Software Requirements Specification

for

Online Trading System

Version 1.0 approved

prepared by,

Group 11
19z334 Nikhil
19z335 Niranjan S
19z339 J .Priyadharshini
19z340 Venkat
19z349 Srikanth K R
19z355 M.Swetha

PSG college of Technology,Coimbatore

26-08-2021

TABLE OF CONTENTS

Revision History

Name	Date	Reason For Changes	Version

1.Introduction

1.1 Purpose

The main purpose of this application is to manage inventory with ease. This Online Trading System enables users to buy or sell shares online. Compared to the traditional stock exchanges between the trader and the investor the online trading system provides a greater transparency in trading.

1.2 Document Conventions

We have used a standard font for documentation - Times New Roman used a font of 14 points.

1.3 Intended Audience and Reading Suggestions

The main purpose of this application is to manage inventory with ease. This Online Trading System enables users to buy or sell shares online.

Compared to the traditional stock exchanges between the trader and the investor the online trading system provides a greater transparency in trading.

1.4 Product Scope

The main scope of our project is to be user-friendly and easy to use. This Online Trading System tends to be cheaper and convenient as it eliminates the direct contact with the broker. The monitoring of the trading throughout the day by the investors provides a hassle free environment. Overall the project tries to overcome the problems faced in the traditional stock exchange.

1.5 References

SRS Template: IEEE Template →
https://web.cs.dal.ca/~hawkey/3130/srs_template-ieee.doc
• Roger S Pressman, Software Engineering - A Practitioner's Approach, McGraw Hill International Edition, Singapore, 2015.

2. Overall Description

2.1 Product Perspective

"Online Trading System" is aimed to provide most if not all the essential data and facilities a typical trading system provides. Customers should be able to use this web-based application to execute a variety of stock-related tasks, such as monitoring stock prices and news, as well as purchasing and selling stocks.

2.2 Product Functions

This product aims to provide the following features to all its users currently (needless to say, the basic login and other generic functions have been omitted in this section), while leaving a huge scope for multiple changes in the near future:

- Visualization of processed and analyzed stock data;
- Separation of volatile stocks from the stagnant ones;
- Filtering products based on their utility;
- Live stock-influencing newsfeed;
- A mock watchlist that holds the user's chosen stock products;

2.3 User Classes and Characteristics

There are various kinds of users for this product:

- Investors:

  - There are many sorts of investors, each of them has their own trading style, but the final aim is to earn a profit or make smart high-return investments for the same; namely four of them include:
  - Active Investors:
    - They are mostly looking for the next big thing in the stock market that will provide them with higher returns.
  - Speculators:
    - Simply put, speculators are investors who wait for the ideal time to invest before making a decision.
  - Passive Investors:
    - Passive investors are people who desire peace of mind and don't want to worry about their assets and they are long-term investors who will sell their holdings if they appear to be lucrative.
  - Value Investors:
    - Value investors, as the name implies, consider the stock's value and price before investing and they are meticulous in their investments and lie halfway between active and passive investors.

- <u>Traders</u>:
  - A stock trader is someone who makes money by buying and selling securities such as stock shares. Individual stock traders or professionals dealing on behalf of a financial firm are both examples of stock traders.
  - A few types of Traders include:
    - Day Trader:
      - Someone who enters and leaves several positions in a single day are known as intraday traders because they never retain a position from one trading day to the next.
    - Swing Trader:
      - A swing trader spends more time monitoring equities and weighing the options available. Swing traders can hold a position for days in order to capture the bulk of a price change in a security.
    - Buy and Hold Trader:

            - A long-term trader is a buy and hold trader and this is the most frequent strategy, in which the trader purchases shares in a solid firm rather than one that is trending.
        - Momentum Trader:
            - A momentum trader takes a long or short position in a stock, concentrating on the stock's price acceleration, or the company's revenue or profits acceleration and they take these positions based on the belief that the current trend will continue.

- <u>Brokers</u>:
    - A broker is a person or an organisation that organises and executes financial transactions on behalf of another person or company.
    - More specifically, a Stock Broker is a registered financial market representative who facilitates the buying and selling of securities for various clients.
    - A few types of Stock Brokers include:
        - Full-Service Stock Broker:
            - A full-service stockbroker provides clients with a wide range of financial services. Clients are usually assigned to specific licenced stockbrokers. Research departments of brokerage companies provide analyst recommendations and early access to initial public offerings (IPOs).
        - Discount Stock Broker:
            - Financial goods, mutual fund access, banking products, and other services are all available through discount stockbrokers. A cheap stock broker provides many of the same goods and services as a full-service stockbroker, but at a lower cost.
        - Online Stock Broker:
            - An online stockbroker, often known as a direct access stockbroker, provides services to active day traders at the lowest possible commission — generally on a per-stock basis. Direct access platforms with routing and charting

capabilities, as well as access to various exchanges, market makers, and electronic communication networks (ECN), are available through online stockbrokers.

- Banks:
    - Banks are companies, and they must generate revenue from a number of sources in order to remain successful.
    - The majority of a bank's revenue comes from the interest it charges on client loans.
    - Fees charged by the bank, as well as profits from investments made by the bank, provide additional revenue.
    - Stock ownership can provide investment income in the form of capital gains on stock sales as well as dividends paid to the bank by the stock issuers.

2.4 Operating Environment

- The website will most likely be hosted online.
- It will be mostly compatible with any device that has an internet connection and a web browser.
- To reduce compatibility concerns to a minimum, the website will be coded using industry standard techniques.

2.5 Design and Implementation Constraints

- The team puts out their utmost effort in the system's development while some design and implementation limitations are enforced in order to ensure the system's dependability and durability.
- The availability of a mobile app for the “Online Trading System” would make the system portable, but this is not achievable owing to schedule constraints.
- We have the opportunity to work with new tools like Vanilla JS and Python/Flask while building the system and its user interfaces.
- We chose to develop such interfaces in a basic, realistic style utilising inexpensive technologies, taking into account the team's expertise in this arena.

2.5 User Documentation

- A User-Manual in the form of another web-page will be coded along with the system.
- It will be written in an easy-to-understand language that hides the system's underlying complexity.
- With the delivery of the system, the recipient will receive a live demonstration too.

2.6 Assumptions and Dependencies

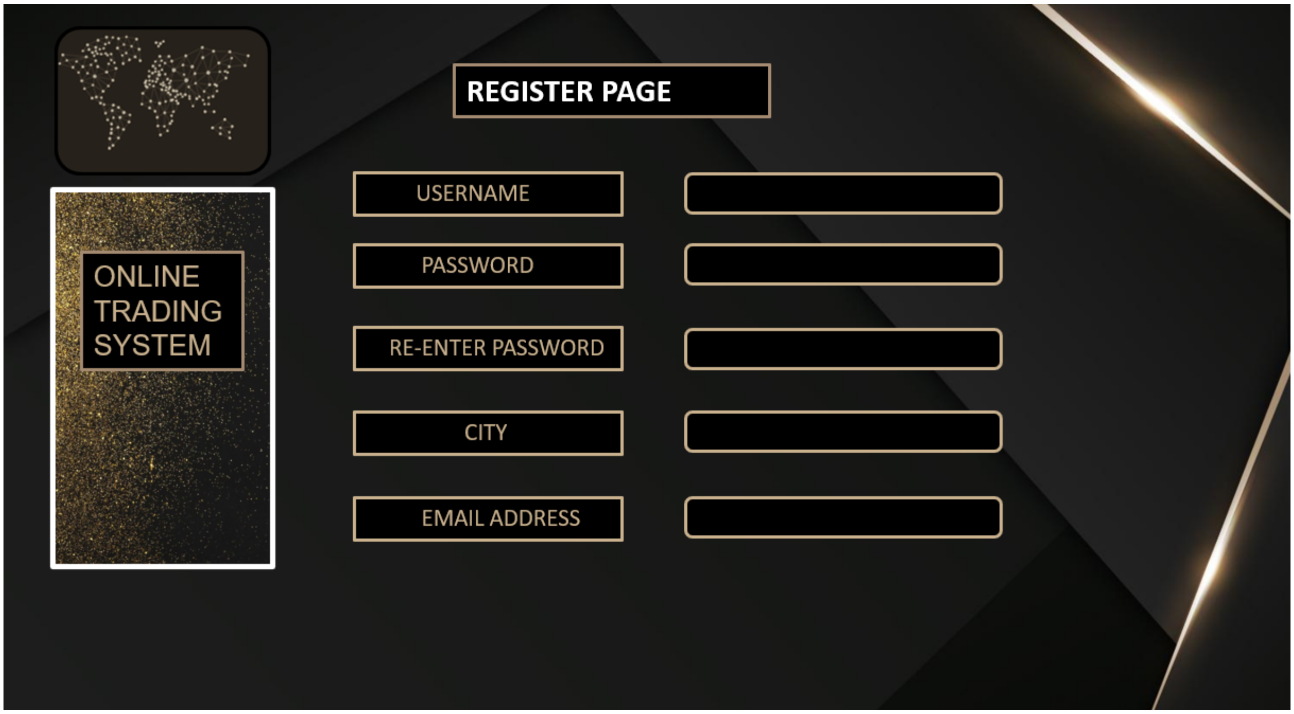
- The opted technology has been tested in similar systems and has shown to operate in gear.
- This is, after all, a fundamental stock trading system. Us developers have a tendency to overlook security and since the number of end users is small, it cuts down on the effort put.
- Hence the recipients are requested to overlook the following assumptions and constraints which the team will put in work to overcome soon:
  - Only the stock markets with widespread recognition are taken into account;
  - Accounts can only be opened by persons above the age of 18;
  - Any sort of network difficulties that may arise should be ignored;
  - Ignore any concerns about contract negotiations, legal issues and IRS tax reporting;
  - Ignore any difficulties that arise after the project is completed; post-project maintenance issues are tedious.

3.External Interface Requirements

3.1 User Interfaces

The first interface the user interacts with is the register page where he/she has to give the following details

- User name
- Password
- Re-enter password
- City
- Email address



REGISTER PAGE

ONLINE TRADING SYSTEM

USERNAME

PASSWORD

RE-ENTER PASSWORD

CITY

EMAIL ADDRESS

Once he/she is registered the user is redirected to the login page where he is prompted for his username and password.

If the login credentials are incorrect then the user is redirected to an apology webpage where the error message is displayed.

GUI standards -

System status - The user is always informed about his/her stock product statuses.This may include the notification of the following actions-

➔ When the user's stock has a hit a huge profit value

➔ When the stock value decreases beyond the expected limit

Match between system and the real world - Though the system has a bunch of data which is non-graphical, the user’s expectation will be to visualize and analyse the trade easily. So, we have designed to express such raw data as a graphical visualization as the value changes in the global market.

Asking help - Also, we will have an additional functionality of ‘help’ where there will be certain guidelines and instructions to be followed if the user is unaware of what is to be done.

Error handling - When an error occurs, the user will be alerted of what should be done at that point. An example of error is when a user tries to buy a stock without mentioning the name which he wants.

Software components -

- ★ The components the user will mostly need will be a search engine where when he types a required text, a pop up will show all matching details.
- ★ Navigational components - For each activity like buy , sell and so on, the user will have a navigation bar with tabs also an icon for help desk.
- ★ Buttons - for user interaction. Each button click will specify a unique task.

  2 buttons are common in all pages. -Help
  -Log out
- ★ Text box- For the user to enter his/her value.

3.2 Hardware Interfaces

Hardware components -

Our website will run on all devices including mobile phones. The main server is our local harddisk. All the requests are handled by HTTP and sent to our server and corresponding actions are taken.

3.3 Software Interfaces

Tools-

Our online trading website uses the following software components

A backend microframework named ‘FLASK’. Version - 2.0.0. It is called microframework because it has the advantage,where we do not need to install any particular tools or libraries.

Language used is Python. Version 3.9

Database used is MySQL. Version 8.0.21

Operating system-

Windows 7 or higher, Mac, Linux.

Integrated commercial components -

For accessing the real world data, we use a component called IFX token. IFX cloud server offers global data resources and allows virtual storage using the internet which is a public connection.

3.4 Communications Interfaces

Our software’s communication interface is through a web browser which is connected to a remote server and communicates with HTTP requests.

Also, once the user buys or sells stock ,he gets an email notification which will be done using Sessions.

For each user, once he is registered, we use encryption to store and access his Session id. The constraint is the user should accept cookies to enable this and here after, the system will remember the user.

4. System Feature

The following explains main features about online trading services.

4.1 Sign up

4.1.1 Description and Priority:

The user must enter their details to create an account on this platform. This is a high priority feature, without creating an account the user cannot use this platform.

4.1.2 Stimulus/Response Sequences:

Details that the user must enter:

- Username
- Email address
- Password
- Ph.no
- Bank account number
- PAN number

4.1.3 Functional Requirements:

REQ-1: A database is created to store the credentials of the user.

REQ-2: A program that acknowledges, approves, stores the client credentials and sends the affirmation mail

4.2 Login

4.2.1 Description and Priority:

The user must provide the username or email address that was provided during the register process, as well as the password. If the credentials are legitimate, the user is granted access to the platform; otherwise, the user must re-enter the credentials. This functionality is of high priority.

4.2.2 Stimulus/Response Sequences:

The has to enter:

- Username
- Password

After the credentials are checked, the user will be given access to the platform on whether the details are valid or not. If the login credentials do not match, the user's request for login will be declined, and the user will have to re-enter the details correctly.

4.2.3 Functional Requirements:

REQ-1: The user credentials which are kept in a database.

REQ-2: A program that accepts and validates the credentials entered. If the details are correct, the application will let the user in; if the details are incorrect, the user will be asked to re-enter their credentials.

4.3 Log Out

4.3.1 Description and Priority:

The user can press the log out button to logout of the platform. This feature is a low priority.

4.3.2 Stimulus/Response Sequences:

The user has to press the logout button. Then the application quits and goes to the login page.

4.3.3 Functional Requirements:

REQ-1: A program that accepts user input, and directs the user to the next step in the process.

4.4 Search For Equities

4.4.1 Description and Priority:

The user can search for any companies' stock or commodities and select them. This feature has a medium priority in this platform.

4.4.2 Stimulus/Response Sequences:

The user has to manually type the name of the equity/commodity name to select them. The platform program accepts and provides a detailed analysis of the selected item.

4.4.3 Functional Requirements:

REQ-1: The database in which the price and analysis of the selected item is stored.

REQ-2: A program that accepts user input, and directs the user to the next step in the process.

4.5 Buy Stocks

4.5.1 Description and Priority:

After selecting the stock as done in the above feature. The user can buy the available stocks on the platform. This feature is of high priority.

4.5.2 Stimulus/Response Sequences:

The user has to press the buy button to buy stocks. Every change is stored and kept track of every time when the user buys stocks.

4.5.3 Functional Requirements:

REQ-1: A program that detects changes when the user buy stocks, stores the changes and keeps track of all the changes

REQ-2: A program that accepts user input, and directs the user to the next step in the process.

4.6 Sell Stocks

4.6.1 Description and Priority:

After selecting the stock as done in the above feature. The user can sell the stocks from his/her on the platform. This feature is of high priority.

4.6.2 Stimulus/Response Sequences:

The user has to press the sell button to sell stocks. Every change is stored and kept track of every time when the user sells stocks.

4.6.3 Functional Requirements:

REQ-1: A program that detects changes when the user sell stocks, stores the changes and keeps track of all the changes

REQ-2: A program that accepts user input, and directs the user to the next step in the process.

4.7 Holdings

4.7.1 Description and Priority:

The user can view their holdings (equities bought). This feature has a low priority, as the user doesn't need to use this feature always. This feature is of low priority.

4.7.2 Stimulus/Response Sequences:

The user has to press the holdings button to view their holdings. The user is sent to the holdings page.

4.7.3 Functional Requirements:

REQ-1: A program that displays the user's holdings from the database.

4.8 View Chart:

4.8.1 Description and Priority:

The user, after searching and selecting their desired stock, can view a chart that gives a detailed analysis of the stock. This feature is of low priority.

4.8.2 Stimulus/Response Sequences:

The user has to press the view chart button to open the chart. The user is sent to the view chart page.

4.8.3 Functional Requirements:

REQ-1: A program that displays the user's holdings from the database after getting the input.

4.9 History:

4.9.1 Description and Priority:

This is a low priority feature, which shows the history of transactions of the user.

4.9.2 Stimulus/Response Sequences:

The user has to press the history button to view transaction history. The user is sent to the history page.

4.9.3 Functional Requirements:

REQ-1: A program that displays the user’s holdings from the database after getting the input.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

1. Website Development

For the first part of the development stage, we need the help of experienced people who provide developers skilled in programming languages such as python or java .

2. Backend Development

The backend development primarily includes the development of the server and database of our desktop-based application. For this, we require a proficient backend developer, basically, someone who deals extensively with database planning and structuring, and Website development services**.** It is during this phase that all the functions such as notifications, sign in, and transactions are going to be developed. For this , we are using the python flask framework to satisfy our first two requirements of having a site that reuses code and is dynamic. It allows you to get a site going with one small python file, and add all the HTML elements in a folder called templates which makes the web application building process much easier.

3. Frontend Development

The frontend development refers to the development of the user-side of the application, the screens, the interface, and everything the user will be interacting with is incorporated in the frontend development. For our front end, HTML is used.

4. Payment gateway integration

The payment gateway must be chosen very carefully, not only it should be safe and secure but provide major options of payment like e-wallets, debit and credit cards, and net banking.

5. The UI/UX Considerations

The stock market is such a place where one incorrect judgment can drive to a major failure. While keeping this in mind, we want to build a successful online trading platform by keeping the design uncomplicated and efficient .We started outlining the UI/UX design while planning for the application, and formulated ideas and requirements for the application.

5.2 Safety Requirements

- Authorized users can update the entity's profile directly on the website.
- Date and user name are logged when profile is changed.
- User bank details are validated before any trading operation and proof documents of users are constantly checked for validation and reviewed.
- Password reminders and resets are handled by the website .
- With the exception of system administrators, all user accounts are tied to our system.
- Change login records details on data modification (date, username).
- Buyers and sellers validated details are accurately specified for a very safe trading operation.

- Stocks that are put on watchlist by the users, if any updates on stock prices , a constant update notification is sent to the user's email for prevention of data miss out.

5.3 Security Requirements

The basic key web services security requirements are authentication, authorization, data protection.

1.Authentication

Authentication ensures that each entity involved in our Web service—the requestor, the provider, and the broker (if there is one)—is what it actually claims to be . While the user signs up in our system, we take all our user's information and also they need to submit a strong validation proof along with that such as passport, citizenship ID and also we keep on checking the information validity and if not , we freeze the accounts till the user updates it.

2.Authorization

Authorization determines whether the service provider has granted access to the Web service to the requestor. Basically, authorization confirms the service requestor's credentials. As the user signs up, he/she needs to fill their username and password and login to our system. Also, while users sign up, we authenticate all the information of the user. In case they forget the password , an email from our system is sent to that particular user email for verification.

3.Data Protection

Data protection ensures that the Web service request and response have not been tampered with enroute. It requires securing both data integrity and privacy. As we are using Flask, it uses cookie based sessions by default, but there is support for custom sessions that store data in other places. In particular, the Flask-Session stores the user session data in the server, giving you a variety of storage options such as plain files, Redis, relational databases, etc. When the session data is stored in the server you can be sure that any data that you write to it is as secure as your server.

5.4 Software Quality Attributes

1.Usability and Reusability

It is an Globally secured site for all our users and our overall objective is very customer specific. We target what is essential for our users satisfaction and is made for very easy use and understanding for all our users. We have made our web application so simple, aesthetic and clear so it pleases our customers when they visit our application. Our Interface offers so many special features for making our customers easy to understand each stock history and make use of our mock watchlist in which they can monitor the prices of their chosen stocks, choose volatile stocks over stagnant ones, and analyse stock based on their utility.

2.Functionality and Maintainability

- Users can search recent stocks which they had viewed recently through a search column so they can recover data easily and efficiently in no time.
- Our navigating features are so precise so it takes to the tasks what our users had exactly expected.
- While our users use our application for the first time, after signing up and logging in,a proper demo is illustrated for our users to guide them throughout our application and tell us about what we exactly offer to them.

After our user uses our system for the first time, they can easily adapt to our interface.

3.Reliability and Efficiency

Our web application has great response time performance, page generation speed,very safe online transactions and so overall, our customers would be pleased.If transactions failed, it would be automatically refunded so we have great error recovery rate. Information regarding user log in details are regularly checked for validation and thus, we prevent any form of spams affecting our system.

5.5 Business Rules

- After entering our application, in order to get in, the user must sign up and add all their validated details and then log in to access our system.
- Information is checked for validity and thus, if any invalid information given by the user, their account would be frozen.
- In order to activate buy/sell stock transactions, their account must have a minimum balance of Rs.1000.
- After finishing the process, the user must log out.

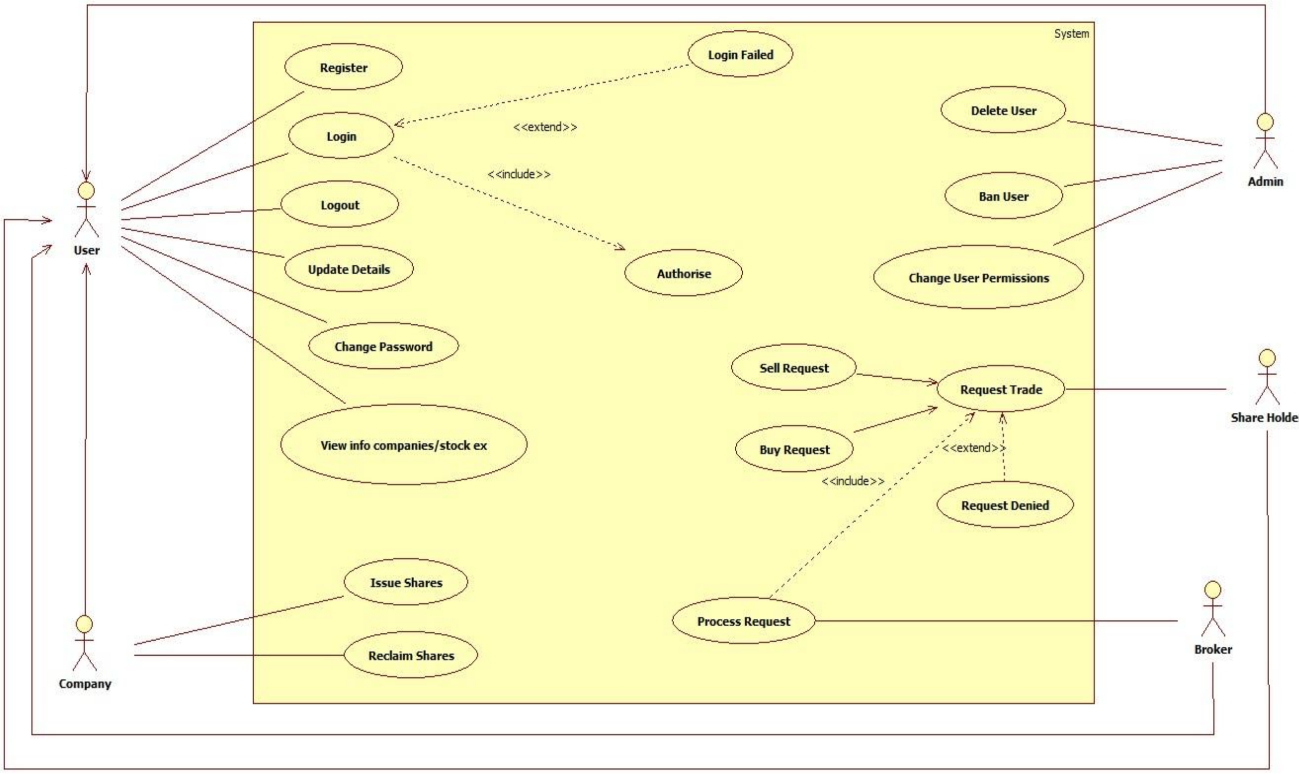
6. Other Requirements

6.1 Appendix A: Glossary

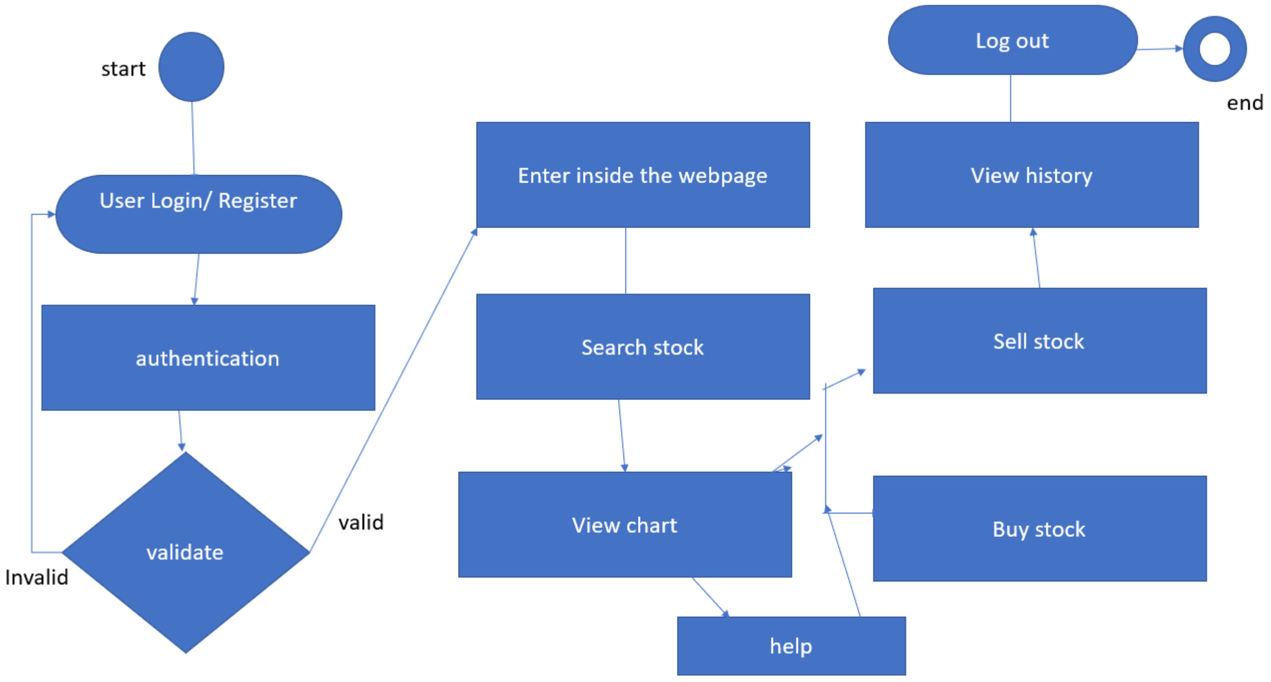
OTS– Online Trading System
SRS – Software Requirement Specifications
OS – Operating System

6.2 Appendix B: Analysis Models

Use case diagram



Activity diagram



6.3 Appendix C: To Be Determined List

1.Storage size – size may vary according to the number of choices we provide for users.
2.Traffic handling - How much time it takes to load for a user if the traffic is heavy.