Software Requirements Specification

for

Stock Simulator

**Version 1.0**

**Prepared by**

**Group Name:**

|  |  |  |
| --- | --- | --- |
| **Rushabh Vinchhi** | **60004150124** | **rushabhuvinchhi@gmail.com** |
| **Pranav Shirke** | **60004150117** | **shirkepranav01@gmail.com** |
| **Prapti Shanbhag** | **60004150115** | **praptee@gmail.com** |
|  |  |  |
|  |  |  |

|  |  |
| --- | --- |
| **Instructor:** |  |
| **Course:** | **SE** |
| **Lab Section:** | **Compters, TE-B** |
| **Date:** | **5th March, 2018** |
|  |  |

**Contents**

**REVISIONS III**

**1** **INTRODUCTION 1**

1.1 Document Purpose 1

1.2 Product Scope 1

1.3 Intended Audience and Document Overview 1

1.4 Definitions, Acronyms and Abbreviations 1

1.5 Document Conventions 1

1.6 References and Acknowledgments 2

**2** **OVERALL DESCRIPTION 3**

2.1 Product Perspective 3

2.2 Product Functionality 3

2.3 Users and Characteristics 3

2.4 Operating Environment 3

2.5 Design and Implementation Constraints 4

2.6 User Documentation 4

2.7 Assumptions and Dependencies 4

**3** **SPECIFIC REQUIREMENTS 5**

3.1 External Interface Requirements 5

3.2 Functional Requirements 6

3.3 Behaviour Requirements 6

**4** **OTHER NON-FUNCTIONAL REQUIREMENTS 7**

4.1 Performance Requirements 7

4.2 Safety and Security Requirements 7

4.3 Software Quality Attributes 7

**5** **OTHER REQUIREMENTS 8**

**APPENDIX A – DATA DICTIONARY 9**

**APPENDIX B - GROUP LOG 10**

**Revisions**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Primary Author(s)** | **Description of Version** | **Date Completed** |
| Draft Type and Number | Pranav Shirke,Prapti Shanbhag, Rushabh Vinchii | Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded. | 00/00/00 |

# Introduction

The aim of the project is to provide the users a stock simulator platform for prospective users to learn about the stock market in and out via a game format. In this section we discuss about the purpose objectives and outlines of the project. Thus we have implemented a useful stock tool for the market.

## Document Purpose

The purpose of the documentation is for prospective investors to make the most out of our web application which mirrors the real stock marketplace. This system will allow its users to participate in stock market simulated games. It could also help people understand the investment strategies of other users, perhaps enabling them to better control their investment in the actual stock market.

## Product Scope

Stock Investments League is geared towards a wide array of audiences and expects a variety of users with varying knowledge levels to participate. In order to maintain appeal amongst these users the platform should provide rewards to users for acheiving particular goals. We would like to replicate the idea of achievements or trophies similar to the Microsoft xBox and Sony Playstation family of systems. These achievements can award users with new abilities or additional cash to their portfolio as they rise up the achievements ladder. Users should also be able to create leagues to help further enhance the competitiveness of the game.

## Intended Audience and Document Overview

This document is intended for:

1. Client
2. Developers
3. Prospective investors
4. Subject Matter expert
5. Analyst
6. Tester

## Document Conventions

The System Requirement Specification document follows the standard IEEE SRS conventions.

## References and Acknowledgments

<http://cs229.stanford.edu/proj2015/151_report.pdf>

<http://cs229.stanford.edu/proj2016/poster/MaoYuWang-Deep%20learning%20based%20food%20recognition-poster.pdf>

<https://simonb83.github.io/machine-learning-food-classification.html>

# Overall Description

## Product Perspective

This website is a standalone system. The entire experience would be unified across mobile, tablet, and the desktop and combined with the above features provide an enthralling core experience for users to learn about the stock market.

## Product Functionality

Leagues exist to allow multiple users to compete against a subset of the global user base with individual league rules. This allows leagues to set particular goals in order to be declared the winner. Leagues will require a cash buy-in that will be pooled together and distributed to the winner(s) as seen fit by the league creator. To help facilitate these leagues, a leader board will be created for each individual league such that users can see their progress. In addition to league leader boards, mutliple global leaderboards will be available providing specific metrics of comparison.

## Users and Characteristics

Investor: A user who has an account in our servers and is logged in to their account. These will be regarded as our primary stakeholders.

Guest: A visitor to the website who has either not logged in or just a simple visitor. Visitors will be regarded as secondary stakeholders.

Administrator: Also referred to as an Admin, is the person in charge of managing the operation of their respective field. For instance, a League admin is in charge of maintaining league settings.

## Operating Environment

The hardware requirements on the server side are the main contribution to the operation of Stock Investments League, leaving the client-side with minimal requirements. In fact, the only requirement of a client will that it runs a browser that is capable of running a modern web browser.

## Design and Implementation Constraints

1. To provide a secure method of entering usernames and passwords, as well as protecting user accounts.
2. The current and uploaded training data must be stored in a database.
3. The system must run 24 hours a day.
4. Users may access from any computer that has internet browsing capabilities and an Internet connection.
5. The system should be able to handle invalid inputs.

## User Documentation

The website is easy to understand and use with a simple registration and login. The graphical user interface is self-explanatory and no tutorials or manuals need to be provided to the user to understand how to use the stock simulator.

## Assumptions and Dependencies

The team is assuming that all desired features are possible within our time constraints. Placeholders must also be created for future features to be implemented.

The team is also assuming that the KNN algorithm will give them enough accuracy and that response time, once image is uploaded, will be reasonable.

# Specific Requirements

## External Interface Requirements

### User Interfaces

User Registration:

For the first time user, they can click Signup on the welcome page which redirects him/her to the signup page

Main Page :

Once registered, the user will be brought to the main page. Here there are several things they can do.

Buy/Sell Stock :

By pressing the purchase stock button on the main page, a small pop up window will appear and the user can enter the stock they want and the amount they want to purchase. The same pop up window will appear if the user clicks the sell stock option

### Hardware Interfaces

Our system will require only Internet connection and web browsers from our users. Our system will run on any web browser. The system won’t require any hardware space for this application. It will save all the information on our ‘MoneyMachine’ servers. Using their preferred browsers, without installing any software, Users easily connecting to their Internet, and enjoy and experience real life Stocks, and It will be an amazing experience for our users.

### Software Interfaces

It will also be compatible with the latest stable version of Google Chrome, Mozilla Firefox and Microsoft Edge on any operating system.

### Communications Interfaces

The user needs a computer, access to the Internet and an internet browser to view and use the services the website provides. The user also needs an email ID to create an account.

## Functional Requirements:

## The prediction shall abide by the following functional requirements:

1. Prior to application of stock recommendations, the database is updated by the latest values.

2. The charts and comparison of the companies would be done only on the latest data stock market data

3. The user is provided with a login, logging into which enables the user to view his past stock purchases and future recommendations.

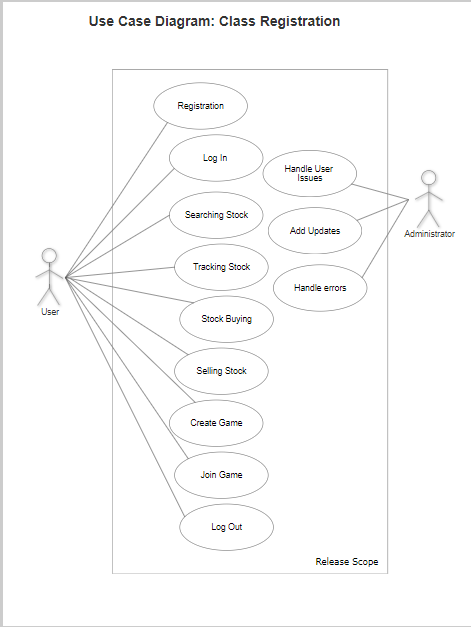
4. The user can look previous data Information which was collected.

5. Each user has a friend list and can also be recommended on their buying patterns.

6. The user can also be recommended on the basis of the trending stocks which would require the data regarding the stocks.

## Behaviour Requirements

### Use Case View



# Other Non-functional Requirements

## Performance Requirements

The stock simulator will have the following performance requirements:

1. A user will be able to access only portfolio at a time.
2. The portfolio size that is alloted should not be over 100MB.
3. The accuracy with which the algorithm will be able to identify the profit margin of a certain stock will not be over a certain percentage.
4. The website will have at least one new blog to feature every day on the ‘Blog’ page.

## Safety and Security Requirements

* Listed companies must be secured and all stock listings must be verified
* Details provided during login are stored in a secure way and are not prone to leak.
* If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage (typically tape) and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed up log, up to the time of failure.

## Software Quality Attributes

* **Availability:** The investor portfolio should be available for each user so the algorithm can learn from them
* **Correctness:** The share prices should be as close to the real value as possible
* **Maintainability:** The administrators and should maintain correct data and remove the incorrect one
* **Usability:** The simulator should satisfy a maximum number of customers’ needs.