

# ■ Guía de Referencia Rápida – Sistema de Órdenes de Servicio para Drones

## ■ Endpoints del Backend (FastAPI)

POST /orders/ → Crear orden

GET /orders/ → Listar todas las órdenes

GET /orders/{id} → Obtener una orden por ID

PUT /orders/{id} → Actualizar orden completa

PATCH /orders/{id}?estado=terminado → Actualizar solo el estado

DELETE /orders/{id} → Eliminar una orden

## ■ Base de Datos (Postgres en Docker)

Nombre de la DB: orders\_db

Usuario: user

Contraseña: password

Host en Docker: db

Puerto: 5432

Cadena conexión: postgresql://user:password@db:5432/orders\_db

## ■ Variables importantes en el backend

db → lista simulada de órdenes (cuando no uses Postgres).

Order → modelo de datos.

order\_id → parámetro de la URL (/orders/3).

orden.id → identificador de cada objeto en la DB.

if orden.id == order\_id → compara el ID en DB con el de la API.

## ■ Rutas clave del frontend (React Router)

/ → Listado de órdenes

/nueva → Crear nueva orden

/actualizar/:id → Actualizar estado de orden

## ■ Comandos clave en terminal

Backend → pip install fastapi uvicorn psycpg2-binary sqlalchemy

Backend → uvicorn main:app --reload

Docker → docker compose up --build / docker compose down

Frontend → npx create-react-app frontend

Frontend → npm install axios react-router-dom

Frontend → npm start

## ■ Flujo de trabajo típico

1. Registrar orden → POST /orders/

2. Listar órdenes → GET /orders/

3. Actualizar estado → PATCH /orders/{id}?estado=en%20proceso

4. Finalizar orden → PATCH /orders/{id}?estado=entregado

5. Eliminar orden → DELETE /orders/{id}