16:332:568 S15

Software Engineering
Web Application

Web Based Stock Forecaster

First Report of Group 2

Chaoran Fu, Kai Kang, Yu Sun

Yue Wu, Wuyang Zhang

April 29, 2015

“All group members contributed equally.”

Table of Contents

Summary of Changes

1. IOS Interface was designed to provide a whole new mobile experience to attract more uses.

2. User Interface was designed completely to provide a more pleasant and fluid user experience.

3. Further prediction algorithms (SVM)were added to give a more precise prediction for a more precise stock prediction.

1. Customer Statement of Requirements

1.1 Problem Statement

The history of stock market goes back to 12th century in France where people were concerned with managing and regulating the debts of agricultural communities on behalf of the banks. Because these men also traded with debts, they could be called the first brokers. Over hundreds of years this kind of market never fails to fascinate human being. Now there are now stock markets in virtually every developed and most developing economies. Not just for it is one of the most important ways for companies to raise money but also because it allows businesses to be publicly traded, and raise additional financial capital for expansion by selling shares of ownership of the company in a public market. Nowadays this magic place gathered businessmen and investors all over the world while businessmen gain capital by selling shares of the companies. On the other hand all the investors have a urge to earn money by purchase those company shares and hoping those companies could prosper. However it is human's nature to seek some effortless way of becoming rich in a very short of time. Regardless of the actual economic development, there should be an easy way to earn profit in the stock market. This is where the financial speculation come in to place. The successful prediction of a stock's future price will no doubt give investors significant payoff and continuously excite their nerves.

We will assume that the stock price is predictable and the most two common prediction methodologies are fundamental analysis and technical analysis. Fundamental analysis is built on the belief that human society needs capital to make progress and if a company operates well, it should be rewarded with additional capital and result in a surge in stock price. That is why the fundamental Analysts are concerned with the company that underlies the stock itself. Technical analysts are not concerned with any company's background. They try to find a way to determine the future price

of a stock based solely on the (potential) trends of the past price. Using charts search for archetypal price chart patterns, technical analysts also widely use market indicators of many sorts, some of which are mathematical transformations of price, often including up and down volume, advance/decline data and other inputs.

Huge profits will come along with risks and that same volatility in the market that can be exploited for gain, can be the cause for loss as well. No doubt stock market will always attract new and passionate traders but the risks will also keep them away at some point. They lack the appropriate tools needed for the decision making process when compared to existing professional traders. Any individuals who would consider spending their spare money trading stocks but may have second thoughts due to the lack of time, resources and knowledge in the market field.

1.2 The Goal of Our Project

The goal of our project is to create a web-based system. The foremost purpose of the system should be to predict future stock prices. Our website should not only be able to automate the prediction process but also produce the prediction results with the high accuracy. Any results generated by the system should also come with a reasonable confidence value shown to the user the possibility of the given decision of "ask", "bid" or "hold" is being correct. The system should use as many as possible various methods to gauge stock trends to increase the prediction accuracy because this will be concerned most by our users. The results our system displays to the user should be simple, straightforward, easy to understand and easy to remember. Our user should be able to obtain all the prediction results and illustrative diagram at a few mouse clicks away.

Some users maybe experienced with the stock market and they certainly will not be satisfied with simple prediction results. That is why the system should also have a professional mode that could access the analysis used to get to the specific result and

reveal an explanation to the user. Thus, the system will serve both sides of the crowd. Users who want easy and quick results and who want more understanding of the technical analysis will both have a great experience using our website.

Most investment advisors suggest maintaining a diverse investment portfolio in order to reduce the risk of investment loss. So if a user would like to broaden the investment portfolio with multiple stocks, our system should provide a proper solution. An account based system is required. The system should be able to track and view multiple predictions simultaneously according to the user's personal interests. Also, when the system notice an impending decision, it should alert and inform the user of any decision may come immediately.

To achieve all the features discussed above, we need several steps to build every bricks of our project. The first step is to build up a highly organized database that will periodically download historical and real time stock price from open source websites. The next step is to implement prediction models and also keep improving the algorithm. Then, we build up the website and the mobile workspace as well.

As the report will show, at this point we have complete the first phase of the project. Our application runs continuously as a background process and periodically retrieves stock information, parses the received responses, and stores the extracted parameters into a local relational database.

2. Glossary of Terms

Fundamental Analysis

Fundamental analysis includes economic analysis, industry analysis, and company analysis. On the basis of these three analyses the intrinsic value of the shares are determined. This is considered as the true value of the share. If the intrinsic value is higher than the market price it is recommended to buy the share. If it is equal to market price hold the share and if it is less than the market price sell the shares.

Machine Learning

Machine learning is a subfield of computer science that evolved from the study of pattern recognition and computational learning theory in Artificial Intelligence. Machine learning explores the construction and study of algorithms that can learn from and make predictions on data. Such algorithms operate by building a model from example inputs in order to make data-driven predictions or decisions, rather than following strictly static program instructions.

Algorithmic Trading

Algorithmic trading, also called black-box trading or algorithmic trading, is the use of electronic platforms for entering trading orders with an algorithm which executes pre-programmed trading instructions accounting for a variety of variables such as timing, price, and volume. Algorithmic trading is widely used by investment banks, pension funds, mutual funds, and other buy-side (investor-driven) institutional traders, to divide large trades into several smaller trades to manage market impact and risk.

Artificial Neural Network (ANN)

Artificial neural network (ANN) is a family of statistical learning algorithms inspired by biological neural networks (the central nervous systems of animals, in

particular the brain) and are used to estimate or approximate functions that can depend on a large number of inputs and are generally unknown. Artificial neural networks are generally presented as systems of interconnected "neurons" which can compute values from inputs, and are capable of machine learning as well as pattern recognition thanks to their adaptive nature.

Support Vector Machine (SVM)

Support vector machines (SVMs) are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. Given a set of training examples, each marked as 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 binary linear 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.

In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.

3. System requirement

3.1 Enumerated Functional Requirement

INDENTIFIER	PW	REQUIREMENT
REQ1	5	The system shall collect current market data.
REQ2	5	The system shall collect historical market data.
REQ3	5	The system shall provide suggested profit stock based on different requirement.
REQ4	5	The system shall provide top 5 stocks picks that best fit for long term prediction.
REQ5	5	The system shall provide top 5 stocks picks that best fit for short term prediction..
REQ6	4	The system shall provide the choice to display the Simple Moving Average indicator (SMA).
REQ7	4	The system shall provide the choice to display the Exponential Moving Average indicator (EMA).
REQ8	4	The system shall provide the choice to display the Money Flow indicator(MFI).
REQ9	4	The system shall provide the choice to display the MACD indicator.
REQ10	4	The system shall provide the choice to display the Relative Strength Index(RSI) indicator.
REQ11	4	The system shall provide the choice to display the Bollinger Bands indicator.
REQ12	4	The system shall provide the choice to display the Stochastic indicator.
REQ13	3	The system shall let administrator add and remove stocks.
REQ14	3	The system shall provide the choice to compare any one stock with Nasdaq index.
REQ15	3	The system shall search any stock for current or historical data.

REQ16	2	The system shall provide login and register to users.
REQ17	2	The system shall visually track user finance via graphs and charts.
REQ18	2	The system shall provide prediction subscription to users through emails.

Table1: Enumerated functional requirement for a web based stock forecasting system

For this system, the main solution is to provide an automated process of predicting future stock prices. In order to improve the accuracy of prediction, the system must need to collect current and historical market data first. Therefore, REQ 1 and REQ 2 is given highest priority. Furthermore, the core function of the system is to help user forecast stock price so that accurate predictions can be made. Thus, REQ 3, REQ4 and REQ 5 respectively are also given highest priority.

Requirement REQ 6- REQ 12 allow users to have different prediction strategy, based on recognizing different indicators, to have a more accurate and convincing prediction. These are added features and not essential (although highly desirable) to the core functionality of the system as stated before, and thus REQ6-REQ12 were given the second highest priority.

Naturally, not all listed companies can be accounted for. Thus, before predictions can even be made, the user must select the range of stocks that the system will handle. In this system we set the number as five. Therefore, the system shall let users add and remove up to five preferred stocks, which is REQ 13.

As professional financial analysts, they cannot only focus on the stocks that have the absolute profit, rather shall compare the stock price with the Nasdaq index to see if this stock is relatively performed better than others. This features is realized by REQ 14.

REQ 15 allows users to be able to search any stock for current to historical data. REQ 16 deals with allowing users to register for an account with the system and login.

REQ 17 and REQ18 provide users much more convenience to keep touch with stock price.

3.2 Enumerated Non-functional Requirement

INDENTIFIER	PW	REQUIREMENT
REQ19	5	The system shall protect stock and user's information security.
REQ20	5	The system shall retrieve one day history stock every market open day.
REQ21	5	The system shall retrieve current stock data every 30 seconds.
REQ22	4	The system shall backup user's information once per week.
REQ23	4	The system shall backup stock information once per week.
REQ24	3	The system shall allow user register and login to obtain customize information.
REQ25	3	The system shall response to user's each requirement no more than 3 seconds.
REQ26	2	The system shall be compatible to different platform.
REQ27	2	The system shall keep the user interface as concise as possible.

Table2: Enumerated non-functional requirement for a web based stock forecasting system

Since the system will be storing user accounts, REQ 19 needs to be specified for security purposes. The system insures that all user's and stocks' information will be backup at least once per week in case the database crack down at times. That's REQ 22 and REQ 23 should realize.

For REQ 24., the system insures users can complete every time of register or login in a successful way.

Furthermore, the customer intends for the service to be as easily accessible as possible and thus REQ 26 finally insures that the system is able to be used for not only desktop browsers but also smartphones and tablets as well. This requirement is of the least priority because we will be working on the functionality for modern desktop web browsers first and then optimize for mobile app if we have enough time.

4. System Design

4.1 Overall system design

The below figure1 is the basic system design diagram

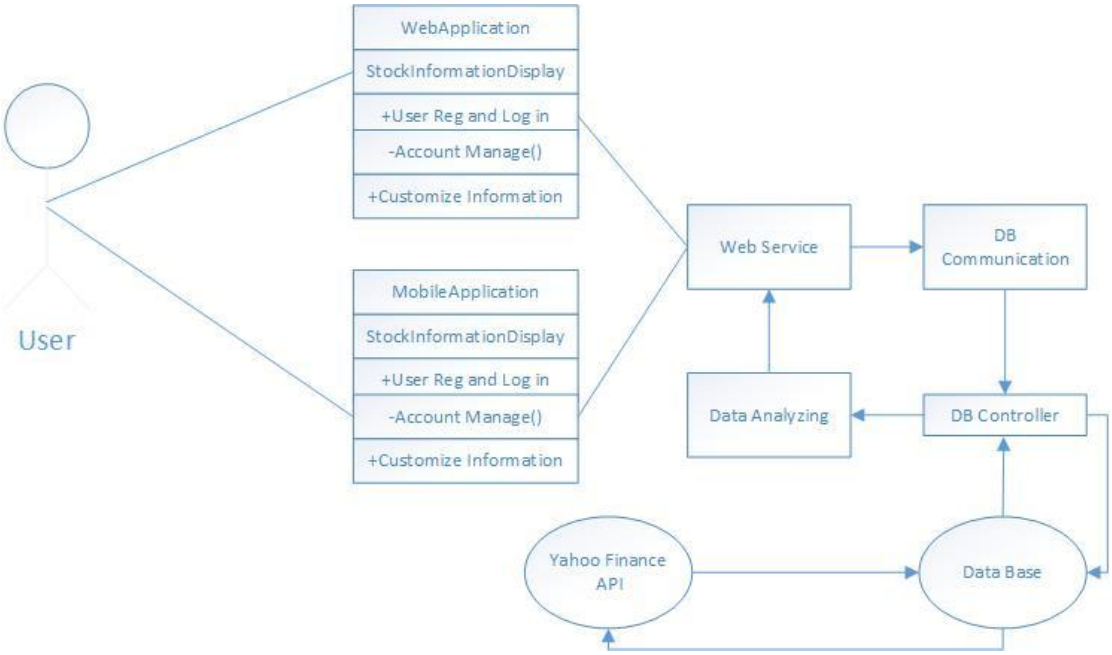

Figure 1. System level interaction diagram

The key point for the stock forecasting system design is to receive user's order, retrieving and processing data, and then send the processed data back to user. In this case, we need to design eight parts to complete the task. First, we decide to design two end points application, one for the web and one for the phone app. This is because stock market is changing at every moment, one minute may result in a huge turnover, therefore, to make the market information accessible for our customers in every place and every time is important, so design another mobile application is necessary.

Both endpoints may have the capability to display basic stock information, and let the user to register and log in to view some particular stock and information he or she may interested in. Besides, the application should have the ability to remember user's preference, so every time user log in, the application will display the most important

information that the user want to see.

The web Service class is use to obtain the user's command and send the command to background to process. Also, the web service need to send the processed data and information back to the endpoints to display.

The DB communication class act as two goals. First handle the request from web service and processed the command to data base understandable language. Second, it connect to the data base controller to ask for requested data.

The DB controller class use to decide if the request is legible, if yes, allow the database to process the request, if not, deny the request. This is use to protect the data base to become overloaded and secure the database.

The database is used to store all the information including the stock market data and user's account information. Also, the database need to store the data comes from Yahoo Finance.

4.2 Endpoint interaction design

The end point should response to the user's request. The ability is shown in figure 2.

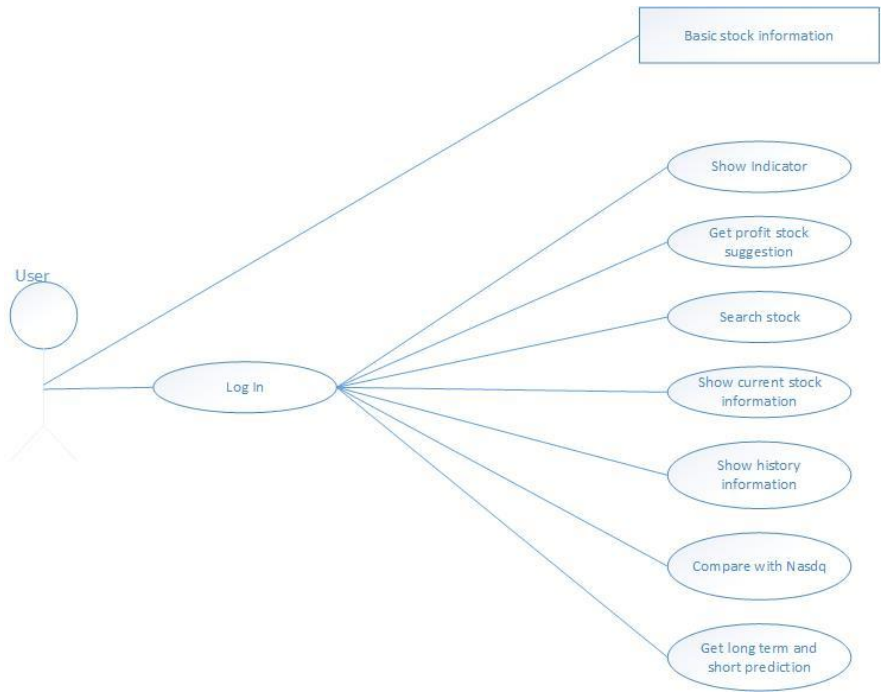

Figure 2 Client side use case

3. Background interaction design

The most important goal for background design is to have the ability to collect data and processing them. The detail can be shown in figure 3.

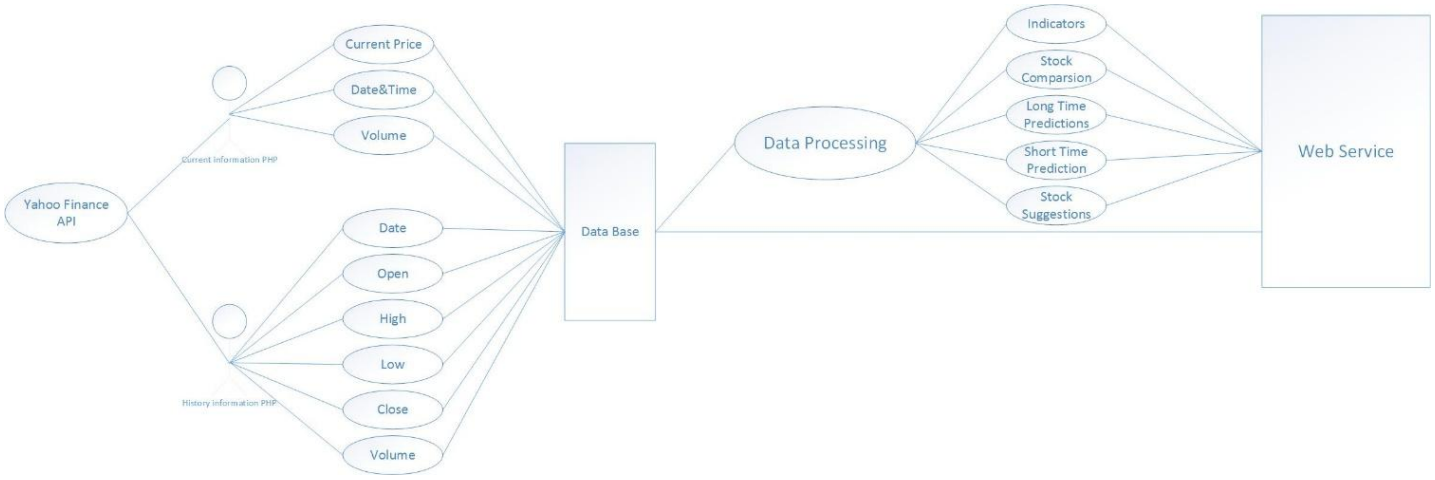

Figure 3 Back interaction design

First, there are two basic PHP scripts that runs background continuously and obtain basic stock data from Yahoo Finance periodically and data base receive and store all the data. Then the data processing class call for this basic data and do analyzing, then the web service receive both the basic data and processed data as user required.

4. Overall interaction.

As the background and endpoints part work together, the system can display request information to the user. The overall interaction can be shown in figure 4.

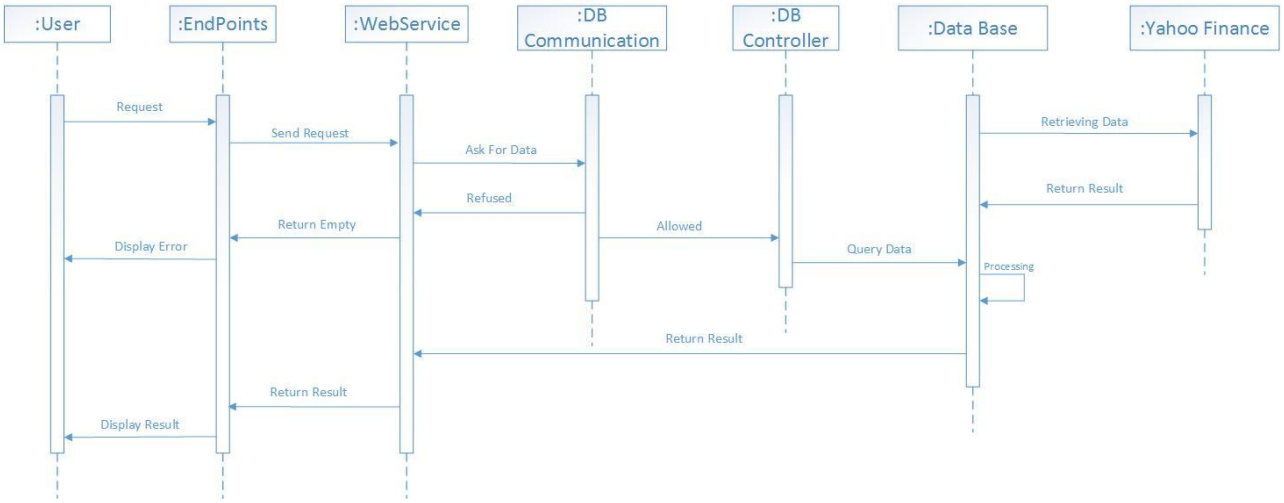

Figure 4 Interaction Diagram

The user ask for information from endpoints and trigger the system, if the ask is not legal, then refuse the request, if the ask is legal, then processing and return the result from database. Meanwhile, the data base retrieve necessary from Yahoo Finance continuously without interruption.

Stock information retrieve from Yahoo Finance

History information	Real time information
Stock	Stock
Date	Current Price
Open Price	Date
High Price	Time
Low Price	Volume
Close Price	
Volume	

Table 3 Stock Information

5. Functional Requirements Specification

5.1 Stakeholders

The stock forecast system concentrates on providing the prediction of stock price in the future time on stocks that users require.

Register Users

Register users are defined as the most principal user objects. They have access to pick any stock they interest and would like to observe from the stock list within the database. They have access to add indicators on stock interactive charts so as to observe and monitor the tendency line in detail of stocks. They can also calculate stock return profits by inputting stock parameters. Moreover, the system will provide ask or bid suggestion of preferred stocks and notify users about the recent price change of stocks.

Systems architects and developers

System architects and developers are defined as system supporters. They take responsibility on building the stock forecast system satisfying the users' requirements, testing and maintaining they system, and providing technical support to other stakeholders.

5.2 Actors and Goals

User (Initiating type)

Goals: to interact with the stock forecast system, to choose preferred stocks and obtain ask and bid suggestions.

Administrator (Initiating type)

Goals: to collect current and history stock price, access, manage and maintain the database as well as the website, providing stock service to the users.

Yahoo Finance server and database (Participating type)

Our server and database (Participating type)

5.3 Use Cases

5.3.1 Casual Description

The summary of use cases are illustrated as following.

UC-1: Login

Allowing a user to access and manage the account and using the system features provided specifically for registered users. [Derived from REQ16]

UC-2: Register

Allowing a visitor to become a registered user through completing the registration procedure. [Derived from REQ16]

UC-3: Manage Stock

Allowing the administrator to add or remove a stock from stock forecast database. (<include> login) [Derived from REQ13]

UC-4: Search Stock

Allowing the user to search from the prediction list one specific stock which has been analyzed and are available to users through the stock ticker or the company name. [Derived from REQ2, REQ15]

UC-5: Add Stock Indicator

Allowing the user to add some indicators on the stock price data to analysis the stock performance from distinct perspectives in details. (<include> login) [Derived from REQ6-REQ12]

UC-6: Suggest Ask or Bid

Allowing the user to execute the system predicting stock price and affording the asked and bid strategies. (<include> login) [Derived REQ3, REQ4, REQ5]

UC-7: Track Stock Price

Allowing the user to create a watch list or portfolio to track the performance of stocks. (<include> login) [Derived from REQ13]

UC-8: Manage Portfolio

Allowing the user to check the profit return of the portfolio by inputting stock

parameters. (<include> login) [Derived from REQ13, REQ14, REQ17]

UC-9: Update Stock Price

Allowing the timer to send the request of the loading the current stock price from Yahoo Finance to the database. (<include> login) [Derived from REQ1]

UC-10: Notify Price Change

Allowing the user to get notification from the system such as email if the prediction has been changed. (<include> login) [Derived from REQ18]

5.3.2 Use Case Diagram

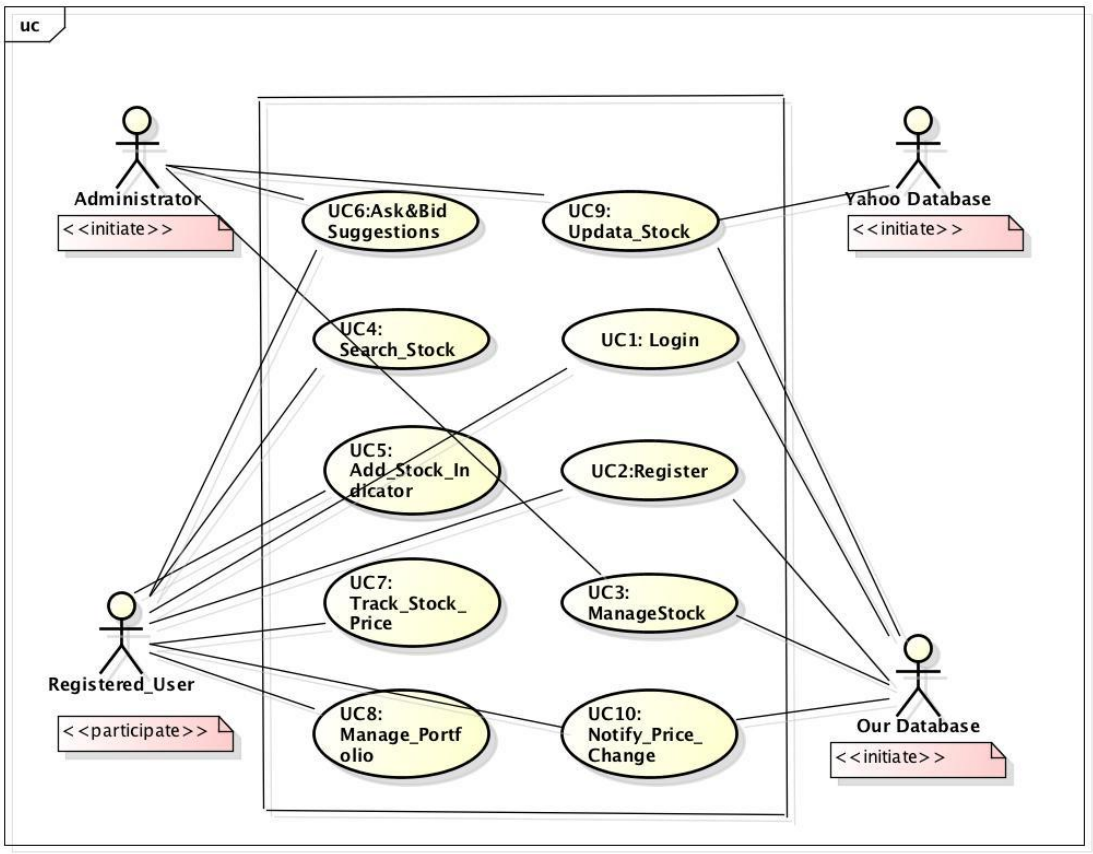

Figure 5 Use Case Diagram

5.3.3 Traceability Matrix

REQ	PW	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10
REQ1	5									X	
REQ2	5				X						
REQ3	5						X				
REQ4	5						X				
REQ5	5						X				
REQ6	4					X					
REQ7	4					X					
REQ8	4					X					
REQ9	4					X					
REQ10	4					X					
REQ11	4					X					
REQ12	4					X					
REQ13	3							X	X		
REQ14	3								X		
REQ15	3				X						
REQ16	2	X	X								
REQ17	2								X		
REQ18	2										X
MAX PX		2	2		5	4	5	3	3	5	2
TOTAL		2	2		8	28	15	3	8	5	2

Table 4 Traceability Matrix

5.3.4 Fully Dressed Description

Use case UC-3: ManageStock, allows the administrator to manage stocks in the system. The administrator can add stocks into the system or remove stocks from the system. After choosing stocks, the system would retrieve the data from the price provider and build the stock database. Usually each stock has the unique identifier, ticker symbol. The system would use the ticker to mark the stock for searcher to visit the stock database.

Use Case UC-3 ManageStock

Related Requirements: REQ13

Initiating Actor: Administrator

Actor's Goal: to add or remove stocks in the database that predictions are made on.

Participating Actors: Price Provider, Database

Preconditions: Administrator is logged in.

Success End Condition: All of the chosen stocks are added to the list of stocks within the database that predictions are made on. Historical prices for the given stocks are retrieved from the Price Provider and stored within the database.

Failed End Condition: The administrator is notified that price provider cannot find the particular stock in the web database.

Flow of Events for Main Success Scenario:

Include::Login(UC-1)

1. Administrator enters ticker symbols for the stocks he wants to add or remove.

2. Administrator clicks the "Add Stocks" button of "Remove Stocks" button.

(loop: for each ticker symbol entered, use Remove Stocks for example)

1. System verifies the stock currently already exist within the **Database**.

2. System sends the remove stock request to the **Database**.

3. Database deletes the stock data and returns the information

4. System notifies **Administrator** that the stock is successfully removed.

Flow of Events for Alternate Scenarios:

1a. System cannot find the stock in the database.

1. **System** notifies **Administrator** the stock that has failed to be added.
2. **System** lets the **Administrator** to remove another stock.

Use case UC-5: Add Indicator, allows the user to add some indicators on the stock price data to analysis the stock performance comprehensively.

Use Case UC-5 Add Indicator

Related Requirements: REQ6-REQ12

Initiating Actor: User

Actor's Goal: to add the indicator on the price data.

Participating Actors: Price Provider, Database

Preconditions: User is logged in.

Success End Condition: The user can view the indicator on the price data.

Failed End Condition: The user is notified the indicator requested is not available for the particular stock data.

Flow of Events for Main Success Scenario:

Include::Login(UC-1)

Include::Search(UC-4)

1. **User** enters the ticker symbols for the stocks to search.
2. **System** verifies that the given stock exists within the Database.
3. **System** offers an indicator list for user to choose (such as SMA, EMA, MFI, MACD, RSI).
4. **User** chooses the indictor he wants and sends to system.
5. **System** gets the historical data from the **Database**.
6. **System** calculates the indictor data and sends to Grapher.
7. **System** displays the graph and historical data to the **User**.

Flow of Events for Alternate Scenario:

3a. The historical data cannot support the particular indicator.

1. System notifies the user that the indicator cannot be calculated based on the stock data in the database.

6. Algorithms Design

6.1 Indicator

Besides the price prediction methods we mentioned above, we decided to offer users more interesting information, which is different kinds of indicators. Although we doesn't give buying or selling instructions to users directly through indicators, however, user can make trade decision more carefully if they combine our prediction and indicators together. For example, when we give a predict price and suggest user to enter the trade , user should also check the volume change status, if the volume change remain slightly, it may suggests that predict price may be wrong or it's may not be a good time for buying. The indicators we offer to users will describe in detail below.

6.1.1 SMA

SMA is short for simple moving average, it's a simple, or arithmetic, moving average that is calculated by adding the closing price of the security for a number of time periods and then dividing this total by the number of time periods. Short-term averages respond quickly to changes in the price of the underlying, while long-term averages are slow to react.

The basic function is

$$SMA = \frac{P_M + P_{M-1} + \ldots + P_{M-(n-1)}}{n}$$

Where P stands for price and n stands for period. In our system, we choose five days as a period for history price and calculate the SMA by using the close price. For current price, we choose five minutes as a time period.

A Simple Moving Average can serve as a line of resistance. At times when price is in a downtrend and the moving average is in a downtrend as well, and price tests the SMA above and is rejected a few consecutive times (i.e. the **moving average is serving as a resistance line**), then buy on the next rally up to the Simple Moving Average.

6.1.2 EMA

EMA is short for exponential moving average, it's an indicator similar to SMA, but more weight is given to the latest data. This type of moving average reacts faster to recent price changes than a simple moving average, the formula can be describe as below:

$$EMA_{today} = EMA_{yesterday} + \alpha(price_{today} - EMA_{yesterday})$$

Where α is a coefficient equals to $2/(N+1)$, Here N is the time period.

If we expand $EMA_{yesterday}$ we can get the below formula

$$EMA_{dotay} = \frac{p_1 + (1-\alpha)p_2 + (1-\alpha)^2 p_3 ...}{1 + (1-\alpha) + (1-\alpha)^2 + (1-\alpha)^3 + ...}$$

Since this is an infinite sum but only N periods will dominate the result, so the above formula can be rewrite as:

$$1-(1-\alpha)^{N+1}$$

For history, we also choose five days as a period, and for current we choose five minutes as a period

Compare with SMA, although EMA response faster than SMA, EMA's are not necessarily faster to give buy and sell signals when using moving average crossovers. Nevertheless, Moving Averages remain the most popular and arguably the most effective technical analysis indicator out on the market today.

6.1.3 MACD

EMA is very useful and can even use to construct some other indicators, MACD is an example of using EMA.

MACD is short for Moving Average Convergence Divergence It's a trend-following momentum indicator that shows the relationship between two moving averages of prices. The MACD is calculated by subtracting the 26-day exponential moving average (EMA) from the 12-day EMA. For current price, we choose 26 minutes and 12 minutes.

The formula can be describe as below:

$$DIF = EMA_{12} - EMA_{26}$$

There should also have a signal lines, and make the use of MACD better, however due to the end of the term, we don't have enough time to calculate the signal line. Even though, the MACD will give trader enough information to consider whether they should enter a trade.

There are three common methods used to interpret the MACD:

Crossovers - As shown in the chart above, when the MACD falls below the signal line, it is a bearish signal, which indicates that it may be time to sell. Conversely, when the MACD rises above the signal line, the indicator gives a bullish signal, which suggests that the price of the asset is likely to experience upward momentum. Many traders wait for a confirmed cross above the signal line before entering into a position to avoid getting "faked out" or entering into a position too early, as shown by the first arrow.

2. Divergence - When the security price diverges from the MACD. It signals the end of the current trend.

3. Dramatic rise - When the MACD rises dramatically - that is, the shorter moving average pulls away from the longer-term moving average - it is a signal that the security is overbought and will soon return to normal levels.

Traders also watch for a move above or below the zero line because this signals the position of the short-term average relative to the long-term average. When the MACD is above zero, the short-term average is above the long-term average, which signals upward momentum. The opposite is true when the MACD is below zero. As you can see from the chart above, the zero line often acts as an area of support and resistance for the indicator.

6.1.4 MFI

The Money Flow Index (MFI) is an oscillator that uses both price and volume to measure buying and selling pressure. MFI is also known as volume-weighted RSI. MFI starts with the typical price for each period. Money flow is positive when the typical

price rises (buying pressure) and negative when the typical price declines (selling pressure). A ratio of positive and negative money flow is then plugged into an RSI formula to create an oscillator that moves between zero and one hundred. As a momentum oscillator tied to volume, the Money Flow Index (MFI) is best suited to identify reversals and price extremes with a variety of signals. Because MFI will use some data that only history price have, so we only implement it in the history data.

There are serval steps involved to calculate the MFI:

1. Calculate the typical price, the typical price is defined by the below formula:

$$typicalprice = \frac{high + low + close}{3}$$

2. Calculate the positive and negative money flow:

$$moneyflow = typicalprice * volume$$

The Money flow is divided into positive and negative money flow.

Positive money flow is calculated by adding the money flow of all the days where the typical price is higher than the previous day's typical price.

Negative money flow is calculated by adding the money flow of all the days where the typical price is lower than the previous day's typical price.

If typical price is unchanged then that day is discarded.

3. Calculate the money ratio:

$$moneyratio = \frac{posMF}{negMF}$$

4. Calculate the MFI:

$$MFI = 100 - \frac{100}{1 + moneyratio}$$

There are two threshold in the MFI: Below 20 is considered oversold; look for buying opportunities. Above 80 is in overbought territory; look for sell signals. Thus a

trader should look carefully with MFI and to seek for a better opportunity to enter the trade. In our system, we choose 10 days as a period to calculate the MFI.

6.1.5 RSI

The relative strength index (RSI) is a technical indicator used in the analysis of financial markets. It is intended to chart the current and historical strength or weakness of a stock or market based on the closing prices of a recent trading period.

The basic formula is

$$RSI = 100 - 100/(1+RS)$$

Where RS=Avg of x days' up closes/Avg of x day's down closes. Because RSI also involve some data only history price have, thus we didn't implement it into current price. For our system, we choose 10 days as a period.

Similar to MFI, RSI also have two threshold, an asset is deemed to be overbought once the RSI approaches the 70 level, meaning that it may be getting overvalued and is a good candidate for a pullback. Likewise, if the RSI approaches 30, it is an indication that the asset may be getting oversold and therefore likely to become undervalued.

6.1.6 ADL

The Accumulation Distribution Line (ADL) is a cumulative measure of each period's volume flow, or money flow. A high positive multiplier combined with high volume shows strong buying pressure that pushes the indicator higher. Conversely, a low negative number combined with high volume reflects strong selling pressure that pushes the indicator lower. Money Flow Volume accumulates to form a line that either confirms or contradicts the underlying price trend. In this regard, the indicator is used to either reinforce the underlying trend or cast doubts on its sustainability. An uptrend in prices with a downtrend in the Accumulation Distribution Line suggests underlying selling pressure (distribution) that could foreshadow a bearish reversal on the price chart. A downtrend in prices with an uptrend in the Accumulation Distribution Line

indicate underlying buying pressure (accumulation) that could foreshadow a bullish reversal in prices.

The basic function is as below:

1) Money Flow Multiplier = [(Close - Low) - (High - Close)] /(High - Low)

2) Money Flow Volume = Money Flow Multiplier x Volume for the Period

3) ADL = Previous ADL + Current Period's Money Flow Volume

The Money Flow Multiplier fluctuates between +1 and -1. As such, it holds the key to the Money Flow Volume and the Accumulation Distribution Line. The multiplier is positive when the close is in the upper half of the high-low range and negative when in the lower half. This makes perfect sense. Buying pressure is stronger than selling pressure when prices close in the upper half of the period's range (and vice versa). The Accumulation Distribution Line rises when the multiplier is positive and falls when the multiplier is negative.

The multiplier adjusts the amount of volume that ends up in the Money Flow Volume. Volume is in effect reduced unless the Money Flow Multiplier is at its extremes (+1 or -1). The multiplier is +1 when the close is on the high and -1 when the close is on the low. All volume is positive when +1 and all volume is negative when -1. At .50, only half of the volume translates into the period's Money Flow Volume. The table below shows the Money Flow Multipliers, Money Flow Volume and Accumulation Distribution Line for Research-in-Motion (RIMM). Notice how the multiplier is between .50 and 1 when the close is strong and between -.50 and -1 when the close is weak.

6.1.7 OBV

On Balance Volume measures buying and selling pressure as a cumulative indicator that adds volume on up days and subtracts volume on down days. OBV was developed by Joe Granville and introduced in his 1963 book, Granville's New Key to

Stock Market Profits. It was one of the first indicators to measure positive and negative volume flow. Chartists can look for divergences between OBV and price to predict price movements or use OBV to confirm price trend

The basic function is as below:

1) If the closing price is above the prior close price then:

Current OBV = Previous OBV + Current Volume

2) If the closing price is below the prior close price then:

Current OBV = Previous OBV - Current Volume

3) If the closing prices equals the prior close price then:

Current OBV = Previous OBV (no change)

Granville theorized that volume precedes price. OBV rises when volume on up days outpaces volume on down days. OBV falls when volume on down days is stronger. A rising OBV reflects positive volume pressure that can lead to higher prices. Conversely, falling OBV reflects negative volume pressure that can foreshadow lower prices. Granville noted in his research that OBV would often move before price. Expect prices to move higher if OBV is rising while prices are either flat or moving down. Expect prices to move lower if OBV is falling while prices are either flat or moving up.

The absolute value of OBV is not important. Chartists should instead focus on the characteristics of the OBV line. First define the trend for OBV. Second, determine if the current trend matches the trend for the underlying security. Third, look for potential support or resistance levels. Once broken, the trend for OBV will change and these breaks can be used to generate signals. Also notice that OBV is based on closing prices. Therefore, closing prices should be considered when looking for divergences or support/resistance breaks. And finally, volume spikes can sometimes throw off the indicator by causing a sharp move that will require a settling period.

6.1.8 BB

The Bollinger Bands automatically widen when volatility increases and narrow when volatility decreases. This dynamic nature of Bollinger Bands also means they can be used on different securities with the standard settings. For signals, Bollinger Bands can be used to identify M-Tops and W-Bottoms or to determine the strength of the trend.

The basic function is as below:

1) Middle Band = 20-day simple moving average (SMA)
2) Upper Band = 20-day SMA + (20-day standard deviation of price x 2)
3) Lower Band = 20-day SMA - (20-day standard deviation of price x 2)

Bollinger Bands reflect direction with the 20-period SMA and volatility with the upper/lower bands. As such, they can be used to determine if prices are relatively high or low. **According to Bollinger, the bands should contain 88-89% of price action, which makes a move outside the bands significant.** Technically, prices are relatively high when above the upper band and relatively low when below the lower band. However, relatively high should not be regarded as bearish or as a sell signal. Likewise, relatively low should not be considered bullish or as a buy signal. Prices are high or low for a reason. As with other indicators, Bollinger Bands are not meant to be used as a stand-alone tool. Chartists should combine Bollinger Bands with basic trend analysis and other indicators for confirmation.

6.2 Prediction Strategy

6.2.1 Bayes Curve Fitting

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

$$p(\mathrm{t} \mid \mathrm{x}, \mathrm{x}, \mathrm{t}) = \int p(\mathrm{t} \mid \mathrm{x}, \mathrm{w})\, \mathrm{p}(\mathrm{w} \mid \mathrm{x}, \mathrm{t})\, \mathrm{dw}$$

Here p (t|x,w) is given by 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, and can be found by normalizing the right-hand side of (1.66). We shall see in Section 3.3 that, for problems such as the curve-fitting example, this posterior distribution is a Gaussian and can be evaluated analytically. Similarly, the integration in (1.68) can also be performed analytically with the result that the predictive distribution is given by a Gaussian of the form.

Here's the mean as the prediction price:

$$m(\mathrm{x}) = \beta \phi(\mathrm{x})^{t} S \sum_{n=1}^{N} \phi(\mathrm{x}_{n}) \phi(\mathrm{x})^{T}$$

6.2.2 Support Vector Machine (SVM)

In machine learning support vector machines is one type of supervised learning models with machine learning algorithm so as to implement data classify and linear predict. First source data can be trained and then a model based on this source data will be constructed. According to this model predict source data will be input and then predict value will be calculated by using the model.

In this system svmlib has been used to implement the algorithm of support vector machine on the stock forecasting system. In the predict accuracy test through using history price data support vector machine achieves 95% predicting accuracy.

In the following the concrete steps of using svmlib will be illustrated.

1. Scale source data and predict data

To keep higher predicting accuracy and higher calculating speed, it is reasonable to implement scaling on source data and predict data. By using svm-scale code data can be set into the range from 0 to 1.

2. Constructing predict model

When scaling source data is prepared svm-train can be used to train a predict model in which epsilon-SVR type and radial basis function are used. Note that two important parameters gamma, a parameter in kernel function, cost, a parameter set in epsilon-SVR are supposed to be set carefully and precisely. To achieve this goal a tool named grid.py is used here. Grid.py is a parameter selection tool for SVM using RBF (radial basis function) kernel. It uses cross validation technique to estimate the accuracy of each parameter combination in the specified range and help to decide the best parameter in a problem. Through using grid.py it provides best parameters including gamma and cost required in the model. After that a predicting model with proper parameters on input source data will be constructed.

3. Predict new prices based on model

When a proper predicting model has been constructed predict source data can be input into the model then it will offer predict prices in long term.

Take Amazon stock data as an example.

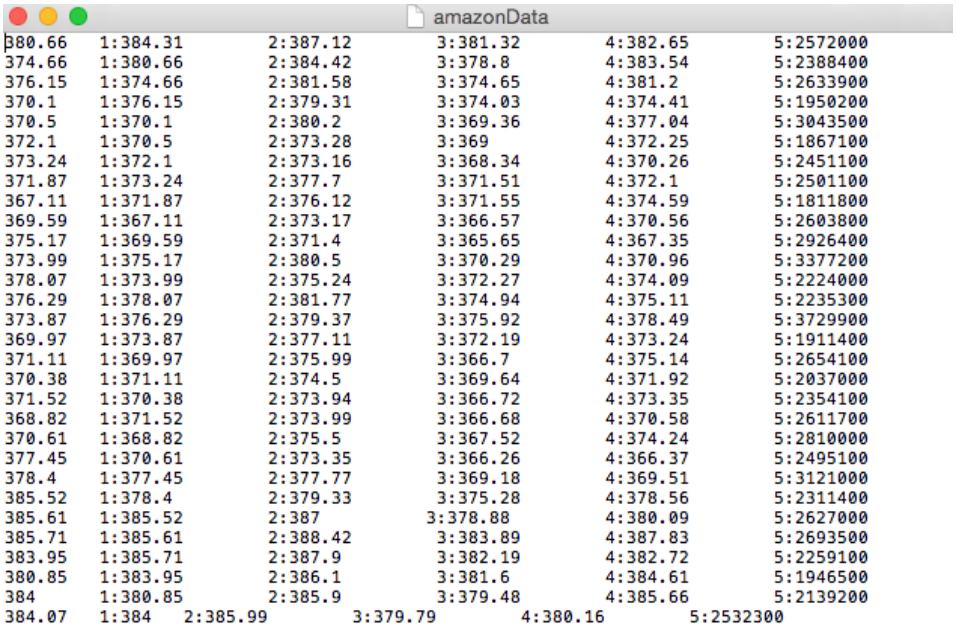

Fig 6-1 amazon source data

In figure 6-1 each line with 6 sets of number becomes a array data. The labels from 1 to 5 represent five different parameters: open price, close price, high price, and low price, volume within a day. And the first number represents the real open price in a next day. Totally we have 30 arrays data within a month as the source input data.

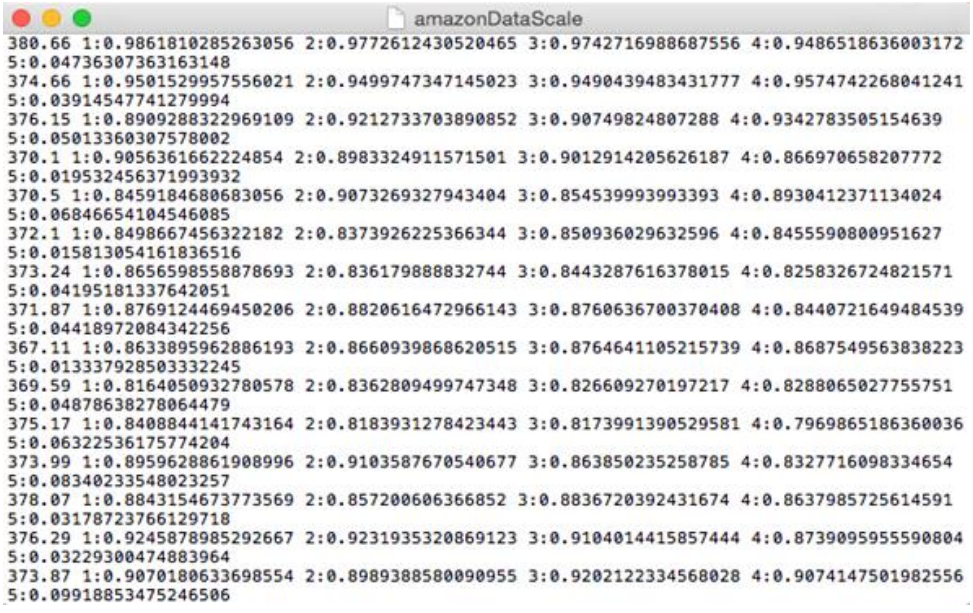

Fig 6-2 amazon source scale data

Figure 6-2 shows the result of scaling source data on source data showed in figure 6-1

This scaling data will be practically used to construct our model.

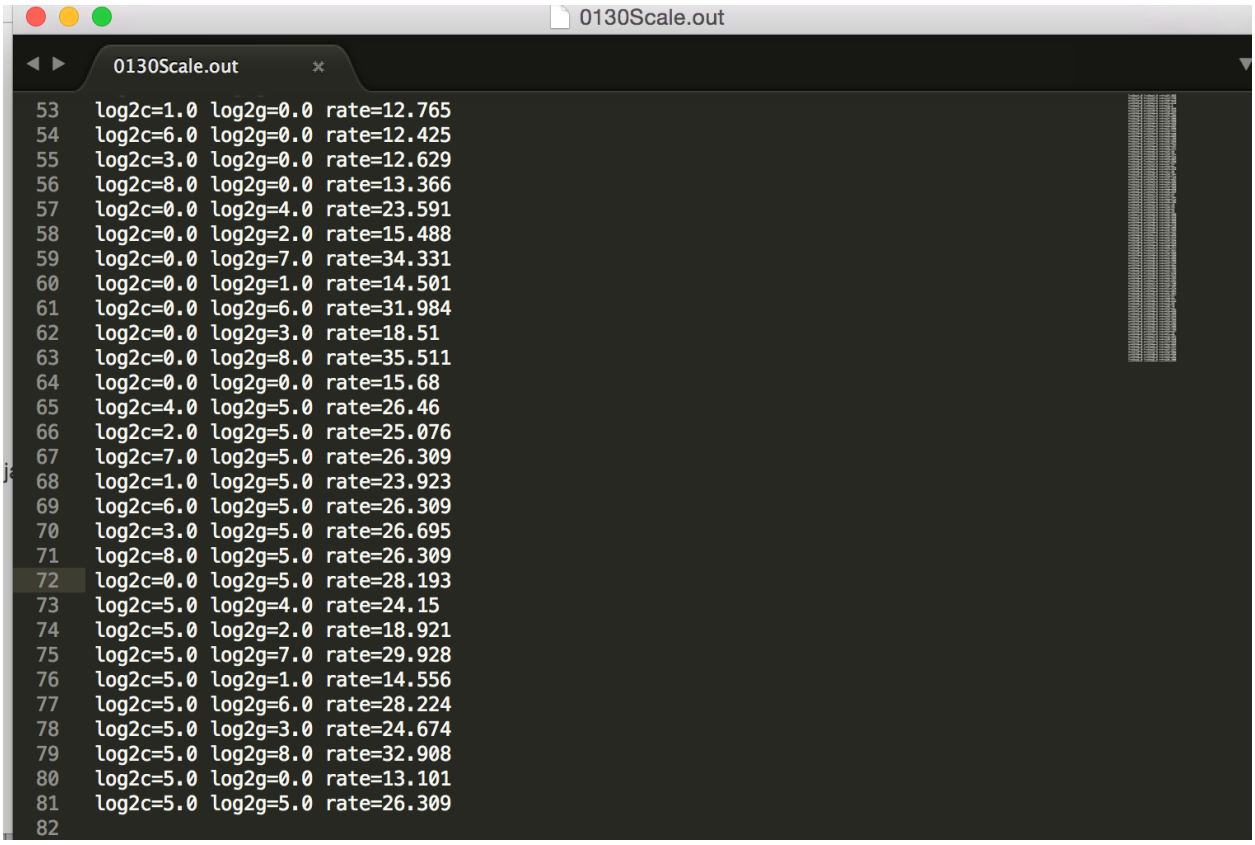

Fig 6-3 Find best parameters gama and cost

Figure 6-3 shows through using grid.py best parameters gama and cost have been found.

Then through using these two parameters we choose epsilon-SVR type of SVM and RBF kernel to build the model. After that predict source data can be input into the model it offers predicting prices.

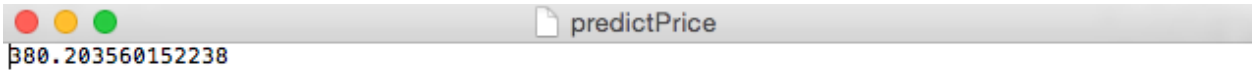

Fig 6-4 Predict Price using the model

Moreover, when to change source data SVM can also predict prices in different time including 7 days, 15days, 30days and etc. For example if the requirement is to predict a price in a next week, then the first values in all sets in a source data will be altered into the prices in a next week as training data. Through using this source data to build a model, this model is the one used to predict prices in a next week.

7. User Interface Design

7.1 Web Interface Design

1. Home Page

Everything begins with our Home Page. The whole website is surrounded by a grey tone giving people a very professional look. Below our website logo, you can see a navigation bar which will take you to other pages of our website.

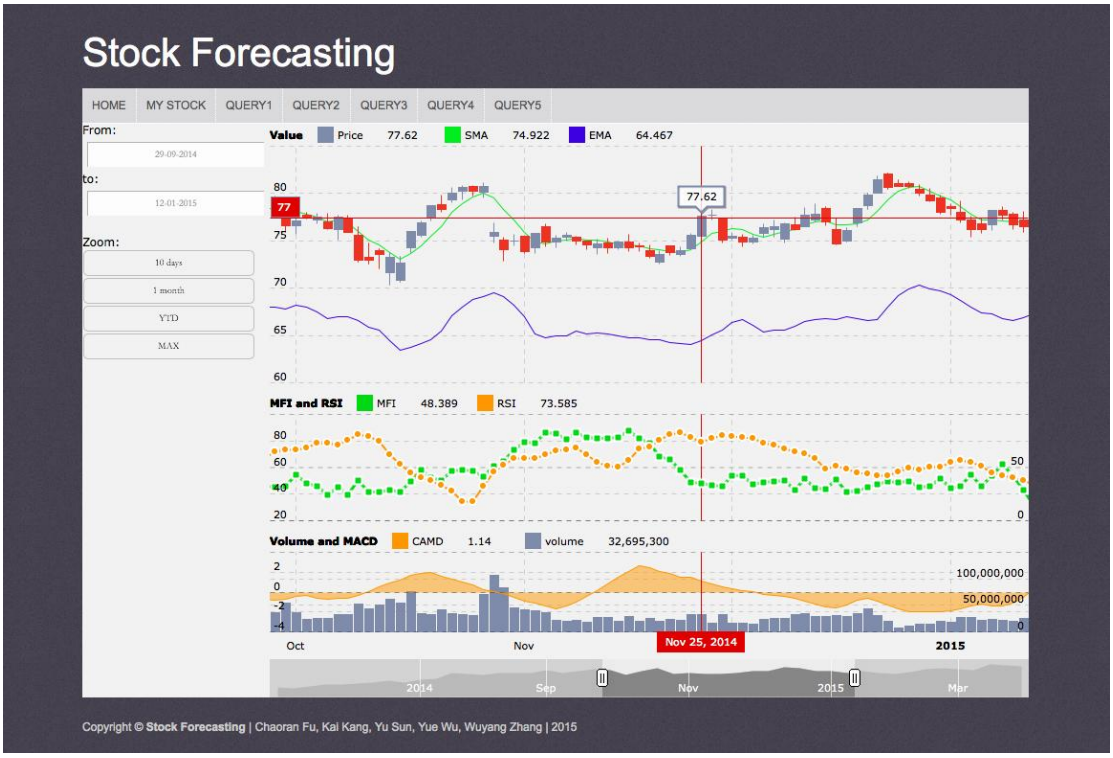

Figure 7-1. Home Page

The most obvious feature of our Home Page is that beautiful stock chart. This chart gives our user a basic understanding of our website before they go through the login/signup system. The stock chart can highly interact with our user and motivates them to look further into our website.

2. Login and Sign Up

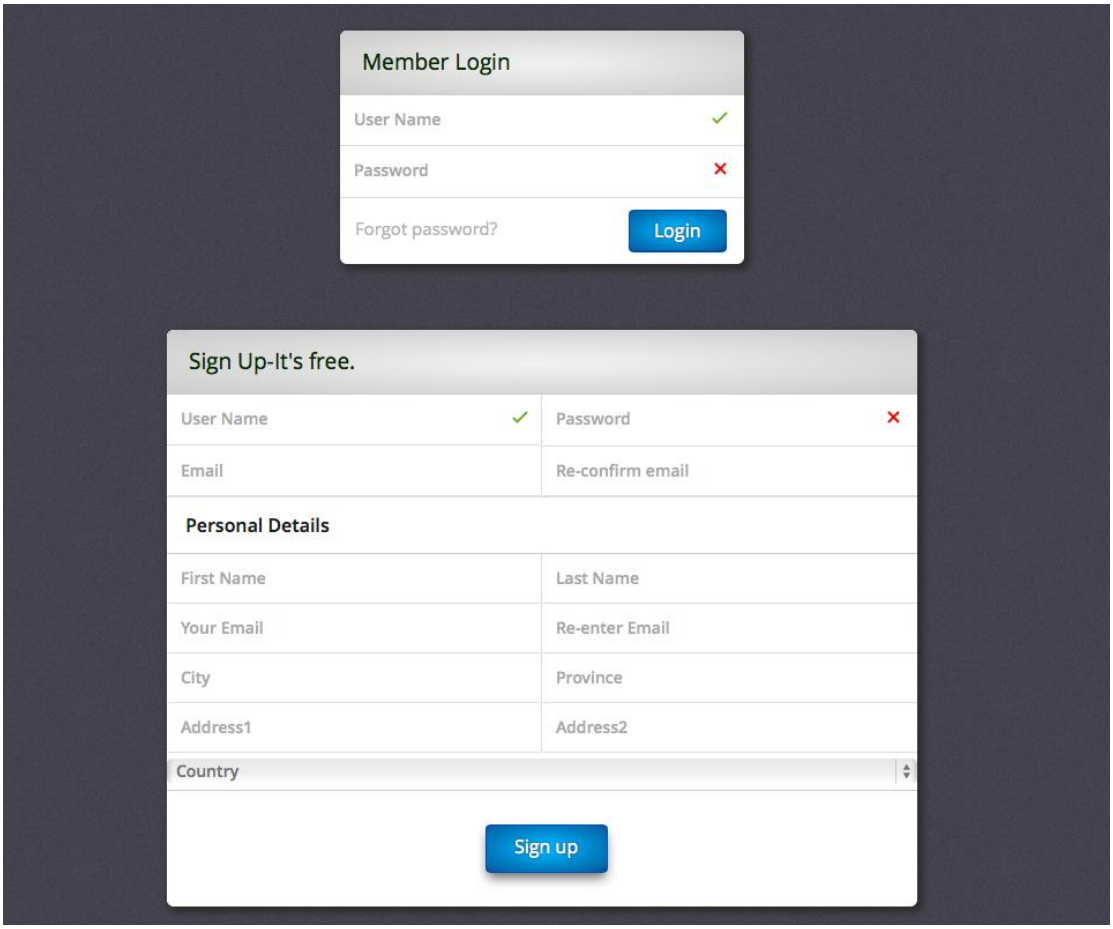

Figure 7-2. Login & Signup Page

Click on the My Stock tab and it will take you to the login and signup page. All of our pages will continue to be the same style as the Home Page providing an associative perception other than incoherent. If a user is already a member, he can simply using his own user name and password to login. For new users, the sign-up is quick and simple. Just type in the user name, the password and the user email and you are ready to create a new account. We also allow user to setup their personal details including their full name, address and even their country. Yet the personal details are optional when creating a new account because most users will expect a quick start up.

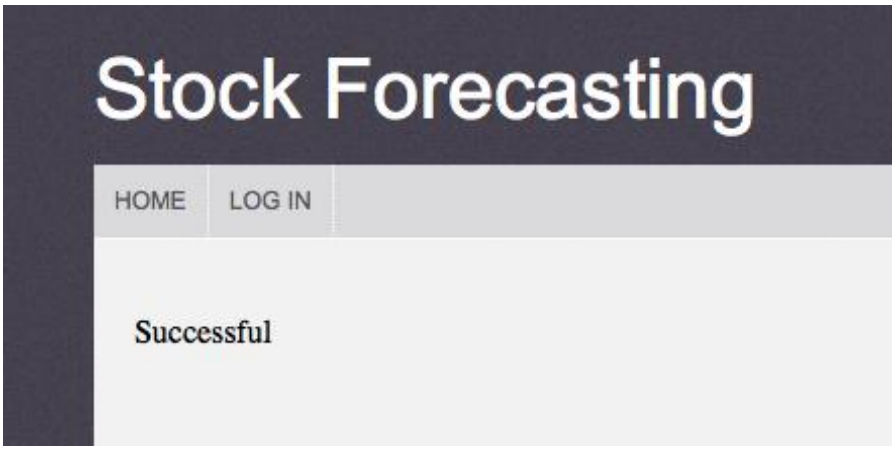

Figure 7-3. Successfully Signed Up

After successfully signed up, the user can click on the log in button to go back to the log in page and access the personal page from there.

3. Personal Page

We provide customized service for our user in the personal page. After our user successfully logged in, he will receive a welcome message. However if this is the first time he logs in, then he has not selected any stock yet. The user can select the stock he or she is interested in and click Submit in the Stocks Selection area.

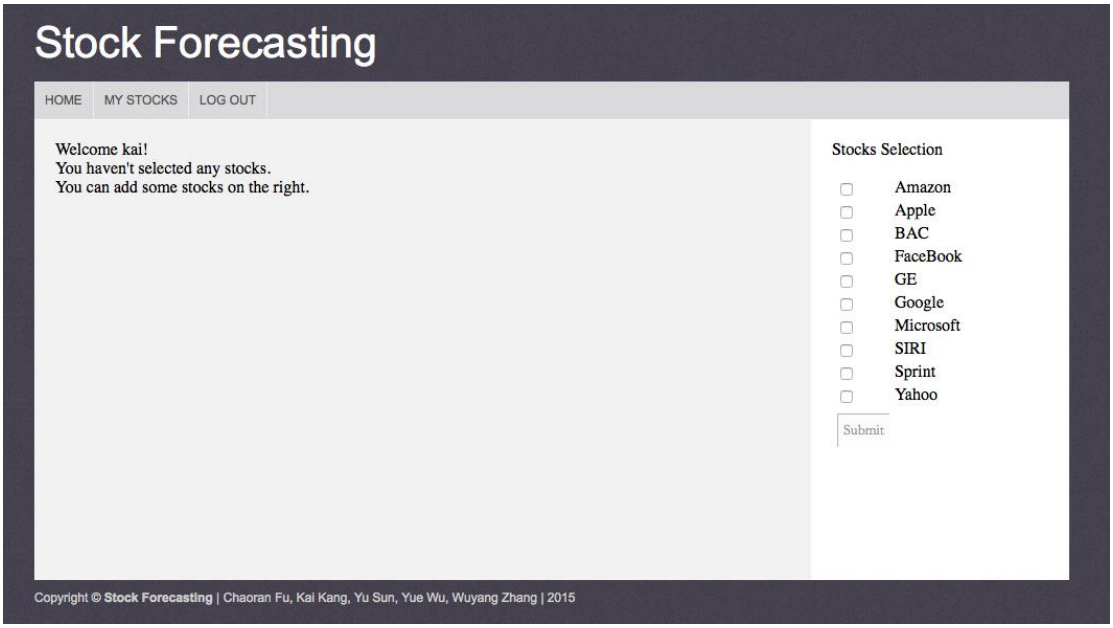

Figure 7-4. Initial Personal Page

After selection and submission, our personal page will display the information of the selected stocks. As shown in Figure 5 Every stock's current price, prediction price

and buying suggestions will be clearly presented. Users can also click on the stock name to view current price chart and the History view for history data. Of course the user can modify the stock selection anytime he want.

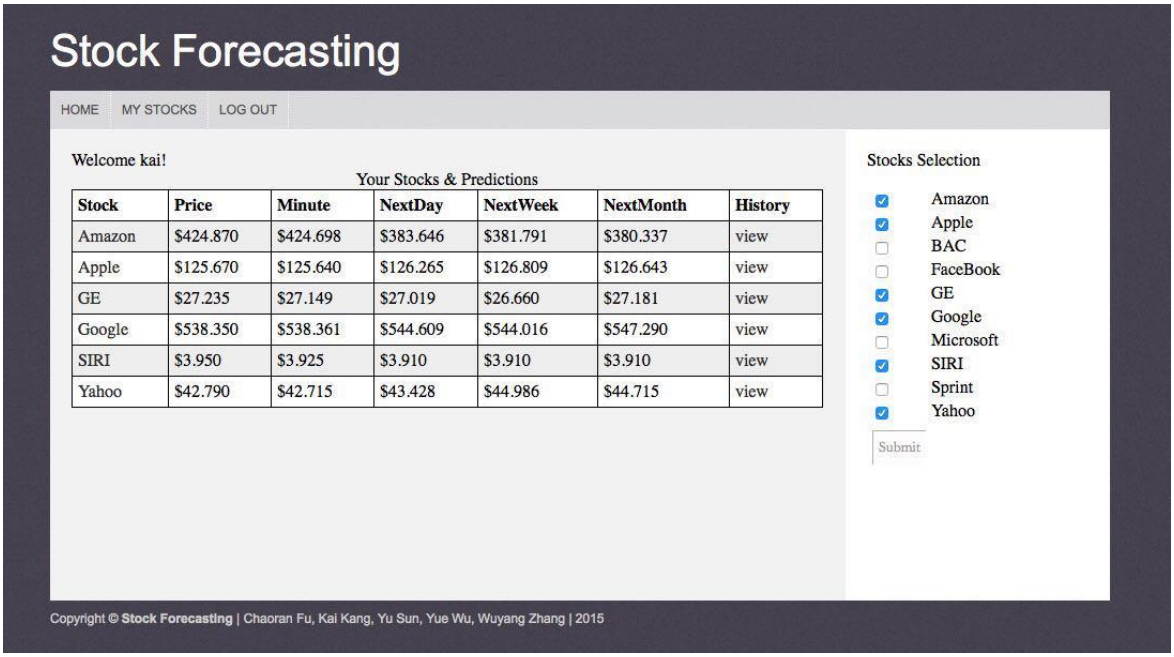

Figure 7-5. Stock Prediction

4. History Price Chart

Figure 7-6. History Price Chart

We use a history price chart to provide information about a stock's history price and its indicators over one year. The history price chart contains five parts. The main part is the candlestick chart located at the up right corner. Every candlestick presents details about one-day stock price: opening price, closing price, ceiling price and also the floor price. Different color is used to separate the price going up from the other way around. From the candlestick chart, we can clearly see the price trend corresponding with two of our indicators: SMA and EMA. The values of price and two indicators are shown on the top and the user can also click on the indicators' color icon to hide the indicator curves for a better look. Under our main chart, there are two more charts showing three other indicators and the volume. Using different type of charts to show different data gives user the clear and straightforward look.

At the very bottom lays the significant part of the whole chart: the timeline scrolling bar. Move the small paddle to select any time period you want. You can even move the whole time period by dragging the selected period left or right. There is also a time period selector located at the left side of the chart. User can type in the exact dates they want or quickly zoom in to the given time period with one click on the period button.

5. Current Price Chart

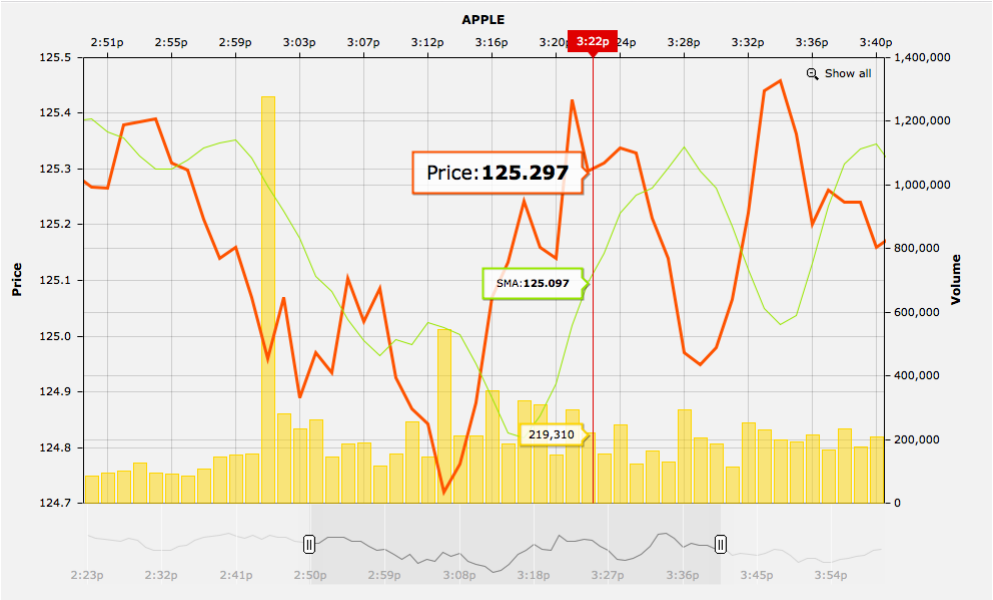

Figure 7- 7. Current Price Chart

We also provide stock current price chart as shown in Figure 7. It contains a stock's most recent one-day price and volume accurate to every minute. As you move the mouse inside the chart, there will be a vertical ruler following the mouse cursor highlighting the exact time on the time axis. The text balloons will also move along with the vertical ruler showing the specific value of price, volume and the SMA indicator. As in the history price chart, the timeline scrolling bar at the bottom allows users to move or zoom into any time period as they wish.

7.2 IOS Interface Design

We also implemented our stock forecasting system on IOS platform so as to satisfy the requirements of more clients. The tendency that people have more chances of using mobile phones compared to computers is obvious. Therefore, implementing our system on IOS is necessary.

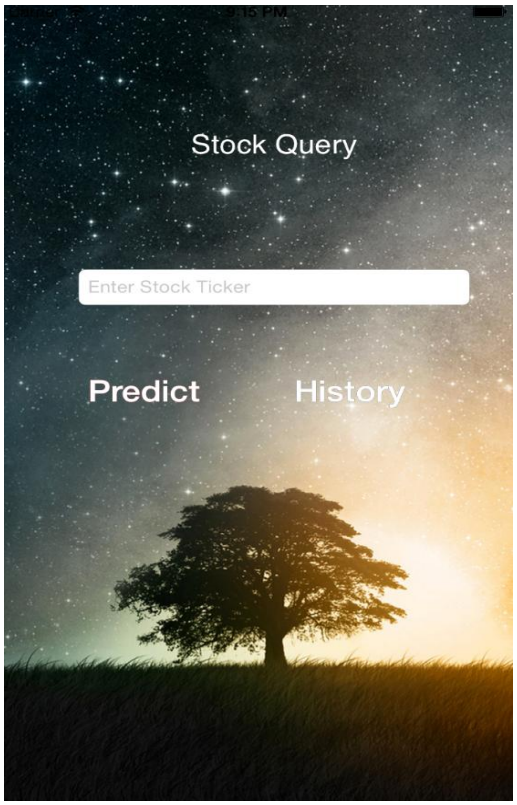

Fig 7-8 Stock Forecasting main interface

The figure 1.1 is the interface of stock forecasting on IOS platform. It divides the stock system into two functions. One can enter a stock ticker and then query related stock information. The first is to view present various prices of a stock including open, close, high, low, volume, and also predict price in its next day.

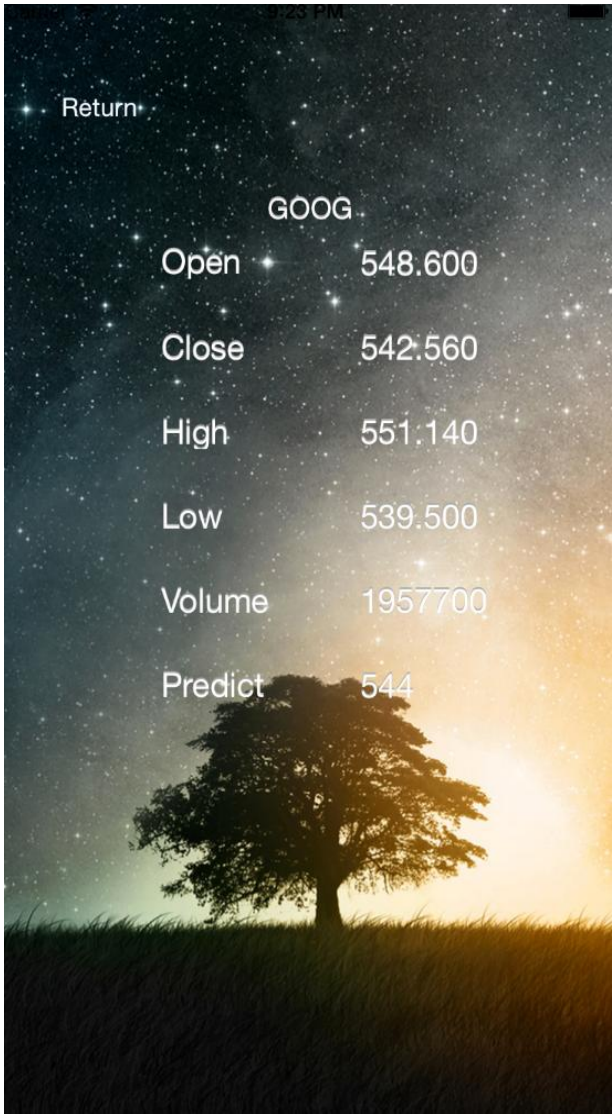

Fig 7-9 Predict Interface

Fig 7-10 AAPL History Price Graph.

Through using the predicting function people know precise immediate prices of a given stock and decide whether buy or sell shares according to it's predict price in a next day.

The second function is to view the graph of the historical close price of a given stock in recent three months. For example if we enter query keyword AAPL in the main interface, then history interface will show the graph of close prices of apple stock. Through this graph one can precisely know the changing tendency of a given stock so as to make proper decisions on this stock.

7.3 Stock Trading System Design

Base on the programming finance project, we design the stock trading system. Here's the main interface of the system.

Fig 7-11 Main Interface

In this system, users can view the portfolio, the stock price, deal history, aslo they can buy or sell share.

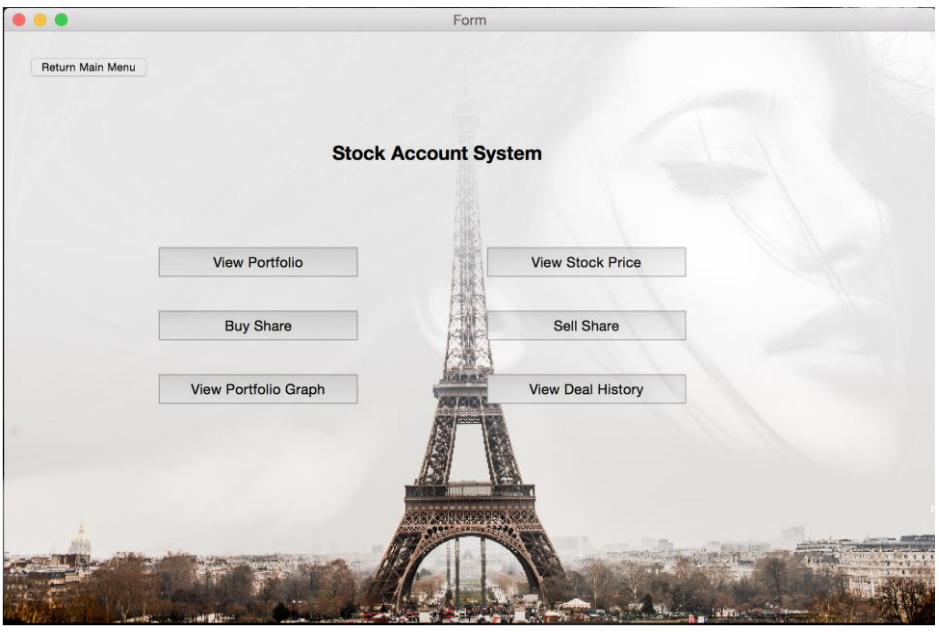

Fig 7-12 System Feature

We take the “Buy share” feature for instance:

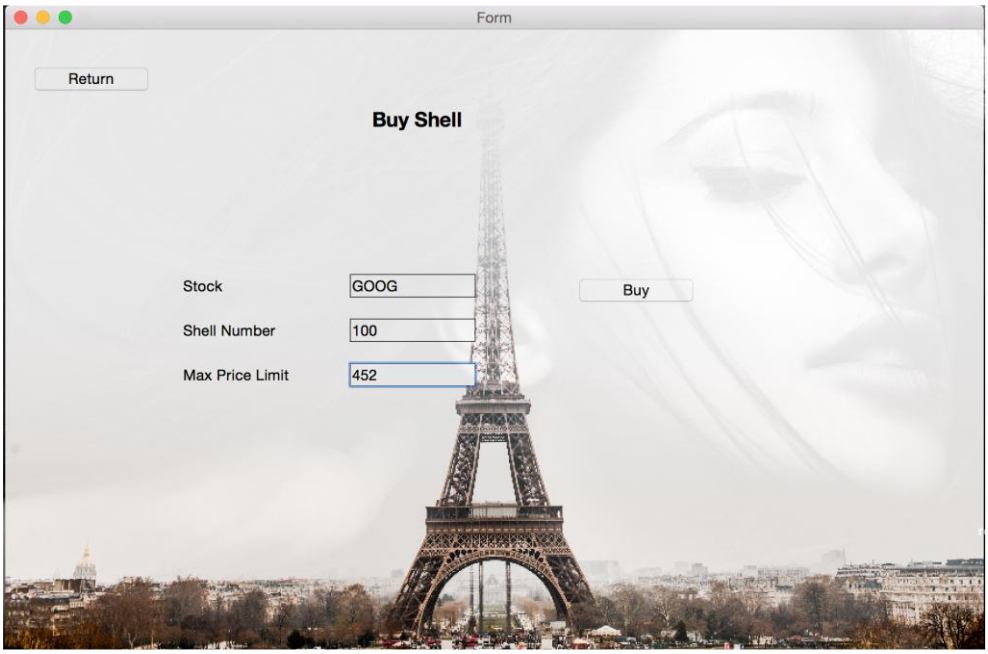

Fig 7-13 Buy Share

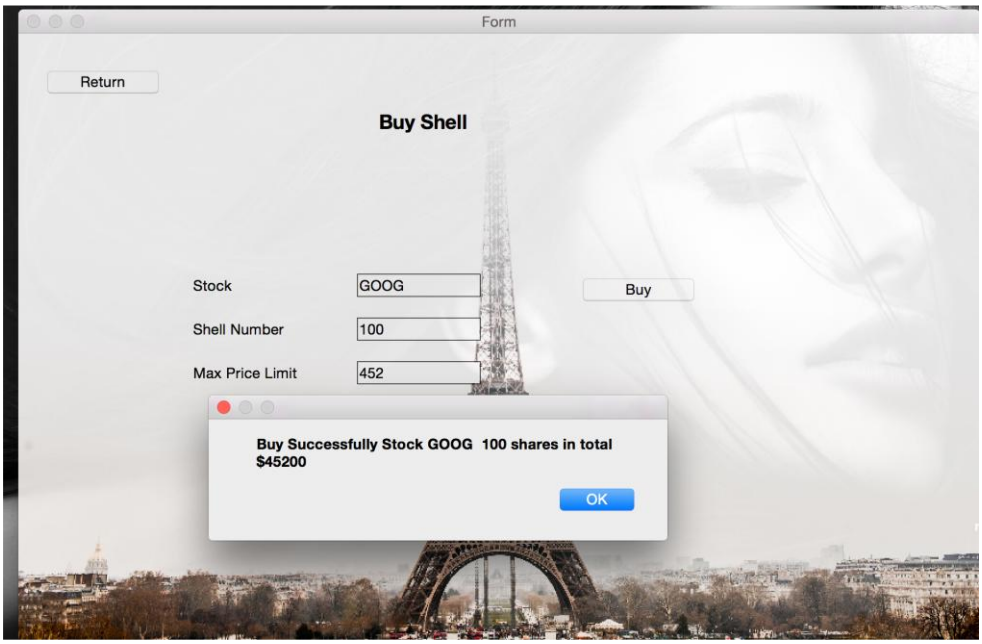

Fig7-14 Show Results

8. Implementation and Results

To get start, we need to implement to methods: downloading current price and history price, and store them into the database.

8.1 Creating Database and Table

To create a database, it's very convenient to do it via phpmyadmin, and we create a database called the StocketForcasting. The next step is to create tables to store the data. Consider there are a lot of tables, we create the database through PHP scripts instead of create them through phpmyadmin. The Following PHP sentence will create a table called CurrentPrice in the database:

```
$sql = "CREATE TABLE IF NOT EXISTS CurrentPrice
      (
        ID INT(6)    NOT NULL AUTO_INCREMENT PRIMARY KEY,
        Stocks VARCHAR(15),
        Price DECIMAL(10,3),
        Volume int(10),
        VolumeChange int(10),
        Date VARCHAR(20) DEFAULT '0',
        Time Varchar(20) Default '0'
              )ENGINE=MyISAM";
```

As we can see from above, this table use to store data related to stocks current price information. The data include stocks name, real time price, the date and time, accumulative total volume per day and the volume change in a short period. We choose to use the engine MyISAM instead of InnoDB because InnoDB has some problem in

display the whole information—though it's not effect the result.

The next table use to store information related to history price from Google.

```
$sql = "CREATE TABLE IF NOT EXISTS Google
    (
      ID INT(6)    NOT NULL AUTO_INCREMENT PRIMARY KEY,
      Stocks VARCHAR(15),
      Open    DECIMAL(10,3),
      High DECIMAL(10,3),
      Low DECIMAL(10,3),
      Close DECIMAL(10,3),
      Volume INT(10),
      Date VARCHAR(20) DEFAULT '0'
      )ENGINE=MyISAM";
```

This table store the stocks name, open price, high price, low price, close price and the volume and date. The other table to obtain history price have the same structure except different company names.

Below figures are the structure of the database, current price table and history table with real data obtain from Yahoo Finance.

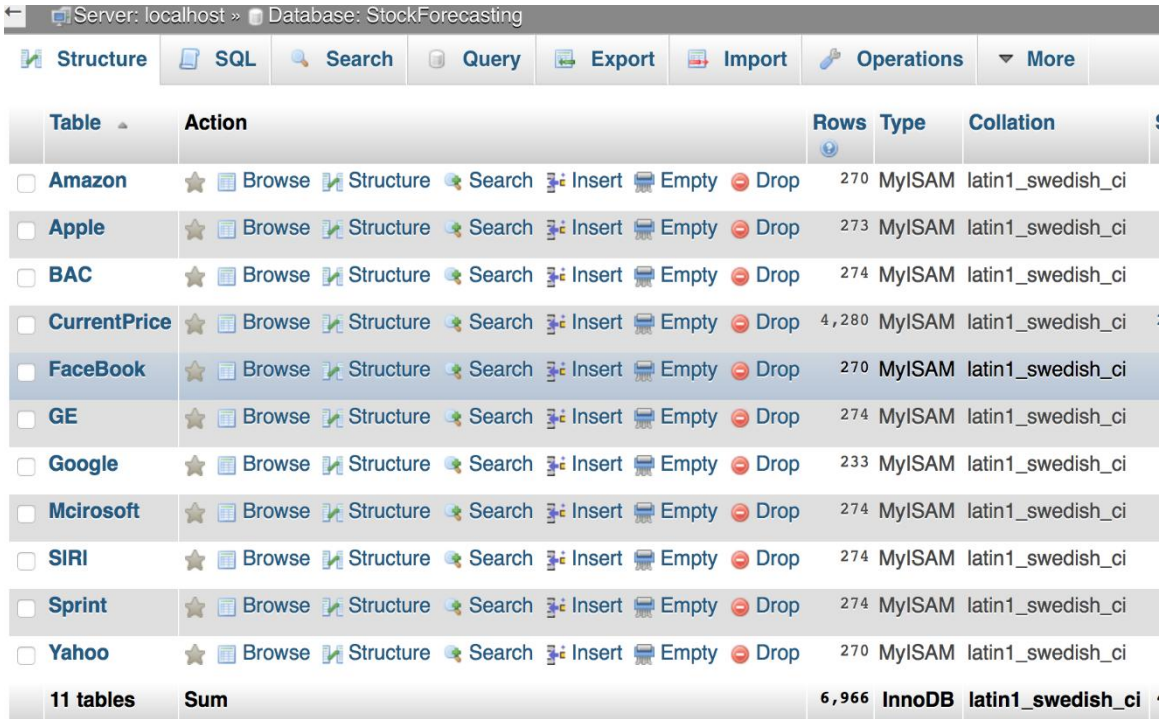

Table	Action	Rows	Type	Collation
Amazon	Browse Structure Search Insert Empty Drop	270	MyISAM	latin1_swedish_ci
Apple	Browse Structure Search Insert Empty Drop	273	MyISAM	latin1_swedish_ci
BAC	Browse Structure Search Insert Empty Drop	274	MyISAM	latin1_swedish_ci
CurrentPrice	Browse Structure Search Insert Empty Drop	4,280	MyISAM	latin1_swedish_ci
FaceBook	Browse Structure Search Insert Empty Drop	270	MyISAM	latin1_swedish_ci
GE	Browse Structure Search Insert Empty Drop	274	MyISAM	latin1_swedish_ci
Google	Browse Structure Search Insert Empty Drop	233	MyISAM	latin1_swedish_ci
Mcirosoft	Browse Structure Search Insert Empty Drop	274	MyISAM	latin1_swedish_ci
SIRI	Browse Structure Search Insert Empty Drop	274	MyISAM	latin1_swedish_ci
Sprint	Browse Structure Search Insert Empty Drop	274	MyISAM	latin1_swedish_ci
Yahoo	Browse Structure Search Insert Empty Drop	270	MyISAM	latin1_swedish_ci
11 tables	Sum	6,966	InnoDB	latin1_swedish_ci

Figure 8-1 Database structure

Options

	ID	Stocks	Open	High	Low	Close	Volume	Date
Edit Copy Delete	1	GOOG	554.240	564.710	552.900	558.400	2344200	2015-02-27
Edit Copy Delete	2	GOOG	543.210	556.140	541.500	555.480	2305200	2015-02-26
Edit Copy Delete	3	GOOG	535.900	546.220	535.450	543.870	1821000	2015-02-25
Edit Copy Delete	4	GOOG	530.000	536.790	528.250	536.090	1002300	2015-02-24
Edit Copy Delete	5	GOOG	536.050	536.440	529.410	531.910	1453900	2015-02-23
Edit Copy Delete	6	GOOG	543.130	543.750	535.800	538.950	1440400	2015-02-20
Edit Copy Delete	7	GOOG	538.040	543.110	538.010	542.870	986400	2015-02-19
Edit Copy Delete	8	GOOG	541.400	545.490	537.510	539.700	1449100	2015-02-18
Edit Copy Delete	9	GOOG	546.830	550.000	541.090	542.840	1612400	2015-02-17
Edit Copy Delete	10	GOOG	543.350	549.910	543.130	549.010	1895100	2015-02-13
Edit Copy Delete	11	GOOG	537.250	544.820	534.670	542.930	1615800	2015-02-12
Edit Copy Delete	12	GOOG	535.300	538.450	533.380	535.970	1374000	2015-02-11
Edit Copy Delete	13	GOOG	529.300	537.700	526.920	536.940	1745100	2015-02-10
Edit Copy Delete	14	GOOG	528.000	532.000	526.020	527.830	1264300	2015-02-09
Edit Copy Delete	15	GOOG	527.640	537.200	526.410	531.000	1744600	2015-02-06
Edit Copy Delete	16	GOOG	523.790	528.500	522.090	527.580	1844700	2015-02-05
Edit Copy Delete	17	GOOG	529.240	532.670	521.270	522.760	1659100	2015-02-04
Edit Copy Delete	18	GOOG	528.000	533.400	523.260	529.240	2033100	2015-02-03
Edit Copy Delete	19	GOOG	531.730	533.000	518.550	528.480	2842000	2015-02-02
Edit Copy Delete	20	GOOG	515.860	539.870	515.520	534.520	5591000	2015-01-30

Figure 8-2 History price structure

+ Options

	ID	Stocks	Price	Volume	VolumeChange	Date	Time
Edit Copy Delete	1	AMZN	384.610	2805	2805	3/3/2015	4:00pm
Edit Copy Delete	2	AAPL	129.360	30638	30638	3/3/2015	4:00pm
Edit Copy Delete	3	FB	79.600	10737	10737	3/3/2015	4:00pm
Edit Copy Delete	4	GOOG	573.640	359	359	3/3/2015	4:00pm
Edit Copy Delete	5	YHOO	42.620	35024	35024	3/3/2015	4:00pm
Edit Copy Delete	6	AMZN	384.610	2805	0	3/3/2015	4:00pm
Edit Copy Delete	7	AAPL	129.360	30638	300	3/3/2015	4:00pm
Edit Copy Delete	8	FB	79.600	10737	500	3/3/2015	4:00pm
Edit Copy Delete	9	GOOG	573.640	359	0	3/3/2015	4:00pm
Edit Copy Delete	10	YHOO	42.620	35224	200	3/3/2015	4:00pm
Edit Copy Delete	11	AMZN	384.610	2805	0	3/3/2015	4:00pm
Edit Copy Delete	12	AAPL	129.360	30638	0	3/3/2015	4:00pm
Edit Copy Delete	13	FB	79.600	10737	0	3/3/2015	4:00pm
Edit Copy Delete	14	GOOG	573.640	359	0	3/3/2015	4:00pm
Edit Copy Delete	15	YHOO	42.620	35224	200	3/3/2015	4:00pm
Edit Copy Delete	16	AMZN	384.610	2805	0	3/3/2015	4:00pm
Edit Copy Delete	17	AAPL	129.360	30638	0	3/3/2015	4:00pm
Edit Copy Delete	18	FB	79.600	10737	0	3/3/2015	4:00pm
Edit Copy Delete	19	GOOG	573.640	359	0	3/3/2015	4:00pm
Edit Copy Delete	20	YHOO	42.620	35924	700	3/3/2015	4:00pm

Figure 8-3 Current Price Structure

Reference

[1]. Wikipedia, “Broker” [Online]. Available: http://en.wikipedia.org/wiki/Broker

[2]. Wikipedia, “Stock market prediction” http://en.wikipedia.org/wiki/Stock_market_prediction

[3] SMA http://www.investopedia.com/terms/s/sma.asp

[4]SMAhttp://www.onlinetradingconcepts.com/TechnicalAnalysis/MASimple.html

[5] MACD http://www.investopedia.com/terms/m/macd.asp

[6]MFIhttp://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:money_flow_index_mfi

[7] RSI http://en.wikipedia.org/wiki/Relative_strength_index

[8]ADLhttp://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:accumulation_distribution_line

[9]OBVhttp://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:on_balance_volume_obv

[10]BBhttp://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:bollinger_bands

[11] Svmlib http://www.csie.ntu.edu.tw/~cjlin/libsvm/