

Tecnicatura: Técnico Superior en Desarrollo Web y Aplicaciones Digitales.

Espacios Curriculares: Proyecto Integrador

Tema: PROYECTO: Planificador de Proyectos Empresariales

Docente: Veronica Nievas

Integrantes:

Alexis Gabriel Baracat Peralta 40107091

Janet Pilar Molina DNI 35109986



Año: 2025

Proyecto:

STOCKPOINT

Desarrollado by MACHINA DEVELOPS

Un sistema diseñado para la gestión de stock para pequeñas empresas o Pymes facilitando y automatizando tareas relacionadas con el stock.



<u>FICHA DEL DOCUMENTO</u>	3
<u>CONTENIDO</u>	4
<u>1 INTRODUCCIÓN</u>	6
<u>1.1 Propósito</u>	6
<u>1.2 Alcance</u>	6
<u>1.3 Herramientas</u>	6
<u>1.4 Personal involucrado</u>	6
<u>1.5 Definiciones, acrónimos y abreviaturas</u>	7
<u>1.6 Referencias</u>	7
<u>1.7 Resumen</u>	7
<u>2 DESCRIPCIÓN GENERAL</u>	7
<u>2.1 Perspectiva del producto</u>	7
<u>2.3 Características de los usuarios</u>	8
<u>2.4 Restricciones</u>	9
<u>3 REQUISITOS ESPECÍFICOS</u>	9
<u>3.2 Product Backlog</u>	17

3.2.1 Product Backlog	17
3.3 Sprints	19
3.3.1 Sprint 1	19
3.3.2 Sprint 2	20

PROYECTO INTEGRADOR

-Especificaciones requisitos de software-

1 Introducción

Este documento es una Especificación de Requisitos Software (ERS) para el Sistema de información para la gestión de procesos y control de inventarios. Esta especificación se ha estructurado basándose en las directrices dadas por el estándar IEEE Práctica Recomendada para Especificaciones de Requisitos Software ANSI/IEEE 830, 1998.

1.1 Propósito

El presente documento tiene como propósito definir las especificaciones funcionales, para el desarrollo de un sistema de información web que permitirá **Control de Stock para pequeños comercios y PYMEs**, destinado a propietarios y empleados de negocios como almacenes, kioscos, depósitos y talleres. El sistema permitirá registrar, consultar y administrar el inventario de forma sencilla y eficiente.

1.2 Alcance

El sistema proporcionará:

- Registro y actualización de productos.
- Control de entradas y salidas de stock.
- Alertas de stock bajo.
- Generación de reportes.
- Gestión básica de usuarios.

Será accesible desde un navegador web, con interfaz amigable y adaptable a distintos niveles de conocimiento técnico

1.4 Personal involucrado

Nombre	Alexis Baracat
Rol	Scrum master
Categoría Profesional	Estudiante
Responsabilidad	
Información de contacto	mail

Nombre	Pilar Molina
Rol	Developers
Categoría Profesional	Estudiante
Responsabilidad	
Información de contacto	mail

1.5 Definiciones, acrónimos y abreviaturas

Nombre	Descripción
Usuario	Administrador, Empleado
SIS-I	Sistema de Control de Stock para pequeños comercios y PYMEs.
ERS	Especificación de Requisitos Software
RF	Requerimiento Funcional
RNF	Requerimiento No Funcional
FTP	Protocolo de Transferencia de Archivos
Moodle	Aula Virtual

1.6 Referencias

Título del Documento	Referencia
Standard IEEE 830 - 1998	IEEE

1.7 Resumen

Este documento describe la Especificación de Requisitos de Software para el sistema **Stock Point**, una aplicación web destinada a pequeños comercios y PYMEs para la gestión de inventarios.

El sistema permitirá registrar productos, controlar entradas y salidas de stock, emitir alertas de bajo inventario y generar reportes.

El proyecto seguirá la metodología ágil Scrum, comenzando con un Sprint 0 para definir alcance, diseño y backlog inicial. El backend se desarrollará en Python, la base de datos será MySQL y el frontend se implementará con HTML, CSS y JavaScript.

Con esta solución se busca optimizar la gestión de inventarios, reducir pérdidas por faltantes o excesos de mercadería y facilitar la toma de decisiones.

2 Descripción general

2.1 Perspectiva del producto

El sistema **Stock Point** será un producto diseñado para operar en entornos web, accesibles desde cualquier dispositivo con conexión a internet y navegador moderno, sin necesidad de que el cliente requiera un software adicional.

Permitirá a los usuarios realizar operaciones de alta, baja y modificación de productos, registrar movimientos de stock y generar reportes, todo desde una interfaz simple y adaptada a distintos niveles de experiencia técnica.

El sistema está integrado a pequeños comercios y PyMes, contemplando distintos roles de uso, con diferentes permisos, pensado como una solución segura, capaz de crecer en cantidad de usuarios y productos.

2.2

Características de los usuarios

Tipo de usuario	Administrador
Formación	Manejo de PC
Actividades	Alta/baja de productos, gestión de usuarios, análisis de reportes, control de stock

Tipo de usuario	Empleado
Formación	Manejo de PC
Actividades	Registro de ventas, control de entradas y salidas de productos, actualización de inventario

2.3 Restricciones

- El sistema funcionará únicamente en navegadores compatibles (Chrome, Firefox, Edge).
- Se requiere conexión a internet para su uso.
- Compatibilidad con pantallas de escritorio y tablet.
- Backend desarrollado en **Python** con frameworks como Flask o FastAPI
- Base de datos: **MySQL**.
- Frontend con **HTML5, CSS3 y JavaScript**.

3 Requisitos específicos

REQUERIMIENTOS FUNCIONALES

RF-01: Registrar nuevos productos con nombre, código SKU, precio y stock mínimo.

RF-02: Modificar y eliminar productos existentes.

RF-03: Registrar movimientos de stock (entradas, salidas, ajustes).

RF-04: Consultar stock actual y filtrar por nombre, código o categoría.

RF-05: Generar alertas de bajo stock.

RF-06: Exportar reportes de stock a PDF o Excel.

RF-07: Gestionar usuarios con roles y permisos.

REQUERIMIENTOS NO FUNCIONALES

RNF-01: Interfaz amigable y responsiva.

RNF-02: Tiempo de respuesta máximo de 2 segundos por operación.

RNF-03: Acceso seguro con autenticación de usuario y contraseñas encriptadas.

RNF-04: Backup automático diario de la base de datos.

RNF-05: Disponibilidad del sistema mínima del 99% mensual

Product Backlog

US01: Como administrador, quiero registrar productos para tenerlos en el inventario.

US02: Como administrador, quiero registrar entradas y salidas de productos para mantener actualizado el stock.

US03: Como administrador, quiero recibir alertas cuando un producto esté bajo el stock mínimo.

US04: Como administrador, quiero generar reportes para analizar el estado del inventario.

US05: Como empleado, quiero poder registrar ventas para que el stock se descuente automáticamente.

US06: Como administrador, quiero crear y gestionar usuarios para definir permisos de acceso.

Sprints.

N° de sprint	00
Sprint Backlog	<p>US01: Registrar productos en el inventario</p> <ol style="list-style-type: none">1. Diseñar formulario de alta de productos.2. Validar campos obligatorios (nombre, SKU, costo, precio, stock mínimo).3. Conectar el formulario con la base de datos.4. Implementar validaciones de duplicados (SKU único).5. Guardar el producto en DB.6. Probar creación y persistencia de productos. <p>US02: Como administrador, quiero registrar entradas y salidas de productos para mantener actualizado el stock.</p> <ol style="list-style-type: none">1. Crear interfaz para registrar movimiento.2. Definir lógica de actualización del stock (suma/resta).3. Validar cantidades ingresadas (no negativas, no mayores a stock disponible en salida).4. Guardar movimiento en DB con fecha y usuario.5. Probar entradas y salidas con diferentes escenarios.

US03: Como administrador, quiero recibir alertas cuando un producto esté bajo el stock mínimo.

1. Definir condición de alerta ($\text{stock_actual} < \text{stock_mínimo}$).
2. Implementar notificación visual o por mensaje en sistema.
3. Configurar alerta automática al consultar inventario o realizar movimientos.
4. Probar con productos en diferentes niveles de stock.

US04: Como administrador, quiero generar reportes para analizar el estado del inventario.

1. Diseñar consultas SQL para obtener datos de inventario.
2. Crear módulo de generación de reportes (tablas, resúmenes).
3. Implementar filtros (por fecha, por producto).
4. Agregar exportación de reportes (PDF/Excel).
5. Probar integridad y actualización de datos en reportes.

US05: Como empleado, quiero poder registrar ventas para que el stock se descuente automáticamente.

1. Crear formulario de ventas.
2. Implementar lógica de descuento automático del stock.
3. Validar existencia suficiente de stock antes de confirmar.
4. Guardar venta en DB.
5. Probar flujo completo de venta y actualización.

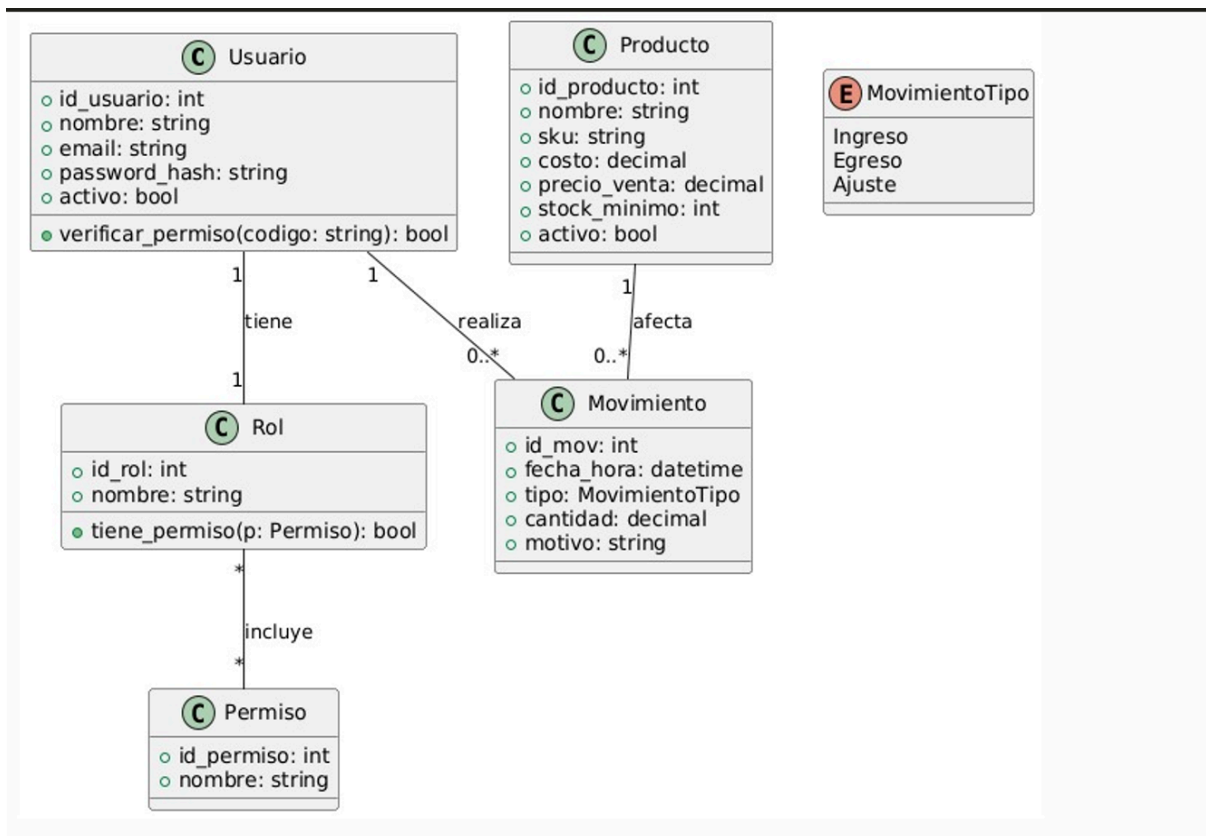
US06: Como administrador, quiero crear y gestionar usuarios para definir permisos de acceso.

1. Diseñar formulario de registro de usuario.
2. Implementar roles (Administrador, Empleado).

	3.Asignar permisos a cada rol. 4.Implementar validaciones de acceso según rol. 5.Probar usuarios con diferentes roles y accesos.
Responsabilidades	
Calendario	
Inconvenientes:	

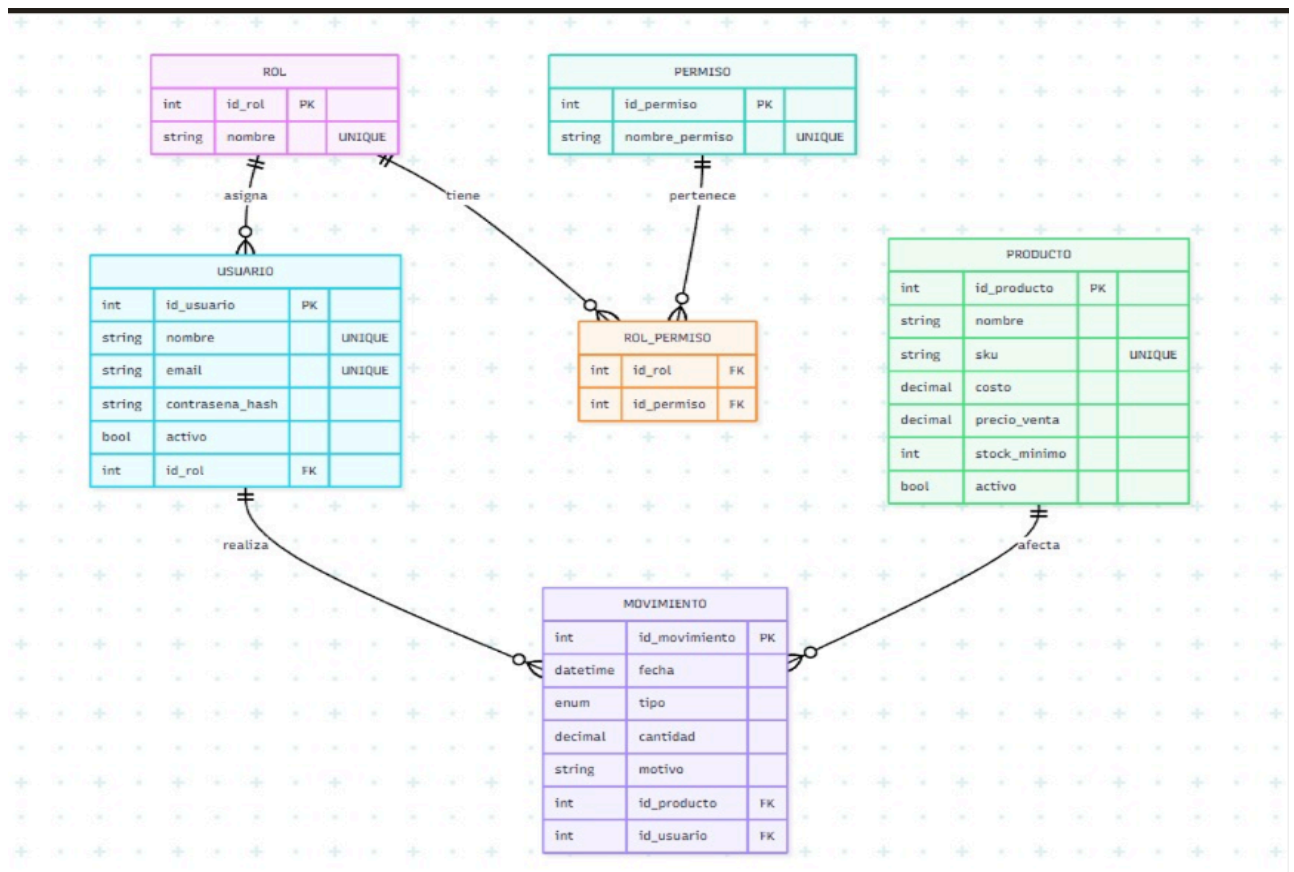
Anexo I

-DIAGRAMA UML



ANEXO II

-Diagrama Entidad Relación-



Entidades y atributos

1) USUARIO

- PK: id_usuario
- Atributos:
 - nombre
 - email (único)
 - contraseña_hash
 - activo (bool)
 - id_rol (FK) → relación con la entidad ROL
- Notas:
 - Un usuario pertenece a un único rol.
 - El rol define los permisos disponibles para el usuario.

2) ROL

- PK: `id_rol`
- Atributos:
 - `nombre` (p. ej., Administrador, Empleado)

3) PERMISO

- PK: `id_permiso`
- Atributos:
 - `nombre_permiso` (p. ej., crear_producto, modificar_producto, borrar_producto)

4) ROL_PERMISO

- PK: combinación de, o bien `id_rol` y `id_permiso` (implícito como clave compuesta)
- Atributos:
 - `id_rol` (FK) → ROL
 - `id_permiso` (FK) → PERMISO
- Notas:
 - Tabla intermedia que modela la relación N:M entre Rol y Permiso.

5) PRODUCTO

- PK: `id_producto`
- Atributos:
 - `nombre`
 - `sku` (único)
 - `costo`
 - `precio_venta`
 - `stock_minimo`
 - `activo` (bool)
- Notas:
 - El stock real no se almacena como atributo estático en PRODUCTO; se calcula a partir de los movimientos (entradas, salidas, ajustes) registrados en MOVIMIENTO.

6) MOVIMIENTO

- PK: `id_movimiento`

- Atributos:
 - fecha (datetime)
 - tipo (entrada, salida, ajuste) – puede mapear a un enumerado
 - cantidad (decimal)
 - motivo
 - id_producto (FK) → PRODUCTO
 - id_usuario (FK) → USUARIO
 - Notas:
 - Cada movimiento pertenece a un producto y a un usuario responsable.
 - El stock actual de un producto se obtiene sumando entradas y restando salidas/ajustes relevantes.
-

Relaciones y cardinalidades

- USUARIO 1:N MOVIMIENTO
 - Un usuario puede realizar múltiples movimientos.
- PRODUCTO 1:N MOVIMIENTO
 - Un producto puede generar múltiples movimientos a lo largo del tiempo.
- ROL 1:N USUARIO
 - Un rol puede estar asignado a varios usuarios.
- ROL 1:N ROL_PERMISO y PERMISO 1:N ROL_PERMISO
 - Un rol puede incluir múltiples permisos y un permiso puede pertenecer a varios roles a través de la tabla intermedia Rol_Permiso.
- Modelo de control de acceso:
 - Las operaciones deben verificar los permisos asociados al rol del usuario antes de permitir acciones (crear/editar productos, registrar movimientos, gestionar usuarios, generar reportes).