# Software Requirements Specification

**for**

## ProfitsRUS Stock Market System
## Version 2

**Prepared by GROUP: D**

**Submitted to Dr. Juerge Rilling**


| | |
|---|---|
| **Tapsvini Shingala** | **40000729** |
| **Kaichen Zhang** | **40000160** |
| **Deepakkumar Subramaniam** | **27894856** |
| **Manjinder Kaur** | **40038081** |
| **Hanbing Zuo** | **40000345** |
| **Hassan Khalid** | **27885490** |


**April 15, 2017**

# Contents

# Revision History

**Version 1:**

| Student Name | Usecase | Sections | Date |
|---|---|---|---|
| Deepakkumar Subramaniam | Read CSV | 3.1, 3.4, Section 1, 3.2, 3.8 | March 3, 2017 |
| Tapsvini Shingala | Display Chart | 3.3, Section 2 , 3.9 | March 4, 2017 |
| Hassan Khalid | Select Date range | 3.4, Remaining Section 3 | March 4, 2017 |
| Kaichen Zhang | Display MA | 3.5, Section 4 | March 3, 2017 |
| Hanbing Zuo | Select MA option | 3.6, Section 5 | March 3, 2017 |
| Manjinder Kaur | Select option buy/sell | 3.7, Remaining Section 4 and Section 5. | March 4, 2017 |

**Version 2: (Only main features are mentioned)**

| Student Name | Usecase | Sections | Date |
|---|---|---|---|
| Deepakkumar Subramaniam | Select stock | 3.1, 3.4, Section 1, 3.2, 3.8 | April 13, 2017 |
| Tapsvini Shingala | Create Panels | 3.3, Section 2 , 3.9 | April 12, 2017 |
| Hassan Khalid | API verification | 3.4, Remaining Section 3 | April 11, 2017 |
| Kaichen Zhang | Watchlist Management | 3.5, Section 4 | April 13, 2017 |
| Hanbing Zuo | Add Stock | 3.6, Section 5 | April 13, 2017 |
| Manjinder Kaur | Create MA panel | 3.7, Remaining Section 4 and Section 5. | April 14, 2017 |

# 1. Introduction

ProfitsRUS Stock Market System is to provide ProfitsRUS with a new technical analysis software for the stock market to guide user's during the decision making whether to buy or sell a stock.[1] Our project group will be focused on the implementation of this system using Java programming language. This document follows Dr. Juergen Rilling [2] slides to specify the system requirements and describe the system design. Based on the course of the Engineering and Computer Science Department at Concordia University. This document follows the SRS template provided by Professor Dr. Juergen Rilling on course website.[3] and will follow the object oriented design methodology.[4]

## 1.1 Purpose

This System is identified as "ProfitsRUS Stock Market System" Version 1.0. This SRS will describe the part of system that provides reading of CSV file, selecting date range, displaying

chart, selecting MAs to be displayed, displaying moving averages and option of displaying buy/sell. The system and documentation are to be designed in terms of usability and user friendliness.

Software Requirement Specification Version 2 for ProfitsRUS Stock Market System has describes other extended features like selecting stock, creating panels, managing watchlist and API verification.

## 1.2 Document Conventions

This document uses the IEEE standard Times New Roman size 11 fonts. Bold/Underline is used to convey important terms. Every requirement statement will have its own priority.

## 1.3 Intended Audience and Reading Suggestions

This document is intended for developers, project manager, marketing staff, users, testers and document writers. The whole document is divided into 6 sections namely: 1) Introduction part contains description about the system and this document, 2) Overall description of the system and its implementation details like the main features, user classes and characteristics, operating environment, design and implementation constrains, assumptions and dependencies and budget. 3) System features: This section has all detailed features of the system with use cases, fully dressed scenario, System sequence diagram, sequence diagram and collaboration diagram. 4) External Interface requirements: This section has interface requirement which includes UI, Hardware interface, software interfaces and communication interfaces. Fifth section has other non-functional requirements in it. 5) Other non-functional requirements like performance, safety, security and software quality attributes. 6) Other requirements. Which is followed by Appendix A and Appendix B.

Above mentioned sections will be read by different stakeholders of the system as per they needs.

## 1.4 Project Scope

This system is a technical analysis software for the stock market, which can assist users for making decision whether to buy or sell the stock. The company is already using some analysis techniques for making clear recommendations for buy/sell and this system is based on that only. This system uses Moving average analysis and it can perform just one stock at a time. This SRS is concerned about reading of CSV file, selecting date range, displaying chart, selecting MAs to be displayed, displaying moving averages and option of displaying buy/sell.

Software Requirement Specification version 2 has additional features like selecting stock, creating panels, managing watchlist and API verification.

PSMS program: implemented in Java
PSMS tool document: This document will take the reader through the software life cycle: Requirements analysis + specification, Design, Implementation, Testing and maintenance.

## 1.5 References

[1]. "Case Study 1" Concordia Course Web Sites.Web. 03 Mar. 2017.
[2]. "Lecture Notes" Concordia Course Web Sites.Web. 04 Mar. 2017.
[3]. “Specification Sample Document.pdf” Concordia Course Web Sites.Web. 05 Mar. 2017.

[4]. “Object-oriented design" *Wikipedia*. Wikimedia Foundation, 21 Jan. 2017. Web. 02 Mar. 2017.
[5]. “COCOMO” *Wikipedia*. Wikimedia Foundation, 25 Feb. 2017. Web. 03 Mar. 2017.

# 2. Overall Description

## 2.1 Product Perspective

The PSMS is being developed for **ProfitsRUS** Company which provides financial service software to end-user. The organization supports both online as well as stand-alone desktop applications. ProfitsRUS is a market leader and aims to expand their products with a new stock market analysis software. [1] It contains following features:

PSMS version 2 have some new features which are mentioned below.

## 2.2 Product features

This system contains following features:

1. Select date range:
   - Allows user to enter the start date and end date of desired duration.

2. Display chart:
   - Allows system to display chart.

3. Read CSV file:
   - Allows system to read the CSV data file.

4. Select MA option:
   - Allows user to select the option for MA.

5. Display MA:
   - Allows system to display MA on the chart.

6. Select option buy/sell:
   - Allows user to decide whether to buy or sell the shares.

New Features:

1. Select Stock:
   - Allows user to select stock based on their demand.

2. Create Panels:
   - Allows user to create the panels based on their requirement.

3. API verification:
   - External database will verify the API.

4. Mange watchlist:

- Allows user to add stock to the watchlist and other functionality like clear and update the watchlist based on their demand.

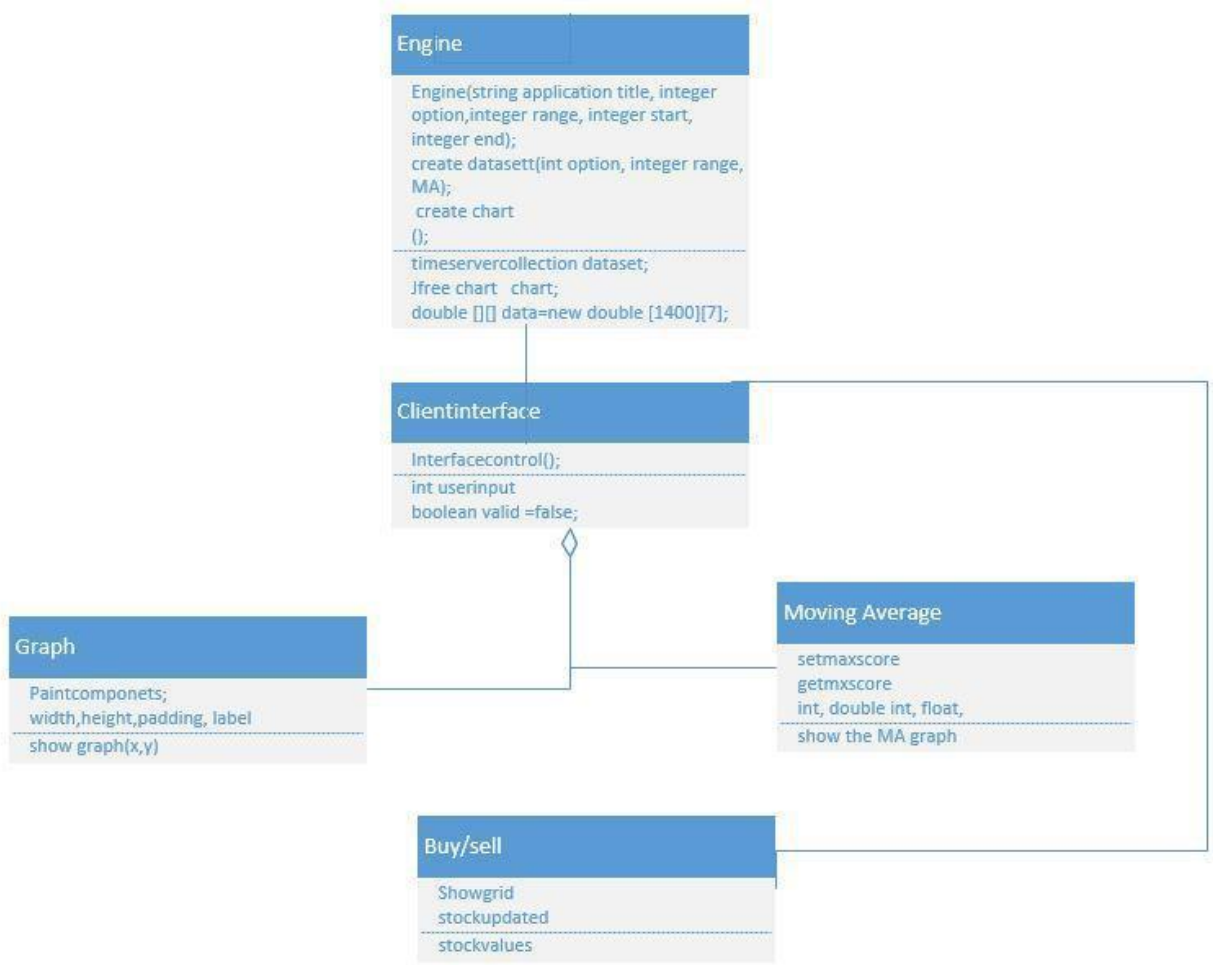
5. Add stock:

- Allows user to add stock to the watchlist.

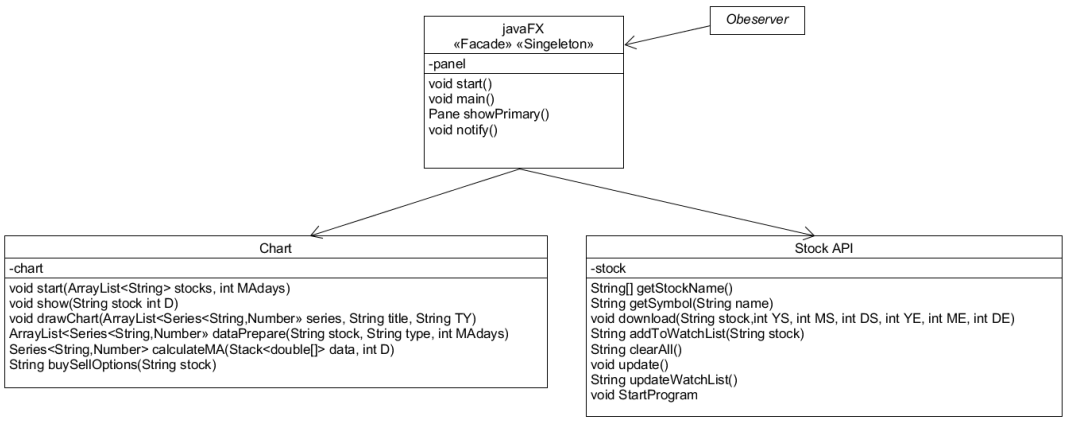
6. Create MA panel:

- Allows user to create MA panel.

## 2.2.1 Class Diagram



### 2.2.1-Version 2: Class Diagram




## 2.3 User Classes and Characteristics

There are two users who interact with the system:

1. User (End user- customer)

   Users are the customer who sell and buy shares from various company. Users do not need to have high level of education. Users can view the charts on selection of valid option.

2. System

   System is a managing entity. Whole software is dependent on system. As per user's request system will take action by using data file.

## 2.4 Operating Environment

The operating system will be Microsoft Windows 7, 8 and 10 running on the x64 and x86 hardware platform. Eclipse SDK Version: 4.6.1 will be used. Other software components include Eclipse Java Development Tools, Eclipse Plug-In development environment, etc.

## 2.5 Design and Implementation Constraints

The programming language used to implement the PSMS is Java. The user will have limited functionalities like selecting the date range, selection MAs and choosing decision option buy/sell and the system will generate charts and MAs on user's request.

Version 2 has additional functionality of selecting stock, API verification, Create panels, Managing Watchlist and Creating MA panel.

## 2.6 User Documentation

This system is easy to use. No other user manual or guide will require to use it. So no additional documents will be provided.

## 2.7 Assumptions and Dependencies

There are two main files which contains whole functionality. ClientInterface.java and Engine.java.

1) ClientInterface.java
   This file contains the implementation of the user side view, the UI while taking inputs.

2) Engine.java
   This file contains the implementation of the main business logic, calculation of MA etc.

The development requires the Microsoft Windows operating system.
There will be 2 user types: User as in customer and other System as in administrator.
At this point, it is not clear whether or not we will use software components from another project.

System version 2 has three classes:

1) Fx.java
   This class contains main logic of the system as it maintains the user requests and according to that take actions.

2) Chart.java
   Main functionality of this class is to generate the charts using the data from external database and based on the given input by the user to the system.

3) Stock.java
   This class is designed to retrieve the data from yahoo finance API based on the request submitted by the user to the system.

## 2.8 Budget

We have estimated our budget for the given six features and which is $5,500.00. The technique which we have used for the calculation of total cost is COCOMO. COCOMO Stands for Constructive Cost Model. It is a procedural software cost estimation model developed by Barry W. Boehm.[5]

As per the specifications and parameters of this model, we assumed that our project is small in size and a very little amount of innovation with no fixed deadlines. For our project, we have considered these attributes:

- Required software reliability(Low) : 0.88
- Complexity of the product (Very low) : 0.70
- Run-time performance constraints(Nominal) : 1.00
- Analyst capability(Low) : 1.19
- Applications experience (Low): 1.13
- Software engineer capability (Very low): 1.42

- Programming language experience (Low): 1.07
- Application of software engineering methods (Very low) : 1.24
- Use of software tools (Nominal): 1.00
- Required development schedule (Very low) : 1.23

All numbers are taken from reference [5].
We took adjust factor 0.565 based on the COCOMO model description in reference document and our team member's past experience. As well as, as per our domain model and other diagrams we came up with
This number.

The implemented system is very small, it required nearly 350 line of code. (0.35 KLOC). This system includes only six features as of now. Further features can be added in near future.

Project can be completed in a half month as we are six members of team. Every person need to write nearly 60 lines.
Therefore, the total estimated cost for this part of the project is
the following: Effort per person per month = $2.4 * 1.6^{1.5} * 0.565$
= 2.744 **person/months** And the total estimated cost is:
**Total Estimated Cost**: 2.744 * $2,000.00= **$5,488.00 (~$5,500.00)**
(Based on a monthly salary rate of $2,000.00)

# 3. System Features

The main objective of the PSMS is provide a better solution for stock analysis using Moving average. Which can help users to make decision about buying or selling. As of now, the system will work for given six features based on six functional requirements. Select date range, Display chart, Read csv file, select MA, Display MA and selection of buy/sell option.

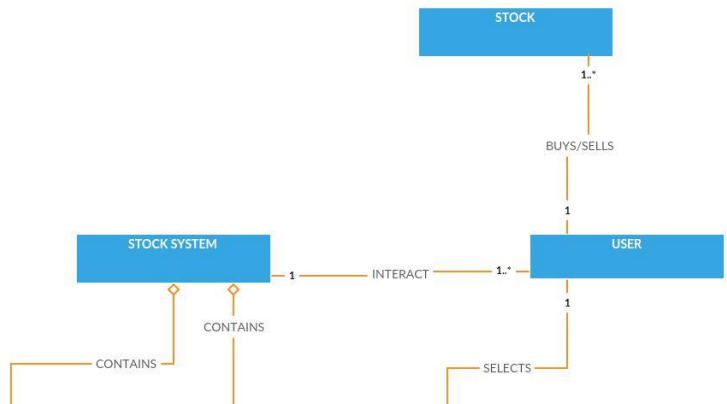
Version 2 has additional features like selecting stock, creating panels, managing watchlist and API verification.

## 3.1 Project Features

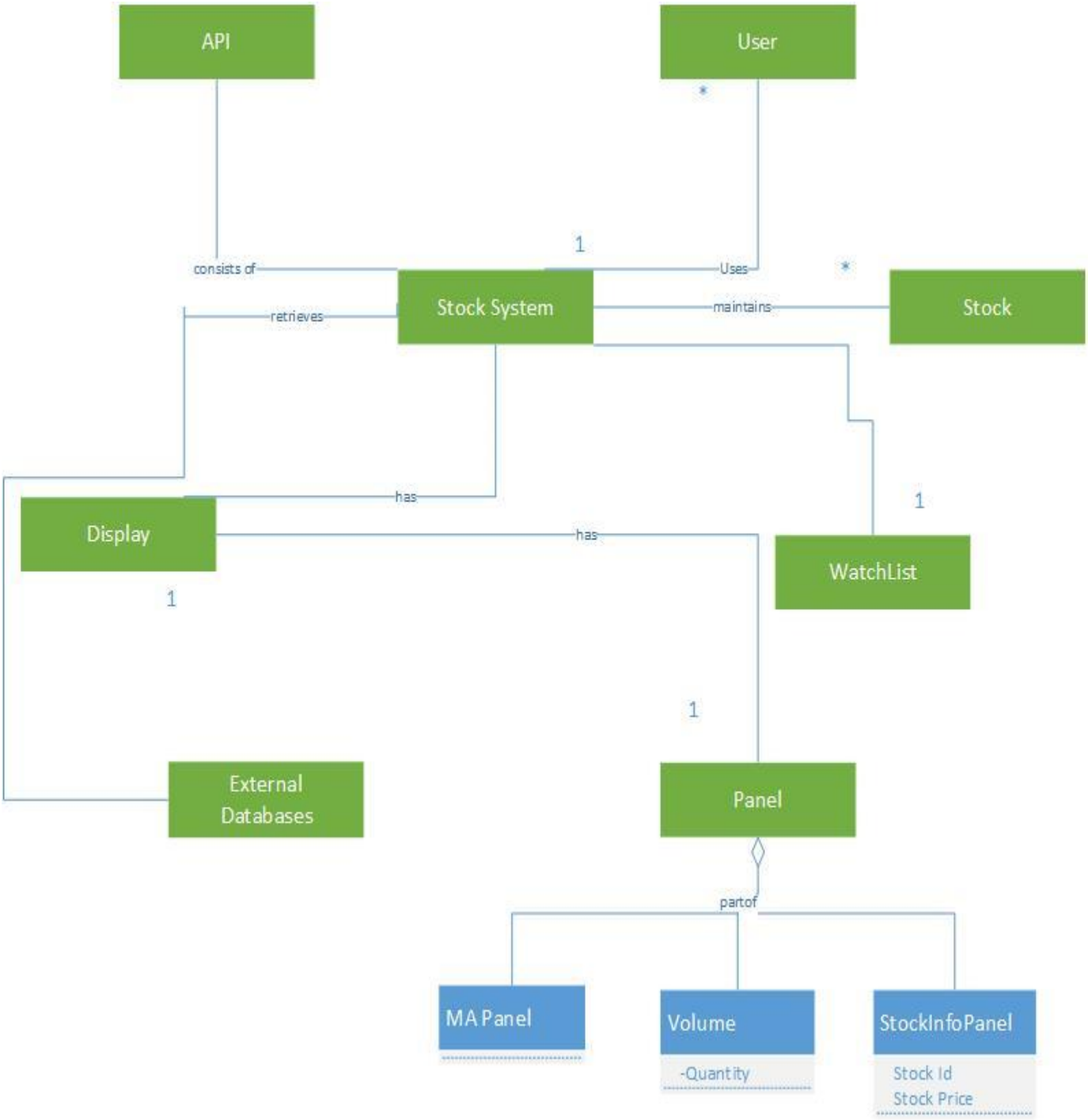
ProfitsRUS Stock Market System will help users(customers) to make decision about buying or selling shares of the company by providing effective analysis technique. The technique which is used for the drawing chart is moving average.

version 2 has additional features like selecting stock, creating panels, managing watchlist and API verification.

### 3.1.1 Project domain model



## 3.1.1-Version 2: Project domain Model



## 3.1.2 Team Use Cases

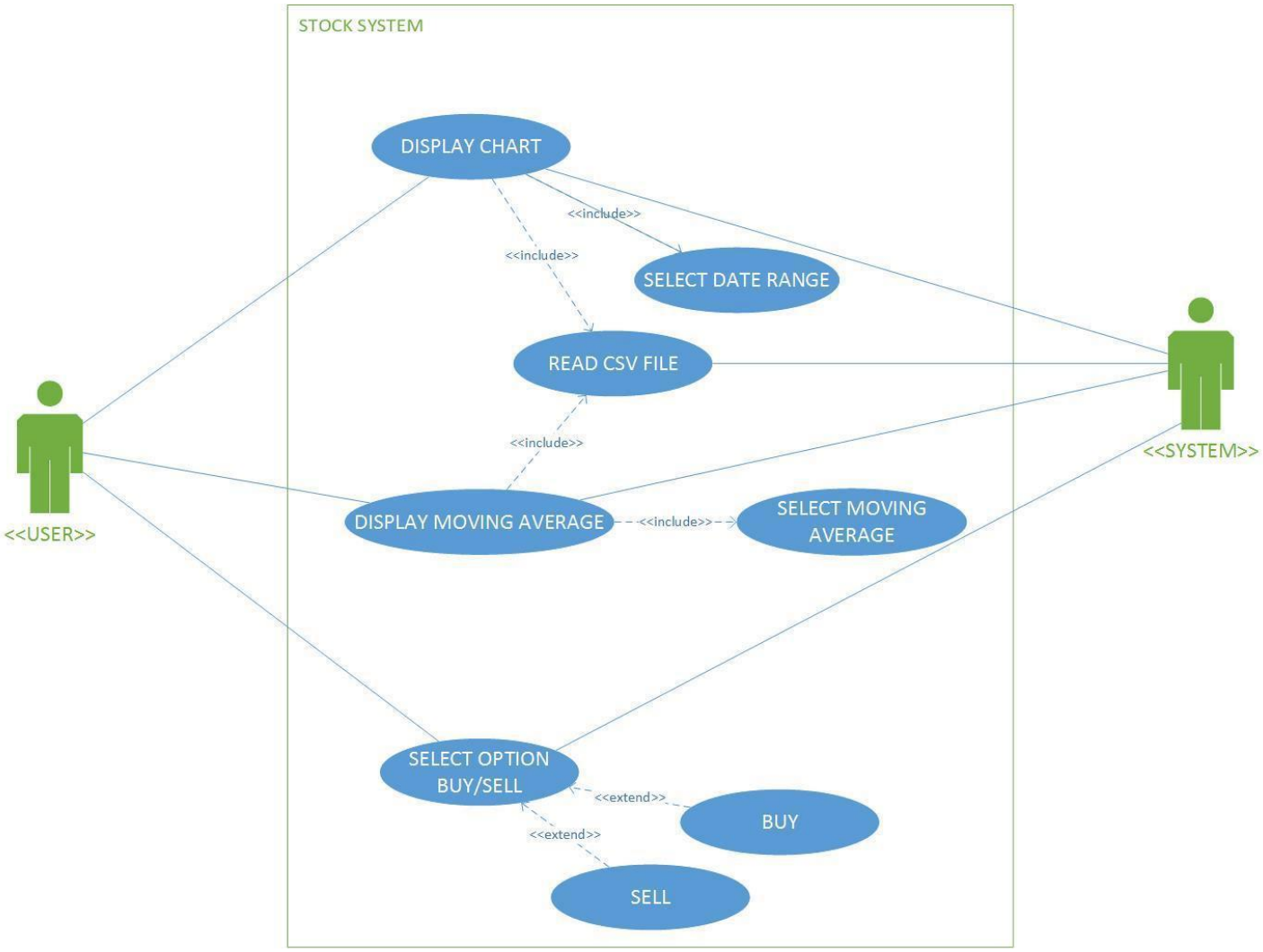
**Version 1:**

| Student Name | Usecase |
|---|---|
| Deepakkumar Subramaniam | Read CSV |
| Tapsvini Shingala | Display Chart |
| Hassan Khalid | Select Date range |
| Kaichen Zhang | Display MA |
| Hanbing Zuo | Select MA option |
| Manjinder Kaur | Select option buy/sell |
| All together | Select buy |
| All together | Select sell |

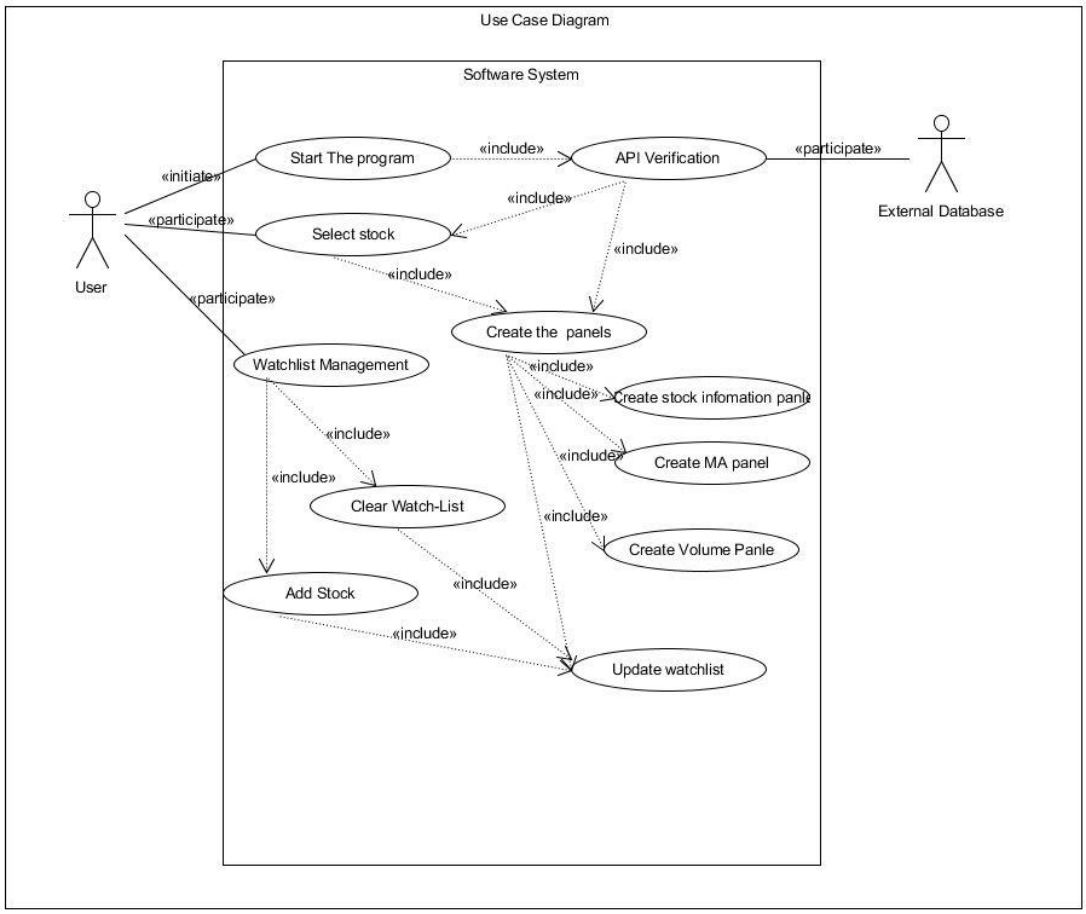
## 3.1.2-Version 2: Team Use Cases

| Student Name | Usecase |
|---|---|
| Deepakkumar Subramaniam | Select stock |
| Tapsvini Shingala | Create Panels |
| Hassan Khalid | API verification |
| Kaichen Zhang | Watchlist Management |
| Hanbing Zuo | Add Stock |
| Manjinder Kaur | Create MA panel |

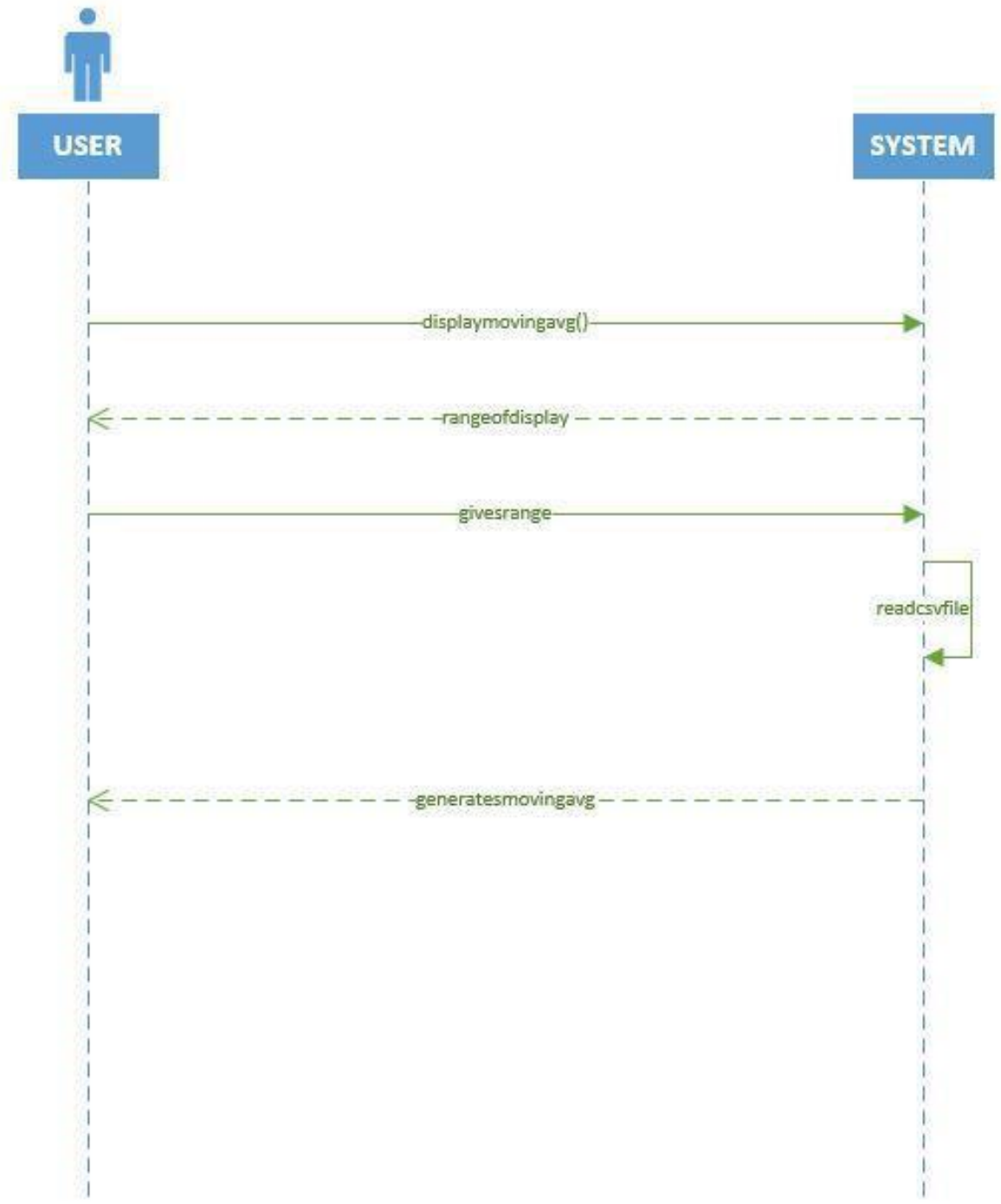
## 3.1.3 Use case Diagram



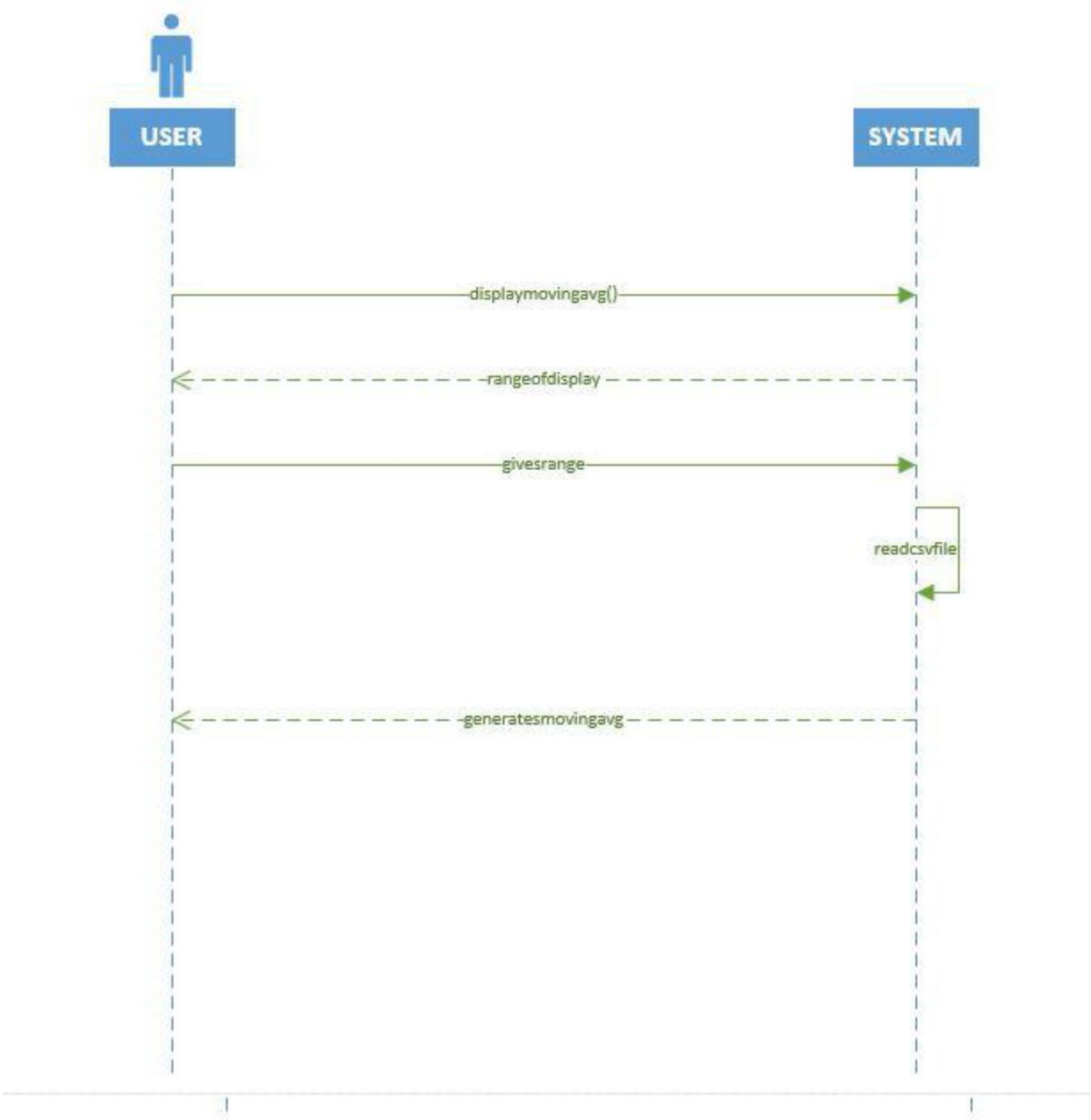
## 3.1.3 Version 2: Use Case Diagram



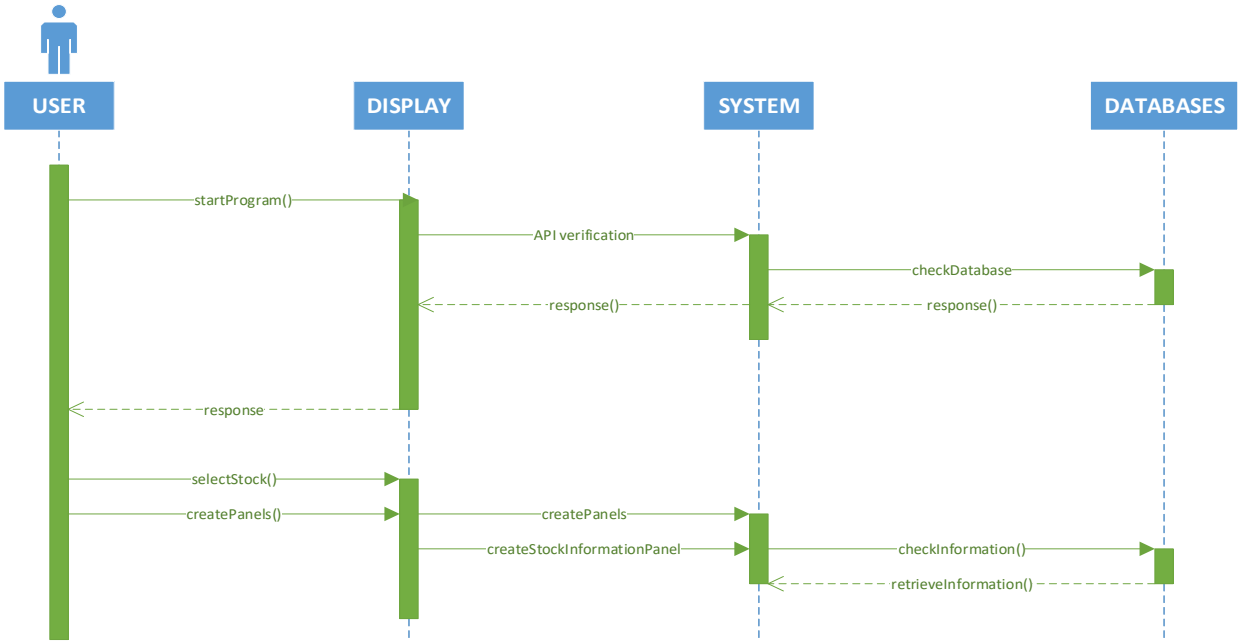
### 3.1.4 System Sequence Diagram (First)

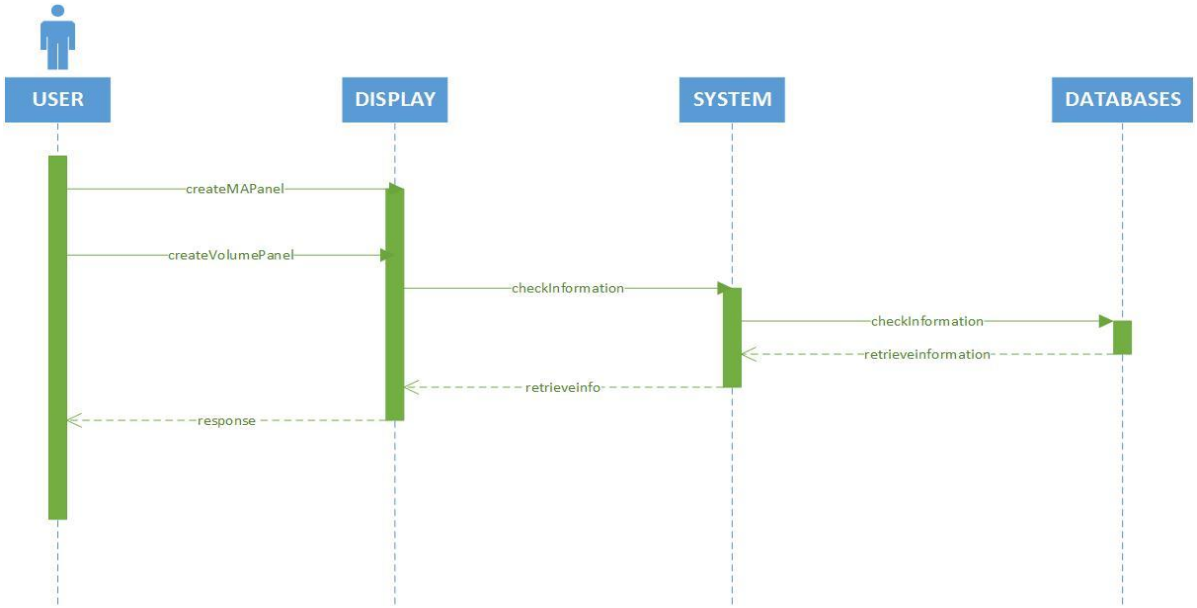


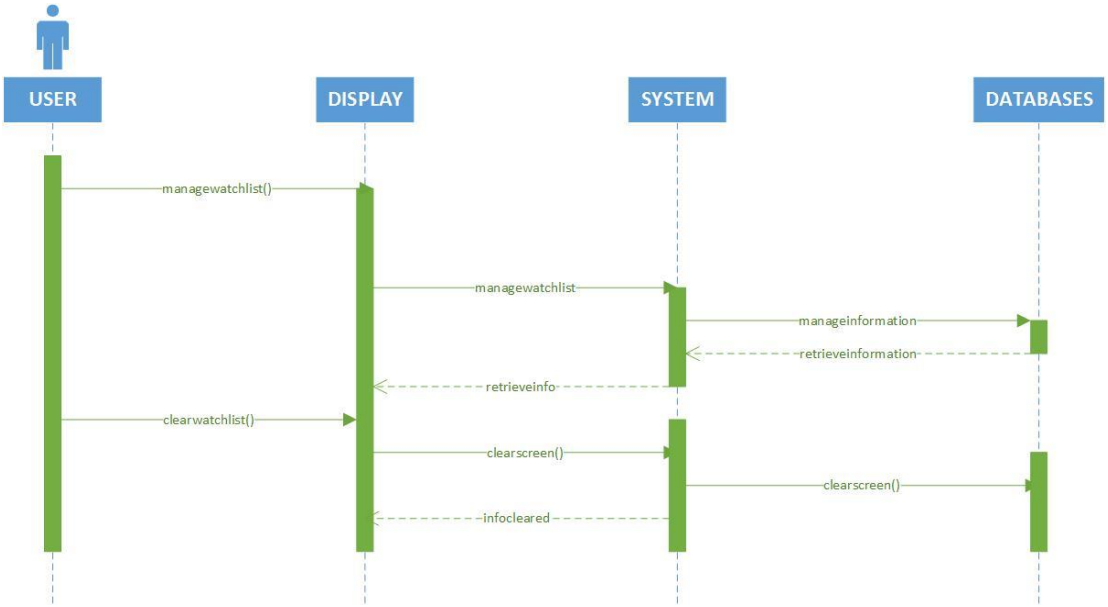
### 3.1.5 System Sequence Diagram (Second)

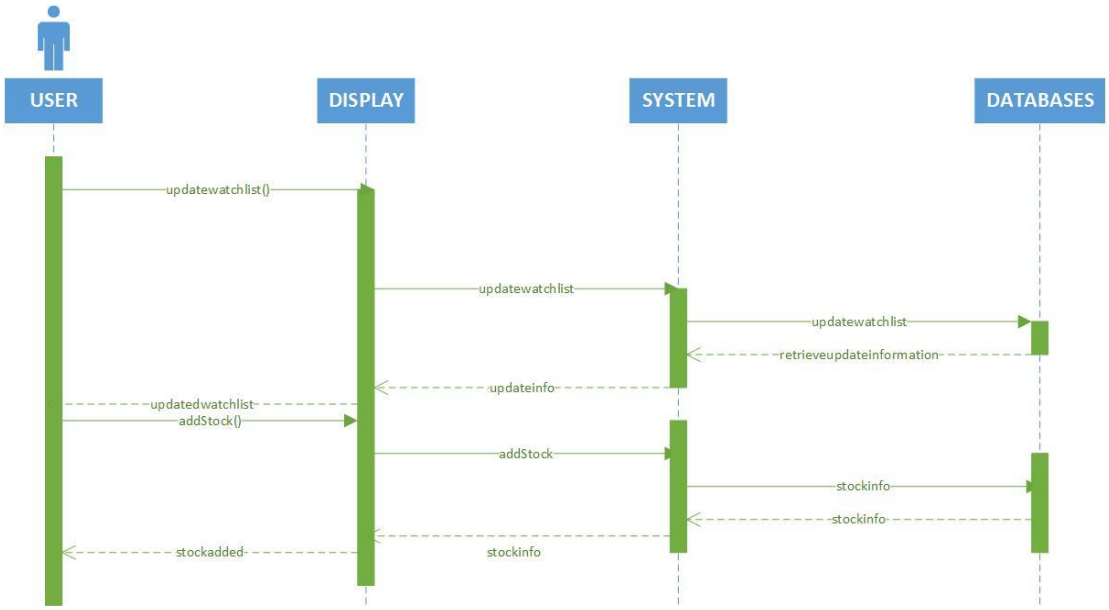


## 3.1.5 Version 2: Sequence Diagrams

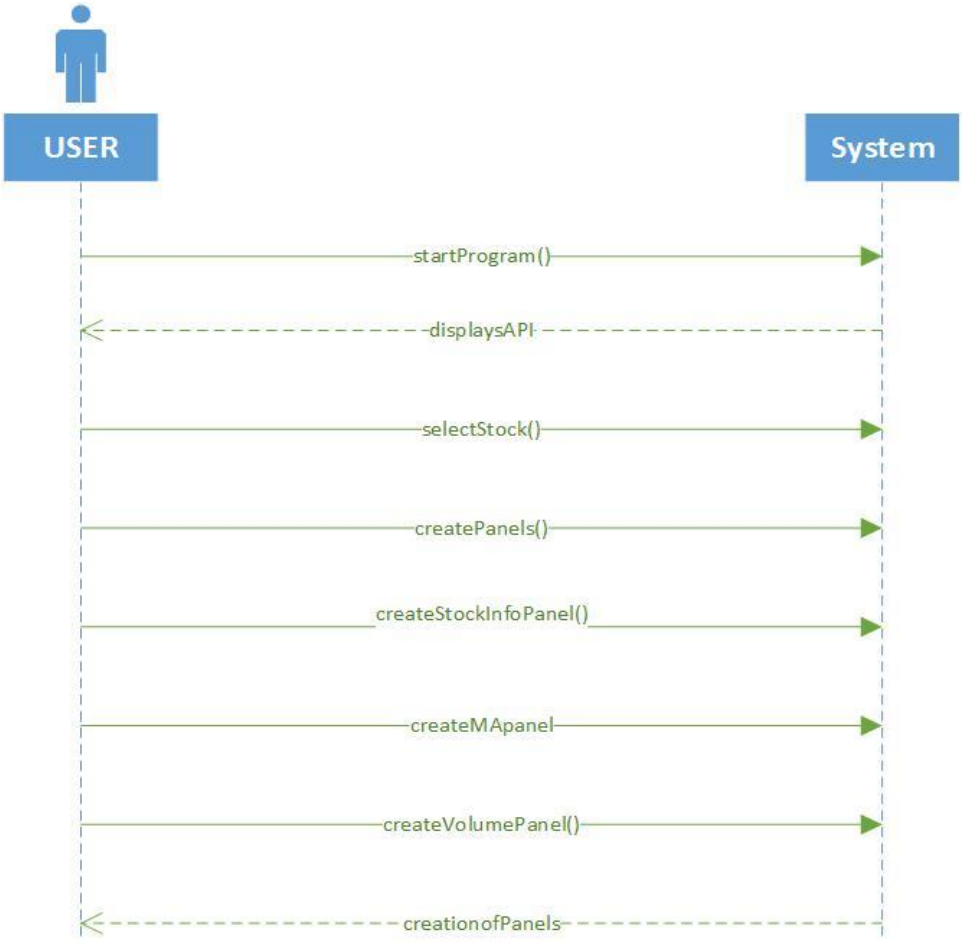


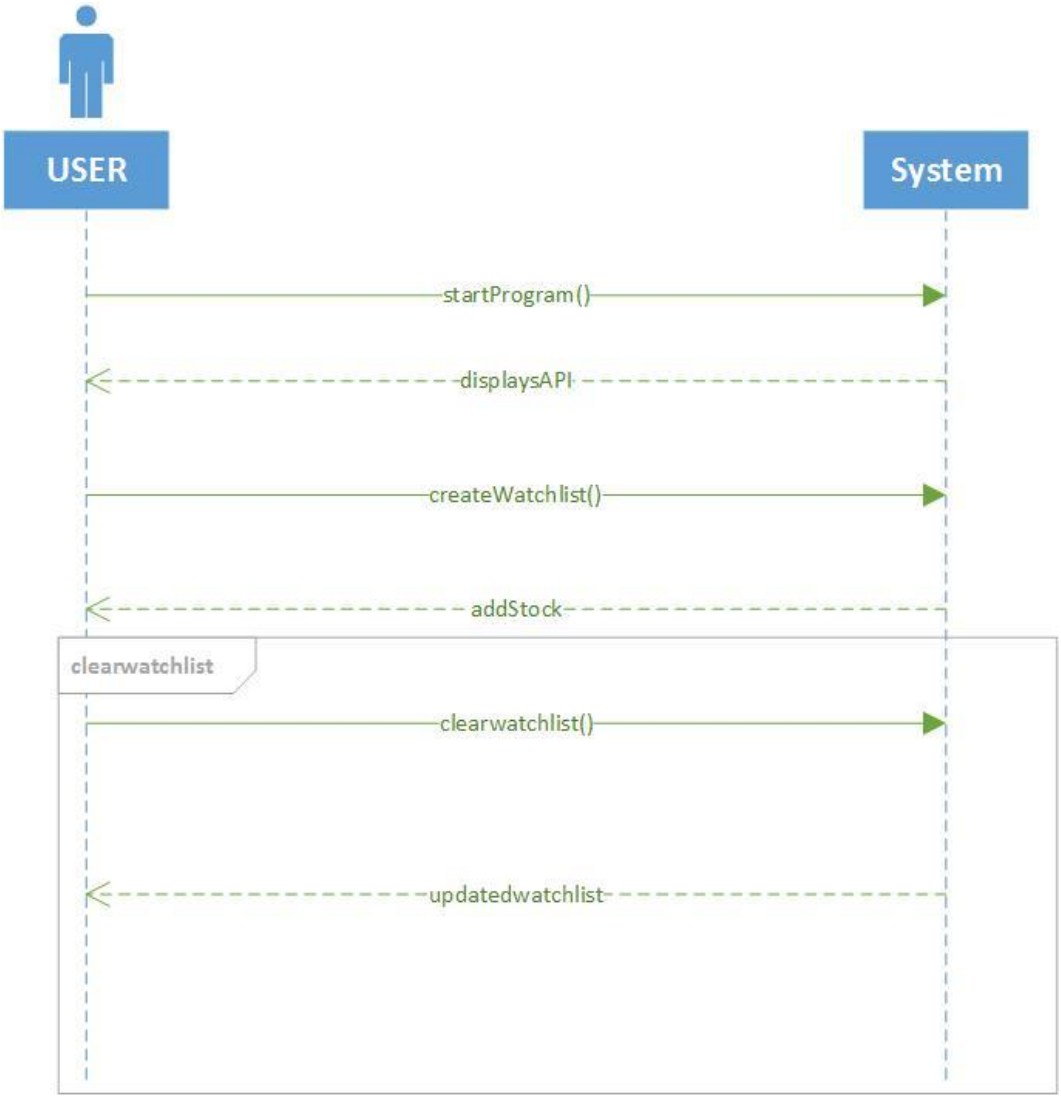




USER
DISPLAY
SYSTEM
DATABASES
managewatchlist()
managewatchlist
manageinformation
retrieveinformation
retrieveinfo
clearwatchlist()
clearscreen()
clearscreen()
infocleared

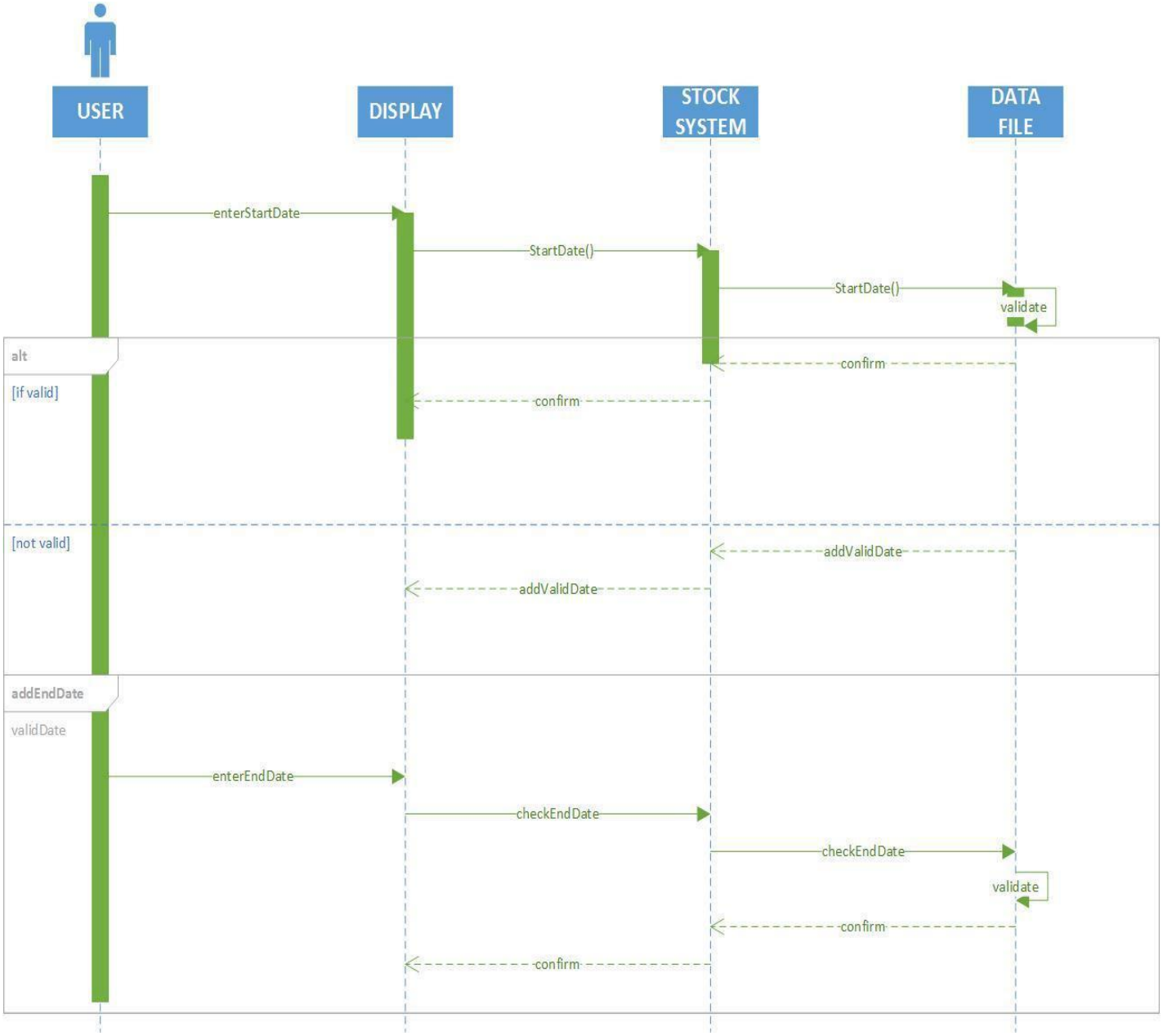

USER
DISPLAY
SYSTEM
DATABASES
updatewatchlist()
updatewatchlist
updatewatchlist
retrieveupdateinformation
updateinfo
updatedwatchlist
addStock()
addStock
stockinfo
stockinfo
stockadded
stockinfo

### 3.1.6 Version 2: System Sequence Diagrams

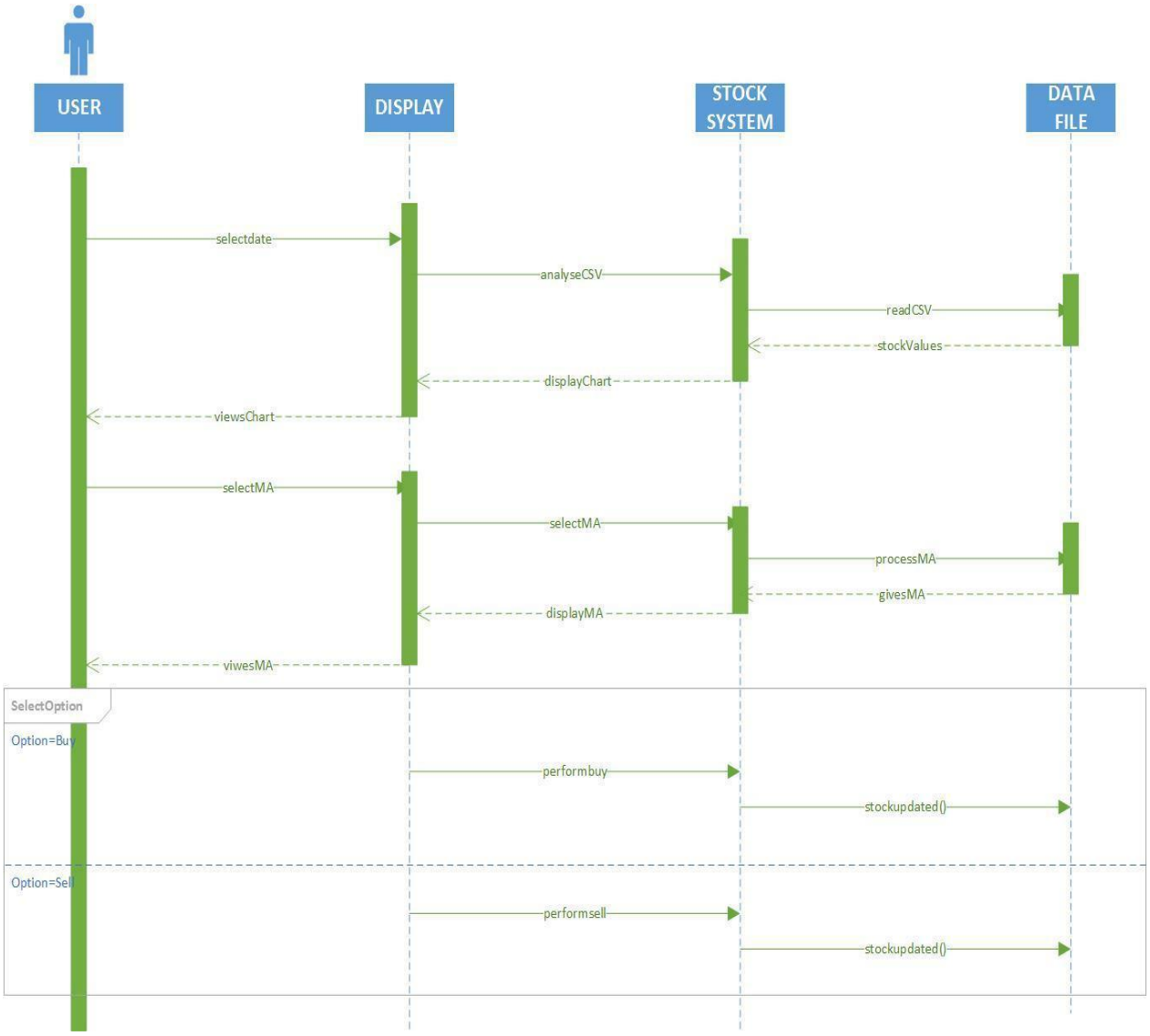



USER
System
startProgram()
displaysAPI
createWatchlist()
addStock
clearwatchlist
clearwatchlist()
updatedwatchlist

## 3.1.6 Sequence Diagram (First)



### 3.1.7 Sequence Diagram (Second)



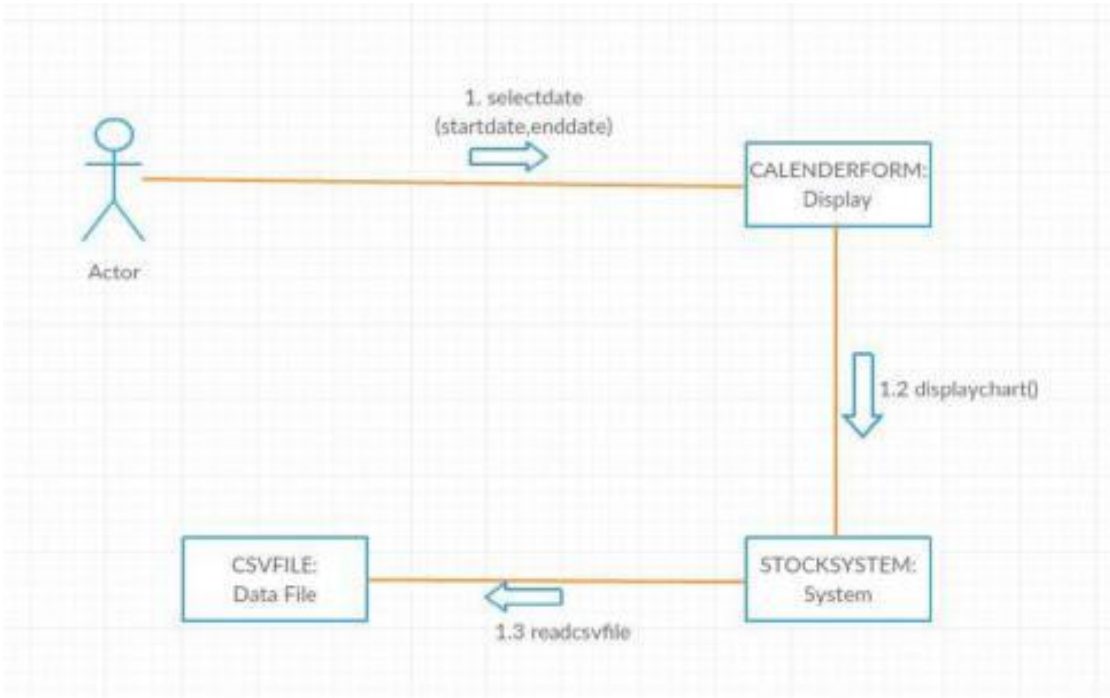
## 3.2 Use Case 1 : Read CSV File

| | |
|---|---|
| Use Case UC-1 | READ CSV FILE |
| Related Requirements | |
| Primary Actor | User |
| Actor's Goal | CSV file to be read by the system |
| Supporting Actor | System |
| Preconditions | CSV file to be Provided |
| Post Conditions | Successful reading of CSV file by the system |
| Main Success Scenario | |
| | 1. The user provides the CSV file |
| | 2. System reads the CSV file to perform the further actions |
| Extensions | |
| | 1.1 CSV file not provided<br>1.1.1 CSV file should be provided |
| | 2.1 Data provided in the CSV file is empty or some missing values.<br>2.1.1 Valid data should be provided in the CSV file |

## 3.3 Use Case 2 : Display Chart

| | |
|---|---|
| Use Case UC-2 | DISPLAY CHART |
| Related Requirements | UC-1 |
| Primary Actor | User |
| Actor's Goal | To display the chart with stock analysis |
| Supporting Actor | System |
| Preconditions | The CSV file should be read by the system |
| Post Conditions | The system displays the chart based on the data in the CSV file |
| Main Success Scenario | |
| | 1. The user provides the CSV file to the system |
| | 2. The system reads the CSV file and displays the chart based on the values in it. |
| Extensions | |
| | 1.1 CSV file not provided<br>1.1.1 CSV file should be provided |

### 3.3.1 Collaboration Diagram for "Display Chart"



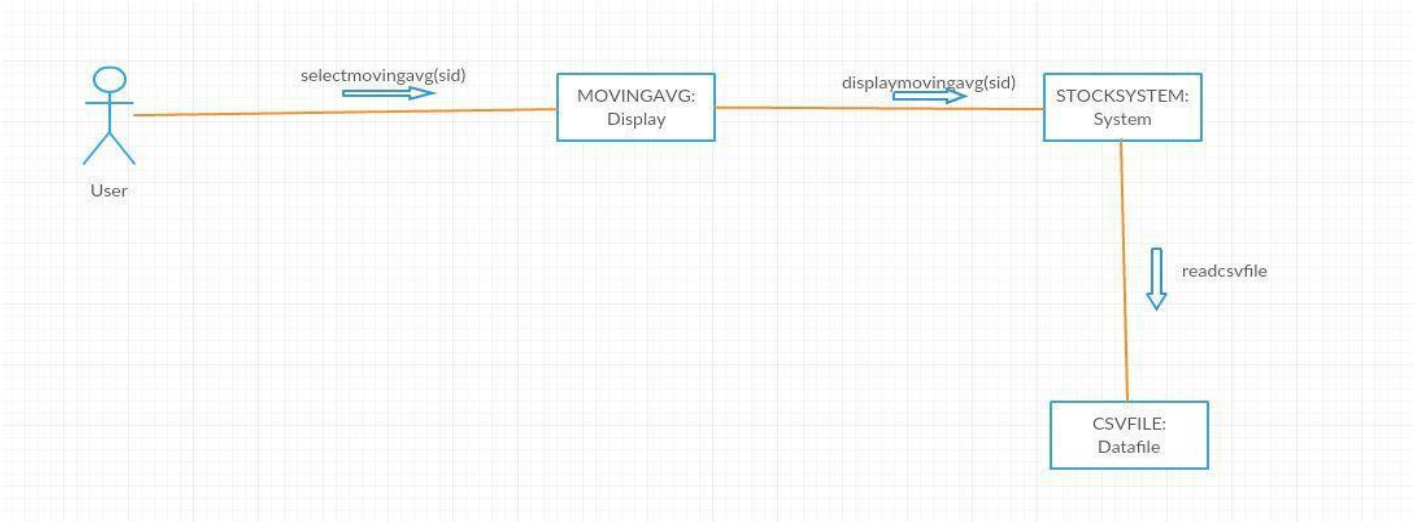

## 3.4 Use Case 3 : Select Date Range

| | |
|---|---|
| Use Case UC-3 | SELECT DATE RANGE |
| Related Requirements | UC-1, UC-2 |
| Primary Actor | User |
| Actor's Goal | To select the start and end date period for the chart to be displayed |
| Supporting Actor | System |
| Preconditions | The CSV file is there to perform the actions in the system |
| Post Conditions | Chart displays the date range which the user has selected |
| Main Success Scenario | |
| | 1. The user selects the date range of the stock value for the particular period. |
| Extensions | |
| | 1.1 If the user selects the date out of the range.<br>1.1.1 The user has to select the correct date range. |

## 3.5 Use Case 4 : Display Moving Average

| | |
|---|---|
| Use Case UC-4 | DISPLAY MOVING AVERAGE |
| Related Requirements | UC-1, UC-5 |
| Primary Actor | User |
| Actor's Goal | To get moving averages to be displayed |
| Supporting Actor | System |
| Preconditions | The user selects the moving average option |
| Post Conditions | The system displays the moving averages |
| Main Success Scenario | |
| | 1. The User selects the moving average<br>2. The System displays the moving averages |
| Extensions | |
| | None |

### 3.5.1 Collaboration Diagram for "Display Moving Average"



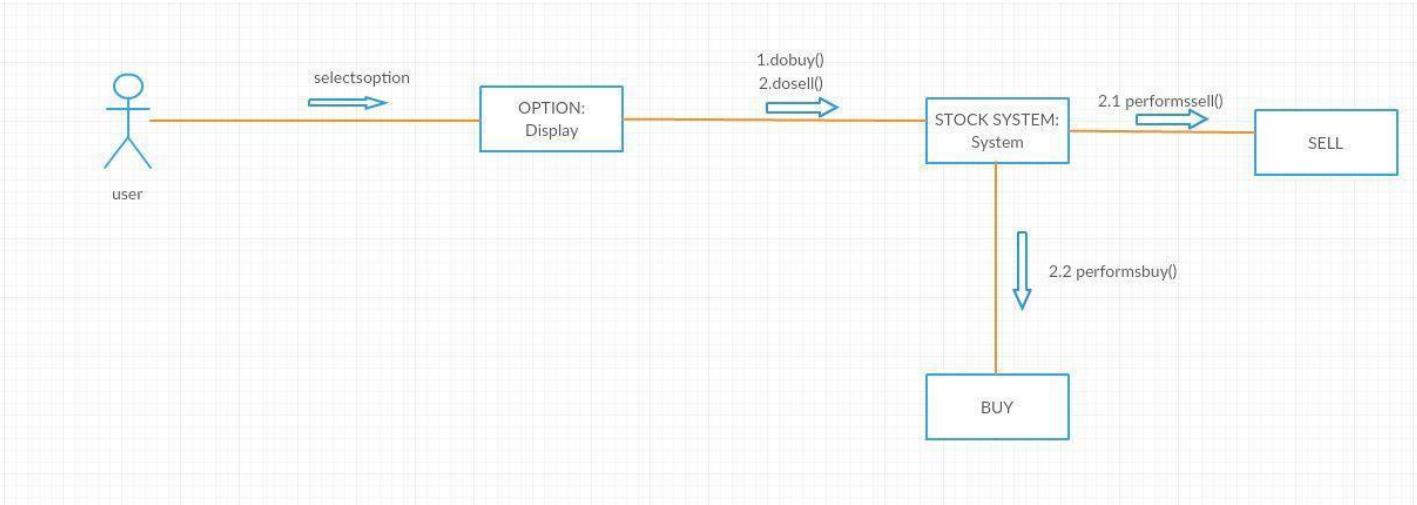
## 3.6 Use Case 5 : Select Moving Average

| | |
|---|---|
| Use Case UC-5 | SELECT MOVING AVERAGE |
| Related Requirements | UC-1, UC-4 |
| Primary Actor | User |
| Actor's Goal | To select the moving averages |
| Supporting Actor | System |
| Preconditions | The system should read the CSV file first |
| Post Conditions | The user should select moving average in order to display the moving averages |
| Main Success Scenario | |
| | 1. The User selects the moving averages option |
| Extensions | |
| | None |

## 3.7 Use Case 6: Select Option Buy/Sell

| | |
|---|---|
| Use Case UC-6 | SELECT OPTION BUY/SELL |
| Related Requirements | UC-7, UC-8 |
| Primary Actor | User |
| Actor's Goal | To select the one option which is buy/sell |
| Supporting Actor | System |
| Preconditions | The user analysis the stock value |
| Post Conditions | Based on the his/her requirements the user selects the option |
| Main Success Scenario | |
| | 1. The user performs the stock analysis |
| | 2. |
| Extensions | None |

### 3.7.1 Collaboration Diagram for "Select Buy/ Sell"



## 3.8 Use Case 7 : Select Buy

| | |
|---|---|
| Use Case UC-7 | SELECT BUY |
| Related Requirements | UC-4, UC-6 |
| Primary Actor | User |
| Actor's Goal | To buy the stock |
| Supporting Actor | System |
| Preconditions | The user has done with the stock analysis |
| Post Conditions | The user prefers to buy the stock |
| Main Success Scenario | 1. The user selects the buy option |
| Extensions | 1.1 The user unsure about the stock<br>1.2 Also prefers the sell option for his own stocks |

## 3.9 Use Case 8 : Select Sell

| | |
|---|---|
| Use Case UC-8 | SELECT SELL |
| Related Requirements | UC-4, UC-6 |
| Primary Actor | User |
| Actor's Goal | To sell the stock |
| Supporting Actor | System |
| Preconditions | The user has done with the stock analysis |
| Post Conditions | The user prefers to sell his stocks |
| Main Success Scenario | |
| | 1. The user selects the sell option |
| Extensions | |
| | 1.1 The user unsure about the stock |
| | 1.2 Also prefers the buy option for his own stocks |

## 3.10 Use case 9: Select stock

| | |
|---|---|
| Use Case 9 | SELECT Stock |
| Related Requirements | NA |
| Primary Actor | User |
| Actor's Goal | To view the stock details |
| Supporting Actor | System, Yahoo |
| Preconditions | The user has valid credentials to enter it the system |
| Post Conditions | The user may access the stock details |
| Main Success Scenario | |
| | 1. The user chooses the API for verification<br>2. The user selects the stock that he shows interest<br>3. The user adds the stocks to the watch list. |
| Extensions | |
| | 2.1 The user selects the wrong stocks<br>2.2 Also prefers the own stocks |

## 3.12 Use case 10: Create Panels

| | |
|---|---|
| Use Case 10 | Create Panel |
| Related Requirements | NA |
| Primary Actor | User |
| Actor's Goal | To create different panels like stock information panel, MA panel, Volume panel. |
| Supporting Actor | System |
| Preconditions | The user has selected the stock that too his interest |
| Post Conditions | The user checks the information's about the stock |
| Main Success Scenario | |
| | 1. The user selects the stock information panel<br>2. The system displays the stock information |
| Extensions | |
| | NA |

## 3.13 Use case 11 : API Verification

| | |
|---|---|
| Use Case 11 | API Verification |
| Related Requirements | NA |
| Primary Actor | User |
| Actor's Goal | To verify the API |
| Supporting Actor | System, External Databases |
| Preconditions | The user has to valid credentials to enter into the system |
| Post Conditions | API gets verified so that the user may selects the stocks |
| Main Success Scenario | |
| | 1. The user enters the valid his/her credentials<br>2. Verifies the API |
| Extensions | |
| | 1.1 The user entered wrong credentials<br>1.1.1 Unauthorized users<br>2.2 API verification fails. |

## 3.14 Use case 12: Manage Watchlist

| | |
|---|---|
| Use Case 12 | Manage Watch list |
| Related Requirements | NA |
| Primary Actor | User |
| Actor's Goal | To manage the watch list |
| Supporting Actor | System, External Databases |
| Preconditions | The user has to valid credentials to enter into the system |
| Post Conditions | The user manages the watch list by add stock and clearing the watch list |
| Main Success Scenario | |
| | 1. The user shows the interest in some stocks<br>2. Adds it to the watch list<br>3. After adding one stock the user may add another to watch |
| Extensions | |
| | 1.1 If the user not interest in stock that is previously added he clears it from the watch list |

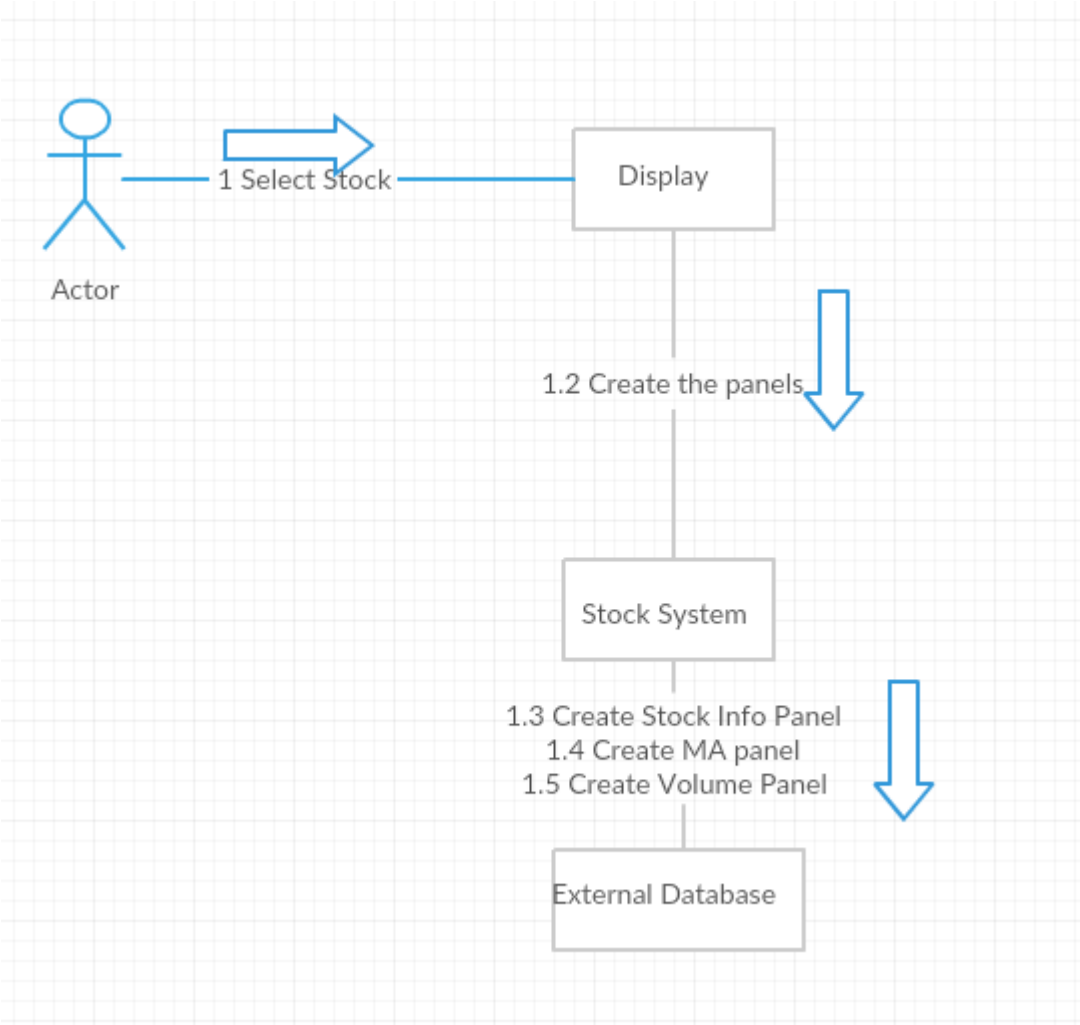
## 3.15 Use case 13: Add stock

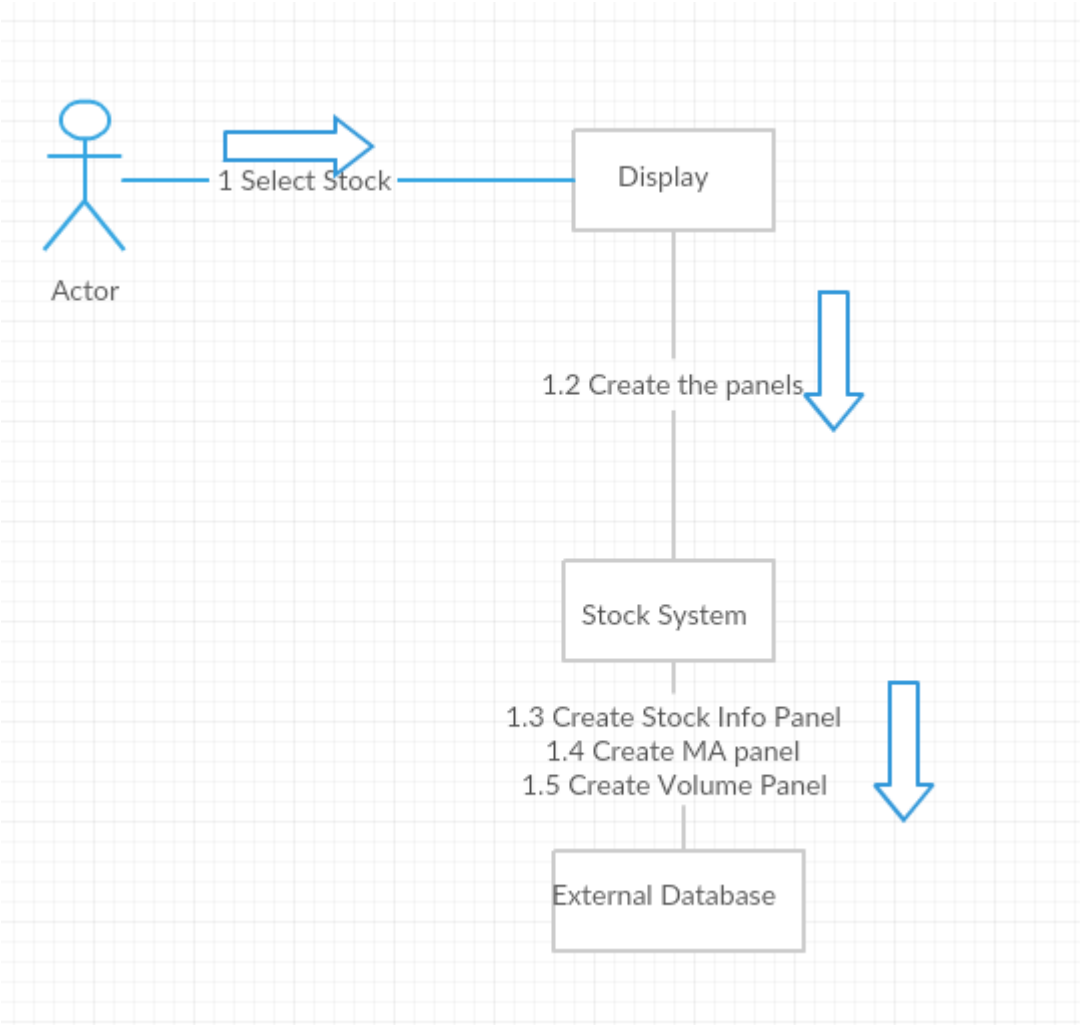
| | |
|---|---|
| Use Case 13 | Add stock |
| Related Requirements | NA |
| Primary Actor | User |
| Actor's Goal | To add the stock od particular company. |
| Supporting Actor | System, External Databases |
| Preconditions | Data should be available in the external database. |
| Post Conditions | System will allow to add the stock |
| Main Success Scenario | |
| | 1. The user can search for the stock<br>2. The user adds the interested stock |
| Extensions | NA |

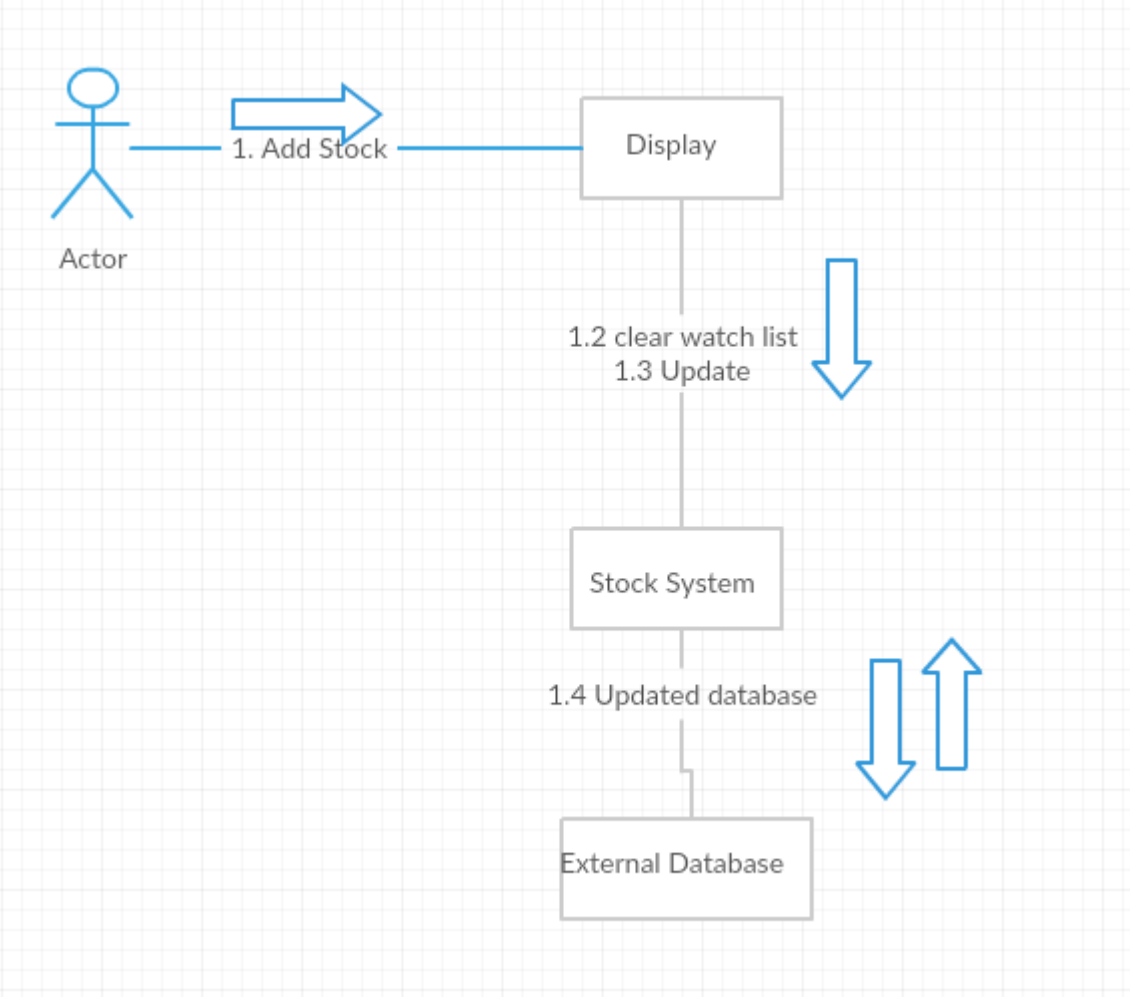
## 3.16 Use case 14: Create MA Panel

| | |
|---|---|
| Use Case 14 | Create MA Panel |
| Related Requirements | NA |
| Primary Actor | User |
| Actor's Goal | To create MA panel of the selected stocks. |
| Supporting Actor | System, External Databases |
| Preconditions | Data should be available in the external database. |
| Post Conditions | System will allow to choose the stock |
| Main Success Scenario | |
| | 1. The user can search for the stock<br>2. The user adds the interested stock<br>3. The use can see the created MA Panel by clicking on button. |
| Extensions | NA |

## 3.16.1 Version 2: Collaboration Diagrams




Actor
1 Select Stock
Display
1.2 Create the panels
Stock System
1.3 Create Stock Info Panel
1.4 Create MA panel
1.5 Create Volume Panel
External Database




# 4. External Interface Requirements

## 4.1 User Interfaces

The primary goal of the group D is to create the simple and reliable chart that displays the stock information through the jfree console. The console displays the chart with interactive feature to access the moving average by selecting the date range by the user. The interactive feature enables the user to choose the option to perform the buy or sell action.

Main Standard Buttons:

1. Select File Path- which is used to load the CSV file to the system, contains the stock value with the features like open, high, low, adj.average which are related to the particular stock at the specific point of time.
2. Choose the Date to Display- It is to choose the date in order to show the moving averages.
3. Calculate Moving Average- It is used to show the moving averages for the particular start date and end date.
4. Start Date- It is used to show the moving average from the beginning of the particular range.
5. End Date- It is used to show the moving average to end of the particular range.
6. Show- It is to display the Moving average.

7. Buy- It is used to buy the particular stock.
8. Sell- It is used to sell the particular stock.

## 4.2 Hardware Interfaces

The ProfitRUS Stock Software is developed using java and It uses abstract computing machine called Java Virtual Machine (JVM) to run the application. Specifically jFree is used. It requires the hardware requirement of 100Mb of storage. It is developed under 2 gb ram.

## 4.3 Software Interfaces

The ProfitRUS stock software system is developed using the JFREE chart 1.0.19. It is a java open source chart library it supports the different types of charts like pie chart, bar chart, Gantt chart, etc.

Jfree generates the chart within the console, so it creates both the interactive and as well as the non-interactive charts.

## 4.4 Communications Interfaces

It uses the java serial communication. If the mouse listener so that the options selected by the user gets the some actions to be done.

# 5. Other Nonfunctional Requirements

## 5.1 Performance Requirements

The ProfitRUS is the standalone application, the response time is quick it fetches the data from the csv file and based the data the system process the moving averages and displays the chart. The reliable buy and sell options are performed in the stock system. The scalability is fair in the system if the multi user performs the system synchronous updates the stock

## 5.2 Safety Requirements

No safety measures are required in this system.

## 5.3 Security Requirements

The security requirements needed in the system is that only authorized users can access the stocks. Once the user performs buy or sell the particular value is updated as well as it volume. CSV file should be inaccessible by the external users.

## 5.4 Software Quality Attributes

Reliability and ease of use will be the system. So that the user can perform the desired function without the external help. Accurate data will be generated each time so it emphasis on the accuracy.

## 6. Other Requirements

No other requirements are required in the particular system

# Appendix A: Glossary

ProfitRUS- Name of the financial concern

SRS- Software Requirements specification

CSV- Comma Separated Values

MA- Moving Averages

UI- User Interface

PSMS- ProfitRUS stock market system

SDK-software Development Kit

Cocomo- Constructive Cost Model

Jfree- open source java framework

# Appendix B: Issues List

N/A, Issues are not reported yet.