

# UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI



## FACULTAD DE INDUSTRIAS AGROPECUARIAS Y CIENCIAS AMBIENTALES

### CARRERA DE COMPUTACIÓN

**Tema:** "Sistema computacional de gestión de inventarios para Minimarket"

Trabajo de Integración Curricular previo a la obtención del  
título de Ingeniero en Ciencias de la Computación

AUTOR: Salazar Burbano Israel Sebastián

TUTOR: Ing. Naranjo Cedeño Jeffery Alex MSc.

Tulcán, 2026.

## **CERTIFICADO DEL TUTOR**

Certifico que el estudiante(s) Salazar Burbano Israel Sebastián y con el número de cédula 0401952445 respectivamente ha desarrollado el Trabajo de Integración Curricular: "Sistema computacional de gestión de inventarios para Minimarkets"

Este trabajo se sujeta a las normas y metodología dispuesta en el Reglamento de la Unidad de Integración Curricular, Titulación e Incorporación de la UPEC, por lo tanto, autorizo la presentación de la sustentación para la calificación respectiva

---

Ing. Naranjo Cedeño Jeffery Alex MSc.

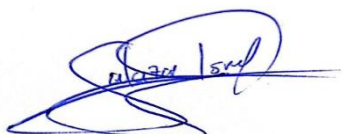
**TUTOR**

Tulcán, junio de 2026

## AUTORÍA DE TRABAJO

El presente Trabajo de Integración Curricular constituye un requisito previo para la obtención del título de Ingeniero en la Carrera de computación de la Facultad de Industrias Agropecuarias y Ciencias Ambientales

Yo, Salazar Burbano Israel Sebastián con cédula de identidad número 0401952445 respectivamente declaro que la investigación es absolutamente original, auténtica, personal y los resultados y conclusiones a los que he llegado son de mi absoluta responsabilidad.



---

Salazar Burbano Israel Sebastián

**AUTOR**

Tulcán, junio de 2026

## ACTA DE CESIÓN DE DERECHOS DEL TRABAJO DE INTEGRACIÓN CURRICULAR

Yo Salazar Burbano Israel Sebastián declaro ser autor de los criterios emitidos en el Trabajo de Integración Curricular: "Sistema computacional de gestión de inventario para Minimarkets" y eximo expresamente a la Universidad Politécnica Estatal del Carchi y a sus representantes de posibles reclamos o acciones legales.



---

Salazar Burbano Israel Sebastián

**AUTOR**

Tulcán, junio de 2026

## AGRADECIMIENTO

Agradezco a Dios, por ser mi guía constante, por brindarme la fortaleza, la sabiduría y la perseverancia necesarias para culminar esta importante etapa de mi vida académica.

A mis padres, quienes con su amor incondicional, esfuerzo y sacrificio han sido el pilar fundamental de mi formación.

A la Universidad Politécnica Estatal del Carchi, por abrirme sus puertas y permitirme formarme como profesional. A todos los docentes de la carrera, quienes con dedicación y vocación compartieron sus conocimientos, experiencias y valores, contribuyendo no solo a mi formación académica, sino también a mi crecimiento personal.

Especialmente agradezco a mi tutor el MSc. Jeffery Naranjo, a la MSc. Georgina Arcos y al ingeniero Yasmany Fernandez por su guía y apoyo durante mi proceso de titulación y en todo el transcurso de la carrera de inicio a fin demostrando ser unos excelentes profesionales educando tanto a nivel profesional como en valores.

## **DEDICATORIA**

A mis padres, que han estado presentes en cada paso de este camino y han sido mi mayor ejemplo de perseverancia y dedicación. Gracias por confiar en mí incluso en los momentos en que yo mismo dudaba, por su paciencia infinita y por enseñarme, con su ejemplo diario, el valor del esfuerzo y la responsabilidad. Todo lo que hoy logro lleva un poco de ustedes.

A mis amigos, por estar ahí en las largas jornadas de estudio, por los ánimos en los momentos difíciles y por hacer más llevadero este recorrido. Y a mis docentes, quienes con su conocimiento, paciencia y vocación supieron orientarme a lo largo de estos años. A todos ellos, gracias por formar parte de esta etapa que hoy culmino con mucha satisfacción.

## ÍNDICE

<b>RESUMEN</b> .....	17
<b>ABSTRACT</b> .....	18
<b>INTRODUCCIÓN</b> .....	19
<b>I. EL PROBLEMA</b> .....	20
<b>1.1. PLANTEAMIENTO DEL PROBLEMA</b> .....	20
<b>1.2. FORMULACIÓN DEL PROBLEMA</b> .....	21
<b>1.3. JUSTIFICACIÓN</b> .....	21
<b>1.4. OBJETIVOS Y PREGUNTAS DE INVESTIGACIÓN</b> .....	23
1.4.1. Objetivo General .....	23
1.4.2. Objetivos Específico .....	23
1.4.3. Preguntas de Investigación .....	23
<b>II. FUNDAMENTACIÓN TEÓRICA</b> .....	24
<b>2.1. ANTECEDENTES DE LA INVESTIGACIÓN</b> .....	24
<b>2.2. MARCO TEÓRICO</b> .....	26
2.2.1 Inventario.....	26
2.2.2 Tipos de Inventarios .....	26
2.2.3 Gestión de Inventarios.....	27
2.2.4 Registro de productos .....	28
2.2.5 Registro de Entradas y Salidas .....	28
2.2.6 Control de Stock y niveles mínimos .....	29
2.2.7 Clasificación de Inventarios: Método ABC .....	30
2.2.8 Sistemas de Control de Inventarios .....	30
2.2.9 Punto de Reorden y Stock de Seguridad .....	32
2.2.10 Indicadores de Gestión de Inventarios.....	32
2.2.11 Tecnologías Aplicadas a la Gestión de Inventarios.....	33

2.2.12 Auditoría de Inventarios .....	34
2.2.13 Kardex .....	35
2.2.14 Valoración del Inventario.....	36
2.2.15 Control de existencias .....	36
2.2.16 Gestión de proveedores .....	37
2.2.17 Facturación y su relación con la gestión de inventarios.....	37
2.2.18 Registro automático de ventas .....	37
2.2.19 Ingeniería de software.....	38
2.2.20 Arquitectura de software .....	38
2.2.21 Tipos de sistemas de gestión de inventarios.....	39
2.2.22 Metodología de Desarrollo .....	39
2.2.23 IDEs y herramientas de desarrollo.....	42
2.2.24 Lenguajes de programación .....	42
2.2.25 Frameworks.....	44
2.2.26 Base de datos .....	45
2.2.27 Seguridad de la información y control de accesos .....	46
2.2.28 Estándares de calidad. ....	47
<b>III. METODOLOGÍA .....</b>	<b>48</b>
<b>3.1. ENFOQUE METODOLÓGICO .....</b>	<b>48</b>
3.1.1. Enfoque .....	48
3.1.2 Enfoque Mixto .....	48
3.1.3. Tipo de Investigación.....	49
<b>3.2. IDEA A DEFENDER .....</b>	<b>50</b>
<b>3.3. DEFINICIÓN Y OPERACIONALIZACIÓN DE LAS VARIABLES.....</b>	<b>50</b>
<b>3.4. MÉTODOS UTILIZADOS .....</b>	<b>53</b>
3.4.1 Métodos.....	53
3.4.2 Técnicas.....	53
<b>3.5. ANÁLISIS ESTADÍSTICO .....</b>	<b>53</b>

<b>IV. RESULTADOS Y DISCUSIÓN</b> .....	54
<b>4.1. RESULTADOS</b> .....	54
4.1.1 Entrevista.....	54
4.1.2 Levantamiento de procesos.....	55
4.1.3 Identificación de problemas y necesidades.....	55
4.1.4 Requerimientos funcionales y no funcionales.....	55
4.1.5 Evaluación de lenguajes de programación .....	56
4.1.6 Evaluación de Frameworks.....	56
4.1.7 Evaluación de sistemas de base de datos.....	57
4.1.8 Evaluación de entornos de desarrollo (IDE) .....	57
4.1.9 Evaluación de metodologías de desarrollo .....	57
4.1.10 Resultados para el diseño del sistema .....	58
<b>4.2 PROPUESTA</b> .....	60
4.2.1. Factibilidad organizacional .....	60
4.2.2. Factibilidad técnica .....	61
4.2.3. Factibilidad Económica .....	62
4.2.4. Factibilidad Operativa .....	63
4.2.5 Metodología XP .....	67
4.2.6 Historias de usuario .....	68
4.2.7 Tareas de usuario .....	71
4.2.8 Requerimientos de software .....	75
4.2.9 Fase de Diseño .....	79
4.2.10 Diagramas de caso de uso.....	88
4.2.11 Fase de Codificación .....	90
4.2.12 Templates .....	96
<b>4.3. DISCUSIÓN</b> .....	101
<b>V. CONCLUSIONES Y RECOMENDACIONES</b> .....	104
<b>5.1. CONCLUSIONES</b> .....	104

<b>5.2. RECOMENDACIONES</b> .....	105
<b>VI. REFERENCIAS BIBLIOGRÁFICAS</b> .....	106
<b>VII. ANEXOS</b> .....	110

## ÍNDICE DE TABLAS

Tabla 1. Métodos de valoración de inventario .....	36
Tabla 2. Tabla comparativa metodologías ágiles.....	41
Tabla 3. IDES y herramientas de desarrollo .....	42
Tabla 4. Comparativa de lenguajes de programación .....	43
Tabla 5. Tabla comparativa de frameworks.....	45
Tabla 6. Tabla comparativa de bases de datos .....	45
Tabla 7. Operacionalización de variables (Variable dependiente) .....	51
Tabla 8. Operacionalización de variables (Variable independiente).....	52
Tabla 9. Recursos software .....	61
Tabla 10. Recursos de Hardware .....	62
Tabla 11. Costos de hardware .....	62
Tabla 12. Costos de Software.....	62
Tabla 13. Talento Humano.....	63
Tabla 14. Materiales de escritorio .....	63
Tabla 15. Resumen de costos.....	63
Tabla 16. Módulos del sistema .....	67
Tabla 17. Historia de usuario 1 .....	68
Tabla 18. Historia de usuario 2 .....	68
Tabla 19. Historia de usuario 3 .....	69
Tabla 20. Historia de usuario 4 .....	69
Tabla 21. Historia de usuario 5 .....	69
Tabla 22. Historia de usuario 6 .....	69
Tabla 23. Historia de usuario 7 .....	70
Tabla 24. Historia de usuario 8 .....	70
Tabla 25. Historia de usuario 9 .....	70
Tabla 26. Historia de usuario 10 .....	70
Tabla 27. Historia de usuario 11 .....	70
Tabla 28. Historia de usuario 12 .....	71
Tabla 29. Historia de usuario 13 .....	71
Tabla 30. Historia de usuario 14 .....	71
Tabla 31. Historia de usuario 15 .....	71
Tabla 32. Tarea de usuario 1.....	72

Tabla 33. Tarea de usuario 2.....	72
Tabla 34. Tarea de usuario 3.....	72
Tabla 35. Tarea de usuario 4.....	72
Tabla 36. Tarea de usuario 5.....	72
Tabla 37. Tarea de usuario 6.....	73
Tabla 38. Tarea de usuario 7.....	73
Tabla 39. Tarea de usuario 8.....	73
Tabla 40. Tarea de usuario 9.....	73
Tabla 41. Tarea de usuario 10.....	73
Tabla 42. Tarea de usuario 11.....	74
Tabla 43. Tarea de usuario 12.....	74
Tabla 44. Tarea de usuario 13.....	74
Tabla 45. Tarea de usuario 14.....	74
Tabla 46. Tarea de usuario 15.....	74
Tabla 47. Tarea de usuario 16.....	74
Tabla 48. Tarea de usuario 17.....	75
Tabla 49. Tarea de usuario 18.....	75
Tabla 50. Tarea de usuario 19.....	75
Tabla 51. Tarea de usuario 20.....	75
Tabla 52. Requerimientos Funcionales .....	75
Tabla 53. Requerimientos No Funcionales.....	76
Tabla 54. Tiempos estimados de tareas de usuario.....	77
Tabla 55. Plan de entrega del proyecto .....	78
Tabla 56. Tarjeta CRC Usuario/Rol.....	79
Tabla 57. Tarjeta CRC Producto.....	80
Tabla 58. Tarjeta CRC StockInicial.....	80
Tabla 59. Tarjeta CRC Proveedor.....	80
Tabla 60. Tarjeta CRC Compra .....	80
Tabla 61. Tarjeta CRC Venta .....	81
Tabla 62. Tarjeta CRC Comprobante .....	81
Tabla 63. Tarjeta CRC Kardex.....	81
Tabla 64. Tarjeta CRC Bitácora .....	81
Tabla 65. Tarjeta CRC AlertasMinimos .....	82
Tabla 66. Tarjeta CRC Precios .....	82

Tabla 67. Tarjeta CRC Cierre Caja .....	82
Tabla 68. Tarjeta CRC Movimiento interno.....	82
Tabla 69. Tarjeta CRC Reporte .....	83
Tabla 70. Tarjeta CRC Exportación .....	83
Tabla 71. Tarjeta CRC Anulación .....	83

## ÍNDICE DE FIGURAS

Figura 1. Ejemplo de Kardex.....	36
Figura 2. Diseño de la base de datos.....	83
Figura 3. Pantalla inicio de sesión al sistema .....	84
Figura 4. Pantalla principal, inicio, funciones principales y dashboard.....	84
Figura 5. Pantalla del panel de ventas.....	84
Figura 6. Pantalla del listado de productos .....	85
Figura 7. Pantalla creación de venta y procesamiento pagos facturación .....	85
Figura 8. Pantalla registro de proveedores .....	85
Figura 9. Pantalla de categorías de los productos .....	86
Figura 10. Pantalla de ubicaciones de los productos.....	86
Figura 12. Pantalla de gestión y creación de usuarios .....	87
Figura 15. Pantalla inventario, stocks y productos por caducar.....	88
Figura 16. Diagrama de caso de uso siendo administrador.....	88
Figura 17. Diagrama de caso de uso del Cajero .....	89
Figura 18. Diagrama de caso de uso del Bodeguero.....	89
Figura 19. Diagrama de caso de uso del Gerente .....	89
Figura 20. Diagrama de caso de uso de funciones que hace el sistema.....	90
Figura 21. Diagrama de caso de uso de funciones del sistema hacia el cliente.....	90
Figura 22. Código función de autenticación y seguridad .....	91
Figura 23. Código función ventas y control de stock .....	91
Figura 24. Código función generar factura .....	92
Figura 25. Código función registrar compras.....	93
Figura 26. Código función listar productos y consultar .....	93
Figura 27. Código función dashboard de movimientos .....	94
Figura 28. Código función login .....	94
Figura 29. Código función ventas.....	95
Figura 30. Código función para ajustar el stock y ver inventario actualizado .....	96
Figura 31. Template login.....	96
Figura 32. Template inicio del sistema y dashboard .....	97
Figura 33. Template realizar ventas y gestión .....	97
Figura 34. Template crear clientes y gestión.....	97
Figura 35. Template gestionar y crear nuevas compras.....	98

Figura 36. Template ingresar nuevos proveedores.....	98
Figura 37. Inventario completo con múltiples funciones .....	98
Figura 38. Template gestión de productos .....	99
Figura 39. Template gestionar ubicaciones .....	99
Figura 40. Template generación de reportes diarios y mensuales .....	99
Figura 41. Template gestión de usuarios y sus roles .....	100
Figura 42. Template auditoría del sistema.....	100
Figura 43. Template recuperar contraseña de acceso.....	101

## ÍNDICE DE ANEXOS

Anexo 1. Acta de la sustentación de Predefensa del TIC .....	110
Anexo 2. Certificado del abstract por parte de idiomas .....	110
Anexo 3. Informe de Turniting.....	113
Anexo 4. Ficha de observación para levantamiento de procesos actuales .....	114
Anexo 5. Ficha de identificación de problemas y necesidades .....	115
Anexo 6. Checklist de requerimientos funcionales y no funcionales.....	116
Anexo 7. Comparativa de lenguajes de programación (matriz de evaluación). ....	116
Anexo 8. Comparativa de Frameworks (matriz de evaluación).....	117
Anexo 9: Sistemas de base de datos (matriz de evaluación).....	117
Anexo 10: Comparativa IDES (matriz de evaluación).....	118
Anexo 11. Metodologías de desarrollo (Fichas técnicas).....	119
Anexo 12. Arquitectura del sistema (Diagrama de 3 capas).....	119
Anexo 13. Entidades de la base de datos .....	120
Anexo 14. Normalización de la base de datos hasta 3FN.....	121
Anexos 15, 16 y 17 .....	121
Anexo 18. Entrevista al dueño del negocio .....	122

## RESUMEN

El Minimarket Otavalo, ubicado en la ciudad de Mira, provincia del Carchi, es uno de esos casos en los que el uso de registros manuales y hojas de Excel ha generado problemas constantes, como errores en los datos, pérdida de productos por vencimiento y decisiones basadas en información desactualizada. Este trabajo de investigación nace de una problemática real que enfrentan muchos pequeños negocios en Ecuador: la dificultad para controlar sus productos de forma ordenada y confiable. El objetivo principal de la investigación fue desarrollar un sistema computacional de gestión de inventarios para el Minimarket "Otavalo" en el cantón Mira en la Provincia del Carchi. Para ello, se revisaron los procesos actuales del negocio utilizando técnicas como la observación directa y el análisis de documentos comerciales. Esto permitió identificar exactamente dónde estaban las fallas y qué necesitaba el negocio para mejorar. Luego, se realizó una evaluación técnica de diversas metodologías ágiles y herramientas de desarrollo existentes. Se compararon lenguajes de programación, frameworks y bases de datos para elegir la combinación más adecuada. Como resultado, se seleccionó XP (Xtreme programming) como metodología ágil de ingeniería de software, Angular para el frontend, Node.js con Express para el backend y MySQL como base de datos, todas herramientas modernas, gratuitas y con amplia documentación. El sistema propuesto incluye módulos para gestionar productos, proveedores, compras, ventas, Kardex, alertas de stock y control de usuarios. Los resultados demuestran que es posible diseñar soluciones tecnológicas adaptadas a las necesidades específicas de los pequeños negocios, siempre que se realice un análisis previo de los procesos y se seleccionen las herramientas adecuadas. Este diseño queda disponible para una futura implementación que beneficie directamente al Minimarket Otavalo.

**Palabras clave:** gestión de inventarios, sistema computacional, minimarket, metodología XP, Angular, Node.js, MySQL.

## ABSTRACT

The Otavalo Minimarket, located in the city of Mira, Carchi Province, is one of those cases where the use of manual records and Excel spreadsheets has led to constant problems, such as data errors, loss of products due to expiration, and decisions based on outdated information. This research project stems from a real problem faced by many small businesses in Ecuador: the difficulty in managing their products in an organized and reliable manner. The main objective of the research was to develop a computerized inventory management system for the "Otavalo" Minimarket in Mira canton of Carchi province. For this purpose, the business's current processes were reviewed using techniques such as direct observation and the analysis of commercial documents. This allowed us to identify exactly where the flaws were and what the business needed to improve. Next, a technical evaluation was conducted of various existing agile methodologies and development tools. Programming languages, frameworks, and databases were compared to select the most appropriate combination. As a result, XP (Extreme Programming) was selected as the agile software engineering methodology, Angular for the front end, Node.js with Express for the back end, and MySQL as the database—all modern, free tools with extensive documentation. The proposed system includes modules for managing products, suppliers, purchases, sales, Kardex, stock alerts, and user control. The results demonstrate that it is possible to design technological solutions tailored to the specific needs of small businesses, provided that a preliminary analysis of the processes is conducted and the appropriate tools are selected. This design is available for future implementation that will directly benefit the Otavalo Minimarket.

**Keywords:** Inventory management, computer system, minimarket, XP methodology, Angular, Node.js, MySQL.

## INTRODUCCIÓN

En muchos Minimarkets, todavía utilizan hojas sueltas o archivos de Excel para controlar su inventario. Por eso, las compras y ventas se registran con retraso, lo que puede generar diferencias entre el stock real y el registrado en el sistema, y tomar decisiones importantes se vuelve más tardío. Además, cuando falta un producto, el cliente se va y se pierden ventas para el negocio. Esta situación se repite día tras día y afecta no solo a la caja, sino también al control de costos y a la organización del trabajo.

Frente a esta situación, esta investigación propone diseñar un sistema informático simple y ordenado para gestionar inventarios y ventas en un minimarket. La idea es que cada entrada, salida y ajuste quede registrado con el responsable y la fecha, y que el stock se actualice en el momento de cada operación. Así, el negocio podrá saber fácilmente qué productos están disponibles, cuáles hay que reponer y cuánto se ha vendido, sin depender de cálculos manuales.

Para desarrollar esta propuesta, primero se describió la situación actual. Se revisaron los documentos que ya utiliza el negocio, como facturas, listados y planillas, y también se observaron los procesos con sus fichas técnicas y matrices de evaluación. Con esta información, se detectaron los puntos más críticos: registros que llegan tarde, falta de un Kardex claro, precios y unidades poco coherentes, y reportes que no ayudan a tomar decisiones rápidamente. Luego, con toda esta base, se definieron los requisitos y se diseñó el sistema en módulos: catálogo de productos y precios, compras y proveedores, ventas con comprobante, Kardex y movimientos, alertas por niveles mínimos, caja y arqueo, reportes y usuarios con auditoría.

También, se eligió una arquitectura web adecuada para el contexto: interfaz en Angular, lógica en Node y Express, y base de datos en MySQL. Esta combinación permite ampliar el sistema poco a poco, mantenerlo rápido y asegurar que todo quede bien registrado. El diseño incluye el modelo de datos, las historias de usuario y los diagramas necesarios para entender cómo se conectan todas las partes.

## I. EL PROBLEMA

### 1.1. PLANTEAMIENTO DEL PROBLEMA

Actualmente, administrar inventarios representa uno de los mayores desafíos para empresas de todo el mundo, independientemente de su tamaño. Las grandes corporaciones han implementado sistemas computacionales avanzados que les ayudan a optimizar procesos, reducir errores humanos y tomar decisiones basadas en datos precisos (Mora García, 2016). Sin embargo, las pymes, particularmente en países en desarrollo, todavía enfrentan obstáculos importantes para adoptar estas soluciones debido a restricciones económicas, falta de capacitación y una dependencia de métodos tradicionales (CEPAL, 2021).

En América Latina, las pymes representan una parte fundamental de la economía, constituyendo aproximadamente el 99% del total de empresas y generando cerca del 67% del empleo formal en la región (CEPAL-CAF, 2020). Sin embargo, muchas todavía dependen de registros manuales o herramientas sencillas, como hojas de cálculo, lo que puede dificultar su competitividad en un entorno cada vez más digital (BID, 2025). Los minimarkets, en particular, enfrentan desafíos constantes con la gestión de inventarios, como datos inexactos, pérdida de productos por vencimiento o deterioro, y una visión limitada en tiempo real del estado de sus existencias (Guerrero Salas, 2022). Estas dificultades impactan directamente en su capacidad para tomar decisiones estratégicas y atender de manera más eficiente la demanda del mercado, pero hay muchas oportunidades para mejorar esa situación.

En Ecuador, esta problemática resulta aún más evidente y preocupante. Según el Instituto Nacional de Estadística y Censos (INEC 2023), las microempresas representan más del 90% del tejido empresarial del país, y muchas de ellas todavía no usan sistemas tecnológicos avanzados para administrar sus operaciones, lo que limita su crecimiento y eficiencia. Esta situación se siente con mayor intensidad en regiones menos urbanizadas, como Mira, en la provincia del Carchi, donde el acceso a tecnologías modernas y a la capacitación aún es un desafío para muchos.

El Minimarket Otavalo, ubicado en esta región, es un claro ejemplo de este contexto. Actualmente, este negocio utiliza métodos tradicionales para la gestión de su inventario, como registros en papel y hojas de cálculo en Excel. Estos métodos, aunque comunes en pequeñas empresas, presentan una serie de desventajas significativas (Andrade y Guerrero, 2023). Entre ellas se encuentra la alta propensión al error humano, que conduce a datos inexactos y discrepancias entre los inventarios reales y los reportados.

Además, la ausencia de un sistema de actualización en tiempo real impide que los empleados respondan rápidamente a cambios en la demanda. Desde una perspectiva económica, estas limitaciones se traducen en sobrecompras, pérdidas de productos perecederos y una gestión ineficiente del capital de trabajo, lo que afecta directamente la rentabilidad del negocio (Causado Rodríguez, 2015).

## **1.2. FORMULACIÓN DEL PROBLEMA**

¿Cómo diseñar un sistema computacional que responda a las necesidades de gestión de inventarios del Minimarket Otavalo, frente a las limitaciones de los métodos tradicionales actualmente utilizados en la ciudad de Mira, provincia del Carchi?

## **1.3. JUSTIFICACIÓN**

Esta investigación sirve para ofrecer una alternativa tecnológica a los problemas de control de inventarios que enfrenta el Minimarket Otavalo. Actualmente, el negocio depende de métodos manuales que generan errores, pérdidas económicas y dificultades para tomar decisiones oportunas. El diseño de un sistema computacional permitirá organizar la información de productos, proveedores, compras y ventas de manera centralizada y accesible.

Los principales beneficiarios de esta investigación son los propietarios y empleados del Minimarket Otavalo, en caso de que deseen adquirir esta herramienta diseñada específicamente para sus necesidades. Además, los clientes del negocio se verán beneficiados indirectamente, ya que un mejor control de inventarios garantiza la disponibilidad de productos y reduce la posibilidad de encontrar artículos vencidos o deteriorados. A nivel más amplio, este trabajo puede servir como referencia para otros pequeños comercios de la zona que enfrentan problemáticas similares, contribuyendo al desarrollo económico de la región del Carchi.

Sí, esta investigación aborda directamente problemas reales identificados en el Minimarket Otavalo: errores en el registro de productos, falta de control sobre fechas de vencimiento, desconocimiento del stock disponible en tiempo real y dificultades para generar reportes confiables. El sistema propuesto brinda soluciones concretas para cada uno de estos problemas mediante módulos específicos de gestión de productos, alertas automáticas de stock mínimo, control de lotes con fechas de caducidad y generación de reportes. Aunque el alcance de este trabajo es el diseño del sistema, los artefactos producidos (modelo de base de datos, historias de usuario, prototipos) pueden ser implementados posteriormente para resolver estos problemas de manera efectiva.

Esta investigación contribuye al conocimiento sobre cómo adaptar metodologías ágiles de desarrollo, específicamente Extreme Programming (XP), al contexto de pequeños comercios minoristas en zonas rurales de Ecuador. Se documenta el proceso de análisis de requerimientos, evaluación técnica de herramientas y diseño de arquitectura para un sistema de inventarios, generando información que puede ser utilizada por otros investigadores o desarrolladores que trabajen en contextos similares. Además, la comparación técnica de frameworks frontend (Angular, React, Vue.js), frameworks backend (Express, Django, Spring Boot) y bases de datos (MySQL, PostgreSQL, MariaDB) proporciona criterios de selección fundamentados que pueden ser replicados en otros proyectos.

La investigación propone instrumentos de recolección de datos adaptados al contexto de minimarkets, incluyendo fichas de observación para procesos de inventario y matrices de evaluación técnica con criterios ponderados. Estos instrumentos pueden ser reutilizados o adaptados por otros investigadores que estudien problemáticas similares en el sector comercial minorista. Asimismo, el modelo de base de datos normalizado y las historias de usuario documentadas constituyen artefactos metodológicos que pueden servir como plantillas para el diseño de sistemas similares en otros negocios pequeños.

## **1.4. OBJETIVOS Y PREGUNTAS DE INVESTIGACIÓN**

### 1.4.1. Objetivo General

- Desarrollar un sistema computacional de gestión de inventarios para el Minimarket "Otavalo" en el cantón Mira en la Provincia del Carchi

### 1.4.2. Objetivos Específico

- Fundamentar los principales enfoques existentes en el campo de la teoría de inventarios y las diversas herramientas de desarrollo de software.
- Determinar los procesos actuales y las necesidades del Minimarket Otavalo para definir los requerimientos del sistema de gestión de inventarios.
- Proponer una solución computacional que automatice los procesos de gestión de inventarios del Minimarket Otavalo bajo los principios de la ingeniería de software.

### 1.4.3. Preguntas de Investigación

- ¿Cuáles son los principales enfoques teóricos y herramientas de desarrollo de software existentes para la gestión de inventarios?
- ¿Cuáles son los procesos actuales y las necesidades del Minimarket Otavalo que definen los requerimientos para un sistema de gestión de inventarios?
- ¿Cómo diseñar una solución computacional que automatice los procesos de gestión de inventarios del Minimarket Otavalo bajo los principios de la ingeniería de software?

## II. FUNDAMENTACIÓN TEÓRICA

### 2.1. ANTECEDENTES DE LA INVESTIGACIÓN

El estudio de Pérez y Ramírez (2020) abordó los efectos negativos del uso de tecnologías obsoletas, como las hojas de cálculo, en la administración del inventario en pequeñas y medianas empresas (PYMEs). Los investigadores utilizaron un enfoque cuantitativo con un diseño descriptivo y correlacional, aplicando encuestas a 45 PYMEs del sector comercial y realizando análisis estadístico de los datos recopilados sobre errores de inventario y tiempos de gestión. La investigación encontró que Excel presenta limitaciones significativas en escalabilidad, integridad de datos y eficiencia. Se determinó que el 68% de las empresas estudiadas reportaban al menos 5 errores de transcripción mensuales, y el 73% identificó la falta de automatización como un problema crítico que afecta la exactitud de los inventarios y la planificación. Este antecedente respalda la idea de que el Minimarket Otavalo debe superar los límites de los métodos tradicionales para mejorar su gestión.

En el trabajo de investigación de Jiménez (2021), se exploró cómo la adopción de sistemas de inventario automatizados contribuye a reducir las pérdidas económicas en empresas del sector retail. Se empleó un enfoque mixto con diseño cuasi-experimental, implementando un sistema de inventario en 3 tiendas piloto durante 6 meses y comparando indicadores antes y después de la implementación mediante análisis estadístico t-Student. El estudio demostró que la digitalización permitió reducir las pérdidas por productos perecederos en un 42%, optimizar las órdenes de compra reduciendo el sobre-stock en un 35%, y disminuir el desperdicio de inventario en un 28%. Este antecedente es relevante para el Minimarket Otavalo, ya que uno de sus problemas clave es la gestión ineficiente de productos perecederos.

García y Martínez (2020) realizaron un estudio sobre los desafíos de la gestión de inventarios en pequeños comercios minoristas en Latinoamérica. Utilizaron un enfoque cualitativo con diseño fenomenológico, realizando entrevistas semiestructuradas a 28 propietarios de comercios minoristas en Colombia, México y Ecuador, complementadas con observación participante de los procesos de inventario. Según su investigación, hasta un 60% de los errores en los inventarios físicos

se deben a discrepancias en el registro de datos y a retrasos en la actualización de la información. Como solución, proponen implementar sistemas computacionales adaptados a las necesidades específicas de cada negocio, destacando que una adecuada estructuración del Kardex digital puede reducir errores hasta un 40% y agilizar las tareas administrativas en un 50%. Este antecedente es importante para esta investigación porque resalta la relevancia de analizar los procesos de gestión actuales antes de sugerir soluciones tecnológicas.

Rodríguez (2020) desarrolló un sistema de gestión de inventarios basado en una base de datos relacional para una pequeña tienda de abarrotes. Se aplicó la metodología de desarrollo en cascada (Waterfall), siguiendo las fases de análisis de requerimientos y diseño de la base de datos relacional normalizada hasta 3FN, codificación en PHP con MySQL, pruebas de integración y despliegue. El proyecto se desarrolló en un período de 4 meses. El sistema logró reducir en un 80% los errores de registro y actualización de inventarios, optimizar en un 60% el tiempo de búsqueda de productos, y mejorar la organización y seguridad en el manejo de la información. Además, el uso de una base de datos relacional incrementó la accesibilidad a los datos y permitió la elaboración de reportes detallados, lo que se tradujo en un aumento del 35% en la eficiencia operativa general de la tienda.

Sihombing (2024) diseñó y puso en marcha un sistema de gestión enfocado en mejorar los procesos internos en una organización de servicios. Se aplicó la metodología Extreme Programming (XP) con ciclos iterativos de 2 semanas, documentando las fases de planificación mediante historias de usuario, diseño colaborativo, codificación en parejas (pair programming), pruebas unitarias con cobertura del 85%, y entregas incrementales validadas por usuarios finales. El desarrollo se completó en 5 iteraciones durante 3 meses. La aplicación de XP redujo en un 47% los errores detectados en las pruebas de integración respecto al proyecto anterior de la organización, mejoró en un 35% la eficiencia de los procesos internos, y alcanzó un incremento del 42% en la satisfacción de los usuarios finales, especialmente en los indicadores de facilidad de uso y estabilidad operativa. Este antecedente valida la selección de XP como metodología para el presente proyecto, demostrando su efectividad en la entrega de sistemas alineados con las necesidades reales del cliente.

## **2.2. MARCO TEÓRICO**

### 2.2.1 Inventario

Un inventario se describe como el catálogo, el conteo y la valoración monetaria de todos los activos que posee una compañía. Estos activos pueden incluir productos terminados preparados para su venta, así como insumos que se emplearán en la producción.

El inventario es esencial porque ayuda a las empresas a optimizar sus gastos, minimizar las pérdidas derivadas de la gestión inadecuada de los insumos y asegurar la disponibilidad de productos diseñados para satisfacer las necesidades de los clientes (Andrade y Guerrero, 2023).

Este principio de inventario resulta crucial para el desarrollo de la presente investigación, dado que permite reconocer que una gestión efectiva del stock no se limita a contar productos, sino que también implica supervisar su valor financiero y su disponibilidad en el momento oportuno. Esta comprensión fundamenta la relevancia de contar con herramientas informáticas que documenten de forma exacta las existencias y mantengan la información actualizada, previniendo así pérdidas o carencias que afecten la operatividad de negocios como el Minimarket Otavalo.

### 2.2.2 Tipos de Inventarios

La clasificación de los inventarios permite establecer estrategias de control diferenciadas según el propósito y las características de cada tipo de existencia. Según Mora García (2016), los inventarios pueden clasificarse de diversas maneras dependiendo del criterio de análisis utilizado. Heizer y Render (2014) identifican tres categorías principales que abarcan la mayoría de los contextos empresariales:

- Inventario de materias primas: Comprende todos los materiales e insumos que una empresa adquiere para ser transformados o incorporados en un proceso productivo. Según Ballou (2004), este tipo de inventario es fundamental en empresas manufactureras, ya que su disponibilidad condiciona directamente la capacidad de producción. En el contexto de un minimarket, aunque no existe transformación industrial, este concepto se aplica a los insumos que se utilizan para la preparación de productos como pan, comidas rápidas o bebidas elaboradas en el local.

- **Inventario de productos terminados:** Incluye todos los artículos listos para la venta directa al consumidor final. Chase et al. (2018) señalan que este inventario representa la mayor proporción de existencias en negocios comerciales minoristas. En un minimarket, esta categoría abarca la totalidad de productos exhibidos para la venta: alimentos envasados, bebidas, productos de limpieza, artículos de uso personal, entre otros. El control adecuado de este inventario es crítico porque determina la capacidad del negocio para satisfacer la demanda de sus clientes.
- **Inventario de productos perecederos:** Corresponde a los artículos que tienen una fecha de caducidad definida y requieren condiciones especiales de almacenamiento y rotación. Chopra y Meindl (2016) destacan que este tipo de inventario demanda un control riguroso para evitar pérdidas económicas por vencimiento. En el contexto de un minimarket, esta categoría incluye lácteos, embutidos, pan, frutas, verduras y otros productos frescos que requieren seguimiento de lotes y alertas automatizadas por fecha de vencimiento.

Esta clasificación resulta relevante para la presente investigación porque permite comprender que un sistema de gestión de inventarios para un minimarket debe contemplar mecanismos diferenciados según el tipo de producto. Los artículos perecederos requieren alertas por fecha de vencimiento y control de lotes, mientras que los productos terminados demandan control de stock en tiempo real y alertas por niveles mínimos.

### 2.2.3 Gestión de Inventarios

La gestión de inventarios constituye un proceso fundamental para cualquier entidad. De acuerdo con Toledo et al. (2023), los inventarios representan activos estratégicos cuya gestión adecuada favorece la productividad y la rentabilidad. Su función principal consiste en regular las discrepancias entre oferta y demanda, además de garantizar la disponibilidad de productos en tiempo y forma (Flores-Tapia & Flores-Ceballos, 2023). Una gestión eficiente no solo contempla el almacenaje de productos, sino también la planificación, el control, la valoración, la rotación, y la toma de decisiones relacionadas.

La revisión bibliográfica indica que administrar inventarios no se limita únicamente a mantener existencias, sino que también implica gestionar información que permita

reducir pérdidas, evitar rupturas de stock y aprovechar oportunidades de compra. Durán (2012) enfatiza que los inventarios mal administrados generan costos innecesarios y afectan de manera directa las utilidades.

La revisión de estos conceptos permite comprender que la administración de los inventarios tiene un impacto directo en la eficacia financiera y operativa de una empresa. Este fundamento resulta relevante para la presente investigación, dado que establece cuáles procesos deben analizarse, qué indicadores deben ser medidos y qué componentes deben considerarse como referentes al plantear soluciones que optimicen el control y la toma de decisiones en contextos comerciales minoristas.

#### 2.2.4 Registro de productos

El registro de productos es fundamental para tener un buen control del inventario. Según Causado Rodríguez (2015), mantener un registro bien organizado nos ayuda a clasificar, analizar y priorizar productos según su importancia económica. La clasificación ABC, que se basa en el principio de Pareto, nos muestra que un pequeño porcentaje de productos genera la mayor parte de los ingresos (Arango et al., 2013).

Un buen registro incluye códigos únicos, nombres estandarizados, presentación, categorías, precios, costos, proveedores, ubicación y unidades de medida.

El análisis del registro de productos permite comprender la importancia de organizar adecuadamente esta información en cualquier negocio comercial. Un proceso estructurado justifica la necesidad de implementar mecanismos de registro organizados y sistemáticos que garanticen la trazabilidad, eviten duplicidades y faciliten la clasificación posterior, especialmente al momento de tomar decisiones sobre compras y reposiciones.

#### 2.2.5 Registro de Entradas y Salidas

Las entradas y salidas constituyen los movimientos principales que afectan el stock. Las entradas incluyen compras, devoluciones y ajustes; las salidas comprenden ventas, consumos internos, daños o mermas (Guerrero, 2022). Registrar estos movimientos con exactitud permite mantener actualizado el kardex y detectar diferencias o fraudes (Guzmán et al., 2021).

### 2.2.5.1 Entradas

Según Gómez (2022), las entradas en la gestión de inventarios representan el ingreso de bienes o productos al almacén, ya sea por compras, devoluciones o producción terminada. Estas entradas permiten mantener actualizado el registro de existencias, conocer el valor real del inventario y garantizar la disponibilidad de productos para la venta o uso interno. Mantener un control adecuado de las entradas es fundamental para planificar el abastecimiento de manera efectiva y prevenir faltantes que puedan interrumpir las operaciones comerciales.

Las entradas abarcan la documentación de todos los artículos, la evaluación de su calidad, y también los cambios relacionados con su recepción insumo.

### 2.2.5.2 Salidas

Gómez (2022) señala que las salidas en la gestión de inventarios corresponden a los movimientos que disminuyen las existencias dentro del almacén. Estas pueden deberse a ventas, consumo interno, transferencias o pérdidas por deterioro o caducidad.

El registro minucioso de las salidas ayuda a entender el movimiento de los productos, evita el desperdicio y asegura un manejo efectivo de los inventarios, algo que es esencial para la gestión de decisiones y la rentabilidad de la empresa.

### 2.2.6 Control de Stock y niveles mínimos

Según la Universidad Tecnológica de Jalisco (2017), se conoce como stock al total de productos que posee una empresa o negocio, almacenados en un sitio específico, en transporte o formando parte del ciclo de producción, y destinados a su uso en actividades industriales o de venta.

- Stock mínimo: La cantidad necesaria de productos para atender los pedidos de los consumidores sin quedarse sin inventario.

Este concepto resulta fundamental para la investigación, dado que permite comprender que el stock no se limita a los productos visibles en los estantes o bodegas, sino que también abarca aquellos que se encuentran en tránsito o en proceso dentro del ciclo comercial. Esta perspectiva orienta hacia la necesidad de contar con herramientas computacionales capaces de registrar todos los niveles del inventario y asegurar un control integral de las existencias.

### 2.2.7 Clasificación de Inventarios: Método ABC

La clasificación ABC constituye una técnica de gestión de inventarios basada en el principio de Pareto, también conocido como la regla 80/20. Según Heizer y Render (2014), este método permite categorizar los productos en tres grupos según su valor económico e impacto en el inventario total, facilitando la asignación eficiente de recursos de control y seguimiento.

Ballou (2004) establece que la clasificación ABC divide el inventario en tres categorías principales:

- **Categoría A:** Comprende aproximadamente el 15-20% de los artículos del inventario, pero representa entre el 70-80% del valor total. Estos productos requieren un control riguroso, revisiones frecuentes y políticas de reposición precisas.
- **Categoría B:** Incluye alrededor del 30-35% de los artículos y representa entre el 15-20% del valor del inventario. Requieren un nivel de control intermedio con revisiones periódicas.
- **Categoría C:** Abarca el 50-55% de los artículos restantes, pero solo representa entre el 5-10% del valor total. Pueden gestionarse con controles más simples y menos frecuentes.

Krajewski et al. (2013) señalan que la clasificación ABC permite a las organizaciones concentrar sus esfuerzos de control en los productos que generan mayor impacto económico, optimizando así el uso de recursos humanos y tecnológicos en la gestión del inventario.

En el contexto de un minimarket, la aplicación de este método implica reconocer que productos como licores, cigarrillos, lácteos y otros de alta rotación (categoría A) representan la mayor parte del valor del inventario y, por tanto, requieren un control más frecuente y detallado. Este principio resulta relevante para la presente investigación, ya que fundamenta la necesidad de contar con mecanismos de control diferenciados según la importancia económica de cada artículo.

### 2.2.8 Sistemas de Control de Inventarios

La literatura especializada distingue dos sistemas fundamentales para el control de inventarios: el sistema periódico y el sistema permanente o perpetuo. Según Chase et

al. (2018), la selección entre ambos sistemas depende del tipo de negocio, el volumen de transacciones y los recursos tecnológicos disponibles.

#### 2.2.8.1 Sistema de Inventario Periódico

El sistema periódico, también denominado sistema de revisión periódica, consiste en realizar conteos físicos del inventario en intervalos de tiempo fijos, como semanal, quincenal o mensual (Heizer & Render, 2014). Durante el período entre conteos, no se registran las variaciones en el nivel de existencias de forma individual.

Krajewski et al. (2013) indican que este sistema presenta las siguientes características:

- El costo del inventario se determina únicamente al momento del conteo físico.
- Requiere menor inversión en tecnología y sistemas de información.
- Resulta adecuado para negocios con bajo volumen de transacciones.
- Presenta mayor riesgo de desabastecimiento al no contar con información en tiempo real.

#### 2.2.8.2 Sistema de Inventario Permanente (Perpetuo)

El sistema permanente o perpetuo mantiene un registro continuo y actualizado de cada movimiento del inventario. Según Ballou (2004), cada entrada por compra y cada salida por venta se registra inmediatamente, permitiendo conocer en todo momento el nivel de existencias disponibles.

Chase et al. (2018) destacan las ventajas de este sistema:

- Proporciona información en tiempo real sobre el stock disponible.
- Facilita la detección oportuna de discrepancias y posibles pérdidas.
- Permite la implementación de alertas automáticas por stock mínimo.
- Mejora la precisión en la planificación de compras y reposición.

La comprensión de estos dos enfoques resulta fundamental para la presente investigación, dado que permite identificar cuál modelo se adapta mejor a las necesidades de un negocio como el Minimarket Otavalo. El modelo de inventario permanente ofrece ventajas significativas para negocios con alto volumen de transacciones diarias, al proporcionar visibilidad continua del stock y facilitar la toma de decisiones oportunas.

### 2.2.9 Punto de Reorden y Stock de Seguridad

El punto de reorden representa el nivel de inventario en el cual debe generarse una nueva orden de compra para evitar el desabastecimiento. Según Heizer y Render (2014), este indicador se calcula considerando la demanda promedio durante el tiempo de entrega del proveedor.

$$\text{Punto de Reorden (ROP)} = \text{Demanda diaria promedio} \times \text{Tiempo de entrega (días)}$$

Chopra y Meindl (2016) explican que, en condiciones de incertidumbre, resulta necesario mantener un stock de seguridad adicional que proteja al negocio contra variaciones en la demanda o retrasos en las entregas de los proveedores.

El stock de seguridad se define como la cantidad adicional de inventario que se mantiene para absorber la variabilidad de la demanda y del tiempo de reabastecimiento (Ballou, 2004). Su cálculo considera factores como:

- La desviación estándar de la demanda histórica.
- El nivel de servicio deseado (probabilidad de no quedarse sin stock).
- La variabilidad en los tiempos de entrega del proveedor.

Estos conceptos resultan relevantes para la gestión del Minimarket Otavalo, ya que fundamentan la importancia de contar con mecanismos de alerta cuando el stock de un producto alcance niveles mínimos predefinidos. La aplicación de estos principios permitiría gestionar los pedidos de reposición de manera oportuna y evitar pérdidas de ventas por desabastecimiento.

### 2.2.10 Indicadores de Gestión de Inventarios

Los indicadores de gestión de inventarios constituyen métricas cuantitativas que permiten evaluar la eficiencia y efectividad de las operaciones de almacenamiento y distribución. Según Mora García (2016), estos indicadores proporcionan información objetiva para la toma de decisiones y la mejora continua de los procesos.

#### 2.2.10.1 Rotación de Inventarios

La rotación de inventarios mide la frecuencia con que el inventario se renueva durante un período determinado. Heizer y Render (2014) definen este indicador mediante la siguiente fórmula:

$$\text{Rotación de Inventarios} = \text{Costo de Ventas} / \text{Inventario Promedio}$$

Una rotación alta indica que los productos se venden rápidamente, lo cual en general se considera positivo, aunque debe equilibrarse para evitar rupturas de stock. Una rotación baja puede señalar exceso de inventario o productos de lenta salida (Chase et al., 2018).

#### 2.2.10.2 Días de Inventario

Los días de inventario representan el tiempo promedio que un producto permanece almacenado antes de ser vendido. Según Krajewski et al. (2013), este indicador se calcula como:

$$\text{Días de Inventario} = 365 / \text{Rotación de Inventarios}$$

Este indicador permite identificar productos con movimiento lento que podrían requerir estrategias de promoción o liquidación, así como optimizar el nivel de inversión en inventario.

#### 2.2.10.3 Exactitud del Inventario

La exactitud del inventario mide la concordancia entre las existencias físicas reales y los registros del sistema. Ballou (2004) presenta la fórmula para su cálculo:

$$\text{Exactitud del Inventario} = (\text{Conteo Físico} / \text{Registro del Sistema}) \times 100$$

Un nivel de exactitud superior al 95% se considera aceptable en la mayoría de las industrias, mientras que valores inferiores indican problemas en los procesos de registro o posibles pérdidas por robo, daño o error humano (Chopra & Meindl, 2016).

Estos indicadores resultan fundamentales para justificar la necesidad de contar con herramientas que generen reportes en negocios como el Minimarket Otavalo. La disponibilidad de estas métricas permite evaluar objetivamente el desempeño de la gestión de inventarios y tomar decisiones informadas sobre compras, promociones y control de mermas.

#### 2.2.11 Tecnologías Aplicadas a la Gestión de Inventarios

La evolución tecnológica ha transformado significativamente la gestión de inventarios en el sector comercial. Según Bowersox et al. (2013), la automatización de procesos mediante tecnologías de identificación y sistemas integrados permite reducir errores, optimizar tiempos y mejorar la trazabilidad de los productos.

### 2.2.11.1 Códigos de Barras

Los códigos de barras constituyen la tecnología de identificación automática más utilizada en el comercio minorista. Chopra y Meindl (2016) señalan que este sistema permite codificar información del producto mediante barras paralelas de diferentes anchos que pueden ser leídas por escáneres ópticos.

- Las ventajas de los códigos de barras incluyen:
- Registro rápido y preciso de productos en el punto de venta.
- Reducción significativa de errores de digitación manual.
- Integración directa con sistemas de inventario y facturación.
- Bajo costo de implementación comparado con otras tecnologías.

### 2.2.11.2 Sistemas Integrados Punto de Venta-Inventario

La integración entre el sistema de punto de venta (POS) y el módulo de inventario constituye un avance fundamental para la gestión eficiente de pequeños comercios. Laudon y Laudon (2016) explican que esta integración permite que cada transacción de venta actualice automáticamente los niveles de stock, eliminando la necesidad de registros manuales posteriores.

Los beneficios de esta integración incluyen:

- Actualización del inventario en tiempo real con cada venta.
- Eliminación de discrepancias por registros tardíos o incompletos.
- Generación automática de alertas cuando el stock alcanza niveles mínimos.
- Disponibilidad inmediata de información para la toma de decisiones.

Esta fundamentación teórica resulta relevante para la presente investigación, ya que describe las características y beneficios de las tecnologías que pueden aplicarse en negocios minoristas como el Minimarket Otavalo. La integración entre ventas e inventario representa una solución a los problemas de actualización tardía y discrepancias que enfrentan los comercios que operan con métodos manuales.

### 2.2.12 Auditoría de Inventarios

La auditoría de inventarios consiste en el proceso sistemático de verificación y validación de las existencias físicas contra los registros contables y operativos. Según Arens et al. (2017), este proceso permite detectar diferencias, identificar sus causas y establecer medidas correctivas para mejorar el control interno.

Whittington y Pany (2016) identifican los principales objetivos de la auditoría de inventarios:

- Verificar la existencia real de los productos registrados.
- Comprobar la exactitud de las cantidades y valores reportados.
- Detectar posibles pérdidas por robo, daño, obsolescencia o errores.
- Evaluar la efectividad de los controles internos implementados.

Estos conceptos resultan esenciales para la presente investigación, dado que fundamentan la importancia de contar con mecanismos de auditoría que registren todos los movimientos del inventario, cambios de precios, anulaciones y demás operaciones sensibles. Esta funcionalidad permite mantener un control completo sobre las actividades del negocio y facilita la detección oportuna de irregularidades, aspecto crítico para negocios como el Minimarket Otavalo donde participan múltiples usuarios.

#### 2.2.13 Kardex

Según Lima E, (2020) un Kardex es un archivo organizado y minucioso de los ingresos y egresos de los artículos o productos que posee una compañía en su inventario. Este archivo ayuda a mantener una supervisión precisa de las existencias, lo que facilita la administración y el seguimiento de los productos. El Kardex es una herramienta esencial en la gestión de inventarios, ya que proporciona datos clave sobre las cantidades disponibles, los costos asociados y las transacciones realizadas.

La relevancia del Kardex radica en su capacidad para ayudar a las organizaciones a mantener un control efectivo de sus inventarios. Al llevar un registro detallado de cada entrada y salida de productos, las empresas pueden tomar decisiones más acertadas sobre compras, almacenamiento y distribución.

El Kardex constituye una herramienta fundamental para gestionar de forma organizada todos los movimientos de entrada y salida en cualquier negocio comercial, facilitando el control operativo y la organización de la información relacionada con el inventario

FÁBRICA XYZ			TARJETA DE KARDEX N° _____						
CÓDIGO: _____		ALMACÉN: _____			INVENTARIO MAX.: _____				
DESCRIPCIÓN: _____									
RECIBIDO				EMITIDO			SALDO		
FECHA	CANT.	PRECIO UNITARIO	VALOR	FECHA	CANT.	VALOR	CANT.	COSTO UNITARIO	VALOR

Figura 1. Ejemplo de Kardex

### 2.2.14 Valoración del Inventario

La valoración asigna el costo a los productos almacenados y vendidos según (Hansen & Mowen, 2018). Los métodos más conocidos son:

Tabla1. Métodos de valoración de inventario

Método	Características	Uso recomendado
<b>FIFO</b>	Asume que los primeros en entrar son los primeros en salir; inventario final a precios recientes.	Productos perecederos, mercados inflacionarios.
<b>LIFO</b>	Últimos en entrar son los primeros en salir; inventario final a precios antiguos.	Usado donde se permiten ventajas fiscales.
<b>Promedio ponderado</b>	Calcula un costo promedio para todas las unidades disponibles.	Inventario homogéneo, variaciones frecuentes de precio.

A partir de esta comparación, se identifica que los métodos de valoración permiten determinar cómo varía el costo del inventario según el método aplicado. Esta comprensión resulta útil para analizar qué método sería más adecuado al momento de generar reportes y calcular los costos de venta de manera precisa y confiable en un contexto comercial.

### 2.2.15 Control de existencias

El control interno permite verificar que las existencias físicas coincidan con los registros. Según (La Febre & Rivera, 2023). Las falencias en los controles generan pérdidas directas, errores contables y rupturas de stock

Los controles de inventario son fundamentales para asegurar que todo opere de manera confiable y sin alternativas. Entender estos procedimientos realmente me ayuda a determinar qué mecanismos implementar, de modo que los datos utilizados en la investigación sean precisos y reflejen fielmente la situación real del inventario.

Este análisis orienta hacia la inclusión de funciones que faciliten la generación de listados de auditoría y la verificación entre el inventario físico y los registros digitales, aspectos que resultan esenciales para cualquier negocio comercial que busque mantener datos confiables y detectar diferencias oportunamente.

#### 2.2.16 Gestión de proveedores

Según (Mora García, 2016). Los proveedores influyen en los precios, la calidad y la disponibilidad del inventario. Evaluarlos permite establecer relaciones estables y mitigar riesgos. Los criterios comunes incluyen plazos, precios, calidad, historial de cumplimiento y comunicación (López-Vera et al., 2023).

Esta revisión fundamenta la importancia de registrar adecuadamente los datos de los proveedores y evaluar su desempeño, así como de contar con mecanismos de gestión que faciliten este control. Estas funcionalidades resultan esenciales para optimizar las decisiones sobre compras, reposición y negociaciones futuras en cualquier contexto de gestión de inventarios.

#### 2.2.17 Facturación y su relación con la gestión de inventarios

La facturación, en el ámbito de un sistema informático, representa el acto que culmina la transacción de venta y, por ello, establece la relación más importante y en tiempo real con la administración del inventario. Esta conexión garantiza la precisión de los registros y la constante disponibilidad de productos, lo cual es esencial para el funcionamiento eficaz de un Minimarket (Chiquito Tigua y Cedeño Muñoz, 2025).

La facturación no se limita a ser una actividad contable; es el proceso logístico que provoca la actualización instantánea del inventario. Como mencionan Toledo y otros (2023), la gestión y el control del inventario dependen de elementos operativos organizados y conectados que determinen cuánto y cuándo se debe reabastecer, donde la gestión de la orden de venta (facturación) es un paso clave que identifica la demanda del mercado y sienta las bases para las operaciones posteriores (Corella-Parra y Olea-Miranda, 2023).

#### 2.2.18 Registro automático de ventas

El registro automático es el mecanismo mediante el cual una venta verificada y facturada se refleja de forma inmediata en una reducción de la cantidad vendida en el inventario digital.

### 2.2.18.1 Impacto Operacional

La automatización garantiza que el saldo reflejado en el sistema sea un reflejo preciso de la mercancía disponible para la venta, previniendo el quiebre o el exceso de stock. La rapidez y precisión en el registro de ventas son, por tanto, directamente proporcionales a la efectividad del proceso de aprovisionamiento (Romero et al., 2021, citado en Veloz & Parada, 2024).

### 2.2.19 Ingeniería de software

La ingeniería de software (IS) es como una guía que nos ayuda a crear, usar y cuidar el software de manera ordenada, disciplinada y comprobable (Pressman & Maxim, 2021, p. 55). No se trata solo de programar o codificar; es un proceso cuidadoso con para desarrollar productos de software efectivos y confiables, realmente satisfagan las necesidades del usuario final, respetando los plazos y presupuestos establecidos. Por ejemplo, en un sistema de gestión de inventarios, la IS garantiza que el software funcione correctamente y que los empleados del minimarket puedan usarlo con facilidad, además de mantenerse al día con las actualizaciones del negocio.

La ingeniería de software es la disciplina que hace posible que el sistema de inventarios del minimarket no solo funcione bien, sino que además sea escalable, resistente y sencillo de mantener a lo largo del tiempo.

### 2.2.20 Arquitectura de software

La arquitectura de software se refiere a la organización estructural de un sistema, incluyendo sus componentes de software, sus propiedades visibles desde el exterior y las relaciones entre ellos. Elegir la arquitectura correcta es crucial, ya que establece las bases para la escalabilidad, el rendimiento y el mantenimiento del sistema a largo plazo (Sommerville, 2011).

Entre los patrones más utilizados en el desarrollo web se encuentra la arquitectura de tres capas (Three-Tier Architecture). Este modelo propone la separación lógica de la aplicación en tres niveles independientes:

- Capa de Presentación: Es la interfaz a través de la cual el usuario final interactúa.
- Capa de Negocio (Lógica): responsable de procesar los datos y aplicar las reglas comerciales.

- Capa de Datos: la sección donde se almacena y administra la información de manera persistente.

Esto permite que cada capa se desarrolle y actualice de forma independiente, facilitando la especialización del equipo de desarrollo y la migración tecnológica sin afectar al sistema completo (IBM Cloud Education, 2021).

#### 2.2.21 Tipos de sistemas de gestión de inventarios

##### 2.2.21.1 Sistemas Locales (On-Premise)

Son aquellos donde el software se instala directamente en los servidores físicos y computadoras de la empresa.

- Característica: La organización mantiene el control total de los datos y la seguridad física, no dependiendo de una conexión a internet externa para operar.
- Limitación: Requiere una alta inversión inicial en hardware (servidores), licencias y personal de TI para el mantenimiento y las copias de seguridad (Oracle NetSuite, 2024).

##### 2.2.21.2 Sistemas en la Nube (Cloud-Based)

El software y los datos se alojan en servidores remotos gestionados por un proveedor de servicios (modelo SaaS) y se accede a ellos a través de internet.

- Característica: Permite el acceso a la información en tiempo real desde cualquier ubicación geográfica y dispositivo.
- Ventaja: Reduce significativamente los costos iniciales de infraestructura y ofrece escalabilidad automática según el crecimiento del negocio (Oracle NetSuite, 2024).
- Sistemas Híbridos: Combinan características de los dos modelos anteriores, manteniendo procesos críticos en servidores locales para garantizar la velocidad, mientras sincronizan datos analíticos o históricos en la nube para el acceso remoto.

#### 2.2.22 Metodología de Desarrollo

Una metodología de desarrollo de software es como un conjunto de guías, procesos y técnicas que acompañan a un desarrollador en las diferentes etapas de planificación, construcción y entrega de un sistema informático (Pressman & Maxim,

2021). Su principal objetivo es garantizar que el software se entregue con la calidad deseada, respetando los plazos y presupuestos establecidos (Sommerville, 2021).

#### 2.2.22.1 Metodología eXtreme Programming (XP)

La programación extrema (XP) es una metodología ágil excelente, diseñada para gestionar proyectos con requerimientos inciertos o en constante cambio. Sus principios son fundamentales para mantener la calidad y la disciplina en el desarrollo individual, aunque, en realidad, está pensada para trabajar en equipo, lo que hace que todo sea más colaborativo y eficiente.

- La planificación iterativa: consiste en dividir el proyecto en ciclos muy cortos, que usualmente duran una a dos semanas. Esto ayuda a los desarrolladores a centrarse en pequeñas partes de la funcionalidad, permitiéndoles entregar valor de manera rápida y efectiva.
- Diseño Simple y Mantenible: Lo más importante es crear un código que sea simple y que cumpla con los requisitos actuales, mientras vamos corrigiendo y mejorando su estructura de forma constante. De esta manera, hacer cambios en el futuro será mucho más fácil, sin tener que depender siempre de un equipo para revisarlo.
- Feedback: El desarrollador se mantiene en contacto estrecho con el Minimarket para confirmar que las funcionalidades entregadas cumplen con sus expectativas y realmente ayudan a resolver sus problemas de inventario, lo que crea una colaboración más fluida y confiable.

**Tabla 2.** Tabla comparativa metodologías ágiles

<b>Criterio</b>	<b>XP (Extreme Programming)</b>	<b>Scrum</b>	<b>RUP</b>	<b>Kanban</b>
Enfoque base	Calidad técnica y simplicidad del diseño; ciclos muy cortos con refactorización y pruebas.	Gestión de trabajo por sprints y roles claros; prioriza valor de negocio.	Proceso unificado por fases (inicio, elaboración, construcción, transición) con mucha documentación.	Flujo continuo; limitar trabajo en curso y mejorar el proceso paso a paso.
Ritmo / cadencia	Iteraciones de 1–2 semanas con entregas pequeñas y frecuentes.	Sprints fijos de 1–4 semanas con demo y retrospectiva.	Fases más largas; iteraciones internas según el proyecto.	Sin sprints; trabajo entra y sale continuamente según capacidad.
Roles clave	Equipo técnico colaborando con "cliente in situ"; pares programando juntos; coach. TDD, integración continua, pair programming, refactorización constante, diseño evolutivo.	Product Owner, Scrum Master y Equipo de Desarrollo.	Arquitecto, analistas, diseñadores, testers y gerente de proyecto.	No impone roles; se adapta a la estructura existente.
Prácticas distintivas	Historias pequeñas y estimadas; se ajusta en cada iteración.	Backlog priorizado, planificación por sprint, daily, review y retro.	Artefactos y diagramas formales, gestión de requisitos, arquitectura definida al inicio.	Tablero visual, límites WIP, métricas de flujo (lead time, throughput).
Planificación	"El diseño más simple que funciona" y que puede crecer con el código. Pruebas automatizadas desde el día uno, cobertura amplia y CI.	Roadmap y sprints; compromiso de alcance por sprint.	Plan maestro por fases, con hitos y entregables definidos.	Priorización dinámica; se atiende lo más importante según capacidad.
Diseño / Arquitectura	Muy bienvenida: se asume y se acomoda en cada iteración. Código limpio y probado, alta adaptabilidad, aprendizaje rápido.	Diseño emergente durante el sprint; suficiente para completar historias.	Arquitectura pensada y documentada desde el inicio.	No prescribe diseño; se centra en el flujo del trabajo.
Gestión de calidad	Se incorpora entre sprints y en grooming del backlog.	QA dentro del sprint; Definition of Done.	Ciclos formales de verificación/validación y revisiones.	Se apoya en políticas de calidad del equipo; no define técnicas.
Cambio de requisitos	Buena organización y foco en valor, transparencia del avance.	Se incorpora entre sprints y en grooming del backlog.	Cambios gestionados mediante control de configuración y alcance.	Se reordena la cola y se mueve el flujo; muy flexible.
Ventajas	Requiere disciplina técnica; sin TDD/CI se degrada.	Puede caer en "ceremonias sin técnica" si falta ingeniería.	Pesado y lento para equipos pequeños o académicos.	Simplicidad, rapidez para empezar, excelente para operaciones. Puede quedar difuso el alcance si no hay metas claras.
Riesgos	Equipos pequeños y medianos, requisitos cambiantes, énfasis en la calidad técnica.	Productos con roadmap, varios interesados y necesidad de cadencia.	Sistemas críticos o regulados que exigen documentación formal.	Soporte, mantenimiento, equipos de Help Desk o DevOps.

Después de analizar las metodologías ágiles, tanto XP (Extreme Programming) como Scrum son las opciones más apropiadas para desarrollar sistemas de gestión de inventarios en pequeños negocios. Ambas metodologías tienen características que encajan bien con las necesidades de estos proyectos: equipos pequeños, requisitos que pueden cambiar durante el proceso y la importancia de entregar resultados funcionales en plazos cortos.

### 2.2.23 IDEs y herramientas de desarrollo

Según Pressman y Maxim (2020), un Entorno de Desarrollo Integrado (IDE) es una herramienta que reúne en una misma interfaz todas las funciones necesarias para el desarrollo de software, como el editor de código, el compilador, el depurador y las utilidades de gestión.

En resumen, un IDE facilita el trabajo del programador porque centraliza actividades que tradicionalmente se realizaban por separado, haciendo el proceso de desarrollo más ágil y ordenado.

**Tabla 3.** IDEs y herramientas de desarrollo

Herramienta	Tipo	Ventajas
Visual Studio Code	Editor avanzado	Extensiones para Angular/Node, integración con Git, gratuitas.
PhpStorm / WebStorm	IDE	Refactorización avanzada.
MySQL Workbench	BD	Diseño y modelado de datos.
Postman	API testing	Pruebas rápidas y documentación.

Después de evaluar las opciones disponibles, Visual Studio Code es la mejor elección por su alta capacidad de extensión, su facilidad para integrar herramientas de frontend y backend, su soporte nativo de Git y su compatibilidad con diversos lenguajes. Además, su ligereza y gratuidad lo convierten en una opción práctica y eficiente para proyectos de cualquier escala.

### 2.2.24 Lenguajes de programación

Letelier (2006) sostiene que la elección del lenguaje de programación depende de factores como la facilidad de mantenimiento, el ecosistema disponible, el rendimiento y la escalabilidad.

**Tabla 4.** Comparativa de lenguajes de programación

Lenguaje / Plataforma	Características	Ventajas	Limitaciones
Node.js / JavaScript	- Node.js es un entorno de ejecución para JavaScript en servidor - -JavaScript es un lenguaje orientado a eventos y usado tanto en navegador como en servidor.	Permite usar el mismo lenguaje en el frontend y el backend; gran ecosistema (npm)	Node.js requiere un buen manejo de sus elementos; por lo tanto , para aplicaciones CPU intensivas puede no ser ideal.
TypeScript	Superset de JavaScript que añade tipado estático, interfaces y compilación a JavaScript	Mejora la mantenibilidad y detección temprana de errores en proyectos grandes; mejor autocompletado y refactorización en IDEs.	Añade el paso de compilación y la curva de aprendizaje por el tipado; requiere configuración inicial.
PHP	Lenguaje interpretado tradicionalmente usado en hosting compartido y desarrollo web.	Fácil de aprender; amplia disponibilidad de hosting y frameworks maduros (p. ej. Laravel).	Menor rendimiento en tareas intensivas que con soluciones optimizadas.
Python	Lenguaje de alto nivel con sintaxis clara; fuerte en scripting, ciencia de datos y automatización.	Excelente para IA/ML y análisis, el cual ofrece muchos frameworks y bibliotecas científicas.	Menor rendimiento en alta concurrencia, sin arquitecturas específicas; además, el GIL de CPython limita algunos patrones.
Java	Lenguaje compilado a bytecode y ejecutado en JVM; fuertemente tipado y orientado a objetos.	Muy escalable y robusto de amplio uso empresarial y ecosistema maduro.	Curva de aprendizaje más pronunciada; más verboso en algunas tareas.

Con base en este análisis, se determina que la combinación de JavaScript y TypeScript resulta ideal para proyectos de APIs modernas y aplicaciones web que crecen. JavaScript ofrece mucha flexibilidad y un ecosistema muy completo, lo que permite desarrollar tanto la parte del cliente como numerosas soluciones del servidor. Por su parte, TypeScript aporta valor mediante el tipado estático, las interfaces y mejores herramientas de verificación durante la compilación. Esto ayuda a reducir errores, facilita las refactorizaciones y hace que los proyectos de tamaño mediano y grande sean más fáciles de mantener.

Según Goldberg, J. (2022). TypeScript es un superconjunto de JavaScript que incorpora tipado estático, clases, interfaces, genéricos y otras herramientas de programación más estructuradas, lo cual facilita el desarrollo de aplicaciones más robustas, escalables y fáciles de mantener.

Este lenguaje proporciona ventajas como la detección temprana de errores, una mejor organización del código y una transición más segura en proyectos de gran envergadura en comparación con JavaScript puro.

#### 2.2.25 Frameworks

Un framework o marco de trabajo es un conjunto estructurado de conceptos, prácticas y criterios que simplifican y hacen más eficiente la resolución de un problema específico. En desarrollo de software, proporciona una estructura fundamental con herramientas y librerías listas para usar, facilitando la codificación, reduciendo tareas repetitivas y garantizando mayor calidad en el código (Pressman, 2010).

##### 2.2.25.1 Angular

Es una plataforma de desarrollo de código abierto que Google mantiene, diseñada con TypeScript. Destaca por ser un framework muy completo que incluye herramientas útiles como enrutamiento, formularios y cliente HTTP. Su estructura se basa en componentes y servicios, lo que la hace muy sólida para construir aplicaciones empresariales que puedan crecer sin problema (Google Developers, 2024).

##### 2.2.25.2 React

Es una biblioteca de JavaScript desarrollada por Meta (Facebook) que facilita la creación de interfaces de usuario de manera sencilla eficiente. Está diseñada para que puedas usar componentes que puedas reutilizar en diferentes partes de tu proyecto, y usa un DOM Virtual para hacer que la interfaz funcione de forma más rápida y suave. Aunque es muy flexible, es posible que necesites integrar algunas librerías externas para que funcione como un framework completo (Meta, 2024).

##### 2.2.25.3 Vue.js

Es un framework progresivo que facilita una adopción paso a paso, perfecto para quienes buscan una integración sencilla pero potente. Su ligereza y rápida curva de aprendizaje lo hacen una opción muy atractiva (Vue Team, 2024).

##### 2.2.25.4 Express.js

Es un framework diseñado para el entorno de Node.js, que destaca por ser sencillo y flexible. Nos ofrece un conjunto de herramientas potentes para crear aplicaciones

web y móviles, especialmente APIs REST, haciendo que la gestión de rutas y middlewares sea más fácil y eficiente (OpenJS Foundation, 2024).

**Tabla 5.** Tabla comparativa de frameworks

Característica	Angular	React	Vue.js
Desarrollador	Google	Meta (Facebook)	Comunidad
Lenguaje Base	TypeScript	JavaScript / JSX	JavaScript
Arquitectura	MVC / Componentes	Componentes / V-DOM	MVVM / Componentes
Estructura	Completa y Estricta	Flexible (Librería)	Progresiva
Curva de Aprendizaje	Alta	Media	Baja

En la investigación actual, tras revisar los diferentes frameworks disponibles, se decide optar por Angular y Express.js. Esto se debe a que, al estar basados en TypeScript, aseguran un tipado estricto que ayuda a mantener la integridad de los datos en sistemas como la facturación y los inventarios, evitando errores de cálculo. Además, se escoge Express.js para el backend por su eficiencia en la gestión de solicitudes HTTP y por permitir que todo el proyecto utilice un único lenguaje de programación, JavaScript/TypeScript, lo cual facilita la estructura del sistema.

#### 2.2.26 Base de datos

Los SGBD permiten almacenar y gestionar datos de forma estructurada. Su elección determina el rendimiento, la integridad de la información y la escalabilidad del sistema (Sánchez et al., 2011).

**Tabla 6.** Tabla comparativa de bases de datos

Base de datos	Tipo	Ventajas	Desventajas	N.º de conexiones	Nivel de seguridad
MySQL (8.x)	Relacional	Buen rendimiento en lectura; amplio uso en web; ecosistema maduro.	Menos eficiente en consultas muy complejas;	151 (por defecto) / ≤ 500 conexiones	Medio
PostgreSQL (13+)	Relacional-objeto	Integridad y consistencia avanzadas; funciones SQL potentes. Mejoras de rendimiento sobre MySQL en varios motores; compatible en gran parte con MySQL.	Requiere administración más técnica para tuning y escalado.	100 (por defecto) / ≤ 300 conexiones	Alto
MariaDB (10.x+)	Relacional	Mejoras de rendimiento sobre MySQL en varios motores; compatible en gran parte con MySQL.	Algunas diferencias/compatibilidad es con MySQL según versión.	151 (por defecto) / ≤ 500 conexiones	Medio-Alto

SQLite (3.x)	Embebida	Muy ligera y portable; sin servidor; ideal para aplicaciones cliente/móvil.	No apta para escrituras concurrentes intensas; dependencia de la app para seguridad.	Conexiones de lectura: cientos–miles (práctico) / Escrituras simultáneas: 1	Bajo–Medio
--------------	----------	---	--	---	------------

Según el análisis comparativo realizado, MySQL se identifica como una opción apropiada para contextos comerciales pequeños y medianos, debido a su facilidad de uso, excelente desempeño en operaciones comunes, amplia compatibilidad y bajo costo de implementación. Estos criterios pueden respaldar técnicamente su selección en proyectos de desarrollo de sistemas de gestión.

### 2.2.27 Seguridad de la información y control de accesos

La protección de la información se fundamenta en principios esenciales como la confidencialidad, la integridad y la disponibilidad (Avenía, 2017). Estos principios nos guían en la elección de controles técnicos, administrativos y físicos que protejan nuestros activos de información frente a amenazas internas y externas. Entre las amenazas más comunes para los sistemas de información están la inyección SQL, el cross-site scripting (XSS), el robo de credenciales y el acceso no autorizado. Como señalan Sanabria & Ospina (2020), estas amenazas pueden afectar tanto la continuidad de las operaciones como la confidencialidad de nuestros datos más sensibles.

#### Características

- Autenticación multifactor (MFA) para cuentas administrativas y sensibles.
- Gestión de identidades y accesos (IAM) con privilegios mínimos y separación de funciones.
- TLS/SSL activo para el cifrado de datos en tránsito.
- Cifrado en reposo aplicado a columnas/tablespaces con datos sensibles.
- Control de versiones y gestión de parches, con un ciclo de vida documentado de vulnerabilidades.

Este análisis aporta claridad sobre los riesgos que deben considerarse en cualquier desarrollo de sistemas de información. Los fundamentos presentados justifican la aplicación de controles de seguridad adecuados y su incidencia directa en la protección de datos y en la continuidad de las operaciones del negocio

#### 2.2.28 Estándares de calidad.

Los estándares de calidad son documentos normativos que establecen requisitos, criterios y buenas prácticas para garantizar que productos, servicios y procesos cumplen niveles aceptables de desempeño, seguridad y consistencia. Estos estándares permiten comparar y armonizar procedimientos entre organizaciones, fomentar la confianza del cliente y promover la mejora continua mediante controles, mediciones y procesos normalizados (ISO, 2015; Hoyle, 2017).

##### 2.2.28.1 Normativa ISO 9001 2015

La norma ISO 9001:2015 establece los requisitos para un sistema de gestión de la calidad (SGC), centrado en la mejora continua, en atender las necesidades y satisfacer a los clientes, en el liderazgo y en la gestión a través de procesos. Además, ISO 9001:2015 presenta una estructura de nivel superior que facilita la integración con otros sistemas de gestión y resalta la importancia del pensamiento basado en riesgos, la gestión del contexto organizacional y la toma de decisiones basadas en evidencia. En la práctica, la norma invita a definir los procesos principales, las responsabilidades, los indicadores de desempeño y los mecanismos para monitorear y mejorar el SGC (ISO, 2015; Hoyle, 2017).

Con base en esta información, la ISO 9001:2015 aporta un marco reconocido internacionalmente para documentar procesos, definir indicadores de calidad y establecer criterios de aceptación en sistemas de software y de gestión.

### **III. METODOLOGÍA**

#### **3.1. ENFOQUE METODOLÓGICO**

##### **3.1.1. Enfoque**

Para esta investigación, se empleó un enfoque metodológico mixto que integra técnicas cualitativas y cuantitativas. Este método permitió comprender y describir los procesos operativos del Minimarket a través de una entrevista al dueño del negocio Minimarket Otavalo, así como también la observación directa y el análisis documental, mientras que también permitió evaluar y comparar objetivamente las herramientas tecnológicas usando matrices de evaluación y fichas técnicas con criterios medibles y también los procesos de gestión de inventarios que se llevan a cabo. Así, se logró una visión clara del problema que respalda las decisiones de diseño del sistema.

##### **3.1.2 Enfoque Mixto**

El enfoque mixto, según Hernández-Sampieri y Mendoza (2018), implica integrar de manera sistemática los métodos cuantitativo y cualitativo en un solo estudio para lograr una comprensión más integral del fenómeno investigado. Este enfoque resultó apropiado para la presente investigación porque permitió analizar cualitativamente los procesos actuales de gestión de inventarios mediante fichas de observación, entrevista y levantamiento de información, mientras que el componente cuantitativo facilita la evaluación técnica de herramientas de desarrollo mediante matrices comparativas con indicadores numéricos, como el número de conexiones, los niveles de seguridad y los tiempos de respuesta en base de datos y comparaciones de diferentes herramientas de desarrollo de software. La combinación de ambos enfoques me proporcionó una base sólida tanto para comprender la realidad operativa del negocio como para justificar técnicamente la selección de tecnologías.

### 3.1.3. Tipo de Investigación

#### 3.1.3.1 Investigación Descriptiva

Según Hernández, Fernández y Baptista (2014), la investigación descriptiva se centra en representar los fenómenos tal como son, sin alterarlos ni intervenir en ellos; su principal finalidad es ofrecer una visión clara y detallada de la situación. En este proyecto, la investigación descriptiva no solo me permitió describir en detalle el proceso actual de gestión de inventarios en el minimarket, incluyendo cómo se registran y controlan, los problemas frecuentes del personal y las limitaciones del sistema actual, sino también documentar de manera sistemática los aspectos tecnológicos relacionados. Esto abarca los recursos tecnológicos disponibles, lenguajes de programación, frameworks y bases de datos, así como los flujos de información entre las áreas y los factores que provocan errores en las operaciones. Con esta información, se pudo priorizar las necesidades reales, establecer requisitos técnicos claros para el diseño del nuevo sistema y justificar las decisiones del proyecto.

#### 3.1.3.2 Investigación de Campo

Según Arias (2016, citado en UNIB.E, 2022), la investigación de campo se basa en recoger datos directamente de las personas o del entorno donde ocurren los hechos, sin intervenir ni modificar las condiciones existentes.

En este proyecto, la investigación de campo fue especialmente valiosa porque permitió observar de primera mano las labores diarias del Minimarket Otavalo sin alterar sus rutinas. Mediante observación directa y entrevistas breves al personal, se identificaron problemas concretos en la gestión de almacén como errores al copiar datos, demoras en las actualizaciones y falta de control sobre productos próximos a su fecha de caducidad. Además, se constató la ausencia de un sistema tecnológico que automatice y centralice los procesos de inventario, lo que aumenta la probabilidad de errores humanos y retrasa las operaciones diarias. Estos hallazgos directos aportaron información práctica y detallada que permitió definir requerimientos funcionales y diseñar soluciones ajustadas a la realidad del negocio.

### **3.2. IDEA A DEFENDER**

El análisis de los procesos de gestión de inventarios y la evaluación técnica de herramientas de desarrollo tecnológico me permitirán fundamentar el diseño de un sistema computacional adaptado a las necesidades del Minimarket Otavalo, definiendo la estructura y la tecnología apropiadas para automatizar el control de existencias y la toma de decisiones.

### **3.3. DEFINICIÓN Y OPERACIONALIZACIÓN DE LAS VARIABLES**

**Tabla 7.** Operacionalización de variables (Variable dependiente)

<b>Variable Dependiente</b>	<b>Definición</b>	<b>Dimensión</b>	<b>Indicador</b>	<b>Instrumento</b>
Gestión de Inventarios	Según Mora García (2016), la gestión de inventarios es el proceso de planificación, control y supervisión de las existencias que posee una organización, incluyendo su registro, movimientos de entrada y salida, y disponibilidad.	Registro y Control de Productos	- Datos de productos registrados	- Entrevista
		Movimientos de Inventario	-Método de codificación utilizado - Control de fechas de vencimiento - Registro de entradas (compras) - Registro de salidas (ventas) - Frecuencia de conteo de stock	- Ficha de observación
		Gestión Administrativa	- Control de proveedores - Emisión de comprobantes - Generación de reportes	- Revisión documental

**Tabla 8.** Operacionalización de variables (Variable independiente)

Variable	Definición	Dimensión	Indicador	Instrumento
Sistema Computacional	Según Pressman y Maxim (2021), un sistema computacional es un conjunto de componentes de software integrados que permiten el procesamiento, almacenamiento y gestión automatizada de información para resolver necesidades específicas.	- Módulos de Gestión de Productos	- Registro de productos	- Matriz de evaluación
		-Módulos de Operaciones Comerciales	- Control de stock (Kardex) - Alertas de stock mínimo - Registro de compras - Registro de ventas - Gestión de proveedores	- Ficha técnica
		- Módulos de Administración	- Control de usuarios - Generación de reportes - Auditoría de operaciones	- Requerimientos

**Nota:** La tabla expone la variable independiente y cómo los instrumentos e indicadores se relacionan con este proyecto

### **3.4. MÉTODOS UTILIZADOS**

#### 3.4.1 Métodos

##### 3.4.1.1 Método Descriptivo

Según Shona McCombes (2019), la investigación descriptiva tiene como objetivo "describir de manera clara y ordenada un fenómeno, situación o población". En esta investigación el método descriptivo se utilizó para observar y detallar cómo se gestionan actualmente los inventarios en el Minimarket, el cual, hasta el momento, se maneja de manera manual a través de registros en hojas de papel o en archivos de Excel.

Este método permitió comprender con precisión y orden de cómo se gestionan los inventarios, qué herramientas se utilizan y qué dificultades surgen en el proceso. A partir de esta información se pudo identificar los aspectos que requieren optimización y, con base en ello, desarrollar un sistema informático que facilite y mejore la gestión de inventarios y otros procesos en el Minimarket.

#### 3.4.2 Técnicas

Según, Martínez-Corona, Palacios Almón y Oliva-Garza (2023), La revisión de documentos, es "el proceso de buscar, organizar y analizar información que ya existe en documentos para responder las preguntas de investigación".

Esta técnica se emplea en esta investigación para examinar los registros que ya existen en el Minimarket, incluyendo facturas, hojas de Excel y otros documentos vinculados con el inventario. La meta es comprender la manera en que se están manejando los inventarios en el presente. Esta revisión permitió determinar qué instrumentos se están empleando, de qué manera se documentan los movimientos de inventario y cuáles son los inconvenientes existentes, sin intervenir ni hacer modificaciones en los procesos.

### **3.5. ANÁLISIS ESTADÍSTICO**

Este estudio no emplea métodos estadísticos tradicionales. Los datos cualitativos se analizan mediante fichas de observación y entrevistas para identificar los procesos actuales y sus problemas. Los datos cuantitativos se obtienen de matrices y tablas comparativas que evalúan las herramientas tecnológicas según criterios técnicos. Ambos análisis permiten fundamentar las decisiones de diseño del sistema.

## IV. RESULTADOS Y DISCUSIÓN

### 4.1. RESULTADOS

En el presente capítulo se exponen los resultados obtenidos durante la investigación, organizados en tres fases: análisis de procesos, evaluación técnica de herramientas y diseño del sistema. Cada resultado se fundamenta en los instrumentos aplicados, los cuales se documentan en los anexos correspondientes.

#### 4.1.1 Entrevista

Mediante la aplicación de una entrevista semiestructurada al propietario del Minimarket Otavalo (**ver Anexo 17**), se identificaron los procesos actuales de gestión de inventarios y los principales problemas que afectan la operación del negocio.

En cuanto al registro de productos, el propietario manifestó que utilizaba un cuaderno para anotar los productos al momento de recibirlos y ocasionalmente transfería esta información a hojas de Excel. Respecto al control de entradas, indicó que la recepción de mercadería se realizaba de forma informal, verificando las facturas de los proveedores, pero sin registrar los movimientos en tiempo real.

El registro de ventas presentaba limitaciones significativas, dado que solo se controlaba el efectivo recaudado al final del día sin mantener un detalle de los productos vendidos. El propietario señaló que la reposición de productos se realizaba únicamente cuando observaba que el producto se había agotado en la percha, evidenciando la ausencia de control automatizado del stock.

Se identificaron pérdidas económicas por productos vencidos, debido a que no existía un mecanismo que permitiera conocer con anticipación cuáles productos estaban próximos a caducar, siendo necesario revisar cada producto de forma individual.

Respecto a la gestión de proveedores, se determinó que la información se encontraba dispersa entre contactos del celular, tarjetas de presentación y facturas guardadas en carpetas, lo cual dificultaba el control de fechas de pago y condiciones de crédito.

El propietario expresó la necesidad de contar con un sistema que permitiera integrar las ventas con el inventario en tiempo real, generar alertas de stock mínimo, controlar fechas de vencimiento, emitir comprobantes de forma automática y generar reportes de ventas y productos más vendidos. Adicionalmente, manifestó que el sistema debía ser intuitivo y de fácil navegación, considerando que sus conocimientos tecnológicos eran básicos.

Estos hallazgos fundamentaron los requerimientos funcionales definidos para el sistema y validaron la necesidad de automatizar los procesos de gestión de inventarios del Minimarket Otavalo.

#### 4.1.2 Levantamiento de procesos

Mediante la aplicación de fichas de observación directa en el Minimarket Otavalo, se identificaron los procesos operativos actuales relacionados con la gestión de inventarios. Se observó que el negocio maneja cinco procesos principales: registro de productos, control de entradas por compras a proveedores, registro de salidas por ventas, control de existencias mediante conteo manual y gestión de proveedores. Actualmente, estos procesos se realizan de forma manual mediante hojas de cálculo de Excel y registros en papel, lo que genera retrasos en la actualización de la información y dificultades para obtener datos precisos del stock disponible **(ver Anexo 3)**.

#### 4.1.3 Identificación de problemas y necesidades

A partir del análisis de los procesos observados, se identificaron los principales problemas que afectan la gestión de inventarios del Minimarket como la falta de actualización en tiempo real del stock, errores de transcripción en los registros manuales, dificultad para rastrear el origen de las discrepancias entre el inventario físico y el registrado, ausencia de alertas por stock mínimo que ocasiona quiebres de inventario, y falta de reportes que faciliten la toma de decisiones. Estas deficiencias justifican la necesidad de diseñar un sistema computacional que automatice los procesos críticos y proporcione información confiable **(ver Anexo 4)**.

#### 4.1.4 Requerimientos funcionales y no funcionales

Con base en las necesidades identificadas, se definieron los requerimientos del sistema mediante un checklist validado. Los requerimientos funcionales incluyen: gestión de catálogo de productos con código, nombre, categoría, precio y stock;

registro de compras vinculadas a proveedores; registro de ventas con generación de comprobantes; control de movimientos tipo Kardex con trazabilidad de usuario, fecha y motivo, alertas automáticas por stock mínimo, gestión de roles de usuario como lo son (administrador, gerente, cajero, bodeguero); control de caja con apertura y cierre de sesiones; y generación de reportes de ventas, inventario y movimientos. Los requerimientos no funcionales establecen que el sistema debe mantener tiempos de respuesta inferiores a 3 segundos, garantizar la seguridad mediante autenticación JWT y permitir la operación con hasta 500 conexiones simultáneas **(ver Anexo 5)**.

#### 4.1.5 Evaluación de lenguajes de programación

Para determinar el lenguaje de programación más adecuado, se elaboró una matriz de evaluación técnica comparando JavaScript/TypeScript, PHP, Python y Java. Los criterios evaluados fueron: curva de aprendizaje, rendimiento en aplicaciones web, disponibilidad de documentación, comunidad de soporte y compatibilidad con frameworks modernos. Los datos se obtuvieron de las fichas técnicas oficiales de cada tecnología. JavaScript/TypeScript obtuvo la mayor puntuación debido a su capacidad para utilizarse tanto en frontend como en backend, su amplia documentación, su comunidad activa y su integración nativa con frameworks como Angular y Node.js. Esta característica permite unificar el desarrollo bajo un mismo lenguaje, reduciendo la complejidad del proyecto **(ver Anexo 6)**.

#### 4.1.6 Evaluación de Frameworks

Se evaluaron los frameworks de desarrollo web más utilizados actualmente: Angular, React y Vue.js para el frontend; Express.js, NestJS y Fastify para el backend. La matriz de evaluación consideró criterios como arquitectura, modularidad, curva de aprendizaje, rendimiento, documentación oficial y soporte de la comunidad. Para el frontend, Angular resultó seleccionado por su arquitectura basada en componentes standalone, su sistema de tipado fuerte con TypeScript, su módulo de Material Design integrado y su estructura que facilita el mantenimiento en proyectos medianos. Para el backend, Express.js fue elegido por su simplicidad, flexibilidad, amplia documentación y compatibilidad con middleware de autenticación y validación **(ver Anexo 7)**.

#### 4.1.7 Evaluación de sistemas de base de datos

Mediante una matriz de evaluación técnica se compararon tres sistemas gestores de bases de datos relacionales: MySQL, PostgreSQL y MariaDB. Los criterios analizados fueron: número máximo de conexiones concurrentes, nivel de seguridad, facilidad de instalación, documentación disponible, compatibilidad con el lenguaje seleccionado y costo de licenciamiento. Los datos fueron obtenidos de la documentación oficial de cada proveedor. MySQL obtuvo la puntuación más alta al soportar hasta 500 conexiones simultáneas, ofrecer un nivel de seguridad adecuado mediante cifrado de conexiones y autenticación robusta, contar con amplia documentación en español, ser compatible con Node.js a través del paquete mysql2, y estar disponible bajo licencia gratuita para proyectos comerciales pequeños. Estas características lo hacen apropiado para el contexto del Minimarket Otavalo **(ver Anexo 8)**.

#### 4.1.8 Evaluación de entornos de desarrollo (IDE)

Se evaluaron los entornos de desarrollo integrado más utilizados para proyectos web: Visual Studio Code, WebStorm y Sublime Text. Los criterios de comparación fueron: soporte para TypeScript y JavaScript, extensiones disponibles, integración con control de versiones, depuración integrada, consumo de recursos y costo. Visual Studio Code fue seleccionado por ser gratuito, ofrecer soporte nativo para TypeScript, contar con extensiones específicas para Angular y Node.js, integrar terminal y depurador, y tener bajo consumo de memoria RAM comparado con alternativas como WebStorm. Adicionalmente, su amplia comunidad facilita la resolución de problemas durante el desarrollo **(ver Anexo 9)**.

#### 4.1.9 Evaluación de metodologías de desarrollo

Se compararon tres metodologías ágiles—Extreme Programming (XP), Scrum y Kanban para determinar la más adecuada. Los criterios evaluados fueron: adaptabilidad a equipos pequeños, enfoque en la calidad del código, flexibilidad ante cambios de requerimientos, documentación requerida y facilidad de implementación. XP obtuvo la mayor puntuación debido a su enfoque en entregas frecuentes, programación en pares, refactorización continua y comunicación directa con el cliente. Esta metodología resulta apropiada para el proyecto porque permite incorporar retroalimentación del negocio durante el diseño y facilita ajustes rápidos en los requerimientos sin afectar significativamente el cronograma **(ver Anexo 10)**.

#### 4.1.10 Resultados para el diseño del sistema

##### 4.1.10.1 Arquitectura del sistema

Con base en la evaluación técnica realizada, se definió una arquitectura de tres capas para el sistema: capa de presentación desarrollada con Angular 17 utilizando componentes standalone y Angular Material para la interfaz de usuario; capa de lógica de negocio implementada con Node.js y Express.js que gestiona las reglas del negocio, la autenticación mediante JWT y las validaciones; y capa de datos con MySQL 8.0 que almacena la información de productos, movimientos, usuarios y transacciones. Esta arquitectura permite separar responsabilidades, facilitar el mantenimiento y escalar cada capa de forma independiente según las necesidades futuras del negocio **(ver Anexo 11)**.

##### 4.1.10.2 Modelo de base de datos

Se diseñó el modelo de base de datos relacional compuesto por 26 tablas que cubren todos los módulos del sistema. Las entidades principales son: productos (almacena catálogo con código, nombre, categoría, marca, precios y stock), categorías y marcas (clasificación de productos), proveedores (datos de contacto y RUC), compras y compras\_detalle (registro de adquisiciones), ventas y ventas\_detalle (registro de transacciones con clientes), movimientos\_stock (Kardex con tipo, cantidad, motivo, usuario y fecha), usuarios y roles (control de acceso), caja\_sesiones (apertura y cierre de caja), ubicaciones y bodegas (localización física de productos), lotes (control de productos perecederos con fecha de caducidad), auditoria (registro de cambios para trazabilidad), y config (parámetros del sistema). El modelo incluye las relaciones entre entidades mediante llaves foráneas que garantizan la integridad referencial de los datos **(ver Anexo 12)**.

##### 4.1.10.3 Normalización de la base de datos

El modelo de base de datos fue diseñado aplicando las tres primeras formas normales para garantizar la integridad y evitar redundancia de datos. En la Primera Forma Normal (1FN), se aseguró que cada campo contenga valores atómicos y que cada tabla tenga una clave primaria única. En la Segunda Forma Normal (2FN), se eliminaron las dependencias parciales separando la información en tablas relacionadas; por ejemplo, los datos del proveedor no se repiten en cada compra, sino que se referencian mediante proveedor\_id. En la Tercera Forma Normal (3FN), se eliminaron las dependencias transitivas; por ejemplo, la información de categorías y

marcas se almacena en tablas independientes referenciadas desde productos. Esta normalización reduce el espacio de almacenamiento, facilita las actualizaciones y previene inconsistencias en los datos **(ver Anexo 13)**.

#### 4.1.10.4 Elaboración de historias de usuario

De acuerdo con la metodología XP, se elaboraron las historias de usuario que representan las funciones del sistema desde la perspectiva del usuario final. Se elaboraron historias para cada rol: el administrador puede gestionar usuarios, roles y configuraciones del sistema; el gerente puede consultar reportes de ventas, inventario y movimientos; el cajero puede registrar ventas, aplicar descuentos y gestionar la caja; el bodeguero puede registrar entradas por compras, realizar ajustes de inventario y consultar existencias. Cada historia de usuario incluye: identificador, rol, funcionalidad deseada, criterios de aceptación y prioridad. Por ejemplo: como cajero, quiero registrar una venta seleccionando productos y cantidades, para generar el comprobante y actualizar automáticamente el stock **(ver Anexo 14)**.

#### 4.1.10.5 Diagramas de casos de uso

Se crearon los diagramas de casos de uso a través de notación UML para mostrar las interacciones entre los actores y el sistema. Se identificaron cuatro actores principales: administrador, gerente, cajero y bodeguero. Los casos de uso se agruparon por módulos: módulo de autenticación (iniciar sesión, cerrar sesión, cambiar contraseña), módulo de productos (crear, editar, eliminar, consultar productos), módulo de compras (registrar compra, consultar compras, gestionar proveedores), módulo de ventas (registrar venta, anular venta, consultar ventas), módulo de inventario (consultar stock, registrar ajuste, ver Kardex, gestionar alertas), módulo de caja (abrir caja, cerrar caja, consultar movimientos) y módulo de reportes que tiene generar reporte de ventas, inventario, productos más vendidos. **(ver Anexo 15)**.

#### 4.1.10.6 Diseño de interfaces

Se diseñaron las interfaces del sistema utilizando los componentes de Angular Material para garantizar la consistencia visual y la usabilidad. Las interfaces incluyen: pantalla de inicio de sesión con validación de credenciales; dashboard principal con accesos rápidos según el rol del usuario; módulo de productos con tabla de datos, búsqueda, filtros y formularios de registro; módulo de ventas con selección de productos, cálculo automático de totales e impuestos y generación de comprobante; módulo de compras con registro de proveedor y detalle de productos; módulo de inventario con

vista de Kardex, alertas de stock mínimo y formulario de ajustes; módulo de caja con resumen de apertura, movimientos y cierre; y módulo de reportes con filtros por fecha y exportación de resultados. El diseño sigue los principios de Material Design para facilitar la navegación y reducir la curva de aprendizaje de los usuarios **(ver Anexo 16)**.

Los resultados presentados demuestran que el análisis sistemático de los procesos del negocio y la evaluación técnica fundamentada de las herramientas de desarrollo permiten diseñar un sistema computacional adaptado a las necesidades específicas del Minimarket Otavalo, cumpliendo con los objetivos planteados en la investigación.

## **4.2 PROPUESTA**

Diseñar un sistema de gestión de inventarios para el Minimarket Otavalo en la ciudad de Mira, este proyecto surge de la necesidad de modernizar y optimizar el control de productos dentro del establecimiento. Actualmente, los procesos se realizan de forma manual, lo que genera demoras, errores y falta de precisión en los registros.

### **Objetivo general**

Desarrollar un sistema computacional de gestión de inventarios para el Minimarket "Otavalo" en el cantón Mira en la Provincia del Carchi

### **Objetivos Específico**

- Analizar los requerimientos en el área de gestión de inventario para comprender las funcionalidades que debe tener el sistema computacional.
- Aplicar la metodología XP durante el desarrollo del sistema, siguiendo sus fases de trabajo rápido y pruebas continuas para mejorar el funcionamiento y la calidad del programa.

#### 4.2.1. Factibilidad organizacional

### **Aspectos generales de la organización**

**Institución:** Minimarket Otavalo

**Ubicación geográfica:** Ciudad de Mira, parte céntrica

**Área:** Comercio minorista

**Objeto social:** Servicio al público

### **Misión**

Brindar a la comunidad productos de primera necesidad con calidad, variedad y precios accesibles, asegurando una atención rápida y eficiente. El Minimarket Otavalo busca optimizar sus procesos de gestión mediante el uso de herramientas tecnológicas que permitan mejorar el control de inventarios y garantizar la disponibilidad continua de productos para satisfacer las necesidades de sus clientes.

### Visión

Ser reconocido en la ciudad de Mira como un Minimarket confiable y eficiente, que combina la atención cercana al cliente con procesos tecnológicos avanzados, garantizando un control de inventarios preciso y una gestión operativa moderna que asegure la disponibilidad continua de productos y la satisfacción total de los clientes.

Para llevar a cabo el sistema, se determinaron los diferentes recursos que se emplearán, tanto de software como de hardware, indispensables para su adecuado funcionamiento.

#### 4.2.2. Factibilidad técnica

El sistema que se va a crear incluirá tecnologías como Node.js, Angular y TypeScript para el desarrollo de la aplicación, además de MySQL como sistema de gestión de bases de datos. Estas herramientas fueron elegidas por su flexibilidad, capacidad de escalabilidad y efectividad, lo cual facilita la creación de un sistema moderno y seguro que asegura un control exacto de los inventarios.

**Tabla 9.** Recursos software

Tipo de recurso	Nombre del Recurso	Descripción	Cantidad
Software	Node.js	Entorno de ejecución de JavaScript para el desarrollo del backend	1
	Angular	Framework para el desarrollo del frontend de la aplicación	1
	TypeScript	Lenguaje de programación que mejora JavaScript con tipado estático y escalabilidad	1
	MySQL	Sistema gestor de base de datos relacional	1
	Visual Studio Code	Editor de texto donde se va a depurar el código	1

La solución es estable, rápida y fácil de mantener; funciona en equipos comunes de punto de venta y no depende de dispositivos especiales (el lector de códigos es opcional). Permite crecer por etapas sin rehacer todo, soporta los módulos definidos

(productos, compras, ventas, Kardex, reportes, caja, auditoría y roles) y mantiene la información segura con respaldos y permisos por usuario y la conexión e internet es estable. Además, el costo de operación es bajo y hay suficiente soporte para resolver problemas y seguir mejorándolos.

**Tabla 10.** Recursos de Hardware

Tipo de recurso	Nombre del recurso	Descripción
Hardware	Computadora de escritorio	PC i5/8GB/256SSD para caja y administración.
Hardware	Impresora	Emisión de facturas, archivos pdf
Hardware (opcional)	Lector de códigos	Captura de códigos de barras en ventas.
Hardware	Cajón de dinero	Manejo seguro de efectivo en caja.
Hardware	Router	Red local para terminales y dispositivos.
Hardware (opcional)	Tablet	Consulta de inventario y precios en el perchero

#### 4.2.3. Factibilidad Económica

El análisis económico considera todos los elementos necesarios para poner en marcha y mantener en operación el sistema. Se detallan los costos de hardware y software, el tiempo y esfuerzo de desarrollo, la capacitación del personal y los insumos habituales del punto de venta. Los montos se estiman con precios de mercado vigentes y, cuando es posible, con referencias locales, de modo que el presupuesto refleje una inversión realista para un minimarket.

**Tabla 11.** Costos de hardware

Descripción	Cantidad	Costo real	Costo referencial
Computadora de escritorio	1	0,00	\$500,00
Impresora	1	0,00	\$120,00
Lector de códigos (opcional)	1	0,00	\$60,00
Cajón de dinero	1	0,00	\$55,00
Router	1	0,00	\$45,00
Tablet (opcional)	1	0,00	\$180,00
<b>Total de hardware</b>			<b>\$960,00</b>

**Tabla 12.** Costos de Software

Descripción	Costo real	Costo referencial
Angular	0,00	\$0,00
Node.js	0,00	\$0,00
MySQL	0,00	\$0,00
Visual Studio Code	0,00	\$0,00
<b>Total de software</b>		<b>\$0,00</b>

**Tabla 13. Talento Humano**

Descripción	Cantidad	Costo real	Costo referencial
Desarrollador (análisis, desarrollo, pruebas, despliegue)	1	0,00	\$900,00
<b>Total, de talento humano</b>			<b>\$900,00</b>

**Tabla 14. Materiales de escritorio**

Descripción	Cantidad	Costo real	Costo referencial
Hojas y rollos de Papel	20	0,00	\$35,00
carpetas	—	0,00	\$20,00
Varios (bolígrafos, etiquetas)	—	0,00	\$15,00
<b>Total, de materiales de escritorio</b>			<b>\$70,00</b>

**Tabla 15. Resumen de costos**

Concepto	Monto
Subtotal (hardware + software + talento + materiales)	\$2.230,00
10% de imprevistos	\$323,00
<b>Total</b>	<b>\$2.253,00</b>

#### 4.2.4. Factibilidad Operativa

Se busca agilizar las ventas, mantener el inventario al día y generar evidencia (kardex y auditoría) para la toma de decisiones de compra y el control de caja.

##### 4.2.4.1 Propósito

Diseñar un sistema computacional que permita al Minimarket gestionar sus inventarios de forma ordenada, facilitando el registro de ventas, el control de existencias, el seguimiento de movimientos y la generación de información para la toma de decisiones.

##### 4.2.4.2 Alcance

Incluye: catálogo de productos, compras/recepción, ventas y facturación, inventario/Kardex, alertas y reposición, caja y arqueo, reportes, usuarios y permisos, auditoría completa y respaldos.

No incluye: contabilidad avanzada ni tienda en línea (posibles fases futuras).

##### 4.2.4.3 Situación actual

- **Registros dispersos y actualización tardía del stock.**

Actualmente la información de inventario se anota en cuadernos o archivos separados. Esto hace que el dato exacto no esté en un solo lugar y que la

actualización se haga al final del día o cuando hay tiempo. Entre que se vende y se registra, el stock queda desfasado y se pierde control sobre lo que realmente hay en bodega.

- **Ventas desconectadas del inventario**

La venta no descuenta existencias en el momento del cobro. El cajero entrega el producto, pero el sistema o la hoja de cálculo no reflejan la salida de inmediato. Este desfase produce diferencias cuando se realiza el conteo, y dificulta detectar pérdidas, mermas o errores de facturación.

- **Reposición reactiva sin alertas de mínimos.**

Los pedidos a proveedores se hacen cuando ya falta, sin una alerta que indique que el producto está por agotarse. Esto genera quiebres de stock en artículos de alta rotación y compras de emergencia con peores precios o tiempos. Además, se inmoviliza dinero en productos que no se necesitan mientras faltan los que sí se venden.

- **Caja y comprobantes con poca trazabilidad.**

La emisión de tickets o notas de venta se realiza con herramientas básicas o de forma manual. Luego es difícil vincular cada comprobante con su detalle, usuario y forma de pago. Ante un reclamo, una anulación o una revisión interna, no se cuenta con un rastro claro para confirmar qué ocurrió y quién lo autorizó.

- **Reportes básicos: sin rotación, productos top ni márgenes por periodo.**

Los reportes actuales se limitan a totales generales y no muestran indicadores clave. No se identifica con precisión qué productos lideran las ventas, cuáles se mueven lento, ni el margen por categoría o periodo. Sin esos datos, las decisiones de precio, compra y promoción se toman "a ciegas" y es más fácil cometer errores que afectan la rentabilidad.

#### 4.2.4.4 Operación propuesta con el sistema

- **Venta integrada al inventario:** el stock baja al emitir el comprobante.
- **Kardex y auditoría:** entradas, salidas y ajustes con usuario, fecha, motivo y costo; bitácora de cambios de precio y anulaciones.
- **Unidades de venta claras:** empaquetados por unidad y fraccionables por libra, arroba, quintal.

- **Reposición guiada:** alertas por mínimos y sugerido de compra por rotación.
- **Caja y arqueo:** apertura/cierre y diferencias automáticas.
- **Reportes útiles:** ventas por día/mes, top de productos, rotación y valorización.

#### 4.2.4.5 Roles y responsabilidades

##### **Administrador**

- **Qué hace:** define parámetros del sistema, precios e impuestos, crea/edita usuarios, aprueba acciones sensibles (anulaciones, cambios de precio), configura respaldos y ve todos los reportes y auditorías.
- **Puede:** ver/crear/editar/eliminar en todos los módulos; restaurar respaldos; cambiar permisos.
- **No puede:** nada está bloqueado para este rol (acceso total).
- **Visibilidad:** todo el sistema.

##### **Gerente**

- **Qué hace:** analiza ventas y márgenes, revisa auditorías, valida compras grandes y autoriza anulaciones y cambios de precio cuando se requiera.
- **Puede:** ver reportes completos; ver kardex y auditoría; aprobar compras y ajustes importantes; autorizar anulaciones/cambios de precio; consultar catálogo y stock.
- **No puede:** crear y eliminar usuarios; cambiar parámetros del sistema; ejecutar restauraciones; borrar datos históricos.
- **Visibilidad:** casi todo en modo lectura; edición solo en compras importantes, precios y aprobaciones.

##### **Cajero**

- **Qué hace:** realiza ventas en el POS, emite comprobantes, reimprime cuando corresponde y registra medios de pago.
- **Puede:** buscar productos; vender por unidad / lb / arroba / quintal; emitir y reimprimir comprobantes; consultar precios y stock; ver su propio historial de ventas.
- **No puede:** crear/editar productos; registrar compras; hacer ajustes de inventario; ver auditorías; cambiar precios o impuestos; anular ventas sin autorización del Gerente o Administrador.
- **Visibilidad:** POS de ventas, consulta de productos y su historial de ventas.

### **Bodeguero:**

- **Qué hace:** recibe compras, registra movimientos internos, realiza conteos cíclicos y propone reposición.
- **Puede:** registrar entradas de mercadería (total o parcial); hacer traslados entre ubicaciones; cargar ajustes por merma/rotura con motivo; generar sugerencias de reposición; consultar kardex.
- **No puede:** emitir ventas; modificar precios/impuestos; autorizar anulaciones; eliminar movimientos cerrados; crear usuarios.
- **Visibilidad:** compras/recepción, inventario/kardex, conteos y reportes básicos de stock.

#### 4.2.4.6 Flujos operativos clave

##### **Apertura de caja**

1. Registrar fondo inicial.
2. Verificar conexión, impresora y papel térmico.

##### **Venta / Facturación**

1. Buscar o escanear producto.
2. Elegir unidad de venta (unidad o lb/arroba/quintal si aplica).
3. Emitir comprobante; el stock se descuenta automáticamente.

##### **Compras / Entradas**

1. Registrar proveedor y factura de compra (total o parcial).
2. Actualizar costos y existencias; generar movimiento en kardex.

##### **Reposición**

1. Revisar alertas por mínimos.
2. Generar sugerido de compra por rotación y stock disponible.

##### **Cierre de caja**

1. Calcular ventas del día por forma de pago.
2. Conciliar con efectivo real y registrar diferencias.

#### 4.2.4.7 Políticas operativas

1. Precios, impuestos y anulaciones: Gerente/Administrador.
2. Todo ajuste con motivo y en auditoría.

3. Conteos: semanal para alta rotación; mensual para el resto.
4. Respaldo automático diario y uno semanal externo (USB/nube).

#### 4.2.4.8 Recursos y capacitación

1. Equipo: PC de escritorio con impresora térmica, lector de código (opcional) y cajón de dinero; router estable.
2. Capacitación: 4–6 horas (venta, entradas, ajustes, cierre de caja y reportes).
3. Manual resumido: pasos de venta, correcciones y contingencia.

#### 4.2.5 Metodología XP

##### 4.2.5.1 Fase de planificación

En nuestro proyecto de gestión de inventario para un Minimarket, la planificación sirve para poner en sintonía al dueño del negocio y al equipo, ordenar qué módulos van primero y armar un plan de trabajo realista en iteraciones cortas. Para este proyecto se priorizan: Ventas y Facturación, Inventario/Kardex, Compras/Recepción y Caja; con ello se crea una base sólida para una ejecución ágil y efectiva, que permita ajustar el rumbo ante cambios del día a día del Minimarket, como variaciones de stock, precios, promociones o tiempos de proveedores, sin perder control ni tiempo.

##### 4.2.5.2 Módulos del sistema

**Tabla 16.** Módulos del sistema

N.º	Módulo	Objetivo	Entregable (propuesta)	Prioridad	Iteración
1	Autenticación (Login/JWT)	Acceso seguro y protección de rutas.	Flujo de inicio de sesión + guard/interceptor.	Alta	11
2	Usuarios y Roles	Gestionar cuentas y permisos (ADMIN, GERENTE, BODEGUERO, CAJERO).	CRUD de usuarios + matriz de permisos por módulo.	Alta	11
3	Productos	Registrar y mantener productos (unidad, precios, stock mínimo).	CRUD de productos + búsqueda/autocompletar.	Alta	11
4	Categorías	Organizar productos por categoría para filtros y reportes.	CRUD de categorías.	Media	11
5	Marcas	Clasificar productos por marca.	CRUD de marcas.	Media	11
6	Clientes	Administrar datos de clientes para ventas.	CRUD de clientes + validación de documento.	Media	11
7	Proveedores	Registrar proveedores para compras.	CRUD de proveedores.	Media	11
8	Compras	Ingresar compras y aumentar stock por bodega.	Flujo de compra + detalle de productos + aumento de stock.	Alta	12
9	Ventas (POS)	Facturar rápido y descontar stock.	Flujo POS con folio, métodos de pago y descuento de stock.	Alta	12

10	Inventario Kardex /	Ver existencias y trazabilidad (entradas, salidas y ajustes).	Consulta de inventario + lista de movimientos por producto.	Alta	12
11	Ubicaciones	Asignar ubicación física (sección-pasillo-estante-nivel).	CRUD de ubicaciones + selector de productos.	Alta	12
12	Reportes	Obtener ventas, compras e inventario con exportación.	Reportes básicos con exportación a Excel/PDF.	Media	13
13	Dashboard	Ver KPIs: ventas hoy, compras hoy, bajo stock, valor inventario.	Tarjetas de KPIs y gráfico de ventas.	Media	13
14	Auditoría	Verificar el stock físico contra el stock registrado en el sistema para detectar diferencias	Formulario de conteo con listado de productos, campo para cantidad física, cálculo automático de diferencias	Alta	13

#### 4.2.6 Historias de usuario

**Tabla 17.** Historia de usuario 1

HISTORIA DE USUARIO	
<b>Número:</b> 1	<b>Usuario:</b> Cajero
<b>Nombre historia:</b> Búsqueda y selección de productos	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Bajo
<b>Estimación:</b> 1	<b>Iteración:</b> 1
<b>Responsable:</b> Israel Salazar (desarrollador)	
<b>Descripción:</b> Cajero, necesito buscar productos por nombre o código para agregarlos rápido a la venta sin perder tiempo con listas largas.	
<b>Detalle:</b> El sistema debe filtrar desde las primeras 3 letras y mostrar precio y stock en la misma vista. Debe permitir agregar con un clic y continuar con la siguiente búsqueda. La consulta no debe tardar más de 2 segundos en condiciones normales. Si el producto no existe, debe indicarlo claramente y ofrecer la opción de avisar al Administrador.	

**Tabla 18.** Historia de usuario 2

HISTORIA DE USUARIO	
<b>Número:</b> 2	<b>Usuario:</b> Cajero
<b>Nombre historia:</b> Venta por unidad o por fracción	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Medio
<b>Estimación:</b> 1	<b>Iteración:</b> 1
<b>Responsable:</b> Israel Salazar (desarrollador)	
<b>Descripción:</b> Como Cajero, necesito vender por unidad o por libra/arroba/quintal cuando el producto lo permita, para cobrar correctamente.	
<b>Detalle:</b> El selector de unidad solo aparece en productos fraccionables. El sistema calcula precio y descuenta stock según la unidad elegida. Si se intenta una unidad no válida, muestra un mensaje claro y no permite continuar.	

**Tabla 19.** Historia de usuario 3

---

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 3	<b>Usuario:</b> Cajero
<b>Nombre historia:</b> Emisión de comprobante (ticket/factura)	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Bajo
<b>Estimación:</b> 1	<b>Iteración:</b>
<b>Responsable:</b> Israel Salazar (desarrollador)	
<b>Descripción:</b> Como Cajero, necesito emitir un comprobante con numeración, IVA y forma de pago para entregar al cliente su respaldo.	
<b>Detalle:</b> Permite imprimir en impresora térmica o descargar PDF. Registra fecha, cajero, totales e ítems. Habilita reimpresión controlada y muestra mensajes simples ante errores.	

---

**Tabla 20.** Historia de usuario 4

---

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 4	<b>Usuario:</b> Bodeguero / Administrador
<b>Nombre historia:</b> Registro de compras y proveedores	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Medio
<b>Estimación:</b> 2	<b>Iteración:</b> 2
<b>Responsable:</b> Israel Salazar (desarrollador)	
<b>Descripción:</b> Como Bodeguero/Administrador, necesito registrar compras con su proveedor para actualizar existencias y costos.	
<b>Detalle:</b> Soporta recepción total o parcial. Actualiza costo promedio y genera entrada en Kardex. Vincula documento del proveedor y deja notas internas.	

---

**Tabla 21.** Historia de usuario 5

---

<b>Historia de Usuario</b>	
<b>Número:</b> 5	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Kardex y bitácora de movimientos	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Medio
<b>Estimación:</b> 2	<b>Iteración:</b> 2
<b>Responsable:</b> Israel Salazar (desarrollador)	
<b>Descripción:</b> Como Administrador, necesito consultar Kardex y bitácora para auditar entradas, salidas y ajustes por usuario y fecha.	
<b>Detalle:</b> Cada movimiento guarda cantidad, costo, saldo y referencia. La bitácora registra altas, ediciones, eliminaciones, anulaciones y reimpressiones. Exporta a PDF/Excel y filtra por producto/fecha/responsable.	

---

**Tabla 22.** Historia de usuario 6

---

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 6	<b>Usuario:</b> Gerente
<b>Nombre historia:</b> Alertas de mínimo y reposición sugerida	
<b>Prioridad:</b> Media	<b>Riesgo:</b> Bajo
<b>Estimación:</b> 1	<b>Iteración:</b> 3
<b>Responsable:</b> Israel Salazar (desarrollador)	
<b>Descripción:</b> Como Gerente, quiero recibir alertas cuando un producto llegue al mínimo y ver una lista sugerida de compra.	
<b>Detalle:</b> Define umbral por producto. Calcula sugeridos con base en rotación y ventas recientes. Muestra las alertas en panel y puede enviar correo si se configura. Guarda historial de revisión.	

---



**Tabla 28.** Historia de usuario 12

---

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 12	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Respaldos y restauración	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Bajo
<b>Estimación:</b> 1	<b>Iteración:</b> 4
<b>Responsable:</b> Israel Salazar (desarrollador)	
<b>Descripción:</b> Como administrador, necesito generar copias de seguridad y poder restaurarlas para proteger la información.	
<b>Detalle:</b> Programa de backup externo diario y semanal (USB/nube). La restauración se prueba en entorno seguro. Todas las acciones quedan registradas en la bitácora.	

---

**Tabla 29.** Historia de usuario 13

---

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 13	<b>Usuario:</b> Bodeguero / Administrador
<b>Nombre historia:</b> Gestión de ubicaciones y asignación a productos	
<b>Prioridad:</b> Media	<b>Riesgo:</b> Bajo
<b>Estimación:</b> 1	<b>Iteración:</b> 2
<b>Responsable:</b> Israel Salazar (desarrollador)	
<b>Descripción:</b> Como Bodeguero/Admin, necesito crear ubicaciones (sección–pasillo–estante–nivel) y asignarlas a productos.	
<b>Detalle:</b> Código único (p. ej., A1-E1-N1); no permite eliminarlo si está en uso; auditoría completa; filtro por ubicación en el inventario.	

---

**Tabla 30.** Historia de usuario 14

---

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 14	<b>Usuario:</b> Gerente
<b>Nombre historia:</b> Dashboard con KPIs	
<b>Prioridad:</b> Media	<b>Riesgo:</b> Bajo
<b>Estimación:</b> 1	<b>Iteración:</b> 3
<b>Responsable:</b> Israel Salazar (desarrollador)	
<b>Descripción:</b> Como Gerente, quiero ver KPIs clave en un dashboard (ventas hoy, compras hoy, productos con stock bajo, valor de inventario).	
<b>Detalle:</b> Tarjetas con números; gráfico de ventas del mes; lista de bajo stock; refrescar datos con un clic.	

---

**Tabla 31.** Historia de usuario 15

---

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 15	<b>Usuario:</b> Gerente / Administrador
<b>Nombre historia:</b> Reportes y exportaciones	
<b>Prioridad:</b> Media	<b>Riesgo:</b> Bajo
<b>Estimación:</b> 1	<b>Iteración:</b> 4
<b>Responsable:</b> Israel Salazar (desarrollador)	
<b>Descripción:</b> Como Gerente/Admin, necesito reportes de ventas, compras e inventario con exportación a Excel/PDF.	
<b>Detalle:</b> Filtros por fechas/proveedor/cliente; top productos vendidos; reporte de bajo stock; exportación con los filtros aplicados.	

---

#### 4.2.7 Tareas de usuario

**Tabla 32.** Tarea de usuario 1

---

<b>Tarea de usuario</b>	
Número de la tarea: 1	Número de historia: H01
Nombre de tarea: Autocompletado y búsqueda rápida de productos	
Tipo de tarea: Frontend	Puntos estimados: 3
Fecha inicio: 01/07/2025	Fecha fin: 07/07/2025
Programador responsable: Israel Salazar	
Descripción: Implementar un campo de búsqueda con sugerencias a partir de 3 letras que muestre el precio y el stock y permita agregar al carrito con un clic.	

---

**Tabla 33.** Tarea de usuario 2

---

<b>Tarea de usuario</b>	
Número de la tarea: 2	Número de historia: H01
Nombre de tarea: API de búsqueda con índices por nombre/código	
Tipo de tarea: Backend	Puntos estimados: 3
Fecha inicio: 08/07/2025	Fecha fin: 14/07/2025
Programador responsable: Israel Salazar	
Descripción: Crear endpoint GET /api/productos?search= con paginación e índices en MySQL para respuestas de hasta 2 segundos.	

---

**Tabla 34.** Tarea de usuario 3

---

<b>Tarea de usuario</b>	
Número de la tarea: 3	Número de historia: H02
Nombre de tarea: Selector de unidad/fracción en POS	
Tipo de tarea: Frontend	Puntos estimados: 2
Fecha inicio: 15/07/2025	Fecha fin: 21/07/2025
Programador responsable: Israel Salazar	
Descripción: Mostrar selector solo en productos fraccionables y validar unidades no permitidas con mensajes claros.	

---

**Tabla 35.** Tarea de usuario 4

---

<b>Tarea de usuario</b>	
Número de la tarea: 4	Número de historia: H02
Nombre de tarea: Conversión y descuento de stock por fracción	
Tipo de tarea: Backend	Puntos estimados: 3
Fecha inicio: 22/07/2025	Fecha fin: 31/07/2025
Programador responsable: Israel Salazar	
Descripción: Aplicar las reglas de conversión de unidades y de fracciones, recalcular los totales y actualizar las existencias.	

---

**Tabla 36.** Tarea de usuario 5

---

<b>Tarea de usuario</b>	
Número de la tarea: 5	Número de historia: H03
Nombre de tarea: Emisión de comprobante PDF/Térmico	
Tipo de tarea: Frontend	Puntos estimados: 3
Fecha inicio: 01/08/2025	Fecha fin: 07/08/2025
Programador responsable: Israel Salazar	
Descripción: Plantillas de ticket/factura con IVA y forma de pago; descarga en PDF e impresión térmica.	

---

**Tabla 37.** Tarea de usuario 6

---

**Tarea de usuario**

Número de la tarea: 6 Número de historia: H03  
Nombre de tarea: Registro de venta y numeración  
Tipo de tarea: Backend Puntos estimados: 3  
Fecha inicio: 08/08/2025 Fecha fin: 14/08/2025  
Programador responsable: Israel Salazar  
Descripción: Endpoint POST /api/ventas con serie/folio, ítems, cajero, totales y control de reimpressiones.

---

**Tabla 38.** Tarea de usuario 7

---

**Tarea de usuario**

Número de la tarea: 7 Número de historia: H04  
Nombre de tarea: CRUD de proveedores y registro de compras  
Tipo de tarea: Backend Puntos estimados: 3  
Fecha inicio: 15/08/2025 Fecha fin: 21/08/2025  
Programador responsable: Israel Salazar  
Descripción: Endpoints y modelo para compras totales y parciales vinculadas a proveedor con adjuntos.

---

**Tabla 39.** Tarea de usuario 8

---

**Tarea de usuario**

Número de la tarea: 8 Número de historia: H04  
Nombre de tarea: UI de compras y actualización de costo promedio  
Tipo de tarea: Full-stack Puntos estimados: 3  
Fecha inicio: 22/08/2025 Fecha fin: 31/08/2025  
Programador responsable: Israel Salazar  
Descripción: Formulario de compra, recepción parcial y cálculo de costo promedio con entrada automática en Kardex.

---

**Tabla 40.** Tarea de usuario 9

---

**Tarea de usuario**

Número de la tarea: 9 Número de historia: H05  
Nombre de tarea: Consulta de Kardex y bitácora con exportación  
Tipo de tarea: Full-stack Puntos estimados: 3  
Fecha inicio: 01/09/2025 Fecha fin: 07/09/2025  
Programador responsable: Israel Salazar  
Descripción: Vistas con filtros por producto, fecha y responsable, y exportación a PDF y Excel; API paginada.

---

**Tabla 41.** Tarea de usuario 10

---

**Tarea de usuario**

Número de la tarea: 10 Número de historia: H06  
Nombre de tarea: Panel de alertas de mínimos y sugeridos  
Tipo de tarea: Full-stack Puntos estimados: 3  
Fecha inicio: 08/09/2025 Fecha fin: 14/09/2025  
Programador responsable: Israel Salazar  
Descripción: Configurar umbrales por producto, calcular los sugeridos por rotación y el panel con historial.

---

**Tabla 42. Tarea de usuario 11**

---

**Tarea de usuario**

Número de la tarea: 11 Número de historia: H07  
Nombre de tarea: Cierre de caja y arqueo diario  
Tipo de tarea: Full-stack Puntos estimados: 3  
Fecha inicio: 15/09/2025 Fecha fin: 24/09/2025  
Programador responsable: Israel Salazar  
Descripción: Totales por forma de pago, observaciones, documento de arqueo y registro en auditoría.

---

**Tabla 43. Tarea de usuario 12**

---

**Tarea de usuario**

Número de la tarea: 12 Número de historia: H08  
Nombre de tarea: Movimientos internos y conteos cíclicos  
Tipo de tarea: Full-stack Puntos estimados: 3  
Fecha inicio: 25/09/2025 Fecha fin: 01/10/2025  
Programador responsable: Israel Salazar  
Descripción: Traslados de bodega a góndola, conteos por categoría y rotación, y ajustes con motivo y responsable.

---

**Tabla 44. Tarea de usuario 13**

---

**Tarea de usuario**

Número de la tarea: 13 Número de historia: H09  
Nombre de tarea: Flujo de anulación con autorización y reverso  
Tipo de tarea: Full-stack Puntos estimados: 3  
Fecha inicio: 02/10/2025 Fecha fin: 08/10/2025  
Programador responsable: Israel Salazar  
Descripción: Solicitud de motivo y validación por rol; reversión en Kardex y caja; registro completo en bitácora.

---

**Tabla 45. Tarea de usuario 14**

---

**Tarea de usuario**

Número de la tarea: 14 Número de historia: H10  
Nombre de tarea: Gestión de precios y promociones  
Tipo de tarea: Full-stack Puntos estimados: 3  
Fecha inicio: 09/10/2025 Fecha fin: 15/10/2025  
Programador responsable: Israel Salazar  
Descripción: Precios por unidad/fracción y reglas de promoción vigentes y aplicación en POS.

---

**Tabla 46. Tarea de usuario 15**

---

**Tarea de usuario**

Número de la tarea: 15 Número de historia: H11  
Nombre de tarea: Usuarios, roles y permisos por módulo  
Tipo de tarea: Full-stack Puntos estimados: 3  
Fecha inicio: 16/10/2025 Fecha fin: 22/10/2025  
Programador responsable: Israel Salazar  
Descripción: CRUD de usuarios, asignación de roles y auditoría de cambios.

---

**Tabla 47. Tarea de usuario 16**

---

**Tarea de usuario**

Número de la tarea: 16 Número de historia: H12  
Nombre de tarea: Respaldos y restauración  
Tipo de tarea: Full-stack Puntos estimados: 3  
Fecha inicio: 23/10/2025 Fecha fin: 29/10/2025  
Programador responsable: Israel Salazar  
Descripción: Programar backup diario/semanal y restauración en entorno seguro; registrar toda acción en bitácora.

---

**Tabla 48.** Tarea de usuario 17

<b>Tarea de usuario</b>	
Número de la tarea:	17
Número de historia:	H03
Nombre de tarea:	Configuración de impresora térmica 80 mm y formato
Tipo de tarea:	Frontend
Puntos estimados:	2
Fecha inicio:	30/10/2025
Fecha fin:	08/11/2025
Programador responsable:	Israel Salazar
Descripción:	Ajustar el tamaño de papel, los márgenes y las pruebas de impresión para ticket/factura; manejo de cortes.

**Tabla 49.** Tarea de usuario 18

<b>Tarea de usuario</b>	
Número de la tarea:	18
Número de historia:	H05
Nombre de tarea:	Exportación PDF/Excel desde interfaz de reportes
Tipo de tarea:	Frontend
Puntos estimados:	2
Fecha inicio:	09/11/2025
Fecha fin:	12/11/2025
Programador responsable:	Israel Salazar
Descripción:	Agregar botones de exportación, plantillas y nombres de archivo; realizar pruebas con grandes volúmenes.

**Tabla 50.** Tarea de usuario 19

<b>Tarea de usuario</b>	
Número de la tarea:	19
Número de historia:	H11
Nombre de tarea:	Interceptor JWT y guard por rol en Angular
Tipo de tarea:	Frontend
Puntos estimados:	3
Fecha inicio:	13/11/2025
Fecha fin:	14/11/2025
Programador responsable:	Israel Salazar
Descripción:	Proteger rutas por rol (Administrador, Gerente, Cajero, Bodeguero), refrescar los tokens y manejar la expiración.

**Tabla 51.** Tarea de usuario 20

<b>Tarea de usuario</b>	
Número de la tarea:	20
Número de historia:	H12
Nombre de tarea:	Script de backup/restore y programación
Tipo de tarea:	Backend
Puntos estimados:	2
Fecha inicio:	15/11/2025
Fecha fin:	15/11/2025
Programador responsable:	Israel Salazar
Descripción:	Scripts para MySQL (dump/restore) y para agendar tareas; verificación de la integridad y de los logs de ejecución.

#### 4.2.8 Requerimientos de software

**Tabla 52.** Requerimientos Funcionales

ID	Nombre	Requisitos
RF-01	Login y roles	Autenticar usuarios y asignar permisos por rol (Admin, Gerente, Cajero, Bodeguero).
RF-02	CRUD Productos	Crear, ver, editar y eliminar productos con código, nombre, unidad/fracción y estado.
RF-03	Stock inicial	Registrar stock inicial por producto y ubicación (bodega/góndola).
RF-04	Compras	Ingresar compras (total/parcial), proveedor y costos; actualizar Kardex.
RF-05	Venta POS	Vender por unidad o fracción, calcular totales/IVA y descontar stock.
RF-06	Comprobantes	Generar ticket/factura en PDF e impresión térmica; reimpresión controlada.
RF-07	Kardex	Mostrar entradas, salidas y saldos por producto y fecha, con referencia al movimiento.
RF-08	Bitácora	Registrar altas, ediciones, anulaciones y reimpressiones con usuario, fecha y hora.

RF-09	Alertas de mínimos	Notificar productos con stock mínimo y listar sugerencias de reposición.
RF-10	Proveedores	CRUD de proveedores y asociación con compras y costos.
RF-11	Precios	Gestionar precios por unidad y por fracción.
RF-12	Cierre de caja	Arqueo diario por forma de pago, registro de diferencias y observaciones.
RF-13	Movimientos internos	Registrar traslados entre ubicaciones y ajustes con motivo y responsable.
RF-14	Reportes	Reportes de ventas, rotación, valorización y top productos con filtros.
RF-15	Exportación	Exportar reportes, Kardex y bitácora a PDF y Excel.
RF-16	Anulaciones controladas	Requerir autorización de rol para anular ventas y revertir efectos en Kardex y caja, manteniendo trazabilidad.

**Tabla 53.** Requerimientos No Funcionales

ID	Nombre	Requisito
RNF-01	Rendimiento búsquedas	Las búsquedas de productos deben responder en $\leq 2$ s y devolver hasta 10 000 productos.
RNF-02	Rendimiento POS	Agregar un ítem y recalcular los totales debe tardar $\leq 1$ s.
RNF-03	Rendimiento reportes	Reportes (ventas, rotación, valorización) de 12 meses en $\leq 5$ s.
RNF-04	Concurrencia	Permitir hasta 5 cajeros a la vez sin errores ni ventas duplicadas.
RNF-05	Disponibilidad	$\geq 99$ % en horario 08:00–21:00.
RNF-06	Integridad de datos	Usar transacciones en ventas y compras para que operación y Kardex sean atómicos.
RNF-07	Seguridad de acceso	Autenticación JWT, contraseñas con bcrypt, cierre por inactividad a los 20 min.
RNF-08	Autorización por rol	Acciones sensibles (anular, ver costos, cambiar precios) solo por rol; negarlas por defecto.
RNF-09	Auditoría	Bitácora de alta/edición/eliminación/anulación/reimpresión con usuario, fecha y referencia; retención de 12 meses.
RNF-10	Usabilidad	Interfaz simple; flujos del cajero en $\leq 3$ clics.
RNF-11	Compatibilidad	Navegadores Chrome/Edge en Windows 10/11, resolución $\geq 1366 \times 768$ .
RNF-12	Escalabilidad ligera	Hasta 20 000 productos y 1 500 ventas/día sin cambiar la arquitectura.
RNF-13	Internacionalización	Moneda: USD, IVA configurable (por defecto: 15 %), fecha: dd/mm/aaaa.
RNF-14	Mantenibilidad	Código comentado y README; pruebas básicas para totales e IVA.
RNF-15	Exportación confiable	Descargas de PDF/Excel sin errores y con encabezados/totales/filtros correctos.
RNF-16	Protección de datos	Costos y márgenes visibles solo para Admin/Gerente; ocultarlos en la UI y en la API para otros roles.

**Tabla 54.** Tiempos estimados de tareas de usuario

Nombre Historia	N.º Tarea	Tarea	Semanas	Días	Horas
<b>Ventas POS y Facturación</b>	1	Búsqueda/selección de productos en POS	1	5	40
	2	Lógica de venta por unidad y fracción	1	5	40
	3	Emisión de comprobante (PDF/térmica)	1	5	40
<b>Catálogo de Productos</b>	4	CRUD backend de productos (modelo + endpoints)	1	5	40
	5	UI de productos (ABM con filtros)	1	5	40
<b>Inventario y Kardex</b>	6	Registro de entradas/salidas/ajustes (actualiza saldos)	1	5	40
	7	Consulta Kardex con filtros y saldos	1	5	40
<b>Compras y Proveedores</b>	8	Registro de compras total/parcial con documento	1	5	40
	9	Actualización de costo promedio + vínculo a Kardex	1	5	40
	10	CRUD de proveedores y asociación en compras	0.5	2.5	20
<b>Caja y Arqueo</b>	11	Cierre diario, diferencias y observaciones	1	5	40
	12	Configurar mínimos + panel de alertas	1	5	40
<b>Alertas y Reposición</b>	13	Login/JWT y gestión de sesiones	1	5	40
	14	Matriz de permisos por rol y módulo	1	5	40
<b>Usuarios y Roles</b>	15	Gestión de precios por unidad/fracción	0.5	2.5	20
	16	Reglas simples de promociones (vigencia/porcentaje)	0.5	2.5	20
<b>Reportes</b>	17	Reportes: ventas y rotación con filtros	1.5	7.5	60
	18	Reportes: valorización y top productos	1	5	40
	19	Exportación a PDF/Excel (reportes/Kardex/bitácora)	0.5	2.5	20
<b>Auditoría y Anulaciones</b>	20	Flujo de anulación con autorización + reversa en Kardex/Caja y bitácora completa	1.0	5	40
<b>Total de tiempo</b>			<b>19.5</b>	<b>97.5</b>	<b>780</b>

**Tabla 55.** Plan de entrega del proyecto

Módulo	Nro.	Nombre de historia	Semanas	Días	Horas	Iteración 1	Iteración 2	Entrega 1	Entrega 2
<b>Ventas (POS/Facturación)</b>	1	POS: búsqueda, fracciones y comprobante	3	15	120	X		X	
<b>Catálogo de Productos</b>	2	CRUD de productos	2	10	80	X			X
<b>Inventario/Kardex</b>	3	Movimientos + consulta Kardex	2	10	80	X			X
<b>Compras y Proveedores</b>	4	Registro de compras y costos	2.5	12.5	100		X	X	
<b>Caja y Arqueo</b>	5	Cierre diario y diferencias	1	5	40		X		X
<b>Alertas y Reposición</b>	6	Mínimos y sugeridos	1	5	40		X		X
<b>Usuarios y Roles</b>	7	Login y permisos por rol	2	10	80	X		X	
<b>Precios</b>	8	Gestión de precios y promociones	1	5	40		X		X
<b>Reportes</b>	9	Ventas/rotación/valorización + exportación	3	15	120		X		X
<b>Auditoría y Anulaciones</b>	10	Anulación con autorización + bitácora	1	5	40		X		X
<b>Movimientos internos</b>	11	Traslados/ajustes con motivo	0.5	2.5	20	X			X
<b>Total</b>									<b>21 semanas</b>

#### 4.2.9 Fase de Diseño

En la metodología XP (Xtreme Programming), el diseño debe ser simple y listo para cambiar cuando el negocio lo pida. Ford, Parsons y Kua explican que un diseño evolutivo, se basa en módulos claros, pruebas automáticas y refactorización frecuente para aceptar cambios sin romper nada. En este caso eso significa mantener separados los componentes de POS, Compras, Kardex y Reportes, con interfaces limpias (APIs) entre Angular y el backend en Node, para mejorar cada parte sin tocar todo el sistema.

Farley señala que un buen diseño es el que reduce el costo del cambio con el tiempo: TDD, integración continua y despliegues pequeños ayudan a detectar errores rápidamente y a ajustar el rumbo. Esto se traduce en: pruebas de servicios (como, por ejemplo, ventas y actualización de stock), pipelines simples de build/test y entregas en pasos cortos (p. ej., primero venta por unidad, luego venta fraccionada, después promociones). Así, la arquitectura en Node + MySQL se mantiene estable mientras se agregan funciones al frontend de Angular.

Además, el diseño en XP es colaborativo y se guía por las historias de usuario. Skelton y Pais proponen organizar el trabajo para que la arquitectura y los equipos fluyan mejor mediante componentes de bajo acoplamiento, límites claros, como ventas, compras, inventario y decisiones arquitectónicas registradas de forma ligera. En tu proyecto, esto ayuda a alinear el diseño con historias como Kardex y bitácora, cierre de caja o alertas de mínimos, validando cada iteración con criterios de aceptación antes de programar.

##### 4.2.9.1 Tarjetas CRC

**Tabla 56.** Tarjeta CRC Usuario/Rol

TARJETA CRC
<b>Nombre:</b> Usuario/Rol
<b>Responsabilidades:</b> <ul style="list-style-type: none"><li>- Autenticar usuario (login).</li><li>- Autorizar el acceso por rol (Admin, Gerente, Cajero, Bodeguero).</li><li>- Administrar permisos por módulo.</li></ul>
<b>Colaboradores:</b> <ul style="list-style-type: none"><li>- Clase BaseDatos (guardar/actualizar usuarios y roles).</li><li>- Clase Bitácora (para registrar accesos y cambios).</li></ul>

**Tabla 57. Tarjeta CRC Producto**

---

<b>TARJETA CRC</b>
<b>Nombre:</b> Producto
<b>Responsabilidades:</b>
- Crear y actualizar datos (código, nombre, unidad/fracción, estado).
- Consultar el stock disponible y el precio vigente.
- Validar si es fraccionable.
<b>Colaboradores:</b>
- Clase Precio (precio por unidad/fracción).
- Clase Kardex (movimientos y saldo).
- Clase Base Datos (persistencia).

---

**Tabla 58. Tarjeta CRC StockInicial**

---

<b>TARJETA CRC</b>
<b>Nombre:</b> StockInicial
<b>Responsabilidades:</b>
- Registrar el stock inicial por producto y ubicación (bodega/góndola).
- Validar duplicidades y consistencia.
- Generar el asiento inicial en el Kardex.
<b>Colaboradores:</b>
- Clase Producto (referencia).
- Clase Kardex (movimiento inicial).
- Clase BaseDatos (persistencia).

---

**Tabla 59. Tarjeta CRC Proveedor**

---

<b>TARJETA CRC</b>
<b>Nombre:</b> Proveedor
<b>Responsabilidades:</b>
- Registrar y actualizar los datos del proveedor.
- Controlar estado (activo/inactivo).
- Asociar compras al proveedor.
<b>Colaboradores:</b>
- Clase Compra (asociación).
- Clase BaseDatos (persistencia).

---

**Tabla 60. Tarjeta CRC Compra**

---

<b>TARJETA CRC</b>
<b>Nombre:</b> Compra
<b>Responsabilidades:</b>
- Registrar compra total o parcial.
- Actualizar el costo promedio del producto.
- Generar entrada en Kardex.
<b>Colaboradores:</b>
- Clase Proveedor (origen de la compra).
- Clase Producto (ítems).
- Clase Kardex (movimiento de entrada).
- Clase Bitácora (alta/edición/anulación).

---

**Tabla 61.** Tarjeta CRC Venta

---

**TARJETA CRC**

---

**Nombre:** Venta

**Responsabilidades:**

- Buscar y añadir productos (unidad o fracción).
- Calcular subtotales, IVA y total.
- Descontar stock y generar salida en Kardex.

**Colaboradores:**

- Clase Producto (validaciones y stock).
  - Clase Precio (cálculo por unidad/fracción).
  - Clase Kardex (movimiento de salida).
  - Clase Comprobante (emisión).
  - Clase Bitácora (registro).
  - Clase BaseDatos (persistencia).
- 

**Tabla 62.** Tarjeta CRC Comprobante

---

**TARJETA CRC**

---

**Nombre:** Comprobante

**Responsabilidades:**

- Generar ticket/factura (PDF/impresora térmica).
- Controlar numeración y reimpressiones.
- Registrar forma de pago.

**Colaboradores:**

- Clase VentaPOS (origen de ítems/valores).
  - Clase BaseDatos (guardar comprobante).
  - Clase Bitácora (reimpressiones/anulaciones).
- 

**Tabla 63.** Tarjeta CRC Kardex

---

**TARJETA CRC**

---

**Nombre:** Kardex

**Responsabilidades:**

- Registrar entradas, salidas y saldos por producto/fecha.
- Mantener referencia al movimiento (compra, venta, ajuste, traslado).
- Consultar historial con filtros.

**Colaboradores:**

- Clase Compra, VentaPOS, MovInterno, StockInicial (asientos).
  - Clase BaseDatos (persistencia).
  - Clase Reporte (consulta).
- 

**Tabla 64.** Tarjeta CRC Bitácora

---

**TARJETA CRC**

---

**Nombre:** Bitácora

**Responsabilidades:**

- Registrar altas, ediciones, eliminaciones, anulaciones y reimpressiones.
- Guardar quién, cuándo, qué y el motivo (si aplica).
- Consultar por usuario/fecha/módulo.

**Colaboradores:**

- Clase Usuario/Rol (responsable).
  - Clases Compra, VentaPOS, Comprobante, Precios, Anulación (eventos).
  - Clase BaseDatos (persistencia).
-

**Tabla 65.** Tarjeta CRC AlertasMinimos

---

**TARJETA CRC**

---

**Nombre:** AlertasMinimos

**Responsabilidades:**

- Detectar productos por debajo del mínimo.
- Sugerir reposición (según rotación/ventas recientes si aplica).
- Notificar en panel (y correo si está configurado).

**Colaboradores:**

- Clase Producto (stock actual y mínimos).
  - Clase de Reporte (rotación/ventas, si aplica).
  - Clase BaseDatos (persistencia).
- 

**Tabla 66.** Tarjeta CRC Precios

---

**TARJETA CRC**

---

**Nombre:** Precios

**Responsabilidades:**

- Gestionar el precio por unidad y por fracción.
- Aplicar promociones y vigencias (si existen).
- Mantener historial de cambios.

**Colaboradores:**

- Clase Producto (asociación).
  - Clase Bitácora (registro de cambios).
  - Clase BaseDatos (persistencia).
- 

**Tabla 67.** Tarjeta CRC Cierre Caja

---

**TARJETA CRC**

---

**Nombre:** Cierre Caja

**Responsabilidades:**

- Consolidar las ventas por forma de pago.
- Registrar diferencias y observaciones.
- Generar documento de arqueo.

**Colaboradores:**

- Clase Comprobante (ventas del día).
  - Clase Bitácora (registro).
  - Clase BaseDatos (persistencia).
- 

**Tabla 68.** Tarjeta CRC Movimiento interno

---

**TARJETA CRC**

---

**Nombre:** Movimiento interno

**Responsabilidades:**

- Registrar traslados entre ubicaciones (bodega/góndola).
- Realizar ajustes con motivo y un responsable.
- Generar movimiento en Kardex.

**Colaboradores:**

- Clase Producto (ítems).
  - Clase Kardex (movimiento).
  - Clase Bitácora (registro).
  - Clase BaseDatos (persistencia).
-

**Tabla 69. Tarjeta CRC Reporte**

<b>TARJETA CRC</b>
<b>Nombre:</b> Reporte
<b>Responsabilidades:</b>
<ul style="list-style-type: none"> <li>- Generar reportes (ventas, rotación, valorización, top productos).</li> <li>- Filtrar por fechas, producto y responsable.</li> <li>- Preparar datos para exportación.</li> </ul>
<b>Colaboradores:</b>
<ul style="list-style-type: none"> <li>- Clase BaseDatos (consultas).</li> <li>- Clase de Exportación (formato de archivo).</li> <li>- Clase Usuario/Rol (permisos).</li> </ul>

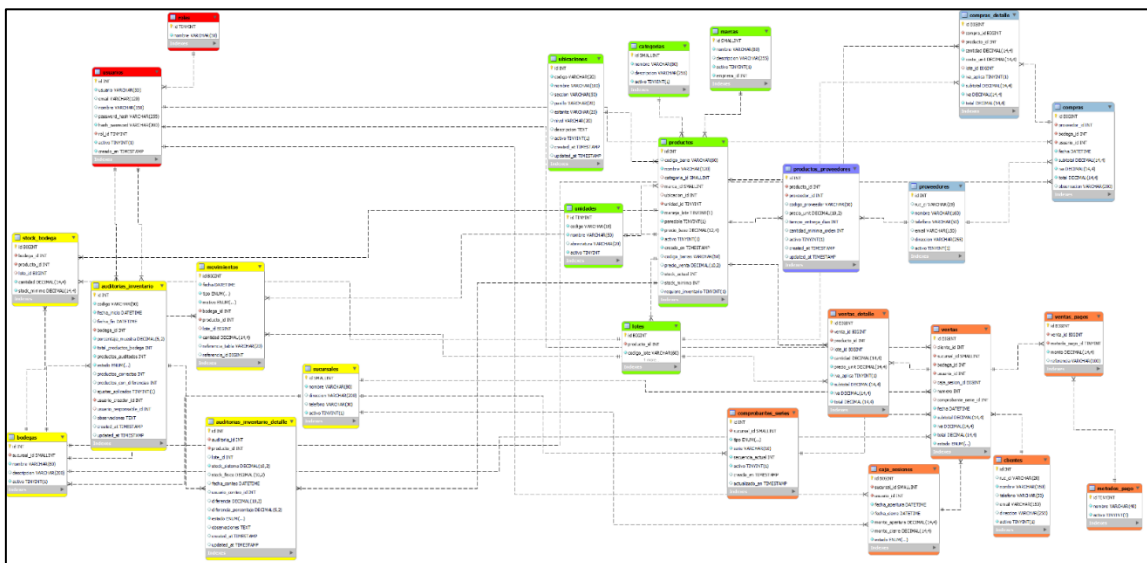
**Tabla 70. Tarjeta CRC Exportación**

<b>TARJETA CRC</b>
<b>Nombre:</b> Exportación
<b>Responsabilidades:</b>
<ul style="list-style-type: none"> <li>- Exportar reportes, Kardex y bitácora a PDF y Excel.</li> <li>- Controlar el nombre del archivo y los metadatos.</li> </ul>
<b>Colaboradores:</b>
<ul style="list-style-type: none"> <li>- Clase Reporte (fuente de datos).</li> <li>- Clases Kardex y Bitácora (cuando se exportan directamente).</li> </ul>

**Tabla 71. Tarjeta CRC Anulación**

<b>TARJETA CRC</b>
<b>Nombre:</b> Anulación
<b>Responsabilidades:</b>
<ul style="list-style-type: none"> <li>- Validar la autorización por rol y motivo.</li> <li>- Revertir los efectos en el Kardex y en la caja (si aplica).</li> <li>- Registrar trazabilidad completa.</li> </ul>
<b>Colaboradores:</b>
<ul style="list-style-type: none"> <li>- Clase Usuario/Rol (autorización).</li> <li>- Clase VentaPOS / Comprobante (objeto a anular).</li> <li>- Clases Kardex y Cierre de Caja (reverso).</li> <li>- Clase Bitácora (registro).</li> </ul>

4.2.9.2 Base de datos (diseño).



**Figura 2. Diseño de la base de datos**

### 4.2.9.3 Prototipo

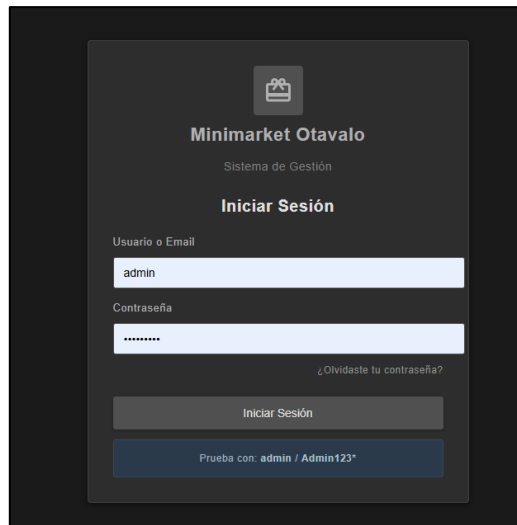


Figura 3. Pantalla inicio de sesión al sistema

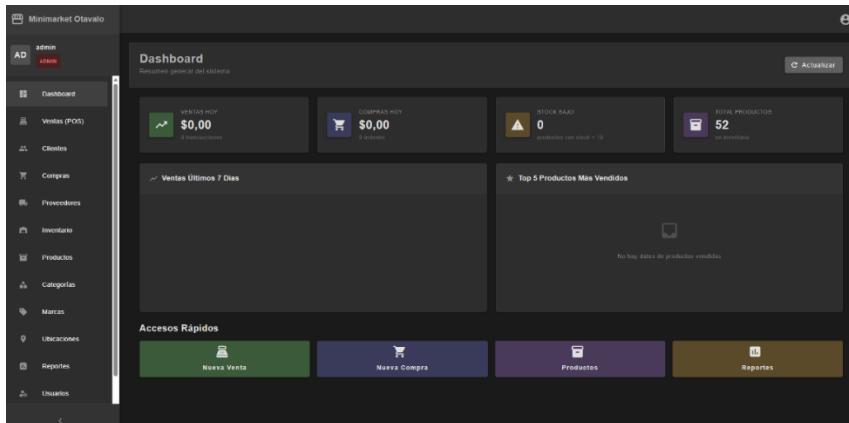


Figura 4. Pantalla principal, inicio, funciones principales y dashboard

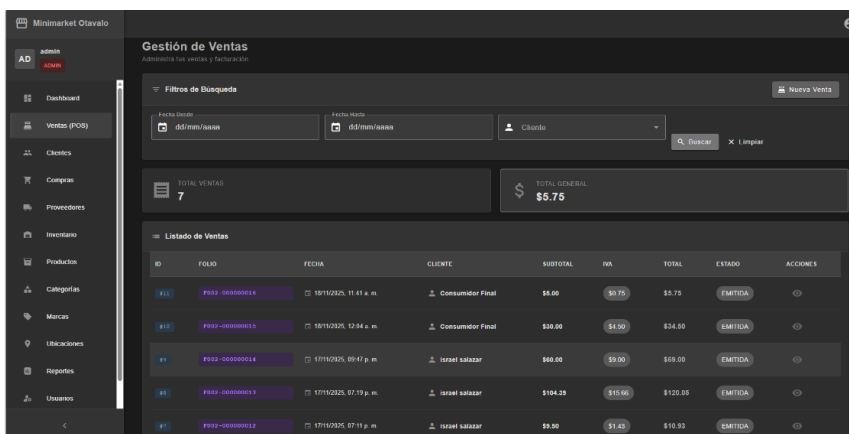


Figura 5. Pantalla del panel de ventas

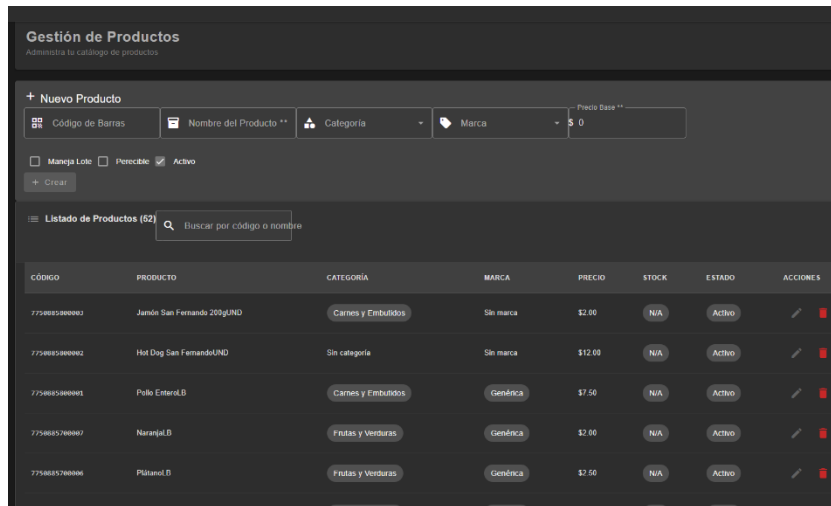


Figura 6. Pantalla del listado de productos

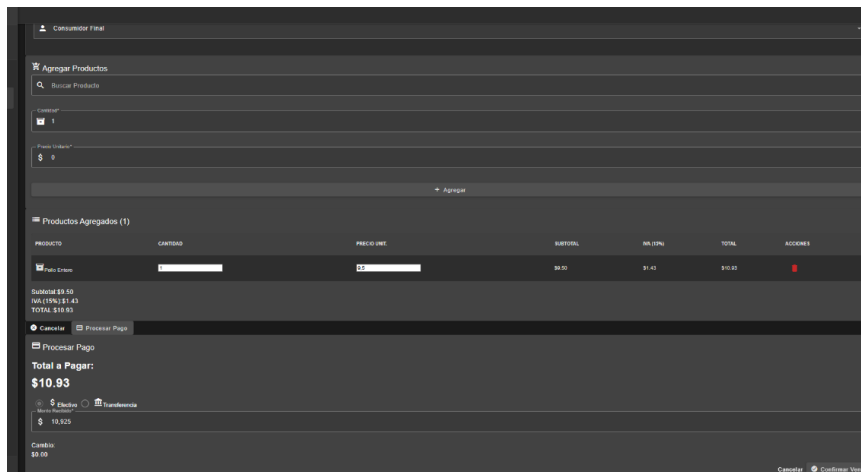


Figura 7. Pantalla creación de venta y procesamiento pagos facturación

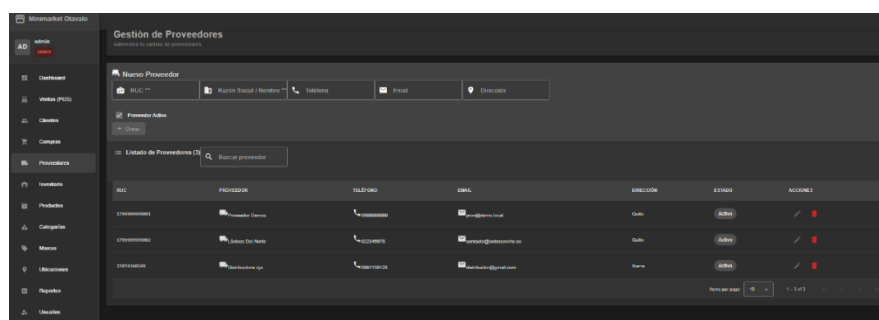


Figura 8. Pantalla registro de proveedores

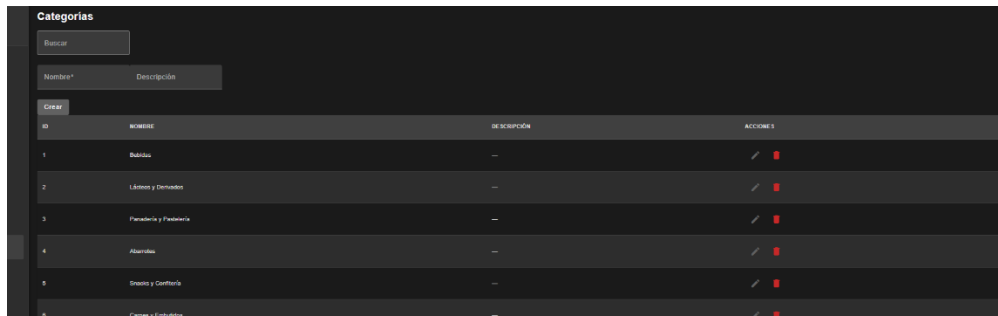


Figura 9. Pantalla de categorías de los productos

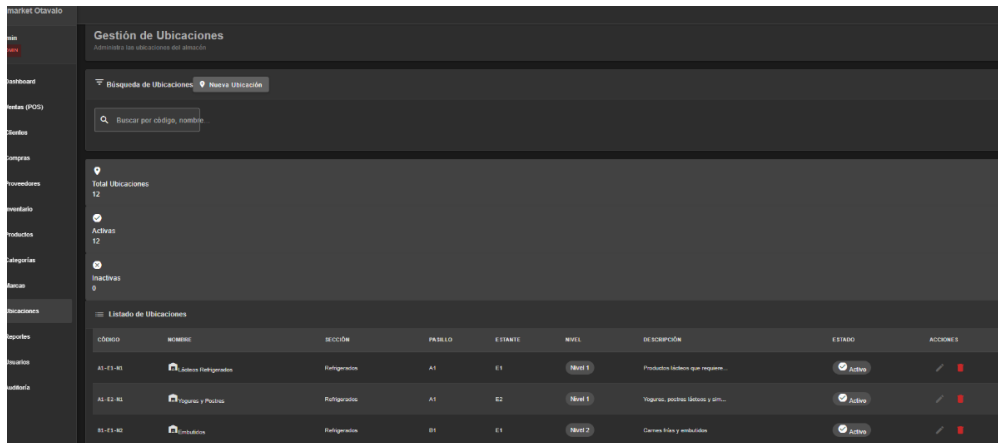


Figura 10. Pantalla de ubicaciones de los productos

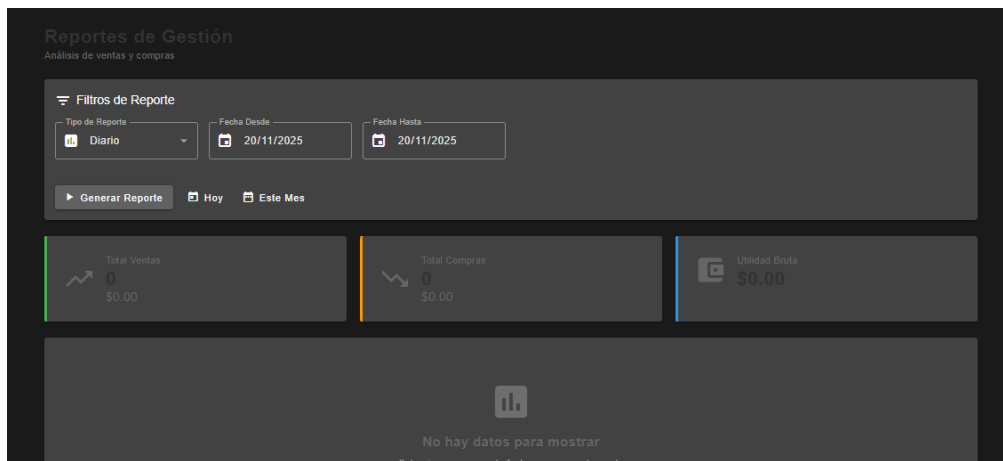


Figura 11. Pantalla de reportes de ventas

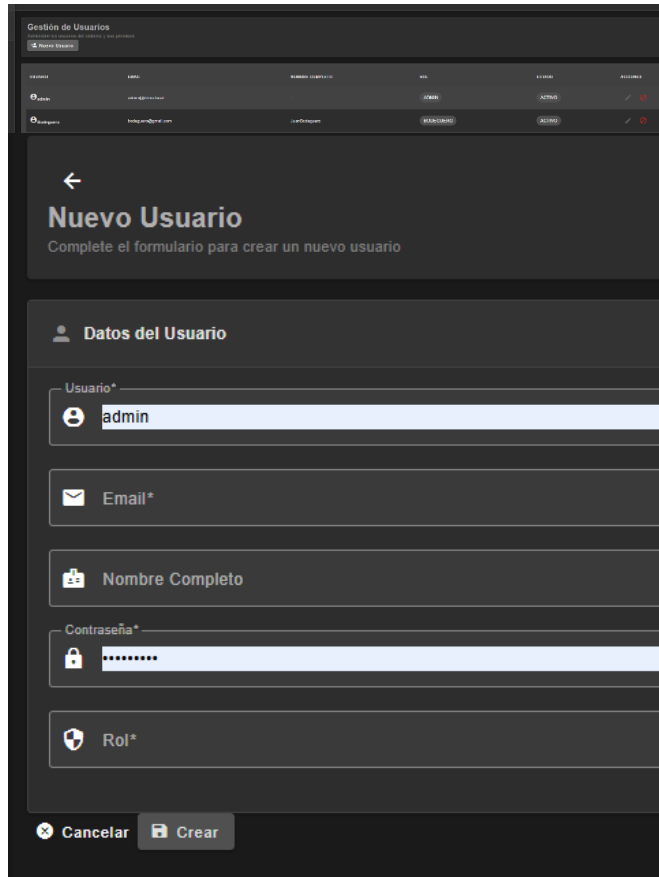


Figura 12. Pantalla de gestión y creación de usuarios

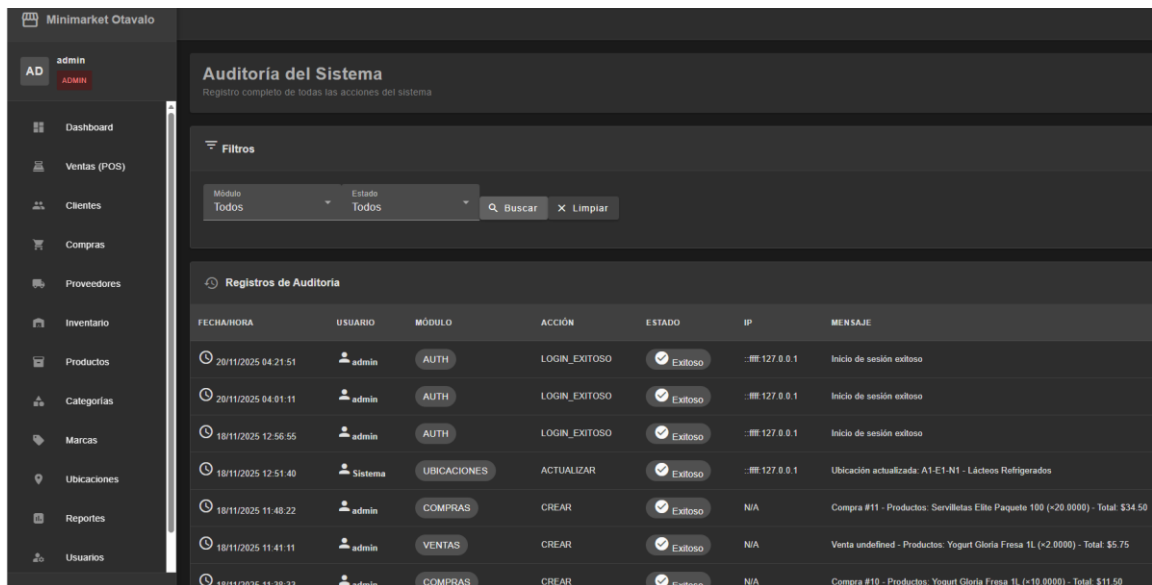


Figura 13. Pantalla de auditoría completa del sistema

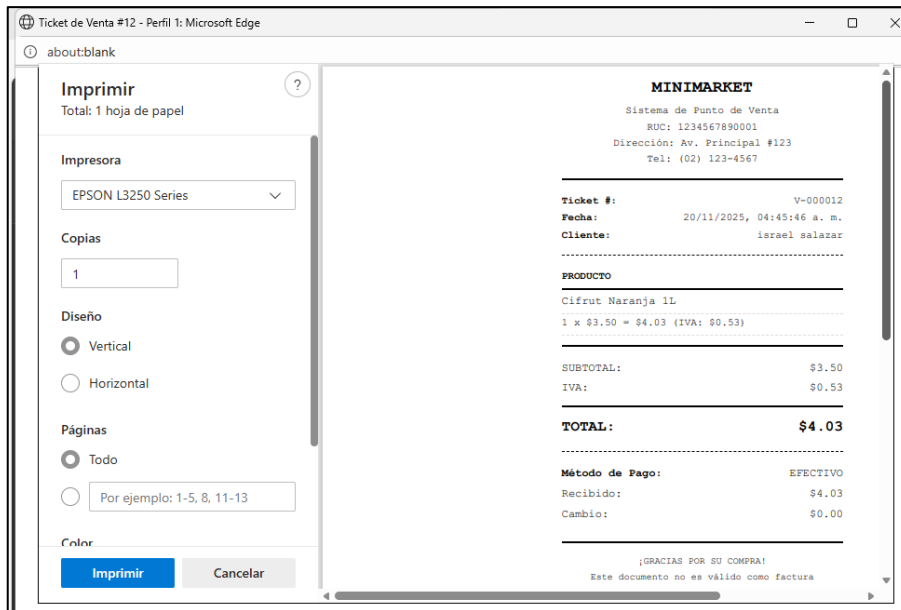


Figura 14. Pantalla de factura del Minimarket

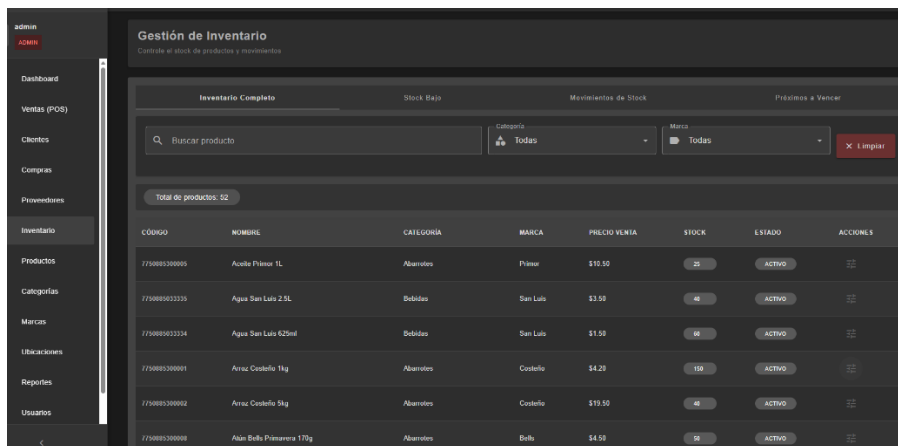
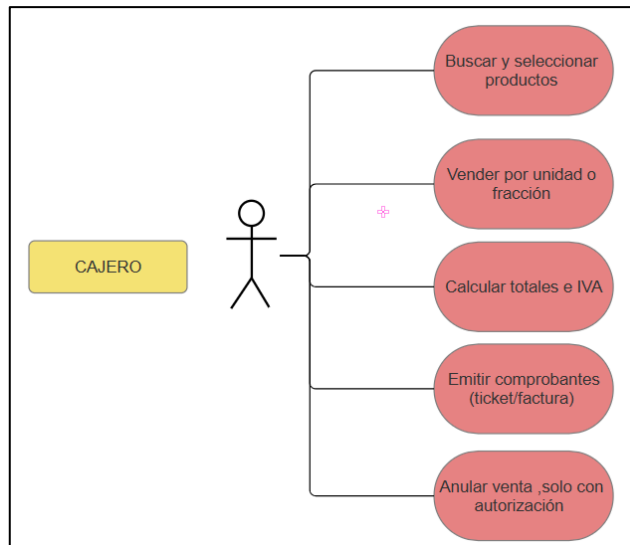


Figura 15. Pantalla inventario, stocks y productos por caducar

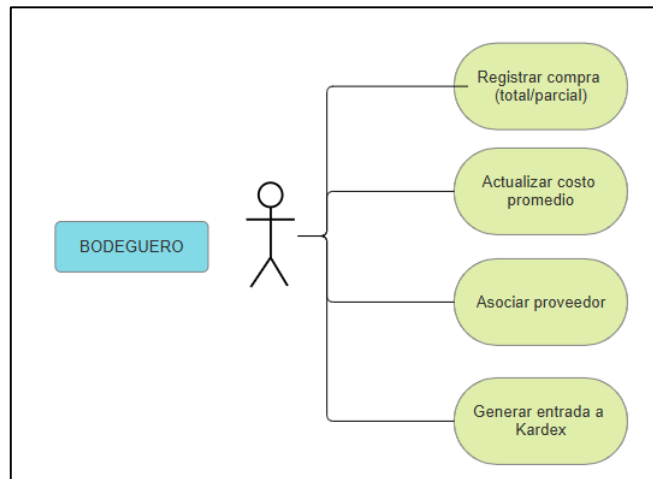
#### 4.2.10 Diagramas de caso de uso



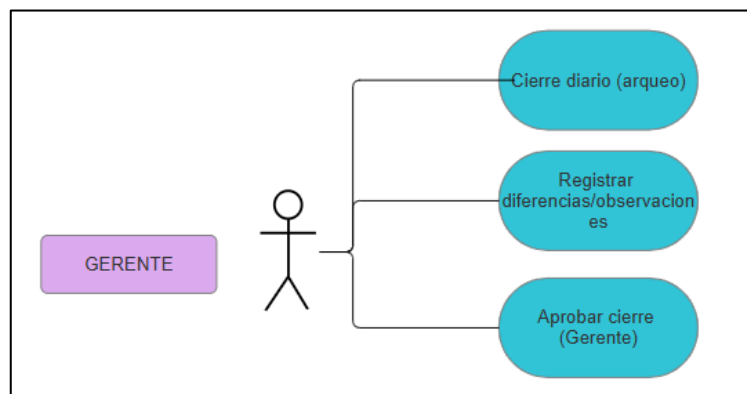
Figura 16. Diagrama de caso de uso siendo administrador



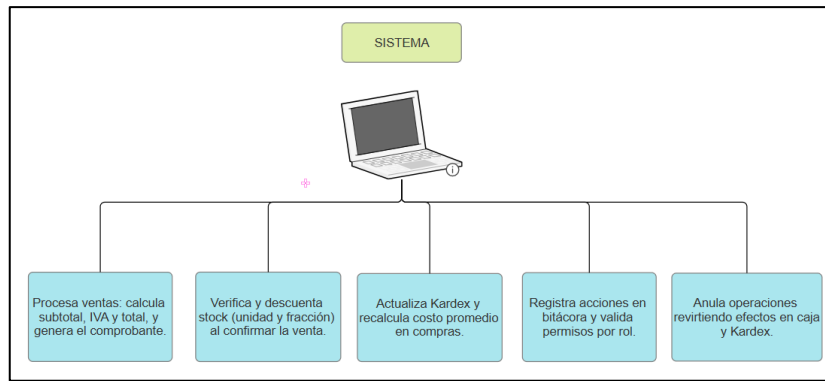
**Figura 17.** Diagrama de caso de uso del Cajero



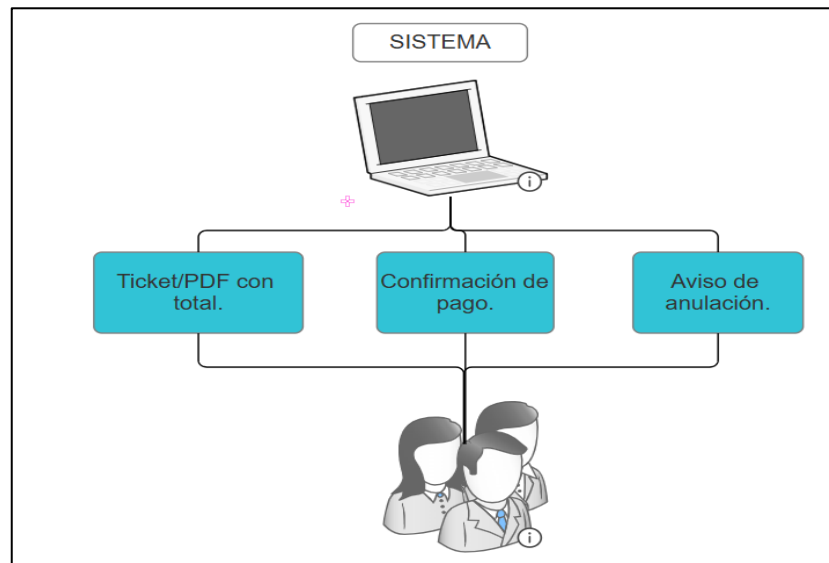
**Figura 18.** Diagrama de caso de uso del Bodeguero



**Figura 19.** Diagrama de caso de uso del Gerente



**Figura 20.** Diagrama de caso de uso de funciones que hace el sistema



**Figura 21.** Diagrama de caso de uso de funciones del sistema hacia el cliente

#### 4.2.11 Fase de Codificación

Durante la etapa de codificación, se utilizó Angular 17 para crear una interfaz moderna y fácil de usar, que incluía componentes, servicios y rutas protegidas para mayor seguridad. En el servidor, se empleó Node.js con Express, conectado a MySQL, y usando JWT para gestionar el acceso de roles como Gerente, Administrador, Cajero y Bodeguero. El Kardex se actualizó automáticamente con las compras y ventas, que también se registraron en la bitácora, asegurando un seguimiento detallado. Además, se implementaron validaciones y una gestión centralizada de errores para mantener la trazabilidad y la coherencia en todo momento. Se dio prioridad a un código fácil de mantener, con una separación clara en capas y pruebas básicas que facilitan la incorporación de nuevas funciones sin afectar las existentes.

## Función de autenticación y seguridad

```
.env JS auth.js X categorias-list.component.html index.html TS api.service.ts dashboard.component.html
minimarket-backend > src > middlewares > JS auth.js > ...
1  const jwt = require('jsonwebtoken');
2
3  exports.auth = (req, res, next) => {
4    const auth = req.headers.authorization || '';
5    if (!auth.startsWith('Bearer ')) {
6      return res.status(401).json({ error: 'Token requerido' });
7    }
8    try {
9      const token = auth.slice(7);
10     const payload = jwt.verify(token, process.env.JWT_SECRET);
11     req.user = payload; // { id, usuario, email, rol }
12     next();
13   } catch (e) {
14     return res.status(401).json({ error: 'Token inválido o vencido' });
15   }
16 };
17
18 exports.authorizeRoles = (...roles) => (req, res, next) => {
19   if (!req.user || !roles.includes(req.user.rol)) {
20     return res.status(403).json({ error: 'No autorizado' });
21   }
22   next();
23 };
```

Figura 22. Código función de autenticación y seguridad

Este middleware garantiza el acceso mediante tokens JWT (JSON Web Tokens). Auth comprueba que cada solicitud contenga un token válido y, en caso afirmativo, obtiene los datos del usuario (nombre, rol e identificación). AuthorizeRoles implementa la regulación de los permisos por rol para cada ruta. Si el token está ausente o no es válido, se niega la solicitud y no se tiene acceso a los recursos.

## Función para proceso de venta y control de stock

```
119 // 1. Obtener stock ANTERIOR (si no está en el Map, leer de BD)
120 let stockAnterior;
121 if (stockPorProducto.has(producto_id)) {
122   // Ya procesamos este producto antes, usar el valor acumulado
123   stockAnterior = stockPorProducto.get(producto_id);
124 } else {
125   // Primera vez que vemos este producto, leer de BD
126   const [[stockAntes]] = await conn.query(
127     'SELECT COALESCE(SUM(cantidad), 0) AS total FROM stock_bodega WHERE producto_id=? AND bodega_id=?',
128     [producto_id, bodega_id]
129   );
130   stockAnterior = Number(stockAntes.total || 0);
131 }
132
133 // 2. Calcular el nuevo stock
134 const stockNuevo = stockAnterior - Number(cantidad);
135
136 // 3. Guardar el nuevo stock en el Map para siguientes iteraciones
137 stockPorProducto.set(producto_id, stockNuevo);
138
139 // 4. Actualizar stock_bodega (restar cantidad) - ASEGURARNOS QUE SE ACTUALICE EN LA BD
140 if (lote_id) {
141   const [result] = await conn.query(
142     'UPDATE stock_bodega SET cantidad = cantidad - ? WHERE producto_id=? AND bodega_id=? AND lote_id=?',
143     [cantidad, producto_id, bodega_id, lote_id]
144   );
145   console.log('UPDATE stock lote: affected ${result.affectedRows} rows');
146   if (result.affectedRows === 0) {
147     throw new Error('No existe registro de stock para producto ${producto_id} en la bodega ${bodega_id} con lote ${lote_id}');
148   }
149 } else {
150   const [[row]] = await conn.query(
151     'SELECT id, cantidad FROM stock_bodega WHERE producto_id=? AND bodega_id=? AND lote_id IS NULL LIMIT 1',
152     [producto_id, bodega_id]
```

Figura 23. Código función ventas y control de stock

Esta sección actualiza el inventario en tiempo real con cada transacción de venta. Saca el stock anterior, halla el nuevo valor restando la cantidad vendida, almacena el resultado en la base de datos y anota un movimiento con el antes y el después, así como la fecha y el usuario. El sistema crea automáticamente un registro de stock con una cantidad negativa para posibilitar la venta y luego realiza el ajuste si no hay un registro previo del producto.

### **Función para generar comprobantes**

```
56 // 2) Serie activa (LOCK)
57 const [series] = await conn.query(
58   `SELECT id, sucursal_id, tipo, serie, secuencia_actual, activo
59   FROM comprobantes_series
60   WHERE sucursal_id=? AND tipo=? AND activo=1
61   ORDER BY id ASC
62   LIMIT 1
63   FOR UPDATE`,
64   [sucursal_id, tipo_comprobante]
65 );
66 if (!series.length) {
67   throw new Error('No hay serie activa configurada para la sucursal/tipo');
68 }
69 const serieRow = series[0];
70 const siguienteNumero = Number(serieRow.secuencia_actual || 0) + 1;
71
72 // 3) Insert cabecera
73 const [venta] = await conn.query(
74   `INSERT INTO ventas
75   (cliente_id, sucursal_id, bodega_id, usuario_id, caja_sesion_id,
76   numero, comprobante_serie_id, subtotal, iva, total, estado)
77   VALUES (?, ?, ?, ?, ?, ?, ?, 0, 0, 0, 'EMITIDA')`,
78   [
79     cliente_id, sucursal_id, bodega_id, usuario_id, caja_sesion_id,
80     siguienteNumero, serieRow.id
81   ]
82 );
83
```

**Figura 24.** Código función generar factura

Este código automatiza la generación de números de comprobante (facturas, notas de crédito) de manera secuencial y segura. Busca la serie activa de comprobantes usando `FOR UPDATE` para bloquear la fila y evitar que dos vendedores obtengan el mismo número si venden simultáneamente. Incrementa el número de secuencia, crea el registro de venta con el nuevo número, y actualiza el contador para la próxima venta.

## Registro de compras

```
15 async function crearCompra(req, res) {
16   const conn = await pool.getConnection();
17   try {
18     const { proveedor_id, bodega_id, usuario_id, observacion = null, items = [] } = req.body;
19
20     if (!proveedor_id || !bodega_id || !usuario_id || !Array.isArray(items) || items.length === 0) {
21       return res.status(400).json({ error: 'Datos incompletos: proveedor_id, bodega_id, usuario_id e items son obligatorios' });
22     }
23
24     await conn.beginTransaction();
25
26     // Cambiar nivel de aislamiento para ver cambios inmediatos
27     await conn.query('SET SESSION TRANSACTION ISOLATION LEVEL READ COMMITTED');
28
29     const [[cfg]] = await conn.query('SELECT iva_porcentaje FROM config WHERE id=1');
30     const ivaPct = Number((cfg && cfg.iva_porcentaje) || 15);
31
32     const [compra] = await conn.query(
33       `INSERT INTO compras (proveedor_id, bodega_id, usuario_id, observacion, subtotal, iva, total)
34        VALUES (?, ?, ?, ?, 0, 0, 0),
35        [proveedor_id, bodega_id, usuario_id, observacion]
36     );
37
38     let subtotal = 0, iva = 0, total = 0;
39
40     // Mapa para rastrear el stock de cada producto en memoria
41     const stockPorProducto = new Map();
42
43     for (const it of items) {
44       const { producto_id, cantidad, costo_unit, iva_aplica = 1, lote_codigo = null, fecha_caducidad = null } = it;
45       if (!producto_id || !cantidad || !costo_unit) {
46         throw new Error('Cada item requiere producto_id, cantidad y costo_unit');
47       }
48
49       let lote_id = null;
```

Figura 25. Código función registrar compras

Esta función documenta las compras y aumenta el inventario; valida el proveedor, la bodega, el usuario y los artículos mediante transacciones para garantizar la integridad. Calcula automáticamente el IVA y los subtotales, actualiza o establece el registro en stock\_bodega sumando las cantidades y marca la transacción como ENTRADA en el historial. Asimismo, permite almacenar lotes y fechas de vencimiento de productos que pueden descomponerse.

## Listado de productos con consulta

```
3 exports.listarProductos = async (req, res) => {
4   try {
5     const pagina = parseInt(req.query.pagina || '1', 10);
6     const tamaño = parseInt(req.query.tamaño || '10', 10);
7     const buscar = (req.query.buscar || '').trim();
8
9     const offset = (pagina - 1) * tamaño;
10
11     let where = '';
12     let params = [];
13     if (buscar) {
14       where = 'WHERE p.nombre LIKE ? OR p.codigo_barra LIKE ?';
15       params.push(`%${buscar}%`, `%${buscar}%`);
16     }
17
18     const [datos] = await pool.query(
19       `
20       SELECT p.id, p.codigo_barra AS codigo, p.codigo_barra, p.nombre, p.precio_base, p.precio_venta, p.activo,
21         c.nombre AS categoria, m.nombre AS marca, u.codigo AS unidad,
22         ub.codigo AS ubicacion_codigo, ub.nombre AS ubicacion_nombre,
23         ub.seccion AS ubicacion_seccion, ub.pasillo AS ubicacion_pasillo,
24         ub.estante AS ubicacion_estante, ub.nivel AS ubicacion_nivel,
25         COALESCE(SUM(sb.cantidad), 0) AS stock_total
26       FROM productos p
27       LEFT JOIN categorias c ON c.id = p.categoria_id
28       LEFT JOIN marcas m ON m.id = p.marca_id
29       INNER JOIN unidades u ON u.id = p.unidad_id
30       LEFT JOIN ubicaciones ub ON ub.id = p.ubicacion_id
31       LEFT JOIN stock_bodega sb ON sb.producto_id = p.id
32       ${where}
33       GROUP BY p.id, p.codigo_barra, p.nombre, p.precio_base, p.precio_venta, p.activo, c.nombre, m.nombre, u.codigo, ub.codigo, ub.nombre, ub.sec
34       ORDER BY p.id DESC
35       LIMIT ? OFFSET ?
36     `
37     [...params, tamaño, offset]
```

Figura 26. Código función listar productos y consultar

La función lista productos con paginación (10-20 por página), permite la búsqueda por nombre o código de barras y fusiona información sobre marcas, categorías, unidades, productos, ubicaciones y existencias en el almacén. Realiza una consulta optimizada que organiza y agrupa para mostrar primero los más recientes.

## Dashboard y estadísticas en tiempo real

```

1  GET /api/dashboard/stats 42
2  * Retorna estadísticas generales del sistema 43
3  // 44
4  export.getStats = async (req, res) => { 45
5  // 46
6  try { 47
7  const connection = await pool.getConnection(); 48
8  // 49
9  try { 50
10 // 3. Total de ventas del día 51
11 let ventasDay = { total_ventas: 0, monto_total: 0 }; 52
12 try { 53
13 const [rows] = await connection.query(` 54
14 SELECT 55
15 COBIT(*) as total_ventas, 56
16 FROM FUNCION(total), 0) as monto_total 57
17 FROM ventas 58
18 WHERE DATE(fecha) = CURDATE() 59
19 `); 60
20 ventasDay = rows[0] || ventasDay; 61
21 } catch (e) { 62
22 console.log('Falta ventas no disponible', e.message); 63
23 } 64
24 // 3. Total de compras del día 65
25 let comprasDay = { total_compras: 0, monto_total: 0 }; 66
26 try { 67
27 const [rows] = await connection.query(` 68
28 SELECT 69
29 COBIT(*) as total_compras, 70
30 FROM FUNCION(total), 0) as monto_total 71
31 FROM compras 72
32 WHERE DATE(fecha) = CURDATE() 73
33 `); 74
34 comprasDay = rows[0] || comprasDay; 75
35
36 // 3. Productos con stock bajo (menos de 10 unidades) 76
37 let stockBajo = { total_producto_bajo_stock: 0 }; 77
38 try { 78
39 const [rows] = await connection.query(` 79
40 SELECT COBIT(DISTINCT p.id) as total_productos_bajo_stock 80
41 FROM productos p 81
42 LEFT JOIN stock_bodega sb ON sb.producto_id = p.id 82
43 GROUP BY p.id 83
44 HAVING COBIT(SUM(sb.cantidad), 0) < 10 84
45 `); 85
46 stockBajo = { total_producto_bajo_stock: rows.length }; 86
47 } catch (e) { 87
48 console.log('Stock bajo no disponible', e.message); 88
49 } 89
50 // 4. Total de productos 90
51 let totalProductos = { total: 0 }; 91
52 try { 92
53 const [rows] = await connection.query(`SELECT COBIT(*) as total FROM productos`); 93
54 totalProductos = rows[0] || totalProductos; 94
55 } catch (e) {} 95
56 // 5. Total de clientes 96
57 let totalClientes = { total: 0 }; 97
58 try { 98
59 const [rows] = await connection.query(`SELECT COBIT(*) as total FROM clientes`); 99
60 totalClientes = rows[0] || totalClientes; 100
61 } catch (e) {} 101
62 // 6. Ventas de los últimos 7 días 102
63 let ventasUltimos7Dias = []; 103
64 try { 104
65 const [rows] = await connection.query(` 105
66 SELECT 106
67 `); 107
68 } catch (e) {} 108
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200

```

Figura 27. Código función dashboard de movimientos

El panel de control te ofrece una visión clara y actualizada de tus métricas más importantes del negocio en tiempo real. Aquí se puede ver fácilmente las ventas diarias, tanto en dinero como en número de transacciones, así como las semanales, para identificar tendencias. También destacan los cinco productos más vendidos, lo que ayuda a tomar decisiones de compra. Además, alerta sobre artículos con poco inventario y productos próximos a caducar, lo que facilita que promociones antes de la fecha límite.

## Interfaz de usuario (Angular 18 + Material Design)

### Componentes del Login

```

15 export class LoginComponent {
16 // Password recovery
17 showRecovery = false;
18 loadingRecovery = false;
19 recoveryError = '';
20 recoverySuccess = false;
21
22 form = this.fb.group({
23   user: ['', [Validators.required]],
24   password: ['', [Validators.required]],
25 });
26
27 recoveryForm = this.fb.group({
28   email: ['', [
29     Validators.required,
30     Validators.email,
31     Validators.pattern(/^[a-zA-Z0-9_]+@[a-zA-Z0-9-]+\.[a-zA-Z]{2,}$/)
32 ]],
33 });
34
35 constructor(private fb: FormBuilder, private auth: AuthService, private router: Router) {}
36
37 submit() {
38   this.errorMsg = '';
39   if (this.form.invalid) return;
40
41   this.loading = true;
42
43   const user = (this.form.get('user')?.value ?? '').toString().trim();
44   const password = (this.form.get('password')?.value ?? '').toString();
45
46   this.auth.login({ user, password }).subscribe({
47     next: () => {
48       this.loading = false;
49       this.router.navigateByHref('/dashboard');
50     }
51   });
52 }
53 }
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200

```

Figura 28. Código función login

El componente de inicio de sesión verifica que el usuario y la contraseña estén completos y correctos antes de habilitar el envío. Envía la información al servidor. Si las credenciales son válidas, recibe un token, lo almacena y luego accede al panel de control. Si existen errores, muestra un mensaje claro y un indicador de carga mientras se revisa con Angular Material.

## Procesamiento de Venta

```
51     MatDialogModule,
52     MatRadioModule,
53   ],
54   templateUrl: './ventas-form.component.html',
55   styleUrls: ['./ventas-form.component.scss']
56 })
57 export class VentasFormComponent implements OnInit {
58   ventaForm: FormGroup;
59   productoForm: FormGroup;
60
61   clientes = signal<Cliente[]>([]);
62   productos = signal<Producto[]>([]);
63   productosFiltrados = signal<Producto[]>([]);
64   items = signal<VentaItem[]>([]);
65   loading = signal(false);
66
67   displayedColumns = ['producto', 'cantidad', 'precio_unit', 'subtotal', 'iva', 'total', 'acciones'];
68
69   // IVA por defecto
70   IVA_PORCENTAJE = 15;
71
72   // Métodos de pago
73   metodosPago = signal<{id: number, nombre: string}[]>([
74     { id: 1, nombre: 'EFECTIVO' },
75     { id: 2, nombre: 'TRANSFERENCIA' }
76   ]);
77
78   pagoForm: FormGroup;
79   mostrarDialogoPago = signal(false);
80   montoPagado = signal(0);
81   cambio = signal(0);
82
83   // Computed totals
84   subtotalGeneral = computed(() =>
85     this.items().reduce((sum, item) => sum + item.subtotal, 0)
```

Figura 29. Código función ventas

Comprueba que haya al menos un producto, que, si corresponde, el cliente esté seleccionado y que el pago cubra el total. Después, compone el contenido de la venta utilizando información del cliente, de la bodega, del local, del usuario, de la lista de ítems y de los pagos; luego, lo envía al servidor. Al obtener la confirmación, se puede imprimir el comprobante, limpiar el formulario y actualizar las estadísticas para la próxima venta.

## Inventarios y Ajuste de Stock

```

98 }
99 export class InventarioComponent implements OnInit {
100 // Signals
101 inventarioCompleto = signal<ProductoInventario[]>({});
102 stockBajo = signal<ProductoInventario[]>({});
103 movimientos = signal<MovimientoStock[]>({});
104 productosProximosVencer = signal<ProductoProximoVencer[]>({});
105 loading = signal(false);
106 tabSeleccionada = signal(0);
107
108 // Filtros
109 busqueda = signal('');
110 categoriaSeleccionada = signal<string>('');
111 marcaSeleccionada = signal<string>('');
112 tipoMovimiento = signal<string>('');
113 filtroVencimiento = signal<string>(''); // 'TODOS', 'VENCIDOS', 'VENICE_HOY', 'VENICE_7_DIAS', 'VENICE_30_DIAS'
114
115 // Listas para filtros
116 categorias = computed(() => {
117   const cats = this.inventarioCompleto()
118     .map(p => p.categoria)
119     .filter((c): c is string => !!c);
120   return [...new Set(cats)].sort();
121 });
122
123 marcas = computed(() => {
124   const marcas = this.inventarioCompleto()
125     .map(p => p.marca)
126     .filter((m): m is string => !!m);
127   return [...new Set(marcas)].sort();
128 });
129
130 // Computed: Inventario filtrado
131 inventarioFiltrado = computed(() => {
132   let productos = this.inventarioCompleto();
133   // ...
134 });

```

Figura 30. Código función para ajustar el stock y ver inventario actualizado

Esta función permite corregir discrepancias entre el stock físico y el del sistema, debidas a pérdidas, errores, robos o daños. Cuando se selecciona un producto, se abre una ventana para revisar el stock actual, ingresar el stock correcto y agregar un comentario obligatorio. El sistema, tras calcular el diferencial, transmite la información al servidor con el propósito de actualizar el stock, generar un movimiento de ajuste y que quede la auditoría correcta.

### 4.2.12 Templates

#### Pantalla Login

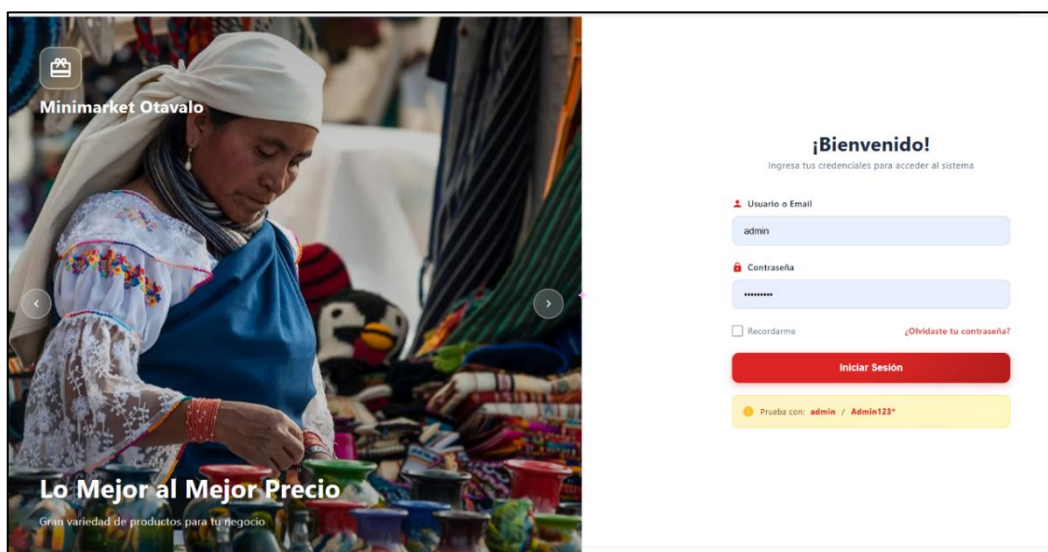


Figura 31. Template login

## Pantalla inicio/Dashboard

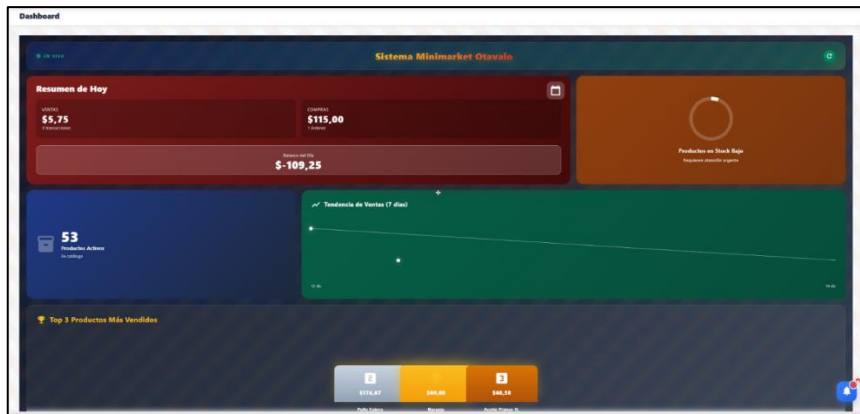


Figura 32. Template inicio del sistema y dashboard

## Pantalla Ventas

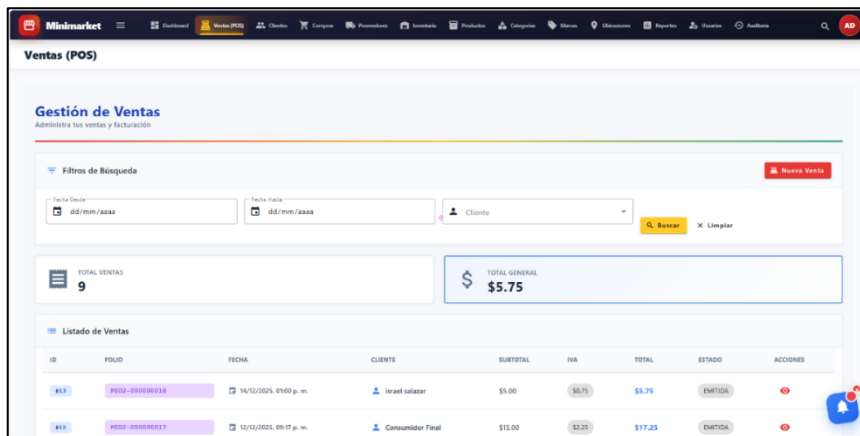


Figura 33. Template realizar ventas y gestión

## Pantalla clientes

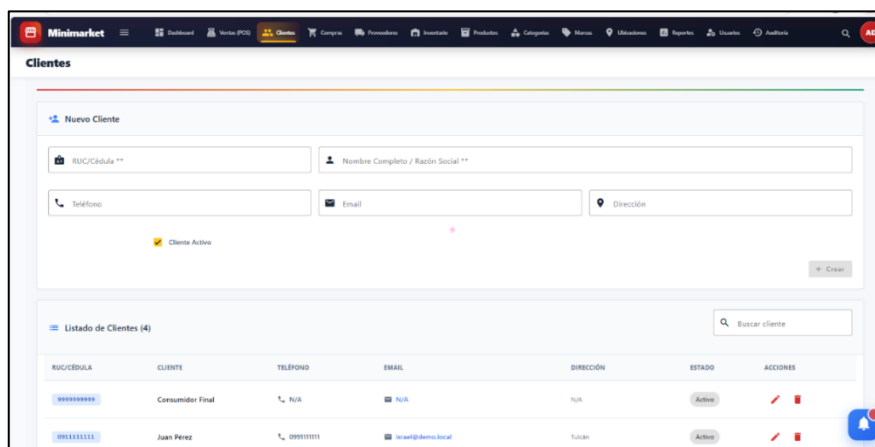


Figura 34. Template crear clientes y gestión

## Pantalla Compra de productos

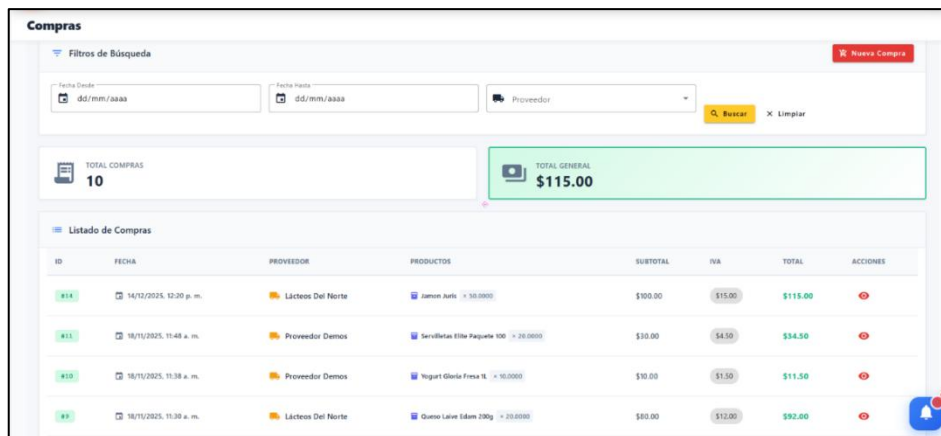


Figura 35. Template gestionar y crear nuevas compras

## Pantalla Proveedores

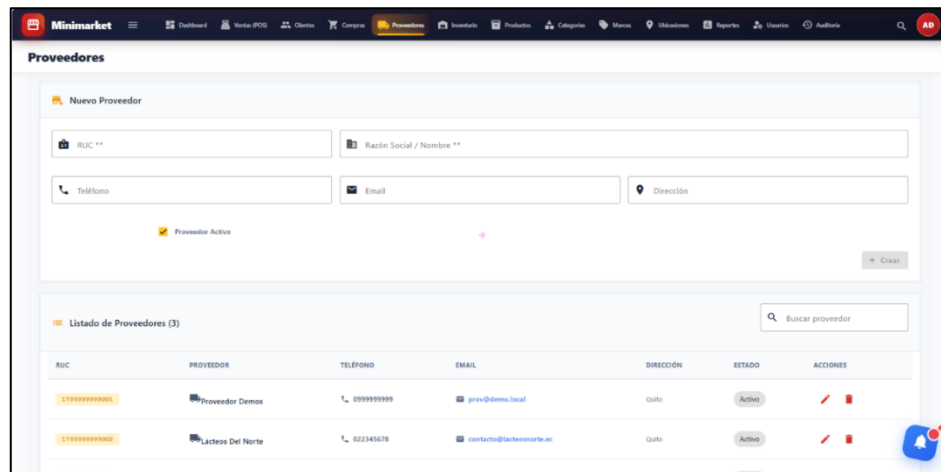


Figura 36. Template ingresar nuevos proveedores

## Pantalla Inventario completo

Niveles mínimos de stock, movimientos, productos por vencer.

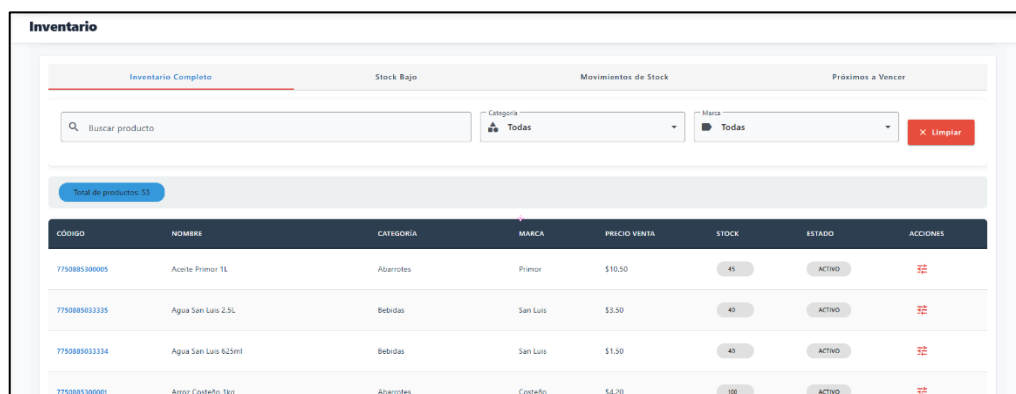


Figura 37. Inventario completo con múltiples funciones

## Pantalla Gestión de productos

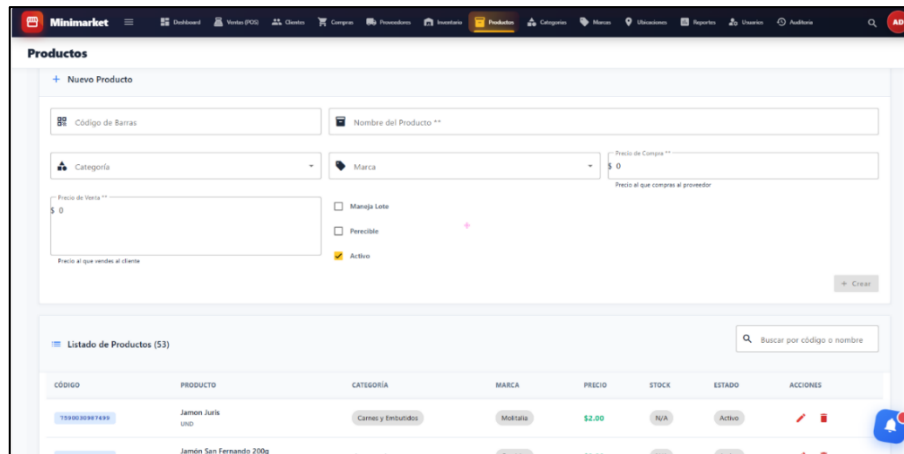


Figura 38. Template gestión de productos

## Pantalla Ubicaciones de productos

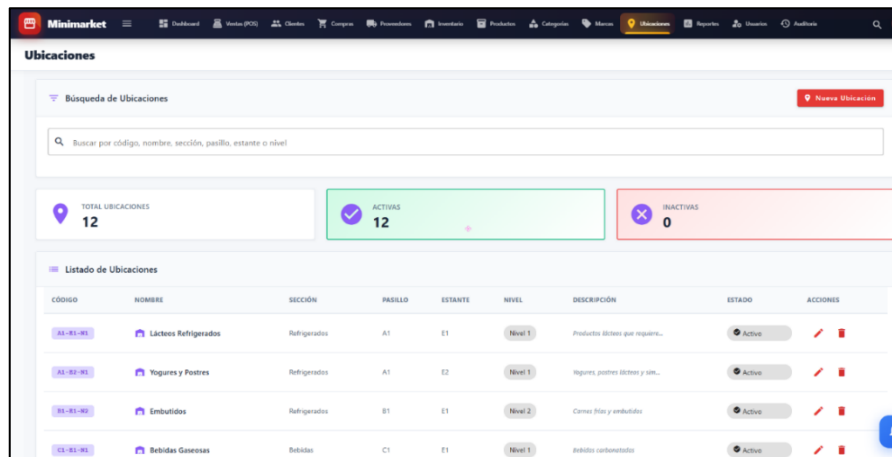


Figura 39. Template gestionar ubicaciones

## Pantalla Reportes y exportación

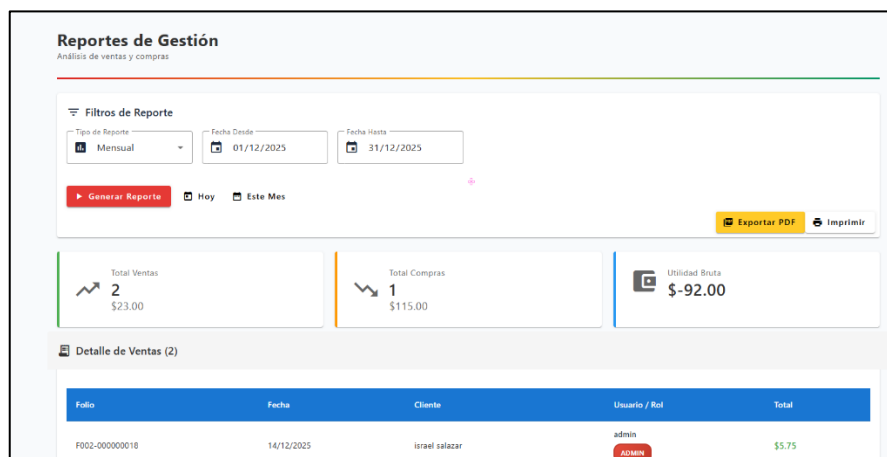


Figura 40. Template generación de reportes diarios y mensuales

## Pantalla Gestionar usuarios y roles

Usuario	Email	Nombre Completo	Rol	Estado	Acciones
admin	admin@demo.local	-	ADMIN	ACTIVO	
Bodeguero	bodeguero@gmail.com	JuanBodeguero	BODEGUERO	ACTIVO	
Cajero	cajero@gmail.com	JuanCajero	CAJERO	ACTIVO	
Gerente	gerente@gmail.com	PedroGerente	GERENTE	ACTIVO	
Proba	israelsalazar164@gmail.com	Israel Salazar	CAJERO	ACTIVO	

Figura 41. Template gestión de usuarios y sus roles

## Pantalla Auditoria

CODIGO	BODEGA	FECHA INICIO	PRODUCTOS	ESTADO	PRECISION	ACCIONES
AUD-20251214-787	Bodega Central	14/12/2025 17:47	7 / 58 (12.06%)	Completada	85.71%	
AUD-20251214-909	Bodega Central	14/12/2025 13:02	6 / 58 (10.34%)	Completada	50.00%	
AUD-20251214-371	Bodega Central	14/12/2025 00:29	6 / 57 (10.53%)	Completada	91.31%	
AUD-20251213-497	Bodega Central	13/12/2025 19:23	1 / 57 (1.75%)	Completada	0.00%	
AUD-20251213-563	Bodega Central	13/12/2025 19:15	1 / 57 (1.75%)	Completada	0.00%	
AUD-20251213-035	Bodega Central	13/12/2025 19:06	3 / 57 (5.26%)	Completada	0.00%	
AUD-20251213-076	Bodega Central	13/12/2025 19:02	6 / 57 (10.53%)	Cancelada	-	
AUD-20251213-534	Bodega Central	13/12/2025 18:55	6 / 57 (10.53%)	Completada	66.67%	

PRODUCTO	LOTE	STOCK SISTEMA	STOCK FISICO	DIFERENCIA	ESTADO	ACCIONES
Acaite Primor 1L 775040300005	-	45	48	0	CONTADO	
Fideos Niccolini Tornillo 500g 7750565300004	-	60	60	0	CONTADO	
Leche Gloria Entera 1L 7720385300001	-	50	50	0	CONTADO	
Papas Lays Clásicas 180g 770050500001	0003	40	40	0	CONTADO	
Papas Lays Limón 180g 7700505400001	-	40	40	0	CONTADO	

Figura 42. Template auditoría del sistema

## Pantalla Recuperar contraseña de acceso con correo electrónico

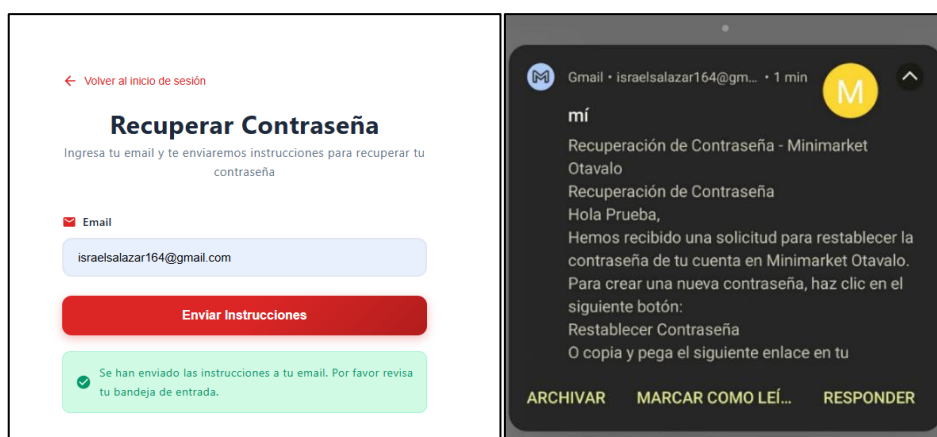


Figura 43. Template recuperar contraseña de acceso

### 4.3. DISCUSIÓN

Los resultados obtenidos en esta investigación permiten establecer comparaciones con estudios previos que abordaron problemáticas similares en la gestión de inventarios de pequeños comercios.

En cuanto al análisis de los procesos actuales del Minimarket Otavalo, se identificaron problemas como errores de transcripción en registros manuales, falta de actualización del stock en tiempo real, ausencia de alertas por stock mínimo y dificultades para generar reportes confiables. Estos hallazgos coinciden con lo reportado por Pérez y Ramírez (2020), quienes encontraron que el 68% de las PYMEs que utilizan hojas de cálculo presentan al menos cinco errores de transcripción mensuales, y que el 73% identificó la falta de automatización como un problema crítico. De manera similar, García y Martínez (2020) señalaron que hasta el 60% de los errores en inventarios físicos se deben a discrepancias en el registro de datos y retrasos en la actualización de información. La situación del Minimarket Otavalo refleja estas mismas condiciones, lo que confirma que los problemas detectados no son casos aislados, sino una realidad común en los pequeños comercios minoristas de la región.

Respecto a la definición de requerimientos funcionales, el sistema diseñado incluye módulos de control de lotes con fechas de caducidad y alertas automáticas de stock mínimo. Esta decisión de diseño se fundamenta en los hallazgos de Jiménez (2021), quien demostró que los sistemas automatizados de inventario permiten un mejor seguimiento de productos perecederos, logrando reducir las pérdidas por vencimiento en un 42% y las de stock en un 35%. Si bien estos beneficios cuantitativos

solo podrían verificarse mediante la implementación del sistema, el diseño propuesto incorpora las funcionalidades necesarias para que, en una fase posterior, el Minimarket Otavalo pueda obtener resultados similares.

En relación con el diseño de la base de datos, se desarrolló un modelo relacional compuesto por 26 tablas normalizadas hasta la tercera forma normal, que incluye un módulo de Kardex con registro de movimientos, usuario responsable, fecha y motivo. Este enfoque es consistente con lo propuesto por García y Martínez (2020), quienes destacaron que una adecuada estructuración del Kardex digital puede reducir errores hasta en un 40% y agilizar las tareas administrativas en un 50%.

Asimismo, Rodríguez (2020) demostró que el uso de bases de datos relacionales incrementa la accesibilidad a los datos, permite consultas segmentadas y facilita la elaboración de reportes detallados. El modelo diseñado para el Minimarket Otavalo sigue estos principios, garantizando integridad en los datos y eliminando redundancia de estos.

En cuanto a la selección de tecnologías, la evaluación técnica determinó que la combinación de Angular para el frontend, Node.js con Express para el backend y MySQL como base de datos constituye la arquitectura más adecuada para el sistema. Esta decisión difiere parcialmente de lo reportado por Rodríguez (2020), quien utilizó PHP con MySQL en su implementación. Sin embargo, la presente investigación optó por tecnologías más actuales que ofrecen mayor escalabilidad, tipado estricto mediante TypeScript y la posibilidad de utilizar un único lenguaje de programación tanto en el cliente como en el servidor. Esta diferencia responde a la evolución tecnológica y a las necesidades específicas identificadas durante el análisis de requerimientos.

Con respecto a la metodología de desarrollo, se seleccionó Extreme Programming (XP) para la documentación de historias de usuario, tarjetas CRC y el diseño iterativo del sistema. Esta elección se respalda en los resultados de Sihombing (2024), quien reportó que la aplicación de XP permitió reducir en un 47% los errores detectados en pruebas de integración y alcanzar un incremento del 42% en la satisfacción de usuarios finales. Aunque la presente investigación se limita a la fase de diseño, la documentación generada mediante 15 historias de usuario, 20 tareas de ingeniería y los diagramas de casos de uso sigue los principios de XP, estableciendo las bases para una futura implementación que pueda replicar estos resultados positivos.

Es importante señalar una diferencia fundamental entre los antecedentes consultados y el presente estudio: las investigaciones de Pérez y Ramírez (2020), Jiménez (2021), García y Martínez (2020), Rodríguez (2020) y Sihombing (2024) llegaron hasta la fase de implementación, lo que les permitió medir resultados cuantitativos como porcentajes de reducción de errores, mejoras en eficiencia y niveles de satisfacción. En contraste, esta investigación se delimita al diseño del sistema computacional, por lo que no es posible realizar comparaciones directas en términos de indicadores de desempeño. No obstante, los artefactos generados, que incluyen el modelo de base de datos normalizado, los requerimientos funcionales y no funcionales validados, las historias de usuario documentadas, los prototipos de interfaces y la arquitectura definida, constituyen una base técnica sólida que puede ser retomada en investigaciones posteriores orientadas a la implementación y evaluación del sistema propuesto.

Finalmente, los resultados de esta investigación confirman lo señalado por García y Martínez (2020) respecto a la importancia de analizar los procesos actuales de gestión antes de proponer soluciones tecnológicas. El levantamiento detallado de los procesos del Minimarket Otavalo, la identificación de problemas específicos y la definición de requerimientos basados en necesidades reales permitieron diseñar un sistema adaptado al contexto del negocio, evitando soluciones genéricas que podrían no ajustarse a su operación diaria.

## V. CONCLUSIONES Y RECOMENDACIONES

### 5.1. CONCLUSIONES

- La revisión bibliográfica permitió comprender los conceptos fundamentales de la gestión de inventarios y evaluar las herramientas de desarrollo disponibles. Se determinó que la combinación de Angular, Node.js con Express y MySQL constituye la arquitectura más adecuada para el sistema, con base en criterios de rendimiento, compatibilidad y escalabilidad.
- El diagnóstico de los procesos actuales del Minimarket Otavalo identificó problemas críticos como errores de transcripción, ausencia de actualización del stock en tiempo real, carencia de alertas por stock mínimo y pérdidas por productos vencidos. Estos hallazgos fundamentaron la definición de los requerimientos funcionales y no funcionales del sistema.
- Se diseñó un sistema computacional compuesto por módulos integrados que cubren las necesidades identificadas: catálogo de productos, compras y proveedores, ventas con facturación, Kardex con trazabilidad, alertas automáticas, control de caja y gestión de usuarios con roles diferenciados. El desarrollo aplicó la metodología XP mediante historias de usuario, tareas de ingeniería y diagramas de casos de uso.
- La investigación demuestra que el análisis previo de los procesos operativos y la evaluación técnica de herramientas son pasos fundamentales para diseñar sistemas que respondan a las necesidades reales de los pequeños comercios, evitando soluciones genéricas que no se adaptan al contexto específico del negocio.

## 5.2. RECOMENDACIONES

- Se recomienda que investigaciones similares en el ámbito de diseño de sistemas computacionales para pequeños comercios apliquen el proceso de levantamiento de procesos mediante fichas técnicas o de observación directa antes de definir los requerimientos funcionales, ya que este enfoque permitió identificar con precisión las necesidades reales del Minimarket Otavalo y evitar el diseño de funcionalidades innecesarias o alejadas del contexto operativo del negocio.
- Es recomendable el uso de matrices de evaluación técnica con criterios adecuados para la elección de tecnologías de desarrollo, como se empleó en este estudio. Esto es porque dicha herramienta posibilita justificar las decisiones tecnológicas de manera objetiva y simplifica la comparación entre las opciones existentes en el mercado, evitando que se seleccionen basándose solo en inclinaciones o gustos individuales.
- Se recomienda que el diseño de bases de datos para sistemas de gestión de inventarios aplique la normalización hasta la tercera forma normal, ya que este proceso garantiza la integridad de los datos, elimina la redundancia de información y facilita el mantenimiento del modelo a largo plazo, aspectos fundamentales para la confiabilidad de cualquier sistema de control de existencias.

## VI. REFERENCIAS BIBLIOGRÁFICAS

- Andrade Clavijo, B. P., y Guerrero Cepeda, M. J. (2023). Evaluación de la gestión de inventarios y facturación en el almacén Peregrine Falcón. *Ciencia Latina Revista Científica Multidisciplinar*, 7(2), 3328-3348. [https://doi.org/10.37811/cl\\_rcm.v7i2.5576](https://doi.org/10.37811/cl_rcm.v7i2.5576)
- Arens, A. A., Elder, R. J., y Beasley, M. S. (2017). *Auditoría: Un enfoque integral* (15.ª ed.). Pearson Educación. <https://www.pearsonenespanol.com/mexico/auditoria-un-enfoque-integral>
- Avenía, C. (2017). *Fundamentos de seguridad informática*. Ecoe Ediciones.
- Ballou, R. H. (2004). *Logística: Administración de la cadena de suministro* (5.ª ed.). Pearson Educación. <https://www.pearson.com/es/universitario/logistica>
- Banco Interamericano de Desarrollo. (2025). *El poder de los datos: Impulsando la transformación digital de las pymes de América Latina*. BID. <https://blogs.iadb.org/innovacion/es/el-poder-de-los-datos-impulsando-la-transformacion-digital-de-las-pymes-de-america-latina/>
- Bowersox, D. J., Closs, D. J., y Cooper, M. B. (2013). *Administración y logística en la cadena de suministros* (13.ª ed.). McGraw-Hill Education.
- Causado Rodríguez, E. (2015). Modelo de inventarios para control económico de pedidos en empresa comercializadora de alimentos. *Revista Ingenierías Universidad de Medellín*, 14(27), 163-177. <https://www.redalyc.org/articulo.oa?id=75045730012>
- Chase, R. B., Jacobs, F. R., y Aquilano, N. J. (2018). *Administración de operaciones: Producción y cadena de suministros* (15.ª ed.). McGraw-Hill Education. <https://www.mheducation.com/highered/product/operations-management>
- Chiquito Tigua, G. P., y Cedeño Muñoz, G. L. (2025). Gestión de inventarios y la eficiencia operativa de Lubrirepuestos Navarrete. *Ciencia y Desarrollo*, 28(1), 448-461. <https://dialnet.unirioja.es/descarga/articulo/10091190.pdf>

Chopra, S., y Meindl, P. (2016). *Administración de la cadena de suministro: Estrategia, planeación y operación* (6.ª ed.). Pearson Educación. <https://www.pearson.com/supply-chain-management>

Comisión Económica para América Latina y el Caribe. (2021). *Transformación digital de las mipymes: elementos para el diseño de políticas*. CEPAL. <https://www.cepal.org/es/publicaciones/47183-transformacion-digital-mipymes-elementos-diseno-politicas>

Comisión Económica para América Latina y el Caribe, y Banco de Desarrollo de América Latina. (2020). *Las oportunidades de la digitalización en América Latina frente al COVID-19*. CEPAL-CAF. <https://www.cepal.org/es/publicaciones/45360-oportunidades-la-digitalizacion-america-latina-frente-al-covid-19>

Durán, Y. (2012). Administración del inventario: elemento clave para la optimización de las utilidades en las empresas. *Visión Gerencial*, (1), 55-78. <https://www.redalyc.org/pdf/4655/465545892008.pdf>

Express.js. (2024). *Express: Fast, unopinionated, minimalist web framework for Node.js*. OpenJS Foundation. <https://expressjs.com/>

Flores-Tapia, C., y Flores-Ceballos, K. (2023). Optimización de inventarios aplicando Investigación de Operaciones. *RECAI Revista de Estudios en Contaduría, Administración e Informática*, 12(34), 1-15. <https://www.redalyc.org/journal/6379/637974848003/html/>

Gómez, R. (2022). *Manual de gestión de almacenes e inventarios*. Ediciones de la U.  
Google Developers. (2024). *Angular: The modern web developer's platform*. <https://angular.dev/>

Guerrero Salas, H. (2022). *Inventarios: manejo y control* (2.ª ed.). Ecoe Ediciones. <https://www.ecoediciones.mx/wp-content/uploads/2017/08/Inventarios.-Manejo-y-control-2da-Edici%C3%B3n.pdf>

Hansen, D., y Mowen, M. (2018). *Administración de costos: Contabilidad y control* (8.ª ed.). Cengage Learning.



Heizer, J., y Render, B. (2014). *Principios de administración de operaciones* (9.ª ed.). Pearson Educación. <https://www.pearson.com/us/higher-education/program/Heizer-Principles-of-Operations-Management>

- Hernández-Sampieri, R., y Mendoza, C. P. (2018). *Metodología de la investigación: Las rutas cuantitativa, cualitativa y mixta*. McGraw-Hill Education. <http://www.biblioteca.cij.gob.mx/Archivos/Materiales de consulta/Drogas de Abuso/Articulos/SampieriLasRutas.pdf>
- IBM Cloud Education. (2021). *What is Three-Tier Architecture?* IBM. <https://www.ibm.com/think/topics/three-tier-architecture>
- Instituto Nacional de Estadística y Censos. (2023). *Registro Estadístico de Empresas 2023 - Boletín Técnico*. INEC. <https://www.ecuadorencifras.gob.ec/documentos/web-inec/Estadisticas Economicas/Registro Empresas Establecimientos/2023/Semestre I/Boletin REEM 2023.pdf>
- Krajewski, L. J., Ritzman, L. P., y Malhotra, M. K. (2013). *Administración de operaciones: Procesos y cadena de suministro* (10.ª ed.). Pearson Educación. <https://www.pearson.com/operations-management>
- La Febre, M., y Rivera, C. (2023). Análisis del control interno de inventarios y su impacto en los productos terminados. *Revista de Ciencias Empresariales*, 8(2), 45-60.
- Laudon, K. C., y Laudon, J. P. (2016). *Sistemas de información gerencial* (14.ª ed.). Pearson Educación. <https://www.pearson.com/management-information-systems>
- Letelier, P., y Penadés, M. C. (2006). Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). *Técnica Administrativa*, 5(26). <https://www.cyta.com.ar/ta0502/v5n2a1.htm>
- McCombes, S. (2019). *Descriptive Research: Definition, Types, Methods & Examples*. Scribbr. <https://www.scribbr.com/methodology/descriptive-research/>
- Meta. (2024). *React: A JavaScript library for building user interfaces*. <https://react.dev/>
- Mora García, L. A. (2016). *Gestión logística integral: Las mejores prácticas en la cadena de abastecimiento* (2.ª ed.). Ecoe Ediciones.
- MySQL. (2024). *MySQL 8.0 Reference Manual*. Oracle Corporation. <https://dev.mysql.com/doc/refman/8.0/en/>
- OpenJS Foundation. (2024). *Node.js Documentation*. <https://nodejs.org/docs/latest/api/>

- Oracle NetSuite. (2024). *Types of Inventory Management Systems*. Oracle. <https://www.netsuite.com/portal/resource/articles/inventory-management/types-inventory-management-systems.shtml>
- Pressman, R. S. (2010). *Ingeniería del software: Un enfoque práctico* (7.ª ed.). McGraw-Hill. <https://profesorezequielruizgarcia.wordpress.com/wp-content/uploads/2015/01/ingenieria-del-software-un-enfoque-practico-roger-s-pressman.pdf>
- Pressman, R. S., y Maxim, B. R. (2020). *Software Engineering: A Practitioner's Approach* (9th ed.). McGraw-Hill Education. <https://www.mheducation.com/highered/product/software-engineering-practitioner-s-approach-pressman-maxim/M9781260548006.html>
- Pressman, R. S., y Maxim, B. R. (2021). *Ingeniería de software: Un enfoque práctico* (9.ª ed.). McGraw-Hill Education.
- Sommerville, I. (2011). *Ingeniería de software* (9.ª ed.). Pearson Educación. [https://gc.scalahed.com/recursos/files/r161r/w25469w/ingdelsoftwarelibro9\\_compressed.pdf](https://gc.scalahed.com/recursos/files/r161r/w25469w/ingdelsoftwarelibro9_compressed.pdf)
- Sommerville, I. (2021). *Software Engineering* (10th ed.). Pearson Education.
- UNIB.E. (2022). *Guía metodológica: Cómo investigar en la Universidad Iberoamericana del Ecuador*. Universidad Iberoamericana del Ecuador. [https://unibe.edu.ec/wp-content/uploads/2022/08/RF\\_LIBRO-Como-Investigar-en-la-UNIB.E-version-19-10-2021.pdf](https://unibe.edu.ec/wp-content/uploads/2022/08/RF_LIBRO-Como-Investigar-en-la-UNIB.E-version-19-10-2021.pdf)
- Universidad Tecnológica de Jalisco. (2017). *Gestión logística: Manual del alumno - Unidad 3*. <https://pabellon.tecnm.mx/CENTRODEINFORMACION/app/files/101050051.pdf>
- Visual Studio Code. (2024). *Documentation for Visual Studio Code*. Microsoft. <https://code.visualstudio.com/docs>
- Vue.js Team. (2024). *Vue.js: The Progressive JavaScript Framework*. <https://vuejs.org/>
- Whittington, O. R., y Pany, K. (2016). *Principios de auditoría* (14.ª ed.). McGraw-Hill Education.

## VII. ANEXOS

### Anexo 1. Acta de la sustentación de Predefensa del TIC

**UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI**

FACULTAD DE INDUSTRIAS AGROPECUARIAS Y CIENCIAS AMBIENTALES

CARRERA DE COMPUTACIÓN

**ACTA**

DE LA INSTANTACIÓN ORAL DE LA PREDEFENSA DEL TRABAJO DE INTEGRACIÓN CURRICULAR CON ENFOQUE EN INVESTIGACIÓN


ESTUDIANTE:	SALAZAR BURBANO ISRAEL SALAZAR	CÉDULA DE IDENTIDAD:	0401952445
PERIODO ACADÉMICO:	2026A	DOCENTE TUTOR:	MSC. JEFFERY NARANJO CEDERO
PRESIDENTE TRIBUNAL:	MSC. FERNÁNDEZ FERNÁNDEZ YASMANY	DOCENTE:	MSC. GEORGINA ARCOS PONCE
TEMA DEL TIC:	"SISTEMA COMPUTACIONAL DE GESTIÓN DE INVENTARIOS PARA MINIMARKET"		

No.	CATEGORÍA	Evaluación cuantitativa	OBSERVACIONES Y RECOMENDACIONES
1	PROBLEMA - OBJETIVOS	9.00	
2	FUNDAMENTACIÓN TEÓRICA	7.47	
3	METODOLOGÍA	9.00	
4	RESULTADOS	9.00	
5	DISCUSIÓN	4.00	
6	CONCLUSIONES Y RECOMENDACIONES	9.00	
7	DEFENSA, ARGUMENTACIÓN Y VOCABULARIO PROFESIONAL	9.00	
8	FORMATO, ORGANIZACIÓN Y CALIDAD DE LA INFORMACIÓN	9.00	

Obteniendo una nota de: **8.57** Por lo tanto, **APRUEBA** : debiendo el o los investigadores acotar el siguiente artículo:

Art. 66.- De la aprobación de la pre defensa del informe final de TIC.- El estudiante deberá obtener una nota mínima de 7/10, al finalizar el finalizar el pre-defensa se procederá a levantar el acta correspondiente. En el caso de aprobar con observaciones el estudiante deberá adjuntar el informe final de cumplimiento de observaciones y recomendaciones emitido por el Tribunal previo a la defensa final en un término máximo de 10 días.

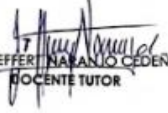
Para constancia del presente, firman en la ciudad de Tulcán el **Miércoles, 6 de mayo de 2026**



MSC. FERNANDO FERNANDEZ YASMANY  
PRESIDENTE TRIBUNAL



MSC. GEORGINA ARCOS PONCE  
DOCENTE



MSC. JEFFERY NARANJO CEDERO  
DOCENTE TUTOR

Escaneado con CamScanner



UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI  
FACULTAD DE INDUSTRIAS AGROPECUARIAS Y CIENCIAS AMBIENTALES



CARRERA DE COMPUTACIÓN  
RÚBRICA DE EVALUACIÓN DE LA SUSTENTACIÓN ORAL DE LA PREDEFENSA DEL  
TRABAJO DE INTEGRACIÓN CURRICULAR CON ENFOQUE EN INVESTIGACIÓN

ESTUDIANTE:	SALAZAR BURBANO ISRAEL SALAZAR	CÉDULA DE IDENTIDAD:	0401952445
PERIODO ACADÉMICO:	2026A	FECHA:	6 de mayo de 2026
PRESIDENTE TRIBUNAL:	MSC. FERNÁNDEZ FERNÁNDEZ YASMANY	DOCENTE TUTOR:	MSC. JEFFERY HARAJJO CEDEÑO
DOCENTE:	MSC. GEORGINA ARCOS PONCE	AULA:	112
TEMA DEL TIC:	"SISTEMA COMPUTACIONAL DE GESTIÓN DE INVENTARIOS PARA MINIMARKET"		

No.	CATEGORÍA	CRITERIO ÓPTIMO DE EVALUACIÓN	PRESIDENTE TRIBUNAL	DOCENTE TUTOR	DOCENTE	
SUSTENTACIÓN ORAL DEFENSA	1	PROBLEMA - OBJETIVOS	Se expone el planteamiento, formulación y justificación, los objetivos son expuestos como científicos para alcanzar el objetivo general, las preguntas de investigación aportan a entender lo que se quiere investigar y son coherentes con los objetivos.	9	9	9
	2	FUNDAMENTACIÓN TEÓRICA	Es un marco de referencia para el desarrollo e interpretación de los resultados de la investigación. Los antecedentes investigativos incluidos tienen relación con el tema planteado. Revisar Metodología de Inventarios	7	7	9
	3	METODOLOGÍA	El estudiante explicó el enfoque de la investigación de manera lógica al análisis estadístico, la población, muestra, técnicas e instrumentos presentados, permitiendo entender que el informe es consistente en resultados y discusión.	9	9	9
	4	RESULTADOS	Se analizó la relación entre las variables de manera cualitativa, cuantitativa y fueron representativas a la profesión. Espasa gráficas, figuras, tablas de frecuencia y contingencia coherentes y de acuerdo a la metodología de investigación. Los datos fueron presentados de forma clara y efectiva a lo observado y no exigen interpretaciones.	9	9	9
	5	DISCUSIÓN	Revisar la aplicación de manera exhaustiva para cada uno de los procesos listados; el proceso de configuración de alertas no queda claro en la plataforma del administrador; cerrar los movimientos. (Actualizar el proceso de inventarios)	6	6	6
	6	CONCLUSIONES Y RECOMENDACIONES	Las conclusiones y recomendaciones expuestas, son claras, concisas y acordes a los objetivos y resultados de la investigación.	9	9	9
	7	DEFENSA, ARGUMENTACIÓN Y VOCABULARIO PROFESIONAL	El estudiante demostró conocimiento y seguridad del objeto de estudio. Relacionó conceptos y teorías. El vocabulario utilizado fue acorde a la terminología de la profesión con un volumen de voz adecuado. Hizo un uso correcto del tiempo. Utilizó recursos didácticos apropiados.	9	9	9
<b>PROMEDIO SOBRE SIETE</b>				5,87		
DOCUMENTO ESCRITO	8	FORMATO, ORGANIZACIÓN Y CALIDAD DE LA INFORMACIÓN	El formato, la organización de contenidos, redacción, uso de gramática y ortografía, aplicación de normas de citas y referencias cumplen con el formato de la UPEC.	9	9	9
	<b>PROMEDIO SOBRE TRES</b>				2,70	
				<b>8,57</b>		

Art. 44.- De la aprobación de la pre defensa del Informe final de TIC.- El estudiante deberá obtener una nota mínima de 7/10; al finalizar el proceso de pre-defensa se procederá a levantar el acta correspondiente. En el caso de aprobar con observaciones el estudiante deberá adjuntar el Informe final de cumplimiento de observaciones y recomendaciones emitido por el Tribunal previo a la defensa final en un término máximo de 10 días.  
Art. 47.- De la re aprobación de la pre defensa del TIC.- Si el estudiante no aprueba la pre defensa tendrá un término de 30 días para realizar los cambios y presentarse por una sola ocasión o una segunda pre defensa, para ello entregará la solicitud dirigida a la Dirección de Carrera. Si el estudiante en la pre defensa adicional no alcanzara la nota establecida para su aprobación, deberá solicitar por una sola ocasión el cambio de opción de titulación.

MSC. JEFFERY HARAJJO CEDEÑO  
DOCENTE TUTOR

MSC. FERNÁNDEZ FERNÁNDEZ YASMANY  
PRESIDENTE TRIBUNAL

MSC. GEORGINA ARCOS PONCE  
DOCENTE

**Anexo 2.** Certificado del abstract por parte de idiomas



UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI FOREIGN AND  
NATIVE LANGUAGES CENTER

ABSTRACT- EVALUATION SHEET				
<b>NAME:</b> Salazar Burbano Israel Sebastián				
<b>DATE:</b> Jueves, 28 de mayo de 2026				
<b>Topic:</b> "Computerized inventory management system for mini markets"				
<b>MARKS AWARDED</b>		<b>QUANTITATIVE AND QUALITATIVE</b>		
<b>VOCABULARY AND WORD USE</b>	Use new learnt vocabulary and precise words related to the topic	Use a little new vocabulary and some appropriate words related to the topic	Use basic vocabulary and simplistic words related to the topic	Limited vocabulary and inadequate words related to the topic
	EXCELLENT: 2 <input type="checkbox"/>	GOOD: 1,5 <input checked="" type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
<b>WRITING COHESION</b>	Clear and logical progression of ideas and supporting paragraphs.	Adequate progression of ideas and supporting paragraphs.	Some progression of ideas and supporting paragraphs.	Inadequate ideas and supporting paragraphs.
<b>De</b>	EXCELLENT: 2 <input checked="" type="checkbox"/>	GOOD: 1,5 <input type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
<b>ARGUMENT</b>	The message has been communicated very well and identify the type of text	The message has been communicated appropriately and identify the type of text	Some of the message has been communicated and the type of text is little confusing	The message hasn't been communicated and the type of text is inadequate
	EXCELLENT: 2 <input checked="" type="checkbox"/>	GOOD: 1,5 <input type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
<b>CREATIVITY</b>	Outstanding flow of ideas and events	Good flow of ideas and events	Average flow of ideas and events	Poor flow of ideas and events
	EXCELLENT: 2 <input type="checkbox"/>	GOOD: 1,5 <input checked="" type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
<b>SCIENTIFIC SUSTAINABILITY</b>	Reasonable, specific and supportable opinion or thesis statement	Minor errors when supporting the thesis statement	Some errors when supporting the thesis statement	Lots of errors when supporting the thesis statement
	EXCELLENT: 2 <input checked="" type="checkbox"/>	GOOD: 1,5 <input type="checkbox"/>	AVERAGE: 1 <input type="checkbox"/>	LIMITED: 0,5 <input type="checkbox"/>
<b>TOTAL/AVERAGE</b>	9 - 10: EXCELLENT 7 - 8,9: GOOD 5 - 6,9: AVERAGE 0 - 4,9: LIMITED	<b>TOTAL 9</b>		



**UNIVERSIDAD POLITÉCNICA ESTATAL DEL  
CARCHI- FOREIGN AND NATIVE LANGUAGES  
CENTER**

**Informe sobre el Abstract de Artículo Científico  
o Investigación.**

**Autor:** Salazar Burbano Israel Sebastián

**Fecha de recepción del abstract:** Miércoles, 20 de mayo de 2026

**Fecha de entrega del informe:** Miércoles, 28 de mayo de 2026

El presente informe validará la traducción del idioma español al inglés si alcanza un porcentaje de: 9 – 10 Excelente.

Si la traducción no está dentro de los parámetros de 9 – 10, el autor deberá realizar las observaciones presentadas en el ABSTRACT, para su posterior presentación y aprobación.

**Observaciones:**

Después de realizar la revisión del presente abstract, éste presenta una apropiada traducción sobre el tema planteado en el idioma inglés. Según la rúbrica de evaluación de la traducción en inglés, ésta alcanza un valor de 9; por lo cual se valida dicho trabajo.

Atentamente



MA. Martha Viveros  
RESPONSABLE CIDEN

### Anexo 3. Informe de Turnitin

**Israel Salazar**  
 tesis final  
 Quick Submit  
 Universidad Politécnica Estatal de Carchi

**Detalles del documento**

Identificador de la entrega trnoid:13549040494	120 páginas
Fecha de entrega 24 abr 2026, 9:11 a.m. GMT-5	25.995 palabras
Fecha de descarga 24 abr 2026, 9:31 a.m. GMT-5	154.376 caracteres
Nombre del archivo Tesis_completa_Israel_Salazar_FINAL_v2.docx	
Tamaño del archivo 5.3 MB	

**9% Similitud general**  
 El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para ca...

**Filtrado desde el Informe**

- Bibliografía
- Coincidencias menores (menos de 8 palabras)

**Exclusiones**

- N° de fuentes excluidas

**Fuentes principales**

- 7% Fuentes de Internet
- 2% Publicaciones
- 3% Trabajos entregados (trabajos del estudiante)

**Marcas de Integridad**  
 N° de alertas de integridad para revisión  
 No se han detectado manipulaciones de texto sospechosas.

Una marca de alerta es la manera en que un indicador de problemas. Sin embargo, recomendamos que preste atención y lo revise.

### Anexo 4. Ficha de observación para levantamiento de procesos actuales

**Instrumento:** Ficha de observación

**Lugar:** Minimarket Otavalo - Ciudad de Mira, Provincia del Carchi

**Fecha de aplicación:** 12/09/2025 **Observador:** Israel Salazar B.

Proceso Observado	Descripción	Herramienta Actual	Problemas Detectados
Control de productos	Registro manual de productos en hojas de cálculo	Excel / Registros manuales	Propenso a errores, sin actualización en tiempo real
Actualización de inventarios	Se actualiza semanalmente o cuando hay tiempo disponible	Excel / Libros físicos	Actualización tardía, datos imprecisos
Entrada de mercancía	Registro de compras a proveedores	Excel	Ingreso tardío, sin trazabilidad
Salida de mercancía	Registro de ventas y retiros	Hojas físicas / Excel	Desorganización, falta de integración

Gestión de proveedores	de Información de proveedores dispersa	de Excel / Papeles	Dificultad para consultar datos
------------------------	--	--------------------	---------------------------------

### Anexo 5. Ficha de identificación de problemas y necesidades

<b>Problema Identificado</b>	<b>Descripción del Impacto</b>	<b>Necesidad Asociada</b>
Falta de actualización en tiempo real	El stock registrado no coincide con el físico, generando ventas de productos agotados	Sistema con actualización automática
Errores de transcripción	Datos incorrectos por digitación manual que afectan inventarios y reportes	Validación automática de datos
Falta de trazabilidad	Imposibilidad de rastrear origen de discrepancias en el inventario	Registro de movimientos con usuario y fecha
Ausencia de alertas	No hay notificación de stock bajo, causando quiebres de inventario	Alertas automáticas por stock mínimo
Dificultad para generar reportes	Toma de decisiones basada en datos incompletos o desactualizados	Generación automática de reportes

## Anexo 6. Checklist de requerimientos funcionales y no funcionales

### Requerimientos Funcionales

ID	Nombre	Descripción	Prioridad
RF-01	Gestión de productos	CRUD de productos con código, nombre, categoría, precio y stock	Alta
RF-02	Registro de compras	Registro de compras vinculadas a proveedores con actualización de stock	Alta
RF-03	Registro de ventas	POS con selección de productos, cálculo de totales y comprobante	Alta
RF-04	Control Kardex	Registro de movimientos con tipo, cantidad, usuario, fecha y motivo	Alta
RF-05	Alertas de stock	Notificación automática cuando el stock alcanza el mínimo definido	Media
RF-06	Gestión de usuarios	Administración de usuarios con roles: Admin, Gerente, Cajero, Bodeguero	Alta
RF-07	Control de caja	Apertura, cierre y arqueo de sesiones de caja	Alta
RF-08	Generación de reportes	Reportes de ventas, inventario, movimientos y productos más vendidos	Media

**Nota:** Para mayor detalle en los requerimientos funcionales ver las tablas de requerimientos en la propuesta ya que aquí están descritos de manera más general.

### REQUERIMIENTOS NO FUNCIONALES

ID	Requerimiento	Descripción
RNF-01	Accesibilidad web	El sistema debe ser accesible mediante navegador web sin instalación adicional
RNF-02	Tiempo de respuesta	Las operaciones deben ejecutarse en menos de 3 segundos
RNF-03	Seguridad	Autenticación mediante JWT y cifrado de contraseñas con bcrypt
RNF-04	Concurrencia	Soporte para hasta 5 usuarios simultáneos sin degradación

**Nota:** Para mayor detalle en los requerimientos no funcionales, ver las tablas de requerimientos en la propuesta, ya que aquí están descritos de manera más general.

## Anexo 7. Comparativa de lenguajes de programación (matriz de evaluación).

**Fuentes:** Documentación oficial de cada tecnología (nodejs.org, php.net, python.org) los enlaces se encuentran en las referencias bibliográficas

Criterio	Node.js/JavaScript	PHP	Python	Java
Curva de aprendizaje	Media - Sintaxis flexible	Baja - Fácil para principiantes	Baja - Sintaxis clara	Alta - Verboso
Rendimiento web	Alto - Asíncrono, event-driven	Medio - Síncrono tradicional	Medio - GIL limita concurrencia	Alto - JVM optimizada
Ecosistema/Paquetes	NPM: +2M paquetes	Composer: +350K paquetes	PyPI: +400K paquetes	Maven: +500K artefactos
Full-stack capability	Sí - Frontend y Backend	No - Solo Backend	No - Solo Backend	No - Solo Backend
Tipado estático	Sí (TypeScript)	Parcial (PHP 8+)	Parcial (Type hints)	Sí - Nativo
PUNTUACIÓN (1-5)	4.5	3.5	3.5	4.0

Node.js/JavaScript con TypeScript obtiene la mayor puntuación (4.5/5) por su capacidad full-stack, ecosistema amplio y soporte de tipado estático, lo que permite unificar el desarrollo frontend y backend bajo un mismo lenguaje.

### Anexo 8. Comparativa de Frameworks (matriz de evaluación)

Fuentes: angular.dev, react.dev, vuejs.org, expressjs.com (enlaces en las referencias)

#### FRAMEWORKS FRONTEND

Criterio	Angular 17	React 18	Vue.js 3
Desarrollador	Google	Meta (Facebook)	Comunidad
Lenguaje base	TypeScript (nativo)	JavaScript/JSX	JavaScript
Arquitectura	Componentes Standalone, MVC	Componentes, Virtual DOM	MVVM, Componentes
Herramientas incluidas	Router, Forms, HTTP, CLI	Solo librería UI	Router, opcionales, Vuex
Curva de aprendizaje	Media-Alta	Media	Baja
PUNTUACIÓN (1-5)	4.5	4.0	4.0

#### FRAMEWORKS BACKEND

Criterio	Express.js	NestJS
Tipo	Minimalista, flexible	Estructurado, opinado
Curva de aprendizaje	Baja	Media-Alta
Documentación	Amplia, comunidad activa	Buena, en crecimiento
PUNTUACIÓN (1-5)	4.5	4.0

Angular obtiene la mayor puntuación (4.6/5) debido a su arquitectura basada en componentes que permite estructurar el sistema en módulos independientes (productos, ventas, inventario, reportes), su tipado estricto con TypeScript que garantiza la precisión en cálculos de precios y cantidades, y la integración nativa de Angular Material que proporciona componentes de interfaz profesionales. Estas características responden a los requerimientos del Minimarket Otavalo, particularmente la necesidad de una interfaz fácil de usar y un sistema modular que facilite el mantenimiento.

Express.js se selecciona para el backend (4.5/5) por su simplicidad para crear APIs REST, su compatibilidad directa con middleware de autenticación JWT requerido para el control de acceso por roles, y su integración eficiente con MySQL mediante el paquete mysql2. Adicionalmente, permite mantener todo el proyecto bajo un único lenguaje (JavaScript/TypeScript), reduciendo la complejidad del desarrollo y facilitando el mantenimiento del sistema.

### Anexo 9: Sistemas de base de datos (matriz de evaluación)

## Fuentes oficiales:

- MySQL: [dev.mysql.com/doc/refman/8.0/en/](https://dev.mysql.com/doc/refman/8.0/en/)
- PostgreSQL: [postgresql.org/docs/current/runtime-config-connection.html](https://postgresql.org/docs/current/runtime-config-connection.html)
- MariaDB: [mariadb.com/docs/server/server-management/variables-and-modes/](https://mariadb.com/docs/server/server-management/variables-and-modes/)

Criterio	MySQL 8.0	PostgreSQL 16	MariaDB 10.x
Tipo	Relacional	Relacional-Objeto	Relacional
Conexiones por defecto	151	100	151
Máximo conexiones	100,000 (según HW)	262,143	100,000 (según HW)
Nivel de seguridad	Medio-Alto (SSL/TLS)	Alto (SSL, Row Security)	Medio-Alto (SSL/TLS)
Licencia	GPL + Comercial	PostgreSQL License	GPL (100% libre)
Compatibilidad Node.js	Excelente (mysql2)	Buena (pg)	Excelente (mysql2)
Documentación en español	Amplia	Limitada	Media
Facilidad de uso	Alta	Media	Alta
PUNTUACIÓN (1-5)	4.5	4.0	4.0

MySQL 8.0 obtiene la mayor puntuación (4.5/5) por su capacidad de soportar hasta 500 conexiones concurrentes (superando ampliamente las necesidades del Minimarket que opera con 2-3 usuarios), su nivel de seguridad mediante cifrado de conexiones y autenticación robusta, su amplia documentación disponible en español, y su compatibilidad probada con Node.js a través del paquete mysql2. Además, se distribuye bajo licencia gratuita para proyectos comerciales pequeños, alineándose con las limitaciones presupuestarias del negocio identificadas durante el levantamiento de información.

## Anexo 10: Comparativa IDEs (matriz de evaluación)

**Fuentes:** [code.visualstudio.com](https://code.visualstudio.com), [jetbrains.com/webstorm](https://jetbrains.com/webstorm)

Criterio	VS Code	WebStorm	Sublime Text
Desarrollador	Microsoft	JetBrains	Sublime HQ
Costo	Gratuito	\$69/año	\$99 (perpetua)
Soporte TypeScript	Nativo, excelente	Nativo, excelente	Vía plugins
Extensiones Angular	Angular Language Service	Integrado	Limitado
Consumo de RAM	~300-500 MB	~1-2 GB	~100-200 MB
Terminal integrada	Sí	Sí	No nativa
Depurador integrado	Sí	Sí	No
PUNTUACIÓN (1-5)	4.5	4.5	3.0

Visual Studio Code se selecciona (4.7/5) por su soporte nativo para TypeScript con autocompletado inteligente y verificación de tipos en tiempo real, su amplio ecosistema de extensiones específicas para Angular, Node.js y MySQL, su integración nativa con Git para el control de versiones, y su bajo consumo de recursos (300-400 MB de RAM) que permite ejecutar simultáneamente el editor y los servidores de

desarrollo. Su distribución gratuita y la amplia comunidad de usuarios garantizan actualizaciones frecuentes y recursos de aprendizaje disponibles.

### Anexo 11. Metodologías de desarrollo (Fichas técnicas)

**Fuentes:** [extremeprogramming.org](http://extremeprogramming.org), [scrum.org](http://scrum.org), [kanban.university](http://kanban.university)

Criterio	XP	Scrum	Kanban
Enfoque principal	Calidad técnica, entregas frecuentes	Gestión por sprints, roles definidos	Flujo continuo, visualización
Tamaño de equipo ideal	1-12 personas	5-9 personas	Cualquier tamaño
Iteraciones	1-2 semanas	2-4 semanas (sprints)	Sin iteraciones fijas
Flexibilidad a cambios	Alta - cambios bienvenidos	Media - cambios entre sprints	Alta - flujo continuo
Documentación requerida	Mínima	Media (backlog, burndown)	Mínima (tablero)
Comunicación con cliente	Constante, cliente in-situ	Por sprint (review)	Según necesidad
PUNTUACIÓN (1-5)	4.5	4.0	3.5

XP obtiene la mayor puntuación (4.4/5) por su adaptabilidad a equipos pequeños de desarrollo, su enfoque en entregas frecuentes e incrementales que permite obtener retroalimentación temprana del propietario del Minimarket, y su énfasis en la comunicación directa con el cliente para validar funcionalidades. A diferencia de Scrum que requiere roles diferenciados, XP puede aplicarse efectivamente en proyectos individuales, priorizando el software funcional sobre la documentación exhaustiva y permitiendo ajustes rápidos en los requerimientos durante el desarrollo.

### Anexo 12. Arquitectura del sistema (Diagrama de 3 capas)

El sistema se va a diseñar bajo una arquitectura de tres capas que separa las responsabilidades en: Presentación, Lógica de Negocio y Datos.

Capa	Tecnología	Responsabilidades
PRESENTACIÓN	Angular 17 + Angular Material + TypeScript	Interfaz de usuario, validaciones del lado cliente, navegación, componentes visuales, formularios reactivos
LÓGICA DE NEGOCIO	Node.js + Express.js + JWT + bcryptjs	API REST, autenticación/autorización, reglas de negocio, validaciones, middleware de seguridad
DATOS	MySQL 8.0 + InnoDB	Almacenamiento persistente, integridad referencial, transacciones ACID, índices para optimización

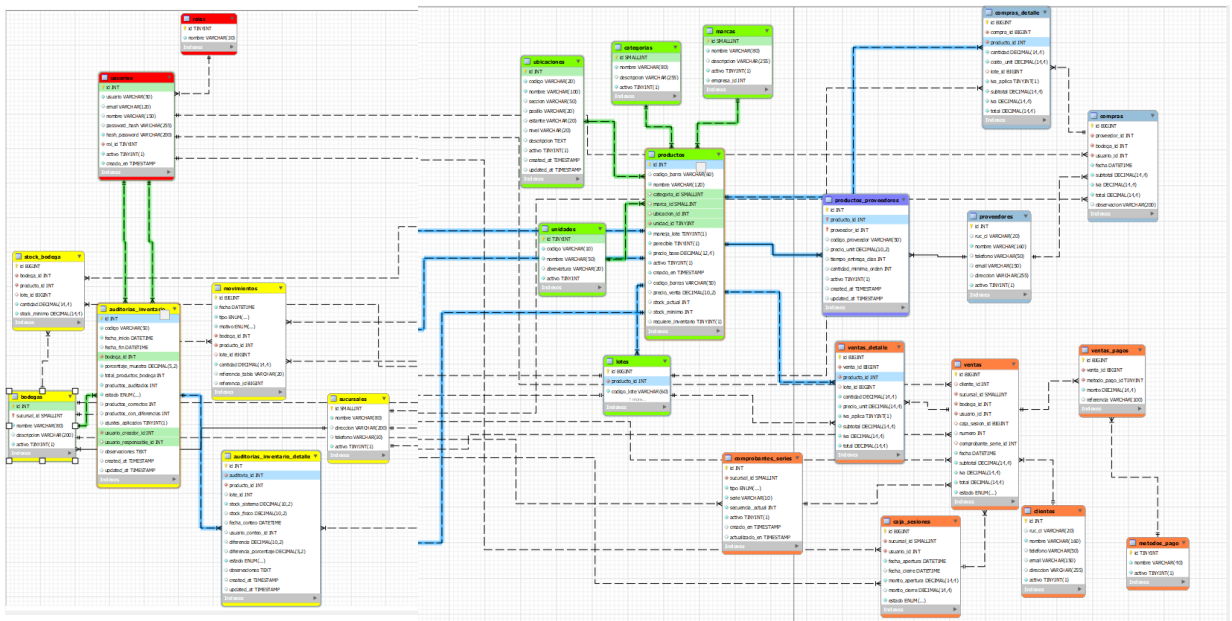
#### Comunicación entre capas:

- Frontend y Backend: HTTP/HTTPS mediante API REST (JSON)
- Backend y base de datos: conexión TCP
- Autenticación: JWT (JSON Web Tokens) con expiración configurable

### Anexo 13. Entidades de la base de datos

El modelo de base de datos está compuesto por 27 tablas que cubren todos los módulos del sistema, pero las entidades principales son:

Entidad	Descripción y Campos Principales
productos	Catálogo de productos: id, codigo_barra, nombre, categoria_id, marca_id, ubicacion_id, unidad_id, precio_base, precio_venta, stock_actual, stock_minimo, perecible, maneja_lote
categorias	Clasificación de productos: id, nombre, descripcion, activo
marcas	Marcas comerciales: id, nombre, descripcion, activo
proveedores	Datos de proveedores: id, ruc_ci, nombre, telefono, email, direccion, activo
compras	Registro de compras: id, proveedor_id, bodega_id, usuario_id, fecha, subtotal, iva, total, observacion
compras_detalle	Detalle de compra: id, compra_id, producto_id, cantidad, costo_unit, lote_id, iva_aplica, subtotal, iva, total
ventas	Registro de ventas: id, cliente_id, sucursal_id, bodega_id, usuario_id, caja_sesion_id, numero, comprobante_serie_id, ip_address, fecha, subtotal, iva, total, estado
ventas_detalle	Detalle de venta: id, venta_id, producto_id, lote_id, cantidad, precio_unit, iva_aplica, subtotal, iva, total
movimientos_stock	Kardex: id, producto_id, lote_id, tipo (ENTRADA/SALIDA/AJUSTE/TRASLADO), cantidad, stock_anterior, stock_nuevo, referencia_tipo, referencia_id, usuario_id, observacion, fecha
usuarios	Usuarios del sistema: id, usuario, email, nombre, password_hash, rol_id, activo, creado_en
roles	Roles de usuario: id, nombre (ADMIN, GERENTE, CAJERO, BODEGUERO)
Caja_sesiones	Control de caja: id, sucursal_id, usuario_id, fecha_apertura, fecha_cierre, monto_apertura, monto_cierre, estado
auditoria	Bitácora: id, usuario_id, usuario_nombre, accion, modulo, tabla_afectada, registro_id, datos_anteriores, datos_nuevos, fecha
lotes	Control de lotes: id, codigo_lote, producto_id, fecha_caducidad
ubicaciones	Localización física: id, codigo, nombre, seccion, pasillo, estante, nivel, descripcion, activo
bodegas	Almacenes: id, sucursal_id, nombre, descripcion, activo



## Anexo 14. Normalización de la base de datos hasta 3FN

Tabla Productos sin normalizar

id	nombre	categoría	marca	proveedor	telefono_prov	ubicacion
1	Arroz 1kg	Granos	La Favorita	Dist. Norte	0998765432	Bodega-Est1-Niv1
2	Aceite 1L	Aceites	La Favorita	Dist. Norte	0998765432	Bodega-Est2-Niv1
3	Leche 1L	Lácteos	Vita	Lácteos EC	0991234567	Bodega-Est3-Niv2

Tablas normalizadas hasta su 3FN

Tabla: categorías

id	nombre	descripcion	activo
1	Granos	Arroz, frijoles, lentejas	1
2	Lácteos	Leche, queso, yogurt	1
3	Aceites	Aceites vegetales y de oliva	1

Tabla: Marcas

id	nombre	descripcion	activo
1	La Favorita	Productos alimenticios	1
2	Vita	Productos lácteos	1

Tabla: roles

id	nombre
1	ADMIN
2	GERENTE
3	CAJERO
4	BODEGUERO

Tabla: Productos con claves foráneas

id	codigo_barra	nombre	categoria_id	marca_id	unidad_id	precio_venta	stock_actual
1	7861234567	Arroz 1kg	1 → FK	1 → FK	1 → FK	1.25	50
2	7869876543	Leche 1L	2 → FK	2 → FK	2 → FK	1.10	30

Tabla: Usuarios

id	usuario	nombre	email	rol_id
1	admin	Administrador Sistema	admin@mini.com	1 → FK
2	cajero1	Juan Pérez	juan@mini.com	3 → FK

## Anexos 15, 16 y 17:

Los anexos 14 Historias de Usuario, 15 Diagramas de Casos de Uso y 16 Diseño de Interfaces se encuentran documentados en la sección de desarrollo del sistema

dentro de la propuesta de la presente tesis, ya que corresponden a los elementos generados durante la aplicación de la metodología XP.

#### **Anexo 18.** Entrevista al dueño del negocio

**Pregunta 1.** ¿Cómo registra actualmente los productos que vende en su Minimarket?

Ahorita lo que hago es anotar los productos en un cuaderno cuando llegan y también tengo unas hojas de Excel en la computadora. En el cuaderno escribo el nombre del producto, cuántas unidades llegaron y el precio de venta. A veces cuando tengo tiempo paso esa información al Excel, pero no siempre alcanzo porque hay mucho que hacer en el negocio."

**Pregunta 2.** Cuando le llega mercadería de los proveedores, ¿cómo la recibe y registra?

Cuando llega el proveedor, primero reviso que venga todo lo que pedí según la factura. Las facturas las guardo en una carpeta y cuando puedo anoto en el cuaderno lo que llegó. El problema es que a veces pasan días y ya no me acuerdo bien de todo lo que vino, especialmente cuando llegan varios proveedores el mismo día.

**Pregunta 3.** ¿Cómo registra las ventas que realiza durante el día?

Para las ventas uso una calculadora para sumar lo que compra el cliente. El dinero va al cajón y al final del día cuento cuánto quedó. Lo que vendí no lo anoto producto por producto porque no hay tiempo, solo sé cuánto dinero entró. Cuando veo que algo se acabó en la percha es cuando me doy cuenta que toca pedir más.

**Pregunta 4.** ¿Cómo sabe cuántos productos tiene disponibles en este momento?

Sinceramente, saber exacto cuánto tengo es difícil. Lo que hago es mirar las perchas y la bodega, y cuando veo que queda poco de algo, ahí apunto en un papelito para pedir. El conteo completo de todo el inventario lo hago como una vez al mes o cada dos meses, y siempre encuentro diferencias con lo que tenía anotado.

**Pregunta 5.** ¿Cómo maneja la información de sus proveedores?

Tengo los contactos de los proveedores en el celular y las tarjetas de presentación en una cajita. Las facturas que me dan las guardo para saber cuánto les debo y cuándo tengo que pagar. A veces me confundo con las fechas de pago porque algunos dan

crédito de 15 días, otros de 30, y cuando son varios proveedores es complicado acordarse de todo.

**Pregunta 6.** ¿Cómo controla los productos que tienen fecha de vencimiento?

Eso sí es un dolor de cabeza. Los lácteos, embutidos, pan de molde, jugos... todos esos hay que estar pendiente. Lo que hago es poner adelante los que van a vencer primero, pero a veces se me pasa y cuando me doy cuenta el producto ya venció. El mes pasado tuve que botar como 50 dólares en productos vencidos entre yogures y quesos.

**Pregunta 7.** ¿Cuáles son los principales problemas que enfrenta para manejar su inventario?

Los problemas son varios. Primero, no sé exactamente cuánto tengo de cada producto. Segundo, cuando hago el conteo siempre falta o sobra mercadería comparado con lo que anoté. Tercero, los productos se vencen porque no tengo cómo saber cuáles van a caducar pronto. Cuarto, hacer las notas de venta a mano es lento y a veces me equivoco.

**Pregunta 8.** ¿Ha identificado pérdidas de dinero relacionadas con el manejo del inventario?

Sí, hay varias formas en que pierdo plata. Los productos vencidos que toca botar, eso me pasa casi todos los meses. También cuando cuento el inventario y falta mercadería, no sé si es que anoté mal o qué pasó. A veces compro de más un producto porque no supe que todavía tenía en bodega, y después se queda ahí ocupando espacio.

**Pregunta 9.** Si tuviera un sistema para manejar el inventario, ¿qué le gustaría que pudiera hacer?

Me gustaría que, al vender, automáticamente se reste del inventario. Que me avise cuando un producto está por acabarse para pedir a tiempo. Que me diga cuáles productos van a vencer pronto. Que haga las facturas automáticas, rápido y sin errores. Y que al final del día me dé un resumen de cuánto vendí, de qué productos, cuánto gané.

**Pregunta 10.** ¿Qué tan familiarizado está con el uso de computadoras? ¿Estaría dispuesto a implementar un sistema?

Yo sé usar el Excel básico, el WhatsApp, navegar en internet. No soy experto pero tampoco me da miedo la tecnología. Lo que sí necesito es que el sistema sea fácil, con botones claros. Sí estaría dispuesto a implementarlo porque la verdad ya necesito ordenarme mejor. Si el sistema me ahorra tiempo y me ayuda a no perder dinero, para mí vale la pena.

### **Manual de usuario**



### **MANUAL DE USUARIO**

---

#### **Sistema de Gestión de Inventarios para Minimarket**

Año 2026

**Elaborado por:** Israel Salazar

**Fecha:** 04/03/2026

## INTRODUCCIÓN AL SISTEMA

### 1. ¿Qué es el sistema?

El Sistema de Gestión de Inventarios para Minimarket es una aplicación web diseñada para administrar de forma integral las operaciones del negocio: el control del inventario, el registro de ventas en el punto de venta, la gestión de compras a proveedores y el control de caja. El sistema centraliza toda la información del minimarket y la mantiene actualizada en tiempo real, lo que permite conocer en cualquier momento el stock disponible, las ventas realizadas y el estado financiero de cada jornada.

El presente manual describe paso a paso cada uno de los módulos del sistema. Los módulos se presentan siguiendo el flujo natural de trabajo del negocio: primero la configuración de los datos base y el ingreso de productos, luego el proceso de ventas y, finalmente, las herramientas de control y supervisión.

### Características generales

- Aplicación web de acceso local a través del navegador.
- Interfaz sencilla e intuitiva, organizada por módulos.
- Control de acceso mediante roles de usuario.
- Actualización del inventario en tiempo real.
- Generación de reportes e indicadores del negocio.

### 2. Requisitos para su uso

Para acceder y utilizar el sistema se requiere lo siguiente:

- Un computador de escritorio o portátil.
- Un navegador web actualizado (se recomienda Google Chrome).
- Conexión al servidor local donde se ejecuta el sistema.
- Dirección de acceso: <http://localhost:4200>
- Credenciales de usuario (nombre de usuario y contraseña) asignadas por el administrador.

### Ficha técnica del sistema

Campo	Detalle
<b>Nombre del sistema</b>	Sistema de Gestión de Inventarios para Minimarket
<b>Tipo</b>	Aplicación web de escritorio (local)
<b>Tecnología del frontend</b>	Angular 17 con Angular Material

Campo	Detalle
<b>Tecnología del backend</b>	Node.js + Express
<b>Base de datos</b>	MySQL
<b>Acceso</b>	Navegador web (http://localhost:4200)
<b>Autenticación</b>	JWT (usuario y contraseña)

### 3. Roles y permisos de usuario

El sistema controla el acceso a sus módulos mediante cuatro roles de usuario. Cada rol determina qué módulos puede ver el usuario y qué acciones puede ejecutar. De esta manera, cada persona accede únicamente a las funciones que le corresponden según su cargo.

Rol	Descripción	Módulos a los que accede
<b>ADMIN</b>	Administrador del sistema	Acceso total a todos los módulos.
<b>GERENTE</b>	Gerente de tienda	Reportes, historial de caja, ventas, compras e inventario. No puede abrir ni cerrar caja.
<b>CAJERO</b>	Cajero de punto de venta	Punto de venta y su propia caja; visualiza únicamente sus ventas.
<b>BODEGUERO</b>	Encargado de bodega	Inventario, percha, compras y kardex.

## ACCESO Y NAVEGACIÓN

### 4. Inicio de Sesión

Es la pantalla de autenticación que permite el acceso al sistema mediante usuario y contraseña. El sistema valida las credenciales, asigna el rol correspondiente y redirige automáticamente al panel principal.

#### Funcionalidades principales

- Campos para ingresar el usuario y la contraseña.
- Botón "Iniciar Sesión".
- Validación de las credenciales en tiempo real.
- Redirección automática al panel principal según el rol.
- Mensaje de error cuando las credenciales son incorrectas.

#### Proceso paso a paso

1. Ingrese su nombre de usuario.
2. Escriba su contraseña.
3. Presione el botón "Iniciar Sesión".
4. El sistema verifica las credenciales contra la base de datos.
5. Si son correctas, genera un token de sesión (JWT) y lo redirige al panel principal.



**¡Bienvenido!**  
Ingresa tus credenciales para acceder al sistema

**Usuario o Email**  
Ingresa tu usuario o email

**Contraseña**  
Ingresa tu contraseña

Recordarme [¿Olvidaste tu contraseña?](#)

**Iniciar Sesión**

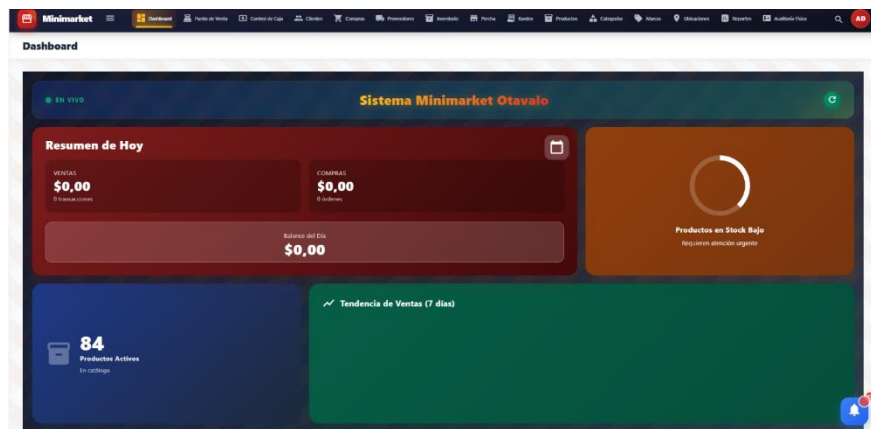
Prueba con: `admin / Admin123*`

© 2025 Minimarket Otavalo. Todos los derechos reservados.

## 5. Panel Principal (Dashboard)

Es la pantalla de inicio que se muestra después de iniciar sesión. Presenta un resumen ejecutivo del estado del negocio en tiempo real, con indicadores clave de ventas, compras, stock e inventario. El contenido puede variar según el rol del usuario.

- Tarjetas de resumen: total de ventas del día, total de compras, productos en stock y alertas activas.
- Menú lateral con acceso rápido a los módulos principales.
- Barra de navegación superior con todos los módulos.
- Menú de perfil en la esquina superior derecha (Configuración, Usuarios, Cerrar sesión).
- Botón de notificaciones con un contador de alertas.

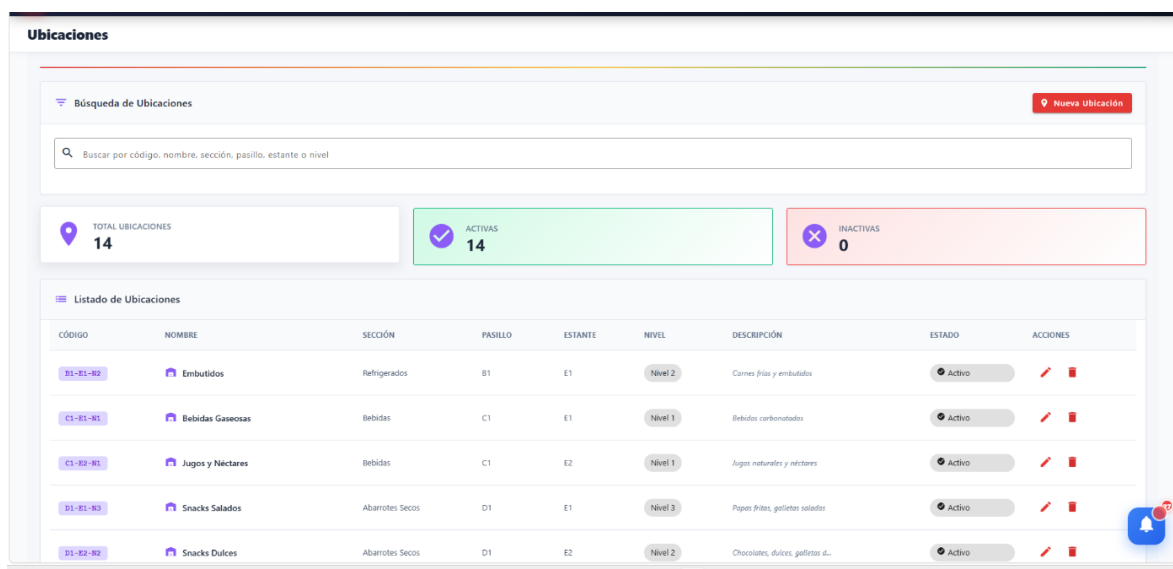


## CONFIGURACIÓN INICIAL

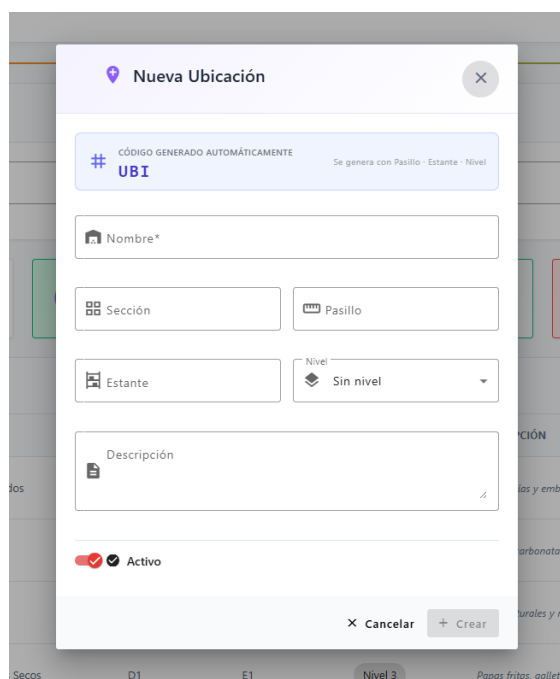
Antes de registrar compras y ventas, el sistema necesita tener configurada la información base del negocio. En esta sección se crean las ubicaciones, las categorías, las marcas, los proveedores y los productos. Esta configuración se realiza una sola vez al inicio y luego se actualiza cuando sea necesario.

### 6. Ubicaciones (Bodega)

Permite gestionar las ubicaciones físicas dentro del minimarket, la bodega donde se almacenan los productos. Las bodegas son indispensables para poder registrar compras e inventario.



CÓDIGO	NOMBRE	SECCIÓN	PASILLO	ESTANTE	NIVEL	DESCRIPCIÓN	ESTADO	ACCIONES
B1-E1-N2	Embutidos	Refrigerados	B1	E1	Nivel 2	Carnes frías y embutidos	Activo	[Edit] [Delete]
C1-E1-N1	Bebidas Gaseosas	Bebidas	C1	E1	Nivel 1	Bebidas carbonatadas	Activo	[Edit] [Delete]
C1-E2-N1	Jugos y Néctares	Bebidas	C1	E2	Nivel 1	Jugos naturales y néctares	Activo	[Edit] [Delete]
D1-E1-N3	Snacks Salados	Albarotes Secos	D1	E1	Nivel 3	Papas fritas, galletas saladas	Activo	[Edit] [Delete]
D1-E2-N2	Snacks Dulces	Albarotes Secos	D1	E2	Nivel 2	Chocolates, dulces, galletas d...	Activo	[Edit] [Delete]



**Nueva Ubicación**

CÓDIGO GENERADO AUTOMÁTICAMENTE  
# UBI (Se genera con Pasillo - Estante - Nivel)

Nombre\*

Sección Pasillo

Estante Nivel: Sin nivel

Descripción

Activo

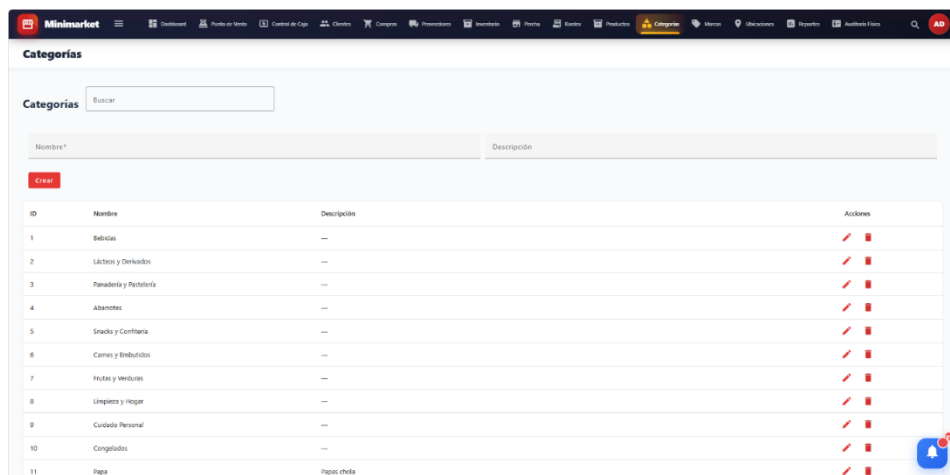
Cancelar + Crear

## 7. Categorías

Permite gestionar las categorías que clasifican los productos del minimarket (por ejemplo: Lácteos, Bebidas, Snacks, etc.). Toda categoría se usa después al crear los productos.

### Funcionalidades principales

- Listado de categorías existentes.
- Crear una nueva categoría (nombre y descripción).
- Editar o eliminar categorías.

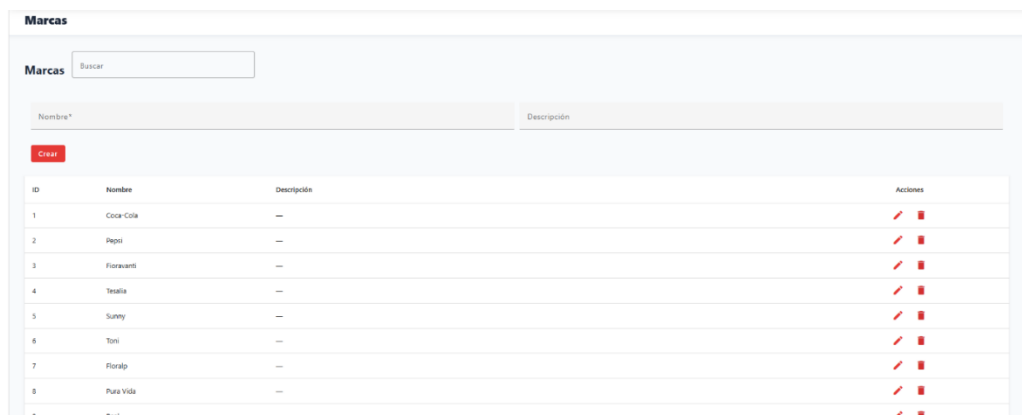


## 8. Marcas

Permite gestionar el catálogo de marcas de los productos. Las marcas se asignan a cada producto al momento de crearlo.

### Funcionalidades principales

- Listado de marcas registradas.
- Crear una nueva marca.
- Editar o eliminar marcas.



## 9. Proveedores

Permite gestionar el catálogo de proveedores del minimarket. El registro de proveedores es un requisito previo para poder ingresar compras de mercadería, ya que toda compra debe asociarse a un proveedor.

### Funcionalidades principales

- Listado de proveedores con búsqueda por nombre o RUC.
- Registrar un nuevo proveedor: RUC, nombre o razón social, teléfono, correo electrónico y dirección.
- Editar la información de un proveedor.
- Activar o desactivar un proveedor.

The screenshot displays the 'Gestión de Proveedores' (Provider Management) interface. It features a 'Nuevo Proveedor' (New Provider) form with the following fields: 'RUC / Cédula \*\*' (10 digits for RUC, 13 for Cédula), 'Razón Social / Nombre \*\*', 'Teléfono' (10 digits), 'Email' (e.g., contacto@empresa.com), and 'Dirección'. A 'Proveedor Activo' checkbox is checked. A '+ Crear' button is at the bottom right. Below the form is a 'Listado de Proveedores (10)' section with a search bar. A table lists the following provider:

RUC	PROVEEDOR	TELÉFONO	EMAIL	DIRECCIÓN	ESTADO	ACCIONES
17999999001	Proveedor Otavalo	099999999	prov@otavalo.local	Quito	Activo	[Edit] [Delete] [Details]

## 10. Productos



















Permite gestionar el catálogo completo de productos disponibles en el minimarket. Cada producto se clasifica por categoría y marca, y se le define su precio de venta, su precio de costo y su stock mínimo para las alertas automáticas.

### Funcionalidades principales

- Listado de productos con búsqueda y filtros.
- Registrar un nuevo producto: nombre, código de barras, categoría, marca, unidad de medida, precio de venta, precio de costo, stock mínimo e imagen.
- Editar un producto existente.
- Activar o desactivar un producto.
- Control de stock mínimo para generar alertas automáticas.

### Productos

Listado de Productos (82) 🔍 Buscar por código o nombre

CÓDIGO	PRODUCTO	CATEGORÍA	MARCA	PROVEEDOR	UBICACIÓN	PRECIO	ESTADO	ACCIONES
11844898442188888478	Pan Integral UND	Panadería	Bimbo	Bimbo	D1-E1-N3	\$1.00	Activo	 
565654645645	Leche Tony UND	Lácteos y Derivados	Tony	Lácteos Del Norte	I1-E1-N1	\$0.80	Activo	 
987897879709	Seven Up UND	Bebidas	La original	Distribuidora de Bebidas S.A.	C1-E1-N1	\$1.25	Activo	 
TEST-LIM-003	Desinfectante Lysol 500ml UND	Limpieza	Genérica	Higiene y Limpieza Cia. Ltda.	G1-E1-N2	\$2.20	Activo	 
TEST-LIM-002	Limpiaplus Fabuloso 1L ARROBA	Limpieza	Genérica	Higiene y Limpieza Cia. Ltda.	—	\$1.80	Activo	 
TEST-LIM-001	Detergente Fab 500g ARROBA	Limpieza	Genérica	Higiene y Limpieza Cia. Ltda.	—	\$1.10	Activo	 
TEST-BIG-003	Shampoo Dove 400ml ARROBA	Higiene Personal	Dove	Higiene y Limpieza Cia. Ltda.	—	\$3.50	Activo	 
TEST-BIG-002	Pasta Dental Colgate 75ml ARROBA	Higiene Personal	Colgate	Higiene y Limpieza Cia. Ltda.	—	\$1.20	Activo	 
TEST-BIG-001	Jabon Dove 90g ARROBA	Higiene Personal	Dove	Higiene y Limpieza Cia. Ltda.	—	\$0.90	Activo	 

### Productos

#### Gestión de Productos

Administra tu catálogo de productos

**Editar Producto**

Código de Bares:  Nombre del Producto:

Categoría:  Marca:  Proveedor:

Ubicación en Bodega:  Precio de Compra:  Precio de Venta:

Activo

Cancelar Actualizar

## **INGRESO Y GESTIÓN DE PRODUCTOS**

Una vez configurados los datos base, esta sección describe cómo ingresa la mercadería al sistema y cómo se gestiona el inventario: el registro de compras, la consulta del inventario, el traslado de productos a la percha y el seguimiento de los movimientos a través del kardex.

### **11. Compras**

Permite registrar las órdenes de compra a los proveedores. Al confirmar una compra, el sistema ingresa los productos al inventario con su costo, su lote y su fecha de vencimiento, e incrementa automáticamente el stock de la bodega seleccionada.

#### **Listado de compras**

- Tabla con: ID, fecha, proveedor, usuario, total, estado y acciones.
- Filtros por fecha y por proveedor.
- Opción para ver el detalle de cada compra.

#### **Nueva compra**

- Selección del proveedor (obligatorio).
- Selección de la bodega de destino.
- Agregar productos indicando cantidad, precio de costo, lote y fecha de vencimiento.
- Tabla de items con los totales de la compra.
- Al confirmar, el stock se actualiza automáticamente.

#### **Proceso paso a paso**

1. Presione "Nueva Compra".
2. Seleccione el proveedor.
3. Seleccione la bodega donde ingresará el producto.
4. Busque y agregue cada producto con su cantidad y su precio de costo.
5. Ingrese el número de lote y la fecha de vencimiento, si corresponde.
6. Revise el total de la compra.
7. Confirme la compra: el sistema incrementa el stock en la bodega.

**Compras**

**Gestión de Compras**  
Administra tus compras y adquisiciones

Filtros de Búsqueda

Fecha Desde: dd/mm/aaaa    Fecha Hasta: dd/mm/aaaa    Proveedor:

TOTAL COMPRAS: **41**    TOTAL GENERAL: **\$62.10**

Listado de Compras

ID	FECHA	PROVEEDOR	PRODUCTOS	SUBTOTAL	IVA	TOTAL	ACCIONES
#43	18/05/2026, 05:52 p. m.	Distribuidora Granos del Sur	Arroz Largo Fino 2kg x 30	\$54.00	\$8.10	\$62.10	
#44	16/05/2026, 11:37 p. m.	Distribuidora de Bebidas S.A.	Aceite El Cocinero 1L x 20	\$50.00	\$7.50	\$57.50	
#43	16/05/2026, 11:09 p. m.	Distribuidora Granos del Sur	Nachos Tostitos 10g x 50	\$44.00	\$6.60	\$50.60	
#42	16/05/2026, 10:56 p. m.	Proveedora Carnicos del Norte	Chorizo Artesanal x 15	\$22.50	\$3.38	\$25.88	

**Compras**

Registra una nueva compra al proveedor

Información de la Compra

Proveedor: Proveedora Carnicos del Norte    Observación (Opcional):

Agregar Productos

IVA %: 15

Buscar Producto:     Cantidad:     Costo Unitario: \$ 0

Código de Lote (Opcional):     Fecha de Caducidad (Opcional): dd/mm/aaaa     Aplica IVA (15%)

Año entre 2026 y 2026

Productos Agregados (1)

Producto	Cantidad	Costo Unit.	Lote	F. Caducidad	Subtotal	IVA (15%)	Total	Acciones
Chorizo Ahumado 250g	<input type="text" value="1"/>	\$1.80	12401240234	-	\$1.80	\$0.27	\$2.07	

Subtotal: \$1.80  
IVA (15%): \$0.27  
**TOTAL: \$2.07**

## 12. Inventario

Muestra una vista consolidada del stock actual de todos los productos del sistema, diferenciando el stock en bodega del stock en percha (estantería de venta). Es la pantalla de referencia para conocer la disponibilidad real de cada producto.

### Funcionalidades principales

- Tabla con todos los productos y su stock actual.
- Columnas: producto, categoría, stock en bodega, stock en percha, stock total, precio de venta y estado.
- Indicadores visuales de stock bajo (rojo) y stock normal (verde).

- Filtros por categoría y por nombre.
- Exportación a Excel o PDF.

**Inventario**  
 Controle el stock de productos y movimientos

Inventario Completo    Stock Bajo    Movimientos de Stock    Próximos a Vencer    Historial de Bajas

Buscar producto (nombre o código)    Categoría: Todas    Marca: Todas    Limpiar

Total de productos: 82

CÓDIGO	NOMBRE	CATEGORÍA	MARCA	UBICACIÓN	PRECIO VENTA	STOCK	VENCIMIENTO	ESTADO	ACCIONES
7750885300005	Aceite El Cocinero 1L	Abarrotes	El Cocinero	11-11-11	\$3.25	9		ACTIVO	
TEST-BEB-003	Agua Nestlé 500ml	Bebidas	Nestlé	—	\$0.60	48	18/05/2026	ACTIVO	
7750885033335	Agua Tesalia 2.5L	Bebidas	Tesalia	11-11-11	\$0.65	23		ACTIVO	
7750885033334	Agua Tesalia 625ml	Bebidas	Tesalia	11-11-11	\$0.35	49	30/08/2026	ACTIVO	
TEST-GRA-001	Arroz Largo Fino 2kg	Granos y Harinas	Genérica	—	\$2.50	30	04/09/2026	ACTIVO	
7750885300001	Arroz Real 1kg	Abarrotes	Real	01-11-13	\$0.95	0		ACTIVO	

### 13. Percha (Transferencia Bodega → Percha)

Permite trasladar productos desde la bodega hacia la percha (estantería de venta). Se utiliza para reabastecer el punto de venta cuando se agota el stock visible al cliente. La transferencia descuenta el producto de la bodega y lo suma a la percha.

#### Transferencia Bodega → Percha

- Seleccionar el producto.
- Ver el stock disponible en bodega.
- Ingresar la cantidad a trasladar.
- Confirmar la transferencia.

#### Historial de transferencias

- Registro de todos los movimientos entre bodega y percha.
- Fecha, producto, cantidad y usuario responsable.

#### Proceso paso a paso

1. Verifique que haya stock disponible en la bodega.
2. Seleccione el producto que desea reabastecer.
3. Ingrese la cantidad a mover hacia la percha.
4. Confirme la transferencia: el sistema actualiza ambos stocks.

Percha

**Gestión de Percha** Actualizar

Control de stock en estantería de venta

1. Compra

Desde Ingresos a bodega

2. Bodega

Almacén principal

3. Percha

Estantería de venta

4. Venta

De documento de percha

**16** Productos en Percha

**4** Requieren Reposición

**55** Disponibles en Bodega

Percha Actual
Bodega - Percha
Historial

CÓDIGO	PRODUCTO	CATEGORÍA	EN PERCHA	STOCK MÍNIMO	DISPONIBLE BODEGA	ESTADO	REPONER
7758085380905	Acelite El Cocinero 1L	Abarrotes	8 uds	5 uds	0 uds	OK	Reponer
TEST-BEB-003	Agua Nestle 500ml	Bebidas	9 uds	5 uds	48 uds	BAJO	Reponer
7758085033335	Agua Tesalia 2.5L	Bebidas	6 uds	5 uds	23 uds	OK	Reponer
7758085022224	Agua Tesalia 625ml	Bebidas	10 uds	5 uds	49 uds	OK	Reponer
7758085380001	Arroz Real 1kg	Abarrotes	8 uds	5 uds	0 uds	BAJO	Reponer
7758085380002	Arroz Real 5kg	Abarrotes	8 uds	5 uds	87 uds	OK	Reponer

Percha

**16** Productos en Percha

**4** Requieren Reposición

**55** Disponibles en Bodega

Percha Actual
Bodega - Percha
Historial

1. Busca el producto → 2. Haz clic en "Enviar a Percha" → 3. Confirma la transferencia

que quieras poner en la estantería      • indica cuántas unidades      • el stock pasa de bodega a percha

55 productos disponibles en bodega

CÓDIGO	PRODUCTO	CATEGORÍA	STOCK EN BODEGA	YA EN PERCHA	ACCIÓN
TEST-BEB-003	Agua Nestle 500ml	Bebidas	48 uds	0 uds	Enviar a Percha
7758085033335	Agua Tesalia 2.5L	Bebidas	23 uds	6 uds	Enviar a Percha
7758085033334	Agua Tesalia 625ml	Bebidas	49 uds	10 uds	Enviar a Percha
TEST-GRN-001	Arroz Largo Fino 2kg	Granos y Harinas	39 uds	0 uds	Enviar a Percha
7758085380002	Arroz Real 5kg	Abarrotes	87 uds	8 uds	Enviar a Percha
7758085380000	Atún Real Primavera 170g	Abarrotes	41 uds	7 uds	Enviar a Percha
7758085380006	Azúcar San Carlos 1kg	Abarrotes	112 uds	8 uds	Enviar a Percha
7758085700003	Cebolla Palteña x lb	Frutas y Verduras	89 uds	10 uds	Enviar a Percha
7758085400005	Chocolate La Universal Blanco 30g	Snacks y Confitería	120 uds	0 uds	Enviar a Percha

**Percha**

Control de stock en estantería de venta

1. Compra → 2. Bodega → 3. Percha → 4. Venta

Stock Ingres a bodega → Almacén principal → Estantería de venta → Se descuenta de percha

16 Productos en Percha | 4 Requieren Reposición | 55 Disponibles en Bodega

Percha Actual | Bodega - Percha | Historial

FECHA / HORA	PRODUCTO	CANTIDAD TRANSFERIDA	BODEGA ANTES	BODEGA DESPUÉS	RESPONSABLE
16/05/2026, 23:59	Arroz Real 1kg 7750885300003	1 uds	93 uds	92 uds	admin
16/05/2026, 23:53	Aceite El Cocinero 1L 7750885300005	24 uds	24 uds	0 uds	admin
16/05/2026, 23:49	Galletas Ricas Soda 315g 7750885300010	30 uds	30 uds	0 uds	admin
16/05/2026, 23:48	Galletas Ricas Soda 315g 7750885300010	20 uds	50 uds	30 uds	admin
16/05/2026, 23:38	Chocolate La Universal Clásico 30g 7750885400004	20 uds	120 uds	100 uds	Bodeguero
16/05/2026, 23:29	Agua Nestle 500ml TEST-BEB-003	2 uds	50 uds	48 uds	Bodeguero
16/05/2026, 23:13	Aceite El Cocinero 1L 7750885300005	20 uds	24 uds	4 uds	Bodeguero
16/05/2026, 23:11	Arroz Real 5kg 7750885300002	1 uds	88 uds	87 uds	Bodeguero
16/05/2026, 22:41	Chorizo Artesanal 02548925458	1 uds	1 uds	0 uds	admin

## 14. Kardex

Es el registro histórico de todos los movimientos de stock de cada producto. Permite rastrear las entradas, salidas y transferencias para realizar una auditoría completa del inventario.

### Funcionalidades principales

- Búsqueda por producto.
- Filtro por rango de fechas.
- Tabla con: fecha, tipo de movimiento, cantidad, stock anterior, stock resultante, usuario y referencia.
- Tipos de movimiento: COMPRA (entrada), VENTA (salida), TRANSFERENCIA y AJUSTE.
- Saldo acumulado por cada movimiento.

**Kardex**

Kardex de Inventario  
Historial completo de entradas, salidas y ajustes de stock

Buscar producto, proveedor, cliente... | Tipo: Todos | Desde: dd/mm/aaaa | Hasta: dd/mm/aaaa | Filtros | Limpiar

4430 Unidades Ingresadas | 1340 Unidades Vendidas/Salidas | 243 Movimientos Totales

Fecha y Hora	Tipo	Producto	Cantidad	Stock Ant.	Stock Act.	Proveedor / Cliente	Observación	Usuario
18/05/2026 03:57 p. m.	SALEDA	Aceite El Cocinero 1L (Proveedor)	-1	5	4		Venta #116 (desde percha)	admin
18/05/2026 03:48 p. m.	SALEDA	Aceite El Cocinero 1L (Proveedor)	-1	10	9		Venta #117 (desde percha)	admin
18/05/2026 03:45 p. m.	SALEDA	Arroz Real 1kg (Proveedor)	-90	90	0		Saldo por VENCIMIENTO etc...	admin
18/05/2026 03:02 p. m.	SALEDA	Arroz Real 1kg (Proveedor)	-3	3	0		Venta #116 (desde percha)	cap02
18/05/2026 02:07 p. m.	SALEDA	Aceite El Cocinero 1L (Proveedor)	-3	12	9		Venta #115 (desde percha)	cap02
18/05/2026 02:55 p. m.	SALEDA	Aceite El Cocinero 1L (Proveedor)	-5	14	9		Venta #114 (desde percha)	Cap0
18/05/2026 02:52 p. m.	COMPRA	Arroz Largo Fino 2kg (TEST-000-000)	30	0	30	Distribuidora Granos del Sur Comercio S.A.	Compra #42	admin
18/05/2026 02:40 p. m.	SALEDA	Agua Nestle 500ml (TEST-000-000)	-1	1	0		Venta #113 (desde percha)	admin

## PROCESO DE VENTAS

Con la mercadería ya ingresada y disponible en percha, esta sección describe el proceso de ventas: la gestión de clientes, el registro de ventas en el punto de venta y la consulta del historial de ventas.

### 15. Clientes

Permite gestionar el catálogo de clientes del minimarket. Posibilita registrar, editar y buscar clientes, con validación automática de la cédula ecuatoriana y del RUC.

#### Funcionalidades principales

- Listado de clientes con búsqueda en tiempo real.
- Registrar un nuevo cliente: cédula o RUC, nombre completo, teléfono, correo electrónico y dirección.
- Validación automática de la cédula y del RUC.
- Editar un cliente existente.
- Activar o desactivar un cliente.











El formulario muestra la opción de seleccionar 'Consumidor Final' o 'Nuevo Cliente'. Incluye campos para 'Cédula / RUC (opcional)', 'Nombre Completo \*\*' y 'Teléfono (opcional)'. Hay un campo para 'Agregar Productos' con un buscador de productos, una cantidad (1) y un precio unitario (\$ 0). Un IVA del 15% está predefinido. Botones de 'Cancelar' y 'Procesar Pago' están al final.

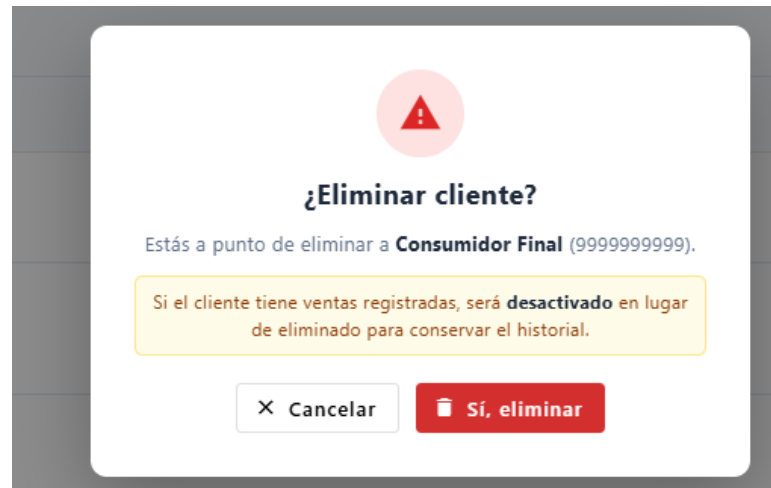
### Clientes

#### Gestión de Clientes

Listado de clientes registrados durante las ventas

Listado de Clientes (17)

RUC / CÉDULA	CLIENTE	TELÉFONO	ESTADO	ACCIONES
999999999	Consumidor Final	Sin teléfono	Activo	 
091111111	Juan Pérez Israel@demo.local Tulcán	099111111	Activo	 
040967972	Israel Salazar	0960976755	Activo	 
0401524893511	Luis Pupiales	0995478215	Activo	 
0401952448	Luis Ramirez	0950986482	Activo	 



## 16. Punto de Venta (Nueva Venta)

Es el módulo principal del cajero para registrar las ventas. Permite buscar productos, agregarlos al carrito, seleccionar el cliente y procesar el pago en efectivo. Para poder vender, la caja debe estar abierta previamente.

### Selección del cliente

- Modo "Consumidor Final" (sin identificación).
- Modo "Buscar Cliente" (búsqueda por nombre o cédula con autocompletado).
- Modo "Nuevo Cliente" (crea el cliente al momento, con validación de cédula o RUC).

### Agregar productos

- Buscador de productos con autocompletado (por nombre o código).
- Muestra el stock disponible en percha y en bodega.
- Campo de cantidad con validación de stock.
- Precio unitario fijo del catálogo (no editable).
- Botón "Agregar" para añadir el producto al carrito.

### Carrito de productos

- Lista de los productos agregados.
- Cantidad editable, con el límite del stock disponible.
- Casilla de IVA por producto (activar o desactivar).
- Subtotal, IVA y total por cada línea.
- Botón para eliminar un producto del carrito.

### Totales

- Subtotal general.

- IVA total (porcentaje configurable, por defecto 15%).
- Total a pagar, mostrado en tamaño grande.

### Proceso de pago

- El botón "Procesar Pago" abre el diálogo de cobro.
- Muestra el total a pagar.
- Indicador "Pago en Efectivo" (único método aceptado).
- Campo "Monto Recibido" para calcular el vuelto.
- Cálculo automático del cambio o vuelto.
- El botón "Confirmar Venta" registra la venta y genera el folio.

### Bloqueos automáticos

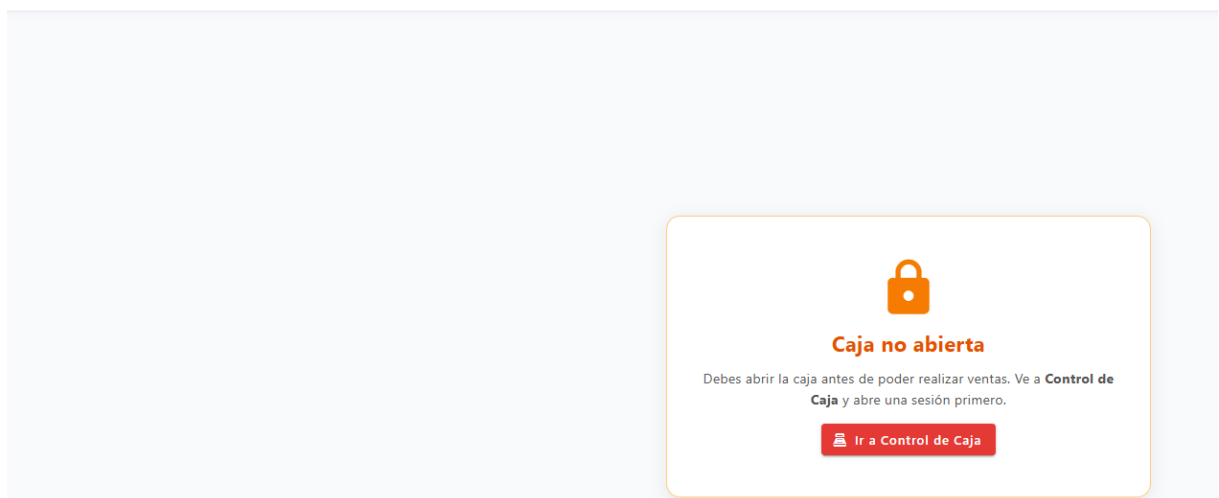
- Si la caja NO está abierta, se muestra la pantalla "Caja no abierta" con un botón para ir a Control de Caja.
- Si no hay stock en percha, se muestra una advertencia indicando el stock disponible en bodega.

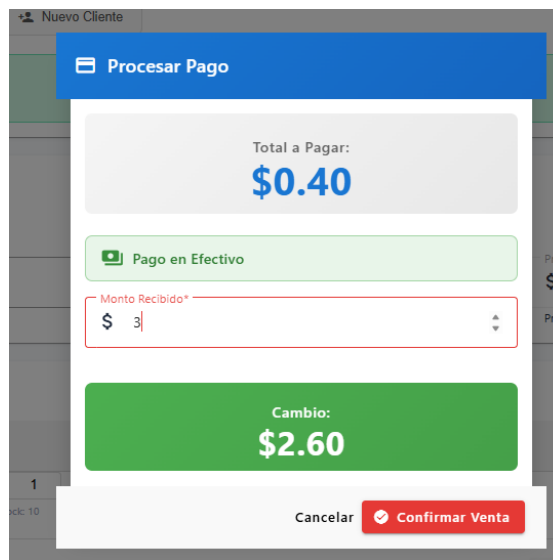
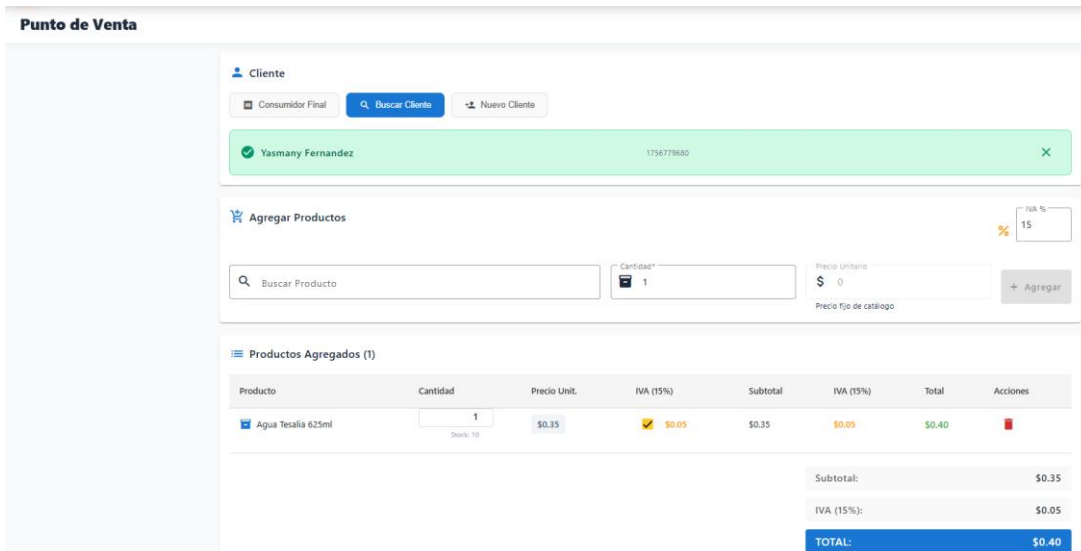
### Proceso paso a paso

1. Verifique que la caja esté abierta (si no lo está, vaya al módulo Control de Caja).
2. Seleccione el tipo de cliente.
3. Busque y seleccione los productos con su cantidad.
4. Revise los totales en el carrito.
5. Presione "Procesar Pago".
6. Ingrese el monto recibido del cliente.
7. Verifique el vuelto calculado.
8. Confirme la venta: el sistema descuenta el stock y genera el folio automáticamente.

---

#### Punto de Venta





## 17. Listado de Ventas

Muestra el historial de todas las ventas registradas. El CAJERO solo ve sus propias ventas; el ADMIN y el GERENTE ven las ventas de todos los cajeros.

### Funcionalidades principales

- Tabla con: ID, folio, fecha, cliente, subtotal, IVA, total, estado y acciones.
- Filtros por fecha (desde / hasta) y por cliente.
- Botón "Ver detalle" para cada venta.
- Resumen del monto total en el período filtrado.
- Paginación de los resultados.

### Detalle de una venta

- Datos del comprobante: folio, fecha, cliente y cajero.

- Tabla de productos vendidos con cantidades y precios.
- Subtotal, IVA y total.
- Información del pago registrado.
- Opción para imprimir o exportar.

**Punto de Venta**

Lista de Ventas

ID	FOLIO	FECHA	CLIENTE	SUBTOTAL	IVA	TOTAL	ESTADO	ACCIONES
#114	F001-00000078	18/05/2026, 03:52 p. m.	Yasmany Fernandez	\$3.25	\$0.49	\$3.74	EMITIDA	
#117	F001-00000077	18/05/2026, 03:48 p. m.	Camandor Final	\$3.25	\$0.49	\$3.74	EMITIDA	
#118	F001-00000076	18/05/2026, 03:02 p. m.	Camandor Final	\$3.25	\$0.49	\$3.74	EMITIDA	
#120	F001-00000075	18/05/2026, 02:57 p. m.	Leo Pagan	\$9.75	\$1.46	\$11.21	EMITIDA	
#121	F001-00000074	18/05/2026, 03:58 p. m.	Camandor Final	\$3.25	\$0.49	\$3.74	EMITIDA	
#123	F001-00000073	18/05/2026, 03:48 p. m.	Camandor Final	\$0.00	\$0.00	\$0.00	EMITIDA	
#127	F001-00000072	17/05/2026, 03:39 p. m.	Camandor Final	\$10.50	\$1.58	\$12.08	EMITIDA	
#128	F001-00000071	17/05/2026, 03:39 p. m.	Camandor Final	\$3.25	\$0.49	\$3.74	EMITIDA	

**Factura / Ticket**  
F001-00000078

FECHA: 19/05/2026, 03:52 p. m.

CLIENTE: Yasmany Fernandez  
CÉDULA / RUC: 1756779680

ESTADO: EMITIDA

PRODUCTO	CANT.	PRECIO	IVA	TOTAL
Acete El Cocinero TL	1	\$3.25	\$0.49	\$3.74
Subtotal:				\$3.25
IVA (15%):				\$0.49
<b>TOTAL:</b>				<b>\$3.74</b>

## CONTROL Y SUPERVISIÓN

Esta sección agrupa las herramientas de control del negocio: la gestión de la caja, los reportes, la auditoría física del inventario y las notificaciones automáticas.

### 18. Control de Caja

Gestiona las sesiones de caja: la apertura y el cierre con fondo inicial, el cuadro del efectivo y el reporte de diferencias. Es obligatorio abrir la caja antes de realizar ventas.

#### Abrir caja

- Campo para ingresar el fondo inicial (dinero de arranque).
- Campo de observaciones de apertura (opcional).
- El botón "Abrir Caja" registra la sesión con fecha y hora.

#### Sesión activa

- Muestra el fondo inicial ingresado.
- Total de ventas en efectivo en tiempo real.
- Número de transacciones realizadas.
- Fecha y hora de apertura.
- Botón "Cerrar Caja" (solo ADMIN y CAJERO).

#### Cerrar caja

- Resumen de la sesión: fondo inicial + ventas en efectivo = efectivo esperado.
- Campo "Efectivo Contado" para ingresar el dinero físico contado.
- Cálculo automático de la diferencia (sobrante o faltante).
- Campo de observaciones de cierre.
- Botón "Confirmar Cierre".

#### Reporte de cierre e historial

- Pantalla de resumen posterior al cierre con todos los datos de la sesión.
- Diferencia resaltada en verde (sobrante) o rojo (faltante).
- Botones “Imprimir Reporte” y “Ver Historial”.
- Historial con todas las sesiones (propias para el CAJERO; de todos para ADMIN y GERENTE).

## Reglas importantes

- Solo puede haber UNA sesión de caja activa por cajero a la vez.
- El GERENTE puede ver el historial y los detalles, pero NO puede abrir ni cerrar caja.
- Si la caja está cerrada, el módulo de ventas queda bloqueado.

## Proceso de cierre

1. Presione “Cerrar Caja”.
2. Cuente el efectivo físico de la caja.
3. Ingrese el monto contado.
4. Verifique la diferencia calculada.
5. Agregue las observaciones del cierre.
6. Confirme el cierre: el sistema genera el reporte.

### Control de Caja

Gestión de apertura y cierre de caja

**Cerrando Caja — Sesión #14**

#### Resumen del Sistema

Según los registros digitales

Fondo inicial	\$40.00
Ventas efectivo	\$0.00
Ventas tarjeta	\$0.00
Ventas transferencia	\$0.00
<b>Total ventas</b>	<b>\$0.00</b>
<b>Efectivo esperado en caja</b>	<b>\$40.00</b>

Fondo inicial = ventas en efectivo

#### Conteo Físico

Dinero real contado por el cajero

Efectivo contado en caja

\$ 40

Cuenta todos los billetes y monedas físicamente

**DIFERENCIA**

**\$0.00**

Caja cuadrada ✓

Observaciones del cierre (opcional)

Cancelar **Confirmar Cierre**

### Control de Caja

Gestión de apertura y cierre de caja

**Minimarket Otavalo**

Reporte de Cierre de Caja

Sesión #14  
Apertura: 04/06/2026, 03:48 p. m.  
Cierre: 04/06/2026, 03:49 p. m.

#### Apertura

Fondo inicial	\$40.00
Transacciones realizadas	0

#### Ventas por Método

<b>Total Ventas</b>	<b>\$0.00</b>
---------------------	---------------

#### Cuadre de Efectivo

Comparación sistema vs. conteo físico

FONDO INICIAL	+ VENTAS EN EFECTIVO	= EFECTIVO ESPERADO	EFECTIVO CONTADO	DIFERENCIA
\$40.00	\$0.00	\$40.00	\$39.00	\$-1.00 FALTANTE

Imprimir Reporte Nueva Apertura Ver Historial

### Control de Caja

Gestión de apertura y cierre de caja

#### Historial de Caja

[← Volver](#)

#14 CERRADA 04/06/2026, 03:48 p. m. → 04/06/2026, 03:49 p. m. 0 minutos

Fondo: \$40.00 Tx: 0 S Efectivo: \$0.00 Tarjeta: \$0.00 Total: \$0.00 Dif: \$-1.00

#13 CERRADA 04/06/2026, 03:42 p. m. → 04/06/2026, 03:47 p. m. 4 minutos

Fondo: \$30.00 Tx: 0 S Efectivo: \$0.00 Tarjeta: \$0.00 Total: \$0.00 Dif: \$0.00

#12 CERRADA 19/05/2026, 03:50 p. m. → 04/06/2026, 03:42 p. m. 383H 51min

Fondo: \$0.00 Tx: 1 S Efectivo: \$3.74 Tarjeta: \$0.00 Total: \$3.74 Dif: \$0.00

#11 CERRADA 19/05/2026, 03:48 p. m. → 19/05/2026, 03:49 p. m. 0 minutos

Fondo: \$50.00 Tx: 1 S Efectivo: \$3.74 Tarjeta: \$0.00 Total: \$3.74 Dif: \$0.00

## 19. Reportes

Es el módulo de análisis y reportes del negocio. Presenta información de ventas, compras, utilidad y sesiones de caja para un período determinado.

### Filtros del reporte

- Tipo: diario, mensual o personalizado.
- Rango de fechas personalizado.
- Botones de acceso rápido "Hoy" y "Este Mes".
- Botón "Generar Reporte".

### Tarjetas de resumen

- Total de ventas (cantidad y monto).
- Total de compras (cantidad y monto).
- Utilidad bruta (ventas menos compras).

### Detalle de ventas y compras

- Detalle de ventas: folio, fecha, cliente, usuario o rol y total.
- Detalle de compras: fecha, proveedor, usuario o rol y total.

### Historial de sesiones de caja

- Tabla con todos los cierres de caja del sistema.
- Diferencias resaltadas: verde (sobrante), rojo (faltante) y azul (exacto).

### Exportación

- Botón "Exportar PDF" para generar el reporte en PDF.
- Botón "Imprimir" que abre el diálogo de impresión del navegador.

#### Reportes de Gestión

Análisis de ventas y compras

---

**Filtros de Reporte**

Tipo de Reporte
Fecha Desde
Fecha Hasta

Mensual
01/05/2026
30/06/2026

General Reporte
Hoy
Este Mes

Exportar PDF
Imprimir

Total Ventas

66

\$360.37

Total Compras

26

\$1086.41

Utilidad Bruta

\$-726.04

**Detalle de Ventas (66)**

Folio	Fecha	Cliente	Usuario / Rol	Total
F001-000000070	19/05/2026	Yasmaly Fernandez	admin <span style="color: red; font-weight: bold;">CERRADO</span>	\$3.74
F001-000000077	19/05/2026	Consumidor Final	admin <span style="color: red; font-weight: bold;">CERRADO</span>	\$3.74
F001-000000076	18/05/2026	Consumidor Final	cajero2 <span style="color: green; font-weight: bold;">CAJERO</span>	\$3.28

**Historial de Sesiones de Caja**

Apertura, cierre y cuadre de efectivo por cajero

#	Cajero	Estado	Apertura	Cierre	Fondo Inicial	Ventas Efectivo	Total Ventas	Tx	Diferencia
#14	admin	CERRADA	04/06/2026, 03:48 p. m.	04/06/2026, 03:49 p. m.	\$40.00	\$0.00	\$0.00	0	\$-1.00
#13	admin	CERRADA	04/06/2026, 03:42 p. m.	04/06/2026, 03:47 p. m.	\$30.00	\$0.00	\$0.00	0	\$0.00
#12	admin	CERRADA	19/05/2026, 03:50 p. m.	04/06/2026, 03:42 p. m.	\$0.00	\$3.74	\$3.74	1	\$0.00
#11	admin	CERRADA	19/05/2026, 03:48 p. m.	19/05/2026, 03:49 p. m.	\$50.00	\$3.74	\$3.74	1	\$0.00
#10	cajero2	CERRADA	18/05/2026, 06:00 p. m.	18/05/2026, 06:03 p. m.	\$100.00	\$14.49	\$14.49	2	\$0.51
#9	Cajero	CERRADA	18/05/2026, 05:56 p. m.	18/05/2026, 05:56 p. m.	\$0.00	\$3.74	\$3.74	1	\$-2.74
#8	admin	CERRADA	18/05/2026, 05:24 p. m.	18/05/2026, 05:46 p. m.	\$0.00	\$0.69	\$0.69	1	\$0.00
#7	Cajero	CERRADA	17/05/2026, 10:39 p. m.	17/05/2026, 10:41 p. m.	\$300.00	\$22.43	\$22.43	1	\$0.00
#6	admin	CERRADA	17/05/2026, 10:17 p. m.	17/05/2026, 10:28 p. m.	\$50.00	\$11.22	\$11.22	2	\$0.00
#5	admin	CERRADA	10/11/2025, 01:40 p. m.	17/05/2026, 10:15 p. m.	\$40.00	\$0.00	\$0.00	0	\$0.00
#4	admin	CERRADA	10/11/2025, 01:35 p. m.	10/11/2025, 01:40 p. m.	\$40.00	\$0.00	\$40.00	0	---

## 20. Auditoría Física

Permite realizar conteos físicos del inventario y compararlos contra el stock registrado en el sistema. Sirve para detectar diferencias entre lo que indica el sistema y lo que realmente hay en la bodega.

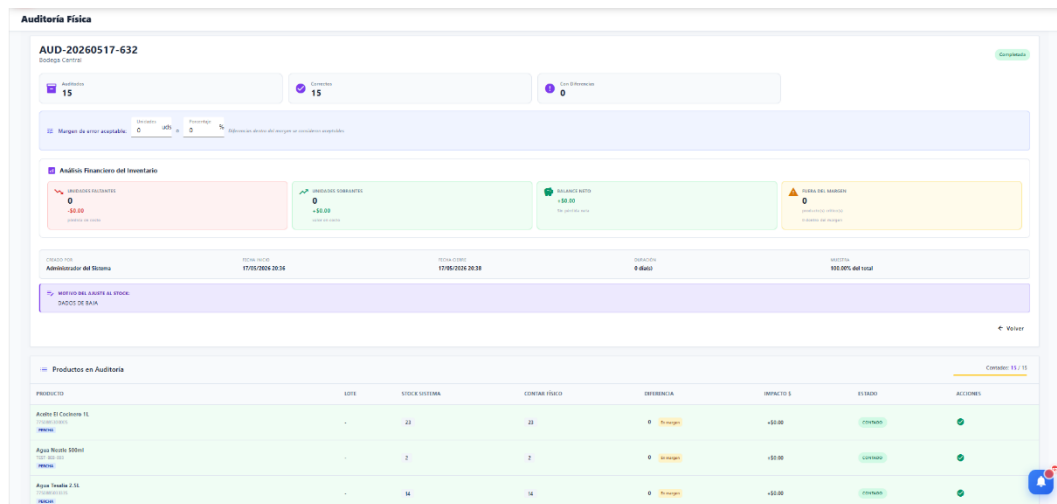
### Funcionalidades principales

- Crear una nueva sesión de auditoría.
- Seleccionar la bodega a auditar.
- Registrar el conteo físico producto por producto.
- El sistema compara el conteo real contra el stock del sistema.
- Genera un reporte de diferencias (sobrantes y faltantes).
- Aprobar o rechazar la auditoría; al aprobarla, el stock se ajusta al conteo real.

### Proceso paso a paso

1. Cree una nueva auditoría seleccionando la bodega.
2. Para cada producto, ingrese la cantidad contada físicamente.
3. El sistema muestra la diferencia respecto al stock registrado.
4. Revise y apruebe el conteo.
5. Al aprobar, el sistema actualiza el inventario con el conteo real.

CÓDIGO	BODEGA	FUENTE	FECHA INICIO	PRODUCTOS	ESTADO	PRECISIÓN	ACCIONES
AUD-20260517-632	Bodega Central	Pecho	17/05/2026 20:36	15 / 15 (100.00%)	Completada	100.00%	
AUD-20260515-863	Bodega Central	Bodega	15/05/2026 12:12	4 / 63 (6.35%)	Completada	50.00%	
AUD-20260514-565	Bodega Central	Bodega	14/05/2026 14:52	4 / 63 (6.35%)	Completada	50.00%	
AUD-20260513-327	Bodega Central	Bodega	13/05/2026 17:08	2 / 62 (3.23%)	Completada	0.00%	
AUD-20260513-802	Bodega Central	Bodega	13/05/2026 17:06	7 / 62 (11.29%)	Completada	85.71%	



## 21. Notificaciones

Es el sistema de alertas automáticas, accesible desde el ícono de campana en la barra superior. Se actualiza cada 30 segundos y avisa sobre situaciones que requieren atención en el inventario.

### Tipos de alertas

- Stock bajo (naranja): producto por debajo del stock mínimo configurado.
- Sin stock en bodega (rojo): producto completamente agotado.
- Percha agotada (naranja): stock en percha en cero, pero hay stock en bodega.
- Por vencer (naranja o rojo): productos con fecha de vencimiento próxima.

### Funcionalidades principales

- Contador de alertas no leídas (distintivo rojo).
- Panel desplegable con la lista de alertas.
- Al hacer clic en una alerta, el sistema redirige al módulo correspondiente.
- Marcar una alerta como leída de forma individual.
- Opciones "Marcar todas como leídas" y "Limpiar todas".

