**URS**

**Group member:**

**Govinda**

**Juarni**

**Ruth**

**Shuying**

**Wenhua**

**Melese**

Contents

[Introduction 3](#_Toc367985316)

[Functional Requirements (USE-cases) 4](#_Toc367985317)

[Select grid 5](#_Toc367985318)

[Add crossing 5](#_Toc367985319)

[Remove crossing 5](#_Toc367985320)

[Start Simulation 6](#_Toc367985321)

[Stop Simulation 6](#_Toc367985322)

[Maximize 7](#_Toc367985323)

[Close Maximize Window 7](#_Toc367985324)

[Adjust Traffic 8](#_Toc367985325)

[Save Simulation 9](#_Toc367985326)

[Open Simulation 10](#_Toc367985327)

[New 11](#_Toc367985328)

[Show Statistic 11](#_Toc367985329)

[Exit 12](#_Toc367985330)

[User Interface 13](#_Toc367985331)

[Main 13](#_Toc367985332)

[Menu Strip 13](#_Toc367985333)

[Tool Box 14](#_Toc367985334)

[Grid 14](#_Toc367985335)

[Customize 15](#_Toc367985336)

[Maximize 16](#_Toc367985337)

[Non-functional requirements 17](#_Toc367985338)

# Introduction

User can achieve the goal of reducing number of accident on the roads in the city of C sharp according to the efficient adoption of traffic light in the city. That can be achieved by using an efficient traffic simulation program before the implementation of traffic lights. The following document contains the functional and non-functional requirements in greater detail.

# Functional Requirements (USE-cases)

The use-cases that were considered for this project are as follows:

* Select Grid
* Add crossing.
* Remove crossing.
* Start simulation.
* Maximize.
* Adjust traffic.
* Save simulation.
* Open simulation.
* Stop simulation.
* New,
* Exit
* Show statistic

Top Left cell is selected

Button Start

Save/open

Button stop

Edit, add, remove

Btn statistic

Button maximize

Close statistic Close custom

Fig.1 Different States of the machine

**NOTE- The above mentioned states will be used as pre and post conditions as required and only the respective letters(A-E) will be used(fig.1).**

## Select grid

**Goal**: to choose specific place for drawing cell

**Actor**: user

**Pre**: state- B

**Post**: a cell with bold border will be displayed

**MSS**:

1. User clicks on a cell
2. System marks the cell as selected and draws border around it

**EXT:**

**2a.** if the clicked position is already selected, nothing happens- and state-B

2b. if there is another grid already selected, it will be cleared from system and borders are removed-

MSS 2

## Add crossing

**Goal**: add a crossing in a cell

**Actor**: user

**Pre**: state-B + cell is selected

**Post**: the crossing is put on a cell + State B

**MSS**:

1. user selects a crossing from toolbox (type 1 or type 2)
2. system checks if it can add in the selected cell
3. the system adds the crossing on that cell
4. the system checks the adjacent four cells for crossing, and adjust the number of cars

of the appropriate roads/lanes according to their input/output ratio

**EXT**:

2a. if the cell is already occupied, System shows a message” can’t put crossing”. Use case ends.

## Remove crossing

**Goal**: remove a crossing from the cell

**Actor**: user

**Pre**: State B+ cell is selected

**Post**: the crossing is removed from the cell + State B

**MSS**:

1. user clicks remove crossing button
2. system checks the process and updates the process(change number of cars of adjacent crossings )
3. the crossing is removed from that cell

**EXT**:

4a. if the selected cell is empty nothing happens; Use case ends.

## Start Simulation

**Goal**: Run the simulation

**Actor**: User

**Pre**: State-B + crossing added

**Post**: State C

**MSS:**

1. User clicks on the button “Start simulation”
2. The system then starts simulation

**EXT:**

2a. If there is traffic jam in any of the crossings message “Traffic jam in the system!” is shown.

* System paints that specific road / lane red

## Stop Simulation

**Goal**: Stop the simulation

**Actor**: User

**Pre**: State C

**Post-condition:** state B

**MSS:**

1. User clicks on the button “Stop simulation”
2. The system then stops simulation

**EXT:** None

## Maximize

**Goal**: increase the size of the crossing

**Actor**: User

**Pre**: State B and crossing is added, cell is selected

**Post**: State E

**MSS**:

1. User clicks on maximized button
2. System checks the selected cell and do processing
3. System opens a panel and paints the selected crossing object with a bigger size.

**Ext**:

4a. if selected cell is empty; nothing happens. Use case ends.

## Close Maximize Window

**Goal**: to close maximize window

**Actor**: User

**Pre**: State E

**Post**: State B

**MSS**:

1. User clicks on close button on maximize window
2. System removes the maximize window

## Adjust Traffic

**Goal**: to adjust number of cars, traffic light timing

**Actor**: User

**Pre**: State B, cell is selected, there is crossing in the cell

**Post**: State B

**MSS**:

1. User clicks the tabs “edit option”
2. System switches tab from “toolbox” to “ Edit Option”
3. System checks the type of crossing in the cell.
4. User chooses the traffic light and changes the timing of path or paths
5. User inputs all the setting he needs and press ok to confirm
6. System checks for the inputs and adjust the traffic light ,

The cars are distributed to the paths/lanes of each edited roads according to the ratios of the current inlet/outlet road.

1. System changes the number of cars of the adjacent inlet/outlet roads as per their ratio
2. System saves the changes.
3. System closes the edit Window
4. System reopens the system

**EXT:**

2a. if selected cell is empty, use case ends.

6a. if user inputs are not correct; message is shown “check the input”;

1. User press OK; goes to MSS:5
2. User press cancel; use case ends

## 

## Save Simulation

**Goal**: to save the simulation

**Actor**: User

**Pre**: State B

**Post**: state B

**MSS**:

1. Actor chose Save option from Menu
2. System Opens Save Dialog Box
3. Actor choose the directory where to save the File & gives file name
4. Then the actor clicks save button.
5. System checks and saves the simulation file and statistical data file.
6. System gives message “successfully saved”.

**EXT**:

4a. Actor chooses Cancel. Use case Ends.

5a. If the file name already exist, system gives message

“File name already exist. Do you want to overwrite?”

Actor chooses “YES”, goes to MSS: 5.

Actor chooses “No”, goes to MSS: 3.

## 

## Open Simulation

**Goal**: To open saved simulation

**Actor**: Users

**Pre**: state B,

**Post**: Pre-saved simulation is opened

**MSS**:

1. Actor Choose Open option from menu
2. System shows the Open file Dialog Box
3. Actor choose the required simulation file from the directory
4. Actor clicks Open option
5. System checks the simulation
6. Chosen simulation is opened

**EXT**:

4a. Actor choose cancel option. Use case ends

5a. if the chosen file is in bad format, system shows message “File cannot be opened.”

Use case ends.

5b. if the chosen file is already open; system shows message “file already opened”.

Use case ends

## New

**Goal**: To open a new simulation project

**Actor**: Users

**Pre**: State A

**Post**: State B

**MSS**:

1. Actor Choose New from Menu.
2. System checks the process
3. System show new simulation grid to work.

**EXT**:

2a. if pre-opened file is not saved system shows message

“Do you want to save the Pre-opened file before opening New”

Actor choose “Yes”; Goes to “Save” use case.

Actor choose “No”; go to MSS iii.

Actor choose “Cancel”; use case ends

## Show Statistic

**Goal**: to show summery of the current traffic system

**Actor**: user

**Pre**: state- B

**Post**: state-F

**MSS**:

1. User clicks on a button show statistic
2. System shows opens note pad with the whole traffic information in new window

**EXT: None**

## Exit

**Goal**: To close the application

**Actor**: User

**Pre**: State B

**Post**: State A

**MSS**:

1. Actor choose exit option from menu
2. System checks the circuit status
3. System close the application

**EXT:**

2a. If the changes made on the configuration and statistical data of the simulation are not saved; system shows the message

“Do you want to save circuit?”

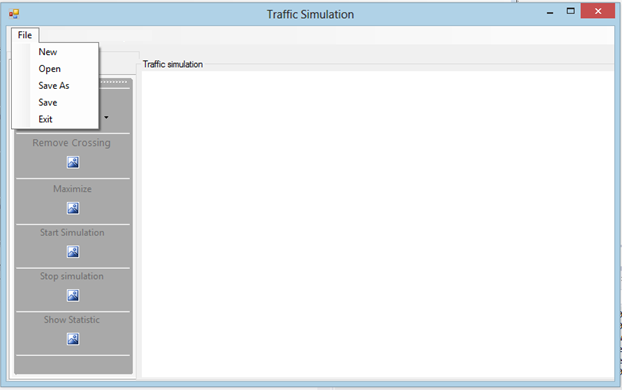
User choose “No”; go to MSS iii.

If user chose “Yes”; Go to Save use case.

# 

# User Interface

## Main



The main page of simulation contains of menu strips, tool strips and tab page.

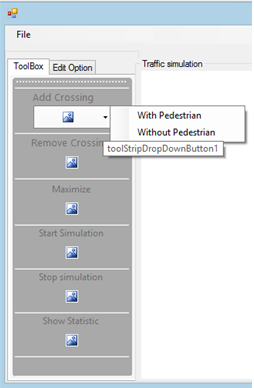
## Menu Strip

Menu strips contains **File**

File has dropdown list and contains:

1. New
2. Open
3. Save
4. Save As
5. Exit

## Tool Box



Tool Strip

1. Add Crossing: it has two option as shown in side figure.

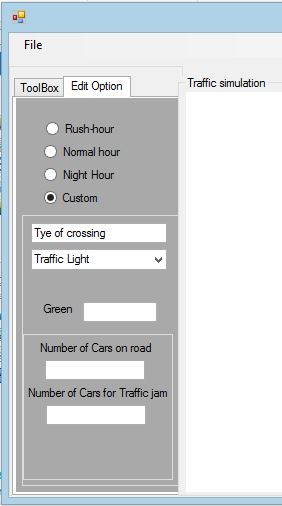
With and without pedestrians

1. Remove Crossing: to remove crossing from cell
2. Maximize: this option helps user to make the picture bigger for clear visibility.
3. Start simulation: user clicks on this button to start simulation
4. Stop simulation: user clicks to stop simulation
5. Show statistic: to view an overview of simulation.

## Grid

We will have a grid on this white space on the application when the application runs.

## Customize



User can change the setting for the traffic light timing, number of cars and others.

For this simulation we are going to work on “custom” part. The other part like “Rush-Hour”, “Normal-Hour”, and “Night-Hour” will be defined after client finds good setting.

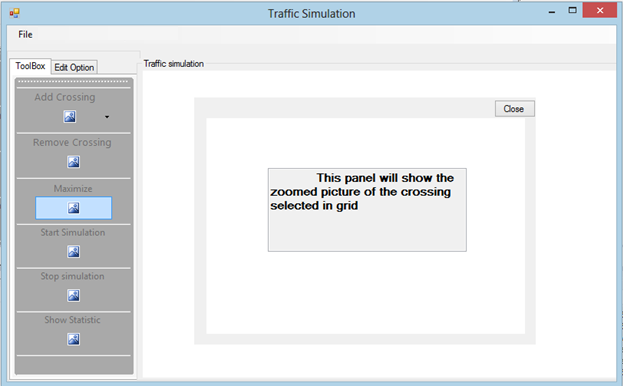
# 

OK

## Maximize

The following panel will be opened when user choose maximize button from tool strip. The picture of crossing selected will be shown 3 times bigger in this panel.

User can easily go out of this maximize part by clicking close button provided on panel.



# Non-functional requirements

**Programming environment:**

* Programming in C#.

**System requirements:**

**Operation systems**: Windows® XP/ Windows® Vista/ Windows® 7/ Windows® 8.

**Software:** Microsoft Visual C# 2012 Redistributable X86 & X64.

**Development System Constraint:**

* The “Could have” may be implemented if the “Must have” and “Should have” have been completely implemented. The MOSCOW table for our use cases is as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Functional requirements | M | S | C | W |
| Add crossing | X |  |  |  |
| Remove crossing |  | X |  |  |
| Start simulation | X |  |  |  |
| Stop simulation | X |  |  |  |
| Maximize |  |  | X |  |
| Adjust traffic | X |  |  |  |
| Save simulation | X |  |  |  |
| Open simulation | X |  |  |  |
| Create new simulation project |  |  |  | X |
| Exit application | X |  |  |  |
| Show Statistic |  | X |  |  |

**Maintainability:**

* Working in a group for the project leads us to have the ability to maintain our project, or improve reliability of project by providing good documentations and a user manual.

**Documentation:**

* Information is written using Microsoft Word for any read or projection.

**User Objectives, Values and Concerns:**

* Usability (ease-of-use, learnability etc.).
* Our target groups are people with sufficient knowledge in traffic control component.

So it will be easily adopted by our users.

* User can achieve the goal of reducing number of accident on the roads in the city of C sharp according to the efficient adoption of traffic light in the city. That can be achieved by using an efficient traffic simulation program before the implementation of traffic lights.

**Development Organization Constraints:**

* Skills of developers in Object Oriented Design, Programming and Management.

**Development Organization Objectives, Values and Concerns**

* The prototype serves all necessary functional requirements (Use cases).
* Final deliverables reaches every requirement of the assignment and our promises.
* Final deliverables is planned to be handed in 7th week of second period.
* Presentation and Final assessment in 8th week of second period.
* Provide good documentation.
* Balance all concerns of team members to the final product.
* Balance time spending on this project among other subjects.