Product Requirement Document for Yihan’s Project

Background 1

Study design 1

Study 1 1

Objective 1

Experimental conditions 2

Study 2 2

Objective 2

Experimental conditions 2

Experimental stages 2

Requirements 3

General requiremnets 3

Study 1 3

Stage 0: Pre-survey + Stage 1: Randomization 3

Stage 2: User-orientation 5

Stage 3: Experiment 6

Stage 4: Evaluation 7

Study 2 8

Stage 0: Pre-survey + Stage 1: Randomization 8

Stage 2: User-orientation 8

Stage 3: Experiment 9

Stage 4: Evaluation 9

# Background

The general objective of this project is to investigate how Fintech service platforms can be designed to facilitate users in making better financial decisions.

The specific objective of this study is to investigate how robo-adviosr’s personality affects investors’ financial decisions and user experience.

# Study design

To investigate the effects of robo-advisor’s personality, we will endow robo-advisor

with two types of personality: dominant and submissive. We will also screen participants based on their personality: dominant, submissive and neutral.

To investigate whether effects vary when robo-advisor’s performance varies, we will simulate three types of performance levels: average, improvement, and deterioration.

The experimental designs are illustrated as follows:

## Study 1

### Objective

To studyinteraction between robo-advisor’s personality with participant’s personality while holding the robo-advisor’s performance as the same

### Experimental conditions

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Condition 1 | Condition 2 | Condition 3 | Condition 4 | Condition 5 | Condition 6 |
| Robo-advisor’s  Performance | **Start date:** Current date – **24** months, **Duration**: **24** months, **Performance level**: **Average** | | | | | |
| Robo-advisor’s  Personality | Dominant | | | Submissive | | |
| Participants’ personality | Dominant | Submissive | Neutral | Dominant | Submissive | Neutral |

## Study 2

### Objective

To studyinteraction between robo-advisor’s personality with participant’s personality while holding the robo-advisor’s performance changes: improve versus deteriorate

### Experimental conditions

#### Performance improvement conditions

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Condition 1 | Condition 2 | Condition 3 | Condition 4 | Condition 5 | Condition 6 |
| Robo-advisor’s  Performance | **Start date:** Current date – **48** months, **Duration**: **48** months, **Performance level**: 24 months **average** + 24 months **excellent** | | | | | |
| Robo-advisor’s  Personality | Dominant | | | Submissive | | |
| Participants’ personality | Dominant | Submissive | Neutral | Dominant | Submissive | Neutral |

#### Performance deterioration conditions

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Condition 7 | Condition 8 | Condition 9 | Condition 10 | Condition 11 | Condition 12 |
| Robo-advisor’s  Performance | **Start date:** Current date – **48** months, **Duration**: **48** months, **Performance level**: 24 months **average** + 24 months **poor** | | | | | |
| Robo-advisor’s  Personality | Dominant | | | Submissive | | |
| Participants’ personality | Dominant | Submissive | Neutral | Dominant | Submissive | Neutral |

# Experimental stages

Both studies are consisted of following stages:

|  |  |  |  |
| --- | --- | --- | --- |
| **Stage** | **Sub-stage** | **Page** | **Purpose** |
| 0: Pre-survey | NA | NA | Participants will answer a survey about their **personality and risk tolerance**, the data will be saved. We’ll categorize them to dominant, submissive, and neutral groups based on their personality scores. We’ll also categorize them into risk-averse and risk-seeking based on their risk tolerance score. |
| 1: Randomization | NA | 1, 2 | Participants will be **randomly allocated** to different experimental conditions |
| 2: User orientation | Self-introduction | 3 | Robo-advisor will introduce its **name** and greet participants |
|  | Q&A session | 4 | Robo-advisor will ask about participants’ personal information and assess their risk profiles |
|  | Task orientation | 5 | Robo-advisor will orient participants toward how to complete the following tasks |
| 3: Experiment | Asset allocation | 6 | Robo-advisor will interact with participants and try to persuade them to invest more with them |
|  | Manipulation check | 7 | Participants will be asked about their assessment of robo-advisor’s financial performance |
| 4: Evaluation | NA | 8 | Participants will be re-directed to Qualtrics to answer about their overall experience |

# Requirements

## General requiremnets

|  |  |
| --- | --- |
| Functional requirements | The design and implementation need to reflect the research objectives |
| Usability requirements | Participants can go through the whole experiments smoothly and interact with robo-advisor naturally |
| Technical requirements | Compatible with NTU and cellular network |
| Environmental requirements | User-friendly for both Windows and Mac users  User-friendly for both web and mobile users |
| Support requirements | Support mainstream browsers: Chrome, Safari, IE, Firefox  Minimizing other support requirements from the participants side |
| Interaction requirements | Strong sense of conversing with an intelligent being;  Feel that the whole conversation is customized, not pre-programmed |

## Study 1

### Stage 0: Pre-survey + Stage 1: Randomization

#### Lab version

##### Procedure

1. We will invite participants to answer a pre-survey about personality and risk tolerance.
2. Participants will enter their metric number and basic demographic info before proceeding to answer the survey questions.
3. We’ll calculate personality score and risk tolerance score based on their answers and categorize them into three groups: dominant, submissive, and neutral.
4. When participants visit the lab, we’ll try to balance their personality types and risk tolerance levels so that each experimental condition will have roughly the same proportion of dominant and submissive participants as well as the same proportion of risk-seeking and risk-averse participants.

##### Specific requirements

Front-end:

1. Create two buttons that correspond to dominant and submissive robo-advisor for experimenter to choose
2. Create a data entry field for participants to enter their metric number

Back-end:

1. Create two robo-advisor versions for experimenter to choose: dominant vs. submissive

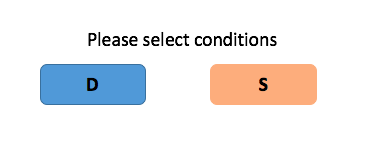
The two robo-advisor versions will have the following attributes:

|  |  |  |
| --- | --- | --- |
|  | Dominant | Submissive |
| Name | Max | Linus |
| Avatar | Robot with opening arms | Robot with folding arms |
| Conversational style | Dominant style | Submissive style |

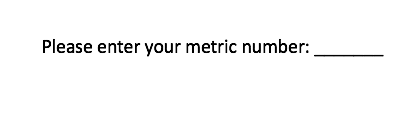
1. Save the metric number
2. Create two libraries to host script of two conversational styles

Mock-up:

Page 1



Page 2



#### Online version

##### Procedure

1. We will recruit participants from an online portal.
2. Participants will first answer the pre-survey and then proceed to the randomization stage

##### Specific requirements

Same as lab version except that participants don’t need to enter metric number nor do experimenters need to choose experimental conditions (No need of page 1, 2)

### Stage 2: User-orientation

#### Self-introduction

##### Specific requirements

Front end

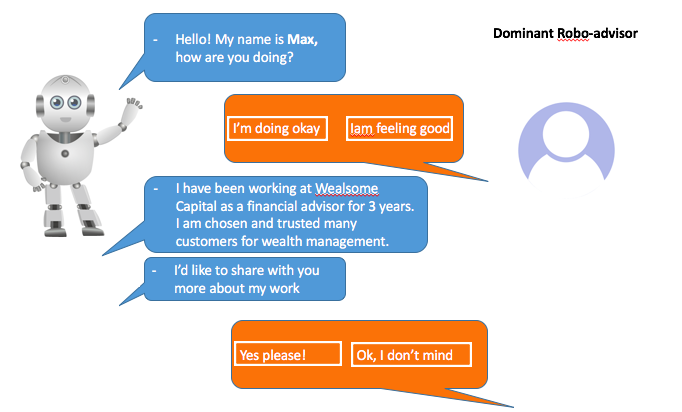
1. In the introduction part, when the robo-advisor greets: Hi, my name is XXX, substitute XXX with its corresponding name
2. Load the corresponding scripts fro each condition
3. Create and embed a logo similar to Cuimin’s project, but with a different name: Wealsome Capital

Back end

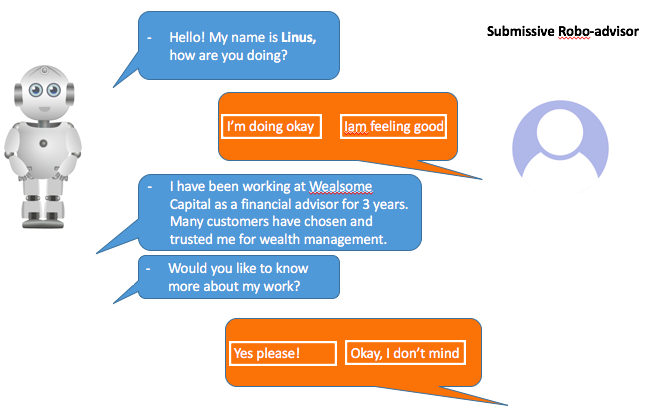
1. Create robo-advisor’s name variable XXX for front-end to quote
2. Create two libraries to host script of two conversational styles

Mock-up:

*Dominant version*



*Submissive version*



#### Q&A session

Similar to above, with different scripts

#### Task-orientation

Similar to above, with different scripts

### Stage 3: Experiment

#### Sub-stage: asset-allocation

##### Procedure:

1. After participants receive the instructions on how to decide on fund allocation, they’re asked how much they are willing to invest with Robo-advisor for a fix period of 12 months. The answer will be saved as trust\_t0.
2. Then participants are invited to go through robo-advisor’s performance of previous 2 years.
3. They are told that unlike previous decision, now they can adjust the investment size every month. They are also told that their bonus will be dependent on how much they earn throughout this period. If they earn $1,000 in the experiment, it will be translated at a ratio of 200:1 to cash, i.e. $5 (rounded off to integers).
4. The monthly commitment will be saved as commit\_n (n🡪 1 ~24)
5. At the end of the 24th months, participants will be asked again how much they are willing to invest with Robo-advisor for a fix period of 12 months. The answer will be saved as trust\_t1.
6. After that, participants are asked to evaluate robo-advisor’s financial performance.

**Note:** for online version, the bonus instruction is different. Participants will instead be notified that the top 10% performers will win a bonus as much as 100% of their HIT payment.

##### Specific requirements

Front end

1. Implement the design according to previous template + company logo
2. Appropriate animation that makes conversation flows more naturally
3. Positive returns will be colored as green; negative as red
4. For different scenarios, load corresponding scripts

Scenarios by robo-advisor’s performance

|  |  |
| --- | --- |
| Scenarios | Script categories |
| **Underperforming** compared to last month by more than 1% | Comforting scripts  (Randomly load 1 out of 12) |
| **Outperforming** compared to last month by more than 1% | Encouraging scripts  (Randomly load 1 out of 12) |
| Performance compared to last month between -1%~ +1% | General investment wisdom  (Randomly load 1 out of 12) |

Back end

1. Save the following data: trust\_t0, trust\_t1, commit\_1~commit\_n
2. Save robo-advisor’s monthly performance data accordingly
3. Create sub-libraries for each scenario category
4. Make sure that the loaded script cannot be repeated more than three times
5. Calculate bonus and save it as bonus
6. Create a dummy variable for the top 10% performers for each condition

Mock-up

1. According to Stella’s design template

#### Sub-stage: manipulation-check

Front end

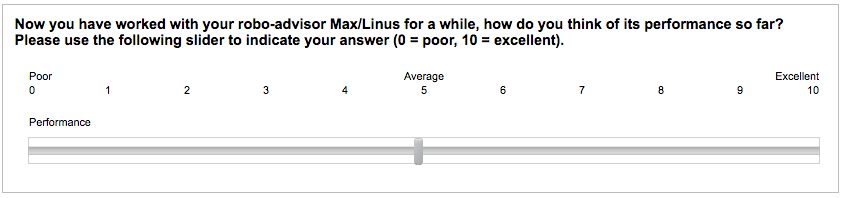
1. Revise the manipulation check according to the following mock-up
2. Note: substitute robo-advisor’s name (Max or Linus) according to the condition

Back end

1. Save the evaluation result as eva\_t1

Mock-up

Manipulation-check after 24th month



### Stage 4: Evaluation

##### Procedure:

##### Participants will be redirected to the Qualtrics platform to answer the survey of overall experience

1. They will be re-directed to the Web-app to be informed of the bonus

##### Specific requirements

Front end

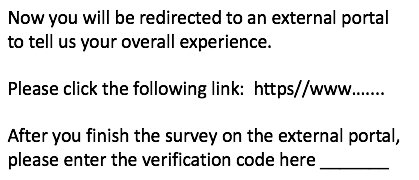
1. Create a page to re-direct participants to Qualtrics, after they finish the survey, they will be given a 3-digit code: verification\_code = 345
2. Create a field for participants to enter the code
3. Show participants the bonus if the code is verified to be true

Back end

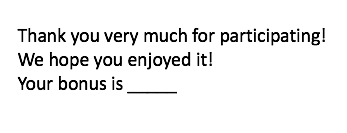
1. Create a verification field and verify whether it equals verification\_code = 345

Mock up

*Redirect to external portal*



*Bonus*



## Study 2

### Stage 0: Pre-survey + Stage 1: Randomization

#### Difference from study 1

##### Procedure

1. After participants are manually assigned to dominant and submissive conditions by experimenters, they will be further randomized to two conditions: improved performance vs. deteriorated performance

##### Specific requirements

Front end

1. Since the experimenter cannot easily tell study 1 from study 2, we may need to insert a sign “V2” somewhere in the webpage to represent study 2. It serves as a secret code, so it needs to be recognizable yet not to distract participants.

Back end

1. Set up the randomization component

### Stage 2: User-orientation

#### Difference from study 1

##### Procedure

1. As study 2 has a longer time horizon, participants need additional instructions on task orientation

#### Task-orientation

Front end

1. Load specific task-orientation scripts for study 2

Back end

1. Create the corresponding script library for task-orientation instruction

### Stage 3: Experiment

#### Sub-stage: asset-allocation

#### Difference from study 1

#### Procedure:

1. Participants are invited to go through robo-advisor’s performance of previous 4 years instead of 2 years in study 1.
2. Participants are instructed that their bonus conversion ratio is between 200:1 to 600:1
3. The monthly commitment will be saved as commit\_n (n🡪 1 ~48)
4. At the end of the 48th months, participants will be asked again how much they are willing to invest with Robo-advisor for a fix period of 12 months. The answer will be saved as trust\_t2.

##### Specific requirements

Front end

1. Adjust the time horizon to 4 years, with a break at the end of 2nd year
2. During the break, show the evaluation question

Back end

1. For participants assigned to the improvement condition, the conversion ratio is 600:1; For participants assigned to the improvement condition, the conversion ratio is 200:1.
2. Calculate participant’s bonus accordingly

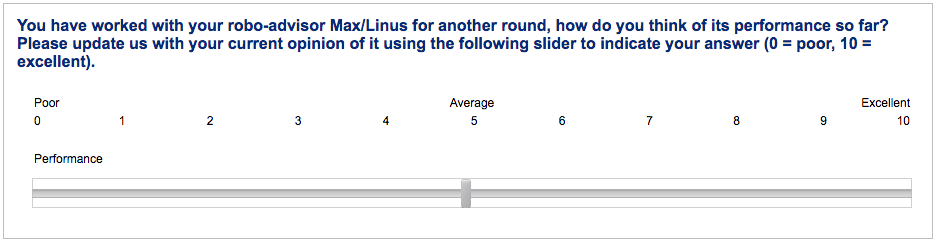
#### Sub-stage: manipulation-check

##### Procedure

1. Add another manipulation check after 48th month

Mock up

Manipulation-check after 48th month



### Stage 4: Evaluation

#### Difference from study 1

##### Procedure:

##### Participants will be redirected to the Qualtrics platform to answer the survey of overall experience, but the link will be specific to study 2.

Front end

1. Update the link address