City-Scale Digital Twin: Real-Time Data Ingestion and Visualization Sequence

City Manager	React Frontend	FastAPI Backend	PostgreSQL DB	HiveMQ MQTT Broker	Data Simulator

Real-Time Data Ingestion (Continuous)

loop [30-Second Interval]

Data Simulator → HiveMQ MQTT Broker: Publish IoT data (e.g., TrafficSensor)

FastAPI Backend → HiveMQ MQTT Broker: Subscribe to data topics

HiveMQ MQTT Broker → FastAPI Backend: Deliver new data message

FastAPI Backend → PostgreSQL DB: Ingest and persist data (async write operation)

PostgreSQL DB ⇢ FastAPI Backend: Acknowledge persistence

Data Retrieval & Visualization (On-Demand)

City Manager → React Frontend: Open Dashboard

React Frontend → FastAPI Backend: Request latest sensor data (GET /api/traffic_sensors/latest)

FastAPI Backend → PostgreSQL DB: Query for latest sensor reading

PostgreSQL DB ⇢ FastAPI Backend: Return data

FastAPI Backend ⇢ React Frontend: Respond with JSON data

React Frontend → React Frontend: Update map with new sensor data

React Frontend → City Manager: View visualized real-time data

City Manager	React Frontend	FastAPI Backend	PostgreSQL DB	HiveMQ MQTT Broker	Data Simulator