

# User Guide: Greater Cairo Smart City Transportation System

## 1. Introduction

Welcome to the Greater Cairo Smart City Transportation Optimization System! This guide will help you understand how to set up, run, and interact with the system and its various modules through the planned dashboard interface (Streamlit application).

## 2. System Overview

The system is designed to analyze and optimize different aspects of urban transportation, including:

- \* **Infrastructure Planning:** Designing optimal road networks.
- \* **Traffic Flow Management:** Finding best routes and analyzing congestion.
- \* **Emergency Response:** Optimizing routes for emergency vehicles.
- \* **Public Transit:** Enhancing bus and metro services.

## 3. Setup and Installation

(Refer to the main `README.md` in the project root for detailed setup and installation instructions.)

Key steps include:

- \* Ensuring Python 3.9+ is installed.
- \* Installing dependencies: `pip install networkx matplotlib pandas streamlit`
- \* Having `ffmpeg` installed is recommended if you wish to generate video animations from simulation outputs.

## 4. Data Requirements

The system relies on several JSON data files located in the `transportation_system/data/` directory:

- \* `road_data.json`: Defines road segments, intersections (nodes), and their properties (distance, capacity, coordinates).
- \* `facilities.json`: Lists important facilities (hospitals, airports, etc.) with their locations and types.
- \* `traffic_data.json`: Provides historical or real-time traffic flow volumes for different road segments at various times of day.
- \* `transit_data.json`: Contains details about metro lines (stations, passenger counts) and bus routes (stops, assigned buses, passenger counts), as well as public transport O-D demand.

Ensure these files are correctly formatted and populated before running analyses.

## 5. Running the Application (Dashboard)

The primary way to interact with the system is through the Streamlit dashboard.

1. **Navigate to the project directory** in your terminal: `bash cd path/to/transportation_system`
2. **Run the Streamlit application:** `bash streamlit run integration/main.py`  
(Or `streamlit run integration/dashboard.py` if that becomes the main entry point)
3. **Access the Dashboard:** Open your web browser and go to the URL provided by Streamlit (usually `http://localhost:8501`).

## 6. Using the Dashboard (Conceptual Features)

The dashboard will provide interactive controls and visualizations for each module:

### 6.1. Infrastructure Module

- **MST Network Design:**
  - Select criteria (e.g., minimize distance, minimize cost for potential roads).
  - Optionally prioritize connections to critical facilities or high-population areas.
  - View the resulting Minimum Spanning Tree overlaid on the city map.
  - See cost and distance summaries for the designed network.
- **Network Visualization:** View the complete road and facility network.

### 6.2. Traffic Flow Module

- **Shortest Path Finder:**
  - Select origin and destination nodes.
  - Choose time of day (e.g., morning peak, off-peak) to consider time-dependent traffic.
  - View the calculated shortest path on the map and its effective length/travel time.
- **K-Alternate Routes:** Find multiple alternative routes between an origin and destination.
- **Congestion Analysis:**
  - Select a time of day.

- View a congestion map showing V/C ratios for road segments (color-coded).
- Identify key bottlenecks.
- **Traffic Simulation (Visualization):** View pre-generated animations or trigger conceptual simulations to observe traffic dynamics (output as frames/video).

### 6.3. Emergency Module

- **Emergency Routing (A\*):**
  - Select an incident location (origin) and a destination (e.g., nearest hospital).
  - Specify time of day for traffic conditions.
  - View the optimized emergency route on the map.
- **Emergency Simulation (Visualization):** View pre-generated animations or trigger conceptual simulations of emergency vehicle dispatch and response.
- **Response Time Analysis:** View summary statistics on simulated emergency response times (dispatch, travel to scene, travel to hospital).

### 6.4. Transit Module

- **Transit Network Map:** View existing metro and bus lines overlaid on the city map.
- **Bus Allocation Optimization:**
  - Input total available bus fleet size.
  - See suggested allocation of buses to routes based on demand.
- **Metro Line Feasibility:**
  - Define a potential new metro line (sequence of stations).
  - Get a conceptual feasibility score based on demand coverage and connectivity.
- **Demand Coverage Analysis:**
  - View statistics on how well current transit services cover O-D passenger demand.
  - Identify top underserved O-D pairs.

### 6.5. System Reports

- Access or generate consolidated reports summarizing analyses from different modules (conceptual).

## 7. Interpreting Outputs

- **Maps and Visualizations:** Network graphs will show nodes (locations) and edges (roads/routes). Colors and line styles will differentiate types of roads, transit lines, congestion levels, etc. Pay attention to legends.

- **Tables and Statistics:** Numerical outputs will summarize costs, distances, travel times, passenger numbers, V/C ratios, etc.
- **Simulation Outputs:** Simulation frames can be compiled into videos to observe dynamic behavior. Analysis scripts will provide aggregate statistics from simulations.

## 8. Troubleshooting

- **Data Errors:** Ensure all JSON data files in the `data/` directory are correctly formatted and present. Missing or malformed data is a common source of issues.
- **Dependency Issues:** Verify all required Python packages are installed (see `README.md` ).
- **Module Not Found Errors:** If running scripts directly, ensure your Python path is set up correctly or you are running commands from the appropriate directory level.
- **Performance:** Complex analyses or large simulations can be computationally intensive.

## 9. Getting Help

- Refer to the module-specific `README.md` files for more details on individual components.
- Consult the `docs/system_documentation.md` for in-depth technical information.

Thank you for using the Greater Cairo Smart City Transportation Optimization System!