REAL TIME TRAFFIC ANALYSIS

A MINI-PROJECT REPORT

Submitted by : MERVIN J – 2116220701163

in partial fulfilment of the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING,

RAJALAKSHMI ENGINEERING COLLEGE,

AUTONOMOUS, THANDALAM, CHENNAI,

NOV/DEC, 2024

RAJALAKSHMI
ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

BONAFIDE CERTIFICATE

.

Certified that this mini project “Nipah Virus (Niv) Testing Management System” is the bonafide work of “Mervin J(2116220701163)” who carried out the project work under my supervision

SIGNATURE

Mrs. JANANEE V,

Assistant Professor,

Computer Science Engineering,

Rajalakshmi Engineering College,

Thandalam, Chennai - 602105 .

Submitted for the End semester practical examination to be held on

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

I express my sincere thanks to my beloved and honourable chairman **MR.S.MEGANATHAN** and the chairperson **DR.M.THANGAM MEGANATHAN** for their timely support and encouragement.

I am greatly indebted to my respected and honourable principal **DR.S.N.MURUGESAN** for his able support and guidance.

No words of gratitude will suffice for the unquestioning support extended to us by my head of the department **DR.P.KUMAR**, and my Academic Head **DR.N.DURAIMURUGAN**, for being ever supporting force during my project work.

I also extend my sincere and hearty thanks to my internal guide **Mrs. JANANEE V** for her valuable guidance and motivation during the completion of this project.

My sincere thanks to my family members, friends and other staff members of Computer Science and Engineering.

ABSTRACT

The increasing trend of urbanization and vehicle use has brought about serious traffic congestion issues, making real-time traffic analysis a critical focus for effective transportation management. This project introduces a robust framework that combines advanced technologies, including machine learning, big data analytics, and the Internet of Things (IoT), to address the challenges associated with urban traffic. By integrating data from traffic sensors, GPS devices, and social media, the system can deliver real-time, dynamic traffic flow analysis, helping transportation authorities monitor and predict traffic patterns more accurately.

Our approach employs real-time data to identify congestion hotspots, optimize traffic signal timings, and recommend alternative routes. This proactive method reduces delays and allows for smoother traffic flow, benefiting both daily commuters and urban infrastructure. The system's AI-driven predictive models also enhance its capabilities, enabling it to forecast traffic conditions based on a combination of historical and real-time data. This predictive power supports faster decision-making, allowing traffic management systems to respond more effectively to changing conditions on the road.

The results of our framework demonstrate substantial improvements in average travel times, traffic fluidity, and road safety. By reducing congestion, this real-time traffic analysis framework not only minimizes economic losses caused by delays but also positively impacts environmental sustainability by lowering vehicle emissions. Our work underscores the transformative potential of real-time traffic solutions in modern urban settings, providing insights for future urban mobility advancements.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Traffic congestion poses major challenges in urban areas, leading to longer travel times, higher fuel consumption, and increased pollution. As urban populations grow, traditional traffic management systems that rely on fixed schedules are insufficient for managing dynamic traffic flow. Advances in big data, machine learning, and the Internet of Things (IoT) now enable real-time traffic analysis by gathering data from GPS devices, sensors, and social media to adapt swiftly to road conditions. This approach optimizes traffic flow, reduces delays, and enhances safety. By using predictive models trained on historical and real-time data, traffic managers can proactively prevent congestion through dynamic signal control and rerouting. This paper explores the potential of real-time traffic analysis to improve urban mobility and driving experiences.

1.2 SCOPE OF THE WORK

This work presents a comprehensive approach to designing, implementing, and evaluating a real-time traffic analysis system aimed at reducing congestion and enhancing traffic flow in urban areas. The system integrates data from traffic sensors, GPS, social media, and CCTV to provide a full, real-time view of traffic conditions. Advanced data processing methods ensure timely, accurate analysis, while AI-driven predictive models forecast congestion and accidents by leveraging both historical and live data. These insights are applied to optimize traffic management through dynamic signal control and rerouting strategies. The system's integration with IoT devices and Vehicle-to-Infrastructure (V2I) communication enables real-time updates for connected vehicles, making the system responsive and adaptable to evolving urban demands. Evaluations through simulated and real-world scenarios validate its impact on reducing travel times and enhancing road safety, while its scalability and adaptability make it suitable for diverse urban settings. This work thus provides a versatile framework for real-time traffic management, addressing the complexities of modern urban environments.

1.3 AIM & OBJECTIVES OF THE PROJECT

The aim of this project is to develop an intelligent, real-time traffic analysis system that leverages data from various sources to optimize traffic flow, reduce congestion, and improve urban mobility. By utilizing advanced technologies such as machine learning, big data, and IoT, the system aims to provide predictive insights and adaptive traffic management solutions that enhance transportation efficiency and safety. To achieve this aim, the project is structured around the following specific objectives:

Develop a Comprehensive Data Collection Framework

- Design a system that integrates real-time data from traffic sensors, GPS devices, vehicle tracking systems, and other relevant sources (e.g., social media and CCTV).
- Ensure the framework can handle large volumes of data with minimal latency.

Implement Real-Time Traffic Data Processing Algorithms

- Create algorithms for efficient processing, filtering, and normalization of real-time traffic data.
- Ensure the system provides near-instantaneous feedback and updates to traffic management systems and users.

Design and Train Predictive Traffic Models

- Utilize machine learning techniques to develop models capable of predicting traffic congestion, accidents, and flow patterns based on both real-time and historical data.
- Evaluate the accuracy and performance of these models under different traffic conditions.

Develop Adaptive Traffic Control Systems

- Design dynamic traffic signal control and route optimization algorithms that respond in real time to changes in traffic conditions.
- Implement vehicle rerouting suggestions to reduce congestion and minimize travel times.

Integrate IoT and Vehicle-to-Infrastructure (V2I) Communication

- Facilitate seamless communication between vehicles and the traffic management system using IoT devices and V2I technologies.
- Ensure vehicles can receive real-time traffic updates and route recommendations.

Conduct System Testing and Validation

- Perform simulations and real-world case studies to evaluate the system’s performance in various urban traffic environments.
- Analyze the system's impact on congestion reduction, travel time improvements, and road safety.

Ensure System Scalability and Adaptability

- Design the system to be scalable and adaptable to different city sizes and infrastructure setups.
- Test the system's capability to handle various traffic conditions and large data volumes.

By accomplishing these objectives, the project aims to provide a robust solution to urban traffic challenges, improving the efficiency of transportation systems and contributing to smarter, more sustainable cities.

CHAPTER 2

SYSTEM SPECIFICATIONS

The system specifications for the real-time traffic analysis system encompass the hardware and software requirements necessary to develop, deploy, and operate the traffic management solution efficiently.

2.1 SOFTWARE SPECIFICATIONS

The following software components are necessary for data handling, machine learning, and system management:

Operating System

- Linux-based OS (e.g., Ubuntu, CentOS) for servers due to its robustness, scalability, and security.
- Windows or macOS for development environments.

Programming Languages

- **Python**: For data processing, machine learning models, and system logic.
- **JavaScript (Node.js)**: For real-time web services and data visualization.
- **SQL/NoSQL Databases**: For data storage and management (PostgreSQL, MongoDB, Cassandra).

Machine Learning Libraries

- **TensorFlow/PyTorch**: For building and training predictive traffic models.
- **Scikit-learn**: For statistical modeling and basic machine learning tasks.
- **Pandas/Numpy**: For data manipulation and analysis.

Data Analytics Tools

- **Apache Kafka**: For real-time data streaming and handling.
- **Spark**: For large-scale data processing and analytics.
- **Elasticsearch**: For real-time data indexing and querying.

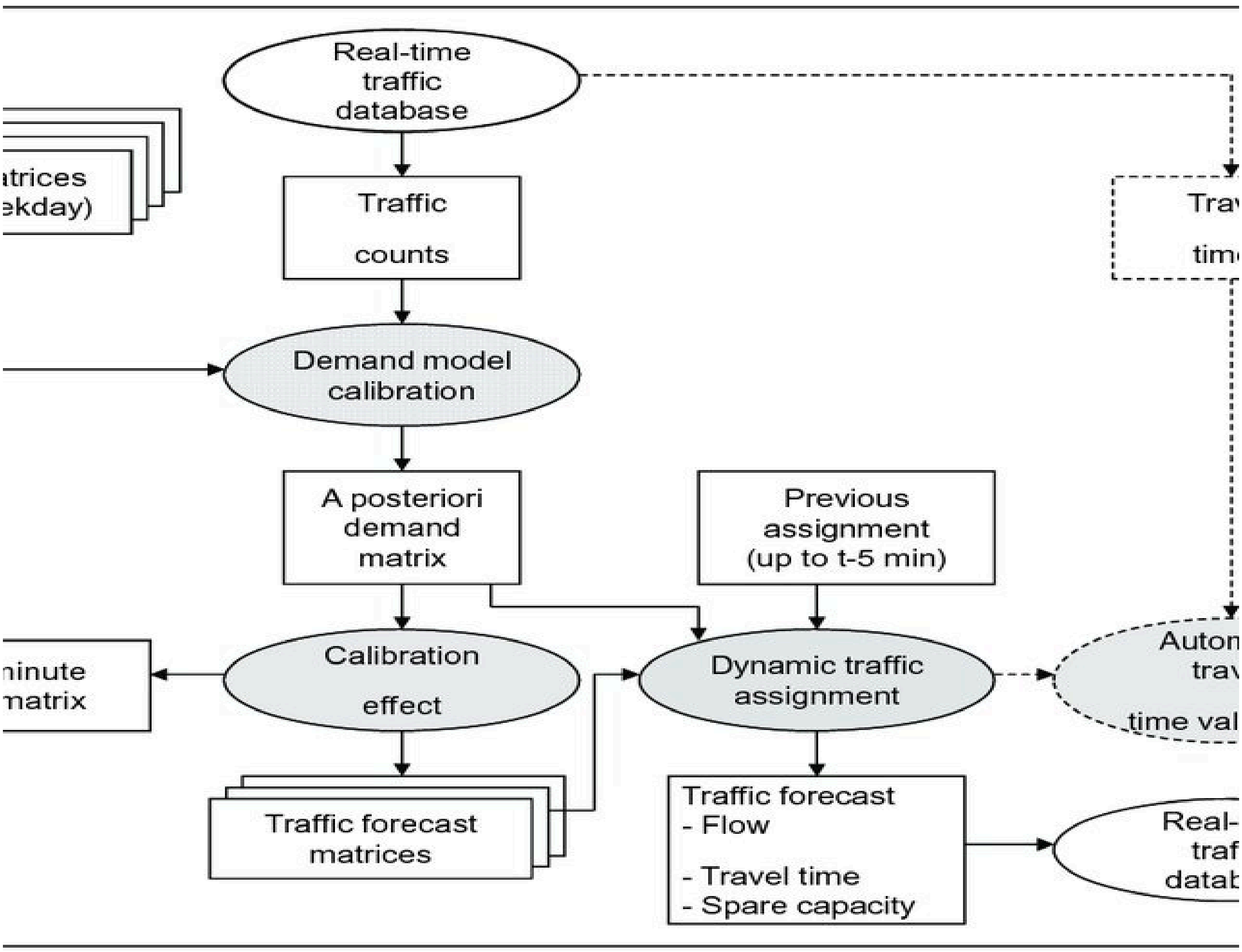
Traffic Management Software

- **SUMO**: For traffic simulation and model validation.
- **VISSIM**: For microscopic traffic simulations.
- Custom traffic control algorithms for signal optimization and vehicle rerouting.

By combining these hardware and software specifications, the system will effectively manage and analyze real-time traffic data, providing an adaptable and scalable solution to urban congestion challenges.

CHAPTER 3

ARCHITECTURE DIAGRAM



CHAPTER 4
MODULE DESCRIPTION

Module 1: Data Collection Module

This module is responsible for gathering real-time data from various sources and ensuring it is properly formatted for analysis.

Data Sources:

- Traffic Sensors: Collect data such as vehicle counts, speed, and occupancy from sensors installed at intersections and along roads.
- GPS Data: Tracks vehicle positions and movement using GPS-enabled devices from smartphones and connected vehicles.
- Camera Feeds: Real-time video from traffic cameras for vehicle detection and incident analysis.
- Social Media & Public Feeds: Collects traffic-related information from social platforms (e.g., Twitter) and external services like Google Maps or Waze.

Responsibilities:

- Collect and timestamp incoming data from multiple sources.
- Preprocess raw data for accuracy by handling missing or noisy data.

- Integrate data from various sensors into a unified format for analysis.

Module 2: Data Processing and Analysis Module

This module processes the collected data to derive meaningful insights and traffic patterns.

Data Preprocessing:

- Cleans, filters, and aggregates real-time traffic data to ensure reliability and consistency.
- Normalizes and formats data from heterogeneous sources (e.g., GPS logs, sensor data) for compatibility with the analytics engine.

Real-Time Analysis:

- Uses data stream processing techniques (e.g., Apache Kafka, Apache Storm) to analyze traffic conditions as they occur.
- Identifies congestion, accidents, or unusual traffic patterns in real time.

Historical Data Analysis:

- Retrieves and processes historical data to understand long-term traffic trends, seasonal variations, and recurring congestion points.
- Trains machine learning models for traffic prediction based on past patterns.

Module 3: Predictive Modeling and Machine Learning Module

This module builds and trains machine learning models to predict traffic conditions and incidents based on historical and real-time data.

Model Development:

- Implements machine learning algorithms (e.g., regression, neural networks) to predict future traffic flow, congestion hotspots, and potential accidents.
- Incorporates weather conditions, event data, and traffic incidents into the prediction models to improve accuracy.

Training and Validation:

- Trains predictive models using historical traffic data, optimizing them for performance under different scenarios.
- Evaluates model performance using metrics like accuracy, precision, recall, and real-world validation through simulations.

Traffic Forecasting:

- Provides short-term (minutes/hours) and long-term (days/weeks) traffic forecasts based on real-time inputs and historical patterns.
- Generates predictive alerts for traffic operators and drivers.

Module 4: Traffic Management and Control Module

This module focuses on real-time traffic control and optimization.

Signal Control Optimization:

- Dynamically adjusts traffic signal timings based on real-time traffic data to reduce congestion at intersections.

- Uses adaptive algorithms that optimize signal timings to respond to changes in vehicle flow and pedestrian activity.

Rerouting and Route Optimization:

- Suggests alternative routes for drivers to avoid congestion and reduce travel times.
- Provides real-time rerouting updates to navigation systems based on current traffic conditions and predicted bottlenecks.

Incident Management:

- Detects accidents, roadblocks, and other traffic incidents and alerts relevant authorities or traffic operators.
- Provides suggested actions such as lane closures, detours, or temporary signal changes to mitigate traffic disruptions.

Module 5: IoT Integration and (V2I) Communication Module

This module facilitates seamless interaction between vehicles, roadside infrastructure, and traffic control systems.

IoT Device Communication:

- Collects real-time data from IoT devices (e.g., connected traffic lights, smart signs) and sends updates back to the central system.
- Integrates with smart city infrastructure for vehicle counting, speed detection, and traffic flow monitoring.

Vehicle-to-Infrastructure (V2I):

- Facilitates communication between vehicles and traffic control systems to provide real-time updates on traffic signals, road conditions, and route changes.
- Supports connected and autonomous vehicles (CAVs) by enabling V2I interaction for smoother traffic flow.

Module 6: Data Storage and Management Module

This module ensures efficient storage, management, and retrieval of large volumes of traffic data.

Data Storage:

- Uses both relational (SQL) and NoSQL databases for storing structured and unstructured traffic data (e.g., GPS logs, sensor readings, video feeds).
- Ensures scalability to handle increasing amounts of data as more sensors and devices are integrated into the system.

Data Security and Privacy:

- Implements data encryption protocols (SSL/TLS) for secure transmission of sensitive information.
- Ensures compliance with data privacy regulations, protecting user identity and location data.

Data Retrieval:

- Provides efficient querying and retrieval mechanisms for real-time and historical data needed for analysis, visualization, and reporting.

Module 7: User Interface and Visualization Module

This module provides an intuitive interface for traffic operators, city planners, and drivers.

Traffic Operator Dashboard:

- Real-time visualizations of traffic conditions on interactive maps, showing vehicle density, congestion areas, and incidents.
- Control panel for adjusting signal timings, viewing system alerts, and monitoring key performance metrics.
- Tools for generating traffic reports and analyzing long-term traffic trends.

Mobile Application for End Users:

- A mobile app or web-based platform providing real-time traffic updates, alternative routes, and congestion alerts.
- Features V2I communication, offering drivers real-time signal timings, road closures, and traffic recommendations.

Module 8: System Testing and Simulation Module

This module is responsible for validating the system's performance before deployment.

Traffic Simulations:

- Uses traffic simulation tools (e.g., SUMO, VISSIM) to model and test the real-time traffic management algorithms.
- Evaluates the system's ability to reduce congestion, optimize signal timings, and provide accurate traffic predictions under simulated conditions.

Performance Monitoring:

- Monitors the system’s responsiveness, latency, and data accuracy during real-time operations.
- Provides feedback for system improvement and optimization.

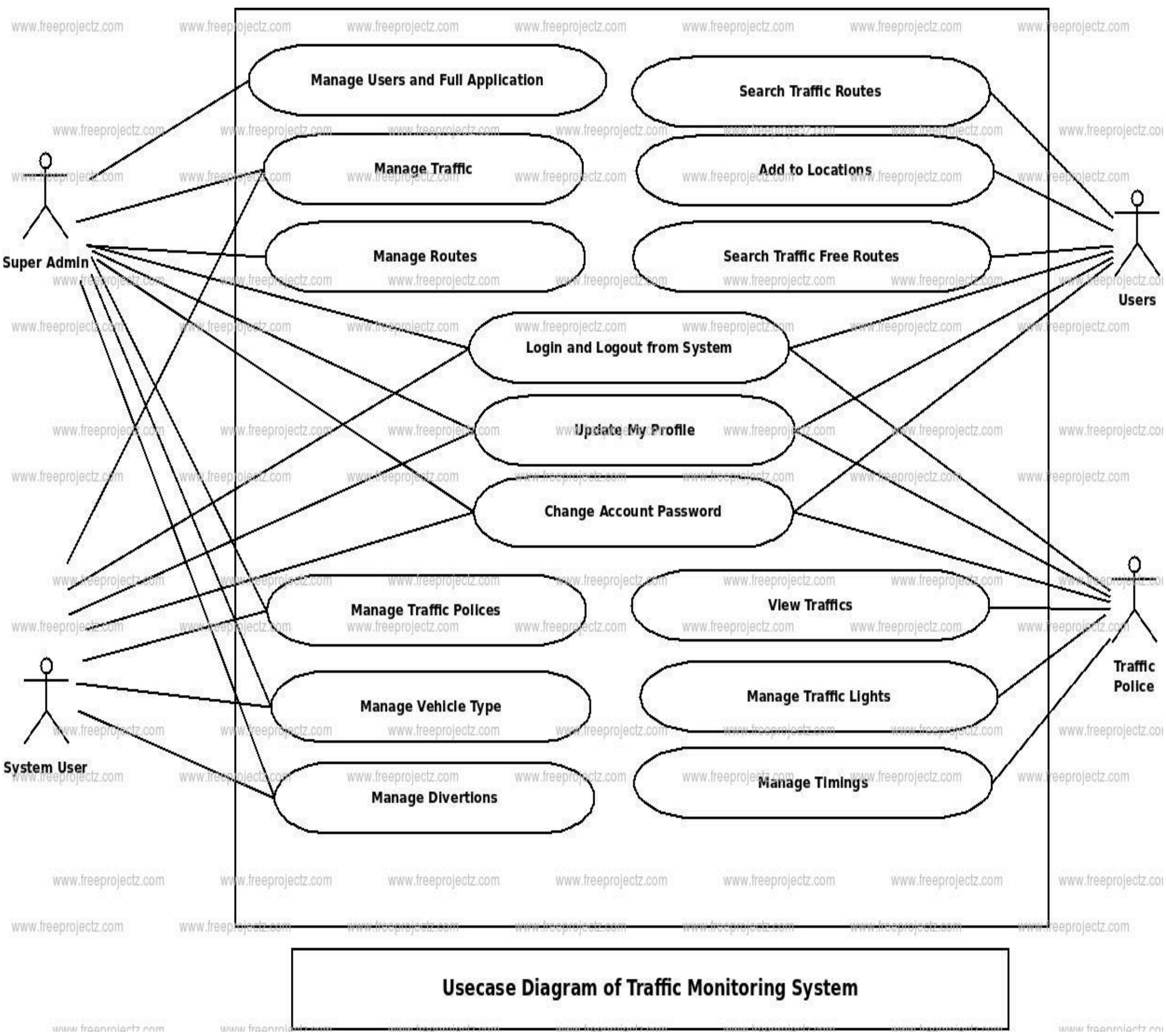
These modules work together to create a robust and scalable real-time traffic analysis system capable of addressing the complexities of modern urban traffic.

CHAPTER 5

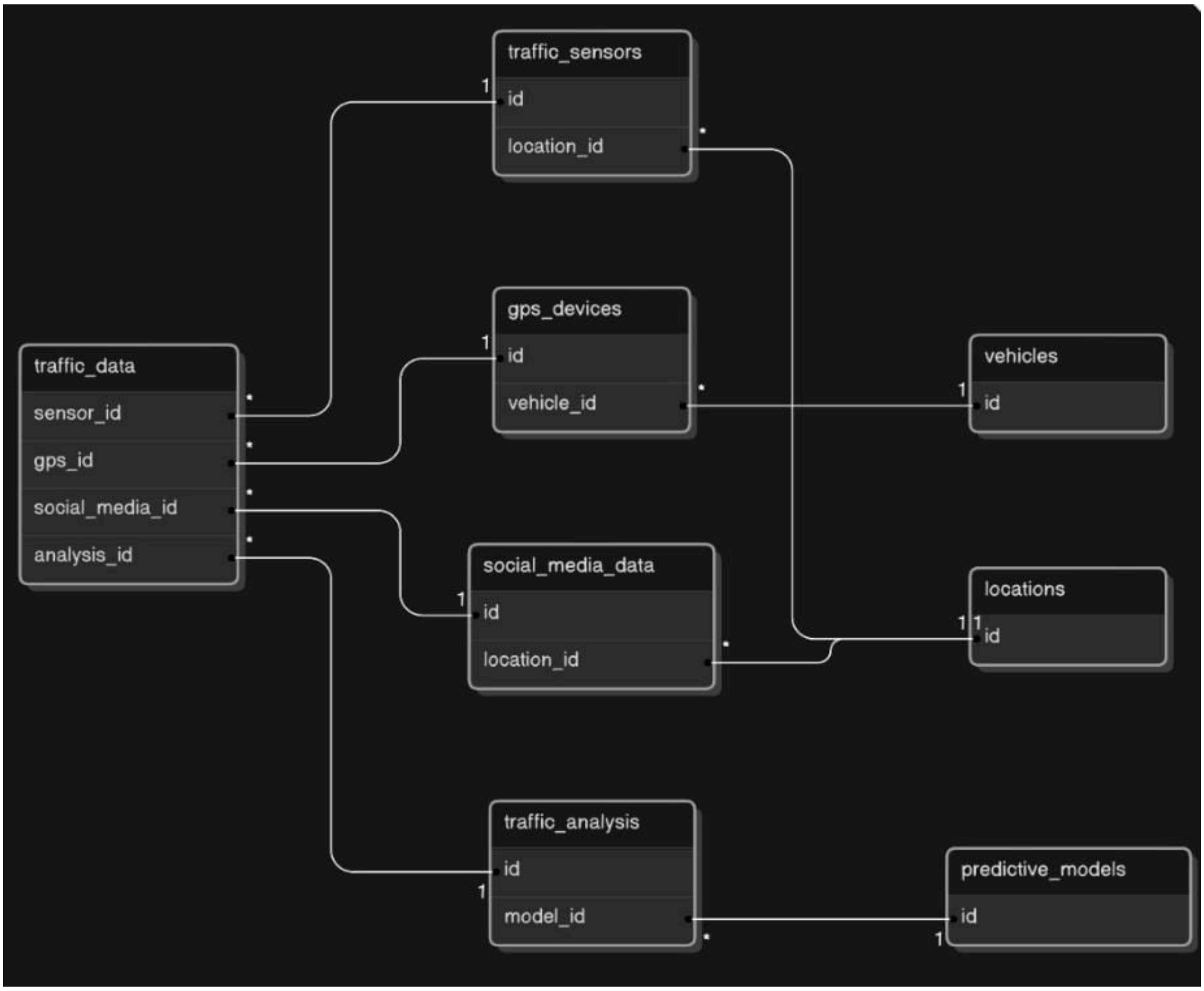
SYSTEM DESIGN

The design focuses on integrating various components and ensuring efficient communication between modules to achieve real-time data collection, processing, analysis, and traffic control.

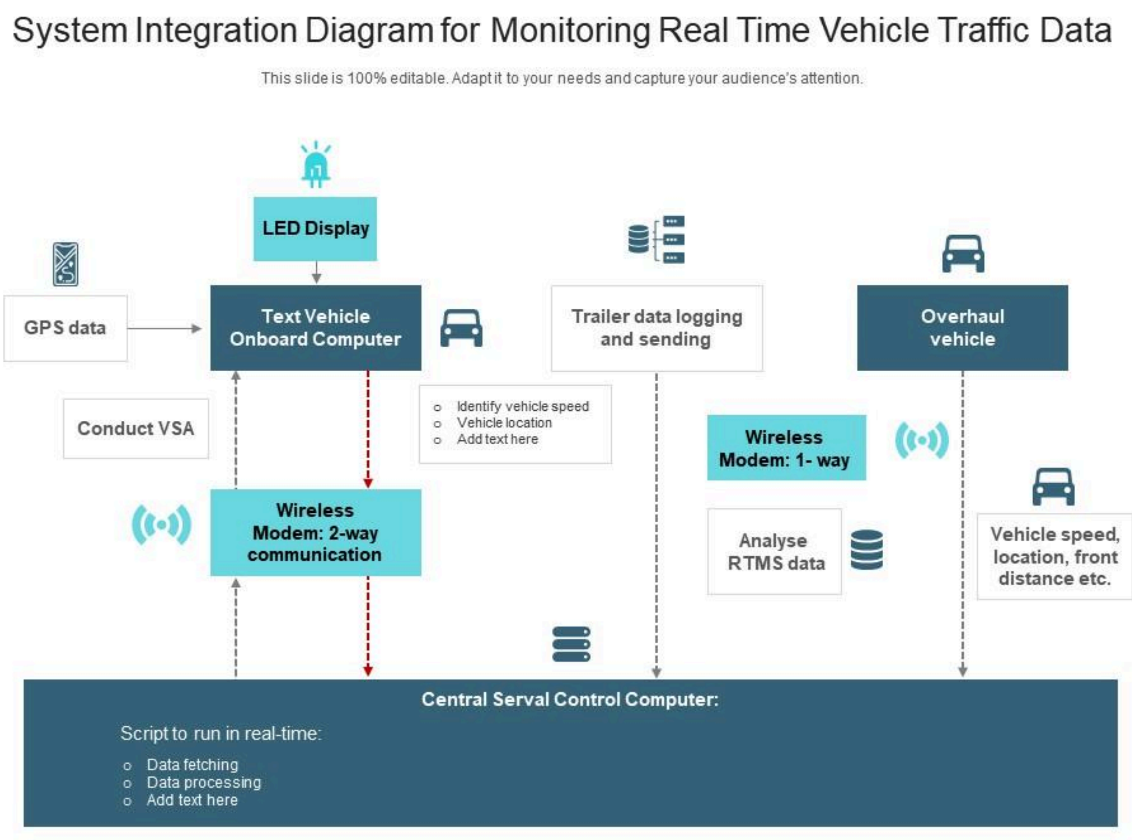
5.1 USE CASE DIAGRAM



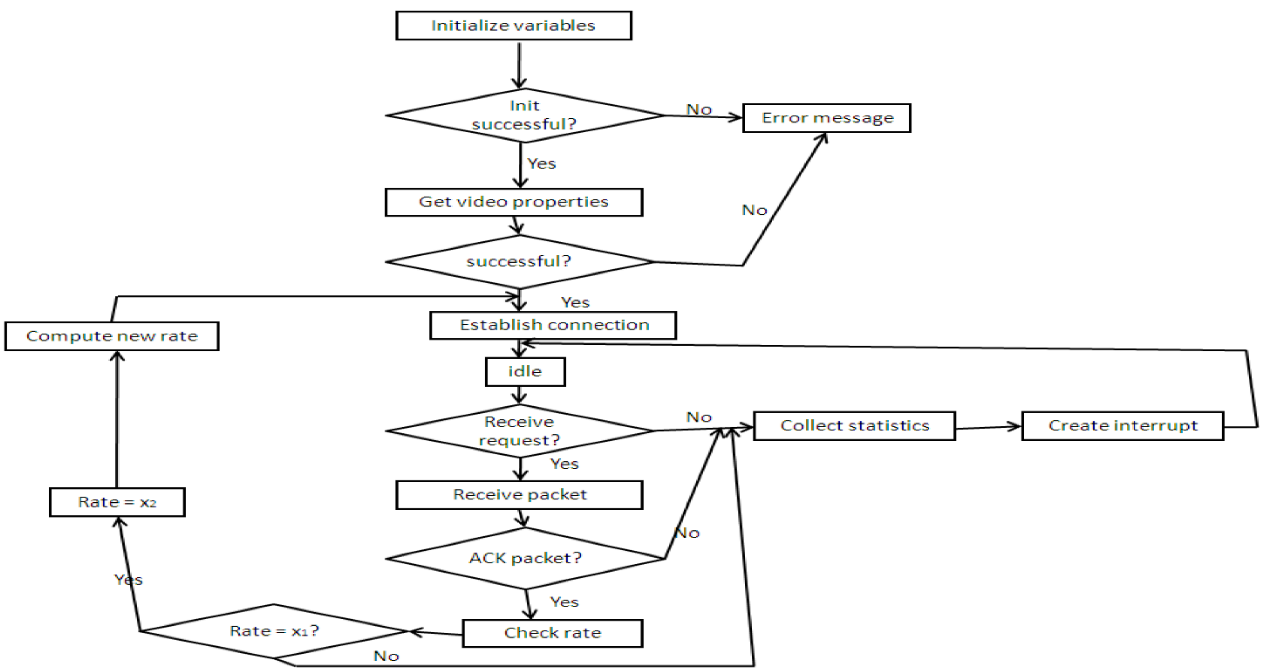
5.2 ENTITY-RELATIONSHIP DIAGRAM



5.3 DATA FLOW DIAGRAM



5.4 ACTIVITY DIAGRAM



CHAPTER 6
SAMPLE CODING

Login page.html:

```html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>Login</title>
</head>
<body>
<h1>Login</h1>
<form action="/login" method="post">
<label for="username">Username:</label>
<input type="text" id="username" name="username" required><br><br>
<label for="password">Password:</label>
<input type="password" id="password" name="password"
required><br><br>
<button type="submit">Login</button>
</form>
<p>Don't have an account? <a href="/signup">Sign up here</a>.</p>
</body>
</html>
```

App.py :

```python
from flask import Flask, Response, render_template, jsonify, request, redirect, url_for,
session, flash
import cv2
from ultralytics import YOLO
import numpy as np
import mysql.connector
```

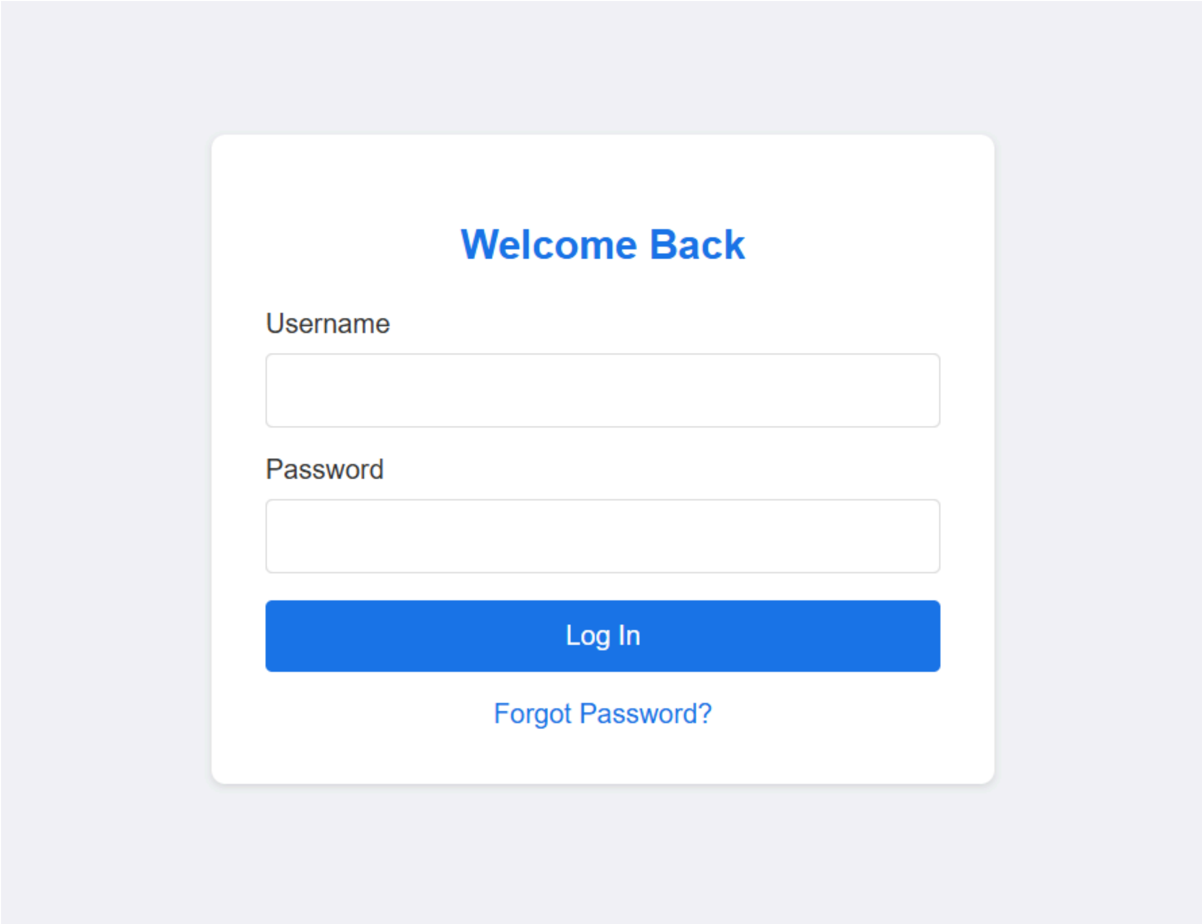
```python
app = Flask(__name__)
app.secret_key = 'your_secret_key'  # Set a secret key for session management

# Load the YOLO model and set up video capture
model = YOLO('models/best.pt')
cap = cv2.VideoCapture('sample_video.mp4')

if not cap.isOpened():
    raise Exception("Could not open video device")

# Define constants (same as in your original code)
heavy_traffic_threshold = 10
vertices1 = np.array([(465, 350), (609, 350), (510, 630), (2, 630)], dtype=np.int32)
vertices2 = np.array([(678, 350), (815, 350), (1203, 630), (743, 630)], dtype=np.int32)
x1, x2 = 325, 635
lane_threshold = 609

def create_connection():
    connection = mysql.connector.connect(
        host='localhost',
        user='host',  # Your MySQL username
        password='root',  # Your MySQL password
        database='traffic_analysis_db'  # Your database name
    )
    return connection

def process_frame():
    while True:
        ret, frame = cap.read()
```

```
        if not ret:
            break

        # Create a copy of the original frame to modify
        detection_frame = frame.copy()
        detection_frame[:x1, :] = 0  # Black out from top to x1
        detection_frame[x2:, :] = 0  # Black out from x2 to the bottom of the frame

        # Perform inference on the modified frame
        results = model.predict(detection_frame, imgsz=640, conf=0.4)
        processed_frame = results[0].plot(line_width=1)

        # Restore the original top and bottom parts of the frame
        processed_frame[:x1, :] = frame[:x1, :].copy()
        processed_frame[x2:, :] = frame[x2:, :].copy()

        # Convert frame to JPEG for video streaming
        ret, buffer = cv2.imencode('.jpg', processed_frame)
        frame = buffer.tobytes()

        # Yield the frame in byte format for streaming
        yield (b'--frame\r\n'
               b'Content-Type: image/jpeg\r\n\r\n' + frame + b'\r\n')
@app.route('/traffic_data')
def traffic_data():
    data = get_traffic_data()
    return jsonify(data)

if __name__ == '__main__':
    app.run(debug=True)
```

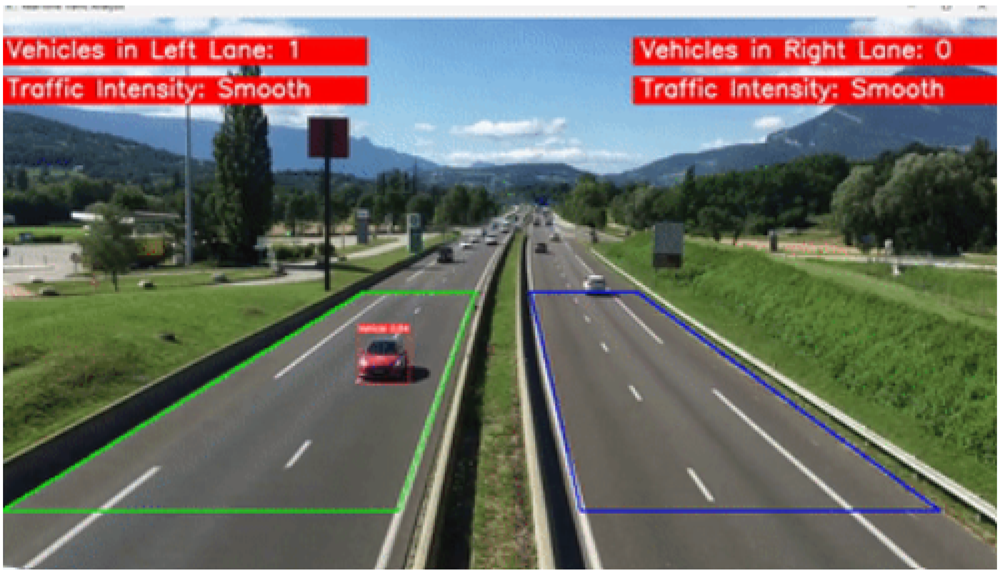
CHAPTER 7

SCREENSHOTS

Login Page:



Video :



CHAPTER 8
CONCLUSION

In conclusion, the proposed framework for real-time traffic analysis offers a promising solution to the challenges of urban traffic congestion. By integrating machine learning, big data analytics, and IoT technologies, the system provides a comprehensive approach to monitoring and managing traffic flow in real time. Through dynamic data collection from traffic sensors, GPS devices, and social media, this model not only detects current bottlenecks and optimizes signal timings but also predicts potential congestion patterns. This capability enables transportation authorities to make informed, immediate decisions, leading to more efficient traffic management and enhanced mobility within cities.

The results demonstrate that our approach can significantly reduce average travel times, improve overall traffic flow, and increase road safety, all of which contribute to a smoother urban experience for commuters. Moreover, the reduction in congestion also has environmental benefits, such as decreased emissions from idling vehicles, supporting a more sustainable urban future. This work highlights the potential for real-time traffic analysis systems to play a pivotal role in the transformation of urban transportation, paving the way for smarter, more resilient cities that can adapt to the evolving demands of mobility

REFERENCES

- ☐ Duckett, J. (2011). HTML and CSS: Design and Build Websites.

  Wiley.
- ☐ Duckett, J. (2014). JavaScript and jQuery: Interactive Front-End Web Development. Wiley.

- ☐ Font Awesome Icons. Retrieved from Font Awesome.

- ☐ W3Schools. (n.d.). HTML, CSS, JS. Retrieved from W3Schools.

- ☐ YouTube. (n.d.). PHP, MySQL Tutorials. Retrieved from YouTube.

- ☐ Haverbeke, M. (2014). Eloquent JavaScript. No Starch Press.