# User Requirements specification

GROUP A

Student name: Student number  
Daniel Todorov [2480417]  
Georgi Chishirkov [2530090]  
Ilia Nikushev [2551721]  
Yu Hong [2598515]

Date : 4.12.2015

Version: final

Contents

[User Requirements specification 1](#_Toc437023098)

[Introduction 2](#_Toc437023099)

[Functional Requirements (use cases ) 2](#_Toc437023100)

[User Interface 7](#_Toc437023101)

[Non Functional Requirements 8](#_Toc437023102)

## Introduction

The network flow system allows the users to plan out a pipeline to see measure how the flow would be distributed. The user is provided with various elements that would replicate real life objects to control the flow on the pipeline.

This document will introduce you to the requirements of this system in order for it to provide an easy and intuitive way to structure the pipeline.

## Apendix 1: Definitions

**Element** – Refers to the pump, merger, splitter, adjustable splitter and sink that are used in the program. A pipeline (connection) is not considered an element.

** Pump –** The flow starts from it. Every pump pumps fuel into a pipeline. It has a certain capacity and current flow.

**Capacity** - The maximum amount of fuel that can leave the pump.

**Current flow** - The current amount of fuel leaving the pump every time-unit.

** Pipeline –** A pipelineis the connection between two components in the flow network. It has a current flow of fuel and a **safety limit (critical flow).**

** Splitter –** It separates one input pipeline to two output.Always half of the incoming fuel leaves the splitter via the upper output and half of it via the lower output

** Adj. Splitter –** It is just like a splitter but it has the feature allowing the user to adjust the percentage of fuel that leaves the splitter via the upper output. The rest leaves via the lower output.

** Merger –** The opposite of a splitter. It has two inputs and one output. It merges the incoming fuel together.

** Sink -** This is the destination for the fuel. Every sink has one input and no outputs.

**Drawing panel** refers to the place where the elements and connections will be visualized in our product and **Toolbox** is the place where all the major tools are located. In the User Interface part of this document you can see how they look.

**Connectable point** is the place where an element can be attached to a pipeline. Each element has different in and out flow connectable points.

**Undo stack** is a list that stores various actions that a user has completed (placing an element, deleting an element, placing a connection etc). It can be used to undo mistakes and is called a stack because the actions are stacked and can only be undone in the order they were done in. **Redo stack** is the same but with the opposite effect, it stores actions that the user has undone.

## Functional Requirements (use cases )

**Draw any *element\** into the *drawing panel\****

1. User clicks on any of the elements from the *toolbox\**
2. The system highlights the element clicked
3. The user left clicks inside the drawing panel
4. The system checks if the area is free
5. The element is placed on the drawing panel by the system

Alternatives:

3.1 The user attempts to draw outside the panel – nothing happens

4.1 The user attempts to draw on top of another element – nothing happens

**Remove a placed element**

1. The user selects the remove tool
2. The system highlights the tool
3. The user clicks on an element inside the drawing panel
4. The system removes that element and any connected pipes to it

Alternatives:

3.1 The user clicks outside or not on a element – nothing happens

**Remove a *pipeline\****

1. The user selects the remove tool
2. The system highlights the tool
3. The user hovers the mouse close to the pipeline
4. The system highlights the pipeline connection in red
5. The user clicks on it
6. The system removes the pipeline

Alternatives:

3.1 The user positions the mouse over 2 or more overlapping pipelines – the system highlights the pipeline that was added latest from all of the overlapping

**Select any element inside the drawing panel**

1. User clicks on the select tool from the toolbox
2. The system highlights the tool in the toolbox
3. The user selects an element
4. The system shows the current flow and max flow for the element

Alternatives:

3.1 The user selects anything outside the drawing panel – nothing happens

3.2 The user selects a pipe – nothing happens

4.1 The user has selected an adjustable splitter - a dropdown menu appears where you can adjust the percentages of the outgoing flow

**Move any element inside the drawing panel**

1. The user clicks on the select tool from the toolbox
2. The system highlights the tool selected
3. The user selects an element to drag
4. The user then drags the element
5. The system sets the final position when the dragging has finished as well as the new position for the pipelines

Alternatives:

4.1 The user attempts to drag outside the panel – the element remains within

4.2 The user attempts to drag on top of another element – the element is reverted back to its last known position

**Connect two elements together**

1. The user clicks on the pipeline element button
2. The user hovers on an element’s *connectable point\**
3. The system highlights the element’s point as available
4. The user clicks on an element’s connectable point
5. The system begins the pipeline
6. The system highlights this point as in use
7. The user repeats 3,4 to connect to an end point
8. The system connects the two points with a straight line

Alternatives

3.1The element has an active pipeline at that point – The system highlights the point as unavailable  
 4.1The element has an active pipeline at that point – The system does not allow this pipeline  
 4.2The user clicks outside of the pipeline point – Nothing happens  
 7.1The pipeline results in a circular network - the point is marked as unavailable

**Check flow for the active pipeline**

Pre requirements: Elements have been placed and pipelines between them exist with an outgoing flow. The user has selected a pump

1. The system shows the current outgoing flow and the maximum flow below the toolbox
2. The user changes the value of the current flow closer to the maximum
3. The system checks all pipelines:  
   All pipelines near *critical level\** are colored in red

Alternatives

2.1 The user changes the flow to a negative whole number – the systems sets it to 0

2.2 The user changes the flow above the maximum – the system sets it to the maximum possible

**Undo**

1. The user clicks the undo button
2. The system removes the last change done
3. The system registers to the *redo stack\** this change

Alternatives

1.1Nothing to undo – the button is disabled, un-clickable

**Redo**

1. The user clicks the redo button
2. The system adds the last change un-done
3. The system registers to the *undo stack\** this change

Alternatives

1.1Nothing to redo – the button is disabled, un-clickable

**Create a New pipeline**

1. User clicks the ‘new’ button
2. System checks if there is an active pipeline on the panel
3. The system clears the panel

Alternatives

2.1There is an active pipeline  
 -User is prompted to save or discard their changes before continuing

**Save a pipeline**

1. User clicks the save button
2. The system opens the file dialog
3. The user gives a name and a location and clicks save
4. The system stores the file as .pipeline

Alternatives

3.1The user specifies a file that exists in that directory – System checks if they want to overwrite  
 3.2The user specifies an invalid file name – System informs of the oddity  
 4.1The user’s operating system does not allow the save to occur – System informs the user

**Load a pipeline**

1. User clicks the load button
2. The system opens the file dialog
3. User selects a .pipeline file to load
4. System loads the pipeline

Alternatives

1.1 User has unsaved changes – User is prompted if they want to save  
 3.1 User has selected a file that is corrupt or invalid – The user is informed of the oddity

**Open Right click menu**

1. User clicks with the mouse’s right button somewhere on the drawing panel
2. User processes the click’s location and stores it
3. System disables interactions which would not be available
   1. When clicked on an element – Remove and Cancel are available
   2. When clicked on an empty area – all Add commands and Cancel are available
4. The system shows the right click menu

**Close Right click menu**

1. User requests the menu
2. System shows the menu
3. User clicks on the Cancel button
4. System closes the menu

Alternatives

* 1. User clicks on any other available button within the menu  
     -The system performs the action  
     -The system closes the menu
  2. Right click menu loses focus ( user clicks outside of the form / outside of the drawing panel / somewhere within the drawing panel )  
     -The system closes the menu

**Use Right click menu to remove an element**

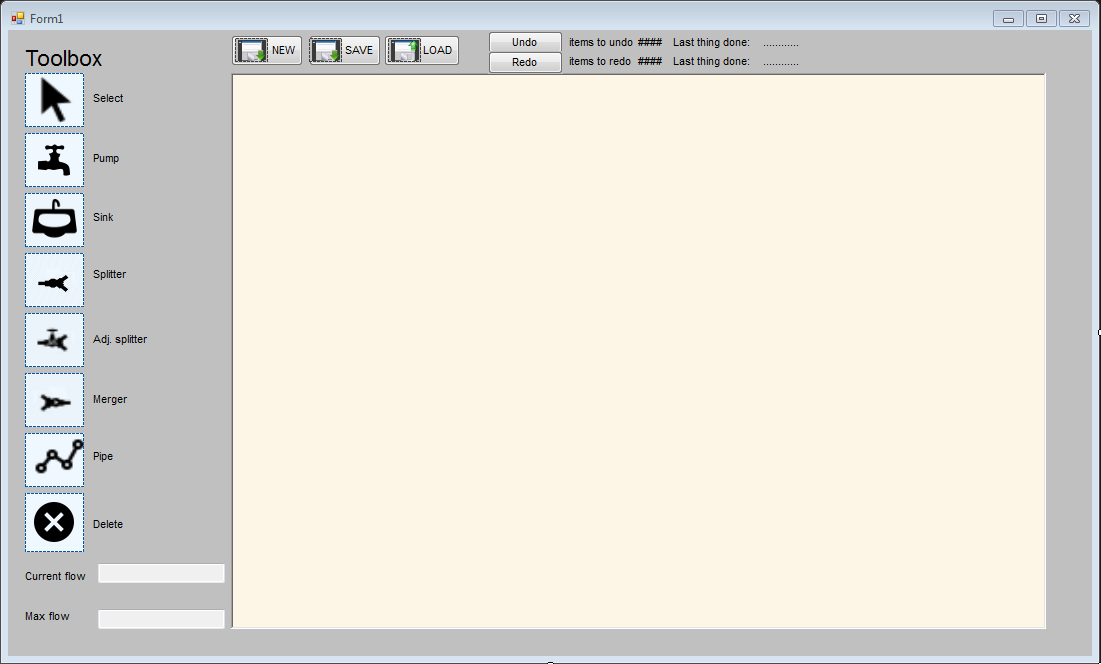
1. System presents the right click menu
2. User clicks on the ‘Remove’ button
3. System removes the element that was targeted

**Use Right click menu to add an element**

1. System presents the right click menu
2. User clicks on any of the Add element buttons
3. System creates a new element designated by the button at the right click position

## User Interface

This is the interface the user will be presented with



## Non Functional Requirements

These are the key features that will outline our projects main quality standards. At run time the 3 main things we are aiming for will be usability, stability and performance. Our program will be easy and intuitive to use, as long as the user has some basic knowledge about flow diagrams. Lastly we will focus on its overall performance and try to ensure everything will run as fast as possible.

The program also focuses on extensibility in the overall structure of the system and the code itself. Meaning that after launch, if we so desire new features will be easily added without having to redesign or recode the program.