



University of Applied Sciences

Traffic Control System Test Plan

Agnes Wadee
Coen Stange
Wen Li
Yongshi Liang

Contents

1	Introduction	2
2	Test Action	2
3	Test tables	3
3.1	Positioning a road	4
3.2	Rotating the component	5
3.3	Positioning a crossroad	6
3.4	Configuring traffic light timing	7
3.5	Deleting the component	8
3.6	Setting up the amount of incoming cars	9
3.7	Running a simulation	10
3.8	Stopping simulation	11
3.9	Pausing simulation	12
3.10	Load file	13
3.11	Save file	14
3.12	Save file as a new file	15
3.13	Resizing the grid	16
3.14	Closing application	17

1 Introduction

The purpose of this test-plan is to see whether all functionalities described in the usecases are working correctly and whether certain undesired actions affect the program's workflow. Instead of constant warning pop-up when a certain action is not permitted we focused on preventing the user from creating mistakes as much as possible, meaning certain functions will be disabled when they're not supposed to be accessed.

We will conduct a small test with a handful of testers and conduct a final reconfiguration before the final acceptance test with the client.

2 Test Action

This test action are based on URS document we already made.

Scenarios:

- Positioning a road
- Rotating the component
- Positioning a crossing
- Configuring traffic light timing
- Deleting the component
- Setting up the amount of incoming cars
- Running simulation
- Stopping the simulation
- Pausing simulation
- Load file
- Save file
- Save file as a new file
- Resizing the grid
- Closing application

3 Test tables

Explanation:

- Target on screen
The actual screen commands the user will interact with.
- Test Data/Simulation
Test actions under different kinds of conditions and with different kinds of data to check if we have captured all the exceptions and if we take necessary precautions to prevent the action from crashing.

3.1 Positioning a road

1. Precondition

- Road Type A on Grid B3.
- Crossroad Type B on Grid C4.
- The other Grids are empty.

2. Target on screen

- User clicks on the Road tab.
- User chooses a road.
- User drags the road from the Road tab.
- User places the road on the grid.

3. Test Data/Simulation

Actions	Expected result	Result
User opens the application.	Application opens.	[PASS] / [FAIL]
User drags a road Type A from the sidebar and places it in Grid B3.	A line with Type a shows in Grid B3.	[PASS] / [FAIL]
User drags a crossroad Type A from the sidebar and places it in Grid C4.	A crossroad with Type A shows in Grid C4.	[PASS] / [FAIL]
User chooses a road Type A, by left clicking the mouse, and then drags it to the grid (grid B2).	A road with TypeA shows in grid B2.	[PASS] / [FAIL]
User chooses a road Type B, by left clicking the mouse, and then drags it in the grid (grid B3).	System gives an error message "Grid occupied". Grid B3 does not change.	[PASS] / [FAIL]
User chooses a road Type A, by left clicking the mouse, and then drags it to the grid (grid C4).	System gives an error message "Grid occupied". Grid C4 does not change.	[PASS] / [FAIL]
User chooses a road TypeB, by left clicking the mouse, and then drags it outside the grid.	System gives an error message "Please places inside the grid."	[PASS] / [FAIL]

Tester name:	Test date:
Result: [PASS] / [FAIL]	Comment:

3.2 Rotating the component

1. Precondition

- Road Type A on Grid B3.
- Crossroad Type B on Grid C4.
- The other Grids are empty.

2. Target on screen

- User chooses a component in the grid.
- User right clicks on the component.
- User selects "Rotate" from right-click menu.

3. Test Data/Simulation

Actions	Expected result	Result
User opens the application.	Application opens.	[PASS] / [FAIL]
User checks whether the play button is enabled.		[PASS] / [FAIL]
User drags a road Type B from the sidebar and places it in Grid B3.	A line with Type a shows in Grid B3.	[PASS] / [FAIL]
User drags a crossroad Type A from the sidebar and places it in Grid C4.	A crossroad with Type A shows in Grid C4.	[PASS] / [FAIL]
User right clicks on grid B3 then chooses rotate option by left clicking the mouse.	Road on grid B3 has rotated 90 degrees clockwise.	[PASS] / [FAIL]
User right clicks on Grid C4 then chooses rotate option by left clicking the mouse.	Crossroad on grid C4 has rotated 90 degrees clockwise.	[PASS] / [FAIL]
User right clicks on grid D1.	Nothing happens and no option pop out.	[PASS] / [FAIL]

Tester name:	Test date:
Result: [PASS] / [FAIL]	Comment:

3.3 Positioning a crossroad

1. Precondition

- Road Type A on Grid B3.
- Crossroad Type B on Grid C4.
- The other Grids are empty.

2. Target on screen

- User click on the Crossroad tab.
- User chooses a crossroad.
- User drags the crossroad from the Crossroad tab.
- User places the crossroad on the grid.
- User sets the initial setting of that crossroad.

3. Test Data/Simulation

Actions	Expected result	Result
User opens the application.	Application opens.	[PASS] / [FAIL]
User checks whether the play button is enabled.		[PASS] / [FAIL]
User drags a road Type B from the sidebar and places it in Grid B3.	A line with Type a shows in Grid B3.	[PASS] / [FAIL]
User drags a crossroad Type A from the sidebar and places it in Grid C4.	A crossroad with Type A shows in Grid C4.	[PASS] / [FAIL]
User chooses a crossroad Type A, by left clicking the mouse, and then drags it to the grid (grid B2). Setting the initial setting in the pop-up window.	A crossroad with TypeA shows in grid B2. System set attributes to that crossroad.	[PASS] / [FAIL]
User chooses a crossroad Type B, by left clicking the mouse, and then drags it to the grid (grid B3).	System gives an error message "Grid occupied". Grid B3 does not change.	[PASS] / [FAIL]
User choose a crossroad Type A, by left clicking the mouse, and then drags it to the grid (grid C4).	System gives an error message "Grid occupied". Grid C4 does not change.	[PASS] / [FAIL]
User chooses a crossroad Type B, by left clicking the mouse, and then drags it to the grid (grid C2).	A crossroad with Type B shows in grid C2. System sets default attributes to that crossroad.	[PASS] / [FAIL]
User chooses a crossroad Type A, by left clicking the mouse, and then drags it to the grid (grid C3). User clicks on the cancel button in the pop-up window.	A crossroad with Type B shows in grid C3. System closes the setting window and without setting attributes.	[PASS] / [FAIL]
User chooses a crossroad Type B, by left clicking the mouse, and then drags it outside the grid.	System gives an error message "Please places inside the grid."	[PASS] / [FAIL]

Tester name:	Test date:
Result: [PASS] / [FAIL]	Comment:

3.4 Configuring traffic light timing

1. Target on screen

- User choose a crossroad in the grid.
- User right clicks on the crossroad.
- User selects "Traffic light configuration" from right-click menu.
- User selects a light group in the listbox of the light groups.
- User change the green light time on the numericUpDown.
- User clicks the "Complete" button.

2. Test Data/Simulation

Actions	Expected result	Result
User opens the application.	Application opens.	[PASS] / [FAIL]
User checks whether the play button is enabled.		[PASS] / [FAIL]
User drags a road Type B from the sidebar and places it in Grid B3.	A line with Type A shows in Grid B3.	[PASS] / [FAIL]
User drags a crossroad Type A from the sidebar and places it in Grid C4.	A crossroad with Type A shows in Grid C4.	[PASS] / [FAIL]
User right clicks on grid C4 then chooses Traffic light configuration option by left clicking the mouse. After selects a light group and changes the time by the numericUpDown, user clicks on Change. User clicks on complete.	System sets the green light time of that chosen light group and close the configuration window.	[PASS] / [FAIL]
User right clicks on grid C4 then choose Traffic light configuration option by left clicking the mouse. After changed the time by the numericUpDown, user click on Change all. User click on complete.	System sets that value to all the light groups and close the configuration window.	[PASS] / [FAIL]
User right clicks on grid C4 then choose Traffic light configuration option by left clicking the mouse. And without changing anything then clicks on close.	System without changing anything and closes the configuration window.	[PASS] / [FAIL]
User right clicks on grid B3.	System only shows the rotate option.	[PASS] / [FAIL]
User right clicks on grid A1.	Nothing happens and no option pop out.	[PASS] / [FAIL]
Tester name:		Test date:
Result: [PASS] / [FAIL]		Comment:

3.5 Deleting the component

1. Target on screen

- User chooses a component in the grid.
- User right clicks on the component.
- User selects "Delete" from right-click menu.

2. Test Data/Simulation

Actions	Expected result	Result
User opens the application.	Application opens.	[PASS] / [FAIL]
User drags a road Type B from the sidebar and places it in Grid B3.	A line with Type a shows in Grid B3.	[PASS] / [FAIL]
User drags a crossroad Type A from the sidebar and places it in Grid C4.	A crossroad with Type A shows in Grid C4.	[PASS] / [FAIL]
User right clicks on grid B3 then chooses rotate option by left clicking the mouse.	Road on grid B3 has deleted.	[PASS] / [FAIL]
User right click on grid C4 then choosea rotate option by left clicking the mouse.	Crossroad on grid C4 has deleted.	[PASS] / [FAIL]

Tester name:	Test date:
Result: [PASS] / [FAIL]	Comment:

3.6 Setting up the amount of incoming cars

1. Target on screen

- User chooses a textbox on one of the components which already in the grid.
- User defines the amount of the cars coming through the roads in that textbox.

2. Test Data/Simulation

Actions	Expected result	Result
User opens the application.	Application opens.	[PASS] / [FAIL]
User checks whether the play button is enabled.		[PASS] / [FAIL]
User drags a road Type B from the sidebar and places it in Grid B3.	A line with Type a shows in Grid B3.	[PASS] / [FAIL]
User drags a crossroad Type A from the sidebar and places it in Grid C4.	A crossroad with Type A shows in Grid C4.	[PASS] / [FAIL]
User clicks on a textbox in grid B3 and input a value.	System sets the amount of incoming car of which user changed in the lane.	[PASS] / [FAIL]
User clicks on a textbox in grid C4 and input a value.	System sets the amount of incoming car of which user changed in the crossroad.	[PASS] / [FAIL]
User clicks on a textbox in grid B3 and without input a value.	System sets the amount of incoming car of which user changed in the lane to default value.	[PASS] / [FAIL]
User doesn't click on a textbox.	Nothing happens.	[PASS] / [FAIL]
Tester name:		Test date:
Result: [PASS] / [FAIL]		Comment:

3.7 Running a simulation

1. Target on screen

- User clicks on "Start/Stop" button.

2. Test Data/Simulation

Actions	Expected result	Result
User opens the application.	Application opens.	[PASS] / [FAIL]
User checks whether the play button is enabled.		[PASS] / [FAIL]
User drags a road Type B from the sidebar and places it in Grid B3.	A line with Type a shows in Grid B3.	[PASS] / [FAIL]
User right clicks on grid B3 then chooses rotate option by left clicking the mouse.	Road on grid B3 has rotated 90 degrees clockwise.	[PASS] / [FAIL]
User drags a crossroad Type A from the sidebar and places it in Grid C4.	A crossroad with Type A shows in Grid C4.	[PASS] / [FAIL]
User right clicks on grid C4 then chooses Traffic light configuration option by left clicking the mouse. After selects a light group and changes the time by the numericUp-Down, user clicks on Change. User clicks on complete.	System sets the green light time of that chosen light group and close the configuration window.	[PASS] / [FAIL]
User clicks on a textbox in grid B3 and input a value.	System sets the amount of incoming car of which user changed in the lane.	[PASS] / [FAIL]
User clicks on a textbox in grid C4 and input a value.	System sets the amount of incoming car of which user changed in the crossroad.	[PASS] / [FAIL]
User click on start.	After user clicks on start, system shows the cars flow go through all the lanes in the components which are already placed in the grid. Also the traffic light time changing like the setting in the configuration.	[PASS] / [FAIL]
The simulation starts.	Cars stop when the light is red and go when the light is green. And the button of pause is available.	[PASS] / [FAIL]

Tester name:	Test date:
Result: [PASS] / [FAIL]	Comment:

3.8 Stopping simulation

1. Target on screen

- User clicks on "Start/Stop" button.

2. Test Data/Simulation

Actions	Expected result	Result
User opens the application.	Application opens.	[PASS] / [FAIL]
User checks whether the play button is enabled.		[PASS] / [FAIL]
User drags a road Type B from the sidebar and places it in Grid B3.	A line with Type a shows in Grid B3.	[PASS] / [FAIL]
User right clicks on grid B3 then chooses rotate option by left clicking the mouse.	Road on grid B3 has rotated 90 degrees clockwise.	[PASS] / [FAIL]
User drags a crossroad Type A from the sidebar and places it in Grid C4.	A crossroad with Type A shows in Grid C4.	[PASS] / [FAIL]
User right clicks on grid C4 then chooses Traffic light configuration option by left clicking the mouse. After selects a light group and changes the time by the numericUp-Down, user clicks on Change. User clicks on complete.	System sets the green light time of that chosen light group and close the configuration window.	[PASS] / [FAIL]
User clicks on a textbox in grid B3 and input a value.	System sets the amount of incoming car of which user changed in the lane.	[PASS] / [FAIL]
User clicks on a textbox in grid C4 and input a value.	System sets the amount of incoming car of which user changed in the crossroad.	[PASS] / [FAIL]
User click on start.	After user clicks on start, system shows the cars flow go through all the lanes in the components which are already placed in the grid. Also the traffic light time changing like the setting in the configuration.	[PASS] / [FAIL]
The simulation starts.	Cars stop when the light is red and go when the light is green. And the button of pause is availabled.	[PASS] / [FAIL]
User clicks on stop.	After user clicks on stop, system stops the simulation. All the cars disappear and the lights stop changing. The pause button is disabled.	[PASS] / [FAIL]
Tester name:		Test date:
Result: [PASS] / [FAIL]		Comment:

3.9 Pausing simulation

1. Target on screen

- User clicks on "Pause" button.

2. Test Data/Simulation

Actions	Expected result	Result
User opens the application.	Application opens.	[PASS] / [FAIL]
User checks whether the play button is enabled.		[PASS] / [FAIL]
User drags a road Type B from the sidebar and places it in Grid B3.	A line with Type a shows in Grid B3.	[PASS] / [FAIL]
User right clicks on grid B3 then chooses rotate option by left clicking the mouse.	Road on grid B3 has rotated 90 degrees clockwise.	[PASS] / [FAIL]
User drags a crossroad Type A from the sidebar and places it in Grid C4.	A crossroad with Type A shows in Grid C4.	[PASS] / [FAIL]
User right clicks on grid C4 then chooses Traffic light configuration option by left clicking the mouse. After selects a light group and changes the time by the numericUp-Down, user clicks on Change. User clicks on complete.	System sets the green light time of that chosen light group and close the configuration window.	[PASS] / [FAIL]
User clicks on a textbox in grid B3 and input a value.	System sets the amount of incoming car of which user changed in the lane.	[PASS] / [FAIL]
User clicks on a textbox in grid C4 and input a value.	System sets the amount of incoming car of which user changed in the crossroad.	[PASS] / [FAIL]
User click on start.	After user clicks on start, system shows the cars flow go through all the lanes in the components which are already placed in the grid. Also the traffic light time changing like the setting in the configuration.	[PASS] / [FAIL]
The simulation starts.	Cars stop when the light is red and go when the light is green. And the button of pause is available.	[PASS] / [FAIL]
User clicks on pause.	After user clicks on pause, system pauses the simulation. All the cars stop and the lights stop changing. The text of the pause button changes into resume.	[PASS] / [FAIL]
Tester name:		Test date:
Result: [PASS] / [FAIL]		Comment:

3.10 Load file

1. Target on screen

- User clicks on "Load" button.

2. Test Data/Simulation

Actions	Expected result	Result
User opens the application.	Application opens.	[PASS] / [FAIL]
User checks whether the play button is enabled.		[PASS] / [FAIL]
User checks that there is at least one saved file with traffic control system on the device.		[PASS] / [FAIL]
User clicks on Load and chooses a file to open.	System closes previous project and loads the chosen file.	[PASS] / [FAIL]
User clicks on Load and chooses a file to open.	System informs user that file cannot be loaded and given the choice to stop or choose another file.	[PASS] / [FAIL]
User clicks on Load and chooses a file to open when another project is open.	System asks if users wants, to save project, that is already open, before closing it and opening another project.	[PASS] / [FAIL]

Tester name:	Test date:
Result: [PASS] / [FAIL]	Comment:

3.11 Save file

1. Target on screen

- User clicks on "Save" button.

2. Test Data/Simulation

Actions	Expected result	Result
User opens the application.	Application opens.	[PASS] / [FAIL]
User checks whether the play button is enabled.		[PASS] / [FAIL]
User checks that there is at least one saved file with traffic control system on the device.		[PASS] / [FAIL]
User clicks on Load and chooses a file to open.	System loads the chosen file.	[PASS] / [FAIL]
User makes a change on the grid by dragging a new line Type A to an unoccupied place in the grid.	System displays the new component in the grid.	[PASS] / [FAIL]
User clicks on Save.	System saves the file.	[PASS] / [FAIL]
User clicks on Save when the project has not been saved on the device yet.	A saving dialogue window pops up.	[PASS] / [FAIL]
User chooses the directory where the file will be stored, and names the file then clicks OK.	System saves the file.	[PASS] / [FAIL]
Tester name:		Test date:
Result: [PASS] / [FAIL]		Comment:

3.12 Save file as a new file

1. Target on screen

- User clicks on "Save as" button.

2. Test Data/Simulation

Actions	Expected result	Result
User opens the application.	Application opens.	[PASS] / [FAIL]
User checks whether the play button is enabled.		[PASS] / [FAIL]
User clicks on Save as.	A saving dialogue window pops up.	[PASS] / [FAIL]
User chooses the directory where the file will be stored, and names the file then clicks OK.	System saves the file.	[PASS] / [FAIL]

Tester name:	Test date:
Result: [PASS] / [FAIL]	Comment:

3.13 Resizing the grid

1. Target on screen

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

2. Test Data/Simulation

Actions	Expected result	Result
User opens the application.	Application opens.	[PASS] / [FAIL]
User checks whether the play button is enabled.		[PASS] / [FAIL]
User clicks on "Edit" on the top bar.	System displays a menu with several options.	[PASS] / [FAIL]
User clicks on "Document settings.	System shows document settings panel.	[PASS] / [FAIL]
User defines the size of the grid by typing numbers in the width and height input boxes, then clicks on OK.	System updates and display the new grid.	[PASS] / [FAIL]
User defines a smaller size of the grid	System discards the objects placed outside of the new grid size.	[PASS] / [FAIL]
Tester name:	Test date:	
Result: [PASS] / [FAIL]	Comment:	

3.14 Closing application

1. Target on screen

- User clicks on "Close" button.

2. Test Data/Simulation

Actions	Expected result	Result
User opens the application.	Application opens.	[PASS] / [FAIL]
User clicks on Close.	System closes the window.	[PASS] / [FAIL]
User clicks on Close when there is unsaved changes.	System asks if user wants to save changes.	[PASS] / [FAIL]

Tester name:	Test date:
Result: [PASS] / [FAIL]	Comment: