ECE 16:332:568

Software Engineering of Web Applications

WEB BASED STOCK FORECASTER

http://se2.peterjiang.me

By Group 4:

Yuwei Jiang

Chenfan Xiao

Cheng Chen

Xinyu Li

Jianing Xu

Instructor:

Shiyu Zhou

Submission Date:

04/28/2016

Contribution Breakdown: All members contributed equally

|  |  |
| --- | --- |
| Prediction Algorithm | Chenfan Xiao, Xinyu Li |
| Web Design | Yuwei Jiang, Cheng Chen, Jianing Xu |
| Report | Yuwei Jiang, Cheng Chen, Jianing Xu, Chenfan Xiao, Xinyu Li |

Table of contents

1. Problem statement and motivation

1.1 Problem statement

1.2 Motivation

2. Glossary of Terms

3. System requirements

3.1 System requirements

3.2 Requirement specification

4. Function requirements specification

4.1 Specifying The Discovered Requirements

4.2 Stakeholders

4.3 Use Case Description

4.3.1 Use cases

4.3.2 Fully dressed description

4.4 Use case diagram

5. Programming Technologies

5.1 Front-end Technologies

5.1.1 JavaScript and Ajax

5.1.2 HTML5

5.1.3 PHP

5.1.4 PaaS

5.1.5 CDN

5.1.6 HTTP

5.1.7 Google Visualization

5.2 Back-end Technologies

5.2.1 Matlab

5.2.2 Mysql

6. Algorithms and Structure

6.1 Bayesian Curve Fitting

6.2 Artificial Neural Network

6.3 Exponential Moving Average (EMA)

6.4 Granville’s Strategy

6.5 Relative Strength Index(RSI)

6.6 Support Vector Machine Machine-Regression(SVR)

7. User interface design and implementation

7.1 Preliminary design

8. Future work

9. Reference

## 1 Problem Statement and Motivation

### 1.1 Problem Statement

Since the stock market came to use, the businessmen became capable of buying shares of different companies and the investors are also available of purchasing the shares in seek of making a huge profits. Although the market can also provide these functions, nowadays it has a tendency that the use of seeking gain for normal investors is exceeding the use of seeking capital for businessmen. Since it is an age of high taxes and low wages, the trading stocks are therefore stand out as a viable and even effortless method of becoming rich in a very short time.  Unfortunately, the fact is that the same volatility in the market can be used for profits, but at the same time, it could bring huge loss as well, even for some knowledgeable investors occasionally.

The following reasons would result such loses: the unawareness of the company’s business model, the difficulty of stock history visualization,  the late acknowledgement of current company economic events and the most important is the lack of capability of stock’s future performance prediction.

First, three basic acknowledgement about the company are required: what does this company do, how the company actually makes profits and the company’s fundamentals. The company’s fundamentals gives us an opportunity to have a deep understanding about the company’s financial operation, including the balance sheet, governmental filings, investor reports, etc. It is not surprising that the individual investors are unaware of the company they are trying to invest in or they have a preconception of how well the company is doing and what they are best known for. The problem with this is, how will investors be able to know when to buy or sell a stock if they don't know exactly how the company is making money and if their assets will still be popular with consumers in the future? Investors are also recommend to know the financial stability of the company.

Second, the visualization of the stock history might be not easy to understand. Therefore, our predictor implements a linear regression graph of time versus price to serve as an easy way for the investors to understand. Since with the help of such graph, investors can view the rise and fall of the stock price. The prediction will be an extension from the current stock value on the graph and will help investors see how the stock will perform relative to the current and past values.

The biggest problem is the lack of capability of predicting.  The stock market is changing from raising and falling all the time, which means the stock would go very high today and go down significantly next day. Neither the long-term nor short-term prediction is one hundred percent precise, but it is possible to minimize the mismatch. The three prediction methods mentioned before are all based on past data and stock performance and are highly mathematical. However, after retrieving the mathematical and logical research, when can you take the volatility and emotional aspects of buying and selling stocks into consideration? Our predictor could help the investors to make decisions by a generated forecast. On the other hand, our website will recommend what kind of prediction model we use to generate the prediction. This is a method of accountability and investors do not need to question where such numbers came from. By utilizing these predictions, it is up to the investors to choose whether they should buy, sell, or hold the stock.

### 1.2 Motivation

     For every day trading, these invisible risks are a main barrier which keep the new, inexperienced traders away from the market. The investor’s ranges from a young college student who wants to feed his own, to the old who might be saving money for his retirement. But certain percentage of these investors leave the market due to the insufficient time, founds and especially knowledge about the invisible world.

      We want to make such unknown risks predictable even to those people who are foreign to the market or stock field, not only to the experts or professional investors. But in the other hand, we could learn from the long-term investors who utilize the entailed analyzing of the financial performance of many companies and choose the stocks that has the best growth potential. Since it is a long-term strategy, it may has little growth at the beginning and time consuming, but it is still superior to human errors.

       Many investment advisors suggest to diverse the investment portfolio to minimize the risk of loss. As the old saying goes “leaving all of the eggs in one basket is not vise”. Thus, we should increase the diversity of investment. But how to implement such strategy? Basically there are 3 methods: fundamental analysis, technical analysis and technological methods. All these methods are based on the analysis of the past data which makes them extremely mathematical. However, after receiving all the mathematical data, when and how can you make a decision whether to buy or sell the stock? Our project gives the solution to this problem by a well-designed stock predictor for individual investors. Our predictor is capable of searching the internet to collect various stock data and generate a forecast automatically.

## 

## 2 Glossary of Terms



|  |  |
| --- | --- |
| One year High/Low | Highest and lowest prices of a stock that has been recorded at during the year. |
| Buy and Hold | The practice of buying a good for a long term rather than trying to turn a profit quickly. |
| Average Volume | The amount of stocks that traded over duration of time. |
| Closing Price | The price a particular stock closes at on a given trading day. |
| Dividend | A distribution of a portion of a company's earnings, decided by the board of directors, to a class of its shareholders. The dividend is most often quoted in terms of the dollar amount each share receives (dividends per share). It can also be quoted in terms of a percent of the current market price, referred to as dividend yield. |
| Earnings per Share (EPS) | The portion of a company's profit allocated to each outstanding share of common stock. Earnings per share serves as an indicator of a company's profitability. It  is calculated as (Net Income -Dividends on Preferred Stock) / divided by the Average Outstanding Shares |
| Forecast | A prediction of the future based on special knowledge |
| Fundamental Trading | Fundamentalists trade companies based on fundamental analysis, which examines things like corporate events such as actual or anticipated earnings reports, stock splits, reorganizations or acquisitions. |
| Individual | An investor who purchases relatively small lots of stocks for his own |
| Investor | Portfolio. |
| Institutional Investor | An entity with large amounts to invest, such as investment companies, brokerages, and investment banks. Institutional investors are covered by fewer protective regulations because it is assumed that they are more knowledgeable and better able to protect themselves. Institutional investors are usually a group of people, rather than individuals. |
| Market Trend | The tendency of a financial market to move in a particular direction over time. Bull market refers to an upward trend, and a bear market refers to a downward trend. |
| Moving Average Prediction Model | A way to predict the future price of stocks based on the assumption of constant underlying mean of given prices |
| Opening Price | The price a stock starts off at a particular trading day. |
| Stock | A type of security that signifies ownership in a corporation and represents a claim on part of the corporation's Assets and earnings. |
| Stock Market | The marketplace for buyers and sellers of stocks. |
| Stock Symbol | A unique set of symbols that represent a particular company. Ex: GOOG is the stock symbol of Google. |
| Ticker | See Stock Symbol |
| Trading Day | The duration of time the stock market is open for buying and selling stock. Ex: For the New York Stock Exchange trading day is 9:30 AM Eastern Time to 4:00 PM Eastern Time, trading days never occur on weekends. |

## 3 System requirements



### 3.1 System Requirements

|  |  |  |
| --- | --- | --- |
| Identifier | Priority | Requirement |
| REQ1 | 5 | The system should gather a set of company’s price history into database. |
| REQ2 | 5 | The system should operate the predictions on the future trend and price of the price with the prediction algorithm based on history data and store them in the database. |
| REQ3 | 5 | The system should allow users to search for a company stock and display history, real time data/graph and prediction suggestion for users. |
| REQ4 | 4 | The system should allow users to choose stocks to track |
| REQ5 | 4 | The system should allow administrator to add/remove stocks in the stock database |
| REQ6 | 3 | The system should allow users to set up accounts(email,password) |
| REQ7 | 2 | The system should provide an index page showing a list of all the stocks that it can provide along with the latest price. |
| REQ 8 | 1 | The system should provide relative information about the stock for users. |

### 3.2 Requirement specification

|  |  |
| --- | --- |
| REQ1a | The system should have two databases. One is used to save the history stock price for past 1 year. The other is used to save the real time data of every minutes. |
| REQ1b | The database should be able to save at least 10 stocks. |
| REQ1c | The database should support extension to save informations of price predictions. |
| REQ1d | The database should support extensions to save user account |
| REQ1e | The database should support allow administrator tw add and delete stocks in database. |
| REQ2a | The prediction should be able to give both long term and short term predictions that can help making stock investment decisions. The long term prediction give prediction of tomorrow based on the data from past 14 days. The short term prediction gives the prediction of next minute based on the short term history. |
| REQ2b | The short term prediction is based on bayesian which gives the price of next minutes |
| REQ2c | The long term prediction is based on RSI, SVM and EMA. RSI and SVM can provide a price prediction. The EMA give a suggestion between sell, hold and buy. |
| REQ3a | The search function should give search result based on the keywords input by the users. The result should be listed according to the relevance with the key words. |

|  |  |
| --- | --- |
| REQ3b | The stock info page should be displayed into two parts. The relevant information should be displayed correctly. The stock name, current price, update time and the graph of the price are shown on the left part of the graph. The stock graph gives the history price line chart of past month and the real time data of the past day. The other part of the graph gives the prediction information along with some statistical history information. The statistical information includes the maximum price in the last 10 days, the minimum price over the last year and the average price in the past year. |
| REQ4a | The system keep the stock list each user choose to track. User can check the price whenever they need it. |
| REQ6a | The system should be able to display the account info about uuid given to the user, the username, the email for registration and the regdate. |

## 

## 4 Function Requirements Specification

### 4.1 Specifying The Discovered Requirements

Architectural style: Web-based – The programs will run on a Web Browser and each user’s portfolio will be stored; their portfolio includes a list of all the stock tickers, information for those stocks, and corresponding graphs for those stocks. All information could be stored in a common database or on an Excel Spreadsheet. There is no form of client communication, but if we feel as if it could be an added benefit to the website, it can be implemented in the future.

### 4.2 Stakeholders

Unlike institutional investors who have a team at their disposal, individual investors don't have such a luxury and will benefit from our web application. These investors have a broad range of assets (usually smaller than institutional investors) and can either invest in the short or long term. They usually purchases small lots of stock (1-5) depending on the value of their assets. The individual investor is your standard person who does not use investing as their primary source of income.

### 

### 4.3 Use Case Description

#### 4.3.1 Use Cases

UC-1: login- Allow users to access their account and view information about the stocks they tracked.

UC-2: Search- Users can search for a stock by keywords and view the relative information about the stock

UC-3: Track- Users can add and delete stocks to their track list.

UC-4: Register- Visitors can register to be a user.

UC-5: Check stock info-visitors and users can check the stock info covering price,graph, time and prediction.

UC-6: Suggest- a user to request for the prediction info about a stock.

UC-7: Add/Delete Stock- administers can change the list of stocks that can be stored in the database.

#### 4.3.2 Fully dressed description

|  |
| --- |
| Use case -login |
| Initiating Actors:  Actor’s Goal: To login in the account and browsers the tracked stocks in the database.  Participating Actors: Database  Preconditions: The user has already registered an account.(UC-4)  Success End Conditions: User login the account and able to browse the information he/she needs.  Failure End Conditions: Login failed because of wrong passwords or have not registered.  Flow of Events for Main Success Scenario:   1. User click login and web for user to login shows up; 2. User type in the email address and password and click login; 3. User is notified a login success and redirected to the index page |

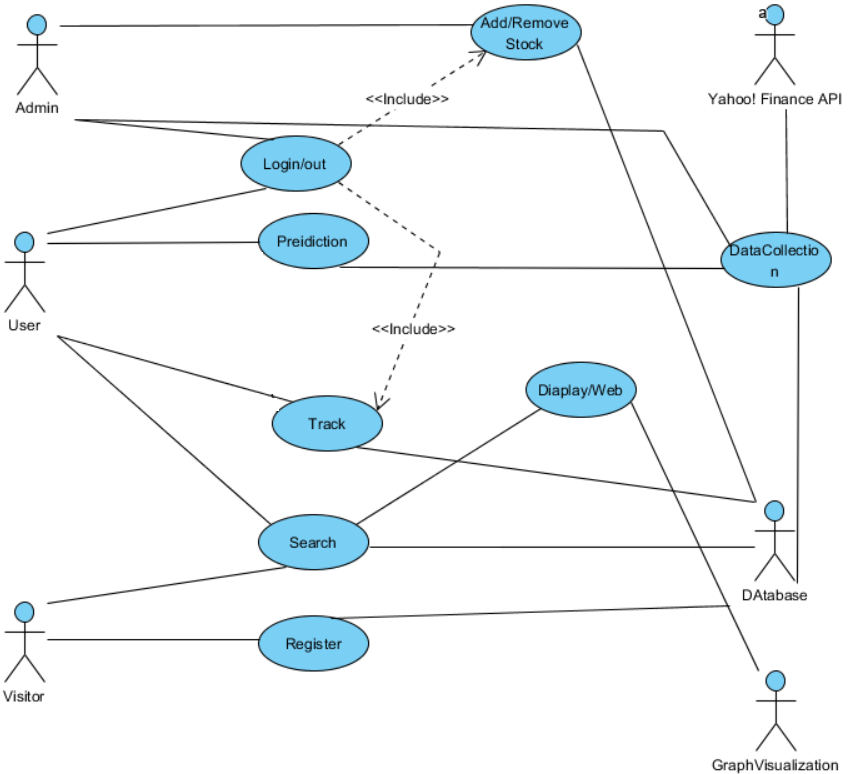
|  |
| --- |
| Use case -Search |
| Initiating Actors:Visitors, Users ,administrators (in the following users represents all three actors.  Actor’s Goal: To search for a stock user is interested in from the database.  Participating Actors: Database  Preconditions: none  Success End Conditions: The user is provided with a list of stock ranked by relevance to the key words.  Failure End Conditions: The search does not provide any result because the key does not match any stock in the database  Flow of Events for Main Success Scenario:   1. User click on the search button in the navigation bar 2. User type several key letters into the search bar 3. A list of possible stocks are shown below the search bar 4. User click on one of the stocks and is redirected to the stock info page. 5. If the result does not satisfy user, user can change the key words by repeating from process one. |

|  |
| --- |
| Use case -Track |
| Initiating Actors:Users  Actor’s Goal:To add a list of stock in the user’s account such that users can check the status of the stock without checking search or finding them elsewhere.  Participating Actors: Database  Preconditions: The user has already registered an account.(UC-4) and already logged in(UC-1)  Success End Conditions: a stock is added to the list in the user account  Failure End Conditions:none  Flow of Events for Main Success Scenario:   1. On the stock info page user click the add button to add the stock he or she wants to follow. The stock is hereby added to the account track list. 2. User click the stocklist button in the upright. 3. The list of tracked stock is shown there. Users can check the info of each stock by clicking on them   Alternative Scenario:   1. User click the remove button on stock info page. The stock is then removed from the tracking list. |

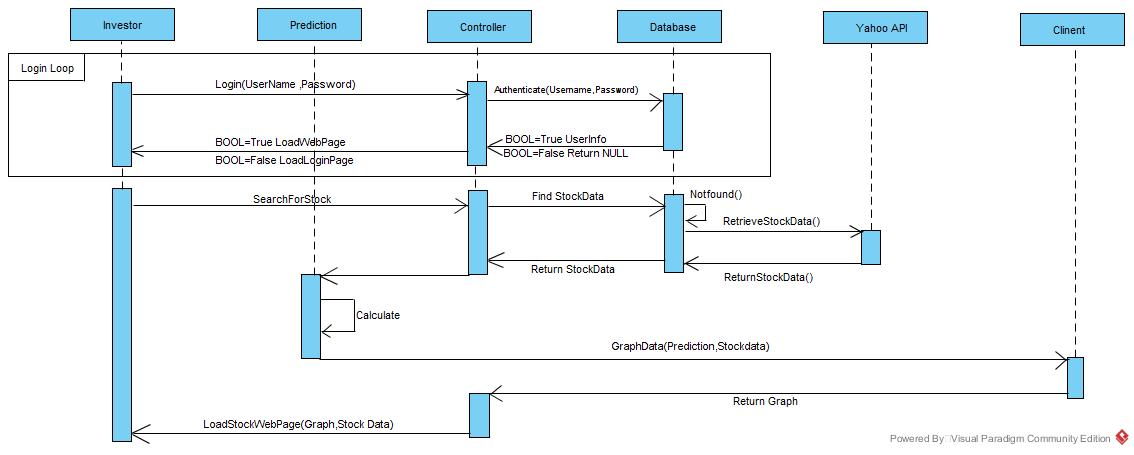
|  |
| --- |
| Use case -Check stock info |
| Initiating Actors:All(Administrator, users and visitors)  Actor’s Goal: See a detailed info about the a stock. The information is discussed in the REQ3  Participating Actors: Database,Graph visualization  Preconditions: The stock has to be inside the database.  Success End Conditions: The info is shown on a stock info page  Flow of Events for Main Success Scenario:   1. User is directed to the info page from different path(search, tracked list, Index page) 2. Relevant data about the the stock is extracted from the database, the graph visualization gives the line chart of the stock, The prediction is data, latest price info is also displayed. 3. User can choose to browse the real time graph( data of the past 24 hours.) or history graph. User can also choose see the history data and set the start time and end time of the graph. Then graph visualization then provide the graph for the user. |

|  |
| --- |
| Use case -Add/delete stock |
| Initiating Actors:Administrator  Actor’s Goal: Add stocks to the list of stocks within the database with history data  Participating Actors: Database  Preconditions: Administrators is logged in the system  Success of end condition: All of the chosen stocks are added to the database. The system then put history price from the source into the database.  Flow of Events for Main Success Scenario:   1. Administrators enter the symbol of the stock that he/she wishes to add to the system in the administrator's management page. 2. The administrators click the add stock button 3. The system check if the stock is already in the database. 4. If the stock is already in the database, then addition failed. 5. Else the system retrieve the relevant data from Yahoo! Stock API and save them into the database. 6. System notified the administrator that the stock is added successfully. 7. Administrator can choose to add more stock or view the stock in the system. |

### 4.4 Use case diagram

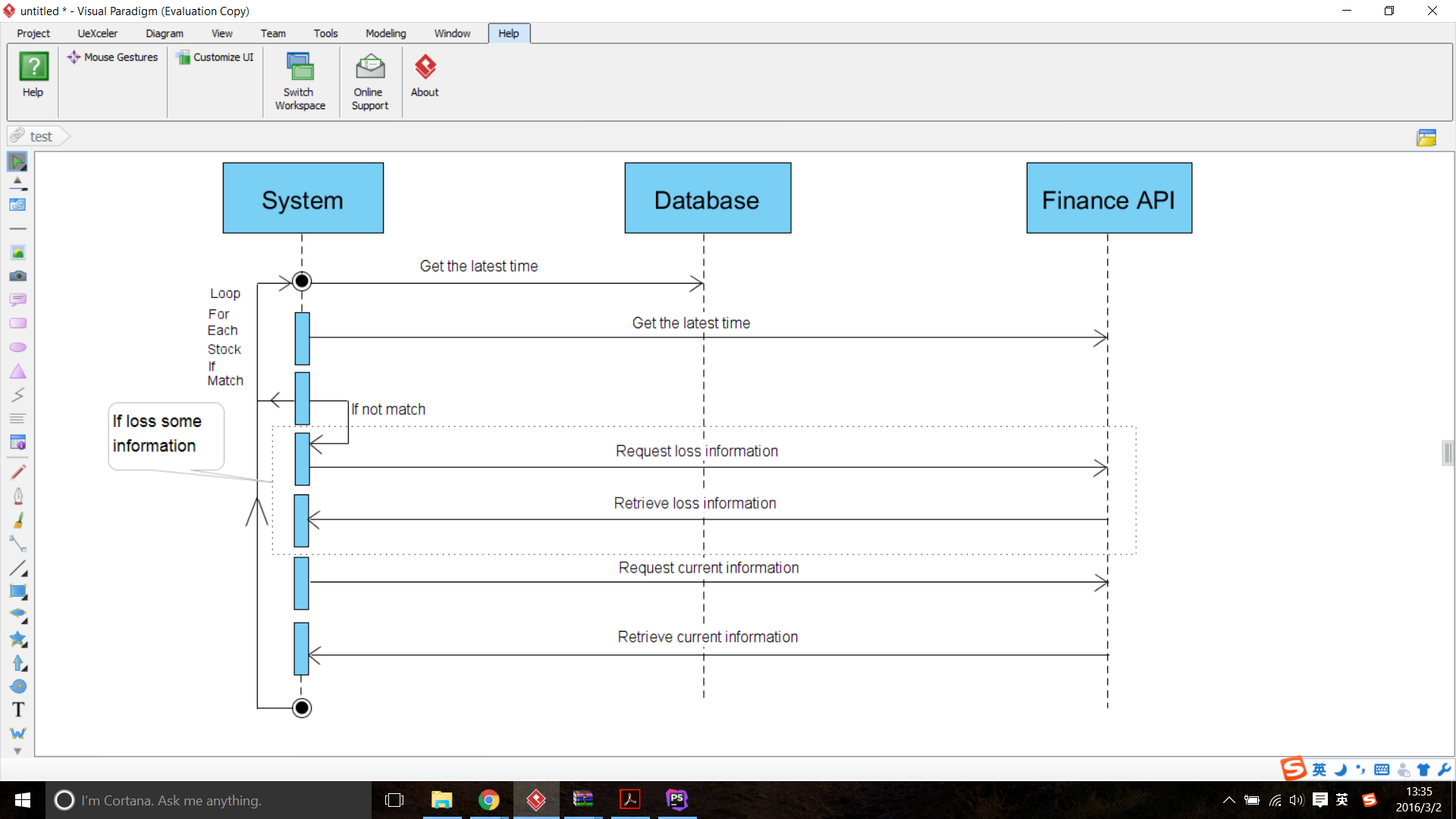


### 4.5 System sequence diagram



### 

### 4.6 Database Diagram



## 5 Programming Technologies



Technical details will be discussed in this chapter. It will be divided into two main parts: front-end and back-end.

### 5.1 Front-end Technologies

#### 5.1.1 JavaScript and Ajax

Many modern approaches were used for our project development. We used JavaScript and Ajax for most interactive content in HTML pages. For example, the search recommendation division used Ajax so that the search results can be shown on visitors’ browsers without redirecting to search result page. This would highly improve the experience of users and they can choose the stock they want without input the name or symbol completely. JavaScript is used for most form. This would help to create an interactive content where users can send request as their wishes to the server but get the content neglecting their demands. For example, they can add or remove stock from their personal stock list after log in, so they do not need to search for stocks they want to track each time. They can also request data and charts between any time interval as they may want.

Several technologies derived from JavaScript are also used including JSON and jQuery. These technologies simplify the development and maintenance of the website. JSON is a language-independent data format. It derives from JavaScript, but as of 2016, code to generate and parse JSON-format data is available in many programming languages. It provides the extendibility for further application might run on the system. jQuery is a cross-platform JavaScript library designed to simplify the client-side scripting of HTML. jQuery is the most popular JavaScript library in use today, with installation on 65% of the top 10 million highest-trafficked sites on the Web.

#### 5.1.2 HTML5

Besides, the design of our pages meets HTML5 standards, which can enhance the user experience and make it more stable and compatible. We also used CSS3 for our UI design and every page is fluid-gridded, which means no matter what and how large the devices are and how much resolution the screens have, the pages will be displayed correctly and friendly on their devices. The users will never see a single page with too small front when they want to track their stocks with their cellphones on the way home.

#### 5.1.3 PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. Every page was generated by PHP, which means it can change dynamically when requested. And with PHP, a user management system and admin system were also implemented so each user can manage their own stock list and more functions can be extended conveniently. The admin system can add or remove stocks globally, both prediction algorithms and data collection function will be adjusted according to the system stocks lists stored also in database automatically. Once the system was deployed, everything maintenance will have friendly UI so the maintenance can be achieved by administrator without programming technologies.

#### 

#### 5.1.4 PaaS

The whole website is deployed on a cloud computing PaaS platform provided by RedHat Company with resources of Amazon. It can be extended to support millions of requests per second. Platform as a service (PaaS) is a cloud computing model that delivers applications over the Internet. In a PaaS model, a cloud provider delivers hardware and software tools -- usually those needed for application development -- to its users as a service. A PaaS provider hosts the hardware and software on its own infrastructure.

#### 5.1.5 CDN

To help visitors have faster access to the website, and to reduce the pressure of the server, CDN was used for our website. Here is the ping time from my computer to the original server provided by Amazon AWS:

|  |
| --- |
| PING ec2-54-85-48-36.compute-1.amazonaws.com (54.85.48.36): 56 data bytes  64 bytes from 54.85.48.36: icmp\_seq=0 ttl=42 time=23.087 ms  64 bytes from 54.85.48.36: icmp\_seq=1 ttl=42 time=22.981 ms  64 bytes from 54.85.48.36: icmp\_seq=2 ttl=42 time=10.726 ms  64 bytes from 54.85.48.36: icmp\_seq=3 ttl=42 time=9.327 ms  64 bytes from 54.85.48.36: icmp\_seq=4 ttl=42 time=10.692 ms  64 bytes from 54.85.48.36: icmp\_seq=5 ttl=42 time=9.534 ms  64 bytes from 54.85.48.36: icmp\_seq=6 ttl=42 time=10.733 ms  64 bytes from 54.85.48.36: icmp\_seq=7 ttl=42 time=11.720 ms  64 bytes from 54.85.48.36: icmp\_seq=8 ttl=42 time=12.662 ms  64 bytes from 54.85.48.36: icmp\_seq=9 ttl=42 time=11.102 ms  ^C  --- ec2-54-85-48-36.compute-1.amazonaws.com ping statistics ---  10 packets transmitted, 10 packets received, 0.0% packet loss  round-trip min/avg/max/stddev = 9.327/13.256/23.087/4.972 ms |

Where we can see the that the average ping time is 13.256 ms even though the server is physically located in East America. Here is the the result of ping time via CDN:

|  |
| --- |
| PING se2cdn.peterjiang.me (104.28.7.117): 56 data bytes  64 bytes from 104.28.7.117: icmp\_seq=0 ttl=53 time=9.910 ms  64 bytes from 104.28.7.117: icmp\_seq=1 ttl=53 time=9.631 ms  64 bytes from 104.28.7.117: icmp\_seq=2 ttl=53 time=10.458 ms  64 bytes from 104.28.7.117: icmp\_seq=3 ttl=53 time=8.728 ms  64 bytes from 104.28.7.117: icmp\_seq=4 ttl=53 time=9.780 ms  64 bytes from 104.28.7.117: icmp\_seq=5 ttl=53 time=9.832 ms  64 bytes from 104.28.7.117: icmp\_seq=6 ttl=53 time=15.039 ms  64 bytes from 104.28.7.117: icmp\_seq=7 ttl=53 time=8.891 ms  64 bytes from 104.28.7.117: icmp\_seq=8 ttl=53 time=8.360 ms  64 bytes from 104.28.7.117: icmp\_seq=9 ttl=53 time=9.488 ms  64 bytes from 104.28.7.117: icmp\_seq=10 ttl=53 time=8.335 ms  ^C  --- se2cdn.peterjiang.me ping statistics ---  11 packets transmitted, 11 packets received, 0.0% packet loss  round-trip min/avg/max/stddev = 8.335/9.859/15.039/1.763 ms |

The average ping time is just 9.859 ms which is dramatically dropped and the longest ping time is 15.039 ms, which is just slightly more than the average time without CDN.  The difference will be even more if the clients visit the website from a longer distance for example in the other side of the country or earth.

A content delivery network (CDN) is a system of distributed servers (network) that deliver webpages and other Web content to a user based on the geographic locations of the user, the origin of the webpage and a content delivery server.

#### 5.1.6 HTTPS

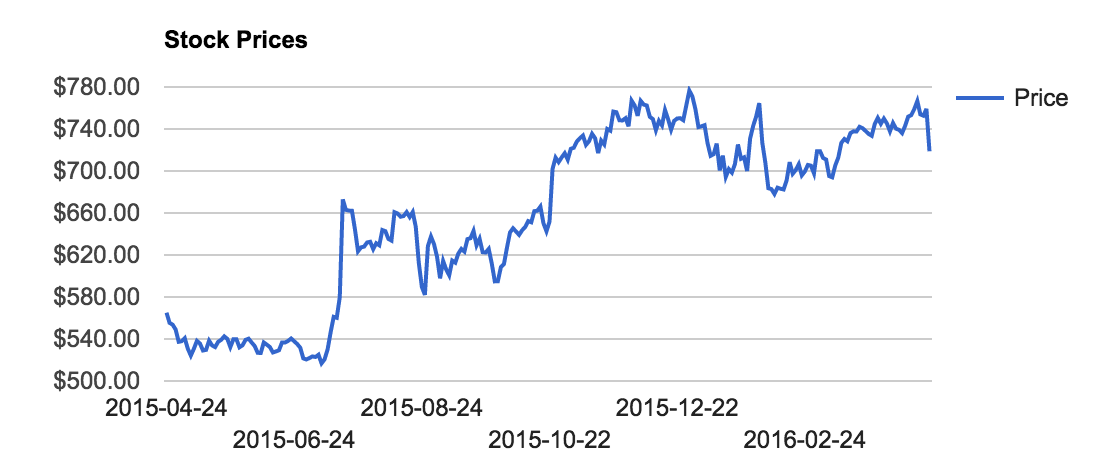
Another measure used on our website is HTTP over TLS, or more widely known as HTTPS. It ensures the sensitive information of users, such as emails, password and the stocks users tracking, will be protected properly and the communication between clients and servers will not be acquired without authority. The website is globally protected by HTTPS with SHA-256, which is also used by Wikipedia, Chase and TOR. Any content without proper encryption will not be transferred to and from users.

In its popular deployment on the internet, HTTPS provides authentication of the website and associated web server with which one is communicating, which protects against man-in-the-middle attacks. Additionally, it provides bidirectional encryption of communications between a client and server, which protects against eavesdropping and tampering with and/or forging the contents of the communication. In practice, this provides a reasonable guarantee that one is communicating with precisely the website that one intended to communicate with (as opposed to an impostor), as well as ensuring that the contents of communications between the user and site cannot be read or forged by any third party.

#### 

#### 5.1.7 Google Visualization

With the help with Google Visualization, which is a widely used and mature API provided by Google, plotting charts is efficiently more than ever. The chart as follows can be created without using much resource of the server.



### 5.2 Back-end Technologies

#### 5.2.1 MATLAB

Most of the algorithms were implemented with Matlab, which provides fast and stable performance for calculation. No matter how complicated the algorithms are, Matlab can calculate and write the result into database fast and stable. It ensures the core of the website functions well.

#### 5.2.2 MySQL

The most important part of the website is the database, which was implemented with MySQL. It ensures all the data can be stored and retrieved conveniently and all of the components and applications can be integrated together easily. MySQL is an open-source relational database management system (RDBMS); in July 2013, it was the world's second most widely used RDBMS, and the most widely used open-source client–server model RDBMS. The structure of the database used is discussed in other chapter.

## 6 Algorithms and Structure

**6.1 Bayesian Curve Fitting**

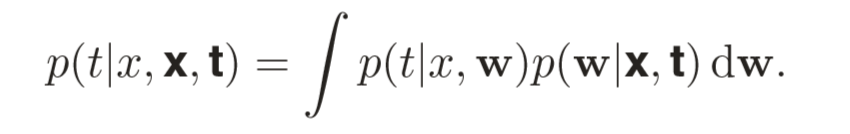
In [statistics](https://en.wikipedia.org/wiki/Statistics), Bayesian linear regression is an approach to [linear regression](https://en.wikipedia.org/wiki/Linear_regression) in which the statistical analysis is undertaken within the context of [Bayesian inference](https://en.wikipedia.org/wiki/Bayesian_inference). We will find the likelihood function and max it to fit curve.

The goal in the curve ﬁtting problem is to be able to make predictions for the target variable t given some new value of the input variable x on the basis of a set of training data comprising N input values x = (x1,...,xN)T and their corresponding target values t = (t1,...,tN)T. We can express our uncertainty over the value of the target variable using a probability distribution. For this purpose, we shall assume that, given the value of x, the corresponding value of t has a Gaussian distribution with a mean equal to the value y(x,w) of the polynomial curve. We have:

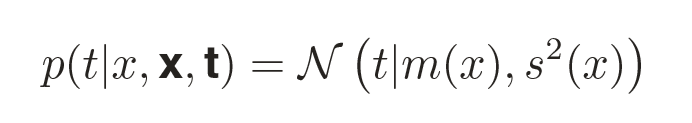


where, for consistency with the notation in later chapters, we have deﬁned a precision parameter β corresponding to the inverse variance of the distribution.

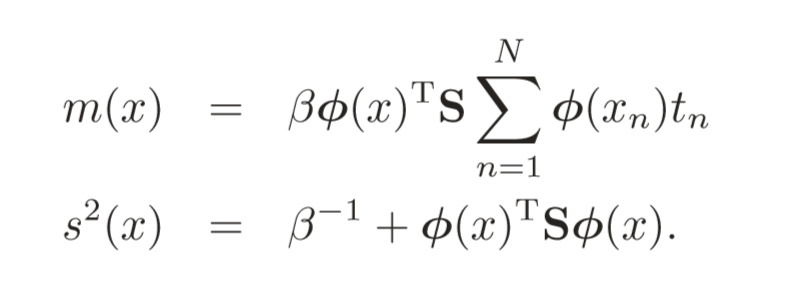
In the curve ﬁtting problem, we are given the training data x and t, along with a new test point x, and our goal is to predict the value of t. We therefore wish to evaluate the predictive distribution p(t|x,x,t). Here we shall assume that the parameters α and β are ﬁxed and known in advance (in later chapters we shall discuss how such parameters can be inferred from data in a Bayesian setting). A Bayesian treatment simply corresponds to a consistent application of the sum and product rules of probability, which allow the predictive distribution to be written in the form



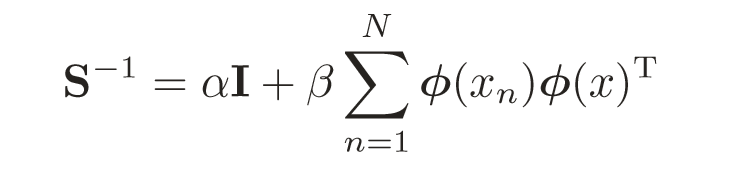
Here p(t|x,w) is given above, and we have omitted the dependence on α and β to simplify the notation. Here p(w|x,t) is the posterior distribution over parameters. We shall see that, for problems such as the curve-ﬁtting example, this posterior distribution is a Gaussian and can be evaluated analytically. Similarly, the integration in above can also be performed analytically with the result that the predictive distribution is given by a Gaussian of the form



where the mean and variance are given by



Here the matrix S is given by



where I is the unit matrix, and we have deﬁned the vector φ(x) with elements φi(x) = xi for i = 0,...,M.

We apply Bayesian Curve Fitting in short term prediction.

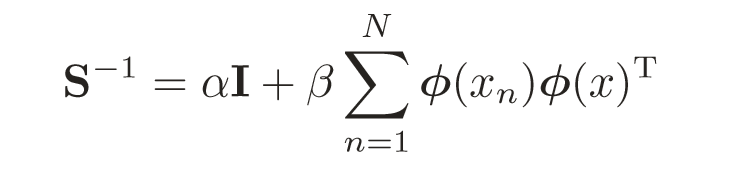
Basically, you need to get the training data to fit the curve, so the first step is choose appropriate data to do the prediction.

We choose the latest 10 real-time prices of one stock. And assuming that every price has an index from 1 to 10. So we can get a set of training data comprising N input values x = (x1,...,xN)T and their corresponding target values t = (t1,...,tN)T.

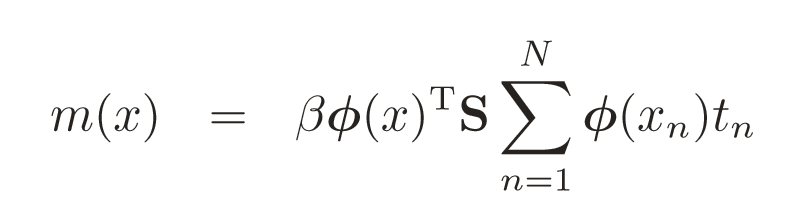
Then, we can increase the index to get the new value of the input variable x.

As for the parameters α and β, we can assume that the values are fixed. We set α=0.0047 and β= 0.225.

After these steps, we can calculate



For the stock prediction, we only care about the price which will has the greatest probability. So we only need to calculate the m(x):



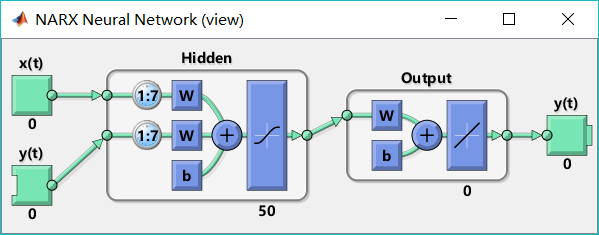
### 6.2 Artificial Neural Network

Artificial neural networks (ANNs) are a family of models inspired by [biological neural networks](https://en.wikipedia.org/wiki/Biological_neural_network) (the [central nervous systems](https://en.wikipedia.org/wiki/Central_nervous_system) of animals, in particular the [brain](https://en.wikipedia.org/wiki/Brain)) and are used to estimate or [approximate](https://en.wikipedia.org/wiki/Universal_approximation_theorem) [functions](https://en.wikipedia.org/wiki/Function_(mathematics)) that can depend on a large number of [inputs](https://en.wikipedia.org/wiki/Argument_of_a_function) and are generally unknown. Artificial neural networks are generally presented as systems of interconnected "[neurons](https://en.wikipedia.org/wiki/Artificial_neuron)" which exchange messages between each other. The connections have [numeric weights](https://en.wikipedia.org/wiki/Weighting) that can be tuned based on experience, making neural nets adaptive to inputs and capable of learning.

These networks are excellent for designing the behavior of more complicated structures because of their ability to learn. A major advantage in using this method for what we seek to accomplish is that they can be used to model a system of events, stock trends, that are totally unknown and how it will react to distortion and interference or in our case the imperfections of the stock market.

We use MATLAB neural network toolbox to generate the ANN. This means through the use of this software we can adapt the information we know about stocks to their system that is already set up to handle neural networks.

In order to use toolbox of neural network in MATLAB, we must prepare data to train the network. This data is imported from database which we fetch from stock website. And we will redefine this data to fit the toolbox.



We use the whole one past year’s close price as the train data. The date will be redefined as a number as input, and the close price can be target output. Once the data is ready, the neutral network can be trained.

We set the delay of input is 7, and the number of hidden layer is 50.

After the networks has been set, we will redefine the next days as numbers, and input these numbers to the network. The network will calculate the predictive value of current input under the training weights. So, we can get the predictive results of the next days.

### 6.3 Exponential Moving Average (EMA)

The exponential moving average (EMA) is a variant of moving averages that looks and acts like any other moving average. If you look at a chart with a simple moving average (SMA) and an exponential moving average, you won’t be able to differentiate at first glance. However, under the hood, there are key differences regarding how the SMA and EMA are calculated.

Let’s say you are trading the daily chart and looking at last month’s [price action](http://tradingsim.com/blog/price-action-trading-strategies/). Would you agree that analyzing last week’s price action would offer a better understanding of how the market is behaving today, and today’s price action would likely dictate tomorrow’s price action?

Since recent price data plays a more relevant compared to older price data in shaping the market, it is common sense that you should give more weight to recent data.

The exponential moving average (EMA) applies this very notion that traders should pay more attention to the recent price action compared to the old ones. Although most modern charting packages automatically calculates and plots the various types of moving averages on a price chart, it is always  a good idea that you know how they are calculated as it helps to increase your understanding regarding why moving averages behave differently.

Basically, you need to go through three steps to calculate the exponential moving average for trading any instrument.

First, we need to figure out the simple moving average (SMA). If we want to calculate the SMA of the last 10 [days](http://tradingsim.com/blog/trading-days-in-a-year/), we simply sum up the values of the last 10 closing prices and divide it by 10 to get the SMA.

Once we have the SMA, next we need to figure out the weighting multiplier for the number of periods we want to calculate for the EMA.

The weighting multiplier is calculated with the following formula:

EMA(current) = ( (Price(current) – EMA(prev) ) x Multiplier) + EMA(prev)

You should always remember that the number of periods will have a profound effect on the weighting multiplier as it places greater importance to the most recent price action.

As we are using 10 days in this exponential moving average example, the weighting multiplier would be calculated like this:

(2 / (Time periods + 1) ) = (2 / (10 + 1) ) = 0.1818 (18.18%)

Finally, once you have calculated the SMA and  weighting multiplier values, you can easily calculate the EMA with the following calculation:

(Closing price - EMA(previous day)) x multiplier + EMA(previous day)

### 6.4 Granville’s Strategy

Once we calculate the EMA, we can make a trading decision by using Granville’s strategy.

(1)If the 200 day average line flattens out following a previous decline, or is advancing, and the price of the stock penetrates that average line on the upside, this comprises a major buying signal.

(2) If the price of the stock falls below the 200 day moving average price line while the average line is still rising, this also is considered to be a buying opportunity.

(3)If the stock price is above the advancing 200-day line and is declining toward that line, fails to go through and starts to turn up again, this is a buying signal.

(4) If the stock price falls too fast under the declining 200-day average line, it is entitled to an advance back toward the average line and the stock can be bought for this short-term technical rise.

(5) If the 200-day average line flattens out following a previous rise, or is declining, and the price of the stock penetrates that line on the  downside, this comprises a major selling signal.

(6) If the price of the stock rises above the 200 day moving average price line while the average line is still falling, this also is considered to be a selling opportunity.

(7) If the stock price is below the falling 200-day line, and is advancing toward that line, fails to go through and starts to turn down again, this is a selling signal.

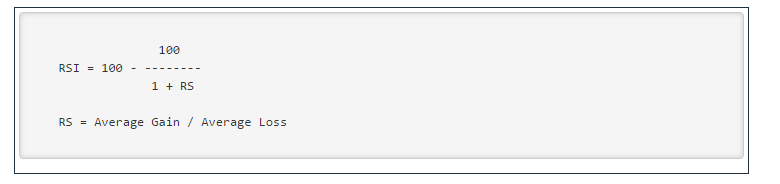
(8) If the stock price advances too fast above the advancing 200 day average line, it is entitled to a reaction back toward the average line and the stock can be sold for this short-term technical reaction.

### 6.5 Relative Strength Index(RSI)

Developed J. Welles Wilder, the Relative Strength Index (RSI) is a momentum oscillator that measures the speed and change of price movements. RSI oscillates between zero and 100. Traditionally, and according to Wilder, RSI is considered overbought when above 70 and oversold when below 30. Signals can also be generated by looking for divergences, failure swings and centerline crossovers. RSI can also be used to identify the general trend.

RSI is an extremely popular [momentum indicator](http://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:introduction_to_technical_indicators_and_oscillators) that has been featured in a number of articles, interviews and books over the years. In particular, Constance Brown's book, Technical Analysis for the Trading Professional, features the concept of bull market and bear market ranges for RSI. Andrew Cardwell, Brown's RSI mentor, introduced positive and negative reversals for RSI. In addition, Cardwell turned the notion of divergence, literally and figuratively, on its head.

The calculation formation is:

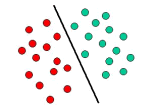


RSI is a versatile momentum oscillator that has stood the test of time. Despite changes in [volatility](http://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:standard_deviation_volatility) and the markets over the years, RSI remains as relevant now as it was in Wilder's days. While Wilder's original interpretations are useful to understanding the indicator, the work of Brown and Cardwell takes RSI interpretation to a new level. Adjusting to this level takes some rethinking on the part of the traditionally schooled chartists. Wilder considers overbought conditions ripe for a reversal, but overbought can also be a sign of strength. Bearish divergences still produce some good sell signals, but chartists must be careful in strong trends when bearish divergences are actually normal. Even though the concept of positive and negative reversals may seem to undermine Wilder's interpretation, the logic makes sense and Wilder would hardly dismiss the value of putting more emphasis on price action. Positive and negative reversals put price action of the underlying security first and the indicator second, which is the way it should be. Bearish and bullish divergences place the indicator first and price action second. By putting more emphasis on price action, the concept of positive and negative reversals challenges our thinking towards momentum oscillators.

### 6.6 Support Vector Machine Machine-Regression(SVR)

In [machine learning](http://learning), support vector machines (SVMs, also support vector networks[[1]](https://en.wikipedia.org/wiki/Support_vector_machine#cite_note-CorinnaCortes-1)) are [supervised learning](http://learning) models with associated learning [algorithms](https://en.wikipedia.org/wiki/Algorithm) that analyze data used for [classification](http://classification) and [regression analysis](http://analysis). Given a set of training examples, each marked for belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other, making it a non-[probabilistic](http://classification) [binary](http://classifier) [linear classifier](http://classifier). An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.

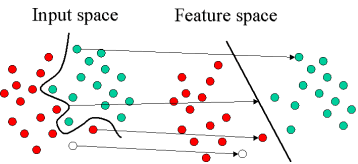
Support Vector Machines are based on the concept of decision planes that define decision boundaries. A decision plane is one that separates between a set of objects having different class memberships. A schematic example is shown in the illustration below. In this example, the objects belong either to class GREEN or RED. The separating line defines a boundary on the right side of which all objects are GREEN and to the left of which all objects are RED. Any new object (white circle) falling to the right is labeled, i.e., classified, as GREEN (or classified as RED should it fall to the left of the separating line).



The above is a classic example of a linear classifier, i.e., a classifier that separates a set of objects into their respective groups (GREEN and RED in this case) with a line. Most classification tasks, however, are not that simple, and often more complex structures are needed in order to make an optimal separation, i.e., correctly classify new objects (test cases) on the basis of the examples that are available (train cases). This situation is depicted in the illustration below. Compared to the previous schematic, it is clear that a full separation of the GREEN and RED objects would require a curve (which is more complex than a line). Classification tasks based on drawing separating lines to distinguish between objects of different class memberships are known as hyperplane classifiers. Support Vector Machines are particularly suited to handle such tasks.



he illustration below shows the basic idea behind Support Vector Machines. Here we see the original objects (left side of the schematic) mapped, i.e., rearranged, using a set of mathematical functions, known as kernels. The process of rearranging the objects is known as mapping (transformation). Note that in this new setting, the mapped objects (right side of the schematic) is linearly separable and, thus, instead of constructing the complex curve (left schematic), all we have to do is to find an optimal line that can separate the GREEN and the RED objects.



Support Vector Machine can also be used as a regression method, maintaining all the main features that characterize the algorithm (maximal margin). The Support Vector Regression (SVR) uses the same principles as the SVM for classification, with only a few minor differences. First of all, because output is a real number it becomes very difficult to predict the information at hand, which has infinite possibilities. In the case of regression, a margin of tolerance (epsilon) is set in approximation to the SVM which would have already requested from the problem. But besides this fact, there is also a more complicated reason, the algorithm is more complicated therefore to be taken in consideration. However, the main idea is always the same: to minimize error, individualizing the hyperplane which maximizes the margin, keeping in mind that part of the error is tolerated.

## 

## 

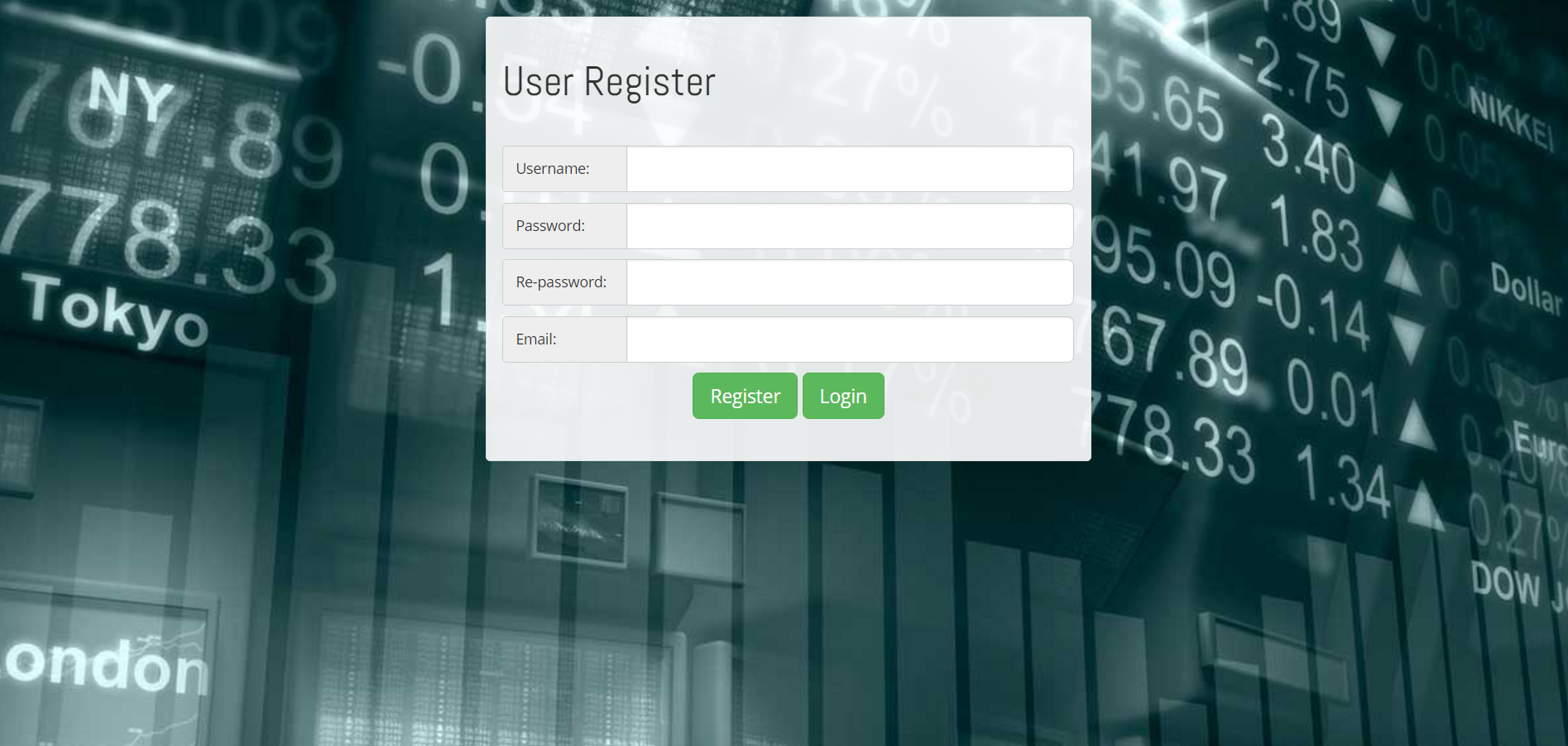
## 7 User interface design and implementation



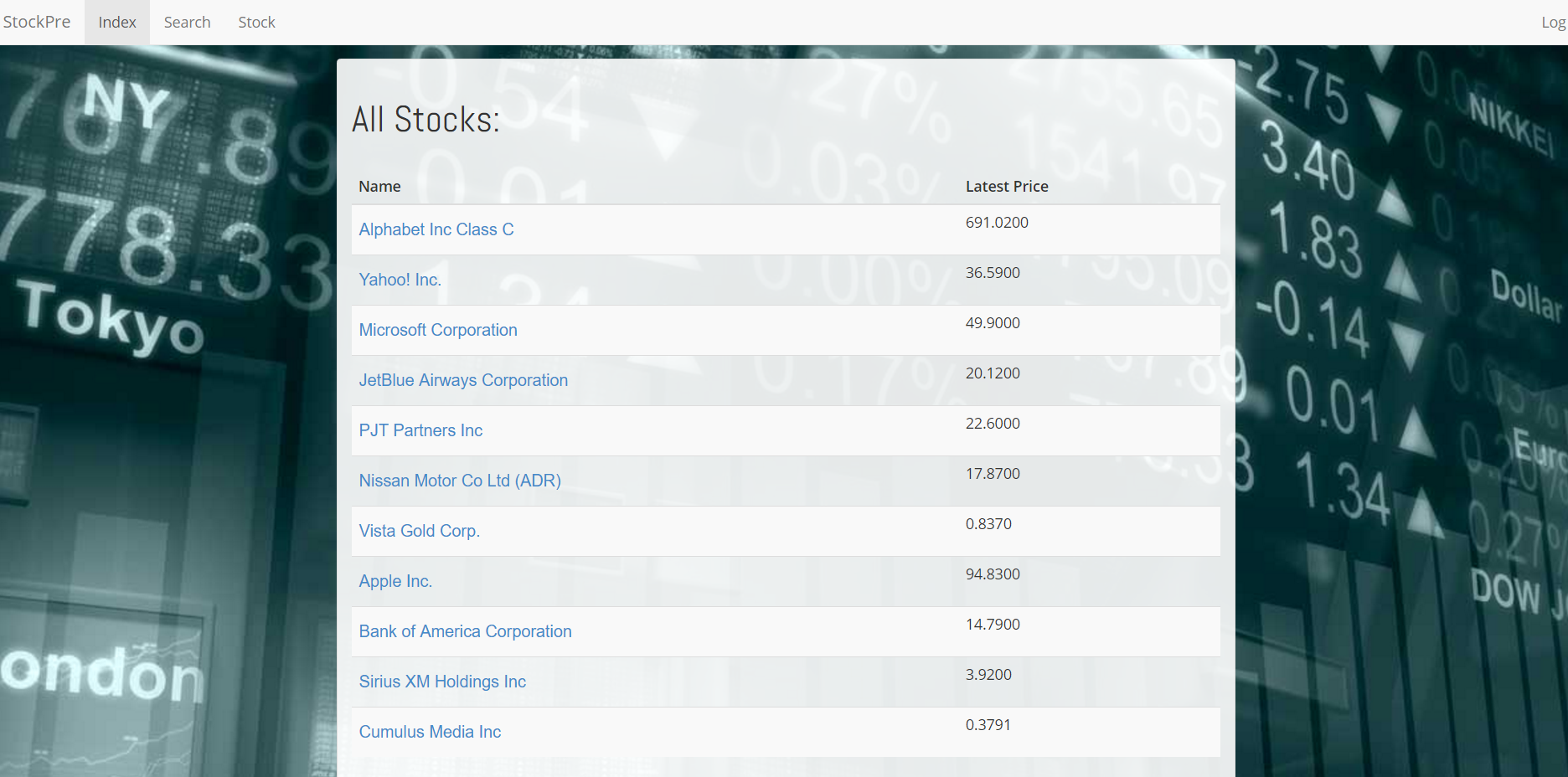
### 7.1 Preliminary design

****

When the user loads our webpage, they do not have access to its content. Therefore, the first thing the user will see is the login screen. On here, if they are already registered, they can simply fill in their username and password in the boxes and click on login to access the site. If they are not registered, they can just click on the "Register" button.

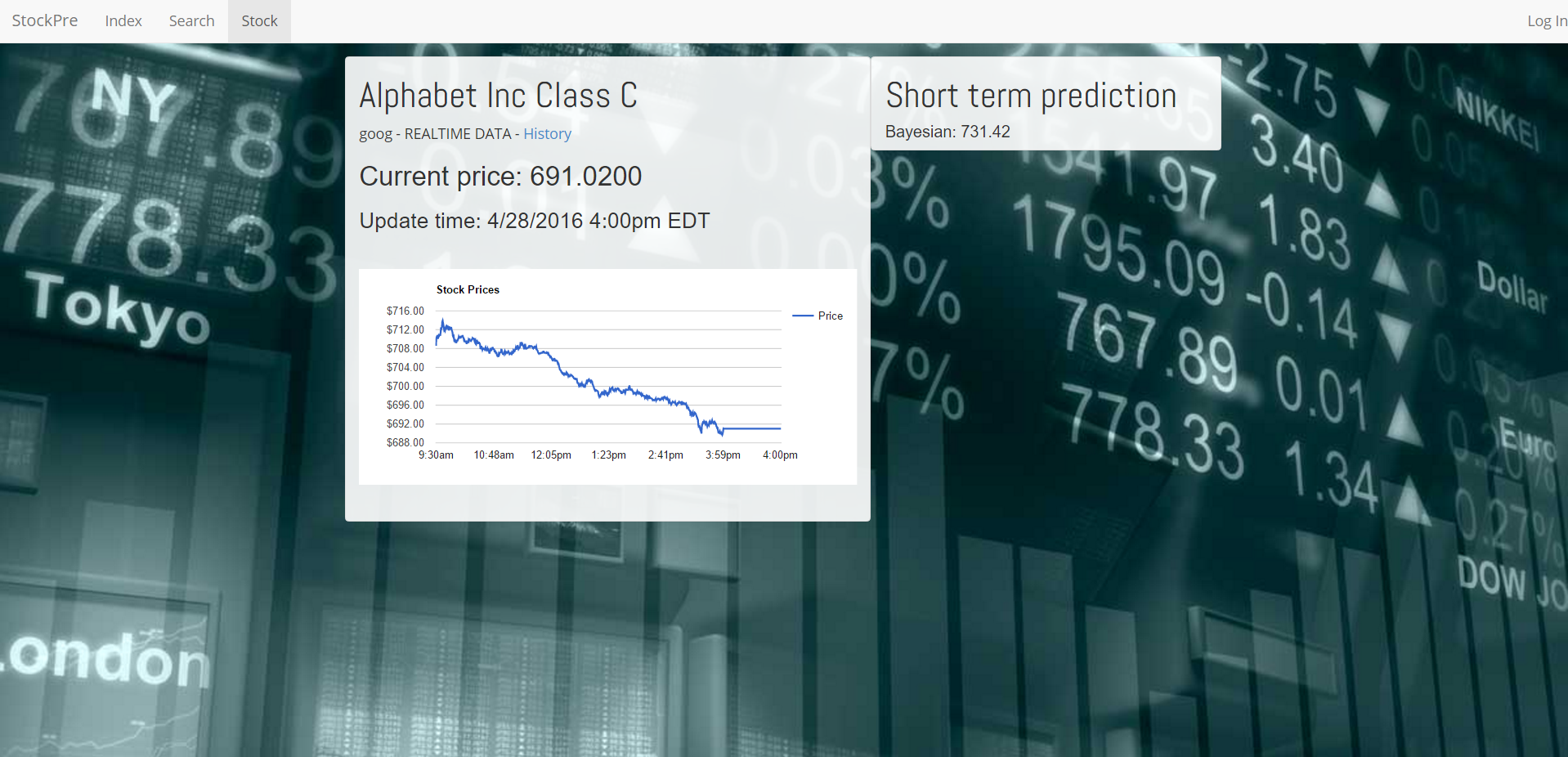


They can then register by filling in the information and hitting the "Register" button. After registration or logging in, the User will be taken to the home page. The home page itself is customized according to the User's portfolio and the current stocks they have. The home page consists of all of the User's stock's shown on one graph. It will also show the User which stock is gaining the most income and which one is the biggest loss. On the right hand side of the page, the User will see news articles and snippets of the stocks he/she has in their portfolio.

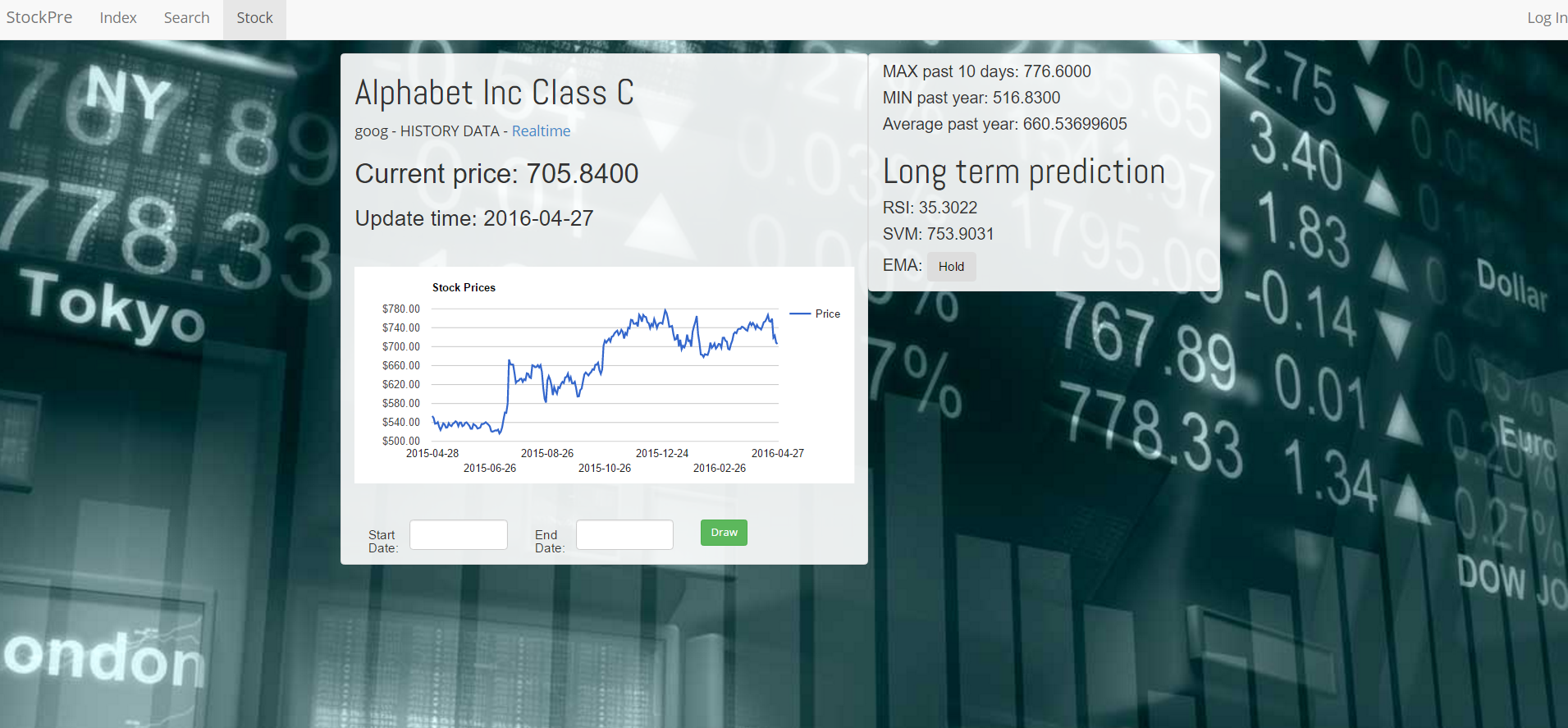




The Index is used to save all the portfolio information listed in the first part named as ‘All Stocks’. The second part is a simple comparison with the Google stock to find all the stock which has lower average price than Google.



The real time stock information



The historical stock information

The ‘Stock’ is used to have a close look at the stock information both for real-time observation and historical analysis. Also in the history of the stock, the MAX, MIN and the average is provided for the investors to have a clear look of the stock,besides, the investors are capable to search for a stock information in a certain period time by the ‘Draw’ function. For the real-time stock, short term prediction strategy is utilized while the long-term prediction strategy is used for historical stock information.

## 8 Future work



We could design a system which should regularly send emails or short messages to the customer informing him about the current trends and future scenarios. It could feed the user current news and conduct surveys to judge its performance. Besides, the system should track various stock movements simultaneously that might not be owned currently by customers and send alerts if it notices a favorable tilt in the graph that hints at immediate purchase.

## 9 Reference

[1] [www.php.net](http://www.php.net)

[2]MATLAB Neural Network Toolbox Users Guide

[3] http://www.w3schools.com/html/default.asp

[4] <http://www.w3schools.com/xml/default.asp>

[5]<http://en.wikipedia.org>

[6]http://phpwebservices.blogspot.com/

[7]<http://www.investopedia.com/>