

Sistema de Gestión de Pedidos - Delivery de Medicamentos

Entrega 1 – Laboratorio de Bases de Datos Relacionales (PostgreSQL)

Grupo 1

1. Introducción

Una empresa de delivery especializada en medicamentos a domicilio necesita una base de datos que le permita gestionar sus pedidos, clientes, farmacias, productos y repartidores. Además de mantener registros, el sistema debe permitir consultas analíticas, automatización de procesos y manejo eficiente de la información.

2. Objetivos

- Diseñar y construir una base de datos relacional en PostgreSQL para un sistema realista de delivery.
- Aplicar consultas SQL avanzadas para obtener información significativa del negocio.
- Crear procedimientos almacenados, triggers y vistas para automatizar procesos y facilitar la toma de decisiones.

3. Requisitos técnicos

Modelo de datos sugerido (mínimo):

- - Clientes
- - Repartidores
- - Farmacias
- - Productos
- - Pedidos
- - Detalle de pedidos
- - Medios de pago
- - Calificaciones

Obligatorio:

- - Llaves primarias, foráneas y restricciones de integridad.
- - Índices donde sea necesario.
- - Carga de datos de ejemplo (incluidos en el script).
- Uso de JWT para el registro de usuarios y login.
- Se debe realizar un api Rest con Spring y Vue.
- Se prohíbe el uso del ORM, se deben realizar las consultas SQL a “mano”.

4. Actividades obligatorias

Consultas SQL complejas (mínimo 6):

1. ¿Qué cliente ha gastado más dinero en pedidos entregados?
2. ¿Cuáles son los productos más pedidos en el último mes por categoría?
3. Listar las farmacias con más entregas fallidas.
4. Calcular el tiempo promedio entre pedido y entrega por repartidor.
5. Obtener los 3 repartidores con mejor rendimiento (basado en entregas y puntuación).
6. ¿Qué medio de pago se utiliza más en pedidos urgentes?

Procedimientos almacenados (mínimo 3):

7. Registrar un pedido completo.
8. Cambiar el estado de un pedido con validación.
9. Descontar stock al confirmar pedido.

Triggers (mínimo 3):

10. Insertar automáticamente la fecha de entrega al marcar como entregado.
11. Registrar una notificación si un medicamento con receta es pedido sin validación.
12. Insertar una calificación automática si no se recibe en 48 horas.

Vistas (mínimo 3):

13. Resumen de pedidos por cliente (monto total, número de pedidos).
14. Vista de desempeño por repartidor.
15. Vista de farmacias con mayor volumen de productos entregados.

5. Actividades opcionales (para nota extra)

- Generar un ranking de productos según devoluciones o cancelaciones.
- Crear una función para calcular el riesgo de cliente por comportamiento (frecuencia de reclamos, pedidos fallidos).
- Agregar tabla `recetas_médicas` con validación especial en triggers.

6. Datos de ejemplo

Cargados en `estructura_postgresql.sql`:

- 10 clientes
- 5 farmacias
- 20 productos (algunos requieren receta)
- 5 repartidores
- 20 pedidos históricos
- Medios de pago variados
- Calificaciones simuladas

7. Entrega

- Archivo `.sql` con el esquema, datos, vistas, procedimientos, triggers y consultas.
- Documento `.pdf` o `.docx` con capturas y resultados de pruebas.
- Fecha de entrega: 12 de mayo (15 min de presentación)