

# 1. Introduction

## 1.1 Purpose

The purpose of the Software Development Plan is to gather all information necessary to control the project. It describes the approach to the development of the software and is the top-level plan generated and used by managers to direct the development effort.

The following people use the *Software Development Plan*:

- The **project manager** uses it to plan the project schedule and resource needs, and to track progress against the schedule.
- **Project team members** use it to understand what they need to do, when they need to do it, and what other activities they are dependent upon.

## 1.2 Scope

This Software Development Plan describes the overall plan to be used by the ISeeFin project, including deployment of the product. The details of the individual iterations will be described in the Iteration Plans. The plans as outlined in this document are based upon the product requirements as defined in the *Use Case Document* (Appendix A).

## 1.3 Definitions, Acronyms, and Abbreviations

- **Rational Unified Process** - An iterative software development process framework.
- **Scrum** - An Agile software development framework for managing software projects.
- **Agile** - A paradigm of software development methods focusing on iteration, adaption to change, and discovery of granularity through refinements to wide band requirements.
- **Sprint** - An iteration within the Scrum framework
- **Iteration** - A timeboxed effort.
- **Phase** - One or many iterations/sprints working towards a single unifying goal.

Note: Sprint and iteration are used interchangeably within this document.

## 1.4 References

- Software Requirements Specification
- Vision Document
- Business Case Document
- Use Case Document / Estimation Analysis (Appendix A)

## 1.5 Overview

This Software Development Plan contains the following information:

**Project Overview** — provides a description of the project's purpose, scope, and objectives. It also defines the deliverables that the project is expected to deliver.

**Project Organization** — describes the organizational structure of the project team.

**Management Process** — explains the estimated cost and schedule, defines the major phases and milestones for the project, and describes how the project will be monitored.

**Applicable Plans and Guidelines** — provides an overview of the software development process, including methods, tools and techniques to be followed.

## 2. Project Overview

### 2.1 Project Purpose, Scope, and Objectives

We envisage building and launching in the market a software tool to support Investment Clubs. An investment club is a group of persons who pool their financial resources (i.e., cash) in order to acquire and manage collectively a portfolio of securities (such as stocks, or bonds, or other publicly traded financial instruments)..

For a more indepth purpose, scope, and objectives statment, refer to Appenix A (Use Case Documents) introduction and features.

### 2.2 Assumptions and Constraints

Version 1.0 of the system must be fully deployed and operational by **October 3, 2014**

### 2.3 Project Deliverables

- Business Use Cases (Appendix A)
- Business Use Case Survey
- Glossary
- Supplementary Specifications
- Creative Design Briefs
- User Interface Prototype
- Use Case Survey
- Data Model
- Design Model
- Database Design
- Software Architecture Document
- Implementation Subsystem

- Test Package
- Change Requests
- Test Summary
- Product Backlog

## 2.4 Evolution of the Software Development Plan

The Software Development Plan will be revised prior to the start of each Iteration phase.

Version	Phase	Target Date
2	Inception	May 10, 2014
3	Elaboration	May 25, 2014
4.1	Construction Iteration 1	June 10, 2014
4.2	Construction Iteration 2	June 24, 2014
4.3	Construction Iteration 3	July 8, 2014
4.4	Construction Iteration 4	July 22, 2014
4.5	Construction Iteration 5	August 5, 2014
4.6	Construction Iteration 6	August 19, 2014
4.7	Construction Iteration 7	September 2, 2014
5	Transition	September 16, 2014

## 3. Project Organization

### 3.1 Organizational Structure

The team will maintain a flat management structure in order to maintain agility and ease communication across the team. However, there will be roles delegated to people to maintain a single point of communication for important jobs as well as maintain accountability for milestones (see 3.3). As such, project roles will be delegated in accordance to those mentioned in the Rational Unified Process (RUP). However, internal development practices will follow more closely with those of a Scrum team.

### 3.2 External Interfaces

The project will maintain entirely in-house and within the one individual development team. However input and feedback will be taken into account from investors and end-user asked to test the system.

### 3.3 Roles and Responsibilities

Person	Role within Rational Unified Process AND/OR Scrum
Evan Louie	Project Manager, Deployment Manager, Requirements Reviewer, Configuration Manager, Change Control Manager
Jeff Chan	Project Reviewer, Requirements Reviewer, Architecture Reviewer
Theo Ng	System Analyst, Requirements Specifier, Test Manager, Test Analyst
Kaya Gayos	Requirements Reviewer, Architecture Reviewer, Design Reviewer, User Interface Designer, Scrum Master
Matt Park	Software Architect, Product Owner
All	Designer, Implementer, Code Reviewer, Integrator, Test Designer, Tester, Technical Writer

## 4. Management Process

### 4.1 Project Estimates

The Inception and Construction phases are estimated to take approximately 3000 Person-Hours to complete (refer to Use-Case Point Analysis of Appendix A). Add to this a minimum of 200 Person-Hours for the Transition Phase, the project will take approximately 3200 Person-Hours (4 Months with 5 staff members) and cost \$110,000.00.

For a more in-depth use case estimation analyses, refer to to Appendix A (Use-Case Analysis Document).

An additional \$30,000.00 will be required for hardware and internal software for development and deployment. This funding will go towards:

Hardware and Internal Expenses: \* Servers \* Development Machines \* Development Environments (IDE Licensing, Bug Tracker Licensing, etc...)

In total, the project is estimated to cost: **\$140,000.00** and be completed by **October 8, 2014**.

### 4.2 Project Plan

#### 4.2.1 Phase Plan

Refer to:

- Attached Gantt Chart (end of document)

## 4.2.2 Iteration Objectives

Phase	Iteration	Objective
Inception	1	Define a set of high level requirements and goals for the final product which both the team and investors can sign off on. Specify what technologies will be used.
Elaboration	1	Iterate and prototype on the high level requirements, refining user stories into tasks lists generating sprint backlogs.
Construction	1	Complete database services layer. First UI prototype completed.
Construction	2	User creation and login/logout, integration of UI assets and business logic.
Construction	3	Portfolio creation and prototyping of automated portfolio valuation.
Construction	4	Connect to external services to get live currency and tax information. Complete automated portfolio valuation
Construction	5	Generate and submit tax forms based on location and custom to province/state/country.
Construction	6	Refine UI/UX.
Construction	7	Change Request freeze. All lingering bugs and tasks in product backlog to be emptied.
Transition	1	Final testing and deployment

## 4.2.3 Releases

**Version Name Convention:** <Major>.<Minor>.<BugFix>

At the very end of every phase/iteration (minus Inception), the product will be in some form of a shippable state. As such, from Elaboration to Construction Iteration 7, there will be 7 versions ranging from 0.0.0 - 0.6.0 which will all be considered alpha/beta releases.

After the completion of the transition phase, 1.0.0 (final) will be released.

## 4.2.4 Project Schedule

Refer to:

- Attached Gantt Chart (end of document)

## **4.2.5 Project Resourcing**

### **4.2.5.1 Staffing Plan**

The project is planned to require the usage of the 5 preexisting members of the ISeeFin team. The team already has the required software developers, UI/UX specialists, and business analysts required to cover all aspects required for development.

### **4.2.5.2 Resource Acquisition Plan**

All the staff needed to complete this project are preexisting members of the ISeeFin team. Planning has been done with the assumption that only these 5 people will be used.

### **4.2.5.3 Training Plan**

Staff members have worked together as a team previously, as such, little to no is required in terms of team dynamics and workflow. However, should the situation arise that a technology is chosen that only a select few have familiarity with:

- Training sessions will be held prior to project kickoff.
- Project estimates will be re-analyzed and changes in the schedule will be escalated to the project sponsors, who will then decide whether to alter scope in order to preserve target completion dates.

## **4.3 Project Monitoring and Control**

### **4.3.1 Requirements Management**

High level requirements and goal are found within the Vision document. During Inception, the product backlog will be generated and act as the pool of requirements. Once Inception is completed and the product backlog generated, adding to the backlog will either occur out of necessity found by the team, or by external stakeholders who will contact the product owner to relay any such change requests. Any change request, whether it be to add/remove/modify a user story in the product backlog, will be evaluated at the end of an iteration during the sprint retrospective.

### **4.3.2 Schedule and Budget Control**

Expenses are monitored by the project manager, and reported and assessed monthly. (See Reporting and Measurement below).

The project manager maintains the Gantt and burn down charts monitoring the expected date of each milestone. The user stories in the sprint backlog are broken down into tasks, said tasks are

assigned to individuals. From the amount of tasks completed for a given user story, the project manager will extrapolate whether or not the project is on schedule. Changes in the schedule will be escalated to the project sponsors, who will then decide whether to alter scope in order to preserve target completion dates.

### 4.3.3 Quality Control

Development will engage in Test Driven Development to minimize defects. However, found defects will be recorded and tracked as "bugs", and defect metrics will be gathered (see Reporting and Measurement below).

All deliverables and code is required to go through their respective review process. All code will go under code review before merging to main code branches.

Any and all defects found during review which are unable to be fixed prior to end-of-sprint must be captured as bugs on the product backlog and be considered high priority within the next sprint.

### 4.3.4 Reporting and Measurement

#### Reporting Artifacts:

- Product Backlog
- Burn-down Chart
- Gantt Chart

The product backlog will measure overall progress and completion of functionality. It will also measure earned value (through story points) and total defects opened/closed.

The Burn-down chart will measure/report velocity of completion and whether or not the project is on schedule.

The Gantt Chart will specify the tentative schedule and work flow for the team over the entire project life. The Gantt chart will be modified as the project proceeds based on the product backlog and burn-down chart.

### 4.3.5 Risk Management

Overarching project risks will be identified in the Inception Phase. Subsequently, during the kickoff of every phase/iteration, potential risks will be identified specific to the phase/iteration and added to this table:

Risk Ranking (High, Medium, Low)	Description/Impact	Mitigation Strategy

### 4.3.6 Configuration Management

Appropriate tools will be selected which provide a database of Change Requests and a controlled versioned repository of project artifacts.

All source code, test scripts, and data files are included in baselines. Documentation related to the source code is also included in the baseline, such as design documentation. All customer deliverable artifacts are included in the final baseline of the iteration, including executables.

The Change Requests are reviewed and approved by a majority vote at the end and/or beginning of a sprint (retrospective or kickoff).

Full backups are performed bi-weekly and incrementals are performed nightly.

## **5. Annexes**

This document refers to many terms and ideas from:

- Rational Unified Process
- Agile Manifesto
- Scrum Guide