

COURSE PROJECT

Universidad Peruana de Ciencias Aplicadas



**Ingeniería de Software - 2025-2
Aplicaciones para Dispositivos Móviles - 12612
Profesor: David Gerardo Quevedo Velasco**

Informe del Trabajo Final

**Startup: Stoq
Producto: StockWise**

Team Members:

Member	Code
Almerco Rojas, Jocelyn Damaly	U20221g068
Sanchez Rios, Camila Cristina	U202210973
Chi Cruzatt, Kevin Jorge	U202313655
Oroncoy Almeyda, Alejandro Daniel	U202313397
Paucar Meneses, Jeremy Alion	U201719449

Abril, 2025

Registro de versiones del Informe

Versión	Fecha	Autor	Descripción de modificaciones
TB1	17/09/2025	<ul style="list-style-type: none">• Jocelyn Almerco• Camila Sanchez• Kevin Chi• Alejandro Oroncoy• Jeremy Paucar	<ul style="list-style-type: none">• Capítulo I: Presentacion• Capítulo II: Requirements Development and Software Solution Design• Avance de Conclusiones, Bibliografía y Anexos

Project Report Collaboration Insights

Link de repositorio del reporte: <https://github.com/upc-pre-202520-1ACC0238-12612-Stoq/Report>

Contenido

Student Outcome

Objetivos SMART

Capítulo I: Presentación

- 1.1. Startup Profile
 - 1.1.1. Descripción de la Startup
 - 1.1.2. Perfiles de integrantes del equipo
- 1.2. Solution Profile
 - 1.2.1 Antecedentes y problemática
 - 1.2.2 Lean UX Process
 - 1.2.2.1. Lean UX Problem Statements
 - 1.2.2.2. Lean UX Assumptions
 - 1.2.2.3. Lean UX Hypothesis Statements
 - 1.2.2.4. Lean UX Canvas
- 1.3. Segmentos objetivo

Capítulo II: Requirements Development and Software Solution Design

- COURSE PROJECT
- Project Report Collaboration Insights
- Contenido
- Student Outcome
- Objetivos SMART
- Capítulo I: Presentación
 - 1.1. Startup Profile
 - 1.1.1. Descripción de la Startup
 - 1.1.2. Perfiles de integrantes del equipo
 - 1.2. Solution Profile
 - 1.2.1 Antecedentes y problemática
 - 1.2.2 Lean UX Process
 - 1.2.2.1. Lean UX Problem Statements
 - 1.2.2.2. Lean UX Assumptions
 - 1.2.2.3. Lean UX Hypothesis Statements
 - 1.2.2.4. Lean UX Canvas
- Capítulo II: Requirements Development and Software Solution Design
 - 2.1. Competidores
 - 2.1.1. Análisis competitivo
 - 2.1.2. Estrategias y tácticas frente a competidores
 - Estrategias Generales de Posicionamiento
 - Estrategias Ofensivas frente a Competidores
 - Estrategias Defensivas
 - 2.2. Entrevistas
 - 2.2.1. Diseño de entrevistas
 - 2.2.2. Registro de entrevistas
 - 2.2.3. Análisis de entrevistas
 - 2.3. Needfinding
 - 2.3.1. User Personas
 - 2.3.2. User Task Matrix
 - 2.3.3. User Journey Mapping
 - 2.3.4. Empathy Mapping
 - 2.3.5. Ubiquitous Language
 - 2.4. Requirements specification
 - 2.4.1. User Stories
 - Technical Stories
 - Spike Story
 - 2.4.2. Impact Mapping
 - Segmento Objetivo 1: Bodegas especializadas por rubro
 - Segmento Objetivo 2: Startups y emprendedores en expansión con necesidades logísticas
 - 2.4.3. Product Backlog
 - Gestión del producto Backlog
 - 2.5. Strategic-Level Domain-Driven Design
 - 2.5.1. EventStorming
 - 2.5.1.1. Candidate Context Discovery
 - Step 1: Unstructured exploration
 - Step 2 : Timeline construction
 - Step 3: Pain Points
 - Step 4: Pivotal Points

- Step 5: Commands
- Step 6: Policies
- Step 7 & 8: Read Models & External Systems
- Step 9 & 10: Aggregates & Bounded Contexts
- 2.5.1.2. Domain Message Flows Modeling
- 2.5.1.3. Bounded Context Canvases
- 2.5.2. Context Mapping
- 2.5.3. Software Architecture
 - 2.5.3.1. Software Architecture Context Level Diagrams
 - 2.5.3.2. Software Architecture Container Level Diagrams
 - 2.5.3.3. Software Architecture Deployment Diagrams
- 2.6. Tactical-Level Domain-Driven Design
 - 2.6.1. Bounded Context: AlertStockManagement Context
 - 2.6.1.1. Domain Layer
 - 2.6.1.2. Application Layer
 - 2.6.1.3. Interface Layer
 - 2.6.1.4. Infrastructure Layer
 - 2.6.1.5. Bounded Context Software Architecture Component Level Diagrams
 - 2.6.1.6. Bounded Context Software Architecture Code Level Diagrams
 - 2.6.1.6.1. Bounded Context Domain Layer Class Diagrams
 - 2.6.2. Bounded Context: Iam Context
 - 2.6.2.1. Domain Layer
 - 2.6.2.2. Interface Layer
 - 2.6.2.3. Application Layer
 - 2.6.2.4. Infrastructure Layer
 - 2.6.2.5. Bounded Context Software Architecture Component Level Diagrams
 - 2.6.2.6. Bounded Context Software Architecture Code Level Diagrams
 - 2.6.2.6.1. Bounded Context Domain Layer Class Diagrams
 - 2.6.2.6.2. Bounded Context Database Design Diagram
 - 2.6.3. Bounded Context: Inventory Context
 - 2.6.3.1. Domain Layer
 - 2.6.3.2. Application Layer
 - 2.6.3.3. Interface Layer
 - 2.6.3.4. Infrastructure Layer
 - 2.6.3.5. Bounded Context Software Architecture Component Level Diagrams
 - 2.6.3.6. Bounded Context Software Architecture Code Level Diagrams
 - 2.6.3.6.1. Bounded Context Domain Layer Class Diagrams
 - Domain: Model
 - Domain: Services
 - 2.6.3.6.2. Bounded Context Database Design Diagram
 - 2.6.4.1. Domain Layer
 - 2.6.4.2. Application Layer
 - 2.6.4.3. Interface Layer
 - 2.6.4.4. Infrastructure Layer
 - 2.6.4.5. Bounded Context Software Architecture Component Level Diagrams
 - 2.6.4.6. Bounded Context Software Architecture Code Level Diagrams
 - 2.6.4.6.1. Bounded Context Domain Layer Class Diagrams
 - Domain: Model
 - Domain: Services
 - 2.6.4.6.2. Bounded Context Database Design Diagram
 - 2.6.5. Bounded Context: Reports Context
 - 2.6.5.1. Domain Layer
 - 2.6.5.2. Application Layer
 - 2.6.5.3. Interface Layer
 - 2.6.5.4. Infrastructure Layer
 - 2.6.5.5. Bounded Context Software Architecture Component Level Diagrams
 - 2.6.5.6. Bounded Context Software Architecture Code Level Diagrams
 - 2.6.5.6.1. Bounded Context Domain Layer Class Diagrams
 - Domain: Model
 - Domain: Services
 - 2.6.5.6.2. Bounded Context Database Design Diagram
 - Capítulo III: Solution UI/UX Design
 - 3.1. Product design
 - 3.1.1. Style Guidelines

- 3.1.1.1. General Style Guidelines
- 3.1.2. Information Architecture
 - 3.1.2.1. Organization Systems
- 3.1.2.1. Organization Systems
 - 3.1.2.2. Labelling Systems
 - 3.1.2.3. SEO Tags and Meta Tags
 - 3.1.2.4. Searching Systems
 - 3.1.2.5. Navigation Systems
 - 3.1.3. Landing Page UI Design
 - 3.1.3.1. Landing Page Wireframe
 - 3.1.3.2. Landing Page Mock-up
 - 3.1.4. Mobile Applications UX/UI Design
 - 3.1.4.1. Mobile Applications Wireframes
 - 3.1.4.2. Mobile Applications Wireflow Diagrams
 - 3.1.4.3. Mobile Applications Mock-ups
 - 3.1.4.4. Mobile Applications User Flow Diagrams
 - 3.1.4.5. Mobile Applications Prototyping
- Capítulo IV: Product Implementation & Validation
 - 4. Product Implementation & Validation
 - 4.1. Software Configuration Management
 - 4.1.1. Software Development Environment Configuration
 - 4.1.2. Source Code Management
 - 4.1.3. Source Code Style Guide & Conventions
 - HTML
 - CSS
 - JavaScript
 - Kotlin
 - 4.1.4. Software Deployment Configuration
 - Deployment Backend:
 - Creación de VPS en Google Cloud Platform (GCP)
 - 1. Configuración Inicial del Proyecto
 - 2. Creación de la Instancia de VM
 - 3. Configuración de Acceso y Seguridad
 - 4.2. Landing Page & Mobile Application Implementation
 - 4.2.1. Sprint 1
 - 4.2.1.1. Sprint Planning 1
 - 4.2.1.2. Sprint Backlog 1
 - 4.2.1.3. Development Evidence for Sprint Review
 - 4.2.1.4. Testing Suite Evidence for Sprint Review
 - Cobertura por pieza de código
 - Relación de tests diseñados (extracto)
 - 4.2.1.5. Execution Evidence for Sprint Review
 - Screenshots
 - Landing Page
 - Funciones claves
 - Sección de Planes Disponibles
 - Sección de cambio de idioma
 - Mobile Frontend
 - 4.2.1.6. Services Documentation Evidence for Sprint Review
 - 4.2.1.7. Software Deployment Evidence for Sprint Review
 - Landing Page
 - 1. Preparación del proyecto
 - 2. Creación de cuenta o acceso a Netlify
 - 3. Nuevo sitio desde Git
 - 4. Autorización y selección del repositorio
 - 5. Configuración del despliegue
 - 6. Despliegue automático
 - 7. Personalización de dominio (opcional)
 - 8. Actualizaciones automáticas
 - 9. Monitoreo del sitio
 - Backend
 - 1. Configuración del Entorno Docker en el VPS
 - 2. Preparación del Proyecto ASP.NET 9
 - 3. Despliegue de la Aplicación

- 4. Configuración de Nginx como Reverse Proxy
 - 4.2.1.8. Team Collaboration Insights during Sprint
- 4.3. Validation Interviews
 - 4.3.1. Diseño de Entrevistas
 - 4.3.2. Registro de Entrevistas
 - 4.3.3. Evaluaciones según heurísticas
- Conclusiones y recomendaciones
- Video App Validation
- Video About-the-Product
- Video About-the-team
- Bibliografía
- Anexos

Student Outcome

ABET - EAC - Student Outcome 7

Criterio: La capacidad de adquirir y aplicar nuevos conocimientos según sea necesario, utilizando estrategias de aprendizaje apropiadas.

En el siguiente cuadro se describe las acciones realizadas y enunciados de conclusiones por parte del grupo, que permiten sustentar el haber alcanzado el logro del ABET – EAC - Student Outcome 7.

Criterio específico	Acciones realizadas	Conclusiones
Actualiza conceptos y conocimientos necesarios para su desarrollo profesional y en especial para su proyecto en soluciones de software.	<p>Jocelyn Almerco</p> <p>TB1:</p> <p>Reconocí que participar en la edición del video de entrevistas y en las herramientas de análisis (Capítulo II, matrices, mapeos y backlog) me exige un aprendizaje permanente para mejorar mi aporte en proyectos de software.</p> <p>TP1:</p> <p>Realice las correcciones del capítulo 2, pantallas de inventario en móviles en Android Studio</p> <p>TB2:</p> <p>..</p> <p>TF:</p> <p>..</p> <p>TF:</p> <p>..</p>	<p>TB1:</p> <p>Como equipo logramos actualizar nuestros conocimientos al aplicar diferentes técnicas y herramientas en el proyecto, lo que nos permitió mejorar nuestra formación profesional y fortalecer el desarrollo de la solución de software.</p> <p>TP1:</p> <p>Como equipo, logramos identificar áreas clave de conocimiento que debíamos reforzar para llevar a cabo las responsabilidades asignadas. Este proyecto nos motivó a actualizar conceptos relacionados con arquitectura de software, diseño UI/UX, frameworks móviles, integración de servicios backend y despliegue en la nube. Gracias a ello, fortaleceremos nuestras competencias profesionales, ya que este conocimiento será fundamental no solo para completar la aplicación, sino también para abordar proyectos futuros con mayor seguridad técnica y claridad metodológica.</p> <p>TB2:</p> <p>..</p> <p>TF:</p> <p>..</p>
Camila Sanchez Rios		
	<p>TB1:</p> <p>Me encargué de crear la organización y los repositorios en el github. Colaboré con Capítulo I: Presentación, Capítulo II: Entrevistas, Competidores</p> <p>TP1:</p> <p>Avance del capítulo 3 y colaboración del Capítulo 4, creación de pantallas IAM en móviles en Android Studio</p>	

TB2:

TF:

Kevin Chi

TB1:

Puse en práctica lo que sabía y reforcé mis conocimientos diseñando el C4 y el modelo de base de datos para el proyecto.

TP1:

Coordinación del grupo, corrección del capítulo 1, creación de pantallas productos en móviles en Android Studio y colaboración en la presentación

TB2:

TF:

Alejandro Oroncoy

TB1:

Mejoré mis conocimientos aplicando el Tactical DDD y organizando el dominio del proyecto de una manera más clara.

TP1:

Corrección del backend, avance capítulo 4 y pantallas de inventario en móviles en Android Studio

TB2:

TF:

Jeremy Paucar

TB1:

Aprendí nuevas formas de mapear el flujo de eventos y actualicé mi conocimiento aplicando EventStorming en el proyecto.

TP1:

Corrección del backend y despliegue, pantallas de productos en móviles en Android Studio

TB2:

TF:

Reconoce la necesidad del aprendizaje permanente para el

Jocelyn Almerco

TB1:

TB1:

desempeño profesional y el desarrollo de proyectos en soluciones de software.

Me encargué de editar el video de las entrevistas. Colaboré con Capítulo II: Entrevistas, User Task Matrix, User Journey Mapping, Empathy Mapping, Requirements specification, User Stories, Impact Mapping, Product Backlog.

TP1:

..

Es importante contar con la capacidad de trabajar en equipo, ya que permite compartir ideas, resolver problemas juntos y mejorar los resultados del proyecto.

TP1:

Durante el desarrollo del proyecto, comprendimos que la ingeniería de software es un campo en constante evolución. La necesidad de aprender nuevas herramientas, frameworks, lenguajes y enfoques fue evidente en cada etapa del trabajo. Como equipo, reconocemos que la mejora continua será indispensable para adaptarnos a los cambios tecnológicos y así poder enfrentar proyectos de mayor complejidad. Esta experiencia nos ha demostrado que el aprendizaje permanente no es una opción, sino una necesidad para mantenernos competitivos y eficientes en nuestra carrera profesional.

TB2:

..

TF:

..

TB2:

..

TF:

..

Camila Sanchez

TB1:

Me encargué de crear la organización y los repositorios en el github. Colaboré con Capítulo I: Presenatcion, Capítulo II: Entrevistas, Competidores

TP1:

..

TB2:

..

TF:

Kevin Chi

TB1:

Me di cuenta de que al trabajar con C4 y el diagrama de base de datos necesito seguir aprendiendo sobre cómo representar bien la arquitectura y los datos de un sistema.

TP1:

..

TB2:

..

TF:

Alejandro Oroncoy

TB1:

Comprendí que usar las tácticas de DDD requiere aprendizaje constante de patrones como

entidades y agregados para aplicarlos bien en el dominio.

TP1:

...

TB2:

...

TF:

...

Jeremy Paucar

TB1:

Entendí que al hacer EventStorming es clave seguir capacitándome para identificar de forma correcta los eventos y procesos del sistema.

TP1:

...

TB2:

...

TF:

...

Objetivos SMART

Jocelyn Damaly Almerco Rojas

Objetivo 1:

Antes del 30/12/2026, habré completado una especialización en Diseño de Interfaces de Usuario (UI/UX), aplicando los conocimientos en un proyecto académico o personal que combine diseño intuitivo y funcionalidad. Para lograrlo, destinaré al menos 5 horas semanales al estudio, finalizaré 3 cursos certificados y diseñaré un prototipo interactivo con Figma o Adobe XD. Consideraré alcanzado el objetivo al obtener la certificación y presentar un portafolio con 2 proyectos funcionales orientados a la experiencia del usuario.

Objetivo 2:

Dentro de los dos primeros años posteriores a mi graduación, me habré incorporado a un equipo de desarrollo como Front-End Developer o UX Designer Junior, donde pueda aplicar principios de accesibilidad, usabilidad y diseño centrado en el usuario. Para lograrlo, mantendré actualizado mi portafolio en línea, realizaré al menos 15 postulaciones mensuales y participaré en 2 hackatones o proyectos colaborativos. Consideraré cumplido el objetivo al obtener una oferta laboral formal y completar mi primer proyecto en producción con retroalimentación positiva del equipo y usuarios finales.

Jeremy Paucar Meneses

Objetivo 1:

Antes del 31/07/2026, habré obtenido una certificación en Desarrollo Full Stack con JavaScript (Node.js y Vue.js), implementando un proyecto completo que integre backend, frontend y base de datos. Para conseguirlo, estudiaré 6 horas semanales, culminaré 3 cursos especializados y desarrollaré una aplicación funcional desplegada en la nube. Consideraré cumplido el objetivo al aprobar la certificación y publicar la aplicación con documentación técnica en un repositorio Git.

Objetivo 2:

En un plazo máximo de 18 meses después de graduarme, habré conseguido mi primer puesto como Software Developer Junior, participando en el desarrollo de módulos o microservicios en un entorno productivo. Para alcanzarlo, realizaré 20 postulaciones mensuales, practicaré algoritmos 2 veces por semana y participaré en al menos 1 proyecto colaborativo o open source. Consideraré logrado el objetivo al obtener una oferta de trabajo estable y contribuir con al menos 1 entrega de código validada por revisión de pares (code review).

Camila Sánchez Ríos

Objetivo 1:

Antes del 31/12/2026, habré completado un diplomado en Calidad de Software y Pruebas Automatizadas, certificándome en herramientas como

Selenium o Postman. Para ello, dedicaré 5 horas semanales al estudio, aprobaré al menos 3 cursos especializados y diseñaré un conjunto de pruebas automatizadas para un sistema real o académico. Consideraré alcanzado el objetivo al obtener la certificación y entregar un informe técnico con métricas de cobertura y defectos detectados.

Objetivo 2:

Dentro de los dos años posteriores a mi graduación, habré logrado un puesto en el área de Aseguramiento de Calidad (QA Engineer) o Analista de Pruebas de Software, participando activamente en el ciclo de vida de pruebas de un producto digital. Para lograrlo, realizaré 10 postulaciones mensuales, asistiré a 2 conferencias técnicas de QA y mantendré un registro de mejoras continuas en metodologías ágiles. Consideraré el objetivo cumplido al obtener una oferta formal y participar en al menos una liberación exitosa de software con métricas de defectos bajo el 2%.

Kevin Chi Cruzatt

Objetivo 1:

Antes del 30/06/2026, obtendré una certificación en Arquitectura de Software con Java y Spring Boot, diseñando una aplicación modular basada en principios de Domain Driven Design (DDD) y CI/CD. Para lograrlo, dedicaré 6 horas semanales al estudio, culminaré 3 cursos especializados y publicaré un repositorio con documentación técnica y métricas de cobertura superiores al 80%. Consideraré logrado el objetivo al aprobar la certificación y desplegar la aplicación en un entorno de prueba funcional.

Objetivo 2:

En un plazo máximo de 18 meses después de mi graduación, me habré incorporado a una empresa tecnológica como Backend Developer o Software Engineer, participando en la construcción de servicios escalables y mantenibles. Para alcanzarlo, enviaré 20 postulaciones mensuales, mantendré mi portafolio con al menos 3 proyectos técnicos y realizaré una entrevista simulada quincenal. Consideraré cumplido el objetivo al recibir una oferta laboral y completar una funcionalidad de backend validada en producción.

Alejandro Oroncoy Almeyda

Objetivo 1:

Durante el primer año tras mi graduación, habré desarrollado un portafolio técnico con al menos 4 proyectos completos en distintas áreas del desarrollo de software (backend, frontend y bases de datos). Para lograrlo, destinaré 6 horas semanales al aprendizaje y práctica, completaré 3 cursos en línea y documentaré los proyectos con pruebas unitarias y despliegue en la nube. Consideraré alcanzado el objetivo al tener el portafolio publicado y revisado por mentores o pares profesionales.

Objetivo 2:

En un plazo máximo de dos años después de graduarme, obtendré mi primer empleo formal en el área de Desarrollo de Software, participando en proyectos colaborativos que fortalezcan mis habilidades de trabajo en equipo y resolución de problemas. Para lograrlo, realizaré 15 postulaciones mensuales, asistiré a 2 ferias laborales o eventos de tecnología al año y mantendré mi perfil de LinkedIn activo y actualizado. Consideraré el objetivo cumplido al firmar mi primer contrato y haber completado al menos una entrega de producto en producción., eso pegalo en smart please, cami eso podrías ponerlo en el GitHub please

Capítulo I: Presentación

1.1. Startup Profile

1.1.1. Descripción de la Startup

StockWise es un aplicativo móvil de gestión de inventarios dirigida a pequeñas y medianas empresas, startups y bodegas especializadas. Su objetivo principal es facilitar el control eficiente de entradas y salidas de productos, la gestión de usuarios, la configuración de alertas inteligentes, la generación de reportes detallados y de boleta de venta, todo a través de una interfaz intuitiva y accesible desde cualquier dispositivo.

La solución responde a una problemática concreta: muchos negocios aún utilizan métodos manuales (como hojas de cálculo o registros en papel) para administrar sus inventarios y pagos, lo cual genera errores, desorganización, sobrecompras y pérdidas económicas. StockWise busca resolver este problema digitalizando y centralizando el control del inventario y de las ganancias de la tienda, permitiendo a los negocios tomar decisiones basadas en datos y optimizar su operación mediante cuatro planes de funcionalidad avanzada:

- **Plan A – Entrada por voz:** Permite registrar productos mediante comandos de voz (ejemplo: "Agregar 20 botellas de agua al inventario"), ofreciendo mayor rapidez y comodidad, especialmente en situaciones donde el personal no pueda usar el teclado.
- **Plan B – Geolocalización (GPS):** Integra funciones de ubicación para optimizar la trazabilidad y distribución de productos, permitiendo identificar su procedencia, registrar puntos de entrega y visualizar en un mapa interactivo las distintas sedes o sucursales vinculadas a la tienda principal.
- **Plan C – Localización y predicción inteligente:**
 - Localiza: Utiliza un mapa interactivo con integración de códigos QR para ubicar productos dentro del almacén de manera precisa.
 - Predice: Implementa un sistema de reabastecimiento inteligente basado en patrones de ventas, que sugiere cuándo y cuánto stock reponer para evitar quiebres de inventario.
- **Plan D – Escaneo por lotes con cámara rápida:** En lugar de códigos de barras, la app toma una foto del producto o lote y una API de visión (como Google ML Kit) devuelve etiquetas genéricas. El usuario confirma el producto exacto y la cantidad en un menú antes de registrarla, pudiendo ver directamente su ubicación en el almacén virtual.

Misión: Brindar a pequeñas y medianas empresas una solución de gestión de inventarios y pagos sencilla, accesible y eficiente, que les permita digitalizar su operación, reducir errores logísticos y tomar decisiones basadas en datos reales, apoyando así su crecimiento sostenible.

Visión: StockWise se enfoca en ser una aplicación preferida de los empresarios que buscan una gestión de inventarios, herramientas de alto valor con un modelo escalable, adaptable y centrado en el usuario.

1.1.2. Perfiles de integrantes del equipo



Jocelyn Damaly Almerco Rohas

Soy estudiante de Ingeniería de Software y actualmente curso el 5to ciclo. En mi tiempo libre me gusta tocar ukelele, leer libros, resolver sudoku, escuchar música y ver series.



Jeremy Paucar Meneses

Estudiante de Ingeniería de Software, apasionado a programar, tengo conocimientos en c++, js, c#, y algunos frameworks como Vue, dentro de mis pasatiempos esta escuchar musica, ver futbol y ver series



Sánchez Ríos, Camila Cristina

Soy estudiante de la carrera de Ingeniería de Software en la Universidad Peruana de Ciencias Aplicadas, actualmente me encuentro en el octavo ciclo. Me gusta escuchar música y leer en los ratos libres y aprender más sobre la carrera.



Chi Cruzatt, Kevin Jorge

Actualmente estoy cursando la carrera de ingeniería de Software y me encuentro en el 6º ciclo. Tengo experiencia utilizando diferentes metodologías de diseño, como Domain Driven Design (DDD) y CMV, además de frameworks como Laravel y Spring en lenguajes como Java, PHP, etc. Me considero una persona paciente y perseverante en situaciones difíciles.



Oroncoy Almeyda, Alejandro Daniel

Mi nombre es Alejandro Oroncoy. Tengo 19 años, soy estudiante de la carrera de ingeniería de software, estoy en 6to ciclo. Me considero una persona proactiva, autodidacta y orientada a objetivos.



1.2. Solution Profile

1.2.1 Antecedentes y problemática

- **Who (¿Quiénes?)**

Emprendedores, startups y pequeñas/medianas empresas con bodegas físicas que almacenan productos de distintos rubros como ropa, calzado, electrodomésticos, ferretería o alimentos.

- **What (¿Qué sucede?)**

A medida que sus negocios escalan, la gestión manual del inventario con hojas de cálculo o registros físicos se vuelve ineficiente, generando pérdidas, errores, quiebres de stock, sobrecompras y desorden logístico.

- **When (¿Cuándo ocurre?)**

En el momento en que el negocio empieza a crecer, aumentar su variedad de productos o abrir múltiples canales de venta, como tiendas físicas y plataformas online.

- **Where (¿Dónde ocurre?)**

En bodegas físicas propias, espacios alquilados o incluso en el hogar del emprendedor, especialmente en etapas tempranas o de expansión del negocio.

- **Why (¿Por qué es un problema?)**

La falta de un sistema centralizado y en tiempo real impide tomar decisiones estratégicas basadas en datos. Esto afecta la planificación de compras, genera pérdidas económicas, y daña la experiencia del cliente final.

- **How (¿Cómo lo solucionan hoy?)**

Mediante herramientas manuales como cuaderno o hojas de papel, inventarios escritos o software no especializado, que resultan limitados, propensos a errores y poco escalables.

- **How much (¿Cuánto cuesta no resolverlo?)**

El costo se traduce en pérdidas económicas significativas por productos no vendidos, errores de stock, tiempo invertido en tareas manuales y menor competitividad frente a negocios más organizados.

1.2.2 Lean UX Process

1.2.2.1. Lean UX Problem Statements

El propósito de StockWise es proporcionar una aplicación intuitiva y accesible para que empresas pequeñas y medianas, startups y bodegas especializadas puedan gestionar su inventario de forma eficiente, digitalizada y sin complicaciones técnicas ni altos costos.

Actualmente, muchos de estos negocios aún dependen de métodos manuales para registrar y controlar su stock. Este enfoque genera errores frecuentes, tales como registros incorrectos, desactualización en tiempo real o pérdidas de productos. Como resultado, se producen compras innecesarias, falta de trazabilidad, escasa visibilidad sobre los niveles de inventario y una gestión logística ineficiente. Según un estudio de NetSuite (2024), el promedio de precisión de inventario en empresas es de aproximadamente 83 %, lo que refleja un margen de error significativo que afecta los costos operativos y la toma de decisiones.

Diversas investigaciones sostienen que una gestión ineficiente del inventario repercute directamente en la productividad empresarial, la satisfacción del cliente y la rentabilidad general del negocio, además de generar sobrecostos en almacenamiento y desperdicio de recursos (Altavant Consulting, 2024; Arce-Gonzales & Sandoval, 2023). A medida que estas empresas escalan, el desorden operativo se vuelve insostenible, reduciendo su capacidad de respuesta ante la demanda y su competitividad en el mercado.

Por lo tanto, surge la necesidad de diseñar una aplicación de gestión de inventarios que sea simple, funcional y adaptable, capaz de atender las necesidades reales de las pymes en crecimiento. Esta herramienta debe facilitar el control exacto del inventario, reducir errores humanos, mejorar la trazabilidad y proporcionar información en tiempo real para respaldar decisiones basadas en datos.

1.2.2.2. Lean UX Assumptions

Business Assumptions:

1. Creemos que los negocios emergentes necesitan digitalizar su gestión de inventarios a través de una solución móvil accesible.
2. Estas necesidades se pueden satisfacer con una aplicación móvil intuitiva, escalable y de bajo costo.
3. Nuestros clientes iniciales serán emprendedores, startups y pequeñas empresas con bodegas especializadas que operan de manera ágil.
4. El principal valor que busca el cliente es tener control y visibilidad total de su inventario en tiempo real, desde cualquier lugar.
5. El cliente obtendrá además alertas inteligentes, reportes automáticos, generación de boletas de venta y una experiencia de usuario móvil optimizada.
6. Adquiriremos la mayoría de nuestros clientes mediante publicidad en redes sociales dirigida, ASO (Optimización de la App Store) y asociaciones con comunidades de emprendedores.
7. Nuestro modelo de ingresos se basará en un esquema freemium dentro de la app, con upgrade a planes premium que desbloqueen funcionalidades avanzadas.
8. Nuestra competencia directa e indirecta incluye aplicaciones genéricas como hojas de cálculo móviles (Excel, Sheets), recordatorios básicos (Calendar) y ERPs complejos con apps móviles poco intuitivas.
9. Nuestra ventaja competitiva radicará en la simplicidad móvil, el enfoque específico en las pymes y la especialización en la gestión de inventarios sobre la marcha.
10. El mayor riesgo para el negocio es una baja tasa de adopción o la percepción de que la app es compleja o redundante frente a métodos manuales.
11. Mitigaremos este riesgo realizando pruebas de usabilidad móvil con usuarios reales, iteraciones rápidas basadas en feedback y una estrategia de onboarding dentro de la app que guíe al usuario paso a paso.

User Assumptions:

- **¿Quién es el usuario?** Dueños de negocios, encargados de bodegas o logística en pymes/startups que necesitan gestionar inventarios de forma remota.
- **¿Qué problemas busca resolver nuestro producto?** La falta de control inmediato del stock, los errores por registros manuales en papel, los sobrecostos por pérdidas y las ventas fallidas debido a quiebres de stock inesperados.
- **¿Qué características son importantes?** Un registro rápido de productos (por voz, cámara o manual), un historial de movimientos accesible, alertas push personalizables, reportes visuales simplificados y la generación de boletas de venta directamente desde el dispositivo móvil.
- **¿Dónde encaja nuestro producto en su trabajo?** Se integra en su flujo de trabajo diario para la gestión del inventario en la bodega, en el punto de venta o durante las rondas de reposición, permitiendo decisiones informadas desde cualquier lugar.
- **¿Cuándo y cómo es usado nuestro producto?** Se utiliza de manera frecuente a lo largo del día, directamente desde sus teléfonos inteligentes, para registrar entradas/salidas al momento, consultar niveles de stock en tiempo real y generar comprobantes al instante.
- **¿Cómo debe verse y comportarse?** Debe tener una interfaz de usuario (UI) móvil limpia, simple y con navegación táctil intuitiva. Debe ser rápida, responsive y funcionar sin conexión para casos de uso críticos.

Feature Assumptions:

Creemos que la aplicación móvil debe contar con una interfaz táctil intuitiva y diseños adaptados a dispositivos móviles que permitan a emprendedores y administradores adoptarla sin dificultad, minimizando la curva de aprendizaje.

Creemos que la app debe proporcionar notificaciones push y alertas personalizables (por stock bajo, fechas de vencimiento o pedidos de proveedores) que mantendrán a los usuarios informados de manera proactiva para prevenir errores operativos.

Creemos que el sistema debe incluir un módulo de reportes y dashboards visuales optimizados para móvil que permitan visualizar de un vistazo métricas clave (productos más vendidos, niveles de stock, tendencias), facilitando la toma de decisiones ágiles.

1.2.2.3. Lean UX Hypothesis Statements

Hipótesis 1: Creemos que las alertas inteligentes push resolverán el problema de pérdidas económicas por quiebres de stock inesperados que enfrentan las pymes.

Sabremos que es cierto

Cuando al menos el 80% de los usuarios activos reporten una reducción del 25% en pérdidas por productos agotados y mejoren su planificación de reabastecimiento en encuestas trimestrales.

Hipótesis 2: Creemos que los reportes visuales móviles resolverán la falta de visibilidad sobre el rendimiento del inventario que impide a los emprendedores tomar decisiones estratégicas basadas en datos.

Sabremos que es cierto

Cuando el 70% de los usuarios utilicen los reportes semanalmente y reporten una mejora del 30% en sus decisiones de compra y gestión de stock.

Hipótesis 3: Creemos que el modelo freemium resolverá la barrera económica que impide a pequeños negocios acceder a herramientas profesionales de gestión de inventarios.

Sabremos que es cierto

Cuando logremos una adopción del 15% de usuarios gratuitos que conviertan a premium en 60 días, demostrando el valor percibido de la solución.

Hipótesis 4: Creemos que las alertas automatizadas por stock bajo resolverán el problema de sobrecostos operativos causados por el monitoreo manual del inventario.

Sabremos que tenemos razón

Cuando los usuarios reduzcan en 40% el tiempo dedicado a control manual de stock y eliminan el 90% de los quiebres de inventario críticos.

Hipótesis 5: Creemos que la generación digital de boletas de venta resolverá los errores de facturación y la desorganización financiera que afecta el flujo de caja de las pymes.

Sabremos que tenemos razón

Cuando los usuarios reporten una reducción del 50% en errores de facturación y mejoren en 35% la precisión de su control de ingresos diarios.

1.2.2.4. Lean UX Canvas

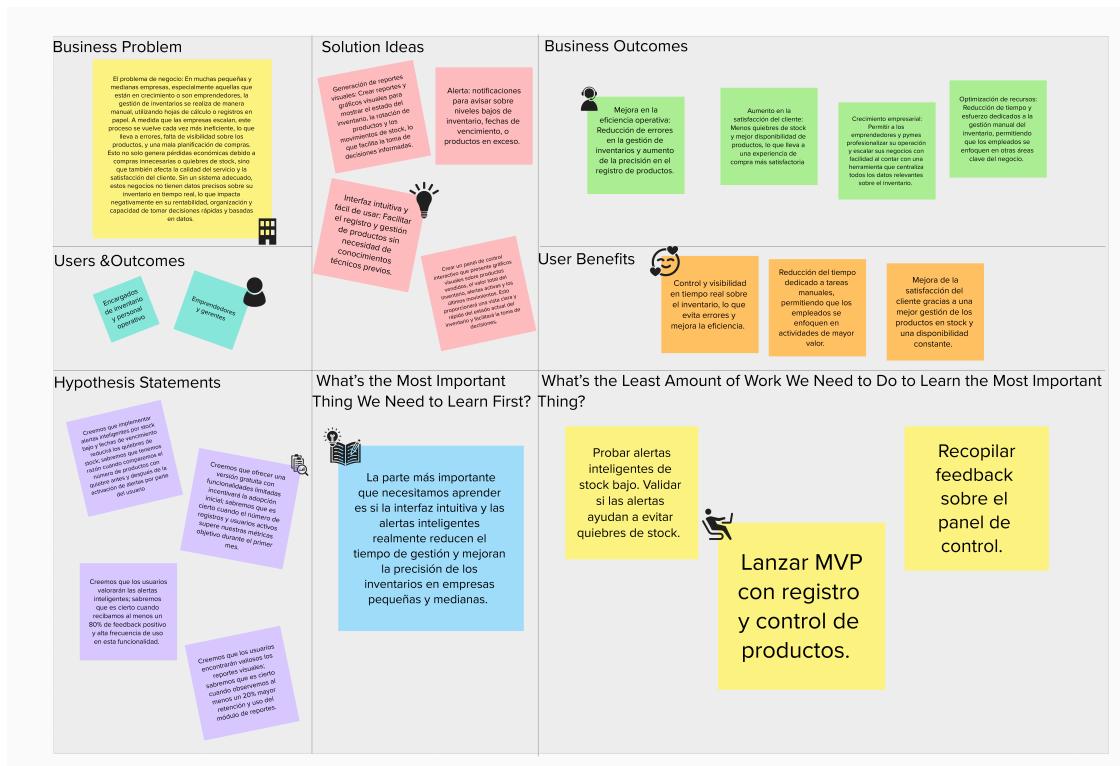


Imagen (Nº1). Elaboración propia. Realizado en Canva

Capítulo II: Requirements Development and Software Solution Design

2.1. Competidores

2.1.1. Análisis competitivo

Objetivo del análisis: Identificar el posicionamiento competitivo de StockWise en el mercado de aplicaciones móviles de gestión de inventarios para pymes, entendiendo las ventajas diferenciales y oportunidades de mejora.

	StockWise 	Zoho Inventory 	Odoo Inventory 	TradeGecko 
--	---	---	--	--

PERFIL

Overview	Solución móvil integral especializada en gestión de inventarios para pymes y emprendedores	Suite de negocio completa con módulo de inventario como parte de su ecosistema	Software de gestión open-source con módulo de inventario personalizable	Plataforma de gestión de inventario centrada en comercio electrónico y retail
Ventaja competitiva	Especialización móvil, interfaz intuitiva, planes funcionales específicos (voz, GPS, IA) y enfoque en pymes	Integración completa con ecosistema Zoho y terceros, robustez funcional	Flexibilidad y personalización por ser open-source	Fuerte integración con canales de e-commerce y plataformas de venta

PERFIL DE MARKETING

Mercado objetivo	Pymes, emprendedores, startups y bodegas especializadas	Medianas empresas y pymes en crecimiento	Empresas de todos los tamaños que buscan personalización	Negocios de e-commerce y retail
Estrategias de marketing	Marketing digital, ASO, comunidades de emprendedores, modelo freemium	Marketing B2B, fuerza de ventas, contenido educativo	Comunidad open-source, partners de implementación	Marketing especializado en e-commerce, ferias sectoriales

PERFIL DE PRODUCTO

Productos & Servicios	App móvil con: gestión inventory, alertas, reportes, boletas, planes voz/GPS/IA/scanner	Software web y móvil con: inventory, órdenes venta, envíos, facturación	Suite modular web/móvil con: inventory, manufactura, ventas, CRM	Plataforma web/móvil: inventory, pedidos, logística, analytics
Precios & Costos	Modelo freemium con planes premium escalables por funcionalidad	Planes por volumen de órdenes, desde gratis hasta enterprise	Freemium modular, costos de implementación y personalización	Planes premium por volumen de ventas y usuarios
Canales de distribución	Móvil (iOS/Android)	Web y móvil (iOS/Android)	Web y móvil (iOS/Android)	Web y móvil (iOS/Android)

Análisis SWOT Fortalezas - Especialización móvil nativa

- Interfaz intuitiva
- Funcionalidades innovadoras (voz, GPS, IA)
- Enfoque específico en pymes - Ecosistema integrado
- Solución completa
- Reconocimiento de marca
- Escalabilidad - Flexibilidad y personalización
- Comunidad activa
- Modelo open-source
- Múltiples módulos integrados - Especialización en e-commerce
- Integraciones con plataformas de venta
- Analytics avanzado **Debilidades** - Marca desconocida

- Ecosistema limitado vs competidores
- Recursos limitados vs grandes competidores - Curva de aprendizaje más pronunciada
- Puede ser complejo para pymes muy pequeñas
- Costo elevado para planes avanzados - Requiere personalización para funcionar óptimamente
- Soporte puede ser limitado en versión community - Enfoque muy específico en e-commerce
- Menos adaptable a otros modelos de negocio
- Precios elevados **Oportunidades** - Crecimiento mercado pymes digitales
- Tendencia hacia soluciones móviles
- Expansión a Latinoamérica
- Alianzas con comunidades de emprendedores - Expansión a nuevos mercados
- Desarrollo de más integraciones
- Consolidación como suite integral - Crecimiento de la comunidad de desarrolladores
- Expansión de funcionalidades móviles
- Alianzas con implementadores - Crecimiento del e-commerce global
- Expansión a nuevos verticales de retail
- Desarrollo de analytics predictivo **Amenazas** - Entrada de competidores establecidos al mercado móvil
- Commoditización de funcionalidades básicas
- Dificultad para captar usuarios frente a alternativas gratuitas - Competencia de soluciones más especializadas
- Precios más competitivos de alternativas
- Cambios en regulaciones internacionales - Fragmentación de la comunidad
- Calidad variable de implementaciones
- Competencia de soluciones SaaS más pulidas - Competencia de plataformas de e-commerce con módulos nativos
- Cambios en APIs de plataformas de venta
- Consolidación del mercado

2.1.2. Estrategias y tácticas frente a competidores

Estrategias Generales de Posicionamiento

1. Especialización en Experiencia Móvil

Objetivo: Diferenciarnos como la solución móvil nativa más intuitiva del mercado.

Tácticas:

- Desarrollar funcionalidades exclusivas móviles (voz, GPS, escaneo por cámara)
- Optimizar la interfaz para uso táctil con gestos intuitivos
- Garantizar funcionamiento offline para operaciones críticas
- Implementar notificaciones push contextuales y personalizables

2. Enfoque en Usabilidad para PYMES

Objetivo: Ofrecer la curva de aprendizaje más corta del mercado.

Tácticas:

- Crear sistema de onboarding guiado dentro de la aplicación
- Desarrollar tutoriales interactivos por funcionalidad
- Implementar asistente inteligente para tareas complejas
- Diseñar plantillas preconfiguradas por tipo de negocio

Estrategias Ofensivas frente a Competidores

1. Contra Zoho Inventory

Debilidad a explotar: Complejidad para PYMES pequeñas

Tácticas:

- Campañas comparativas destacando simplicidad vs complejidad
- Programa de migración asistida desde Zoho
- Planes precio más competitivos para segmento pequeño-mediano
- Funcionalidades "justo lo necesario" sin sobrecarga de opciones

2. Contra Odoo Inventory

Debilidad a explotar: Requiere personalización e implementación

Tácticas:

- Mensajería centrada en "funciona desde el día 1"

- Precios transparentes sin costos ocultos de implementación
- Casos de éxito de implementación en menos de 24 horas
- Servicio de configuración inicial incluido en planes premium

3. Contra TradeGecko

Debilidad a explotar: Enfoque limitado a e-commerce

Tácticas:

- Posicionamiento como solución multi-canal integral
- Funcionalidades para negocios físicos + digitales
- Campañas dirigidas a retail tradicional con interés en digitalizarse
- Precios más accesibles para negocios mixtos

Estrategias Defensivas

1. Ante posible commoditización

Tácticas:

- Desarrollo continuo de funcionalidades innovadoras (voz, IA, realidad aumentada)
- Programa de fidelización con beneficios escalables
- Integraciones exclusivas con proveedores locales
- Comunidad de usuarios para feedback y co-creación

2. Protección frente a entrada de grandes competidores

Tácticas:

- Alianzas estratégicas con asociaciones de PYMES
- Contratos a largo plazo con precios congelados
- Desarrollo de especialización vertical por industria
- Branding fuerte centrado en el enfoque PYME

2.2. Entrevistas

La finalidad de realizar entrevistas es obtener un alcance más completo sobre las experiencias, perspectivas y opiniones de los segmentos de mercado definidos. Nuestro objetivo es recolectar información valiosa que nos permita conocer mejor a nuestro público objetivo. Con estas entrevistas se tendrá una visión más clara de las necesidades y expectativas de nuestros usuarios.

2.2.1. Diseño de entrevistas

Segmento #1: Bodegas especializadas por rubro

Preguntas principales

1. ¿Podrías contarme cómo gestionas actualmente el inventario de tu bodega?
2. ¿Cuáles son los mayores retos que enfrentas al momento de organizar tus productos?
3. ¿Has tenido pérdidas o problemas por errores en el inventario? ¿Cómo los resolviste?
4. ¿Qué tan importante es para ti tener un control en tiempo real de tu stock?
5. ¿Utilizas algún sistema o herramienta digital? ¿Cuál y cómo te va con ella?

Preguntas complementarias

1. ¿Cómo te enteras cuando falta un producto o está a punto de vencerse?
2. ¿Qué tipo de reportes o información te gustaría tener sobre tu inventario?
3. ¿Qué dispositivos usas más en tu trabajo (laptop, celular, tablet)?
4. ¿Cómo crees que una plataforma digital podría ayudarte a mejorar tu operación?

Segmento #2: Startups y emprendedores en expansión con necesidades logísticas

Preguntas principales

1. ¿Cómo manejas actualmente el inventario de tu negocio?
2. ¿En qué momentos has sentido que el control del stock te limita o te hace perder tiempo?
3. ¿Cómo llevas el registro de entradas y salidas de productos?
4. ¿Qué te gustaría mejorar de tu proceso logístico actual?
5. ¿Has considerado usar alguna plataforma para gestionar tu inventario? ¿Por qué sí o por qué no?

Preguntas complementarias

1. ¿Qué herramientas digitales usas en tu negocio actualmente?
2. ¿Dónde almacenas tus productos?
3. ¿Qué tan seguido necesitas revisar el stock?
4. ¿Qué redes sociales o canales digitales usas para vender?

2.2.2. Registro de entrevistas

Segmento 1: Bodegas especializadas por rubro

Entrevista	1	Nombre	Milagros Almerco
Edad	28	Distrito	San Martín de Porres
Captura de la entrevista:	Milagros Almerco, evaluó la app móvil StockWise y destacó su facilidad de uso al registrar productos y gestionar inventario. Aunque al inicio tuvo dudas sobre la diferencia entre "stock" y "lote", logró comprenderlo rápidamente gracias a la interfaz intuitiva. Resaltó como funcionalidades más valiosas el control de productos próximos a vencer y la visualización por lotes, que considera claves para evitar pérdidas. Valoró también el diseño moderno y claro, sin presentar dificultades técnicas. Sugirió como mejora incluir un sistema para registrar pagos pendientes o créditos. En general, calificó la app como práctica, útil y recomendada para otros dueños de bodegas.		
URL de la grabación	Ver grabación		
Timing	00:00 - 5:48		

Entrevista	2	Nombre	Elvis Aranga Mesa
Edad	31	Distrito	Santiago de Surco
Captura de la entrevista:	Elvis Aranga Meza, encargado de un mini market en Surco, compartió su experiencia utilizando la app móvil para la gestión de bodegas, destacando la facilidad de uso tanto para agregar productos como para navegar por las funciones principales, y calificando su utilidad con un 9 sobre 10. Consideró el diseño visual adecuado y sencillo, aunque sugirió mejorar el tamaño de la interfaz en dispositivos móviles. Resaltó la utilidad de funciones como la visualización de fechas de vencimiento para evitar mermas y la organización de productos por lote, lo cual facilita la gestión de promociones. Sugirió incorporar reportes y estadísticas que permitan proyectar ventas y establecer metas. Finalmente, recomendó la plataforma a otros bodegueros, especialmente por su precio y utilidad, aunque señaló la importancia de brindar asistencia a usuarios con menos experiencia tecnológica.		
URL de la grabación	Ver grabación		
Timing	5:48 - 11:47		
Entrevista	3	Nombre	Catalina Villa Guerra
Edad	28	Distrito	Breña
Captura de la entrevista:	Catalina es una administradora de 28 años, encargada de gestionar una bodega familiar en el distrito de Breña. Ella menciona que las funciones que más le gustaron fueron la de alerta de productos próximos a vencer y la de control de stock, ya que considera que le ayudarían a evitar pérdidas por productos caducados y a mantener un mejor seguimiento de sus existencias. Sin embargo, percibe que estas funciones aún son básicas y podrían ofrecer opciones más avanzadas, como recordatorios personalizados o reportes automáticos. En cuanto a la interfaz, a Catalina le agrado el diseño sencillo y organizado, destacando que los botones son fáciles de identificar y las categorías están bien distribuidas. También comentó que la aplicación le resultaría útil para delegar mejor el control del inventario entre los miembros de su familia, facilitando así la administración general de la bodega.		
URL de la grabación	Ver grabación		
Timing	11:48 - 21:31		

Segmento 2: Startups y emprendedores en expansión con necesidades logísticas

Entrevista	4	Nombre	Juan Carlos Ramírez
Edad	49	Distrito	Surquillo
Captura de la entrevista:	Juan Carlos Ramírez, emprendedor de 49 años, compartió su experiencia al utilizar la aplicación para la gestión de inventarios, destacando la facilidad de uso en el registro y control de productos, así como la organización del historial de movimientos. Consideró que estas herramientas podrían ayudarlo significativamente en su proceso de digitalización, ya que actualmente gestiona las entradas y salidas de manera manual mediante boletas y facturas físicas. En cuanto al diseño, señaló que la interfaz es clara y ordenada, aunque sugirió mejorar la personalización de reportes para adaptarlo mejor a las necesidades de su negocio. Finalmente, destacó que funcionalidades como la alerta de stock bajo y la gestión por lotes serían de gran apoyo para tener un control más preciso del		

inventario, y que la incorporación de estas herramientas digitales le permitiría automatizar procesos y liberar tiempo operativo en su gestión diaria.

URL de la grabación	Ver grabación		
Timing	21:31 - 25:14		
Entrevista	5	Nombre	Leonardo Gamboa
Edad	26	Distrito	San Miguel
Captura de la entrevista: 	Leonardo G., emprendedor que maneja actualmente su inventario de manera manual con hojas de Excel y una libreta, probó la plataforma y comentó que le pareció intuitiva, útil y eficaz para el control de su negocio. Señaló que la sección de alertas de stock le resultó especialmente atractiva, ya que le permitiría evitar ventas de productos agotados y responder más rápido a la demanda constante que maneja. En cuanto al diseño, mencionó que la interfaz es clara y sencilla, con botones fáciles de identificar y una navegación fluida. Leonardo destacó que la herramienta podría ayudarlo a organizar mejor sus pedidos y actualizar su inventario sin esfuerzo, reemplazando las hojas manuales que utiliza actualmente. También valoró que la plataforma sea accesible y de fácil uso, algo clave para quienes aún no están acostumbrados a los sistemas digitales, pero buscan dar ese paso hacia la automatización de su negocio.		
URL de la grabación	Ver grabación		
Timing	25:14 - 29:36		
Entrevista	6	Nombre	Gael Rivera
Edad	xx	Distrito	xx
Captura de la entrevista: 	Eduardo de Rivera Sosa, un emprendedor que importa productos desde Asia para venderlos en Perú, comentó que la idea del proyecto le pareció atractiva y muy necesaria para negocios que manejan múltiples referencias de productos. Durante la prueba, resaltó la facilidad de uso y la organización del panel principal, especialmente en el registro de productos y control de stock. Mencionó que actualmente lleva su inventario en Excel y organiza sus pedidos en Notion, por lo que considera que la app podría integrar y simplificar ambos procesos en un solo espacio digital. Valoró la función de alerta de stock bajo y las opciones de historial de movimientos, que le permitirían tener una visión más clara de sus existencias y reducir pérdidas por errores humanos o por actualizaciones tardías. En cuanto a la apariencia visual, Eduardo comentó que la landing page le pareció moderna y atractiva, destacando que transmite confianza y profesionalismo. Sin embargo, sugirió que la exportación de datos a Excel o PDF podría mejorarse con formatos más personalizables.		
URL de la grabación	Ver grabación		
Timing	29:36 - 33:36		

2.2.3. Análisis de entrevistas

Segmento 1: Bodegas especializadas por rubro

Los entrevistados (Milagros, Elvis, Catalina) gestionan inventarios de forma manual, con cuadernos o Excel, lo que dificulta mantener la información precisa y actualizada. Cuando los datos llegan tarde, los productos se vencen sin que nadie lo note y se pierden ventas; además, los cambios de precio no quedan registrados. El problema no es falta de interés por la tecnología, sino que registrar toma tiempo mientras atienden a clientes. Para este grupo la solución debe reducir pasos: registro rápido (por voz en el celular, duplicar fichas o cargar varios productos de una vez), una pantalla que muestre "lo que vence pronto" con avisos simples y la ubicación por estantería para encontrar productos sin perder minutos. Con ello deberían bajar las pérdidas, reducir el tiempo de registro y aumentar la confianza en el sistema.

Además de los aspectos técnicos, los entrevistados expresan frustración por el desorden y la pérdida de productos, pero también muestran disposición a adoptar soluciones simples que les permitan sentir mayor control y tranquilidad en la gestión diaria.

Característica	Frecuencia (n/3)	Porcentaje	Entrevistas relacionadas
Control manual del inventario (cuaderno/Excel)	3/3	100%	1(Milagros), 2(Elvis), 3(Catalina)
Pérdidas por productos vencidos / falta de alertas	3/3	100%	1, 2, 3
Precios desactualizados o cambios no registrados	2/3	67%	1, 2
Falta de tiempo por atención a clientes (se omiten registros)	2/3	67%	1, 3
Preferencia por solución simple y rápida (alertas de vencimiento, registro ágil)	3/3	100%	1, 2, 3
Necesidad de ubicar productos por estantería/sector	2/3	67%	1, 3

Segmento 2: Startups y emprendedores en expansión con necesidades logísticas

Los entrevistados Juan Carlos, Leonardo y Eduardo manejan inventarios mayormente manuales o semiautomatizados (Excel, cuadernos y sistemas en transición). El reto principal es coordinar más volumen y varios canales (WhatsApp y redes). Muchas salidas no quedan en un solo registro y preparar pedidos demora porque no se ubican los productos con precisión; además, varias personas cambian precios y no se sabe quién hizo qué. Necesitan orden sin un sistema pesado: roles y permisos con historial para saber quién hizo cada cambio, zonas y códigos QR para acelerar el picking, combos que resten insumos automáticamente y reportes simples por canal para ver qué se mueve más. Así pueden crecer sin perder control.

Pese a estas limitaciones operativas, en este grupo se percibe una actitud proactiva y de mejora continua: buscan herramientas que reflejen su crecimiento y les brinden seguridad al delegar tareas, evidenciando motivación por alcanzar mayor eficiencia y profesionalización.

Característica	Frecuencia (n/3)	Porcentaje	Entrevistas relacionadas
Gestión manual o semiautomatizada (Excel/cuadernos/sistema parcial)	3/3	100%	4(Juan Carlos), 5(Leonardo), 6(Eduardo)
Ventas y coordinación por WhatsApp/redes sociales	3/3	100%	4, 5, 6
Problemas para ubicar productos y preparar pedidos rápido	2/3	67%	4, 5
Falta de control centralizado / trazabilidad de cambios de precio y salidas	2/3	67%	5, 6
Interés en digitalizar para ganar eficiencia (sin sistemas pesados)	3/3	100%	4, 5, 6
Necesidad de roles/permisos e historial de cambios	2/3	67%	5, 6
Reportes simples por canal y mayor visibilidad del stock	2/3	67%	4, 5

Ambos segmentos comparten el manejo manual, los problemas con perecibles y la falta de visibilidad, por lo que requieren automatización. La diferencia es que las bodegas piden simplicidad inmediata, mientras que las startups necesitan además trazabilidad y velocidad. Por ello se propone empezar con un núcleo fácil de usar (alertas de caducidad, registro rápido, ubicación clara y cambios de precio con historial) y activar, solo si se requiere, funciones de escala como roles, QR, combos y reportes por canal. Si con esto bajan las mermas, disminuye el tiempo de registro y sube la exactitud del stock, la solución será escalable, intuitiva y eficiente, alineada con lo que se recogió en las entrevistas.

2.3. Needfinding

En el siguiente apartado, analizaremos a nuestros segmentos objetivos para identificar sus necesidades y en base a esto ofrecerles soluciones óptimas a sus problemas.

2.3.1. User Personas

Segmento 1: Bodegas especializadas por rubro

NOMBRE	TAMAÑO DEL MERCADO	TIPO								
Carla Rodríguez	 40 %	Guardian								
										
Metas <ul style="list-style-type: none"> Tener control total del stock de sus productos (evitar quiebres y sobrestock). Minimizar pérdidas por productos vencidos o mal almacenados. Optimizar la reposición de productos con alertas automáticas. Tomar decisiones basadas en reportes visuales (ventas, rotación, márgenes). Reducir el tiempo invertido en conteos físicos y registros manuales. 										
Frase <p>“Quiero hacer crecer mi negocio y agilizar todos los procesos posibles”</p>										
Trasfondo <p>Carla comenzó ayudando en el negocio familiar desde joven. Luego estudió una carrera técnica en Administración de Empresas con enfoque en logística. Hace más de 10 años decidió tomar las riendas de la bodega mayorista que hoy lidera, enfocada en distribución de productos de consumo masivo (alimentos, limpieza y bebidas) hacia tiendas minoristas y restaurantes.</p> <p>Ha aprendido a negociar precios, manejar inventarios grandes y liderar un pequeño equipo de operarios y asistentes administrativos. Aunque no es experta en tecnología, si está acostumbrada a usar herramientas como Excel, WhatsApp Business y apps bancarias para gestionar partes de su operación.</p>										
HABILIDADES <table border="1"> <tr> <td>Atención al cliente</td> <td>85</td> </tr> <tr> <td>Comunicación efectiva</td> <td>80</td> </tr> <tr> <td>Manejo del dinero y finanzas</td> <td>82</td> </tr> <tr> <td>Perseverancia</td> <td>84</td> </tr> </table>			Atención al cliente	85	Comunicación efectiva	80	Manejo del dinero y finanzas	82	Perseverancia	84
Atención al cliente	85									
Comunicación efectiva	80									
Manejo del dinero y finanzas	82									
Perseverancia	84									
Motivaciones <ul style="list-style-type: none"> Tener tiempo libre real para compartir con su familia: Le gustaría poder desconectarse del negocio después del horario de atención, simplemente descansar sin preocuparse por el inventario. Modernizar su bodega y sentirse profesionalmente realizada: Más allá de lo operativo, quiere sentir que su esfuerzo vale la pena y que su negocio está al nivel de empresas más grandes, usando herramientas modernas y eficientes. Tener mayor control sin depender de su presencia constante: Quiere que el negocio funcione aunque ella no esté, para poder delegar tareas con confianza y dedicar más tiempo a sí misma y su familia. Reducir el estrés del día a día: Busca simplificar su rutina y eliminar tareas repetitivas o manuales que la agotan, con el objetivo de sentirse más tranquila y con energía para otras áreas de su vida. 										
Frustraciones <ul style="list-style-type: none"> Carga operativa excesiva genera estrés laboral: Carla se siente sobrepasada por las tareas del negocio, ya que muchos procesos se hacen de forma manual y repetitiva, lo que le consume tiempo y energía diariamente. Depende demasiado de su presencia en el almacén: Si no está físicamente presente, el control del inventario se vuelve inexacto o desorganizado, y le cuesta delegar por falta de una herramienta confiable. Le cuesta entender reportes complicados o usar herramientas muy técnicas: Aunque domina lo básico de Excel, se le complica generar reportes detallados y visualizar tendencias en su inventario, lo que dificulta tomar decisiones informadas. Hace pedidos de forma reactiva y no planificada: Normalmente espera a que el stock esté muy bajo o se acabe para hacer pedidos, lo que genera retrasos en la atención de clientes y estrés logístico. 										

Imagen (Nº2). Elaboración propia. Realizado en UXPressia

Segmento 2: Startups y emprendedores en expansión con necesidades logísticas

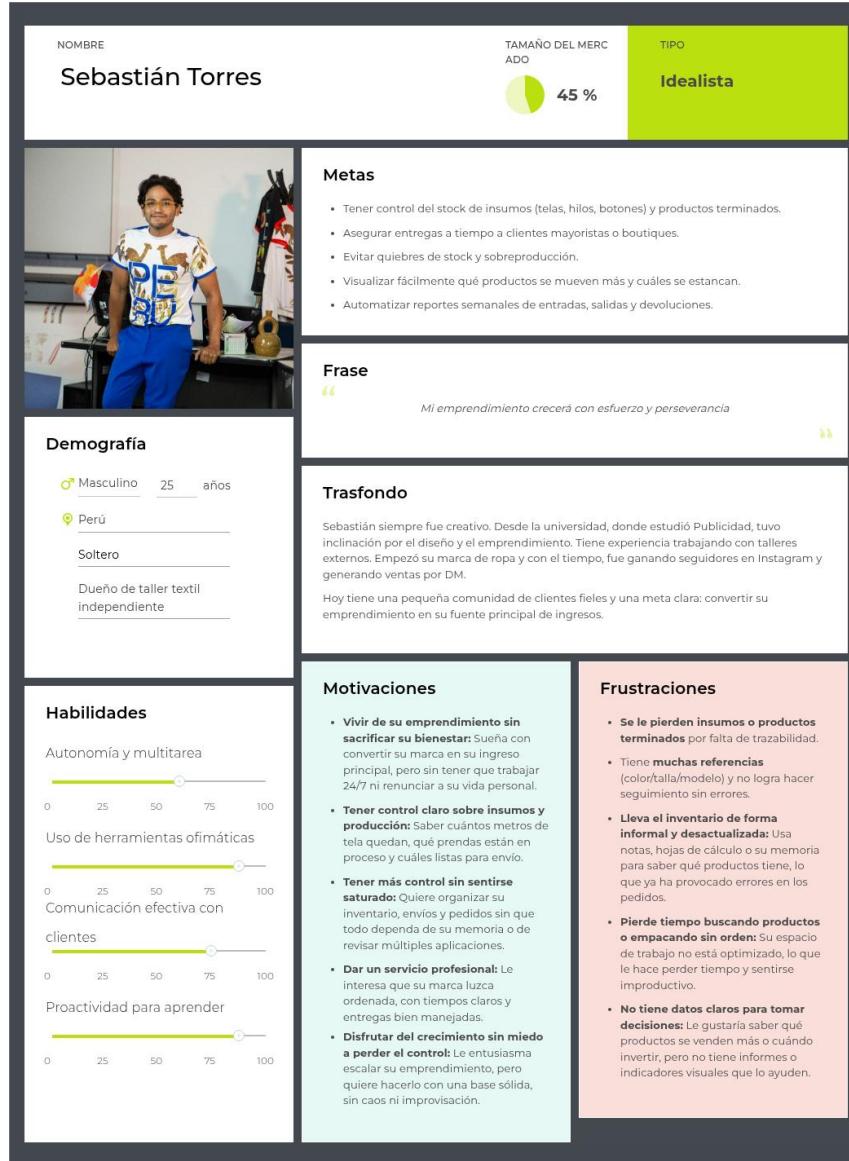


Imagen (Nº3). Elaboración propia. Realizado en UXPressia

2.3.2. User Task Matrix

Segmento 1: Bodegas especializadas por rubro

Task Matrix	Frecuencia	Importancia
Supervisar el stock y revisar niveles de inventario	Alta	Alta
Realizar conteos físicos o auditorías manuales	Media	Alta
Negociar precios y coordinar con proveedores	Alta	Alta
Revisar reportes de ventas, rotación y márgenes	Media	Alta
Ingresar datos en Excel o sistemas básicos de control	Media	Media
Delegar tareas a sus asistentes o empleados	Media	Alta
Atender clientes en tienda	Alta	Alta
Coordinar pedidos con mayoristas o distribuidores	Alta	Alta
Capacitarse en nuevas herramientas tecnológicas	Baja	Media

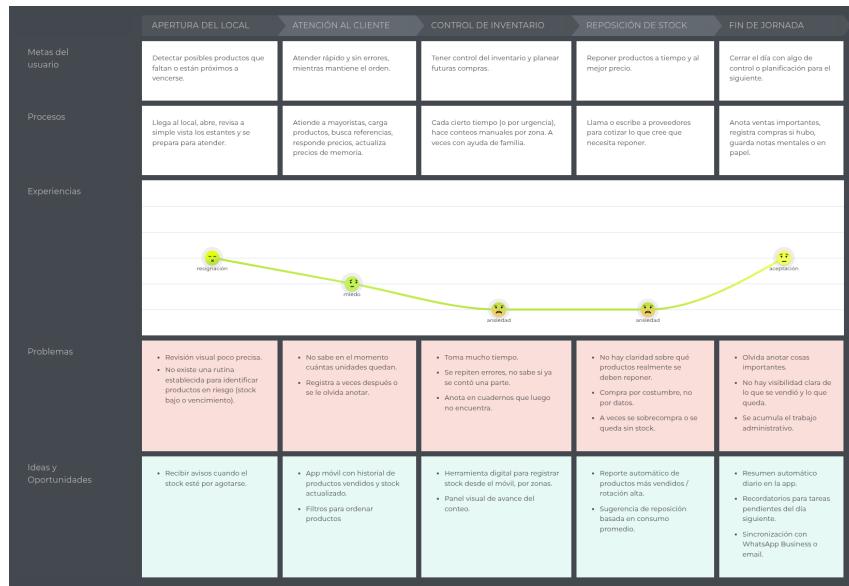
Task Matrix	Frecuencia	Importancia
Resolver errores de inventario (<i>sobrestock</i> , productos vencidos)	Alta	Alta
Registrar productos por voz y confirmar datos	Media	Alta
Escanear lotes con cámara rápida y validar	Alta	Alta
Ubicar productos en almacén con QR/mapa	Alta	Alta
Revisar alertas de predicción de stock	Media	Alta
Visualizar sedes y rutas en mapa (GPS)	Baja	Media

Segmento 2: Startups y emprendedores en expansión con necesidades logísticas

Task Matrix	Frecuencia	Importancia
Supervisar la producción (propia o con talleres externos)	Alta	Alta
Gestionar inventario de insumos y productos terminados (telas, hilos, prendas)	Alta	Alta
Registrar entradas, salidas y devoluciones de productos	Media	Alta
Coordinar pedidos con clientes o puntos de venta	Alta	Alta
Diseñar nuevas piezas o adaptar modelos a la demanda	Alta	Alta
Publicar contenido y gestionar redes sociales	Alta	Media
Empaquetar productos y organizar despachos	Media	Alta
Analizar qué productos se venden más y qué insumos se usan más	Media	Alta
Actualizar listas de precios, catálogos o colecciones	Media	Media
Registrar ventas y organizar información contable o administrativa	Media	Alta
Aprender herramientas nuevas para mejorar su operativa (apps, control, automatización)	Media	Alta
Dictar insumos o productos por voz	Alta	Alta
Registrar clientes y entregas con GPS	Alta	Alta
Analizar predicciones de producción/ventas	Media	Alta
Escanear productos en lotes y asignar ubicación	Alta	Alta
Recibir alertas de sobrestock o quiebre	Media	Alta

2.3.3. User Journey Mapping

Segmento 1: Bodegas especializadas por rubro



Segmento 2: Startups y emprendedores en expansión con necesidades logísticas

	APERTURA DEL LOCAL	ATENCIÓN AL CLIENTE	CONTROL DE INVENTARIO	REPÓSITO DE STOCK	FIN DE JORNADA
Metas del usuario	Detectar posibles productos que faltan o están próximos a vencerse.	Atender rápido y sin errores, mientras mantiene el orden.	Tener control del inventario y planear futuras compras.	Reponer productos a tiempo y al mejor precio.	Cerrar el día con algo de control o planeación para el siguiente.
Procesos	Llega al local, abre, revisa a simple vista los existentes y se prepara para atender.	Atiende a clientes, carga productos, busca referencias, responde preguntas, actualiza precios de memoria.	Cada cierto tiempo (o por urgencia), hace conteos manuales por zona. A veces con ayuda de familia.	Llama o escribe a proveedores para cotizar lo que cree que necesita reponer.	Anota ventas importantes, registra compras si hubo; guarda notas mentales o en papel.
Experiencias					
Problemas	<ul style="list-style-type: none"> Revisión visual poco precisa. No existe una rutina establecida para identificar productos en riesgo (stock bajo o vencimiento). 	<ul style="list-style-type: none"> No sabe en el momento cuántas unidades quedan. Registra a veces después o se le olvida anotar. 	<ul style="list-style-type: none"> Toma mucho tiempo. Se repiten errores, no sabe si ya se contó una parte. Registra a veces después o se le olvida anotar. 	<ul style="list-style-type: none"> No hay claridad sobre qué productos realmente se deben reponer. Compra por costumbre, no por datos. A veces se sobrecompra o se queda sin stock. 	<ul style="list-style-type: none"> Obliga a mudar cosas importantes. No hay visibilidad clara de lo que se vendió y lo que queda. A veces se sobrecompra o se queda sin stock. Se acumula el trabajo administrativo.
Ideas y Oportunidades	<ul style="list-style-type: none"> Recibir avisos cuando el stock esté en agotarse. 	<ul style="list-style-type: none"> App móvil con historial de productos vendidos y stock actualizado. Filtros para ordenar productos 	<ul style="list-style-type: none"> Herramienta digital para registrar stock desde el móvil, por zonas. Panel visual de avance del conteo. 	<ul style="list-style-type: none"> Reporte automático de productos más vendidos / rotación alta. Sugerencia de reposición basada en consumo promedio. 	<ul style="list-style-type: none"> Resumen automático directo en la app. Recordatorios para tareas pendientes del día siguiente. Sincronización con WhatsApp Business o email.

2.3.4. Empathy Mapping

Segmento 1: Bodegas especializadas por rubro

<p>1.¿Con quién estamos empatizando?</p> <p>Carla es una mujer peruana que es gerente de una tienda mayorista la cual desea organizarse mejor para hacer crecer su negocio y llevar un mejor balance en todos los aspectos de su vida.</p>	<p>7.¿Qué es lo que piensa y siente?</p> <p>“ Estoy cansada de hacer todo a mano, pero tampoco quiero perder el control. Mi trabajo no termina cuando cierro la tienda. Me llevo el estrés a casa.”</p>	<p>2.¿Qué es lo que necesita hacer?</p> <p>Estar dispuesta a aceptar cambios para el progreso de su bodega. Informarse de nuevas tecnologías para no quedarse atrás.</p>
<p>6.¿Qué es lo que escucha?</p> <p>Recomendaciones de amigos comerciantes que le hablan de apps para controlar ventas o inventario. Opiniones de su familia sobre lo mucho que trabaja y lo poco que descansa.</p>		<p>3.¿Qué es lo que ve?</p> <p>Un mercado competitivo, con negocios similares que ofrecen precios agresivos. Que otros negocios usan herramientas tecnológicas, pero le parecen muy complicadas.</p>
<p>5.¿Qué es lo que hace?</p> <p>Reconoce los puntos débiles de su gestión. Busca nuevas tecnologías y las compara para elegir la que más se adapte a sus necesidad.</p>		
<p>Dolores</p> <p>¿Cuáles son sus frustraciones y miedos? Tareas repetitivas que consumen tiempo valioso. No tener información clara para tomar decisiones. Dependencia de su presencia física para que todo funcione.</p>	<p>Motivaciones</p> <p>¿Cuáles son sus necesidades, esperanzas y deseos? Automatizar procesos sin perder el control. Tomar decisiones informadas (con reportes visuales, simples). Sentirse profesionalizada y al nivel de empresas grandes.</p>	<p>4.¿Qué es lo que dice?</p> <p>“ Tengo que seguir esforzándome por darle un servicio de calidad a mis clientes sin sacrificar mi tiempo libre.”</p>

Imagen (Nº6). Elaboración propia. Realizado en UXPressia

Segmento 2: Startups y emprendedores en expansión con necesidades logísticas

<p>1.¿Con quién estamos empatizando?</p> <p>Sebastián es el fundador de su propio emprendimiento digital el cual es un una tienda de ropa en redes. Quiero expandirme para escalar más rápido y posicionar su negocio entre los mejores.</p>	<p>7.¿Qué es lo que piensa y siente?</p> <p>“Siento que pierdo dinero en materiales y tiempo. Estoy cansado de repetir los mismos errores por no tener control.”</p>	<p>2.¿Qué es lo que necesita hacer?</p> <p>Reconocer los puntos débiles en la logística de su emprendimiento para mejorarlo. Informarse acerca de nuevas tecnologías.</p>
<p>6.¿Qué es lo que escucha?</p> <p>A talleristas diciendo: "Ya entregué todo, revisa tu lista". A su entorno recomendándole que digitalice su operación.</p>		<p>3.¿Qué es lo que ve?</p> <p>Sus estantes llenos, pero sin saber qué está realmente disponible. A sus talleres o trabajadores haciendo su parte sin mucha supervisión.</p>
<p>5.¿Qué es lo que hace?</p> <p>Elije las tecnologías que se adaptan a sus necesidades y trabaja para suplir con las fallas en su gestión que reconoció antes.</p>		
<p>Dolores</p> <p>¿Cuáles son sus frustraciones y miedos?</p> <p>Pérdidas de materiales sin explicación clara.</p> <p>Errores en entregas por falta de trazabilidad.</p> <p>Falta de información para planificar o comprar inteligentemente.</p>	<p>Motivaciones</p> <p>¿Cuáles son sus necesidades, esperanzas y deseos?</p> <p>Tener control real del negocio, sin depender 100% de su presencia.</p> <p>Reducir errores y estrés operativo.</p> <p>Escalar su empresa con procesos profesionales, sin perder tiempo ni creatividad.</p>	<p>4.¿Qué es lo que dice?</p> <p>“No puedo seguir así, tengo que organizarme por mí y mi negocio.”</p>

Imagen (Nº7). Elaboración propia. Realizado en UXPressia

2.3.5. Ubiquitous Language

El siguiente glosario presenta los términos clave utilizados a lo largo del desarrollo del proyecto StockWise. Este lenguaje común busca asegurar que todos los miembros del equipo (tanto técnicos como no técnicos) compartan una comprensión unificada de los conceptos centrales del sistema, facilitando así la comunicación y el diseño colaborativo.

Palabra	Descripción
Landing Page	Página principal de presentación al llegar al sitio oficial de StockWise.
Inventario	Conjunto de productos almacenados en una bodega con detalles como cantidad, ubicación y fecha de vencimiento.
Producto	Artículo registrado en la plataforma con nombre, descripción, stock mínimo, unidades, fecha de vencimiento y categoría.
Stock	Cantidad disponible de un producto específico en el inventario.
Stock bajo	Estado que indica que la cantidad de un producto está por debajo del mínimo definido por el usuario.
Panel de Control	Interfaz principal para visualizar métricas, movimientos y reportes del inventario.
Movimiento de Inventario	Registro de cambios en cantidades de productos por compras, ventas, pérdidas o ajustes.

Palabra	Descripción
Reporte	Documento visual o estadístico que resume información clave del inventario para la toma de decisiones.
Bodega	Negocio pequeño con venta de productos de primera necesidad, como alimentos y bebidas.
Usuario Administrador	Persona con acceso total que puede gestionar usuarios, configuraciones y datos.
Emprendedor	Usuario en etapa de crecimiento que busca profesionalizar la gestión de su negocio.
Dueña de bodega	Usuario con experiencia comercial que desea optimizar la gestión operativa con herramientas digitales.
Versión Freemium	Modelo que permite acceso gratuito a funciones básicas y pago por funciones premium.
Onboarding	Proceso de bienvenida y guía para enseñar a nuevos usuarios a usar la plataforma.
MVP (Producto Mínimo Viable)	Versión inicial con funciones mínimas para validar hipótesis clave con usuarios reales.

2.4. Requirements specification

El siguiente glosario presenta los términos clave utilizados a lo largo del desarrollo del proyecto StockWise. Este lenguaje común busca asegurar que todos los miembros del equipo (tanto técnicos como no técnicos) comparten una comprensión unificada de los conceptos centrales del sistema, facilitando así la comunicación y el diseño colaborativo.

1. **Landing Page:** Página de presentación a la que llegan los usuarios cuando visitan el sitio oficial de stockwise.
2. **Inventario:** Conjunto de productos almacenados en una bodega. Incluye información como cantidad disponible, ubicación, fecha de vencimiento, entre otros.
3. **Producto:** Bien que se registra en la plataforma, con atributos específicos como nombre, descripción, stock mínimo, unidades, fecha de vencimiento, y categoría.
4. **Stock:** Cantidad disponible de un producto exacto en el inventario
5. **Stock bajo:** Estado de un producto que ha superado la mínima cantidad definida por el usuario.
6. **Panel de Control:** Interfaz principal de la plataforma donde el usuario puede visualizar métricas clave del inventario, movimientos y reportes
7. **Movimiento de Inventario:** Registro de cualquier cambio en la cantidad de productos: entrada (nueva compra), salida (venta o pérdida), ajuste manual, etc.
8. **Reporte:** Documento visual o estadístico que resume información clave del inventario
9. **Bodega:** Negocio pequeño donde se venden productos de primera necesidad, como alimentos y bebidas.
10. **Usuario Administrador:** Persona que tiene acceso total a la plataforma, puede crear usuarios, modificar configuraciones y visualizar todos los datos.
11. **Emprendedor:** Usuario que está en proceso de expansión de su negocio, y busca profesionalizar su gestión
12. **Dueña de bodega:** Usuario generalmente con experiencia en el rubro comercial que busca optimizar su gestión operativa con herramientas digitales
13. **Versión Freemium:** Modelo de negocio que permite a los usuarios acceder a funciones básicas de forma gratuita, con la posibilidad de pagar por funciones premium.
14. **Onboarding:** Proceso de bienvenida y guía para nuevos usuarios que les enseña cómo usar la plataforma y aprovechar sus funcionalidades.
15. **MVP (Producto Mínimo Viable):** Versión inicial de la plataforma con las funciones mínimas necesarias para validar hipótesis clave con usuarios reales.
16. **Registro por Voz:** Funcionalidad que permite ingresar datos dictando productos, cantidades o insumos, validados por el sistema.
17. **Escaneo QR:** Proceso de registrar productos o lotes mediante el uso de lector, facilitando la velocidad de captura.
18. **Mapa/GPS:** Herramienta que permite ubicar bodegas, almacenes o rutas de distribución, y registrar entregas.
19. **Alertas de Stock:** Notificaciones automáticas que advierten sobre riesgos de quiebre, sobrestock o vencimiento de productos.
20. **Rotación de Productos:** Indicador que mide la velocidad de venta o uso de un producto en el inventario.
21. **Gestión de Producción:** Capacidad de registrar y controlar procesos productivos.
22. **Contabilidad Básica:** Registro y organización de ventas, costos y márgenes, integrados a reportes financieros simples.

2.4.1. User Stories

Las User Stories son clave en metodologías ágiles porque traducen los requisitos funcionales desde la mirada del usuario. Cada historia especifica una necesidad concreta, lo que permite planificar, priorizar y construir el sistema de forma iterativa. Así se asegura que cada función aporte valor real y permanezca alineada con las expectativas del usuario final.

Story ID	User	Priority	Epic
US01	Usuario de inventario	Alta	EP01 - Funciones básicas de inventario

Title Registrar producto nuevo Description Como usuario, quiero registrar un nuevo producto en mi inventario, para tener un control detallado de su existencia desde el comienzo. Acceptance Criteria

Escenario 01: Registro exitoso de producto

- Dado que el sistema recibe datos válidos para un producto,
- Cuando procesa el registro,
- Entonces guarda el producto en la base de datos
- Y lo incluye en el inventario.

Escenario 02: Interno de registro en campos vacíos

- Dado que el sistema recibe un producto con campos requeridos vacíos,
- Cuando intenta registrar el producto,
- Entonces el sistema rechaza la operación
- Y muestra mensajes de validación sobre los campos faltantes.

Story ID	User	Priority	Epic
US02	Usuario de inventario	Alta	EP01 - Funciones básicas de inventario

Title Editar información de producto Description Como usuario, quiero editar la información de un producto registrado, para corregir o actualizar datos.

Acceptance Criteria

Escenario 01: Edición con éxito

- **Dado que** el usuario ya tiene un producto en el inventario
- **Cuando** modifica campos de información del producto
- **Y** procesa el registro adecuadamente
- **Entonces** el sistema guarda los cambios

Escenario 02: Edición con datos inválidos

- **Dado que** el usuario está editando información de un producto
- **Cuando** ingrese datos inválidos
- **Entonces** el sistema le muestra los errores
- **Y** no se guardan los cambios

Story ID	User	Priority	Epic
US03	Usuario de inventario	Alta	EP01 - Funciones básicas de inventario

Title Registrar salida de producto Description Como usuario, quiero registrar la salida de productos del inventario, para mantener actualizado el stock en tiempo real. Acceptance Criteria

Escenario 01: Registro válido de salida de stock

- **Dado que** un producto tiene suficiente stock disponible
- **Cuando** el sistema recibe una salida con cantidad permitida,
- **Entonces** descuenta la cantidad del stock
- **Y** registra el movimiento.

Escenario 02: Intento de salida mayor al stock disponible

- **Dado que** que el stock actual es menor que la cantidad solicitada
- **Cuando** valida la salida
- **Entonces** rechaza la operación
- **Y** expone el motivo "cantidad supera stock disponible"

Story ID	User	Priority	Epic
US04	Usuario de inventario	Alta	EP01 – Funciones básicas de inventario
Title			
Ver historial de movimientos			
Description			
Como usuario, quiero ver el historial de entrada y salida de mis productos, para rastrear los cambios y movimientos en el inventario.			
Acceptance Criteria			
Escenario 01: Visualización del historial completo			
<ul style="list-style-type: none"> • Dado que el usuario está en la sección historial de un producto • Cuando este producto tiene movimientos registrados • Entonces el sistema muestra una lista con toda la información del producto 			
Escenario 02: Producto sin historial de movimientos			

- Dado que el usuario accede al historial de un producto nuevo o sin registro
- Cuando no hay datos previos
- Entonces el sistema muestra un mensaje indicando que no hay movimientos registrados

Story ID	User	Priority	Epic			
US05	Usuario de inventario	Alta	EP01 - Funciones básicas de inventario			
Title	Generar alertas por bajo stock					
Description						
Como usuario, quiero recibir alertas cuando un producto está debajo del stock mínimo, para reabastecerme a tiempo.						
Acceptance Criteria						
Escenario 01: Activación automática de alerta por stock bajo						
<ul style="list-style-type: none"> • Dado que un producto tiene un límite mínimo configurado • Cuando su stock baja por debajo del mínimo • Entonces el sistema genera una alerta visual y notifica al usuario 						
Escenario 02: Personalización de la alerta						
<ul style="list-style-type: none"> • Dado que el usuario desea configurar alertas específicas • Cuando accede a la edición del producto y define un nuevo valor de stock mínimo • Entonces el sistema guarda la alerta y la activa con el nuevo valor 						

Story ID	User	Priority	Epic			
US06	Usuario de inventario	Alta	EP01 - Funciones básicas de inventario			
Title	Buscar productos en inventarios					
Description						
Como usuario, quiero buscar productos en el inventario por nombre o categoría, para encontrarlo de manera rápida y fácil.						
Acceptance Criteria						
Escenario 01: Búsqueda por nombre o código						
<ul style="list-style-type: none"> • Dado que el usuario está en la sección "Inventario" • Cuando ingresa el nombre o código de un producto en el buscador • Entonces se muestran los productos que coinciden con los términos de búsqueda 						
Escenario 02: Búsqueda sin coincidencias						
<ul style="list-style-type: none"> • Dado que el usuario realiza una búsqueda con término no registrado • Cuando busca el producto • Entonces el sistema muestra el mensaje "No se encontraron resultados" 						

Story ID	User	Priority	Epic			
US07	Usuario de inventario		EP01 - Funciones básicas de inventario			
Title	Añadir etiquetas para productos					
Description						
Como usuario, quiero poder clasificar los productos mediante etiquetas, para organizar el stock y agilizar búsquedas.						
Acceptance Criteria						
Escenario 01: Añadir etiqueta correctamente						
<ul style="list-style-type: none"> • Dado que el usuario está en el formulario de "Nuevo producto" • Cuando escribe una o varias etiquetas personalizadas y lo guarda exitosamente • Entonces las etiquetas se registran con el producto y son visibles en su detalle 						
Escenario 02: Usar etiquetas existentes						

- Dado que el usuario comienza a escribir una etiqueta
- Cuando ya existe una similar en el sistema
- Entonces el sistema sugiere etiquetas existentes para evitar duplicados

Escenario 03: Filtrar por etiquetas

- Dado que el usuario está en el módulo de inventario
- Cuando selecciona una etiqueta desde el filtro
- Entonces se muestran solo los productos que contienen esa etiqueta

Story ID	User	Priority	Epic		
US08	Usuario de inventario	Media	EP02 - Funciones de analítica, reportes y control avanzado		
Title		Description			
Como usuario, quiero ver un listado con los productos más vendidos durante un periodo de tiempo, para identificar los de mayor demanda y planificar mejor.					
Acceptance Criteria					
Escenario 01: Ver productos más vendidos en un periodo					
<ul style="list-style-type: none"> • Dado que el usuario accede al módulo de estadísticas • Cuando selecciona un rango de fechas • Entonces el sistema muestra un ranking de productos más vendidos y permite ordenarlos por unidades o ingresos 					
Escenario 02: Exportar reporte					
<ul style="list-style-type: none"> • Dado que el usuario visualiza el listado • Cuando solicita la exportación • Entonces el sistema genera un archivo PDF con el ranking 					

Story ID	User	Priority	Epic		
US09	Usuario de inventario	Media	EP02 - Funciones de analítica, reportes y control avanzado		
Title		Description			
Como usuario, quiero visualizar qué categorías o líneas de productos son las que más se venden, para saber cuál promocionar o reducir del catálogo.					
Acceptance Criteria					
Escenario 01: Ver categorías más vendidas					
<ul style="list-style-type: none"> • Dado que el usuario accede al módulo de estadísticas • Cuando selecciona un rango de fechas • Entonces se muestra un gráfico con la participación de cada categoría en las ventas 					
Escenario 02: Ver detalle de productos por categoría					
<ul style="list-style-type: none"> • Dado que el usuario visualiza las categorías más vendidas • Cuando busca en categoría • Entonces se muestra la lista con los productos que la componen y sus ventas 					

Story ID	User	Priority	Epic		
US10	Usuario de inventario	Media	EP02 - Funciones de analítica, reportes y control avanzado		
Title		Description			
Como usuario, quiero conocer el ticket promedio de mis ventas para entender cuánto gasta en promedio cada cliente y evaluar la efectividad de los precios.					
Acceptance Criteria					

Escenario 01: Visualizar ticket promedio

- Dado que el usuario accede al módulo de estadísticas
- Cuando selecciona un rango de fechas
- Entonces se muestra el ticket promedio, el total de ingresos y el número de ventas

Escenario 02: Comparar ticket con periodo anterior

- Dado que el usuario ve el ticket promedio actual
- Cuando habilita la opción de comparación
- Entonces el sistema muestra la variación respecto al periodo anterior

Story ID	User	Priority	Epic			
US11	Usuario de inventario	Media	EP02 - Funciones de analítica, reportes y control avanzado			
Title	Generar estadística de stock promedio					
Description						
Como usuario, quiero visualizar el promedio de stock disponible para identificar niveles óptimos y evitar quiebres o exceso de stock.						
Acceptance Criteria						
Escenario 01: Visualizar stock promedio						
<ul style="list-style-type: none">• Dado que el usuario accede al módulo de estadísticas• Cuando selecciona un producto o categoría y un rango de fechas• Entonces se muestra el promedio de stock disponible en ese periodo						
Escenario 02: Detectar stock por fuera del rango óptimo						
<ul style="list-style-type: none">• Dado que el usuario ve el reporte de stock promedio• Cuando el valor está por debajo o por encima del stock ideal• Entonces el sistema resalta los productos con riesgo de quiebre o sobrestock y puede sugerir acciones						

Story ID	User	Priority	Epic			
US12	Usuario de inventario	Media	EP02 - Funciones de analítica, reportes y control avanzado			
Title	Generar reportes de inventario					
Description						
Como usuario, quiero generar reportes detallados del inventario para analizar el estado general y tomar decisiones informadas.						
Acceptance Criteria						
Escenario 01: Generar reporte completo						
<ul style="list-style-type: none">• Dado que el usuario accede al módulo de reportes• Cuando selecciona parámetros como fecha, información, categorías o stock mínimo• Entonces el sistema genera un reporte con esa información						
Escenario 02: Exportar reporte						
<ul style="list-style-type: none">• Dado que el usuario visualiza el reporte generado• Cuando hace clic en "Exportar"• Entonces el sistema descarga el reporte en PDF o Excel						

Story ID	User	Priority	Epic			
US13	Usuario de inventario	Media	EP02 - Funciones de analítica, reportes y control avanzado			
Title	Controlar productos caducados					
Description						
Como usuario, quiero identificar productos próximos a caducar o ya caducados, para evitar pérdidas y mantener la calidad del inventario.						
Acceptance Criteria						

Escenario 01: Visualizar productos próximos a caducar

- Dado que el usuario accede al módulo de inventario
- Cuando filtra por fecha de caducidad próxima
- Entonces el sistema muestra una lista con los productos que caducan pronto

Escenario 02: Alerta de productos caducados

- Dado que un producto está caducado
- Cuando el usuario accede al inventario
- Entonces el sistema muestra una alerta visual destacando esos productos

Story ID	User	Priority	Epic			
US14	Usuario de inventario	Media	EP02 - Funciones de analítica, reportes y control avanzado			
Title	Registrar devolución de productos					
Description						
Como usuario, quiero registrar devoluciones de productos al inventario, para mantener el stock actualizado y controlar incidencias.						
Acceptance Criteria						
Escenario 01: Registro de devolución válida						
<ul style="list-style-type: none">• Dado que un producto fue devuelto por un cliente• Cuando el usuario ingresa la cantidad devuelta y la razón y confirma• Entonces el sistema incrementa el stock y guarda el movimiento						
Escenario 02: Devolución con cantidad inválida						
<ul style="list-style-type: none">• Dado que el usuario intenta registrar una devolución con cantidad negativa o mayor a la vendida• Cuando hace clic en "Guardar"• Entonces el sistema muestra un error y no registra la devolución						

Story ID	User	Priority	Epic			
US15	Usuario de inventario	Media	EP02 - Funciones de analítica, reportes y control avanzado			
Title	Configurar roles y permisos de usuario					
Description						
Como usuario, quiero definir roles y permisos para los usuarios del sistema.						
Acceptance Criteria						
Escenario 01: Crear nuevo rol con permisos específicos						
<ul style="list-style-type: none">• Dado que el administrador accede a la configuración de usuarios• Cuando crea un nuevo rol y asigna permisos• Entonces el rol se guarda y puede ser asignado						
Escenario 02: Asignar rol a usuario						
<ul style="list-style-type: none">• Dado que el administrador selecciona un usuario existente• Cuando asigna un rol específico• Entonces el sistema muestra los permisos del rol asignado						

Story ID	User	Priority	Epic			
US16	Usuario de inventario	Alta	EP01 - Funciones básicas de inventario			
Title	Gestionar proveedores					
Description						
Como usuario, quiero registrar y gestionar la información de proveedores, para mantener contactos actualizados y facilitar la reposición.						

Acceptance Criteria

Escenario 01: Registro exitoso de proveedor

- Dado que el usuario está en “Nuevo proveedor”
- Cuando completa los campos obligatorios y guarda
- Entonces el proveedor se registra y aparece en la lista

Escenario 02: Edición de datos de proveedor

- Dado que el usuario selecciona un proveedor existente
- Cuando modifica la información y guarda
- Entonces el sistema actualiza los datos del proveedor

Story ID	User	Priority	Epic			
US17	Usuario de inventario	Alta	EP01 - Funciones básicas de inventario			
Title	Registrar lote de productos					
Description						
Como usuario, quiero registrar productos por lote, para controlar el ingreso masivo con un mismo proveedor, fecha y condiciones.						
Acceptance Criteria						
Escenario 01: Registro de lote con múltiples productos						
<ul style="list-style-type: none"> • Dado que el usuario accede a “Nuevo lote” • Cuando añade varios productos con fecha, proveedor y condiciones • Entonces el sistema guarda el lote y vincula los productos 						
Escenario 02: Visualizar lote registrado						
<ul style="list-style-type: none"> • Dado que el usuario visualiza la lista de productos • Cuando selecciona “Ver lote” • Entonces se muestra la información común del lote 						

Story ID	User	Priority	Epic			
US18	Usuario de inventario	Media	EP02 - Funciones de analítica, reportes y control avanzado			
Title	Visualizar historial de stock					
Description						
Como usuario, quiero ver el historial de stock de un producto, para analizar su comportamiento y tomar mejores decisiones de compra.						
Acceptance Criteria						
Escenario 01: Visualizar historial por producto						
<ul style="list-style-type: none"> • Dado que el usuario accede a la ficha de un producto • Cuando selecciona “Ver historial” • Entonces se muestra una gráfica con variaciones de stock en el tiempo 						
Escenario 02: Filtro por periodo						
<ul style="list-style-type: none"> • Dado que el usuario visualiza el historial de stock • Cuando aplica un filtro por rango de fechas • Entonces la gráfica se actualiza con el periodo seleccionado 						

Story ID	User	Priority	Epic			
US19	Usuario de inventario	Media	EP02 - Funciones de analítica, reportes y control avanzado			
Title	Configurar alertas de stock mínimo					
Description						
Como usuario, quiero configurar alertas cuando un producto alcance su stock mínimo, para poder reabastecerme a tiempo.						
Acceptance Criteria						

Acceptance Criteria			
Escenario 01: Configurar stock mínimo			
<ul style="list-style-type: none"> • Dado que el usuario accede a la configuración de productos • Cuando establece un valor de stock mínimo por producto • Entonces el sistema guarda ese valor para alertas 			
Escenario 02: Alerta de bajo stock			
<ul style="list-style-type: none"> • Dado que un producto llega o baja del stock mínimo configurado • Cuando el usuario accede al panel o inventario • Entonces se muestra una alerta visual indicando que se debe reponer 			

Story ID	User	Priority	Epic			
US20	Usuario de inventario	Baja	EP03 - Innovación y valor agregado			
Title	Añadir etiquetas a productos					
Description						
Como usuario, quiero añadir etiquetas personalizadas a los productos, para facilitar su clasificación y búsqueda.						
Acceptance Criteria						
Escenario 01: Etiquetado desde el formulario del producto						
<ul style="list-style-type: none"> • Dado que el usuario está registrando o editando un producto • Cuando escribe nuevas etiquetas o selecciona existentes • Entonces las etiquetas se asocian al producto guardado 						
Escenario 02: Buscar productos por etiqueta						
<ul style="list-style-type: none"> • Dado que el usuario está en el módulo de búsqueda o lista • Cuando ingresa el nombre de una etiqueta • Entonces se filtran los productos asociados a esa etiqueta 						

Story ID	User	Priority	Epic			
US21	Usuario de inventario	Alta	EP01 - Funciones básicas de inventario			
Title	Buscar productos en inventario					
Description						
Como usuario, quiero buscar productos en el inventario por nombre o categoría, para encontrarlos de forma rápida y eficiente.						
Acceptance Criteria						
Escenario 01: Búsqueda por nombre						
<ul style="list-style-type: none"> • Dado que el usuario accede al buscador • Cuando ingresa parte del nombre del producto • Entonces se muestran resultados coincidentes en tiempo real 						
Escenario 02: Búsqueda por categoría						
<ul style="list-style-type: none"> • Dado que el usuario está en el buscador • Cuando selecciona una categoría del filtro • Entonces se muestran solo productos de esa categoría 						

Story ID	User	Priority	Epic
US22	Usuario de inventario	Baja	EP03 - Innovación y valor agregado
Title	Generar alertas por productos caducados		
Description			

Como usuario, quiero recibir alertas automáticas de productos caducados, para poder retirarlos del inventario a tiempo.

Acceptance Criteria

Escenario 01: Activar alertas automáticas

- Dado que el usuario accede a configuración de alertas
- Cuando activa la opción de alerta por caducidad
- Entonces el sistema enviará notificaciones cuando haya productos vencidos

Escenario 02: Visualización de productos vencidos

- Dado que hay productos caducados en el inventario
- Cuando el usuario accede al sistema
- Entonces se muestra un panel o lista resaltando estos productos

Story ID	User	Priority	Epic
US23	Usuario de inventario	Alta	EP01 - Funciones básicas de inventario
Title	Registrar ingreso de productos		

Description

Como usuario, quiero registrar el ingreso de productos al inventario, para mantener actualizado el stock disponible.

Acceptance Criteria

Escenario 01: Ingreso individual de producto

- Dado que el usuario accede al formulario de ingreso
- Cuando registra los datos del producto y la cantidad recibida
- Entonces el sistema actualiza el stock del producto

Escenario 02: Ingreso masivo por archivo

- Dado que el usuario cuenta con un archivo de productos
- Cuando carga el archivo en el sistema
- Entonces se actualiza el inventario con los productos del archivo

Story ID	User	Priority	Epic
US24	Usuario de inventario	Media	EP02 - Funciones de analítica, reportes y control avanzado
Title	Visualizar resumen de stock por categoría		

Description

Como usuario, quiero ver un resumen del stock agrupado por categoría, para conocer el estado general del inventario de forma ordenada.

Acceptance Criteria

Escenario 01: Visualización del resumen por categoría

- Dado que el usuario accede al módulo de inventario
- Cuando selecciona "Resumen por categoría"
- Entonces se muestra una tabla con cada categoría, cantidad total de productos y stock promedio

Escenario 02: Acceso al detalle desde el resumen

- Dado que el usuario visualiza el resumen por categoría
- Cuando hace clic en una categoría
- Entonces se despliega la lista de productos que la conforman

Story ID	User	Priority	Epic
US25	Usuario de inventario	Media	EP02 - Funciones de analítica, reportes y control avanzado
Title	Comparar ventas entre períodos		

Description

Como usuario, quiero comparar las ventas entre dos períodos, para identificar tendencias y evaluar el impacto de promociones.

Acceptance Criteria

Escenario 01: Comparación de períodos seleccionados

- Dado que el usuario accede al módulo de estadísticas
- Cuando selecciona dos períodos a comparar
- Entonces se muestra una gráfica con la diferencia de ventas

Escenario 02: Comparación por categoría o producto

- Dado que el usuario visualiza la comparación general
- Cuando filtra por categoría o producto
- Entonces la gráfica se actualiza con los datos seleccionados

Story ID	User	Priority	Epic
US26	Usuario de inventario	Baja	EP03 - Innovación y valor agregado
Title	Registrar productos por voz		

Description

Como usuario, quiero registrar productos mediante comandos de voz (ej. "Agregar 20 botellas de agua al inventario"), para ganar rapidez y comodidad cuando no pueda usar el teclado.

Acceptance Criteria

Escenario 01: Comando válido

- Dado que el usuario activa el modo voz,
- Cuando dicta "Agregar <cantidad> <producto> al inventario",
- Entonces el sistema reconoce cantidad y producto, muestra un resumen y al confirmar registra el movimiento.

Escenario 02: Ambigüedad o error

- Dado que el audio no es claro o hay varias coincidencias,
- Cuando el sistema no puede identificar el producto,
- Entonces solicita confirmación/sugerencias y permite corregir o escribir manualmente.

Escenario 03: Dictado continuo

- Dado que el usuario permanece en modo voz,
- Cuando dicta varios comandos consecutivos,
- Entonces el sistema procesa cada ítem y permite "cancelar" o "terminar" por voz.

Story ID	User	Priority	Epic
US27	Usuario de inventario	Baja	EP03 - Innovación y valor agregado
Title	Registrar geolocalización de productos y entregas		

Description

Como usuario, quiero integrar GPS para registrar procedencia, puntos de entrega y visualizar en un mapa las sedes vinculadas a la tienda principal, para optimizar trazabilidad y distribución.

Acceptance Criteria

Escenario 01: Captura de ubicación

- Dado que el usuario registra un movimiento/entrega,
- Cuando concede permisos de ubicación,
- Entonces el sistema guarda latitud/longitud y muestra un pin en el mapa del producto o entrega.

Escenario 02: Mapa de sedes

- Dado que el usuario abre el mapa,
- Cuando selecciona "Sedes y rutas",
- Entonces visualiza marcadores por sede y puede filtrar por estado.

Escenario 03: Sin permisos

- Dado que el usuario deniega GPS,
- Cuando intenta registrar ubicación,
- Entonces el sistema muestra aviso y permite ingresar dirección manual.

Story ID	User	Priority	Epic			
US28	Usuario de inventario	Baja	EP03 - Innovación y valor agregado			
Title	Localizar productos con mapa y QR					
Description						
Como usuario, quiero ubicar productos dentro del almacén usando un mapa interactivo e integración de códigos QR, para encontrarlos con precisión.						
Acceptance Criteria						
Escenario 01: Asociar ubicación por QR <ul style="list-style-type: none"> • Dado que el usuario escanea un QR de estantería/pasillo, • Cuando guarda la ubicación en la ficha del producto, • Entonces el sistema vincula el producto a ese punto del mapa. 						
Escenario 02: Búsqueda y ruta <ul style="list-style-type: none"> • Dado que el usuario busca un producto, • Cuando selecciona "Ver ubicación", • Entonces el mapa señala pasillo/estante y puede mostrar ruta sugerida. 						
Escenario 03: QR inválido <ul style="list-style-type: none"> • Dado que el QR no existe o está dañado, • Cuando intenta asociarlo, • Entonces el sistema alerta y permite ingresar ubicación manual. 						

Story ID	User	Priority	Epic			
US29	Usuario de inventario	Baja	EP03 - Innovación y valor agregado			
Title	Gestionar predicción de reabastecimiento					
Description						
Como usuario, quiero recibir sugerencias de cuándo y cuánto reponer según patrones de ventas y stock mínimo, para evitar quiebres y sobrestock.						
Acceptance Criteria						
Escenario 01: Sugerencia de compra <ul style="list-style-type: none"> • Dado que el sistema analiza histórico y lead time, • Cuando el stock proyectado cae bajo el nivel objetivo, • Entonces propone fecha y cantidad de reabastecimiento y permite aceptarla/editarla. 						
Escenario 02: Alerta de quiebre inminente <ul style="list-style-type: none"> • Dado que el consumo supera la previsión, • Cuando se detecta riesgo de quiebre, • Entonces se muestra alerta prioritaria y recomendaciones de acción. 						
Escenario 03: Parámetros ajustables <ul style="list-style-type: none"> • Dado que el usuario abre "Predicción", • Cuando ajusta stock mínimo, tiempo de reposición y estacionalidad, • Entonces el modelo recalcula y actualiza las sugerencias. 						

Story ID	User	Priority	Epic
US30	Usuario de inventario	Baja	EP03 - Innovación y valor agregado

Title	Escanear por lotes con cámara rápida	Description
Como usuario, quiero registrar lotes tomando una foto del producto/lote para que una API de visión devuelva etiquetas genéricas; luego confirmar producto exacto y cantidad y ver su ubicación en el almacén virtual.		
Acceptance Criteria		
Escenario 01: Registro por visión		
<ul style="list-style-type: none"> Dado que el usuario abre "Escaneo rápido", Cuando toma una foto del lote, Entonces la API (p.ej., ML Kit) devuelve etiquetas; el usuario elige el producto, ingresa cantidad y confirma el registro del lote. 		
Escenario 02: Etiquetas ambiguas		
<ul style="list-style-type: none"> Dado que la API devuelve varias opciones o no encuentra coincidencias, Cuando el usuario revisa sugerencias, Entonces puede seleccionar/corregir manualmente y continuar sin perder el flujo. 		
Escenario 03: Ubicación en almacén virtual		
<ul style="list-style-type: none"> Dado que el lote quedó registrado, Cuando el usuario finaliza, Entonces el sistema muestra o permite elegir la ubicación del lote en el mapa del almacén. 		

Technical Stories

En esta sección se describen las historias técnicas que desarrollamos para implementar las funcionalidades clave. Cada historia define tareas específicas que el equipo de desarrollo debe realizar, como crear endpoints, manejar validaciones, controlar el stock, generar reportes, entre otros.

Story ID	User	Priority	Epic			
TS01	Developer	Alta	EP04 - Cuentas y seguridad			
Title	API Registro de usuarios					
Description						
Como desarrollador, necesito exponer endpoints para registrar nuevos usuarios, asegurando validaciones y persistencia correcta en el backend.						
Acceptance Criteria						
Escenario 01: Registro exitoso						
Dado que el endpoint de registro está disponible y la app valida email/contraseña, Cuando el usuario envía datos válidos desde el móvil, Entonces el backend crea la cuenta (201), retorna token y la app lo guarda de forma segura (Keychain/Keystore).						
Escenario 02: Datos inválidos o duplicados						
Dado el endpoint de registro, Cuando el email ya existe o los campos no cumplen las reglas (formato/fortaleza), Entonces se responde 400/409 con mensaje claro y la app muestra los errores sin duplicar el envío.						

Story ID	User	Priority	Epic			
TS02	Developer	Alta	EP04 - Cuentas y seguridad			
Title	API Autenticación de usuarios					
Description						
Como desarrollador, necesito implementar endpoints de autenticación para que la app móvil inicie sesión de forma segura (emisión y validación de tokens).						
Acceptance Criteria						
Escenario 01: Login válido						
Dado el endpoint de autenticación, Cuando se envían credenciales válidas desde la app, Entonces se retorna 200 con access/refresh token y datos mínimos del perfil; la app inicia sesión persistente y puede habilitar biometría.						
Escenario 02: Login inválido						

Dado el endpoint de autenticación,
 Cuando las credenciales son incorrectas o el usuario está inactivo,
 Entonces se retorna 401/403 con mensaje adecuado y la app ofrece "recuperar contraseña".

Story ID	User	Priority	Epic			
TS03	Developer	Alta	EP01 - Funciones básicas de inventario			
Title	API Creación de productos					
Description						
Como desarrollador, necesito exponer endpoints para registrar productos con validaciones de campos requeridos y unicidad.						
Acceptance Criteria						
Escenario 01: Producto creado Dado que la app permite capturar foto/código de barras, Cuando se envían datos correctos desde el móvil, Entonces el backend crea el producto (201) y la app sincroniza la imagen en segundo plano.						
Escenario 02: Datos inválidos Dado el endpoint de productos, Cuando el código está duplicado o faltan campos requeridos, Entonces se retorna 400/409 con detalle y la app resalta los campos a corregir.						

Story ID	User	Priority	Epic			
TS04	Developer	Alta	EP01 - Funciones básicas de inventario			
Title	API Consulta de productos					
Description						
Como desarrollador, necesito listar productos con soporte de paginación y orden para su visualización en la app móvil.						
Acceptance Criteria						
Escenario 01: Lista disponible Dado que la app lista con "infinite scroll", Cuando consulta el endpoint, Entonces se retorna colección paginada (200) mostrando placeholders y usando caché local si no hay conexión.						
Escenario 02: Sin registros Dado el endpoint de consulta, Cuando no existen productos, Entonces se retorna 200 con lista vacía y la app muestra estado vacío con CTA para crear.						

Story ID	User	Priority	Epic			
TS05	Developer	Alta	EP01 - Funciones básicas de inventario			
Title	API Productos por categoría					
Description						
Como desarrollador, necesito filtrar productos por categoría para agilizar búsquedas desde la app.						
Acceptance Criteria						
Escenario 01: Categoría existente Dado que el usuario selecciona una categoría (chips/filtros), Cuando se consulta el endpoint con la categoría, Entonces se retorna 200 con resultados ignorando mayúsculas/acentos y la app recuerda el filtro.						
Escenario 02: Sin coincidencias Dado el filtro por categoría, Cuando no hay productos asociados, Entonces se retorna 200 con lista vacía y la app permite limpiar filtros.						

Story ID	User	Priority	Epic

TS06	Developer	Alta	EP01 - Funciones básicas de inventario			
Title	API Productos por etiqueta					
Description						
Como desarrollador, necesito permitir la consulta de productos por etiqueta para mejorar la organización y búsqueda.						
Acceptance Criteria						
<p>Escenario 01: Etiqueta válida Dado que la app ofrece autocompletado de etiquetas recientes, Cuando el usuario selecciona una etiqueta y consulta, Entonces el backend retorna los productos (200) y la app permite combinar con otros filtros activos.</p> <p>Escenario 02: Etiqueta sin resultados Dado que se consulta por una etiqueta inexistente o sin datos, Cuando el backend no encuentra coincidencias, Entonces responde 200 con lista vacía y la app sugiere etiquetas relacionadas.</p>						

Story ID	User	Priority	Epic			
TS07	Developer	Alta	EP01 - Funciones básicas de inventario			
Title	API Registro de inventario					
Description						
Como desarrollador, necesito endpoints para registrar movimientos/stock de inventario con validaciones de negocio.						
Acceptance Criteria						
<p>Escenario 01: Registro válido Dado que la app permite escanear código/QR, Cuando se registra una entrada o salida con cantidad válida, Entonces el backend guarda el movimiento (201) y retorna el stock actualizado para mostrar en la app.</p> <p>Escenario 02: Registro inválido Dado el endpoint de inventario, Cuando la cantidad es negativa o la salida supera el stock, Entonces se retorna 400 con detalle y la app resalta el campo con error.</p>						

Story ID	User	Priority	Epic			
TS08	Developer	Alta	EP02 - Funciones de analítica, reportes y control avanzado			
Title	API Inventarios por precio					
Description						
Como desarrollador, necesito consultar inventarios filtrados por rango de precio para análisis desde la app.						
Acceptance Criteria						
<p>Escenario 01: Rango válido Dado que el usuario ajusta un rango (slider), Cuando se consulta el endpoint con la moneda/locale del dispositivo, Entonces se retornan resultados (200) y pueden ordenarse por precio asc/desc.</p> <p>Escenario 02: Sin resultados Dado un rango sin coincidencias, Cuando el backend no encuentra registros, Entonces retorna 200 con lista vacía y la app sugiere ampliar el rango.</p>						

Story ID	User	Priority	Epic			
TS09	Developer	Alta	EP02 - Funciones de analítica, reportes y control avanzado			
Title	API Inventario por producto					
Description						
Como desarrollador, necesito consultar el inventario de un producto específico para mostrar su disponibilidad.						

Acceptance Criteria

Escenario 01: Producto existente

Dado que la app escanea o selecciona un producto,
 Cuando se consulta por su ID,
 Entonces el backend retorna 200 con stock actual y últimos movimientos.

Escenario 02: Producto inexistente

Dado el endpoint por producto,
 Cuando el ID no existe o está inactivo,
 Entonces se retorna 404 con mensaje "no encontrado" y la app ofrece buscar otro producto.

Story ID	User	Priority	Epic
TS10	Developer	Alta	EP02 - Funciones de analítica, reportes y control avanzado
Title			API Inventarios por fecha de ingreso

Description

Como desarrollador, necesito filtrar inventarios por fecha de ingreso para consultas cronológicas desde la app.

Acceptance Criteria

Escenario 01: Fecha con registros

Dado que el usuario selecciona un rango con el date picker del móvil,
 Cuando la app consulta con la zona horaria del dispositivo,
 Entonces el backend retorna 200 con los registros del período.

Escenario 02: Fecha sin registros

Dado el endpoint por fecha,
 Cuando el rango no tiene datos,
 Entonces se retorna 200 con lista vacía y la app propone rangos cercanos.

Story ID	User	Priority	Epic
TS11	Developer	Alta	EP02 - Funciones de analítica, reportes y control avanzado
Title			API Inventarios por cantidad

Description

Como desarrollador, necesito consultar inventarios por cantidad exacta o rango para monitorear niveles críticos o excedentes.

Acceptance Criteria

Escenario 01: Rango/cantidad válida

Dado que la app permite filtrar por cantidad o rango,
 Cuando se consulta el endpoint,
 Entonces se retornan productos (200) e indicadores de "bajo stock" si están bajo el mínimo configurado.

Escenario 02: Sin coincidencias

Dado el filtro por cantidad/rango,
 Cuando no hay resultados,
 Entonces se retorna 200 con lista vacía y la app permite guardar el filtro.

Story ID	User	Priority	Epic
TS12	Developer	Alta	EP02 - Funciones de analítica, reportes y control avanzado
Title			API Inventarios por proveedor

Description

Como desarrollador, necesito consultar inventarios asociados a un proveedor para análisis y gestión.

Acceptance Criteria

Escenario 01: Proveedor existente

Dado que el usuario selecciona un proveedor en la app,
 Cuando se consulta el endpoint con su identificador,
 Entonces retorna 200 con la lista paginada de inventarios.

Escenario 02: Proveedor sin productos

Dado el endpoint por proveedor,
 Cuando no hay inventarios vinculados,
 Entonces se retorna 200 con lista vacía y la app muestra un estado informativo.

Story ID	User	Priority	Epic		
TS13	Developer	Alta	EP05 - Alertas y notificaciones		
Title		API Registro de alertas			
Description					
Como desarrollador, necesito registrar alertas del sistema (p.ej., bajo stock) para su notificación en la app móvil.					
Acceptance Criteria					
Escenario 01: Alerta registrada Dado un evento de bajo stock detectado en la app o backend, Cuando se envía la alerta con datos válidos al endpoint, Entonces se registra (201) y la app puede disparar notificación local/push.					
Escenario 02: Datos inválidos Dado el endpoint de alertas, Cuando el tipo es inválido o faltan campos requeridos, Entonces se retorna 400 con detalle de validaciones.					

Story ID	User	Priority	Epic		
TS14	Developer	Alta	EP05 - Alertas y notificaciones		
Title		API Consulta de alertas			
Description					
Como desarrollador, necesito listar alertas para su visualización y seguimiento en la aplicación móvil.					
Acceptance Criteria					
Escenario 01: Listado disponible Dado que la app muestra un feed con "pull to refresh", Cuando consulta el endpoint, Entonces se retornan alertas paginadas (200) ordenadas por severidad/fecha y se pueden marcar como leídas.					
Escenario 02: Sin alertas Dado el endpoint de consulta de alertas, Cuando no existen registros, Entonces se retorna 200 con lista vacía y la app ofrece ir a configuración de alertas.					

Story ID	User	Priority	Epic		
TS15	Developer	Alta	EP01 - Funciones básicas de inventario		
Title		API Registro de combos			
Description					
Como desarrollador, necesito endpoints para registrar combos de productos respetando reglas de composición.					
Acceptance Criteria					
Escenario 01: Combo registrado Dado que el usuario arma un combo en la app seleccionando productos y cantidades, Cuando envía la creación, Entonces el backend valida reglas (stock mínimo, no repetidos) y crea el combo (201) de forma átomica.					
Escenario 02: Datos inválidos Dado el endpoint de combos, Cuando la composición es inválida o hay productos inexistentes, Entonces se retorna 400 con detalle y la app conserva la selección para corregir.					

Story ID	User	Priority	Epic			
TS16	Developer	Alta	EP01 - Funciones básicas de inventario			
Title	API Consulta de combos					
Description						
Como desarrollador, necesito consultar combos registrados para mostrarlos como opciones agrupadas en la app móvil.						
Acceptance Criteria						
Escenario 01: Listado de combos Dado que la app lista combos con imágenes y precio total, Cuando consulta el endpoint, Entonces el backend retorna 200 con lista paginada y la app permite ordenar y guardar favoritos.						
Escenario 02: Sin combos Dado el endpoint de combos, Cuando no existen registros, Entonces se retorna 200 con lista vacía y la app muestra un estado sin datos con botón para crear combo.						

Spike Story

Story ID	User	Priority	Epic			
SPK01	Equipo de desarrollo	Baja	EP03 - Innovación y valor agregado			
Title	Implementar entrada por voz, comandos estructurados e interpretación					
Description						
Como equipo de desarrollo, investiga gramática mínima y extracción de entidades (cantidad, producto) para registrar movimientos por voz con manejo de ambigüedad.						
Acceptance Criteria						
Escenario 01: Extracción de entidades con confianza mínima <ul style="list-style-type: none"> Dado que se recibe un comando válido, Cuando procesa audio o texto, Entonces extrae cantidad y referencia de producto con un umbral de confianza definido. 						
Escenario 02: Manejo de ambigüedad <ul style="list-style-type: none"> Dado que existen múltiples coincidencias, Cuando detecta ambigüedad, Entonces solicita desambiguación dentro del flujo técnico. 						
Escenario 03: Dictado consecutivo <ul style="list-style-type: none"> Dado que el usuario dicta varios comandos, Cuando los recibe en secuencia, Entonces procesa cada uno y conserva estado hasta terminación. 						

Story ID	User	Priority	Epic			
SPK02	Equipo de desarrollo	Alta	EP04 - Cuentas y seguridad			
Title	Roles y permisos en API					
Description						
Como equipo de desarrollo, investiga un modelo RBAC con matriz rol permiso y puntos de control en API para operaciones de inventario y administración.						
Acceptance Criteria						
Escenario 01: Autorización según política <ul style="list-style-type: none"> Dado que un rol tiene permisos definidos, Cuando solicita una operación restringida, 						

- Entonces autoriza o rechaza de acuerdo con la política.

Escenario 02: Intentos no autorizados

- Dado que un usuario carece del permiso,
- Cuando intenta modificar recursos,
- Entonces rechaza la acción y registra el intento.

Escenario 03: Efecto de cambios de rol

- Dado que se actualizan las asignaciones de rol,
- Cuando el cambio entra en vigor,
- Entonces aplica permisos vigentes sin inconsistencias.

Story ID	User	Priority	Epic			
SPK03	Equipo de desarrollo	Baja	EP03 - Innovación y valor agregado			
Title	Geolocalizar y modelar la ubicación (GPS y QR)					
Description						
Como equipo de desarrollo, investiga captura de lat/long, precisión, almacenamiento, permisos y asociación de ubicación por QR para productos y entregas.						
Acceptance Criteria						
Escenario 01: Persistencia de coordenadas						
<ul style="list-style-type: none"> Dado que los permisos de ubicación están concedidos, Cuando captura la ubicación, Entonces persiste coordenadas con precisión y timestamp. 						
Escenario 02: Flujo sin permisos						
<ul style="list-style-type: none"> Dado que los permisos son denegados, Cuando se requiere ubicación, Entonces documenta flujo de entrada manual de dirección o coordenadas. 						
Escenario 03: Asociación por QR						
<ul style="list-style-type: none"> Dado que se dispone de un QR válido, Cuando asocia la ubicación, Entonces vincula el producto al punto definido y lo hace consultable. 						

Story ID	User	Priority	Epic			
SPK04	Equipo de desarrollo	Baja	EP03 - Innovación y valor agregado			
Title	Configurar predicción de reabastecimiento					
Description						
Como equipo de desarrollo, investiga una fórmula base (media móvil + lead time) con parámetros ajustables para sugerir fecha y cantidad de reabastecimiento.						
Acceptance Criteria						
Escenario 01: Sugerencia con supuestos explícitos						
<ul style="list-style-type: none"> Dado que existe histórico de ventas y stock mínimo, Cuando calcula demanda proyectada, Entonces sugiere fecha y cantidad con supuestos explícitos. 						
Escenario 02: Recálculo por parámetros						
<ul style="list-style-type: none"> Dado que los parámetros son ajustables, Cuando actualiza valores, Entonces recálcula sugerencias con trazabilidad de versiones. 						
Escenario 03: Detección de quiebre inminente						

- Dado que el consumo supera la previsión,
- Cuando se detecta desvío relevante,
- Entonces marca riesgo de quiebre y registra recomendación prioritaria.

Story ID	User	Priority	Epic		
SPK05	Equipo de desarrollo	Baja	EP03 - Innovación y valor agregado		
Title		Description			
La app toma una foto al lote; una API de visión (p.ej., Google ML Kit) devuelve etiquetas genéricas. El usuario confirma producto exacto y cantidad antes de registrar; se puede saltar a su ubicación en el almacén virtual.					
Acceptance Criteria					
Escenario 01: Etiquetado y confirmación					
<ul style="list-style-type: none"> • Dado una imagen de lote, • Cuando el servicio devuelve top-N etiquetas, • Entonces el usuario selecciona producto y define cantidad en un paso. 					
Escenario 02: Registro por lotes					
<ul style="list-style-type: none"> • Dado múltiples fotos consecutivas, • Cuando procesa cada captura, • Entonces registra movimientos en cola con feedback de éxito/error por ítem. 					
Escenario 03: Acceso a ubicación					
<ul style="list-style-type: none"> • Dado un producto confirmado, • Cuando el usuario lo solicita, • Entonces navega a su ubicación (QR o GPS) en el mapa del almacén. 					

2.4.2. Impact Mapping

Segmento Objetivo 1: Bodegas especializadas por rubro

IMPACT MAP: stockwise - impact map



UXPRESSIA

This map was built in uxpresa.com

Imagen (Nº12). Elaboración propia. Realizado en UXPRESSIA

Segmento Objetivo 2: Startups y emprendedores en expansión con necesidades logísticas

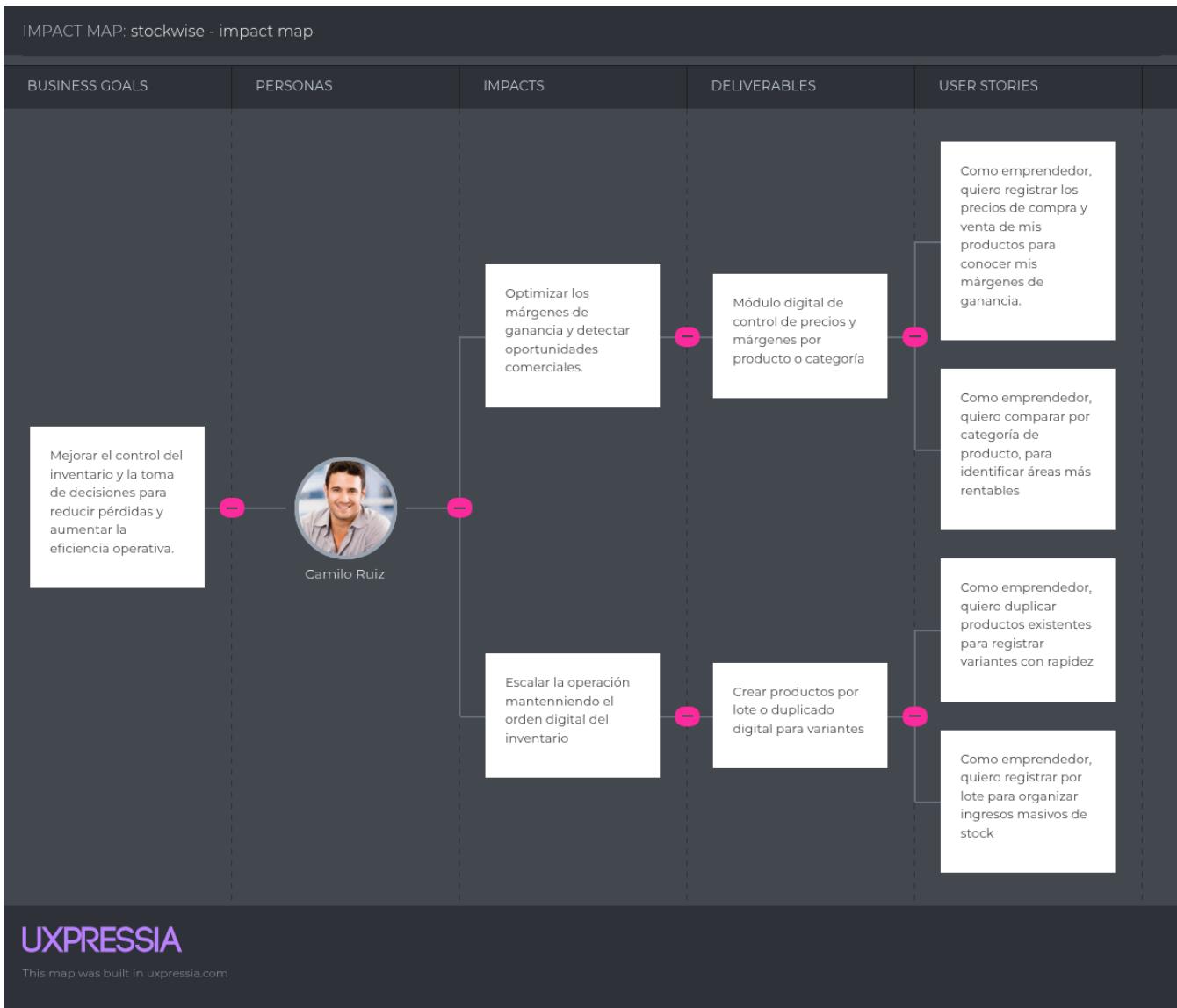


Imagen (Nº13). Elaboración propia. Realizado en UXPRESSIA

2.4.3. Product Backlog

#	User Story ID	Título	Story Points
1	US01	Registrar producto nuevo	5
2	US05	Generar alertas por bajo stock	5
3	US07	Añadir etiquetas para productos	5
4	US08	Generar estadística de productos más vendidos	5
5	US09	Generar estadística de categoría más vendida	5
6	US11	Generar estadística de stock promedio	5
7	US15	Configurar roles y permisos	5
8	US17	Registrar lote de productos	5
9	US20	Añadir etiquetas a productos	5
10	US26	Registrar productos por voz	5
11	US27	Registrar geolocalización de productos y entregas	5
12	US28	Localizar productos con mapa y QR	5
13	US02	Editar información de producto	3
14	US03	Registrar salida de producto	3

#	User Story ID	Título	Story Points
15	US04	Ver historial de movimientos	3
16	US06	Buscar productos en inventario	3
17	US10	Generar ticket promedio	3
18	US12	Generar reportes de inventario	3
19	US13	Controlar productos caducados	3
20	US14	Registrar devolución de productos	3
21	US16	Gestionar proveedores	3
22	US18	Visualizar historial de stock	3
23	US19	Configurar alertas de stock mínimo	3
24	US21	Buscar productos en inventario	3
25	US22	Generar alertas por productos caducados	3
26	US23	Registrar ingreso de productos	3
27	US25	Comparar ventas entre periodos	3
28	US29	Gestionar predicción de reabastecimiento	3
29	TS03	Generar API Creación de productos	3
30	TS06	Generar API Productos por etiqueta	3
31	SPK01	Implementar entrada por voz, comandos estructurados e interpretación	3
32	SPK03	Geolocalizar y modelar la ubicación	3
33	SPK04	Configurar predicción de reabastecimiento	3
34	TS07	Generar API Registro de inventario	3
35	US30	Escanear por lotes con cámara rápida	3
36	TS14	Generar API Consulta de alertas	3
37	US24	Visualizar resumen de stock por categoría	2

Gestión del producto Backlog

The screenshot shows a Trello board titled "Stoq - Product Backlog". The board is organized into five columns:

- Product Backlog:** Contains stories like US16: Gestionar proveedores (EP01, 3 points), US21: Buscar productos en inventario (EP01, 3 points), US23: Registrar ingreso de productos (EP01, 3 points), and US25: Comparar ventas entre periodos (3 points, EP02).
- EP01 - Funciones básicas de inventario:** Contains stories like US05: Generar alertas por bajo stock (EP01, 5 points), US01: Registrar producto nuevo (EP01, 5 points), US07: Añadir etiquetas para productos (EP01, 5 points), US17: Registrar lote de productos (EP01, 5 points), and US18: Visualizar historial de stock (EP01, 3 points).
- EP02 - Funciones de analítica, reportes y control avanzado:** Contains stories like US08: Generar estadística de productos más vendidos (5 points, EP02), US09: Generar estadística de categoría más vendida (5 points, EP02), US11: Generar estadística de stock promedio (5 points, EP02), and US18: Visualizar historial de stock (3 points, EP02).
- EP03 - Innovación y valor agregado:** Contains stories like US20: Añadir etiquetas a productos (5 points, EP03), US26: Registrar productos por voz (5 points, EP03), US27: Registrar geolocalización de productos y entregas (5 points, EP03), and US30: Escanear por lotes con cámara rápida (3 points, EP03).
- EP05 - Alertas y notificaciones:** Contains stories like TS14: Generar API Consulta de alertas (3 points, EP05).

At the bottom of the board, there are navigation buttons: "Bandeja de entrada", "Planificador", "Tablero", and "Cambiar de tablero".

Trello del Product Backlog: <https://trello.com/invite/b/68d9c7c26edf04fc488ca657/ATTIe2ace23b94bb268e8de0b90039ed6ef756FAD28C/stoq-product-backlog>

2.5. Strategic-Level Domain-Driven Design

En esta sección se describe el proceso utilizado en las decisiones del negocio, aplicando el enfoque de Domain Driven Design. A través de esta metodología, se busca identificar y definir los distintos Bounded Contexts que conforman el sistema, es decir, subconjuntos con límites lógicos y naturales en función del dominio del problema. Para lograrlo, el equipo emplea herramientas como EventStorming y el Bounded Context Canvas, las cuales permiten visualizar, analizar y delimitar las responsabilidades, flujos de eventos y relaciones clave dentro del sistema.

2.5.1. EventStorming

En este apartado se documenta el proceso del Event Storming, una herramienta clave para construir una primera aproximación al modelado general del dominio del negocio. Esta técnica colaborativa permite identificar eventos relevantes, actors, commands y aggregates dentro del sistema. Con ello, el objetivo principal es explorar nuestro dominio para descubrir patrones de software relevantes, límites y conexiones claras entre nuestros bounded context y sentar las bases para una buena arquitectura. Para ello, se realizaron los siguientes pasos en la herramienta Lucidchart bajo la normativa del DDD Estratégico. https://miro.com/app/board/uXjVJlfQcmY=/?share_link_id=739758304264

2.5.1.1. Candidate Context Discovery

A continuación se detalla el proceso y la evolución final del event storming.

Step 1: Unstructured exploration

Primero tuvimos una lluvia de ideas en la cual nos propusimos identificar exhaustivamente todos los eventos que ocurrirán en nuestro aplicativo, para asegurarnos de abarcar una amplia gama de posibilidades y ofrecer una experiencia completa y satisfactoria a nuestros usuarios.

A continuación, se presentan una serie de figuras que evidencian este proceso de exploración inicial.

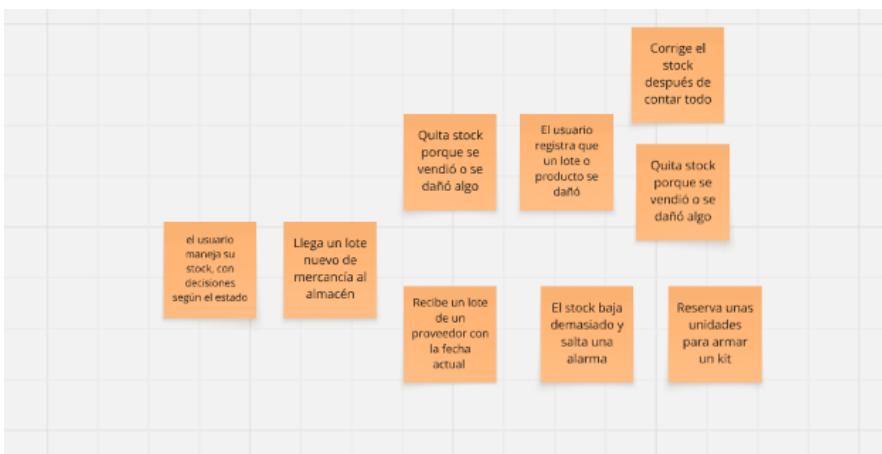
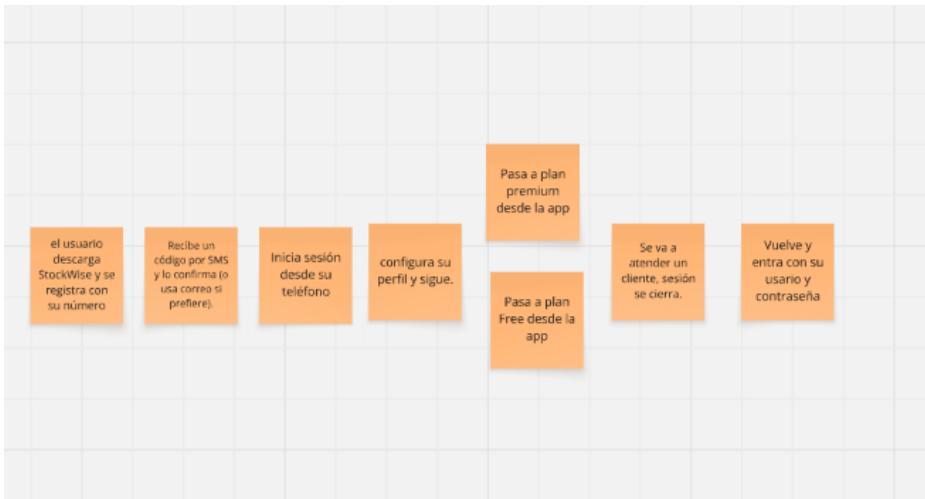
Unstructured exploration									
Alguien crea una cuenta nueva en la app	El usuario entra con su mail y contraseña.	Se cierra la sesión porque se va a almorzar.	Un jefe le da permisos de admin a un empleado.	El usuario cambia su foto de perfil o el idioma a inglés.	Pide resetear la contraseña porque la olvidó.	Termina de cambiar la contraseña porque ya puede entrar de nuevo.	Sube de plan gratis a uno pagado para más productos.	Baja de plan porque ya no necesita tanto.	La cuenta se duerme por no usarla en meses.
Agrega un producto nuevo, como "camisetas rojas".	Cambia el precio de ese producto porque subió el dólar.	Copia un producto para hacer una variante, tipo talla S.	Archiva un producto viejo que ya no vende.	Pone fecha de caducidad a un yogur que acaba de llegar.	Agrega una etiqueta como "alta demanda" al producto.	Quita una etiqueta porque ya no aplica.	Sube una foto del producto para que se vea mejor.	Cambia la categoría de "ropa" a "accesorios".	Ajusta el precio por una promo de fin de semana.
Llega un lote nuevo de mercancía al almacén.	Quita stock porque se vendió o se dañó algo.	Corrige el stock después de contar todo manualmente.	Recibe un lote de un proveedor con la fecha de hoy.	Un lote de comida se vence y hay que tirarlo.	Termina el conteo mensual y ve si hay errores.	El stock baja demasiado y salta una alarma.	Reserva unas unidades para armarse un kit.	Libera esa reserva porque al final no lo usa.	Toma una foto rápida del inventario actual para un reporte.
Salta una alerta porque el stock de leche está bajo.	Alerta de que algo se va a vencer en una semana.	Avisa que hay demasiado stock de algo que no se mueve.	El usuario dice "ok, vi la alerta, la marco como hecha".	Ignora una alerta porque ya lo maneja de otra forma.	Manda la alerta al jefe porque es grave.	Envía un mail o notificación al teléfono del usuario.	Cambia el límite de stock bajo de 10 a 5 unidades.	Procesa un montón de alertas al final del día.	Guarda el historial viejo de alertas en un archivo.
Arma un kit nuevo con camisetas y pantalones.	Cambia el kit agregando más de un producto.	Publica el kit para que lo vean en la tienda online.	Desarma el kit porque ya no lo quiere.	Chequea si hay stock suficiente para vender el kit.	Vende un kit y resta todo el stock de adentro.	Genera un reporte de qué se vendió esta semana.	Exporta el reporte a Excel para mandárselo al contador.	Registra una entrada o salida en el historial.	Captura números como "stock promedio" para ver tendencias.
El usuario confirma su correo tras registrarse, para activar su cuenta y empezar a usar la app	La sesión se cierra automáticamente tras un tiempo sin actividad, para proteger la cuenta.	El usuario pone un descuento temporal en un producto, como "20% off en algún producto".	El usuario define un stock mínimo o máximo para un producto, para personalizar alertas.	El usuario registra que un lote o producto se dañó (e.g., latas aplastadas), ajustando el inventario.	El usuario mueve stock de una bodega a otra (útil para startups con múltiples almacenes)	El usuario crea una alerta personalizada, como "avisar si el stock de un producto baja de 10".	El usuario marca una alerta como "urgente" para que salte primero en el dashboard.	El usuario crea un reporte que muestra tendencias de ventas (e.g., qué productos se mueven más).	El usuario envía el reporte por email o lo comparte con el equipo desde la app

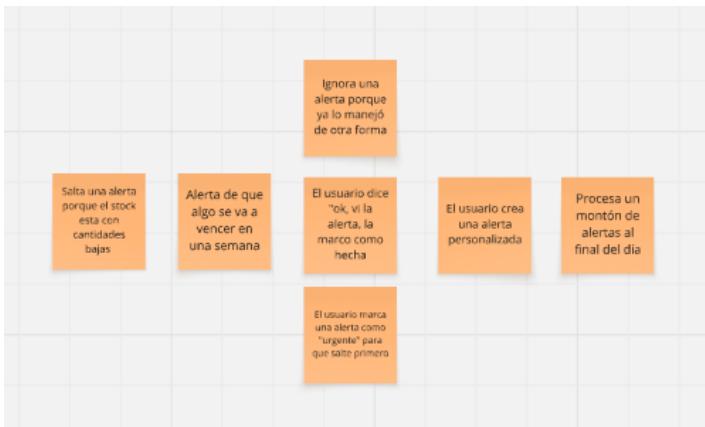
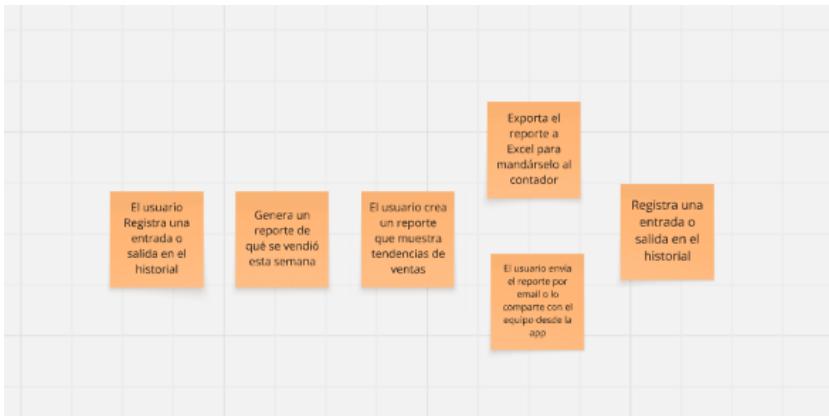
En esta figura se aprecia un mural con todos los eventos del dominio que fueron rescatados de la lluvia de ideas que realizó el equipo.

Step 2 : Timeline construction

Una vez identificados los eventos, organizamos la secuencia cronológica en la que estos ocurren dentro del flujo del sistema. Esto nos permitió identificar la secuencia natural del negocio y clarificar la narrativa del dominio.

A continuación, se presentan una serie de capturas de pantalla que ilustran cómo estructuramos esta línea temporal.

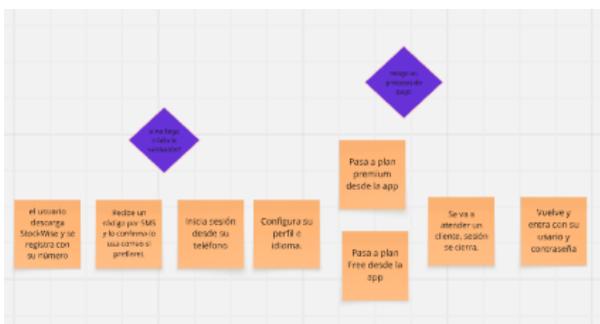




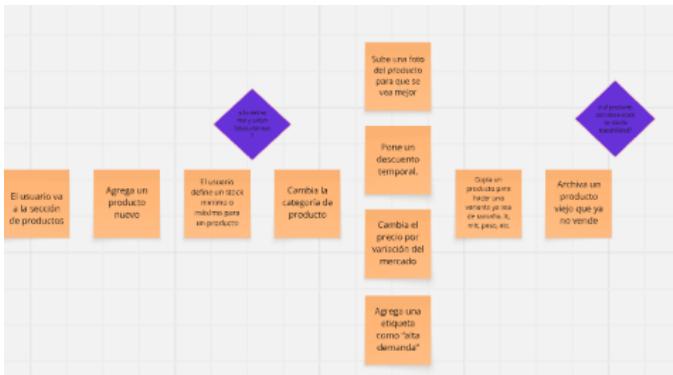
Step 3: Pain Points

En este paso identificamos los puntos críticos que requieren especial atención dentro de nuestro dominio, ya sea por su nivel de impacto en el negocio, riesgo potencial o complejidad. Para facilitar su análisis y visualización, se ha elaborado un diagrama colaborativo que refleja dichos puntos críticos.

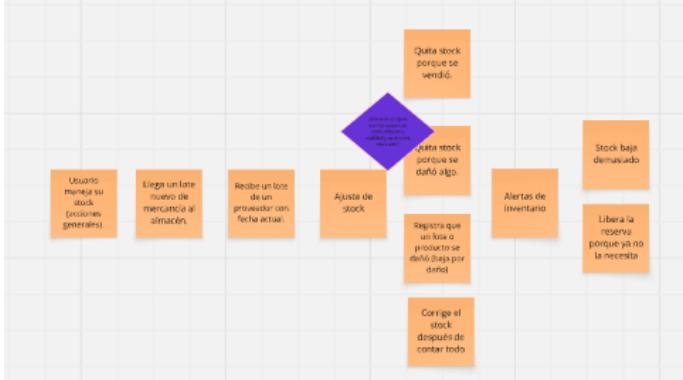
A continuación, se presentan una serie de figuras que evidencian el proceso realizado durante la identificación y análisis de los puntos críticos del dominio. Estas capturas complementan el diagrama general elaborado en Miro y permiten observar con mayor detalle los elementos clave identificados.



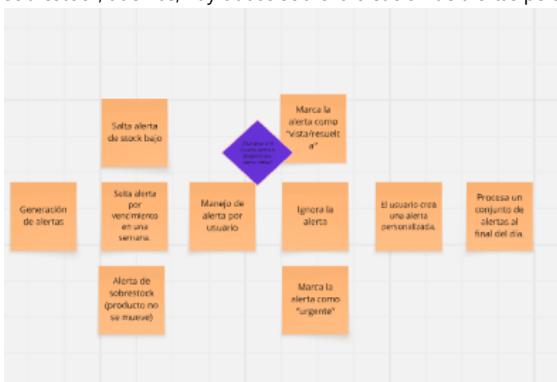
En esta imagen se aprecia la identificación de incertidumbres sobre cómo manejar la verificación (canal, reintentos, expiración y fallback) y cuándo activar la cuenta si el código no llega o falla. Al resolver estas interrogantes —reenvío OTP, cambio de canal, rate-limit y soporte— mejoraremos el alta de usuarios y reduciremos abandonos.



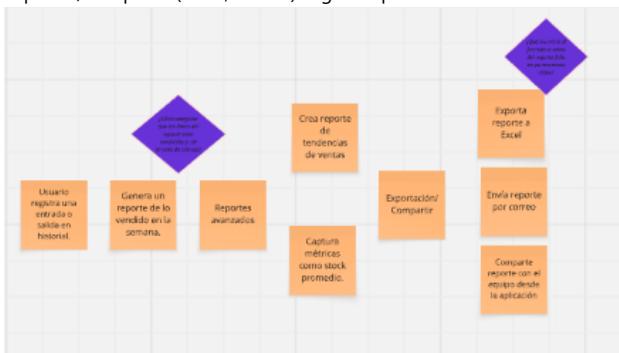
En esta imagen se aprecia la identificación de incertidumbres sobre de dónde salen los valores por defecto, si son obligatorios y cómo impactan las alertas. Al imponer validación y defaults por categoría, habilitamos reposición oportuna y alertas útiles sin fricción para el usuario.



En esta imagen se aprecia la identificación de incertidumbres sobre cómo cerrar una alerta: si se exige confirmación de atención o escalamiento previo, quién puede marcarla como "vista/resuelta", y cómo tratar estados como ignorada o urgente cuando hay alertas por stock bajo, vencimiento o sobrestock; además, hay dudas sobre la creación de alertas personalizadas y el procesamiento por lotes al final del día.



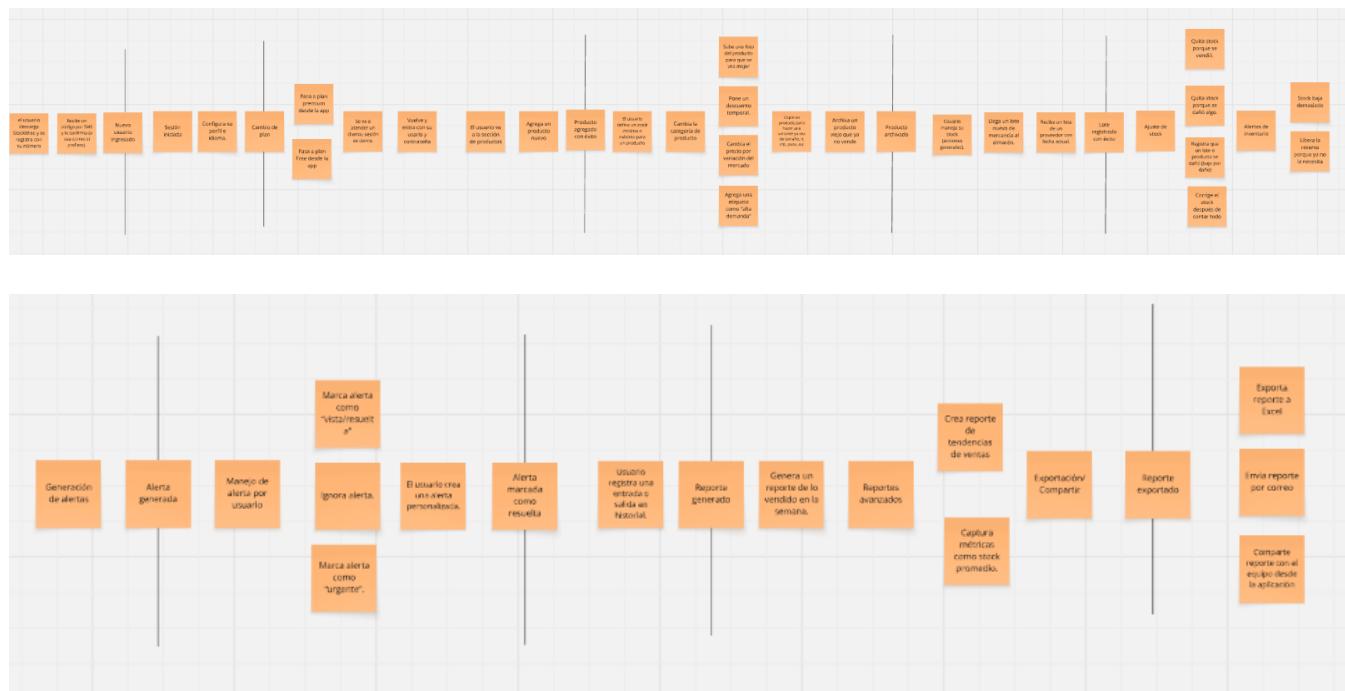
En esta imagen se aprecia la identificación de incertidumbres sobre qué datos "validados" deben alimentar la auditoría y los reportes (para evitar inconsistencias), así como el alcance y filtros de los informes (ventas semanales, tendencias, métricas como stock promedio) y quién puede exportar/compartir (Excel, correo) según el plan.



En esta imagen se aprecia la identificación de incertidumbres sobre cómo garantizar que un "ajuste de stock" refleje la realidad y no oculte errores o maniobras: cuándo corresponde usar ajuste versus quitar stock por venta o baja por daño, qué evidencias (motivo, usuario, documento/foto) y doble validación exigir, cómo conciliar el conteo físico sin duplicar salidas previas, cómo impactar lotes y reservas (p. ej., liberar reservas ya innecesarias) y cómo disparar alertas por umbral después de cada movimiento.

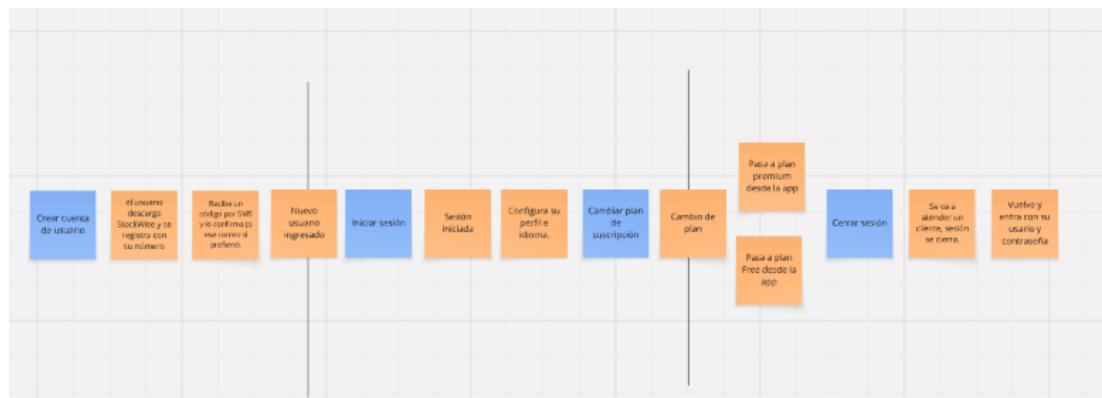
Step 4: Pivotal Points

En esta sección se marcaron los puntos clave que denotan transiciones significativas dentro del proceso de desarrollo, especialmente aquellos eventos o decisiones que implican un cambio en el comportamiento del sistema o en la forma en que los distintos Bounded Contexts interactúan entre sí. Como se puede observar, los pivotal points son sumamente importantes. Por ello, para poder comprender y apreciar el diseño realizado en este paso, se muestran las siguientes imágenes.

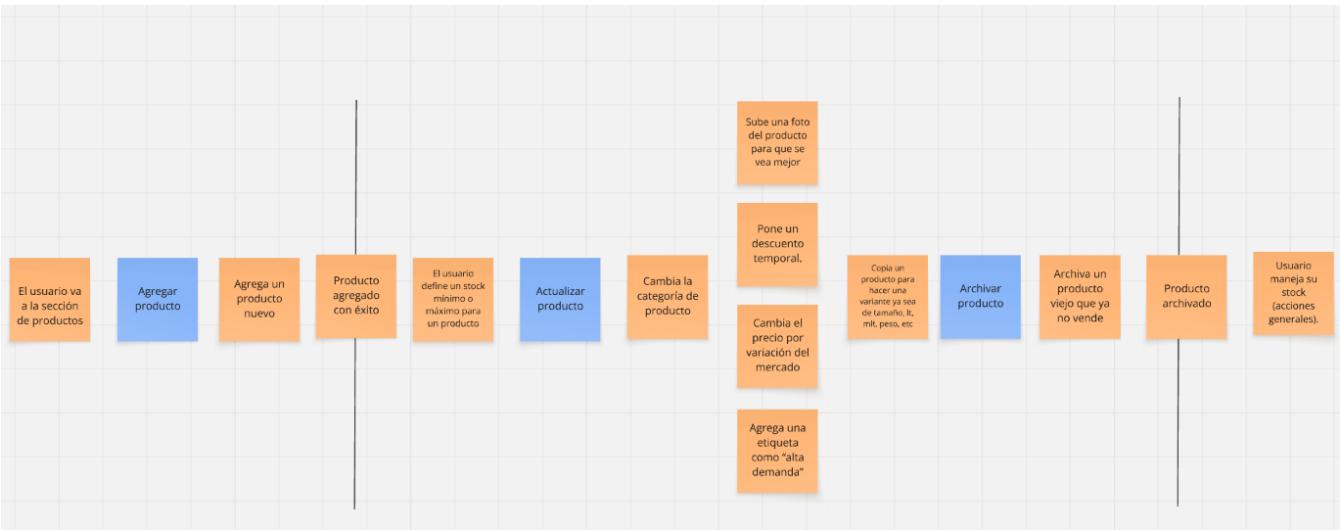


Step 5: Commands

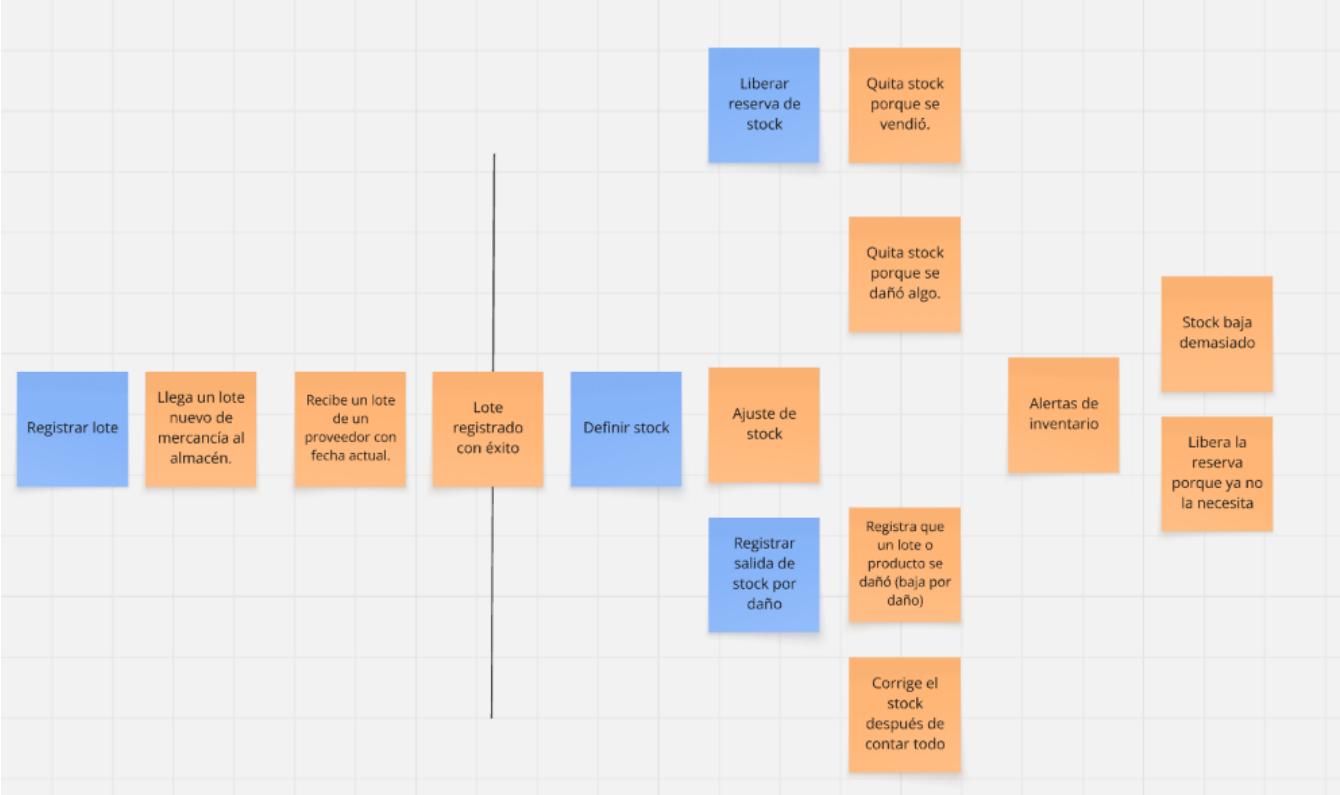
En este paso se definen los comandos que representan las acciones que los usuarios pueden ejecutar en el sistema. Estos comandos son fundamentales para desencadenar eventos y modificar el estado del sistema, y deben estar alineados con los eventos identificados previamente. A continuación, se muestran los comandos relevantes para nuestro dominio.



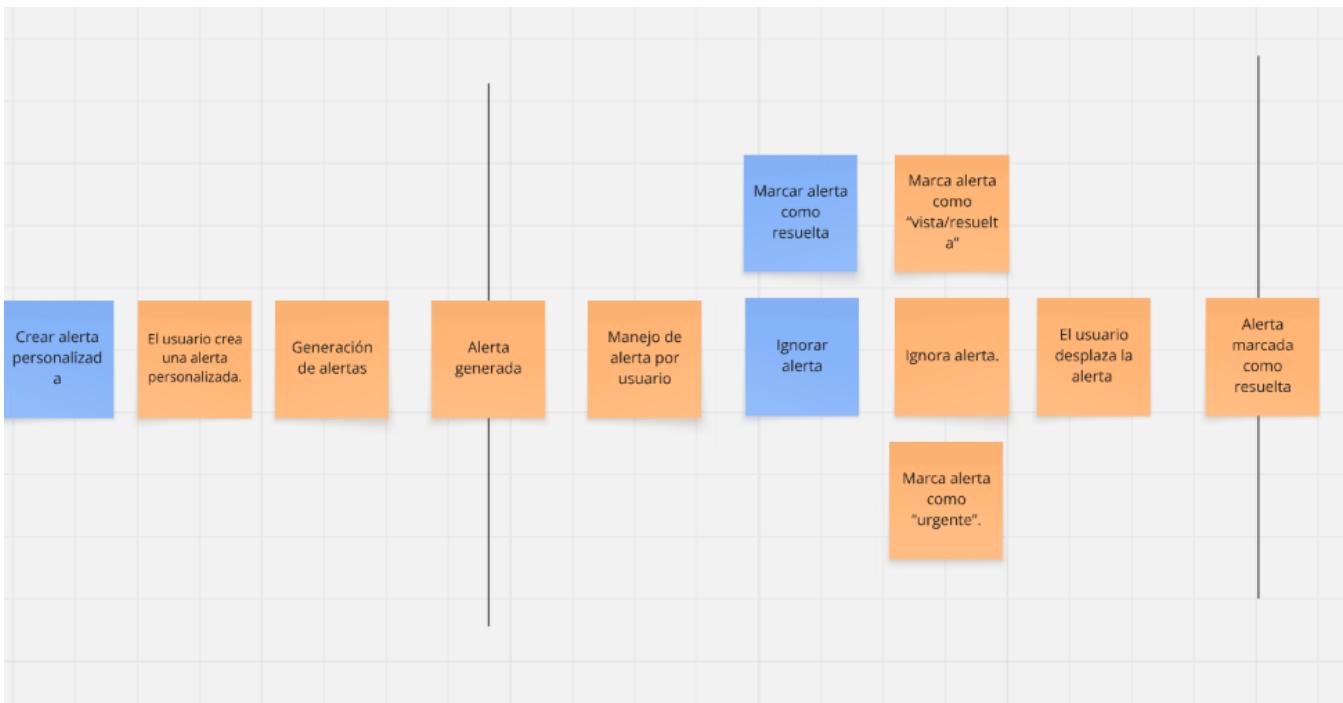
Crear cuenta de usuario registra la cuenta y dispara la verificación (SMS/Email); Iniciar sesión autentica factores y abre una sesión válida; Cambiar plan de suscripción solicita upgrade/downgrade, coordina el cobro y aplica el plan activo tras confirmación; Cerrar sesión invalida la sesión/token en curso.



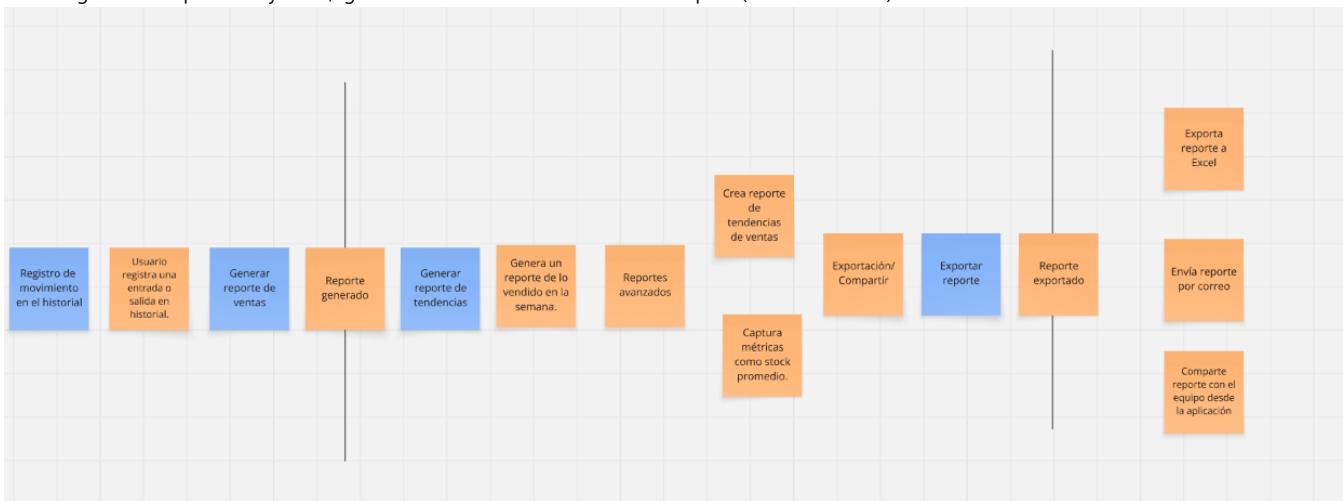
Agregar producto da de alta un ítem con sus datos base; Actualizar producto modifica atributos (nombre, descripción, imágenes, etc.); Cambiar la categoría de producto reasigna su clasificación; Archivar producto cambia el estado a "archivado" cuando no existen existencias ni reservas activas.



Registrar lote ingresa stock por lote (cantidad, almacén, vencimiento); Definir stock establece umbrales mínimo/máximo para alertas y reposición; Ajuste de stock corrige existencias con doble validación cuando procede; Liberar reserva de stock devuelve a disponible las cantidades apartadas que ya no se necesitan.



Generar alerta crea una alerta con severidad y canales según reglas; Marcar alerta como resuelta cierra el caso tras confirmación de atención; Marcar como urgente eleva prioridad y canal; Ignorar alerta cambia el estado cuando aplica (con trazabilidad).



Generar reporte ejecuta una consulta/KPI sobre datos validados; Exportar reporte materializa el resultado (Excel/PDF) con sello/firma; Enviar/Compartir reporte distribuye el archivo por correo o canal elegido y registra la entrega.

Step 6: Policies

En este paso se describen políticas que rigen decisiones y acciones dentro del sistema, asegurando que este se alinee con las necesidades del usuario y los objetivos del negocio.

Los siguientes frames contienen políticas que indican diferentes restricciones o condiciones descritas a continuación:

-Validar contacto antes de activar la cuenta. El correo o teléfono debe confirmarse (OTP) antes de permitir inicio de sesión o uso de la app.

-Un solo plan activo por cuenta. La suscripción vigente es única (Free o Premium) y el cambio de plan solo se aplica cuando el pago está confirmado.

-Producto con datos mínimos obligatorios. Todo producto debe tener categoría y precio; además se definen umbrales de stock min/máx para integrarse con inventario.

-No archivar con stock/Reserva. Prohibido archivar si existen existencias o reservas; primero se libera/traslada el stock.

-Perecibles con fecha de vencimiento. Lotes de productos perecibles deben registrar expiryDate.

-Ajuste con doble validación. Cualquier ajuste de inventario requiere aprobación en un segundo paso.

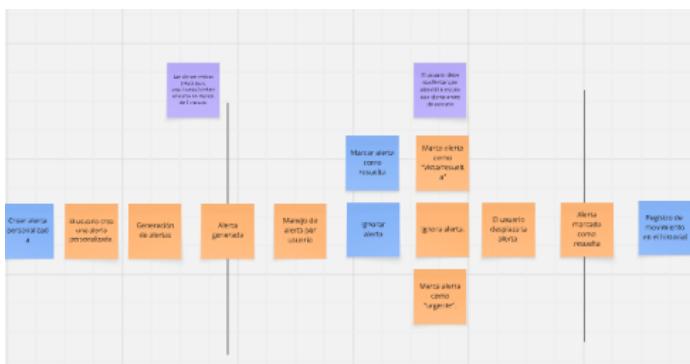
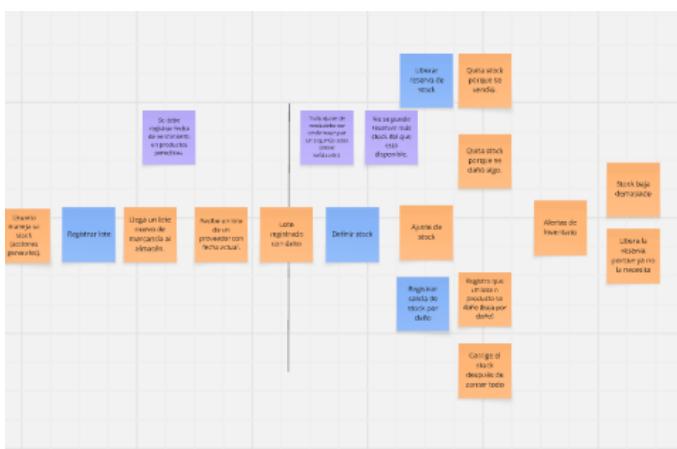
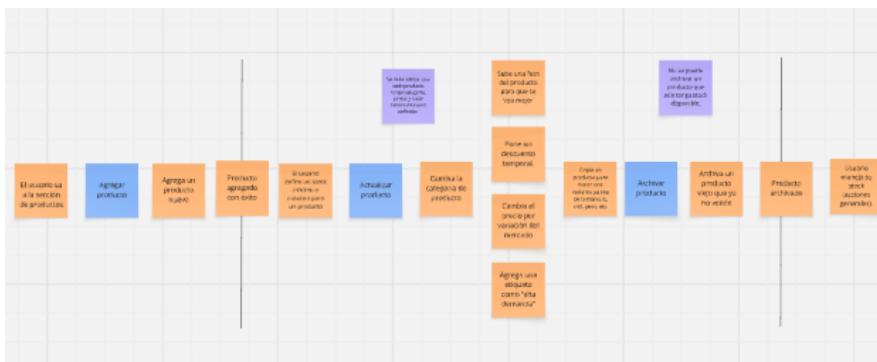
-No reservar más que disponible. Las reservas nunca pueden exceder el stock disponible.

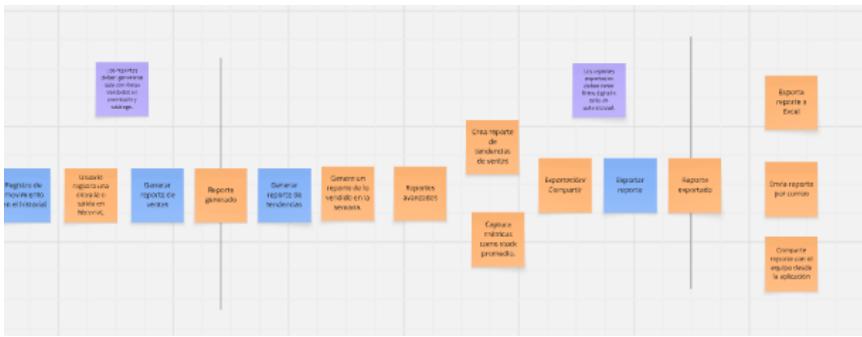
-Críticas en < 1 minuto. Las alertas de alta severidad (stock bajo, vencimiento) deben generarse y enviarse en menos de un minuto.

-ACK obligatorio antes de cerrar. El usuario debe reconocer/escalar una alerta antes de marcarla como resuelta.

-Reportes solo con datos validados. Los informes se generan a partir de eventos confirmados (inventario/catálogo).

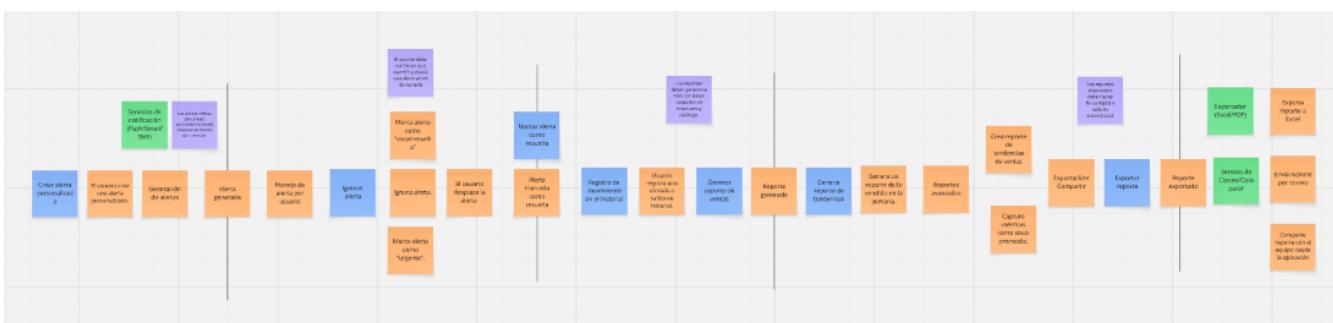
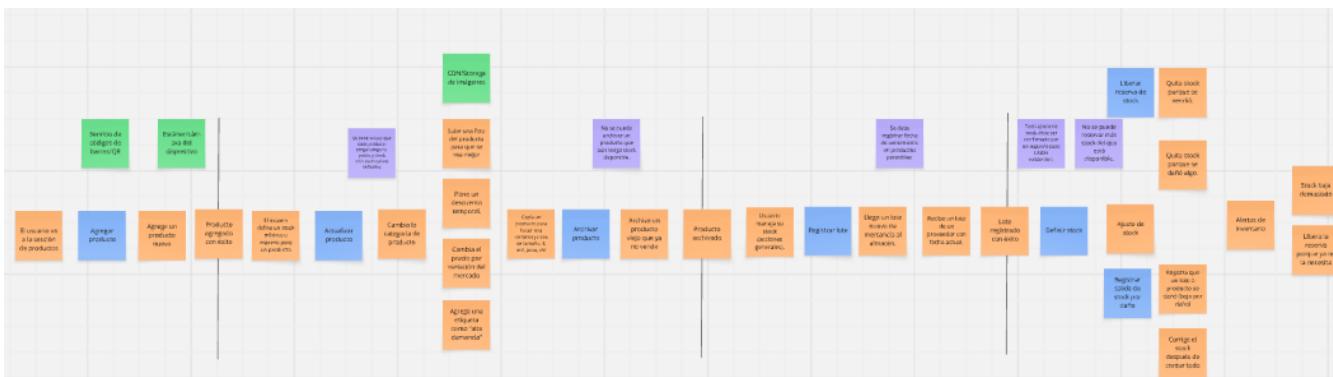
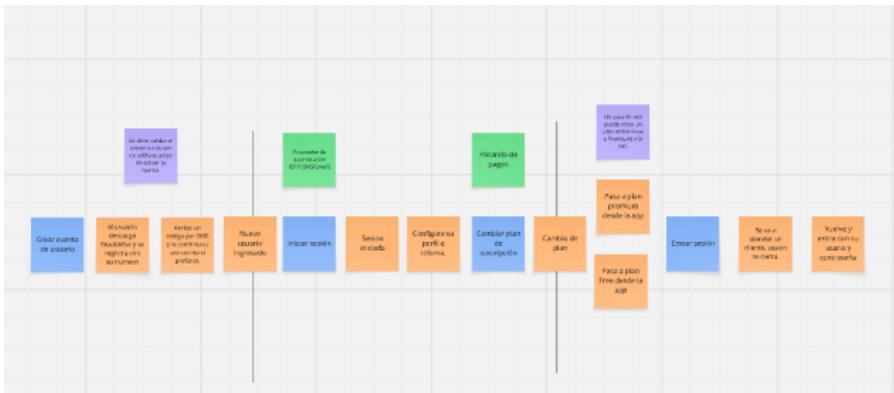
-Exportos con firma/sello. Los archivos exportados (Excel/PDF) deben incluir firma digital o sello de autenticidad.





Step 7 & 8: Read Models & External Systems

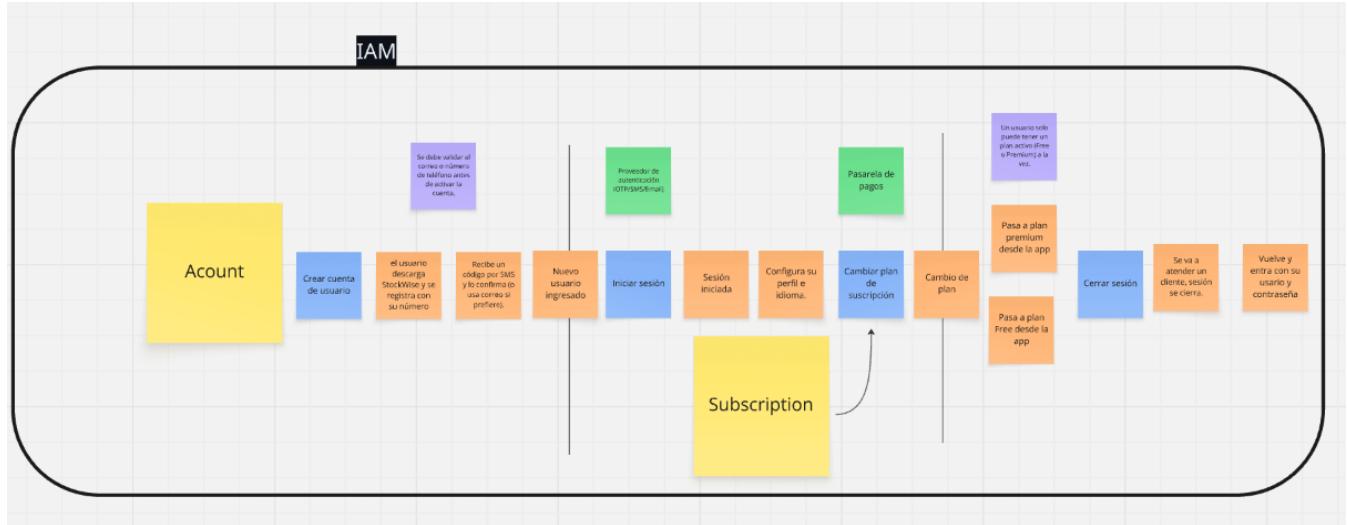
Estos dos pasos conforman la aparición de los read models y los external systems respectivamente. Por un lado, los read models son proyecciones que representan vistas personalizadas, dashboards, o reportes. Por otro lado, los external systems en DDD representan aquellos componentes o servicios externos con los que el sistema necesita interactuar, como pasarelas de pago, servicios de autenticación, sistemas de terceros, o integraciones con otras plataformas empresariales.



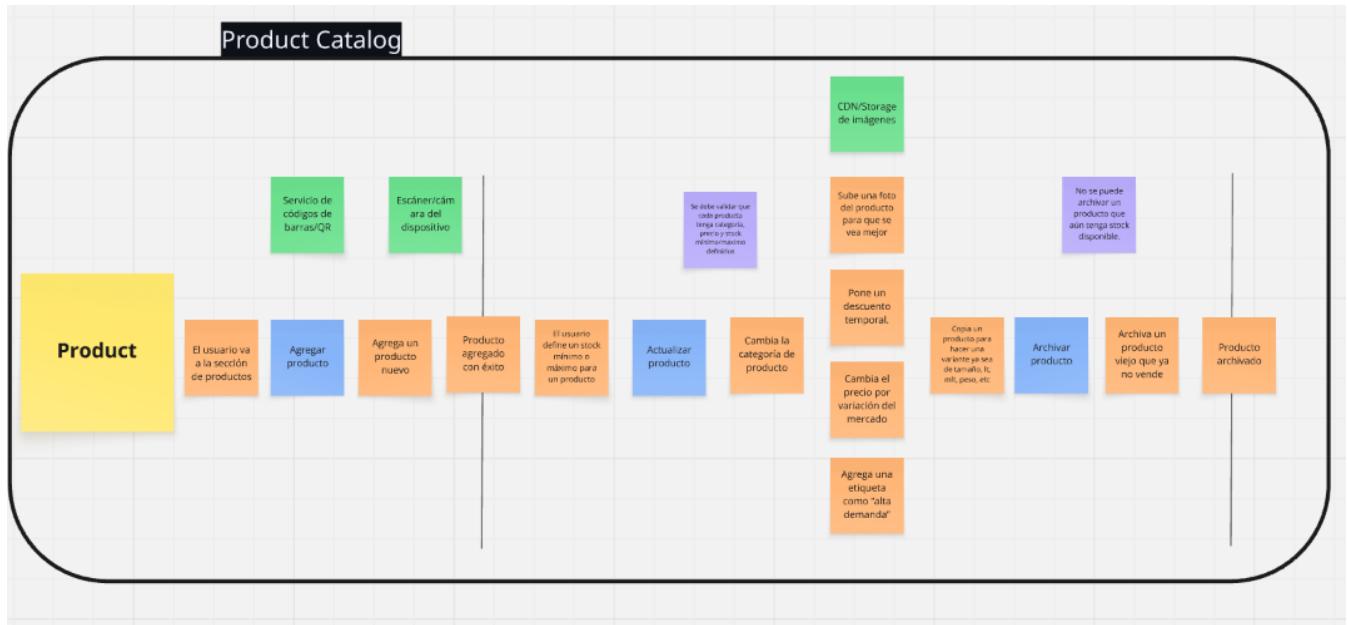
Step 9 & 10: Aggregates & Bounded Contexts

En los pasos 9 y 10 se introducen dos conceptos clave del Domain-Driven Design: Aggregates y Bounded Contexts. Los aggregates representan unidades de consistencia dentro del modelo de dominio, agrupando entidades y objetos de valor que deben mantenerse coherentes bajo una misma raíz (aggregate root). Por otro lado, los bounded contexts definen límites en los cuales un modelo tiene significado y coherencia, estableciendo una frontera clara para evitar ambigüedades conceptuales.

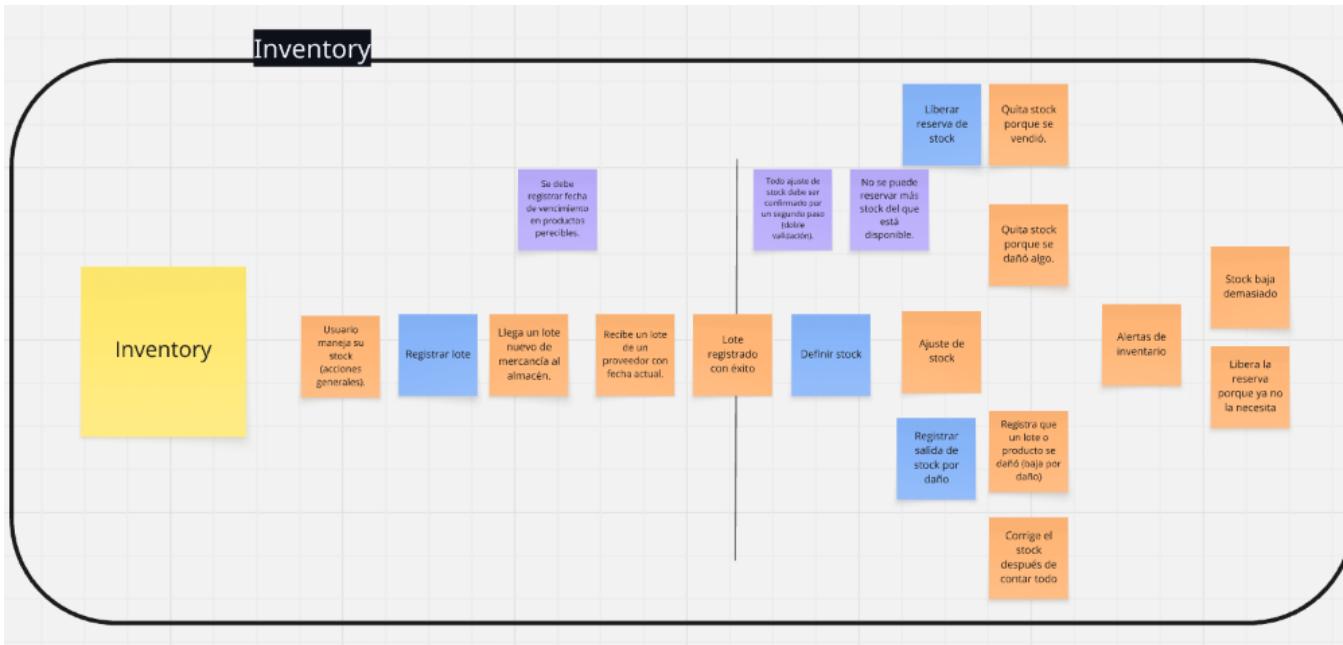
Bounded Context: IAM



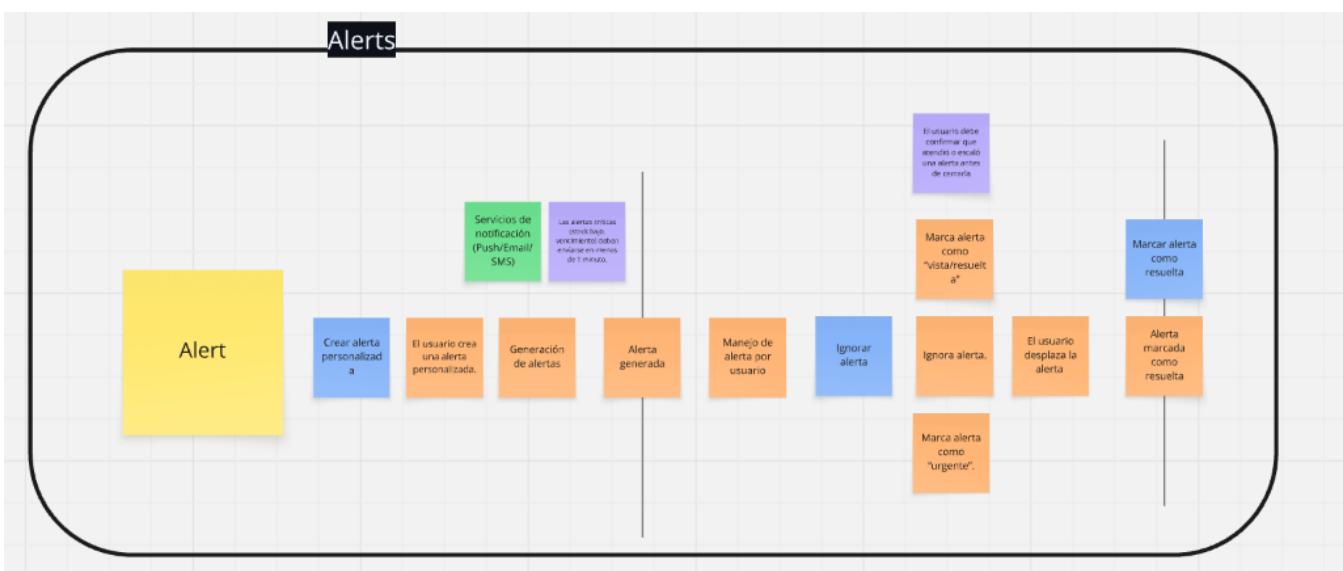
Bounded Context: Product catalog



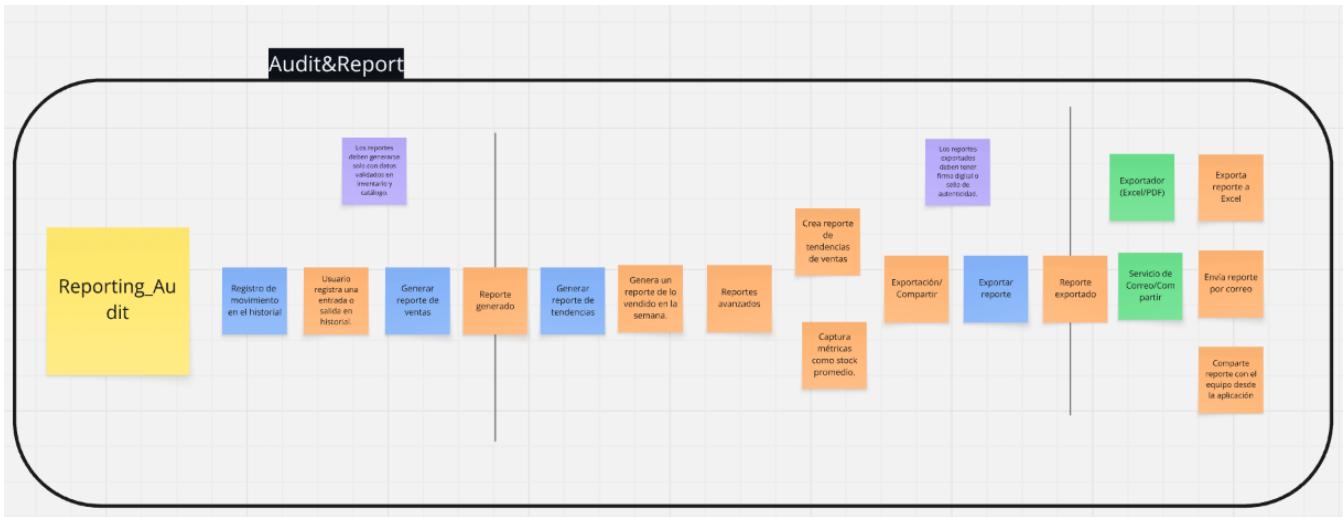
Bounded Context: Inventory



Bounded Context: Alerts

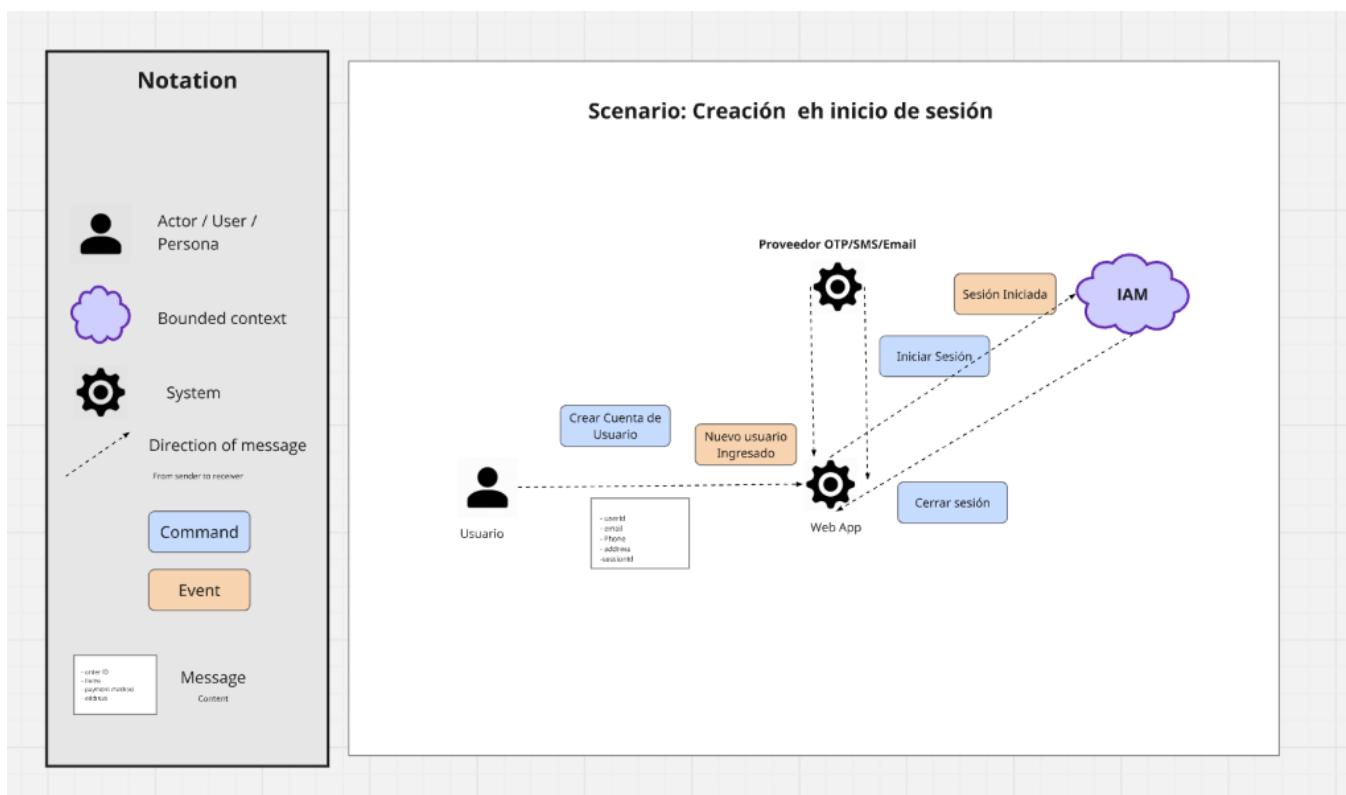


Bounded Context: AuditReport

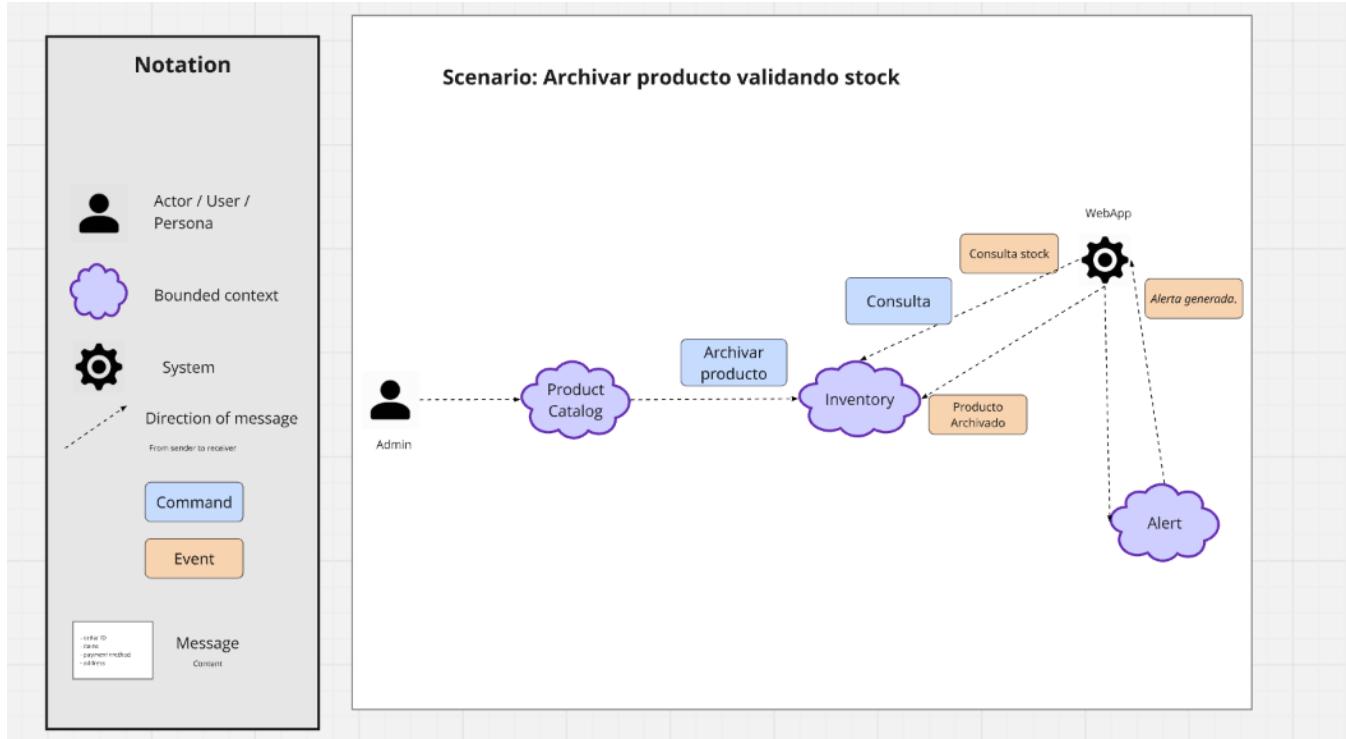


2.5.1.2. Domain Message Flows Modeling

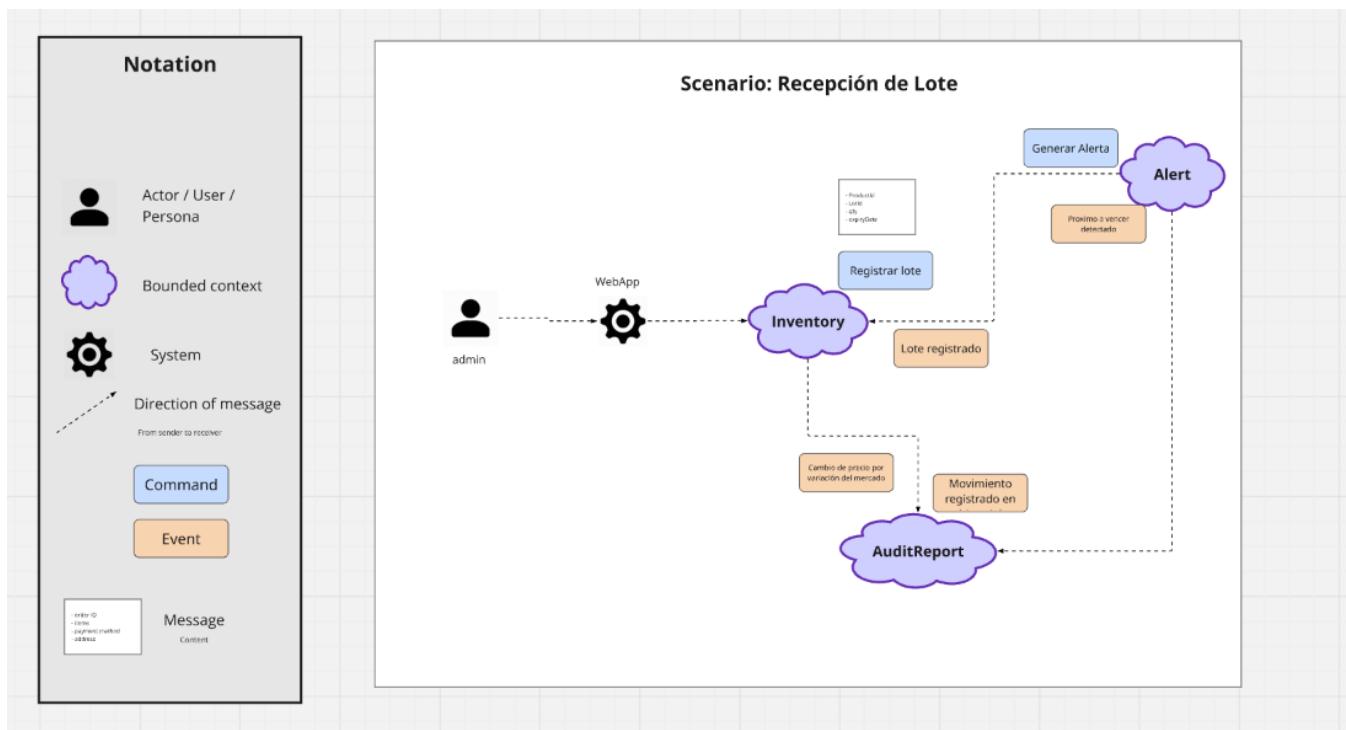
En los siguientes gráficos, se representa el proceso seguido para visualizar cómo deben colaborar los bounded contexts para resolver distintos escenarios que pueden presentarse en el negocio.



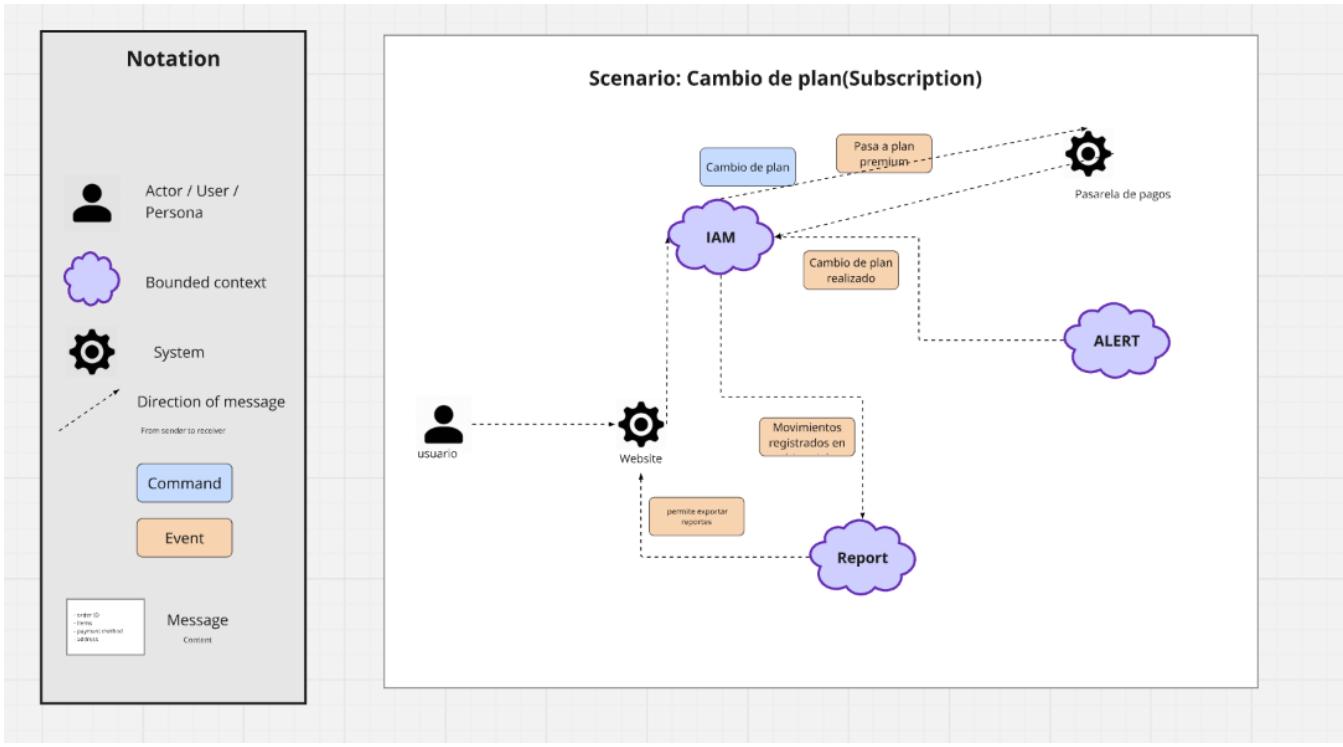
Este es un escenario de creación e inicio/cierre de sesión para nuestra app de bodega: primero, el usuario registra su cuenta en la Web App enviando su correo o teléfono; la aplicación remite el comando a IAM, que coordina con el proveedor OTP/SMS/Email para verificar el contacto y, una vez confirmado, publica el evento "Nuevo usuario ingresado" dejando la cuenta activa. A continuación, el usuario solicita iniciar sesión; la Web App envía el comando a IAM, que valida credenciales/OTP y emite "Sesión iniciada", habilitando el acceso a las demás funciones. Finalmente, cuando el usuario decide salir, la Web App manda "Cerrar sesión" a IAM, que invalida el token/sesión y concluye el ciclo de autenticación.



Este es un escenario de archivado de producto con validación de stock: el Admin solicita en Product Catalog el comando Archivar producto y, antes de ejecutar el cambio, el catálogo consulta a Inventory el estado de existencias y reservas. Con la Consulta de stock, Inventory responde si hay unidades o reservas activas; si aún existe stock o reservas, desde la Web App se genera la notificación correspondiente y se envía a Alerts (Alerta generada) para informar y bloquear la acción. Si el resultado confirma stock=0 y sin reservas, el catálogo procede a completar la operación y se publica el evento Producto archivado, quedando el ítem fuera de publicación.



Este es un escenario de recepción de lote: el admin, a través de la Web App, envía a Inventory el comando Registrar lote con productId, lotId, qty y expiryDate; Inventory valida la política de perecibles (exigir fecha de vencimiento) y confirma la alta con el evento Lote registrado; a partir de ese hecho aplica sus reglas: si el lote es perecible y cae dentro del umbral, detecta "Próximo a vencer" y ordena en Alerts el comando Generar Alerta para notificar al usuario; en paralelo, registra la trazabilidad de la operación en AuditReport mediante el evento Movimiento registrado en historial, dejando el ingreso listo para reportes e indicadores.

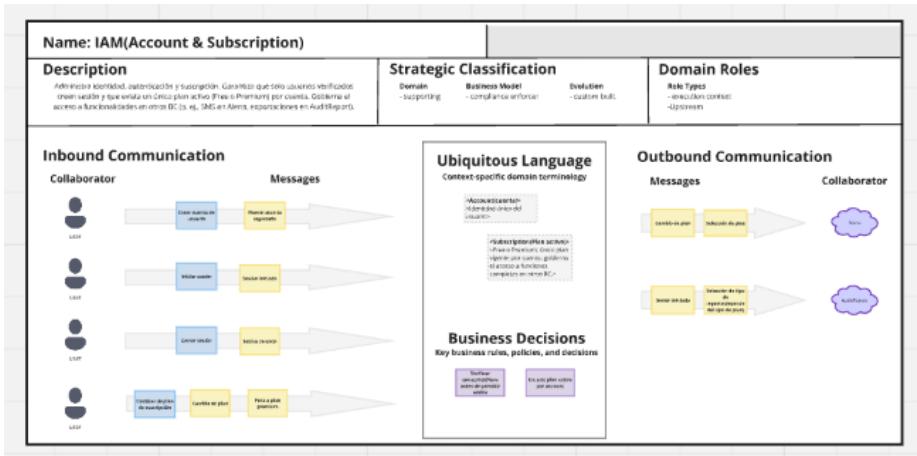


Este es un escenario de cambio de plan (Subscription): el usuario solicita en el Website el comando Cambio de plan, que es atendido por el BC IAM; IAM coordina con la pasarela de pagos para capturar el cobro y, una vez confirmado, publica los eventos "Cambio de plan realizado" y "Pasa a plan premium", los cuales habilitan capacidades en los demás contextos: Alerts reconfigura sus canales (por ejemplo, activa SMS para notificaciones críticas) y Report registra el movimiento de suscripción en la auditoría; como efecto visible, el Website recibe el permiso de exportar reportes (gating por plan) y el usuario continúa operando con las nuevas funcionalidades asociadas a su plan activo..

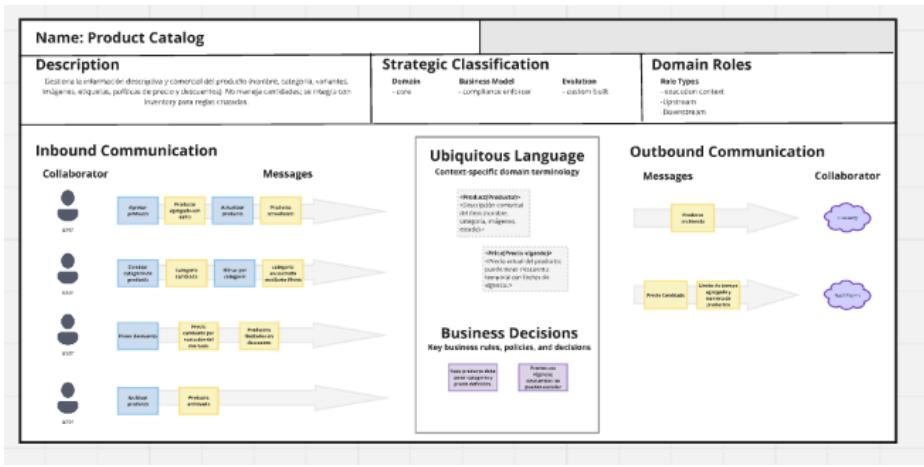
2.5.1.3. Bounded Context Canvases

El Bounded Context Canvas es una herramienta visual del Domain-Driven Design (DDD) que permite definir, entender y comunicar de forma clara los límites, responsabilidades y elementos clave de un Bounded Context. Ayuda a alinear al equipo en torno a una visión compartida del dominio, identificando entidades, eventos, comandos y relaciones externas. Además, elaborar un diseño modular del sistema de manera coherente es posible gracias a las nomenclaturas establecidas por un Bounded Context Canvas

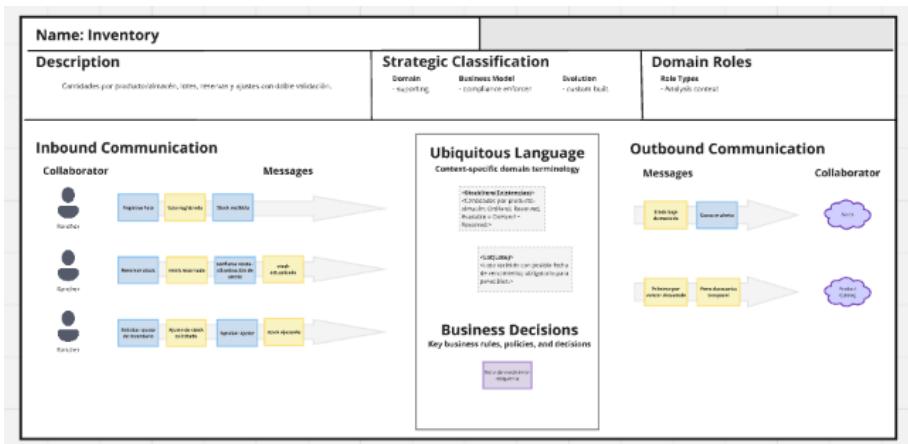
Bounded Context: IAM



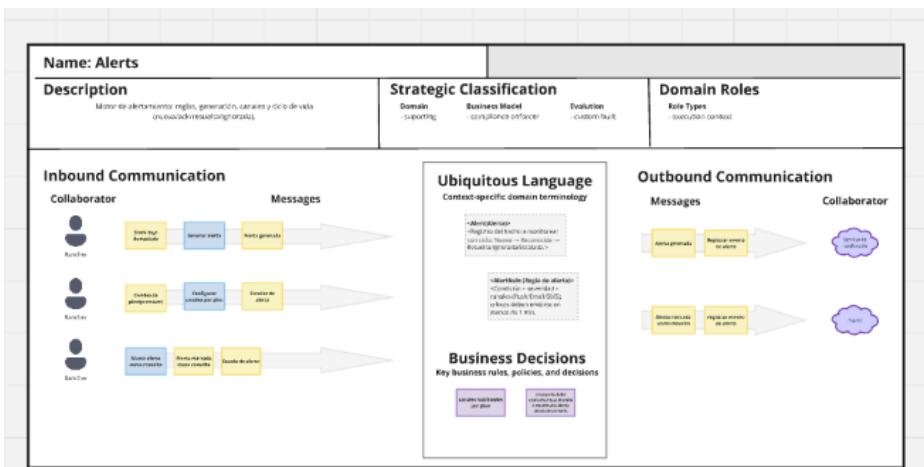
Bounded Context: Product catalog



Bounded Context: Inventory



Bounded Context: Alerts



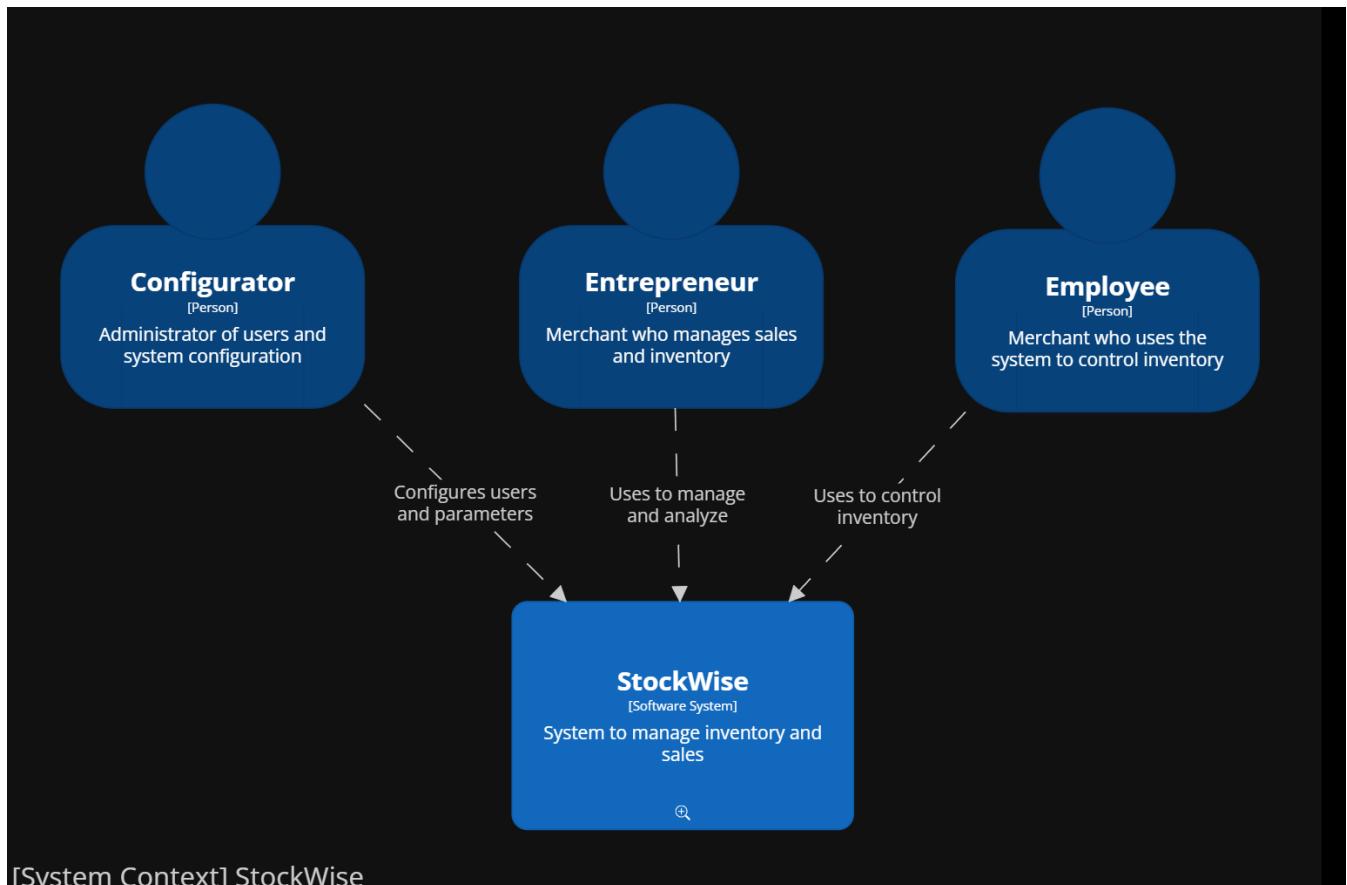
Bounded Context: AuditReport

Name: AuditReport		
Description Translational and analytical. Registers overview of business validated and generates monthly reports y exportaciones.	Strategic Classification Domain - value Business Model - compliance en forcing Evaluation - customer trial	Domain Roles Role Types - monitoring content
Inbound Communication Collaborator Customer: "Envío informe Q3 actualizado", "requisitos más recientes", "información capitalizada". Supplier: "Favorable resultado", "información actualizada", "información capitalizada". Partner: "Favorable resultado", "información actualizada", "información capitalizada". Messages Request for report: "Puedes enviar el informe Q3?", "¿Podrías enviar el informe Q3?". Response regarding: "Sí, ya lo envíe", "Sí, ya lo envíe, se adjunta el informe Q3 para tu revisión de acuerdo a los requisitos". Cancellation of prior contract: "Cancelo la petición anterior", "Cancelo la petición anterior".	Ubiquitous Language Context-specific domain terminology - informe Q3: Registro de actividad realizada. - información capitalizada: datos que sirven para informar sobre las actividades de acuerdo a los requisitos. - respuesta favorable: indica que se ha cumplido con los requisitos establecidos. - cancelación de contrato anterior: se cancela la solicitud anterior.	Outbound Communication Messages Response regarding: "Sí, ya lo envíe", "Sí, ya lo envíe, se adjunta el informe Q3 para tu revisión de acuerdo a los requisitos". Response enclosed: "Adjunto informe Q3", "Adjunto informe Q3". Collaborator Customer: "Puedes enviar el informe Q3?", "Cancelo la petición anterior". Supplier: "Sí, ya lo envíe", "Adjunto informe Q3".
	Business Decisions Key business rules, policies, and decisions Informe de cumplimiento: "informe de cumplimiento", "informe de cumplimiento". Informe de revisión: "informe de revisión", "informe de revisión".	

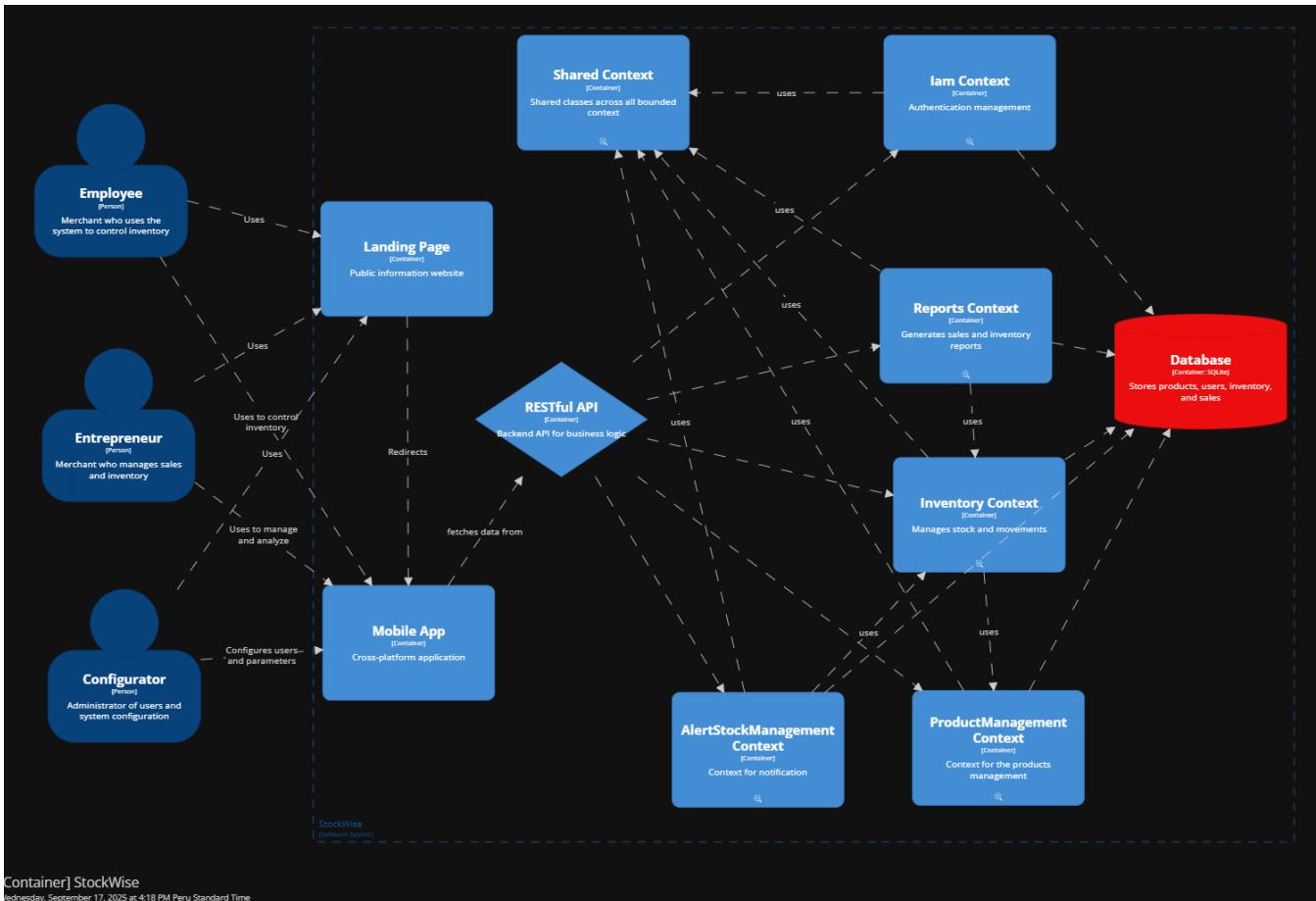
2.5.2. Context Mapping

2.5.3. Software Architecture

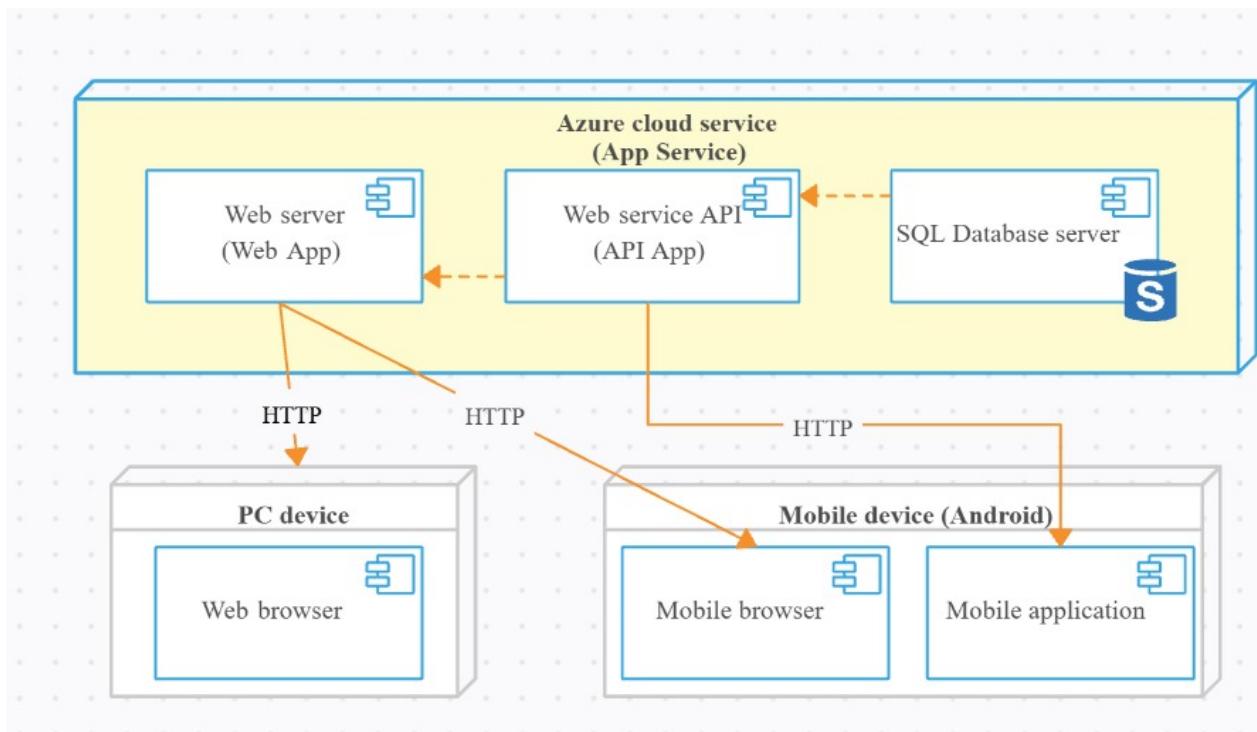
2.5.3.1. Software Architecture Context Level Diagrams



2.5.3.2. Software Architecture Container Level Diagrams



2.5.3.3. Software Architecture Deployment Diagrams



2.6. Tactical-Level Domain-Driven Design

2.6.1. Bounded Context: AlertStockManagement Context

2.6.1.1. Domain Layer

Clase/Interfaz	Tipo	Propósito	Atributos / Métodos

Clase/Interfaz	Tipo	Propósito	Atributos / Métodos
StockAlertItem	Aggregate Root (Entity)	Representa una alerta de stock para un producto	ProductName, Quantity, MinStock, EntryDate, IsLowStock
StockAlertQuery	Query	Criterio de consulta de alertas de stock	IncludeLowStock
IStockAlertQueryService	Domain Service (Interface)	Define contrato para consultas de alertas de stock	GetAlertsAsync(StockAlertQuery): Task<IEnumerable>
IInventoryReadRepository	Repository (Interface)	Lectura de inventario para generar alertas	GetStockAlertsAsync(StockAlertQuery): Task<List>
IStockAlertReadRepository	Repository (Interface)	Lectura de alertas de stock ya calculadas	GetStockAlertsAsync(StockAlertQuery): Task<List>

2.6.1.2. Application Layer

Clase/Interfaz	Tipo	Propósito	Atributos / Métodos
StockAlertQueryService	Application Service	Implementa consultas de alertas de stock contra inventario	repository, GetAlertsAsync(StockAlertQuery): Task<List>
IStockAlertQueryService	Application Service (Interface)	Contrato de aplicación para consultas de alertas de stock	GetAlertsAsync(StockAlertQuery): Task<List>

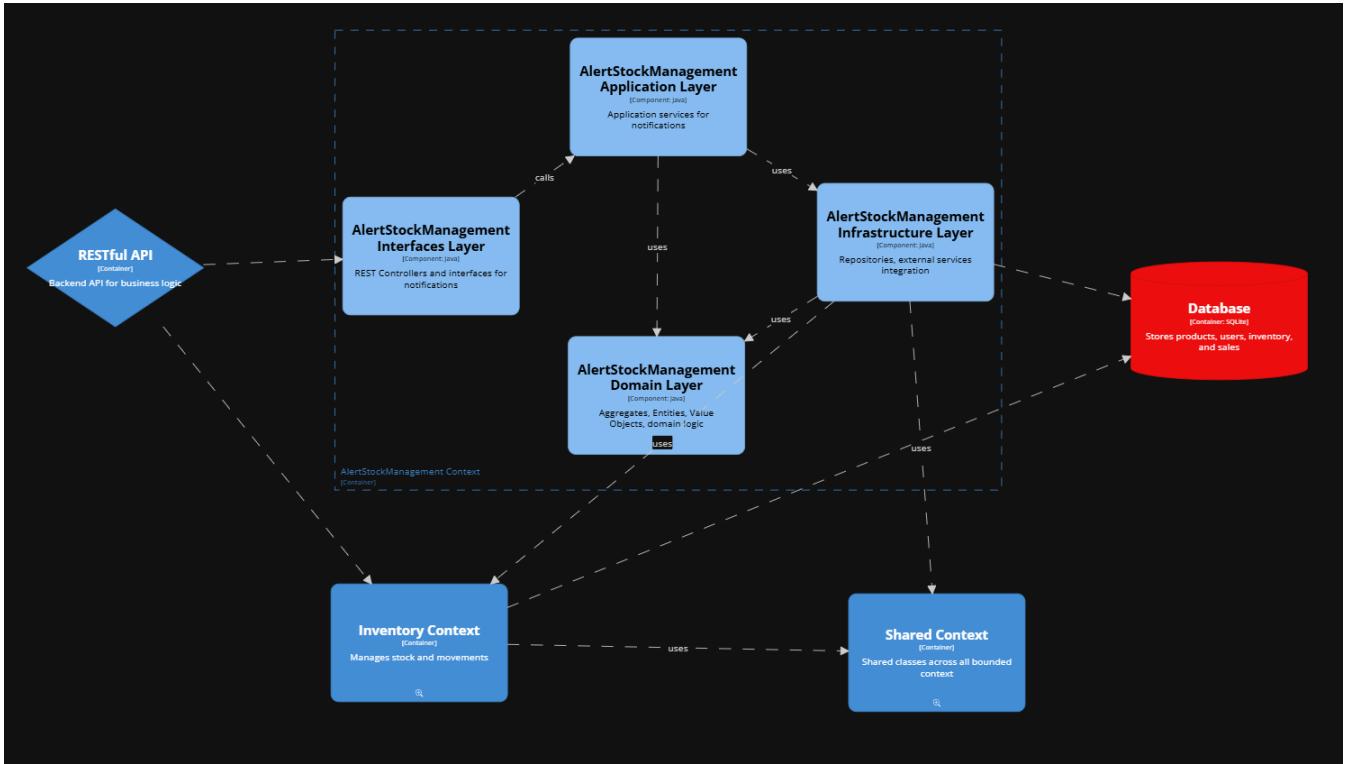
2.6.1.3. Interface Layer

Clase/Interfaz	Tipo	Propósito	Atributos / Métodos
StockAlertResource	Resource (DTO)	Representa alerta de stock en respuestas REST	ProductName, Quantity, MinStock, EntryDate, IsLowStock
StockAlertTransform	Assembler	Convierte entidad StockAlertItem en DTO	ToResource(StockAlertItem): StockAlertResource
StockAlertController	REST Controller	Expone endpoints para consultar alertas de stock	_service, GetAlerts(): Task

2.6.1.4. Infrastructure Layer

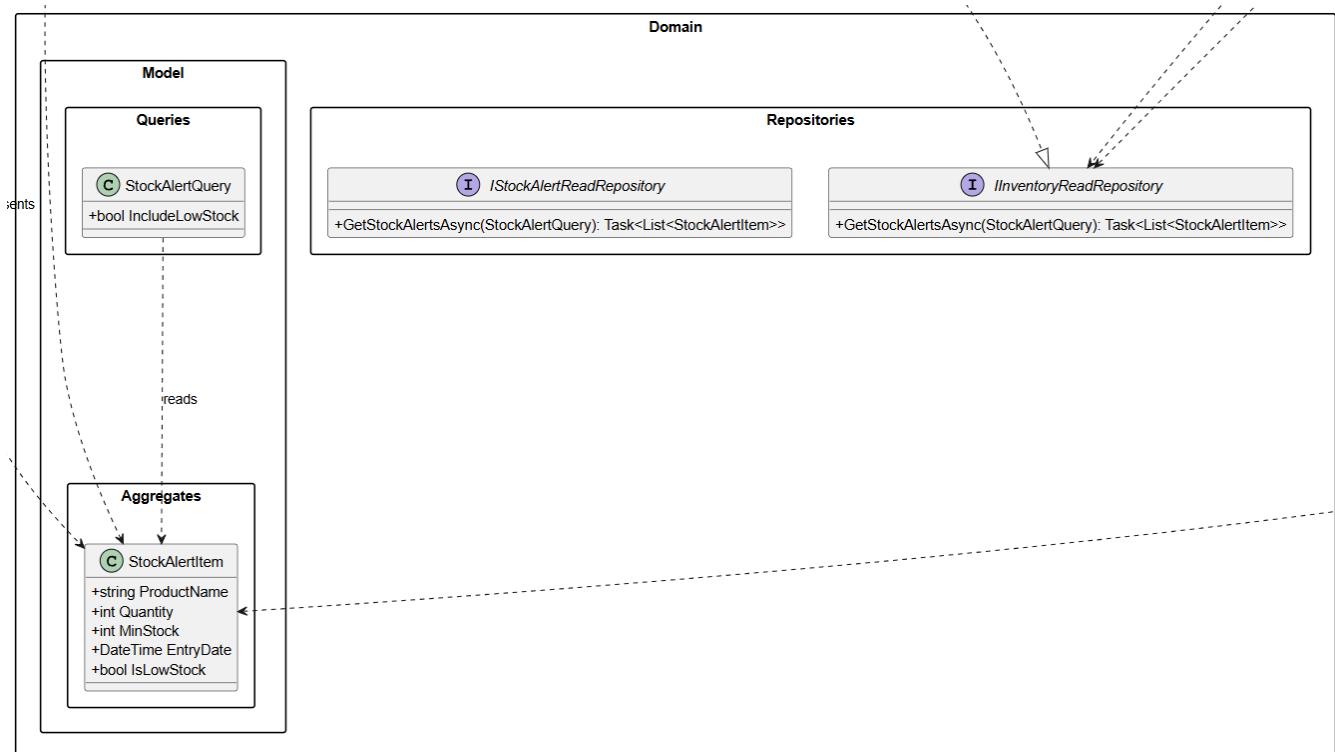
Clase	Tipo	Propósito	Atributos / Métodos
InventoryReadRepository	Repository Implementation	Implementación EF Core para leer stock y generar alertas	context, GetStockAlertsAsync(StockAlertQuery): Task<List>

2.6.1.5. Bounded Context Software Architecture Component Level Diagrams



2.6.1.6. Bounded Context Software Architecture Code Level Diagrams

2.6.1.6.1. Bounded Context Domain Layer Class Diagrams



2.6.2. Bounded Context: IAM Context

2.6.2.1. Domain Layer

Clase/Interfaz	Tipo	Propósito	Atributos / Métodos
User	Aggregate Root (Entity)	Representa un usuario dentro del sistema IAM	Id, Name, LastName, Email, Password, Role, User(), User(SignInCommand), User(SignUpCommand)
UserRole	Enum (Value Object)	Define los roles posibles de usuario	Employee, Administrator

Clase/Interfaz	Tipo	Propósito	Atributos / Métodos
SignUpCommand	Command	Solicitud para registrar un nuevo usuario	Name, LastName, Email, Password
SignInCommand	Command	Solicitud para iniciar sesión	Email, Password
ChangeUserRoleCommand	Command	Solicitud para cambiar el rol de un usuario	UserId, NewRole
 GetUserByIdQuery	Query	Consulta para obtener un usuario por ID	Id
IUserCommandService	Domain Service (Interface)	Define el contrato de servicios para manejar comandos de usuario	Handle(SignUpCommand), Handle(SignInCommand), Handle(ChangeUserRoleCommand)
IUserQueryService	Domain Service (Interface)	Define el contrato de servicios para consultas de usuario	Handle(GetUserByIdQuery)
IUserRepository	Repository (Interface)	Acceso abstracto a la persistencia de usuarios	FindByEmailAsync, FindByldAsync, AddAsync, UpdateUserRoleAsync

2.6.2.2. Interface Layer

Clase	Tipo	Propósito	Atributos / Métodos
SignUpResource	Resource (DTO)	Representa los datos de entrada para registrarse	Name, LastName, Email, Password
SignInResource	Resource (DTO)	Representa los datos de entrada para iniciar sesión	Email, Password
UserResource	Resource (DTO)	Representa un usuario simplificado en respuesta	Id, Name, Email
AuthenticatedUserResource	Resource (DTO)	Representa un usuario autenticado junto con token	Id, Name, LastName, Token, Role
ChangeUserRoleResource	Resource (DTO)	Datos de entrada para cambiar el rol de un usuario	UserId, NewRole
SignUpCommandFromResourceAssembler	Assembler	Convierte un recurso en un comando de dominio	ToCommandFromResource(SignUpResource)
SignInCommandFromResourceAssembler	Assembler	Convierte un recurso en un comando de dominio	ToCommandFromResource(SignInResource)
UserResourceFromEntityAssembler	Assembler	Convierte una entidad en un recurso DTO	ToResourceFromEntity(User)
ChangeUserRoleCommandFromResourceAssembler	Assembler	Convierte un recurso en un comando de dominio	ToCommandFromResource(ChangeUserRoleResource)
AuthenticatedUserResourceFromEntityAssembler	Assembler	Convierte entidad y token en recurso de autenticación	ToResourceFromEntity(User, Token)
AuthenticationController	REST Controller	Expone endpoints de autenticación y gestión de usuario	SignUp(SignUpResource), SignIn(SignInResource), ChangeUserRole(ChangeUserRoleResource)

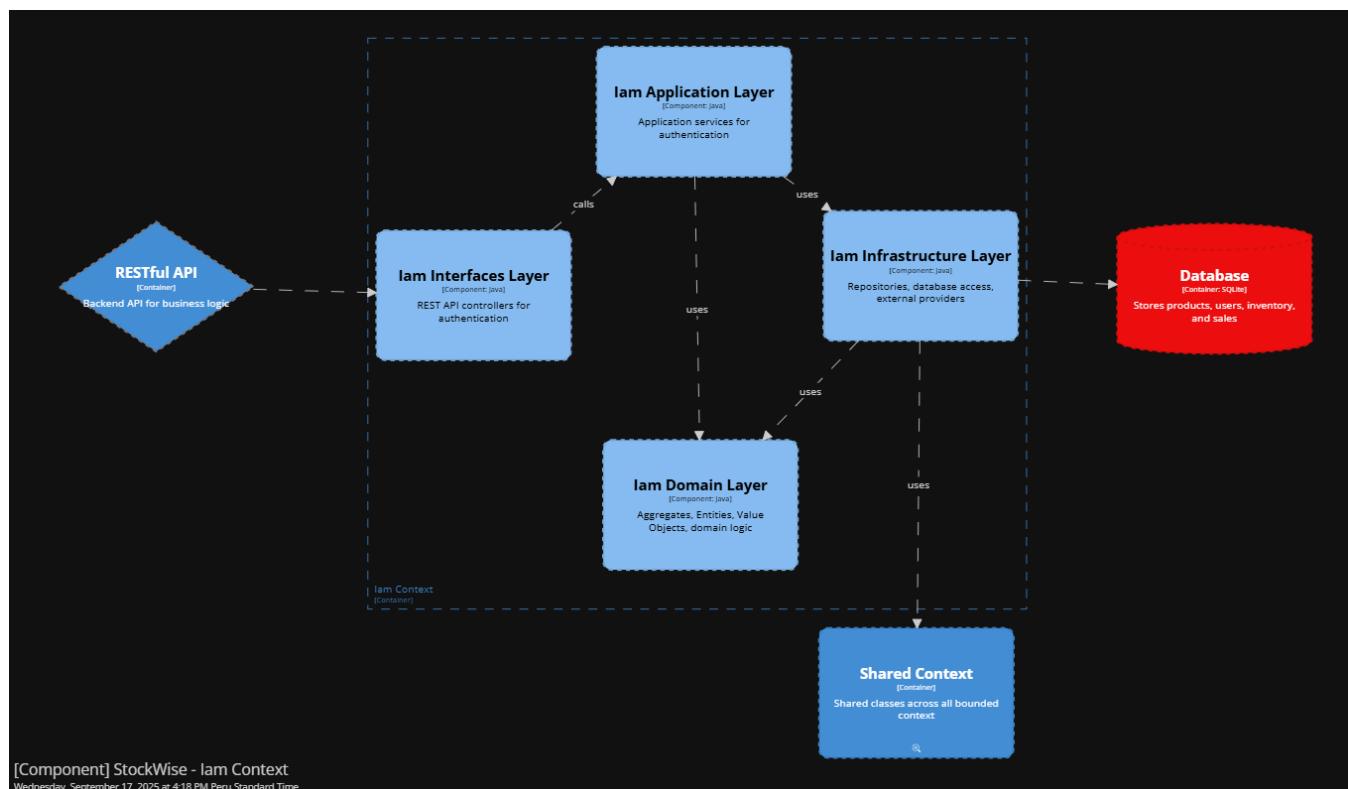
2.6.2.3. Application Layer

Clase/Interfaz	Tipo	Propósito	Atributos / Métodos
UserCommandService	Application Service	Implementa los servicios de comando de usuarios	userRepository, tokenService, hashingService, unitOfWork, Handle(SignUpCommand), Handle(SignInCommand), Handle(ChangeUserRoleCommand)
UserQueryService	Application Service	Implementa los servicios de consulta de usuarios	userRepository, GetAllAsync(), FindByIdAsync(int)
ITokenService	Outbound Service (Interface)	Define generación y validación de tokens	GenerateToken(User), ValidateToken(string)
IHashingService	Outbound Service (Interface)	Define operaciones de hashing de contraseñas	GenerateHash(string), VerifyHash(string, string)

2.6.2.4. Infrastructure Layer

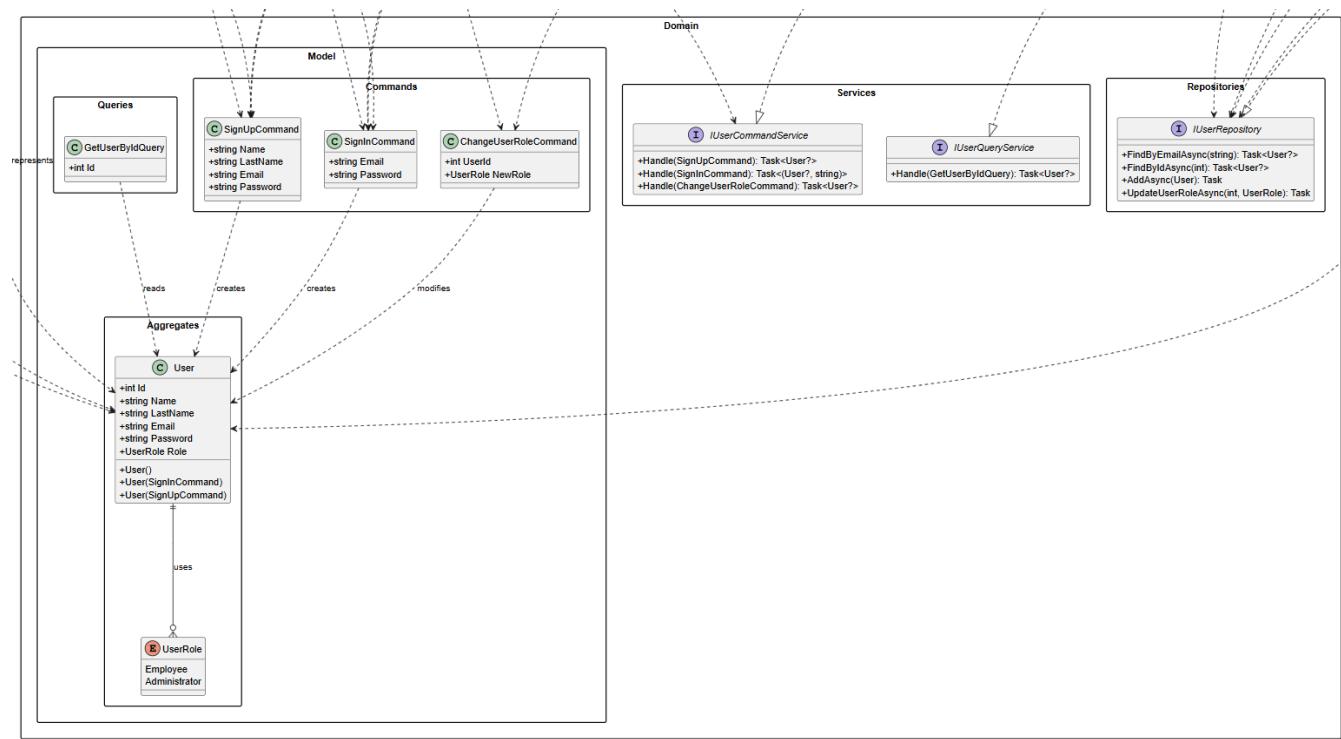
Clase	Tipo	Propósito	Atributos / Métodos
UserRepository	Repository (Implementation)	Implementación de acceso a datos de usuarios	context, FindByEmailAsync, FindByIdAsync, AddAsync, UpdateUserRoleAsync
TokenService	Outbound Service Implementation	Implementación de generación y validación de tokens JWT	GenerateToken(User), ValidateToken(string)
HashingService	Outbound Service Implementation	Implementación de hashing de contraseñas con BCrypt	GenerateHash(string), VerifyHash(string, string)
TokenSettings	Configuration	Configuración de secret key para JWT	Secret
AuthorizeRolesAttribute	Authorization Filter	Atributo para restringir acceso por roles en controllers	_roles, OnAuthorization(AuthorizationFilterContext)

2.6.2.5. Bounded Context Software Architecture Component Level Diagrams



2.6.2.6. Bounded Context Software Architecture Code Level Diagrams

2.6.2.6.1. Bounded Context Domain Layer Class Diagrams



2.6.2.6.2. Bounded Context Database Design Diagram

Users					
					comment
Id	INT	NULL	default	comment	
Name	VARCHAR(255)	N-N	default	comment	
LastName	VARCHAR(255)	N-N	default	comment	
Email	VARCHAR(255)	N-N	default	comment	
Password	VARCHAR(255)	N-N	default	comment	
Role	ENUM(Employee,Administrator)	N-N	Administrator	comment	
CreatedAt	TIMESTAMP	NULL	CURRENT_TIMESTAMP	comment	
UpdatedAt	TIMESTAMP	NULL	CURRENT_TIMESTAMP	comment	

2.6.3. Bounded Context: Inventory Context

2.6.3.1. Domain Layer

Clase/Interfaz	Tipo	Propósito	Atributos / Métodos
InventoryByProduct	Aggregate Root (Entity)	Representa inventario por producto	Id, Categoria, Producto, FechaEntrada, Cantidad, Precio, StockMinimo, UnidadMedida, InventoryByProduct(), InventoryByProduct(...)
InventoryByBatch	Aggregate Root (Entity)	Representa inventario por lotes	Id, Proveedor, Producto, FechaEntrada, Cantidad, Precio, Unidad, Total, InventoryByBatch(), InventoryByBatch(...)
Cantidad	Value Object	Representa una cantidad	Value, Cantidad(int)
Precio	Value Object	Representa un precio monetario	Value, Precio(decimal)
StockMinimo	Value Object	Define el stock mínimo de un producto	Value, StockMinimo(int)
Unidad	Value Object	Representa la unidad de medida	Value, Unidad(string)
CreateInventoryByProductCommand	Command	Crear inventario por producto	Categoría, Producto, FechaEntrada, Cantidad, Precio, StockMinimo, UnidadMedida
CreateInventoryByBatchCommand	Command	Crear inventario por lote	Proveedor, Producto, FechaEntrada, Cantidad, Precio, Unidad

Clase/Interfaz	Tipo	Propósito	Atributos / Métodos
GetInventoryByBatchQuery	Query	Consultar todos los inventarios por lote	-
GetInventoryByProductQuery	Query	Consultar todos los inventarios por producto	-
GetInventoryByProductIdQuery	Query	Consultar inventario por producto por Id	Id
GetInventoryByBatchByIdQuery	Query	Consultar inventario por lote por Id	Id
GetGeneralInventoryQuery	Query	Consultar inventario general	-
IInventoryByProductCommandService	Domain Service (Interface)	Define contrato para comandos de inventario por producto	Handle(CreateInventoryByProductCommand), DeleteAsync(int)
IInventoryByProductQueryService	Domain Service (Interface)	Define contrato para consultas de inventario por producto	GetAllAsync(), GetByIdAsync(int)
IInventoryByBatchCommandService	Domain Service (Interface)	Define contrato para comandos de inventario por lote	Handle(CreateInventoryByBatchCommand), DeleteAsync(int)
IInventoryByBatchQueryService	Domain Service (Interface)	Define contrato para consultas de inventario por lote	Handle(GetInventoryByBatchQuery)
IInventoryByProductRepository	Repository (Interface)	Acceso abstracto a inventarios por producto	ListAsync(), FindByIdAsync(int), AddAsync(InventoryByProduct), DeleteAsync(int)
IInventoryByBatchRepository	Repository (Interface)	Acceso abstracto a inventarios por lote	ListAsync(), FindByIdAsync(int), AddAsync(InventoryByBatch), DeleteAsync(int)

2.6.3.2. Application Layer

Clase	Tipo	Propósito	Atributos / Métodos
InventoryByProductCommandService	Application Service	Implementa comandos de inventario por producto	repository, unitOfWork, Handle(CreateInventoryByProductCommand), DeleteAsync(int)
InventoryByBatchCommandService	Application Service	Implementa comandos de inventario por lote	repository, unitOfWork, Handle(CreateInventoryByBatchCommand), DeleteAsync(int)
InventoryByProductQueryService	Application Service	Implementa consultas de inventario por producto	repository, GetAllAsync(), GetByIdAsync(int)
InventoryByBatchQueryService	Application Service	Implementa consultas de inventario por lote	repository, Handle(GetInventoryByBatchQuery)

2.6.3.3. Interface Layer

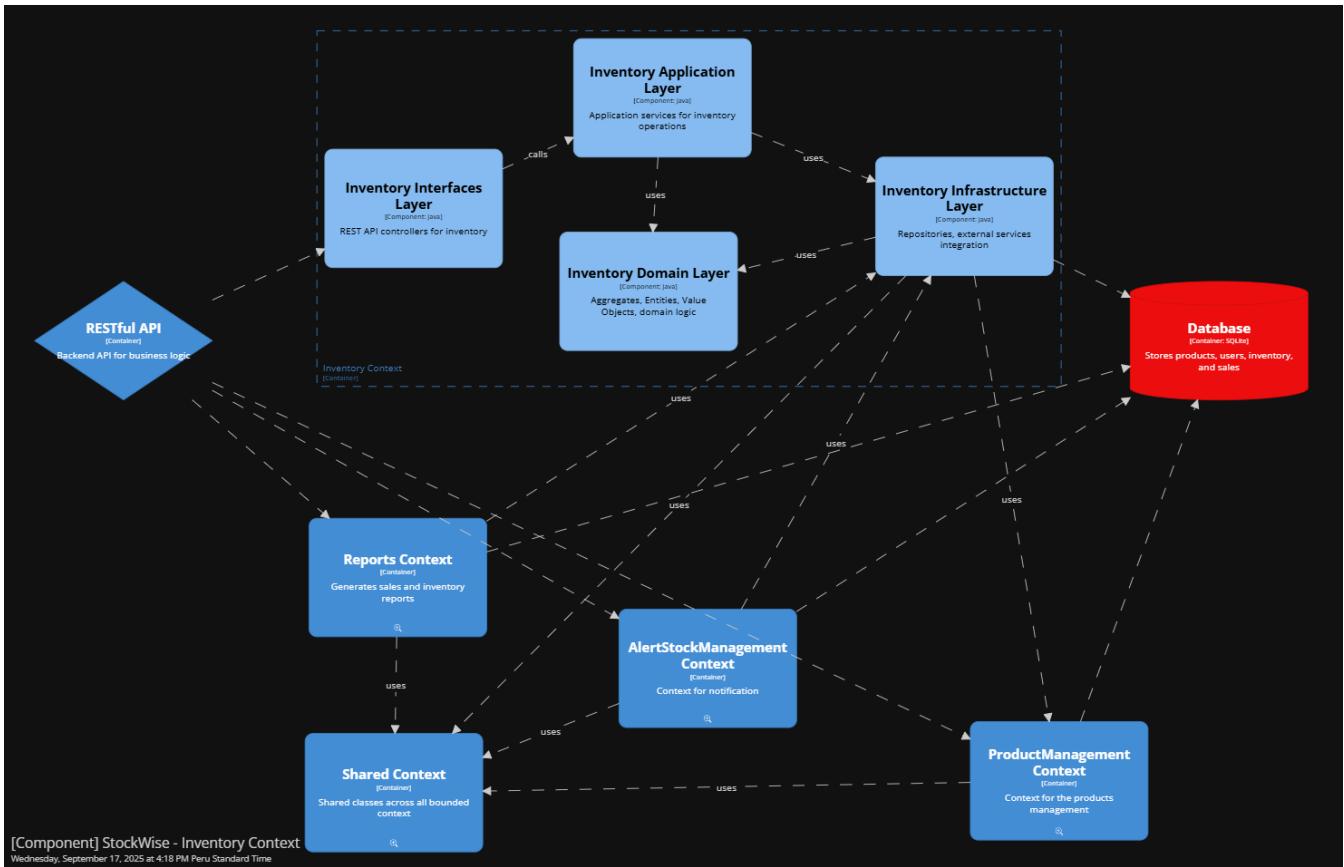
Clase	Tipo	Propósito	Atributos / Métodos
InventoryGeneralResource	Resource (DTO)	Representa el inventario general (productos y lotes)	Productos, Lotes
InventoryByProductResource	Resource (DTO)	Representa inventario por producto en respuestas	Id, Categoria, Producto, FechaEntrada, Cantidad, Precio, StockMinimo, UnidadMedida
InventoryByBatchResource	Resource (DTO)	Representa inventario por lote en respuestas	Id, Proveedor, Producto, FechaEntrada, Cantidad, Precio, Unidad, Total

Clase	Tipo	Propósito	Atributos / Métodos
InventoryTransform	Assembler	Convierte entidades de inventario a recursos DTO	ToResourceFromEntity(InventoryByProduct), ToResourceFromEntity(InventoryByBatch)
InventoryController	REST Controller	Expone endpoints de inventarios	GetAllInventory()

2.6.3.4. Infrastructure Layer

Clase	Tipo	Propósito	Atributos / Métodos
InventoryByProductRepository	Repository Implementation	Implementación acceso a inventario por producto (EF Core)	context, ListAsync(), FindByIdAsync(int), AddAsync(InventoryByProduct), DeleteAsync(int)
InventoryByBatchRepository	Repository Implementation	Implementación acceso a inventario por lote (EF Core)	context, ListAsync(), FindByIdAsync(int), AddAsync(InventoryByBatch), DeleteAsync(int)

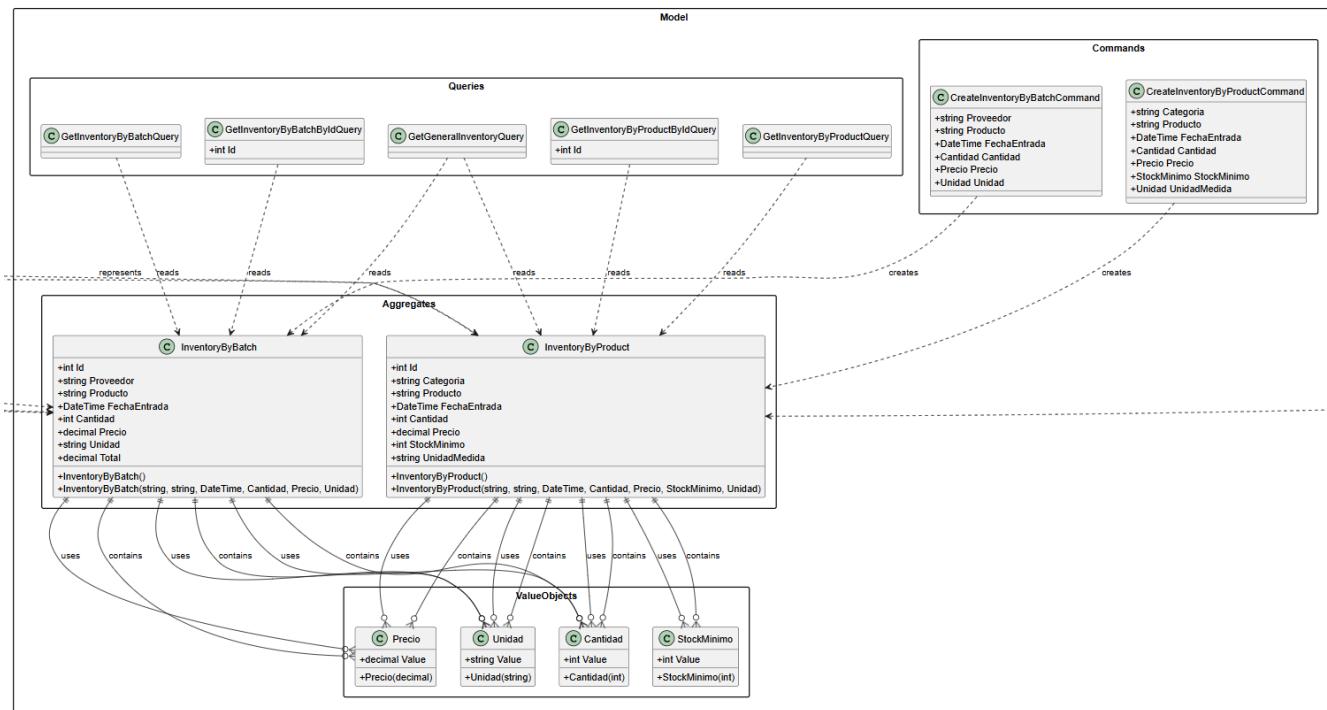
2.6.3.5. Bounded Context Software Architecture Component Level Diagrams



2.6.3.6. Bounded Context Software Architecture Code Level Diagrams

2.6.3.6.1. Bounded Context Domain Layer Class Diagrams

Domain: Model



Domain: Services



2.6.3.6.2. Bounded Context Database Design Diagram



2.6.4.1. Domain Layer

Clase/Interfaz	Tipo	Propósito	Atributos / Métodos
Product	Aggregate Root (Entity)	Representa un producto dentro del sistema	Id, Name, Description, PurchasePrice, SalePrice, InternalNotes, CategoryId, UnitId, Category, Unit, ProductTags, Product(), Product(...), Product(CreateProductCommand), UpdateProduct(...), AddTag(int), RemoveTag(int)
Category	Entity	Agrupa productos por categoría	Id, Name, Products, Category(), Category(string), UpdateName(string)
Unit	Entity	Representa la unidad de medida de un producto	Id, Name, Abbreviation, Products, Unit(), Unit(string, string), UpdateUnit(string, string)

Clase/Interfaz	Tipo	Propósito	Atributos / Métodos
ProductTag	Entity (Join)	Relaciona productos con etiquetas	ProductId, TagId, ProductTag(int, int)
Tag	Entity	Representa una etiqueta asociada a productos	Id, Name, Description, Tag(), Tag(string, string), UpdateTag(string, string)
Combo	Entity	Representa un combo de productos	Id, Name, Description, Price, ComboItems, Combo(), Combo(...), UpdateCombo(...)
ComboItem	Entity	Relaciona un producto con un combo	Id, Comboid, ProductId, Quantity, Combo, Product, ComboItem(), ComboItem(int, int, int), UpdateQuantity(int)
ProductId	Value Object	Identificador de producto	Value
UnitId	Value Object	Identificador de unidad	Value
TagId	Value Object	Identificador de etiqueta	Value
CategoryId	Value Object	Identificador de categoría	Value
CreateProductCommand	Command	Crear un nuevo producto	Name, Description, PurchasePrice, SalePrice, InternalNotes, CategoryId, UnitId, TagIds
UpdateProductCommand	Command	Actualizar datos de un producto	Id, Name, Description, PurchasePrice, SalePrice, InternalNotes, CategoryId, UnitId
DeleteProductCommand	Command	Eliminar un producto	Id
CreateComboCommand	Command	Crear un combo	Name, Description, Price
UpdateComboCommand	Command	Actualizar datos de un combo	Id, Name, Description, Price
CreateComboItemCommand	Command	Crear ítem en un combo	Comboid, ProductId, Quantity
UpdateComboItemCommand	Command	Actualizar ítem de un combo	Id, Quantity
DeleteComboItemCommand	Command	Eliminar ítem de un combo	Id
DeleteComboCommand	Command	Eliminar un combo	Id
GetProductByIdQuery	Query	Consultar producto por Id	Id
GetAllProductsQuery	Query	Consultar todos los productos	-
GetProductsByCategoryQuery	Query	Consultar productos por categoría	CategoryId
GetProductsByTagQuery	Query	Consultar productos por etiqueta	TagId
GetProductsByPriceRangeQuery	Query	Consultar productos por rango de precios	MinPrice, MaxPrice

Clase/Interfaz	Tipo	Propósito	Atributos / Métodos
GetComboByIdQuery	Query	Consultar combo por Id	Id
 GetAllCombosQuery	Query	Consultar todos los combos	-
GetComboItemByIdQuery	Query	Consultar ítem de combo por Id	Id
GetComboItemsByComboidQuery	Query	Consultar ítems de un combo	Comboid
IProductCommandService	Domain Service (Interface)	Define el contrato para manejar comandos de productos	Handle(CreateProductCommand), Handle(UpdateProductCommand), Handle>DeleteProductCommand)
IProductQueryService	Domain Service (Interface)	Define el contrato para consultas de productos	Handle(GetProductByIdQuery), Handle(GetAllProductsQuery), Handle(GetProductsByCategoryQuery), Handle(GetProductsByTagQuery), Handle(GetProductsByPriceRangeQuery)
IComboCommandService	Domain Service (Interface)	Define el contrato para comandos de combos	Handle(CreateComboCommand), Handle(UpdateComboCommand)
IComboQueryService	Domain Service (Interface)	Contrato para consultas de combos	GetAllAsync()
IUnitQueryService	Domain Service (Interface)	Contrato para consultas de unidades	GetAllAsync()
ITagQueryService	Domain Service (Interface)	Contrato para consultas de etiquetas	GetAllAsync()
IProductRepository	Repository (Interface)	Acceso abstracto a productos	FindProductNameAsync, FindAllProductsAsync, FindProductsByCategoryAsync, FindProductsByTagAsync, FindProductsByPriceRangeAsync, FindProductWithRelationsAsync
ICategoryRepository	Repository (Interface)	Acceso abstracto a categorías	FindCategoryByNameAsync, FindAllCategoriesAsync
IUnitRepository	Repository (Interface)	Acceso abstracto a unidades	FindUnitByNameAsync, FindAllUnitsAsync
ITagRepository	Repository (Interface)	Acceso abstracto a etiquetas	FindTagByNameAsync, FindAllTagsAsync, FindTagsByIdsAsync
IComboRepository	Repository (Interface)	Acceso abstracto a combos	FindComboByNameAsync, FindAllCombosAsync

2.6.4.2. Application Layer

Clase	Tipo	Propósito	Atributos / Métodos
ProductCommandService	Application Service	Implementa comandos de producto	productRepository, tagRepository, unitOfWork, Handle(CreateProductCommand), Handle(UpdateProductCommand)
ProductQueryService	Application Service	Implementa consultas de productos	productRepository, Handle(GetProductByIdQuery), Handle(GetAllProductsQuery), Handle(GetProductsByCategoryQuery), Handle(GetProductsByTagQuery), Handle(GetProductsByPriceRangeQuery)
ComboCommandService	Application Service	Implementa comandos de combos	comboRepository, unitOfWork, Handle(CreateComboCommand), Handle(UpdateComboCommand)
ComboQueryService	Application Service	Consultas sobre combos	comboRepository, GetAllAsync()

Clase	Tipo	Propósito	Atributos / Métodos
UnitQueryService	Application Service	Consultas sobre unidades	unitRepository, GetAllAsync()
TagQueryService	Application Service	Consultas sobre etiquetas	tagRepository, GetAllAsync()

2.6.4.3. Interface Layer

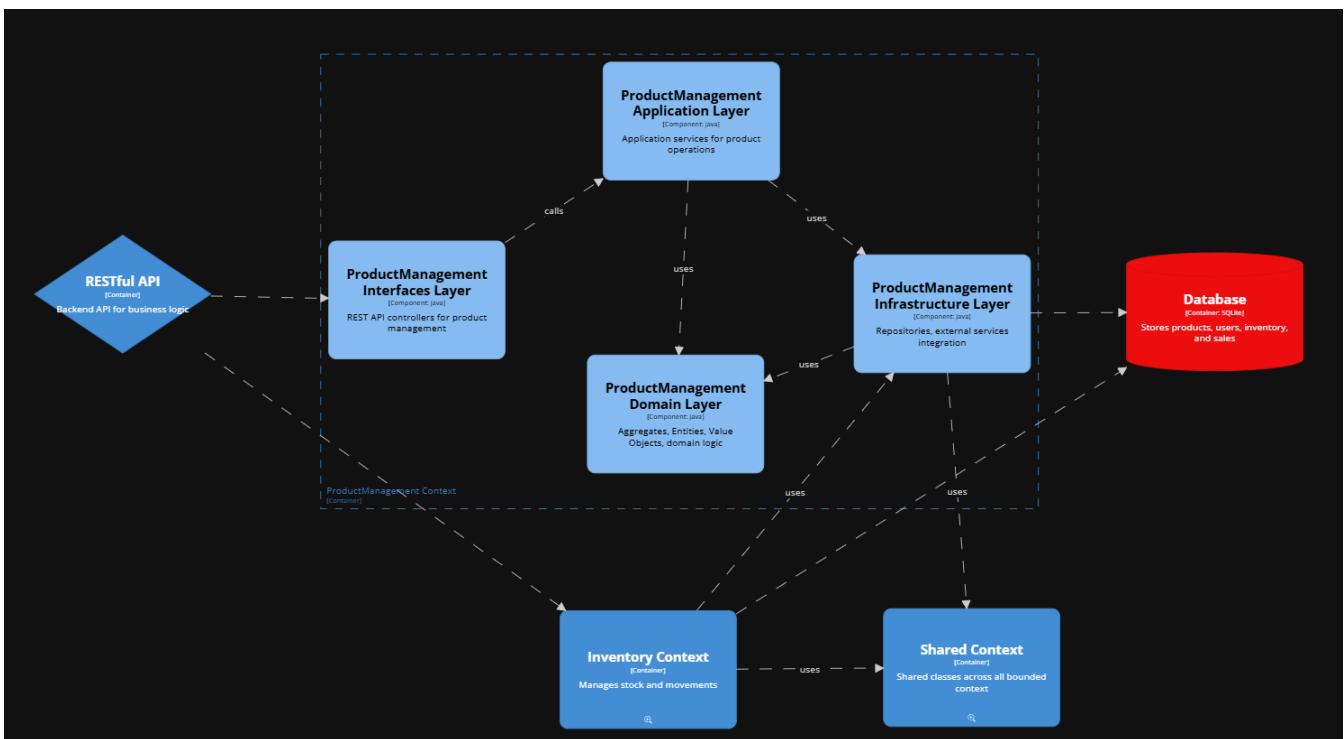
Clase	Tipo	Propósito	Atributos / Métodos
CreateProductResource	Resource (DTO)	Datos de entrada para crear producto	Name, Description, PurchasePrice, SalePrice, InternalNotes, CategoryId, UnitId, TagIds
UpdateProductResource	Resource (DTO)	Datos de entrada para actualizar producto	Id, Name, Description, PurchasePrice, SalePrice, InternalNotes, CategoryId, UnitId
ProductResource	Resource (DTO)	Representa un producto completo en respuesta	Id, Name, Description, PurchasePrice, SalePrice, InternalNotes, CategoryId, UnitId, Category, Unit, Tags
CategoryResource	Resource (DTO)	Representa una categoría	Id, Name
UnitResource	Resource (DTO)	Representa una unidad	Id, Name, Abbreviation
TagResource	Resource (DTO)	Representa una etiqueta	Id, Name, Description
ComboResource	Resource (DTO)	Representa un combo	Id, Name, Description, Price
CreateComboResource	Resource (DTO)	Datos de entrada para crear combo	Name, Description, Price
CreateProductCommandFromResourceAssembler	Assembler	Convierte recurso a comando	ToCommandFromResource(CreateProductResource)
UpdateProductCommandFromResourceAssembler	Assembler	Convierte recurso a comando	ToCommandFromResource(UpdateProductResource)
ProductResourceFromEntityAssembler	Assembler	Convierte entidad a recurso DTO	ToResourceFromEntity(Product)
ProductsController	REST Controller	Expone endpoints de productos	GetProductById, CreateProduct, GetAllProducts, GetProductsByCategory, GetProductsByTag
UnitsController	REST Controller	Expone endpoints de unidades	GetAllUnits
TagsController	REST Controller	Expone endpoints de etiquetas	GetAllTags
CombosController	REST Controller	Expone endpoints de combos	GetComboById, CreateCombo, GetAllCombos

2.6.4.4. Infrastructure Layer

Clase	Tipo	Propósito	Atributos / Métodos
ProductRepository	Repository Implementation	Implementación acceso a productos (EF Core)	context, FindProductByNameAsync, FindAllProductsAsync, FindProductsByCategoryAsync, FindProductsByTagAsync, FindProductsByPriceRangeAsync, FindProductWithRelationsAsync
CategoryRepository	Repository Implementation	Implementación acceso a categorías (EF Core)	context, FindCategoryByNameAsync, FindAllCategoriesAsync

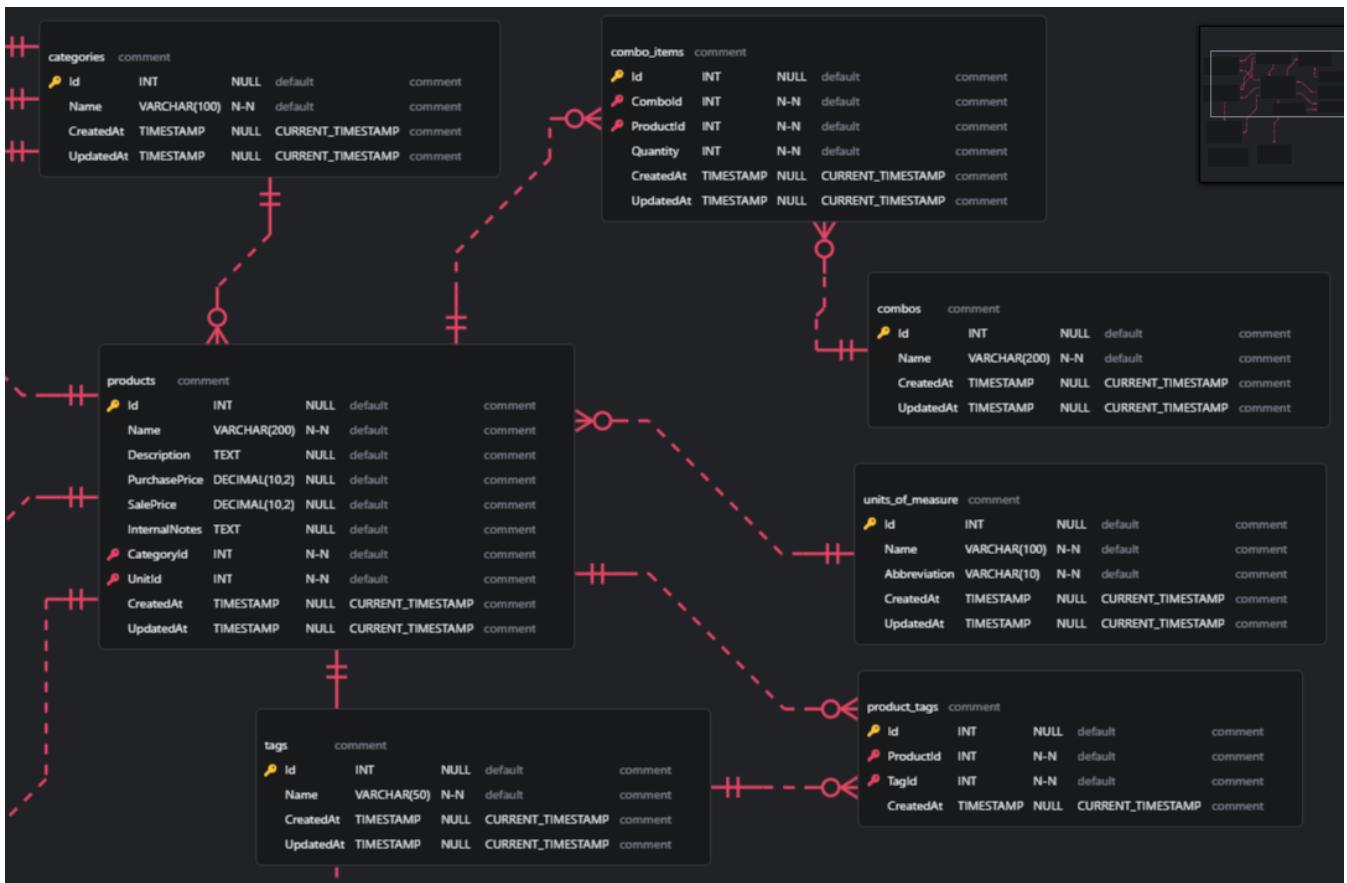
Clase	Tipo	Propósito	Atributos / Métodos
UnitRepository	Repository Implementation	Implementación acceso a unidades (EF Core)	context, FindUnitByNameAsync, FindAllUnitsAsync
TagRepository	Repository Implementation	Implementación acceso a etiquetas (EF Core)	context, FindTagByNameAsync, FindAllTagsAsync, FindTagsByIdsAsync
ComboRepository	Repository Implementation	Implementación acceso a combos (EF Core)	context, FindComboByNameAsync, FindAllCombosAsync

2.6.4.5. Bounded Context Software Architecture Component Level Diagrams

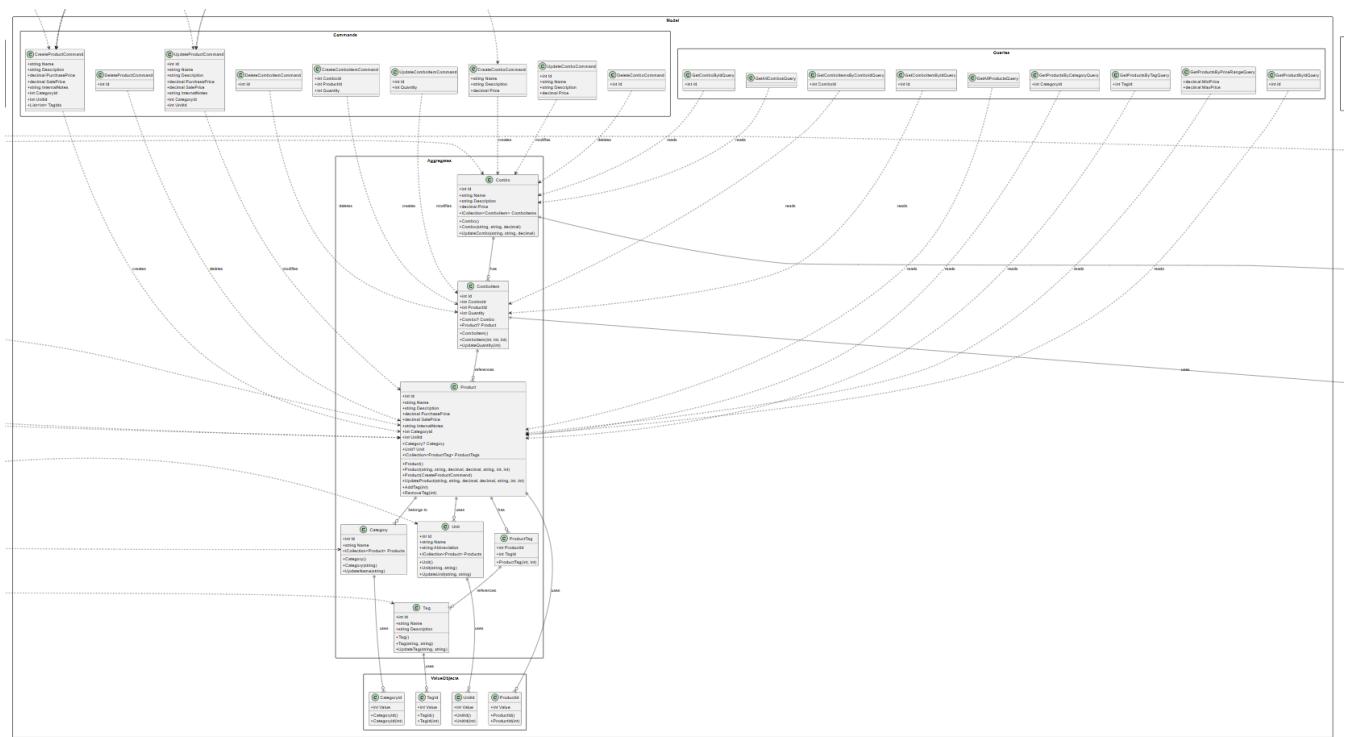


2.6.4.6. Bounded Context Software Architecture Code Level Diagrams

2.6.4.6.1. Bounded Context Domain Layer Class Diagrams



Domain: Model



Domain Services



2.6.4.6.2. Bounded Context Database Design Diagram

2.6.5. Bounded Context: Reports Context

2.6.5.1. Domain Layer

Clase/Interfaz	Tipo	Propósito	Atributos / Métodos
CategoryReport	Aggregate Root (Entity)	Reporte de productos por categoría	Id, Categoria, Producto, FechaConsulta, PrecioUnitario, Cantidad, Total, CategoryReport(), CategoryReport(...), UpdateCantidad(int), UpdatePrecio(decimal)
StockAverageReport	Aggregate Root (Entity)	Reporte de stock promedio por producto y categoría	Id, StockPromedio, Categoria, Producto, FechaConsulta, StockIdeal, Estado, StockAverageReport(...)
ReportType	Enum	Define el tipo de reporte	Category, StockAverage
ReportDate	Value Object	Fecha de un reporte	Value, ReportDate(DateTime)
StockStatus	Value Object	Estado de stock	Value, StockStatus(string)
ProductName	Value Object	Nombre del producto	Value, ProductName(string)
CategoryName	Value Object	Nombre de la categoría	Value, CategoryName(string)
CreateCategoryReportCommand	Command	Crear reporte por categoría	Categoría, Producto, FechaConsulta, PrecioUnitario, Cantidad
CreateStockAverageReportCommand	Command	Crear reporte de stock promedio	StockPromedio, Categoria, Producto, FechaConsulta, StockIdeal, Estado
GetAllCategoryReportsQuery	Query	Consultar todos los reportes de categoría	-
GetCategoryReportsByDateQuery	Query	Consultar reportes de categoría por fecha	FechaConsulta
GetCategoryReportByIdQuery	Query	Consultar reporte de categoría por Id	Id
 GetAllStockAverageReportsQuery	Query	Consultar todos los reportes de stock promedio	-
GetStockAverageReportsByDateQuery	Query	Consultar reportes de stock promedio por fecha	FechaConsulta
GetStockAverageReportByIdQuery	Query	Consultar reporte de stock promedio por Id	Id
ICategoryReportCommandService	Domain Service (Interface)	Define contrato para comandos de reportes de categoría	Handle(CreateCategoryReportCommand)
ICategoryReportQueryService	Domain Service (Interface)	Define contrato para consultas de reportes de categoría	Handle(GetAllCategoryReportsQuery), Handle(GetCategoryReportsByDateQuery), Handle(GetCategoryReportByIdQuery)
IStockAverageReportCommandService	Domain Service (Interface)	Define contrato para comandos de reportes de stock promedio	Handle(CreateStockAverageReportCommand)
IStockAverageReportQueryService	Domain Service (Interface)	Define contrato para consultas de reportes de stock promedio	Handle(GetAllStockAverageReportsQuery), Handle(GetStockAverageReportsByDateQuery)
ICategoryReportRepository	Repository (Interface)	Acceso a reportes de categoría	FindAllAsync(), FindByDateAsync(DateTime), AddAsync(CategoryReport), FindByIdAsync(int)

Clase/Interfaz	Tipo	Propósito	Atributos / Métodos
IStockAverageReportRepository	Repository (Interface)	Acceso a reportes de stock promedio	FindAllAsync(), FindByDateAsync(DateTime), AddAsync(StockAverageReport), FindByIdAsync(int)

2.6.5.2. Application Layer

Clase	Tipo	Propósito	Atributos / Métodos
CategoryReportCommandService	Application Service	Implementa comandos de reportes de categoría	repository, unitOfWork, Handle(CreateCategoryReportCommand)
StockAverageReportCommandService	Application Service	Implementa comandos de reportes de stock promedio	repository, unitOfWork, Handle(CreateStockAverageReportCommand)
CategoryReportQueryService	Application Service	Implementa consultas de reportes de categoría	repository, Handle(GetAllCategoryReportsQuery), Handle(GetCategoryReportsByDateQuery), Handle(GetCategoryReportByIdQuery)
StockAverageReportQueryService	Application Service	Implementa consultas de reportes de stock promedio	repository, Handle(GetAllStockAverageReportsQuery), Handle(GetStockAverageReportsByDateQuery)

2.6.5.3. Interface Layer

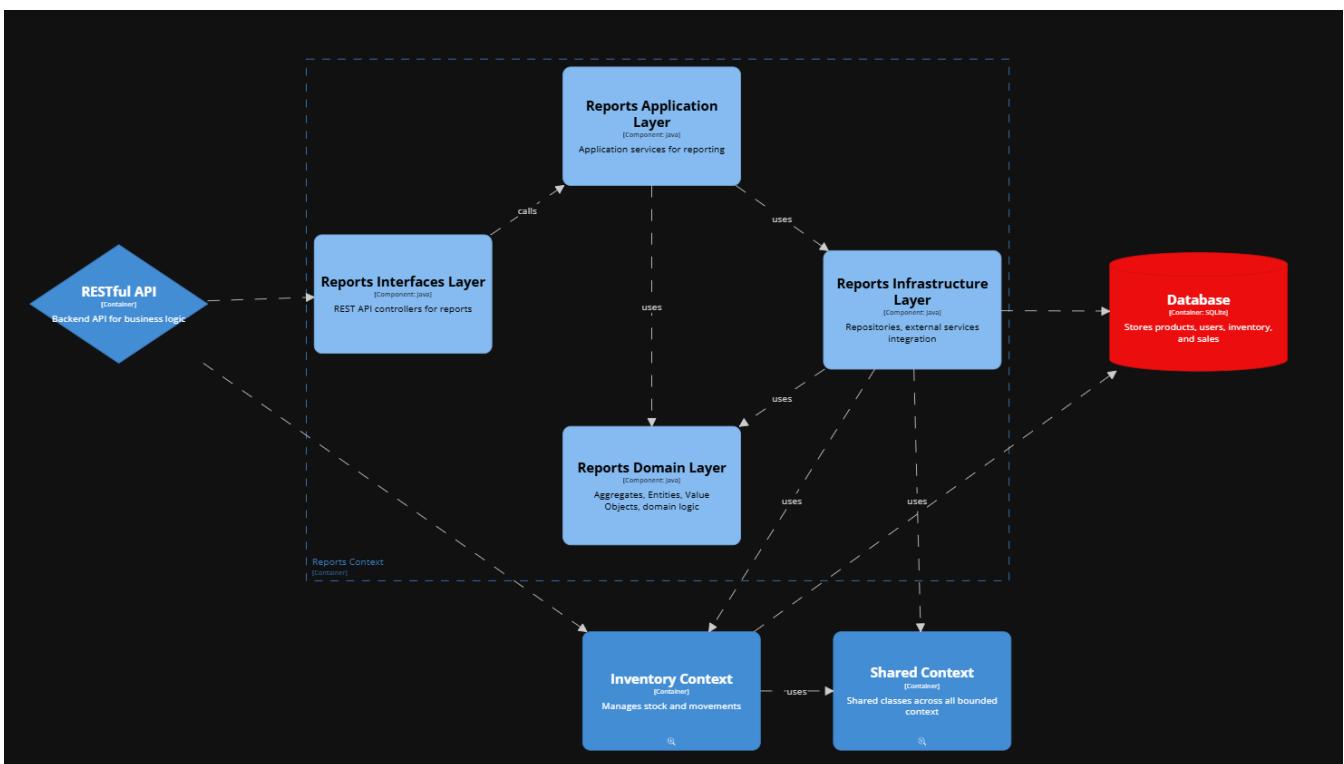
Clase	Tipo	Propósito	Atributos / Métodos
GeneralReportResource	Resource (DTO)	Representa todos los reportes (categoría y stock promedio)	CategoryReports, StockAverageReports
CategoryReportResource	Resource (DTO)	Representa reporte de categoría en respuestas	Id, Categoria, Producto, FechaConsulta, PrecioUnitario, Cantidad, Total
StockAverageReportResource	Resource (DTO)	Representa reporte de stock promedio en respuestas	Id, StockPromedio, Categoria, Producto, FechaConsulta, StockIdeal, Estado
CreateCategoryReportResource	Resource (DTO)	Datos de entrada para crear un reporte de categoría	Categoria, Producto, FechaConsulta, PrecioUnitario, Cantidad
CreateStockAverageReportResource	Resource (DTO)	Datos de entrada para crear un reporte de stock promedio	StockPromedio, Categoria, Producto, FechaConsulta, StockIdeal, Estado
CategoryReportResourceAssembler	Assembler	Convierte entidad CategoryReport en DTO	ToResource(CategoryReport)
StockAverageReportResourceAssembler	Assembler	Convierte entidad StockAverageReport en DTO	ToResource(StockAverageReport)
CreateCategoryReportCommandAssembler	Assembler	Convierte recurso en comando	ToCommand(CreateCategoryReportResource)
CreateStockAverageReportCommandAssembler	Assembler	Convierte recurso en comando	ToCommand(CreateStockAverageReportResource)

Clase	Tipo	Propósito	Atributos / Métodos
ReportController	REST Controller	Expone endpoints para reportes	GetAllReports(), CreateCategory(CreateCategoryReportResource), GetAllCategoryReports(), GetCategoryReportById(int), GetCategoryByDate(DateTime), CreateStockAverage(CreateStockAverageReportResource), GetAllStockAverageReports(), GetStockAverageReportById(int), GetStockAverageByDate(DateTime)

2.6.5.4. Infrastructure Layer

Clase	Tipo	Propósito	Atributos / Métodos
CategoryReportRepository	Repository Implementation	Implementación EF Core para reportes de categoría	context, FindAllAsync(), FindByDateAsync(DateTime), AddAsync(CategoryReport), FindByIdAsync(int)
StockAverageReportRepository	Repository Implementation	Implementación EF Core para reportes de stock promedio	context, FindAllAsync(), FindByDateAsync(DateTime), AddAsync(StockAverageReport), FindByIdAsync(int)

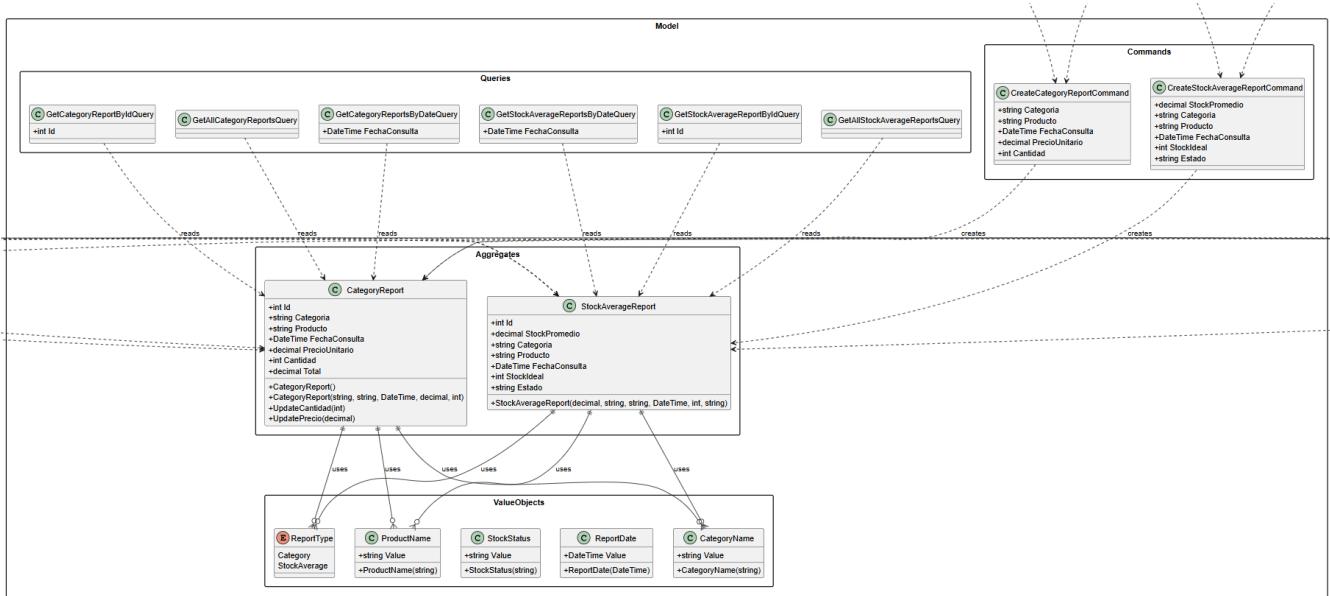
2.6.5.5. Bounded Context Software Architecture Component Level Diagrams



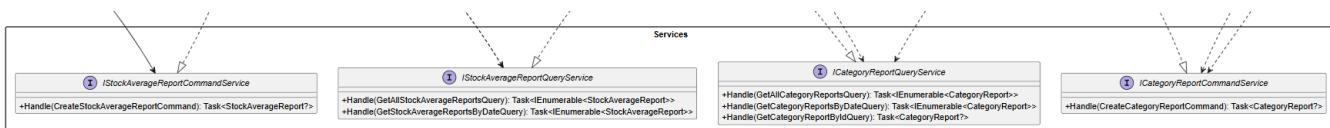
2.6.5.6. Bounded Context Software Architecture Code Level Diagrams

2.6.5.6.1. Bounded Context Domain Layer Class Diagrams

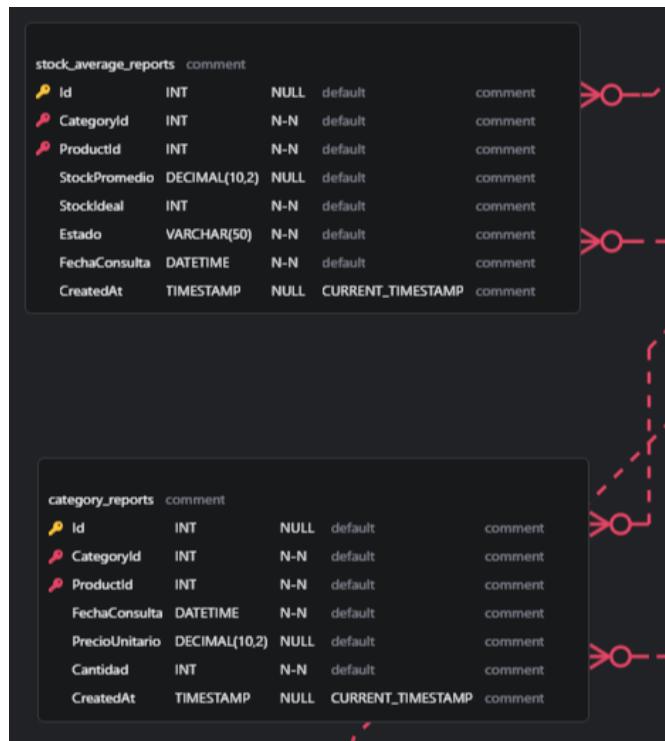
Domain: Model



Domain: Services



2.6.5.6.2. Bounded Context Database Design Diagram



Capítulo III: Solution UI/UX Design

3.1. Product design

3.1.1. Style Guidelines

Un "style guideline" o guía de estilo es un conjunto de reglas y pautas que establecen la forma en que se deben escribir, diseñar o presentar documentos, contenido web, software, o cualquier otro tipo de trabajo creativo. A continuación, se otorga especificación a los parámetros implementados en la estructura del proyecto:

3.1.1.1. General Style Guidelines

Branding

StockWise es una marca pensada para ofrecer confianza, cercanía y eficiencia a pequeñas y medianas empresas, especialmente bodegas de barrio y emprendimientos. El branding refleja accesibilidad, modernidad y calidez. El enfoque está en facilitar la transformación digital de la gestión de inventarios con una interfaz clara, amigable y funcional.

Logotipo

El logo combina una bodega estilizada con colores cálidos y un código de barras, representando tanto la esencia física del negocio como la modernización a través de tecnología.

- La palabra "Stock" en rojo (#BC162A) resalta la acción y lo esencial del inventario.
- La palabra "Wise" en marrón oscuro (#302325) sugiere inteligencia y fiabilidad.

El código de barras integrado representa la gestión estructurada y la automatización del stock.



Tono de comunicación

Nos comunicamos como lo haría un buen amigo del barrio, claro, sin complicaciones, y con buena onda. Queremos transmitir una sensación de seguridad y eficiencia, mientras mantenemos una comunicación cercana y amigable.

Lenguaje (Language):

Usamos términos fáciles de entender: "productos", "alertas", "entradas/salidas", "usuarios". Evitamos jerga técnica innecesaria o anglicismos si no son esenciales.

Colores

La paleta de colores de StockWise ha sido cuidadosamente seleccionada para transmitir profesionalismo, confianza y accesibilidad.

Primary Colors:

- #BC162A (Rojo intenso): Botones principales, acciones críticas, alertas importantes
- #EE7F27 (Naranja vibrante): Botones secundarios, estados activos, notificaciones
- #302325 (Marrón oscuro): Texto principal, headers, elementos de navegación

Background Colors:

- #F5E1A4 (Fondo cálido claro): Fondo general de la aplicación
- #D9D593 (Gris suave): Fondos de secciones, cards secundarios

Functional Colors:

- #27A300 (Verde éxito): Confirmaciones, estados positivos
- #FFC107 (Amarillo advertencia): Alertas moderadas, advertencias
- #DC3545 (Rojo error): Estados de error, acciones destructivas



Tipografía

La elección tipográfica para StockWise es un componente esencial que complementa la identidad visual de la marca. Se han seleccionado dos familias tipográficas que juntas ofrecen versatilidad y coherencia, asegurando que la comunicación sea clara y efectiva en todos los medios.

Font Families:

- Inter: Para textos largos, body copy y contenido principal
- Nunito: Para headers, botones y elementos de interfaz

Font Scale (Mobile First):

- h1: 24px / 1.1 / Bold
- h2: 20px / 1.1 / SemiBold
- h3: 18px / 1.2 / Medium
- Body Large: 17px / 1.4 / Regular
- Body: 16px / 1.4 / Regular
- Small: 14px / 1.4 / Regular
- Caption: 12px / 1.3 / Regular

Weights:

- Bold (700): Títulos principales, acciones críticas
- SemiBold (600): Subtítulos, botones importantes
- Medium (500): Etiquetas, elementos interactivos
- Regular (400): Texto body, contenido principal
- Light (300): Texto secundario, descripciones

Border Radius:

- Small: 8px (botones pequeños, inputs)
- Medium: 12px (cards, modales)
- Large: 16px (containers principales)

Spacing System (8px base):

- xs: 4px
- sm: 8px
- md: 16px
- lg: 24px
- xl: 32px
- xxl: 48px

Shadows:

- Low: 0 2px 4px rgba(48, 35, 37, 0.1)
- Medium: 0 4px 8px rgba(48, 35, 37, 0.15)
- High: 0 8px 16px rgba(48, 35, 37, 0.2)

3.1.2. Information Architecture

La arquitectura de la información, también conocida como Information Architecture (IA), implica la organización de la información de manera clara y lógica, de modo que los usuarios puedan comprender su ubicación, lo que han descubierto, qué pueden esperar y qué está disponible a su alrededor. Esto tiene como objetivo permitir a los usuarios encontrar con facilidad lo que están buscando, y a los clientes, comprender las capacidades de la plataforma. Además, la arquitectura de la información posibilita la incorporación de nuevas funciones y la expansión del producto sin generar una estructura compleja o de difícil comprensión (Rosenfeld, Morville & Arango 2015).

3.1.2.1. Organization Systems

La interfaz se divide en módulos bien definidos, accesibles desde un panel de navegación estructurado jerárquicamente. Estos módulos incluyen: Inicio, Home, inventario y Configuración. Cada sección agrupa funciones específicas según su propósito, permitiendo que las tareas clave estén siempre al alcance del usuario.

3.1.2.1. Organization Systems

La arquitectura de organización de StockWise está diseñada siguiendo principios de agrupación lógica y progresiva de la información, permitiendo a los usuarios acceder rápidamente a las funciones necesarias según su rol y contexto de uso.

Estructura Organizacional Principal

Módulo	Descripción	Funciones Principales	Acceso por Rol
Dashboard Principal	Vista consolidada del estado del inventario y métricas clave	<ul style="list-style-type: none"> - Resumen ejecutivo de stock - Alertas prioritarias - Acciones rápidas - Métricas en tiempo real 	Admin, Encargado
Gestión de Inventario	Núcleo operativo para administración completa de productos	<ul style="list-style-type: none"> - Registro y edición de productos - Control de niveles de stock - Categorización y etiquetado - Gestión de ubicaciones físicas 	Admin, Encargado, Empleado
Operaciones Diarias	Módulo para transacciones y movimientos regulares	<ul style="list-style-type: none"> - Registro de entradas/salidas - Ajustes de inventario - Historial de movimientos - Devoluciones y mermas 	Admin, Encargado, Empleado
Reportes y Analytics	Sistema de generación y visualización de datos	<ul style="list-style-type: none"> - Reportes personalizados - Análisis de tendencias - Métricas de rendimiento - Exportación de datos 	Admin, Encargado
Configuración del Sistema	Administración de preferencias y usuarios	<ul style="list-style-type: none"> - Gestión de perfiles de usuario - Configuración de empresa - Preferencias de notificaciones - Backup y seguridad 	Admin

Principios de Organización Aplicados

1. Agrupación por Funcionalidad

Las características se organizan según su propósito común, facilitando la asociación mental y reduciendo la carga cognitiva.

2. Jerarquía Visual Progresiva

La información se presenta desde lo general hacia lo específico, permitiendo drill-down controlado según las necesidades del usuario.

3. Contextualización Dinámica

Las opciones disponibles se adaptan según el rol del usuario y el módulo activo, mostrando solo las funciones relevantes.

4. Consistencia Transversal

Mismos patrones organizativos se aplican en todos los módulos, creando una experiencia unificada y predecible.

Organización de Contenido por Módulo

Módulo de Inventario

- **Agrupación Primaria:** Por estado de stock (Normal, Bajo, Crítico)
- **Agrupación Secundaria:** Por categorías de producto
- **Agrupación Terciaria:** Por ubicación física en bodega

Módulo de Reportes

- **Agrupación Temporal:** Diario, Semanal, Mensual
- **Agrupación por Métrica:** Ventas, Stock, Rentabilidad
- **Agrupación por Producto:** Individual, Por categoría, Global

3.1.2.2. Labelling Systems

El sistema de etiquetado en StockWise sigue principios de claridad, consistencia y contexto, asegurando que los usuarios comprendan inmediatamente la función de cada elemento.

Principios de Etiquetado:

- Lenguaje Natural: Usamos términos del dominio del usuario ("productos", "proveedores", "ventas")
- Consistencia: Mismo término para misma función en toda la aplicación
- Jerarquía Visual: Tamaño y peso tipográfico reflejan importancia
- Contexto: Las etiquetas cambian según el módulo y las acciones disponibles

Sistema de Iconografía:

- **Acciones Principales:**

- Agregar/Registrar
- Editar/Modificar
- Eliminar/Descartar
- Buscar/Filtrar
- Exportar/Compartir

- **Módulos y Secciones:**

- Dashboard - Vista general
- Inventario - Gestión de productos
- Movimientos - Entradas y salidas
- Reportes - Analytics y métricas
- Configuración - Ajustes del sistema

- **Estados del Sistema:**

- Completado/Éxito
- Advertencia/Alerta
- Error/Problema
- Procesando/En curso

Microcopy y Mensajes:

- **Botones de Acción:** "Agregar Producto", "Registrar Entrada", "Generar Reporte"
- **Mensajes de Confirmación:** "¿Estás seguro de eliminar este producto?"
- **Estados Vacíos:** "Aún no tienes productos registrados"
- **Guías Contextuales:** "Usa el código de barras para buscar más rápido"

3.1.2.3. SEO Tags and Meta Tags

SEO Tags

SEO (Search Engine Optimization) Tags son elementos de HTML que ayudan a los motores de búsqueda a entender el contenido y la estructura de una página web. Estos tags influyen en cómo los motores de búsqueda indexan y clasifican tu sitio en los resultados de búsqueda.

Algunos ejemplos importantes de SEO Tags para StockWise:

Title Tags para Páginas Principales:

```
<!-- Página de Inicio -->
<title>StockWise - App de Gestión de Inventarios para PYMES</title>

<!-- Página de Características -->
<title>Gestión de Inventario Móvil | StockWise App</title>

<!-- Página de Precios -->
<title>Planes de Gestión de Inventario | StockWise</title>
```

Header Tags Estructurados:

```
<h1>Gestiona Tu Inventario desde tu Móvil</h1>
<h2>La solución simple para negocios en crecimiento</h2>
<h3>Control total de tu stock en tiempo real</h3>
```

Meta Descriptions Optimizadas:

```
<meta name="description" content="StockWise - App móvil de gestión de inventarios para PYMES. Controla stock, genera reportes y evita quiebres de inventario desde tu teléfono.">

<meta name="keywords" content="gestión inventarios, app móvil, control stock, PYMES, bodegas, emprendedores">
```

Open Graph Tags para Redes Sociales:

```
<meta property="og:title" content="StockWise - App de Inventarios">
<meta property="og:description" content="Gestiona tu inventario desde cualquier lugar con StockWise">
<meta property="og:image" content="/images/stockwise-social-preview.png">
```

3.1.2.4. Searching Systems

El sistema de búsqueda de StockWise está diseñado para ser intuitivo y potente, permitiendo a los usuarios encontrar rápidamente la información que necesitan en su inventario.

Características del Sistema de Búsqueda:

- **Búsqueda Inteligente:**
 - Búsqueda por texto libre en múltiples campos (nombre, - código, categoría, proveedor)
 - Búsqueda por código de barras usando la cámara del dispositivo
 - Búsqueda por voz para hands-free operation
 - Búsqueda predictiva con sugerencias en tiempo real
- **Resultados de Búsqueda:**
 - Ordenamiento por relevancia, nombre, stock, fecha
 - Vista de tarjetas con información esencial
 - Acciones rápidas desde los resultados
 - Historial de búsquedas recientes
- **Búsqueda por Lotes:**
 - Escaneo múltiple de códigos de barras
 - Reconocimiento de imágenes para productos sin código
 - Procesamiento offline con sincronización posterior

3.1.2.5. Navigation Systems

El sistema de navegación de StockWise está optimizado para dispositivos móviles, priorizando la accesibilidad y la eficiencia en las tareas diarias.

Patrones de Navegación Principal:

Bottom Navigation Bar:



[Inicio] [Inventario] [Agregar] [Reportes] [Perfil]

Navegación por Gestos:

- Deslizar hacia la derecha: Menú lateral
- Deslizar hacia abajo: Actualizar contenido
- Deslizar hacia arriba: Acciones rápidas
- Toque largo: Opciones contextuales

Navegación Contextual:

- Breadcrumbs para ubicación en estructuras profundas
- Botón "Atras" nativo del dispositivo
- Navegación por pestañas en secciones complejas
- Accesos directos personalizables según uso frecuente

Navegación para Accesibilidad:

- Soporte completo para lectores de pantalla
- Navegación por teclado en versiones tablet
- Tamaños de touch target mínimos de 44px
- Alto contraste y modos de daltonismo

3.1.3. Landing Page UI Design

3.1.3.1. Landing Page Wireframe

El prototipado de la landing page cuenta diversas secciones:

- Header: Incluye botones para facilitar la navegación
- Hero Section: Con un botón CTA principal, un título y una imagen.
- Sección de Beneficios / Características: mostrara un resumen de 3 beneficios que ofrece la app

- Planes / Precios: Tendrá 2 planes con su referente título y lista de características y contará con un botón CTA secundario por cada plan.
- Testimonios: Contará de un título y subtítulo, así como de unos testimonios de usuarios de cada segmento, lo que aumenta la confianza en los potenciales clientes.
- Llamado a la acción final (CTA grande) : Tiene un título y subtítulo, un botón CTA grande para "Try our app"
- Footer: Sección que da fin a la landing page, cuenta con las redes sociales de la plataforma.

Wireframe Desktop

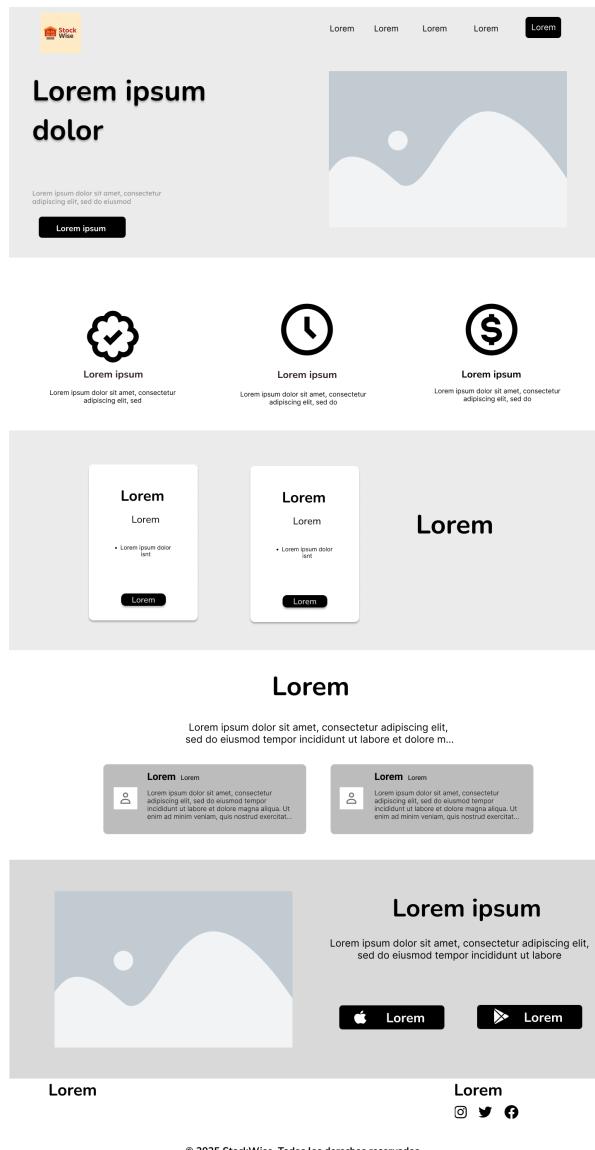


Imagen (Nº). Elaboración propia. Realizado en Figma

Wireframe Mobile

En la versión móvil el navbar se reemplaza por un menu desplegable.





Lorem ipsum

Lorem ipsum dolor sit amet,
consectetur adipiscing elit, sed



Lorem ipsum

Lorem ipsum dolor sit amet,
consectetur adipiscing elit



Lorem ipsum

Lorem ipsum dolor sit amet

Lorem

Lorem

Lorem

Lorem ipsum
dolor sit

Lorem

Lorem

Lorem

Lorem ipsum
dolor sit

Lorem

Lorem

Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do

John Doe Worker



Lorem ipsum dolor sit amet,
consectetur adipiscing elit, sed do
eiusmod tempor incididunt ut labore

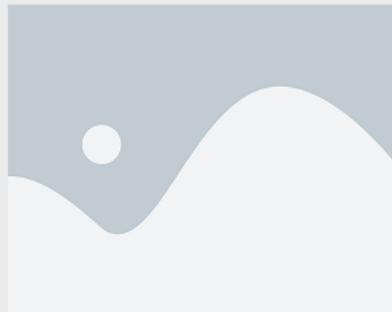
Sarah Miller Worker

Do

Lorem ipsum dolor sit amet,
consectetur adipiscing elit, sed do
eiusmod tempor incididunt ut labore

Lorem ipsum

Lorem ipsum dolor sit amet, consectetur adipiscing elit,
sed do eiusmod tempor incididunt ut labore



Lorem

Lorem

Lorem

Lorem ipsum dolor sit amet,
consectetur adipiscing elit

Lorem



© 2025 StockWise. Todos los derechos reservados.

Imagen (Nº). Elaboración propia. Realizado en Figma

3.1.3.2. Landing Page Mock-up

Wireframe Desktop



Boost your business with StockWise

Discover how our advanced features can transform your inventory management.



Plans that fit your business

Find the perfect plan for your needs. Start free and scale as you grow.



Need something custom? Contact us for enterprise plans

What our clients say

Thousands of companies trust StockWise to efficiently manage their inventory

"Thanks to StockWise, I now know exactly which products need restocking. Much more order and less stress!"
Carlos Rodriguez
Owner

"StockWise has completely transformed the way we manage inventory. It's like having a 24/7 assistant."
Maria Gonzalez
Owner

Try our App

The perfect app to manage your inventory intelligently and efficiently. Available for iOS and Android.

● Inventory control ● Real-time reports ● Smart management

Join thousands of users who already trust StockWise for their inventory management.

Download on the App Store GET IT ON Google Play



Stock Wise

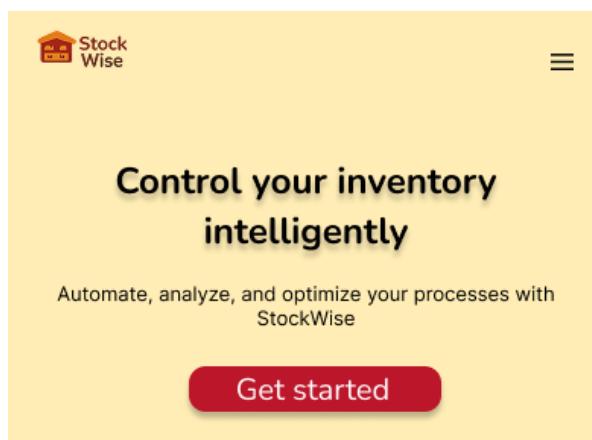


© 2025 StockWise. Todos los derechos reservados.

Imagen (Nº). Elaboración propia. Realizado en Figma

Wireframe Mobile

En la versión mobile el navbar se reemplaza por un menu desplegable.



Boost your business with StockWise

Discover how our advanced features can transform your inventory management.



Real-time stock management

Control your inventory instantly and make better decisions.



Smart reports

Visualize key data to optimize your business.



Integration with your system

Easily connect with your current tools.

Plans that fit your business

Find the perfect plan for your needs. Start free and scale as you grow.

Free

s./0

- Up to 30 registered products
- 1 administrator user
- Tag sorting
- Product search and filtering
- Discreet advertising

[Get started](#)

Entrepreneur Plan

s./30

- Up to 200 products
- Up to 2 users (admin + assistant)
- Basic statistics
- Category sorting
- No ads

[Access](#)

Business Plan

s./70

- Unlimited products
- Unlimited users
- Complete statistics and graphs
- Transaction history
- Minimum stock alerts

[Access](#)

What our clients say

Thousands of companies trust StockWise to efficiently manage their inventory

★★★★★
"Thanks to StockWise, I now know exactly which products need restocking. Much more order and less stress!"



Carlos Rodriguez
Worker

★★★★★
"StockWise has completely transformed the way we manage inventory. It's like having a 24/7 assistant."



Maria Gonzalez
Worker

Try our App

The perfect app for managing your inventory intelligently and efficiently

- Inventory control • Real-time reports
- Smart management

Join thousands of users who already trust Stock Wise.

[!\[\]\(fa6eaef7168d92d4c5e905c067c3b755_img.jpg\) Download on the App Store](#)
[!\[\]\(f6804ce3356c98fc6c3a345129e4e421_img.jpg\) GET IT ON Google Play](#)



Stock Wise



© 2025 StockWise. Todos los derechos reservados.

Imagen (Nº). Elaboración propia. Realizado en Figma

3.1.4. Mobile Applications UX/UI Design

3.1.4.1. Mobile Applications Wireframes

Los wireframes ayudan a los diseñadores y desarrolladores a planificar la arquitectura y la funcionalidad de la aplicación, permitiendo visualizar cómo los usuarios interactuarán con ella. Son esenciales en las primeras etapas de desarrollo, ya que facilitan la comunicación de ideas, la identificación de problemas potenciales y la alineación de todos los involucrados en el proyecto antes de pasar al diseño detallado y la programación.

Iniciar sesión

La siguiente imagen evidencia el wireframe de la sección de "inicio de sesión" desde la vista desktop web browser. En ella, los usuarios deberán ingresar las credenciales necesarias para poder acceder a la aplicación, siempre y cuando posean con una cuenta previamente creada.

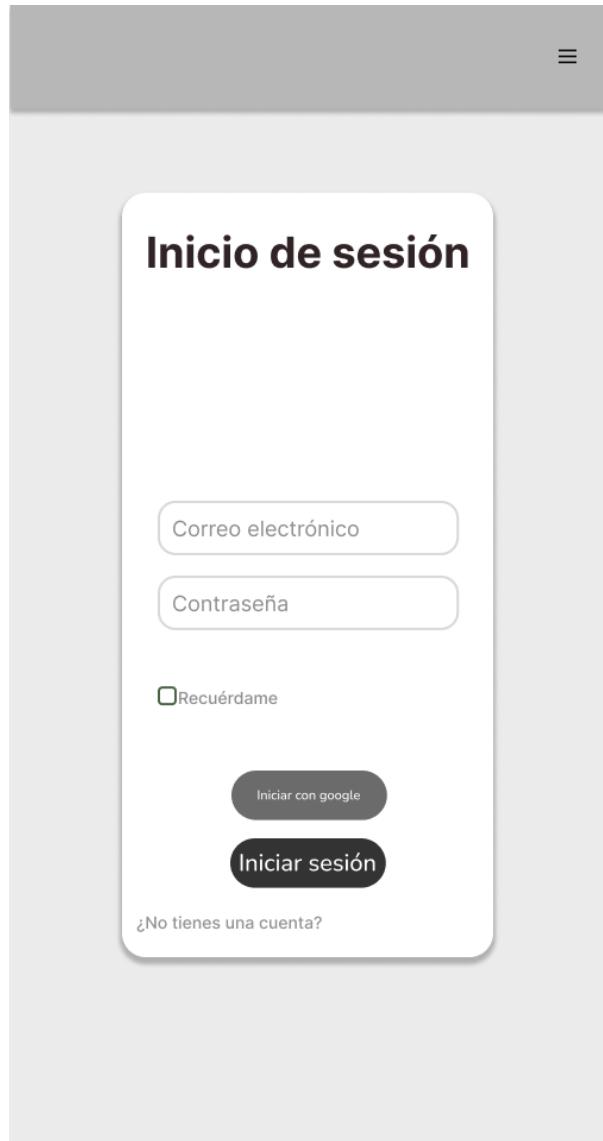


Imagen (Nº). Elaboración propia. Realizado en Figma

Crear cuenta

La siguiente imagen presenta la sección de “registro de usuario”, a través de la cual los administradores podrán remitir a la aplicación la información necesaria para crear una cuenta.

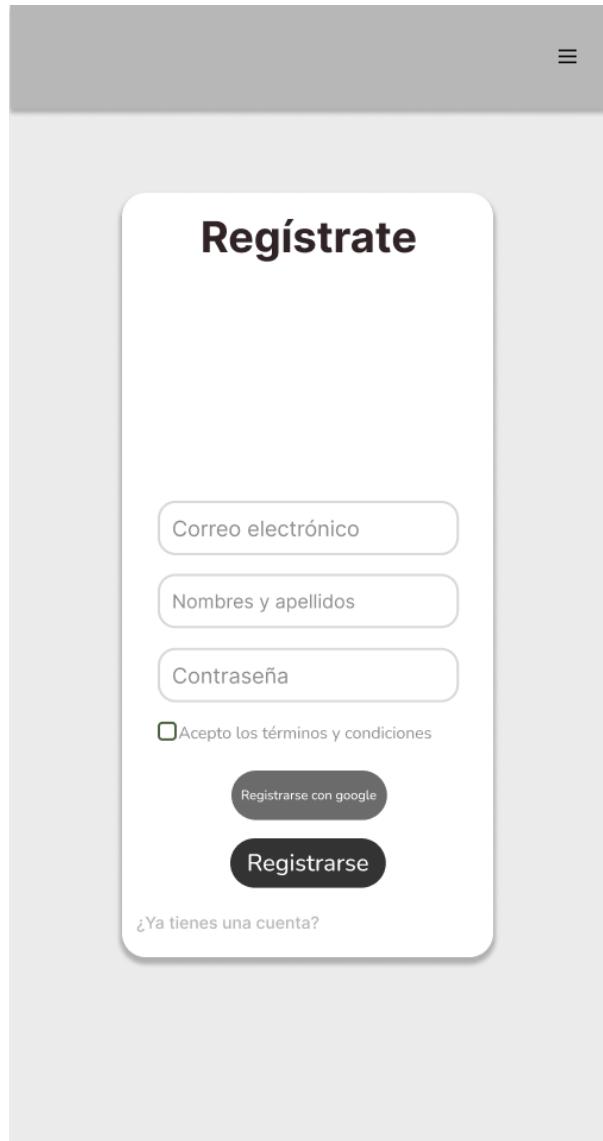


Imagen (Nº). Elaboración propia. Realizado en Figma

Elegir plan de pago

La siguiente imagen presenta la sección "Elegir plan de pago", la cual despliega las categorías de planes disponibles en la plataforma. Los tres planes están diseñados para adaptarse a las necesidades de los administradores de tienda acorde a sus necesidades.



Imagen (Nº). Elaboración propia. Realizado en Figma

Pasarela de pagos

La siguiente imagen corresponde a la sección “Pasarela de pagos” donde los usuarios son dirigidas a un formulario para seleccionar y registrar su método de pago. Una vez que el proceso de pago es completado exitosamente, se notifica al usuario con un mensaje confirmando el vinculo de su tarjeta con la plataforma. Del mismo modo, si el usuario desea retirar su información o actualizarlo podrá hacerlo a través de su perfil.



Imagen (Nº). Elaboración propia. Realizado en Figma

Mi perfil

La sección titulada "Mi perfil" solicita al usuario que complete información en dos categorías para registrar sus datos. Adicionalmente, la categoría "cambiar cuenta" ofrece la opción de permitir al usuario cambiar de cuenta a una de empleador y utilizar las herramientas de este por la plataforma. También, tiene libertad de cambiar el plan si es necesario.

Perfil Administrador

En este perfil, el usuario puede ver a la derecha sus herramientas correspondientes a su cuenta.



Imagen (Nº). Elaboración propia. Realizado en Figma

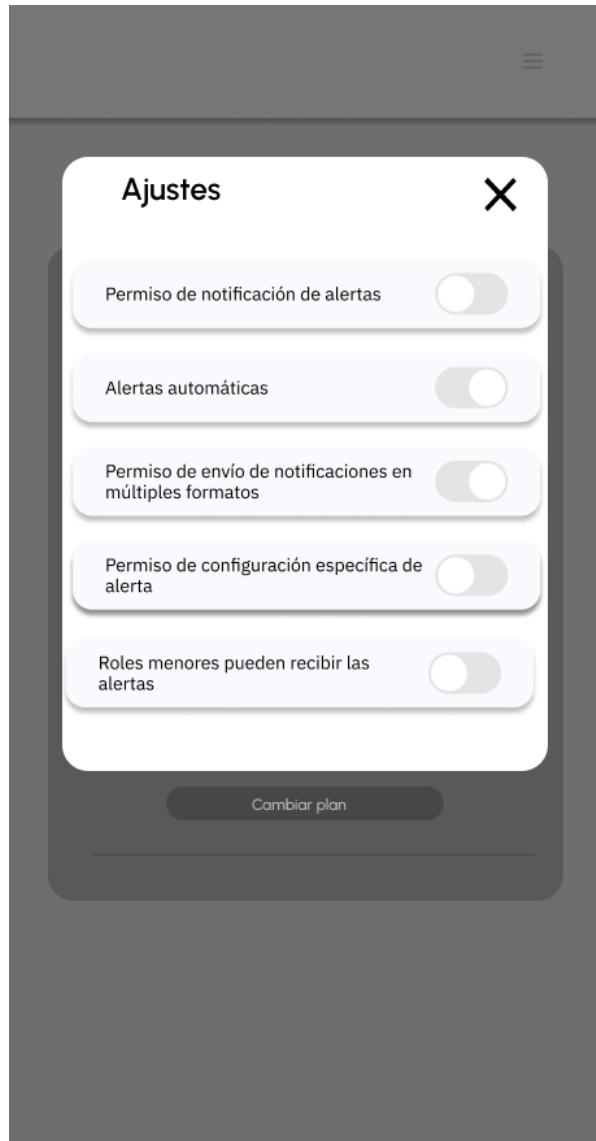
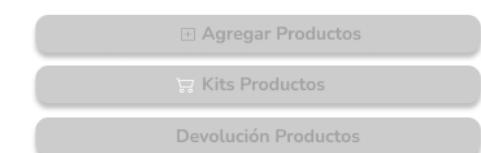


Imagen (Nº). Elaboración propia. Realizado en Figma

Panel de control (Dashboard)

Esta pantalla representa el panel de control principal del sistema StockWise. Su objetivo es proporcionar al usuario una vista general del estado del inventario y acceso rápido a funciones clave. Entre sus elementos principales se encuentran:

- Resumen de información:
 - Total de Productos: número total de ítems registrados.
 - Fecha del Último Proveedor: muestra la última fecha de ingreso de productos (por proveedor).
 - Próximos a Caducar: productos con fecha de vencimiento próxima, indicando la fecha y el stock restante.
- Accesos rápidos:
 - Historial de Movimientos: seguimiento de entradas y salidas del inventario.
 - Inventario: acceso directo a la lista completa de productos.
- Botones de acción:
 - Agregar Productos: para registrar nuevos productos en el sistema.
 - Kits de Productos: permite combinar productos individuales en un kit.
 - Devolución de Productos: facilita el registro de productos devueltos por proveedores o clientes.



Leche 15/05/2024	Stock 5
Leche 15/05/2024	Stock 5
Leche 15/05/2024	Stock 5

Imagen (Nº). Elaboración propia. Realizado en Figma

Combinación de Productos en Kits

Esta sección está dedicada a la funcionalidad de crear kits de productos, una herramienta útil para negocios que venden combos o paquetes predefinidos. Las características principales incluyen:

Buscador de productos: permite al usuario localizar rápidamente productos por nombre.

Listado de productos existentes:

Muestra el nombre, precio y stock disponible de cada producto.

Incluye un botón de acción en cada fila para añadir productos al kit.

Botón "Seleccionar para kit": una vez seleccionados los productos deseados, el usuario puede avanzar para crear el kit final.

Esta funcionalidad mejora la gestión del stock agrupado y permite optimizar estrategias de venta, como combos promocionales o paquetes de temporada.



lorem ipsum

 Lorem

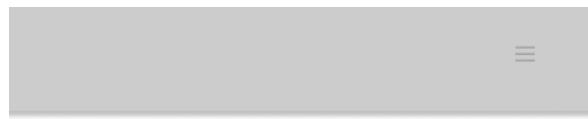
Lorem ipsum

Imagen (Nº). Elaboración propia. Realizado en Figma

Agregar Producto

Esta pantalla permite visualizar todos los productos actualmente registrados en el inventario, con acceso rápido para editar o ver más detalles. Las funcionalidades clave incluyen:

- Tarjetas de producto con información relevante:
 - Nombre del producto (ej. Galleta).
 - Categoría/etiqueta (ej. Golosina).
 - Stock disponible.
 - Botón “+ Detalle” para ver información adicional o editar.
 - Botón “+”: permite agregar un nuevo producto desde cero
 - Buscador de productos y opción de filtro por categoría o stock.



Agregar Producto



Buscar Productos

Filtro

Galleta	20 Stock
+	+ Detalle

Galleta	20 Stock
+	+ Detalle

Galleta	20 Stock
+	+ Detalle

Galleta	20 Stock
+	+ Detalle

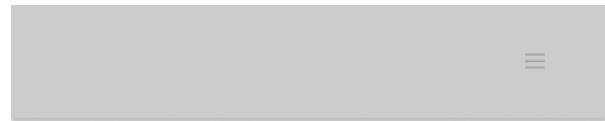
Imagen (Nº). Elaboración propia. Realizado en Figma

Registro de Nuevo Producto

Pantalla que aparece al presionar el botón "+", diseñada para agregar un nuevo producto al sistema. Incluye los siguientes campos:

- Nombre del producto.
- Etiquetas/Categorías (por ejemplo: Golosina, Bebida).
- Precio de compra y de venta.
- Cantidad inicial del producto.
- Número de lote y fecha de caducidad.
- Notas adicionales, si se desea registrar observaciones específicas.

Una vez completado el formulario, se debe presionar "Guardar" para registrar el nuevo ítem.



lorem ipsum

Lorem

Lorem

 Lorem

 Lorem

 Lorem

 Lorem

 Lorem

 Lorem

 Lorem

 Lorem

 Lorem

Guardar

Imagen (Nº). Elaboración propia. Realizado en Figma

Edición Rápida de Producto

Este modal aparece al hacer clic en “+ Detalle” sobre cualquier tarjeta de producto en la pantalla principal. Permite realizar modificaciones rápidas sobre un producto específico:

- Editar etiquetas.
- Actualizar cantidad.
- Modificar fecha de caducidad.
- Agregar notas.

Incluye botones para “Duplicar” (crear una copia rápida del producto) y “Eliminar” (quitar del sistema).

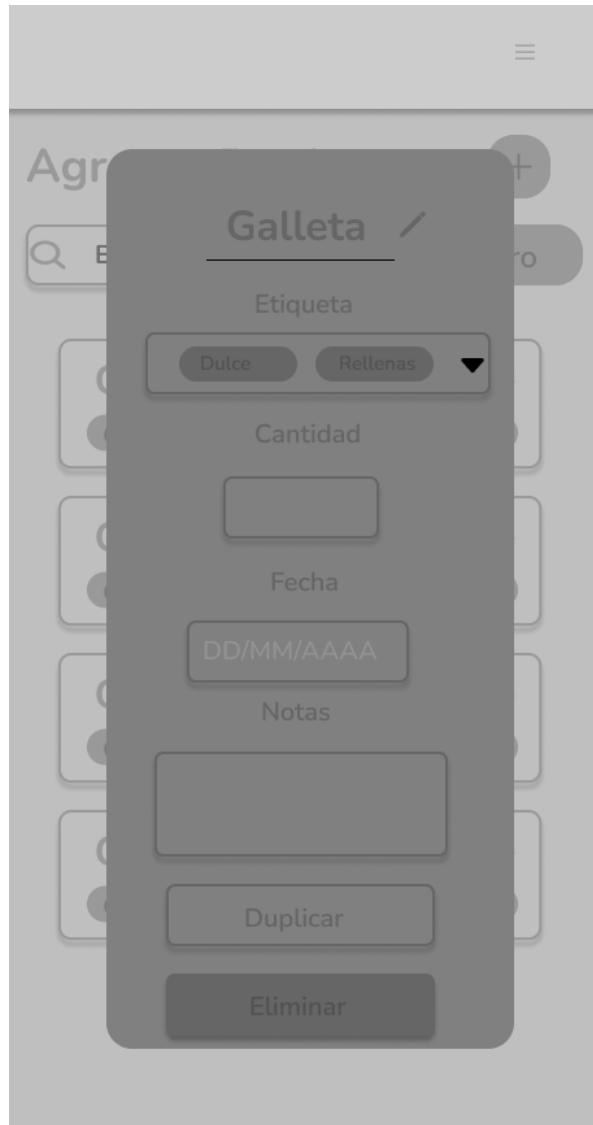


Imagen (Nº). Elaboración propia. Realizado en Figma

Historial de Movimientos

Esta vista permite al usuario consultar todos los movimientos registrados en el sistema, ya sea por producto individual, por categoría o mediante cálculo del ticket promedio.

- Filtros disponibles:
 - Tipo de gestión: Producto, Categoría o Stock Promedio.
 - Rango de fecha: permite buscar registros por fechas específicas.
- Resultados organizados en tarjetas que muestran:
 - Producto o categoría
 - Fecha del movimiento
 - Precio unitario (si aplica)
 - Cantidad gestionada
 - Total del movimiento
- También cuenta con botones para:
 - Editar un registro
 - Eliminar un registro (con confirmación previa).

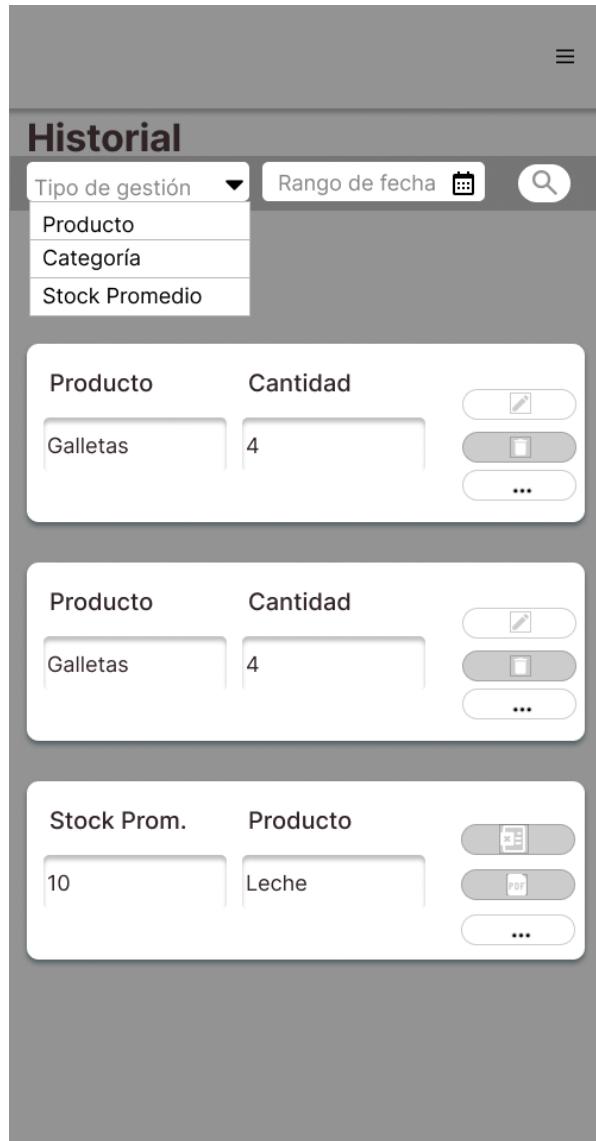


Imagen (Nº). Elaboración propia. Realizado en Figma

Confirmación de Eliminación

Al presionar el botón de eliminar, se despliega una ventana emergente de confirmación que advierte al usuario antes de borrar un registro del historial.



Imagen (Nº). Elaboración propia. Realizado en Figma

Ticket Promedio

Al seleccionar "Stock Promedio" como tipo de gestión, se habilita el acceso al cálculo de Ticket Promedio, que estima el valor promedio de venta por unidad en un rango de fechas. Este recurso permite realizar un análisis financiero rápido sobre el comportamiento de productos específicos.

- Campos del formulario:
 - Rango de fecha.
 - Producto a analizar.
 - Cantidad de ventas.
 - Precio total de ventas.
 - Resultado del ticket promedio (calculado automáticamente).
- Botones:
 - Crear: Guarda el registro del ticket.
 - Cancelar: Cierra el modal sin guardar.



Imagen (Nº). Elaboración propia. Realizado en Figma

Pantalla principal de Inventario por Lote

Esta pantalla permite la visualización y gestión del inventario agrupado por lote de entrada.

- Filtros en la parte superior:
 - Búsqueda por Producto
 - Proveedor
 - Fecha de ingreso
 - Cantidad
 - Precio
- Lista de productos ingresados por lote:
 - Proveedor
 - Producto
 - Fecha de entrada
 - Cantidad por unidad
 - Precio por unidad
 - Unidad de medida
- Botón rojo “Generar Nuevo Lote” ubicado a la derecha, que permite agregar un nuevo registro.

The image shows a mobile application interface for managing inventory. At the top, there is a header bar with the title "Inventario por Lote". Below the header, there are two tabs: "Por producto" and "Por lote", with "Por producto" being the active tab. Underneath the tabs are search and generate buttons: "Buscar" with a magnifying glass icon, "Generar nuevo lote", and "Búsqueda avanzada" with a gear icon.

The main content area displays two product entries in a card format:

Producto	Proveedor
Leche	Rosaura Lopez

Below this, there are four input fields:

Fecha de entrada	Cantidad por unidad
14/04	20

Precio por unidad	Unidad de medida
s/. 5	ml

At the bottom of each card are two buttons: a dark blue button with a plus sign and a white button with a plus sign.

The second card below it has identical data:

Producto	Proveedor
Leche	Rosaura Lopez

Below this second card are the same four input fields as the first one.

Imagen (Nº). Elaboración propia. Realizado en Figma

Busqueda Avanzada



Imagen (Nº). Elaboración propia. Realizado en Figma

Pantalla de Comentarios del Lote

Este modal aparece al hacer clic en el icono de "Comentarios", permite dejar notas asociadas a un producto, útiles para incidencias, condiciones especiales del lote o instrucciones de seguimiento.

- Campos de entrada para seleccionar el Lote y el Producto.
- Área para ingresar el comentario del usuario.
- Botones:
 - "Guardar" (acción de confirmar)
 - "Cancelar" (descartar el comentario)

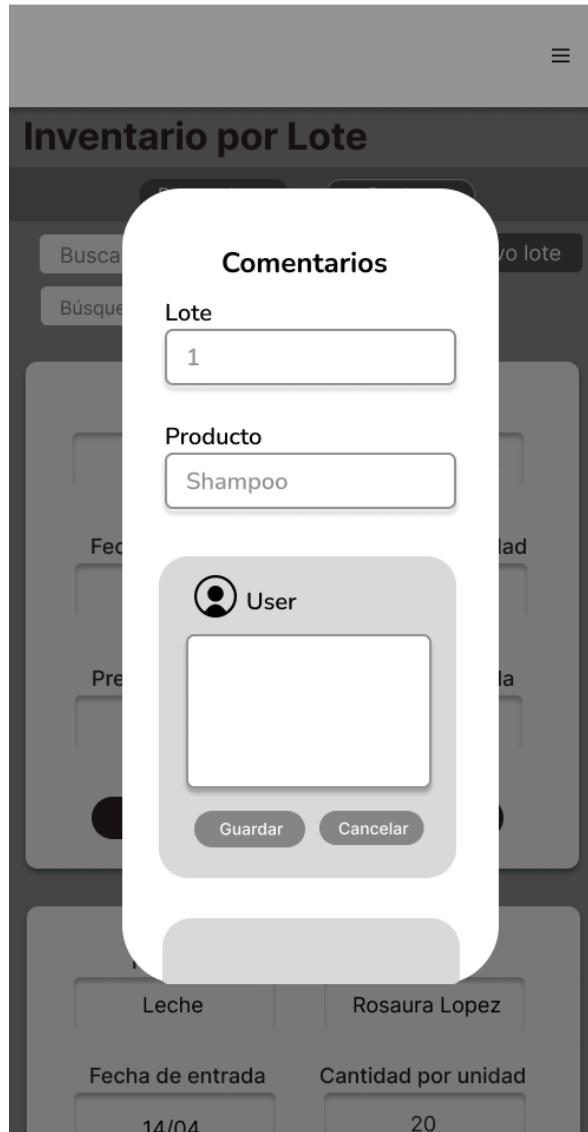


Imagen (Nº). Elaboración propia. Realizado en Figma

Pantalla de Exportación

Permite obtener un respaldo o trabajar fuera del sistema con los datos del inventario. Al dar click, saldra una ventana emergente con el ícono de exportación. Mostrara un texto “¿Deseas exportar en excel?” y botones de acción: “Volver” y “Continuar”.



Imagen (Nº). Elaboración propia. Realizado en Figma

Pantalla de Inventario por Producto

Visualización clara de todos los productos del inventario, con opción de filtrar por condiciones específicas, lo cual optimiza el control individualizado del stock.

- Filtros:
 - Categoría
 - Productos
 - Rango de Fecha
 - Stock Mínimo.
- Tabla:
 - Categoría
 - Producto
 - Fecha de entrada
 - Cantidad
 - Precio
 - Stock mínimo
 - Unidad de medida.
- Botón: "Generar Nuevo Lote".

Imagen (Nº). Elaboración propia. Realizado en Figma

Edición de Registro

Facilita la actualización de datos en tiempo real, permitiendo mantener el inventario siempre actualizado y preciso.

- Elementos mostrados:
 - Formulario editable con campos: Categoría, Producto, Cantidad por unidad, Precio por unidad, Unidad de medida.
 - Botones de acción: "Crear" (confirmar edición) y "Cancelar".

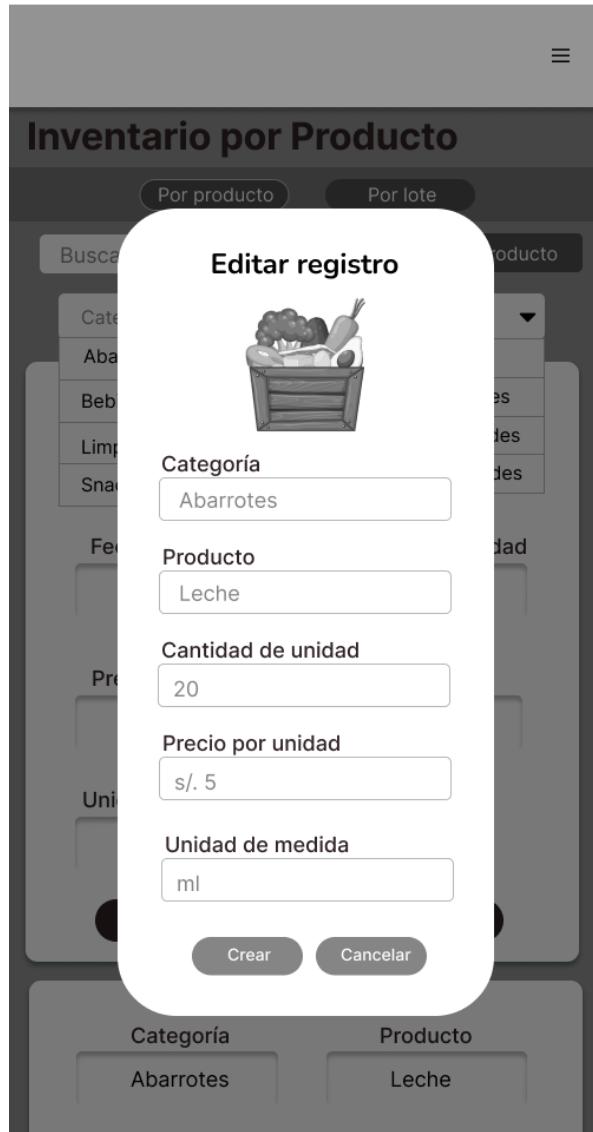


Imagen (Nº). Elaboración propia. Realizado en Figma

3.1.4.2. Mobile Applications Wireflow Diagrams

Los Mobile Applications Wireflow Diagrams son una combinación de wireframes y flujos de usuario (user flows) que ilustran no solo la estructura y disposición de las pantallas de una Mobile Application, sino también cómo los usuarios navegarán entre ellas. Estos diagramas proporcionan una visión detallada del recorrido del usuario, mostrando las interacciones clave y las transiciones de una pantalla a otra dentro de la aplicación.

User Goal: Registrar

El usuario selecciona la opción "Register", completa los campos solicitados y hace clic en el botón "Registrar". A continuación, se muestra el panel "Add Card", donde debe llenar los campos relacionados con su tarjeta y correo electrónico. Una vez que el proceso de pago se complete exitosamente, se notifica al usuario con un mensaje confirmando el vínculo de su tarjeta con la plataforma. Del mismo modo, si el usuario desea retirar su información o actualizarlo podrá hacer a través de su perfil. Finalmente hacer clic en el botón "Aceptar".

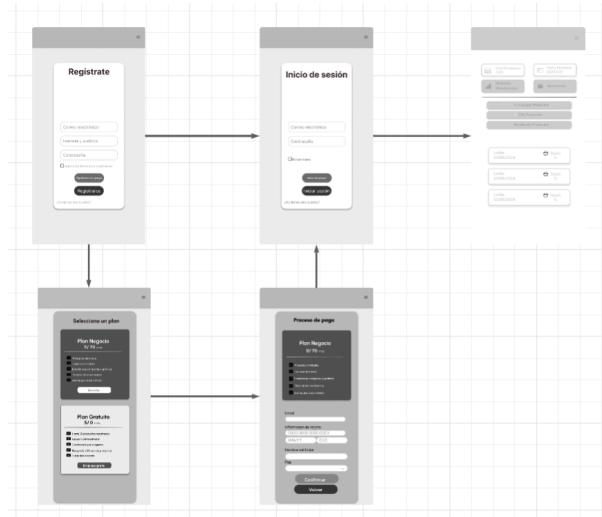


Imagen (Nº). Elaboración propia. Realizado en Figma

User Goal: Iniciar sesión

El usuario introduce su correo y contraseña, luego hace clic en el botón "Log In". Después, se le redirige al panel de perfil, donde podrá editar su información personal y acceder a las herramientas según su perfil "Administrador" o "Empleado".

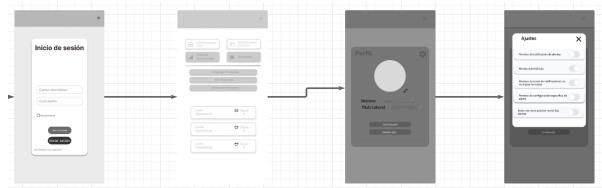


Imagen (Nº). Elaboración propia. Realizado en Figma

User Goal: Navegar por el Dashboard

1. El usuario inicia sesión desde la Landing Page.
2. Ingresa a la vista principal del Dashboard.
3. Visualiza el total de productos registrados y la fecha del último proveedor.
4. Visualiza un resumen de productos próximos a caducar con su respectiva fecha y stock disponible.
5. Accede a botones de acción rápida como "Historial", "Inventario", "Añadir Productos", "Kits" y "Devolución de productos".

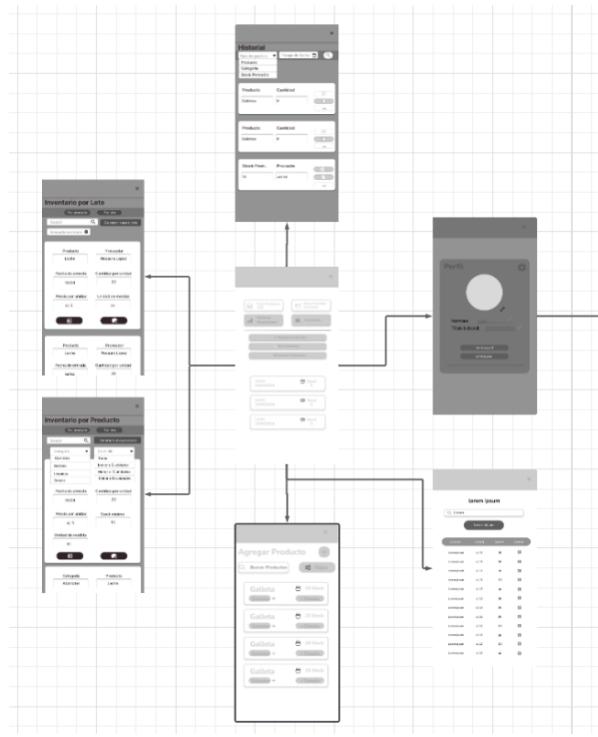


Imagen (Nº). Elaboración propia. Realizado en Figma

User Goal: Inventario (Producto / Lote)

1. Ingresa a la sección de Inventario.
2. Revisa el listado de productos presionando el botón "por producto".
3. Filtra los productos por categoría, nombre del producto, fecha o stock mínimo.
4. Consulta el listado con información clave: fecha de entrada, cantidad por unidad, precio, stock mínimo y unidad de medida.

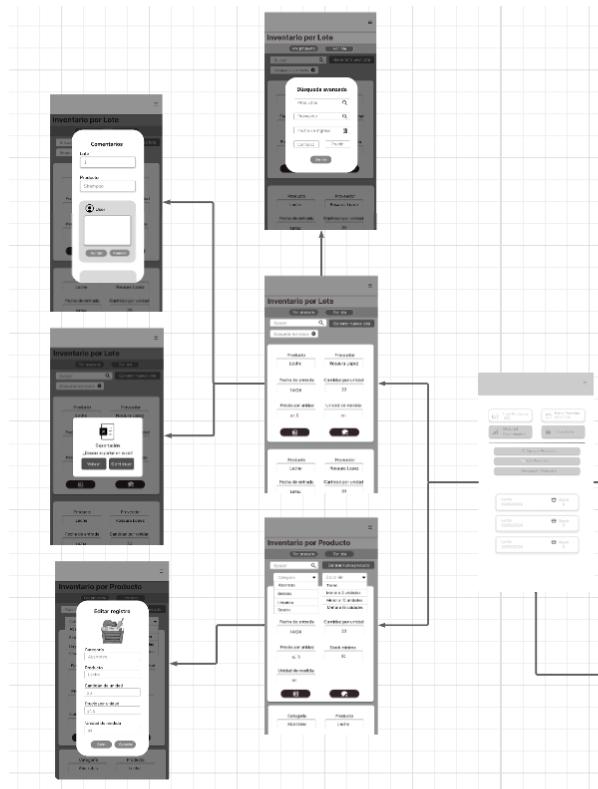


Imagen (Nº). Elaboración propia. Realizado en Figma

User Goal: Botones Principales (Aregar Producto y Kits)

1. Pulsa el botón "Aregar Producto".

2. Rellena los campos solicitados para registrar uno nuevo.
3. Pulsa el botón "Crear Kit".
4. Combina productos existentes para crear un kit nuevo.
5. El usuario pulsa el botón "Añadir Productos" desde el Dashboard.
6. Visualiza una galería de productos existentes y accede a opciones para editarlos o duplicarlos.
7. Puede agregar uno nuevo haciendo clic en el botón "+", donde se despliega un formulario con campos como nombre, etiquetas, cantidad, lote, precios, fecha de caducidad y notas.
8. Desde el menú principal, también accede a la opción "Kits".

En esta sección, selecciona productos existentes y los combina mediante el botón "Seleccionar para kit", indicando cantidad e inventario disponible.

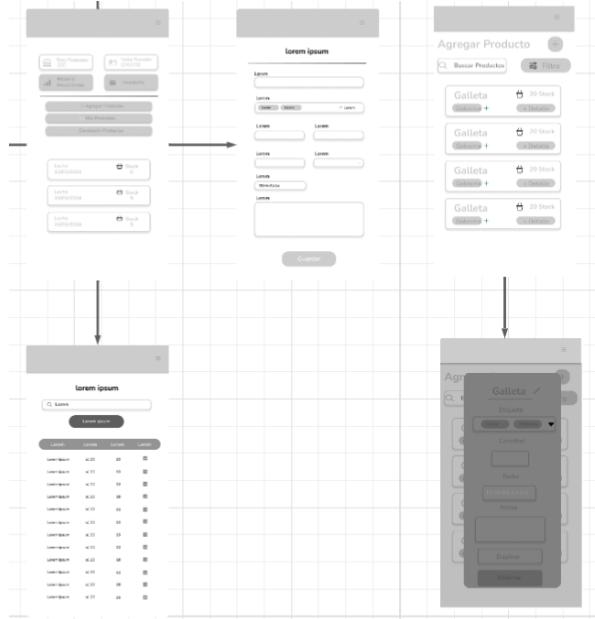


Imagen (Nº). Elaboración propia. Realizado en Figma

User Goal: Historial de Movimientos

1. Navega a la sección de Historial.
2. Visualiza entradas y salidas de productos.
3. Filtra movimientos por fecha, producto o lote.
4. El usuario accede a la sección de Historial desde el panel principal.
5. Filtra los registros por tipo de gestión, categoría, stock promedio y fecha.
6. Visualiza los movimientos realizados, incluyendo datos como nombre del producto, fecha de consulta, precio unitario, cantidad y total.
7. Consulta métricas como el stock promedio, estado del producto y stock ideal.
8. Cuenta con botones para editar o eliminar cada registro y, para los stock promedio, exportar la información y realiza un ticket promedio.

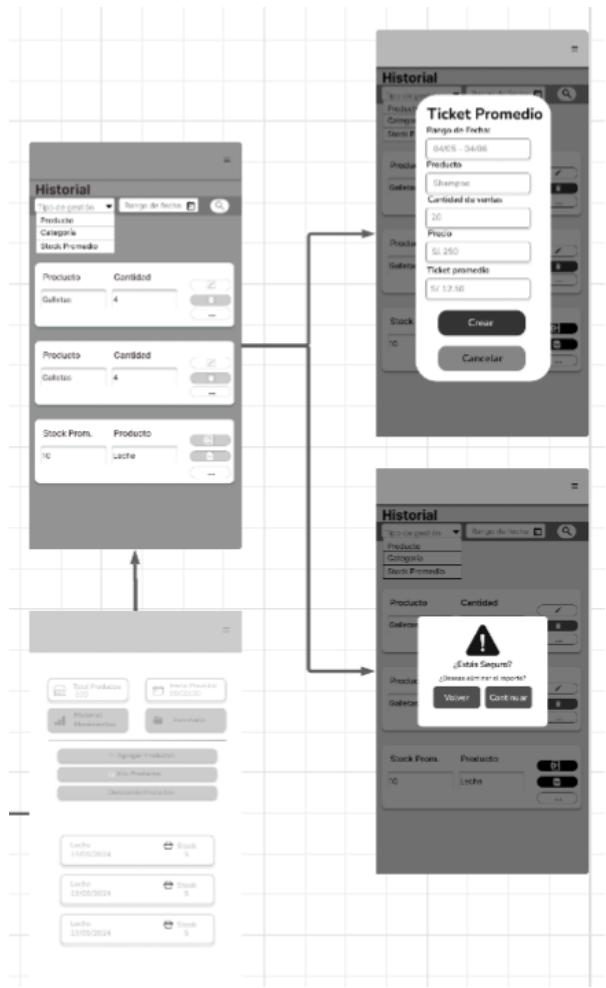


Imagen (Nº). Elaboración propia. Realizado en Figma

3.1.4.3. Mobile Applications Mock-ups

Iniciar sesión

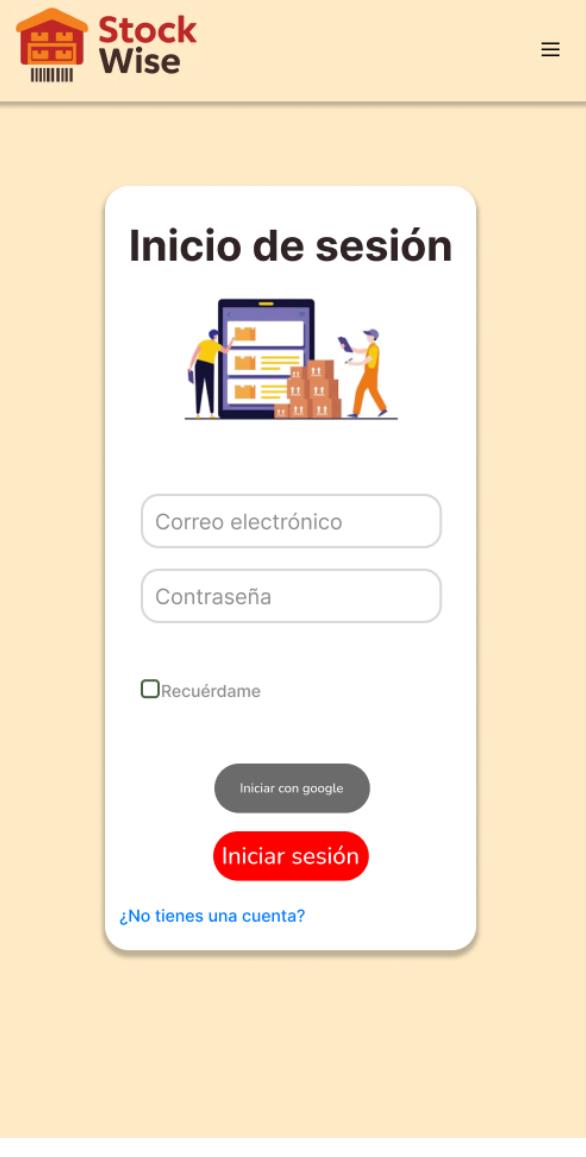


Imagen (Nº). Elaboración propia. Realizado en Figma

Crear cuenta

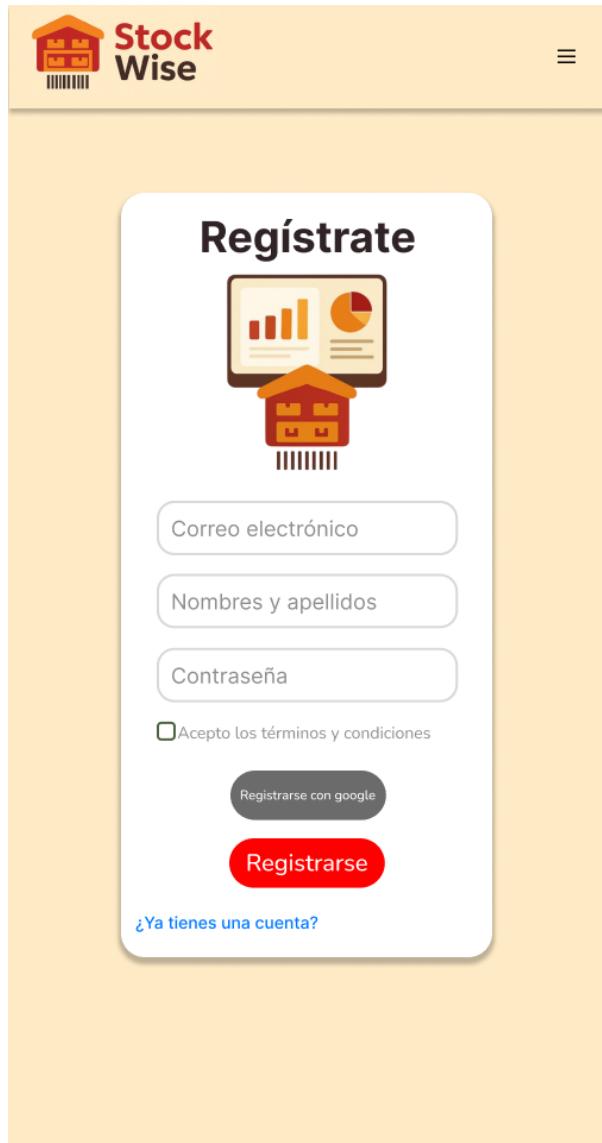


Imagen (Nº). Elaboración propia. Realizado en Figma

Elegir plan de pago

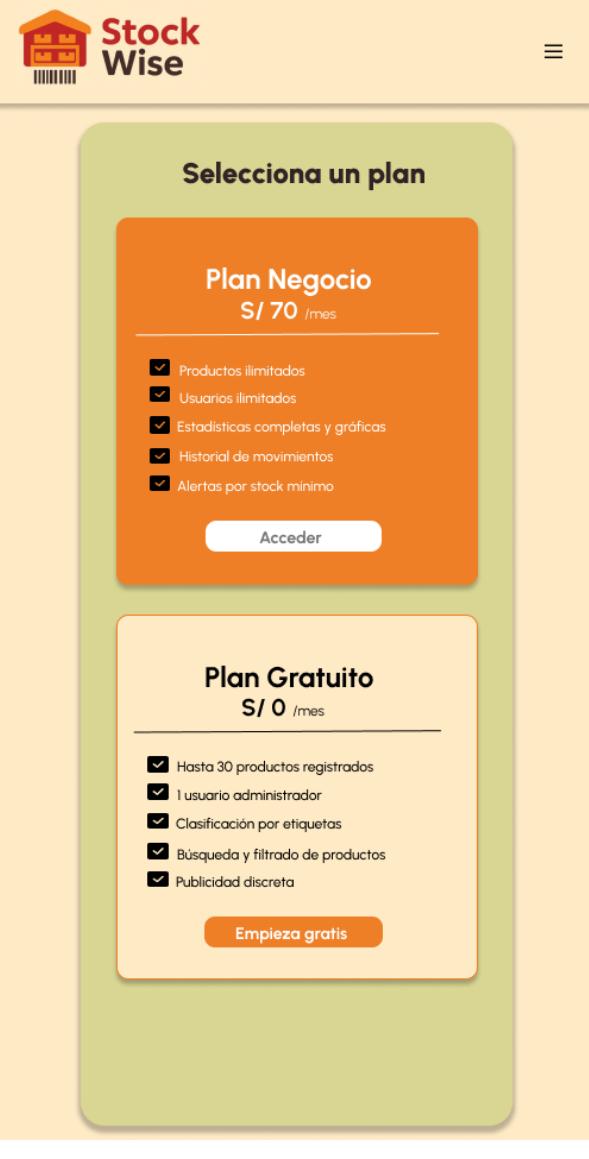


Imagen (Nº). Elaboración propia. Realizado en Figma

Pasarela de pagos



Imagen (Nº). Elaboración propia. Realizado en Figma

Mi perfil

Perfil Administrador

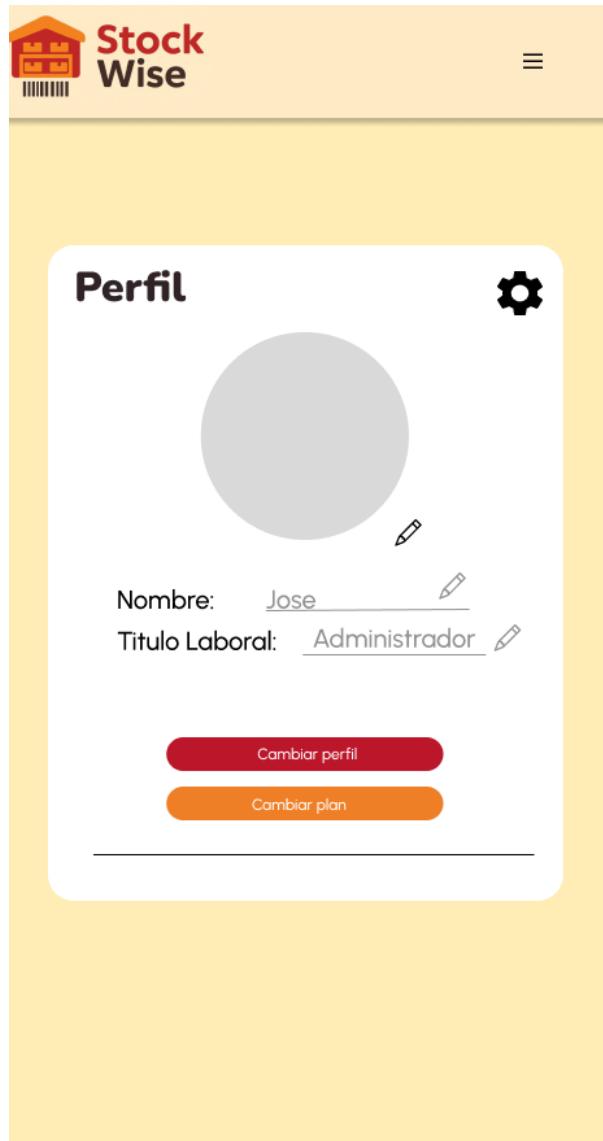


Imagen (Nº). Elaboración propia. Realizado en Figma

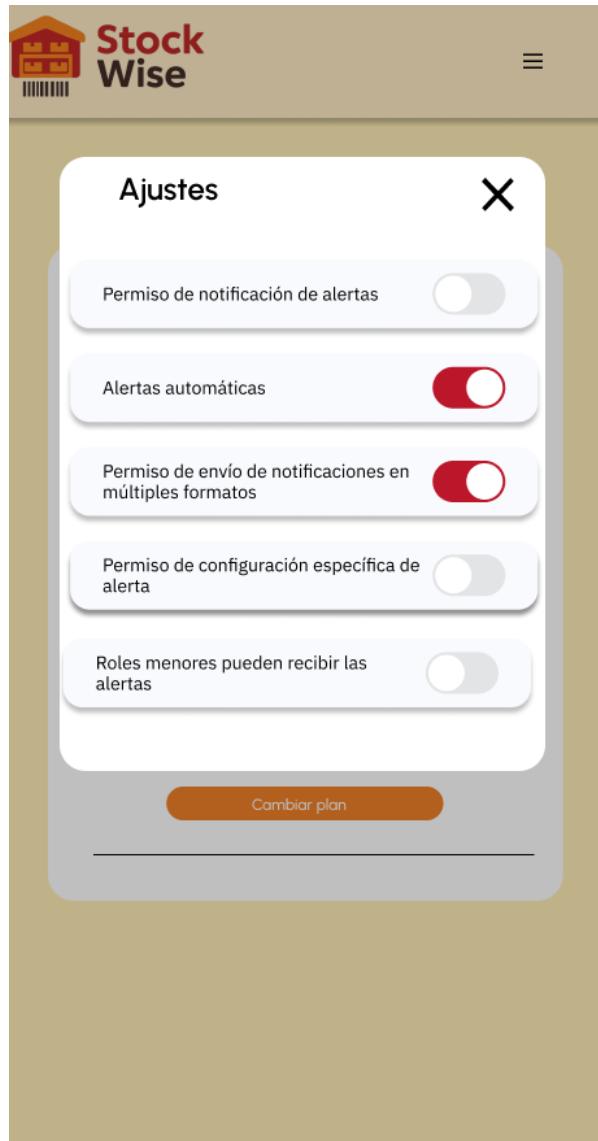


Imagen (Nº). Elaboración propia. Realizado en Figma

Panel de control (Dashboard)

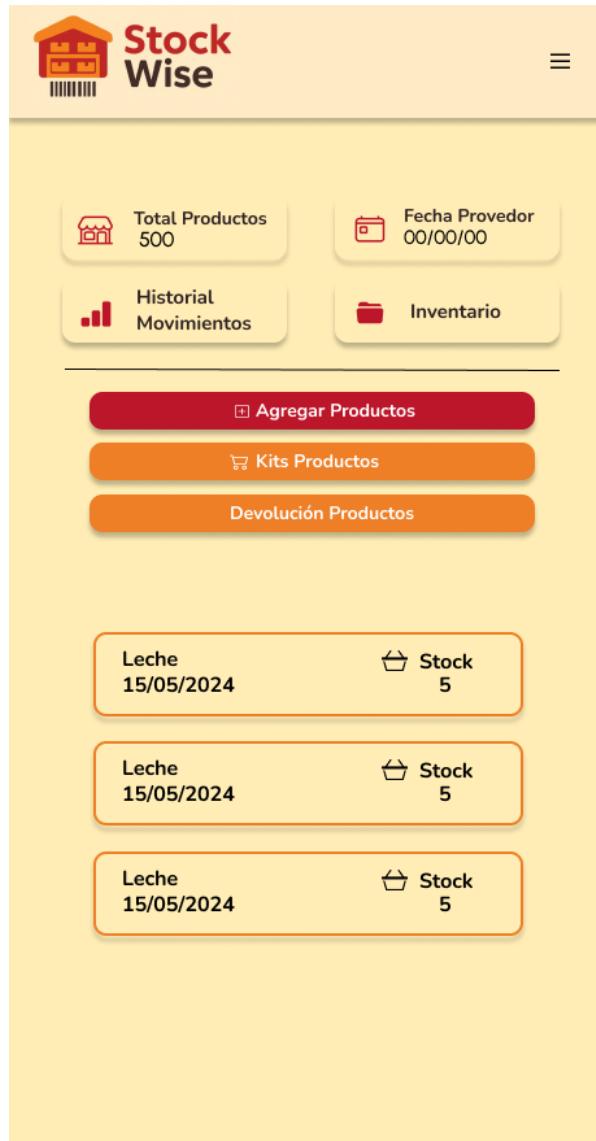


Imagen (Nº). Elaboración propia. Realizado en Figma

Combinación de Productos en Kits

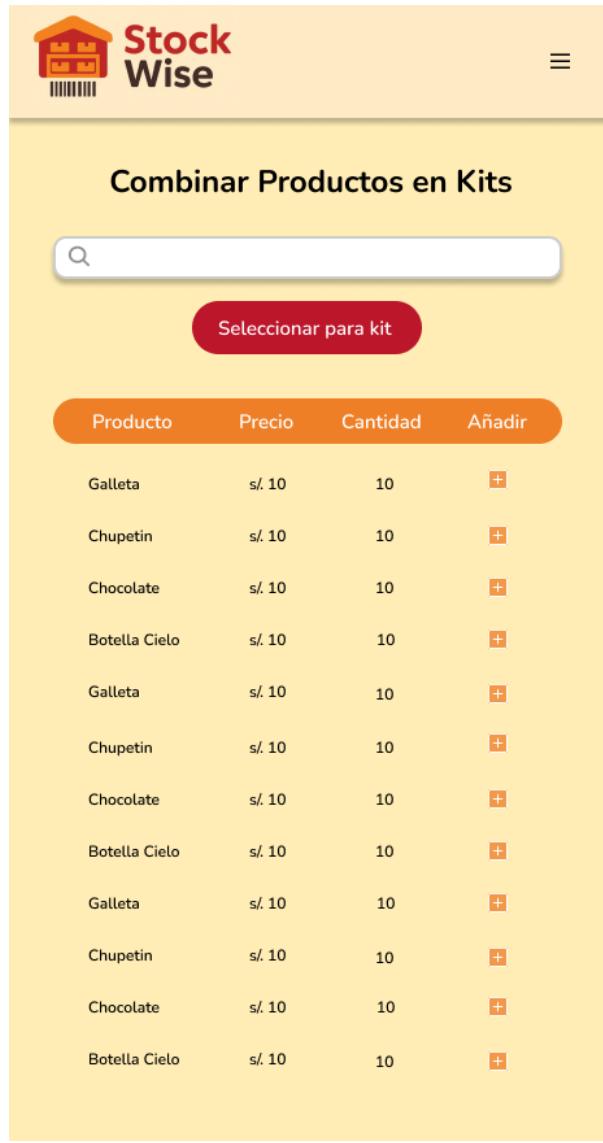


Imagen (Nº). Elaboración propia. Realizado en Figma

Agregar Producto



Imagen (Nº). Elaboración propia. Realizado en Figma

Registro de Nuevo Producto

The image shows a mobile application interface for 'Stock Wise'. At the top, there is a logo of a red building with the text 'Stock Wise' next to it. On the right side of the header is a three-line menu icon. The main screen has a light yellow background and features the title 'Agregar productos' (Add products) in orange at the top center. Below the title are several input fields and controls:

- A text input field labeled 'Nombre' (Name).
- A text input field labeled 'Etiquetas' (Labels).
- Two input fields for price: 'Precio de compra' (Purchase Price) showing '\$ 00.00' and 'Precio de venta' (Sale Price) showing '\$ 00.00'.
- Two dropdown menus: 'Cantidad' (Quantity) and 'Lote' (Lot).
- A date input field labeled 'Fecha de caducidad' (Expiration Date) with the placeholder 'DD/MM/AAAA'.
- A text input field labeled 'Nota' (Note).

At the bottom center of the screen is a large red button with the white text 'Guardar' (Save).

Imagen (Nº). Elaboración propia. Realizado en Figma

Edición Rápida de Producto



Imagen (Nº). Elaboración propia. Realizado en Figma

Historial de Movimientos

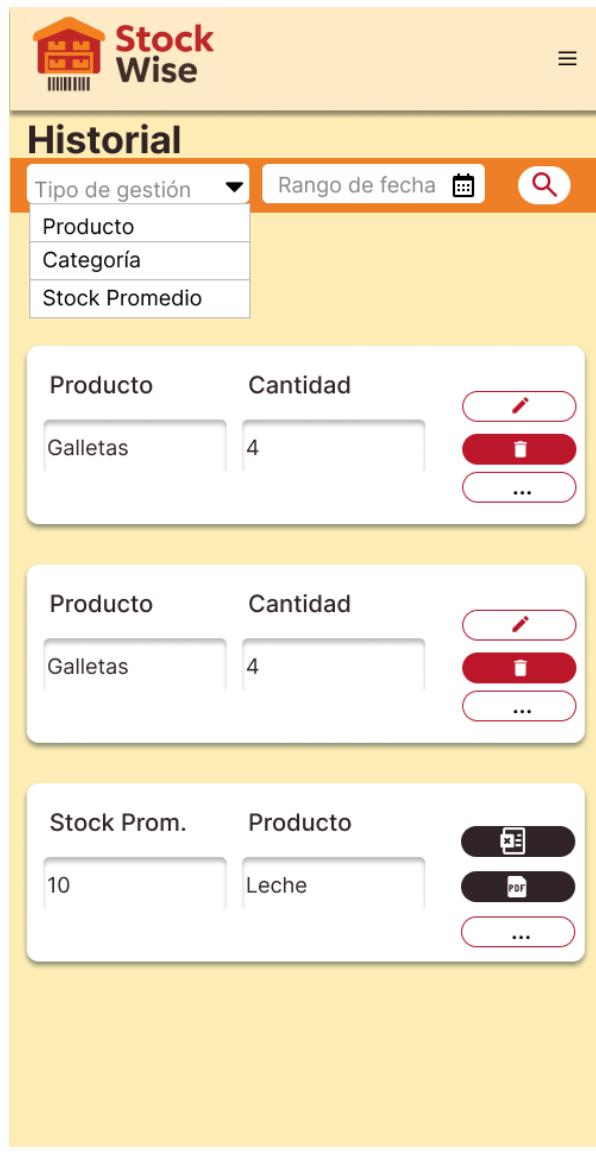


Imagen (Nº). Elaboración propia. Realizado en Figma

Confirmación de Eliminación



Imagen (Nº). Elaboración propia. Realizado en Figma

Ticket Promedio



Imagen (Nº). Elaboración propia. Realizado en Figma

Pantalla principal de Inventario por Lote



Imagen (Nº). Elaboración propia. Realizado en Figma

Busqueda Avanzada

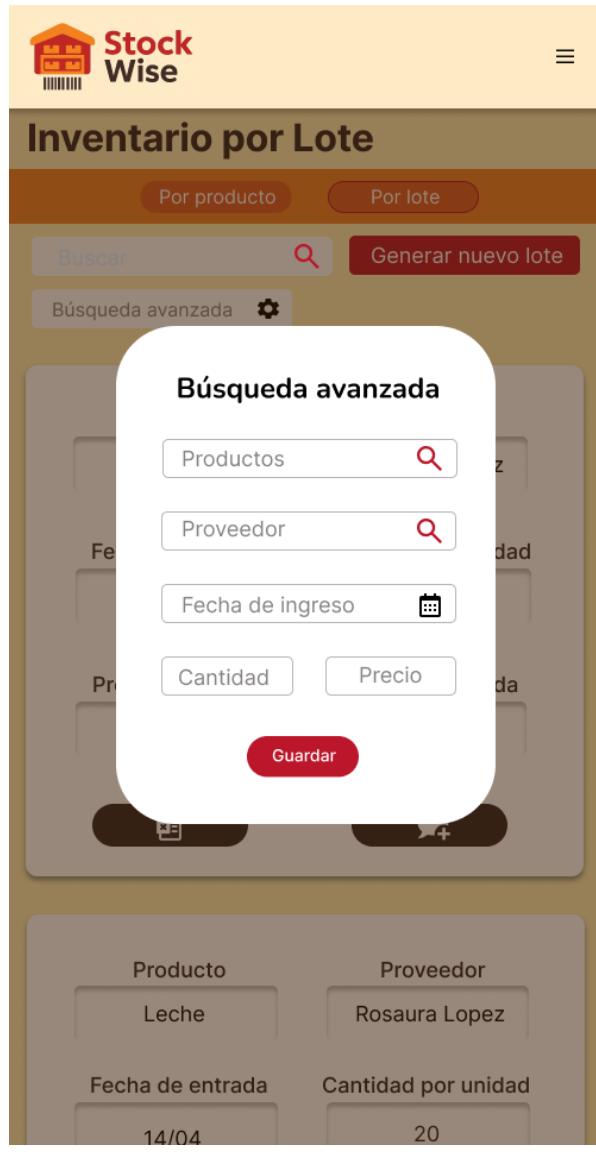


Imagen (Nº). Elaboración propia. Realizado en Figma

Pantalla de Comentarios del Lote

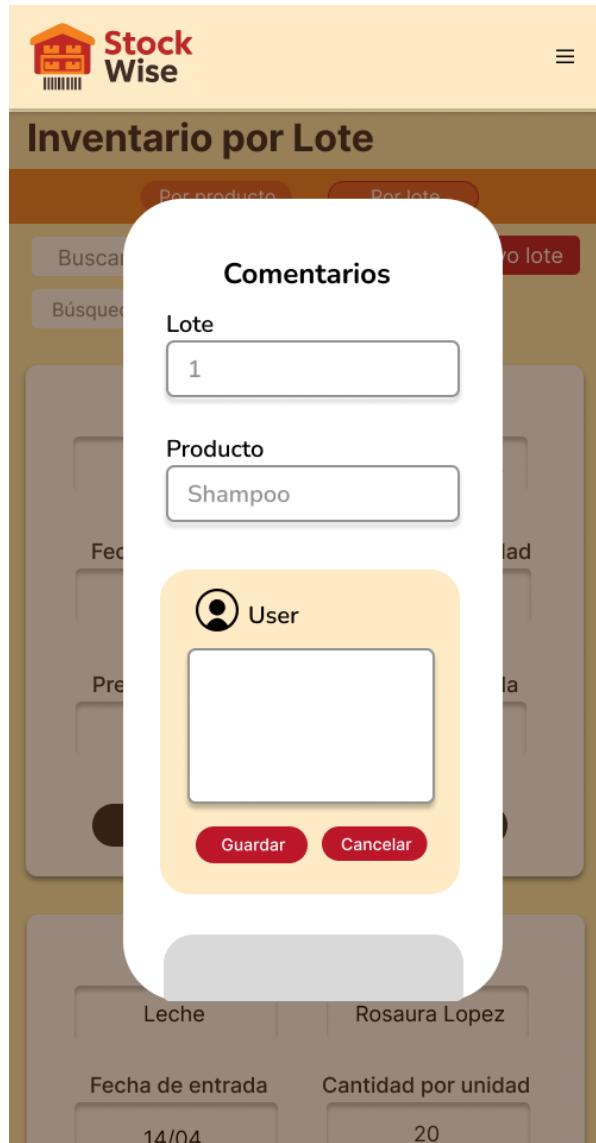


Imagen (Nº). Elaboración propia. Realizado en Figma

Pantalla de Exportación



Imagen (Nº). Elaboración propia. Realizado en Figma

Pantalla de Inventario por Producto

The image shows a user interface for a stock management system named "Stock Wise". The top navigation bar includes a logo of a red building with a yellow roof, the text "Stock Wise", and a menu icon (three horizontal lines). Below the header, the title "Inventario por Producto" is displayed. A secondary navigation bar offers two options: "Por producto" (selected) and "Por lote".

The main content area contains several input fields and dropdown menus:

- Search:** A search bar labeled "Buscar" with a magnifying glass icon.
- Product Generation:** A button labeled "Generar nuevo producto".
- Categoría:** A dropdown menu listing "Abarrotes", "Bebidas", "Limpieza", and "Snacks".
- Stock Min:** A dropdown menu listing "Todos", "Menor a 5 unidades", "Menor a 10 unidades", and "Menor a 15 unidades".
- Fecha de entrada:** An input field showing the date "14/04".
- Cantidad por unidad:** An input field showing the quantity "20".
- Precio por unidad:** An input field showing the price "s/. 5".
- Stock mínimo:** An input field showing the minimum stock level "10".
- Unidad de medida:** An input field showing the unit of measurement "ml".
- Action Buttons:** Two dark blue rounded rectangular buttons with white icons: one with a plus sign and another with a minus sign.

At the bottom of the form, there are two additional input fields:

- Categoría:** An input field showing "Abarrotes".
- Producto:** An input field showing "Leche".

Imagen (Nº). Elaboración propia. Realizado en Figma

Edición de Registro

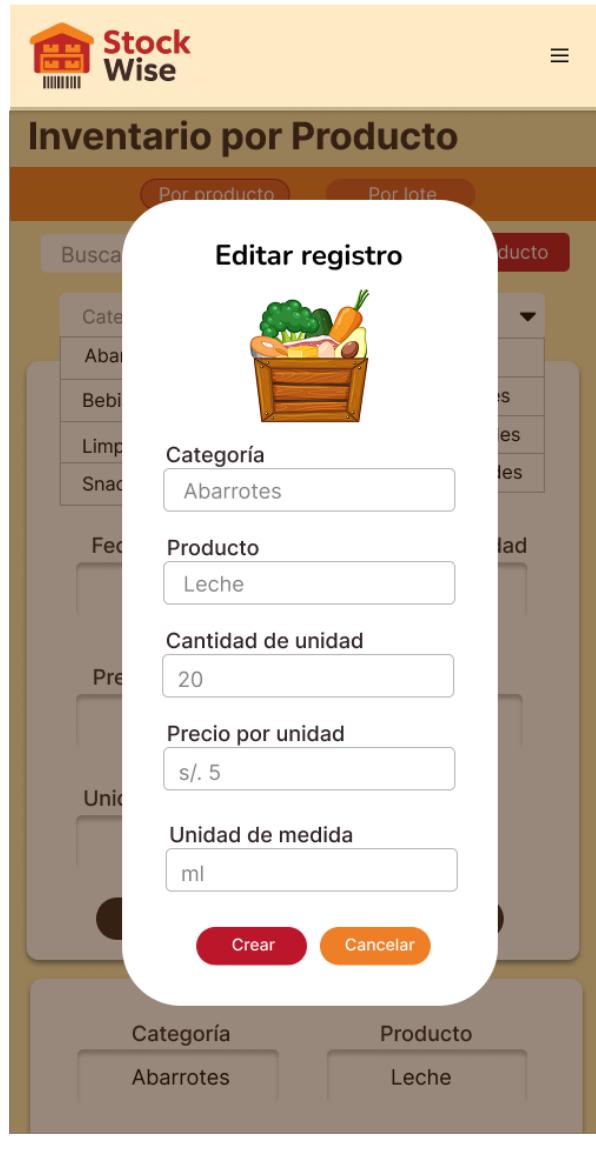


Imagen (Nº). Elaboración propia. Realizado en Figma

3.1.4.4. Mobile Applications User Flow Diagrams

User Goal: Registrar

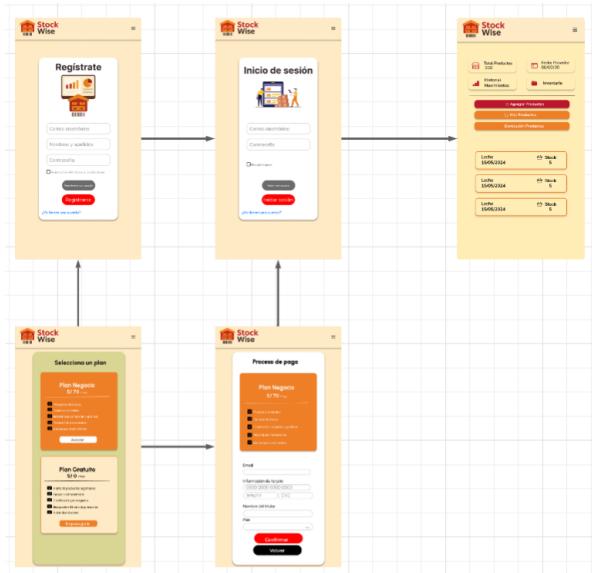


Imagen (Nº). Elaboración propia. Realizado en Figma

User Goal: Iniciar sesión

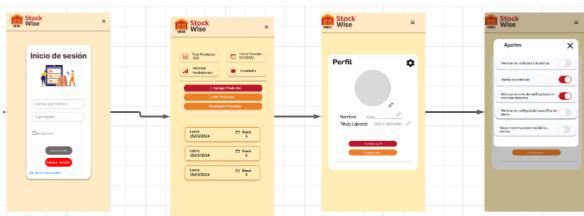


Imagen (Nº). Elaboración propia. Realizado en Figma

User Goal: Navegar por el Dashboard



Imagen (Nº). Elaboración propia. Realizado en Figma

User Goal: Inventario (Producto / Lote)

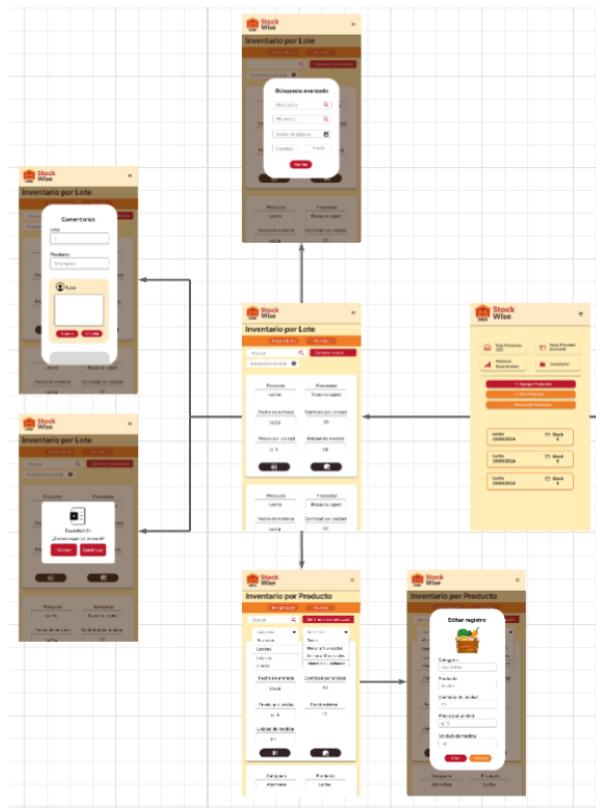


Imagen (Nº). Elaboración propia. Realizado en Figma

User Goal: Botones Principales (Agregar Producto y Kits)

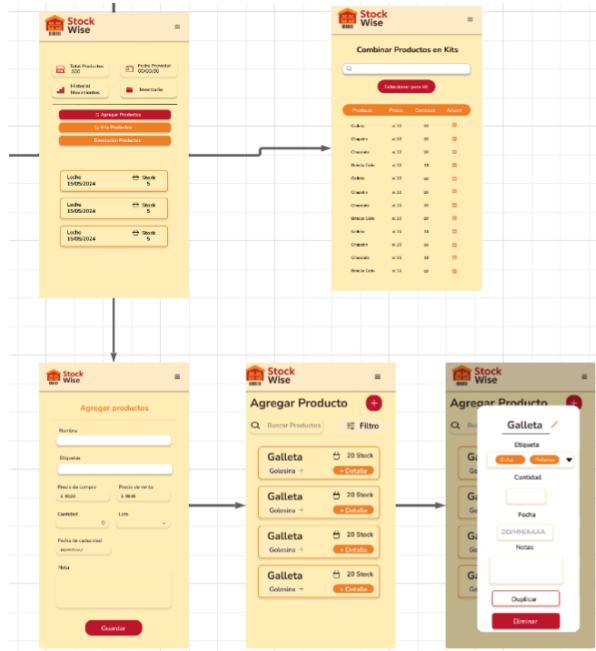


Imagen (Nº). Elaboración propia. Realizado en Figma

User Goal: Historial de Movimientos

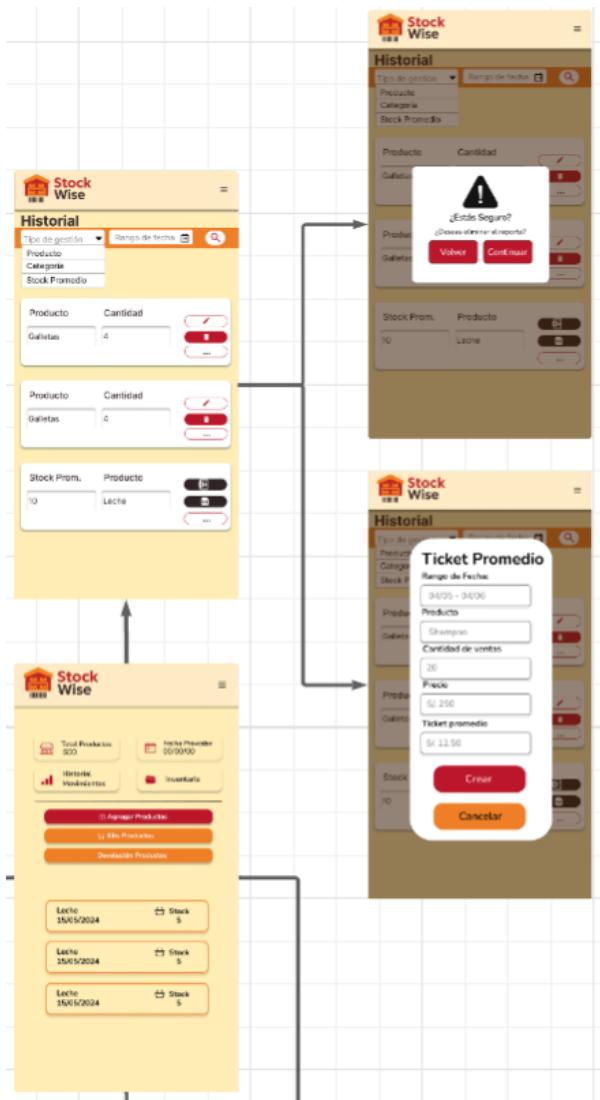


Imagen (Nº). Elaboración propia. Realizado en Figma

3.1.4.5. Mobile Applications Prototyping

El apartado de Mobile Application Prototyping muestra el primer acercamiento visual e interactivo al diseño de la aplicación. A través del uso de herramientas de prototipado, se crean representaciones dinámicas que permiten simular la navegación, disposición de módulos y flujos de interacción del usuario antes de pasar a la fase de desarrollo.

Durante este proceso, se estructuran pantallas clave como el login, panel de control, gestión de productos, movimientos de stock, generación de reportes y alertas de inventario. El objetivo es anticipar posibles mejoras de usabilidad y validar la propuesta funcional con los usuarios finales.

El prototipo también permite identificar ajustes necesarios en la experiencia de usuario (UX) y verificar que los requisitos funcionales definidos en las User Stories se reflejen correctamente en las interfaces propuestas.

A continuación, se presenta el enlace para visualizar el video de navegación del prototipo interactivo:

Inicio de sesión



Correo electrónico

Contraseña

Recuérdame

Iniciar con google

Iniciar sesión

[¿No tienes una cuenta?](#)

Capítulo IV: Product Implementation & Validation

En este apartado se resume todo el contenido recopilado, examinando los procedimientos a seguir y evaluando el estado emocional.

4. Product Implementation & Validation

4.1. Software Configuration Management

En la siguiente sección se detalla la ruta de acceso de cada uno de los productos de software, facilitando a cualquier miembro del equipo el desarrollo de cada aspecto del trabajo:

4.1.1. Software Development Environment Configuration

- **Android Studio:** Entorno de desarrollo.



- **GitHub:** Repositorio colaborativo en la nube.



- **Netlify:** Plataforma que facilita implementar despliegues sencillos para nuestras páginas web.



- **Vertabelo:** Plataforma colaborativa para la creación de diagramas de base de datos.



- **Visual Studio Code:** Entorno de desarrollo para diseño de base de datos.



- **Figma:** Herramienta colaborativa que permite elaborar wireframes y mockups.



4.1.2. Source Code Management

Repositorio de la Landing Page:

Implementación de GitFlow:

Para nuestra estrategia de gestión de versiones con Git, nos hemos inspirado en el artículo "A successful Git branching model" de Vincent Driessen, adoptando el modelo de ramificación GitFlow. Este enfoque nos permite establecer claramente las convenciones de ramificación que aplicamos en nuestro proyecto.



- **Rama Principal (Main branch):** Contiene el código en producción y se conoce como la Master branch o Main branch.
 - Notación: main
- **Rama de Desarrollo (Develop branch):** Acumula las últimas actualizaciones y cambios para la próxima versión. Funciona como un entorno de integración y prueba continua.
 - Notación: develop
- **Rama de Lanzamiento (Release branch):** Facilita la preparación de una nueva versión del producto, permitiendo correcciones de errores y recibiendo más actualizaciones de Develop.
 - Debe derivarse de: develop
 - Debe fusionarse con: develop y master/main
 - Notación: release
- **Rama de Características (Feature branch):** Se utiliza para desarrollar nuevas funcionalidades para la siguiente versión o futuras iteraciones.
 - Debe derivarse de: develop
 - Debe fusionarse de vuelta a: develop
 - Notación: feature
- **Rama de Corrección Rápida (Hotfix branch):** Aborda errores críticos en producción, permitiendo la implementación rápida de soluciones.
 - Debe derivarse de: master/main
 - Debe fusionarse con: develop y master/main
 - Notación: hotfix

Conventional Commits:

Adoptamos esta metodología para estructurar los mensajes de confirmación de cambios de manera estándar y semántica, lo que facilita la comunicación y la automatización de registros de cambios.

Tipos de Commits Convencionales:

- feat: Nuevas características o funcionalidades.
- fix: Correcciones de errores.
- docs: Cambios o mejoras en la documentación.
- style: Cambios de formato que no afectan la funcionalidad.
- refactor: Mejoras en la estructura o legibilidad del código.
- test: Adición o modificación de pruebas.
- chore: Cambios en el proceso de construcción o tareas de mantenimiento.
- perf: Mejoras de rendimiento en el código.

4.1.3. Source Code Style Guide & Conventions

HTML

Regla	Ejemplo / Explicación
Etiquetas y atributos en minúsculas	<code><div class="container">, </code>
Atributos ordenados lógicamente	<code>class, id, name, type, value, etc.</code>
Uso de comillas dobles	<code><input type="text" name="username"></code>
Indentación consistente (2 o 4 espacios)	No mezclar espacios con tabs

CSS

Regla	Ejemplo / Explicación
Nombres de clases en kebab-case	.main-header, .user-profile-card
Propiedades en minúsculas y ordenadas	color: #333; font-size: 16px; margin-top: 20px;
Uso de comentarios	/* Sección de estilos para el header */
Indentación consistente	2 o 4 espacios, no usar tabs

JavaScript

Regla	Ejemplo / Explicación
Variables y funciones en camelCase	let userName = "Juan"; function getUserData() {}
Clases en PascalCase	class UserProfile {}
Constantes en UPPER_SNAKE_CASE	const API_URL = "https://api.example.com";
Uso de const y let	Evitar var, usar const por defecto y let si se necesita mutabilidad
Punto y coma al final de líneas	let nombre = "Carlos";
Indentación consistente (2 o 4 espacios)	Mantener el mismo estilo en todo el proyecto

Kotlin

Regla	Ejemplo / Explicación
Variables y funciones en camelCase	val userName = "Juan", fun getUserData() {}
Clases y objetos en PascalCase	class UserProfile, object AppConfig
Constantes en UPPER_SNAKE_CASE	const val MAX_USERS = 100
Archivos nombrados igual que la clase	UserProfile.kt
Indentación con 4 espacios	No usar tabs
Uso de val por defecto, var si mutable	Promueve inmutabilidad
Expresiones lambda con it	users.filter { it.isActive }

4.1.4. Software Deployment Configuration

Deployment Landing Page:

En esta sección, detallamos el proceso de implementación de nuestra landing page en la plataforma de GitHub.

1. Se crea un repositorio en GitHub para alojar el código de nuestra landing page.

Landing-Page Public

main 1 Branch 0 Tags

Go to file Add file Code

C4m174 feat: actualizar título de la página de inicio a 'StockWise - Gestión...' cd7585e · last week 5 Commits

public feat: initialize React application with Vite, add global styles a... last week

src feat: agregar soporte multilingüe en la sección de características last week

.gitignore feat: agregar soporte multilingüe en la sección de características last week

README.md feat: initialize React application with Vite, add global styles a... last week

index.html feat: actualizar título de la página de inicio a 'StockWise - Ge... last week

package-lock.json feat: initialize React application with Vite, add global styles a... last week

package.json feat: agregar script de vista previa y cambiar directorio de sa... last week

vite.config.ts feat: agregar script de vista previa y cambiar directorio de sa... last week

README

Landing Page

Tecnologías Utilizadas

- React — Librería principal para la construcción de interfaces.
- TypeScript — Tipado estático para mayor robustez.
- Vite — Herramienta de desarrollo rápida para proyectos React.
- Tailwind CSS — Utilidades para estilos rápidos y responsive.

2. Agregamos a los participantes:

Insights / People

Find a user Everyone Outside collaborators

5 people have access to this repository

User	Access Level
Alejandro Daniel Oroncoy Almeyda	Read
Jeremy	Read
C4m174	Admin
Damaly29	Admin
Krillson	Read

1. Habilitamos Netlify para poder importar nuestro proyecto:

Watch NTL Deploy News Support

1. Connect to Git provider 2. Select repository 3. Configure project and deploy

Let's deploy your project with...

upc-pre-202520-1ACC0238-12612-Stoq

Search your repos

Landing-Page TypeScript 6 days ago Public

Native-mobile-development Kotlin 5 hours ago Public

Report 20 days ago Public

4. Finalmente, se confirma el despliegue de nuestra página web después de completar todo el procedimiento.

Controla tu inventario inteligentemente

Automatiza análisis, optimiza tus procesos y toma decisiones inteligentes con StockWise. La plataforma de gestión de inventario que tu negocio necesita.

Comenzar Gratis

Activar Windows
Ve a Configuración para activar Windows.

Este proceso garantiza el despliegue satisfactorio de nuestra landing page en la plataforma de Netlify, siguiendo las especificaciones y requisitos de nuestro proyecto.

Enlace de la Landing Page: <https://stockwiselanding.netlify.app/>

Deployment Backend:

En esta sección, detallamos el proceso de implementación de nuestro backend en la plataforma de Google Cloud.

Enlace del Backend: <http://34.39.181.148:8080/swagger/index.html>

Creación de VPS en Google Cloud Platform (GCP)

A continuación se detallan los pasos para crear un servidor privado virtual (VPS) en Google Cloud Platform para el despliegue del backend:

1. Configuración Inicial del Proyecto

Paso 1: Crear un Proyecto en GCP

1. Iniciar sesión en [Google Cloud Console](#)
2. Hacer clic en el selector de proyectos en la parte superior
3. Seleccionar "NUEVO PROYECTO"
4. Asignar un nombre descriptivo al proyecto (ej: "stockwise-backend")
5. Hacer clic en "CREAR"

Paso 2: Configurar Facturación

1. Navegar a "Facturación" en el menú de navegación
2. Vincular una cuenta de facturación válida
3. Seleccionar el proyecto creado como proyecto de facturación

2. Creación de la Instancia de VM

Paso 3: Habilitar Compute Engine API

1. En la barra de búsqueda, escribir "Compute Engine API"
2. Seleccionar la API y hacer clic en "HABILITAR"

Paso 4: Crear Instancia de VM

1. Navegar a "Compute Engine" > "Instancias de VM"
2. Hacer clic en "CREAR INSTANCIA"
3. Configurar los siguientes parámetros:

Detalles de la Instancia:

- **Nombre:** stockwise-backend-server
- **Región:** us-central1 (o la región más cercana a los usuarios)
- **Zona:** us-central1-a

Configuración de Máquina:

- **Tipo de máquina:** e2-medium (2 vCPU, 4 GB RAM)
- **Serie:** E2
- **Tipo de máquina:** Uso general

Disco de arranque:

- **Sistema operativo:** Ubuntu
- **Versión:** Ubuntu 20.04 LTS
- **Tipo:** Disco persistente equilibrado
- **Tamaño:** 30 GB

Firewall (Reglas de red):

- Permitir tráfico HTTP
- Permitir tráfico HTTPS
- Permitir tráfico SSH

4. Hacer clic en "CREAR" para iniciar la creación de la instancia

3. Configuración de Acceso y Seguridad

Paso 5: Configurar Acceso SSH

1. Una vez creada la instancia, hacer clic sobre el nombre de la instancia
2. En la sección "Conectarse", hacer clic en "Modificar" en "Control de acceso"
3. Agregar la clave SSH pública del equipo de desarrollo
4. Alternativamente, usar "Abrir en ventana del navegador" para acceso directo

Paso 6: Configurar Reglas de Firewall Personalizadas

1. Navegar a "Redes VPC" > "Firewall"
2. Hacer clic en "CREAR REGLA DE FIREWALL"
3. Configurar para el backend:
 - **Nombre:** backend-api-allow
 - **Dirección IP de origen:** 0.0.0.0/0 (para desarrollo)
 - **Protocolos y puertos:** TCP, puerto 8080 (o puerto del API)
4. Hacer clic en "CREAR"

Detalles Observabilidad Información del SO Captura de pantalla

descripción general de los riesgos de seguridad

SSH Conectar a consola en serie

La conexión a puertos en serie está inhabilitada [?](#)

Registros

[Registros](#)
[Puerto en serie 1 \(consola\)](#)

▼ Mostrar más

Información básica

Nombre	aplicaciones-moviles
ID de instancia	6187908988709181653
Descripción	Ninguna
Tipo	Instancia
Estado	<input checked="" type="checkbox"/> Activa
Fecha y hora de creación	sept 27, 2025, 10:06:38 p. m. UTC-05:00
Ubicación ?	southamerica-east1-a
Imagen de origen del disco de arranque	ubuntu-minimal-2404-noble-amd64-v20250923a
Arquitectura del disco de arranque	X86_64
Tipo de licencia del disco de arranque	Sin cargo
Plantilla de instancia	Ninguna
En uso por	Ninguno
Host físico ?	Ninguna
Estado del mantenimiento ?	—
Reservas	Elegir automáticamente
Etiquetas	goog-ops-a... : v2-x86-tem...
Etiquetas ?	—

Lot 1.0 OAS 3.0

/swagger/v1/swagger.json

Authorize 

Report

Puntos de acceso disponibles para la gestión de reportes de categoría y de stock promedio.

GET	/api/v1/reports	Obtener todos los reportes generales	 
POST	/api/v1/reports/category	Crear CategoryReport	 
GET	/api/v1/reports/category	Listar CategoryReports	 
GET	/api/v1/reports/category/{id}	Obtener CategoryReport por ID	 
GET	/api/v1/reports/category/by-date	Filtrar CategoryReports por fecha	 
POST	/api/v1/reports/stock-average	Crear StockAverageReport	 
GET	/api/v1/reports/stock-average	Listar StockAverageReports	 
GET	/api/v1/reports/stock-average/{id}	Obtener StockAverageReport por ID	 
GET	/api/v1/reports/stock-average/by-date	Filtrar StockAverageReports por fecha	 

Combos

Puntos de acceso disponibles para la gestión de combos (kits).

GET	/api/v1/combos/{comboId}	Obtener Combo por Id	🔒	▼
POST	/api/v1/combos	Crear Combo	🔒	▼
GET	/api/v1/combos	Obtener Todos los Combos	🔒	▼

Products

Puntos de acceso disponibles para la gestión de productos.

GET	/api/v1/products/{productId}	Obtener Producto por Id	🔒	▼
POST	/api/v1/products	Crear Producto	🔒	▼
GET	/api/v1/products	Obtener Todos los Productos	🔒	▼
GET	/api/v1/products/by-category/{categoryId}	Obtener Productos por Categoría	🔒	▼
GET	/api/v1/products/by-tag/{tagId}	Obtener Productos por Etiqueta	🔒	▼

Tags

Puntos de acceso para la gestión de etiquetas.

GET	/api/v1/tags	Obtener todas las etiquetas	🔒	▼
-----	--------------	-----------------------------	---	---

Units

Puntos de acceso para la gestión de unidades de medida.

GET	/api/v1/units	Obtener todas las unidades de medida	🔒	▼
-----	---------------	--------------------------------------	---	---

Inventory

Operaciones del inventario general, por producto y por lote.

GET	/api/v1/inventory	Obtener todos los registros de inventario	🔒	▼
POST	/api/v1/inventory/by-product	Crear Inventario por Producto	🔒	▼
GET	/api/v1/inventory/by-product	Listar todos los Inventarios por Producto con filtros	🔒	▼
GET	/api/v1/inventory/by-product/{id}	Obtener Inventario por Producto por ID	🔒	▼
DELETE	/api/v1/inventory/by-product/{id}	Eliminar Inventario por Producto por ID	🔒	▼
POST	/api/v1/inventory/by-batch	Crear Inventario por Lote	🔒	▼
GET	/api/v1/inventory/by-batch	Listar todos los Inventarios por Lote con filtros	🔒	▼
GET	/api/v1/inventory/by-batch/{id}	Obtener Inventario por Lote por ID	🔒	▼
DELETE	/api/v1/inventory/by-batch/{id}	Eliminar Inventario por Lote por ID	🔒	▼

Authentication

POST	/api/v1/authentication/sign-up	Register a new user account	🔒	▼
POST	/api/v1/authentication/sign-in	Authenticate and sign in user	🔒	▼
PUT	/api/v1/authentication/change-role	Change user role	🔒	▼

StockAlert

GET	/api/alerts		🔒	▼
-----	-------------	--	---	---

4.2. Landing Page & Mobile Application Implementation

4.2.1. Sprint 1

Este informe documenta el progreso realizado durante la fase de definición de requisitos del proyecto, que incluye entrevistas con los interesados y creación de artefactos antes y después de la implementación de la aplicación móvil. Proporcionar una visión clara del avance y garantizar una comunicación efectiva entre el equipo de desarrollo y los interesados son los objetivos principales.

Durante esta etapa, se realizaron extensas entrevistas con los interesados para comprender sus necesidades, expectativas y requisitos particulares para la aplicación. Las entrevistas proporcionaron información útil que ayudó a definir los requisitos del proyecto.

Se realizaron actividades de creación de artefactos antes y después de la implementación de la aplicación mobile, además de entrevistas. Estos objetos fueron:

4.2.1.1. Sprint Planning 1

Sprint #	Sprint 1
Sprint Planning Background	
Date	2025-10-8
Time	22:23 PM
Location	UPC - Monterrico
Prepared by	Camila Sanchez Rios
Attendees (to planning meeting)	Jocelyn Almerco, Kevin Chi, Alejandro Oroncoy, Jeremy Pauar
Sprint n-1	
Review	En resumen se desarrollaron avances en la landing page, backend y frontend mobile
Summary	
Sprint Planning Background	Durante esta etapa, se llevó a cabo una exhaustiva verificación de la funcionalidad de la landing page diseñada para el proyecto. El objetivo principal fue asegurar que la landing page cumpla con los estándares de calidad y proporcionar una experiencia óptima para los visitantes. Asimismo, se realizó la primera implementación de la aplicación móvil y el despliegue del backend
Sprint Goal & User Stories	
Sprint 1 Goal	Desarrollar la funcionalidad de la página web con i18n e implementar el backend. Se considerará que el objetivo del sprint se ha cumplido si todas las historias de usuario relacionadas con la landing page están implementadas y si todas las historias de usuario relacionadas con la implementación de APIs para la gestión de una entidad están completadas en un 80%.
Sprint Velocity	Se establece un Velocity de 24 Story Points para este Sprint.
Sum of Story Points	24 Story Points

4.2.1.2. Sprint Backlog 1

Objetivo del Sprint:

Desarrollar la funcionalidad principal del sistema de gestión de inventario para bodegas, implementando los módulos de productos, inventario y reportes, así como los servicios API de autenticación y creación de productos.

El Sprint se considera completado si las historias de usuario relacionadas con la gestión de inventarios y APIs de usuario están implementadas y validadas mediante pruebas unitarias e integración.

Sprint #	Sprint n	User Story	Work-Item / Task	Status			
				(To-do / In-	Estimation (Hours)	Assigned To	Process / To-Review / Done)

Sprint #	Sprint n	User Story	Work-Item / Task					
1	Sprint 1	US01	Registrar producto nuevo	TSK-01	Implementar formulario de registro	Formulario para crear producto con validación de campos requeridos; al guardar debe persistir en BD y aparecer en inventario.	6	Jeremy Done
1	Sprint 1	US02	Editar información de producto	TSK-02	Desarrollar módulo de edición	Edición de un producto existente con validaciones; guarda cambios válidos y muestra errores si son inválidos.	5	Jocelyn Done
1	Sprint 1	US03	Registrar salida de producto	TSK-03	Control de stock de salida	Registrar salida si hay stock suficiente, descontar cantidad y crear movimiento; rechazar si supera stock.	5	Kevin Done
1	Sprint 1	US05	Usuario de inventario	TSK-04	Crear interfaz de autenticación	UI para registro/login del usuario de inventario integrada con APIs de autenticación.	6	Alejandro Done
1	Sprint 1	US06	Buscar productos en inventarios	TSK-05	Implementar búsqueda avanzada	Búsqueda por nombre/código con resultados en tiempo real y mensaje "No se encontraron resultados" si no hay coincidencias.	5	Jeremy Done
1	Sprint 1	US07	Añadir etiquetas para productos	TSK-06	Crear módulo de etiquetas	Gestión de etiquetas: crear/sugerir existentes y filtrar por etiqueta en inventario.	5	Jocelyn Done
1	Sprint 1	US08	Generar estadística de productos más vendidos	TSK-07	Implementar gráfico de ventas	Ranking por periodo (unidades/ingresos) y opción de exportar a PDF.	6	Kevin Done
1	Sprint 1	US12	Generar reportes de inventario	TSK-08	Crear vista de reportes	Generación de reportes por fecha/categoría/stock mínimo con opción de exportar a PDF o Excel.	6	Alejandro Done
1	Sprint 1	US17	Registrar lote de productos	TSK-09	Programar ingreso por lotes	Registro de múltiples productos en un lote con proveedor, fecha y condiciones; visualizar detalle del lote.	5	Kevin Done
1	Sprint 1	TS01	API Registro de usuarios	TSK-10	Desarrollar endpoint /sign-up	Endpoint de registro que retorna token (201) y valida duplicados/formatos (400/409).	4	Alejandro Done
1	Sprint 1	TS02	API Autenticación de usuarios	TSK-11	Desarrollar endpoint /sign-in	Endpoint de login que emite access/refresh tokens (200) y maneja 401/403 para credenciales inválidas.	4	Jeremy Done
1	Sprint 1	TS03	API Creación de productos	TSK-12	Crear endpoint /products	Crear producto (201) con validaciones y unicidad; soportar imagen/código de barras; errores 400/409.	5	Jocelyn Done
1	Sprint 1	TS04	API Consulta de productos	TSK-13	Implementar GET /products/by-category	Listado paginado/ordenado; estado vacío si no hay registros.	4	Kevin Done
1	Sprint 1	TS05	API Productos por categoría	TSK-14	GET /products/by-tag	Filtro por categoría/etiqueta ignorando mayúsculas/acentos; permitir limpiar filtros.	4	Jeremy Done

Sprint #	Sprint n	User Story		Work-Item / Task					
1	Sprint 1	TS06	API Productos por etiqueta	TSK-15	Endpoint /products/by-tag	Autocompletado y combinación con otros filtros activos; lista vacía si no hay coincidencias.	4	Kevin	Done
1	Sprint 1	TS09	API inventario por producto	TSK-16	Endpoint /inventory/by-product	Consultar stock y últimos movimientos por producto; 404 si no existe.	6	Alejandro	Done
1	Sprint 1	US18	Visualizar historial de stock	TSK-17	Crear historial de cambios	Gráfica de variación de stock por producto con filtro de periodo.	5	Jocelyn	To-Do
1	Sprint 1	US20	Añadir etiquetas a productos	TSK-18	Implementar visualización en interfaz	Alta/edición con etiquetas nuevas o existentes; búsqueda por etiqueta.	4	Jeremy	To-Do
1	Sprint 1	US21	Buscar productos en inventario	TSK-19	Integrar búsqueda en inventario	Búsqueda por nombre y filtros por categoría desde módulo de inventario.	5	Kevin	To-Do
1	Sprint 1	US23	Registrar ingreso de productos	TSK-20	Crear flujo de entrada	Ingreso individual desde formulario y soporte para carga masiva por archivo.	5	Alejandro	To-Do
1	Sprint 1	US24	Visualizar resumen de stock por categoría	TSK-21	Mostrar gráfico resumen	Tabla/gráfico con totales y stock promedio por categoría, con acceso al detalle.	5	Jocelyn	To-Do

SPRINT BACKLOG 1: <https://trello.com/b/8Uj3aB6o/stockwise-sprint1>

4.2.1.3. Development Evidence for Sprint Review

Durante este sprint, se han realizado avances significativos en la implementación de la Landing Page, backend y frontend del aplicativo mobile. Se han completado varias historias de usuario tanto de la landing page como del backend y se han realizado múltiples commits en los repositorios correspondientes.

Repository	Branch	Component	Commit Id	Commit Message	Committed on (Date)
https://github.com/upc-pre-202520-1ACC0238-12612-Stoq/Web-Services	Main	Backend	a83775a0459b3c8f7ca1a8e41782c349a9e849b3	initial commit>	Octubre 09, 2025
https://github.com/upc-pre-202520-1ACC0238-12612-Stoq/Landing-Page	Develop	Landing Page	27811d1b1f7fae66c103cf20dd69dde3615c416b	initialize React application with Vite	Octubre 2, 2025
https://github.com/upc-pre-202520-1ACC0238-12612-Stoq/Native-mobile-development	Development	Mobile Frontend	13b7a69ca9f225b3b39bd1e6bd28397243c16596	fix response to endPoint login	Ocrubre 9, 2025

4.2.1.4. Testing Suite Evidence for Sprint Review

En esta sección se presenta el conjunto de **Unit Tests, Integration Tests y Acceptance Tests (BDD)** automatizados desarrollados durante el Sprint, para los Web Services del módulo **Inventories** del proyecto **Stoq (Web-Services)**.

El stack de testing utilizado fue **C#/.NET 8, xUnit, Moq, EF Core InMemory, Microsoft.AspNetCore.Mvc.Testing y SpecFlow (Gherkin)**.

Cobertura por pieza de código

- **InventoryByProductQueryService**
 - Unit: verifica que `GetAllAsync()` delega en `IInventoryByProductRepository.ListAsync()`.
 - Unit: verifica que `GetByIdAsync(id)` llama `FindByIdAsync(id)` y retorna `null` si no existe.
- **InventoryByBatchCommandService**

- Unit: `Handle(CreateInventoryByBatchCommand)` crea `InventoryByBatch`, llama `AddAsync` y `unitOfWork.CompleteAsync()`.
- Unit: `DeleteAsync(id)` elimina y confirma unidad de trabajo cuando existe; retorna `false` si no encuentra.
- **Integration (Web API)**
 - Endpoints de Inventario por Lote y por Producto con **EF Core InMemory**, levantados con **WebApplicationFactory**.
- **Acceptance (BDD)**
 - .feature "Crear inventario por lote" y "Eliminar inventario por lote" con escenarios Gherkin; *step definitions* en C# validando respuestas HTTP y estado persistido.

A continuación, la tabla de commits **relacionados exclusivamente con testing**:

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
https://github.com/upc-pre-202520-1ACC0238-12612-Stoq/Web-Services	feature/testing-suite	1a9c7e4	test(unit): add xUnit+Moq tests for InventoryByProductQueryService	Se agregaron pruebas unitarias que validan <code>GetAllAsync</code> y <code> GetByIdAsync</code> usando dobles de prueba de <code>IInventoryByProductRepository</code> . Se verifican llamadas a <code>ListAsync()</code> / <code>FindByIdAsync()</code> y retornos nulos cuando no existe el Id.	08/10/2025
https://github.com/upc-pre-202520-1ACC0238-12612-Stoq/Web-Services	feature/testing-suite	3f21bd0	test(unit): cover InventoryByBatchCommandService (Handle/DeleteAsync)	Se añadieron pruebas con Moq para asegurar que <code>Handle</code> llama <code>AddAsync</code> y <code>CompleteAsync</code> . <code>DeleteAsync</code> elimina cuando encuentra el Id y retorna <code>false</code> si no existe.	08/10/2025
https://github.com/upc-pre-202520-1ACC0238-12612-Stoq/Web-Services	feature/integration-tests	7c5d2a9	test(integration): API endpoints with WebApplicationFactory + EF InMemory	Se configuró <code>WebApplicationFactory</code> y una BD InMemory para probar endpoints de creación/consulta/eliminación de inventarios por lote y por producto, verificando códigos 201/200/404.	09/10/2025
https://github.com/upc-pre-202520-1ACC0238-12612-Stoq/Web-Services	feature/acceptance-bdd	c84a110	test(bdd): SpecFlow features and step definitions for inventory flows	Se añadieron .feature con escenarios Gherkin (crear y eliminar inventario por lote) y steps en C# que consumen la API, asertan payload y estado en BD InMemory.	09/10/2025

Relación de tests diseñados (extracto)

- **Unit (xUnit/Moq)**
 - `InventoryByProductQueryServiceTests.GetAllAsync_ReturnsRepositoryList`
 - `InventoryByProductQueryServiceTests.GetByIdAsync_ReturnsEntityOrNull`
 - `InventoryByBatchCommandServiceTests.Handle_PersistsAndCommits`
 - `InventoryByBatchCommandServiceTests.DeleteAsync_DeletesWhenFound_ReturnsTrue`
 - `InventoryByBatchCommandServiceTests.DeleteAsync_WhenNotFound_ReturnsFalse`
- **Integration (WebApplicationFactory + EF InMemory)**
 - POST `/api/inventories/batches` → 201 y body con Id generado.
 - GET `/api/inventories/batches/{id}` → 200 con entidad o 404 si no existe.
 - DELETE `/api/inventories/batches/{id}` → 204 cuando elimina.
- **Acceptance BDD (SpecFlow/Gherkin) – ejemplo de .feature (resumen):**
 - **Crear inventario por lote**

```

Feature: Crear inventario por lote
  Scenario: Crear inventario con datos válidos
    Given un payload válido de inventario por lote
    When envío POST a /api/inventories/batches
    Then la respuesta es 201 Created
    And el inventario persiste en la base de datos
  
```

- o **Eliminar inventario por lote**

```
Scenario: Eliminar inventario existente
Given existe un inventario por lote con Id 10
When envio DELETE a /api/inventories/batches/10
Then la respuesta es 204 No Content
And el recurso ya no está disponible
```

Repository and test routes: [Web-Services/tests/Inventories.*](#) (folders **Unit**, **Integration**, **Acceptance** with SpecFlow).

4.2.1.5. Execution Evidence for Sprint Review

During this Sprint, several milestones have been reached in the implementation of the Landing Page. In addition, progress was made in the frontend of the mobile applications. The following tasks were completed:

- Implementation of the Benefits section
- Visualization of functionalities.
- Visualization of available plans.
- Implementation of Internationalization

Screenshots

Landing Page

The screenshot shows the StockWise landing page. The main headline reads "Controla tu inventario inteligentemente". Below it, a subtext says "Automatiza análisis, optimiza tus procesos y toma decisiones inteligentes con StockWise. La plataforma de gestión de inventario que tu negocio necesita." A "Comenzar Gratis" button is visible. To the right, there is a preview of a dashboard titled "USERS: LAST 7 DAYS USING MEDIAN" which displays various inventory metrics such as "Stock Level (%)", "Order Picking Speed (SPS)", and "Inventory Turnover". At the bottom, there is a section titled "Potencia tu negocio con StockWise" with a "Activar Windows" link.

Funciones claves

Potencia tu negocio con StockWise

Descubre cómo nuestras características avanzadas pueden transformar la gestión de tu inventario



Sección de Planes Disponibles

Planes que se adaptan a tu negocio

Encuentra el plan perfecto para tus necesidades. Comienza gratis y escala según crezcas.

The chart compares three plans:

- Plan Gratuito**: Perfecto para comenzar. \$0 /mes. Includes: Hasta 50 productos, 1 usuario, Reportes básicos, Soporte por email, Actualizaciones gratuitas. Call-to-action: Comenzar Gratis.
- Plan Emprendedor**: Para pequeñas empresas. \$30 /mes. Call-to-action: Comenzar Prueba. Includes: Hasta 500 productos, 5 usuarios, Reportes avanzados, Soporte prioritario, Integraciones básicas, Backup automático.
- Plan Empresarial**: Para empresas en crecimiento. \$70 /mes. Call-to-action: Contactar Ventas. Includes: Productos ilimitados, Usuarios ilimitados, Reportes personalizados, Soporte 24/7, Todas las integraciones, API completa, Gestor de cuentas dedicado.

Activar Windows
Ve a Configuración para activar Windows.

¿Necesitas algo personalizado? [Contáctanos para planes empresariales](#)

Sección de cambio de idioma

The screenshot shows the StockWise website's header with a language dropdown set to 'EN'. A red arrow points from the 'EN' button down towards the testimonial section below.

StockWise

Home Features Pricing Testimonials Sign In Free Trial EN

Need something custom? [Contact us for enterprise plans](#)

What our clients say

Thousands of companies trust StockWise to efficiently manage their inventory

★★★★★ "StockWise has revolutionized our inventory management. Real-time reports have helped us reduce costs by 30%."

Maria Gonzalez Worker

★★★★★ "The integration with our existing systems was perfect. Now we have complete visibility of our inventory."

Carlos Rodriguez Worker

★★★★★ "Since implementing StockWise, our stock issues have drastically decreased. It's an indispensable tool."

Ana Lopez Worker

Mobile Frontend

The screenshot shows an IDE interface with several windows:

- Left Panel:** Shows the project structure under "Android". It includes "app", "Resource Manager", "shared", "sharedkernel.infrastructure.network", and "StockApplication.kt".
- Middle Panel:** Displays the Java code for a class named "ReportsView". The code includes imports like "com.stoq.StockWise", "Alerts", "AuditReport", "di", "Iam", "Inventory", "ProductCatalog", and "StockApplication.kt".
- Right Panel:** Shows a mobile application screenshot titled "Stock Wise". The screen is titled "Inicio de sesión" and contains fields for "Correo electrónico" (kevin@gmail.com) and "Contraseña" (redacted). There are buttons for "Iniciar con google", "Iniciar sesión", and "¿No tienes una cuenta? Registrarse".
- Bottom Panel:** A terminal window showing command-line activity related to the project.

4.2.1.6. Services Documentation Evidence for Sprint Review

En esta sección se evidencia la documentación de los servicios implementados para el sprint actual que conforman el backend.

Backend

The screenshot shows the Swagger UI for the "Lot" API version 1.0, OAS 3.0. The URL is /swagger/v1/swagger.json. The interface includes a navigation bar with "Select a definition" set to "Lot API v1" and a search bar. Below the navigation, there is a section for "Authentication" which lists three endpoints:

- POST /api/v1/authentication/sign-up**: Register a new user account.
- POST /api/v1/authentication/sign-in**: Authenticate and sign in user.
- PUT /api/v1/authentication/change-role**: Change user role.

Authentication

This is a continuation of the previous screenshot, focusing on the "Authentication" section of the Lot API. It lists the same three endpoints as above, each with a lock icon indicating security status.

Inventory

Inventory	Operaciones del inventario general, por producto y por lote.	^
GET	/api/v1/inventory Obtener todos los registros de inventario	🔒 ▾
POST	/api/v1/inventory/by-product Crear Inventario por Producto	🔒 ▾
GET	/api/v1/inventory/by-product Listar todos los Inventarios por Producto con filtros	🔒 ▾
GET	/api/v1/inventory/by-product/{id} Obtener Inventario por Producto por ID	🔒 ▾
DELETE	/api/v1/inventory/by-product/{id} Eliminar Inventario por Producto por ID	🔒 ▾
POST	/api/v1/inventory/by-batch Crear Inventario por Lote	🔒 ▾
GET	/api/v1/inventory/by-batch Listar todos los Inventarios por Lote con filtros	🔒 ▾
GET	/api/v1/inventory/by-batch/{id} Obtener Inventario por Lote por ID	🔒 ▾
DELETE	/api/v1/inventory/by-batch/{id} Eliminar Inventario por Lote por ID	🔒 ▾

Products

Products	Puntos de acceso disponibles para la gestión de productos.	^
GET	/api/v1/products/{productId} Obtener Producto por Id	🔒 ▾
POST	/api/v1/products Crear Producto	🔒 ▾
GET	/api/v1/products Obtener Todos los Productos	🔒 ▾
GET	/api/v1/products/by-category/{categoryId} Obtener Productos por Categoría	🔒 ▾
GET	/api/v1/products/by-tag/{tagId} Obtener Productos por Etiqueta	🔒 ▾

Combos

Combos	Puntos de acceso disponibles para la gestión de combos (kits).	^
GET	/api/v1/combos/{comboId} Obtener Combo por Id	🔒 ▾
POST	/api/v1/combos Crear Combo	🔒 ▾
GET	/api/v1/combos Obtener Todos los Combos	🔒 ▾

Report

Report	Puntos de acceso disponibles para la gestión de reportes de categoría y de stock promedio.	^
GET	/api/v1/reports Obtener todos los reportes generales	🔒 ▾
POST	/api/v1/reports/category Crear CategoryReport	🔒 ▾
GET	/api/v1/reports/category Listar CategoryReports	🔒 ▾
GET	/api/v1/reports/category/{id} Obtener CategoryReport por ID	🔒 ▾
GET	/api/v1/reports/category/by-date Filtrar CategoryReports por fecha	🔒 ▾
POST	/api/v1/reports/stock-average Crear StockAverageReport	🔒 ▾
GET	/api/v1/reports/stock-average Listar StockAverageReports	🔒 ▾
GET	/api/v1/reports/stock-average/{id} Obtener StockAverageReport por ID	🔒 ▾
GET	/api/v1/reports/stock-average/by-date Filtrar StockAverageReports por fecha	🔒 ▾

4.2.1.7. Software Deployment Evidence for Sprint Review

Landing Page

Para el despliegue de la landing page se realizaron los siguientes pasos:

1. Preparación del proyecto

Se organizó el proyecto con todos los archivos necesarios del sitio web:

- Archivos HTML, CSS, JavaScript e imágenes
- Estructura de carpetas clara (`/css`, `/js`, `/images`, etc.)

2. Creación de cuenta o acceso a Netlify

Se accedió a <https://www.netlify.com> para iniciar sesión o crear una cuenta, vinculándola con un proveedor de repositorios como GitHub, GitLab o Bitbucket.

3. Nuevo sitio desde Git

Se eligió la opción "**Add new site**" > "**Import an existing project**" para conectar el repositorio del proyecto de la landing page.

4. Autorización y selección del repositorio

Se autorizó a Netlify a acceder al repositorio y se seleccionó el repositorio correspondiente al proyecto.

5. Configuración del despliegue

Durante la configuración:

- Se indicó la rama que contiene el código (por ejemplo, `main`)
- Se dejó vacío el campo de build si el proyecto no requiere compilación
- Se indicó el directorio de publicación (por ejemplo, `/` si los archivos están en la raíz)

6. Despliegue automático

Se lanzó el primer despliegue, y Netlify generó automáticamente una URL pública para acceder al sitio.

7. Personalización de dominio (opcional)

Se puede añadir un dominio personalizado desde la sección de configuración de dominio. Netlify gestiona automáticamente el certificado SSL (HTTPS).

8. Actualizaciones automáticas

Cada vez que se realice un push a la rama seleccionada, Netlify desplegará automáticamente los nuevos cambios.

9. Monitoreo del sitio

Desde el panel de control de Netlify es posible:

- Consultar el historial de despliegues
- Ver errores si los hay
- Configurar variables de entorno
- Ver estadísticas básicas del sitio

Backend

Para el despliegue del backend se realizaron los siguientes pasos:

1. Configuración del Entorno Docker en el VPS

Paso 1: Instalación de Docker y Docker Compose

```
# Actualizar paquetes del sistema
sudo apt update && sudo apt upgrade -y

# Instalar dependencias necesarias
sudo apt install -y apt-transport-https ca-certificates curl software-properties-common

# Agregar la clave GPG oficial de Docker
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg

# Agregar el repositorio de Docker
echo "deb [arch=amd64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg]
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list >
/dev/null

# Actualizar paquetes e instalar Docker
sudo apt update
sudo apt install -y docker-ce docker-ce-cli containerd.io docker-compose-plugin

# Iniciar y habilitar Docker
sudo systemctl start docker
sudo systemctl enable docker

# Añadir el usuario actual al grupo docker
sudo usermod -aG docker $USER
```

Paso 2: Verificación de la instalación

```
# Verificar versión de Docker
docker --version
docker compose version

# Probar Docker con un contenedor de prueba
docker run hello-world
```

2. Preparación del Proyecto ASP.NET 9

Paso 3: Creación del Dockerfile

```
# Usar imagen base de ASP.NET 9
FROM mcr.microsoft.com/dotnet/aspnet:9.0 AS base
WORKDIR /app
EXPOSE 8080
EXPOSE 8081

FROM mcr.microsoft.com/dotnet/sdk:9.0 AS build
WORKDIR /src
COPY ["StockWise.API/StockWise.API.csproj", "StockWise.API/"]
RUN dotnet restore "StockWise.API/StockWise.API.csproj"

COPY ..
WORKDIR "/src/StockWise.API"
RUN dotnet build "StockWise.API.csproj" -c Release -o /app/build

FROM build AS publish
RUN dotnet publish "StockWise.API.csproj" -c Release -o /app/publish /p:UseAppHost=false

FROM base AS final
WORKDIR /app
COPY --from=publish /app/publish .
ENTRYPOINT ["dotnet", "StockWise.API.dll"]
```

Paso 4: Creación del archivo docker-compose.yml

```
version: '3.8'

services:
  stockwise-api:
    build:
      context: .
      dockerfile: Dockerfile
    ports:
      - "8080:8080"
      - "8081:8081"
    environment:
      - ASPNETCORE_ENVIRONMENT=Production
      - ASPNETCORE_URLS=http://+:8080
      - ConnectionStrings__DefaultConnection=Host=stockwise-
db;Database=StockWiseDB;Username=stockwise_user;Password=TuPasswordSeguro123!
    depends_on:
      - stockwise-db
    networks:
      - stockwise-network
    restart: unless-stopped

  stockwise-db:
    image: postgres:15-alpine
    environment:
      - POSTGRES_DB=StockWiseDB
      - POSTGRES_USER=stockwise_user
      - POSTGRES_PASSWORD=TuPasswordSeguro123!
    ports:
      - "5432:5432"
    volumes:
      - postgres_data:/var/lib/postgresql/data
    networks:
      - stockwise-network
    restart: unless-stopped

volumes:
  postgres_data:

networks:
  stockwise-network:
    driver: bridge
```

3. Despliegue de la Aplicación

Paso 5: Clonación del Repositorio y Configuración

```
# Crear directorio para la aplicación
sudo mkdir -p /opt/stockwise
cd /opt/stockwise

# Clonar el repositorio
git clone https://github.com/upc-pre-202520-1ACC0238-12612-Stoq/Web-Services.git .

# Construir las imágenes Docker
docker compose build

# Iniciar los servicios en modo detached
docker compose up -d

# Verificar el estado de los contenedores
docker compose ps
```

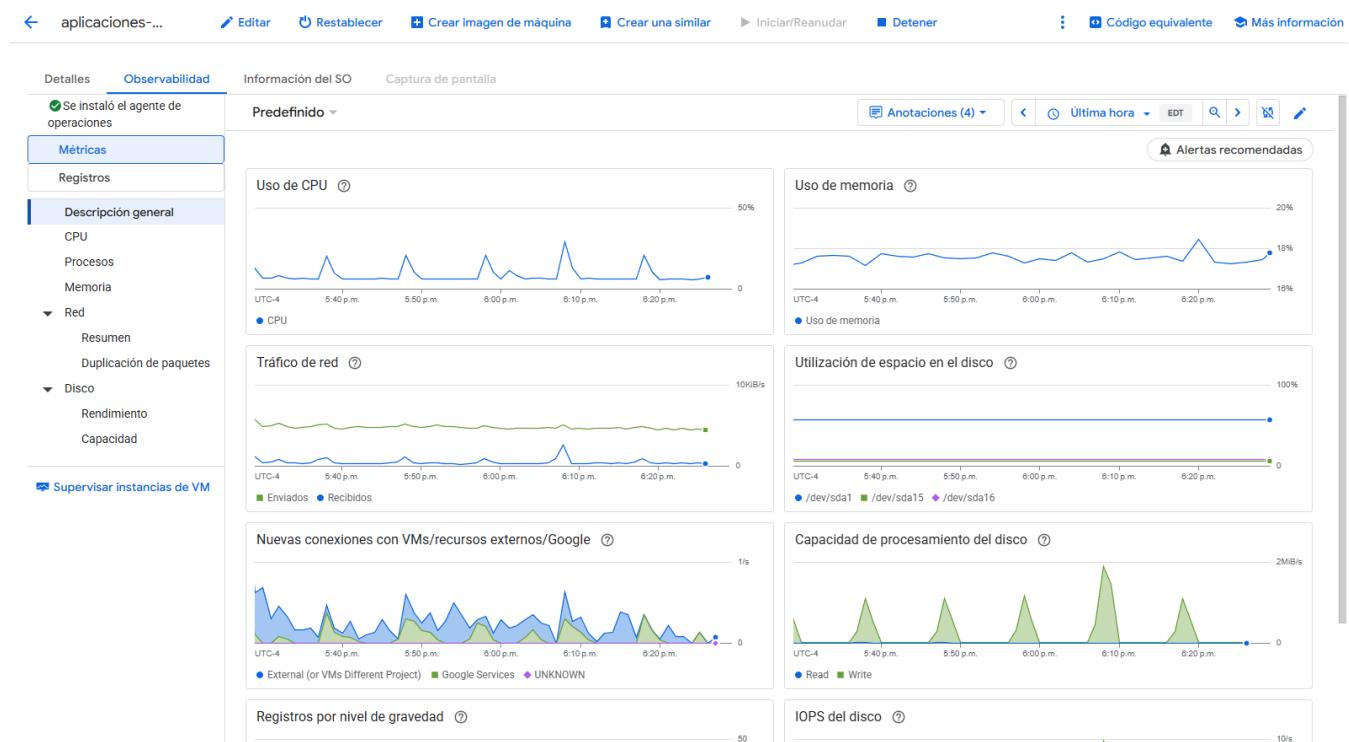
4. Configuración de Nginx como Reverse Proxy

Paso 6: Instalación y Configuración de Nginx

```
# Instalar Nginx
sudo apt install -y nginx

# Configurar SSL con Let's Encrypt
sudo apt install -y certbot python3-certbot-nginx
sudo certbot --nginx -d tu-dominio.com
```

Este proceso completo asegura el despliegue robusto y escalable de la API ASP.NET 9 con PostgreSQL utilizando contenedores Docker en un VPS.



4.2.1.8. Team Collaboration Insights during Sprint

En esta sección se proporcionan los insights para el sprint 1.

Documentation

En esta sección, el equipo destacó la importancia de mantener una documentación clara y actualizada que facilite la colaboración y el entendimiento común. Se identificó que una buena documentación agiliza la resolución de dudas y mejora la calidad del desarrollo.

A continuación se muestran los insights del equipo para la sección correspondiente:

Landing Page

El equipo observó que la landing page funciona como la primera ventana de contacto con el usuario, por lo que es vital que sea atractiva y fácil de navegar. Se resaltó la necesidad de optimizar los textos y elementos visuales para maximizar la conversión y el enganche inicial.

A continuación se muestran los insights del equipo para la sección correspondiente:

Backend

En equipo analizamos que la arquitectura del backend debía ser robusta y escalable para soportar las futuras demandas. Identificamos retos en la gestión de datos y seguridad, lo que motivó la implementación de buenas prácticas para garantizar la integridad y eficiencia.

A continuación se muestran los insights del equipo para la sección correspondiente:

Mobile Frontend

Se destacó la relevancia de adaptar la experiencia a distintos dispositivos, asegurando una interfaz intuitiva y rápida; además de evitar la saturación de pantallas.

4.3. Validation Interviews

4.3.1. Diseño de Entrevistas

Segmento #1: Bodegas especializadas por rubro

Presentación del entrevistado

- ¿Cuál es tu nombre?
- ¿Qué edad tienes?
- ¿Desde cuándo gestionas esta bodega?
- ¿Cuál es el rubro principal de tu bodega?

El usuario interactúa con la app móvil de la plataforma para gestión de inventarios.

Landing page

Acciones clave dentro del sistema

Navegación interactivo por la interfaz app móvil

User Goal: Registrar

El usuario selecciona la opción "Register", completa los campos solicitados y hace clic en el botón "Registrar".

A continuación, se muestra el panel "Add Card", donde debe llenar los campos relacionados con su tarjeta y correo electrónico.

Una vez que el proceso de pago se complete exitosamente, se notifica al usuario con un mensaje confirmando el vínculo de su tarjeta con la plataforma.

Del mismo modo, si el usuario desea retirar su información o actualizarla, lo podrá hacer a través de su perfil.

Finalmente, hacer clic en el botón "Aceptar".

User Goal: Iniciar sesión

El usuario introduce su correo y contraseña.

Luego hace clic en el botón "Log In".

Después, se le redirige al panel de perfil.

Allí puede editar su información personal y acceder a herramientas según su perfil ("Administrador" o "Empleado").

User Goal: Navegar por el Dashboard

El usuario inicia sesión desde la Landing Page.

Ingresa a la vista principal del Dashboard.

Visualiza el total de productos registrados y la fecha del último proveedor.

Visualiza un resumen de productos próximos a caducar con su respectiva fecha y stock disponible.

Accede a botones de acción rápida como "Historial", "Inventario", "Añadir Productos", "Kits" y "Devolución de productos".

User Goal: Inventario (Producto / Lote)

Ingresa a la sección de Inventario.

Revisa el listado de productos presionando el botón "por producto".

Filtrá los productos por categoría, nombre del producto, fecha o stock mínimo.

Consulta el listado con información clave: fecha de entrada, cantidad por unidad, precio, stock mínimo y unidad de medida.

User Goal: Botones Principales (Agregar Producto y Kits)

Pulsa el botón "Agregar Producto".

Rellena los campos solicitados para registrar uno nuevo.

Pulsa el botón "Crear Kit".

Combina productos existentes para crear un kit nuevo.

El usuario pulsa el botón "Añadir Productos" desde el Dashboard.

Visualiza una galería de productos existentes y accede a opciones para editarlos o duplicarlos.

Puede agregar uno nuevo haciendo clic en el botón "+", donde se despliega un formulario con campos como nombre, etiquetas, cantidad, lote, precios, fecha de caducidad y notas.

Desde el menú principal, también accede a la opción "Kits".

En esta sección, selecciona productos existentes y los combina mediante el botón "Seleccionar para kit", indicando cantidad e inventario disponible.

User Goal: Historial de Movimientos

Navega a la sección de Historial.

Visualiza entradas y salidas de productos.

Filtrá movimientos por fecha, producto o lote.

El usuario accede a la sección de Historial desde el panel principal.

Filtra los registros por tipo de gestión, categoría, stock promedio y fecha.

Visualiza los movimientos realizados, incluyendo datos como nombre del producto, fecha de consulta, precio unitario, cantidad y total.

Consulta métricas como el stock promedio, estado del producto y stock ideal.

Cuenta con botones para editar o eliminar cada registro y, para los stock promedio, exportar la información y realiza un ticket promedio.

User Goal: Alerta de Stock

El usuario accede a la sección de Alertas desde el menú superior.

Consulta una lista de productos que presentan alertas por stock mínimo o por cercanía a su fecha de vencimiento.

Visualiza detalles como categoría de alerta, tipo de producto, nombre y fecha de alerta.

Puede ingresar a una vista de detalles, generar reportes o confirmar acciones desde botones individuales por producto.

También puede eliminar una alerta específica luego de una confirmación emergente.

Preguntas principales:

- ¿Te resultó fácil encontrar cómo agregar un producto?
- ¿Qué tal fue el proceso para registrar el stock por producto y por lote?
- ¿Te resultó claro el apartado de "Próximos a caducar"? ¿Te pareció útil?
- ¿Probaste la sección de kits? ¿Te resultó útil combinar productos?
- ¿Hubo algo que no entendiste o que te confundió mientras usabas la app?
- ¿Qué te pareció el diseño general de la landing page?
- ¿Tuviste alguna dificultad visual o técnica durante la navegación?

Valoración de experiencia

- Del 1 al 10, ¿qué tan útil te pareció la app para tu bodega?
- ¿Qué funcionalidad te pareció más valiosa?
- ¿Qué función le agregarías sí o sí?
- ¿Recomendarías esta plataforma a otros dueños de bodegas? ¿Por qué?

Segmento #2: Startups y emprendedores con necesidades logísticas

Presentación del entrevistado

- ¿Cuál es tu nombre?
- ¿Qué edad tienes?
- ¿Qué tipo de producto vendes o almacenas?
- ¿Tienes local físico, almacén propio o trabajas con terceros?

Interactúa el usuario con la landing page y la versión web de nuestra plataforma para gestión gestionar stock, registrar compras y generar reportes.

Landing page

Acciones clave dentro del sistema

Navegación interactivo por la interfaz app móvil

User Goal: Registrar

El usuario selecciona la opción "Register", completa los campos solicitados y hace clic en el botón "Registrar".

A continuación, se muestra el panel "Add Card", donde debe llenar los campos relacionados con su tarjeta y correo electrónico.

Una vez que el proceso de pago se complete exitosamente, se notifica al usuario con un mensaje confirmando el vínculo de su tarjeta con la plataforma.

Del mismo modo, si el usuario desea retirar su información o actualizarla, lo podrá hacer a través de su perfil.

Finalmente, hacer clic en el botón "Aceptar".

User Goal: Iniciar sesión

El usuario introduce su correo y contraseña.

Luego hace clic en el botón "Log In".

Después, se le redirige al panel de perfil.

Allí puede editar su información personal y acceder a herramientas según su perfil ("Administrador" o "Empleado").

User Goal: Navegar por el Dashboard

El usuario inicia sesión desde la Landing Page.

Ingrera a la vista principal del Dashboard.

Visualiza el total de productos registrados y la fecha del último proveedor.

Visualiza un resumen de productos próximos a caducar con su respectiva fecha y stock disponible.

Accede a botones de acción rápida como "Historial", "Inventario", "Añadir Productos", "Kits" y "Devolución de productos".

User Goal: Inventario (Lote)

Accede a la sección de Inventario por lote.

Filtra los lotes por nombre del producto, proveedor, fecha de entrada, cantidad o precio.

Consulta el listado con información detallada como proveedor, producto, fecha, cantidad por unidad, precio y unidad de medida.

Gestiona las acciones disponibles para cada lote desde la columna correspondiente.

User Goal: Botones Principales (Agregar Producto y Kits)

Pulsa el botón "Agregar Producto".

Rellena los campos solicitados para registrar uno nuevo.

Pulsa el botón "Crear Kit".

Combina productos existentes para crear un kit nuevo.

El usuario pulsa el botón "Añadir Productos" desde el Dashboard.

Visualiza una galería de productos existentes y accede a opciones para editarlos o duplicarlos.

Puede agregar uno nuevo haciendo clic en el botón "+", donde se despliega un formulario con campos como nombre, etiquetas, cantidad, lote, precios, fecha de caducidad y notas.

Desde el menú principal, también accede a la opción "Kits".

En esta sección, selecciona productos existentes y los combina mediante el botón "Seleccionar para kit", indicando cantidad e inventario disponible.

User Goal: Historial de Movimientos

Navega a la sección de Historial.

Visualiza entradas y salidas de productos.

Filtra movimientos por fecha, producto o lote.

El usuario accede a la sección de Historial desde el panel principal.

Filtra los registros por tipo de gestión, categoría, stock promedio y fecha.

Visualiza los movimientos realizados, incluyendo datos como nombre del producto, fecha de consulta, precio unitario, cantidad y total.

Consulta métricas como el stock promedio, estado del producto y stock ideal.

Cuenta con botones para editar o eliminar cada registro y, para los stock promedio, exportar la información y realiza un ticket promedio.

User Goal: Alerta de Stock

El usuario accede a la sección de Alertas desde el menú superior.

Consulta una lista de productos que presentan alertas por stock mínimo o por cercanía a su fecha de vencimiento.

Visualiza detalles como categoría de alerta, tipo de producto, nombre y fecha de alerta.

Puede ingresar a una vista de detalles, generar reportes o confirmar acciones desde botones individuales por producto.

También puede eliminar una alerta específica luego de una confirmación emergente.

Preguntas principales:

- ¿Qué opinas de las alertas de stock mínimo/máximo y vencimiento? ¿Son suficientes?
- ¿La sección de historial de movimientos te pareció útil para revisar tus registros?
- ¿Te resultó útil la opción de devolución de productos?
- ¿Los reportes del historial son útiles para tomar decisiones?
- ¿Hubo algo que no entendiste o que te confundió mientras usabas la app?
- ¿Qué te pareció el diseño general de la landing page?
- ¿Tuviste alguna dificultad visual o técnica durante la navegación?

Valoración de experiencia

- Del 1 al 10, ¿qué tan útil te pareció la app móvil para tu bodega?
- ¿Qué funcionalidad te pareció más valiosa?
- ¿Qué función le agregarías sí o sí?
- ¿Recomendarías esta plataforma a otros emprendedores? ¿Por qué?

4.3.2. Registro de Entrevistas

Segmento 1: Bodegas especializadas por rubro

Entrevista	1	Nombre	Milagros Almerco
Edad	28	Distrito	San Martín de Porres
Captura de la entrevista:	Milagros Almerco, evaluó la app móvil StockWise y destacó su facilidad de uso al registrar productos y gestionar inventario. Aunque al inicio tuvo dudas sobre la diferencia entre "stock" y "lote", logró comprenderlo rápidamente gracias a la interfaz intuitiva. Resaltó como funcionalidades más valiosas el control de productos próximos a vencer y la visualización por lotes, que considera claves para evitar pérdidas. Valoró también el diseño moderno y claro, sin presentar dificultades técnicas. Sugirió como mejora incluir un sistema para registrar pagos pendientes o créditos. En general, calificó la app como práctica, útil y recomendada para otros dueños de bodegas.		
URL de la grabación	Ver grabación		
Timing	00:00 - 5:48		

Entrevista	2	Nombre	Elvis Aranga Mesa
Edad	31	Distrito	Santiago de Surco
Captura de la entrevista:	Elvis Aranga Meza, encargado de un mini market en Surco, compartió su experiencia utilizando la app móvil para la gestión de bodegas, destacando la facilidad de uso tanto para agregar productos como para navegar por las funciones principales, y calificando su utilidad con un 9 sobre 10. Consideró el diseño visual adecuado y sencillo, aunque sugirió mejorar el tamaño de la interfaz en dispositivos móviles. Resaltó la utilidad de funciones como la visualización de fechas de vencimiento para evitar mermas y la organización de productos por lote, lo cual facilita la gestión de promociones. Sugirió incorporar reportes y estadísticas que permitan proyectar ventas y establecer metas. Finalmente, recomendó la plataforma a otros bodegueros, especialmente por su precio y utilidad, aunque señaló la importancia de brindar asistencia a usuarios con menos experiencia tecnológica.		
URL de la grabación	Ver grabación		
Timing	5:48 - 11:47		
Entrevista	3	Nombre	Catalina Villa Guerra
Edad	28	Distrito	Breña
Captura de la entrevista:	Catalina es una administradora de 28 años, encargada de gestionar una bodega familiar en el distrito de Breña. Ella menciona que las funciones que más le gustaron fueron la de alerta de productos próximos a vencer y la de control de stock, ya que considera que le ayudarían a evitar pérdidas por productos caducados y a mantener un mejor seguimiento de sus existencias. Sin embargo, percibe que estas funciones aún son básicas y podrían ofrecer opciones más avanzadas, como recordatorios personalizados o reportes automáticos. En cuanto a la interfaz, a Catalina le agrado el diseño sencillo y organizado, destacando que los botones son fáciles de identificar y las categorías están bien distribuidas. También comentó que la aplicación le resultaría útil para delegar mejor el control del inventario entre los miembros de su familia, facilitando así la administración general de la bodega.		
URL de la grabación	Ver grabación		
Timing	11:48 - 21:31		

Segmento 2: Startups y emprendedores en expansión con necesidades logísticas

Entrevista	4	Nombre	Juan Carlos Ramírez
Edad	49	Distrito	Surquillo
Captura de la entrevista: 	Juan Carlos Ramírez, emprendedor de 49 años, compartió su experiencia al utilizar la aplicación para la gestión de inventarios, destacando la facilidad de uso en el registro y control de productos, así como la organización del historial de movimientos. Consideró que estas herramientas podrían ayudarlo significativamente en su proceso de digitalización, ya que actualmente gestiona las entradas y salidas de manera manual mediante boletas y facturas físicas. En cuanto al diseño, señaló que la interfaz es clara y ordenada, aunque sugirió mejorar la personalización de reportes para adaptarlo mejor a las necesidades de su negocio. Finalmente, destacó que funcionalidades como la alerta de stock bajo y la gestión por lotes serían de gran apoyo para tener un control más preciso del inventario, y que la incorporación de estas herramientas digitales le permitiría automatizar procesos y liberar tiempo operativo en su gestión diaria.		
URL de la grabación	Ver grabación		
Timing	21:31 - 25:14		
Entrevista	5	Nombre	Leonardo Gamboa
Edad	26	Distrito	San Miguel
Captura de la entrevista: 	Leonardo G., emprendedor que maneja actualmente su inventario de manera manual con hojas de Excel y una libreta, probó la plataforma y comentó que le pareció intuitiva, útil y eficaz para el control de su negocio. Señaló que la sección de alertas de stock le resultó especialmente atractiva, ya que le permitiría evitar ventas de productos agotados y responder más rápido a la demanda constante que maneja. En cuanto al diseño, mencionó que la interfaz es clara y sencilla, con botones fáciles de identificar y una navegación fluida. Leonardo destacó que la herramienta podría ayudarlo a organizar mejor sus pedidos y actualizar su inventario sin esfuerzo, reemplazando las hojas manuales que utiliza actualmente. También valoró que la plataforma sea accesible y de fácil uso, algo clave para quienes aún no están acostumbrados a los sistemas digitales, pero buscan dar ese paso hacia la automatización de su negocio.		
URL de la grabación	Ver grabación		
Timing	25:14 - 29:36		
Entrevista	6	Nombre	Gael Rivera
Edad	xx	Distrito	xx
Captura de la entrevista: 	Eduardo de Rivera Sosa, un emprendedor que importa productos desde Asia para venderlos en Perú, comentó que la idea del proyecto le pareció atractiva y muy necesaria para negocios que manejan múltiples referencias de productos. Durante la prueba, resaltó la facilidad de uso y la organización del panel principal, especialmente en el registro de productos y control de stock. Mencionó que actualmente lleva su inventario en Excel y organiza sus pedidos en Notion, por lo que considera que la app podría integrar y simplificar ambos procesos en un solo espacio digital. Valoró la función de alerta de stock bajo y las opciones de historial de movimientos, que le permitirían tener una visión más clara de sus existencias y reducir pérdidas por errores humanos o por actualizaciones tardías. En cuanto a la apariencia visual, Eduardo comentó que la landing page le pareció moderna y atractiva, destacando que transmite confianza y profesionalismo. Sin embargo, sugirió que la exportación de datos a Excel o PDF podría mejorarse con formatos más personalizables.		
URL de la grabación	Ver grabación		
Timing	29:36 - 33:36		

4.3.3. Evaluaciones según heurísticas

APP A EVALUAR

StockWise – App móvil de gestión de inventario para bodegas y emprendimientos

TAREAS A EVALUAR

El alcance de esta evaluación incluye la revisión de la usabilidad en las siguientes tareas:

1. Registro de un nuevo usuario.
2. Inicio de sesión con credenciales existentes.
3. Registro y edición de productos.
4. Búsqueda y filtrado de productos por nombre, categoría o lote.
5. Visualización de productos próximos a vencer.
6. Registro de salida o venta de productos.

7. Consulta del historial de movimientos.
8. Visualización de stock y control por lotes.
9. Cierre de sesión.

No están incluidas en esta versión de la evaluación las siguientes tareas:

1. Integración con sistemas externos de pago.
2. Sincronización con otros dispositivos.
3. Administración de usuarios con distintos roles.
4. Funcionalidades futuras de geolocalización o predicción de demanda.

ESCALA DE SEVERIDAD

Nivel	Descripción
1	Problema superficial: se supera fácilmente y no interfiere con la tarea.
2	Problema menor: ocurre con cierta frecuencia o requiere más esfuerzo.
3	Problema mayor: impide completar tareas o genera frustración frecuente.
4	Problema crítico: bloquea la tarea o impide el uso del sistema.

TABLA RESUMEN DE HALLAZGOS

#	Problema	Escala de severidad	Heurística/Principio violado(a)
1	No hay indicador visual claro de diferencia entre "Stock" y "Lote", lo cual puede generar confusión inicial.	2	Usability – Claridad del lenguaje y correspondencia con el mundo real.
2	No existe un acceso rápido desde la pantalla principal al registro de ventas o salidas de productos.	3	Usability – Flexibilidad y eficiencia de uso.
3	No hay opción para registrar créditos o pagos pendientes dentro del flujo principal.	3	Information Architecture – Is it complete?
4	El tamaño de algunos botones y textos resulta pequeño en ciertos dispositivos móviles.	2	Inclusive Design – Accesibilidad visual.
5	No existe vista resumida o gráfica de reportes o estadísticas de ventas.	2	Usability – Visibilidad del estado del sistema.
6	La interfaz no muestra recordatorios personalizables para productos próximos a vencer.	2	Usability – Ayuda y prevención de errores.

DESCRIPCIÓN DE PROBLEMAS

PROBLEMA #1: Confusión inicial entre "Stock" y "Lote"

Severidad: 2

Heurística violada: Usabilidad – Correspondencia entre el sistema y el mundo real.

Problema:

Durante las pruebas, los usuarios Milagros y Catalina manifestaron dudas iniciales sobre la diferencia entre "stock" y "lote". Aunque lograron comprenderlo después de explorar la app, el término no es intuitivo para todos los perfiles de usuarios, especialmente aquellos con poca experiencia digital.

Recomendación:

Agregar una breve descripción o globo de ayuda que explique la diferencia entre ambos términos (por ejemplo: "Stock: cantidad total disponible. Lote: grupo de productos con la misma fecha de vencimiento").

PROBLEMA #2: Falta de acceso rápido al registro de ventas

Severidad: 3

Heurística violada: Usabilidad – Flexibilidad y eficiencia de uso.

Problema:

Usuarios como Elvis y Juan Carlos señalaron que, aunque el registro de productos es sencillo, no existe un acceso directo desde el dashboard para registrar ventas o salidas. Esto implica más pasos de navegación y retrasa la gestión diaria.

Recomendación:

Incluir un botón visible o un acceso rápido en el dashboard para registrar ventas o salidas directamente, sin pasar por el menú lateral.

PROBLEMA #3: Ausencia de módulo de pagos o créditos pendientes

Severidad: 3

Heurística violada: Information Architecture – Is it complete?

Problema:

Varios usuarios (Milagros y Juan Carlos) mencionaron la necesidad de registrar pagos pendientes o créditos, dado que muchas bodegas operan con ventas fiadas. La ausencia de esta función limita la utilidad integral de la app.

Recomendación:

Agregar un submódulo para registrar clientes con deudas y fechas de pago, con alertas visuales y posibilidad de marcar los pagos como cancelados.

PROBLEMA #4: Tamaño reducido de texto y botones

Severidad: 2

Heurística violada: Inclusive Design – Accesibilidad visual.

Problema:

Elvis mencionó que el tamaño de algunos elementos no se adapta correctamente en pantallas pequeñas, dificultando la interacción para usuarios con menos destreza táctil.

Recomendación:

Optimizar el diseño responsive para móviles, asegurando un tamaño mínimo de toque de 44x44 px y contraste adecuado según WCAG 2.1.

PROBLEMA #5: Falta de reportes visuales o estadísticas

Severidad: 2

Heurística violada: Usabilidad – Visibilidad del estado del sistema.

Problema:

Los usuarios destacaron que la app podría mejorar si mostrara reportes o gráficos de ventas y stock. Actualmente, la información está distribuida, pero no se resume de forma visual.

Recomendación:

Incorporar una vista de “Resumen de actividad” con gráficos simples (barras o pastel) sobre productos más vendidos, alertas y stock por categoría.

PROBLEMA #6: Recordatorios limitados para productos próximos a vencer

Severidad: 2

Heurística violada: Usabilidad – Prevención de errores.

Problema:

Catalina valoró la alerta de productos próximos a vencer, pero comentó que sería mejor si pudiera configurar recordatorios personalizados según tipo de producto o periodo (por ejemplo, 5, 10 o 30 días antes del vencimiento).

Recomendación:

Permitir que el usuario ajuste la anticipación de alertas y el canal de notificación (push o correo).

CONCLUSIÓN GENERAL

La app **StockWise** presenta una excelente base de usabilidad y accesibilidad, destacando por su claridad visual, fluidez de navegación y aprendizaje rápido incluso en usuarios con baja experiencia tecnológica.

Las oportunidades de mejora se centran en profundizar funciones avanzadas de gestión (créditos, reportes, recordatorios personalizados) y refinar detalles visuales en dispositivos móviles.

En general, los usuarios calificaron la experiencia como positiva, práctica y recomendable, con potencial para convertirse en una herramienta estándar para la gestión de bodegas pequeñas y medianas.

Conclusiones y recomendaciones

El desarrollo de las funcionalidades propuestas para StockWise responde a las necesidades reales de emprendedores, pymes y startups que buscan optimizar su gestión de inventarios de manera simple, rápida y eficiente. A través de un enfoque centrado en el usuario, se priorizan características esenciales como el control de stock, la generación de alertas inteligentes, reportes visuales y una experiencia de uso intuitiva. Además, la estrategia de diferenciación frente a la competencia, basada en la simplicidad, accesibilidad y enfoque especializado, refuerza el potencial de StockWise para posicionarse como una solución líder en el mercado latinoamericano. Con esta base sólida, el proyecto avanza hacia la creación de una aplicación móvil de alto valor, adaptable y preparada para el crecimiento continuo.

Video App Validation

Video About-the-Product

Enlace del video: <https://youtu.be/gKQCMO4rORw?si=igx6rLDbzR-ltvqh>

Video About-the-team

Bibliografía

1. Dux Software. (2025, 9 abril). Dux Software: El Sistema de Gestión para tu Negocio. <https://www.duxsoftware.com.ar/>
2. Mecalux. (s. f.). Store fulfillment. <https://www.mecalux.pe/software/store-fulfillment>
3. SoftDolt. (s. f.). >Vendus: agiliza la gestión de tu punto de venta. <https://www.softwaredoit.es/vendus/vendus.html>

Anexos

Video de Exposición TP: