**3.1 Requirements**

The traffic simulation application needs to be built according to a certain set of rules and must adhere to a certain guideline which, will be listed here. This section will suggest certain methods that the application should possess to help focus the development of the application. These requirements have been made in order to fit the needs of the project guidelines as well as the requirements decided on by the team.

**3.1.1 Functional Requirements**

* The application should be able to randomize the number of cars entering the simulation.
* The application should be able to have individual cars randomize their behavior at any given point.
* Vehicles must be able to analyze events taking place with other cars and the road environment.
* The application must be able to display a large amount of cars concurrently.
* The application must update the simulation constantly.
* Vehicles must be able to move along a given road network without difficulty.

**3.1.2 User Requirements**

* The user must be able to change the amount of cars entering the simulation at any given point.
* The user must be able to change the speed of cars at any given point.
* The user must be able to change the traffic lights at any given point.

**3.2 Design**

The aim of this project is to develop a data visualization tool to display how vehicles interact with each other and a particular environment as well as simulate a variety of traffic paradigms. This tool will be primarily developed in JAVA and will be divided into multiple layers with each layer having a specific objective. The overall purpose of the layers is to divide complex parts of the tool into simpler parts that will allow the team to program more efficiently. The team has decided that the tool will initially consist of the following five layers:

**3.2.1 Vehicle Layer**

This layer is representing the various types of vehicles given simple movement state (acceleration, slowing down, randomization, and car motion). Vehicles are represented on the graphical interface as one single cell or two cells based on the vehicle size (one cell for cars and two cells for bus/Coaches/ambulance). The vehicles’ states demonstration is decided by the velocity value of the vehicle and available space in front of it. Therefore, the team has agreed to follow the Nagel–Schreckenberg model that is based on cellular automaton model to implement this layer.

**3.2.2 Map Layer**

This layer is representing the road networks (straight road and multi-lane junction). It is an independent layer, which means no vehicles are featured. The road network drawn on the graphical interface in 2D style (X and Y cells) where X and Y are the width and length of the road lane. Number and location of the cells contributed on the layout to draw the road network will be decided when the user select one of the road patterns available.

**3.2.3 Interaction Layer**

This layer is a combination of the vehicle and map layers. It is an implementation of different types of vehicles following specific routes on different road patterns. Vehicles will enter specific road and exit from one of the linked roads based on drawn road network.

**3.2.4 Event Layer**

This layer is designed to add different events. These events could be either road events such as traffic jam and traffic lights or vehicle events such as reckless, cautious, or normal. The user is allowed to insert more than one event at the same time before the simulation of complete collective actions.

**3.2.5 Simulation Layer**

This layer is the final layer where all designed traffic model is animated. Each vehicle will behave in dissimilar mode according to the traffic model. In addition, the simulation will generate results and information for the number of current vehicles, and their position, also the number of vehicles entered the road network and the number of vehicles exit the road network. It will also record the time of journey of vehicles based on the traffic model designed.

Each layer has a specific purpose towards the tool and is necessary that a layer is completed before advancing to the next layer. This approach is being taken to ensure that every layer has a strong foundation and has the absolute minimum amount of errors possible. The vehicle layer and map layer form the basic part of the tool therefore the team will focus on developing these two aspects first. If these two layers are not functioning properly then the interaction layer will not work as desired and will cause other layers problems in later stages of development. Through this we believe that it will be easier to catch errors or identify areas that need to further improved.

The event and simulation layers will serve as the advanced part of the tool and focus on meeting the main objectives of the project. These two layers will require more focus than the others and can only be developed when the interaction layer functions properly without errors. As the event layer will introduce the concepts of road emergencies and restrictions it will help the team make a more realistic traffic simulation engine. The simulation will be the last layer to be developed and will allow the user to vies various traffic scenarios and how they could play out.