

---

# **Software Requirement Specification**

## **Stock Reader**

### **For Department of Software Engineering**

**Version 1.0 Approved**

**Prepared by  
Azmia Hoque Radhio  
ID: 171-35-1932  
Section: B  
Department of Software Engineering**

**15<sup>th</sup> December, 2020**

## Table of Contents

List of Figures .....	ii
1. Introduction.....	3
1.1 Purpose .....	3
1.2 Documentation Conventions .....	3
1.3 Product Scope .....	3
1.4 Overview .....	3
1.5 References .....	4
2. User Classes & Characteristics .....	4
2.1 User .....	4
3. Design & Implementation Constraints .....	4
3.1 Operating Environment: .....	4
3.2 Software Language Used: .....	4
3.3 Development Tools: .....	4
3.4 Database Support: .....	5
4. User Documentation.....	5
5. Assumptions & Dependencies .....	5
5.1 Stock Reader Protocol.....	5
5.2 Data Entry.....	5
5.3 Hardware Dependencies.....	5
5.4 Browser Dependencies.....	6
6. Functional Requirements.....	6
6.1 User Management: .....	6
6.2 Login & Retrieve Password: .....	6
6.3 Stock Management:.....	7
7. External Interface Requirements.....	7
7.1 User Interface.....	7
7.2 Hardware Interface.....	7
7.3 Software Interface .....	7
7.4 Communication Interface .....	8
8. Requirement Engineering Process .....	8
8.1 Feasibility Study .....	8
8.1.1 Economic Feasibility.....	8

8.1.2	Technical Feasibility .....	8
8.1.3	Behavioral Feasibility .....	8
8.2	Requirement Validation .....	9
8.3	Use Case Diagrams .....	10
8.3.1	User Use Case.....	10
8.4	System Architecture Description .....	13
8.4.1	Overview of Modules and Components.....	13
8.5	Class Diagram.....	14
8.6	Sequence Diagram .....	15
8.7	Activity Diagram .....	16
8.7.1	Overall Tasks by User.....	16
8.8	Data Flow Diagram.....	17
8.9	User Interface Design.....	17
8.9.1	Input Design .....	17
8.9.2	Output Design.....	18
8.10	Entity Relationship Diagram .....	18
9.	Non-Functional Requirements .....	19
9.1	Performance Requirements .....	19
9.2	Safety Requirements .....	19
9.3	Security Requirements .....	19
9.4	Maintainability Requirements .....	19

## List of Figures

Figure 1: Use Case – User.....
Figure 3: Basic Modules of the Stock Reader.....
Figure 4: Components of Add Symbols Modules .....
Figure 5: Components of Filter Stock Data Modules .....
Figure 6: Model and View classes for Stock Reader .....
Figure 7: Sequence Diagram for Stock Reader .....
Figure 8: Activity Diagram for Stock Reader .....
Figure 9: Context Diagram for Stock Reader .....
Figure 10: Entity Relationship Diagram for Stock Reader .....

# 1. Introduction

## 1.1 Purpose

The purpose of this document is to describe all the requirements for the targeted system- **Stock Reader**. The intended audience includes all stakeholders in the potential system. These include, but are not necessarily limited to, the following: **Shareholder/Stockholder**.

Developers should consult this document and its revisions as the only source of requirements for the project. They should not consider any requirements statements, written or verbal as valid until they appear in this document or its revision.

The Shareholder should use this document and its revisions as the primary means to communicate confirmed requirements to the development team. The development team expects many face-to-face conversations that will undoubtedly be about requirements and ideas for requirements. Please note that only the requirements that appear in this document or a future revision, however, will be used to define the scope of the system.

## 1.2 Documentation Conventions

This SRS is divided up into sections detailing an overall description, the external interface requirements, system features, and other non-functional requirements. As this is the final draft, any future modifications of this document would involve adapting the product to changing systems and uses. I hope to have the product evolve to changing times as to ensure continued use and success. The Document and Specification team have prepared the overall information in this document to the best of their ability. Once read, it is evident that each section is important to the overall SRS and significant to the project in its own right.

## 1.3 Product Scope

The proposed software product is **Stock Reader**. It will be used to maintain a Stock Scanner site. Users can add company symbols, View Stock Data and Filter on those data by different category. The current system is much vulnerable and risky.

## 1.4 Overview

This Software Requirements Specification (SRS) specifies all the requirements for **Stock Reader**. Various techniques such as interviews, brain storming and idea reduction, use cases and prototyping were used to elicit the requirements and I have identified the needs, analyzed and refined them. The objective of this document therefore is to formally describe the system's high-level requirements including functional requirements, non-functional requirements and constraints. The detail structure of this document is organized as follows:

Section 2 of this document provides an overview of the domain that the proposed software Stock Reader will support. These include a general description of the product, user characteristics, general constraints, and assumptions for this system. To contrast, in section 3 all the design and Implementation constraints are given as well the Assumptions and Dependencies are described concisely in the Section 5. Section 6 includes the system features having all the functional requirements along with their rationale. Section 7 presents the details of the external interface requirements. Finally, section 9 contains the other non-functional requirements.

## 1.5 References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications*. IEEE Computer Society, 1998.

## 2. User Classes & Characteristics

### 2.1 User

In the **Stock Reader** the Shareholders are the users. They can add company symbols, View Stock Data and Filter on those data by different category.

## 3. Design & Implementation Constraints

### 3.1 Operating Environment:

The Stock Reader will be web-based system. Thus, anyone having a browser can hit the specific link and can get access to it. Thus, it will ensure its best usage and will ease the means of getting access to the system. Moreover, it will remove the complexities of running the system in multiple platforms as it will be deployed in a web server.

### 3.2 Software Language Used:

The application will be developed using Fastapi. The used language will be Python and the front end will be developed using HTML. Besides for eye soothing user interface experience CSS 3 and Jinja2Templates will also be used.

### 3.3 Development Tools:

For the development purpose Visual Studio Code will be used. For handling different database operations Sqlalchemy ORM will be used.

### **3.4 Database Support:**

The database that will be used is Sqlite. Fastapi will be used from the applications end to insert, update and delete the data.

## **4. User Documentation**

According to the schedule, after four months of the start of the project, I will hand-over the complete Stock Reader to Software Engineering Department Authority. At the same time, I will also provide a user manual, where all the how-to will be put together. On the other hand, the project should be launched as a pilot project for about 4 months to get more feedback from the end-users and responsible to change with newbie requirements.

## **5. Assumptions & Dependencies**

### **5.1 Stock Reader Protocol**

In this system there is an option to access from the Internet along with Intranet and Ethernet. Here, the user can add company symbols, View Stock Data and Filter on those data by different category.

### **5.2 Data Entry**

Though the data entry operation is out of the scope of this project, but for giving it a standard look I have added some meaningful data to check the compatibility of the system. To include, this information has collected from the requirement elicitation process from the various Stock Scanner sites. Supply of correct information is possible only when valid data is entered in the database. Since the data entry is a separate task and will be performed by users, the authority will be responsible for the validity of the information to be provided to the user through Stock Reader.

### **5.3 Hardware Dependencies**

To operate the system the following hardware dependencies are needed:

- Runs on any x86-64 machine.
- Depending on the number of users its server, it'll need a reasonably powerful machine to perform its tasks. The actual requirements will be profiled at a later phase.
- Every user must have internet connectivity devices to use the system.

## 5.4 Browser Dependencies

The system is based on web; therefore, no custom-tailored client is required to access it. However, Blogger will be compatible with any JavaScript enabled open standard browsers, and it will also support Internet Explorer (IE), Mozilla Firefox (any latest version) and other compatible browsers.

## 6. Functional Requirements

Before identification of the requirements, I need the comprehensive engagement and lighting quick coordination with the stakeholders. This accelerates the entire requirements management process by orchestrating the flow of information and processes across stakeholders. Again, this is combined with hybrid agile and waterfall development methodologies and tools. Flexible workflows and automatic notifications streamline communication, review, and approval of requirements across stakeholders, while common metrics and dashboards ensure everyone is on the same page. So, the listed requirements go with all the previous processes.

### 6.1 User Management:

User management is the task of admin where the key roles is to give input the user ids and approves the users.

Requirement No.	Requirement
UM-001	Add user with username, email id, first name, last name and password
UM-002	Password should be given twice to match
UM-003	Password should be in complex format

### 6.2 Login & Retrieve Password:

Here, the requirements are based on the task of login system and password retrieval by the users including teachers, students and coordinators.

Requirement No.	Requirement
LP-001	While login match the username
LP-002	User will get the functionalities of his/her
LP-003	Login time should be stored in the log file

LP-004	While retrieving the password (if forgot) user should provide the username and email id
LP-005	System will mail a link to that user containing the password
LP-006	With the new password link user will give his/her favorable password
LP-007	Update the new reset password with the log file

### 6.3 Stock Management:

Here, the requirements are based on the task of login system and password retrieval by the users including stock financials/statistics.

Requirement No.	Requirement
SP-001	User will add Symbols
SP-002	User will view Stock financials/statistic
SP-003	User will Filter Stock Data by Different Category

## 7. External Interface Requirements

### 7.1 User Interface

The user interface is a key to application usability. The application should include content presentation, application navigation, and user assistance. In the Stock Reader there should be different portal for each type of users.

### 7.2 Hardware Interface

In the current version of the software, it will have no special hardware interface with other external systems. It will run in a general-purpose computer system with general-purpose hardware and software. Actually, the assumption of the hardware is already given to the upper section 5.3.

### 7.3 Software Interface

To the end-user there is no need of any extra software to be installed. It is to be mentioned that, the user need JavaScript enabled browsers to run the system. For OS, there has no boundary or strict rules, can run smoothly in any OS. However, through the channel the cryptography should be maintained through the whole system as the user can access it through internet also.



## 7.4 Communication Interface

All sorts of communications between server and client programs will be using Hyper Text Transmission Protocol (HTTP) and the messaging will be done by XML format. As a result, any user using standard communication protocols can communicate with the Blogger without any protocol conversion or any other hassles.

# 8. Requirement Engineering Process

## 8.1 Feasibility Study

In this document of Stock Reader, I am also providing some feasibility which will support the system and also give more liveness. For these I will give emphasize on the following topics.

### 8.1.1 Economic Feasibility

Economic analysis is most frequently used for evaluation of the effectiveness of the system. More commonly known as cost/benefit analysis the procedure is to determine the benefit and saving that are expected from a system and compare them with costs, decisions is made to design and implement the system.

In the system, the authority is most satisfied by economic feasibility. Because, just only with the initial server settlement, it need not require any additional hardware resources as well as it will be saving lot of time. Again, the total system is going to develop step by step by me, so other external costs can be cut off.

### 8.1.2 Technical Feasibility

Technical feasibility centers on the existing manual system of the test management process and to what extent it can support the system. According to feasibility analysis procedure the technical feasibility of the system is analyzed and the technical requirements such as software facilities, procedure, inputs are identified. It is also one of the important phases of the system development activities. The system offers greater levels of user friendliness combined with greater processing speed. Therefore, the cost of maintenance can be reduced. Since, processing speed is very high and the work is reduced in the maintenance point of view management convince that the project is operationally feasible. In addition, I am promised to give an understandable user manual which will help the admin, teachers and coordinator as well as the students to cope up with the system effortlessly.

### 8.1.3 Behavioral Feasibility

People are inherently resistant to change and computer has been known to facilitate changes. An estimate should be made of how strong the user is likely to move towards the development of computerized system. These are various levels of users in order to ensure proper authentication and authorization and security of sensitive data of SIS.

## 8.2 Requirement Validation

Through the all process of Software Requirement Elicitation I'm endure different type and level of requirements. From these huge requirements I have already thrashed out the functional requirements. But there are some validations of these requirements like in the registration process of the user in the system. However, all these are not accepted and for that I have ensured these with the stakeholders throughout the elicitation process.

As, the requirements validation is critical to successful system product development and implementation. Requirements are validated when it is certain that the subject set of requirements describes the input requirements and objectives such that the resulting system products can satisfy the requirements and objectives.

The Requirements Validation Process helps ensure that the requirements are necessary and sufficient for creating design solutions appropriate to meeting the exit criteria of the applicable engineering life cycle phase and of the enterprise-based life cycle phase in which the reengineering efforts occur.

Key activities of requirements validation are:

- Conduct requirements reviews to validate that requirements are correct, unambiguous, complete, consistent, ranked for importance, verifiable (testable), modifiable, and traceable. Review teams should include end user representatives and customer representatives, in addition to the developer participants. Use quality checklists as an aid to the review process.
- Use prototyping to validate requirements. Prototypes demonstrate assumptions and actual understandings and can alert the team to mismatches between the written requirement and the interpretation carried forward in the prototype.
- Validate the conceptual models developed during analysis.
- Plan how each requirement will be verified establish acceptance tests.

Perform the validation of any requirements document provided by the acquirer at the time of contract award I can ensure that our requirements specifications have the following characteristics:

- Lack of ambiguity
- Conciseness – Minimal number of words used and presented in a distinct visual form
- Completeness – The specification contains all requirements known to date
- Consistency – There are no conflicting requirements

Traces to origins – The source/origin of each requirement are identified. It may have evolved from a more general requirement, result from a conversation with a user, result from adoption of a standard, or adhering to a new regulation.

## 8.3 Use Case Diagrams

### 8.3.1 User Use Case

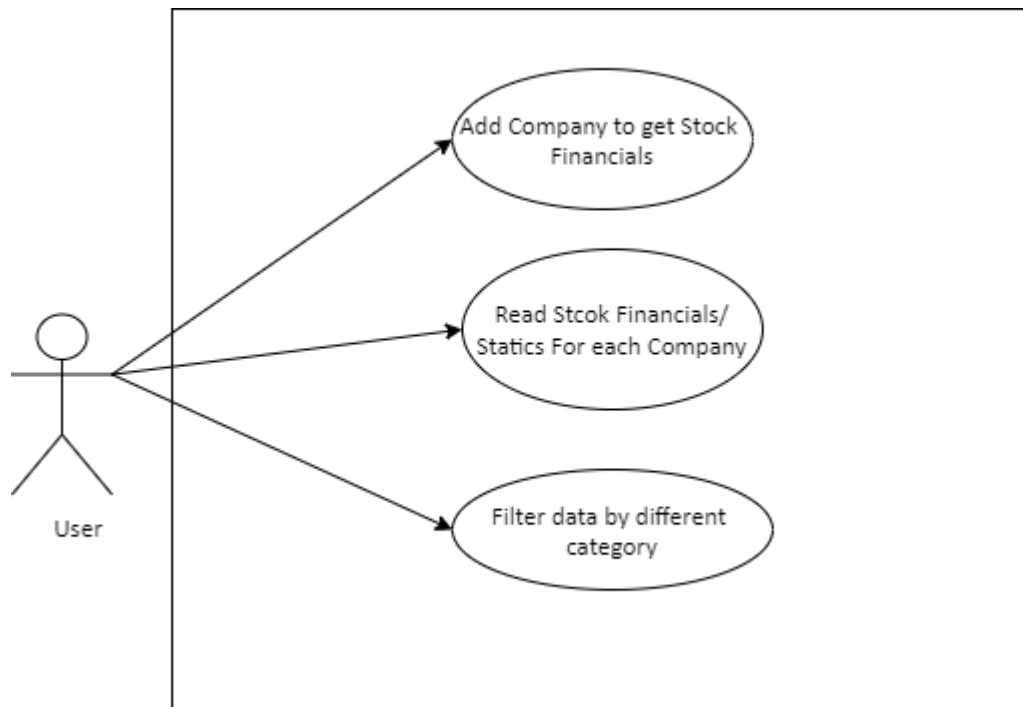


Figure 1: Use Case - User

<b>Use Case No.</b>	1.1
<b>Use Case Name</b>	Login
<b>Actor</b>	User
<b>Description</b>	Allowing users to login to the system
<b>Precondition</b>	User should remain in the login page
<b>Trigger</b>	Click the “Login” Link
<b>Flow of Events</b>	<ul style="list-style-type: none"> <li>➤ Two text fields to give input of the username and password respectively</li> <li>➤ Write the username and password on that field and click the login button</li> </ul>
<b>Post Condition</b>	User logged into the system

<b>Use Case No.</b>	1.2
<b>Use Case Name</b>	Reset Password

<b>Actor</b>	User
<b>Description</b>	Allowing the user to reset the password for security purpose
<b>Precondition</b>	Login
<b>Trigger</b>	Click the “Reset Password” Link
<b>Flow of Events</b>	□ Three text fields to give input of the old password, new password and confirming new password respectively □ Fill up those fields and click reset button
<b>Post Condition</b>	The password is reset and stored in the database

<b>Use Case No.</b>	1.3
<b>Use Case Name</b>	Retrieve Forgotten Password
<b>Actor</b>	User
<b>Description</b>	Allowing user to retrieve forgotten password
<b>Precondition</b>	User should remain in the login page
<b>Trigger</b>	Click the “Forgot Password” Link
<b>Flow of Events</b>	➤ A text field to give input of an email address ➤ Write the email address on the field
<b>Post Condition</b>	If the email address is present in the database a password reset link is sent to that address

<b>Use Case No.</b>	1.4
<b>Use Case Name</b>	Add Symbols
<b>Actor</b>	User
<b>Description</b>	Allowing user to Fetch Stock Data
<b>Precondition</b>	User should remain in the login
<b>Trigger</b>	Click the “Add Symbols” Link
<b>Flow of Events</b>	➤ A text field to write the symbols
<b>Post Condition</b>	If the topic name is present then change the name.

<b>Use Case No.</b>	1.5
<b>Use Case Name</b>	View Stock Statistic

<b>Actor</b>	User
<b>Description</b>	Allowing user to view Stock financials
<b>Precondition</b>	User should remain in the login
<b>Flow of Events</b>	None
<b>Post Condition</b>	User reading the post

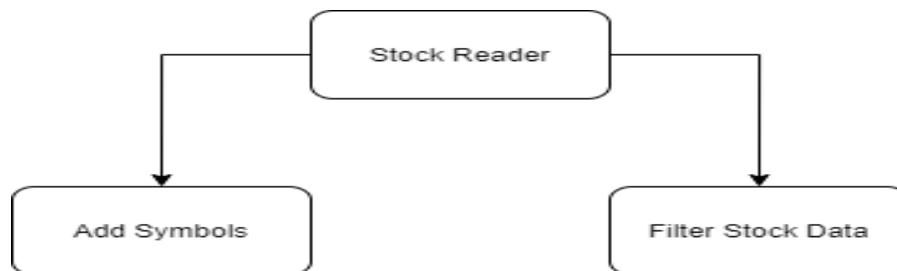
<b>Use Case No.</b>	1.6
<b>Use Case Name</b>	Filter Stock Data
<b>Actor</b>	User
<b>Description</b>	Allowing user to filter on stock
<b>Precondition</b>	User should remain in the login
<b>Trigger</b>	Click the “Filter” Link
<b>Flow of Events</b>	None
<b>Post Condition</b>	User Add a Symbol on a post

## 8.4 System Architecture Description

The core models and functionalities which are derived from the functional requirements are generated as some basic components of the systems. By this section it will be very clear for the development team to find out all these at a glance.

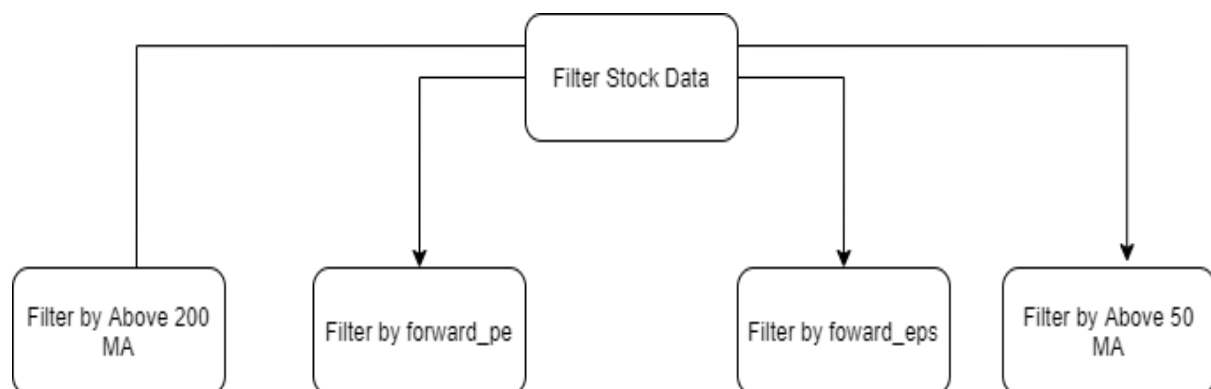
### 8.4.1 Overview of Modules and Components

The Stock Reader should have the two basic modules -



**Figure 3: Add Symbols Modules of the Stock Reader**

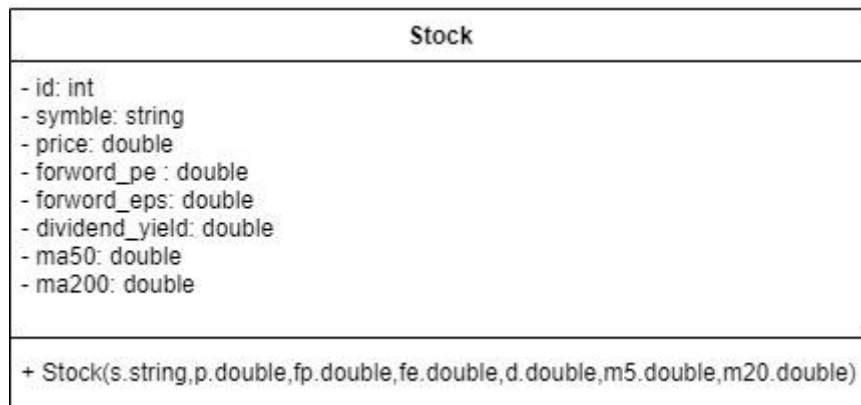
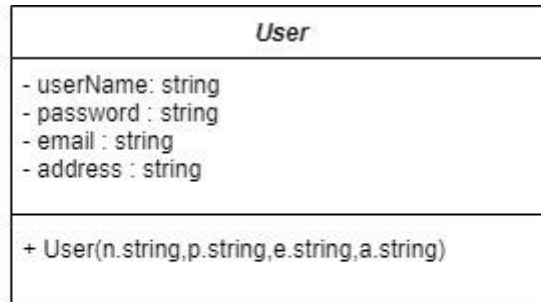
The Stock Reader will handle the addition and update of different users Its components will be -



**Figure 3: Filter Stock Data Modules of the Stock Reader**

## 8.5 Class Diagram

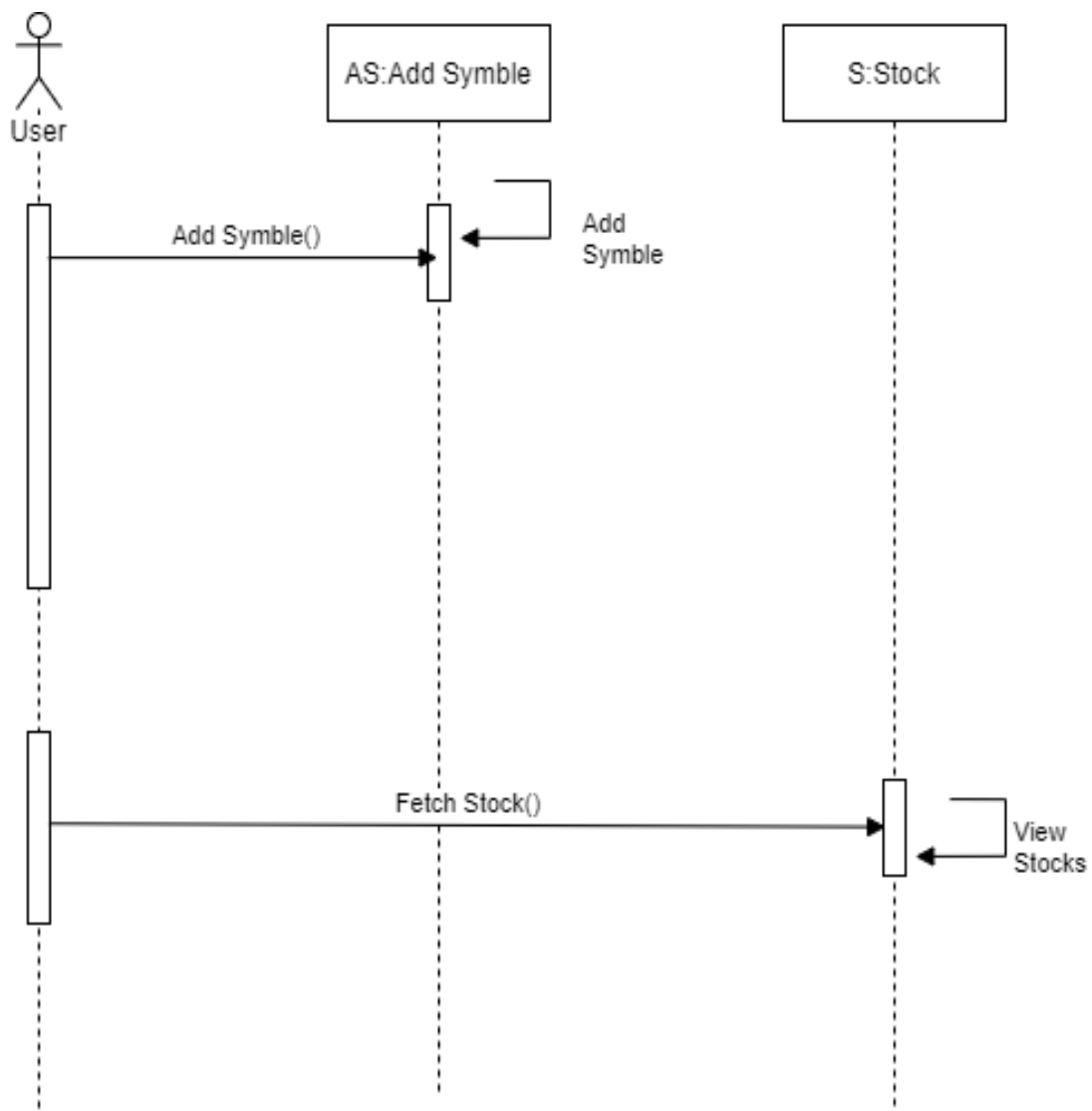
The total system will follow the MVT pattern for these the model and view class diagrams are given below:



**Figure 6: Model and View classes for Stock Reader**

## 8.6 Sequence Diagram

Here is the sequence diagram and overall representation is shown through following figure –



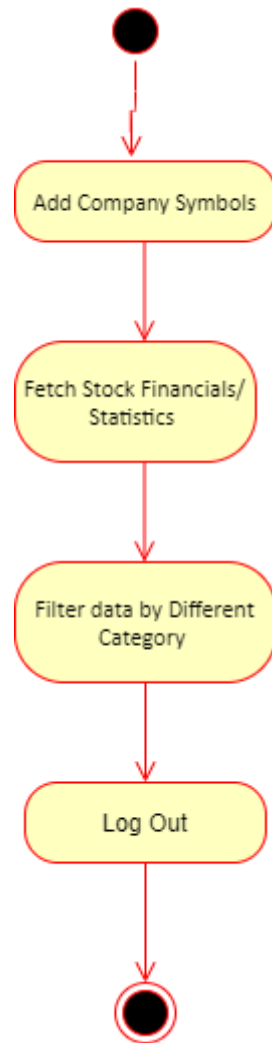
**Figure 7: Sequence Diagram for Stock Reader**



## 8.7 Activity Diagram

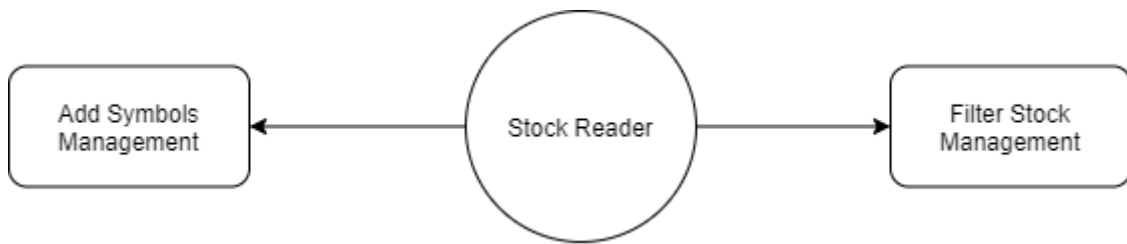
To describe the SDS more specifically there are some activity diagrams to elucidate the system more distinctively.

### 8.7.1 Overall Tasks by User



**Figure 8: Activity Diagram for Stock Reader**

## 8.8 Data Flow Diagram

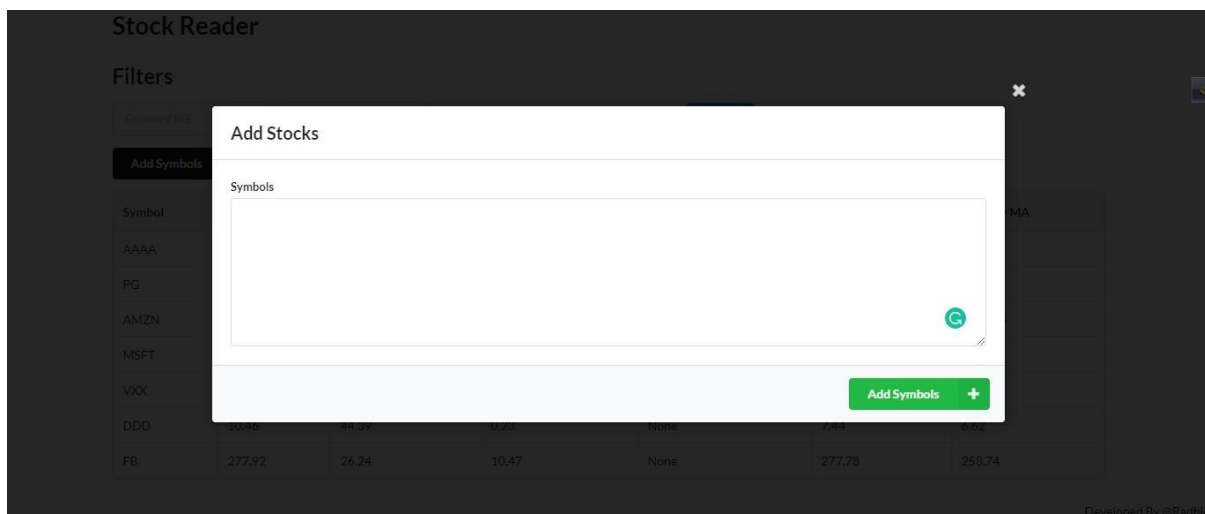


**Figure 9: Context Diagram for Stock Reader**

## 8.9 User Interface Design

In this section there is a concise physical view how the system is intended with the input and output. With the design the development team can find ease to validate the system with Graphical representation.

### 8.9.1 Input Design



## 8.9.2 Output Design

### Stock Reader

#### Filters

☐ Above 50 Day MA
 ☐ Above 200 Day MA

Add Symbols

Symbol	Price	Forward P/E	Forward EPS	Dividend Yield	50 Day MA	200 Day MA
AAAA	None	None	None	None	None	None
PG	137.47	23.16	5.91	2.30	140.26	132.18
AMZN	3162.58	69.56	45.37	None	3161.83	3047.35
MSFT	None	None	None	None	None	None
VXX	17.11	None	None	None	20.76	27.03
DDD	10.46	44.39	0.23	None	7.44	6.62
FB	277.92	26.24	10.47	None	277.78	258.74

Developed By @Radhio

## 8.10 Entity Relationship Diagram

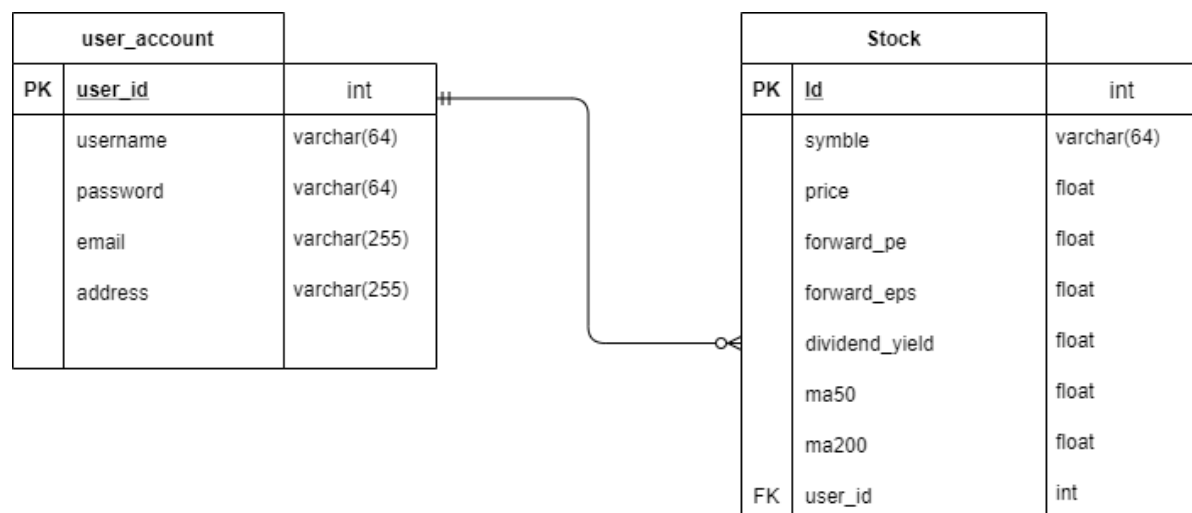


Figure 10: Entity Relationship Diagram for Stock Reader

## **9. Non-Functional Requirements**

### **9.1 Performance Requirements**

Server software does not require any special hardware other than the minimum hardware required for running enterprise OS. Extra disk storage will be required for archives and electronic documents. Increases of memory enables efficient query processing, which is required for quick bibliographic search. Two server grade processors with clock speed 3.0 GHz, at least 8GB RAM and 300 GB hard disk is recommended for the server. Client machine with recommended hardware required for desktop operating system and web browser (with open JavaScript enable).

### **9.2 Safety Requirements**

As per any work place safety rules and their server room where the server is supposed to be placed and the monitoring people.

### **9.3 Security Requirements**

Each time there is a security violation, the log file will be updated with the login, date, and time. Again, high level cryptography and checking should be kept to make it more secured. However, while email or request from any unwanted client the request should drop and let that user know about the fault.

### **9.4 Maintainability Requirements**

At least one backup server with same configuration as in main server is also recommended for fault tolerance and better performance. Separate storage (with backup) for database, electronic document, and manuscript is also recommended. Multiple computing nodes with the storage are required for high availability and to enhance the performance of the application. Again, after a certain period the preliminary manuscript files and other files related with that can be deleted manually from the database to increase the performance.