**Software Requirements**

**Specification**

**for**

***Traffic Modeling System***

# Version 4.0

NWMSU

Feb 17, 2017

**Prepared by:**Sahil Nokhwal, Rupanandha Moori, Manikanta Nomula, Rama Naveen Kommuri, Ashwini Cherukuri, Vamsy Chowdary Bobba, Laxmi Sai Teja Naraharasetty

**Overview**:

Traffic modeling system is a system which simulates the behavior of road’s vehicles in the town. This system has capability to create the simulating behavior. Rest of the document will discuss it in more elaborated way.

Here in this document, all the requirements from the client has been mentioned. The document first addresses the purpose of this document in detail then proceed with the scope of project. After adding these things, the product perspective would be mentioned followed by product features.

Similarly, User classes, Operating environment, design and implementation constraints, user documentation, assumptions and dependencies etc. are mentioned.

**TABLE OF CONTENTS**

1. **Introduction** **4**
   1. Purpose 4
   2. Project Scope 4
2. **Overall Description** **5**
   1. Product Perspective 5
   2. Product Features 5
   3. Core features 6
   4. Additional features 7
   5. User Classes and Characteristics 7
   6. Operating Environment 8
   7. Design and Implementation Constraints 8
   8. User Documentation 8
   9. Assumptions and Dependencies 9
3. **System Features** **10**

**Core Features** **10**

* 1. User Welcome 10

1. **Functional Requirements** **12**
2. **Other Nonfunctional Requirements** **12**
   1. Performance Requirements 12
   2. Safety Requirements 12
   3. Security Requirements 13
   4. Software Quality Attributes 13

**1. Introduction**

1.1 PURPOSE

The purpose of this Software Requirements Specification (SRS) document is to provide a detailed description of the functionalities of the Traffic Modeling System. This document will cover each of the system’s intended features, as well as offer a preliminary glimpse of the software application’s User Interface (UI). The document also cover hardware, software, and various other technical dependencies.

1.2 PROJECT SCOPE

The Traffic Modeling System is composed of a main component: a standalone application which will run any local system as long as the system meets the hardware requirements need. The system is designed to facilitate the process of tracking and simulating the traffic behavior.

**Note:** Hardware requirements is discussed further in this document

**2. Overall Description**

2.1 PRODUCT PERSPECTIVE

The Traffic Modeling System project is a full featured, self-contained i.e. standalone application used for the sole purpose of road traffic simulation. The scope of the project encompasses on client-side functionalities, so this aspect is covered in detail within this document. Below figure 2.1, is a diagram of the Traffic Modeling System which illustrates the interactions between the user and simulator. Few buttons are there to interact with the system and as below is only the snap shot of Interactions between different vehicles so, buttons were taken down.



**Diagram 2.1:** Interactions between different vehicles

2.2 PRODUCT FEATURES

The following are description of the all the features and functionalities of the Traffic Modeling System. The features are split into two major categories: core features and additional features. Core features are essential to the application’s operation, whereas additional features simply add new functionalities. The latter features will only be implemented as time permits.

2.3 CORE FEATURES

1. USER Input for:
   1. Drag and drop the road components i.e. 2-way, 4-way, roundabout, T junction (3-way), 4 -way signals on the simulator panel in any order to make the road path.
   2. Enables the user to customize the order of dragged road components
   3. There would be space of maximum of 10 road components that a user can drop on the simulation panel. On considering the screen space it would be good to show maximum of 10 components. .
   4. Arrival rate: avg. number of car per minute. e.g. 30 cars/min, and # secs/car per road per direction.
   5. User should input speed for the cars and every car would get random speed from the input value with +/- % mph.
   6. Reaction time of cars: random value i.e. random value would range from 0 to 1. As every car on the road has different reaction rate therefore the reaction time would be a random value.
   7. Distance between consecutive cars.
2. Input on simulator panel:
   1. Percentage of vehicle go straight, left, right. And the percentages will sum up to 100%. As there are four directions, the input parameters should be 12 for each road component and eventually **Total input parameters = 12 \* no. of road components.**

1. Simulation panel:
   1. Stores and maintain the order of all the road components which the user has dragged to the simulation panel.

* 1. Play the simulation and road vehicles should start their simulating behavior i.e. the cars will generate and simulate the traffic behavior and behave similar to the actual traffic on the roads. There would be one stop button to stop the simulation.
  2. Efficiently distributes the total incoming from the 3 different sides of the road traffic by using the % age rule (discussed earlier)for left, right , and straight
  3. The speed of the car should be display on the car itself.

2.4 ADDITIONAL FEATURES feauFEATURES

1. Traffic modeling system would support for the 2-D layout.
2. Roads should have multiple lanes.

2.5 USER CLASSES AND CHARACTERISTICS

The Traffic Modeling System project is meant to offer a simulation system solution that is faster, easier, and more convenient than manually calculating and handling road traffic volume. Consequently, the application will have little or no learning curve. Thus, technical expertise should not be an issue. Instead, anticipated users can be defined by how they will use the product in a particular situation. The following list categorizes the scenarios in which Traffic Modeling System is expected to be utilized.

2.6 USER CLASSES AND CHARACTERISTICS

1. Town planning
   * Key functions:
     + Keep track of the vehicles and their behavior on road
     + Record the changed values by the user and save them to a file
     + Load the file to set the parameters for their default value
     + After the user input all the parameters it can simulate the behavior
   * Requirements:
     + Provide the user interface for adding parameters or to change the default values
     + Speed tracking system
     + Automated, background algorithm for calculating speeds, directions
2. See simulation behavior for existing town
   * Key functions:
     + Keep track of the vehicles and their behavior on road
     + Record the changed values by the user and save them to a file
     + Load the file to set the parameters for their default valued
     + After the user input all the parameters it can simulate the behavior
   * Requirements:
     + Simple user interface for adding parameters
     + Speed tracking system
     + Automated, background algorithm for calculating speeds, directions

These groups are not meant to separate or categorize users, just the different situations in which Traffic Modeling System is likely to be used. In fact, a user may utilize the application for all of these scenarios simultaneously.

It is crucial that each of these situations be fully supported in the final product so as to maximize the overall value of the product. It is also important that the application must be as user- friendly means the GUI of the product would be self-explanatory in terms of user input, drag and drop components etc. otherwise it would not be a viable alternative to handling road traffic manually. Most importantly, the application must be reliable. Regardless of the situation, the application must accurately distribute traffic. There is zero tolerance for error when dealing with safety.

2.7 OPERATING ENVIRONMENT

The main component of the Traffic Modeling System project is the software application, which will be limited to the system operating system. The application is not resource- or graphics-intensive, so there are no practical hardware constraints. The application is a self-contained unit and will not rely on any other software related components.

The machine will operate on any Linux/Windows/Mac. The Traffic Modeling System uses flat file as the database.

2.5 DESIGN AND IMPLEMENTATION CONSTRAINTS

The primary design constraint is the system platform. Since the Traffic modeling system is designated for Linux, Mac, or Windows based systems, therefore it would not work on mobile devices which has limited screen size and limited resolution. Creating a user interface which is both effective and easily navigable for a user to work with the software, would pose a difficult challenge.

2.6 USER DOCUMENTATION

The primary goal of Traffic Modeling System is to *facilitate* the process of managing traffic. Consequently, the application will be designed to be as simple to use as possible. Nonetheless, users may still require some supplementary information about each component of the Traffic Modeling System.

2.7 ASSUMPTIONS AND DEPENDENCIES

TIME DEPENDENCIES

Optional features, however, are *not* critical to the function of the application. They are usability improvements and convenience enhancements that may be added after the application has been developed. Thus, the implementation of these features is entirely dependent upon the time spent designing and implementing the core features. The final decision on whether or not to implement these features will be made during the later stages of the design phase.

HARDWARE DEPENDENCIES

In the project there would not be any hardware requirements. The project would be a standalone project and would not depend on any external hardware.

**3. System Features**

Traffic Modeling System’s system features are divided into two main categories: core features and additional features. Core features form the body of the application and include any features that are essential to the functionality of the Traffic Modeling System. These features must be implemented in order to have a fully-functioning application. Additional features, however, are not required for the application to function. They include any features which, if time permits, will be added to the application in order to provide extra functionality.

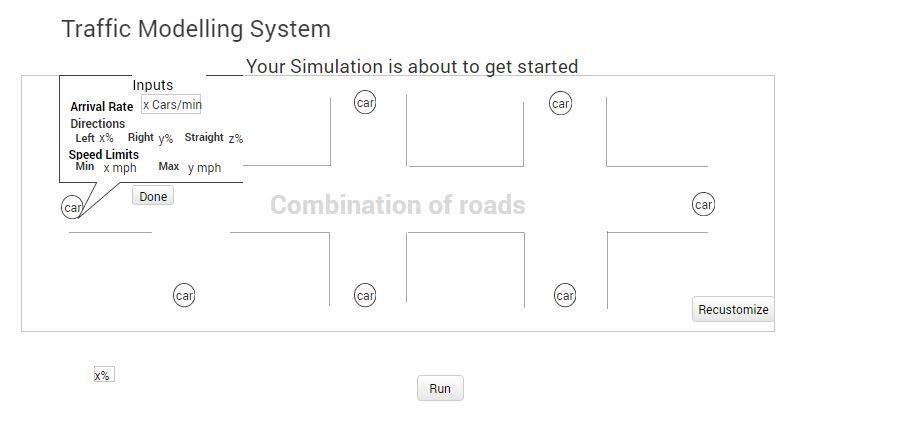


**CORE FEATURES**

3.1 USER WELCOME

When the application is installed and run for the very first time, the user is presented with an initial input screen. The screen prompts the user to input the parameters for Traffic Modeling System.

Below figure 3.1 shows the user input box.



3.1.1 STIMULUS

Figure 3.1

**Step 1** Traffic Modeling System application launched from the system

**Step 2** The user is prompted to browse file for default values

**Step 3** The user is prompted to drag and drop some road components like roundabouts etc.

**Step 4** The user has choice to enter input parameters for these components like % of traffic would go left, right, straight.

**Step 5** Click on simulateto run simulation

**Step 6** The simulation will start and the outcome would be displayed on screen like # of cars on screen etc.

**Step 7** User can stop the simulation and result would be displayed on the screen

3.1.3 SYSTEM REQUIREMENTS

1. There would not be any specific hardware requirement and the Traffic modeling system application is solely depends on the operating system’s hardware requirements.

**4. Functional Requirements**

TMS System’s features are described functional requirements of the system. Each functionality is described in detail inside the core features. Core features are the essential functionalities of the system. In the detail description of the system from the client these essential requirements are identified. These are core features.

1. The user should drag and drop the road components i.e. 2-way, 4-way, roundabout, T junction (3-way), 4 -way signals on the simulator panel in any order to make the road path.
2. If the user does not drag and drop any road component then a single road would appear on the simulation panel.
3. Enables the user to customize the order of dragged road component.
4. There would be max 3 road components on the simulation panel since it’s not a 2-D layout. It is because at any point of time the screen is able to show 3 road components and increasing more would hide other road components. So, 3 would be the better number.
5. The user input the arrival rate: avg. number of car per minute. e.g. 30 cars/min, and # secs/car per road per direction
6. User should input speed for the cars and some percentage. Now the final speed of the any car would be the assigned speed and some -+ percentage which the user inputted.
7. The application would assign default values from a file and if the user wants to change the default values then the can change it by clicking on simulation area. These default values will guide the application about how much the percentage of road traffic would go in the directions like left, right, and straight.
8. Reaction time of cars: random value i.e. random value would range from 0 to 1. As every car on the road has different reaction rate therefore the reaction time would be a random value.
9. Cars will simulate the traffic behavior.
10. Once the simulation starts, then the user does not has permission to change any values i.e. either the default values or the user inputted value e.g. input speed, arrival rate etc. In addition to this the user would not be permitted to change the road components i.e. any modification on the sequence of road components.
11. There would be 1 sec. delay between 2 consecutive cars so that they do not collide with one another.
12. The simulation will run as long as the user would not press stop button.
13. The result of the simulation would be displayed on the next screen or stored onto a file. The result would include: number of cars reached at a particular destination in a particular time frame.

**5. OTHER NONFUNCTIONAL REQUIREMENTSNTS**

5.1 PERFORMANCE REQUIREMENTS

Performance would not be an issue because once the user start the simulation the application will simulate the road traffic behavior and as the application is a standalone application and solely depends on the operating system’s hardware configuration, so performance would not be an issue.

5.2 SAFETY REQUIREMENTS

Traffic Modeling System will not affect data stored outside of its system nor will it affect any other applications installed on the system. It cannot cause any damage to the system or its internal components. Traffic Modeling System should not be used while operating a vehicle or in any other situation where the user’s attention must be focused elsewhere.

5.3 SECURITY REQUIREMENTS

There is not any such requirements

5.4 SOFTWARE QUALITY ATTRIBUTES

The graphical user interface of Traffic Modeling System is to be designed with usability as the first priority. The app will be presented and organized in a manner that is both visually appealing and easy for the user to navigate.

To ensure reliability and correctness, there will be zero tolerance for errors in the algorithm that computes and splits traffic between different roads.