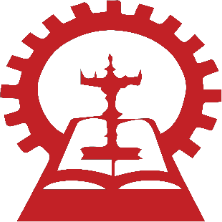
### **“INTELLIGENT TRAFFIC MONITORING SYSTEM”**

**Synopsis of Major Project**

**Submitted to:**

****

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**TECHNOCRATS INSTITUTE OF TECHNOLOGY & SCIENCE, BHOPAL (M.P.)**

**SESSION: 2025**

**Submitted by:**

**Abhimanyu Payasi (0192CS211003)**

**Ankit Kumar (0192CS211028)**

**Manish Kumar (0192CS211081)**

**Rahul Kumar (0192CS211121)**

**“INTELLIGENT TRAFFIC MONITORING SYSTEM”**

**Software Requirements Specification**

**Document**

**Table of Contents**

1. Introduction 3

1.1 Purpose 3

1.2 Scope 3

1.3 Definitions, Acronyms, and Abbreviations 3

1.4 References 3

1.5 Overview 3

2. The Overall Description 4

2.1 Product Perspective 4

2.1.1 Interfaces 5

2.1.2 Hardware Interfaces 5

2.1.3 Software Interfaces 5

2.1.4 Memory Constraints 5

2.1.5 Operations 5

2.1.6 System Adaptation Requirements 5

2.2 Product Functions 6

2.3 User Characteristics 6

2.4 Constraints 7

2.5 Assumptions and Dependencies 7

2.6 Apportioning of Requirements 7

3. Specific Requirements 8

3.1 Functions 8

3.2 Logical Database Requirements 9

3.3 Design Constraints 9

3.4 Software System Attributes 9

3.4.1 Security 9

3.5 Organizing the Specific Requirements 10

3.5.1 System Mode 10

3.5.2 User Class 10

3.5.3 Objects 10-11

# 1. Introduction

## 1.1 Purpose

This document designed for give a complete description of the requirements for the Traffic Monitoring Systemt 1.0 software (ITMS`1+.0). It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team.

## 1.2 Scope

This software is designed to simulate work of traffic lights with introducing algorithms that will improve its performance. This system can be used by the Ministry of Transport and Road Safety or the municipality. The traffic lights will be improved by collecting information and transmitting it in real time, which will lead to the correct distribution of traffic and reduce waiting time on the roads. In the first version of the program, a model will be created to simulate the work of traffic lights on the basis of traffic data already collected, which will affect the change in the work of traffic lights in the interval between two intersections. Our task will be to show the user using the graphical interface that the waiting time is reduced and that the traffic light works optimally in relation to this situation on the road.

## 1.3 Definitions, Acronyms, and Abbreviations.

* **Python**: Main programming language for building system logic.
* **venv**: For managing Python packages and dependencies in an isolated environment.
* **pygame**: Visualizes traffic flow and behavior simulations.
* **YOLO**: Real-time vehicle detection system using deep learning.
* **CSV/Database**: Stores traffic statistics for analysis and predictions.

## 1.4 References

* *India traffic regulations 2007*<https://morth.nic.in/carriage-road-act-2007>
* Traffic rules & Regulation Book

<https://www.indiacode.nic.in/bitstream/123456789/19026/1/motor_transport_workers_act_1961.pdf>

* Java Platform (JDK) 13   
  <https://www.oracle.com/technetwork/java/javase/downloads/index.html>
* MySQL Platform  
  <https://www.mysql.com/downloads/>
* NetBeans Java IDE

## 1.5 Overview

The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.

The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

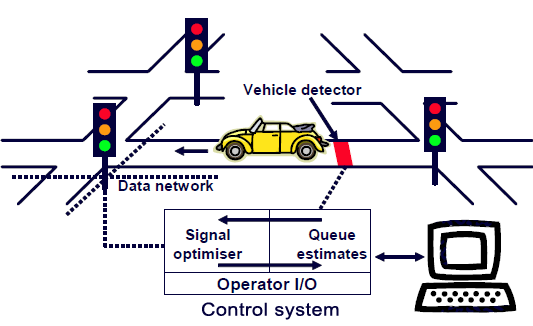
Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

# 2. The Overall Description

## 2.1 Product Perspective

Smart Traffic Light is a new system, that could replace the current simple system placed on majority of traffic lights, for managing color changing of traffic lights based on real time information of road traffic. The context diagram in Figure 1 illustrates the external entities and system interfaces. The system is expected to evolve, ultimately allowing for complete improving of the traffic lights work, specified more clearly in STL 1.0 Vision and Scope.

Fig. 1



### 2.1.1 Interfaces

In this project, a simulation will be provided that will illustrate the operation of the STL system using the following interfaces:

GUI - the interface provides the ability to select the type of user, select various traffic modes, open ready-made data about what is happening on the road, save data to a database, and most importantly, show the operation of the STL system that will solve the problem chosen by the user.

### 

### 2.1.2 Hardware Interfaces

* The system will be used on the Windows platform.
* Development and illustration will be supported by Java technologies.

### 2.1.3 Software Interfaces

* MySQL Workbench 8.0 CE  
  The application stores project data in SQL format enable easy integration with 3rd party applications.
* MS Excel 2007  
  The application allows import / export a ready data of road traffic from / to MS Excel sheet via CSV data format.

### 2.1.4 Memory Constraints

For optimal use of the program, a computer with at least 1 GB of RAM is required, and the user must provide at least 512 MB of secondary memory for installing the program by our prediction.

### 2.1.5 Operations

The user will be required to enter all the necessary data before starting the algorithm, such as traffic congestion, color states of traffic lights and the number of incoming cars.

The system will be able to restart in the event of a long wait or a malfunction.

### 2.1.6 System Adaptation Requirements

For proper operation of the system, the user will need to use the original version of the software, as well as provide all the required parameters, a list of which is provided in the paragraphs above.

## 2.2 Product Functions

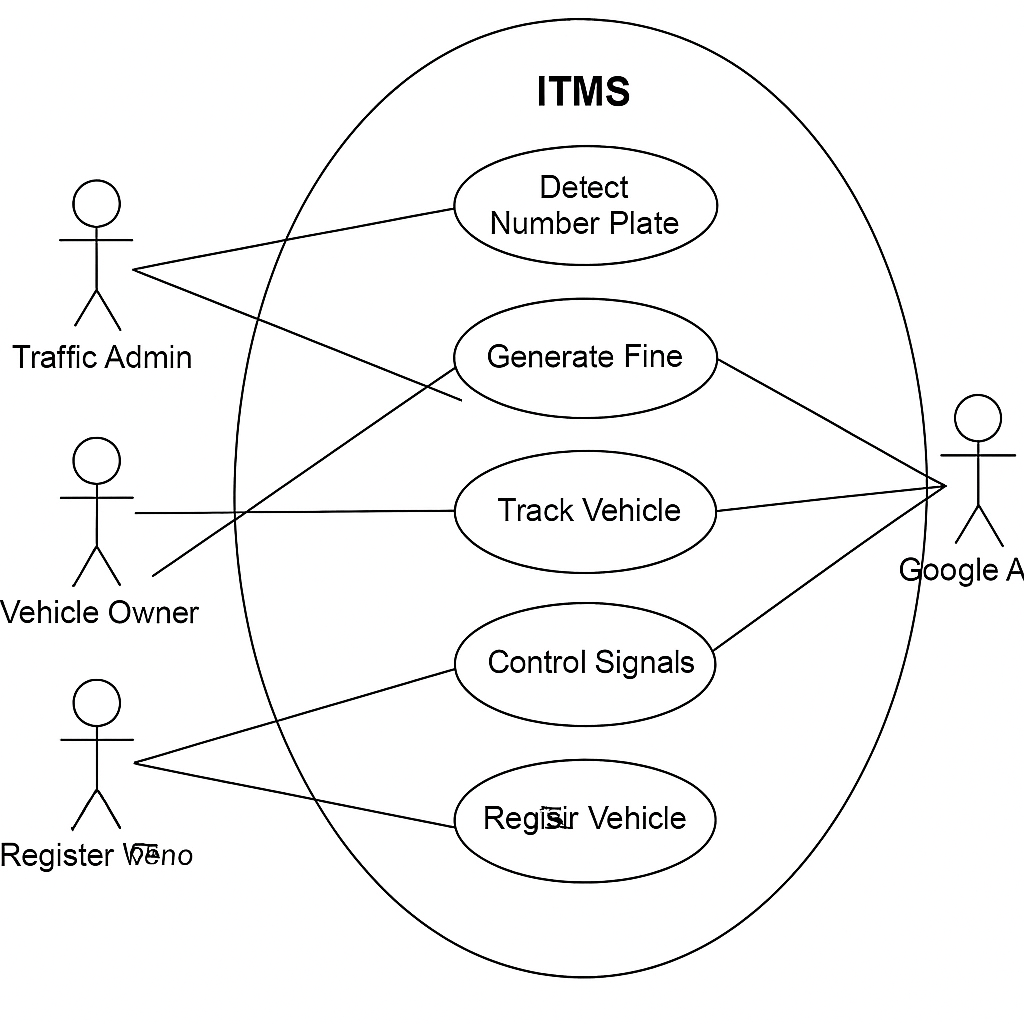
User - Use Case illustration

Fig. 2

* User selection - is a function that will allow the user to choose the type of his further work with the system.
* Opening files - is a function for opening ready-made traffic data. This data contains the number of cars on the road, the state of traffic lights.
* Data lottery - instead of entering data manually, the computer itself will choose the state of movement and traffic lights.
* Data entry - the user will manually enter all the data necessary for the system to work.
* Start of the system - only after the user selects all the data about the state of the road section he will be able to start the system.
* Saving results - after starting the system, the user will be able to save the results of the algorithm to the database.

## 2.3 User Characteristics

Users of the system will be people working in the ministry of transport, the municipality or the police. Work with the program assumes that users will be qualified specialists in the field of road traffic. Use of the system is permitted to persons who possess the following documents:

* driver license
* high school education
* form of vision correctness

## 2.4 Constraints

When developing the system, traffic rules must be taken into account in order to avoid not correct calculating of the system and we are forced to assume that all calculations will be performed on a sufficiently strong computer that can take on huge amounts of data that it will have to analyze in real time and will not lead to system failures and its collapse.

Also, the system must be safe and bear in mind that the switching of traffic light colors should occur at a speed that is understandable to a person, in order to avoid cases when the driver does not have time to react to changes in traffic light colors.

## 2.5 Assumptions and Dependencies

* It is assumed that the system must cope with a large amount of information about what is happening on the roads.
* It is assumed that the data that the system will receive is close to reality.
* It is assumed that all drivers have traffic rights and are familiar with traffic rules, as well as they must comply with them.

## 2.6 Apportioning of Requirements

At this stage of development, the main goal is to implement the first version of the system, which will receive ready-made data on traffic density on the roads. Real-time data collection will be performed in future versions.

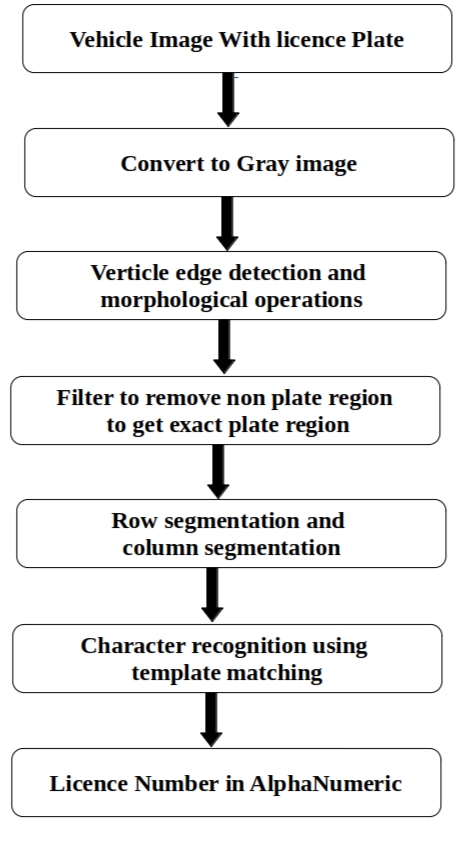
Also in this version, the task will be limited to an example from two crossroad. For us, the primary task is the implementation of a smart change in the work of the traffic light in connection with traffic information.

# 3. Specific Requirements

## 3.1 Functions

* The system shall allow traffic-light switch between 3 colors: red, yellow and green.
* The system shall allow traffic light to direct several types of driving directions: straight only, right only, left only, straight and right only, straight and left only, straight right and left only.
* The system shall allow traffic-light count the amount of cars and estimate the speed of cars on its road in each time cycle.
* The system shall allow crossroad contains between 1 to 4 traffic-lights.
* The system shall allow crossroads can send messages to its traffic-lights.
* The system will receive new information every 10 seconds from the crossroads on its traffic lights, roads and vehicles.
* The system for each crossroad will calculate the timing of its traffic lights.
* The system will enable to save the data of the crossroads to database before and after the calculation process (algorithm for each time cycle) is executed.
* The system will assign unique id for each object: crossroads, traffic lights, vehicles, roads.
* The system will allow the user to choose enter old data from database about crossroads details by time, in order to watch the output of algorithm.
* The system will allow the user to choose manually enter data about certain scenarios on the crossroads.
* The system will allow the user to choose to enter new data by the Excel file on certain crossroad scenarios.
* The system will allow the addition of new crossroads, traffic-lights, roads, vehicles.
* The system will allow the user to enter data randomly and automatically on crossroads, traffic-lights, roads, vehicles (and what they contain).
* The system will allow a visual simulation of the solution calculated for any scenario.
* The system will allow to retrieve information from the database from the analysis tables for analysis.

**3.2 Logical Database Requirements**



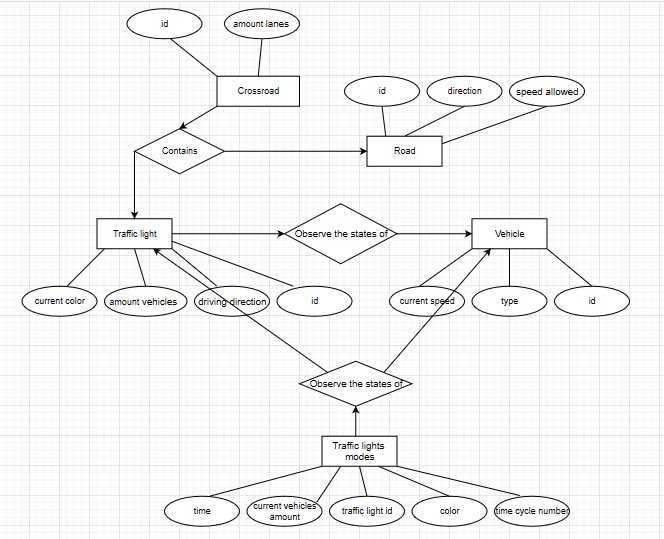


Fig. 3

**3.3 Design Constraints**

* GUI shall be composed using Python libraries

**3.4 Software System Attributes**

3.4.1 Security

* Each program will be installed locally on the computer; it will not have Internet access.
* MySQL - will be a program for saving information. Their system support guarantees us the reliability of information storage.

**3.5 Organizing the Specific Requirements**

3.5.1 System Mode

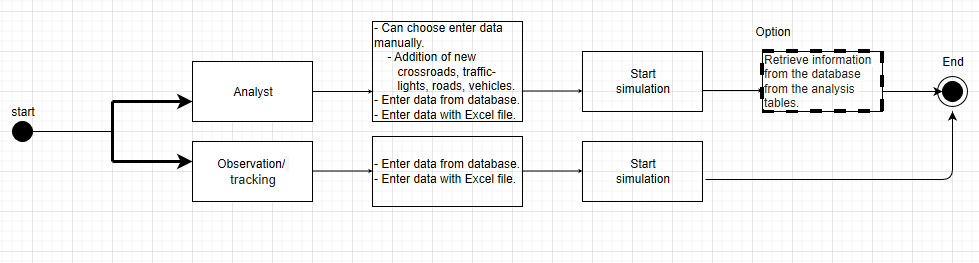


Fig. 4

3.5.2 User Class

The system contains a single user (no login required). This user can perform all processes and access the entire system (no permission restriction).

The user can choose whether they want to analyze data and do extensive and complex tests on the system (manual data entry, manually adding objects) and finally get analyzed data and conclusions from the analysis, or check previously entered data (run the simulation) or new data by Excel file to keep in the database (Or will not be saved) and to conclude simple visual and final results.

3.5.3 Objects

Traffic-light:

* Attributes:

Id, now time, vehicle amount, current color, driving direction, my crossroad, my roads[ ], my neighbors[ ]

* Functions:
* Send message to my neighbor – the traffic light can send specific information about traffic controlling and timing to other traffic lights.
* Get id
* Get/set time
* Get/set current color
* Write to DB
* Read from DB
* Update on time cycle – the details of traffic light will be update after all time cycle.

Crossroad:

* Attributes:

Id, my traffic-lights[ ], my neighbors[ ]

* Functions:
* Send message to my neighbor (same as traffic light "send message to my neighbor")
* Send message to my traffic-lights
* Get id
* Write to DB
* Read from DB
* Update on time cycle – the details of traffic light will be update after all time cycle.

Road:

* Attributes:

Id, direction, speed allowed

* Functions:
* Get id
* Get direction
* Get speed allowed
* Write to DB
* Read from DB

Vehicle:

* Attributes:

Id, type, current speed, current road

* Functions:
* Get id
* Get/set current speed
* Get/set current road
* Get type
* Write to DB
* Read from DB
* Update on time cycle

