Trafic Control System
User Requirements Document

Agnes Wadee
Coen Stange
Wen Li
Yongshi Liang

Background and Context

This user requirements document specifies the software requirements for the "Traffic Control System". This application allows traffic to be simulated with the purpose of noticing traffic jams related to traffic lights.

Definitions and abreviations

User	The person who is controlling this application.
System	The implementation of this application.
Grid	A place on the screen where a component can be added for the traffic situation.
Component	A visible representation of an object on the screen of the user.
Crossroad	A component that can be used in the traffic simulation which has traffic lights.
Traffic light	A component of the crossing which controlls the traffic by displaying colors red, yellow green. For which green the traffic is allowed to go.
Pedestrian	A simulation of a pedestrian crossing a road from the traffic light.
Lane	A component that represent a piece of road.
Group of lanes	A group of incoming lanes at a crossing which have green light at the same time.
Cars	A component that represent a car on the road.

Contents

1   Requirements

The following chapter describes the functionality that the application has to implement.

1.1   General requirements

Rxx	Code	Requirement
R01	GEN-010	The program is compatible with Windows 7.
R01	GEN-020	The system allows to design a traffic situation.
R01	GEN-020A	The traffic situation can be designed with the following components - Crossroad without pedestrian lane - Crossroad with pedestrian lane - Straight road - Curved road
R01	GEN-020B	All crossroads have sensors for cars and pedestrians, traffic lights don't go green for no cars or pedestrians.
R01	GEN-020C	Components can be rotated.
R01	GEN-025	Components can be dragged onto the grid.
R01	GEN-025A	The grid by default is 4x3 but it is possible to change the size of the grid before adding components.
R01	GEN-030	From the traffic lights of the crossroads it is possible to change the amount of time that traffic light is green.
R01	GEN-040	The system allows simulate traffic in realtime, and allow to change the simulation speed.
R02	GEN-040A	The system can return a report in which it highlight where many traffic jams are in a graphical representation of the traffic situation. Which can be saved as an image file.
R01	GEN-050	The system allows to open and save the traffic situation to a file.
R01	GEN-060	The system allows to specify for each open incoming lane to set the amount of traffic coming.
R01	GEN-200	The sytem will be delivered as a standalone application (.exe file).

1.2   Non functional requirements

Rxx	Code	Requirement
R01	NFR-010	The system can run on a regular computer/notebook build less than 4 years ago running Windows 7.
R01	NFR-020	The user interface of the system is straightforward and easy to learn according to the client.
R01	NFR-030	The application can be used offline.
R01	NFR-040	The system is stable, and is not allowed to give unclear error messages.
R01	NFR-050	The system is designed to be used with mouse and keyboard.

2   Specification

The following chapter describes the implementation of the application.

2.1   Main window

The main window is divided into two parts, see figure 1. On the very top of the application is a menubar where the user can do actions like saving their work. Then the window is split up in two parts. On the left side is the menubar and on the right side the grid. The grid is the representation of the traffic situation. The components (see section 2.2) can be dragged from the sidebar to the grid. To remove a component right-click on it and press on Delete in the context-menu. All open incoming lanes have a text-box which allows the user to specify the amount of traffic coming. The simulation can simply be started by the play button and the simulation speed can simply be changed with a slider. With the button "Show Report" the user can get a report of that moment of the simulation. The report will contain a still image of the current situation. It highlights the traffic jams and the image can be saved.
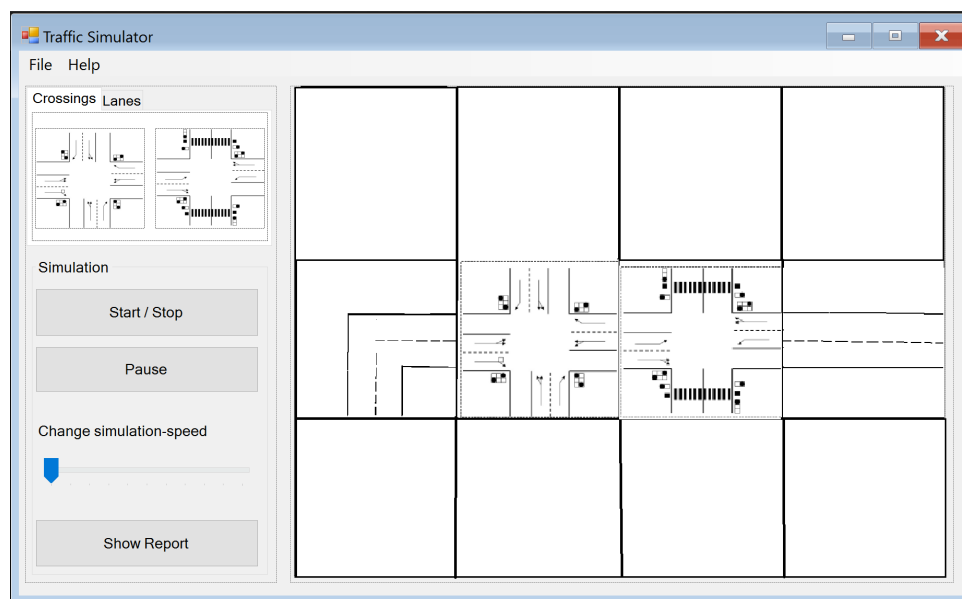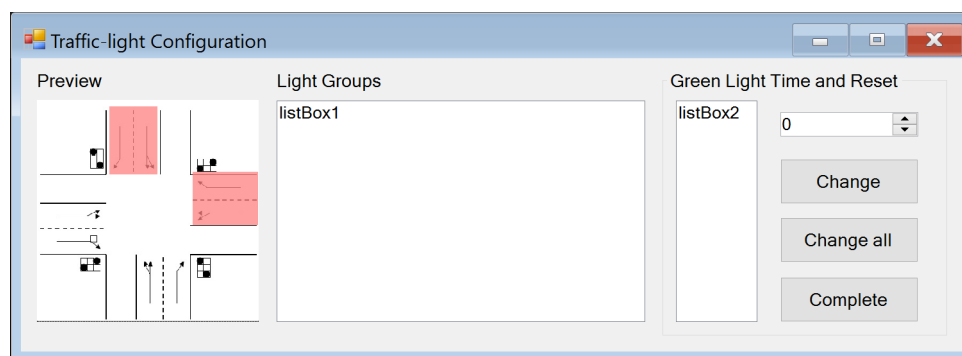
Figure 1: Mockup of the Main window

Figure 2: Mockup of the Configuration window

Rxx	Code	Specification
R01	MWS-010	When the application just started it will create a new grid 4x3
R01	MWS-020	The main window has a menubar.
R01	MWS-020A	The menu bar has the following structure: • File – New – Open – Save – Save As • Help
R01	MWS-020B	A new simulation can be started by pressing on new, a window will prompt for the width and height for the size of the grid.
R01	MWS-020C	The manual can be opened by pressing on Help.
R01	MWS-030	The window has a sidebar on the left.
R01	MWS-032	The sidebar contains all the components described in section 2.2.
R01	MWS-032A	The component can be added to the grid by dragging it to the desired location.
R01	MWS-034	The sidebar contains a button which allows the simulation to start/stop.
R01	MWS-035	The sidebar contains a button which allows the simulation to pause.
R01	MWS-036	The simulation-speed can be changed by adjusting the slider.
R02	MWS-038	In simulation the button "Show report" will generate a report.
R01	MWS-038A	The report is shown in a new window and contains the current traffic situation including cars and pedestrians.
R01	MWS-038B	In the report the traffic jams are highlighted.
R01	MWS-038C	The report can be saved as an image file.

To change the amount of time each traffic light is green press right-click on the crossroad and click in the context-menu on "Traffic-light configuration". A new window will pop up which allows to set the time for each group of lanes.

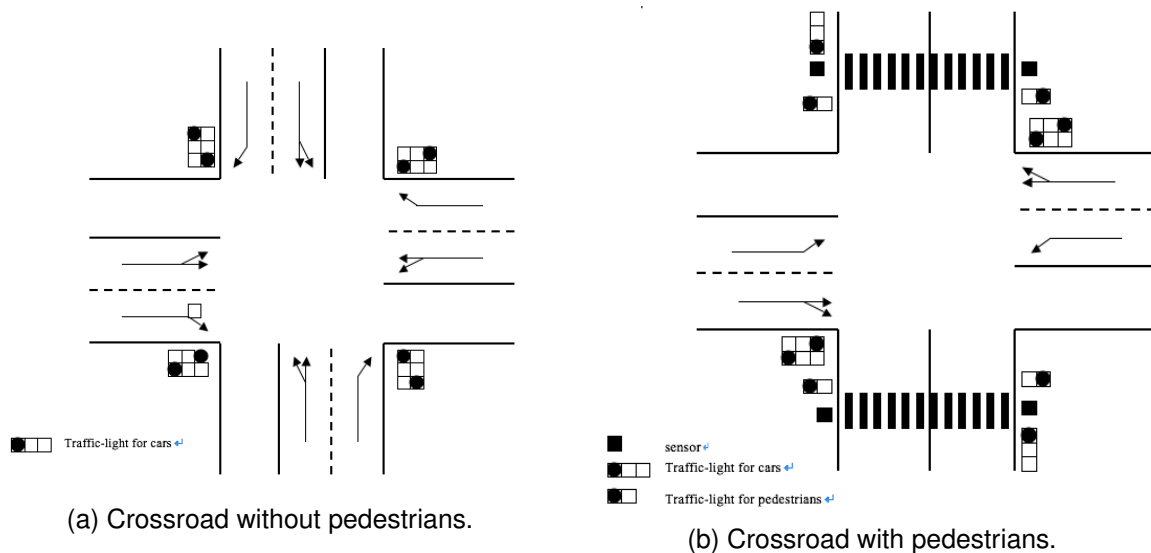| Rxx	Code	Specification
R01	MWS-100	All open incoming lanes have a textbox to specify the amount of traffic coming in.
R01	MWS-110	When pressing right-click on any component placed on the grid a context-menu appears which allows to rotate or delete the component.
R01	MWS-120	When pressing right-click on a crossroad it gives an option "Traffic-light configuration"
R01	MWS-120A	A new window will pop-up with a list of all the lane groups.
R01	MWS-120B	The user can select a lane group and change the amount of time the traffic-light is green.

(a) Crossroad without pedestrians.

(b) Crossroad with pedestrians.

Figure 3: Crossways

2.2 Components

2.2.1 Crossroad

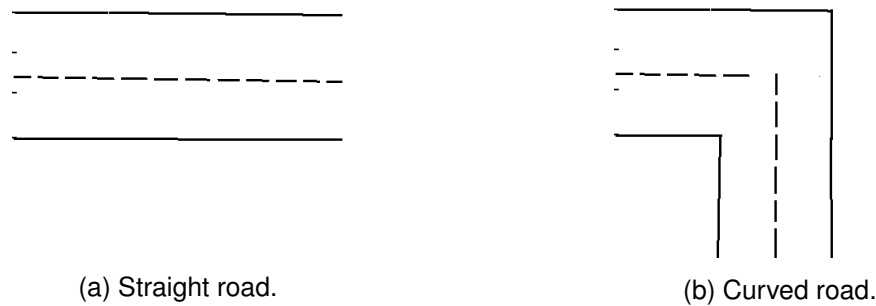| Rxx	Code	Specification
R01	CWC-010	All cossways are connected to 4 roads
R01	CWC-020	There are 2 different types of crossroads see figure 3.
R01	CWC-020	Type A crossroad is without pedestrian lane.
R01	CWC-020A	From each side the crossroad A has 2 incoming lanes and 1 outgoing lane.
R01	CWC-025	Type B crossroad is with an pedestrian lane.
R01	CWC-025A	Type B has from 2 opposite sides a crossroad for pedestrians, which only has 1 incoming lane.
R01	CWC-030	Traffic light for cars have the colors red, orange and geen.
R01	CWC-035	Traffic light for pedestrians have the colors red and green.
R01	CWC-040	Traffic light for cars only turn orange after green.
R01	CWC-040A	Amount of time for the orange light is fixed, and is set to 2 seconds in normal simulation time.
R01	CWC-045	Traffic light for green are set to default for 4 seconds.
R01	CWC-045A	The amount of time each light group of an crossroad can be changed.
R01	CWC-050	The order of the light groups are fixed.
R01	CWC-050A	When there are no cars or pedestrians on the sensors for the according lightgroup it will be skipped in the simulation.
R01	CWC-060	Unconnected incoming lanes have a textbox which allows to change the number of cars comming in when the simulation is running.

(a) Straight road.

(b) Curved road.

Figure 4: Crossways

2.2.2 Road

| Rxx	Code	Specification
R01	RCP-010	There are two types of roads, a straight road and a curved road.
R01	RCP-020	All roads have 2 lanes, in both directions.
R01	RCP-030	Unconnected incoming lanes have a textbox which allows to change the number of cars comming in when the simulation is running.
R01	RCP-030A	Just like crossroad see CWC-060.

2.2.3 Car

Rxx	Code	Specification
R01	CAR-010	All cars run at the same speed, the speed allows 1 car to pass the green light per second.
R01	CAR-020	Cars do not collide.
R01	CAR-020A	Cars hold distance from each other when driving.
R01	CAR-030	No cars go through red light.
R01	CAR-040	Cars will not go through orange light, in case the car is in front of the traffic light.
R01	CAR-050	Cars will take a random direction on the crossroad.
R02	CAR-060	Cars accelerate and break at a realistic speed.

2.2.4 Pedestrian

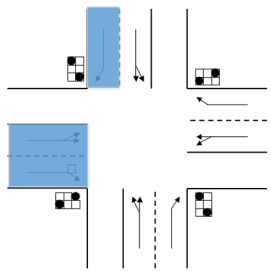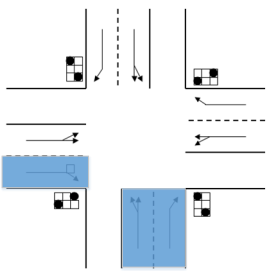| Rxx	Code	Specification
R01	PED-010	Pedestrians have 50% change on being at the crosswalk.
R01	PED-020	Nothing can be configured of the pedestrians.
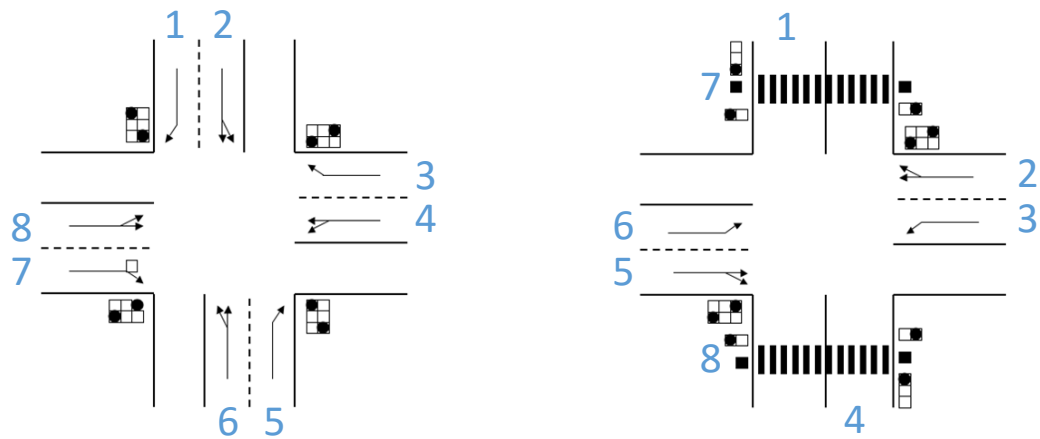
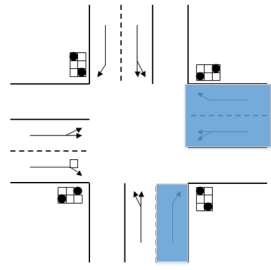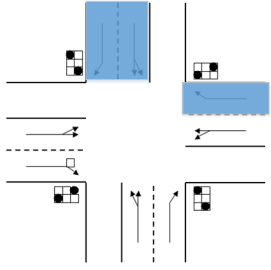(a) Crossroad A incoming lane numbers.   (b) Crossroad B incoming lane numbers.
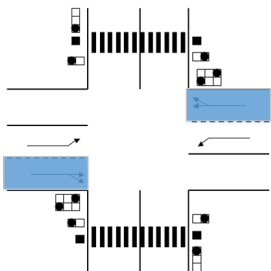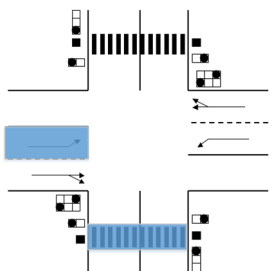
Figure 5: Crossways

2.3 Traffic light groups

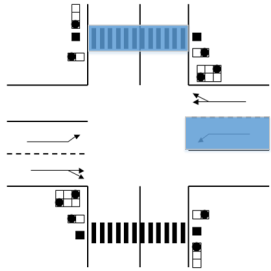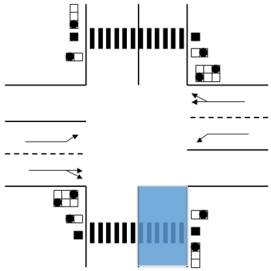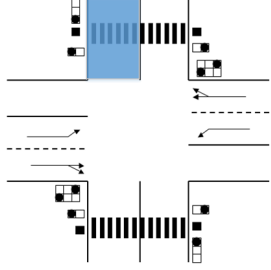The group of lanes which have green light at the same moment are defined in this section. The traffic lights follow the order as defined in this document, Traffic lights will skip a light group if no cars are present on the group of lanes.

2.3.1 Crossroad A

ID	Image	Lanes	ID	Image	Lanes
A1		1, 7, 8	A2		5, 6, 7
A3		3, 4, 5	A4		1, 2, 3

2.3.2 Crossroad B

ID	Image	Lanes	ID	Image	Lanes
B1		2, 5	B2		6, 8
B3		3, 7	B4		4
B5		1			

3 Use cases

Use Case 1	Positioning a lane

Level:	User-goal

Primary Actor:	End-User

Preconditions:	The application is open and no simulation is running.

Main Success Scenario:

1. User drags the lane from the sidebar.

2. User places the lane on the grid.

3. System sets the new placed lane and updates the grid.

Extensions:

2.a If the space where the new lane is put is already engaged, then the system will ignore the new lane and goes back to step 1.

    1. System informs end-user that the lane cannot be placed, because at that space already exists a lane.

    2. End of use case

2.b If the lane is placed outside the grid, then the system will ignore the new lane and goes back to step 1.

    1. System informs end-user that the lane must be placed inside the grid.

    2. End of use case

Post condition:	The program stays in "Positioning a road" state.

Use Case 2	Rotating the component

Level:	User-goal

Primary Actor:	End-User

Preconditions:	There is at least one lane on the grid. No simulaion is running.

Main Success Scenario:

1. User right clicks on a lane.

2. System shows a menu with several options

3. User select "Rotate".

4. System turns the lane 90 degrees.

5. System updates the grid.

Extensions:

   3 User does not want to execute any operation from the right click menu.

      1 User clicks on any space outside the right click menu area.

      2 End of use case.

Post condition: The program stays in "Rotating the lane" state.

Use Case 3	**Positioning a crossroad**
Level:	User-goal
Primary Actor:	End-User
Preconditions:	The application is open and no simulation is running.

Main Success Scenario:

1. User drags the crossroad from the sidebar.

2. User places the crossroad on the grid.

3. System displays a pop-up window.

4. User sets the initial settings for the selected crossroad.

5. System sets the new placed crossroad and updates the grid.

Extensions:

2.a If the space where the new crossroad is put is already engaged, then the system will ignore the new crossroad and goes back to step 1.

    1. System informs end-user that the crossroad cannot be placed, because at that space already exists a crossroad.

    2. End of use case

2.b If the crossroad is placed outside the grid, then the system will ignore the new crossroad and goes back to step 1.

    1. System informs end-user that the crossroad must be placed inside the grid.

    2. End of use case

4.a User does not set any of the required crossroad attributes

    (a) A system default value will be applied to undefined attributes.

4.b User does not want to set any settings.

    1. User clicks on "Cancel" button.

    2. System closes the setting window.

Post condition:	The system displays the updated grid.

Use Case 4	**Configurating traffic light timing**
Level:	User-goal
Primary Actor:	End-User
Preconditions:	There is at least one crossroad on the grid and no simulation is running.

Main Success Scenario:

1. User right clicks on a crossroad.

2. System shows a menu with several options.

3. User chooses "Traffic light configuration" from the menu.

4. System pops up a new window with configuration options.

5. User defines the amount of the time the traffic light is green.

6. User clicks on the "OK" button.

7. System closes the configuration window.

8. System updates the crossroad.

Extensions:
   3. User doesn not want to configurate anything.

       1. User clicks on "Cancel" button.

       2. System closes the configuration window.

       3. End of use case.

Post condition:	The system displays the updated grid.

Use Case 5	**Deleting an crossroad**
Level:	User-goal
Primary Actor:	End-User
Preconditions:	The application is open. There is at least one crossroad on the grid. No simulation is running.

Main Success Scenario:

1. User right clicks on a crossroad.

2. System shows a menu of several options.

3. User clicks on "Delete".

4. System deletes the crossroad and updates the grid.

Extensions:
   3 User does not want to execute any operation from the right click menu.

       1 User clicks on any space outside the right click menu area.

       2 End of use case.

Post condition:	The system displays the updated grid.

Use Case 6	**Setting up a simulation**
Level:	User-goal
Primary Actor:	End-User
Preconditions:	There are components on the grid. No simulation is running.

Main Success Scenario:

1. System displays input boxes of for each incoming lane which is not connected.

2. Users defines the amount of the cars coming through the lanes and fills in the input boxes.

Post condition: The system is ready for running the simulation.

Use Case 7	**Running a simulation**

Level:	User-goal

Primary Actor:	End-User

Preconditions:	The simulation is set up.

Main Success Scenario:

1. User clicks on "Play" button.

2. System runs the simulation.

Extensions:
  2 System detects errors.

    1. System stops running the simulation and gives an error message.

Post condition: The simulation is running.

Use Case 8	**Stopping the simulation**

Level:	User-goal

Primary Actor:	End-User

Preconditions:	The system is running a simulation.

Main Success Scenario:

1. User clicks on "Stop" button.

2. System stops the simulation.

Post condition: The system is not running the simulation.

Use Case 9	Load file

Level:	User-goal

Primary Actor:	End-User

Preconditions:	The application is open. No simulation is running. There is at least one saved file with traffic control system.

Main Success Scenario:

1. User selects file to open.

2. System closes previous project.

3. System loads the file.

Extensions:

2.a File can't be loaded.

    1. System informs user that file can't be loaded and given the choice to stop or choose another file.

    2. End of use case.

2.b Another project is open

    1. System asks if users wants, to save project, that is already open, before closing it and opening another project.

    2. End of use case.

Post condition:	The system has loaded an existing file.

Use Case 10	Save file

Level:	User-goal

Primary Actor:	End-User

Preconditions:	The application is open. No simulation is running.

Main Success Scenario:

1. User click the "Save" button

2. System saves the file.

Extensions:

2.a Project file has not been saved on the device yet.

    1. A saving dialogue window pops up.

    2. User chooses the directory where the file will be stored, and names the file.

    3. User clicks "OK".

    4. System saves the file.

Post condition:	The system has saved a file.

Use Case 11	**Save file as a new file**
Level:	User-goal
Primary Actor:	End-User
Preconditions:	The application is open. Noo simulation is running.

Main Success Scenario:

1. User clicks "Save as" button.

2. A saving dialogue window pops up.

3. User chooses the directory where the file will be stored, and names the file.

4. User clicks "OK".

5. System saves the file.

Post condition:	The system has saved a file.

Use Case 12	**Resizing the grid**
Level:	User-goal
Primary Actor:	End-User
Preconditions:	The application is open. Noo simulation is running.

Main Success Scenario:

1. User clicks on "Edit" on the top bar.

2. System displays a menu with several options.

3. User clicks on "Document settings".

4. System shows document settings panel.

5. User defines the size of the grid by tying numbers in the width and height input boxes.

6. User clicks on "OK".

7. System updates and display the new grid.

Extensions:
   5 Use wants to make the grid size smaller.

   (a) System discards the objects placed outside of the new grid size.

Post condition: System displays the resized grid.

Use Case 13	Closing application
Level:	User-goal
Primary Actor:	End-User
Preconditions:	The application is open.

Main Success Scenario:

1. User presses the close button of the window.

Extensions:
   1 User has unsaved changes

   (a) Ask if user wants to save changes, if so go to use case 10.
   (b) Close application.

Post condition: System displays the resized grid.

3.1   General Requirements with corresponding use cases

This section elaborates more on the general requirements by illustrating the link between these requirement and their corresponding use cases.

General Requirement	Corresponding Use Case
GEN-010	9:Load file
GEN-020	9:Load file
GEN-020A	1:Positioning a lane, 3:Positioning a crossroad, 2:Rotating the lane
GEN-020b	4:Configuring traffic light timing
GEN-020C	2:Rotating the lane
GEN-025	1:Positioning a lane, 3:Positioning a crossroad
GEN-025A	12: Resizing the grid
GEN-030	4:Configuring traffic light timing
GEN-040	7:Running a simulation
GEN-040A	10:Save file
GEN-050	9:Load file, 10: Save file, 11:Save file a new file
GEN-060	3:Positioning a crossroad, 6: Setting up a simulation