

Especificación de requisitos de software

Proyecto: EcoMarket+

Revisión: [99.99]

[Seleccionar fecha]

Contenido

1. Introducción	5
1.1. Propósito.....	5
1.2. Ámbito del Sistema	5
1.3. Definiciones, Acrónimos y Abreviaturas.....	6
1.4. Referencias	6
1.5. Visión General del Documento	8
2. Descripción General	9
2.1. Perspectiva del Producto	9
2.2. Funciones del Producto.....	9
2.3. Características de los Usuarios.....	10
2.4. Restricciones	11
2.5. Suposiciones y Dependencias	12
2.6. Requisitos Futuros	13
3. Requisitos Específicos.....	14
3.1 Requisitos comunes de los interfaces	14
3.1.1 Interfaces de usuario	14
3.1.2 Interfaces de hardware.....	15
3.1.3 Interfaces de software	15
3.1.4 Interfaces de comunicación	15
3.2 Requisitos funcionales	15
3.3 Requisitos no funcionales	22
3.3.1 Requisitos de rendimiento.....	23
3.3.2 Seguridad.....	24
3.3.3 Fiabilidad	24
3.3.4 Disponibilidad	24
3.3.5 Mantenibilidad	24

3.3.6	Portabilidad	24
4.	Diseño de la Nueva Arquitectura.....	25
5.	Planificación de la Migración.....	26
5.1	Estrategia de Migración.....	26
5.2	Riesgos identificados y plan de mitigación	27
6.	Microservicios	28
6.1	Microservicio Usuario.....	28
6.2	Microservicio Inventario.....	29
6.3.	Microservicio Sucursales.	29
6.4.	Microservicio Compras.....	30
6. 5.	Microservicio Pago.	31
6.6	Microservicio Reportes.....	31
6.7	Microservicio Seguimiento	32
6.8	Microservicio Soporte	32
6.9	Microservicio Notificaciones.	33
6.10	Microservicio Catálogo	34
7.	Diagramas UML	35
7.1	Diagramas de Caso de uso por Actor.....	35
7.1.1	Administrador.	35
7.1.2	Gerente de Tienda.	35
7.1.3	Empleado de Ventas.	36
7.1.4	Logística.	36
7.1.5	Cliente	37
7.2	Diagrama de Clases.	38
7.3	Diagramas de Arquitectura de Microservicios.....	39
8.	Aspectos Éticos.....	40
9.	Anexos.....	41
9.1	Fuentes y url Consultadas.	41

Ficha del documento

Fecha	Revisión	Autor	Modificación
[Fecha]	[Rev]	[Descripción]	[Descripción]
[Fecha]	[Rev]	[Descripción]	[Descripción]

Documento validado por las partes en fecha: [Fecha]

Por el cliente

Por la empresa suministradora

[Firma]

[Firma]

Sr./Sra. [Nombre]

Sr./Sra.[Nombre]

1. Introducción

En esta sección se proporcionará una introducción a todo el documento de Especificación de Requisitos Software (ERS). Consta de varias subsecciones: propósito, ámbito del sistema, definiciones, referencias y visión general del documento.

1.1. Propósito

Documentar los requisitos funcionales y no funcionales del sistema de EcoMarket+, migrando de una arquitectura monolítica a una basada en microservicios, para mejorar escalabilidad, rendimiento y mantenibilidad.

1.2. Ámbito del Sistema

Nombre del Sistema:

EcoMarket+

Qué incluye:

- **Gestión de usuarios:** Creación, permisos y roles (admin, gerente, logística, vendedor, cliente).
- **Inventario:** Actualización en tiempo real, reportes (PDF/Excel), alertas de stock.
- **Ventas:** Facturación electrónica, descuentos, historial de compras.
- **Logística:** Seguimiento de envíos, optimización de rutas, gestión de proveedores.
- **Integraciones:** APIs REST de pago (WebPay) y notificaciones en tiempo real.

Qué excluye:

- Desarrollo de hardware específico.
- Procesamiento offline (requiere conexión permanente).

Beneficios y Metas:

- **Escalabilidad:** Arquitectura de microservicios para soportar crecimiento regional (ej: Valdivia, Antofagasta).
- **Rendimiento:** Reducción de tiempos de respuesta (<5 segundos en operaciones críticas).
- **Disponibilidad:** 99.9% uptime (excluyendo mantenimiento programado).
- **Mantenibilidad:** Facilidad para actualizar módulos sin afectar el sistema completo.

Objetivo final: Migrar de un sistema monolítico a uno modular basado en microservicios y APIs REST, que mejore la eficiencia operativa y la experiencia del usuario, alineado con la expansión de EcoMarket SPA.

1.3. Definiciones, Acrónimos y Abreviaturas

A continuación, se presentan los términos, siglas y abreviaciones utilizadas en este documento, junto con su respectiva definición, para facilitar la comprensión del contenido técnico relacionado con el sistema EcoMarket+:

- **API (Application Programming Interface):** Interfaz que permite la comunicación entre aplicaciones a través de métodos definidos. Utilizado en EcoMarket+ para exponer funcionalidades como productos, pagos o seguimiento.
- **REST (Representational State Transfer):** Estilo de arquitectura para diseñar servicios web mediante operaciones HTTP. Utilizado en la comunicación entre microservicios.
- **CRUD (Create, Read, Update, Delete):** Operaciones básicas que se realizan sobre los datos.
- **JWT (JSON Web Token):** Estándar para el intercambio seguro de información mediante tokens firmados. Utilizado para la autenticación.
- **OAuth 2.0:** Protocolo de autorización que permite a usuarios compartir datos sin exponer sus credenciales.
- **UI (User Interface):** Interfaz de usuario, capa visual del sistema.
- **DB (Database):** Base de datos. Sistema de almacenamiento estructurado de información.
- **API Externa:** Servicio de terceros accesible por Internet que complementa funcionalidades del sistema.
- **API Gateway:** Componente intermediario que gestiona y enruta solicitudes entre el cliente y los microservicios, aplicando autenticación, control de acceso y monitoreo centralizado.
- **Auth0:** Plataforma externa para autenticación segura y gestión de identidades.
- **WebPay:** Pasarela de pago de Transbank utilizada para procesar transacciones en línea.
- **Trello:** Herramienta de gestión de tareas utilizada por el equipo para planificación colaborativa.
- **Microservicio:** Componente independiente que ejecuta una funcionalidad específica del sistema y se comunica mediante APIs.
- **EcoMarket+:** Nombre del sistema central desarrollado para la gestión de productos, ventas y operaciones de la empresa EcoMarket SPA.

1.4. Referencias

A continuación, se listan las fuentes de información, herramientas tecnológicas, APIs externas y documentación de apoyo que fueron utilizadas para la elaboración del presente documento de Especificación de Requisitos del Software (ERS) del sistema EcoMarket+:

Normativas y Documentación Técnica

IEEE Std 830-1998 – Recommended Practice for Software Requirements Specifications –
<https://ieeexplore.ieee.org/document/720574>

Autenticación y Seguridad

Auth0 – <https://auth0.com>

JWT.io – <https://jwt.io>

Notificaciones y Comunicación

SendGrid API

Procesamiento de Pagos

WebPay Plus REST API (Transbank Chile)

Tecnologías de Desarrollo

Spring Boot – <https://spring.io>

Node.js – <https://nodejs.org>

MySQL – <https://dev.mysql.com>

RabbitMQ – <https://www.rabbitmq.com>

Generación de Documentos

Apache POI – <https://poi.apache.org>

iText – <https://itextpdf.com>

Planificación y Colaboración

Trello – <https://trello.com>

GitHub – <https://github.com>

Modelado y Diagramación

DiagramasUML.com – <https://diagramasuml.com>

Apoyo Inteligente y Asistido

ChatGPT – <https://chat.openai.com>

DeepSeek – <https://deepseek.com>

Recursos Académicos

Notas de clases DUOC UC y Caso de Estudio 3 – EcoMarket SPA –

1.5. Visión General del Documento

Este documento de Especificación de Requisitos del Software (ERS) tiene como objetivo definir el desarrollo del sistema EcoMarket+, desarrollado para gestionar de manera eficiente el cambio del sistema monolítico original a una arquitectura de microservicio, contemplando los procesos operativos y comerciales de EcoMarket SPA, dedicada a la venta de productos ecológicos y sustentables.

La estructura del documento está organizada en secciones temáticas que permiten una lectura progresiva y coherente:

Sección 1 – Introducción: Presenta el propósito del documento, su alcance, definiciones clave, fuentes de referencia y una visión general de su contenido.

Sección 2 – Descripción General del Sistema: Proporciona el contexto del negocio, los perfiles de usuario, restricciones, y una visión preliminar de las funcionalidades que cubrirá el sistema.

Sección 3 – Requisitos Específicos: Define los requisitos funcionales (RF) y no funcionales (RNF) que el sistema debe cumplir, organizados con identificadores únicos. Estos requisitos están alineados con los procesos reales de la empresa y con la arquitectura planteada.

Sección 4 – Arquitectura de Microservicios: Detalla la estructura modular del sistema, describiendo cada microservicio, su responsabilidad, APIs REST expuestas, tecnologías asociadas.

Sección 5 – Aspectos Éticos y Legales: Incluye consideraciones sobre la privacidad de los datos, cumplimiento de normativas, transparencia en el uso de la información y uso responsable de tecnologías de inteligencia artificial.

Sección 6 – Diagramas del Sistema: Contiene representaciones visuales del sistema, como diagramas de casos de uso, de clases y de microservicios, para facilitar la comprensión técnica del diseño.

Sección 7 – Anexos: Presenta información complementaria como fuentes consultadas, url, y cualquier otro material relevante que respalde el desarrollo del sistema y del presente documento.

2. Descripción General

En esta sección se describe el contexto del sistema EcoMarket+, incluyendo su relación con otros componentes, usuarios clave y limitaciones técnicas. Esto sirve como base para entender los requisitos detallados en la Sección 3.

2.1. Perspectiva del Producto

EcoMarket+ es una solución modular basada en microservicios que reemplaza el sistema monolítico actual de la empresa. Está diseñada para operar de forma independiente, escalable y mantenible, a través de APIs RESTful que permiten la integración entre sus componentes internos. Cada microservicio tiene responsabilidades específicas como usuarios, inventario, compras, pagos, seguimiento, soporte, catálogo, notificaciones y reportes. Esta arquitectura desacoplada permite que cada módulo evolucione sin afectar el sistema completo.

Relación con otros sistemas:

- **WebPay:** Procesamiento de pagos en línea.

Infraestructura:

El sistema se apoya en los siguientes pilares tecnológicos:

- **MySQL** como motor de base de datos relacional por microservicio.
- **RabbitMQ** para mensajería asíncrona entre microservicios.
- **Despliegue escalable en la nube**, permitiendo crecimiento horizontal por demanda.
- **JWT + Auth0** para autenticación segura y control de acceso.
- **Microservicios desacoplados**, comunicados vía HTTP/HTTPS y formato JSON.

2.2. Funciones del Producto

El sistema EcoMarket+ permite a los distintos actores (clientes, personal interno y administradores) realizar operaciones. A través de sus microservicios, el sistema entrega las siguientes funciones clave:

Visualización y compra de productos

- Consultar el catálogo actualizado de productos ecológicos.
- Agregar productos a un carrito y realizar compras en línea.
- Revisar el historial de pedidos realizados.

Pago de pedidos

- Conexión con WebPay para realizar pagos en línea de manera segura.
- Confirmación de transacciones y registro del estado del pago.

Gestión de inventario

- Control de stock por producto y sucursal.
- Ajustes automáticos de inventario según las compras realizadas.
- Alertas por bajo stock.

Seguimiento de entregas

- Registro y consulta del estado de los pedidos (preparación, en ruta, entregado).

- Acceso a esta información tanto por clientes como por el área logística.

Generación de reportes

- Creación de reportes de ventas, usuarios e inventario en formatos PDF y Excel.
- Posibilidad de filtrar y programar reportes desde el panel de administración.

Administración de usuarios y roles

- Registro y autenticación segura de usuarios mediante Auth0 y JWT.
- Asignación de distintos roles (cliente, repartidor, gerente, etc.) con sus respectivos permisos.

Soporte al cliente

- Envío y gestión de solicitudes de ayuda o reclamos por parte de los clientes.
- Seguimiento y respuesta por parte del equipo de soporte.

Sistema de notificaciones

- Envío automático de mensajes sobre eventos relevantes del sistema (compra, estado del pedido, soporte, etc.).
- Visualización de notificaciones en el perfil del usuario.

2.3. Características de los Usuarios

La plataforma estará orientada a cinco tipos de usuarios o actores principales, cada uno con distintos niveles de acceso, responsabilidades y necesidades dentro del sistema. A continuación, se describen sus características:

1. Administrador General

Perfil: Usuario con conocimientos avanzados del sistema y encargado de la gestión global.

Responsabilidades: Crear, editar o eliminar usuarios; visualizar reportes globales; configurar parámetros generales del sistema.

Habilidades: Alto manejo de herramientas digitales, conocimientos en gestión empresarial y administración de plataformas.

Necesidades: Acceso completo al sistema, paneles de control personalizables, seguridad de la información y trazabilidad de las operaciones.

2. Gerente de Tienda

Perfil: Encargado de supervisar la operación de una tienda física o virtual.

Responsabilidades: Controlar el inventario local, aprobar pedidos, monitorear ventas, coordinar con logística y el equipo de ventas.

Habilidades: Conocimiento medio-alto en herramientas de gestión, liderazgo de equipos, análisis de ventas.

Necesidades: Interfaz clara para gestión de inventario, acceso a reportes de ventas, alertas por quiebres de stock, coordinación fluida con logística.

3. Encargado de Logística

Perfil: Usuario responsable de coordinar la recepción, almacenamiento y despacho de productos.

Responsabilidades: Registrar ingresos y salidas de mercancía, programar despachos, verificar entregas y gestionar devoluciones.

Habilidades: Conocimientos en logística y manejo de bodegas, organización y planificación.

Necesidades: Acceso al módulo de stock y movimientos de bodega, visualización de rutas y entregas, reportes de trazabilidad.

4. Vendedor

Perfil: Usuario encargado de realizar ventas y atender consultas de los clientes.

Responsabilidades: Registrar ventas, verificar disponibilidad de productos, emitir boletas o facturas, asesorar a clientes.

Habilidades: Manejo básico a intermedio de sistemas de ventas, habilidades comunicativas y comerciales.

Necesidades: Interfaz rápida e intuitiva para registrar ventas, consulta de inventario en tiempo real, historial de clientes.

5. Cliente

Perfil: Usuario final de la plataforma, que accede a productos y servicios ofrecidos por la tienda.

Responsabilidades: Navegar por el catálogo, realizar pedidos, gestionar su cuenta y hacer seguimiento de sus compras.

Habilidades: Uso básico de plataformas web o móviles.

Necesidades: Experiencia de usuario fluida, información clara de productos, métodos de pago y envío accesibles, soporte en línea.

2.4. Restricciones

A continuación, se detallan las restricciones que deben considerarse durante el desarrollo, implementación y operación del sistema EcoMarket+. Estas limitaciones están definidas por políticas organizacionales, condiciones técnicas y requerimientos de interoperabilidad:

-Políticas de la empresa

El sistema debe cumplir con los principios de sostenibilidad, seguridad de la información y transparencia definidos por EcoMarket SPA. Solo se utilizarán tecnologías que respeten la protección de datos y buenas prácticas de desarrollo.

-Limitaciones del hardware

El sistema debe ser funcional en los dispositivos existentes de la empresa, incluyendo computadores de oficina estándar y dispositivos móviles Android utilizados en tiendas o repartos. No se considera inversión en hardware especializado.

-Interfaces con otras aplicaciones

El sistema debe integrarse correctamente con servicios externos como WebPay, utilizando sus APIs REST públicas. Se deberá respetar el formato y protocolo definido por cada proveedor.

-Operaciones paralelas

Los microservicios deben permitir la ejecución de operaciones simultáneas (por ejemplo, múltiples compras o solicitudes de soporte), sin afectar la consistencia de los datos ni el rendimiento del sistema.

-Funciones de auditoría

El sistema deberá registrar eventos importantes como inicio de sesión, cambios en inventario, transacciones de pago y solicitudes de soporte. Estos logs permitirán trazabilidad en caso de incidentes.

-Funciones de control

Los roles de usuario definirán los permisos de acceso. Solo administradores podrán modificar datos críticos como precios, stock o reportes.

-Lenguaje(s) de programación

La implementación debe realizarse utilizando lenguajes compatibles con Node.js y/o Java (Spring Boot), permitiendo escalabilidad y compatibilidad con librerías modernas.

-Protocolos de comunicación

Toda la comunicación entre microservicios se realizará a través de HTTP/HTTPS y formato JSON. Para mensajería asíncrona se utilizará RabbitMQ.

-Requisitos de habilidad

Los desarrolladores y el personal encargado del mantenimiento deben tener conocimientos básicos en arquitectura de microservicios, REST, autenticación OAuth 2.0 y bases de datos relacionales (MySQL).

-Críticidad de la aplicación

El sistema es considerado crítico para la continuidad operativa de EcoMarket SPA, ya que involucra ventas, pagos, inventario y atención al cliente. Debe funcionar con alta disponibilidad.

-Consideraciones acerca de la seguridad

Se requiere autenticación robusta (OAuth 2.0 + JWT), cifrado de contraseñas, validación de entradas y prevención de accesos no autorizados. Se utilizarán servicios como Auth0, SendGrid para reforzar la seguridad.

2.5. Suposiciones y Dependencias

- Se asume que EcoMarket+ dispone de la infraestructura tecnológica mínima necesaria para implementar y mantener una arquitectura basada en microservicios, incluyendo servidores, redes y servicios de balanceo de carga.
- Se considera que el equipo técnico de la empresa será capacitado adecuadamente en tecnologías y gestión de microservicios.
- Se supone que, durante la fase de implementación, el sistema monolítico coexistirá temporalmente con la nueva arquitectura, para garantizar continuidad operacional y permitir una transición progresiva.
- El sistema propuesto dependerá del correcto funcionamiento del motor de base de datos MySQL, así como de una conexión estable y segura entre microservicios.

- La implementación exitosa del sistema depende de la disponibilidad del personal clave de EcoMarket SPA para realizar validaciones durante el proceso de desarrollo, pruebas y despliegue.

2.6. Requisitos Futuros

- Integración con plataformas de e-commerce y marketplaces externos como Mercado Libre, Amazon u otros para expandir la presencia digital de EcoMarket+.
- Incorporación de herramientas de analítica avanzada que permitan conocer mejor los hábitos de compra de los clientes, detectar tendencias y optimizar el inventario.
- Desarrollo de funcionalidades basadas en inteligencia artificial para ofrecer recomendaciones personalizadas a los clientes.
- Desarrollo de una aplicación móvil nativa (Android e iOS) que ofrezca una experiencia de usuario optimizada en dispositivos móviles.
- Creación de un sistema de fidelización que incluya puntos por compra, recompensas, descuentos y programas de membresía.

3. Requisitos Específicos

Esta sección define de manera estructurada los requisitos funcionales (RF) y no funcionales (RNF) del sistema EcoMarket+, siguiendo el estándar IEEE 830 para garantizar claridad, Trazabilidad y verificabilidad.

Principios clave aplicados:

Corrección:

Cada requisito refleja una necesidad real de EcoMarket SPA, relacionada directamente con una funcionalidad esperada del sistema.

No Ambiguos:

Se utiliza lenguaje técnico claro, evitando interpretaciones múltiples. Los términos clave se definen en el glosario, y para procesos complejos se apoyan con diagramas.

Completo:

Se han considerado todos los escenarios relevantes, incluyendo interacciones, flujos alternativos, errores y respuestas del sistema ante diferentes entradas.

Consistentes:

No existen contradicciones entre los requisitos definidos. Todos mantienen coherencia entre sí y con los objetivos generales del sistema.

Clasificados:

Los requisitos se agrupan según su tipo (funcionales y no funcionales) y pueden etiquetarse por prioridad (esenciales, opcionales, condicionales).

Verificables:

Cada requisito puede ser validado mediante pruebas, métricas u observación directa. Se evita el uso de términos subjetivos como “rápido”, “intuitivo” o “óptimo” sin cuantificación.

3.1 Requisitos comunes de los interfaces

A continuación, se describen los distintos tipos de interfaces que forman parte del sistema EcoMarket+, abarcando tanto la interacción con el usuario como las conexiones entre hardware, software y protocolos de comunicación.

3.1.1 Interfaces de usuario

Descripción:

La interfaz de usuario será desarrollada como una aplicación de página única (SPA), con diseño responsivo compatible con dispositivos móviles y de escritorio. Se prioriza una navegación intuitiva, tiempos de carga inferiores a 2 segundos y una experiencia visual coherente con la identidad corporativa de EcoMarket+.

Características clave:

- Diferenciación visual clara de roles de usuario (por ejemplo: administrador, cliente, soporte).
- Acceso a funcionalidades específicas según permisos asignados.
- Diseño centrado en la usabilidad y accesibilidad.

3.1.2 Interfaces de hardware

Requisitos:

Compatibilidad con dispositivos estándar (lectores de código de barras, impresoras fiscales).

Soporte para servidores en cloud con autoescalado.

Compatibilidad con dispositivos móviles.

3.1.3 Interfaces de software

El sistema EcoMarket+ se basa en una arquitectura de microservicios desacoplados que se comunican internamente mediante APIs REST, utilizando formato de intercambio de datos **JSON** y autenticación segura mediante **JWT**.

A continuación, se detallan las interfaces externas necesarias:

-Pasarela de pagos: WebPay (Transbank).

-Autenticación: Auth0 como proveedor externo de identidad.

El resto de los módulos del sistema (gestión de compras, inventario, seguimiento, soporte, catálogo, reportes y notificaciones) se comunican exclusivamente mediante APIs propias internas.

3.1.4 Interfaces de comunicación

Protocolos:

-HTTP/HTTPS: Todos los microservicios se comunican a través de solicitudes HTTP sobre protocolos seguros (HTTPS), utilizando URL REST definidos.

-RabbitMQ: Se emplea mensajería asíncrona para procesos que no requieren respuesta inmediata, como envío de notificaciones, alertas de stock o actualización logística (ver RF-012).

-JSON: Formato estándar para el intercambio de datos entre componentes internos y externos del sistema.

3.2 Requisitos funcionales

RF-001: Gestión de Usuarios - Crear, Actualizar, Desactivar y Eliminar Cuentas

Descripción: El sistema debe permitir al administrador crear, modificar, desactivar y eliminar cuentas de usuarios.

Historia de Usuario:

Como Administrador, quiero poder gestionar las cuentas de los usuarios, para mantener el control de acceso al sistema.

Criterio de Aceptación:

El administrador puede crear, modificar, desactivar y eliminar cuentas de usuarios en menos de 5 segundos por operación.

Los cambios se reflejan inmediatamente en la base de datos y en la interfaz de administración.

RF-002: Gestión de Usuarios - Asignar Permisos de Acceso

Descripción: El sistema debe permitir al administrador asignar y modificar permisos de acceso a diferentes módulos y funciones.

Historia de Usuario:

Como Administrador, quiero poder asignar permisos específicos, para que cada usuario tenga acceso solo a las funciones que necesita.

Criterio de Aceptación:

El administrador puede asignar y modificar permisos de acceso a diferentes módulos y funciones en menos de 5 segundos.

Los permisos se aplican inmediatamente y los usuarios solo pueden acceder a las funciones asignadas.

RF-003: Gestión de Inventario - Agregar, Actualizar y Eliminar Productos

Descripción: El sistema debe permitir al gerente de tienda agregar, modificar y eliminar productos del inventario.

Historia de Usuario:

Como Gerente de Tienda, quiero poder gestionar los productos del inventario, para mantener la información actualizada y precisa.

Criterio de Aceptación:

El gerente de tienda puede agregar, modificar y eliminar productos del inventario en menos de 3 segundos por operación.

RF-004: Gestión de Inventario - Ajustar Cantidades de Stock y Generar Reportes

Descripción: El sistema debe permitir al gerente de tienda ajustar las cantidades de stock y generar reportes de inventario.

Historia de Usuario:

Como Gerente de Tienda, quiero poder ajustar el stock y generar reportes, para tener un control preciso del inventario.

Criterio de Aceptación:

El gerente de tienda puede ajustar las cantidades de stock y generar reportes de inventario en menos de 5 segundos.

Los reportes se generan en formatos como Excel o PDF y contienen información precisa y actualizada.

RF-005: Ventas y Atención al Cliente - Registrar Ventas, Aplicar Descuentos y Generar Facturas

Descripción: El sistema debe permitir al empleado de ventas registrar ventas, aplicar descuentos automáticos o manuales, y generar facturas electrónicas con opciones de personalización, como agregar notas o detalles adicionales.

Historia de Usuario:

Como Empleado de Ventas, quiero poder registrar ventas, aplicar descuentos flexibles y generar facturas personalizadas de manera eficiente, para mejorar la experiencia del cliente y agilizar el proceso de compra.

Criterio de Aceptación:

El empleado de ventas puede registrar una venta, aplicar descuentos y generar una factura en menos de 10 segundos.

La factura se envía automáticamente por correo electrónico al cliente.

RF-006: Ventas y Atención al Cliente - Atender Devoluciones y Reclamaciones

Descripción: El sistema debe permitir al empleado de ventas gestionar devoluciones y reclamaciones de clientes.

Historia de Usuario:

Como Empleado de Ventas, quiero poder manejar devoluciones y quejas, para mejorar la satisfacción del cliente.

Criterio de Aceptación:

El empleado de ventas puede gestionar devoluciones y reclamaciones en menos de 5 minutos por caso.

El sistema registra el estado de la devolución o reclamación y notifica al cliente.

RF-007: Logística - Gestionar Envíos, Optimizar Rutas y Actualizar Estados de Pedidos

Descripción: El sistema debe permitir al equipo de logística gestionar envíos, optimizar rutas y actualizar el estado de los pedidos.

Historia de Usuario:

Como responsable de logística, quiero poder organizar envíos y rutas, para asegurar que los pedidos lleguen a tiempo.

Criterio de Aceptación:

El equipo de logística puede gestionar envíos, optimizar rutas y actualizar el estado de los pedidos en menos de 5 segundos por operación.

Los estados de los pedidos se actualizan en tiempo real.

RF-008: Logística - Gestionar Proveedores y Recepción de Mercancías

Descripción: El sistema debe permitir al equipo de logística gestionar proveedores y la recepción de mercancías.

Historia de Usuario:

Como responsable de Logística, quiero poder manejar proveedores y recibir productos, para mantener un flujo constante de stock.

Criterio de Aceptación:

El equipo de logística puede gestionar proveedores y recibir mercancías en menos de 5 minutos por operación.

El sistema registra la recepción de mercancías y actualiza el inventario.

RF-009: Interacción del Cliente - Registro, Inicio de Sesión y Gestión de Perfil

Descripción: El sistema debe permitir a los clientes registrarse, iniciar sesión y gestionar su perfil.

Historia de Usuario:

Como Cliente, quiero poder crear una cuenta y gestionar mi perfil, para tener una experiencia personalizada.

Criterio de Aceptación:

Los clientes pueden registrarse, iniciar sesión y gestionar su perfil en menos de 5 segundos por operación.

Los cambios en el perfil se reflejan inmediatamente en la base de datos.

RF-010: Interacción del Cliente - Navegación, Compra y Consulta de Historial de Pedidos

Descripción: El sistema debe permitir a los clientes navegar por el catálogo, realizar compras y consultar su historial de pedidos.

Historia de Usuario:

Como Cliente, quiero poder comprar y ver mis pedidos anteriores, para saber qué he comprado y cuándo.

Criterio de Aceptación:

Los clientes pueden navegar por el catálogo, realizar compras y consultar su historial de pedidos en menos de 3 segundos por operación.

El historial de pedidos muestra información precisa y actualizada.

RF-011: Búsqueda Avanzada de Productos

Descripción: El sistema debe permitir a los clientes buscar productos utilizando filtros.

Historia de Usuario:

Como Cliente, quiero poder buscar productos fácilmente usando filtros, para encontrar lo que necesito rápidamente.

Criterio de Aceptación:

Los clientes pueden buscar productos utilizando filtros avanzados en menos de 2 segundos.

Los resultados de búsqueda son precisos y relevantes.

RF-012: Notificaciones en Tiempo Real

Descripción: El sistema debe enviar notificaciones automáticas (correo electrónico, SMS) sobre el estado de los pedidos, promociones y actualizaciones.

Historia de Usuario:

Como Cliente, quiero recibir notificaciones sobre mis pedidos y ofertas, para estar siempre informado.

Criterio de Aceptación:

El sistema envía notificaciones automáticas (correo electrónico, SMS) sobre el estado de los pedidos, promociones y actualizaciones en menos de 10 segundos.

Las notificaciones llegan a los clientes de manera oportuna.

RF-013: Gestión de Reseñas y Calificaciones

Descripción: El sistema debe permitir a los clientes dejar reseñas y calificaciones de los productos.

Historia de Usuario:

Como Cliente, quiero poder dejar reseñas y calificaciones de los productos que compro, para compartir mi experiencia con otros compradores.

Criterio de Aceptación:

Los clientes pueden dejar reseñas y calificaciones de los productos en menos de 5 segundos.
Las reseñas y calificaciones se moderan y se muestran en el catálogo.

RF-014: Integración con Métodos de Pago

Descripción: El sistema debe soportar múltiples métodos de pago, como tarjetas de crédito, billeteras digitales y transferencias bancarias.

Historia de Usuario:

Como Cliente, quiero poder pagar con mi método de pago preferido, para tener una experiencia de compra más flexible.

Criterio de Aceptación:

El sistema soporta múltiples métodos de pago (tarjetas de crédito, billeteras digitales, transferencias bancarias) y procesa pagos en menos de 5 segundos.
Los pagos se registran correctamente en el sistema.

RF-015: Gestión de Cupones y Descuentos

Descripción: El sistema debe permitir a los clientes aplicar cupones y descuentos durante el proceso de compra.

Historia de Usuario:

Como Cliente, quiero poder aplicar cupones y descuentos en mis compras, para ahorrar dinero.

Criterio de Aceptación:

Los clientes pueden aplicar cupones y descuentos durante el proceso de compra en menos de 3 segundos.
Los descuentos se aplican correctamente al total de la compra.

RF-016: Soporte al Cliente en Línea

Descripción: El sistema debe ofrecer un chat en línea con integración de respuestas automáticas y un formulario de contacto, permitiendo que los clientes puedan recibir asistencia inmediata o en su defecto, ser atendidos dentro de un tiempo estimado.

Historia de Usuario:

Como Cliente, quiero poder contactar al soporte rápidamente a través de un chat en línea o formulario, con la posibilidad de recibir respuestas automáticas inmediatas o seguimiento de mi solicitud para resolver mis problemas o dudas eficientemente.

Criterio de Aceptación:

Los clientes pueden contactar al soporte a través de un chat en línea o un formulario de contacto en menos de 10 segundos.

Las consultas se registran y se responden en un tiempo máximo de 24 horas.

RF-017: Gestión de Pedidos Recurrentes

Descripción: El sistema debe permitir a los clientes configurar pedidos recurrentes para productos que compran regularmente.

Historia de Usuario:

Como Cliente, quiero poder programar pedidos recurrentes, para no tener que hacer pedidos manualmente cada vez.

Criterio de Aceptación:

Los clientes pueden configurar pedidos recurrentes para productos que compran regularmente en menos de 5 segundos.

Los pedidos recurrentes se procesan automáticamente en las fechas programadas.

RF-018: Gestión de Múltiples Direcciones de Envío

Descripción: El sistema debe permitir a los clientes guardar y gestionar múltiples direcciones de envío.

Historia de Usuario:

Como Cliente, quiero poder guardar varias direcciones de envío, para elegir fácilmente a dónde enviar mis pedidos.

Criterio de Aceptación:

Los clientes pueden guardar y gestionar múltiples direcciones de envío en menos de 5 segundos por operación.

Las direcciones se seleccionan correctamente durante el proceso de compra.

RF-019: Exportación de Datos para Reportes

Descripción: El sistema debe permitir a los gerentes de tienda exportar datos de inventario, ventas y otros reportes en formatos como Excel o PDF.

Historia de Usuario:

Como Gerente de Tienda, quiero poder exportar reportes en diferentes formatos, para analizar los datos fuera del sistema.

Criterio de Aceptación:

Los gerentes de tienda pueden exportar datos de inventario, ventas y otros reportes en formatos como Excel o PDF en menos de 10 segundos.

Los reportes contienen información precisa y actualizada.

3.3 Requisitos no funcionales

1. Seguridad

RNF-SEG-001:

- **Descripción:** Autenticación multifactorial obligatoria para operaciones administrativas críticas.
- **Vinculado a:** RF-001, RF-002.

RNF-SEG-002:

- **Descripción:** Cifrado avanzado para datos sensibles (ej: información de pago).
- **Vinculado a:** RF-005, RF-014.

RNF-SEG-003:

- **Descripción:** Registro detallado de auditoría para cambios en permisos y datos críticos.
- **Vinculado a:** RF-002, RF-008.

2. Rendimiento

RNF-Ren-001:

- **Descripción:** Operaciones de gestión de usuarios en 3 segundos.
- **Vinculado a:** RF-001.

RNF-Ren-002:

- **Descripción:** Procesamiento de pagos en menos de 5 segundos.
- **Vinculado a:** RF-005, RF-014.

RNF-Ren-003:

- **Descripción:** Búsqueda de productos con filtros en 3 segundos.
- **Vinculado a:** RF-011.

3. Disponibilidad

RNF-DISP-001:

- **Descripción:** 99.95% disponibilidad para módulo de pagos.
- **Vinculado a:** RF-014.

RNF-DISP-002:

- **Descripción:** 99.9% disponibilidad para gestión de inventario.
- **Vinculado a:** RF-003, RF-004.

4. Escalabilidad

RNF-ESC-001:

- **Descripción:** Soporte para 10K usuarios concurrentes.
- **Vinculado a:** RF-010, RF-016.

RNF-ESC-002:

- **Descripción:** Manejo de +1M productos en inventario.
- **Vinculado a:** RF-003, RF-011.

5. Usabilidad

RNF-USAB-001:

- **Descripción:** Interfaz intuitiva para gestión de permisos.
- **Vinculado a:** RF-002.

RNF-USAB-002:

- **Descripción:** procesar compra en 4 pasos.
- **Vinculado a:** RF-005, RF-015.

6. Cumplimiento Legal

RNF-LEG-001:

- **Descripción:** Protección de datos según Ley 19.628 (Chile).
- **Vinculado a:** RF-009, RF-018.

3.3.1 Requisitos de rendimiento

El sistema EcoMarket+ debe mantener un desempeño eficiente bajo distintas condiciones de carga, asegurando tiempos de respuesta adecuados y estabilidad para los usuarios finales y el personal administrativo.

A continuación, se detallan los principales requisitos de rendimiento:

- El sistema deberá soportar al menos 100 usuarios conectados simultáneamente sin degradar la experiencia de navegación.
- El 95% de las solicitudes HTTP (GET, POST, PUT, DELETE) deberá recibir respuesta en menos de 1 segundo.
- Las operaciones críticas, como el pago en línea o la confirmación de pedidos, deberán completarse en un tiempo máximo de 2 segundos en condiciones normales de red.
- El sistema deberá ser capaz de procesar un mínimo de 50 transacciones por segundo en horas punta, considerando consultas de catálogo, actualizaciones de stock, creación de pedidos y envío de notificaciones.
- Las interfaces administrativas (como reportes o gestión de usuarios) deberán responder en un tiempo inferior a 3 segundos para mantener la fluidez operativa.
- Los microservicios deberán mantenerse estables ante un crecimiento de carga progresiva de hasta un 200% del tráfico promedio mediante escalamiento horizontal.

3.3.2 Seguridad

El sistema EcoMarket+ deberá garantizar altos niveles de seguridad en el acceso, transmisión y almacenamiento de la información, para proteger los datos de clientes, pagos y operaciones internas.

Se considerarán las siguientes medidas:

- Empleo de técnicas criptográficas como hashing de contraseñas (ej. bcrypt) y cifrado de tokens JWT.
- Registro de logs de actividad para trazabilidad y auditoría.
- Asignación de funcionalidades específicas según el rol del usuario (admin, cliente, repartidor, etc.).
- Restricción de acceso entre módulos sensibles mediante políticas de control de acceso.
- Validaciones de integridad para operaciones críticas como pagos y gestión de inventario.
- Uso de autenticación externa segura con Auth0.

3.3.3 Fiabilidad

EcoMarket+ deberá ser confiable en su operación continua. Se espera que el sistema tenga una alta tolerancia a fallos, especialmente durante procesos críticos como compras, pagos y atención al cliente. Los errores deben recuperarse sin pérdida de datos significativos.

3.3.4 Disponibilidad

El sistema deberá mantener una disponibilidad mínima del 99% mensual, considerando su uso activo en múltiples sucursales y canales digitales. Se deben implementar mecanismos de redundancia y autoscalado para asegurar que el sistema funcione incluso en horarios de alta demanda o durante fallas parciales.

3.3.5 Mantenibilidad

El sistema debe ser fácil de mantener por desarrolladores y personal técnico. Se deberá:

- Permitir actualizaciones por microservicio sin afectar el sistema completo.
- Generar reportes de uso y errores automáticos semanalmente.
- Documentar todas las APIs y flujos internos para facilitar el soporte.
- Permitir tareas de mantenimiento programadas sin interrumpir el servicio al cliente.
- Definir responsables para la ejecución de mantenimientos periódicos (ej. soporte TI de la empresa o proveedor externo).

3.3.6 Portabilidad

EcoMarket+ debe estar diseñado para ser portable entre entornos de desarrollo y producción.

Para ello:

- El sistema deberá contener componentes desacoplados y contenerizados (ej. Docker).
- El código debe evitar dependencias rígidas con el servidor o sistema operativo.
- Se emplearán tecnologías multiplataforma como Node.js y Spring Boot.

- El sistema debe ser implementable tanto en infraestructura propia como en servicios cloud (ej. AWS, Azure).
- Las bases de datos deben ser fácilmente migrables mediante scripts o herramientas de respaldo compatibles (ej. MySQL dump).

4. Diseño de la Nueva Arquitectura

El sistema EcoMarket+ ha sido reestructurado utilizando una arquitectura basada en microservicios (REST), con el objetivo de reemplazar el sistema monolítico actual por una solución más modular, escalable y mantenible. Cada microservicio es autónomo, cuenta con su propia base de datos y expone una interfaz web basada en REST que permite su comunicación con los demás servicios.

Además, se incluirá un API Gateway para gestionar el tráfico entre el frontend y los microservicios, aplicando control de acceso, redireccionamiento y seguridad de forma centralizada.

Esta arquitectura favorece el desarrollo distribuido, el despliegue independiente y la evolución progresiva del sistema, reduciendo riesgos y mejorando la experiencia tanto de usuarios como de administradores.

Microservicios propuestos:

Microservicio de Usuarios

Gestiona el registro, login, recuperación de contraseña, asignación de roles y autenticación con JWT.

Microservicio de Inventario

Controla los productos, stock en tiempo real, alertas por bajo inventario y generación de reportes.

Microservicio de Sucursales

Permite registrar y administrar información sobre cada sucursal física, su inventario y personal asociado.

Microservicio de Compras

Se encarga de crear y consultar pedidos, ver historial de compras y cancelar operaciones.

Microservicio de Pagos

Integra WebPay como pasarela externa para gestionar pagos y confirmar transacciones.

Microservicio de Reportes

Genera reportes automáticos (PDF/Excel) para usuarios autorizados, filtrables por fecha, zona o sucursal.

Microservicio de Seguimiento

Controla el estado de los pedidos (en preparación, en ruta, entregado), brindando trazabilidad.

Microservicio de Soporte

Administra solicitudes de atención al cliente: consultas, reclamos, sugerencias y sus respuestas.

Microservicio de Notificaciones

Envía alertas al cliente o personal mediante correos y notificaciones internas (integración con SendGrid).

Microservicio de Catálogo

Gestiona el catálogo público: búsqueda avanzada, filtros, promociones y visualización de productos.

Tecnologías de soporte:

API Gateway para centralizar el acceso, enrutar solicitudes y aplicar seguridad en todos los microservicios.

RabbitMQ para mensajería entre microservicios.

JWT + Auth0 para seguridad y control de acceso.

MySQL como base de datos distribuida por microservicio.

Spring Boot y **Node.js** para el backend según el servicio.

5. Planificación de la Migración

La migración desde el sistema monolítico actual hacia una arquitectura basada en microservicios requiere una planificación cuidadosa para asegurar la continuidad operativa del negocio, minimizar riesgos y garantizar una transición progresiva y controlada.

5.1 Estrategia de Migración

Se propone una migración incremental y por dominios funcionales, es decir, trasladar progresivamente las funcionalidades del sistema actual hacia nuevos microservicios, respetando los límites del negocio y asegurando compatibilidad temporal entre ambos entornos (monolítico y microservicios).

Fases propuestas:**Fase 1 – Preparación del entorno**

- Levantar infraestructura básica para los microservicios (bases de datos, servidor backend, APIs Gateway, Docker).
- Definir estándar de comunicación entre microservicios

Fase 2 – Migración de servicios independientes

- Desarrollar y poner en marcha microservicios que no dependan fuertemente de otros, como:
 - Usuarios, sucursales, catálogo y reportes.

Fase 3 – Migración de procesos clave

- Migrar progresivamente funcionalidades críticas, garantizando compatibilidad temporal:
 - Inventario, compras, pagos (con integración a WebPay) y seguimiento de pedidos.

Fase 4 – Soporte, Notificaciones y Logística

- Reemplazar las herramientas externas de atención al cliente con microservicios propios.
- Conectar notificaciones automáticas entre servicios usando RabbitMQ.

Fase 5 – Desactivación del sistema monolítico

- Una vez que todos los procesos estén cubiertos por microservicios y hayan sido verificados, se procede a desactivar gradualmente los módulos del sistema monolítico original.

5.2 Riesgos identificados y plan de mitigación

Riesgo	Descripción	Mitigación
Interrupción del servicio	Corte o mal funcionamiento durante la transición	Migración por fases, pruebas en entornos controlados
Pérdida de datos	Incompatibilidad en el traspaso de información	Backups completos y pruebas de importación de datos
Errores de comunicación entre microservicios	Mal diseño de APIs o falta de sincronización	Definir contratos claros, aplicar pruebas.
Problemas de rendimiento inicial	Microservicios no optimizados en primeras versiones	Pruebas de carga y monitoreo constante
Falta de formación del equipo	Nuevas tecnologías y estructura desconocida	Capacitación y documentación técnica para los equipos
Sobrecoste por duplicación temporal	Costos de mantener monolito y microservicios a la vez	Plazos claros para el retiro del monolito y planificación de recursos

6. Microservicios

Como parte del objetivo de migración desde una arquitectura monolítica a una arquitectura moderna y escalable, EcoMarket+ será implementado utilizando un enfoque basado en microservicios. Esta arquitectura permite independencia entre módulos funcionales, escalabilidad horizontal, despliegue continuo y mayor mantenibilidad.

A continuación, se detallan los principales microservicios identificados para el sistema:

6.1 Microservicio Usuario.

Este microservicio se encarga de gestionar el ciclo de vida de los usuarios dentro del sistema EcoMarket+, incluyendo su autenticación, permisos y seguridad. Todo se gestiona mediante una API REST.

Las funcionalidades principales incluyen:

- Registrar nuevos usuarios.
- Iniciar sesión con autenticación segura.
- Recuperar contraseñas mediante verificación interna (email o código TOTP).
- Actualizar y eliminar cuentas de usuario.
- Asignar roles y permisos a cada tipo de usuario.
- Validar sesiones mediante tokens JWT.
- Activar autenticación multifactor (MFA) para usuarios con acceso crítico.

USUARIO API:

El microservicio expone sus funcionalidades a través de una interfaz RESTful, con endpoints (urls) como:

POST /api/usuarios/register	(Registro de nuevos usuarios)
POST /api/usuarios/login	(Inicio de sesión y generación de JWT)
PUT /api/usuarios/:id	(Actualización de datos del usuario)
DELETE /api/usuarios/:id	(Eliminación de cuenta de usuario)
POST /api/usuarios/recover	(Solicitud de recuperación de contraseña)
POST /api/usuarios/mfa/verify	(Verificación de código TOTP para MFA)
POST /api/usuarios/roles	(Asignación o modificación de roles)
GET /api/usuarios/:id/permisos	(Consulta de permisos asignados)

Este microservicio cuenta con una base de datos propia en MySQL, denominada “**usuario**”, la cual gestiona la información.

Herramientas sugeridas:

- JWT**: Para generación y validación de tokens de sesión seguros.
- bcrypt**: Para hash de contraseñas en base de datos.
- TOTP (Time-based One-Time Password)**: Para MFA con códigos temporales.

6.2 Microservicio Inventario.

Microservicio que entrega rutas REST para manejar productos, stock y consultas. Algunas de estas rutas son:

Las funcionalidades principales incluyen:

- Agregar, modificar y eliminar productos del inventario.
- Ajustar cantidades de stock en tiempo real.
- Generar alertas automáticas por quiebre de stock.
- Exportar reportes de inventario en formatos PDF o Excel.

INVENTARIO API:

Microservicio que expone endpoints RESTful para el manejo completo de productos, stock y consultas. Algunos endpoints son:

GET /api/inventario	(Listar todos los productos)
POST /api/inventario	(Agregar nuevo producto)
PUT /api/inventario/:id	(Editar producto existente)
DELETE /api/inventario/:id	(Eliminar producto del inventario)
PATCH /api/inventario/:id/stock	(Ajustar stock de un producto)
GET /api/inventario/alertas	(Consultar productos con stock bajo)
GET /api/inventario/reporte/pdf	(Generar reporte PDF del inventario)
GET /api/inventario/reporte/excel	(Generar reporte Excel del inventario)

Este microservicio cuenta con una base de datos propia en MySQL, denominada **inventario**.

Herramientas sugeridas:

Spring Boot: Para construir la API REST con rapidez y buenas prácticas.

6.3. Microservicio Sucursales.

Este microservicio se encarga de gestionar la información de las distintas sucursales físicas de EcoMarket SPA.

Las funcionalidades principales incluyen:

- Registrar, editar y eliminar sucursales.
- Consultar datos de cada sucursal (ubicación, estado, zona de cobertura).
- Asociar empleados y productos a cada sucursal.
- Obtener el inventario y desempeño por sucursal.
- Filtrar datos por ubicación para reportes y logística.

APIs REST (propias):

El microservicio expone sus funcionalidades a través de una interfaz RESTful, con endpoints (url) como:

GET /api/sucursales	(Listar todas las sucursales)
---------------------	-------------------------------

POST	/api/sucursales	(Crear una nueva sucursal)
PUT	/api/sucursales/:id	(Editar información de una sucursal)
DELETE	/api/sucursales/:id	(Eliminar una sucursal)
GET	/api/sucursales/:id/inventario	(Consultar inventario de una sucursal)
GET	/api/sucursales/:id/empleados	(Consultar empleados de una sucursal)

Herramientas sugeridas:**Google Maps API / OpenStreetMap (opcional):**

Servicio externo para obtener y visualizar la ubicación geográfica de cada sucursal.

Este microservicio cuenta con una base de datos propia en MySQL, denominada **sucursal**, la cual gestiona la información asociada a las tiendas físicas de la empresa.

6.4. Microservicio Compras.

Este microservicio se encarga de gestionar todo el flujo de compras realizadas por los clientes dentro de EcoMarket+, desde la creación de pedidos hasta su consulta y cancelación.

Las funcionalidades principales incluyen:

- Crear un nuevo pedido a partir del carrito del cliente.
- Consultar el historial de compras de cada cliente.
- Ver el detalle de un pedido específico.
- Cancelar pedidos en caso de errores o devoluciones.
- Asociar el estado de pago y logística a cada pedido.

COMPRAS API:

Microservicio que expone rutas RESTful para gestionar pedidos y compras. Algunos endpoints disponibles son:

GET	/api/compras	(Listar todos los pedidos)
POST	/api/compras	(Crear un nuevo pedido)
GET	/api/compras/:id	(Consultar detalle de un pedido)
GET	/api/compras/cliente/:idCliente	(Consultar historial de compras de un cliente)
PUT	/api/compras/:id/cancelar	(Cancelar un pedido)
PATCH	/api/compras/:id/estado	(Actualizar estado de un pedido: pendiente, pagado, enviado, entregado)

Este microservicio puede tener una base de datos local en MySQL denominada **compras** para almacenar pedidos.

Herramientas sugeridas:

-**Spring Boot**: Para implementar la API REST con seguridad y eficiencia.

-**JWT**: Para verificar la autenticación del cliente antes de crear o consultar pedidos.

6.5. Microservicio Pago.

Este microservicio se encarga de gestionar el proceso de pago de las compras realizadas en EcoMarket+, integrándose con la pasarela de pagos WebPay de Transbank (Chile). Es responsable de validar transacciones, actualizar el estado de los pedidos y generar confirmaciones para el cliente.

Las funcionalidades principales incluyen:

- Iniciar una transacción de pago con WebPay.
- Confirmar si un pago fue exitoso o rechazado.
- Asociar el pago a la compra correspondiente.
- Consultar pagos por cliente o por compra.
- Notificar a otros microservicios (compras, logística, notificaciones) al completar el pago.

PAGO API (con WebPay):

POST	/api/pagos/iniciar	(Iniciar transacción con WebPay)
GET	/api/pagos/confirmar/:token	(Confirmar transacción desde WebPay)
GET	/api/pagos	(Listar todos los pagos)
GET	/api/pagos/:id	(Consultar detalles de un pago)
GET	/api/pagos/cliente/:idCliente	(Consultar pagos por cliente)
GET	/api/pagos/compra/:idCompra	(Consultar pago asociado a una compra)

Tecnologías asociadas:

WebPay Plus REST API de Transbank: Para iniciar, confirmar y anular pagos.

JWT: Validación de identidad antes de permitir pagos.

Este microservicio utiliza una base de datos propia en MySQL, denominada **pagos**.

6.6 Microservicio Reportes

Este microservicio se encarga de generar reportes automáticos y personalizados para los administradores y gerentes de EcoMarket+, usando los datos internos del sistema (ventas, inventario, usuarios, sucursales, etc.)

Funcionalidades principales:

- Generar reportes en formato PDF, Excel o CSV.
- Consultar reportes filtrando por fecha, zona o sucursal.
- Exportar reportes de ventas, inventario, clientes o productos.
- Programar reportes automáticos.
- Visualizar datos en formato de tabla o gráfico (opcional, para interfaz futura).

REPORTES API:

GET	/api/reportes/ventas	(Generar reporte de ventas)
GET	/api/reportes/inventario	(Generar reporte de inventario)

GET /api/reportes/usuarios	(Generar reporte de usuarios)
GET /api/reportes/sucursales	(Generar reporte por sucursal)
GET /api/reportes/personalizados	(Generar reporte filtrado por fechas y zona)
GET /api/reportes/:tipo/pdf	(Exportar reporte en PDF)
GET /api/reportes/:tipo/excel	(Exportar reporte en Excel)
POST /api/reportes/programados	(Programar envío automático de reportes)

Herramientas sugeridas:

Spring Boot: Para manejar las rutas de generación de reportes.

Apache POI Para generar reportes en formato Excel.

iText: Para generar archivos PDF de manera programada o bajo demanda.

Quartz : Para programación automática de reportes.

Este microservicio utiliza una base de datos propia en MySQL, denominada **pagos**.

6.7 Microservicio Seguimiento

Este microservicio permite rastrear el estado de los pedidos una vez que han sido confirmados y despachados, proporcionando visibilidad tanto para el cliente como para el equipo de logística.

Funcionalidades principales:

- Registrar el estado de un pedido (pendiente, en bodega, en ruta, entregado, fallido).
- Consultar la ubicación y estado actual de un pedido.
- Asociar información de seguimiento con una compra específica.
- Notificar cambios de estado a los microservicios de notificaciones y soporte.
- Cargar eventos de seguimiento manualmente por parte de los encargados de logística.

SEGUIMIENTO API:

GET /api/seguimiento	(Listar todos los seguimientos)
GET /api/seguimiento/:id	(Ver detalle del seguimiento por ID)
GET /api/seguimiento/compra/:idCompra	(Ver seguimiento de una compra específica)
PATCH /api/seguimiento/:id/estado	(Actualizar estado del pedido)
POST /api/seguimiento	(Crear seguimiento para un nuevo pedido)
DELETE /api/seguimiento/:id	(Eliminar seguimiento si el pedido fue cancelado)

Este microservicio cuenta con una base de datos propia en MySQL, denominada **seguimiento**.

6.8 Microservicio Soporte

Este microservicio se encarga de gestionar las solicitudes de ayuda, preguntas, reclamos o sugerencias que los clientes puedan tener al usar EcoMarket+.

Funcionalidades principales:

- Enviar solicitudes de soporte desde la plataforma.

- Clasificar las solicitudes por tipo: consulta, reclamo, sugerencia, etc.
- Ver el historial de solicitudes por usuario.
- Responder y cambiar el estado de cada solicitud (abierto, en progreso, resuelto, cerrado).
- Notificar al cliente cuando su solicitud es atendida.

SOPORTE API:

POST /api/soporte	(Enviar nueva solicitud de soporte)
GET /api/soporte	(Listar todas las solicitudes)
GET /api/soporte/:id	(Ver detalle de una solicitud)
GET /api/soporte/usuario/:idUsuario	(Ver historial de solicitudes de un usuario)
PATCH /api/soporte/:id/estado	(Actualizar estado de la solicitud)
POST /api/soporte/:id/respuesta	(Responder a una solicitud)
DELETE /api/soporte/:id	(Eliminar una solicitud si es inválida)

Este microservicio cuenta con una base de datos propia en MySQL, denominada **soporte**.

6.9 Microservicio Notificaciones.

Este microservicio se encarga de enviar mensajes automáticos a clientes y empleados del sistema, notificando sobre eventos importantes como confirmaciones de pedido, actualizaciones de estado, alertas de stock o respuestas del soporte.

Funcionalidades principales:

- Enviar notificaciones por correo electrónico.
- Enviar alertas internas a través del sistema (tipo bandeja de entrada).
- Registrar el historial de notificaciones enviadas.
- Permitir al usuario marcar notificaciones como leídas.

NOTIFICACIONES API:

POST /api/notificaciones	(Crear y enviar una notificación)
GET /api/notificaciones	(Listar todas las notificaciones)
GET /api/notificaciones/usuario/:idUsuario	(Ver notificaciones de un usuario)
PATCH /api/notificaciones/:id/leida	(Marcar una notificación como leída)
DELETE /api/notificaciones/:id	(Eliminar una notificación)

Tecnologías sugeridas:

RabbitMQ: para recibir eventos desde otros microservicios como Compras, Seguimiento o Soporte

Logger: para trazabilidad de mensajes enviados

Este microservicio cuenta con una base de datos propia en MySQL, denominada **notificaciones**.

6.10 Microservicio Catálogo

Este microservicio se encarga de gestionar y mostrar el catálogo de productos disponibles en EcoMarket+, permitiendo búsquedas, filtros y visualización detallada, todo desde datos internos del sistema.

Funcionalidades principales:

- Listar productos disponibles.
- Buscar productos por nombre, categoría o etiquetas.
- Filtrar por precio, stock o ubicación (sucursal).
- Consultar detalle completo de un producto.
- Mostrar productos destacados o en promoción.

CATÁLOGO API:

GET /api/catalogo	(Listar todos los productos del catálogo)
GET /api/catalogo/:id	(Ver detalle de un producto)
GET /api/catalogo/buscar	(Buscar productos por nombre o palabra clave)
GET /api/catalogo/filtro	(Filtrar productos por precio, stock, sucursal, etc.)
GET /api/catalogo/promociones	(Listar productos en oferta o destacados)
GET /api/catalogo/categorias	(Listar todas las categorías disponibles)

Tecnologías sugeridas:

Cloud Storage o CDN (opcional) para servir imágenes de productos

Este microservicio cuenta con su propia base en MySQL, llamada **catálogo**.

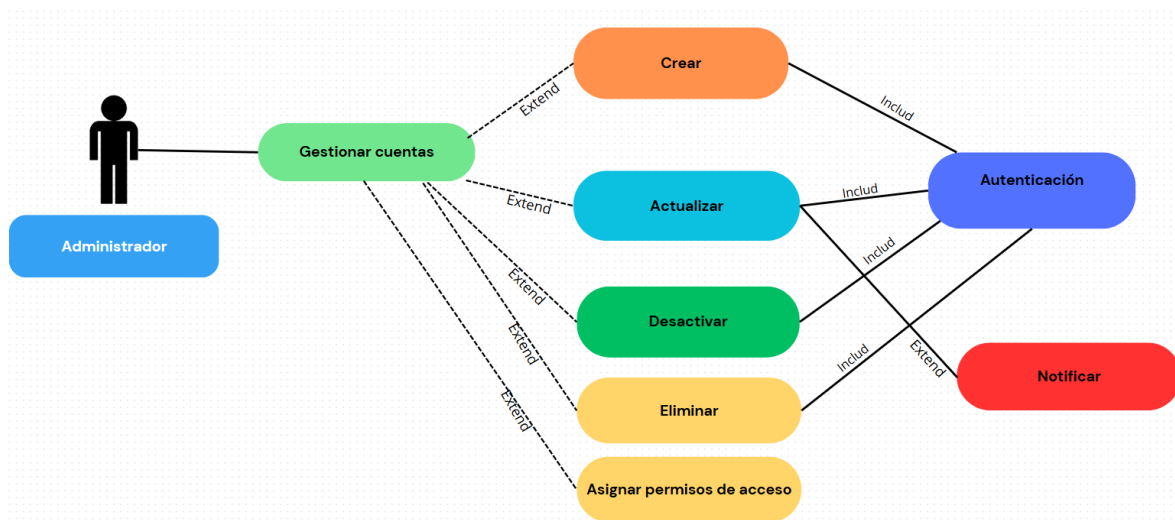
7. Diagramas UML

La migración de EcoMarket+ a una arquitectura basada en microservicios requiere un modelado claro de sus componentes, interacciones y flujos de datos. Los diagramas UML (Unified Modeling Language) son esenciales para visualizar:

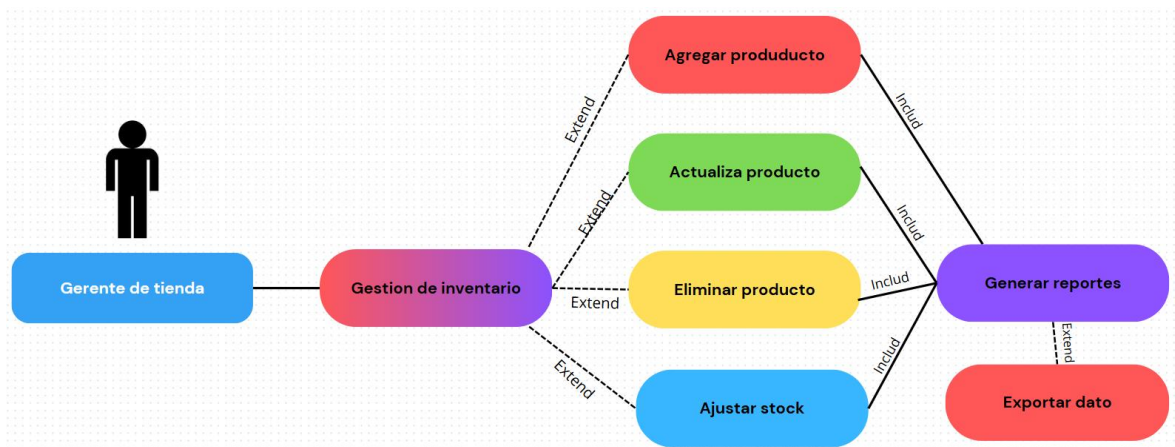
- Estructura del Sistema.
- Comportamiento y procesos
- Despliegue infraestructura.

7.1 Diagramas de Caso de uso por Actor.

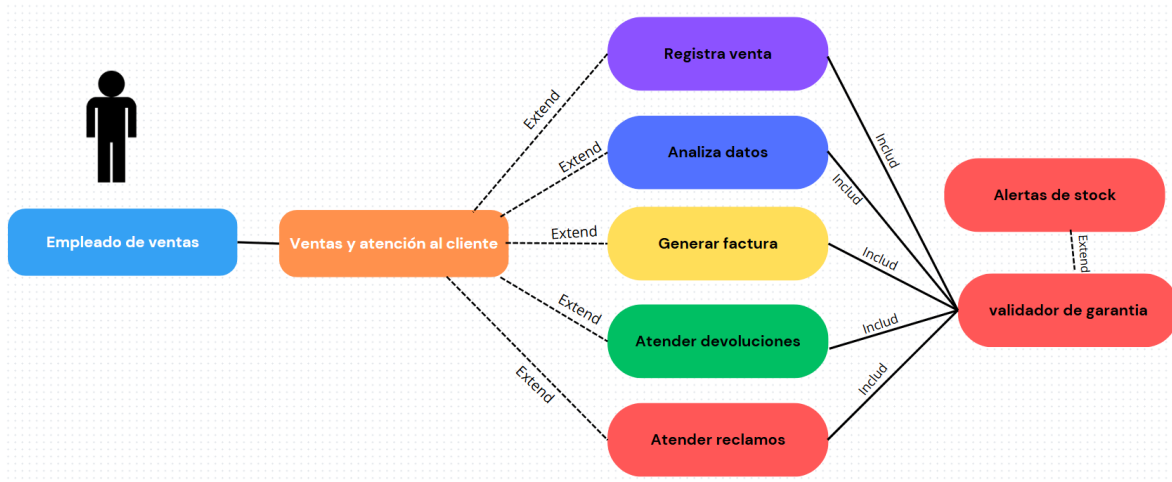
7.1.1 Administrador.



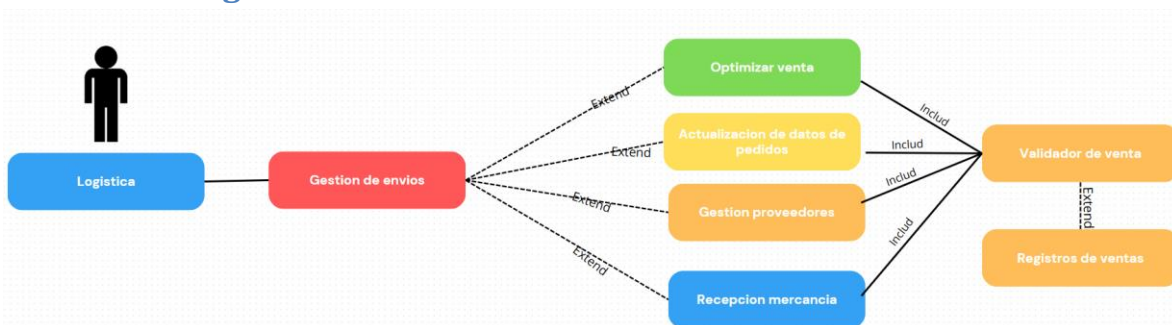
7.1.2 Gerente de Tienda.



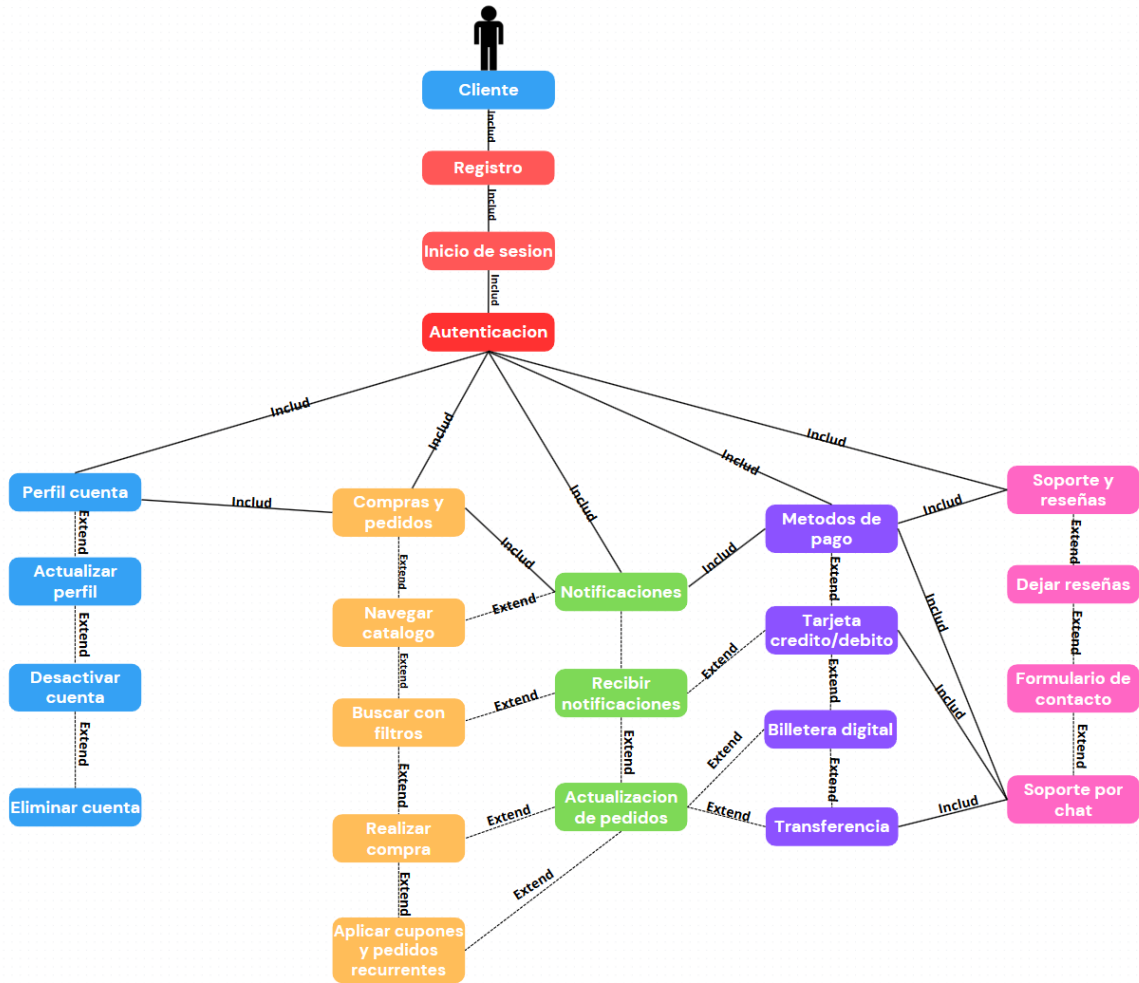
7.1.3 Empleado de Ventas.



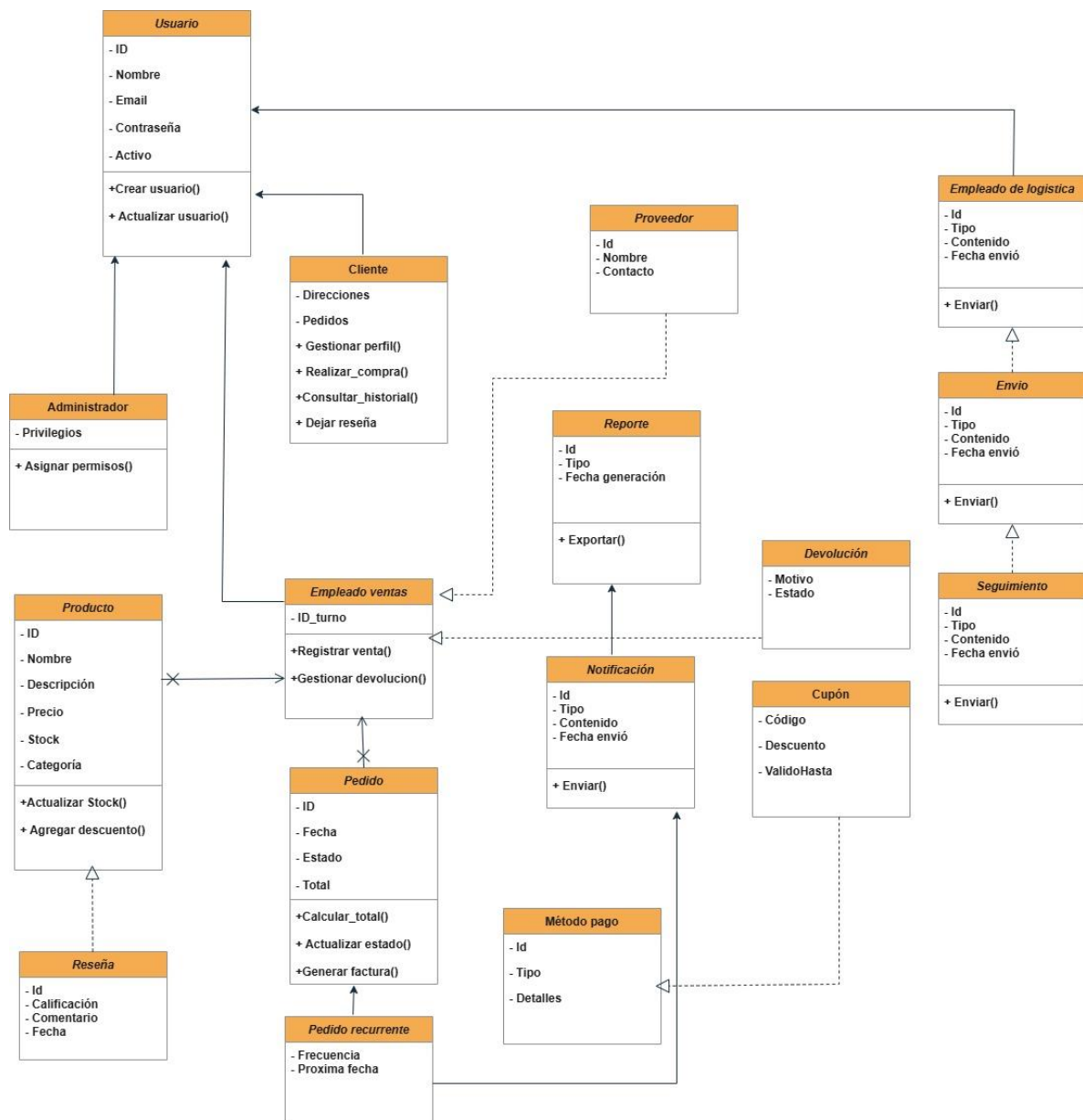
7.1.4 Logística.



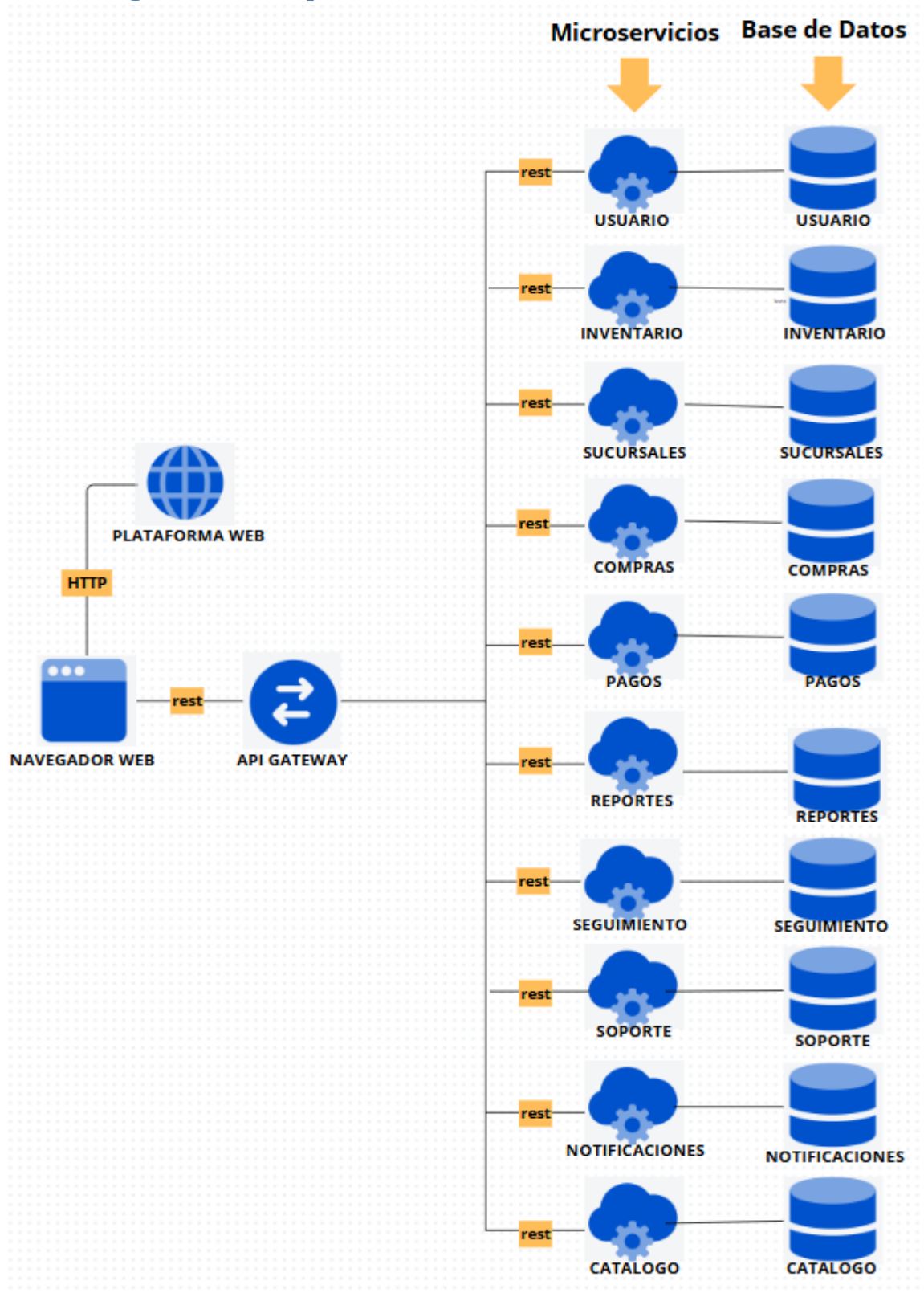
7.1.5 Cliente



7.2 Diagrama de Clases.



7.3 Diagramas de Arquitectura de Microservicios.



8. Aspectos Éticos.

El desarrollo de EcoMarket+ incorpora principios éticos clave para garantizar el uso responsable del sistema, la protección de sus usuarios y el cumplimiento de buenas prácticas tecnológicas.

Privacidad y Seguridad:

EcoMarket+ protege la información de los usuarios mediante el uso de autenticación segura con OAuth 2.0 (Auth0), cifrado de contraseñas, y control de acceso según roles. Se registra la actividad crítica (logs) para prevenir accesos no autorizados y garantizar la trazabilidad.

Inclusión y Accesibilidad:

La interfaz está diseñada para adaptarse a distintos dispositivos (móviles y de escritorio) con navegación intuitiva, permitiendo el acceso fácil a usuarios de distintas edades y niveles técnicos, incluidos trabajadores de sucursales o usuarios con conexión limitada.

Transparencia y Responsabilidad:

El sistema informa al usuario sobre cada acción relevante (registro, compra, pago, soporte). Las operaciones quedan registradas y auditables, y se respetan los principios de claridad y responsabilidad en el uso de datos y servicios integrados.

9. Anexos.

9.1 Fuentes y url Consultadas.

A continuación, se presentan las principales fuentes, herramientas y APIs utilizadas en el desarrollo del sistema EcoMarket+, especialmente aquellas relacionadas con microservicios, seguridad, comunicación y respaldo del desarrollo colaborativo.

Categoría	Recurso / Herramienta	URL de Referencia
Autenticación y Seguridad	Auth0 + OAuth 2.0	https://auth0.com
	JWT.io (tokens seguros)	https://jwt.io
Notificaciones	SendGrid API (correo electrónico)	https://sendgrid.com
	Twilio API (SMS / MFA)	https://www.twilio.com
Pagos	WebPay Plus REST API (Transbank Chile)	https://transbankdevelopers.cl
Desarrollo colaborativo	GitHub (repositorio de código)	https://github.com
Mensajería interna	RabbitMQ (mensajería entre microservicios)	https://www.rabbitmq.com
Base de datos	MySQL (gestión de datos relacional)	https://dev.mysql.com
Programación backend	Spring Boot / Node.js (Microservicios)	https://spring.io / https://nodejs.org
Generación de documentos	Apache POI, iText (Excel y PDF)	https://poi.apache.org/ / https://itextpdf.com
Diagramación	Herramienta UML online	https://diagramasuml.com
Planificación	Trello (gestión de tareas)	https://trello.com

Categoría	Recurso / Herramienta	URL de Referencia
Inteligencia Artificial	ChatGPT / DeepSeek	https://chat.openai.com/ / https://deepseek.com