

User Requirements Specification Document for Pipelines

Coen Stange, Edgar Kruze, Wen Li

2015/12/25

Background and Context

This User Requirements Specification document specifies the requirements for the application "Pipelines in a flow network". To transport fuel from one place to another place pipelines are used. The system is becoming more complex over time therefore it becomes difficult to manage the flow in the pipeline system. For safety purposes it is very important that the pressure in the pipes don't become too high. The following components are part of the flow system: pump, sink, adjustable splitter and merger.

Contents

1	Functional Requirements	2
1.1	Requirements	2
1.1.1	General requirements	2
1.1.2	System requirements	3
1.2	Use cases	3
2	User Interface	11
2.1	Specifications	12
3	Nonfunctional Requirements	13
3.1	Usability factors	13
4	Decisions and Rules	14
4.1	Algorithm chosen	14
4.2	Default values	14

1 | Functional Requirements

1.1 Requirements

1.1.1 General requirements

Code	Requirement
GEN-010	The program is compatible with Windows 7.
GEN-020	The system allows to design a flow system using the following components: <ul style="list-style-type: none">• Pump• Sink• Adjustable splitter• Merger
GEN-030	The system shows the flow through the pipelines.

1.1.2 System requirements

Code	Requirement
SYS-010	The system allows to add components to the flow network.
SYS-015	The system allows to duplicate components (excluding pipelines).
SYS-018	The system allows to move components.
SYS-020	The components are not allowed to overlap each other.
SYS-030	The system allows to connect a pipeline between an unused output of a component to an unused input of a component.
SYS-040	The system allows to remove components including pipelines.
SYS-040A	All the pipelines connected to the removed component will also be removed.
SYS-050	The system allows to change the flow output of a pump within his range.
SYS-060	The system allows to change the distribution of the flow from splitter going to the top and bottom output.
SYS-070	The system displays the flow going through the pipelines.
SYS-070A	The system displays the pipeline red in case the flow exceeds the safety limit.
SYS-080	The system displays the flow transported to a sink.
SYS-090	The system allows to export the flow network to a file.
SYS-100	The system allows to import a flow network from a file.

1.2 Use cases

Use Case 1	Adding new component on flow network
-------------------	---

<i>Level:</i>	User-goal
---------------	-----------

<i>Primary Actor:</i>	End-User
-----------------------	----------

<i>Preconditions:</i>	Component selected from toolbar
-----------------------	---------------------------------

<i>Main Success Scenario:</i>	
-------------------------------	--

1. User selects place on flow network grid
2. System places component on grid with default values see section 4.2
3. System updates internal state

Extensions:

- 2.a System can't place component in selected area
 1. System informs end-user that component can't be placed, because on that point already exists component.
 2. End of use case
-

Use Case 2 Changing flow output of pump

Level: User-goal

Primary Actor: End-User

Preconditions: User selected pump on flow network grid

Main Success Scenario:

1. System opens property sidebar
 2. User can change current flow and maximum flow of pump
 3. System updates values of component and calculates the flow through the network
 4. System updates the changes on the screen
-

Extensions:

2.a User gives a value for current flow larger than maximum flow

1. Current flow is set to the maximum flow

2.b User gives a negative number or non-numeric value

1. Current flow is set to 0

Use Case 3	Changing flow output in splitter component
-------------------	---

<i>Level:</i>	User-goal
---------------	-----------

<i>Primary Actor:</i>	End-User
-----------------------	----------

<i>Preconditions:</i>	User selected splitter on flow network grid
-----------------------	---

Main Success Scenario:

1. System opens property sidebar
 2. User set the division between outputs in the property sidebar using a slider
 3. System updates values of component and calculates the flow through the network
 4. System updates the changes on the screen
-

Use Case 4	Creating connection between components
-------------------	---

<i>Level:</i>	User-goal
---------------	-----------

<i>Primary Actor:</i>	End-User
-----------------------	----------

<i>Preconditions:</i>	Selected pipeline drawing tool from toolbar
-----------------------	---

Main Success Scenario:

1. User selects starting component
 2. User selects point on grid, through which pipeline will go
 3. User repeats step 2, until he has selected all points, through where pipeline will go
 4. User selects ending component
 5. System updates internal state
 6. System updates the changes on the screen
-

Extensions:

- 1.a Selected component already has no unused output
 1. System informs end-user that the selected component can't be used.
 2. User returns to step 1.
 3. End of use case.
 - 2.a Selected point on grid is occupied by component.
 1. System ignores the selected point.
-

Use Case 5	Deleting component on flow network grid
-------------------	--

<i>Level:</i>	User-goal
---------------	-----------

<i>Primary Actor:</i>	End-User
-----------------------	----------

<i>Preconditions:</i>	Component selected
-----------------------	--------------------

Main Success Scenario:

1. User right clicks on the component
2. System shows context menu with options
3. User clicks on Delete
4. System removes selected component
5. System removes pipelines connected to component, if component isn't pipeline
6. System updates internal state
7. System updates the changes on the screen

Use Case 6	Duplicating component on flow network grid
-------------------	---

<i>Level:</i>	User-goal
---------------	-----------

<i>Primary Actor:</i>	End-User
-----------------------	----------

<i>Preconditions:</i>	Component selected
-----------------------	--------------------

Main Success Scenario:

1. User right clicks on the component
2. System shows context menu with options
3. User clicks on Duplicate
4. Go to Use case 1
5. System sets the properties the same as original component

Extensions:

2.a User right click on pipeline

1. Context menu don't show the option to duplicate
2. End of use case.

Use Case 7	Moving component (excluding pipeline) on flow network grid
-------------------	---

<i>Level:</i>	User-goal
---------------	-----------

<i>Primary Actor:</i>	End-User
-----------------------	----------

<i>Preconditions:</i>	Component selected
-----------------------	--------------------

Main Success Scenario:

1. User drags component to new position
 2. System redraws flow network
-

Use Case 8	Load file
-------------------	------------------

<i>Level:</i>	User-goal
---------------	-----------

<i>Primary Actor:</i>	End-User
-----------------------	----------

<i>Preconditions:</i>	User clicks on "Open" file button
-----------------------	-----------------------------------

Main Success Scenario:

1. System shows a file dialog-box.
2. User selects file to open.
3. System closes previous project.
4. System loads the file.

Extensions:

1.a Another project is open

1. System asks if users wants, to save the project, if the user chooses cancel that will end the use case.

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.

Use Case 9 Save file

Level: User-goal

Primary Actor: End-User

Preconditions: User click the "Save" file button

Main Success Scenario:

1. System saves the file.

Extensions:

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

1. Go to Use Case 10
-

Use Case 10	Save file as a new file
<i>Level:</i>	User-goal
<i>Primary Actor:</i>	End-User
<i>Preconditions:</i>	User clicks "Save as" button.
<i>Main Success Scenario:</i>	
<ol style="list-style-type: none"> 1. System shows a file dialog-box. 2. User chooses the directory where the file will be stored, and names the file. 3. System saves the file. 	
<i>Extensions:</i>	
<ol style="list-style-type: none"> 1.a User didn't intend to save file <ol style="list-style-type: none"> 1. User clicks "Cancel". 2. End of use case 	

2 | User Interface

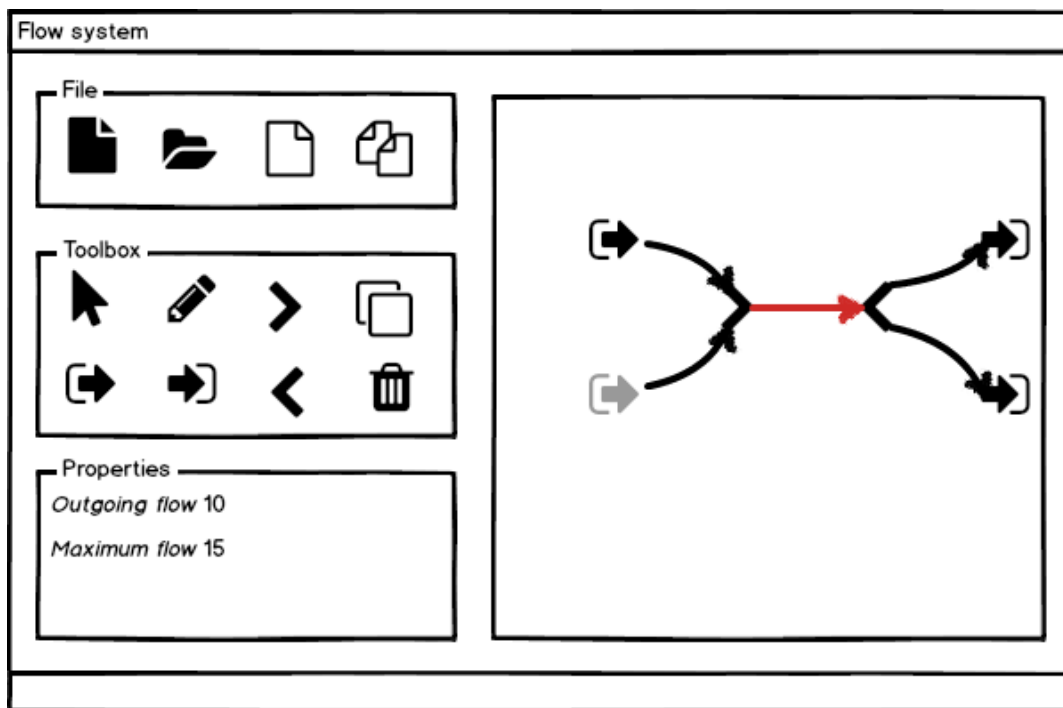











Figure 2.1: Mockup

2.1 Specifications

Code	Specification
UIS-010	An open file dialog box will be showed by pressing  .
UIS-020	A save file dialog box will be showed by pressing  .
UIS-025	A save as file dialog box will be showed by pressing  .
UIS-030	The sidebar contains a toolbox.
UIS-030A	The application has different modes, only one button can be selected in the toolbox.
UIS-040	The normal mode is selected by default  .
UIS-040A	In normal mode a component can be selected by clicking on it.
UIS-040B	A selected item is outlined and the properties can be changed in the properties window.
UIS-040C	A selected item can be moved by dragging.
UIS-040D	In normal mode a context-menu appears when pressing right-click on a component which gives the following options to the component: duplicate and delete.
UIS-050	The draw mode can be selected by pressing  .
UIS-050A	The draw mode allows to draw a pipeline between an unused output of a component to an unused input of another component.
UIS-60	A component can be added to the network by selecting a component from the toolbox and then click the drawing.  Pump  Sink  Splitter  Merger
UIS-100	After a component is added to the flow network the mode is set back to normal.
UIS-110	The sidebar has a properties box which allows to edit the properties of the a selected component.
UIS-120	The flow network is shown in the drawing.
UIS-130	Each component displays the current flow except for splitter and merger.
UIS-140	Components that have missing connectors on the input or output are displayed in orange.
UIS-150	Pipelines that exceed the maximum flow are shown in red.

3 | Nonfunctional Requirements

Code	Requirement
NON-010	The program has an user friendly design.
NON-020	The software is made testable
NON-030	The program has good performance whereby the application is responsive.
NON-050	The program is always available when the user needs it.
NON-060	The program is maintainable.
NON-070	The program is efficient so the task can be done as quick as possible.
NON-080	The program is stable, the program doesn't show any errors.
NON-090	The program has a good user experience which means it is a pleasant working experience for the user.

3.1 Usability factors

Code	Requirement
USF-010	The application is designed to be used with a mouse.
USF-020	Buttons have appropriate sizes to avoid mis-clicks.
USF-030	The colours used for representing different meanings should have clear definitions.

4 | Decisions and Rules

4.1 Algorithm chosen

We choose to use the mathematical way to calculate the flows. The reason is that it is easier for us to do the calculation and programming, though in the reality, the way the system works may seem bizarre. For example when an output is not connected to a splitter the flow is mathematically divided.

4.2 Default values

We set the default numbers of pipes and pumps to 0, and the default value of each splitter to 50/50. However, the value of each splitter is adjustable. With these default settings, our system has a clearer starting point since relative information is not mentioned in the requirement. Since an normal splitter can also be achieved with an adjustable splitter it is easier to remove the normal splitter to prevent confusion.