SRV'D Take-Home – Code Explanation

Overview

This app is a order routing dashboard. It helps admins view, manage, and optimize delivery orders by borough.

Tech Stack

- Frontend: ReactJS + MUI
- Backend: Node.js + Express + Sequelize
- Database: PostgreSQL
- WebSocket: Socket.IO

Backend Explanation

We use Express with Sequelize ORM. Models define `Order` with fields like customer, status, and borough. Routes allow admins to fetch orders, update statuses, and retrieve borough-optimized route batches.

Frontend Explanation

The React app fetches orders and displays them in a responsive table. Admins can filter by borough, update statuses, and fetch optimized batches. Batches and alerts are shown in expandable accordions. WebSocket handles live updates.

Routing & Optimization

The backend groups orders by borough, batching them in groups of 5. Alerts are triggered when a borough has 10+ pending orders. This logic runs on `/routes/optimized`.

Live Refresh

Using Socket.IO, the backend emits updates when order statuses change. The frontend listens for `order-updated` events and refreshes the data in real-time.

Run Instructions

1. Start PostgreSQL and set credentials in `server/config/config.json`.
2. Migrate and seed database using Sequelize CLI.
3. Run server (`npm start`) and client (`npm run dev`).
4. Open the frontend in your browser.