COMSATS University Islamabad
Sahiwal Campus

Project Proposal

REAL-TIME TRAFFIC ANALYSIS

By

FA24-BCS-380 Muneeb Farid

FA24-BCS-395 Dawood Baig

INSTRUCTOR

Mina Iqbal

Bachelor of Science in Software Engineering (2024–2028)

Date: 19 December 2025

Table of Contents

1. Abstract

- This project aims to develop a Real-Time Traffic Analysis System using Microsoft Visual Studio. The system will help monitor, analyze, and visualize traffic data dynamically, providing real-time insights into congestion, vehicle count, and flow patterns. By integrating a user-friendly GUI with efficient data structures, the project seeks to optimize city traffic management and reduce delays.

2. Introduction

- In modern cities, traffic congestion has become a growing issue that directly affects transportation efficiency, fuel consumption, and commuter stress levels. This project provides a technological solution that leverages data structures, algorithms, and GUI design to analyze traffic conditions in real time. The proposed system will visualize live traffic flow using a graphical interface built in web, allowing users to monitor routes and detect bottlenecks.

3. Problem Statement

- The increasing number of vehicles and limited infrastructure capacity cause severe traffic congestion. This problem affects daily commuters, logistics operations, and emergency services. A real-time traffic analysis system can significantly improve urban mobility by collecting and analyzing live data to provide actionable insights.
    - Affects: Commuters, public transport systems, and emergency services.
    - Effects: Time wastage, higher fuel costs, pollution, and stress.
    - Benefits of Solution: Reduced congestion, optimized routes, and improved traffic flow.

4. Merits of Proposed System

- The proposed system offers efficient visualization, accurate analysis, and real-time updates. Using Web for GUI ensures cross-platform compatibility and an interactive interface. Efficient use of data structures such as graphs and priority queues enhances the speed and accuracy of traffic computations.

5. Scope

- This system can be deployed in smart cities for real-time monitoring of roads and highways. It can also be integrated with IoT devices and GPS sensors to enhance functionality. The project serves as a strong foundation for AI-powered traffic prediction and control systems.

6. Modules

- Module 1: Data Collection – Gathers live data from simulated traffic sensors or datasets.
- Module 2: Data Processing – Applies algorithms to analyze traffic density and flow patterns.
- Module 3: Visualization – Displays traffic information dynamically using web.
- Module 4: Database Module – Stores past traffic data for analysis and prediction.
- Module 5: Report Generation – Generates summary reports for decision-makers.

7. Memebers individual Work on project:

- **Muneeb Farid** works (coding, analyzing real time problem gathering data, converting into modulus and testing of the project.)
- **Dawood Baig** works (proposal, powerpoint slides and helping in coding.)

8. Pseudocode:

Start Program

Initialize traffic data list

Initialize Graph for road network

Initialize Queue for incoming traffic data

Initialize Stack for traffic alerts

While system is running:

    Collect traffic data

    Store data in Queue

    Update Graph with traffic flow

    If congestion detected:

      Push alert to Stack

    Display traffic status

    Save records

End Program

9. **Role of Each Member & Data Structure Implementation**
- **Muneeb Farid**

- Implemented **Graph data structure**

- Worked on traffic flow and route analysis

- Managed system logic and coordination

- **Dawood Baig**

- Implemented **Stack**

- Handled traffic alerts and real-time data processing

- Assisted in testing and debugging

- Each member implemented different data structures to ensure modular and efficient development.

10. **Usage of Data Structures**
- **Graph**
  Used to represent the road network where nodes are intersections and edges are roads.
- **Queue**
  Used to process real-time traffic data in the order it arrives.
- **Stack**
  Used to manage urgent alerts such as accidents or heavy congestion.
- **List / Array**
 Used to store and update vehicle counts and speed values.
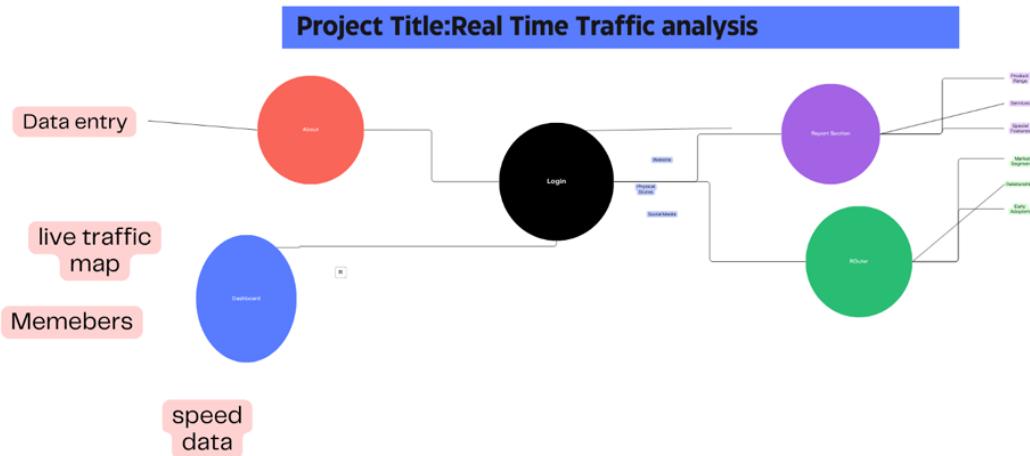
11.System Limitations / Constraints
- Requires consistent data input for accurate analysis.
  - Real-time implementation depends on hardware sensors.
  - System performance may vary depending on data volume.

12.Tools and Technologies

Tools / Technology	Version	Rationale
Microsoft Visual Studio	2022	Integrated development environment for Python development.
Web Framework	6.10	Used to design the interactive GUI for traffic visualization.
Python	3.10	Core programming language used for logic and data structure implementation.
Csv file	3	Lightweight database for storing traffic data records.

13.Hard Prototyping of the System AS Shown:

Project Title:Real Time Traffic analysis

Data entry

live traffic map

Memebers

speed data
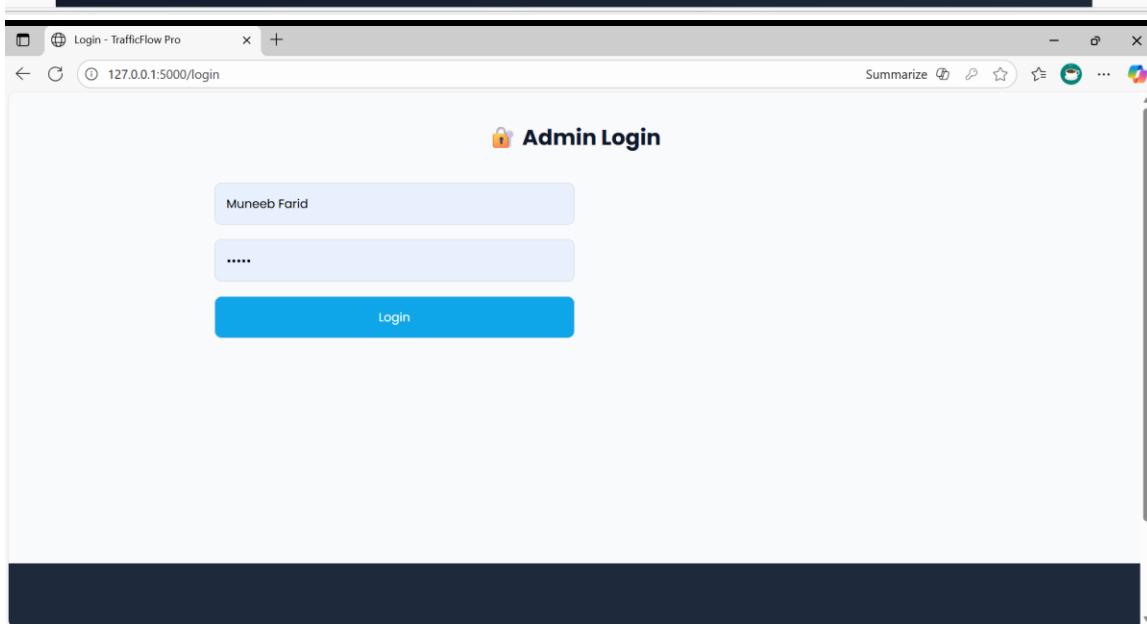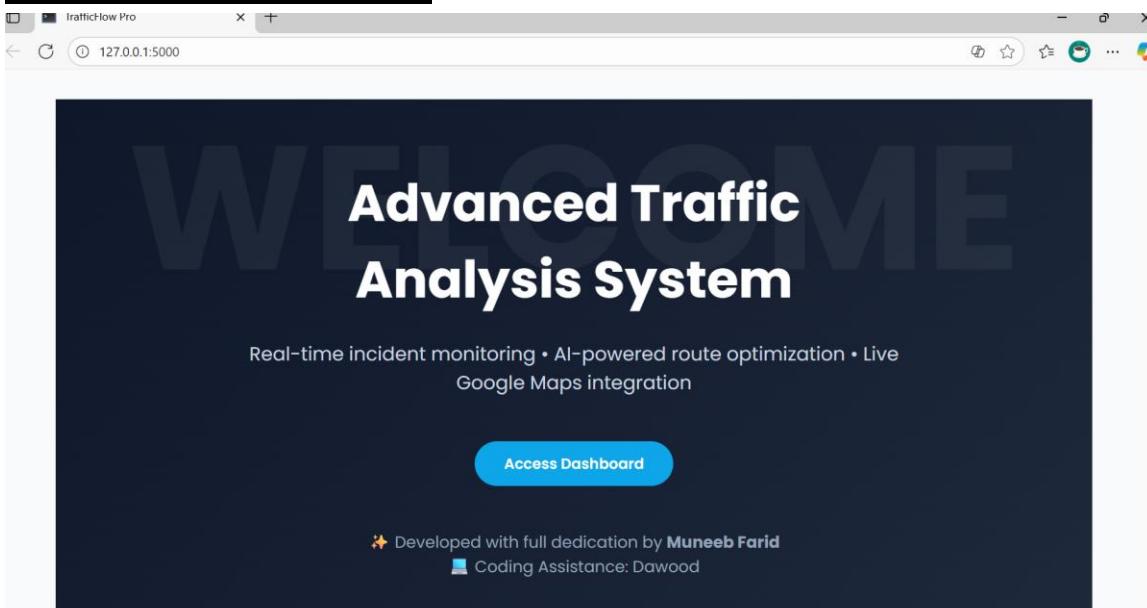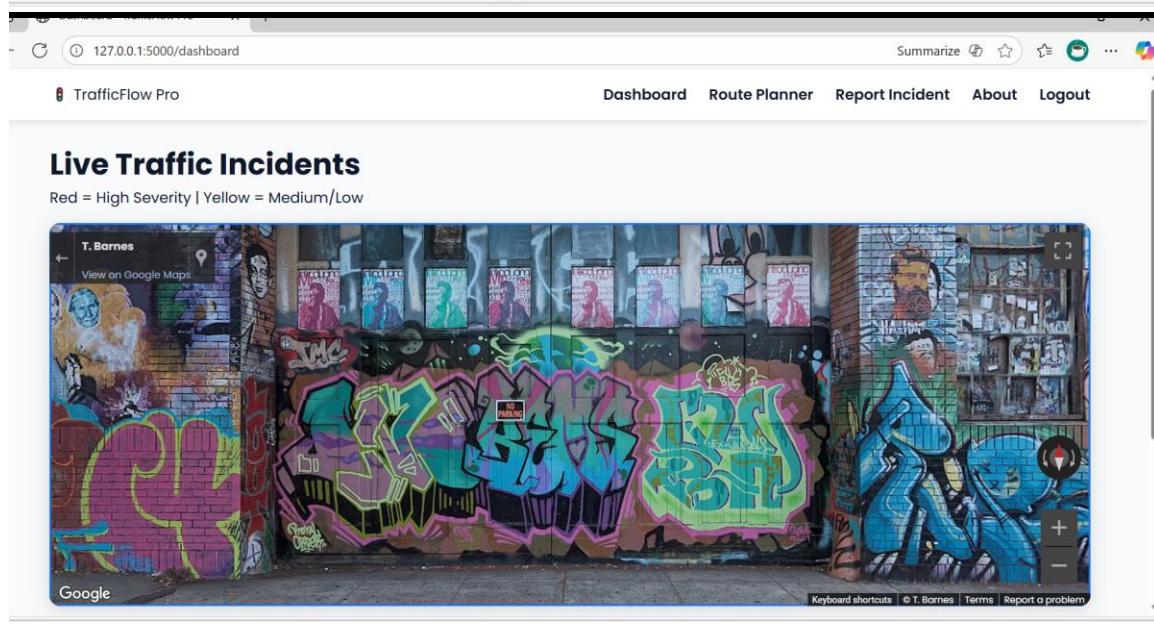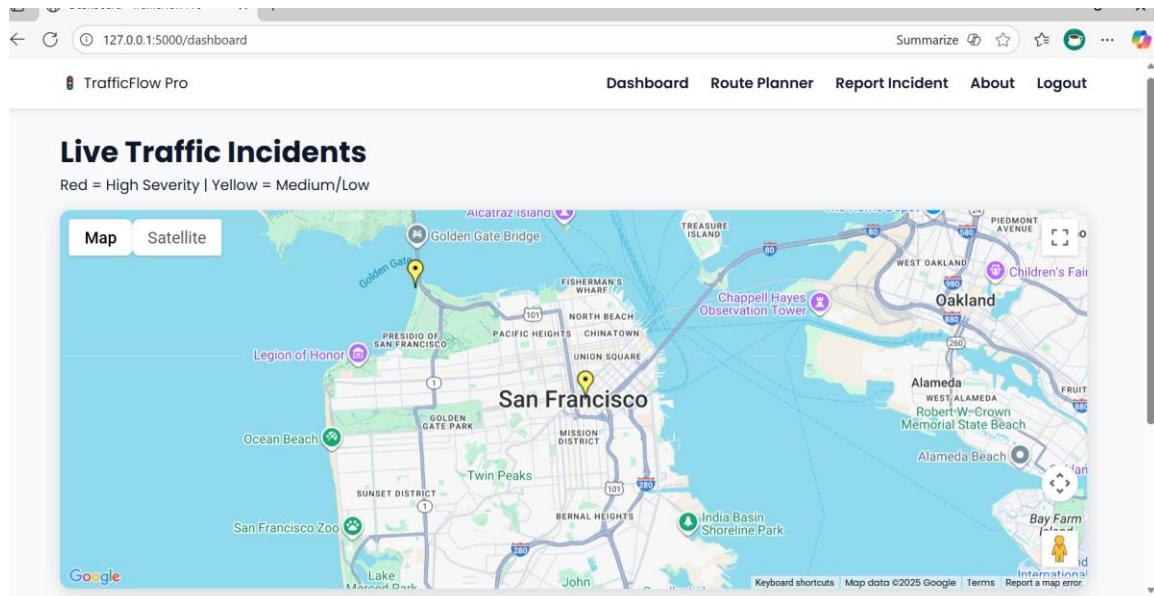
. 14.Hard Prototyping of Data Structures

- **Graph Prototype:**
  Nodes represent intersections, edges represent roads.

- **Queue Prototype:**
  Incoming traffic data processed sequentially.

- **Stack Prototype:**
  Latest alerts placed on top and handled first.

These prototypes ensured clear understanding before implementation.

15.Project Images:

🚦 TrafficFlow Pro

Dashboard   **Route Planner**   **Report Incident**   **About**   **Logout**

Live Traffic Incidents

Red = High Severity | Yellow = Medium/Low

🚦 TrafficFlow Pro

Dashboard   **Route Planner**   **Report Incident**   **About**   **Logout**

Live Traffic Incidents

Red = High Severity | Yellow = Medium/Low

TrafficFlow Pro

Dashboard   Route Planner   Report Incident   About   Logout

🏞️ Route Planner (Dijkstra Demo)

Enter source and target nodes (e.g., A → C)

Source (e.g., A)

Target (e.g., C)

Calculate Shortest Path

Report Incident - TrafficFlow Pro

TrafficFlow Pro

Dashboard   Route Planner   Report Incident   About   Logout

# 🚨 Report Traffic Incident

Latitude (e.g., 37.7749)

Longitude (e.g., -122.4194)

Low (e.g., slow traffic)

Describe the incident...

Submit Report

16.Data Gathering Approach

Data will be gathered from simulated sensors, GPS-based datasets, and sample traffic data sources. These inputs will help test and validate the algorithm's ability to detect congestion patterns, traffic density, and speed variations.

17. Conclusion:

The Traffic Analysis System demonstrates effective use of data structures, algorithms, and Python programming.
It fulfills all instructor requirements and provides a strong foundation for future enhancements such as AI-based traffic prediction.

18:References:

1. Cormenet al., Introduction to Algorithms
2. Data Structures and Algorithms in Python – Wiley
3. https://www.python.org/
4. https://www.geeksforgeeks.org/
5. Instructor Lecture Notes – Miss Mina Iqbal