombre de la empresa	NOT NULL
NIT	VARCHAR (50)	Número de identificación tributaria	NULL
Dirección	VARCHAR (255)	Dirección de la empresa	NULL
Teléfono	VARCHAR (50)	Teléfono de contacto	NULL
Correo	VARCHAR (100)	Correo electrónico de la empresa	NULL
Logo	VARBINARY(MAX)	Imagen del logo	NULL
Moneda	VARCHAR (10)	Tipo de moneda utilizada	DEFAULT 'COP'

Nota. Los Autores

Tabla 5. Proveedores

Nombre del Campo	Tipo de Dato	Descripción	Restricciones
IDProveedor	INT	Identificador único del proveedor	PRIMARY KEY, IDENTITY
Nombre	VARCHAR (100)	Nombre del proveedor	NOT NULL
IdentificadorFiscal	VARCHAR (50)	Identificación tributaria del proveedor	UNIQUE, NULL
Teléfono	VARCHAR (50)	Teléfono de contacto	NULL
Dirección	VARCHAR (255)	Dirección del proveedor	NULL
Ciudad	VARCHAR (100)	Ciudad donde opera el proveedor	NULL
Estado	VARCHAR (50)	Estado del proveedor	DEFAULT 'Activo'

Nota. Los Autores

Tabla 6. Productos

Nombre del Campo	Tipo de Dato	Descripción	Restricciones
IDProducto	INT	Identificador único del producto	PRIMARY KEY, IDENTITY
Descripción	VARCHAR (100)	Nombre o descripción del producto	NOT NULL
Marca	VARCHAR (50)	Marca del producto	NULL
UnidadMedida	VARCHAR (20)	Unidad de medida (ej. Unidad, Litro)	DEFAULT 'Unidad'
Stock	DECIMAL (18,2)	Cantidad disponible en inventario	DEFAULT 0, CHECK (Stock >= 0)
FechaVencimiento	DATE	Fecha de caducidad del producto	NULL
CodigoBarras	VARCHAR (50)	Código de barras único del producto	UNIQUE, NULL
PrecioCompra	DECIMAL (18,2)	Precio de compra	NOT NULL, CHECK (PrecioCompra >= 0)
Flete	DECIMAL (18,2)	Costo del transporte	DEFAULT 0, CHECK (Flete >= 0)
Utilidad	DECIMAL (18,2)	Margen de ganancia sobre el costo	DEFAULT 0, CHECK (Utilidad >= 0)
ValorVenta	DECIMAL (18,2)	Precio de venta calculado	AS (PrecioCompra + Flete + Utilidad) PERSISTED
IDProveedor	INT	Relación con el proveedor	FOREIGN KEY REFERENCES Proveedores (IDProveedor)
Estado	VARCHAR (50)	Estado del producto	DEFAULT 'Activo', CHECK ('Activo', 'Inactivo')
UbicacionEstante	VARCHAR (50)	Ubicación física en la tienda	NULL
UbicacionSeccion	VARCHAR (50)	Sección dentro del estante	NULL

Nota. Los autores

Tabla 7. Usuarios:

Nombre del Campo	Tipo de Datos	Descripción	Restricciones
IDUsuario	INT	Identificador único del usuario	PRIMARY KEY, IDENTITY
Nombre	VARCHAR (100)	Nombre completo del usuario	NOT NULL
Login	VARCHAR (50)	Nombre de usuario (único)	UNIQUE, NOT NULL
Password	VARCHAR (255)	Contraseña cifrada	NOT NULL
Rol	VARCHAR (50)	Tipo de rol (Admin, Vendedor, Cajero)	CHECK ('Admin', 'Vendedor', 'Cajero'), NOT NULL
Estado	VARCHAR (50)	Estado del usuario	DEFAULT 'Activo', CHECK ('Activo', 'Inactivo')

Nota. Los autores

Tabla 8. Clientes

Nombre del Campo	Tipo de Datos	Descripción	Restricciones
IDCliente	INT	Identificador único del cliente	PRIMARY KEY, IDENTITY
Documento	VARCHAR (50)	Número de identificación única	UNIQUE, NOT NULL
Nombre	VARCHAR (100)	Nombre completo del cliente	NOT NULL
Teléfono	VARCHAR (20)	Número de contacto	NULL
Estado	VARCHAR (10)	Estado del cliente	DEFAULT 'Activo', CHECK ('Activo', 'Inactivo')

Nota. Los autores

Tabla 9. Ventas

Nombre del Campo	Tipo de Datos	Descripción	Restricciones
IDVenta	INT	Identificador único de la venta	PRIMARY KEY, IDENTITY
IDCliente	INT	Cliente asociado a la venta	FOREIGN KEY Clientes
FechaVenta	DATETIME	Fecha y hora de la venta	DEFAULT GETDATE ()
TipoVenta	VARCHAR (10)	Contado o crédito	CHECK ('Contado', 'Crédito')-
TotalVenta	DECIMAL (18,2)	Monto total de la venta	NOT NULL,
MontoAbonado	DECIMAL (18,2)	Cantidad abonada	DEFAULT 0,
SaldoPendiente	DECIMAL (18,2)	Diferencia entre total y abonado	PERSISTED
Estado	VARCHAR (20)	Pendiente, pagado, cancelado	DEFAULT 'Pendiente'
FechaEstimadaPago	DATE	Fecha estimada de pago (si es crédito)	NULL

Nota. Los autores

Tabla 10. Detalle de ventas.

Nombre del Campo	Tipo de Datos	Descripción	Restricciones
IDDetalleVenta	INT	Identificador único del detalle	PRIMARY KEY, IDENTITY
IDVenta	INT	Venta asociada	FOREIGN KEY Ventas
IDProducto	INT	Producto vendido	FOREIGN KEY REFERENCES Productos (IDProducto), NOT NULL
Cantidad	DECIMAL (18,2)	Cantidad vendida	CHECK (Cantidad > 0)
PrecioUnitario	DECIMAL (18,2)	Precio por unidad	CHECK (PrecioUnitario >= 0)
Total	DECIMAL (18,2)	Total, por cantidad x precio	AS (Cantidad * PrecioUnitario) PERSISTED

Nota. Los autores

Tabla 11. Pagos

Nombre del Campo	Tipo de Datos	Descripción	Restricciones
IDPago	INT	Identificador único del pago	PRIMARY KEY, IDENTITY
IDCliente	INT	Cliente que realiza el pago	FOREIGN KEY Clientes
MontoAbonado	DECIMAL (18,2)	Monto pagado	NOT NULL, CHECK (MontoAbonado > 0)
FechaPago	DATETIME	Fecha y hora del pago	DEFAULT GETDATE ()

Nota. Los autores

Tabla 12. Compras

Nombre del Campo	Tipo de Datos	Descripción	Restricciones
IDCompra	INT	Identificador único de la compra	PRIMARY KEY, IDENTITY
IDProveedor	INT	Proveedor que suministra la compra	FOREIGN KEY REFERENCES Proveedores (IDProveedor), NOT NULL
FechaCompra	DATETIME	Fecha en la que se realizó la compra	DEFAULT GETDATE ()
TotalCompra	DECIMAL (18,2)	Monto total de la compra	NOT NULL, CHECK (TotalCompra >= 0)
Flete	DECIMAL (18,2)	Costo del transporte de la compra	DEFAULT 0, CHECK (Flete >= 0)
Estado	VARCHAR (20)	Estado de la compra	DEFAULT 'Finalizada', CHECK ('Pendiente', 'Finalizada', 'Cancelada')

Nota. Los Autores

Tabla 13. Detalle de compra

Nombre del Campo	Tipo de Datos	Descripción	Restricciones
IDDetalleCompra	INT	Identificador único del detalle de compra	PRIMARY KEY, IDENTITY
IDCompra	INT	Compra asociada	FOREIGN KEY REFERENCES Compras (IDCompra), NOT NULL
IDProducto	INT	Producto comprado	FOREIGN KEY REFERENCES Productos (IDProducto), NOT NULL
Cantidad	DECIMAL (18,2)	Cantidad comprada	CHECK (Cantidad > 0)
PrecioUnitario	DECIMAL (18,2)	Precio por unidad	CHECK (PrecioUnitario >= 0)
Total	DECIMAL (18,2)	Total, del detalle de compra	AS (Cantidad * PrecioUnitario) PERSISTED

Nota. Los Autores

Tabla 14. Caja

Nombre del Campo	Tipo de Datos	Descripción	Restricciones
IDCaja	INT	Identificador único de la caja	PRIMARY KEY, IDENTITY
Descripcion	VARCHAR (100)	Nombre o descripción de la caja	UNIQUE, NOT NULL
Impresora	VARCHAR (200)	Nombre de la impresora asociada	NULL
Copia de seguridad	VARCHAR (200)	Ruta de la copia de seguridad	NULL
Estado	VARCHAR (20)	Estado de la caja	CHECK ('Activa', 'Inactiva')

Nota. Los autores

4.3. OBJETIVO 3

Construir el código del software aplicando prácticas de programación y refactorización continua para garantizar un diseño simple y eficiente.

4.3.1. FASE 3: CODIFICACIÓN

En la fase de **codificación**, comienza el proceso de programación en parejas frente a un mismo ordenador. El objetivo es crear un código de propiedad colectiva, evitando que un solo programador lo personalice. Esto permite que todo el equipo avance de manera simultánea y esté al tanto del progreso del proyecto.

4.3.1.1. Desarrollar el código de programación que cumpla con las historias de usuario

```
1 reference
public async Task<int> AplicarPagoFacturaAsync(int idCliente, decimal montoAbonado)
{
    int idPago = 0;

    try
    {
        using (var connection = await GetConnectionAsync())
        using (var command = new SqlCommand("AplicarPagoFactura", connection))
        {
            command.CommandType = CommandType.StoredProcedure;
            command.Parameters.AddWithValue("@IDCliente", idCliente);
            command.Parameters.AddWithValue("@MontoAbonado", montoAbonado);

            var outputParam = new SqlParameter("@IDPago", SqlDbType.Int) { Direction = ParameterDirection.Output };
            command.Parameters.Add(outputParam);

            await command.ExecuteNonQueryAsync();
            idPago = (int)outputParam.Value;
        }
    }
    catch (Exception ex)
    {
        Console.WriteLine($"X Error al aplicar el pago: {ex.Message}");
        throw new Exception("Error al aplicar el pago.");
    }

    return idPago;
}
```

Figura 17. Fragmento del código del método CrearProductoAsync.
Nota. Los autores

En la implementación del sistema, se desarrolló el método CrearProductoAsync, encargado de registrar productos en la base de datos utilizando un **procedimiento almacenado**. Este método, escrito en **C# .NET**, establece una conexión asíncrona con la base de datos, añade los parámetros requeridos y ejecuta la inserción de manera segura. Se manejan excepciones para garantizar la estabilidad del sistema y se notifica al usuario el resultado del proceso.

```

/// <summary>
/// Registra una venta con sus detalles.
/// </summary>
1 reference
public async Task<int> RegistrarVentaAsync(Venta venta, List<DetalleVenta> detalles)
{
    try
    {
        using (var connection = await GetConnectionAsync())
        using (var command = new SqlCommand("InsertarVenta", connection))
        {
            command.CommandType = CommandType.StoredProcedure;
            command.Parameters.AddWithValue("@IDCliente", venta.IDCliente);
            command.Parameters.AddWithValue("@TipoVenta", venta.TipoVenta);
            command.Parameters.AddWithValue("@TotalVenta", venta.TotalVenta);

            string estadoVenta = venta.TipoVenta == "Contado" ? "Pagado" : "Pendiente";
            command.Parameters.AddWithValue("@Estado", estadoVenta);
            command.Parameters.AddWithValue("@FechaEstimadaPago",
                venta.TipoVenta == "Crédito" ? (object)venta.FechaEstimadaPago : DBNull.Value);

            var idVentaParam = new SqlParameter("@IDVenta", SqlDbType.Int) { Direction = ParameterDirection.Output };
            command.Parameters.Add(idVentaParam);

            await command.ExecuteNonQueryAsync();
            int idVenta = (int)idVentaParam.Value;

            foreach (var detalle in detalles)
            {
                await RegistrarDetalleVentaAsync(idVenta, detalle);
            }

            return idVenta;
        }
    }
    catch (Exception ex)
    {
        Console.WriteLine($"✘ Error al registrar venta: {ex.Message}");
        throw new Exception("Error al registrar la venta.");
    }
}

```

Figura 18. Fragmento del código del método RegistrarVentaAsync.
Nota. Los autores

En la implementación del sistema, se desarrolló el método RegistrarVentaAsync, encargado de registrar una venta junto con sus detalles en la base de datos mediante un **procedimiento almacenado**. Este método, escrito en **C# .NET**, establece una conexión asíncrona, inserta la venta, obtiene el ID generado y registra los productos vendidos. Se maneja el estado de la venta según el tipo de pago y se controlan excepciones para garantizar la estabilidad del proceso.

4.4. OBJETIVO 4

Realizar una prueba de aceptación de usuario que permita verificar la funcionalidad del software con los requisitos y funcionalidades establecidas en el diseño.

4.4.1. FASE 4: PRUEBAS Y LANZAMIENTO

En la fase de **pruebas**, el código de cada función se somete a pruebas unitarias continuas para detectar y corregir fallos de manera regular, ya que la metodología XP implica cambios constantes. Dado que se trabaja con tiempos cortos, el control automatizado es esencial. En ocasiones, el propio cliente, si tiene conocimientos de programación, actúa como tester, lo que garantiza que sus comentarios sean útiles para el equipo.

En la fase de **lanzamiento**, se presenta el software final. Si las etapas previas se han seguido correctamente, no deberían surgir problemas. Todas las historias de usuario han sido probadas y ajustadas según los requisitos del cliente, logrando un software que cumple con las expectativas y ha superado las pruebas tanto del tester como del equipo. (Mancuzo, 2020)

4.4.1.1. Realizar las pruebas unitarias continuas para detectar y corregir errores durante el desarrollo de cada historia

```
public class ProductoControllerTests
{
    private readonly Mock<ProductoRepository> _mockRepo;
    private readonly ProductoController _productoController;

    0 references
    public ProductoControllerTests()
    {
        _mockRepo = new Mock<ProductoRepository>();
        _productoController = new ProductoController();
    }

    [Fact]
    0 references
    public async Task ListarProductosPorProveedorAsync_DeberiaRetornarProductos()
    {
        var productos = new List<Producto>
        {
            new Producto { Descripcion = "Producto 1" },
            new Producto { Descripcion = "Producto 2" }
        };

        _mockRepo.Setup(repo => repo.ListarProductosPorProveedorAsync(It.IsAny<int>(), true)).ReturnsAsync(productos);

        var result = await _productoController.ListarProductosPorProveedorAsync(1, true);

        Assert.NotNull(result);
        Assert.Equal(2, result.Count);
        Assert.All(result, producto => Assert.False(string.IsNullOrEmpty(producto.Descripcion)));
    }
}
```

Figura 19. Registro de Producto

Nota. Los autores

```

[Fact]
0 references
public async Task CrearClienteAsync_DeberialanzarExcepcionCuandoNombreEsNulo()
{
    var exception = await Assert.ThrowsAsync<ArgumentException>(C =>
        _clienteController.CrearClienteAsync(null, "123456", "1234567890"));

    Assert.Equal("El nombre del cliente es obligatorio.", exception.Message);
}

[Fact]
0 references
public async Task CrearClienteAsync_DeberiallamarCrearClienteEnRepositorio()
{
    var cliente = new Cliente
    {
        Nombre = "Cliente Test",
        Documento = "123456",
        Telefono = "1234567890"
    };

    _mockRepo.Setup(repo => repo.CrearClienteAsync(It.IsAny<Cliente>())).ReturnsAsync(1);

    var result = await _clienteController.CrearClienteAsync(cliente.Nombre, cliente.Documento, cliente.Telefono);

    _mockRepo.Verify(repo => repo.CrearClienteAsync(It.IsAny<Cliente>()), Times.Once);
    Assert.Equal(1, result);
}

```

Figura 20. Crear cliente
Nota. Los autores

4.4.1.2. Colocar en funcionamiento el software en el micro mercado Provisiones Dylan para la prueba piloto

Se ha colocado en funcionamiento el software en el micro mercado Provisiones Dylan, habiendo completado satisfactoriamente la fase de prueba piloto. A lo largo de este proceso, se verificaron las funciones de gestión, inventario y facturación en un entorno operativo real. Se ha recogido la evidencia del correcto funcionamiento, lo que demuestra que el sistema satisface los requisitos establecidos y está preparado para su uso continuo en el negocio.



Figura 21. Evidencia del funcionamiento del software en el micromercado provisiones Dylan.
Nota. Los autores

4.4.1.3. Ejecución de pruebas de aceptación: permitir que el cliente valide funcionalidades clave.

Se llevaron a cabo las pruebas de aceptación, lo que permitió al cliente verificar las funcionalidades principales del software. En el transcurso de este proceso, se realizaron chequeos en la administración de inventario, facturación y otros módulos fundamentales. La validación del cliente ratificó que el sistema satisface los requisitos establecidos, asegurando su correcto funcionamiento y usabilidad en el entorno real del micro mercado Provisiones Dylan.

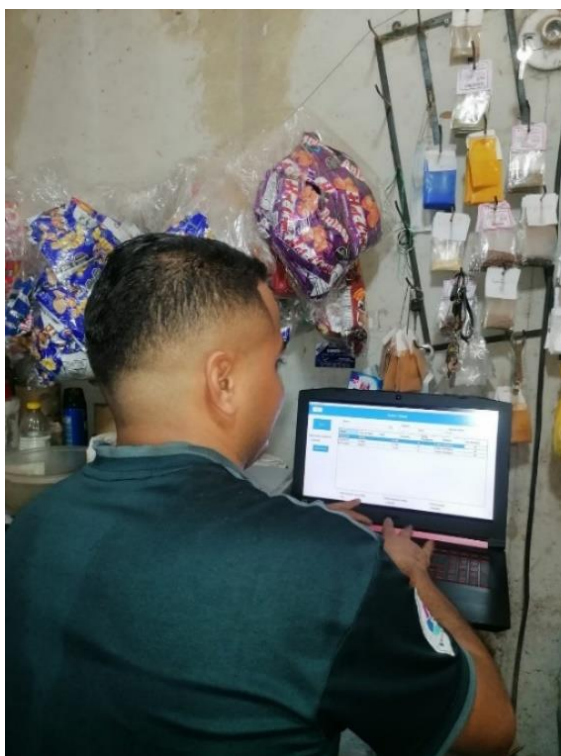


Figura 22. Evidencia de las pruebas de aceptación

Nota. Los autores

4.5. PRINCIPIO SOLID

En la creación del software de gestión, inventario y facturación para el micro mercado Provisiones Dylan, se utilizaron como fundamentos arquitectónicos los principios SOLID, esenciales en la programación orientada a objetos. La integración de estos principios permitió el desarrollo de un sistema sólido, que se puede mantener fácilmente y escalar, lo que a su vez facilita su evolución y adaptación a cambios futuros. Cada módulo del software fue elaborado conforme a buenas prácticas de ingeniería de software, con el objetivo de mantener un diseño limpio, organizado y adaptable, asegurando así un crecimiento ordenado del sistema.

El primer principio implementado fue el **Principio de Responsabilidad Única (Single Responsibility Principle)**, que indica que cada clase debe tener una sola responsabilidad. En nuestro proyecto, esto se manifestó en la división de funcionalidades en módulos específicos, como la gestión de inventarios, la facturación de ventas y la administración de usuarios. Así, cualquier modificación o mejora futura en un módulo solo impactará ese módulo y no al sistema completo, lo que favorece la facilidad de mantenimiento. Además, se llevó a cabo la aplicación del **Principio Abierto/Cerrado (Open/Closed Principle)**, que permitió ampliar las funcionalidades del sistema “como añadir reportes de ventas filtrados”, sin necesidad de modificar el código ya existente, lo que reduce el riesgo de errores.

Siguiendo con los principios, se respetó el **Principio de Sustitución de Liskov (Liskov Substitution Principle)**, garantizando que las clases derivadas pudieran sustituir a las clases base sin afectar el correcto funcionamiento del sistema. Un claro ejemplo de esta aplicación fue la creación de perfiles de usuario (administrador y cajero) que compartían funcionalidades en común, pero contaban con permisos y accesos diferenciados. Además, se llevó a cabo el **Principio de Segregación de Interfaces (Interface Segregation Principle)**, previniendo que las clases implementaran métodos que no utilizaban. Esto se logró mediante la creación de interfaces específicas según la funcionalidad, favoreciendo así un diseño más limpio y enfocado.

Finalmente, el **Principio de Inversión de Dependencias (Dependency Inversion Principle)** fue fundamental para desacoplar las clases de las implementaciones concretas. Esto permitió que el sistema se basara en abstracciones, lo que facilitó el cambio de componentes internos como la base de datos o el motor de autenticación, sin que esto afectara el funcionamiento general. En conclusión, la aplicación de los principios SOLID no solo mejoró la calidad estructural del software desarrollado, sino que también aseguró su sostenibilidad, escalabilidad y facilidad de mantenimiento, siendo este un aspecto clave en el éxito del proyecto implementado para el micro mercado Provisiones Dylan.

5. CONCLUSIONES

La ejecución de la prueba piloto del software de gestión, inventario y facturación en el micro mercado Provisiones Dylan permitió observar su desempeño en un entorno real y analizar su influencia en los procesos del negocio. A partir del uso de la metodología ágil Extreme Programming (XP), fue posible estructurar un sistema que responde a los requisitos definidos, permitiendo un flujo de trabajo organizado.

Durante las pruebas de aceptación, se revisaron funcionalidades clave como la gestión de inventarios, el registro de ventas y la facturación. En este proceso, se recopiló observaciones del cliente que orientaron la identificación de aspectos susceptibles de mejora para futuras versiones del sistema. La retroalimentación obtenida proporcionó elementos para evaluar el impacto del software en la administración del negocio.

Si bien la prueba piloto no representó una implementación definitiva, ofreció datos relevantes sobre la viabilidad del sistema en el contexto del comercio local. La capacitación y la entrega de documentación técnica buscaron facilitar el uso de la herramienta por parte de los usuarios. La experiencia adquirida durante esta etapa permitió identificar oportunidades de mejora y aspectos a fortalecer en el desarrollo continuo del sistema.

Este proyecto permitió explorar el papel de la tecnología en la gestión de pequeños negocios. A través de la prueba piloto, se identificaron factores críticos que podrían influir en futuras mejoras o ampliaciones del sistema, resaltando la importancia de ajustar las soluciones tecnológicas a las condiciones particulares de cada entorno comercial.

6. RECOMENDACIONES

A lo largo del desarrollo del sistema, se han identificado algunas áreas en las que se pueden hacer mejoras para asegurar que el sistema siga siendo útil y eficiente a largo plazo. Aquí van algunas recomendaciones clave para mantener el sistema en buen estado y ajustarlo a las necesidades futuras.

Mantenimiento Continuo

Es importante establecer un plan de mantenimiento regular para corregir errores, actualizar el sistema y agregar nuevas funciones. Esto ayudará a que el sistema se mantenga actualizado y funcionando correctamente con el tiempo.

Recoger retroalimentación

Se recomienda crear una forma sencilla de obtener comentarios de los usuarios sobre el sistema. Esto permitirá ajustar el sistema según sus necesidades y seguir mejorando la experiencia a medida que evoluciona.

7. REFERENCIAS

- Alvarado, J. T. (2021). IMPLEMENTAR UN SOFTWARE PARA EL CONTROL DE INVENTARIO EN EL NEGOCIO "ADELITA" UTILIZANDO LA METODOLOGÍA ÁGIL XP. Obtenido de [http://repositoriodigital.itstena.edu.ec:8080/jspui/bitstream/123456789/194/1/Tra bajo%20de%20Integraci%3%b3n%20Curricular.pdf](http://repositoriodigital.itstena.edu.ec:8080/jspui/bitstream/123456789/194/1/Tra%20bajo%20de%20Integraci%3%b3n%20Curricular.pdf)
- Álvarez Martínez, E. R. (enero de 2022). SISTEMA PARA LA FACTURACIÓN Y VENTAS EASY COMPTE. Obtenido de <https://repositorio.unitec.edu/bitstream/handle/123456789/10383/31251087-enero2022-i06-pg.pdf?sequence=1&isAllowed=y>
- Amazon Web Services. (2023). Obtenido de <https://aws.amazon.com/es/what-is/web-application/>
- Beltran, N. (30 de enero de 2023). *¿Para qué sirve el software? Categorías y desarrollo de calidad*. Obtenido de <https://tuatara.co/blog/software/categorias-desarrollo-calidad/>
- Bogota, C. d. (2023). *Número de empresas activas*. Obtenido de <https://www.ccb.org.co/informacion-especializada/observatorio/dinamica-empresarial/empresas-activas?>
- Cabanes, N. (2022). *Introducción a la programación con C#*. Obtenido de https://csharp.com.es/wp-content/uploads/2013/12/introCsharp_version099.pdf
- Cabrera, J. B. (2021). IMPLEMENTACIÓN DE UN SISTEMA WEB PARA FACTURACIÓN ELECTRÓNICA Y CONTROL DE INVENTARIOS EN EL LOCAL COMERCIAL "IMPORTADORA EL CINCUENTAZO MATRIZ" DEL CANTÓN TENA. Obtenido de [http://repositoriodigital.itstena.edu.ec:8080/jspui/bitstream/123456789/258/1/Tra bajo%20de%20Integraci%3%b3n%20Curricular.pdf](http://repositoriodigital.itstena.edu.ec:8080/jspui/bitstream/123456789/258/1/Tra%20bajo%20de%20Integraci%3%b3n%20Curricular.pdf)
- Cámara de Comercio de Cali. (2023). Informe de gestión 2023. Obtenido de <https://www.ccc.org.co/inc/uploads/2024/03/INFORME-GESTION-CCC-2023.pdf>
- Cámara de Comercio de Casanare. (2024). Informe de gestión 2023. Obtenido de <https://cccasanare.co/wp-content/uploads/2024/02/Informe-de-Gestion-Camara-de-Comercio-de-Casanare-2023-06022023-comprimido-1.pdf>
- CANTOS, J. A., & ESTEFANÍA, M. O. (2021). SISTEMA WEB PARA GESTIONAR LOS SERVICIOS DE SPA, CONTROL DE INVENTARIO Y FACTURACIÓN DE LA VETERINARIA CANDYPET. Obtenido de <https://cia.uagraria.edu.ec/Archivos/MORENO%20ORTIZ%20KIRA%20ESTEFANIA.pdf>
- CFOremoto, & Philippe. (2021). Método FIFO: concepto, ejemplos y diferencias con el LIFO. Obtenido de <https://blog.cforemoto.com/metodo-fifo-concepto-ejemplos-diferencias-lifo/>
- Clavijo, B. P., & Cepeda, M. J. (18 de 04 de 2023). Evaluación de la gestión de inventarios y facturación en el almacén Peregrine Falcón. Obtenido de <https://ciencialatina.org/index.php/cienciala/article/view/5576/8423>
- CNI. (14 de 05 de 2020). Impactos del covid-19 en la industria. Obtenido de <https://www.portaldaindustria.com.br/estatisticas/sondesp-77-impactos-da-covid-19-na-industria/>
- Comunidad Andina, 2000. (s.f.). Obtenido de https://www.redjurista.com/Documents/decision_486_de_2000_.aspx#/
- Confecamaras, C. c. (2020). Encuesta de las cámaras de comercio sobre el impacto del covid-19 sobre las empresas Colombianas. Obtenido de <https://confecamaras.org.co/phocadownload/2020/Encuesta%20Empresarial%20C%3%A1maras%20de%20Comercio%20-%20Emergencia%20Covid-19.pdf>

- Congreso de Colombia, 1982; 2018. (s.f.). Obtenido de <https://www.cvc.gov.co/sites/default/files/2018-10/Ley%202023%20de%201982-Ene-28.pdf>
- Congreso de Colombia, 1999. (s.f.). Obtenido de <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=4132>
- Congreso de Colombia, 2012. (s.f.). Obtenido de <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=53646>
- Cruz, A. R. (2020). SISTEMA DE CONTROL DE INVENTARIOS Y FACTURACIÓN”CASO: “EMPRESA SHARPPPOINT. Obtenido de <https://repositorio.upea.bo/jspui/bitstream/123456789/159/1/P.D.G-%20ALVARO%20ROJAS%20CRUZ.pdf>
- DANE. (28 de 02 de 2023). Boletín Técnico: Encuesta de micronegocios (EMICRON). Obtenido de <https://www.dane.gov.co/files/investigaciones/boletines/ech/micro/bol-micronegocios-IV-2022.pdf>
- Dávila, N. A., & abril, Á. S. (2020). DESARROLLO DE UN PROTOTIPO DE SISTEMA DE FACTURACIÓN E INVENTARIOS PARA TIENDAS MINORISTAS DE ROPA QUE MEDIANTE REDES NEURONALES MEJORE EL CONTROL DE INVENTARIOS. Obtenido de https://repository.unab.edu.co/bitstream/handle/20.500.12749/12050/2020_Tesis_Nicolas_Andres_Bernal_Davila.pdf?sequence=1
- DÁVILA, N. A., & ABRIL, Á. S. (2020). DESARROLLO DE UN PROTOTIPO DE SISTEMA DE FACTURACIÓN E INVENTARIOS PARA TIENDAS MINORISTAS DE ROPA QUE MEDIANTE REDES NEURONALES MEJORE EL CONTROL DE INVENTARIOS. Obtenido de https://repository.unab.edu.co/bitstream/handle/20.500.12749/12050/2020_Tesis_Nicolas_Andres_Bernal_Davila.pdf?sequence=1
- Decreto 2420 de 2015. (s.f.). Obtenido de <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=76745>
- Delgado, A. L. (2022). APLICACIÓN INFORMÁTICA PARA LA GESTIÓN DE PROCESOS DE FACTURACIÓN E INVENTARIO PARA EL HOSTAL EL CAFÉ DE JC DE LA CIUDAD JIPIJAPA. Obtenido de https://repositorio.unesum.edu.ec/bitstream/53000/3569/1/ZAMBRANO%20DE LGADO%20ANA_Pdf.pdf
- DELGADO, A. L. (2022). APLICACIÓN INFORMÁTICA PARA LA GESTIÓN DE PROCESOS DE FACTURACIÓN E INVENTARIO PARA EL HOSTAL EL CAFÉ DE JC DE LA CIUDAD JIPIJAPA. Obtenido de https://repositorio.unesum.edu.ec/bitstream/53000/3569/1/ZAMBRANO%20DE LGADO%20ANA_Pdf.pdf
- Delgado, D. o. (2017). *OpenWebinars*. Obtenido de <https://openwebinars.net/blog/que-es-c-introduccion/>
- Digitalgrow. (2023). Obtenido de <https://digitalgrow.es/manual-de-sql-server-en-espanol-en-formato-pdf-recursos-y-descarga-gratuita/>
- DNP. (2020). Guía para la elaboración y preparacion de casos de usos. Obtenido de <https://colaboracion.dnp.gov.co/CDTI/Oficina%20Informatica/Sistemas%20de%20informaci%C3%B3n/Gu%C3%ADas%20Formatos%20Plantillas/Gu%C3%A Da%20para%20la%20Elaboraci%C3%B3n%20y%20Presentaci%C3%B3n%20de%20Casos%20de%20Uso.pdf>
- Dobaño, R. (11 de enero de 2024). Facturación de una empresa: qué es y su importancia. Obtenido de <https://getquipu.com/blog/facturacion-de-una-empresa/#que-es>
- DocuSign. (14 de septiembre de 2022). ¿Qué es un proceso? Conozca los tipos y ejemplos. Obtenido de <https://www.docusign.com/es-mx/blog/que-es-proceso>

domainlogi. (2024). Clean Architecture: qué es, importancia y beneficios para tu empresa. Obtenido de <https://domainlogic.io/clean-architecture-que-es-importancia-y-beneficios-para-tu-empresa/>

Equipo editoria. (27 de agosto de 2020). Información. Obtenido de <https://concepto.de/informacion/>

Faqsensei. (2024). *Faqsensei*. Obtenido de <https://faqsensei.com/que-es-y-para-que-sirve-xampp>

Franzolini, D. (27 de Enero de 2023). Los 7 tipos de aplicaciones web que existen (y ejemplos). *HubSpot*. Obtenido de <https://blog.hubspot.es/website/tipos-aplicaciones-web>

Gasbarrino, S. (25 de abril de 2023). Qué es un inventario: concepto, tipos y ejemplos. Obtenido de <https://blog.hubspot.es/sales/que-es-inventario#que-es>

González, K. B., Barreto, V. P., Wah, J. P., & Lombardo, M. V. (2020). Implementación de las TIC'S en la gestión de inventario dentro de la cadena de suministro. Obtenido de <https://core.ac.uk/download/pdf/234019677.pdf>

Graus. (14 de noviembre de 2023). Qué es un sistema. Obtenido de <https://www.significados.com/sistema/>

Historiadelapempresa. (2024). ¿Qué es un caso de uso? (Con elementos clave y un ejemplo). Obtenido de <https://historiadelapempresa.com/caso-de-uso>

Imperia. (2025). El punto de Reorden (ROP) o cómo evitar la falta de stock. Obtenido de <https://imperiascm.com/es-es/blog/punto-de-reorden-rop>

Junquera, P. M. (2021). Diseo y desarrollo de una aplicacion de escritorio dedicada a la compocision fotografica. Obtenido de <https://uvadoc.uva.es/bitstream/handle/10324/15170/TFG-G1646.pdf?sequence=1&isAllowed=y>

La Ley 1314 de 2009. (s.f.). Obtenido de <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=36833>

La Ley 23 de 1982. (s.f.). Obtenido de <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=3431>

La Ley 527 de 1999. (s.f.). Obtenido de <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=4276>

Ley 1480 de 2011. (2024). Obtenido de http://secretariassenado.gov.co/senado/basedoc/ley_1480_2011.html

Ley 1915 de 2018. (s.f.). Obtenido de <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=87419>

Ley 23 de 1982 propiedad intelectual. (2024). Obtenido de <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=3431>

Ley 590 de 2000. (s.f.). Obtenido de <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=12672>

Ley 905 de 2004. (s.f.). Obtenido de <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=14501>

LEY ESTATUTARIA 1581 DE 2012. (2022). Obtenido de <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=49981>

Lidefer. (31 de 8 de 2023). Obtenido de <https://www.lifeder.com/sistema-de-inventarios/>

LOGGRO, L. S. (2024). *LOGGRO*. Obtenido de <https://loggro.com/blog/articulo/9-problemas-que-evitara-si-usa-un-software-de-inventarios-en-su-negocio/>

Luz, O. F., & Valentina, G. Y. (17 de 01 de 2023). Sistema de Control de la Producción Para el Mejoramiento de los Sistemas Productivos del Taller Mecánico Agroindustria. Obtenido de <https://repositorio.udes.edu.co/server/api/core/bitstreams/0898fb26-eb5f-4a90-a004-b557d56c41e5/content>

- Mancuzo, G. (08 de Agosto de 2020). Metodología XP: La Mejor Vía para el Desarrollo de Software. *ComparaSoftware*. Obtenido de <https://blog.comparasoftware.com/metodologia-xp/>
- Marques, K. A. (s.f.). Fundamentos de aplicaciones web. MuyTecnologicos. (2023). Investigación tecnológica. Obtenido de <https://muytecnologicos.com/diccionario-tecnologico/investigacion-tecnologica>
- Northware. (26 de Mayo de 2022). Requerimientos en el desarrollo de software y aplicaciones. Obtenido de <https://www.northware.mx/blog/requerimientos-en-el-desarrollo-de-software-y-aplicaciones/>
- Oracle. (s.f.). *¿Qué es una base de datos?* Recuperado el 2024, de <https://www.oracle.com/co/database/what-is-database/>
- Palacio, A. V. (1988). *Administración Política*. 265.
- Parada, M. (2019). *Openwebinars*. Obtenido de <https://openwebinars.net/blog/que-es-sql-server/>
- Peña, M. L., Machado, H. R., & Dávila, J. A. (2020). *Sistema web para el control de inventario y facturación de la Distribuidora Villareyna Utilizando la metodología SCRUM, en la ciudad de Estelí, segundo semestre 2019*. Estelí. Obtenido de <https://repositorio.unan.edu.ni/id/eprint/13519/1/20083.pdf>
- Pereira, A. D. (2019). DESARROLLO DE SOFTWARE WEB PARA LA GESTIÓN DE INVENTARIO, CON. Obtenido de <https://repository.udistrital.edu.co/bitstream/handle/11349/24811/DiazPereiraAlejandro2020.pdf?sequence=3>
- Pérez, R., Piguave, D., & Lozada, F. (22 de Julio de 2022). Los micromercados y su efecto en la comercialización de productos de la canasta. Obtenido de <https://dialnet.unirioja.es/descarga/articulo/9042913.pdf>
- PICÓN, S. T. (2019). DISEÑO DE UN PLAN DE CONTINGENCIA INFORMÁTICO BASADO EN LAS NORMAS ISO/IEC 22301 E ISO/IEC 27031 PARA LA FERRETERÍA CESAR S.A.S EN LA CIUDAD DE VALLEDUPAR. Obtenido de <https://repository.unad.edu.co/bitstream/handle/10596/21434/1065563682.pdf?sequence=1&isAllowed=y>
- Pinto, E. (24 de Julio de 2020). *Metodología XP o programación extrema de software*. Obtenido de Universo del Marketing: <https://eluniversodelmarketing.com/c-metodologias/metodologia-xp/>
- Ramírez, D. A. (17 de 02 de 2024). Transformación Tecnológica en el Modelo de Gestión de Inventarios en las Mipymes, Revisión Bibliográfica. Obtenido de <https://ciencialatina.org/index.php/cienciala/article/view/9701>
- Refactoring_Guru. (2024). *¿Qué es un patrón de diseño?* Obtenido de <https://refactoring.guru/es/design-patterns/what-is-pattern>
- Riaño, M. L., Alonso, V. D., & Sinvesol, K. L. (2023). APLICACIÓN WEB PARA EL CONTROL Y GESTIÓN DE INVENTARIO DE MERCANCÍAS Y HERRAMIENTAS DE SOLINCORP S.A.S. - SINVESOL. Obtenido de <https://repository.libertadores.edu.co/server/api/core/bitstreams/3da15efd-d7f8-4c1c-80f8-0e4c772bcd7f/content>
- RIAÑO, M. L., ALONSO, V. D., & SINVESOL, K. L. (2023). APLICACIÓN WEB PARA EL CONTROL Y GESTIÓN DE INVENTARIO DE MERCANCÍAS Y HERRAMIENTAS DE SOLINCORP S.A.S. - SINVESOL. Obtenido de <https://repository.libertadores.edu.co/server/api/core/bitstreams/3da15efd-d7f8-4c1c-80f8-0e4c772bcd7f/content>
- Ribeter. (2024). *Qué es el método LIFO y ejemplos*. Obtenido de <https://ribeter.org/que-es-el-metodo-lifo-y-ejemplos/>
- Roblenado, A. (2019). *OpenWebinars*. Obtenido de <https://openwebinars.net/blog/que-es-mysql/>

- Rodriguez, A. (12 de 11 de 2022). *DIGITALWEEK*. Obtenido de El impacto de la tecnología en el desarrollo humano: <https://digitalweek.es/el-impacto-de-la-tecnologia-en-el-desarrollo-humano/>
- Rodríguez, L. d., Carmona, E. A., & Arredondo, L. R. (09 de 05 de 2022). Factores que influyen en la Gestión de la Innovación en empresas financiadas por el Gobierno Federal. Obtenido de https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S2448-63882022000100037
- Sánchez, O. J., & Galeano, L. G. (18 de 06 de 2021). DESARROLLO DE UN SISTEMA DE GESTIÓN DE INVENTARIO PARA EL CONTROL DE PRODUCTOS EN LAS FASES DE SOLICITUD, APRESTAMIENTO, ENVÍO, RECEPCIÓN Y RETORNO. Obtenido de <https://repository.usta.edu.co/bitstream/handle/11634/42792/2021luisgomez.pdf?sequence=1>
- SCM, & Arilla, S. (2021). FEFO, FIFO, LIFO. Tres técnicas gestión almacén. Obtenido de <https://www.scmlogistica.es/fefo-fifo-lifo-gestion-almacen-control-stock/>
- sunrise, D. (2025). *Diccionario de datos*. Obtenido de <https://www.datasunrise.com/es/centro-de-conocimiento/diccionario-de-datos/>
- Sydle. (08 de Enero de 2023). Extreme Programming: ¿qué es y cómo funciona? Obtenido de <https://www.sydle.com/es/blog/extreme-programming-602ee205da4d096809438c9c>
- TechEdu. (2024). SOLID principios de diseño de software. Obtenido de <https://techlib.net/techedu/solid-principios-de-diseno-de-software/>
- TecnoEmpresa. (2025). Qué es una aplicación de escritorio: definición y beneficios. Obtenido de <https://tecnoempresa.mx/que-es-una-aplicacion-de-escritorio-definicion-y-beneficios/>
- Trejo, F. R. (2021). SISTEMATIZACIÓN DE PROCESOS DE COMERCIALIZACIÓN DE PRODUCTOS LÁCTEOS, FACTURACIÓN E INVENTARIO EN EL CENTRO AGRÍCOLA DEL CANTÓN TULCÁN. Obtenido de <http://repositorio.upec.edu.ec/bitstream/123456789/1290/1/044-%20VELA%20FABRICIO-YAZAN%20YAMIL%20c3%89.pdf>
- Universitat Carlemany. (21 de Marzo de 2024). Metodologías de desarrollo de software. Obtenido de <https://www.universitatcarlemany.com/actualidad/blog/metodologias-de-desarrollo-de-software/#:~:text=La%20metodolog%C3%ADa%20de%20desarrollo%20de,por%20una%20cuesti%C3%B3n%20de%20organizaci%C3%B3n.>
- Valledupar, A. d. (16 de 05 de 2024). *Alcaldía de Valledupar*. Obtenido de <https://valledupar-cesar.gov.co/MiMunicipio/Paginas/Informacion-del-Municipio.aspx>
- Varela, M. C. (2023). *Introduccion a la tecnologia .NET*. Obtenido de <https://repositorio.utn.edu.ec/bitstream/123456789/1116/1/04%20ISC%20064%20CAPITULO%20I.pdf>
- Victoria, Z. G. (2020). IMPLEMENTACION DE UN SOFTWARE DE GESTION DE LAS OPERACIONES INTERNAS Y EXTERNAS DE LA FERRETERIA "CARDENAS" DE LA CIUDAD MILAGRO. Obtenido de https://cia.uagraria.edu.ec/Archivos/ZAMORA%20GALLEGOS%20MAYERLY%20VICTORIA_compressed.pdf
- VICTORIA, Z. G. (2020). IMPLEMENTACION DE UN SOFTWARE DE GESTION DE LAS OPERACIONES INTERNAS Y EXTERNAS DE LA FERRETERIA "CARDENAS" DE LA CIUDAD MILAGRO. Obtenido de https://cia.uagraria.edu.ec/Archivos/ZAMORA%20GALLEGOS%20MAYERLY%20VICTORIA_compressed.pdf

Vivas Riaño, M. L., Rodríguez Alonso, V. D., & Barbosa Moreno, K. L. (2023).
APLICACIÓN WEB PARA EL CONTROL Y GESTIÓN DE INVENTARIO DE
MERCANCÍAS Y HERRAMIENTAS DE SOLINCORP S.A.S. - SINVESOL.
Obtenido de
<https://repository.libertadores.edu.co/server/api/core/bitstreams/3da15efd-d7f8-4c1c-80f8-0e4c772bcd7f/content>

Westreicher, G. (19 de Febrero de 2024). Gestión: Qué es, pasos a seguir y tipos.
Obtenido de <https://economipedia.com/definiciones/gestion.html>

8. ANEXOS

Anexo A: Entrevista a la representante legal del Micro mercado provisiones Dylan, de Valledupar, Cesar, Mayra Alejandra Chinchilla Orozco.

¿Cómo lleva actualmente el control del inventario y la facturación en su negocio?

Respuestas: En la actualidad, el manejo del inventario y la facturación se lleva a cabo de forma manual, utilizando libretas y anotaciones escritas. Cada vez que se concreta una venta, se registran los productos vendidos y el total correspondiente. Respecto al inventario, se efectúa un conteo físico de los productos disponibles en el almacén y se documenta en una libreta. Sin embargo, este método presenta varias limitaciones, ya que es sencillo cometer errores en los cálculos, olvidar, registrar algunas ventas o extraviar información clave. Además, cuando hay un gran flujo de clientes, el proceso se torna más lento, lo que puede ocasionar retrasos y perjudicar la experiencia del comprador.

¿Cuáles son los principales problemas que enfrenta en la gestión de su micro mercado debido a la falta de un sistema automatizado?

Respuestas: Uno de los principales inconvenientes es la falta de exactitud en el manejo del inventario. Cuando se realiza manualmente, es frecuente que existan discrepancias entre los productos que tenemos registrados y los que realmente hay en existencia. Esto nos ha llevado a quedarnos sin ciertos productos en momentos cruciales o a adquirir de más algunos artículos que no se venden rápidamente, lo que implica una pérdida económica. Otro problema significativo es la facturación, dado que, al ser manual, el proceso resulta más lento y es más susceptible a errores en los precios o en los cálculos, lo que puede originar descontento en los clientes. Esto también impacta la gestión del negocio, ya que sin datos organizados y en tiempo real, se dificulta la toma de decisiones estratégicas sobre compras, promociones o ajustes en los precios.

¿Cómo afecta la gestión manual del inventario y facturación a la rentabilidad de su negocio?

Respuestas: La gestión manual impacta la rentabilidad de diversas maneras. En primer lugar, cuando se pierden productos debido a la falta de control, estamos perdiendo dinero sin percatarnos. También pasa que ciertos productos permanecen almacenados por más tiempo del necesario, lo que genera pérdidas, ya que ocupan espacio y pueden caducar. Además, al no disponer de un sistema preciso, a menudo no sabemos exactamente cuáles son los productos que más se venden o los de menor rotación, lo que complica la planificación de compras y estrategias de venta. Por otro lado, el tiempo que se gasta en realizar registros manuales también tiene un costo, puesto que

podríamos utilizarlo para mejorar el servicio, atender de mejor forma a los clientes o explorar nuevas oportunidades de negocio.

¿Ha considerado antes implementar un software para mejorar estos procesos?
¿Por qué no lo ha hecho hasta ahora?

Respuestas: Sí, he pensado en varias ocasiones en la posibilidad de implementar un software, pues sé que muchas empresas han mejorado sus procesos gracias a la tecnología. Sin embargo, hasta el momento no lo he llevado a cabo principalmente por dos motivos: primero, el desconocimiento acerca de qué tipo de software se ajusta mejor a mi negocio y cómo podría implementarlo sin afectar la operatividad diaria; segundo, el temor o la dificultad para aprender a utilizarlo y que implique una inversión muy elevada. Además, tras años utilizando el método manual, ha sido complicado romper con esa costumbre, aunque cada vez es más claro que es necesario un sistema más moderno y eficiente.

¿Qué funcionalidades considera indispensables en un software de gestión para su negocio?

Respuestas: Es crucial que el software nos permita conocer la cantidad de productos disponibles y que emita alertas cuando el stock sea bajo. Además, sería vital que tenga un sistema de facturación ágil y eficiente, que registre automáticamente cada venta y calcule el total sin cometer errores, un plus que me gustaría que tuviera sería que al momento de emitir la factura no se entregara de manera impresa en papel, sino que se enviara vía correo electrónico o WhatsApp, de esta manera estaríamos contribuyendo al medio ambiente con un pequeño granito de arena.

¿Cómo cree que un software de gestión impactaría en la organización y productividad de su negocio?

Respuestas: Un software de gestión revolucionaría por completo la manera en que operamos el negocio. Primero que todo, disminuiría los errores humanos, puesto que toda la información de ventas y de inventario estaría centralizada y automatizada. Esto nos permitiría economizar tiempo en los trámites administrativos, lo que se traduce en una mayor eficacia operativa. Asimismo, enriquecería la experiencia del cliente, ya que la facturación sería más rápida y exacta. Además, el acceso a datos en tiempo real nos facultaría para tomar mejores decisiones estratégicas, como ajustar precios, planear promociones y controlar con mayor eficacia las compras.

¿Ha tenido problemas con clientes por errores en la facturación o falta de productos en stock?

Respuestas: Sí, esto ha sucedido en múltiples ocasiones. Algunas veces, los clientes llegan en busca de un producto que creemos tener en stock, pero al verificar, nos damos cuenta de que ya no está disponible. Esto genera frustración en los clientes y, en algunas ocasiones, se marchan sin hacer ninguna compra. También hemos enfrentado errores en la facturación, como registrar incorrectamente los precios o calcular mal el total de la compra, lo que genera confusiones y desconfianza. Un sistema automatizado podría ayudar a evitar estos inconvenientes, garantizando que la información esté siempre actualizada y disminuyendo los errores humanos en las transacciones.

¿Qué aspectos considera más importantes al momento de adoptar una nueva tecnología en su negocio?

Respuestas: Para mí, lo primordial es que la tecnología realmente haga más sencilla la labor y no la vuelva complicada. Debe ser intuitiva y ajustarse a las necesidades particulares de mi micro mercado. Además, es crucial que sea fiable y segura, dado que manejará información clave sobre las ventas y el inventario. Por último, me gustaría que el sistema sea flexible para acomodarse al crecimiento del negocio, sin que se quede desactualizado pronto.

¿Qué expectativas tiene con la implementación de este software en su negocio?

Respuestas: Confío en que el software contribuirá a optimizar la organización y eficiencia del negocio, posibilitando un manejo más exacto del inventario y una facturación más ágil y fiable. También espero que facilite la toma de decisiones, proporcionando informes detallados sobre las ventas y el rendimiento del negocio. En términos generales, se espera que este sistema me asista en la reducción de tiempos, minimización de errores y mejora de la rentabilidad del negocio. Además, considero que, con una herramienta tecnológica adecuadamente implementada, podremos brindar un servicio de mayor calidad a los clientes y mantenernos en una posición competitiva dentro del mercado.

Anexo B: Cronograma de actividades

	OBJETIVOS ESPECIFICOS	ACTIVIDADES	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana	Semana				
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	
FASE DE PLANIFICACION	Establecer los requisitos funcionales y no funcionales para diseñar los módulos de gestión, inventario y facturación para el micro mercado "Provisiones Dylan".	Visitar el micro mercado "Provisiones Dylan" para observar cómo se gestionan los procesos de inventario y facturaciones.																															
		Realizar la entrevista con el representante legal del micro mercado como técnica de recolección de necesidades y expectativas del cliente respecto al software.																															
		Elaborar las historias de usuario que describan las funcionalidades con las que va a interactuar el usuario.																															
FASE DE DISEÑO	Diseñar el software que integre las funcionalidades que son requeridas para la gestión, inventario y facturación del micro mercado.	Diseñar las vistas del aplicativo de escritorio para determinar los colores, fuentes y que el diseño sea agradable.																															
		Creación de la base de datos																															
FASE DE COMERCIALIZACION	Construir el código del software aplicando prácticas de programación y refactorización continua para garantizar un diseño simple y eficiente.	Desarrollar el código de programación que cumple con las historias de usuario																															
FASE DE PRUEBAS Y LANZAMIENTO	Realizar una prueba de aceptación de usuario que permita verificar la funcionalidad del software con los requisitos y funcionalidades establecidas en el diseño.	Realizar las pruebas unitarias continuas para detectar y corregir errores durante el desarrollo de cada historia																															
		Colocar en funcionamiento el software en el micro mercado Provisiones Dylan para la prueba piloto																															
		Ejecución de pruebas de aceptación: permitir que el cliente valide funcionalidades clave.																															

Figura 23. Cronograma de actividades
Nota. Los autores

Anexo C: Acta de aceptación de historias de usuario

ACTA DE ACEPTACIÓN DE HISTORIAS DE USUARIO

Software de Gestión de Inventario y Facturación

Empresa: Provisiones Dylan

Gerente: Mayra Alejandra Chinchilla Orozco

Teléfono: 3102348848

Dirección: Torres Nando Marín. Manzana 15, bloque R, apartamento 104

NIT o RUT: 1067035441-0, matrícula: 141745

Fecha: 18 de diciembre de 2024

Objetivo del Acta:

Este documento tiene como objetivo formalizar la aceptación de las historias de usuario definidas en el proyecto Software de Gestión, Inventario y Facturación para el micro mercado Provisiones Dylan. Las historias de usuario detallan las funcionalidades que serán desarrolladas y entregadas, y representan el alcance de las tareas acordadas entre Provisiones Dylan y el equipo de desarrollo.

Descripción de las Historias de Usuario aceptadas:

1. HU01 - Gestión de Usuarios y Roles

Descripción: Como administrador, quiero crear, editar y filtrar usuarios con diferentes roles (Administrador, Vendedor, Supervisor) para controlar el acceso al sistema según sus permisos.

Criterios de aceptación:

Debe permitir la creación, edición y filtrado de usuarios por nombre o rol.
Asignar y modificar roles específicos con permisos definidos.
Restringir funcionalidades según el rol del usuario.

2. HU02 - Gestión de Proveedores

Descripción: Como administrador, quiero registrar, actualizar y filtrar proveedores para vincularlos a los productos del inventario.

Criterios de aceptación:

- Permitir registrar nuevos proveedores con datos de contacto.
- Permitir editar la información de un proveedor.
- Filtrar proveedores por nombre o categoría.

3. HU03 - Gestión de Productos

Descripción: Como administrador, quiero crear, editar y filtrar productos con su proveedor, categoría, precio y stock para tener control de los artículos disponibles.

Criterios de aceptación:

- Crear productos con nombre, descripción, proveedor, precio, stock inicial.
- Editar productos existentes.
- Filtrar productos por nombre, categoría o proveedor.

3. HU04 - Control de Inventario y Stock

Descripción: Como administrador, quiero recibir notificaciones cuando un producto esté por agotarse para realizar reposiciones a tiempo.

Criterios de aceptación:

- El sistema debe generar alertas cuando el stock esté por debajo del umbral definido.
- Posibilidad de configurar el umbral por producto.
- Permitir filtrar productos con stock crítico.

4. HU05 - Gestión de Clientes

Descripción: Como vendedor, quiero registrar, editar y buscar clientes para asociarlos a ventas y créditos.

Criterios de aceptación:

- Crear clientes con nombre, cédula, contacto.
- Editar datos de clientes.
- Filtrar por nombre o número de identificación.

6. HU06 - Registro de Ventas

Descripción: Como vendedor, quiero registrar, editar y buscar ventas realizadas para llevar el control de ingresos.

Criterios de aceptación:

- Registrar productos vendidos, cantidades, método de pago y cliente.
- Calcular automáticamente el total.

- Filtrar ventas por fecha, cliente o método de pago.

7. HU07 - Gestión de Créditos

Descripción: Como vendedor, quiero gestionar créditos asociados a clientes para llevar el control de saldos pendientes.

Criterios de aceptación:

- Registrar ventas a crédito con plazo y monto.
- Consultar historial de créditos por cliente.
- Filtrar créditos por cliente o estado (pendiente/pagado).

8. HU08 - Reportes de Ventas y Créditos:

Descripción: Como administrador, quiero generar reportes filtrados de ventas y créditos para analizar el rendimiento del negocio.

Criterios de aceptación:

Generar reportes en PDF/Excel con opción de filtrar por fecha, cliente y producto.

- Visualizar totales por método de pago, producto y estado de crédito.
- Incluir gráficos con tendencias de ventas y deudas.

Confirmación de aceptación:

El cliente ha revisado y aceptado las historias de usuario descritas en este documento. Los criterios de aceptación y los puntos de historia correspondientes han sido comprendidos y aprobados, por lo que el equipo de desarrollo puede proceder con la implementación según lo acordado.

Firmas:

Por parte de Provisiones Dylan:

Firma: Mayra Chinchilla O.

Nombre: Mayra Alejandra Chinchilla Orozco

Cargo: Gerente

Por parte del Equipo de Desarrollo:

Firma: Jhon Jaider Manosalva

Nombre: Jaider Manosalva Rangel

Cargo: Desarrollador

Firma: Santiago Pupo C.

Nombre: Santiago Pupo Cogollo

Cargo: Desarrollador

Esta acta de aceptación es vinculante para ambas partes y confirma que las historias de usuario han sido aprobadas por el cliente para su implementación en el proyecto.