

Eco Market

Integrantes: Miguel Laguela

Rodrigo Valdes

Matias Torres

Asignatura: Full Stack 1

Fecha entrega: 06/04/2025

**1. Análisis de Requerimientos:**

Requisitos funcionales:

1. Administrador del Sistema:

Gestión de usuario

Ajuste de permisos

Monitorear y gestión del sistema

Respaldo y restauración de datos

1. Gerente de Tienda:

Administración de inventario

Crear reportes (ventas, inventario, rendimiento de tienda)

Administración de tiendas (horario, personal asignado, políticas de la empresa

Administración de pedidos (supervisar y autorizar pedidos de inventario)

1. Empleado de Ventas:

Registro ventas (realizar ventas, actualizar ofertas)

Atención al cliente (devolución y reclamaciones)

Revisión de inventario (disponibilidad productos)

Generación de facturas (emitir y enviar facturas)

1. Logística:

Gestión Envios (crear, actualizar y seguir envíos de productos)

Optimización rutas envío (planificar rutas de entrega)

Actualizar estado de pedidos

Gestión de Proveedores (actualizar información proveedores y pedidos de mercancía)

1. Clientes:

Login (crear, iniciar sesion y actualizar clave)

Navegación del sitio web

Realización de compras y modificaciones

Compra de productos

Pago vía web

Consultar historial de pedidos

Gestión de perfil

Soporte del sitio

Reseña y calificación

Aplicación cupones y descuentos

Requisitos No Funcionales:

1. Seguridad: El proceso creado por la empresa debe asegurar la seguridad del cliente en la realización de su compra
2. Disponibilidad: La tienda debe estar disponible las 24 horas del día
3. Escalabilidad: crecimiento del servidor sin comprometer su rendimiento
4. Rendimiento:que soporte el aumento de cliente, transacciones y pagos.

**2. Análisis del Sistema Actual:**

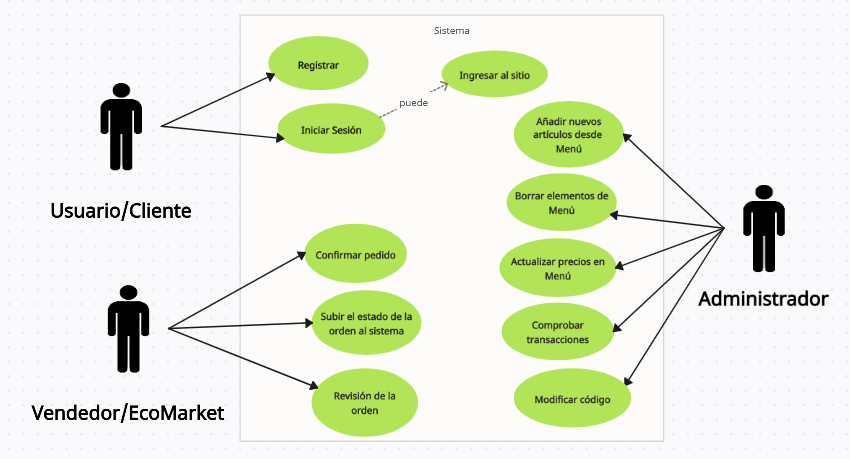
La tienda EcoMarket Spa está asociada ya a un sistema monolítico es decir pertenece todo a un solo código, con todo nos referimos a gestión de usuario, tienda, ventas, etc.

Problemas relacionados:

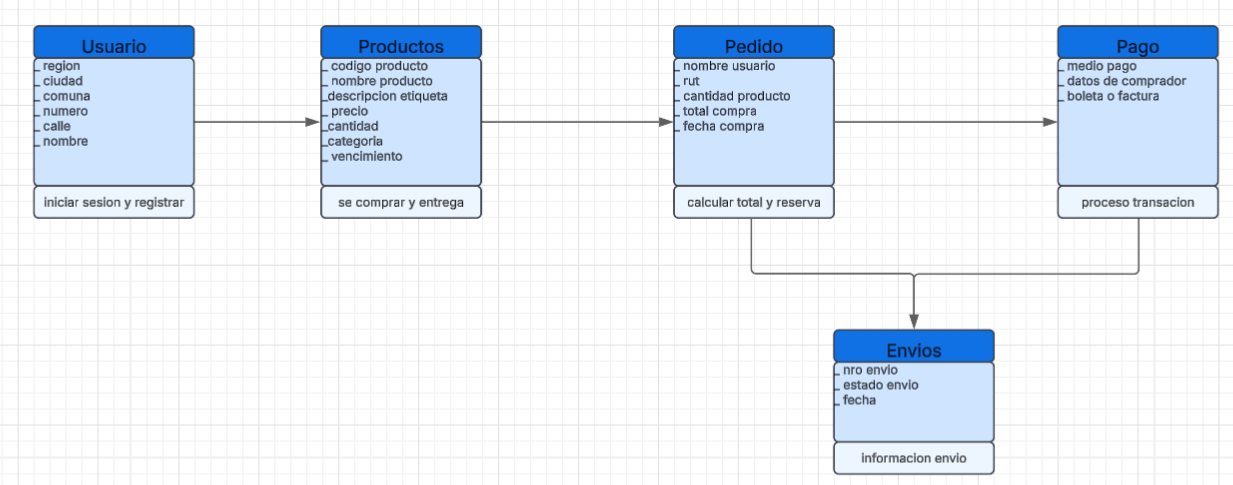
* Código extenso y complejo
* Problema de rendimiento: el costo aumenta según aumenta las funciones del sistema y así se hará inclusive más complejo
* Escalabilidad horizontal difícil: no se puede dividir sencillamente en componentes más pequeños para distribuir en múltiples servidores
* Problemas de disponibilidad
* Afectación a la experiencia del cliente
* Ineficiencia operativa

**3. Diseño de la Nueva Arquitectura:**

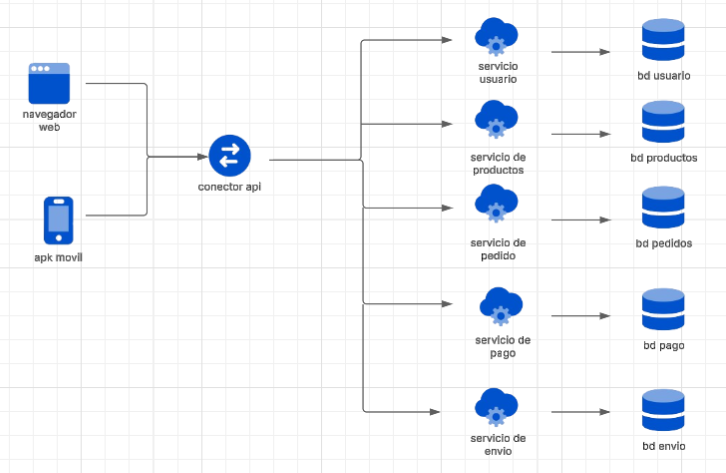
* Microservicio basado en la administración del sistema. (crear y modificar usuarios, asignar permisos, respaldo y restauración de datos)
* Microservicio basado en el inventario. (Agregar, eliminar y actualizar productos. Reporte de ventas y rendimiento. Autorización de pedidos para reabastecimiento de la tienda)
* Microservicio basado en ventas del producto. (transacciones de ventas, descuentos, devoluciones, consulta de disponibilidad del producto y emitir facturas a SII y enviar a correo al cliente.)
* Microservicio basado en logística de envío. ( Crear y actualizar estado y envíos, planificación de rutas, actualización de información y petición de pedido al proveedor).
* Microservicio basado en clientes vía web. (creación y actualización de login, buscador, filtros de productos, carro de compras, detalle de compra y envíos, historial de pedidos,soporte, reseñas y calificaciones de producto, agregar descuentos).
* Diagrama de Casos de Uso



* Diagrama de clases



* Diagrama de Despliegue



**4. Planificación de la Migración:**

Para esto hay que planificar la estrategia de migración y una vez realizado ir de manera gradual trabajando en equipos de trabajo, definiendo los frameworks a utilizar. Luego de esto planificar los microservicio tal como:

-Servicio usuario

-producto

-pedido

-pago

-envío

Por último es necesario realizar pruebas de despliegue y escalabilidad para asegurar que este funcione correctamente para así desactivar gradualmente el uso del sistema monolítico.

Para no tener problemas por operación de negocios separaremos en partes

* parte 1: desplazamiento de servicios básicos (gestión productos y clientes)
* parte 2: uso procesos de ventas y logística
* parte 3: juntar servicios y comprobar que el sistema funcione

| **Riesgos** | **Mitigación** |
| --- | --- |
| Cuello de botella | creando más base de datos para el paso de la información de manera escalonada |
| Pérdida de datos | generando un respaldo de datos |
| Filtración de datos | encriptación de datos |
| Dependencia local | establecer una comunicación a través de la red puede ser por API o web service |