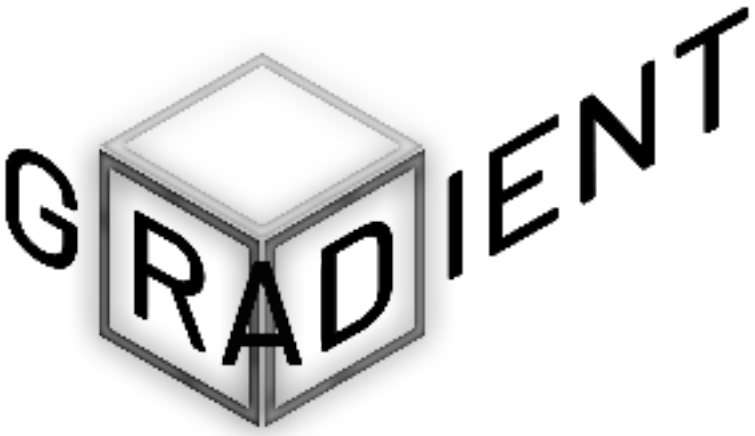

Software Requirements Specification

COS 301 Capstone Project: Demo 1

Members:

Maria Petronella Laura-Lee Strydom	*u04974359*
Mohammed Gangat	*u17058849*
Rahul Kapoor	*u16034130*
Thiveshan Pillay	*u15007911*
Jared Gratz	*u16054972*

Team Gradient

LightBot Adaptive Traffic Control System

Contents

1. Introduction

1.1 Purpose

The purpose of LightBot is to minimize the delays that are caused by traffic congestions at various intersections. The LightBot system will be used to prove a concept that the addition of a machine learning algorithm will develop a good policy for which to control the traffic lights at a given intersection. This policy should minimize the possibility of traffic congestion of an intersection, thus having a positive impact on productivity and maximise flow of traffic, particularly during peak hours.

1.2 Scope

The scope of the system would be to provide simulations to demonstrate the effectiveness in the case of the implementation of a machine learning algorithm for the traffic lights of a given intersection layout. This machine learning algorithm will use statistical metrics from a simulated traffic flow and real-time traffic flow data. The system will be accessible through a web interface, specifically designed to control the state of the simulation.

1.3 Definitions, Acronyms and Abbreviations

1. API - Application Programming Interface
2. GUI - Graphical User Interface

1.4 Overview

The remainder of this document is structured as follows: Section 2 provides the User Characteristics of the system. In Section 3, the Functional Requirements are listed, along with Constraints, Use Cases, Use Case Diagrams and a list of the Subsystems. The Domain Model of the system is depicted in Section 4. Section 5 deals with the Quality Requirements of the system. Traceability Matrices pertaining to Use Cases and Subsystems are shown in Section 6. Finally, any works cited are referenced in Section 7.

2. User Characteristics

2.1 User Stories

I. As an administrator I would like to have the ability to set the privilege levels of other users, so that certain functions are restricted to authorized users.
II. As an administrator I would like the ability to remotely interact with the system, so that I can use the system on any web portal.
III. As an administrator I would like the ability to switch between automated control of the traffic lights and manual control, so that I can see the effectiveness of the reinforcement learning algorithm.
IV. As a regular user, I would like to have access to the system via a web portal, so that I can view the simulation, and see the effectiveness of the reinforcement learning algorithm on reducing congestion.
V. As an overall user, I would like the system to be modeled on a real-world intersection, so that I can see the benefits of having an reinforcement learning optimized traffic light system in the real world.

3. Functional Requirements

3.1 Requirements & Constraints

Requirements:

- R1: The system will gather traffic data
    - R1.1: The system will gather data from a real-world source for the simulation.
    - R1.2: The system will randomly generate data for the simulation.
- R2: The system will have a graphical user interface (GUI)
    - R2.1: The system will have a web interface for viewing the traffic simulation.
    - R2.2: The system will have a web interface for controlling the state of the traffic simulation.
        - R2.2.1: The interface will display a dashboard with statistics.
        - R2.2.2: The interface will display the simulation.
        - R2.2.3: The interface will display the controller options
- R3: The system will have user controller options for the traffic lights.
    - R3.1: The system will allow for manual control of traffic lights
    - R3.2: The system will allow for automated control of traffic lights
    - R3.3 The system will allow for switching between manual and automated control
        - R3.3.1: The controller can be changed on the interface.
        - R3.3.2: The controller change should notify the system.
- R4: The system will collect statistical metrics that describe the efficiency of the current traffic flow.
- R5: The system will provide an API for accessing the current state of the traffic system.
    - R5.1: The interface will use the API to retrieve state data.

- R6: The system will have controllers for the traffic lights.
    - R6.1: It will have at least one controller that actively controls the state of traffic lights.
    - R6.2: It will also have a fixed time controller.
- R7: The system should be able to allow management of the databases.
    - R7.1: An administrator can access and edit any data stored in the databases.

Constraints:

1. The GUI must be able to run in a web browser on a remote machine.

3.2 Use Cases & Diagrams

- U1: Access System GUI
    - U1.1 Register
    - U1.2 Log In
    - U1.3 Reset Password
- U2: View System GUI
    - U 2.1 View state of system
    - U 2.2 View simulated traffic flow
    - U 2.3 View active traffic controllers
    - U 2.4 Access settings
    - U 2.5 Log out of system
- U3: System Management
    - U 3.1 Manual control of traffic lights
    - U 3.2 Automated control of traffic lights
    - U 3.3 View controller states
    - U 3.4 Access simulation
- U4: User Database Management
    - U 4.1 View users
    - U 4.2 Add users
    - U 4.3 Delete users
    - U 4.4 Edit access
    - U 4.5 Revoke access
    - U 4.6 Activate user status
    - U 4.7 Deactivate user status
    - U 4.8 View user role
    - U 4.9 Edit user role
- U5: Simulation Control
    - U 5.1 Create real-world scenarios
    - U 5.2 View scenario results
    - U 5.3 Test emergency scenarios
    - U 5.4 Test situational scenarios
    - U 5.5 Change location & road layout

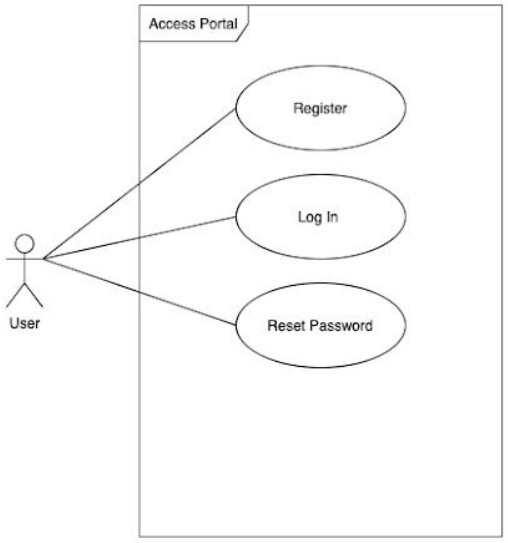

Figure 1: Use Case 1 Diagram

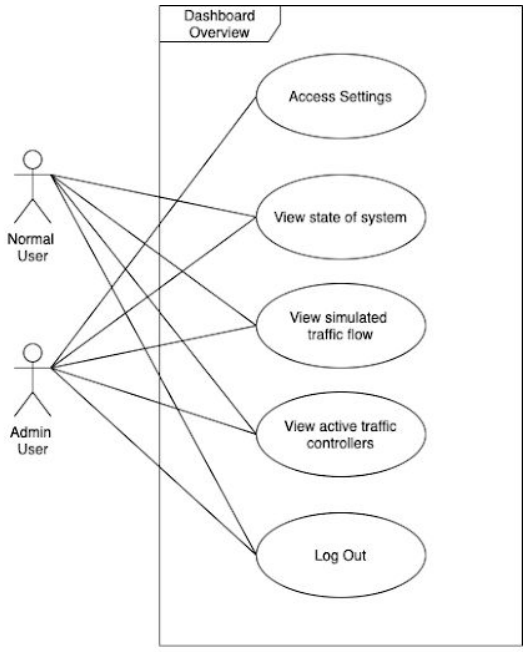

Figure 2: Use Case 2 Diagram

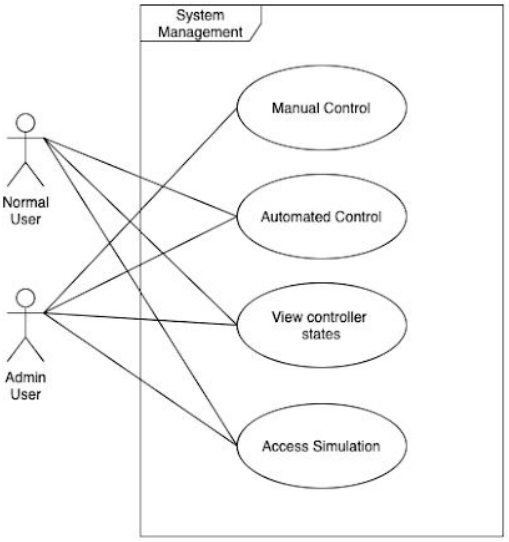

Figure 3: Use Case 3 Diagram

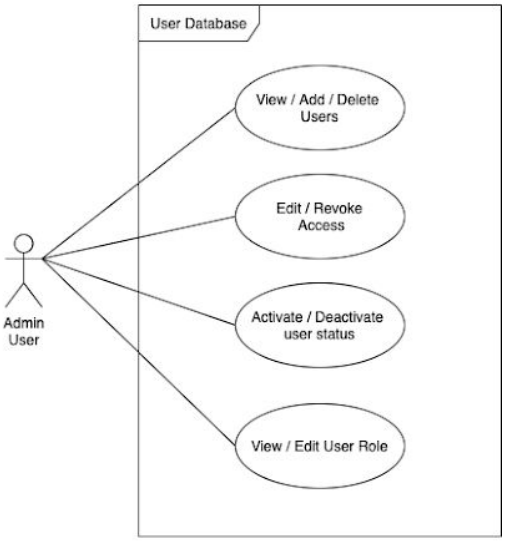

Figure 4: Use Case 4 Diagram

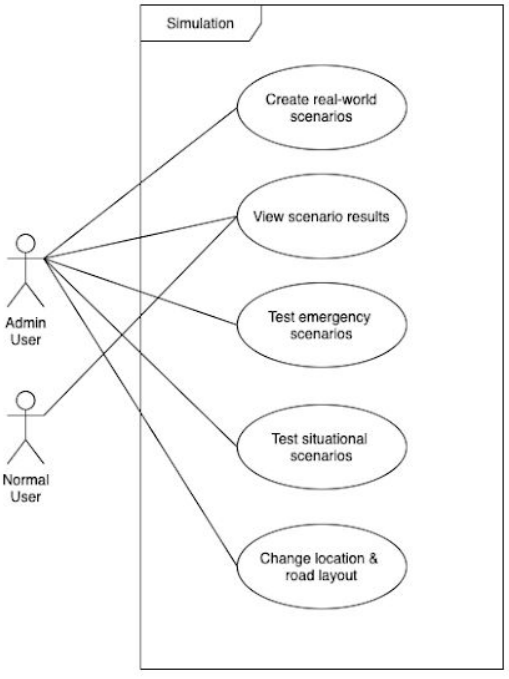

Figure 5: Use Case 5 Diagram

4. Domain Model

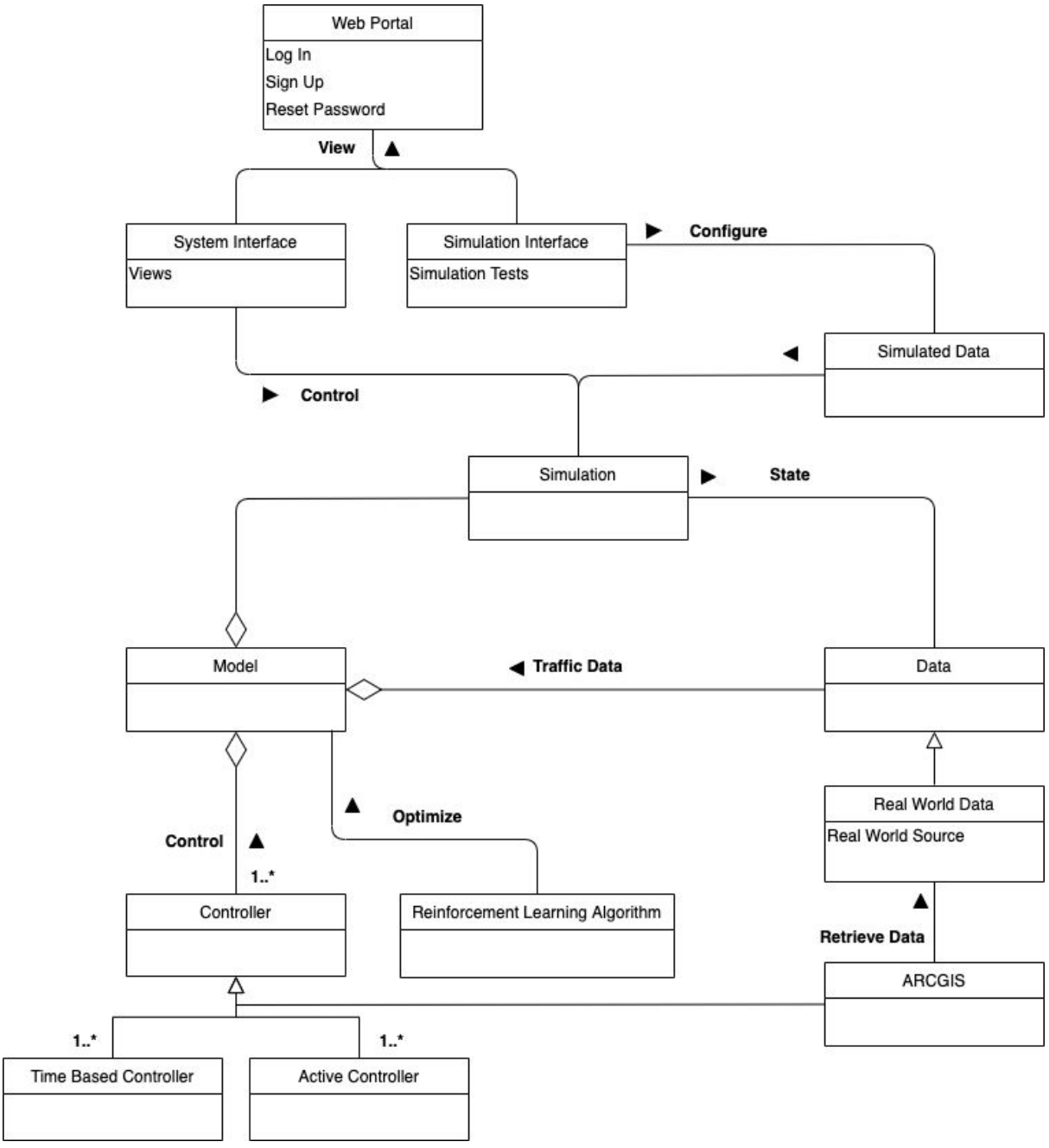

Figure 6: Domain Model of LightBot Adaptive Control System

5. Quality Requirements

1. Performance

- The response time of the system will be measured according to its real-time data processing and feedback to decide how to change the state of the traffic lights in order to maximize the flow of traffic at all times. This includes measuring the execution speed of the GUI, simulation and database at the same time.
- The capacity of the system is justified by the amount of average/maximum users that the system can support at once, where the amount of users at this time is fairly low. The only users to account for in this sense, would be the users from 5DT that would manage the system. No outside users will have access to this system.
- The simulation needs to ensure that simulated vehicle metrics, such as acceleration, are as accurate as possible so that the statistics of the optimized traffic light system have real world value.

2. Reliability

- The system needs to consistently provide accurate output under certain conditions .
- Proper computer networking will have an impact on the reliability of the system as well. The accessibility to the databases will be monitored and tested during different types of network connections to verify that the system responds correctly.

3. Security

- The system will grant each user secure access to the system management GUI, where each users' login data will be encrypted and stored securely on the Management database. Hashing and salting of login passwords will occur.
- The system management GUI, which is a web application, will adhere to having a secure connection over the internet, as well as server connection and api calls made.

4. Maintainability

- The system should be easy to update and maintain. If the layout of an intersection needs to be edited, the system will be able to accommodate such changes to the simulation.
- The dependability of the different components in the system will be kept minimal as excessive dependability has a negative impact on maintainability.
- The different components of the system will be kept separate in design, to ensure that future updates and maintainability of different modules can be accommodated.

5. Useability
    - The amount of interaction with the system from a user to accomplish a certain task will be kept minimal. Users prefer minimal effort to accomplish more tasks. This will positively affect the system's efficiency and user satisfaction.
    - User satisfaction will also be addressed by how easy the system will be to use, taking care of where design elements are placed and how the system reacts to the users decisions.

6. Traceability Matrix

6.1 Traceability Matrix for Use Cases

Use Case / Requirement	U1	U1.1	U1.2	U1.3	U2	U2.1	U2.2	U2.3	U2.4	U2.5	U3	U3.1	U3.2	U3.3	U3.4	U4 - 4.9	U5	U5.1	U5.2	U5.3	U5.4	U5.5
R1.																	X	X		X	X	X
R1.1																	X	X		X	X	X
R1.2																	X			X	X	X
R2.	X	X	X	X	X	X	X	X	X	X												
R2.1					X		X															
R2.2					X	X		X									X					
R2.2.1					X	X		X							X							
R2.2.2					X		X								X				X			
R2.2.3					X				X													
R3.											X	X	X	X	X							
R3.1											X	X										
R3.2											X		X									
R3.3											X	X	X									
R3.3.1											X	X	X	X								
R3.3.2											X	X	X									
R4.																	X	X		X	X	X
R5.																	X	X		X	X	X
R5.1																	X	X		X	X	X
R6.													X									
R6.1													X									
R6.2													X									
R7.																X						
R7.1																X						

Figure 7: Traceability Matrix for LightBot Use Cases & Requirements

7. References

Kung, D. C., 2014. *Object-Oriented Software Engineering: An Agile Unified Methodology*. New York: McGraw-Hill.