Smart Taxi Dispatch System: Balancing Taxis in Real-Time

:: Project Objectives

Project Goal

Build a live system that:

1. Tracks where taxis are right now across Singapore.
2. Identifies areas with too many taxis (low demand) or too few (high demand).
3. Suggests where drivers should move to pick up more passengers quickly.

The Problem Today

- **Drivers lose time (and money)** sitting in areas with few passengers.
- **Passengers wait longer** because taxis aren't where demand is high.
- **Empty trips ("dead mileage")** hurt driver earnings and increase costs.

Who Benefits & How

- **Taxi companies**: Gain real-time visibility into taxi locations and availability through a live data map interface. This helps improve fleet distribution and response times, ensuring better service coverage across the city.
- **Independent drivers**: Access a real-time map visualizing current taxi positions and activity, helping them identify areas with potential passenger demand or reduced competition.

Expected Improvements

- ✅ **Shorter wait times for passengers** (goal: 15% faster pickups).
- ✅ **Higher earnings for drivers** (goal: 8–12% more rides per hour).
- ✅ **Less wasted fuel** (goal: 10% fewer empty trips).

How It Works

1. **Live taxi locations**: Pulls real-time data from LTA.
2. **Smart matching**: Maps taxis to neighborhoods and predicts where demand will rise.
3. **Live Map Guidance:** Provides real-time visibility into taxi distribution, helping drivers identify under-served areas or regions with low taxi presence for improved service coverage.

Milestone	Tasks	Deliverables/Documentation
1	**Identify and collect relevant datasets from open sources, APIs, or internal databases.** **Ensure proper access (API keys, licenses, or permissions) are in place.** **Data Sourcing**	Data Sources and API access setup: 1. API - LTA Data Mall (Taxi Availability) - Provides live taxi locations across Singapore; core dataset for supply tracking 2. Excel - Kaggle Singapore City Geo-Coordinates - Maps postal codes and coordinates to planning areas; used for spatial tagging of taxis
2	**Develop scripts/tools to fetch data automatically at defined intervals.** **Store raw data outputs in a staging area (CSV/JSON/filesystem).** **Implement logging to track fetch dates and errors.** **Data Extraction Setup**	- Python data fetching scripts and documentation. - Raw data samples (CSV/JSON) staged and documented. => Refer to Jupyter Notebook
3	**Standardize values (e.g., names, formats, coordinate systems).** **Handle missing, inconsistent, or duplicate records.** **Transform raw data into a structured format aligned with project needs.** **Data Cleaning & Transformation**	- Transformation/ETL scripts. - Data validation (missing values, duplicates, and standardization). • Pluralise, lowercase, split geolocations table into places and bounding_box tables. - Cleaned dataset prepared for PostgreSQL database. => Refer to Jupyter Notebook

4	**Design schema with relevant tables, relationships, and indexes.** **Insert cleaned data into the database for efficient querying.** **Set up update pipelines to refresh data regularly.** **Database Storage**	- Database schema & ERD diagram - Normalized tables - Populated clean database - Updated pipeline ● Pipeline can be manually triggered to run infinitely while appending captured data to DBMS by batch. => Refer to Jupyter Notebook and presentation slides
5	**Explore data with descriptive / advanced analysis methods.** **Build visualizations (charts, dashboards, maps) to highlight insights.** **Summarize findings and provide recommendations or action points.** **Data Analysis & Visualization**	- SQL query for aggregated data ● Lowest/highest taxi availability - Matplotlib for visuals and predictive analysis ● Recommended Top 10 places likely to benefit from the TRS to pilot the system => Refer to presentation slides