**HSBC Fund Rebalancing Project**

Project Plan

4/1/2019  
Version 1.1

**Team REST**

Aurora (Yutian) Qiu

Cathy Leung

Leo Cheng

Leo (Jixing) Li

Shaw (Xiao) Lu

Stanley Ye

Tanya (Yi) Tan

## **Table of Contents**

[**Table of Contents**](#_71u1uwhem4h3) **2**

[**Revision History**](#_3ktrdv8oyvym) **3**

[**Project Plan Overview**](#_9b76xnulohx4) **3**

[**Project Timeline**](#_chu1nh8krart) **3**

[Gantt Chart](#_84z542m32m3r) 3

[Planning Phase](#_elr0j212uwui) 4

[Team Formation (Jan 2 - 9, 8 Days)](#_y5ofz7i2d4ax) 4

[Project Exploration (Jan 2 - 13, 12 Days)](#_rvpeghgxsj30) 4

[Requirement Gathering Phase](#_ehnyz0i2qc2s) 4

[Gather Requirements from Sponsors (Jan 14 - 17, 4 Days)](#_giovtdmg6bc9) 4

[Consolidate Requirements Document (Jan 14 - 27, 14 Days)](#_x53alz2ugiw) 4

[Set Up Ticket Tracking System (Jan 14, 1 Day)](#_u4zs2kpt9ilc) 4

[Break Down into Executable Tasks (Jan 14 - 16, 2 Days)](#_q1tvi3l3cxx8) 5

[Write Up Acceptance Criteria (Jan 14 - 20, 4 Days)](#_4rzg8uyvfv0e) 5

[Design Phase (Jan 28 - Feb 3)](#_wsnccarf77ph) 5

[Architecture Document Design (Jan 28 - 30, 3 Days)](#_etm6pya27y3f) 5

[Critical Priority Analysis (Jan 31, 1 Day)](#_v5lbg15dpdyy) 5

[Implementation Plan Design (Feb 1 - 2, 2 Days)](#_qk5doesfgza) 6

[Test Plan Design (Feb 3, 1 Day)](#_7mo3sck99f4n) 6

[Implementation Phase](#_y4uoq5j2hxxu) 6

[Task Assignment (Feb 4 - 6, 3 Days)](#_k464kbabt2t9) 6

[Coding Endpoints (Feb 4 - Mar 3, 28 Days)](#_7m0c9f8fxx33) 6

[UI Development (Feb 18 - Mar 3, 14 Days)](#_td8c6a37wca5) 7

[Testing Phase](#_pr8uibtnx6l3) 7

[Functional Testing (Feb 4 - Mar 3, 28 Days)](#_m8duyau8y8vo) 7

[UI Testing (Mar 4 - 17, 21 Days)](#_5f8uxuislc1) 7

[User Acceptance Testing (UAT) (Mar 18 - 31, 21 Days)](#_l3e79fa6bh9p) 7

[Project Release and Maintenance Phase](#_52yqotmnsq0n) 8

[**Appendix I - Gantt Chart**](#_pb83g891jgkl) **9**

## 

## **Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Name** | **Description of Change** |
| 1.1 | 4/1/2019 | Team REST | Final version |
| 1.0 | 1/14/2019 | Team REST | Initial draft |

## **Project Plan Overview**

Following the Software Development Life Cycle, our project will be executed in the order of Planning, Requirement Gathering, Design, Implementation, Testing and Release and Maintenance Phase.

In terms of the project timeline, our critical path is identified on the Gantt chart with one phase executing after another except a concurrent execution of API Implementation and Functional Testing. We believe that by having API implementation, Unit Testing, Integration Testing at a concurrent time, we will be able to ensure code quality and the success criteria in a timely fashion.

In terms of the resource plan, we have assigned two leaders per phase, who have past experience performing the work during each phase to lead the team and provide guidance. Below are the Project Timeline Chart, a detailed description of each phase and the estimated effort.

## **Project Timeline**

### **Gantt Chart**

Refer to Appendix I for a visual overview of the project timeline.

### **Planning Phase**

Lead: Shaw (Xiao) Lu

Duration: Jan 2 - Jan 13

#### **Team Formation (Jan 2 - 9, 8 Days)**

During this period, we finalized our project group of 7 and submitted our team information in Piazza.

#### **Project Exploration (Jan 2 - 13, 12 Days)**

Our team have explored all the available projects, ranked them by preference and landed on HSBC. We completed the Terms of Reference to define the purpose and structure of the project.

### **Requirement Gathering Phase**

Lead: Aurora (Yutian) Qiu and Tanya (Yi) Tan

Duration: Jan 14 - Jan 27

#### **Gather Requirements from Sponsors (Jan 14 - 17, 4 Days)**

The Requirement Gathering Phase starts with gathering more information from sponsors. We will continue to ask clarification questions regarding the mock system as well as the detailed requirements for each API endpoint to clarify both functional as well as non-functional requirements.

#### **Consolidate Requirements Document (Jan 14 - 27, 14 Days)**

We will be capturing all requirements divided into each endpoint as a reference for implementation. In the documents, we will be highlighting the background of each endpoint as well as an overview of the expected outcome. This activity will take most time of the Requirement Phase as we would like to capture the requirements in details and give our developers a holistic view of what each endpoint should look like fulfilling the success criteria.

#### **Set Up Ticket Tracking System (Jan 14, 1 Day)**

We choose Trello as our ticket tracking system to keep track of our development progress. Trello is an easy-to-use project management tool that allows us to create tickets and assign to designated developers during the span of the project.

#### **Break Down into Executable Tasks (Jan 14 - 16, 2 Days)**

Tentatively, each person in the team will be in charge of each API in this project. We will break down each API development into executable tasks so that each task can be tested separately to improve our code quality.

#### **Write Up Acceptance Criteria (Jan 14 - 20, 4 Days)**

At the end of this phase, we will write up the acceptance criteria for the tickets created for our developers to check-off.

### **Design Phase (Jan 28 - Feb 3)**

Lead: Leo (Jixing) Li and Stanley Ye

Duration: Jan 28 - Feb 3

Our design phase consists of 4 different stages in this order: Architecture Document Design, Critical Priority Analysis, Implementation Plan Design, and Test Plan Design. The order is important as the next stage depends on the previous stage. The documents in each of these stages will be used in the later phases of the project plan.

#### **Architecture Document Design (Jan 28 - 30, 3 Days)**

The design phase begins with architecture design. The technical and business requirements will be mapped into an architecture design document. This document maps out the application components, interfaces, and behaviours. The document will also entail technical details such as the technologies and programming languages that will be used in the application. This will take approximately 3 days due to the work involved in gathering the requirements and mapping out these requirements into an architecture. Some research into different technologies will be involved as well.

#### **Critical Priority Analysis (Jan 31, 1 Day)**

In the section, we will analyze the issues that need to be prioritized by splitting the success criteria into a set of important tasks. The stretch goals will also be considered. This will take no more than a day as it mainly requires analyzing the technical and business requirements.

#### **Implementation Plan Design (Feb 1 - 2, 2 Days)**

In this stage, we will be using the architecture document design in order to design an implementation plan. Implementation trade-offs will be considered. Implementations for the database, backend, and frontend will be planned out. This will take approximately 2 days as we need to dive deeper into specific software scenarios.

#### **Test Plan Design (Feb 3, 1 Day)**

In the test plan, we will define the necessary criteria that our application needs to meet to be a high-quality application. We will design a plan to completely test the system and meet the requirements outlined in the HSBC document through functional, UI and user acceptance testing. This should take approximately 1 day as we need to outline how each specific testing plan will be done and the test cases in each plan.

### **Implementation Phase**

Lead: Cathy Leung and Leo Cheng

Duration: Feb 4 - Mar 3

#### **Task Assignment (Feb 4 - 6, 3 Days)**

In terms of assigning work during the coding phase, we want everyone to get the chance to work on end to end development. We will do this by assigning one endpoint to each team member and as there are only 6 endpoints to implement and 7 team members we will have two people work on one of the larger endpoints such as /rebalance. This process should not take more than a couple of days so developers can begin implementation immediately.

#### **Coding Endpoints (Feb 4 - Mar 3, 28 Days)**

During the coding phase, everyone will be working simultaneously on their assigned endpoints. Following the implementation design plan, we expect a similar code structure but anticipate some refactoring will be required for consistency. The main focus will be on ensuring we can retrieve and update from a database which we can use POSTMAN to test; saving UI development for later. As this is the main development period we’ve allocated 4 weeks of time to ensure we follow proper coding practices and cover all edge cases. The stretch goal will be considered during the development period depending on our progress.

#### **UI Development (Feb 18 - Mar 3, 14 Days)**

Once our endpoints are up and running we can begin developing a simple UI. We’ve allocated 2 weeks time to build a simple interface for users to test the different endpoints we have implemented. The UI will provide the ability to hit all API endpoints, along with a display for viewing suggested instructions provided by the rebalancing.

### **Testing Phase**

Lead: Shaw (Xiao) Lu and Aurora (Yutian) Qiu

Duration: Feb 4 - Mar 31

#### **Functional Testing (Feb 4 - Mar 3, 28 Days)**

For Functional Test phase, we will use unit tests and integration tests as our methodology of testing. Unit tests are performed at the same time as the coding process to ensure that individual API method will work appropriately. Integration tests are introduced when the database system and API implementation are completed. Integrations test checks the interaction between the database, the rest API (by us), and the mock system (by sponsor). As this phase of testing ensures the success criteria of our project, it takes the longest with the most effort from the group.

#### **UI Testing (Mar 4 - 17, 21 Days)**

For UI testing, we are going to manually check user interaction with our application as well as improve design elements. We estimated that UI testing will take less of time than the previous testing phase.

#### **User Acceptance Testing (UAT) (Mar 18 - 31, 21 Days)**

User acceptance testing (UAT) is the last phase in testing and it is a chance for sponsors and intended clients to interact with this application to find out if everything works as it should and if any features are miscommunicated, not communicated and so on.

### **Project Release and Maintenance Phase**

Lead: Leo (Jixing) Li

Duration: Apr 1 - Apr 30

At this point, we are almost done. We will deploy our application with the database server on the Google Cloud Platform and maintain our project.

## 

## **Appendix I - Gantt Chart**

