Software Requirements Specification

For

Virtual Stock Market

Prepared by: Aneesh S (12IT06)

Kartik Koralla (12IT33)

Keerthi Prasad N (12IT34)

Suhas H S (12IT85)

Table of Contents

1. Introduction

1.1 Purpose

The aim of this project is to design and implement a web-based software system for virtual stock trading. Users can log in to system and they can buy and sell socks. The virtual system will permit the users to practice and learn how the trade market works without real money. Moreover, the advantage is that you can practice anywhere because it is an online application and it is open 24 hours a day. The trading prices will be obtained by matching the buying and selling orders, each with their target price, as in the real market. The system will record the history of stock prices and provide graphical tools to present this data. Users will be able to view their personal information, own stocks, transaction history and their current orders to buy and sell and the prices of the stocks. The system is web-based application designed using appropriate technologies and will support a relatively large number of users. The application will show a list of the company's stocks with their current prices and a list of buyers and sellers with the name of the stocks that they can buy or sell with the number of stocks and their prices. Moreover, users will be able to see the orders sorted. The best order in the case of buy is the major price will appear first and the worst, in this case the lower price will be appear at last and the orders to sell, the lower price will appear first and the major price will be at last.

1.2 Objectives

The main objective of my project is to set an Apache Web Server up and running and a Database server in MySQL. The web pages will be done in HTML, PHP, CSS and JavaScript. These tools will permit users connect to the system through the Internet from anywhere with any Operating System and browser. The data will be saved in the SQL database and the information about the purchases and sales can be accessed with RSS. The users can see how the transactions change and how the orders of buying and selling match them. If there is a buyer who wants to buy a stock for a certain amount of money

and there is a seller who wants to sell this stock, it will be a transaction. In this application the current price of the stocks in the trade market will be the price of the last sale. Users would be able to see the graphics about the history of the stock. The graphics will be done with a PHP's library. Moreover, users can generate a PDF file with his information, his data and his stocks and can get update notifications using RSS.

1.3 Project Features

The product is restricted to beginners in the trade. Users can see the variations of the prices of stocks in the trade by means of graphics. Also, users can see their own stocks and they can buy and sell stocks. Users will be able to see all information anywhere because the application is in a server connected in the Internet. Henceforth, users can consult the information and do changes when and where they want. The application is programmed using HTML, PHP, JavaScript and XML to do the RSS. Moreover, users can export a PDF file with their stocks and their personal information. Furthermore, users can receive RSS about the new purchases and sales.
The system works in any Operating System and browser.

1.4 Product Scope

PDF: (Portable Document Format) is a file format created by Adobe Systems for document exchange. PDF is used for representing two-dimensional documents in a manner independent of the application software, hardware, and operating system. The documents in a PDF format have the smaller in size. PDF is an open standard.

RSS: is a family of Web feed formats used to publish frequently updated works in a standardized format. They benefit readers who want to subscribe to timely updates from favored websites or to aggregate feeds from many sites into one place. RSS feeds can be read using software called an "RSS reader", "feed reader", or "aggregator", which can be web-based, desktop-based, mobile device or any computerized Internet connected device.

A standardized XML file format allows the information to be published once and viewed by many different programs.

PHP: is a scripting language originally designed for producing dynamic web pages. It has evolved to include a command line interface capability and can be used in standalone software and graphical applications. PHP was created by Rasmus Lerdorf in 1995. PHP is produced by **The PHP Group.** PHP is free software released under the PHP License. However it is incompatible with the GNU General Public License (GPL), due to restrictions on the usage of the term

PHP is a widely-used general-purpose scripting language that is especially suited for web development and can be embedded into HTML. It generally runs on a web server, taking PHP code as its input and creating web pages as output. It can be deployed on most web servers and on almost every operating system and platform.

HTML (Hypertext Markup Language): is the predominant mark-up language for web pages. It is used to describe the structure and the content in a text manner and to complete the text with objects like images. HTML is written in the form of tags that are surrounded by angle brackets (<,>). HTML can also describe the appearance of a document, and can include a script (such as JavaScript) that can affect the behavior of Web browsers and other HTML processors. HTML also is used to refer to the MIME type content text/html or XML (like XHTML 1.0 or subsequent versions) or SGML (like HTML 4.01 and previous versions). HTML files use the extension .htm or .html.

JavaScript: is an interpret language program, it does not require compilation. It is usually used in web pages. It has syntax similar to Java and C language. JavaScript is an object oriented language like Java. It includes inheritance using the Prototype-based programming, the new classes are generated cloning the base classes (prototypes) and extending their functionality. All of the modern browsers interpret JavaScript code integrated in the web pages. It works with a DOM implementation. JavaScript was invented by Brendan Eich in the company Netscape Communications. This company developed the first commercial web browsers. The first time that appeared JavaScript was

in the Netscape product Netscape Navigator 2.0. Years before, it was using in a HTML web pages, to realize tasks and operations in the framework of a costumer application, without access to server functions. JavaScript is executed in the user agent and the sentences are downloading with the HTML code.

.

DOM (Document Object Model): is a cross-platform and language convention for representing and interacting with objects in HTML, XHTML and XML documents. The rules for programming and interacting with the DOM are specified in the DOM Application Programming Interface (API). The programs can access and modify the content, structure and style in the HTML and XML documents with DOM. W3C (World Wide Web Consortium) is the responsible of DOM.

1.5 References

- Stock-broking Today, J. Dundas Hamilton (MacMillan Press Ltd. ISBN: 0-333-24237-8)
- McGraw-Hill Dictionary of Modern Economics, Greenwald & Associates (McGraw-Hill)
- A Guide To Financial Markets, Charles R. Geisst (MacMillan Press Ltd. ISBN: 0-333-30919-7)
- The Work of the Stock Exchange, Sir Stephen Killik (E. Couchman & Co. Ltd.)
- Bulls, Bears, and Microcomputers: Programming for Successful Investment, G. T. Childs (Sigma)
- http://finesse.ac.uk/tags/finesse.html
- http://game.marketwatch.com/
- http://www.tradingsimulation.com/tsimlite/

2. Overall Description

2.1 System Features

The product is a web-based software system for virtual stock trading.

Functionalities

Database: the system keeps the information about users, companies, orders and stock's prices.
User management: users can register in the system and they can edit their personal information: name, address, age.

Company management: the information about companies (name, address, Website, total) is kept in the system.

Stock management: the system keeps each the stocks with his owner.

Order management: the system keeps the orders to buy and sell. Sellers can order sell stocks and buyers can order buy stocks.

Consult options: users can consult their stocks with the history of the prices.

.

2.2 Functional Requirements

Register: users can register on the system by giving their details, name, address and age.

Open session: users can open session to see more options.

Close session: users can close session.

See information about a specific ticker: both *logged users* and *anonymous users* can see the information about a specific ticker to see their evolution of prices in graphics.

See orders: users can see a list of orders with their prices and details.

See tickers: users can see a list of tickers with their prices and details.

Sell stocks: sellers can sell their stocks. They have to detail the price and the volume.

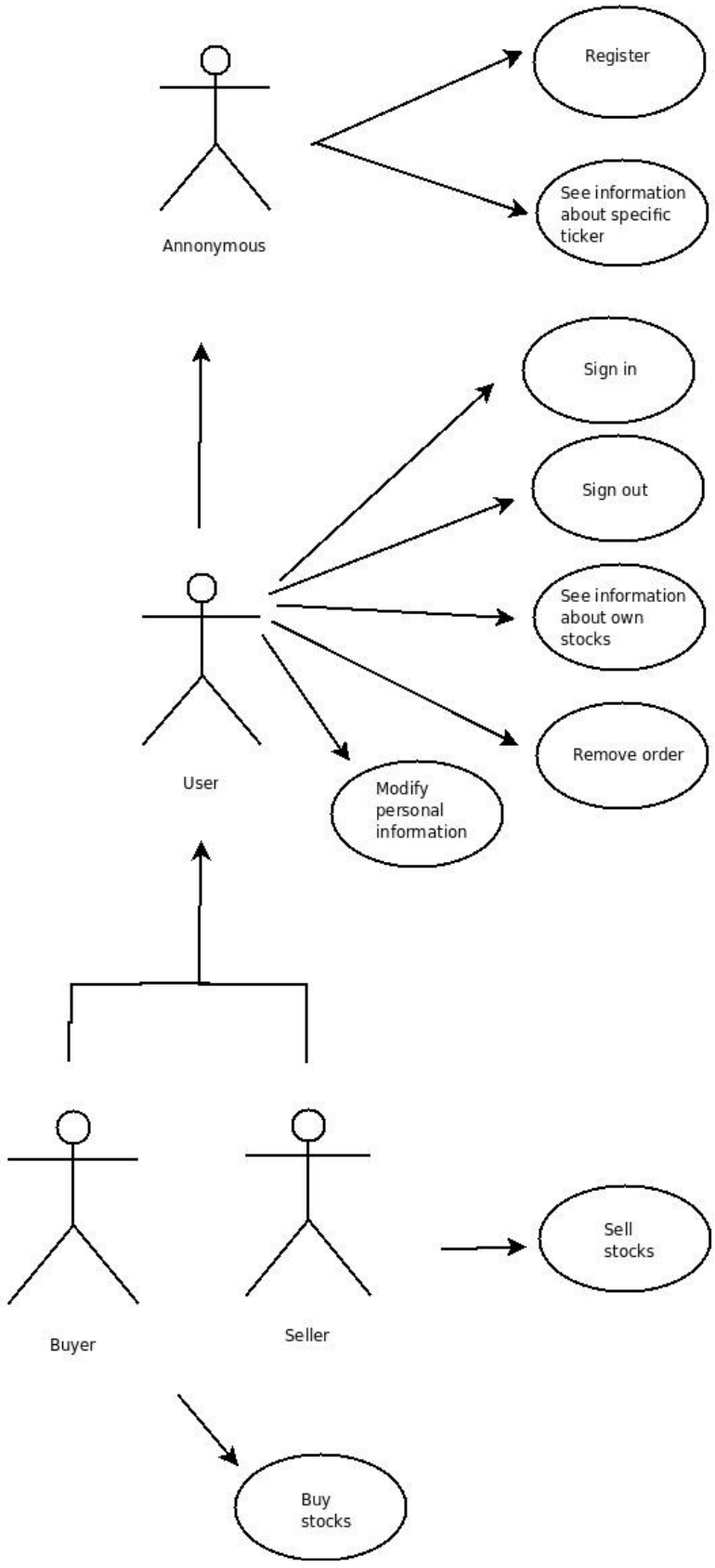
Buy stocks: buyers can buy their stocks. They have to detail the price and the volume user

2.3 Design

The design of this project is based in the UML diagram Case Uses and entity relationship model (ERM). Moreover, I will explain the structure of the directories in the application, the interface and I will describe how the project works with examples.

2.4 Use Case Diagram

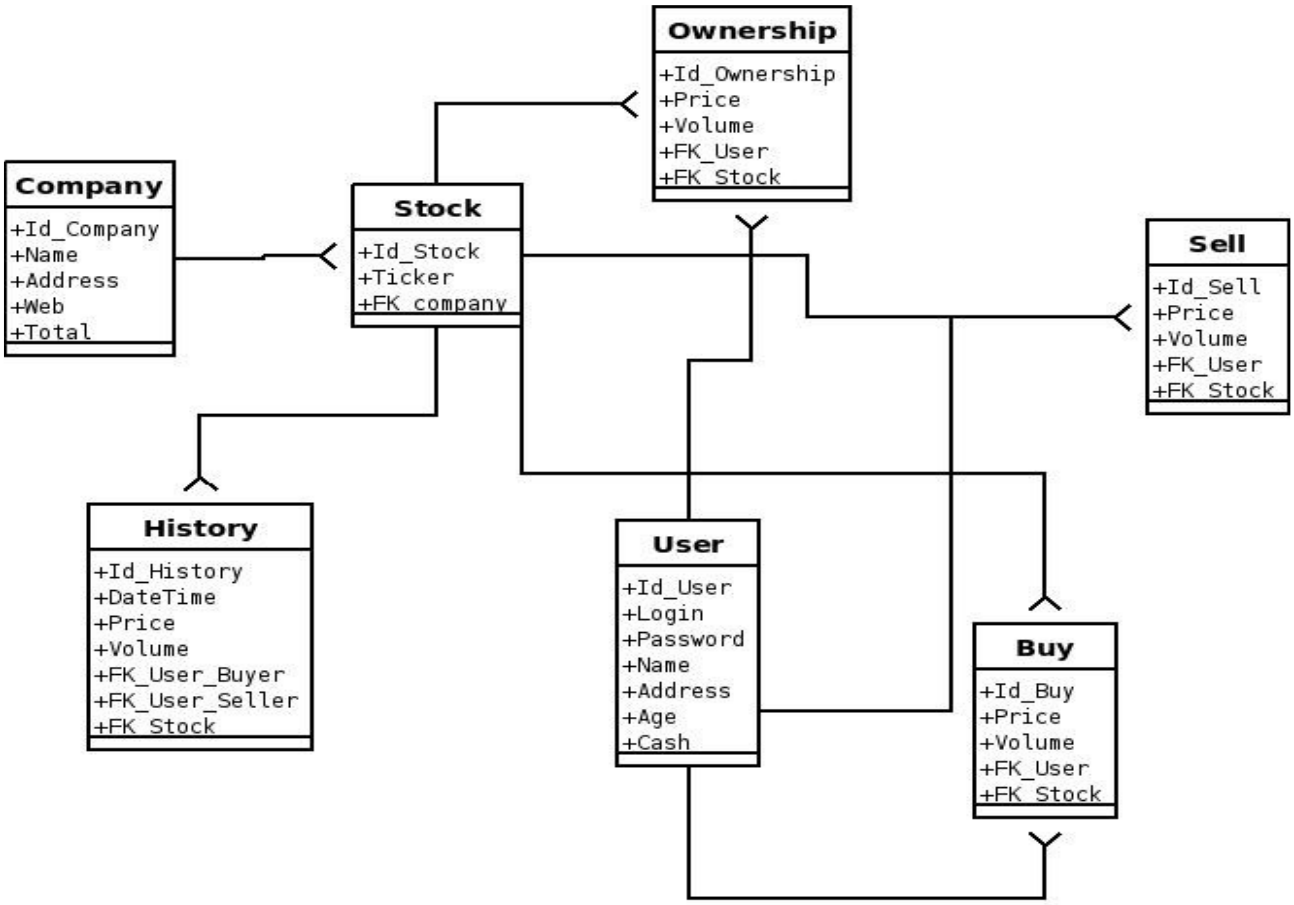
The use case diagram is one of the diagrams of in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be represented. In this project the case uses diagram is based in the several function of the application like register, sign in and sign out, see details, modify data or sell and buy stocks.


Register
See information about specific ticker
Annonymous
Sign in
Sign out
See information about own stocks
Remove order
Modify personal information
User
Sell stocks
Buyer
Seller
Buy stocks

2.5 Entity- Relational Diagram

An Entity-Relationship Model (ERM) is an abstract and conceptual representation of data. Entity-relationship modelling is a database modelling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database, and its requirements in a top-down fashion. In this project I did the Entity-Relationship Model with the relationship of the tables in the Database.

You can see 7 tables, “Ownership”, “Company”, “Stock”, “History”, and “User”, Buy”and“Sell. “Stock” has a foreign key of “Company”, “Ownership” has a foreign key of “Stock” and another foreign key of “User”, “Sell” has a foreign key of “User” and a foreign key of “Stock”, “Buy” has a foreign key of “User” and a foreign key of “Stock”, “History” has two foreign key of “User” and a foreign key of “Stock”.



In the following paragraph I will explain the tables of the Data Base:

- **Company**: this table contains the information of the companies, name address, web and the total of stocks that the company has.
- **Stock**: this table contains the name of the ticker in each stock.
- **User**: this table contains the information about the users, login, password, name, address, age and the cash that the user has.
- **Ownership**: this table links “User” with “Stock”. It contains the stocks of each user.
- **Sell**: this table links “User” with “Stock”. It contains the stocks that the user wants to sell.
- **Buy**: this table links “User” with “Stock”. It contains the stocks that the user wants to buy.
- **History**: this table links “User” with “Stock”. It contains all the transactions done in the trade market, it has the information about the buyer, seller and number of stocks sold, price, ticker and date.

3. Non-Functional Requirements

3.1 *Performance*

The system is expected to do all the operations of buying, selling, maintenance and updating of stock prices in real time in a reasonable amount of time. This can be considered to be efficient enough since price management has to be done on a large number of firms. The market virtualization has to be close to the real world and periodical update of real world prices is necessary.

3.2 *Reliability*

The application is expected to allow login and buying or selling of shares. Naturally we can expect errors in the virtual prices from the real prices, but by employing regular price update process, we can make the system more reliable.

3.3 Security

All the data is stored in a database and is not open to public use. It is not stored in a simple csv file which can be opened and read. Thus we can say that the classified data are kept sufficiently private and confidential. Access rights are provided based on the user privileges which is obtained from the login information.

3.4 Availability

The system will not have any issues with availability as long as the server is not overloaded. Our design keeps in mind the maximum expected number of concurrent users. Also, there is no reason to expect unavailability for longer durations since the client is given a chance to connect as soon as load decreases even at peak usage.

3.5 Portability

The application is independent of the resident operating system and needs a compatible browser for execution. It is expected that JavaScript and Cookie features are enabled on the default browser.