Software Requirements Specification

for

Traffic Light Optimization
Imagine Interactive Systems (Iminsys)

Prepared by Team Alpha:

1. TK Lebelo u15209190 (team leader)
2. RW van Graan u16040865
3. R Rammbuda u16207948
4. OM Mogoathle u14254922
5. TR Sithole u15360441

Submitted to : Stacey Baror
Avinash Singh
Eunice Hammond

May - November 2020

Contents

1 Introduction **4**
- 1.1 Purpose . . . 4
- 1.2 Project Scope . . . 4
  - 1.2.1 Project Name . . . 4
  - 1.2.2 Objectives, Goals and Benefits . . . 4
- 1.3 Definitions, Abbreviations and Acronyms . . . 4

2 User Characteristics **5**
- 2.1 Education Level . . . 5
- 2.2 Experience . . . 5
- 2.3 Technical Skills . . . 5

3 Functional Requirements **6**
- 3.1 Use Case Description and Priority . . . 6
- 3.2 Requirements . . . 6
- 3.3 Subsystems . . . 7
  - 3.3.1 Data Processor . . . 7
  - 3.3.2 Road Network . . . 7
  - 3.3.3 Moving Objects . . . 7
  - 3.3.4 Traffic Controller . . . 7
  - 3.3.5 GUI Simulator . . . 7
- 3.4 Use Case Diagram . . . 8

4 Traceability Matrix **9**
  - 4.0.1 Functional Requirements . . . 9
  - 4.0.2 Use Cases . . . 9

5 Domain Model **10**
- 5.1 Main Classes . . . 10
  - 5.1.1 User Class . . . 10
  - 5.1.2 Traffic Class . . . 10
  - 5.1.3 Simulator Class . . . 10
  - 5.1.4 Optimizer Class . . . 10
- 5.2 Subclasses . . . 10
  - 5.2.1 Road Artefacts Class . . . 11
  - 5.2.2 Road Network Class . . . 11

6 Quality Requirements **13**
6.1 High Priority . . . 13
6.1.1 Auditability/Monitorability . . . 13
6.1.2 Cost . . . 13
6.1.3 Flexibility . . . 13
6.1.4 Reliability . . . 13
6.1.5 Scalability . . . 13
6.2 Low Priority . . . 13
6.2.1 Performance . . . 13
6.2.2 Security . . . 14
6.2.3 Usability . . . 14

References **15**

1 Introduction

1.1 Purpose

The purpose of this system is to receive time series data of an arbitrary road intersection, and use artificial intelligence to generate (and simulate) an optimal schedule for the traffic lights. This result will be able to be viewed on the system's accompanying web application. The web application would also be used to upload time series data to the system's database, and adjust the AI's parameters.

1.2 Project Scope

1.2.1 Project Name

This system is referred to as "Traffic Light Optimization". The name was decided on by our clients - Imagine Interactive Systems (Iminsys).

1.2.2 Objectives, Goals and Benefits

Our goal with this project is to create a working web app that allows a user to upload real-world or artificial traffic light schedules (time series data), and have an AI generate an optimal schedule based on that data. They would also be able to watch a simulation of the generated data.

This system would be highly beneficial for real-life road intersections. The data could be used to mitigate traffic during rush hours, such as when people are going to/coming home from work or school. The optimal schedules would ensure that as many vehicles as possible are allowed to safely pass through an intersection at a time.

1.3 Definitions, Abbreviations and Acronyms

- InfluxDB: An online database managed by InfluxData, which was specifically built to handle time series data.[2]
- TraCI SUMO: TraCI stands for "**Tra**ffic **C**ontrol **I**nterface" and SUMO stands for "**S**imulation of **U**rban **MO**bility". SUMO is open source software that allows for simulation of traffic, created by Eclipse Foundation. TraCI is its online interface for adjusting simulations and retrieving data.[1]

2 User Characteristics

This system would, ideally, only have one user, which would be the person or company operating the simulations. This system is not intended for general or public use, and would primarily be used to generate ideal intersection schedules, which could then be given to, for example, the South African Department of Transport to implement in actual intersections.

2.1 Education Level

The user does not need to be an expert in how intersections work, but they must be able to collect time series data that can be used.

2.2 Experience

Experience with formatting time series data in a way that can be understood by the system is necessary. Being able to adjust the AI's parameters is also important, but not crucial.

2.3 Technical Skills

The user of this system must be able to:

1. Navigate a web page to upload time series data.
2. Adjust the AI's simulation parameters via the web page.
3. View a graphical representation of the simulation's outcome.
4. Download the results via the web page in a time series format.

3 Functional Requirements

3.1 Use Case Description and Priority

The following use cases are ranked from most important to least important. Their relationships with the requirements are detailed in the use case diagram and traceability matrix below.

1. Upload/Store Time Series Data: The user will be able to upload and store time series data in the system's database, via the web app. The data will be stored in an InfluxDB database.

2. Enable Artificial Intelligence: Once time series data has been uploaded, the user can turn on the AI so that it can produce a result.

3. Retrieve Time Series Data: The user will be able to retrieve the optimal time series data generated by the AI, for use in real traffic lights.

4. Upload Configuration Files: The user will be able to upload text files that will configure the AI and the TraCI SUMO simulation.

5. View Simulated Result: Using TraCI SUMO, the result generated by the AI will be displayed in a visual simulation.

3.2 Requirements

The following requirements are ranked from most important to least important.

1. The user must be able to access, view and interact with the system via a website interface (web app).

2. The system must be able to take time series data and analyse it with artificial intelligence for optimal traffic light schedules.

3. The system must be able to store and retrieve time series data from a database.

4. The AI's parameters must be able to be altered via a configuration file provided by the user.

5. A simulation should be generated and displayed from the results given by the AI.

3.3 Subsystems

3.3.1 Data Processor

- Provides CRUD (Create, Read, Update, Delete) function to the database through API call. Data:
  - Time Series Data from traffic network
  - Configuration files and simulation results
- The time series data should be stored in InfluxDB
- Performs analysis on time series data
- Provides statistics of simulation results

3.3.2 Road Network

- Maps out the road network architecture using data structures and algorithm
  - Edges: Streets
  - Nodes: Intersections/Junction
- The network can be created by input file (configuration file)

3.3.3 Moving Objects

- Models the behaviour of moving objects within a traffic network
  - Vehicle [Truck, Bus, Taxi, Car]
  - Train
  - Bicycle
  - Pedestrian
- The objects can be initiated by input file (configuration file)

3.3.4 Traffic Controller

- Controls the traffic flow by manipulating the traffic lights
- Uses K53 road rules to simulate the flow allowing the right of way amongst the different objects.
- Uses threads to simulate the traffic light behaviour

3.3.5 GUI Simulator

- Maps the traffic flow simulation onto a GUI for real time rendering
- Implemented using TraCI SUMO

3.4 Use Case Diagram

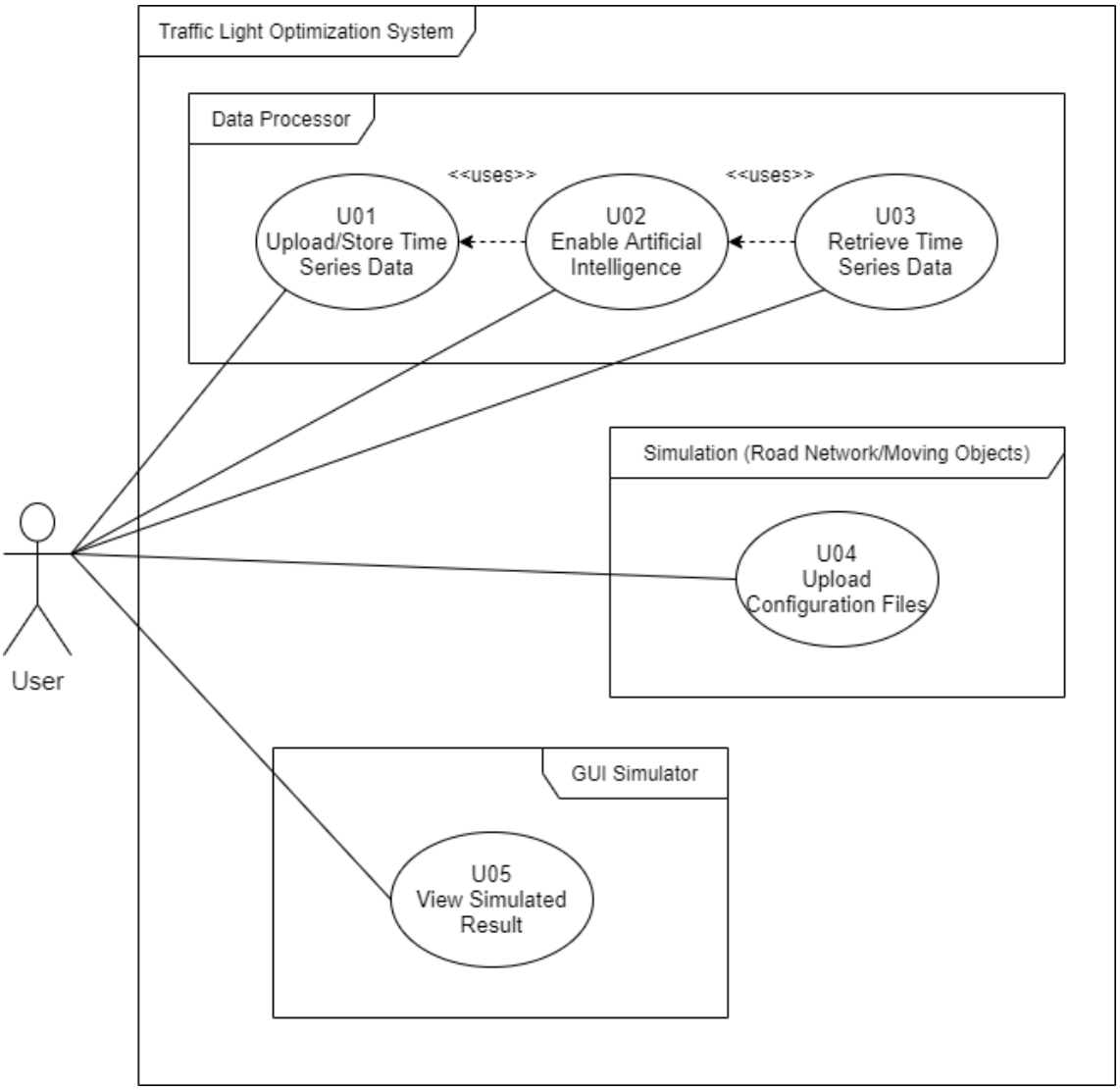

Figure 3.1: Use case diagram

4 Traceability Matrix

	Priority	U01	U02	U03	U04	U05
R01	1	x	x	x	x	x
R02	2		x	x		
R03	3	x		x	x	
R04	4				x	
R05	5					x
Priority		1	2	3	4	5

Table 4.1: Traceability matrix

4.0.1 Functional Requirements

- R01: User access and interaction with the web app.
- R02: Analysis and optimization of time series data by the system.
- R03: Storage and retrieval of time series data in database.
- R04: Upload and usage of configuration files.
- R05: Visual simulation generation based on results provided by AI.

4.0.2 Use Cases

- U01: Upload/Store Time Series Data
- U02: Enable Artificial Intelligence
- U03: Retrieve Time Series Data
- U04: Upload Configuration Files
- U05: View Simulated Result

5 Domain Model

5.1 Main Classes

The domain model below (Figure 5.1) consists of four main classes which interact with each other as follows:

5.1.1 User Class

The user does most of their interaction through the web app. Via the web app, they can upload and retrieve traffic data (optimized or unoptimized) and configure the road network. The user can also view the generated simulation of the system via the web app.

5.1.2 Traffic Class

The traffic class consists of time stamped road artefacts (such as pedestrians, cars and buses) that are derived from data collected from real life traffic lights. For example, this class may contain an entry for 14 June, 2021 at 13:00 that states that there were six cars and two buses waiting at a particular intersection.

5.1.3 Simulator Class

The simulator (TraCI SUMO) simulates traffic data from the traffic class on road networks provided by the user. The simulation is displayed on the web app for the user to see.

5.1.4 Optimizer Class

The optimizer uses intersections in the road network and unoptimized data from the traffic class to generate optimal schedules for traffic lights.

5.2 Subclasses

Other important classes that the four main ones interact with are as follows:

5.2.1 Road Artefacts Class

Road artefacts are representations of different kinds of vehicles and people who may be using a particular intersection. Pedestrians represent people who walk through intersection crossings, domestic vehicles represent cars, bicycles and motorcycles, and buses represent buses and minibus taxis.

5.2.2 Road Network Class

The road network class represents a network of roads and intersections. All traffic exists on a road network of some kind.

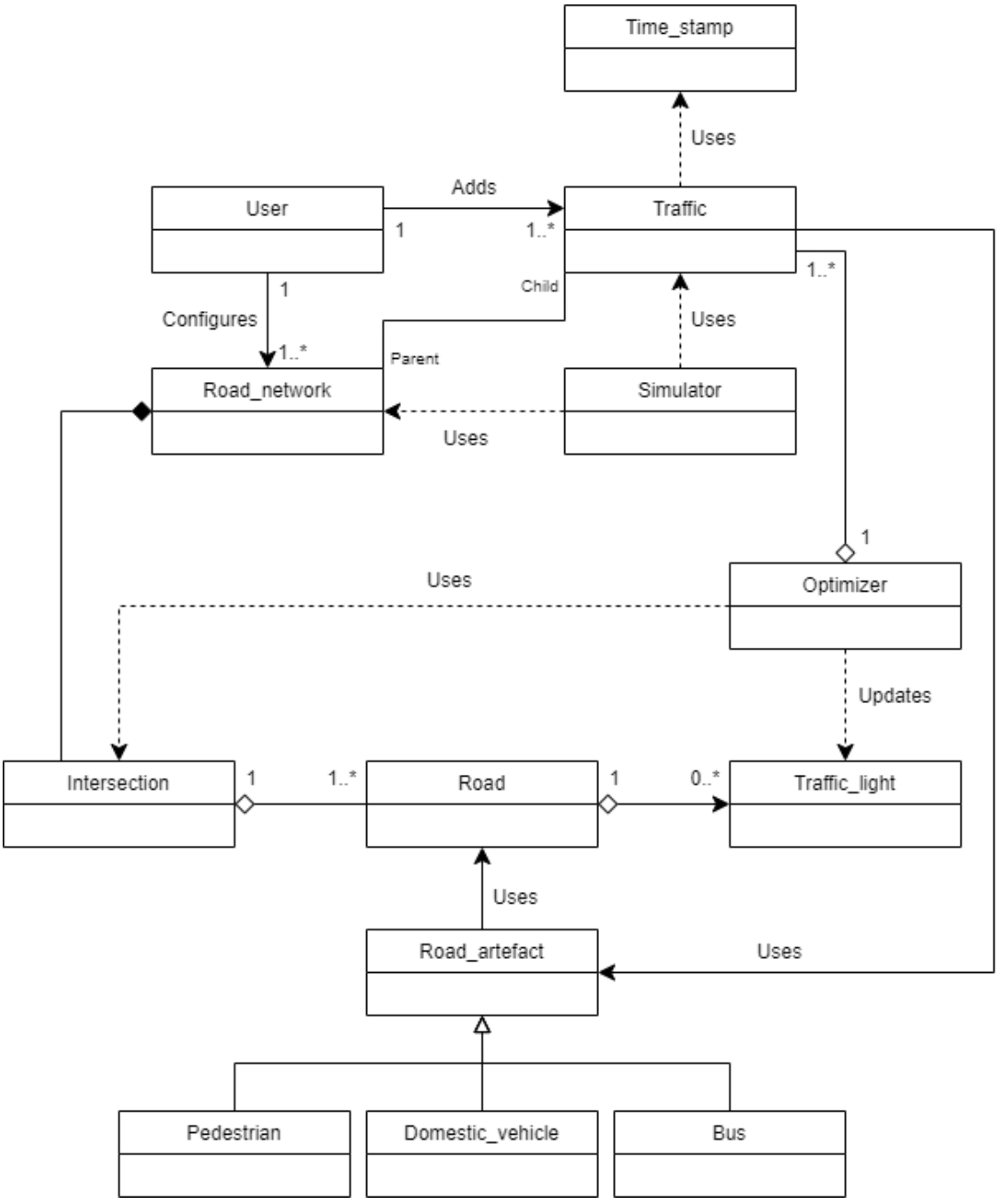

Figure 5.1: Domain model

6 Quality Requirements

The quality requirements for this system can be divided into two categories, high priority and low priority. Note that even though more focus is given on high priority requirements, the low priority ones should still be taken into account and not ignored.

6.1 High Priority

6.1.1 Auditability/Monitorability

The user will have some idea of the inner workings of the system, as they will be able to configure the various parts with configuration files.

6.1.2 Cost

It is emphasized in the specification from the clients that we should use open source and free software for the system.

6.1.3 Flexibility

The user should be able to fine-tune the parameters of the artificial intelligence and the simulation, if they choose to, and see the change in results.

6.1.4 Reliability

As this system could potentially be used to schedule real-life traffic lights, it is important that the result the artificial intelligence produces is always efficient and safe.

6.1.5 Scalability

One of the optional requirements given by the clients is that the system could potentially be expanded to manage a whole network of intersections, instead of just one.

6.2 Low Priority

6.2.1 Performance

It is not very important that a result is produced instantaneously. It is only important that a correct result is produced in a reasonable time. However, if we can get the system to work faster while still being correct, that would be preferable.

6.2.2 Security

No sensitive personal information is stored in this system. It only stores time series data.

6.2.3 Usability

This system is not for everyone to use, and it requires basic knowledge of time series data, artificial intelligence and simulation to use properly.

References

[1] Eclipse Foundation. *TraCI SUMO*. URL: https://sumo.dlr.de/docs/index.html. (accessed: 06.06.2020).

[2] InfluxData. *InfluxDB*. URL: https://www.influxdata.com/. (accessed: 06.06.2020).