** MINISTRY OF EDUCATION AND TRAINING**

**FPT UNIVERSITY**

Capstone Project Document

Smart Buy

|  |  |
| --- | --- |
| **Group 2** | |
| **Group members** | Tran Khac Vy – Team Leader – SE60964  Ho Doan Trung – Team Member – SE60882  Tran Quang Phuc – Team Member - SE60993  Vu Nhat Anh Khoa – Team Member – SE60817  Le Tuan Anh – Team Member – SE60685 |
| **Supervisor** | Mr. Kieu Trong Khanh |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** | SFM |

-Ho Chi Minh City, 01/2014-

*This page is intentionally left blank*

***ACKNOWLEDGEMENTS***

We wish to thank various people for their contribution to this project: Our teachers for their advice and participation in the final review, our friends for their valuable technical support.

Special thanks should be given to Mr.Kiều Trọng Khánh, our research supervisor for his professional guidance and the useful, constructive recommendations throughout the course of this project.

# Table of Contents

[Table of Contents 4](#_Toc377250788)

[List of Tables 5](#_Toc377250789)

[List of Figures 6](#_Toc377250790)

[Definitions, Acronyms, and Abbreviations 6](#_Toc377250791)

[Report No.2 Software Project Management Plan 7](#_Toc377250792)

[1. Problem Definition 7](#_Toc377250793)

[1.1 Name of this Capstone Project 7](#_Toc377250794)

[1.2 Problem Abstract 7](#_Toc377250795)

[1.3 Project Overview 7](#_Toc377250796)

[2. Project organization 9](#_Toc377250797)

[2.1 Software Process Model 9](#_Toc377250798)

[2.2 Roles and responsibilities 9](#_Toc377250799)

[2.3 Tools and Techniques 10](#_Toc377250800)

[3. Project Management Plan 11](#_Toc377250801)

[3.1 Iteration 11](#_Toc377250802)

[3.2 Iteration Detail 12](#_Toc377250803)

[3.3 All Meeting Minutes 14](#_Toc377250804)

[4. Coding Convention 14](#_Toc377250805)

# List of Tables

[Table 1: Hardware Requirement for Server 8](#_Toc377250806)

[Table 2: Hardware Requirement for Mobile App 8](#_Toc377250807)

[Table 3: Roles and Responsibilities Details 10](#_Toc377250808)

[Table 4: Iteration 12](#_Toc377250809)

[Table 5: Phase 1: Preliminary Investigation or Analysis 12](#_Toc377250810)

[Table 6: Phase 2: Data Management 12](#_Toc377250811)

[Table 8: Phase 4: User Related Functions 13](#_Toc377250812)

[Table 7: Phase 3: Suggestion Algorithm 13](#_Toc377250813)

[Table 9: Phase 5: Market Management 13](#_Toc377250814)

# List of Figures

[Figure 1: Agile Development Model 9](#_Toc377233927)

# Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
|  |  |

# Report No.2 Software Project Management Plan

## Problem Definition

### Name of this Capstone Project

Student Finance Management (SFM)

### Problem Abstract

Currently at FPT University, the record and the financial information of each student are stored separately in different Excel spreadsheets. Therefore, the process of information retrieval of each student record or listing registered students in each trimester is extremely complicated. Besides, the school’s staff faces difficulties in managing student records and notifying required tuition fees to each specific student.

In order to reduce the cost of those process, SFM system was created to simplify the process of organizing records. The system will support the information retrieval of student records, automatically sends notification mail to students, and optimizes the monitoring and the financial management procedure.

### Project Overview

#### Current Situation

FPT University currently manages student records manually through Excel spreadsheets. However:

* Spreadsheets only support analyzing and storing data.
* Since it is difficult to retrieve and email to each specific student their record and tuition fees information for the next trimester, the school only sends students notification of deadline for registering to study in the next trimester.

The system will solve those problems by supporting the school’s staff to:

* Import all data from Excel spreadsheets to the system’s database
* Retrieve the information of a specific student easily
* Calculate plan for each student.
* Automatically email each student notification for their specific situation (study record, required tuition fees)

The system also let students keep track on their records

#### The Proposed System

The Student Finance Management system is intended to help managing student financial records and data. It requires web browser and Internet connection to operate. The system will have following functions:

* Admins can manage accounts, manage and configure the system.
* Staff can import excel file or manually enter new student record data.
* System can match existing records with newly imported data, calculate financial plan for students, notify and send email to students according to the plan.
* Managers can use the system to search information based on student id, name, and payment status.
* Students can request to view their own records and financial status.

#### Boundaries of the System

* The system can be used by every people with a laptop/computer.
* The system is **not intended** for managing these aspects:

+ Managing learning outcome detail.

+ Managing exam’s information.

+ Managing student’s schedule.

* The language of the system is English.
* The complete product includes:

+ The website, for staff and user.

+ All the process document involved.

#### Development Environment

##### Hardware requirements

**For server**

|  |  |  |
| --- | --- | --- |
| Windows | Minimum Requirements | Recommended |
| Internet Connection | Cable, Wifi (4 Mbps) | Cable, Wifi (8 Mbps) |
| Operating System | XP, Vista, 7, 8 | XP, Vista, 7, 8 |
| Computer Processor | Intel® Core 2 Duo | Intel® Core(TM) i5 CPU , M 460 @ 2.53GHz |
| Computer Memory | 1GB RAM | 3GB or more |

Table 1: Hardware Requirement for Server

##### Software requirements

* Microsoft Windows 7 Ultimate: operating system and platform for development.
* IntelliJ 14.1 : used to implement website.
* MySQL Community Server 5.6.26 : used to create and manage the database of system.
* Apache Tomcat 7.0.64 : Application server.
* Slack : used for communication and meeting.
* StarUML: used to create models and diagrams.
* Github & TortoiseSVN: used for source control.

## Project organization

### Software Process Model

Project is developed under agile model.



Figure 1: Agile Development Model

For more information: <http://www.indicthreads.com/1439/quick-introduction-to-agile-software-development/>

(Owner: IndicThreads.com. Online Software Developer Magazine and Conferences)

### Roles and responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full name** | **Role in Group** | **Responsibilities** |
| **1** | Kieu Trong Khanh | Project manager | * Specify user requirement * Control the development process * Give out technique and business analysis support |
| **2** | Tran Khac Vy | Team Leader, BA, DEV, Tester | * Managing process * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **3** | Ho Doan Trung | Team Member, BA, DEV, Tester | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **4** | Le Tuan Anh | Team Member, BA, DEV, Tester | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **5** | Nguyen Khoa Anh Tuan | Team Member, BA, DEV, Tester | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **5** | Tran Quan Phuc | Team Member, BA, DEV, Tester | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |

Table 3: Roles and Responsibilities Details

### Tools and Techniques

- Front-end technologies: Thymeleaf engine, CSS3, JavaScript, jQuery, AJAX.

- Back-end: Spring framework and Hibernate framework.

- Web Server: Apache Tomcat 7.0.64

- Database Management System: MySQL Community Server 5.6.26.

## Project Management Plan

### Iteration

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Phase**  **/Iteration** | **Description** | **Deliverables** | **Resource needed** | **Dependencies and Constrains** | **Risks** |
| **Preliminary Investigation or Analysis** | - Study similar existing systems.  -Identify and clarify requirements for the system in general. | -Introduction of proposed system.  -Main functions.  -Project Iteration Plan. | 30 man-days | N/A | Project may  not be feasible  for developing  because lack of technologies  and/or data |
| **Data management** | - Parse data from websites.  - Input data manually.  - Import data from excel files. | - Data management service. | 30 man-days | N/A | Lack of experience.  The implemented parsers are not the best.  Lack of test data |
| **Main user’s functions** | - Let user update price for current day.  - User can search a product price. | - Main user’s functions on web and mobile. | 30 man-days | Depend on “Data management”. | Lack of experience.  Not have a clear understanding about business process. |
| **Suggestion algorithm** | - Build algorithm to calculate the best way to buy a list of products. | - Suggestion service.  - User now can ask for the best way to buy a list of products. | 20 man-days | Depends on “Data management”. | The implemented algorithm is not the best.  Lack of test data. Lack of experience on making and deploying web service. |
| **Market and User Account management** | - Manage markets in the system.  - Manage user accounts in the system | - Market management system.  - User account management system. | 15 man-days | N/A | Lack of experience.  Not have a clear understanding about business process. |

Table 4: Iteration

### Iteration Detail

#### Phase 1: Preliminary Investigation or Analysis

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Identifying and studying existing systems** |  |  |
| **2. Identifying and clarifying main functions.** |  |  |
| **3. Introduction.** |  |  |
| **4. Project Management**  **Plan.** |  |  |
| **5. Website Prototype.** |  |  |
| **6. Mobile Prototype.** |  |  |
| **7. Design ER diagram.** |  |  |

Table 5: Phase 1: Preliminary Investigation or Analysis

#### Phase 2: Data Management

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Identifying Requirement and Planning** |  |  |
| **2. Create parsers** |  |  |
| **3. Input data** |  |  |
| **4. Implement GUI** |  |  |
| **5. Testing** |  |  |
| **6. Document** |  |  |

Table 6: Phase 2: Data Management

#### Phase 3: Main User’s Functions

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Identifying Requirement and Planning** |  |  |
| **2. Manage User** |  |  |
| **3. Update Product Price** |  |  |
| **4. Search Product Price** |  |  |
| **5. Testing** |  |  |
| **6. Document** |  |  |

Table 8: Phase 4: User Related Functions

#### Phase 4: Suggestion Algorithm

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Identifying Requirement and Planning** |  |  |
| **2. Choose algorithm** |  |  |
| **3. Implement algorithm** |  |  |
| **4. System suggestion function** |  |  |
| **5. Testing** |  |  |
| **6. Document** |  |  |

Table 7: Phase 3: Suggestion Algorithm

#### Phase 5: Market Management

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Identifying Requirement and Planning** |  |  |
| **2. Manage market** |  |  |
| **3. Testing** |  |  |
| **4. Document** |  |  |

Table 9: Phase 5: Market Management

### All Meeting Minutes

Refer to Meeting Minutes folder.

## Coding Convention

Java: Using to develop Website.

Summary:

* Naming Convention.
  + Use camel case for both variable and function name.
  + Use pascal case for class name.
* Indentation.
  + Four spaces should be used as the unit of indentation. The exact construction of the indentation (spaces vs. tabs) is unspeciﬁed. Tabs must be set exactly every 8 spaces (not 4).
  + Avoid lines longer than 80 characters, since they’re not handled well by many terminals and tools.
* Declaration.
  + One declaration per line is recommended since it encourages commenting.
  + In absolutely no case should variables and functions be declared on the same line.
  + Do not put different types on the same line.
* Code Examples

Follow “Code Conventions for the Java TM Programming Language, by Sun Microsystems, rev April 20, 1999”.

<http://www.oracle.com/technetwork/java/codeconventions-150003.pdf>