

REPÚBLICA DEL ECUADOR



**INSTITUTO SUPERIOR
TECNOLÓGICO TENA**
Tecnología, Innovación y Desarrollo



**DESARROLLO DE UNA APLICACIÓN MÓVIL CON BASE DE DATOS
LOCAL PARA LA GESTIÓN DE VENTAS E INVENTARIO DE LA
“TIENDA GÉNESIS”**

Trabajo de Integración Curricular, presentado como requisito parcial para optar por el título de Tecnólogo Superior en Desarrollo de Software.

AUTORA: Sara Pamela Andi Tapuy

TUTOR: Ing. Juan Espín Montesdeoca

Tena - Ecuador

2025-IIS

APROBACIÓN DEL TUTOR


ING. JUAN ESPÍN MONTESDEOCA

PROFESOR DEL INSTITUTO SUPERIOR TECNOLÓGICO TENA.

CERTIFICA:

En calidad de Tutor del Proyecto Integrador denominado: **Desarrollo de una aplicación móvil con base de datos local para la gestión de ventas e inventario de la “Tienda Génesis”**, de autoría de la señora Sara Pamela Andi Tapuy, con CC. **1501185670** estudiante de la Carrera de Desarrollo de Software del Instituto Superior Tecnológico Tena, **CERTIFICO** que se ha realizado la revisión prolija del Trabajo antes citado, cumple con los requisitos de fondo y de forma que exigen los respectivos reglamentos e instituciones.

Tena, 05 de enero de 2026



Ing. Juan M. Espín Montesdeoca, Mg.

TUTOR DEL TIC

CERTIFICACIÓN DEL TRIBUNAL CALIFICADOR

Tena, 23 de enero de 2026

Los Miembros del Tribunal de Grado abajo firmantes, certificamos que el Trabajo de Titulación denominado: **Desarrollo de una aplicación móvil con base de datos local para la gestión de ventas e inventario de la “Tienda Génesis”**, presentado por Sara Pamela Andi Tapuy, con CC: 1501185670, estudiante de la Carrera de Desarrollo de Software del Instituto Superior Tecnológico Tena, ha sido corregida y revisada; por lo que autorizamos su presentación.

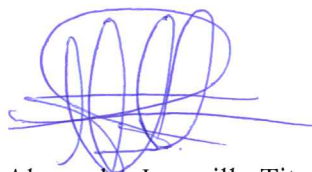
Atentamente;



Ing. Ítalo Marcelo Lara Pilco., MsC.
PRESIDENTE DEL TRIBUNAL



Ing. Agustín Gonzalo Guanipatín Ramírez
MIEMBRO DEL TRIBUNAL



Ing. Betty Alexandra Jaramillo Tituaña., Med.
MIEMBRO DEL TRIBUNAL

AUTORÍA

Yo, **Sara Pamela Andi Tapuy**, con CC: **1501185670**, declaro ser autora del presente Trabajo de Titulación denominado: **Desarrollo de una aplicación móvil con base de datos local para la gestión de ventas e inventario de la “Tienda Génesis”** y absuelvo expresamente al Instituto Superior Tecnológico Tena, y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido de la misma.

Adicionalmente acepto y autorizo al Instituto Superior Tecnológico Tena, la publicación de mi trabajo de Titulación en el repositorio institucional- biblioteca Virtual.

AUTORA:



Sara Pamela Andi Tapuy

CÉDULA: 1501185670


FECHA: Tena, 05 de enero de 2026

CARTA DE AUTORIZACIÓN POR PARTE DEL AUTOR

Yo, **Sara Pamela Andi Tapuy**, declaro ser autora del Trabajo de Titulación titulado: **Desarrollo de una aplicación móvil con base de datos local para la gestión de ventas e inventario de la “Tienda Génesis”**, como requisito para la obtención del Título de: **TECNÓLOGO SUPERIOR EN DESARROLLO DE SOFTWARE**: autorizo al Sistema Bibliotecario del Instituto Superior Tecnológico Tena, para que con fines académicos, muestre al mundo la producción intelectual del Instituto, a través de la visualización de su contenido que constará en el Repositorio Digital Institucional.

Los usuarios pueden consultar el contenido de este trabajo en el RDI, en las redes de información del país y del exterior, con las cuales tenga convenio el Instituto. El Instituto Superior Tecnológico Tena, no se responsabiliza por el plagio o copia del presente trabajo que realice un tercero. Para constancia de esta autorización, en la ciudad de Tena, 05 de enero de 2026, firma la autora.

AUTORA: Sara Pamela Andi Tapuy

FIRMA: 

CÉDULA: 1501185670

DIRECCIÓN: Cotundo_Barrío Laguna Azul

CORREO ELECTRÓNICO: sara.andi@est.itstena.edu.ec

CELULAR: 0994272167

DATOS COMPLEMENTARIOS

TUTOR: Juan M. Espín Montesdeoca, Mg.

TRIBUNAL DEL GRADO:

Ing. Ítalo Marcelo Lara Pilco., MsC. (presidente).

Ing. Agustín Gonzalo Guanipatín Ramírez (Miembro).

Ing. Betty Alexandra Jaramillo Tituaña., Med. (Miembro).

DEDICATORIA

Dedico este proyecto de titulación primeramente a Dios, por guiar cada paso de mi vida y darme la fortaleza necesaria para no rendirme ante las dificultades. A mi esposo Jhonny Huatatoa, por su amor, apoyo incondicional y confianza constante en mí. A mis hijas Génesis Huatatoa y Evelyn Huatatoa, quienes son mi mayor motivación y la razón para seguir superándome cada día, y de manera muy especial a mi hija que hoy se encuentra en el cielo, cuya presencia espiritual ha sido una fuente de fuerza y amor permanente en mi vida.

Dedico también este logro a mis padres, por su esfuerzo, consejos y valores inculcados desde mi niñez, y a todas las personas que de una u otra manera formaron parte de este proceso académico, acompañándome con su apoyo y palabras de aliento.

AGRADECIMIENTO

Primeramente, agradezco a Dios por brindarme la fortaleza, la salud y la perseverancia necesarias para culminar esta importante etapa de mi vida académica. Expreso mi sincero agradecimiento a mi tutor, Ing. Juan Espín, por su guía, acompañamiento y valiosos conocimientos aportados durante el desarrollo de este proyecto de titulación.

De manera muy especial, agradezco a mi esposo Jhonny Huatatocha, quien siempre estuvo a mi lado brindándome su apoyo incondicional, comprensión y motivación constante. A mis hijas, Génesis Huatatocha y Evelyn Huatatocha, quienes han sido mi mayor inspiración para seguir adelante y no rendirme ante las dificultades.

Agradezco con profundo cariño a mis padres, Virginia Tapuy y Pedro Andi, por su amor, consejos y valores inculcados desde mi niñez, los cuales han sido fundamentales en mi formación personal y profesional. Asimismo, agradezco a mis hermanos, especialmente a mi hermana Marina Andi, por motivarme a continuar con mis estudios y creer siempre en mis capacidades.

Reconozco también mi propio esfuerzo, constancia y dedicación, ya que este logro es resultado del compromiso asumido durante todo mi proceso académico. Finalmente, expreso mi agradecimiento a mis compañeros por el compañerismo y apoyo brindado, así como a todos mis profesores y al Instituto Superior Tecnológico Tena, por la formación académica y profesional recibida a lo largo de mi carrera.

ÍNDICE DE CONTENIDO

PORTADA.....	i
APROBACIÓN DEL TUTOR.....	ii
CERTIFICACIÓN DEL TRIBUNAL CALIFICADOR	iii
AUTORÍA.....	iv
CARTA DE AUTORIZACIÓN POR PARTE DEL AUTOR.....	v
DEDICATORIA	vi
AGRADECIMIENTO.....	vii
ÍNDICE DE CONTENIDO.....	viii
ÍNDICE DE TABLAS	xii
ÍNDICE DE ILUSTRACIONES.....	xiii
1. TEMA	15
2. RESUMEN.....	16
ABSTRACT.....	17
3. FUNDAMENTACIÓN DEL TEMA.....	18
3.1. Necesidad.....	18
3.2. Actualidad	18
3.3. Importancia	19
3.4. Presentación del problema profesional a responder.....	20
3.5. Delimitación.....	20
3.5.1. Delimitación Espacial	20
3.5.2. Delimitación Temporal	20
3.5.3. Delimitación técnica.....	21
3.5.4. Unidades de Observación.....	21

3.6. Beneficiarios	21
3.6.1. Directos	21
3.6.2. Indirectos.....	22
4. OBJETIVOS	23
4.1. Objetivo General	23
4.2. Objetivos Específicos.....	23
5. ASIGNATURAS INTEGRADORAS	24
6. FUNDAMENTACIÓN TEÓRICA.....	25
6.1. Transformación digital de las microempresas.....	25
6.2. Adopción de Tecnologías Móviles (TAM/UTAUT)	25
6.3. Sistemas de información para control de ventas e inventario	26
6.4. Gestión de ventas y registro digital.....	26
6.5. Gestión de inventarios, categorías y alertas (stock y caducidad).....	26
6.6. Operación sin conexión y persistencia local (enfoque “offline”)	27
6.7. Flutter	27
6.8. Visual Studio Code (VS Code)	27
6.9. Arquitectura de software en aplicaciones móviles.....	28
6.10. Arquitectura MVVM.....	28
6.11. SQLite	28
6.12. Gestión de inventario automatizada	28
6.13. Calidad del software y evaluación funcional (ISO/IEC 25010).....	29
6.14. Usabilidad y experiencia de usuario (ISO 9241)	29
6.15. Seguridad y privacidad de datos	29
6.16. Resiliencia tecnológica.....	30
6.17. Scrum	30

7. MARCO LEGAL	32
Constitución de la República del Ecuador (2008).....	32
Ley Orgánica de Protección de Datos Personales (LOPDP)	32
Ley Orgánica de Emprendimiento e Innovación (2020).....	33
Ley Orgánica de Telecomunicaciones (2015).....	33
Código Orgánico de la Producción, Comercio e Inversiones (COPCI, 2010).....	33
Ley de Propiedad Intelectual y Código Ingenios (2016)	33
Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos (2002).	34
8. METODOLOGÍA	35
8.1. Materiales.....	35
8.2. Ubicación del Área de Estudio.....	36
8.3. Tipo de Investigación / Estudio.....	37
8.4. Enfoque de la investigación	37
8.5. Metodología para cada objetivo	38
8.5.1. Primer objetivo: Análisis de los procesos actuales de registro de ventas e inventario.....	38
8.5.2. Segundo Objetivo: Diseño de la arquitectura del sistema y la base de datos	38
8.5.3. Tercer Objetivo: Programación de la aplicación móvil	39
8.5.4. Cuarto Objetivo: Validación del funcionamiento de sistema	40
9. RESULTADOS.....	42
9.1. Primer objetivo: Análisis de los procesos actuales de registro de ventas e inventario.....	42
9.1.1. Resultados del levantamiento de información mediante entrevistas.....	42
9.1.2. Resultados de la observación directa del proceso operativo	43
9.1.3. Problemas detectados en los procesos actuales.....	44

9.1.4. Requerimientos funcionales definidos a partir del análisis.....	45
9.1.5. Requerimientos no funcionales definidos	46
9.1.6. Elaboración de historias de usuario y Product Backlog.....	46
9.2. Segundo Objetivo: Diseñar la arquitectura del sistema y la base de datos	47
9.3. Tercer Objetivo: Programación de la aplicación móvil de ventas e inventario	53
9.3.1. Desarrollo de módulos implementados	54
9.3.2. Integración de base de datos local (SQLite)	55
9.3.3. Resultados visuales del sistema implementado.....	56
9.4. Cuarto Objetivo: Validación del funcionamiento del sistema	61
9.4.1. Resultados de la matriz de casos de prueba pruebas funcionales por módulo	61
9.4.2. Resultados de pruebas de integración	63
9.4.3. Resultados de validación de aceptación y usabilidad (usuario final).....	65
9.4.4. Incidencias y ajustes aplicados.....	65
9.4.5. Resultados cuantitativos y cualitativos	66
9.4.6. Validación general del sistema.....	68
9.4.7. Evaluación de la calidad del sistema según ISO/IEC 25010	68
10. CONCLUSIONES	71
11. RECOMENDACIONES	72
12. BIBLIOGRAFÍA.....	73
13. ANEXOS.....	77
13.1. Entrevistas	77
13.2. Fotografías.....	79
13.3. Código	82
13.3.1. Código del Módulo de Ventas en Flutter (Dart)	82

ÍNDICE DE TABLAS

Tabla 1. Asignaturas integradoras.....	24
Tabla 2. Resultados del Levantamiento de Información Mediante Entrevistas....	43
Tabla 3. Resultados de la Observación Directa del Proceso Operativo	44
Tabla 4. Problemas Detectados en los Procesos Actuales	44
Tabla 5. Requerimientos Funcionales Definidos a Partir del Análisis.....	45
Tabla 6. Requerimientos No Funcionales Obtenidos.....	46
Tabla 7. Product Backlog (Historias de Usuario)	47
Tabla 8. Plan de Sprints y Entregables.....	54
Tabla 9. Módulos Implementados y Funcionalidad Alcanzada	55
Tabla 10. Resultados de Pruebas Funcionales por Módulo	61
Tabla 11. Resultados de Pruebas de Integración.....	63
Tabla 12. Validación de Aceptación y Usabilidad.....	65
Tabla 13. Incidencias y Correcciones Aplicadas	66
Tabla 14. Comparación de Tiempos (Registro Manual vs Aplicación Móvil).....	67
Tabla 15. Evaluación del Sistema Según Normas ISO/IEC 25010	70

ÍNDICE DE ILUSTRACIONES

Ilustración 1. Ubicación Geográfica	36
Ilustración 2. Diagrama de Casos de Uso del Sistema de Gestión de Ventas e Inventario.	48
Ilustración 3. Diagrama Entidad - Relación del Sistema de Gestión de Ventas e Inventario.	49
Ilustración 4. Arquitectura del Proyecto Basada en el Patrón MVVM.	49
Ilustración 5. Prototipo de Pantalla de Inicio de Sesión del Sistema.	50
Ilustración 6. Prototipo de Menú Principal del Rol Administrador.	50
Ilustración 7. Prototipo de Menú Principal del Rol Empleado.	51
Ilustración 8. Prototipo del Módulo de Registro de Ventas.	51
Ilustración 9. Prototipo del Módulo de Control de Inventario.	52
Ilustración 10. Prototipo del Módulo de Reportes del Sistema.	52
Ilustración 11. Diagrama de Clases del Sistema de Gestión de Ventas e Inventario.	53
Ilustración 12. Pantalla de Inicio de Sesión del Sistema Móvil.	56
Ilustración 13. Menú Principal del Rol Administrador.	57
Ilustración 14. Gestión de clientes del sistema.	57
Ilustración 15. Gestión de Productos del Sistema.	58
Ilustración 16. Registro de Ventas del Sistema.	58
Ilustración 17. Control de Inventario del Sistema.	59
Ilustración 18. Módulo de Reportes del Sistema.	59
Ilustración 19. Reporte de Ventas Filtrado por Rango de Fechas.	60
Ilustración 20. Reporte de Ventas Generado en Formato PDF.	60
Ilustración 21. Registro de Ventas Durante las Pruebas Funcionales.	63

Ilustración 22. Comprobante de Venta Generado en Formato PDF Durante la Validación.	64
Ilustración 23. Características del Dispositivo Utilizado en las Pruebas.	67
Ilustración 24. Socialización de Actividades.	79
Ilustración 25. Entrevista al Propietario.	79
Ilustración 26. Revisión del Avance del Proyecto	80
Ilustración 27. Elaboración del Proyecto.	80
Ilustración 28. Vista del IDE (Visual Studio Code): Estructura del Proyecto y Archivo main.dart (Flutter)	81
Ilustración 29. Vista del IDE (Visual Studio Code): Codificación del Módulo de Ventas en Flutter	81

1. TEMA

DESARROLLO DE UNA APLICACIÓN MÓVIL CON BASE DE DATOS LOCAL PARA LA GESTIÓN DE VENTAS E INVENTARIO DE LA “TIENDA GÉNESIS”

2. RESUMEN

En la Tienda Génesis, ubicada en la comunidad Los Andes (Cotundo, Napo), el registro manual de ventas e inventario en cuadernos generaba pérdida de información, errores de cálculo, descontrol de stock y demora en la atención al cliente. Por ello, el objetivo del proyecto fue desarrollar una aplicación móvil offline que automatice la gestión de ventas e inventario y permita disponer de información confiable para la toma de decisiones. La metodología fue de enfoque mixto: se levantó información mediante entrevistas al propietario y colaboradores, y observación directa del proceso operativo; luego se diseñó la solución con base de datos SQLite, arquitectura MVVM y prototipos en Figma; el desarrollo se ejecutó con la metodología ágil Scrum en sprints usando Flutter (Dart). Finalmente, se validó el funcionamiento mediante casos de prueba por módulo, pruebas de integración y validación de aceptación/usabilidad con el usuario final. Los resultados evidenciaron que el sistema registra y almacena datos correctamente en SQLite, genera comprobantes y reportes en PDF, mantiene coherencia entre ventas, inventario y reportes, y mejora los tiempos de atención, reduciendo errores asociados al cálculo manual.

Palabras clave: Flutter, SQLite, MVVM, ventas e inventario, aplicación móvil offline.

ABSTRACT

In Tienda Génesis, located in the Los Andes community (Cotundo, Napo), the manual recording of sales and inventory in notebooks led to information loss, calculation errors, poor stock control, and delays in customer service. Therefore, the objective of this project was to develop an offline mobile application to automate sales and inventory management and provide reliable information for decision-making. A mixed-method approach was applied: information was gathered through interviews with the owner and staff, as well as direct observation of the operational process. The solution was then designed using a SQLite database, MVVM architecture, and interface prototypes in Figma; development was carried out using the Scrum agile methodology in sprints with Flutter (Dart). Finally, the system was validated through module-based test cases, integration testing, and user acceptance/usability validation with the end user. The results showed that the system correctly records and stores data in SQLite, generates invoices and PDF reports, maintains consistency between sales, inventory, and reports, and improves service times by reducing errors associated with manual calculations.

Keywords: Flutter, MVVM, offline mobile application, sales and inventory, SQLite.

Reviewed by:


.....

B.A. Carolina Romero M.Ed.

Professor Language Center

3. FUNDAMENTACIÓN DEL TEMA

3.1. Necesidad

La gestión de ventas e inventario en la Tienda Génesis representó uno de los principales problemas en el funcionamiento diario del negocio. Al tratarse de un establecimiento comercial de atención constante, era necesario registrar de manera simultánea múltiples aspectos como la venta de productos, el control del stock disponible, el registro de clientes, el manejo de gastos y la verificación de productos próximos a caducar. Todo este proceso se realizaba de forma manual, utilizando cuadernos y anotaciones informales, lo que generaba errores frecuentes y pérdida de información.

Realizar el control manual de ventas e inventario consumía una cantidad considerable de tiempo y, aun así, no garantizaba exactitud en los datos. Con frecuencia surgían inconsistencias entre el stock real y el stock registrado, dificultades para identificar productos más vendidos o faltantes, y problemas para conocer con claridad los ingresos diarios. Además, cualquier corrección implicaba revisar nuevamente registros anteriores, provocando retrasos y confusión en la información.

Ante esta situación, se identificó la necesidad de contar con una herramienta tecnológica que permitiera organizar y automatizar los procesos de ventas e inventario, reduciendo errores, optimizando el tiempo de atención y facilitando el control del negocio. No se trató únicamente de ahorrar tiempo, sino de mejorar la organización general, ya que una adecuada gestión desde el registro inicial permite que las demás actividades del negocio se desarrollen de manera más ordenada y eficiente.

3.2. Actualidad

En el momento de realizar el estudio, la Tienda Génesis manejaba sus procesos comerciales de forma completamente manual, sin el apoyo de una aplicación móvil o sistema digital especializado. Aunque en algunos casos se utilizaban herramientas básicas como calculadoras o registros aislados, no existía

una plataforma centralizada que integrara las ventas, el inventario, los clientes y los reportes del negocio.

Esta forma de trabajo ocasionaba un uso ineficiente de los recursos disponibles, ya que gran parte del tiempo del propietario y de los empleados se destinaba a registrar información, verificar precios o revisar existencias, en lugar de enfocarse en mejorar la atención al cliente o la planificación del negocio. Asimismo, la falta de información actualizada dificultaba la toma de decisiones oportunas, como la reposición de productos o el control de gastos.

Con el desarrollo de la aplicación móvil propuesta, se implementó una solución tecnológica acorde a las necesidades actuales del comercio local, integrando en un solo sistema los procesos de ventas, inventario, clientes y reportes. Esta solución respondió a las tendencias actuales del desarrollo de software, priorizando el uso de dispositivos móviles, la facilidad de uso y la posibilidad de operar sin conexión a internet.

3.3. Importancia

La importancia de este proyecto estuvo en el alivio que representó para la gestión diaria de la Tienda Génesis. En lugar de depender de registros manuales y revisiones constantes, el control de ventas e inventario se realizó en menos tiempo del que se empleaba anteriormente, permitiendo que el propietario y los empleados dedicaran su atención a actividades donde su experiencia y trato con los clientes marcaron una verdadera diferencia.

En cuanto a la organización del negocio, se logró un mayor orden en la información. Las ventas se registraron de manera más clara, el inventario se mantuvo actualizado y, cuando surgieron imprevistos como faltantes de productos o ajustes en precios, los cambios pudieron realizarse en minutos y no después de largas revisiones de cuadernos o anotaciones previas.

Los empleados también se vieron beneficiados, ya que dejaron de realizar registros repetitivos y confusos. Contaron con una herramienta sencilla que les

permitió ingresar información de manera directa, consultar datos cuando lo necesitaron y reducir errores en el manejo de productos y ventas. Esto contribuyó a que el trabajo diario se desarrollara de forma más ordenada y con menos presión.

Aunque los clientes no interactuaron directamente con el sistema, el cambio fue perceptible. La atención fue más ágil, hubo menos confusiones al momento de cobrar, y se evitó la falta de productos por una mejor planificación del inventario. Todo esto permitió que la experiencia de compra fuera más fluida y organizada, lo cual se reflejó en una mejor percepción del servicio ofrecido por la Tienda Génesis.

3.4. Presentación del problema profesional a responder

La Tienda Génesis carece de un sistema digital para el control de ventas e inventario, lo que ha ocasionado errores en el registro de información, desactualización del stock y dificultad para conocer las ganancias reales del negocio. El uso de métodos manuales ha limitado la organización de los datos y ha afectado la toma de decisiones del negocio.

Línea de Investigación: Tecnologías de la Información.

Sub-Línea de Investigación: Desarrollo de Software.

Área: Informática.

Aspecto: Desarrollo Móvil.

Sector: Programación Móvil.

3.5. Delimitación

3.5.1. Delimitación Espacial

El Trabajo de Integración Curricular se realizó en la Tienda Génesis, establecimiento comercial dedicado a la venta de víveres, abarrotos y productos de consumo diario, ubicado en la comunidad Los Andes, parroquia Cotundo, cantón Archidona, provincia de Napo, Ecuador.

3.5.2. Delimitación Temporal

El proyecto se ejecutó en el Periodo Académico 2025-IIS.

3.5.3. Delimitación técnica

La aplicación se desarrolló con Flutter y el lenguaje Dart, usando el entorno Visual Studio Code. Se utilizó SQLite como base de datos local, permitiendo operar sin conexión a internet, lo cual es clave para zonas con conectividad limitada.

El sistema contó con los siguientes módulos:

- Autenticación y gestión de usuarios por roles.
- Gestión de productos e inventarios.
- Registro de ventas y generación de comprobantes en PDF.
- Generación de reportes automáticos.
- Interfaz gráfica adaptable al usuario final.
- Almacenamiento y procesamiento de datos en local.

Se aplicó la metodología ágil Scrum, dividiendo el trabajo en sprints que permitan validar cada módulo progresivamente. Esta delimitación técnica garantiza un producto funcional y adaptado al entorno rural.

3.5.4. Unidades de Observación

Las unidades de observación que se contemplan para este trabajo están enfocadas directamente en el negocio “Tienda Génesis”.

3.6. Beneficiarios

3.6.1. Directos

El beneficiario directo del trabajo Integración Curricular es:

- El propietario de la Tienda Génesis, quien contará con una herramienta tecnológica que facilitará el control de productos, ventas y balances financieros en tiempo real.

3.6.2. Indirectos

- Los clientes del establecimiento.
- La comunidad local, que se beneficia del fortalecimiento económico del negocio.

4. OBJETIVOS

4.1. Objetivo General

Desarrollar una aplicación móvil con base de datos local que permita gestionar de manera eficiente las ventas e inventario de la Tienda Génesis, optimizando el control de productos y la administración de la información.

4.2. Objetivos Específicos

- Analizar los procesos actuales de registro de ventas e inventario de la Tienda Génesis, identificando las necesidades funcionales del sistema.
- Diseñar la arquitectura del sistema y la base de datos de la Tienda Génesis, definiendo la estructura y relaciones necesarias para el registro y control de la información.
- Programar la aplicación móvil de ventas e inventario, para mejorar el control y reducir errores en los registros.
- Validar el funcionamiento del sistema desarrollado en la Tienda Génesis, mediante pruebas locales que permitan comprobar su usabilidad y correcto registro de la información.

5. ASIGNATURAS INTEGRADORAS

Tabla 1. *Asignaturas integradoras*

ASIGNATURAS INTEGRADORAS	
Asignaturas	Resultados de Aprendizaje
Programación Móvil	Aplica conceptos, técnicas, herramientas de programación, que contribuyan con la implementación de soluciones de software.
Base de Datos Avanzada	Brinda asistencia técnica en el diseño, modelamiento e ilustración del proceso de base de datos.
Análisis y Diseño de Sistemas	Brinda asistencia técnica en el desarrollo de aplicaciones de software, desde el análisis del problema y la planificación del proyecto, hasta la implementación, el mantenimiento, la prueba y la documentación.

Nota: Relación de asignaturas integradoras y resultados de aprendizaje vinculados al TIC. Fuente:

Elaboración propia (2025)

6. FUNDAMENTACIÓN TEÓRICA

6.1. Transformación digital de las microempresas

La transformación digital en las micro, pequeñas y medianas empresas se entiende como un proceso de incorporación y uso de tecnologías digitales para mejorar la productividad, optimizar procesos internos y fortalecer la competitividad. En América Latina, este proceso suele enfrentar barreras relacionadas con conectividad, recursos y capacidades digitales, por lo que las soluciones tecnológicas deben adaptarse al contexto real de operación de cada negocio. En este sentido, la digitalización no se limita a “usar tecnología”, sino a mejorar la forma de registrar, almacenar y consultar información para apoyar decisiones más rápidas y confiables (Dini et al., 2021).

Cuando los registros de ventas e inventario se llevan de forma manual, es frecuente que existan demoras en la consulta, inconsistencias en los apuntes y baja confiabilidad de la información. En experiencias aplicadas en microempresas, se evidencia que el uso de cuadernos y el manejo manual incrementan errores y retrasan la atención, afectando el control administrativo (Abril & Javier, 2022).

6.2. Adopción de Tecnologías Móviles (TAM/UTAUT)

La adopción de tecnologías móviles en microempresas puede analizarse mediante modelos de aceptación como TAM y UTAUT, porque permiten identificar factores que influyen en el uso continuo o abandono de una herramienta digital. En contextos reales (ventas, inventario, atención), la adopción depende de beneficios concretos, facilidad de uso e integración con la rutina diaria del negocio, (Palos-Sánchez et al., 2019)

En TAM, dos variables resultan especialmente relevantes: utilidad percibida (si la herramienta mejora resultados) y facilidad de uso percibida (si aprender y operar el sistema no requiere esfuerzo excesivo). Este enfoque es pertinente en microempresas porque el usuario suele trabajar con poco tiempo, múltiples tareas y necesita pantallas simples y registros rápidos. En el caso de UTAUT, la medición

adecuada de aceptación en población hispanohablante se fortalece cuando se usan instrumentos validados y consistentes (Michel Madera et al., 2012).

6.3. Sistemas de información para control de ventas e inventario

Un sistema de información orientado a ventas e inventario permite registrar transacciones, administrar catálogos de productos, controlar existencias y generar reportes. En microempresas, el inventario es crítico porque influye en costos, disponibilidad, reducción de pérdidas y continuidad operativa (Vásconez et al., 2020). Asimismo, una gestión organizada del inventario contribuye a disminuir costos por administración desordenada y a mejorar reposición y planificación (Mera Cantos & Zambrano Intriago, 2024).

Integrar ventas e inventario en un sistema digital reduce la dependencia de registros dispersos y mejora consistencia: lo vendido se refleja en existencias y en reportes, lo que fortalece el control y la toma de decisiones.

6.4. Gestión de ventas y registro digital

La gestión de ventas implica registrar productos, cantidades, precios, totales y fechas, además de habilitar consultas por periodos para identificar ingresos, rotación de productos y tendencias. En microempresas, el registro digital reduce errores de cálculo y omisiones frecuentes en anotaciones manuales, especialmente cuando el volumen de ventas dificulta mantener consistencia (Palma Muñoz et al., 2021).

Además, la generación de reportes (por rangos de fechas y exportación a PDF) convierte registros operativos en información útil para control y respaldo administrativo (Abril & Javier, 2022).

6.5. Gestión de inventarios, categorías y alertas (stock y caducidad)

El control de inventarios puede apoyarse en modelos como mínimos y máximos, útiles para prevenir quiebres de stock y acumulaciones innecesarias; estos modelos facilitan reposición mediante parámetros de control (Peñate, 2022). En sistemas digitales, la clasificación por categorías mejora organización del catálogo

y consulta rápida. Adicionalmente, las alertas por stock bajo o proximidad de caducidad actúan como mecanismo preventivo para reducir pérdidas y mantener continuidad operativa (Vásconez et al., 2020).

6.6. Operación sin conexión y persistencia local (enfoque “offline”)

En entornos con conectividad limitada o intermitente, la persistencia local permite operar sin depender de internet. Android recomienda el uso de bases de datos para datos estructurados y ofrece guías para trabajar con SQLite en el almacenamiento local. Esto habilita que una aplicación registre ventas y actualice inventario aun sin red, manteniendo continuidad operativa (Android Developers, 2025)

Desde un enfoque de arquitectura, Android define “offline-first” como la capacidad de ejecutar todas o una parte crítica de las funciones del sistema sin acceso a internet, priorizando el funcionamiento del negocio incluso ante fallas de red (Android Developers, 2025).

6.7. Flutter

Flutter es un framework de código abierto para desarrollo multiplataforma que permite construir aplicaciones desde un único código base, optimizando tiempos y recursos. En el ámbito de persistencia local, la documentación oficial presenta el uso de SQLite como alternativa para almacenar y recuperar datos estructurados dentro de la aplicación (Flutter, 2025)

6.8. Visual Studio Code (VS Code)

Visual Studio Code es un editor de código multiplataforma ampliamente utilizado por su ligereza, rapidez y compatibilidad con múltiples lenguajes y frameworks. En el desarrollo móvil, facilita la integración de herramientas como Flutter, Dart, Git, depuración y extensiones, mejorando productividad y organización del proyecto (García & Salazar, 2023).

6.9. Arquitectura de software en aplicaciones móviles

La arquitectura de una aplicación móvil influye en mantenibilidad, escalabilidad y facilidad de pruebas. Android establece que una arquitectura bien definida permite crear aplicaciones escalables y mantenibles que se adaptan a múltiples dispositivos y escenarios de uso (Android Developers, 2026).

Cuando el sistema incluye módulos como usuarios, ventas, inventario y reportes, una arquitectura clara evita dependencias rígidas y reduce riesgos de inconsistencias entre operaciones y resultados reportados.

6.10. Arquitectura MVVM

El patrón MVVM separa responsabilidades entre Modelo (datos/reglas), Vista (interfaz) y ViewModel (lógica de presentación), facilitando mantenimiento y evolución del software. Esta separación favorece pruebas, reduce errores de integración y mejora control del estado de la interfaz, resultando adecuado en aplicaciones móviles con múltiples módulos (Sánchez & Morales, 2020).

6.11. SQLite

SQLite es un sistema de base de datos relacional embebido que no requiere servidor externo y resulta apropiado para almacenamiento local en aplicaciones móviles. Android incluye guías para su uso con datos estructurados, y el enfoque de persistencia local permite continuidad operativa en escenarios sin conexión (Android Developers, 2025).

6.12. Gestión de inventario automatizada

La automatización del inventario optimiza control de productos, reduce errores de registro y mejora planificación operativa. Implementar un sistema digital contribuye a disminuir pérdidas económicas y fortalece decisiones basadas en información confiable y actualizada (Martínez, J., 2024).

6.13. Calidad del software y evaluación funcional (ISO/IEC 25010)

La evaluación de calidad del software se utiliza para verificar si un sistema cumple lo requerido y si es apropiado para su uso. Estudios recientes aplican criterios basados en ISO/IEC 25010 para establecer indicadores de evaluación de calidad, considerando dimensiones como adecuación funcional, usabilidad y eficiencia, (Cortés Rojas, 2024). En aplicaciones de ventas e inventario, estos criterios son relevantes porque el sistema debe registrar correctamente información, mantener consistencia y responder con tiempos adecuados durante la operación diaria.

Asimismo, la arquitectura y las pruebas apoyan la calidad al validar que la información registrada se refleje coherentemente en reportes y que el inventario se actualice correctamente con cada operación, evitando inconsistencias que afecten la toma de decisiones. Por ello, la validación mediante pruebas funcionales, integración y aceptación/usabilidad se justifica para confirmar desempeño en el entorno real.

6.14. Usabilidad y experiencia de usuario (ISO 9241)

La usabilidad y la experiencia de usuario son factores determinantes para el éxito de una aplicación móvil. La norma ISO 9241-210:2019 establece que un sistema usable debe ser efectivo, eficiente y satisfactorio para el usuario. En aplicaciones dirigidas a microempresas, una interfaz clara y sencilla resulta fundamental para garantizar la adopción tecnológica.

(López & Rodríguez, 2023), indican que el uso de botones grandes, colores contrastantes y flujos de navegación simples incrementa la aceptación de aplicaciones móviles en usuarios con poca experiencia tecnológica.

6.15. Seguridad y privacidad de datos

La seguridad de los datos busca proteger la información frente a accesos no autorizados, alteración o pérdida, mientras que la privacidad se enfoca en que el tratamiento de datos personales se realice con medidas preventivas y control sobre

su uso. Para reducir riesgos en sistemas de información, se recomienda aplicar protección de datos desde el diseño, incorporando controles desde las primeras fases del desarrollo (minimización de datos, control de accesos y medidas técnicas acordes al riesgo) y evitando que la privacidad sea una “capa” añadida al final (Agencia Española de Protección de Datos, 2019).

Asimismo, la protección de datos por defecto exige configurar el sistema para que, de manera predeterminada, solo se traten los datos necesarios, se limite la accesibilidad y se definan periodos de conservación adecuados (Agencia Española de Protección de Datos, 2020)

En el contexto ecuatoriano, la guía técnica de la Superintendencia de Protección de Datos Personales enfatiza que estas obligaciones aplican durante la concepción y diseño de proyectos que tratan datos personales y deben basarse en **gestión de riesgos** y medidas proporcionales (Superintendencia de Protección de Datos Personales, 2025).

6.16. Resiliencia tecnológica

La resiliencia tecnológica se refiere a la capacidad de un sistema para mantener su funcionamiento frente a fallas de red, energía u otros eventos adversos. En aplicaciones móviles, esta resiliencia se logra mediante el uso de bases de datos locales, manejo de errores y respaldo de información.

(Khurana et al., 2022), destacan que los sistemas diseñados para entornos de baja conectividad deben priorizar la funcionalidad offline para garantizar la continuidad operativa de los negocios.

6.17. Scrum

Scrum es un marco de trabajo ágil utilizado para el desarrollo de software de manera iterativa e incremental, organizando el trabajo en ciclos cortos denominados sprints. Este enfoque permite entregar versiones funcionales del producto de forma continua, facilitando la adaptación a cambios y la mejora progresiva del sistema.

De acuerdo con (Schwaber & Sutherland, 2020), Scrum mejora la colaboración del equipo y la calidad del producto final mediante la retroalimentación constante del usuario. Asimismo, (Verwijs & Russo, 2023), señalan que la aplicación de Scrum incrementa la productividad y reduce riesgos durante el desarrollo de proyectos de software.

7. MARCO LEGAL

El desarrollo de una aplicación móvil orientada a la gestión de ventas e inventario de la Tienda Génesis se fundamenta en el cumplimiento de las leyes ecuatorianas y estándares internacionales que regulan la tecnología, la protección de datos, la seguridad de la información y la innovación empresarial. Estas normas garantizan que el sistema desarrollado sea legal, ético y técnicamente confiable. A continuación, se detallan las disposiciones más relevantes aplicables al presente proyecto:

Constitución de la República del Ecuador (2008)

La Constitución reconoce en su Artículo 16 el derecho de todas las personas a acceder a las tecnologías de la información y comunicación, y en el Artículo 66, numeral 19, el derecho a la protección de datos personales (Constitución de la República del Ecuador, 2008). Por ello, la aplicación móvil desarrollada garantiza el uso ético y responsable de la información, resguardando la privacidad y los datos de clientes y usuarios de la Tienda Génesis mediante mecanismos de cifrado local y control de acceso por roles. Este enfoque se alinea con lo establecido en la (Ley Orgánica de Protección de Datos Personales, 2021), que en su Artículo 8 exige implementar medidas técnicas y organizativas para proteger los datos tratados en sistemas automatizados.

Ley Orgánica de Protección de Datos Personales (LOPDP)

La Ley Orgánica de Protección de Datos Personales regula el tratamiento automatizado de datos personales en el Ecuador y, en su Artículo 8, establece que el responsable del tratamiento debe aplicar medidas técnicas y organizativas adecuadas para proteger la información (Ley Orgánica de Protección de Datos Personales, 2021). En concordancia con esta normativa, el sistema utilizará mecanismos de cifrado, contraseñas seguras y control de acceso por roles para proteger la información almacenada en la base de datos local (SQLite).

Ley Orgánica de Emprendimiento e Innovación (2020)

Fomenta la creación y fortalecimiento de emprendimientos tecnológicos. En su Artículo 5, promueve el desarrollo de soluciones digitales que mejoren la competitividad de las micro, pequeñas y medianas empresas (MIPYMES).

La aplicación de la Tienda Génesis responde a los principios de innovación y digitalización empresarial que la ley impulsa, fortaleciendo el desarrollo económico local en la provincia de Napo (Ley Orgánica de Emprendimiento e Innovación, 2020).

Ley Orgánica de Telecomunicaciones (2015)

En su Artículo 3, esta ley promueve el acceso equitativo y la inclusión digital de la ciudadanía. La aplicación móvil contribuye al cumplimiento de esta ley al promover el uso responsable de las tecnologías en el ámbito rural, fomentando la inclusión digital de los pequeños comerciantes (Ley Orgánica de Telecomunicaciones, 2015).

Código Orgánico de la Producción, Comercio e Inversiones (COPCI, 2010)

El Título II de esta norma impulsa la innovación, la eficiencia productiva y el fortalecimiento de la competitividad empresarial mediante la adopción tecnológica.

El proyecto contribuye al cumplimiento de esta normativa al incorporar tecnologías digitales que mejoran la productividad y reducen los errores administrativos en la gestión comercial (Código Orgánico de la Producción, Comercio e Inversiones (COPCI), 2010).

Ley de Propiedad Intelectual y Código Ingenios (2016)

En el Artículo 114 se establece que los programas informáticos (software) son obras protegidas por derechos de autor.

El software desarrollado para la Tienda Génesis constituye propiedad intelectual de la autora Sara Pamela Andi Tapuy, y está protegido bajo la legislación ecuatoriana vigente en materia de propiedad intelectual (Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación, 2016).

**Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos
(2002)**

Reconoce la validez jurídica de los mensajes de datos, documentos y firmas electrónicas. El Artículo 6 garantiza la autenticidad e integridad de la información digital.

Los comprobantes o reportes de ventas generados por la aplicación pueden tener respaldo digital con validez legal, fortaleciendo la transparencia y trazabilidad del sistema (Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos, 2002).

8. METODOLOGÍA

8.1. Materiales

a) Recursos tecnológicos

❖ Hardware:

- Computadora portátil con procesador Intel Core i7, 16 GB de RAM y 512 GB SSD.
- Dispositivo móvil Android (versión 10 o superior) para pruebas de campo.
- Router Wi-Fi local para pruebas de conexión.

❖ Software:

- Sistema operativo: Windows 11 Pro.
- Entorno de desarrollo: Visual Studio Code.
- Framework y lenguaje: Flutter (Dart).
- Gestor de base de datos local: SQLite (versión 3.39.2).
- Control de versiones: Git y GitHub.
- Herramientas de diseño: Figma (interfaz de usuario).
- Documentación: Microsoft Word, Excel, Google Workspace.

b) Recursos humanos

- Autora: Sara Pamela Andi Tapuy, estudiante de Tecnología Superior en Desarrollo de Software.
- Tutor académico: Ing. Juan Espín Montesdeoca, Mg.
- Colaboradores: Propietario y personal de la Tienda Génesis, quienes participaron en la recolección de datos, validación de requerimientos y pruebas de la aplicación.

c) Recursos teóricos y normativos

- Documentación técnica de Flutter y SQLite.
- Manuales de implementación de arquitecturas MVVM.

8.3. Tipo de Investigación / Estudio

Este trabajo corresponde a una investigación de tipo aplicada, descriptiva y tecnológica, desarrollada con un enfoque mixto que integra métodos cualitativos y cuantitativos. La investigación se orienta a la solución de un problema real identificado en la Tienda Génesis, relacionado con la gestión manual de ventas e inventario, el cual generaba errores en el control de la información y dificultades en la toma de decisiones.

- Aplicada, porque se utilizan conocimientos de ingeniería de software y desarrollo móvil para resolver un problema práctico del negocio.
- Descriptiva, porque se analizan los procesos manuales existentes de ventas e inventario, permitiendo identificar necesidades, fallas y oportunidades de mejora.
- Tecnológica, porque se diseña y desarrolla una aplicación móvil como solución informática que automatiza los procesos analizados.

8.4. Enfoque de la investigación

La investigación se desarrolla bajo un enfoque mixto:

- Cualitativo, mediante la observación directa y entrevistas al propietario y empleados de la Tienda Génesis, con el fin de comprender el funcionamiento real del negocio y las necesidades de los usuarios.
- Cuantitativo, a través de la medición de tiempos de atención, registro de ventas, control de inventario y reducción de errores antes y después del uso del sistema.

El desarrollo de la solución tecnológica se realiza mediante la metodología ágil Scrum, organizando el trabajo en sprints que permite una entrega incremental de funcionalidades y la validación continua del sistema con los usuarios finales.

8.5. Metodología para cada objetivo

8.5.1. Primer objetivo: Análisis de los procesos actuales de registro de ventas e inventario

Para el cumplimiento de este objetivo se aplica una investigación de tipo aplicada con enfoque cualitativo, orientada a comprender de manera detallada los procesos actuales de ventas e inventario que se desarrollan en la Tienda Génesis. El análisis de requerimientos se realiza directamente en el entorno real del negocio, permitiendo identificar las necesidades específicas del propietario y de los empleados.

La recolección de información se lleva a cabo mediante entrevistas estructuradas dirigidas al propietario y a los colaboradores, lo que permite conocer las principales dificultades del registro manual de ventas, el control del inventario y la elaboración de reportes. La entrevista facilita la identificación de necesidades, tales como el control de stock, el registro de clientes y la generación de comprobantes.

De manera complementaria, se realiza la observación directa del funcionamiento diario del negocio, registrándose las actividades relacionadas con la atención al cliente, el registro de ventas y la verificación de productos disponibles.

Con base en la información recopilada, los requerimientos del sistema se organizan y priorizan en forma de historias de usuario, las cuales conforman el Product Backlog del proyecto, asegurando que el desarrollo responda a las necesidades reales del negocio bajo enfoque de Scrum.

8.5.2. Segundo Objetivo: Diseño de la arquitectura del sistema y la base de datos

Para el cumplimiento de este objetivo se realiza el diseño del sistema estableciendo, en primer lugar, el alcance funcional mediante la definición de requerimientos y la elaboración del diagrama de casos de uso, con el fin de

identificar las funcionalidades principales y la interacción de los actores con el sistema.

Posteriormente, se diseña el modelo de datos utilizando SQLite, identificando las entidades necesarias para la gestión de productos, ventas, inventario, clientes, usuarios y gastos. Con base en ello se elabora el modelo entidad-relación (ER), definiendo relaciones y restricciones que aseguren la integridad de la información almacenada.

A continuación, se define la arquitectura del proyecto utilizando el patrón Model-View-ViewModel (MVVM), con el propósito de separar responsabilidades entre la capa de datos (Model), la lógica de negocio (ViewModel) y la presentación (View), facilitando la mantenibilidad y escalabilidad del sistema.

Seguidamente, se elaboran prototipos de interfaz en Figma de las pantallas principales, priorizando simplicidad, claridad visual y facilidad de navegación. Estos prototipos permiten validar el flujo de uso del sistema y orientar el desarrollo posterior en dispositivos móviles Android, considerando el contexto real de la Tienda Génesis.

Finalmente, como apoyo al diseño estructural, se desarrolla el diagrama de clases, representando la estructura lógica del sistema (clases, atributos, métodos y relaciones), con el fin de complementar la definición técnica y facilitar la implementación.

8.5.3. Tercer Objetivo: Programación de la aplicación móvil

El desarrollo de la aplicación móvil se realiza mediante la metodología ágil Scrum, organizando el trabajo en sprints de duración definida. Al inicio del proyecto se elabora el Product Backlog, en el cual se incluyen las funcionalidades identificadas durante la fase de análisis de requerimientos.

Durante cada sprint se seleccionan historias de usuario priorizadas, las cuales se desarrollan de manera incremental. En cada iteración se programan funcionalidades específicas como: inicio de sesión, gestión de productos, registro

de ventas, control de inventario, registro de gastos y generación de reportes. El desarrollo se realiza utilizando el framework Flutter y el lenguaje Dart, integrando una base de datos local SQLite que permite el funcionamiento del sistema sin conexión a internet. Al finalizar cada sprint se realiza la revisión de las funcionalidades desarrolladas, identificando ajustes necesarios antes de continuar con la siguiente iteración.

Este enfoque permite un avance progresivo del sistema, asegurando que cada módulo sea desarrollado y revisado antes de incorporar nuevas funcionalidades, manteniendo el control del proyecto y la calidad del software.

8.5.4. Cuarto Objetivo: Validación del funcionamiento de sistema

Para el cumplimiento de este objetivo se realiza la validación del sistema mediante pruebas locales, con el propósito de comprobar su funcionamiento, la correcta captura y almacenamiento de datos en SQLite y la usabilidad en el contexto real de la Tienda Génesis. En primer lugar, se elabora una matriz de casos de prueba por módulo como inicio de sesión, productos, clientes, ventas, inventario, gastos y reportes, registrando para cada caso: datos de entrada, procedimiento, resultado esperado, resultado obtenido y estado (aprobado/no aprobado).

Posteriormente, se ejecutan pruebas funcionales para verificar el comportamiento individual de cada módulo, y pruebas de integración para comprobar la consistencia entre procesos relacionados al registrar una venta, generar comprobante en PDF y visualizar la venta en reportes. Finalmente, se realiza una validación de aceptación y usabilidad con el usuario final, solicitando la ejecución de tareas reales del negocio (registrar venta, consultar inventario, registrar gasto y generar reportes), registrando observaciones y mejoras.

Los hallazgos detectados durante la validación se documentan como incidencias y se aplican ajustes hasta obtener una versión estable del sistema. Adicionalmente, se obtiene resultados cuantitativos y cualitativos: se registran tiempos de atención antes y después del uso del sistema y se evaluó la calidad del sistema con base en características seleccionadas del modelo ISO/IEC 25010

(funcionalidad, usabilidad, rendimiento, seguridad y portabilidad), utilizando evidencias obtenidas durante las pruebas realizadas.

9. RESULTADOS

9.1. Primer objetivo: Análisis de los procesos actuales de registro de ventas e inventario

Como resultado del levantamiento de información realizado en la Tienda Génesis, se identificaron las principales deficiencias del método manual de registro de ventas e inventario, así como las necesidades funcionales y no funcionales que debe cubrir el sistema propuesto. Para alcanzar este objetivo se aplicaron entrevistas estructuradas al propietario y a los colaboradores, complementadas con observación directa del proceso operativo.

Los hallazgos generales evidenciaron pérdida frecuente de información por el uso de cuadernos físicos, errores en cálculos diarios, ausencia de control real del stock disponible, duplicidad o inconsistencias en registros y lentitud en la atención al cliente, lo que limita la toma de decisiones del negocio y afecta su eficiencia operativa. En función de estos hallazgos, se definieron requerimientos que posteriormente se transformaron en historias de usuario para estructurar el Product Backlog bajo el enfoque de Scrum.

9.1.1. Resultados del levantamiento de información mediante entrevistas

Las entrevistas permitieron confirmar que el proceso manual utilizado en la Tienda Génesis generaba errores constantes, pérdida de información y retrasos operativos, evidenciando la necesidad de un sistema digital automatizado. Se identificaron necesidades explícitas como control de stock, registro de clientes, generación de comprobantes y reportes, además de una mayor organización de la información de ventas e inventario.

Tabla 2. Resultados del Levantamiento de Información Mediante Entrevistas

Actor entrevistado	Rol en el negocio	Principales hallazgos identificados
Jhonny Belisario Huatatoca Alvarado	Propietario / Administrador	Se identificó pérdida frecuente de información debido al uso de cuadernos físicos, errores en el cálculo diario de ventas, ausencia de control del stock real y dificultad para conocer las ganancias reales del negocio.
Juan Carlos Andi Tapuy	Empleado	Se evidenció desorganización en el registro manual de productos, inexactitud en el inventario, duplicación de registros, dificultad para identificar productos agotados o caducados y lentitud en la atención al cliente.
Raúl Fidel Andi Tapuy	Empleado	Se identificaron dificultades en el control de stock, problemas para detectar productos agotados o caducados y limitaciones del sistema manual para registrar correctamente las operaciones diarias.

Nota: Información recopilada mediante entrevistas estructuradas al propietario y colaboradores de la Tienda Génesis. Fuente: Elaboración propia (2025).

9.1.2. Resultados de la observación directa del proceso operativo

La observación directa evidenció que el método manual afectaba la eficiencia del negocio, incrementando los tiempos de atención y reduciendo la confiabilidad de la información registrada. Se verificó que el registro de ventas se realizaba únicamente en cuadernos, el inventario no contaba con control en tiempo

real y existía deterioro o pérdida de registros físicos. Además, se detectaron tiempos prolongados de atención debido a la búsqueda manual de precios y productos.

Tabla 3. *Resultados de la Observación Directa del Proceso Operativo*

Aspecto observado	Situación identificada
Registro de ventas	Se realizaba exclusivamente en cuadernos, lo que provocaba errores en precios y totales.
Control de inventario	No existía control en tiempo real del stock disponible.
Conservación de información	Se observó deterioro y pérdida de registros físicos.
Atención al cliente	Se detectaron tiempos prolongados por búsqueda manual de precios y productos.

Nota: Registro de hallazgos obtenidos a partir de la observación directa del proceso de ventas e inventario en la Tienda Génesis. Fuente: Elaboración propia (2025).

9.1.3. Problemas detectados en los procesos actuales

Con base en entrevistas y observación, se consolidaron los principales problemas detectados en las áreas de ventas, inventario, control financiero y operación diaria. Esta síntesis permitió orientar la definición de requerimientos del sistema de forma coherente con el contexto real del negocio.

Tabla 4. *Problemas Detectados en los Procesos Actuales*

Área	Problemas identificados
Ventas	Errores frecuentes en cálculos, registros incompletos y diferencias en los totales diarios.

Área	Problemas identificados
Inventario	Desactualización del stock, productos agotados no detectados a tiempo y registros duplicados.
Control financiero	Desconocimiento de ganancias reales y dificultad para identificar gastos diarios.
Operación diaria	Lentitud en la atención y dependencia total de registros manuales.

Nota: Síntesis de problemas identificados a partir de entrevistas y observación directa. Fuente: Elaboración propia (2025).

9.1.4. Requerimientos funcionales definidos a partir del análisis

A partir del diagnóstico realizado, se definieron requerimientos funcionales orientados a automatizar el registro de ventas e inventario, reducir errores y permitir reportes oportunos. Se priorizó además el funcionamiento local sin conexión, debido a las limitaciones del entorno.

Tabla 5. *Requerimientos Funcionales Definidos a Partir del Análisis*

Categoría	Requerimientos obtenidos
Ventas	Registro de ventas en tiempo real, cálculo automático de subtotales y totales, registro de método de pago.
Inventario	Actualización automática de stock, alertas de bajo inventario y productos agotados.
Productos	Visualización de productos disponibles, búsqueda rápida y resaltado de stock bajo.
Reportes	Generación de reportes de ventas e inventario en formato PDF.

Categoría	Requerimientos obtenidos
Funcionamiento	Operación completa sin conexión a internet mediante base de datos SQLite.

Nota: Requerimientos funcionales establecidos a partir del diagnóstico de necesidades del negocio. Fuente: Elaboración propia (2025).

9.1.5. Requerimientos no funcionales definidos

De forma complementaria, se establecieron requerimientos no funcionales que aseguren calidad del sistema en seguridad, rendimiento, usabilidad y mantenibilidad. Estos criterios permiten medir el desempeño del sistema y garantizar su adecuación al uso real.

Tabla 6. *Requerimientos No Funcionales Obtenidos*

Dimensión	Resultados definidos
Seguridad	Autenticación por usuario y contraseña, control de acceso por roles e integridad de datos.
Rendimiento	Respuesta del sistema inferior a 2 segundos durante operaciones frecuentes.
Usabilidad	Interfaz sencilla, navegación intuitiva y adaptación a dispositivos móviles.
Mantenibilidad	Arquitectura MVVM y código modular para futuras mejoras.

Nota: Requerimientos no funcionales definidos con base en criterios de seguridad, rendimiento, usabilidad y mantenibilidad del sistema. Fuente: Elaboración propia (2025).

9.1.6. Elaboración de historias de usuario y Product Backlog

Como salida final del objetivo 1, los requerimientos funcionales y no funcionales fueron transformados en historias de usuario, las cuales se priorizaron para conformar el Product Backlog. Esta organización permitió planificar el

proyecto mediante Scrum, asegurando que el desarrollo se realizó por incrementos y alineado a necesidades reales del negocio.

Tabla 7. *Product Backlog (Historias de Usuario)*

ID	Historia de usuario	Prioridad	Criterio de aceptación (resumen)
HU-01	Como usuario, quiero iniciar sesión para acceder al sistema	Alta	Acceso válido y controlado.
HU-02	Como usuario, quiero gestionar productos para mantener precios correctos	Alta	CRUD completo de productos.
HU-03	Como usuario, quiero registrar ventas para no olvidar transacciones	Alta	Guarda venta con total y método de pago.
HU-04	Como usuario, quiero controlar inventario para evitar faltantes	Alta	Consulta y actualiza stock.
HU-05	Como usuario, quiero registrar gastos para conocer ganancia real	Media	Guarda gasto con fecha y monto.
HU-06	Como usuario, quiero generar reportes para conocer ventas/ganancias	Alta	Reporte generado correctamente.

Nota: Extracto del Product Backlog elaborado en formato de historias de usuario priorizadas. Fuente: Elaboración propia (2025).

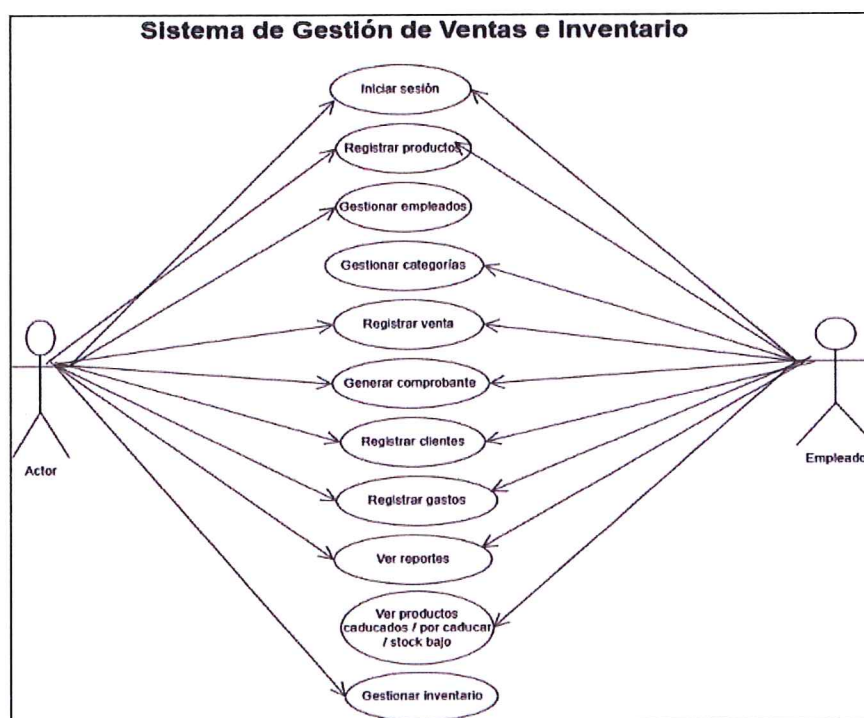
9.2. Segundo Objetivo: Diseñar la arquitectura del sistema y la base de datos

Como resultado de la etapa de diseño, se definió la estructura técnica del sistema de gestión de ventas e inventario para la Tienda Génesis, estableciendo una base sólida para garantizar la integridad de los datos y facilitar el desarrollo y mantenimiento del software. Para ello se desarrollaron los componentes de diseño en el siguiente orden: definición funcional mediante casos de uso, modelado de

datos en SQLite, definición de arquitectura MVVM, prototipado de interfaz en Figma y modelado estructural mediante diagrama de clases.

En primer lugar, se construyó el diagrama de casos de uso, con el objetivo de representar las funcionalidades principales del sistema y la interacción de los actores con cada una de ellas. Este diagrama permitió identificar claramente los procesos vinculados a la gestión de ventas, inventario, clientes, gastos y reportes, delimitando el alcance del sistema y orientando el diseño de módulos y pantallas.

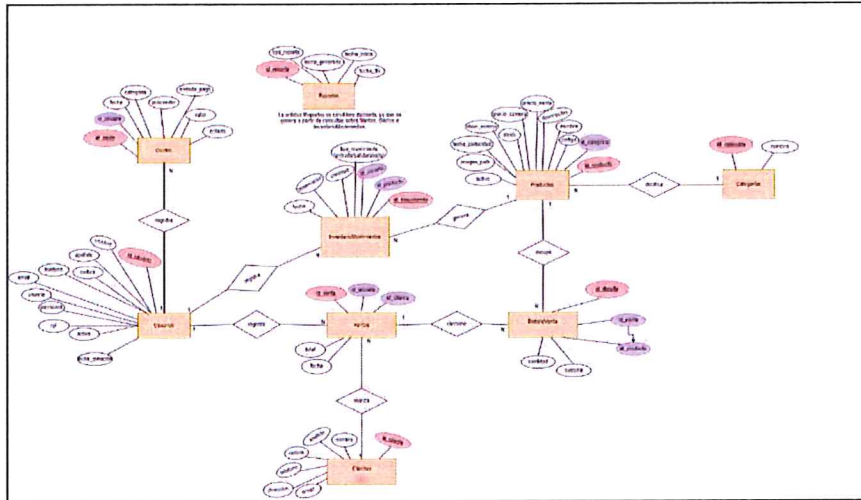
Ilustración 2. Diagrama de Casos de Uso del Sistema de Gestión de Ventas e Inventario.



Nota: Funcionalidades principales y relación entre actores y procesos del sistema. Fuente: Elaboración propia (2025).

Posteriormente, se elaboró el modelo entidad-relación (ER) para la base de datos en SQLite, representando las entidades principales del sistema y sus relaciones. Este diseño permitió definir las tablas necesarias y su interdependencia, asegurando una correcta organización y almacenamiento de la información del negocio.

Ilustración 3. Diagrama Entidad - Relación del Sistema de Gestión de Ventas e Inventario.



Nota: Representación de entidades, atributos y relaciones definidas para la base de datos del sistema. Fuente: Elaboración propia (2025).

A continuación, se definió la arquitectura del proyecto basada en el patrón Model-View-ViewModel (MVVM), logrando una separación de responsabilidades entre la capa de datos (Model), la lógica de negocio (ViewModel) y la presentación (View). Esta arquitectura contribuyó a mejorar la organización del código y facilitar el mantenimiento del sistema.

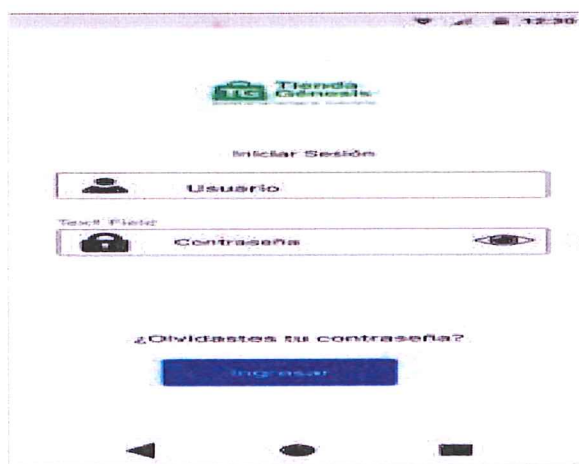
Ilustración 4. Arquitectura del Proyecto Basada en el Patrón MVVM.



Nota: Organización del proyecto en capas (Model, ViewModel y View) para separar responsabilidades. Fuente: Elaboración propia (2025).

Seguidamente, se elaboraron prototipos en **Figma** para definir los módulos y la información que maneja cada rol (administrador y empleado). Estos prototipos sirvieron como base para el desarrollo posterior de las pantallas en Flutter, priorizando una interfaz clara, simple y fácil de usar, acorde con el entorno real de la Tienda Génesis y orientada a dispositivos móviles Android.

Ilustración 5. *Prototipo de Pantalla de Inicio de Sesión del Sistema.*



Nota: Prototipo de interfaz para acceso mediante usuario y contraseña. Fuente: Elaboración propia (Figma); (2025).

Ilustración 6. *Prototipo de Menú Principal del Rol Administrador.*



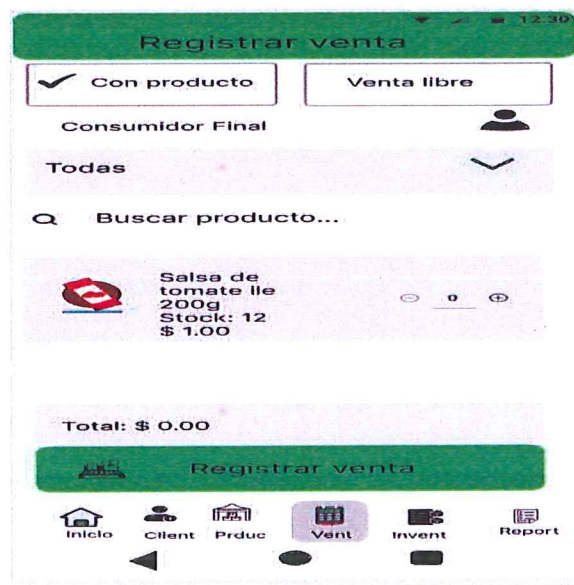
Nota: Prototipo de opciones principales disponibles para el rol Administrador. Fuente: Elaboración propia (Figma); (2025).

Ilustración 7. Prototipo de Menú Principal del Rol Empleado.



Nota: Prototipo de opciones operativas disponibles para el rol Empleado. Fuente: Elaboración propia (Figma); (2025).

Ilustración 8. Prototipo del Módulo de Registro de Ventas.



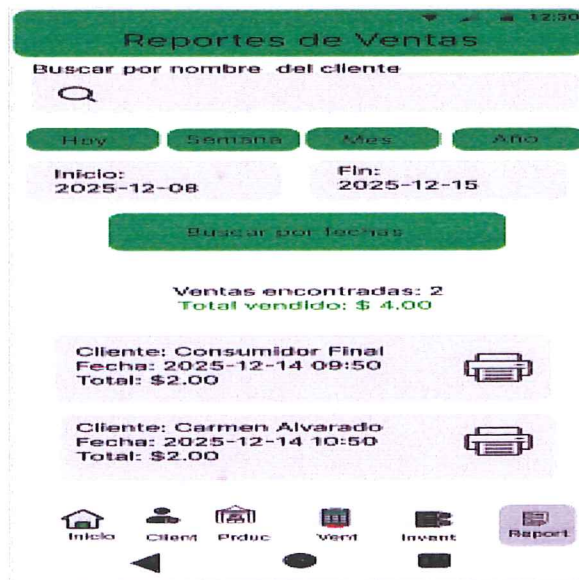
Nota: Prototipo del flujo de registro de ventas y cálculo de totales. Fuente: Elaboración propia (Figma); (2025).

Ilustración 9. Prototipo del Módulo de Control de Inventario.



Nota: Prototipo de entradas/salidas y actualización de stock.
Fuente: Elaboración propia (Figma); (2025).

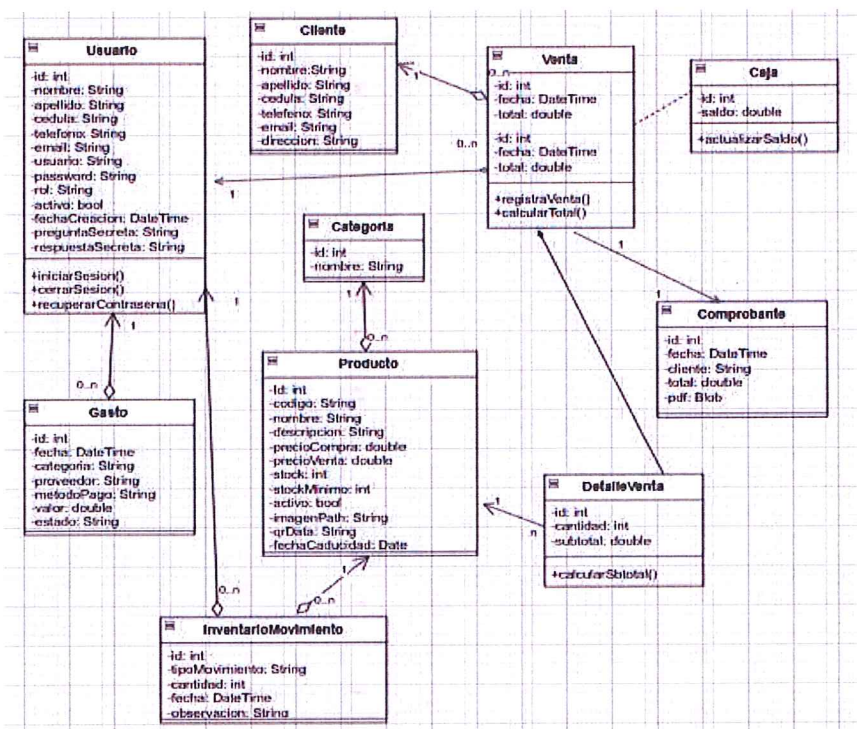
Ilustración 10. Prototipo del Módulo de Reportes del Sistema.



Nota: Prototipo para consulta de reportes del negocio. **Fuente:** Elaboración propia (Figma); (2025).

Finalmente, como apoyo al diseño estructural, se desarrolló el diagrama de clases, el cual representa la estructura lógica del sistema mostrando las clases principales, atributos, métodos y relaciones. Este diseño permitió comprender la organización interna de los componentes y su interacción para gestionar ventas, inventario, usuarios y comprobantes.

Ilustración 11. Diagrama de Clases del Sistema de Gestión de Ventas e Inventario.



Nota: Estructura lógica del sistema: clases principales, atributos y relaciones. Fuente: Elaboración propia (2025).

9.3. Tercer Objetivo: Programación de la aplicación móvil de ventas e inventario

Como resultado del desarrollo, se implementó la aplicación móvil de gestión de ventas e inventario para la Tienda Génesis utilizando Flutter y Dart, integrando una base de datos local SQLite, lo que permitió el funcionamiento del sistema sin conexión a internet. El proceso de construcción se ejecutó bajo la metodología ágil Scrum, lo cual permitió desarrollar el sistema de forma incremental por módulos y

realizar revisiones periódicas de cada incremento antes de incorporar nuevas funcionalidades.

Durante la programación se consolidó un producto funcional que automatizó el registro de ventas y el control de inventario, integrando módulos de inicio de sesión, clientes, productos, ventas, inventario, gastos y reportes, incluyendo la consulta de ventas por rango de fechas. Asimismo, se implementó el cálculo automático del total en ventas, el almacenamiento de información en SQLite y la generación de reportes, fortaleciendo la organización y trazabilidad de los datos del negocio.

Tabla 8. *Plan de Sprints y Entregables*

Sprint	Objetivo del sprint	Historias / funcionalidades incluidas	Incremento entregado
Sprint 1	Preparar base del sistema	Login + estructura base + navegación inicial	Acceso al sistema y estructura inicial
Sprint 2	Gestión de datos del negocio	Clientes + Productos	CRUD de clientes y productos
Sprint 3	Operación principal	Ventas + cálculo automático + registro en SQLite	Registro de ventas funcionando
Sprint 4	Control y análisis	Inventario + Gastos + Reportes	Módulos integrados y reportes

Nota: Planificación del desarrollo incremental mediante Scrum, organizado por sprints y entregables. Fuente: Elaboración propia; (2025).

9.3.1. Desarrollo de módulos implementados

Como resultado de la programación, el sistema incorporó módulos que cubren los procesos del negocio. Cada módulo fue implementado de forma progresiva, asegurando que la información se almacene en SQLite y pueda

consultarse posteriormente. Además, se priorizó una interfaz clara para facilitar el uso por parte del propietario y colaboradores.

Tabla 9. *Módulos Implementados y Funcionalidad Alcanzada*

Módulo	Funcionalidad desarrollada	Resultado
Inicio de sesión	Acceso con credenciales y control de ingreso	Implementado
Clientes	Registro, edición, listado y consulta de clientes	Implementado
Productos	Registro, edición, control de precios, stock y caducidad	Implementado
Ventas	Selección de productos, cantidades, total automático	Implementado
Inventario	Entradas/salidas y actualización de stock	Implementado
Gastos	Registro y consulta de egresos	Implementado
Reportes	Acceso a reportes por categoría	Implementado
Reporte de ventas	Consulta por rango de fechas y detalle	Implementado

Nota: Relación de módulos implementados y funcionalidades alcanzadas durante el desarrollo de la aplicación. Fuente: Elaboración propia; (2025).

9.3.2. Integración de base de datos local (SQLite)

Como parte del resultado técnico, se implementó el almacenamiento en SQLite para asegurar la persistencia de datos y el funcionamiento en entornos con conectividad limitada. La base local permitió registrar ventas, actualizar inventario y consultar reportes sin depender de internet, garantizando continuidad operativa. Adicionalmente, la estructura de datos permitió mantener consistencia entre los procesos de ventas e inventario, evitando pérdida de información y facilitando el análisis del negocio.

9.3.3. Resultados visuales del sistema implementado

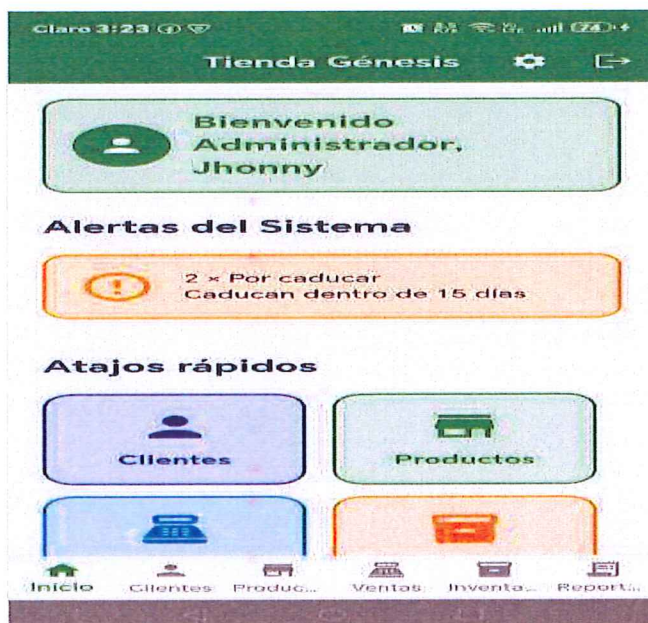
A continuación, se presentan capturas de pantalla del sistema en funcionamiento, evidenciando la implementación real de los módulos desarrollados. Estas ilustraciones demuestran el cumplimiento del objetivo 3, al mostrar pantallas operativas construidas en Flutter con navegación funcional y acceso a datos almacenados en SQLite.

Ilustración 12. *Pantalla de Inicio de Sesión del Sistema Móvil.*



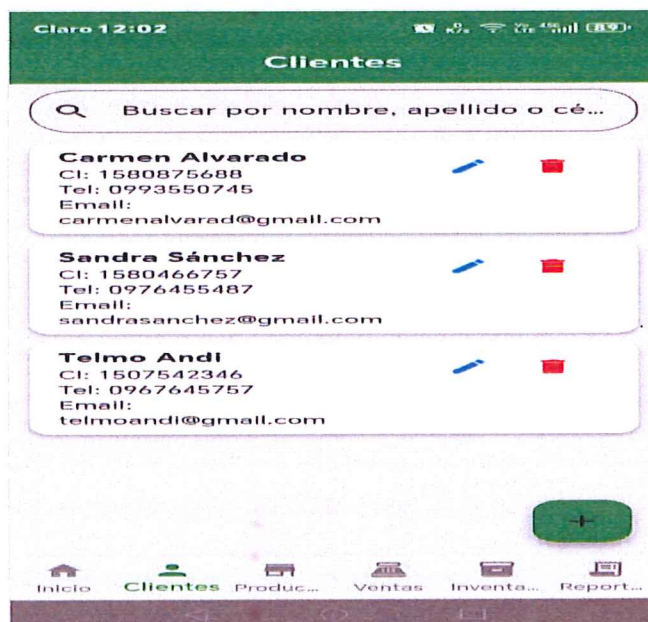
Nota: Acceso al sistema mediante credenciales de usuario. Fuente: Elaboración propia; (2025).

Ilustración 13. Menú Principal del Rol Administrador.



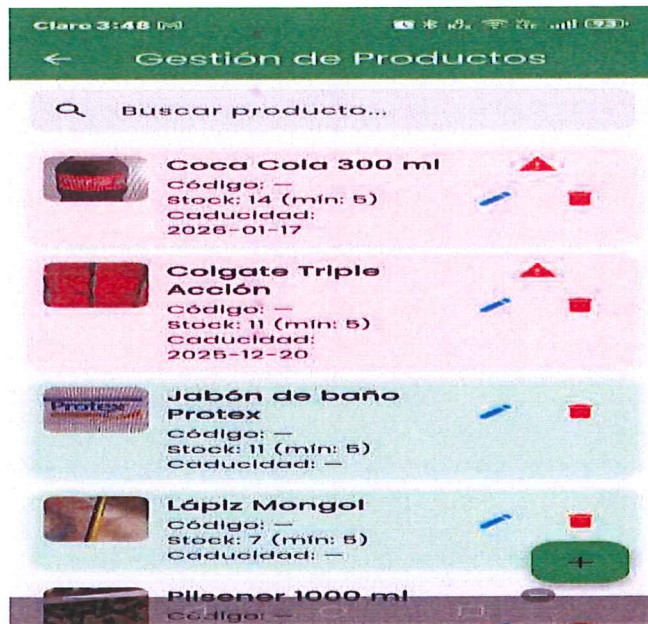
Nota: Módulos disponibles para administración del negocio. Fuente: Elaboración propia; (2025).

Ilustración 14. Gestión de clientes del sistema.



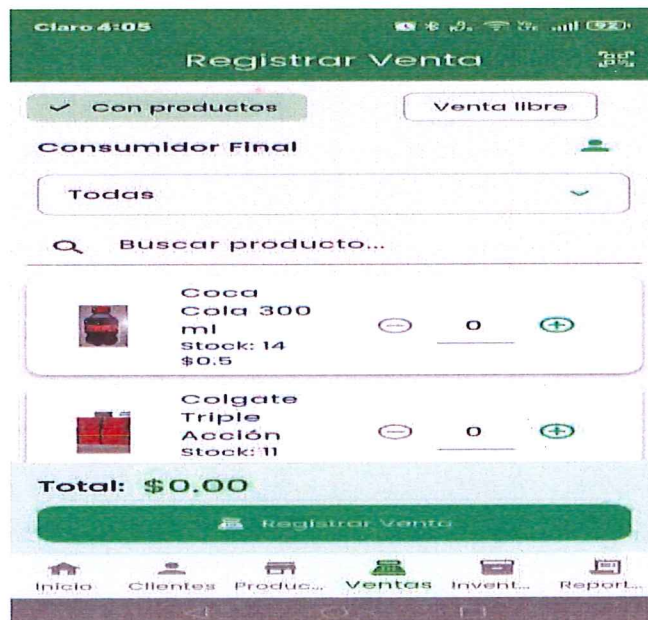
Nota: Registro, consultas y administración de clientes. Fuente: Elaboración propia; (2025).

Ilustración 15. Gestión de Productos del Sistema.



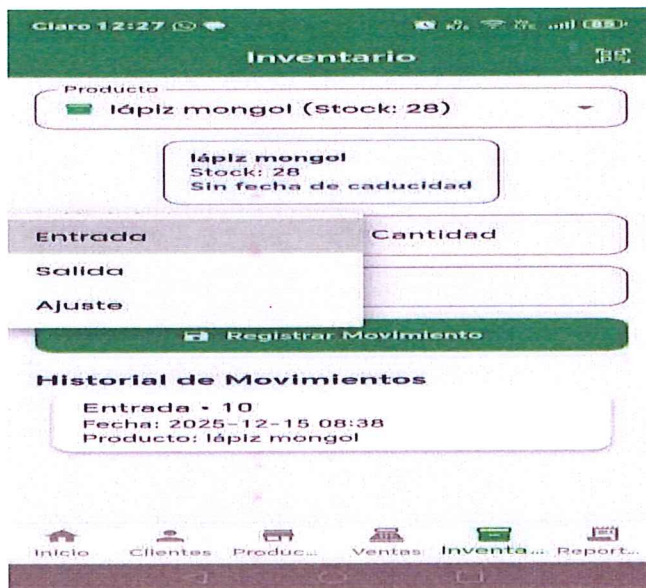
Nota: Registro y control de productos, precios, stock y caducidad.
Fuente: Elaboraci3n propia (2025).

Ilustraci3n 16. Registro de Ventas del Sistema.



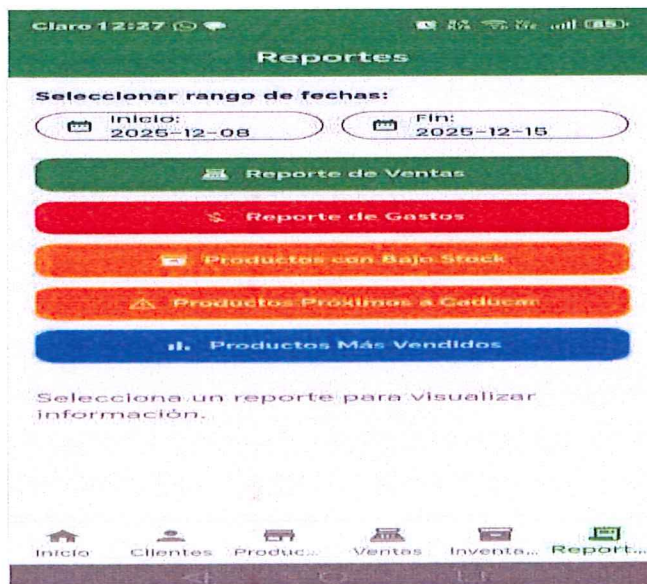
Nota: Selecci3n de productos y c3lculo autom3tico del total de venta.
Fuente: Elaboraci3n propia; (2025).

Ilustración 17. Control de Inventario del Sistema.



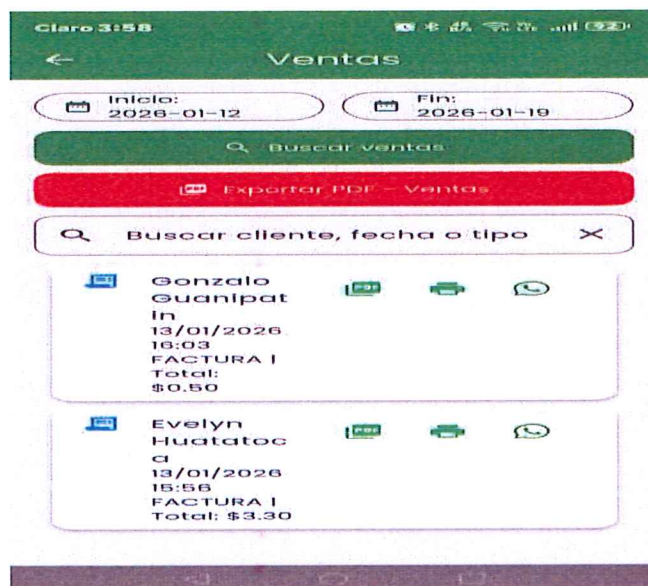
Nota: Registro de movimientos del inventario y actualización de stock.
Fuente: Elaboración propia; (2025).

Ilustración 18. Módulo de Reportes del Sistema.



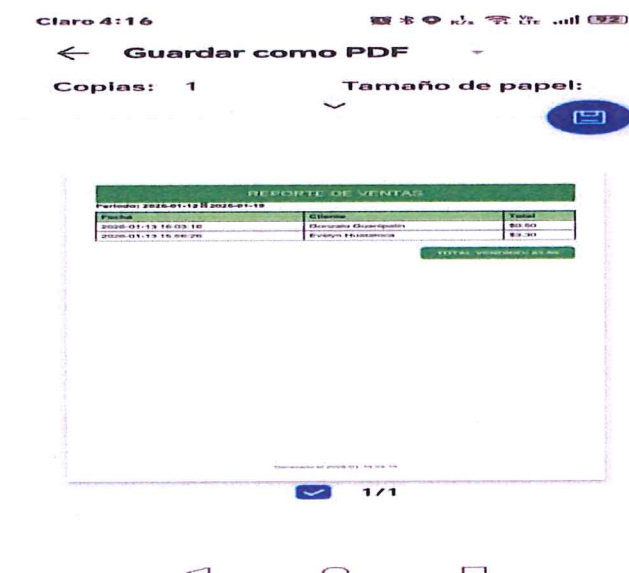
Nota: Acceso a reportes para análisis y control del negocio. **Fuente:** Elaboración propia; (2025).

Ilustración 19. Reporte de Ventas Filtrado por Rango de Fechas.



Nota: Consulta de ventas por fechas con detalle de registros. Fuente: Elaboración propia; (2025).

Ilustración 20. Reporte de Ventas Generado en Formato PDF.



Nota: Reporte exportado en PDF con detalle de ventas y total acumulado. Fuente: Elaboración propia (2025).

9.4. Cuarto Objetivo: Validación del funcionamiento del sistema

Como resultado de la validación del sistema, se comprobó su funcionamiento mediante pruebas locales, verificando el correcto registro de los datos ingresados y el almacenamiento de información en SQLite, así como la usabilidad del sistema en el contexto real de la Tienda Génesis. Para ello se aplicó una matriz de casos de prueba por módulo, posteriormente se ejecutaron pruebas funcionales y pruebas de integración entre procesos relacionados. Finalmente, se desarrolló una validación de aceptación y usabilidad con el usuario final mediante la ejecución de tareas reales del negocio. Los resultados evidenciaron que el sistema cumple con las operaciones principales: acceso, gestión de clientes y productos, registro de ventas, control de inventario, registro de gastos, generación de reportes y emisión de comprobantes y reportes en formato PDF.

9.4.1. Resultados de la matriz de casos de prueba pruebas funcionales por módulo

La ejecución de la matriz de casos de prueba permitió comprobar el funcionamiento de cada módulo del sistema de manera individual. En cada caso se registraron los elementos establecidos en la matriz: datos de entrada, procedimiento, resultado esperado, resultado obtenido y estado (aprobado/no aprobado). Se evidenció que la información ingresada se almacena correctamente en SQLite y puede consultarse posteriormente sin inconsistencias, confirmando estabilidad del sistema en entorno local.

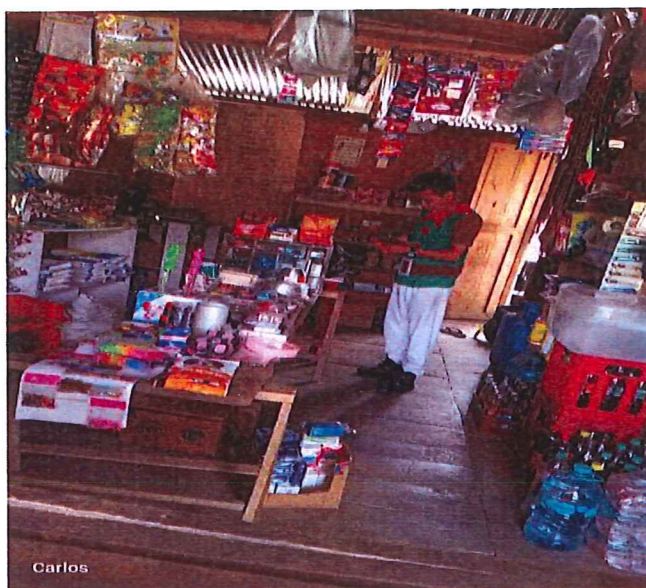
Tabla 10. *Resultados de Pruebas Funcionales por Módulo*

Código	Módulo	Caso de prueba	Resultado esperado	Resultado obtenido	Estado
CP-01	Inicio de sesión	Ingreso con credenciales válidas	Acceso correcto al sistema	Acceso correcto	Aprobado

Código	Módulo	Caso de prueba	Resultado esperado	Resultado obtenido	Estado
CP-02	Clientes	Registrar cliente	Cliente guardado y visible en listado	Registro correcto y visible	Aprobado
CP-03	Productos	Registrar producto	Producto guardado y consultable	Registro correcto y consultable	Aprobado
CP-04	Ventas	Registrar venta	Total correcto y venta registrada	Total correcto y venta registrada	Aprobado
CP-05	Inventario	Registrar entrada/salida	Stock actualizado correctamente	Stock actualizado correctamente	Aprobado
CP-06	Gastos	Registrar gasto	Gasto almacenado y reflejado en el módulo	Registro correcto y consultable	Aprobado
CP-07	Reportes	Generar reporte de ventas	Reporte generado por rango de fechas	Reporte generado correctamente	Aprobado
CP-08	PDF	Emitir comprobante	Creación de archivo PDF con el detalle de la venta	PDF generado correctamente	Aprobado

Nota. Casos de prueba funcionales ejecutados por módulo para verificar el comportamiento funcional del sistema en entorno local. Fuente: Elaboración propia (2025).

Ilustración 21. Registro de Ventas Durante las Pruebas Funcionales.



Nota: Evidencia del registro de ventas ejecutado durante pruebas locales de validación funcional. Fuente: Elaboración propia (2025).

9.4.2. Resultados de pruebas de integración

Las pruebas de integración permitieron verificar que los módulos relacionados funcionan de manera coherente y que la información se mantiene consistente entre los datos registrados y los reportes generados. Se comprobó que, al registrar una venta, los datos se guardan en SQLite, se reflejan correctamente en el reporte de ventas por rango de fechas y permiten generar el comprobante correspondiente en formato PDF. Además, se validó que los movimientos de inventario actualizan el stock de forma inmediata dentro del sistema.


Tabla 11. Resultados de Pruebas de Integración

Código	Integración evaluada	Acción realizada	Resultado esperado	Resultado obtenido	Estado
PI-01	Ventas y Reportes	Registrar venta y generar	Venta aparece en el reporte	Venta reflejada correctamente	Aprobado

Código	Integración evaluada	Acción realizada	Resultado esperado	Resultado obtenido	Estado
		reporte por fechas			
PI-02	Ventas y Comprobante PDF	Registrar venta y emitir comprobante	PDF generado con detalle y total	PDF generado correctamente	Aprobado
PI-03	Inventario y Stock	Registrar movimiento y consultar stock	Stock consistente y actualizado	Stock consistente y actualizado	Aprobado

Nota: Pruebas de integración ejecutadas para comprobar consistencia entre módulos relacionados (ventas-reportes-PDF; inventario-stock). Fuente: Elaboración propia (2025).

Ilustración 22. Comprobante de Venta Generado en Formato PDF Durante la Validación.



TIENDA GÉNESIS
RUC: 1501176109
Los Andes - Cotundo - Napo
Teléfono: 0983613075

COMPROBANTE DE VENTA
Factura N°: 10
Fecha: 19/01/2026 09:33
Método de pago: Efectivo
Vendedor: Jhonny Huatatoca

Ciente: Marlene Chimbo

Producto	Cant.	Precio	Subtotal
Coca Cola 300 ml	1	\$0.50	\$0.50
Jabón de baño Protex	1	\$1.00	\$1.00
Lápiz Mongol	1	\$0.50	\$0.50

TOTAL: \$2.00

Nota: Comprobante emitido automáticamente con detalle de productos, cantidades, método de pago y total. Fuente: Elaboración propia (2025).

9.4.3. Resultados de validación de aceptación y usabilidad (usuario final)

Como resultado de la validación de aceptación, el usuario final realizó tareas representativas del negocio, como iniciar sesión, registrar ventas, consultar el inventario y generar reportes. Se evidenció que el sistema cuenta con una navegación clara y módulos bien identificados, lo que facilita la ejecución de las actividades diarias. Además, el usuario confirmó que el sistema reduce el riesgo de errores en los cálculos al generar los totales de forma automática y mejora la disponibilidad de la información, ya que permite generar reportes por rangos de fechas.

Tabla 12. *Validación de Aceptación y Usabilidad*

Tarea evaluada	Actividad	Resultado
T-01	Iniciar sesión	Acceso logrado sin dificultad.
T-02	Registrar venta	Venta registrada correctamente con total automático.
T-03	Consultar inventario	Stock consultado de forma clara y consistente.
T-04	Registrar gasto	Gasto registrado correctamente y visible en el listado.
T-05	Generar reporte	Reporte generado correctamente por rango de fechas.
T-06	Emitir comprobante PDF	Documento PDF emitido correctamente con detalle de venta.

Nota: Tareas representativas ejecutadas por el usuario final para validar usabilidad y aceptación del sistema en el contexto real. Fuente: Elaboración propia (2025).

9.4.4. Incidencias y ajustes aplicados

Durante la validación del sistema no se encontraron fallos graves que impidan su uso. Los cambios que se realizaron fueron principalmente ajustes pequeños en la parte visual, como mejorar la claridad de algunas etiquetas y

organizar mejor ciertos campos. Con estas mejoras se buscó que la aplicación sea más fácil de usar para el usuario y que el sistema quede en una versión más estable.

Tabla 13. *Incidencias y Correcciones Aplicadas*

Código	Tipo	Descripción	Corrección aplicada	Estado
M-01	Mejora	Ajuste de presentación de campos en pantallas	Reorganización en visual	Aplicada
M-02	Mejora	Claridad de textos/etiquetas en formularios	Corrección de etiquetas	Aplicada

Nota: Registro de incidencias detectadas durante la validación y ajustes aplicados para mejorar estabilidad y experiencia de usuario. Fuente: Elaboración propia (2025).

9.4.5. Resultados cuantitativos y cualitativos

Resultados cuantitativos

Los resultados cuantitativos evidenciaron mejoras en el desempeño del sistema. El tiempo promedio de respuesta al registrar operaciones sin conexión fue de 1,40 segundos, cumpliendo el criterio establecido de respuesta menor a 2 segundos. Este valor corresponde al promedio de 10 ejecuciones consecutivas del registro de venta, realizadas en un dispositivo HONOR modelo ALI-NX3P (Android 15, MagicOS 9.0), utilizando base de datos local SQLite.

Ilustración 23. Características del Dispositivo

Utilizado en las Pruebas.



Nota: Captura de “Acerca del teléfono”. Fuente: Elaboración propia (2025).

Asimismo, el tiempo de atención por venta se redujo de aproximadamente 3–5 minutos con el registro manual a 1–2 minutos utilizando la aplicación móvil, con base en la observación del proceso y la ejecución completa de la tarea de registro de venta.

Tabla 14. Comparación de Tiempos (Registro Manual vs Aplicación Móvil)

Actividad	Registro manual	Con aplicación	Evidencia
Registro completo de una venta	3–5 min	1–2 min	Observación directa y ejecución de la tarea
Tiempo de respuesta del sistema (registro)	—	1,40 s (promedio n=10)	Pruebas locales (SQLite, sin conexión)

Nota: Tiempos obtenidos en pruebas locales y observación del proceso en la Tienda Génesis. Fuente: Elaboración propia (2025).

Resultados cualitativos

Desde el enfoque cualitativo, el usuario final manifestó satisfacción con el sistema durante la validación, destacando su sencillez, rapidez y utilidad para la operación diaria. Se valoró especialmente la posibilidad de operar sin conexión a internet, debido a las limitaciones de conectividad del entorno. Además, las observaciones realizadas se relacionaron con mejoras menores de presentación (claridad de etiquetas y organización visual), las cuales fueron corregidas para consolidar una versión más estable.

9.4.6. Validación general del sistema

En resumen, la validación confirmó que el sistema funciona correctamente en pruebas locales, registra y conserva información en SQLite y permite generar comprobantes y reportes en formato PDF. Las pruebas funcionales e integración evidenciaron consistencia en los procesos evaluados, mientras que la validación con el usuario final confirmó la usabilidad del sistema para la operación diaria de la Tienda Génesis, consolidando una versión estable lista para su implementación.

9.4.7. Evaluación de la calidad del sistema según ISO/IEC 25010

El sistema desarrollado para la Tienda Génesis fue evaluado considerando el modelo de calidad ISO/IEC 25010, con el fin de valorar las características más relevantes de acuerdo con el alcance del proyecto y el contexto de uso. La evaluación se sustentó en evidencias obtenidas durante la ejecución de pruebas funcionales, pruebas de integración y la validación de aceptación y usabilidad.

En cuanto a la funcionalidad, se verificó que la aplicación cumple con los requerimientos definidos, permitiendo el registro de ventas, la gestión de clientes y productos, el control de inventario, el registro de gastos y la generación de reportes, incluyendo la emisión de comprobantes y reportes en formato PDF.

Respecto a la usabilidad, durante la validación con el usuario final se evidenció que la navegación es comprensible y que las operaciones principales pueden ejecutarse sin dificultad, debido a que los módulos se encuentran

identificados de forma clara y el flujo de uso resulta coherente con el proceso del negocio.

En relación con la eficiencia del desempeño (rendimiento), se evaluó el tiempo de respuesta de la aplicación durante operaciones clave del sistema. Las mediciones se realizaron en un dispositivo HONOR X9b 5G, con procesador Snapdragon 6 Gen 1, 8,0 GB de RAM y sistema operativo Android 15 (MagicOS 9). Para obtener un valor representativo, cada operación se ejecutó 10 veces y se calculó el tiempo promedio. Como resultado, la aplicación presentó un tiempo promedio de respuesta de 1,40 s, lo cual se considera adecuado para el uso operativo del negocio. Este rendimiento se obtuvo durante pruebas locales utilizando base de datos SQLite, permitiendo una operación fluida sin conexión a internet, aspecto relevante para el entorno de implementación.

Operaciones evaluadas:

- Registro de venta
- Búsqueda de cliente/producto
- Generación de comprobante/PDF
- Consulta de inventario

En términos de seguridad, se comprobó que el sistema cuenta con control de acceso mediante credenciales (usuario y contraseña) y gestión de permisos por roles. Se establecieron criterios mínimos de contraseña, tales como mínimo 8 caracteres e inclusión de letras y números, con el fin de reducir accesos no autorizados. Además, el acceso a funcionalidades se gestiona por módulo, permitiendo que cada rol acceda únicamente a las secciones que le corresponden (por ejemplo: Administrador: gestión de usuarios y reportes; Vendedor: registro de ventas y consultas), garantizando la confidencialidad y el control de la información.

Ejemplo de control por módulo:

- **Administrador:** gestiona usuarios, productos y reportes.
- **Vendedor:** registra ventas, consulta stock y genera comprobantes.

Finalmente, la portabilidad se evidenció mediante su funcionamiento en dispositivos Android, lo que permitió su instalación y uso en teléfonos móviles compatibles con las necesidades del negocio.

Alcance de la evaluación: La evaluación se enfocó en las características de funcionalidad, usabilidad, rendimiento, seguridad y portabilidad, debido a que son las más relevantes para el alcance del proyecto y el contexto real de uso en el negocio. Estas características permiten verificar que el sistema cumple con los requerimientos principales, es fácil de usar, responde en tiempos adecuados, protege la información y puede operar en dispositivos Android sin inconvenientes.

Con base en esta evaluación, se determinó que el sistema cumple con los criterios de calidad considerados según ISO/IEC 25010 para el presente proyecto.

Tabla 15. *Evaluación del Sistema Según Normas ISO/IEC 25010*

Característica	Evidencia en el sistema	Resultado
Funcionalidad	Registro de ventas, control de inventario, gestión de clientes/productos, reportes y generación de PDF verificados en pruebas	Cumple
Usabilidad	Validación con usuario final: navegación comprensible y ejecución de tareas sin dificultad	Cumple
Rendimiento	Tiempo de respuesta promedio de 1,40s en operaciones sin conexión (pruebas locales)	Cumple
Seguridad	Acceso mediante usuarios y contraseñas con roles definidos	Cumple
Portabilidad	Funcionamiento en dispositivos Android	Cumple

Nota: Evaluación de características seleccionadas del modelo ISO/IEC 25010 basada en evidencias obtenidas durante las pruebas. Fuente: Elaboración propia (2025).

10. CONCLUSIONES

El análisis de los procesos actuales de registro de ventas e inventario de la Tienda Génesis permitió identificar limitaciones del manejo manual (cuadernos y cálculos), tales como inconsistencias en registros, dificultad para consultar información histórica y falta de control sistemático de stock y ventas. Este diagnóstico facilitó definir con precisión las necesidades funcionales del sistema, considerando el contexto de operación y las limitaciones de conectividad.

El diseño de la arquitectura del sistema bajo el patrón MVVM permitió estructurar de forma organizada la interfaz, la lógica de negocio y la gestión de datos, favoreciendo el mantenimiento y la escalabilidad. Asimismo, el uso de SQLite como base de datos local aseguró la persistencia e integridad de la información y permitió el funcionamiento del sistema sin conexión a internet, acorde con las necesidades de la Tienda Génesis.

El desarrollo de la aplicación móvil con Flutter (Dart) permitió implementar los módulos necesarios para la gestión de productos, clientes, ventas, inventario, gastos y reportes, incluyendo consulta por rango de fechas y exportación de reportes en PDF. Además, se incorporaron categorías para organizar productos y alertas de inventario (stock bajo, productos caducados y productos por caducar), fortaleciendo el control preventivo y reduciendo errores asociados a cálculos y registros manuales mediante la automatización del total y el registro digital.

La validación del sistema mediante pruebas funcionales, de integración y de aceptación/usabilidad en el entorno real permitió comprobar el correcto registro de la información y la operación de las funcionalidades principales (ventas, actualización de inventario, reportes por fechas, exportación PDF y alertas). Con ello se confirmó el cumplimiento de los objetivos planteados y la viabilidad del sistema para su uso cotidiano en la Tienda Génesis.

11. RECOMENDACIONES

Realizar una revisión periódica (mensual o bimestral) de los procesos del negocio y de las necesidades funcionales del sistema (nuevos reportes, cambios en productos, ajustes en inventario), con el fin de mantener la aplicación alineada a la operación real de la Tienda Génesis.

Implementar una rutina de respaldos de la base de datos SQLite (semanal o quincenal) y mantener credenciales/roles actualizados para proteger la información. Esto permitirá prevenir pérdida de datos por fallas del dispositivo y asegurar continuidad del control de ventas e inventario.

Usar la aplicación como único medio oficial para registrar ventas, movimientos de inventario y gastos (evitando registros paralelos en cuadernos), de manera que los reportes por fechas y el control de stock reflejen información completa y consistente. Además, mantener actualizadas las categorías para facilitar búsquedas y análisis.

Capacitar al propietario y colaboradores en tareas clave (registro de venta, control de inventario, reportes por rango de fechas y exportación PDF) y aplicar una mejora continua basada en incidencias reales. Se recomienda revisar diariamente las alertas del inventario (stock bajo, productos caducados y por caducar) para prevenir desabastecimientos y pérdidas por caducidad, mejorando la rotación de productos.

12. BIBLIOGRAFÍA

- Abril, C., & Javier, H. (2022). *Sistema de control de ventas de insumos agrícolas y veterinarios para la microempresa agropen mediante el uso del pronóstico cuantitativo de suavizamiento exponencial*. La Libertad, Santa Elena, Ecuador.: Universidad Estatal Península de Santa Elena.
- Agencia Española de Protección de Datos. (octubre de 2019). *Guía de privacidad desde el diseño (v. octubre 2019)*. Agencia Española de Protección de Datos. <https://www.aepd.es/guias/guia-privacidad-desde-diseno.pdf>
- Agencia Española de Protección de Datos. (2020). *Guía de protección de datos por defecto*. Agencia Española de Protección de Datos. <https://www.aepd.es/guias/guia-proteccion-datos-por-defecto.pdf>
- Android Developers. (10 de febrero de 2025). *Save data using SQLite*. Recuperado el 20 de enero de 2026, de Android Developers. <https://developer.android.com/training/data-storage/sqlite>
- Android Developers. (10 de febrero de 2026). *Guide to app architecture*. Android Developers. <https://developer.android.com/topic/architecture>
- Angúlo Daza, N., & Obando Cruz, Ó. (2025). Patrones arquitectónicos y disponibilidad en implementaciones de federación IoT: Una revisión sistemática de literatura. *Revista Instituto Superior Tecnológico Vicente Rocafuerte (REVISTVR)*, 1(1), 1-22. <https://revista.istvr.edu.ec/wp-content/uploads/2025/03/PATRONES-ARQUITECTONICOS-Y-DISPONIBILIDAD-EN-IMPLEMENTACIONES-DE-FEDERACION-IOIOT-UNA-REVISION-SISTEMATICA-DE-LITERATURA.pdf>
- Asamblea Nacional del Ecuador. (2002). *Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos*. Quito: Registro Oficial.
- Asamblea Nacional del Ecuador. (2010). *Código Orgánico de la Producción, Comercio e Inversiones (COPCI)*. Quito: Registro Oficial.

- Asamblea Nacional del Ecuador. (2015). *Ley Orgánica de Telecomunicaciones*. Quito: Registro Oficial.
- Cortés Rojas, H. F. (2024). Uso de la norma ISO 25010 para establecer requerimientos de calidad en el diseño de un prototipo tecnológico educativo basado en realidad aumentada para la enseñanza de programación básica en estudiantes de educación media. *Arandu UTIC*, 11(2), 2603-2621. <https://doi.org/10.69639/arandu.v11i2.452>
- Dini, M., Gligo, N., & Patiño, A. (2021). *Transformación digital de las mipymes: elementos para el diseño de políticas*. Comisión Económica para América Latina y el Caribe (CEPAL). CEPAL. <https://www.cepal.org/es/publicaciones/47183-transformacion-digital-mipymes-elementos-diseno-politicas>
- Ecuador, A. N. (2008). *Constitución de la República del Ecuador*. Quito: Asamblea Nacional del Ecuador.
- Ecuador, A. N. (2016). *Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación*. Quito: Registro Oficial.
- Ecuador, A. N. (2020). *Ley Orgánica de Emprendimiento e Innovación*. Quito: Registro Oficial.
- Flutter. (30 de octubre de 2025). *Persist data with SQLite*. Recuperado el 20 de enero de 2026, de Flutter documentation. <https://docs.flutter.dev/cookbook/persistence/sqlite>
- García, J., & Salazar, A. (2023). *Visual Studio Code: guía práctica para el desarrollo de software*. Quito: Editorial Tecnología y Educación.
- ISO. (2018). *ISO 9241-11: Ergonomía de la interacción humano-sistema*. Organización Internacional de Normalización. S.L.: Organización Internacional de Normalización. <https://www.iso.org>

- Khurana, A., Singh, R., & Patel, S. (2022). Building resilient mobile systems for low-connectivity environments. *IEEE Access*, *10*, 77891–77905.
- López, D., & Rodríguez, M. (2023). Diseño de interfaces móviles centradas en el usuario. *Revista de Usabilidad y Experiencia de Usuario*, *5*(2), 41–55.
- Martínez, J. (2024). Avances en la transformación digital de las MiPymes: aprendizajes post-pandemia. *Journal of Technology Management & Innovation*, *19*(1), 52–68.
- Mera Cantos, K., & Zambrano Intriago, M. M. (2024). Gestión de inventarios y su incidencia en la competitividad en microempresas de venta de repuestos de automóviles, Santa Ana – Ecuador. *Reincisol*, *3*(6), 6234–6253. [https://doi.org/10.59282/reincisol.V3\(6\)6234-6253](https://doi.org/10.59282/reincisol.V3(6)6234-6253)
- Michel Madera, M., Torres Nabel, L., & Quevedo Huerta, L. (2012). Estudio de traducción y confiabilidad del instrumento de la Teoría Unificada de la Aceptación y Uso de la Tecnología (UTAUT). *Apertura: Revista de Innovación Educativa*, *4*(2), 96–105.
- Palma Muñoz, K. A., Garzón García, J. J., Delgado Zambrano, J. D., Zambrano Alcívar, K. G., Párraga Zambrano, L. A., & Mendoza Navarrete, M. L. (2021). El impacto de las aplicaciones móviles, orientado a las MIPYMES de la ciudad de Chone. *ULEAM Bahía Magazine*, *1*(1), 31–41. https://revistas.ulead.edu.ec/index.php/uleam_bahia_magazine/article/view/84
- Palos-Sánchez, P., Reyes-Menendez, A., & Saura, J. (2019). Modelos de adopción de tecnologías de la información y cloud computing en las organizaciones. *Información Tecnológica*, *30*(3), 3–12. <https://doi.org/10.4067/S0718-07642019000300003>
- Personales, L. O. (2021). *Ley Orgánica de Protección de Datos Personales*. Quito: Registro Oficial.

- Sánchez, E., & Morales, J. (2020). Arquitecturas de software para aplicaciones móviles. *Revista de Ingeniería de Sistemas*, 33–44. <https://doi.org/10.4067/S0718-33052020000100033>
- Schwaber, K., & Sutherland, J. (2020). *La guía Scrum: la guía definitiva de Scrum*. Washington D.C.: Scrum.org. <https://scrumguides.org/>
- Superintendencia de Protección de Datos Personales. (2025). *Guía de protección de datos personales desde el diseño y por defecto*. Quito, D. M.: Superintendencia de Protección de Datos Personales. <https://spdp.gob.ec/wp-content/uploads/2025/10/40.02-Guia-de-Proteccion-de-Datos-desde-el-Diseno-y-por-Defecto.pdf>
- Vásconez, V., Mayorga, M., Moreno, M., Arellano, A., & Pazmiño, C. (2020). Gestión del sistema de inventarios orientado a pequeñas y medianas empresas, PYMEs, ecuatorianas del sector ferretero: caso de estudio. *Revista ESPACIOS*, 41(3), 7. <https://www.revistaespacios.com/a20v41n03/a20v41n03p07.pdf>
- Verwijns, C., & Russo, D. (2023). An evidence-based theory of Scrum team effectiveness. *Journal of Systems and Software*, 111567.

13. ANEXOS

13.1. Entrevistas

ENTREVISTA ESTRUCTURADA - PROPIETARIO

Nombre del entrevistado: _____

Cargo: Propietario/administrador de la Tienda Génesis

Lugar: Comunidad Los Andes, parroquia Cotundo, cantón Archidona, provincia de Napo

Fecha: ____ / ____ / 2025

Entrevistadora: Sara Pamela Andí Tapuy

Técnica aplicada: Entrevista estructurada

Objetivo de la entrevista: Recopilar información sobre el manejo actual de ventas e inventario en la Tienda Génesis, identificar los principales problemas del sistema manual y conocer las expectativas del propietario frente al desarrollo de una aplicación móvil con base de datos local.

1. ¿Desde cuándo funciona la Tienda Génesis y cuántas personas trabajan actualmente?
2. ¿Cómo se realizan los registros de ventas y control de inventario en la tienda?
3. ¿Qué dificultades o errores suelen presentarse con el método manual actual?
4. ¿Qué herramientas o dispositivos se utilizan para anotar o calcular las ventas?
5. ¿Quiénes colaboran en la tienda y cómo es su participación en el registro diario?
6. ¿Qué tipo de información considera importante registrar en una aplicación?
7. ¿Con qué frecuencia revisa el stock de la tienda?
8. ¿Qué funciones le gustaría que tenga la aplicación móvil?
9. ¿Cuenta con conexión estable a internet?
10. ¿Qué beneficios espera obtener con la aplicación?

ENTREVISTA A LOS EMPLEADOS

Nombre del entrevistado: _____

Cargo: Empleado de la Tienda Génesis

Lugar: Comunidad Los Andes, parroquia Cotundo, cantón Archidona, provincia de Napo

Fecha: ____ / ____ / 2025

Entrevistadora: Sara Pamela Andi Tapuy

Técnica aplicada: Entrevista estructurada

Objetivo de la entrevista: Identificar las actividades que realizan los empleados en la Tienda Génesis, las dificultades que enfrentan en el registro manual de ventas e inventario y su percepción sobre el uso de una aplicación móvil como herramienta para mejorar la organización y eficiencia de sus labores diarias.

1. ¿Cuáles son sus tareas principales dentro de la tienda?
2. ¿Cómo registra actualmente las ventas y qué dificultades encuentra?
3. ¿Con qué frecuencia revisa el inventario y qué problemas se presentan?
4. ¿Qué herramientas utiliza para el registro del inventario?
5. ¿Qué dificultades tiene con el sistema manual que utilizan?
6. ¿Qué funciones considera útiles en una aplicación móvil para la tienda?
7. ¿Cree que una aplicación móvil mejoraría sus actividades diarias? ¿Por qué?

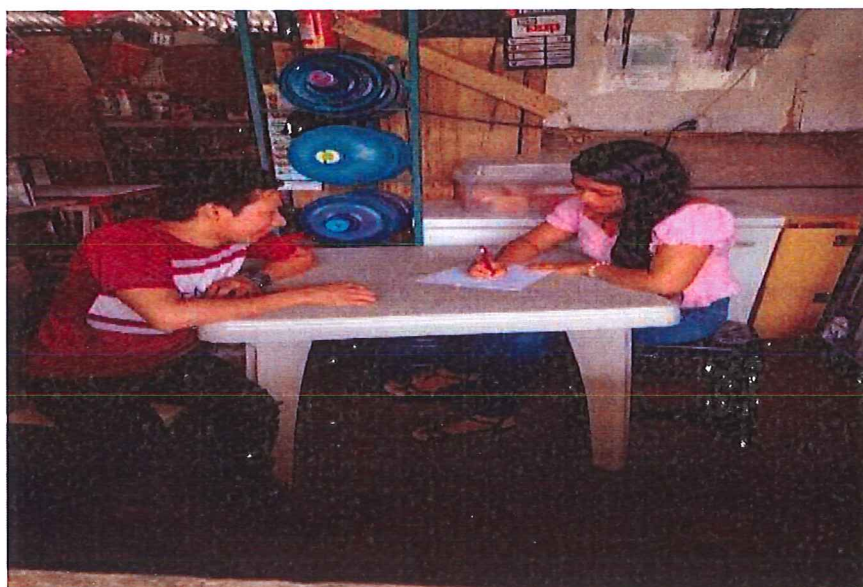
13.2. Fotografías

Ilustración 24. *Socialización de Actividades.*



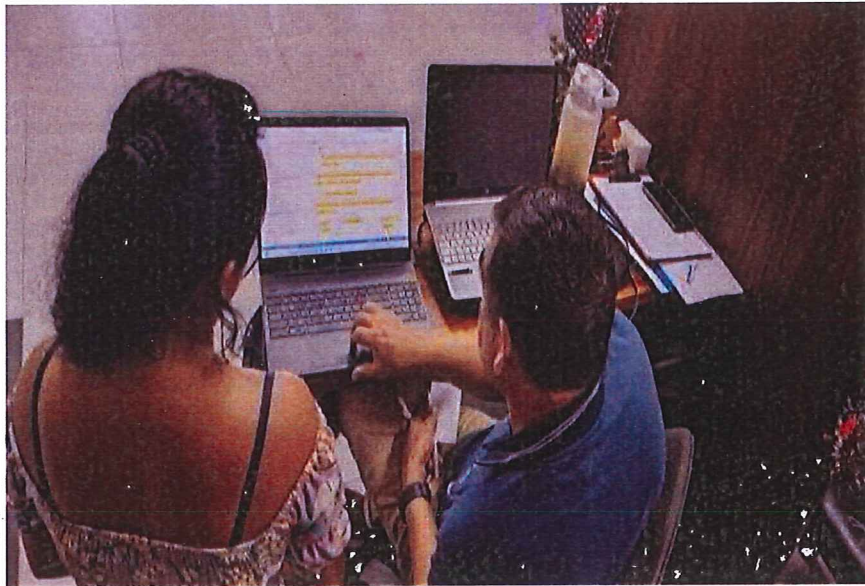
Nota: Socialización e indicaciones proporcionadas por el tutor asesor durante el seguimiento del proyecto. Fuente: Elaboración propia (2025).

Ilustración 25. *Entrevista al Propietario.*



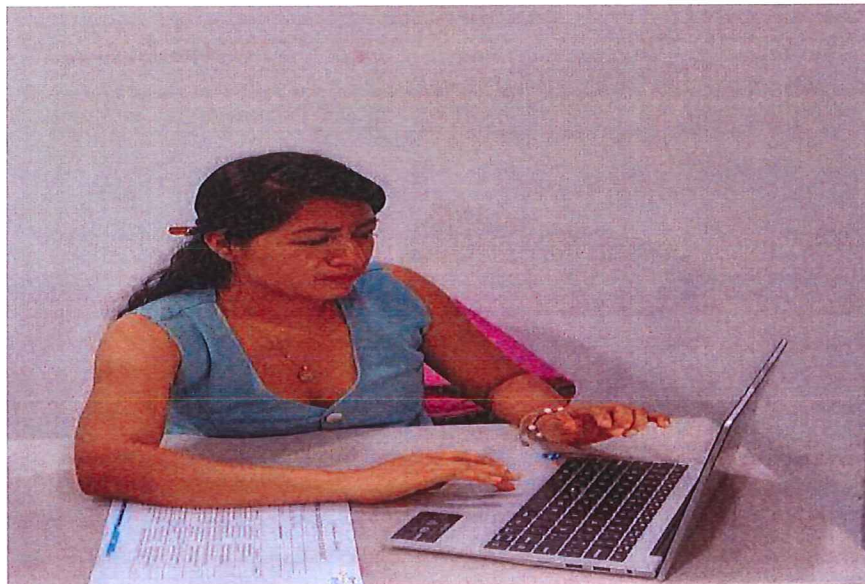
Nota: Entrevista realizada al propietario de la Tienda Génesis para identificar los procesos de ventas e inventario y levantar requerimientos del sistema. Fuente: Elaboración propia (2025).

Ilustración 26. *Revisión del Avance del Proyecto*



Nota: Revisión del avance del proyecto con el docente asesor, en la que se recibieron observaciones y recomendaciones técnicas para la mejora del sistema. Fuente: Elaboración propia (2025).

Ilustración 27. *Elaboración del Proyecto.*



Nota: Desarrollo de la aplicación móvil mediante Flutter, orientado a la implementación de los módulos del sistema y la integración con la base de datos local. Fuente: Elaboración propia (2025).

13.3. Código

13.3.1. Código del Módulo de Ventas en Flutter (Dart)

```
1. import 'dart:io';
2. import 'package:flutter/material.dart';
3. import 'package:intl/intl.dart';
4. import 'package:provider/provider.dart';
5. import 'package:shared_preferences/shared_preferences.dart';
6. // MODELOS
7. import '../models/venta.dart';
8. import '../models/detalle_venta.dart';
9. import '../models/producto.dart';
10. import '../models/cliente.dart';
11. // VIEWMODELS
12. import '../viewmodels/venta_viewmodel.dart';
13. import '../viewmodels/producto_viewmodel.dart';
14. import '../viewmodels/cliente_viewmodel.dart';
15. // WIDGETS
16. import '../widgets/modal_categorias_venta.dart';
17. // PÁGINAS
18. import 'venta_pago_page.dart';
19. import 'venta_resumen_page.dart';
20. import 'crear_cliente_page.dart';
21. import 'qr_scanner_page.dart';
22.
23. class VentasPage extends StatefulWidget {
24.   const VentasPage({super.key});
25.   @override
26.   State<VentasPage> createState() => _VentasPageState();
27. }
28. class _VentasPageState extends State<VentasPage> {
29.   List<DetalleVenta> carrito = [];
30.   Cliente? clienteSeleccionado;
31.   String metodoPago = "Efectivo";
32.   bool modoLibre = false;
33.   String buscador = "";
```

```

34. String categoriaSeleccionada = "Todas";
35. final TextEditingController _montoLibreCtrl =
    TextEditingController();
36. final TextEditingController _descLibreCtrl =
    TextEditingController();
37. // Bloqueo para evitar duplicar ventas por doble click
38. bool _guardandoVenta = false;
39. double get totalCarrito =>
40.     carrito.fold(0.0, (sum, item) => sum + item.subtotal);
41. @override
42. void dispose() {
43.     _montoLibreCtrl.dispose();
44.     _descLibreCtrl.dispose();
45.     super.dispose();
46. }
47. // FILTRAR PRODUCTOS POR CATEGORÍA + BUSCADOR
48. List<Producto> filtrarProductos(List<Producto> productos) {
49.     return productos.where((p) {
50.         final coincideBusqueda =
51.             p.nombre.toLowerCase().contains(buscador.toLowerCase());
52.         final coincideCategoria = categoriaSeleccionada == "Todas"
53.             ? true
54.             : (p.categoria ?? "").trim() ==
                categoriaSeleccionada.trim();
55.         return coincideBusqueda && coincideCategoria;
56.     }).toList();
57. }
58. // AGREGAR PRODUCTO CON CONTROL DE STOCK
59. void _agregarProducto(Producto p) {
60.     if (p.stock <= 0) {
61.         ScaffoldMessenger.of(context).showSnackBar(
62.             SnackBar(content: Text("SIN STOCK: ${p.nombre}")),
63.         );
64.         return;
65.     }
66.     setState(() {

```

```

67. final index = carrito.indexWhere((d) => d.idProducto ==
    p.id);
68.     if (index != -1) {
69.         if (carrito[index].cantidad < p.stock) {
70.             carrito[index].cantidad++;
71.             carrito[index].subtotal = carrito[index].cantidad *
    p.precioVenta;
72.         }
73.     } else {
74.         carrito.add(
75.             DetalleVenta(
76.                 idVenta: 0,
77.                 idProducto: p.id!,
78.                 cantidad: 1,
79.                 subtotal: p.precioVenta,
80.             ),
81.         );
82.     }
83. });
84. }
85. // CAMBIAR CANTIDAD
86. void _cambiarCantidad(Producto p, int nuevaCantidad) {
87.     if (nuevaCantidad > p.stock) {
88.         ScaffoldMessenger.of(context).showSnackBar(
89.             SnackBar(content: Text("Máximo disponible: ${p.stock}")),
90.         );
91.         return;
92.     }
93.     setState(() {
94.         if (nuevaCantidad <= 0) {
95.             carrito.removeWhere((d) => d.idProducto == p.id);
96.             return;
97.         }
98.         final index = carrito.indexWhere((d) => d.idProducto == p.id);
99.         if (index != -1) {
100.             carrito[index].cantidad = nuevaCantidad;

```

```

101.   carrito[index].subtotal = nuevaCantidad * p.precioVenta;
102.   } else {
103.     carrito.add(
104.       DetalleVenta(
105.         idVenta: 0,
106.         idProducto: p.id!,
107.         cantidad: nuevaCantidad,
108.         subtotal: nuevaCantidad * p.precioVenta,
109.       ),
110.     );
111.   }
112. });
113. }
114. // ESCANEAR QR
115. Future<void> _scanProducto() async {
116.   final codigo = await Navigator.push(
117.     context,
118.     MaterialPageRoute(builder: (_) => const QRScannerPage()),
119.   );
120.   if (!mounted || codigo == null) return;
121.   final productos =
122.     Provider.of<ProductoViewModel>(context, listen:
123.       false).productos;
124.   final encontrado = productos.firstWhere(
125.     (p) => p.codigo == codigo || p.qrData == codigo,
126.     orElse: () => Producto(
127.       id: -1,
128.       nombre: "",
129.       precioCompra: 0,
130.       precioVenta: 0,
131.       stock: 0,
132.       stockMinimo: 0,
133.     ),
134.   );
135.   ScaffoldMessenger.of(context).showSnackBar(

```

```

136.   const SnackBar(content: Text("Producto no encontrado")),
137.     );
138.     return;
139.   }
140.   _agregarProducto(encontrado);
141. }
142. // SELECCIONAR CLIENTE
143. Future<void> _seleccionarCliente() async {
144.   final vm = Provider.of<ClienteViewModel>(context, listen:
145.     false);
146.     await vm.cargarClientes();
147.     if (!mounted) return;
148.     final seleccionado = await showModalBottomSheet<Cliente?>(
149.       context: context,
150.       isScrollControlled: true,
151.       builder: (_) => _modalClientes(vm.lista),
152.     );
153.     if (!mounted) return;
154.     setState(() => clienteSeleccionado = seleccionado);
155. }
156. Widget _modalClientes(List<Cliente> lista) {
157.   List<Cliente> filtrados = List.from(lista);
158.   final TextEditingController searchCtrl =
159.     TextEditingController();
160.   return Padding(
161.     padding: const EdgeInsets.all(20),
162.     child: SizedBox(
163.       height: 600,
164.       child: StatefulBuilder(
165.         builder: (context, setSheetState) {
166.           return Column(
167.             children: [
168.               const Text(
169.                 "Seleccionar Cliente",

```

```

170.         const SizedBox(height: 10),
171.         TextField(
172.           controller: searchCtrl,
173.           decoration: const InputDecoration(
174.             prefixIcon: Icon(Icons.search),
175.             hintText: "Buscar cliente...",
176.           ),
177.           onChanged: (txt) {
178.             txt = txt.toLowerCase();
179.             setSheetState(() {
180.               filtrados = lista.where((c) {
181.                 return c.nombre.toLowerCase().contains(txt) ||
182.                   c.apellido.toLowerCase().contains(txt) ||
183.                   c.cedula.contains(txt);
184.               }).toList();
185.             });
186.           },
187.         ),
188.         const SizedBox(height: 10),
189.         Expanded(
190.           child: ListView.builder(
191.             itemCount: filtrados.length,
192.             itemBuilder: (_, i) {
193.               final c = filtrados[i];
194.               return ListTile(
195.                 title: Text("${c.nombre} ${c.apellido}"),
196.                 subtitle: Text(c.cedula),
197.                 onTap: () => Navigator.pop(context, c),
198.               );
199.             },
200.           ),
201.         ),
202.         ElevatedButton(
203.           onPressed: () async {
204.             Navigator.pop(context);
205.             final nuevo = await Navigator.push(

```

```

206.         context,
207.         MaterialPageRoute(
208.           builder: (_) => const CrearClientePage(),
209.         ),
210.       );
211.       if (!mounted) return;
212.       if (nuevo is Cliente) {
213.         setState(() => clienteSeleccionado = nuevo);
214.       }
215.     },
216.     child: const Text("Agregar Cliente"),
217.   ),
218.   TextButton(
219.     onPressed: () => Navigator.pop(context, null),
220.     child: const Text("Consumidor Final"),
221.   ),
222. ],
223. );
224. },
225. ),
226. ),
227. );
228. }
229. // REGISTRAR VENTA LIBRE
230. Future<void> _registrarVentaLibre() async {
231.   if (_guardandoVenta) return;
232.   final monto = double.tryParse(_montoLibreCtrl.text) ?? 0;
233.   final desc = _descLibreCtrl.text.trim();
234.   if (monto <= 0 || desc.isEmpty) {
235.     ScaffoldMessenger.of(context).showSnackBar(
236.       const SnackBar(content: Text("Monto o descripción
237.         inválidos")),
238.     );
239.     return;
240.   }
240.   setState(() => _guardandoVenta = true);

```

```

241. try {
242.     // PAGO
243.     final pago = await Navigator.push(
244.         context,
245.         MaterialPageRoute(
246.             builder: (_) =>
247.             VentaPagoPage(total: monto, metodoInicial: metodoPago),
248.         ),
249.     );
250.     if (!mounted || pago == null) {
251.         setState(() => _guardandoVenta = false);
252.         return;
253.     }
254.     metodoPago = pago["metodo"];
255.     // USUARIO
256.     final prefs = await SharedPreferences.getInstance();
257.     if (!mounted) return;
258.     final idUsuario = prefs.getInt("id_usuario") ?? 1;
259.     // VENTA
260.     final venta = Venta(
261.         fecha: DateFormat('yyyy-MM-dd
262.             HH:mm:ss').format(DateTime.now()),
263.         total: monto,
264.         idUsuario: idUsuario,
265.         idCliente: clienteSeleccionado?.id, // NULL si consumidor
266.         final
267.     );
268.     final ventaVM = Provider.of<VentaViewModel>(context,
269.         listen: false);
270.     final idVenta = await ventaVM.registrarVenta(
271.         venta,
272.         const [],
273.         const [],
274.     );
275.     venta.id = idVenta;
276.     // CLIENTE

```

```

274.     Cliente? clienteData;
275.     if (venta.idCliente != null) {
276.         clienteData = await
ventaVM.obtenerClientePorId(venta.idCliente!);
277.     }
278.     if (!mounted) return;
279.     //     RESUMEN
280.     await Navigator.push(
281.         context,
282.         MaterialPageRoute(
283.             builder: (_) => VentaResumenPage(
284.                 venta: venta,
285.                 metodoPago: metodoPago,
286.                 cliente: clienteData,
287.                 descripcion: desc,
288.                 detalles: const [],
289.                 productos: const [],
290.             ),
291.         ),
292.     );
293. } catch (e) {
294.     if (!mounted) return;
295.     ScaffoldMessenger.of(context).showSnackBar(
296.         SnackBar(
297.             content: Text("Error al registrar venta: $e"),
298.             backgroundColor: Colors.red,
299.         ),
300.     );
301. } finally {
302.     if (mounted) {
303.         setState(() => _guardandoVenta = false);
304.     }
305. }
306. }
307. // REGISTRAR VENTA CON PRODUCTOS
308. Future<void> _registrarVentaProductos() async {

```

```

309.   if (_guardandoVenta) return;
310.   if (carrito.isEmpty) {
311.     ScaffoldMessenger.of(context).showSnackBar(
312.       const SnackBar(content: Text("No hay productos en el
313.         carrito")),
314.     );
315.     return;
316.   }
317.   final productosVM =
318.     Provider.of<ProductoViewModel>(context, listen: false);
319.   // Validar stock ANTES
320.   for (var item in carrito) {
321.     final prod =
322.       productosVM.productos.firstWhere((p) => p.id ==
323.         item.idProducto);
324.     if (item.cantidad > prod.stock) {
325.       ScaffoldMessenger.of(context).showSnackBar(
326.         SnackBar(content: Text("Stock insuficiente para
327.           ${prod.nombre}")),
328.       );
329.       return;
330.     }
331.   }
332.   setState(() => _guardandoVenta = true);
333.   try {
334.     // PAGO
335.     final pago = await Navigator.push(
336.       context,
337.       MaterialPageRoute(
338.         builder: (_) => VentaPagoPage(
339.           total: totalCarrito,
340.           metodoInicial: metodoPago,
341.         ),
342.     );
343.   } catch (e) {
344.     ScaffoldMessenger.of(context).showSnackBar(
345.       SnackBar(content: Text("Error al pagar: $e"))
346.     );
347.   }
348.   if (!mounted || pago == null) {
349.     ScaffoldMessenger.of(context).showSnackBar(
350.       SnackBar(content: Text("Error al pagar: $e"))
351.     );
352.   }
353. }

```

```

342.     setState(() => _guardandoVenta = false);
343.     return;
344. }
345.     metodoPago = pago["metodo"];
346.     // USUARIO
347.     final prefs = await SharedPreferences.getInstance();
348. //     FIX DEL WARNING
349. if (!mounted) return;
350. final idUsuario = prefs.getInt("id_usuario") ?? 1;
351.     // VENTA
352. final venta = Venta(
353.     fecha: DateFormat('yyyy-MM-dd
HH:mm:ss').format(DateTime.now()),
354.     total: totalCarrito,
355.     idUsuario: idUsuario,
356.     idCliente: clienteSeleccionado?.id, // NULL si consumidor
    final );
357. final ventaVM = Provider.of<VentaViewModel>(context, listen:
false);
358.     final detalles = carrito.map((d) {
359.         return DetalleVenta(
360.             idVenta: 0,
361.             idProducto: d.idProducto,
362.             cantidad: d.cantidad,
363.             subtotal: d.subtotal,
364.         );
365.     }).toList();
366.     final idVenta = await ventaVM.registrarVenta(
367.         venta,
368.         detalles,
369.         productosVM.productos,
370.     );
371.     venta.id = idVenta;
372.     // CLIENTE
373.     Cliente? clienteData;
374.     if (venta.idCliente != null) {

```

```

375.   clienteData = await
      ventaVM.obtenerClientePorId(venta.idCliente!);
376. }
377.   // LIMPIAR
378.   setState(() {
379.     carrito.clear();
380.     clienteSeleccionado = null;
381.   });
382.   // NAVEGAR
383.   if (!mounted) return;
384.   await Navigator.push(
385.     context,
386.     MaterialPageRoute(
387.       builder: (_) => VentaResumenPage(
388.         venta: venta,
389.         metodoPago: metodoPago,
390.         cliente: clienteData,
391.         descripcion: "Venta con productos",
392.         detalles: detalles,
393.         productos: productosVM.productos,
394.       ),
395.     ),
396.   );
397.   } catch (e) {
398.     if (!mounted) return;
399.
400.     ScaffoldMessenger.of(context).showSnackBar(
401.       SnackBar(
402.         content: Text("Error al registrar venta: $e"),
403.         backgroundColor: Colors.red,
404.       ),
405.     );
406.   }
407.   finally {
408.     if (mounted) {
409.       setState(() => _guardandoVenta = false);

```

```

410.     }
411.   }
412. }
413. // UI PRINCIPAL
414. @override
415. Widget build(BuildContext context) {
416.   final productos =
417.     Provider.of<ProductoViewModel>(context).productos;
418.   final filtrados = filtrarProductos(productos);
419.   return Scaffold(
420.     appBar: AppBar(
421.       title: const Text("Registrar Venta"),
422.       backgroundColor: Colors.green,
423.       actions: [
424.         IconButton(
425.           icon: const Icon(Icons.qr_code_scanner),
426.           onPressed: _scanProducto,
427.         ),
428.       ],
429.     body: SafeArea(
430.       child: SingleChildScrollView(
431.         padding: const EdgeInsets.only(bottom: 120),
432.         child: Column(
433.           children: [
434.             const SizedBox(height: 10),
435.             Row(
436.               children: [
437.                 Expanded(
438.                   child: ChoiceChip(
439.                     label: const Text("Con productos"),
440.                     selected: !modoLibre,
441.                     onPressed: (_) => setState(() => modoLibre = false),
442.                   ),
443.                 const SizedBox(width: 10),

```

```

445.         Expanded(
446.             child: ChoiceChip(
447.                 label: const Text("Venta libre"),
448.                 selected: modoLibre,
449.                 onSelect: (_) => setState(() => modoLibre = true),
450.             ),
451.         ),
452.     ],
453. ),
454. ListTile(
455.     title: Text(
456.         clienteSeleccionado == null
457.         ? "Consumidor Final"
458.         : "${clienteSeleccionado!.nombre}
    ${clienteSeleccionado!.apellido}",
459.     style: const TextStyle(fontWeight: FontWeight.bold),
460.     ),
461.     trailing: const Icon(Icons.person),
462.     onTap: _seleccionarCliente,
463.     ),
464.     if (!modoLibre) _selectorCategorias(),
465. modoLibre ? _uiVentaLibre() : _uiVentaProductos(filtrados),
466. ],
467. ),
468. ),
469. ),
470. bottomNavigationBar: Container(
471.     padding: const EdgeInsets.all(16),
472.     color: Colors.green.shade50,
473.     child: Column(
474.         mainAxisAlignment: MainAxisAlignment.min,
475.         children: [
476.             Row(
477.                 children: [
478.                     const Text("Total:",
479.

```

```

480.     TextStyle(fontSize: 18, fontWeight: FontWeight.bold)),
481.         const SizedBox(width: 10),
482.         Text(
483.             modoLibre
484.             ? "\${(_montoLibreCtrl.text.isEmpty ? 0 :
double.tryParse(_montoLibreCtrl.text) ?? 0).toStringAsFixed(2)}"
485.             : "\${totalCarrito.toStringAsFixed(2)}",
486.             style: const TextStyle(
487.                 fontSize: 22,
488.                 fontWeight: FontWeight.bold,
489.                 color: Colors.green,
490.             ),
491.         ),
492.     ],
493. ),
494.     const SizedBox(height: 10),
495.     SizedBox(
496.         width: double.infinity,
497.         child: ElevatedButton.icon(
498.             icon: _guardandoVenta
499.             ? const SizedBox(
500.                 width: 18,
501.                 height: 18,
502.                 child: CircularProgressIndicator(strokeWidth: 2),
503.             )
504.             : const Icon(Icons.point_of_sale),
505.             label: Text(_guardandoVenta
506.                 ? "Guardando..."
507.                 : "Registrar Venta"),
508.             onPressed: _guardandoVenta
509.                 ? null
510.             : (modoLibre ? _registrarVentaLibre : _registrarVentaProductos),
511.         ),
512.     ),
513. ],
514. ),

```

```

515.     ),
516.   );
517. }
518. // SELECTOR DE CATEGORÍAS
519. Widget _selectorCategorias() {
520.   return Padding(
521.     padding: const EdgeInsets.symmetric(horizontal: 16,
522.       vertical: 8),
523.     child: GestureDetector(
524.       onTap: () async {
525.         final seleccion = await showModalBottomSheet<String>(
526.           context: context,
527.           isScrollControlled: true,
528.           builder: (_) => const ModalCategoriasVenta(),
529.         );
530.         if (!mounted) return;
531.         if (seleccion != null) {
532.           setState(() => categoriaSeleccionada = seleccion);
533.         }
534.       },
535.       child: Container(
536.         padding: const EdgeInsets.symmetric(horizontal: 16, vertical:
537.           14),
538.         decoration: BoxDecoration(
539.           border: Border.all(color: Colors.grey.shade400),
540.           borderRadius: BorderRadius.circular(10),
541.         ),
542.         child: Row(
543.           mainAxisAlignment: MainAxisAlignment.spaceBetween,
544.           children: [
545.             Text(categoriaSeleccionada,
546.               style: const TextStyle(fontSize: 16)),
547.             const Icon(Icons.keyboard_arrow_down),
548.           ],
549.         ),
550.       ),
551.     ),
552.   );
553. }

```

```

549.     ),
550.   ),
551. );
552. }
553. // UI DE VENTA LIBRE
554. Widget _uiVentaLibre() {
555.   return Padding(
556.     padding: const EdgeInsets.all(20),
557.     child: Column(
558.       children: [
559.         TextField(
560.           controller: _montoLibreCtrl,
561.           keyboardType: TextInputType.number,
562.           decoration: const InputDecoration(labelText: "Monto"),
563.           onChanged: (_) => setState(() {}),
564.         ),
565.         const SizedBox(height: 10),
566.         TextField(
567.           controller: _descLibreCtrl,
568.           decoration: const InputDecoration(labelText: "Descripción"),
569.         ),
570.         const SizedBox(height: 20),
571.       ],
572.     ),
573.   );
574. }
575. // UI PRODUCTOS
576. Widget _uiVentaProductos(List<Producto> filtrados) {
577.   return Column(
578.     children: [
579.       Padding(
580.         padding: const EdgeInsets.all(10),
581.         child: TextField(
582.           decoration: const InputDecoration(
583.             hintText: "Buscar producto...",
584.             prefixIcon: Icon(Icons.search),

```

```

585.         ),
586.         onChanged: (v) => setState(() => buscador = v),
587.     ),
588. ),
589. ListView.builder(
590.     shrinkWrap: true,
591.     physics: const NeverScrollableScrollPhysics(),
592.     itemCount: filtrados.length,
593.     itemBuilder: (_, i) {
594.         final p = filtrados[i];
595.
596.         final cantidad = carrito
597.             .where((d) => d.idProducto == p.id)
598.             .fold(0, (a, b) => a + b.cantidad);
599.
600.         final ctrlCantidad =
601.             TextEditingController(text: cantidad.toString());
602.         return Card(
603.             child: ListTile(
604.                 leading: p.imagenPath != null &&
605.                     p.imagenPath!.isNotEmpty &&
606.                     File(p.imagenPath!).existsSync()
607.                 ? Image.file(File(p.imagenPath!), width: 55, height: 55)
608.                 : const Icon(Icons.image_not_supported),
609.                 title: Text(p.nombre),
610.                 subtitle: Column(
611.                     crossAxisAlignment: CrossAxisAlignment.start,
612.                     children: [
613.                         Text("Stock: ${p.stock}"),
614.                         Text("\${p.precioVenta}"),
615.                         if (p.stock <= 0)
616.                             const Text(
617.                                 "SIN STOCK DISPONIBLE",
618.                                 style: TextStyle(
619.                                     color: Colors.red, fontWeight: FontWeight.bold),
620.                             ),

```

```

621.         ],
622.     ),
623.     trailing: SizedBox(
624.         width: 140,
625.         child: Row(
626.             children: [
627.                 IconButton(
628.                     icon: const Icon(Icons.remove_circle_outline),
629.                     onPressed: cantidad > 0
630.                     ? () => _cambiarCantidad(p, cantidad - 1)
631.                     : null,
632.                 ),
633.                 Expanded(
634.                     child: TextField(
635.                         enabled: p.stock > 0,
636.                         controller: ctrlCantidad,
637.                         textAlign: TextAlign.center,
638.                         keyboardType: TextInputType.number,
639.                         onSubmitted: (text) {
640.                             final val = int.tryParse(text) ?? 0;
641.                             if (val > p.stock) {
642.                                 ScaffoldMessenger.of(context).showSnackBar(
643.                                     SnackBar(
644.                                         content:
645.                                             Text("Stock insuficiente. Máx: ${p.stock}"),
646.                                     ),
647.                                 );
648.                                 return;
649.                             }
650.                             _cambiarCantidad(p, val);
651.                         },
652.                     ),
653.                 ),
654.                 IconButton(
655.                     icon: const Icon(Icons.add_circle_outline),
656.                     onPressed: p.stock > 0 && cantidad < p.stock

```

```
657.         ? () => _cambiarCantidad(p, cantidad + 1)
658.         : null,
659.     ),
660.     ],
661.     ),
662.     ),
663.     ),
664.     );
665. }
666. ),
667. ],
668. );
669. }
670. }
```

CONSTANCIA DE IMPLEMENTACIÓN Y USO DEL SISTEMA

Cotundo, 20/01/2026

Yo, **JHONNY BELISARIO HUATATOCA ALVARADO**, con C.I. N.º **1501176109**, en calidad de **Propietario/Administrador** de la **Tienda Génesis**, ubicada en Cotundo-Comunidad Los Andes, certifico que:

La aplicación móvil denominada "**Sistema de Ventas e Inventario – Tienda Génesis**", desarrollada por la estudiante **Sara Pamela Andi Tapuy**, se encuentra implementada y en uso activo en el establecimiento desde el día **25/01/2026**.

Se confirma que el sistema se utiliza para:

- Registro de ventas,
- Control de inventario y productos,
- Consulta de reportes y exportación de reportes en PDF,
- Control de acceso por roles (Administrador y Empleado), conforme a las necesidades del negocio.

La presente constancia se emite a solicitud de la interesada para fines académicos y de sustentación del proyecto de titulación.

Atentamente,



JHONNY BELISARIO HUATATOCA ALVARADO
Propietario/Administrador – Tienda Génesis
C.I.: 1501176109
Teléfono: 0983613075