Software Requirements Specification (SRS)

Smart Traffic Management System

1. Introduction

1.1 Purpose of this Document

The main aim of this document is to clearly outline the requirements for developing the Smart Traffic Management System. This document will act as a guide for developers, stakeholders, and users to understand the system's functionalities and features. By ensuring all aspects are detailed, it will help in designing and developing a robust system that meets everyone’s needs.

1.2 Scope

Traffic congestion is one of the biggest challenges faced in urban areas. The Smart Traffic Management System (STMS) is designed to tackle this problem using advanced technologies such as IoT (Internet of Things), AI (Artificial Intelligence), and real-time data analysis.

The system’s goals include:

- Efficient Traffic Flow: Adjust traffic signal timings dynamically to reduce waiting times for vehicles.
- Emergency Management: Give priority to ambulances, fire trucks, and other emergency vehicles by creating green corridors.
- Scalability: The system can easily adapt to growing cities and increasing numbers of vehicles.
- Integration with External Systems: Provide APIs to connect with other systems like Google Maps for better navigation.

1.3 Overview

The Smart Traffic Management System is made up of the following components:

- A central server to manage all traffic data.
- Edge devices like smart traffic signals equipped with IoT sensors.
- Vehicle transponders to detect and prioritize emergency vehicles.
- A mobile app that users can access to get live traffic updates and suggested routes.

The design is modular, meaning new features can be added easily, and it provides data insights to improve traffic management continuously.

2. General Description

2.1 Product Perspective

This system will fit into the current traffic infrastructure without requiring major changes. By using IoT sensors and AI algorithms, the system will collect, analyze, and act on real-time traffic data to manage congestion efficiently.

2.2 User Characteristics

- Drivers: Everyday commuters will use a mobile app to get real-time traffic updates and optimized routes.
- Traffic Authorities: Officials will monitor traffic and control signals using a web dashboard.
- Emergency Services: Ambulances, fire trucks, and other emergency vehicles will get prioritized green signals for faster travel.

2.3 Features

- Real-Time Traffic Monitoring: Analyze traffic density using sensors and cameras.
- Emergency Vehicle Priority: Detect ambulances and fire trucks within 500 meters and create a green corridor for them.
- City-Wide IoT Integration: Connect with other IoT devices in the city for seamless traffic management.
- Predictive Analytics: Use historical data to predict and prevent future traffic congestion.

2.4 Benefits

- Reduced Travel Time: Minimize waiting at traffic lights and ensure smoother commutes.
- Improved Emergency Responses: Help emergency vehicles reach their destinations faster.
- Cleaner Environment: Decrease air pollution by reducing vehicle idling time.

3. Functional Requirements

3.1 Traffic Signal Control

- Automatically manage signal timings based on real-time traffic density.
- Allow manual override in case of emergencies or technical issues.

3.2 Emergency Vehicle Detection

- Identify emergency vehicles within a 500-meter range using vehicle transponders.
- Prioritize these vehicles by creating green corridors.

3.3 Data Analytics

- Provide insights into traffic patterns using predictive analytics.
- Suggest traffic adjustments based on historical data.

3.4 Mobile App Features

- Show real-time traffic conditions and suggest optimal routes.
- Allow users to report traffic incidents like accidents or jams.

4. Interface Requirements

4.1 User Interfaces

- A web dashboard for traffic authorities to monitor and manage traffic.
- A mobile app for drivers to get real-time updates and navigation.

4.2 Software Interfaces

- APIs to integrate with navigation systems like Google Maps.

4.3 Hardware Interfaces

- IoT sensors at intersections to monitor traffic density.
- Vehicle-mounted transponders for emergency vehicles.

5. Performance Requirements

- The system must process traffic data and adjust signals within 3 seconds to ensure real-time functionality.
- It should handle up to 1,000 simultaneous connections efficiently.
- Ensure 99.9% uptime for reliable operation.

6. Design Constraints

- Must work seamlessly with the existing traffic signal infrastructure.

- Comply with local government regulations for traffic management systems.

7. Non-Functional Attributes

7.1 Security

- All data transmissions will be encrypted to ensure security and prevent misuse.

7.2 Scalability

- The system will be designed to support future expansions, such as adding more IoT devices.

7.3 Reliability

- Backup systems will be in place to minimize downtime and ensure continuous operation.

7.4 Portability

- The mobile app will be compatible with both Android and iOS devices.

8. Preliminary Schedule and Budget

8.1 Development Timeline

The project will be completed in 2 months with the following milestones:

- Requirement Analysis: 1 week.
- System Design: 2 weeks.
- Development: 4 weeks.
- Testing and Deployment: 1 week.

8.2 Estimated Budget

The total cost of the project is ₹40,00,000, which will cover:

- Hardware (IoT sensors, transponders): ₹20,00,000
- Software Development: ₹15,00,000
- Miscellaneous Costs: ₹5,00,000