Final Version of User Requirement Specification

|  |  |
| --- | --- |
| A picture containing person, clothing  Description automatically generated  USER REQUIREMENT SPECIFICATION  City traffic Simulation Software | ABSTRACT  This document describes about the interface screens of traffic city Simulation Software Application and their description  Fontys University Applied Sciences  Traffic City Simulation Software |

[**Introduction**](#_heading=h.gjdgxs) **2**

[**Functional Requirements**](#_heading=h.30j0zll) **2**

[**Non-Functional Requirements**](#_heading=h.3znysh7) **2**

[**User case Diagram**](#_heading=h.2et92p0) **3**

[**User Interface Design**](#_heading=h.r23egyd7dovv) **3**

[**User Requirement Specification**](#_heading=h.tyjcwt) **4**

[I) Use case 1: Drag the crossing tools to the panel](#_heading=h.3dy6vkm) 4

[II) Use case 2: Starting the simulation](#_heading=h.1t3h5sf) 5

[III) Use Case 3: Stopping the Simulation](#_heading=h.4d34og8) 6

[IV Use case 4: Saving a project](#_heading=h.2s8eyo1) 6

[V) Use case 5: Loading a project](#_heading=h.17dp8vu) 7

[VI) Use Case 6: Displaying Traffic Graph](#_heading=h.3rdcrjn) 7

Table of Contents

[Introduction 2](#_heading=h.gjdgxs)

[Functional Requirements 2](#_heading=h.30j0zll)

[Non-Functional Requirements 2](#_heading=h.3znysh7)

[User case Diagram 2](#_heading=h.2et92p0)

[User Requirement Specification 3](#_heading=h.tyjcwt)

[I) Use case 1: Drag the crossing tools to the panel 3](#_heading=h.3dy6vkm)

[II) Use case 2: Starting the simulation 3](#_heading=h.1t3h5sf)

[III) Use Case 3: Stopping the Simulation 4](#_heading=h.4d34og8)

[IV Use case 4: Saving a project 5](#_heading=h.2s8eyo1)

[V) Use case 5: Loading a project 5](#_heading=h.17dp8vu)

[VI) Use Case 6: Displaying Traffic Graph 6](#_heading=h.3rdcrjn)

# Introduction

This document describes what The City Traffic Simulation application should do as specified in the section “Functional Requirements” and requirements which specifies how the City Traffic Simulation Application performs a certain function as mentioned in the “Non- Functional Requirements”.

It also gives an overview of the platform in which system will run and user interface of functional requirements as well as explain the scenarios how a user can perform certain functionalities as described in “Use Case”.

# Functional Requirements

→ The application should be able to create a city using multiple crossings

→ The application should be able to able to start a simulation in which the cars will move and obey the traffic light.

→ The application should be able to reset the simulation.

→ The application should be able to pause the simulation

→ The application should be able to save the data into a file.

→ The application should be able to load the file and read data from that file

# Non-Functional Requirements

→Performance

→Reliability

→Maintainability

→Availability

→Usability

# User case Diagram

A close up of a map

Description automatically generated

# User Interface Design

1.The main interface should contain a canvas where we can drag and drop crossing .

2. The main GUI contains two different types of crossings that can be added to the canvas by the user.

3.The GUI contains the following buttons that help in using the application

1. the start button that enables the user to start a simulation.
2. The paused button that enables the user to stop a simulation at run time .
3. The restart button that enables the user to restart a simulation at the end of an already runned simulation.
4. The save button that enable the user to save the result of a simulation into a file .
5. The print button that enable the user can print or load the data of an already runed simulation.

# User Requirement Specification

## I) Use case 1: Drag the crossing tools to the panel

Actor: User

Goal Level: Sea-level

Main success Scenario:

|  |
| --- |
| 1.User drags the first crossing tool to the canvas |
| 2. System displays crossing to the canvas |
| 3. User drags the second crossing tool to the canvas |
| 4. System joins the second crossing with first crossing and display to the panel |
| 5.User drag the third crossing to the canvas |
| 6. System joins the third crossing with first and second crossing and display to the crossings on the canvas. |
| 7. User drag the fourth crossing to the canvas . |
| 8. System joins the fourth crossing to the first, second, and third crossings and system displays a city with four crossings. |

Extension(s):

## II) Use case 2: Starting the simulation

Actor: User

Goa-level: Sea-level

Pre-conditions: The project already designed with the requirement design

Main Success scenario:

1. User clicks the start button

2. System displays moving cars on the first crossing (cars moving from North ,East, South and West,)

Extension(s):

1.1. User can not start the application

1.1.1. System informs the user that the application cannot be started.

1.1.2. User tries to restart the application.

1.1.3. Back to step 1 of the use case.

2)User does not want to start the simulation

1. User does not press the button ‘Start’.
2. User exits the application.
3. End of a use case.

## III) Use Case 3: Stopping the Simulation

Actor: User

Goa-level: Sea-level

Main Success scenario:

1. Application is already running

2. User then clicks the stop button

3. System stops the application

Extension(s):

* 1. Application is not running
  2. User performs the use case named ‘Starting the Simulation’.
  3. After the above-mentioned use case is finished, the user performs ‘Stopping the Simulation’ use case.
  4. End of a use case

2.1 The user cannot stop the simulation

2.1.1. User tries to click the ‘Stop’ button.

2.1.2. The simulation does not stop itself.

2.1.3. User then clicks the ‘X’ in the top right corner of the app.

2.1.4. The app exits and the simulation is stopped.

2.1.5. End of a use case.

2.2. The user does not want to stop the simulation

2.2.1. User does not press the button ‘Stop’.

2.2.2. The simulation continues until user wants it to stop.

2.2.3. User performs the use case ‘Stopping the Simulation’.

2.2.4. End of a use case.

## IV Use case 4: Saving a project

Actor: User

Goal-level: Sea-level

Main Success scenario:

1.User clicks the save button

2.System opens the file dialogue from the user desktop

3.User selects the folder and clicks the save button

4.System saves the project

Extension(s):

## V) Use case 5: Loading a project

Actor: User

Goa-level: Sea-level

Pre-conditions: The project already saved to the user desktop

Main Success scenario:

1.User clicks the load button

2.System opens the folder from the user desktop

3.User selects the project

4.System displays the project.

Extension(s):