***Design***

The model-view-controller design is going to be used.

The model will handle the data processing from the user and traffic objects. The controller handles user inputs and the view handles what the user sees. The user will give specifications such as number of cars and the controller will send this information to the model which in turn will process the data and update the view.

***Model classes***

Roads, car, vehicles, motorbike, bus, 3-way, 4-way, straight, traffic light

**Road**

Variables

* Segment count

Methods

* Set segment count
* Get segment count
* Set traffic light
* Get traffic light

**Car**

Variables

* segment
* Road

Methods

* Move
* Position set
* Position get
* Road set
* Road get

**Traffic light**

Variables

* Colour
* Rate of change

Methods

* Stop car
* Set colour
* Get colour
* Set rate of change
* Get rate of change

***Controller***

**Input**

* City or simulation
* City mode: create, edit, open, save
* City editing mode: selection of road shapes and traffic lights to place on city
* Simulation mode: Run simulator, stop simulator, set vehicle spawn rate, update rate

***View***

**Outputs**

* Jframe for program GUI
* Buttons for new, open, save and exit
* Jlabel do show the program status
* JMenuBar for city editing and simulation modes
* Derived JPanel to display the main screen
* Different coloured rectangles for each of the simulator elements

***Milestones***

* Get a car to move across the frame and another car to move in the other direction on a straight road and both successfully stop at the end

**UML Diagrams**