

*Software Engineering*

Smart Robo-Advisor

REQUIREMENTS DOCUMENT

**GROUP REPORT**

1801212963 Ye Juelin (叶珏琳)

1801212938 Wang Rusi (王如思)

1801212920 Shi Yunxia (史云霞)

1801212932 Wang Bingquan (王柄荃)

1801212832 Shen Zimin (谌子民)

# CATALOG

[1. Preface 3](#_Toc3107222)

[2. Introduction 4](#_Toc3107223)

[3. Glossary 5](#_Toc3107224)

[4. User Requirement Definition 6](#_Toc3107225)

[5. System Architecture 8](#_Toc3107233)

[6. System requirements Specification 9](#_Toc3107234)

[6.1 System Use Cases 9](#_Toc3107235)

[6.2 Functional Requirement 9](#_Toc3107236)

[6.3 Non-functional requirements 14](#_Toc3107237)

[7. System Model 19](#_Toc3107238)

[8. System Evolution 22](#_Toc3107241)

[9. Appendices 24](#_Toc3107242)

# Preface

This is a requirement document for our first version Smart Robo-Advisor system. Our requirements document has a diverse set of users, ranging from the investors who are paying for the system to the engineers responsible for developing the software. Picture 1.1 shows possible users of the document and how they use it.

Use the requirements to understand how the system works.

Investors

Use the requirements to understand what system is to be developed.

System

Engineers

Use the requirements to develop validation tests for

the system.

System

Test engineers

Use the requirements to understand the system and the relationships between its parts.

System

Maintenance

Engineers

**Picture 1.1 – Users of the System**

# Introduction

Our system is a smart Robo-Advisor platform that intelligently provides a full range of customized financial portfolio advice for those who do not have access to professional advisory services and lack of professional wealth investment knowledge and experience.

The system contains functions including customer risk assessment, quantitative strategies building, asset allocation, portfolio selection through back-test results, portfolio tracking and timely advice sending. The core strategy algorithm will cover all areas of classical quantitative trading models, e.g., multi-factor model, CTA model, statistical arbitrage model, industry rotation strategy, etc. The investment targets will include stocks, bonds and futures. Since customers’ risk taking level assessment is critical, the system will help users to identify their investment goals by analyzing factors such as investor’s age, salary, family status, asset liquidity and investment duration. And by combining the expected return rate, risk-taking level, periodic characteristics and correlation of different financial assets, this will system finally construct the best portfolio that matches the investment objectives of the users and give investment advice. These features are all managed automatically at a much lower cost, compared with traditional financial advisor service.

Apart from the robo-advisor function, this system also provides additional valuable services, such as strategy sharing forum. Besides, users can sign up personal center to have a comprehensive monitoring and analysis of their historical portfolio performance directly on the platform.

# Glossary

This part lists the definitions of the glossary used in the document in Table 3.1, so that everyone who looks at the document can have a basic understanding of each term in the document.

|  |  |
| --- | --- |
| Glossary | Definition |
| Quantitative Trading | Use computer technology to select strategies from a large historical data that can generate excess returns with a variety of ‘high probability’ events. |
| Artificial Intelligence | [Intelligence](https://en.wikipedia.org/wiki/Intelligence) demonstrated by [machines](https://en.wikipedia.org/wiki/Machine). |
| Asset Allocation | Allocate funds in different asset classes based on investment demand. |
| Back-test | Test the performance of strategy with historical data. |
| Multi-factor | Use a series of factors as the stock selection criteria and select stocks with higher scores to build a portfolio. |
| CTA | Timing strategy in the futures market. |
| Statistical Arbitrage | Find out the two assets with similar trend, buy low and sell high. |

**Table 3.1 - Glossary**

# User Requirement Definition

The user requirements for the Smart Robo-Advisor are specified in natural language precisely.

## 4.1 Risk Lever Assessment

1. Build intelligent risk assessment model.
2. Users fill in questionnaires related to individual financial conditions.
3. Measure the user's real risk-taking level.
4. The result should guide the system to co-determine the final investment strategy as one of the key parameters.

## 4.2 Portfolio Management Advisor

1. Based on users’ input, e.g., risk preference, expected rate of return, frequency of changing portfolio, investment time duration and other important indicators, the system should give users investment advices on how to allocate their asset on different investment targets.
2. Display the back-test result graph of recommended investment strategies and relevant statistical information based on historical trading data.
3. Send the portfolio investing plan and investing instructions to users according to their expected trading frequency.

## 4.3 Personal Center

1. Users can query and modify their personal information.
2. Users can query the back-test results of previous investment strategies to assess the performance of the portfolio based on historical data.

## 4.4 Strategy Book

Users can view existing investment strategy models, the back-test result, as well as some key statistical performance indicators such as maximum withdrawal, annual return rate, Sharpe rate etc.

## 4.5 Strategy Sharing Forum

1. Users can share their own strategic ideas and experience
2. Users can give compliment to other strategies ideas that they agree with.

## 4.6 Product requirements

The Smart Robo-Advisor shall be available to all users every day. Downtime shall not exceed five seconds in any one day.

## 4.7 External requirements

1. The system shall implement patient privacy provisions as set out in HStan-03-2006-priv.
2. Regulators requires smart clients to obtain a securities investment advisory license.
3. Regulators require investors to fully disclose information, report the main parameters of the model and the main logic of asset allocation.

# System Architecture

Strategy

Sharing Forum

Strategy

Book

Personal Center

Portfolio Management Advisor

Risk Level Assessment

Architecture

Risk Level & System Feedback

Risk Parameter

Back-test

Asset allocation

Develop Strategy

Investing Advice

Previous Service

Personal Information

Existing Strategy & Result

Give Compliment

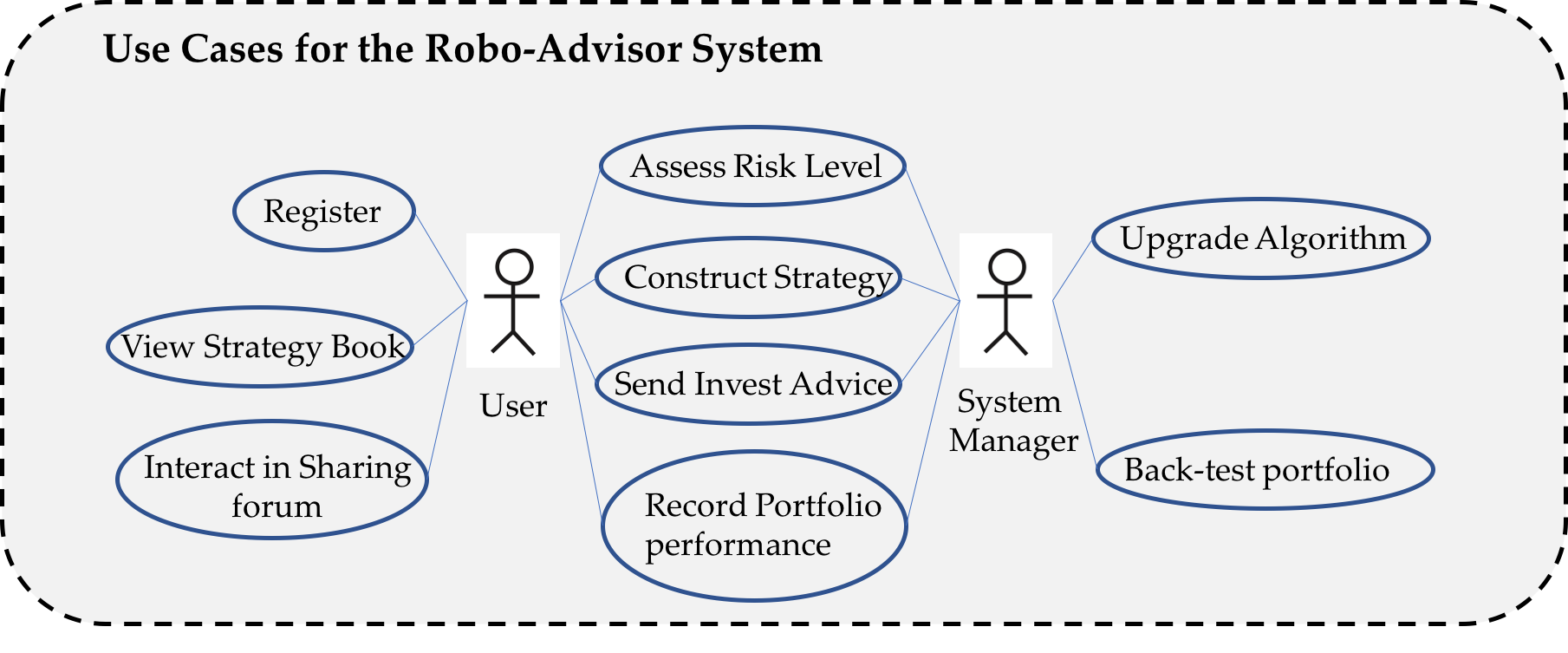
Share Strategic ideas

**Picture 5.1 - System Architecture**

# System requirements Specification

## 6.1 System Use Cases

The system use cases are shown in Picture 6.1.

****

**Picture 6.1- System Use Cases**

## 6.2 Functional Requirement

**1. Core function: Risk Level Assessment**

|  |
| --- |
| Function |
| Risk Level Assessment |
| **Description** |
| * For experienced users who know their risk-taking level, they choose the correspond risk preference value. * For users who have vague risk tolerance level self-recognition, the system provides a questionnaire and then measures the user's real risk-taking level according to the intelligent risk assessment model. |
| **Inputs** |
| * Directly input the risk level. * Using *Individual Investor Risk Tolerance Assessment Questionnaire* to input users’ financial conditions: income flow, age, investment experience, asset liquidity, financial knowledge, investment target preference. |
| **Requirements flow** |
|  |
| **Outputs** |
| * To system: the user’s actual real risk-taking level. * To the user: Readable risk assessment result feedback. |
| **Destination** |
| The risk-taking level will be an important parameter input in the decision process of asset category allocation and quantitative strategy model. |

**2. Core function: Portfolio Management Advisor**

|  |
| --- |
| **Function** |
| Portfolio Management Advisor |
| **Description** |
| Based on users’ input, including expected rate of return, frequency of changing portfolio, investment time duration and other important indicators, the system utilizes all kinds of strategy models to give users investment advices on how to allocate their asset on different investment targets, along with showing the back-test result. If the user accepts the customized strategy, the system further timely sends investment instruction as well as the performance, e.g., net value and accelerated return to the user. |
| **Inputs** |
| * User investment preference: investment amount, expected rate of return, frequency of changing portfolio, investment time duration * Whether accept the strategy or not. |
| **Requirements flow** |
| ../Pictures/图片%202.png |
| **Outputs** |
| Portfolio management strategy back-test performance:   * Visualized diagram * Back-test parameters: total yield, volatility and risk, maximum withdrawal, Sharp rate, etc. * Readable strategy instruction and result description |
| **Destination** |
| If the strategy is accepted by the user, it will be stored in system database. Further, according to the market dynamics, the corresponding investment managing advice and the performance record will be timely sent to the user. |

**3. Additional function: Personal Center**

|  |
| --- |
| **Function** |
| Personal Center |
| **Description** |
| * Users can query or modify their own related information in this function, such as user passwords, used strategies, risk history, and so on. |
| **Inputs** |
| * Username and password |
| **Requirements flow** |
|  |
| **Outputs** |
| * New password * Strategy information * Risk history |
| **Destination** |
| Give users a method to view and modify their personal information. |

**4. Additional function: Strategy Book**

|  |
| --- |
| **Function** |
| Strategy Book |
| **Description** |
| Users can see all the relevant statistics of all our quantitative investment strategies in this interface and invest on the promising quantitative strategies. |
| **Inputs** |
| None |
| **Outputs** |
| Information (introduction, back-test results, graphs) of strategies in our strategy book |
| **Destination** |
| Give the users a way to view the performance of our strategies visually and a channel to invest on our quantitative strategies. |

**5. Additional function: Strategy Sharing Forum**

|  |
| --- |
| **Function** |
| Strategy Sharing Forum |
| **Description** |
| In this module, users can share their own strategic ideas and experience, and they can also give compliment to other strategies ideas that they agree with. |
| **Inputs** |
| Sharing contents. |
| **Outputs** |
| Information uploaded by other users. |
| **Destination** |
| Build a community where users can develop sharing habits. At the same time, expand the channels for users to obtain information. |

## 6.3 Non-functional requirements

**1. Performance requirements:**

(1) Response time:

* In 95% of cases, the general time response time does not exceed 1 second, and during the peak period the response time does not exceed 3 seconds.
* The time required from the click to the next screen should not exceed 300 milliseconds.
* When the network is unblocked, the time required to dial-up to the GPRS network must not exceed 5 seconds.
* In the recommended configuration environment: the login response time is within 2 seconds, the refresh response time of the column is within 2 seconds, the response time of the entry page list is refreshed within 2 seconds, and the response time of the information entry is turned on within 1 second.

(2) Business volume:

* The system allows 50,000 users to be online at the same time.
* The system can simultaneously satisfy 10,000 user requests and provide browsing capabilities for 25,000 concurrent users.

(3) System capacity:

* The database can hold at least 100GB of data.

(4) Accuracy:

* The backtracking tracking error rate does not exceed 1%.
* The accuracy of the calculation is 4 digits after the decimal point.

(5) Resource usage rate:

* CPU usage <=50%.
* Memory usage <=50%.

**2. Security requirements:**

* Strict access control system. After the user is authenticated, he can only access the data within its scope of authority, and can only perform operations within its scope of authority.
* Different users have different identities and rights. They need to provide trusted authorization management services under the premise of authenticity of users, to protect data from illegal/over-authorized access and tampering, and to ensure the confidentiality and integrity of data.
* Provides operational log management and security auditing to track historical usage of the system.
* General can withstand malicious attacks from the Internet. Such as viruses (including Trojans) attacks, password guessing attacks, hacking, etc.
* At least 99% of attacks need to be detected within 10 seconds.

**3. Reliability requirements:**

* There is a prompt for the input, and the data has an inspection mechanism to prevent data anomalies.
* The system is robust and should be able to handle various abnormal situations that occur during system operation, such as human error, illegal data input, hardware device failure, etc. The system should be able to handle it correctly and avoid it properly.
* The probability that the service cannot be completed due to the failure of the software system is less than 5‰.
* The system is required to run 24 hours a day, and the total number of outages for continuous operation throughout the year cannot exceed 10 hours.
* The system defect rate has a maximum of 1 failure every 1,000 hours.
* In 1,000,000 operations, there is a maximum of one time when the system needs to be restarted.

**4. Compatibility requirements:**

* The system should support IOS, Android, Windows operating system;
* The average time to replace the system database is no more than 2 hours and no data loss is guaranteed.

**5. Data security needs:**

* Network delivery data should be encrypted. It is necessary to ensure that data is not sneaked, stolen or tampered with during collection, transmission and processing. Business data needs to be encrypted at the time of storage to ensure that it is not hackable.

**6. Ease of use needs:**

* Within 3 months of using the product, 60% of users should be able to use it to complete the portfolio configuration function within 1 minute, and the failure rate is controlled within one ten thousandth.
* 60% of users will realize that this is a smart investment system within 5 seconds of seeing the product for the first time.
* 80% of users receive asset allocation advice within 5 minutes after receiving a 2-hour system introduction training.

**7. Availability requirements:**

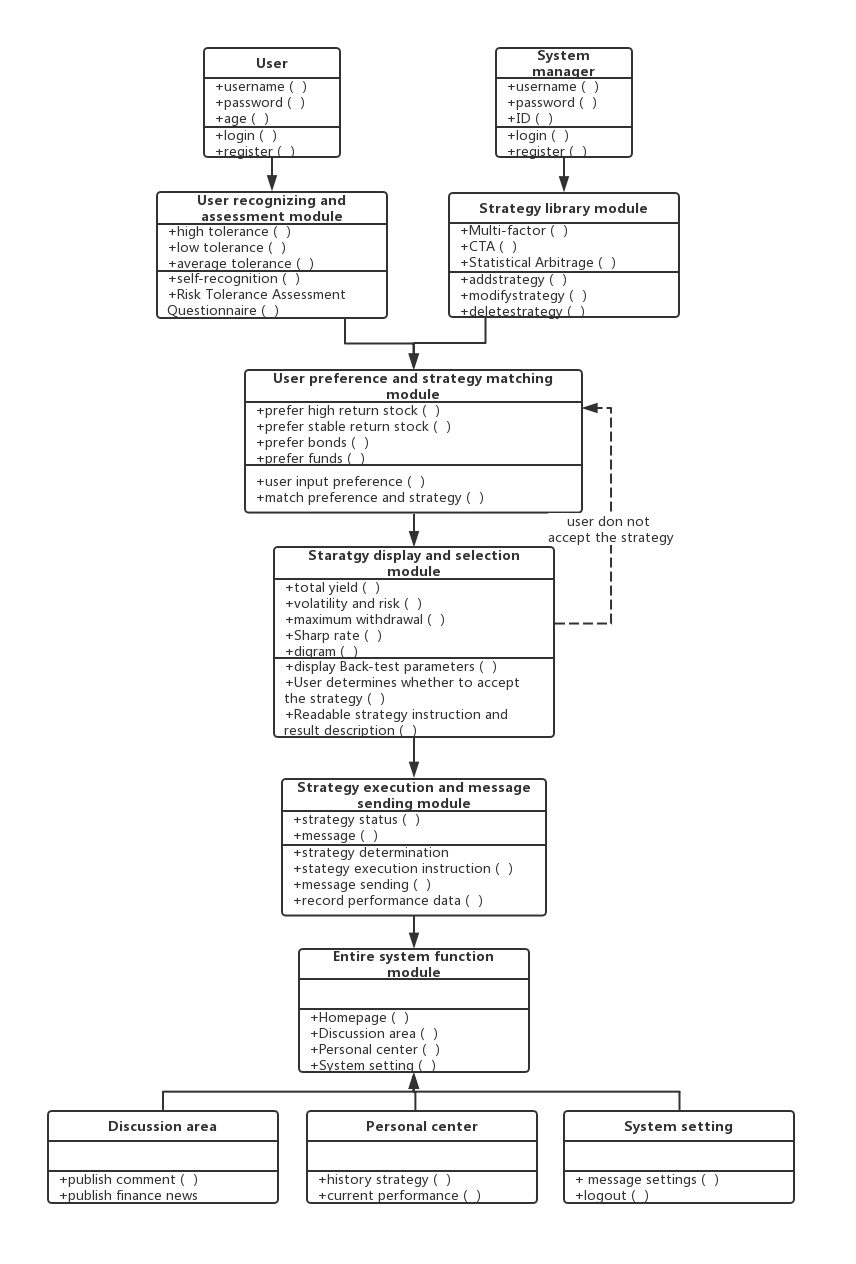
* Provides data backup and recovery functions to enable timely recovery and restore of data (provided by hardware and third-party software) when system data is lost due to system errors or other reasons or system data is corrupted.

**8. Maintainability requirements:**

* 90% of the BUG modification time does not exceed 1 working day, and the other does not exceed 2 working days.
* Any method of any object does not allow more than 200 lines of code.
* Installing the new version must keep all database content and all personal settings unchanged.
* The product must provide a tool that can track any database field.

# System Model

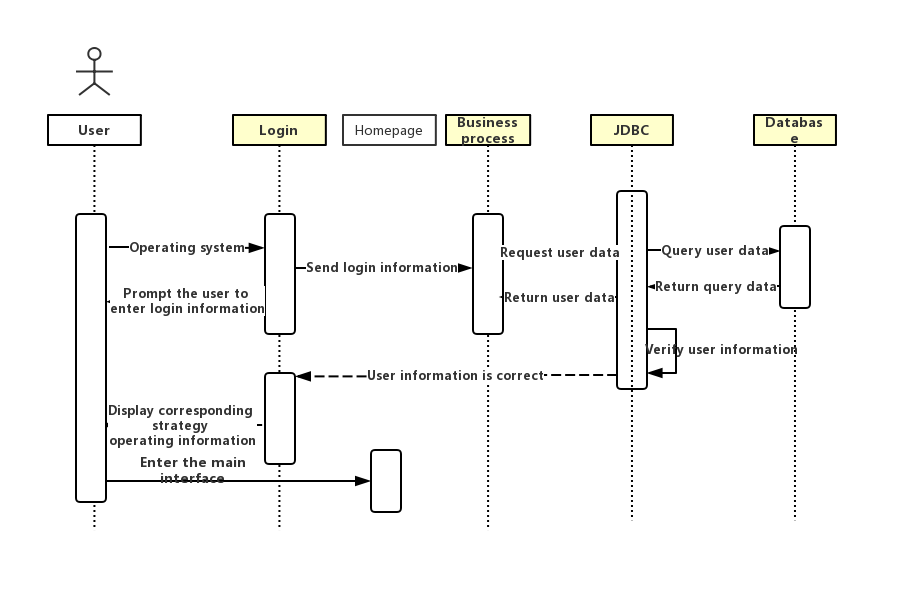
**7.1 System Overall Model Architecture**



**Picture 7.1 - System Overall Model**

## 7.2 System Timeline

Picture 7.2 describes the system operating timeline from the user coming in to the system, to the system giving back the strategy operating instruction.

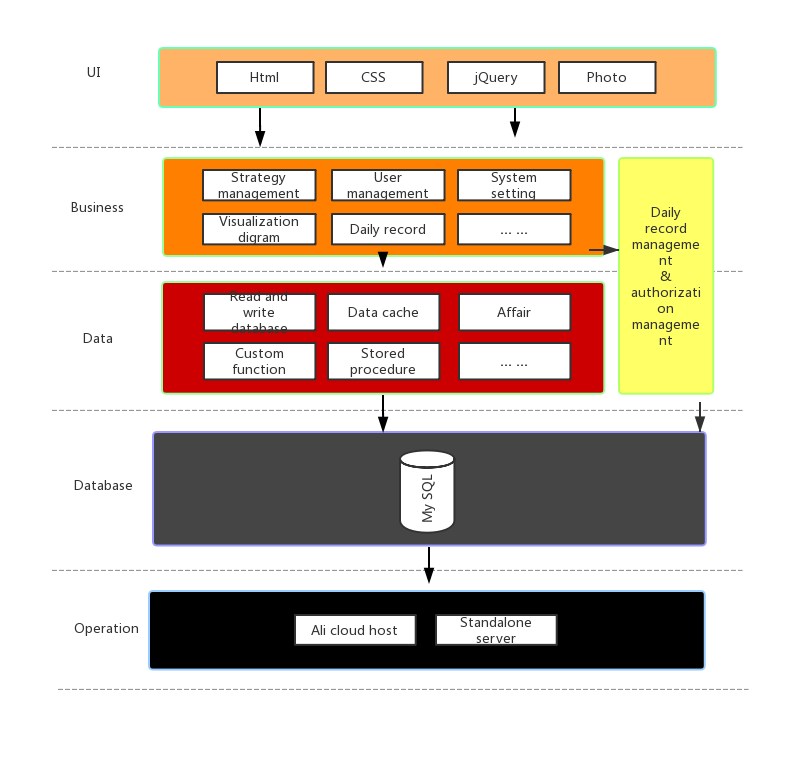


**Picture 7.2- System Operating Timeline**

## 7.3 System Structure

In this part, Picture 7.3 introduces the following part of the system and the relation and logic between them.

* UI design
* Business line
* Data line
* Database design
* Server design



**Picture 7.3- System Logic Structure**

# System Evolution

With the development of users' demand, our system has to change according to the new requirements.

1. New functions for users to design their own strategy. Our system is for people who are not professional in quantitative trading industry. It can help them know the knowledge and invest their money by quantitative strategies. With more use of our system, users could get familiar with quantitative strategies and they may want to design their own strategies. What's more, the users may talk about the strategies in the forum and develop some ideas about how to improve the strategies or develop new strategies. In these case, we need to provide users tools to build strategies such as API. When we are building the system, we should package every function well which make it easier for developers to change them to API.

2. Human services and special trading interface for high value consumers. The main consumers of our software are low value people whose asset are smaller than ￥500,000. The volume of these consumers are big so we can earn money by advertisement. However, with the more and more users come into our software the high value consumers are significant for profits. We should focus on these consumers and give them special services such as human services giving them more accurate investment suggestions. These users may want to trade automatically instead of trading by themselves according to the trade messages. We can cooperate security companies and build some special trading interfaces for these consumers. These consumers did not need to trade by themselves by paying more fees and we can divide these fees with security companies.

3. Automatically trading interface. According to the policy of securities regulatory commission, the trading interface for stocks will open soon. These means that quantitative trading strategies may buy or sell the stocks automatically through the interface. Therefore, we should reserve some interfaces for this functions and provide this service for all the users in the future.

4. More accurate risk preference estimation. Sometimes the consumers don't choose the strategies provided by the system or even give up our software because the risk of strategies cannot meet user's expectations. We shall get more accurate estimation of risk preference for users. To achieve this aim, one way is that the system utilizes the user database which contains the strategies they choose to use. The system my use machine learning to deal with these data in order to predict the risk preference more accurately according to users' questionnaires answer and their choice of strategies. These functions only can be effective when the volume of users are much bigger but we should reserve each user's behavior data first.

# Appendices

The hardware in the system mainly includes servers and database. The servers are used to connects all consumers sending and receiving messages such as user account data which are stored in database. Most calculations are done in the client terminals which will not give pressure to severs. It also does not need occupy the messages flow between users and servers providing more spaces for transmission of stocks or futures data. The database also should store the data of stocks, futures and other targets asset. Therefore, the database need more storage space while the resources requirement of servers is not too much.

In the database, user account data could be packaged by individual user account. An object could be created which includes user ID, asset volume, risk preference, strategies used and return. For the stock and futures data, the system just acquires them through the interface of other financial database software such as Wind. The data will be divided by time and classifications including price, volume, position and everyday price change rate for easier to be utilized in back-test. When the user uses the software to back-test the strategies, the stock and futures data will be downloaded through network and the format of the data will not change.