

*Software Engineering*

Smart Robo-Advisor

PLANNING

**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. Introduction 2](#_Toc4328161)

[1.1 Project Overview 2](#_Toc4328162)

[1.2 Project Deliverables 3](#_Toc4328163)

[1.3 Business Opportunities and Pricing 3](#_Toc4328164)

[2. Project Organization 4](#_Toc4328165)

[2.1 Organizational Structure 4](#_Toc4328166)

[3 Managerial Process 8](#_Toc4328167)

[3.1 Management Objectives and Priorities 8](#_Toc4328168)

[3.2 Assumptions, Dependencies and Constraints 8](#_Toc4328169)

[3.3 Risk Management 9](#_Toc4328170)

[3.4 Monitoring and Controlling 9](#_Toc4328171)

[3.5 Staffing Plan 9](#_Toc4328172)

[5 WBS, Schedule, and Budget 12](#_Toc4328173)

[5.1 Work Breakdown Structure 12](#_Toc4328174)

[5.2 Dependencies 13](#_Toc4328175)

[5.3 Resource Requirements 15](#_Toc4328176)

[5.4 Schedule 15](#_Toc4328177)

[5.5 Budget and Resource Allocation 16](#_Toc4328178)

# 1. Introduction

## 1.1 Project Overview

Our agile project planning, divided into five parts, is supposed to be finished in 6 weeks. The detailed plan is showed in Picture 1. Using it for guide, we can make reasonable management and realize efficient cooperation.



**Picture 1.1-Agile Project Planning**

## 1.2 Project Deliverables

The deliverables of our project include platform system and interface, back-end database and the relevant document materials shown in Table 1.

|  |  |
| --- | --- |
| **Category** | **Deliverables** |
| Platform | Whole system including algorithms, interface and database |
| Documents | Market survey, system description, requirement specification, test and validation report, customer feedback report, version records etc. |

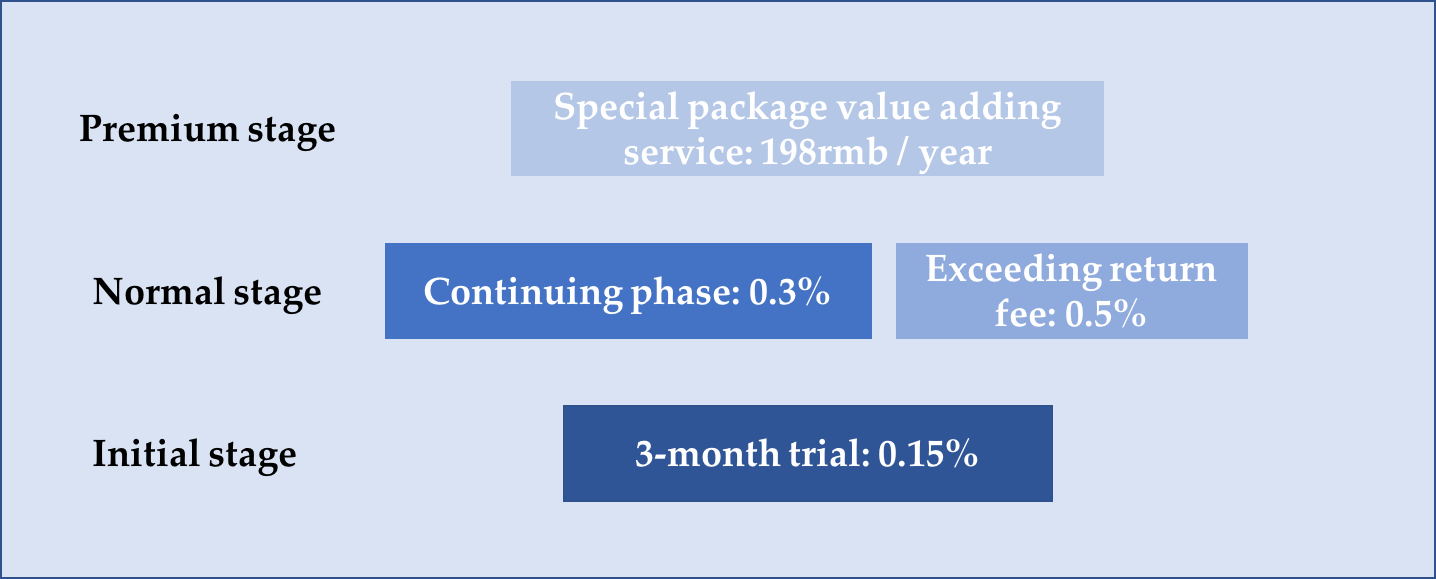
**Table 1- Project Deliverables**

## 1.3 Business Opportunities and Pricing

In the first stage of pricing, since we are the first mover in this area, we charge significantly lower management fee than other investment management services in order to attract the market, which is 0.15% for 3-month trial for new customers.

In the second stage, after 3 months, if the user keeps using our platform, the fee will be raised up to 0.3% as normal charging. In terms of additional pricing, we charge extra fees for 0.5% of return that exceeds the expected return threshold, which is still lower than the traditional management fee.

Meanwhile, we also set up premium package (198rmb per year) that provide value-added services, for example, giving rationality diagnosis of asset use and sending customized financial investment knowledge instead of merely investment instructions. Picture 1.3 illustrates the general pricing strategy of our system.



**Picture 1.3-Pricing Strategy**

# Project Organization

## 2.1 Organizational Structure

This part describes the way in which the development team is organized, the people involved, and their roles in the team. In order to improve work efficiency and ensure the quality of work, the organization, division and management of software developers is a very important task, which directly affects the success and failure of software projects.

First of all, because the personal qualities of software developers vary greatly, it is obviously important to make full use of each person's strengths and experience for the selection and division of software developers.

Second, the organization of software personnel is critical. An important principle is that the software developer's organizational structure corresponds to the software project development model and the structure of the software product, so that the software development method, tools, and personnel can be unified, thereby reducing the complexity of the management system and facilitating the complexity. Management and quality control of the software development process. We mainly use **a tree structure**, the root of the tree is the project manager, and the node of the tree is the programmer team. In order to reduce the complexity of the system and facilitate project management, the number of nodes in the tree should not exceed seven, on this basis. Try to reduce the number of layers in the tree. The number of programmers should be based on the size of the task and the time it takes to complete the task, typically 2 to 5 people.

The development of software projects shows that the types, levels, and number of technicians required at each stage of software development are different. The planning and analysis phase of a software project requires only a few people, mainly system analysts, software senior engineers and project senior managers engaged in software system demonstration and summary design. When designing in detail, add software engineers and programmers, and add programmers and software testers during the coding and testing phases. By the end of the testing phase, the number of software project developers reached its peak. At the beginning of the software operation, there are many people who participate in software maintenance. After the software runs for a period of time, because the software developers participate in error correction maintenance, the software error rate will quickly deplete, and the software developers can gradually withdraw.

The following people are required in a typical software development project:

|  |  |
| --- | --- |
| People | Roles |
| Project manager | Project implementation, project planning, and tracking management |
| Demand analyst | Demand acquisition and analysis |
| Designer | System design and guide programmers in system development |
| Programmer | General module detailed design and coding test |
| Database administrator | Database construction and database maintenance |
| Tester | Tests at all stages of the project, including module testing, system requirements testing, integration testing, system testing, etc. |
| Artist | Design UI interface |

**Table 2.1 – Team & Roles**

**Due to the short cycle of our projects and the limited staff, we are unable to assign staff in strict accordance with the traditional methods. We will adopt a tree structure to divide the work according to their respective strengths and experience. The specific personnel composition and their roles are shown in Picture 2.1.**

**Picture 2.1 - Personnel Composition**

UI Design

Manager

Wang Rusi (王如思)

Risk Assessment

Ye Juelin (叶珏琳)

Multi-factor Strategy

Shi Yunxia (史云霞)

CTA Strategy & Database

Wang Bingquan (王柄荃)

Style Rotation Strategy & Choose Portfolio

Shen Zimin (谌子民)

Ye Juelin

(叶珏琳)

# 3 Managerial Process

## 3.1 Management Objectives and Priorities

The objectives of system management are divided into three parts: front desk system (the user interaction interface), the middle desk system (core algorithm of our system), and the backend system (system platform and database).

The priorities of management can be determined by importance of each part. The front desk system is vital to our system. Because the front desk directly contacts the user, it can directly affect the user experience. Users are the most critical role in our entire system, so we should pay the most attention to the management of the front-end interactive interface. The strategy part of the middle platform is to provide users with strategic support. It is the core part of our entire system, so the management of the middle-end algorithm is also very important. Finally, the back-end system and the stability of the database operation can directly affect the performance of the front-end platform of the entire system, so we should also pay attention to the management of the background.

## 3.2 Assumptions, Dependencies and Constraints

(1) Assume that the database source is stable, it can provide us with stable and timely data sources.

(2) Assume that the backend system is stable and will not be affected by extreme weather.

(3) Assume that the mid-stage algorithm has a high data fault tolerance and will not crash due to cluttered data.

(4) Assume that the front-end system is highly maintainable, and the system interface can be modified at any time according to user needs.

## 3.3 Risk Management

The core data of this system is basically stored in the back-end database, and the operation of the whole system is extremely the data of the back-end system database. Therefore, ensuring the stability of the back-end database has become the top priority of the risk control link of this system. In order to avoid sudden data loss, we need to create a backup database to synchronize with the primary database. At the same time, it is necessary to set up hardware facilities such as standby power supply to ensure the long-term stability of the database.

## 3.4 Monitoring and Controlling

Due to the importance of the back-end system, we need to monitor the back-end system and database operations 24 hours a day. If you find problems with the database, look for remedial measures in a timely manner, such as replacing temporary power, switching backup databases, and so on.

## 3.5 Staffing Plan

The team has five core members and five core members have extensive quantitative investment experience. Among them, the No. 1 member is a system-wide planner, responsible for coordinating the whole process, and the macro control system is moving; the second member is the strategist, who is mainly responsible for the R&D quantitative investment strategy; the third member is a developer, who is mainly responsible for the background system development and data monitoring; the fourth member is the product planner, responsible for product design and business model design; the fifth member is the designer, mainly responsible for system UI design and system overall style design.

The division of labor of the five individuals is clear, but not completely independent. Members work relatively independently, but for better cooperation, there are also overlaps in order to maximize utility.

**4.Technical process**

**4.1 Methods tools and techniques**

Our system is to provide trading strategies for customers automatically which contains multi-factors strategy, statistical arbitrage strategy, commodity trading strategy，Industry rotation strategy. These are all widely used in quantitative trading areas now.

**(1) Multi-factors strategy** is mainly used for choosing the stocks or selecting trading period in stocks trading. The theory of it is based on CAPM, ATP and Fama-French factors model. The basic idea is to find some indicators that are most relevant to the rate of return, and build a stock portfolio based on that indicator, expecting the portfolio to outperform or underperform the index in the future. If it is outperforming, buy more of the combination while shorting the index; if it is underperforming, buy the index and short-selling combination. This will earn the alpha income. The key to a multi-factors model is to find useful factors which is related to return.

**(2) Statistical arbitrage strategy** is to use statistical analysis tools to study and analyze historical data of a group of related prices, study the historical stability of the relationship, and estimate its probability distribution to determine the extremes in the distribution which is called negative region. When the price relationship in the real market enters the negative region, it is considered that the price relationship cannot be maintained for a long time, and the arbitrageur has a higher probability of success to enter the arbitrage.

**(3) Commodity trading strategy** mainly focus on stock index futures, commodity futures and government bond futures. It analyzes the volume and price data in the history of a single species and find the law with probability advantage. The the law will be implemented in code and researchers assumed that such laws will still exist in the future. if it is true profits can be obtained.

**(4) Industry rotation strategy** is an active trading strategy that leverages market trends. The essence is to use the misplacement of strong investment time of different investment varieties to change the investments in different industry to maximize the return on investment. It is based on an analysis of specific institutional system, policy, and developmental stages and also focuses on cyclical changes in the economy.

To implement those strategy, the system used several packages in python such as Pandas, Numpy and Sklearn. The quota data is from Wind by the Windpy.

**4.2 Development environment**

We use python as the programming language. The main IDE for developing is Pycharm. The database is organized by mysql.

# 5 WBS, Schedule, and Budget

## 5.1 Work Breakdown Structure

The WBS is as followed:

## 5.2 Dependencies

Here are the dependencies of the software. We numbered each task, and the “Dependencies” line shows the predecessor need of current process.

Table 2 Schedule and dependencies

|  |  |  |  |
| --- | --- | --- | --- |
| Task  id | Schedule | Dependencies (Predecessor task id) | Duration (days) |
| 1 | 0 Software project planning | | 5 |
| 2 | 1 Demand analysis | 1 | 6 |
| 3 | 1.1 Demand capture | 1 | 1 |
| 4 | 1.2 design business flow chart | 3 | 1 |
| 5 | 1.3 Establishing a UML model | 3 | 1 |
| 6 | 1.4 writing requirements size specification | 4,5 | 1 |
| 7 | 1.5 demand size test | 6 | 1 |
| 8 | 1.6 demand size confirmation | 7 | 1 |
| 9 | 2 system design | 8 | 8 |
| 10 | 2.1 system architecture design | 8 | 2 |
| 11 | 2.2 database design | 8 | 2 |
| 12 | 2.3 system detailed module function design | 8 | 2 |
| 13 | 2.4 software design test | 10,11,12 | 1 |
| 14 | 2.5 software design confirmation | 13 | 1 |
| 15 | 3 system implementations | 14 | 25 |
| 16 | 3.1 data persistence layer implementation | 14 | 5 |
| 17 | 3.2 system framework construction | 14 | 5 |
| 18 | 3.3 system performance layer implementation | 16,17 | 5 |
| 19 | 3.4 system server-side application implementation | 16,17 | 5 |
| 20 | 3.5 system integration | 18,19 | 5 |
| 21 | 4 software testing | 20 | 12 |
| 22 | 4.1 integration test | 20 | 3 |
| 23 | 4.2 functional test | 22 | 3 |
| 24 | 4.3 system test | 23,22 | 3 |
| 25 | 4.4 Acceptance test | 24 | 3 |
| 26 | 5 system acceptance | 25 | 2 |
| 27 | 5.1 Software Delivery | 25 | 2 |
|  | Total days |  | 58 |
|  | Total months |  | 3 |

## 5.3 Resource Requirements

We divide the resource requirements into software resources, hardware resources, and human resources.

Human resources are shown in detail in Table 1, namely five developers, three months.

The other resources are 5 laptops for development (installed MySQL, Python, Wind, Bootstrap) and corresponding power support.

## 5.4 Schedule

The project cycle is 3 months. The project is expected to complete requirement analysis, overall framework design and system design in the first month. In the second month, the software code is written and the test session begins. In the last month, complete the software's various functions, performance and security tests. After 3 months, the project will be completed and accepted. The specific is as followed in (Table 3 Schedule and dependencies).

## 5.5 Budget and Resource Allocation

It is expected that the project will be developed and tested by 5 employees for 3 months. So the project budget is as follows:

Labor cost: 6,000 yuan / month × 5 × 3 = 90,000 (unit: yuan)

Equipment cost: 90,000 × 20% = 18,000

Total cost: 90,000 + 18,000 = 108,000

Resources are evenly allocated among each developer.