Guía Informe final de proyecto de Desarrollo de SIAs

# Introducción

El sistema propuesto busca optimizar la gestión de inventario para una tienda minorista, brindando a los usuarios un control en tiempo real de la disponibilidad de productos, reduciendo pérdidas y mejorando la satisfacción tanto de los vendedores como de los clientes. Con esta solución se espera generar un impacto positivo en la eficiencia y en el crecimiento de la tienda minorista.

# Etapa 1: Visión y Alcance del Proyecto

1. Visión

"Ser el sistema de inventario más rápido y eficiente del mercado, permitiendo optimizar sus flujos de trabajo, reducir tiempos y disminuir pérdidas".

1.2 Problema y contexto

La tienda actualmente gestiona su inventario de forma manual, lo que genera varios impactos negativos:

* Desconocimiento del stock real, dificultando la planificación y la toma de decisiones.
* Pérdida de ventas debido a la falta de productos disponibles.
* Mayor tiempo y esfuerzo en la gestión y actualización de registros de inventario.
* Mayor riesgo de errores operativos y desajustes en el stock.

Estos problemas evidencian la necesidad de implementar un sistema automatizado de gestión de inventarios que permita mejorar la eficiencia, reducir pérdidas y garantizar la disponibilidad de productos para los clientes.

1.3 Usuarios y stakeholders

* **Dueño del almacén**

*Dolencias:* No cuenta con un control preciso y actualizado del inventario, lo que provoca pérdidas por exceso de stock o quiebres de productos. Esto le dificulta planificar compras adecuadamente y anticiparse a la demanda.  
*Objetivos:* Contar con información clara, en tiempo real y confiable del inventario; anticiparse a los quiebres de stock; optimizar la organización de las compras; reducir pérdidas y maximizar la rentabilidad general del negocio.

* **Vendedor de tienda**

*Dolencias:* Tiene dificultad para conocer con precisión qué productos están disponibles o en qué cantidad. Esto genera pérdida de tiempo al buscar o consultar stock y puede provocar errores en el registro de ventas o reposición.

*Objetivos:* Verificar rápidamente la disponibilidad de productos; registrar ingresos y salidas de manera ágil y sin equivocaciones; reducir el tiempo de gestión para enfocarse en la atención al cliente..

* **Clientes**

*Dolencias:* Enfrentan largas esperas en caja o durante la atención, y a menudo no encuentran los productos que buscan por falta de stock o desorganización del local. Esto reduce su satisfacción y confianza en el almacén.

*Objetivos:* Encontrar siempre los productos disponibles; realizar sus compras de forma rápida, cómoda y eficiente; recibir una atención fluida que mejore su experiencia de compra y los motive a regresar.

* **Proveedor**

*Dolencias:* Recibe pedidos incompletos, desordenados o fuera de plazo; existe poca claridad sobre las necesidades reales del almacén y problemas de comunicación que dificultan la planificación de abastecimiento.

*Objetivos:* Recibir pedidos claros, completos y oportunos; mantener una comunicación constante y eficiente con el almacén; fortalecer la relación comercial mediante una gestión coordinada y predecible del inventario.

1.4 Propuesta de valor

La solución propuesta permitirá a la tienda gestionar su inventario de manera efectiva en aspectos como:

* Gestión efectiva del stock en tiempo real.
* Disminución de pérdidas por falta de productos.
* Ahorro de tiempo en la gestión del inventario.
* Aumento en la satisfacción del cliente debido a la disponibilidad de artículos.
* Planificación de compras más eficiente.

Con todo lo anterior se mejorará tanto los procesos internos como los resultados comerciales de la tienda.

# Etapa 2: Organización y Planificación del Proyecto

* 1. Definición de Roles y Responsabilidades.

1. Cliente/usuario final- Yahima Hadfeg, Valentina Duran, Valentina Fuentes, Ginna Cespedes, Maria Quezada:

* **Rol:** Persona que realiza las compras
* **Responsabilidades:**
  + Identificar los requerimientos del sistema.
  + Determinar las metas del negocio.
  + Revisar y aprobar los resultados del proyecto.

1. Jefe de Proyecto/Proyect Manager - Valentina Duran:

* **Rol:** Encargada de la gestión integral del proyecto
* **Responsabilidades:**
  + Organizar las actividades y etapas del proyecto.
  + Dirigir al grupo de trabajo.
  + Supervisar el calendario, los posibles problemas y la calidad del proyecto.
  + Actuar como intermediario entre el cliente y el equipo técnico.

1. Arquitecto del Software/Diseñador - Valentina Fuentes:

* **Rol:** Encargada de las decisiones técnicas de alto nivel
* **Responsabilidades:**
  + Crear el diseño del sistema.
  + Establecer el formato de la base de datos.
  + Elegir las herramientas, modelos de diseño y conexiones.
  + Garantizar la uniformidad técnica en todo el proyecto

1. Equipo de desarrollo (Programadores) - María Quezada:

* **Rol:** Desarrolladora principal del sistema
* **Responsabilidades:**
  + Estructurar el sistema de acuerdo con los planos y especificaciones.
  + Ejecutar las funciones esenciales.
  + Llevar a cabo pruebas unitarias del código.
  + solucionar problemas encontrados a lo largo del desarrollo.

1. Administrador de sistemas/ Infraestructura – Ginna Cespedes:

* **Rol:** Responsable de la infraestructura tecnológica
* **Responsabilidades:**
  + Establecer el entorno local ya sea servidores, base de datos MySQL.
  + Asegurar la red, la protección y la operatividad del sistema.
  + Hacer copias de seguridad y restaurar datos.
  + Supervisar el funcionamiento de los recursos del sistema.

1. Encargado de documentación – Ginna Cespedes:

* **Rol:** Responsable de la documentación del proyecto
* **Responsabilidades:**
  + Asegurar que el equipo tenga documentación actualizada y clara
  + Escribir y estructurar los documentos del sistema.

2.2 Planificación de Tareas (Carta Gantt/ Kanban.).

La planificación del proyecto se representará a través de una carta Gantt, la cual permitirá visualizar de manera clara las actividades, duración y la relación entre ellas. Este recurso facilita la organización del trabajo en equipo, los plazos establecidos y las responsabilidades. De este modo, se garantiza un seguimiento eficiente del progreso y una adecuada gestión del tiempo, contribuyendo al cumplimiento de los objetivos definidos.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tareas** | **Agosto** | | | | **Septiembre** | | | | **Octubre** | | | | **Noviembre** | | | |
| **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** |
| Realizar reunión inicial |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Identificar necesidades |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Definir funcionalidades principales |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Identificar usuarios y roles |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Crear modelo identidad -relación |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Crear base de datos |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Diseñar interfaces graficas |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Definir arquitectura MVC |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Implementar conexión a MySQL |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollar módulo de productos |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollar modulo de movimientos de inventario |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo modulo de consultas de stock |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Realizar pruebas unitarias |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Realizar pruebas de integración con MySQL |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Realizar pruebas de seguridad |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Revisar documentación y correcciones |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Entregar y presentar |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

2.3 Herramientas de Organización y Gestión del Proyecto.

Para garantizar una adecuada colaboración entre los miembros del equipo y el cumplimiento de los plazos fijados, se emplearán las siguientes herramientas para ayudar en la gestión del proyecto:

1. **Word compartido**

* ***Finalidad:*** Centralizar la edición y almacenamiento de documentos relacionados con el proyecto.
* **Justificación:** Permite que los miembros del equipo trabajen de manera simultánea en informes, requisitos y documentación, asegurando consistencia, accesibilidad desde distintos dispositivos y resguardo de la información.

1. **GitHub**

* *Finalidad:* Plataforma en la nube para almacenar, gestionar y colaborar en proyectos de código.
* *Justificación:* Facilita el trabajo colaborativo de los programadores mediante control de versiones, lo que permite llevar un historial de cambios y evitar la pérdida de avances.

1. **NetBeans IDE**

* ***Finalidad:*** Entorno de desarrollo integrado para la construcción de la aplicación en Java, incluyendo interfaz gráfica y conexión a la base de datos.
* ***Justificación:*** Su estructura permite organizar de manera clara los paquetes, clases y controladores bajo el modelo MVC, garantizando orden y escalabilidad del software. Además, simplifica la depuración y pruebas iniciales de la aplicación.

1. **MySQL Workbench**

* *Finalidad:* diseño, creación y gestión de la base de datos para el sistema de inventario.
* *Justificación:* facilita la estructuración de tablas, relaciones y pruebas de consultas. Su integración con el entorno de desarrollo asegura que la lógica de datos sea confiable, optimizando el rendimiento del sistema.

2.4 Matriz de riesgos inicial (seguridad, datos, continuidad).

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | ***Riesgo identificado*** | ***Categoría (Seguridad/Datos/Continuidad)*** | ***Probabilidad (Alta/Media/Baja)*** | ***Impacto (Alto/Medio/Bajo)*** | ***Mitigación inicial*** | ***Responsable*** | | --- | --- | --- | --- | --- | --- | | *Fallo en el registro de entradas/ salidas de los productos* | *Datos* | *Media* | *Alto* | Llevar un registro manual o en planilla mientras se arregla el sistema, revisando que las entradas y salidas coincidan*.* | *Programador* | | *Acceso no autorizado al sistema de inventario* | *Seguridad* | *Media* | *Alto* | *Autentificación con usuario/contraseña segura y roles diferenciados* | *Administrador de sistemas* | | *Pérdida de datos por corte de luz o falla técnica* | *Continuidad* | *Bajo* | *Alta* | *Respaldos diarios y UPS para proteger servidores* | *Administrador de sistemas* | | *Manipulación incorrecta de stock por error humano* | *Datos* | *Alta* | *Medio* | *Capacitación a usuarios y registros con logs de cambios* | *Dueño/vendedor* | | *Falla en la conexión a internet que impida usar el sistema (si está en la nube)* | *Continuidad* | *Media* | *Medio* | *Habilitar modo offline temporal o exportar reportes en Excel* | *Jefe de proyecto* | | *Errores en la base de datos (corrupción, mala normalización)* | *Datos* | *Media* | *Alto* | *Revisiones periódicas de integridad, normalización adecuada y respaldos incrementales.* | *Programador* | | *Resistencia al cambio por parte de los usuarios* | *Continuidad* | *Alta* | *Alto* | |  | | --- | |  |   *Plan de gestión del cambio como capacitaciones, apoyo continuo y comunicación clara de beneficios* | *Jefe de proyecto* | | *Uso indebido de credenciales compartidas* | *Seguridad* | *Alta* | *Alto* | *Políticas de credenciales únicas y monitoreo de accesos.* | *Administrador de sistemas* | | *Ataques externos (malware, ransomware)* | *Seguridad* | *Media* | *Alto* | *Copias de seguridad y monitoreo de amenazas.* | *Administrador de sistemas* | |

2.5 Checklist de decisión rápida (producto vs SaaS, nube vs on-premise, etc.).

|  |
| --- |
| 1. ****Producto instalado vs SaaS (Software como servicio)**** **Producto instalado (On-Premise)**  **SaaS (Servicio en la nube)**  **Justificación:**  **El sistema será creado usando Java con NetBeans y MySQL, operando de manera local en la computadora del almacén, sin necesidad de internet o servicios externos.** 2. ****Infraestructura: Nube vs On-Premise**** **On-Premise.**  **Nube (IaaS/PaaS/DBaaS)**  **Justificación:**  Se utilizará un ambiente local (MySQL instalado en la computadora), lo que permitirá evitar gastos en servidores en la nube y hará más sencillo el desarrollo académico. 3. ****Arquitectura: Monolito vs Microservicios/Servicios**** **Monolito modular**  **Microservicios / SOA**  **Justificación:**  El proyecto es académico y de pequeña escala, un monolito facilita el desarrollo y mantenimiento sin la complejidad de microservicios. 4. ****Proceso de desarrollo: Plan-Dirigido vs Ágil/DevOps**** **Plan-dirigido (Cascada / V-Model)**  **Ágil / DevOps (Scrum, Kanban, XP)**  **Justificación:**  permite planificar y controlar cada fase de manera secuencial, asegurando mayor control de calidad, documentación completa y seguimiento estructurado. |

# Etapa 3: Selección del Modelo de Desarrollo

3.1 Descripción del Proyecto

El proyecto implica crear un sistema de gestión de inventario para una tienda minorista (almacén), ayudando a registrar la entrada y salida de los productos, verificar la cantidad de stock en tiempo real y a optimizar la forma en que se gestionan las actividades relacionadas con el control del inventario, logrando disminuir quiebres y perdidas de stock, para así mejorar las ventas, los tiempos de control de inventario, una mayor satisfacción al cliente y una mejor planificación de compras.

3.2 Modelos de Desarrollo Considerados (cascada, incremental, iterativo, ágil)

* **Modelo** **iterativo:** se basa en desarrollar un producto a través de versiones sucesivas. Es decir, se comienza con una versión inicial del producto y a través de varias iteraciones, se va mejorando progresivamente. Cada nueva versión incorpora mejoras, correcciones o nuevas funciones basadas en lo aprendido en la versión anterior. El objetivo principal es mejorar el producto hasta que cumpla con los requisitos esperados.
* **Modelo Incremental:** se centra en construir el sistema de manera gradual, dividiéndolo en partes o módulos funcionales que se entregan y ponen en uso progresivamente. Cada incremento añade nuevas funcionalidades completas al producto, de manera que al final de cada etapa se cuenta con un sistema operativo más robusto que el anterior. El valor clave aquí es que el cliente recibe partes utilizables del software antes de que el proyecto esté terminado en su totalidad.

La diferencia entre estos modelos es que iterativo se centra en refinar continuamente versiones del mismo producto hasta alcanzar el resultado esperado, mientras que el modelo incremental se enfoca en entregar funcionalidades completas por etapas, agregando nuevas piezas al sistema hasta completarlo. En la práctica, ambos enfoques suelen combinarse: se puede avanzar de manera incremental en las funcionalidades, y dentro de cada incremento aplicar ciclos iterativos de mejora.

3.3 Justificación del Modelo Seleccionado (Cómo este modelo se adapta al ciclo de vida del proyecto elegido)

Se eligió la mezcla entre los modelos iterativo e incremental permite aprovechar las ventajas de ambos enfoques y adaptarse mejor a las necesidades del proyecto, ya que por un lado, el modelo incremental asegura que el sistema se vaya construyendo por etapas, entregando funcionalidades completas y utilizables en plazos más cortos, lo que facilita al cliente obtener valor temprano y comprobar que el desarrollo responde a sus requerimientos. Esto reduce la incertidumbre y mejora la percepción de avance del trabajo, mientras que, el modelo iterativo permite que cada uno de esos incrementos se refine en ciclos sucesivos, corrigiendo errores, ajustando detalles y perfeccionando la solución antes de avanzar a nuevas funcionalidades. De esta manera, no solo se asegura la entrega gradual del producto, sino también la mejora continua de la calidad y la capacidad de adaptarse a cambios en los requerimientos o condiciones del entorno.

# Etapa 4: Recolección y Análisis de Requerimientos

En esta fase, deberás identificar y documentar los requisitos del sistema. Estos pueden dividirse en requisitos funcionales (qué debe hacer el sistema) y requisitos no funcionales (cómo debe comportarse el sistema). Identificar y justificar la técnica que ocuparon para identificar los requisitos (Como documento Anexo deben entregar el [DER](about:blank))

4.1 Identificación de la o las técnicas para la recolección de los requerimientos (Mostrar la evidencia).

Para la recolección de requerimientos se optó por utilizar dos técnicas complementarias, entrevista no estructurada y encuesta mediante Google Forms. La entrevista no estructurada se aplicó con el objetivo de obtener información más abierta, detallada y flexible, permitiendo que el cliente exprese sus necesidades y expectativas sin limitaciones rígidas en las preguntas. En complemento, la encuesta vía Google Forms se eligió para recopilar datos de manera rápida y sistemática. La combinación de ambas técnicas nos permitió obtener una visión integral tanto de los requerimientos como de las necesidades del cliente para el sistema de gestión de inventarios de la tienda minorista

4.2 Requisitos Funcionales y No funcionales (listado).

En este apartado se presenta un listado de los requerimientos del sistema, tanto funcionales como no funcionales, donde el primero corresponde al de mayor importancia y el último al de menor relevancia, también, se clasificarán los tipos de requerimientos no funcionales que permiten cumplir los objetivos del usuario asegurando un funcionamiento eficiente y confiable.

* Requisitos funcionales

1. Modulo de usuarios y roles

01 – Registro de usuarios con roles  
02 – Restricción de funcionalidades por rol  
03 – Recuperación o restablecimiento de contraseña

1. Modulo de proveedores

04 – Gestión de proveedores   
05 – Búsquedas parametrizadas de proveedores

1. Modulo de productos

06 – Gestión de productos  
07 – Búsquedas de productos por parámetros  
08 – Importación y exportación de productos

1. Modulo de inventarios (entradas, salidas y ajustes)

09 – Gestión de movimiento de inventario  
10 – Modificación de notas en movimientos   
11 – Actualización automática de stock  
12 – Consulta de movimientos   
13 – Registro de ajustes de inventario

1. Modulo de reportes   
   14 – Generación de reportes de inventario   
   15 – Reportes analíticos y gráficos   
   16 – Historial de operaciones (trasavilidad)
2. Modulo de respaldo y recuperación de datos  
   17 – Gestión de respaldo y recuperación de la base de datos

* Requisitos No Funcionales

01 – Cifrado de contraseñas (seguridad)  
02 – Tiempo de respuesta en búsquedas (Rendimiento/Eficiencia)  
03 – Conexiones simultáneas soportadas (Escalabilidad)  
04 – Escalabilidad del sistema (Escalabilidad)  
05 – Compatibilidad en instalación (Portabilidad/Compatibilidad/interacción)  
06 – Interfaz intuitiva y validaciones (Usabilidad/Interfaz de usuario)  
07 – Limpieza automática de formularios (Usabilidad/ Interfaz de usuario)  
08 – Mensajes claros de éxito y error (Usabilidad/ Interfaz de usuario)  
09 – Arquitectura MVC (Mantenibilidad/Interacción)

# Etapa 5: Diseño del Sistema

En esta fase, debes diseñar la arquitectura general del sistema. Define los componentes principales y sus interacciones, así como la base de datos que soportará al sistema.

* 1. Diagrama de Flujo de Datos.

A continuación, se representa gráficamente la estructura funcional y el intercambio de información entre los distintos procesos del sistema, para así, asegurar un control eficiente sobre las operaciones. Además, contribuye a mantener la integridad y trazabilidad para una administración precisa y confiable del inventario dentro de la tienda minorista.

* Imagen que contiene Gráfico

  El contenido generado por IA puede ser incorrecto.Nivel 0: Sistema de gestion de inventario para “VMG”
* Gráfico, Diagrama

  El contenido generado por IA puede ser incorrecto.Nivel 1: Procesos del sistema de gestion de inventario
* Diagrama

  El contenido generado por IA puede ser incorrecto.Nivel 1: Procesos del sistema de gestion de inventario
* Gráfico, Gráfico en cascada

  El contenido generado por IA puede ser incorrecto.Nivel 2: Gestion de entradas
* Diagrama

  El contenido generado por IA puede ser incorrecto.Nivel 2: Gestion de productos

En el proceso de **gestión de productos**, la entidad “usuario” representa los diferentes roles que pueden interactuar con este módulo dentro del sistema de inventario. El administrador cuenta con acceso total, pudiendo registrar, modificar, buscar y eliminar productos según sea necesario. El bodeguero tiene permisos limitados principalmente a la consulta y actualización de la información relacionada con el stock. Mientras que, el vendedor no posee acceso directo a este módulo, ya que sus funciones se centran en el registro de salidas y la generación de reportes.

5.2 Diagrama de Clases.

El siguiente diagrama clases para el sistema de inventario representa la estructura interna del sistema, mostrando de manera general cómo se organizan las clases, sus atributos, métodos y las relaciones existentes entre ellas, permitiendo comprender la lógica del sistema.

Imagen que contiene Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

5.3 Diseño de la Base de Datos (Modelo Relacional) scripts.

Diagrama

El contenido generado por IA puede ser incorrecto.El modelo relacional del sistema de inventario presenta la estructura de la base de datos, incluyendo las tablas, claves primarias, foráneas y relaciones que permiten gestionar de forma eficiente la información de productos, como las entradas, salidas y proveedores.

5.4 Diseño de la Interfaz de Usuario (imágenes del prototipo en un entorno real).

A continuación, se presenta el prototipo de la interfaz de usuario del sistema de inventario, que muestra las pantallas principales que son:

1. Login

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

1. Inicio

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

1. Proveedores

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

1. Usuario

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

1. Interfaz de usuario gráfica, Aplicación

   El contenido generado por IA puede ser incorrecto.Entradas
2. Interfaz de usuario gráfica, Aplicación

   El contenido generado por IA puede ser incorrecto.Salidas
3. Interfaz de usuario gráfica, Aplicación

   El contenido generado por IA puede ser incorrecto.Ajustes
4. Interfaz de usuario gráfica, Aplicación

   El contenido generado por IA puede ser incorrecto.Reportes
5. Interfaz de usuario gráfica, Aplicación, Word

   El contenido generado por IA puede ser incorrecto.Historial

5.5 Política de acceso y roles.

A continuación, se presenta la tabla de políticas de acceso correspondientes a los roles definidos en el sistema de inventario.

| **Módulo / Función** | **Administrador/Dueño de almacen** | **Vendedor** | **Bodeguero** |
| --- | --- | --- | --- |
| Gestión de usuarios y roles | ✔ | ✖ | ✖ |
| Registrar proveedores | ✔ | ✖ | ✖ |
| Registrar productos | ✔ | ✔ | ✔ |
| Modificar notas (entradas/salidas) | ✔ | ✖ | ✔ |
| Registrar entradas de productos | ✔ | ✖ | ✖ |
| Registrar salidas de productos | ✔ | ✔ | ✔ |
| Ajuste de inventario | ✔ | ✖ | ✔ |
| Generar reportes | ✔ | ✔ | ✖ |
| Ver historial del sistema | ✔ | ✔ | ✖ |
| Respaldo y restauración de la base de datos | ✔ | ✖ | ✖ |
| Configurar parámetros del sistema | ✔ | ✖ | ✖ |

**Descripción de roles:**

* *Administrador / Dueño de Almacén*: tiene control total sobre el sistema; puede registrar, modificar, eliminar y configurar. Supervisa toda la gestión de inventario.
* *Vendedor*: registra ventas (salidas), consulta reportes y el historial, pero no puede modificar notas ni realizar ajustes.
* *Bodeguero:* gestiona físicamente entradas, salidas y ajustes, sin acceso a configuración o reportes estratégicos.