LIQUID: 2-D FLUID DYNAMICS SIMULATOR

Software Requirements Specification

Version 1.1

10/05/2015

PRYMM
Pavan Kumar Gade, Radhika Panchal,
Yashraj Sinha, Minghua Liu and Manoj Mathe

Prepared for
Principles of Software Engineering
Fall 2015

Revision History

Date	Description	Author	Comments
09/22/15	Initial draft version	Pavan Kumar Gade, Radhika Panchal, Yashraj Sinha, Minghua Liu and Manoj Mathe	Version 1.0 – Initial draft prepared and reviewed by complete team. SRS has been documented based on list of requirement listed in Project Requirements document on Blackboard Individual contributions in documentation: 1 (Pavan,Yashraj and Minghua) 2 (Pavan and Yashraj) 3.1 (Yashraj and Minghua) 3.2 (Minghua and Yashraj) 3.3 (Radhika, Minghua and Yashraj) 3.4 (Minghua, Yashraj and Radhika) 3.5 (Manoj) 3.6 (Minghua and Yashraj) 3.7 (Manoj) 3.9 (Minghua and Yashraj) 4.1 (Radhika, Minghua and Yashraj) 4.2 (Radhika, Minghua and Yashraj) 4.3 (Radhika, Minghua and Yashraj) 5 (Yashraj and Minghua)
10/05/15	Version 1.1 – Updated version includes modifications based on client review comments. Changes were mainly in non-functional requirement section.	Pavan Kumar Gade, Radhika Panchal, Yashraj Sinha, Minghua Liu and Manoj Mathe	Version 1.1 – Updated version includes modifications based on client review comments. Changes were mainly in non-functional requirement section. Below are the sections updated with contributors: 1.4 References - (Minghua and Yashraj) 2.1 Product Perspective (Yashraj, Minghua and Pavan) 2.3 User Characteristics (Minghua and Yashraj) 3.1.1.2 Input interface (Minghua and Yashraj) 3.2.2 Configuration parameters added (Minghua and Yashraj)

			3.3 Use Case removal Radhika 3.5 Non-Functional Requirements (Manoj,Minghua and Yashraj) 3.7 Design Constraints (Manoj,Minghua and Yashraj) 4.2 DFD update (Minghua,Yashraj and Radhika)

Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date

Table of Contents

1. Introduction

This section provides an overview of the Software Requirements Specification (SRS) for the project called LIQUID which is a 2-D Fluid Dynamics Simulator. The purpose of this document is to present in depth clarity of the requirements associated with the project to its intended audience. And the scope of the project gives a particular focus on what the software will do and the benefits related to it. The complete overview of the project is provided in this document for a complete and easy understanding for the reader.

1.1 Purpose

The purpose of the SRS is to outline both the functional and non-functional requirements of the project Fluid dynamics simulation. This document also provides a detailed profile of external interfaces, design constraints and performance considerations. This document is created to contain the detailed information of the project like the software requirements needed to develop the project. The document should act as a foundation for efficient and well-managed project completion and further serve as an accurate reference in the future.
And overall the document is divided for two types of audience. The primary audience of the SRS document will be development group, where they plan the project and develop the project and implements the project on the basis of client requirements. These are all done with the help of the SRS document only. And the secondary document audience comprises the other stakeholders of the project. The SRS will convey and confirm the required functionality and represent an agreement between the involved parties.

1.2 Scope

The software product to be produced is fluid dynamic simulator. The scope of the project is to demonstrate the behavior of various fluids in different environment. This shall be demonstrated by highlighting properties of fluid like viscosity, density, velocity etc. The software product will help us to read the properties of the fluid used for simulation such as density and velocity at any point of the running simulation. Here mainly we are trying to find out the behavior of certain fluid in certain condition so that we can clearly understand the complete picture of that fluid for any of its application in similar condition. The objective and goal of the product is to show the data and record the data during the simulation run in order to understand the dynamics of fluid for certain environment. Here it provides both textual and animated representation of what is happening during execution.

1.3 Definitions, Acronyms, and Abbreviations

Here in this section it describes about the definitions and acronyms and abbreviations that we are using all over the SRS document. This is done for better understanding of the reader about the terms that is used throughout the document.

- SRS: Software Requirement Specifications

- cms: Centimeters
- MS: Microsoft
- GUI: Graphical User Interface
- DFD: Data Flow Diagram
- STD: State Transition Diagram
- FPS: Frames per Second
- MTBF: Mean Time Between Failure

1.4 References

1. Oliver's simple fluid dynamics simulator based on the Naiver-Stokes equations, *https://nerget.com/fluidSim/*
2. Daniel V. Schroeder, A lattice-Boltzmann fluid simulation in JavaScript, *http://physics.weber.edu/schroeder/fluids/*

1.5 Overview

This SRS document Contains general description of the product's perspective and its functions. This section also covers different characteristics of a user and list of constraints in this project. It also has Specific Requirements which is the major part of this SRS Document. In here the requirements are mainly classified into functional and non-functional requirements. It also describes the External Interface Requirements and use-case diagrams which has Classes and Objects. There are few other requirements mentioned in this Specific Requirements, those are the Inverse Requirements, Design Constraints, Logical Database Requirements and Other Requirements .This SRS Document also depicts Sequence Diagrams, Data Flow Diagrams and State-Transition Diagrams. Change Management Process is also included in order to mention manage change during the complete life cycle of the project.

2. General Description

Here in this section, general description of the product is covered. In this section product is compared with the other similar projects. Functionality of the project is also covered in this section along with any assumptions and dependencies.

2.1 Product Perspective

The application has to be an installable application. Being an installable. Like other existing web based application which are mentioned in the reference section of SRS, this project also uses thermal lattice Boltzmann method to simulate fluid behavior and it also provide similar user

interface to setup the configuration for the simulation. This project is a complete project in itself.

2.2 Product Functions

This section gives an overview of functioning of the product. First of all this software is an executable file and will run on MS-Windows platform. The software allows the end user to understand the behavior and properties of the fluid. The software shall allow the user to configure simulation settings such as viscosity, fluid density, flow speed etc. Software should also provide output to the end user in both textual data and in animation form. It shows clearly how the fluid is behaving under certain conditions given configured by the end user. System shall also provide an option to pause the fluid simulation whenever the user wants to. Along with that system should also provide the option to add or remove barriers and modify shape of the barrier.

2.3 User Characteristics

There is no specific user characteristics for this software

2.4 General Constraints

This section describes the restrictions on the development environment of the application.
These constraints can be an algorithm or a development environment. In this project development team should use Lattice Boltzmann method to simulate the behavior of the fluid. The project should be developed to be an installable application which is modular in nature. By modular it means that the GUI and rendering engine should be able to be separated.

2.5 Assumptions and dependencies

This application will run on MS-Windows platform. The application which we are about to create will contain a GUI and a back end. Both of these units should be separable in order to work with any other GUI or back end with minimal integration.

3. Specific Requirements

This section will provide more clarity to the design and development team of the project. This section covers detailed requirement of the project. This section comprises of requirements related to external interface, functional requirement, use cases, classes and non-functional requirement.

3.1 External Interface Requirements

3.1.1 User Interfaces

The system user interface should provide both input and output interfaces to the user to interact with the software. Firstly, the user input interface should allow user to configure the simulation environment so that he can setup the configuration to be simulated. Secondly, the system shall provide output to the user in visual and textual form for the behavior of the fluid under simulation.

3.1.1.1 Output interface
Visual Output: Visual representation of the output shall be represented in a 2-Dimentional rectangular screen showing various properties of fluid such as fluid density at specific space and time in the rectangular screen.

Textual Output: Textual representation of the output shall be displaying parameter names and their values such as density at flow meters and break points.

Logging capability: User should be provided a log at the end of the simulation run.

3.1.1.2 Input interface
User should be provided with interface to control and configure various simulation parameters listed below:

- Viscosity: Ranging from 0.005 to 0.2000 cm2/s
- Type of liquid: At least water and glycerin
- Flow speed of liquid: Ranging from 0.000 to 0.120 cm/s
- Size of rectangular simulation container: 6x12cms and 5x10cms
- Temperature: Ranging from -100 to +100 *c
- Density of fluid: Ranging from 1 to 3500 kg/m3
- Exit path: User can enable exit which resembles a small diameter pipe than that of the diameter of the pipe used for the simulation.

3.1.2 Hardware Interfaces

The system has no hardware interface requirements.

3.1.3 Software Interfaces

The system has no software interface requirements.

3.1.4 Communications Interfaces

The application shall be developed in such a way that the rendering engine and the GUI are independent. Hence system should have a communication interface layer between GUI and rendering engine allowing developers to use other GUIs created on the same operating system with minimal integration

3.2 Functional Requirements

This section describes specific features of the software project. If desired, some requirements may be specified in the use-case format and listed in the Use Cases Section.

3.2.1 The system shall enable users to set up the spatial configuration

- Introduction:

The system should only allow user to configure the spatial configuration through the control panel.

- Inputs:

None

- Processing:

The system will read the values configured by users in control panel for later process.

- Outputs:

None

- Error Handling:

All the controls should be display a legal range as defined for values. User should not configure by self-input thus no error input will occur.

3.2.2 The system shall be able to control various parameters: viscosity, type of liquid, flow speed, temperature, size of rectangular simulation container, density of fluid and exit path from the container.

- Introduction:

The system should pass the values configured by users to viscosity, type of liquid, flow speed, temperature, size of rectangular simulation container, density of fluid and exit path from the container in order to adjust the simulation accordingly

- Inputs:

None

- Processing:

Values read from the user input should be assigned to viscosity, type of liquid, flow speed, Size of rectangular simulation container, Temperature, Density of fluid and Exit path in the system accordingly

- Outputs:

None

- Error Handling

The value read from user input should be in the legal range of each factors

3.2.3 The system shall visualize flow density dynamically

- Introduction

The system should display in a 2-Dimentional rectangular screen showing various properties of fluid such as fluid density at specific space and time in the rectangular screen

- Inputs

User configurations

- Processing

The system should apply user configurations and calculate necessary

- Outputs

New simulation should be shown on the display frame.

- Error Handling

System will discard invalid user input

3.2.4 The system shall provide both textual and animated representation of what is happening during execution

- Introduction

The system should provide animated representation based on the configurations and textual representation of the output displaying parameter values such as density at flow meters and break points

- Inputs

Any event happening during execution

- Processing

System should gather data of events that is happening during execution

- Outputs

System should give textual as well as animated output for any event.

- Error Handling

None

3.2.5 The system shall be able to provide on-demand information about flow values about any location in the environment

- Introduction

The system should give user detailed flow values about available locations in the environment if user requests such information.

- Inputs

User's requests for flow information about location in the environment

- Processing

System should gather all the information as per user's request and give such back to user.

- Outputs

System will display user's requested information

- Error Handling

System should not response to an invalid request

3.2.6 The system shall provide the ability to the user to place monitoring points (flow meters) to show information during execution without affecting the behavior of the liquid

- Introduction

The system should let user place monitoring points on the display frame and show information on those points about the current liquid information.

- Inputs

User's placing monitoring points on the display frame

- Processing

System should collects fluid information on the points where user put for monitoring.

- Outputs

System should show fluid info on the points to user.

- Error Handling

System should not enable user set points on the initial force and outside the area of container..

3.2.7 The system shall be able to clearly indicate when activities begin and end

- Introduction

The system should record begin and end time for each event and activity

- Inputs

System activities and events caused by user behaviors

- Processing

System should add begin and end time information for each events and activities.

- Outputs

In system log file, each activity and event should encompass begin and end time information

- Error Handling

None

3.2.8 The system shall be able to provide detailed logging about state of fluids at break points, flow meters, overall environment, provide information about properties and activities at each increment of execution

- Introduction

The system should generate log file recording state of fluids at break points, flow meters, overall environment; provide information about properties and activities at each increment of execution.

- Inputs

User's requests during execution

- Processing

System should record state of fluids at break points, flow meters, overall environment, information about properties and activities at each increment of execution.

- Outputs

Log file should be generated by system recording state of fluids at break points, flow meters, overall environment and information about properties and activities at each increment of execution.

- Error Handling

System should not respond to user's illegal behavior thus log file should not contain such information of requests.

3.2.9 The system shall be able to provide text file logging capability so users can retrieve their results and trace the execution

- Introduction

The system should provide text file logging function for users to retrieve their results and trace execution

- Inputs

User's request for retrieving their tracing results during the execution

- Processing

System should generate log file for user's request

- Outputs

Log file containing requested information should be provided to user.

- Error Handling

System should not respond to user's illegal behavior thus log file should not be provided for such information of requests

3.2.10 The system shall be able to provide replay log capability (replay the animation of a prior execution) by reading a previously saved log file

- Introduction

The system should replay animation of a prior execution by reading a previously saved log file

- Inputs

User's file of previously saved log for replay

- Processing

System should be able to read the log file and verify the validity of the value in it.

- Outputs

Replay the animation as per the log file

- Error Handling

System should not replay if there's any illegal values or operations in that log file

3.2.11 The system shall accommodate change in temperature and fluid density.

- Introduction

The system should replay respond to the change in temperature and fluid density

- Inputs

User's settings on temperature and fluid density

- Processing

System should process the change of temperature and fluid density and display the same

- Outputs

Graphic display as per the change of temperature and fluid density

- Error Handling

System should not let user configure invalid values for temperature and fluid density

3.2.12 The system shall use the Metric System.

- Introduction

The system should integrate Metric System

- Inputs

None

- Processing

None

- Outputs

Metric system is integrated and deployed on the system

- Error Handling

None

3.2.13 The system should support the simulation of H2O and glycerin

- Introduction

The system should simulate fluid of H20 and glycerin

- Inputs

User's choosing type of the fluid between H20 and glycerin

- Processing

System should display the type of fluid that user has chosen. There will be a default value before the execution

- Outputs

System will display the simulation of the chosen type of fluid

- Error Handling

None

3.2.14 User shall manually indicate the end of the simulation

- Introduction

The system should let user manually indicate the end of simulation

- Inputs

User's behavior of terminating the execution

- Processing

System should stop itself from running if receive command from user

- Outputs

System is no longer simulating the fluid

- Error Handling

None

3.2.15 Product environment shall be a rectangle with configurable sides, like a glass of water or a pipe.

- Introduction

The system should be a rectangle with configurable sides containing all the controlling features

- Inputs

None

- Processing

None

- Outputs

System should include its configurable sides in a rectangle

- Error Handling

None

3.2.16 Rectangular and spherical obstacles shall be addable to the container which will inherit height from container height in the simulation

- Introduction

The system should enable user add rectangular and spherical obstacles on the display frame

- Inputs

User adding rectangular and spherical obstacles on the display frame

- Processing

System should be able to let user add 2 different obstacles: rectangular and spherical.
System should also change the fluid dynamically depending on the obstacles where and which user has added

- Outputs

System should display the fluid effect after user's placing the obstacles

- Error Handling

System should not display or let user add obstacles where initial force is located.

3.2.17 The configuration shall have the ability to set-up an exit way from the container to examine fluid behavior (exit can resemble a pipe)

- Introduction

The system should enable user set an exit way from the container

- Inputs

Adding an exit way in the display frame

- Processing

System should re-calculate and alter the fluid near the exit way after user put one.

- Outputs

System should display the updated fluid effect after the exit way is added

- Error Handling

System should not display and let user add this exit way where the initial force is located.

3.2.18 Breakpoint monitor data at certain point in container

- Introduction

The system should calculate monitor and display flow location, velocity, density of fluid and at breakpoints

- Inputs

Breakpoints should be placed in advance

- Processing

System should monitor and record data where breakpoints are placed

- Outputs

Monitoring data should be displayed accordingly once the breakpoint is set

- Error Handling

None

3.2.19 Breakpoint can be labeled by increment.

- Introduction

The system should be able to label breakpoints by increment

- Inputs

Breakpoints should be placed in advance

- Processing

System should add label to these breakpoints placed by increment

- Outputs

Breakpoints are labeled and can be seen by user

- Error Handling

None

3.2.20 All the results of simulation shall be written into a log file, settings and breakpoint info.

- Introduction

The system should generate log containing settings and breakpoints info during the execution of the simulation

- Inputs

Breakpoints should be placed in advance

- Processing

System should record the data of breakpoints and other settings and write such info into a log file

- Outputs

Log file should contain settings and breakpoint info

- Error Handling

None

3.2.21 There will only be one source of initial force in a given simulation. Initial force is configurable

- Introduction

The system should have only one initial force generating the fluid and this force is configurable

- Inputs

None

- Processing

System should only enable user set one initial force and configure the force by default value. The values for the force can be modified by user in a given legal range respectively

- Outputs

Configuration frame should contain buttons for setting initial force for user. Initial force should be seen after user put onto the display frame

- Error Handling

Values configured out of the range should be discarded and cannot be applied to the initial force
System should not enable user set a second force

3.2.22 Fluid velocity (steady flow) is configurable.

- Introduction

System should enable user to configure fluid velocity

- Inputs

User's configuring fluid velocity

- Processing

System should provide interface for user to configure fluid velocity and apply the value to the simulation

- Outputs

Fluid display should be updated as per the settings of fluid velocity

- Error Handling

Fluid velocity should be defined in a certain range and forbid user configure outside of the range

3.2.23 The scale of the simulation (container size) will be configurable to either meters (m) or centimeters (cm) and will remain constant during execution.

- Introduction

The system should enable user configure the container size in centimeter before the execution

- Inputs

User's changing the container size of the simulation

- Processing

System should change its display frame as per the user's configuration

- Outputs

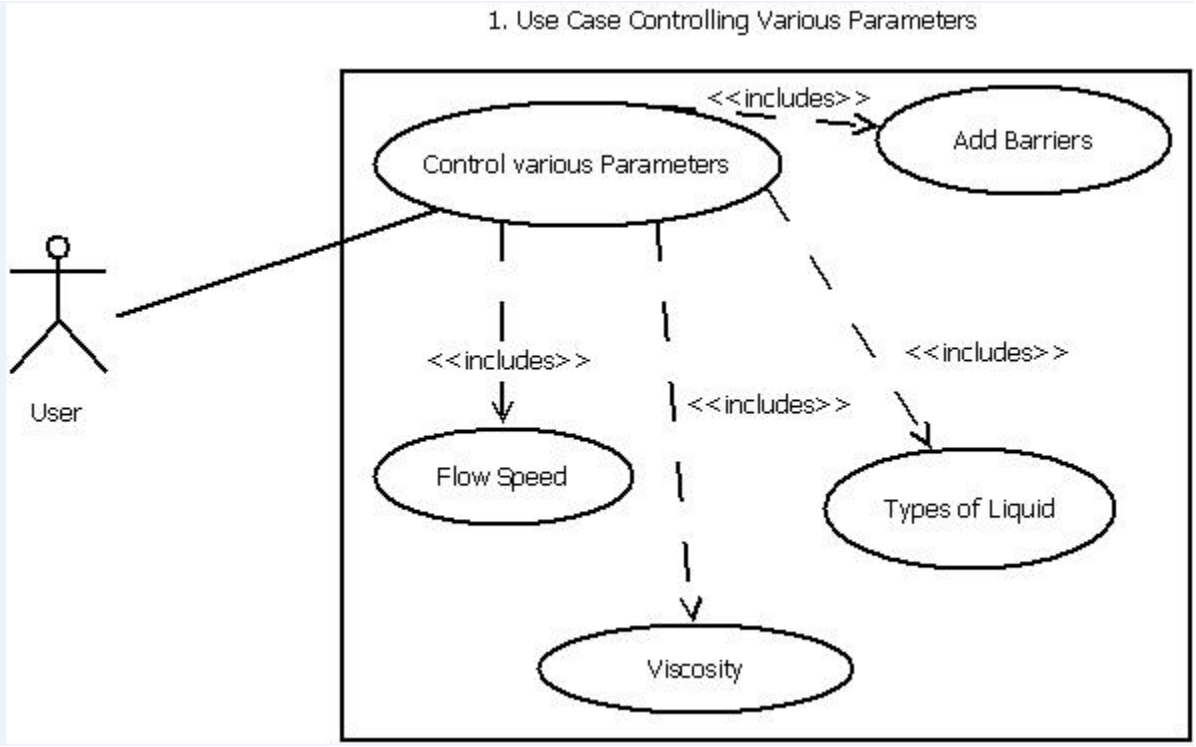
System's display frame is changed according to the values configured by user

- Error Handling

System should no longer change the frame after the execution. All the values of the container size should be constant during the execution

3.3 Use Cases

3.3.1 Use Case 1

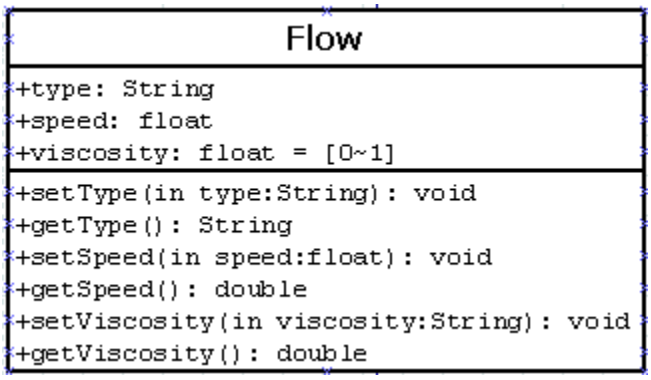


3.4 Classes / Objects

This section covers classes and functions. This section helps design and development team in better understanding the software requirement. The classes covered in this section will have two units in it attributes and functions. Attributes sub-section will have data or properties of the object whereas the other sub-section which is function will contain methods to operate on attributes.

3.4.1 Flow / flow

Class Flow contains basic attributes of the fluid and operations that are needed for setting and getting values of each attribute.



3.4.1.1 Attributes

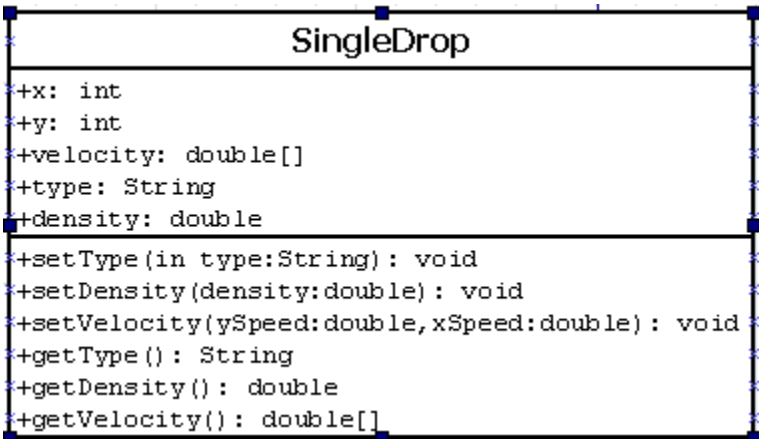
As depicted above

3.4.1.2 Functions

As depicted above

3.4.2 SingleDrop / singleDrop

Class SingleDrop contains information of every points defined in the rectangle. For each SingleDrop instances, it should be initialized at the beginning and can be changed as per the fluid simulation.



3.4.2.1 Attributes

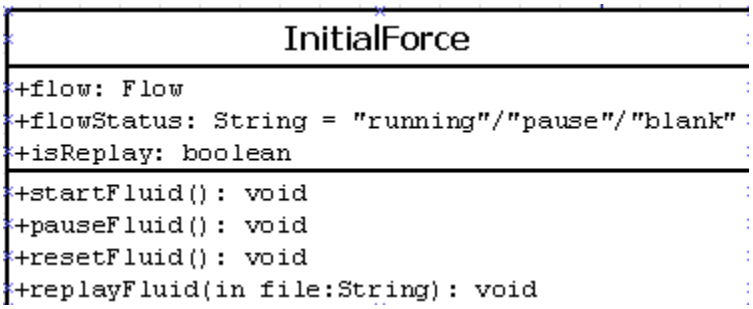
As depicted above

3.4.2.2 Functions

As depicted above

3.4.3 InitialForce / initialForce

Class InitialForce is what we use to generate fluid thus it contains basic information of the fluid itself. It triggers the fluid motion and provide interfaces for pausing, resetting, replaying fluid simulation.



3.4.3.1 Attributes

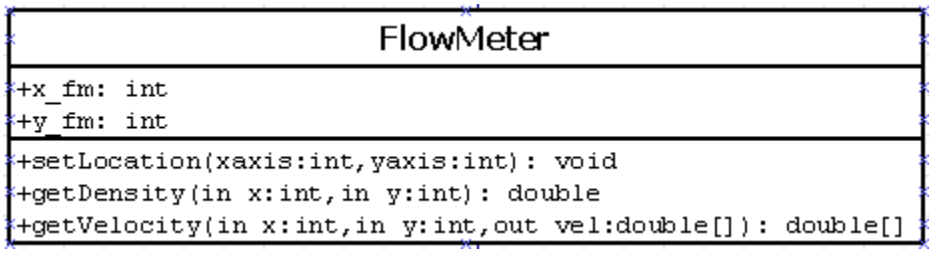
As depicted above

3.4.3.2 Functions

As depicted above

3.4.4 FlowMeter / flowMeter

Class FlowMeter is instantiated whenever user place a flow meter so that the values at the flow meter can be monitored.



3.4.4.1 Attributes

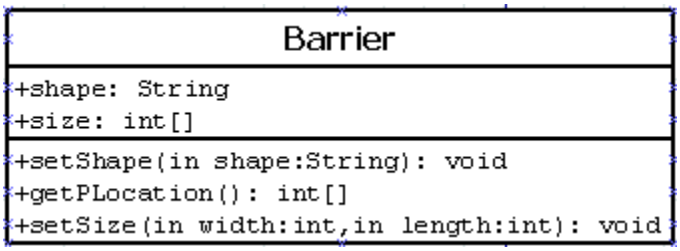
As depicted above

3.4.4.2 Functions

As depicted above

3.4.5 Barrier / barrier

Class Barrier is instantiated whenever user place a barrier with different shapes to observe the behavior of the fluid under force.



3.4.5.1 Attributes

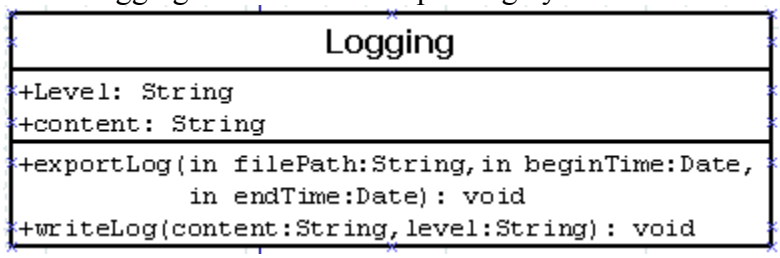
As depicted above

3.4.5.2 Functions

As depicted above

3.4.6 Logging / logging

Class Logging can be used to export log by user and write log to file by the system.



3.4.6.1 Attributes

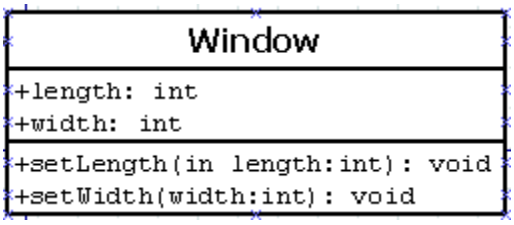
As depicted above

3.4.6.2 Functions

As depicted above

3.4.7 Window / window

Class Window can be used to configure the size of rectangle which is used for the visualization of the simulation.



3.4.7.1 Attributes

As depicted above

3.4.7.2 Functions

As depicted above

3.5 Non-Functional Requirements

This section will cover the various topics related to non-functional requirements such as maintainability, performance, availability.

3.5.1 Performance

- The Frame rate of the fluid simulator should be more than 70fps.
- The response time for user to retrieve the log from the system should not be more than 25 seconds.
- The response time for the system to start replay from the entered log file should not exceed more than 30 seconds.
- The software should log the data at every one second into the log file.
- The log file of the software should not exceed 5MB.
- The system should not allow user to import a file which is larger than 5MB.

3.5.2 Reliability

Considering an average simulation execution time of 5 minutes. Application shall work for 20 cycles without crashing, resulting in an MTBF of 100 minutes for the software.

3.5.3 Availability

The application shall be responsive at all times during the simulation.

3.5.4 Security

Not applicable to this project

3.5.5 Maintainability

- The product must include an installer script or an installer program
- The product must include source code with compilation instructions if any and list of used libraries
- The product must include all documents artifacts generated during the engineering life cycle
- The product must include an installation guide
- The product must include a user guide with screen shots
- The product must include a programmers guide
- All document artifacts produced during the project developments and for the final release must be typed and electronically generated

3.5.6 Portability

- The application is installable for MS-Windows environment.

3.6 Inverse Requirements

- The system should keep every controls off during the execution.(non- functional)

3.7 Design Constraints

- The application shall be developed in such a way that the rendering engine and the GUI are independent. Allowing developers to use other GUI's created on the same operating system (but not necessarily with the same language) with minimal integration.
- The software can include only one third party component.

3.8 Logical Database Requirements

Not applicable to this project

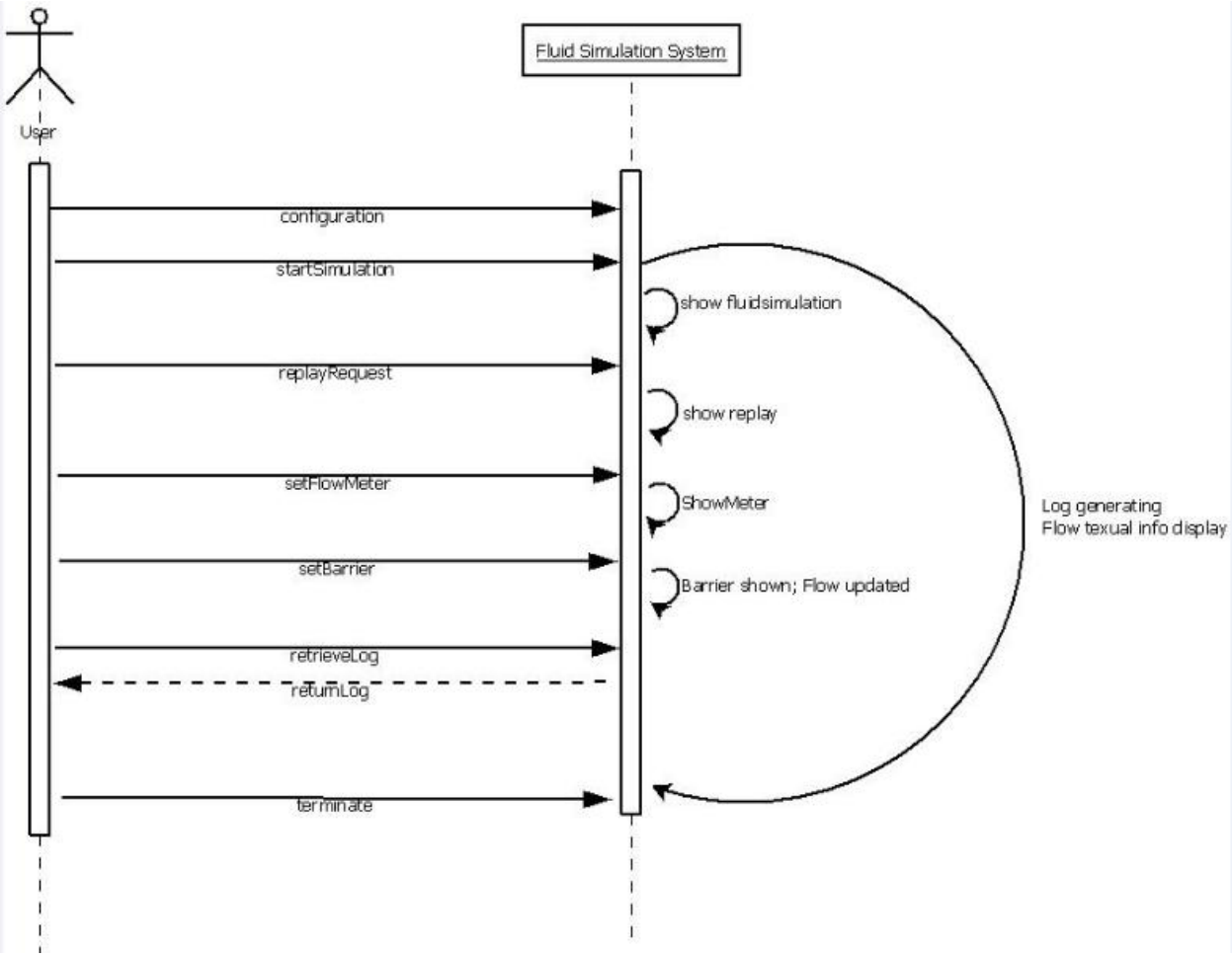
3.9 Other Requirements

Point of view is as if user is looking down from above the container.

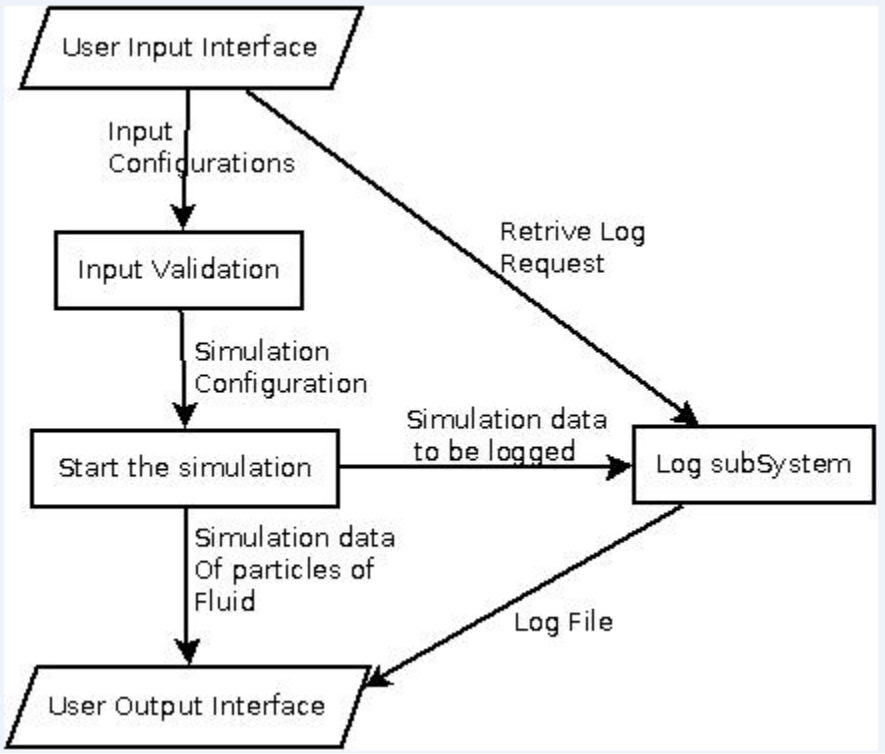
4. Analysis Models

This section consist of sequence diagram, Data Flow Diagram and State Transition Diagram to represent the behavior of the system in pictorial form.

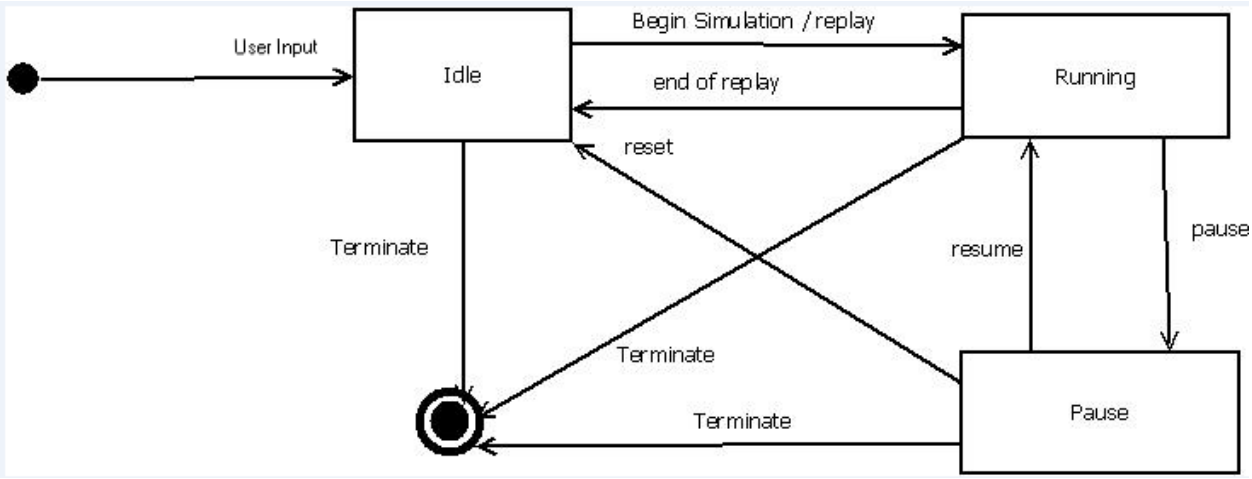
4.1 Sequence Diagrams



4.2 Data Flow Diagrams (DFD)



4.3 State-Transition Diagrams (STD)



5. Change Management Process

Customer or project liaison shall submit request for any change in requirement or its scope but, the change request for any requirement shall go through the process of requirement negotiations. Only after approval from all stakes holders of the project shall the change be applied to SRS

document. Detailed information of the impact on other existing requirement shall be mentioned in revision section of SRS for traceability purpose. Every new version of SRS document shall be shared across to all the stake holders.

A. Appendices

None