CSC4350/6350

Spring 2017

A stock trading software. The price is real-time and the exchange is virtual.

Virtual Stock

BATMAN

2/6/2017

Mengyuan Zhu, Sungjae Kim, Sharon Kim, Jakub Pietrasik, Hyeun Kang

**Contents**

Contents

[Horizontal Prototype 2](#_Toc475363770)

[Problem Statements 6](#_Toc475363771)

[Use cases and Interaction Diagrams 8](#_Toc475363772)

[Function Point Cost Analysis 8](#_Toc475363773)

[Database to be used 10](#_Toc475363774)

[Work Structure Document 10](#_Toc475363775)

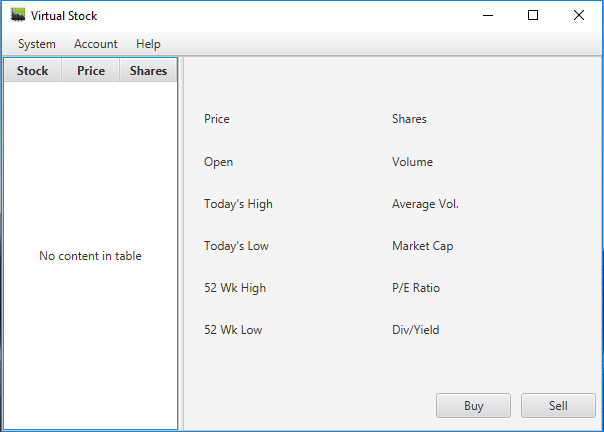
[Dictionary 10](#_Toc475363776)

[Gantt Chart 11](#_Toc475363777)

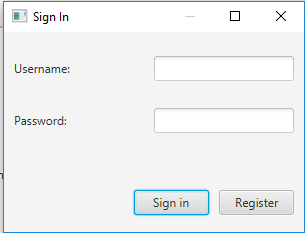
# Horizontal Prototype

The following pages comprise Virtual Stock’s horizontal prototype. The layout may be

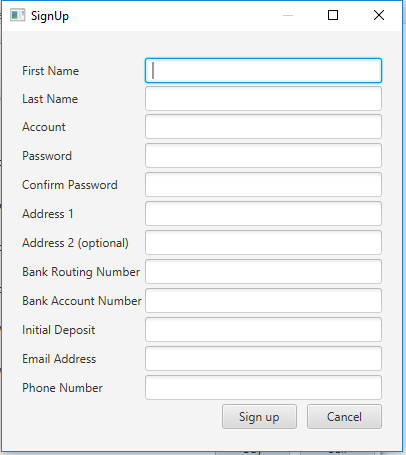
graphically represented as:



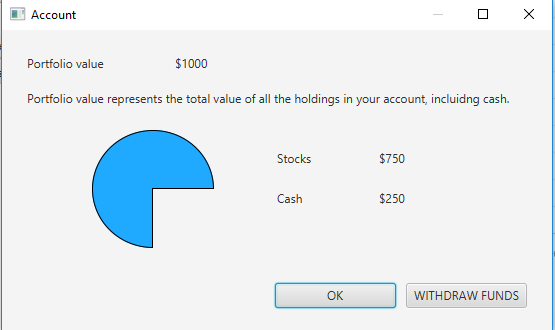
Main Window



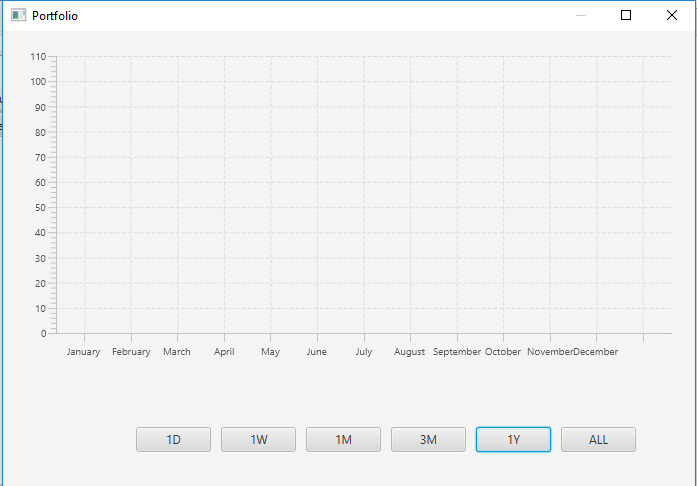
Sign In



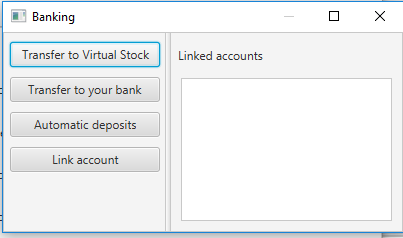
Sign up



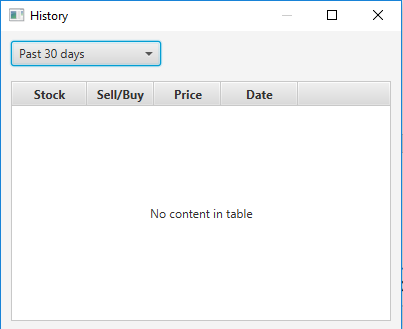
Account



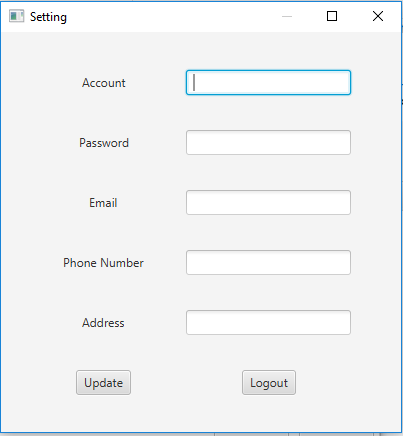
Portfolio



Banking



History



Settings

# Problem Statements

**Virtual Stock-1.0 Introduction**

Batman shall develop a stock trading software with a user-friendly GUI. The software is PC based. The price is real-time from the Internet while the exchange is virtual. It shall be designed for beginners who want to learn stock trading.

**Virtual Stock-2.0 User Registration**

The system shall provide a window for user to register an account in Batman. Each user shall have his/her own account after registration. Users shall provide the following information: First Name, Last name, Address, Birthday, SSN, Bank account routing number and account number, Email address, Nationality and Agreement Form.

**Virtual Stock-2.1**

If a same account has been built up, the system shall let user know and ask the user to refill the form for registration

**Virtual Stock-3.0 User Log In**

The system shall have a window for user to log in.

**Virtual Stock-3.1**

The window shall provide a user name input, a password input, a forget password button, a sign-in button and a sign-up button.

**Virtual Stock-3.2**

The system shall let user into the “Portfolio” interface if the user type the right user name and password.

**Virtual Stock-3.3**

If the user failed to log in to the system, the system shall ask the user to refill the password.

**Virtual Stock-3.4**

The system shall provide a pin number for quick log in.

**Virtual Stock-4.0 Portfolio**

The system shall have a portfolio interface.

**Virtual Stock-4.1**

In this interface, the system shall show a chart of user’s balance chart of today, a total balance value, a breaking news of today, stocks that a user keeps and his/her Shares, and a Watch list.

**Virtual Stock-4.2**

A search button shall also be shown in the corner for user to search a specific stock.

**Virtual Stock-4.3**

The color of the GUI shall be green if the user’s balance goes up and be red if the user’s balance goes down.

**Virtual Stock-5.0 Account**

The system shall provide a window to show user’s account.

**Virtual Stock-5.1**

The account shall show the Total balance, Stocks balance and Cash balance.

**Virtual Stock-6.0 Banking**

The system shall give a function for user to transfer money to a bank or to the Batman app.

**Virtual Stock-6.1**

The linked accounts shall be shown in the bottom.

**Virtual Stock-6.2**

The system shall provide a bank account for user to deposit money.

**Virtual Stock-7.0 History**

The system shall provide a list of history of user’s trading log. The list shows the deposit form a bank, the buying and selling with the corresponding money.

**Virtual Stock-7.1**

The system shall also show the date.

**Virtual Stock-7.2**

The system shall allow the user to see the history from/to a specific date.

**Virtual Stock-8.0 Setting**

The system shall have a setting interface for user to reset pin number and update user information including name, password, email, phone number and address.

**Virtual Stock-8.1**

It shall also have a log out button for user to log out.

**Requirements Traceability Matrix**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Entry#** | **Para#** | **Requirements Traceability Matrix** | **Type** | **Use Case Name** |
| 1 | 1 | Batman shall develop a stock trading software with a user-friendly GUI. | SW, HW |  |
| 2 | 2 | The system shall provide a window for user to register an account in Batman. | SW |  |
| 3 | 2.1 | the system shall let user know and ask the user to refill the form for registration. | SW |  |
| 4 | 3 | The system shall have a window for user to log in. | SW |  |
| 5 | 3.1 | If the user failed to log in to the system, the system shall ask the user to refill the password. | SW |  |
| 6 | 3.2 | The system shall provide a pin number for quick log in. | NTH |  |
| 7 | 4 | The system shall have a portfolio interface. | SW |  |
| 8 | 4.1 | The color of the GUI shall be green if the user’s balance goes up and be red if the user’s balance goes down. | NTH |  |
| 9 | 5 | The system shall provide a window to show user’s account. | SW |  |
| 10 | 6 | The system shall give a function for user to transfer money to a bank or to the Batman app. | SW |  |
| 11 | 6.1 | The system shall provide a bank account for user to deposit money. | SW |  |
| 12 | 7 | The system shall provide a list of history of user’s trading log. | SW |  |
| 13 | 7.1 | The system shall allow the user to see the history from/to a specific date. | NTH |  |
| 14 | 8 | The system shall have a setting interface for user to reset pin number and update user information including name, password, email, phone number and address. | SW |  |

# Use cases and Interaction Diagrams

Use cases here

# Function Point Cost Analysis

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Measurement Parameter** | **Count** | **Simple** | **Average** | **Complex** |  |
|  | x | 3 | 4 | 6 | = |
|  | x | 4 | 5 | 7 | = |
|  | x | 3 | 4 | 6 | = |
|  | x | 7 | 10 | 15 | = |
|  | x | 5 | 7 | 10 | = |
| Count = Total   --------------------------------------------------------------------------------------- | | | | | |

*Note*. By clicking on the buttons above more information about the measurement parameters will be available.

Rate each factor (Fi, i=1 to14) on a scale of 0 to 5:

|  |  |
| --- | --- |
| F1.   Does the system require reliable backup and recovery? |  |
| F2.   Are data communications required? |  |
| F3.   Are there distributed processing functions? |  |
| F4.   Is performance critical? |  |
| F5.   Will the system run in a existing, heavily utilized operational environment? |  |
| F6.   Does the system require on-line data entry? |  |
| F7.   Does the on-line data entry require the input transaction to be built over multiple screens or operations? |  |
| F8.   Are the master files updated on-line? |  |
| F9.  Are the inputs, outputs, files or inquiries complex? |  |
| F10. Is the internal processing complex? |  |
| F11. Is the code designed to be reusable? |  |
| F12. Are conversion and installation included in the design? |  |
| F13. Is the system designed for multiple installations in different organizations? |  |
| F14. Is the application designed to facilitate change and ease of use by the user? |  |



***Result****.*  According to the input your project has:  

# Database to be used

XML

# Work Structure Document

Group name: **Batman**

|  |  |
| --- | --- |
| **Name** | **Tasks** |
| Mengyuan Zhu | Team Coordinator  Documents handler  Java coder  Problem Statement  Requirements Traceabillity Matrix  Dictionary  A horizontal prototype of the software to be developed  Use cases and Interaction Diagrams - example as per given in class  Database to be used |
| Sungjae Kim | Finalize code documentation  Java coder  Gannt Chart |
| Sharon Kim | User Guide  Function Point Cost Analysis.  Program tester  Rationale |
| Jakub Pietrasik | Program tester |
| Hyeun Kang | Program tester |

# Dictionary

**Portfolio:** In finance, a portfolio is a collection of investments held by an investment company, hedge fund, financial institution or individual.

**Broker:** A person who buys or sells an investment for you in exchange for a fee (a commission). Here is Tim’s favorite broker.

**Dividend:** this is a portion of a company’s earnings that is paid to shareholders, or people that own hat company’s stock, on a quarterly or annual basis. Not all company’s do this.

**Exchange:** An exchange is a place in which different investments are traded. The most well-known in the United States are the New York Stock Exchange and the Nasdaq.

**Quote:** Information on a stock’s latest trading price. This is sometimes delayed by 20 minutes unless you are using an actual broker trading platform.

**Volume:** The number of shares of stock traded during a particular time period, normally measured in average daily trading volume.

**Yield:** This usually refers to the measure of the return on an investment that is received from the payment of a dividend. This is determined by dividing the annual dividend amount by the price paid for the stock. If you bought stock XYZ for $40-a-share and it pays a $1.00-per-year dividend, you have a “yield” of 2.5%.

**JDK:** The Java Development Kit (JDK) is an implementation of either one of the Java Platform, Standard Edition; Java Platform, Enterprise Edition or Java Platform, Micro Edition platforms released by Oracle Corporation in the form of a binary product aimed at Java developers on Solaris, Linux, Mac OS X or Windows.

**GUI:** The graphical user interface is a type of user interface that allows users to interact with electronic devices through graphical icons and visual indicators such as secondary notation, instead of text-based user interfaces, typed command labels or text navigation.

**HTTP:** The Hypertext Transfer Protocol (HTTP) is an application protocol for distributed, collaborative, and hypermedia information systems. HTTP is the foundation of data communication for the World Wide Web.

**XML:** In computing, Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The W3C's XML 1.0 Specification and several other related specifications-all of them free open standards—define XML.

# Gantt Chart