# Software Project Management Plan

## Problem Definition

### Name of this Capstone project

* **Information Management website for High School (IMWHC)**

### Problem Abstract

Old school information management method requires a lot of paperwork. Staff and teachers must be well trained or experienced to ensure the accuracy and punctuality. However, it takes a lot of time not only to train staff and teachers to get used to their work but also to do the management work itself, also the improvement of social media and technology demands high schools to have a better system. Parent demands to have a clear view of their children study progress and better communication with the school. Many high schools still find it hard to afford a more advanced and expensive system. Many teachers and staff find it hard to get used with the new and complex system. The solution is creating a simpler website which to help them not only easy to get used to but also help them to do work faster and enhance the accuracy and punctuality.

### Project Overview

#### Current Situation

The current most used system in many schools in Vietnam is VEMIS (Vietnamese Educational Management Information System) started from 4/2006 based on SREM Project with the help of the Europe Community chosen by the Vietnamese Ministry of Education. It’s a software and has to be installed into PC or laptop, it should help staff to manage schedule and activities of teachers and students, manage student grades, manually input data into the system and store information in database, if there are conflicts, the system could notify and leave a warning. The main reason it’s most used because it’s free. However, many staff and teachers still find it not only hard to install but also too complex to use. In order to work, the system requires .NET Framework 3.5, SQL Server 2005 and the software environment itself, it usually causes error when install and takes a lot of time to get used.

#### The proposed system

The IMWHC system is user-friendly and strictly designed to avoid as many errors as possible, it should support the most basic needs in management tasks and provide reports and logs including all activities or errors.

* Data management:
* Admin could fill manually or import data from excel file.
* System could export formatted Excel files.
* Admin could manage information about school year, class, students and teachers.
* Schedule management:
* Staff could create and update schedules.
* System could check and notify when there are conflicts.
* Student’s study result management:
* Teachers should be able to input and modify student’s grade in their responsible class only.
* System could calculate average grade based on inputted ingredient grades.
* System could evaluate student based on average grade.
* Teachers should be able to manually evaluate the conduct of student in their class only.
* Account management:
* System could create account based on the inputted information.
* Admin could active/inactive an account.
* Admin should be able to modify the account’s profile.
* Statistics and reports management:
* Head teacher could view report and statistics based on student result.

#### Boundaries of the System

* The system could be used by staffs and teachers, students and parents with a laptop, PC or MAC.
* The used language of the system is Vietnamese.
* The complete product includes:

+ The website

+ All the process document involved.

#### Development Environment

##### Hardware requirements

**For client**

|  |  |  |
| --- | --- | --- |
| Windows | Minimum Requirements | Recommended |
| Operating System | Windows 7, 8 | Windows 7, 8 |
| Computer Processor | Intel® Core 2(TM) i3 CPU M370 @2.4GHz 3.39GHz | Intel® Core(TM) i5-2410M CPU @ 2.30GHz |
| Computer Memory | 2GB RAM | 4GB or more |
| Internet Connection | Cable, Wi-Fi (2 Mbps) | Cable, Wi-Fi (12 Mbps) |

Table 2: Hardware Requirement for client

**For server**

|  |  |  |
| --- | --- | --- |
| Windows | Minimum Requirements | Recommended |
| Operating System | Window Server 2008 | Window Server 2008 |
| Computer Processor | CPU Intel Xeon E3 2.0GHz | 2.0 GHz CPU 6 core E5-2620 |
| Computer Memory | 2 GB RAM DDR3 | 4 GB RAM DDR2 |
| HDD | 100 GB HDD Raid 0,1 | 500 GB HDD Raid 0,1 |
| Bandwidth | 300Mbps | 500Mbps |
| International bandwidth | 2Mbps | 10Mbps |

Table 3: Hardware Requirement for Server

##### Software requirements

* Window Server 2008: operating system for deploy web service.
* Microsoft Windows 7 Professional: operating system and platform for development.
* SQL Server 2008: used to create and manage the database for system.
* StarUML 5.0.2: used to create models and diagrams.
* Skype 7.0: used for communication and meeting.
* Visual Studio 2013: used to implement web-application

## Project organization

### Software Process Model

Project is developed under Sashimi Waterfall Model.

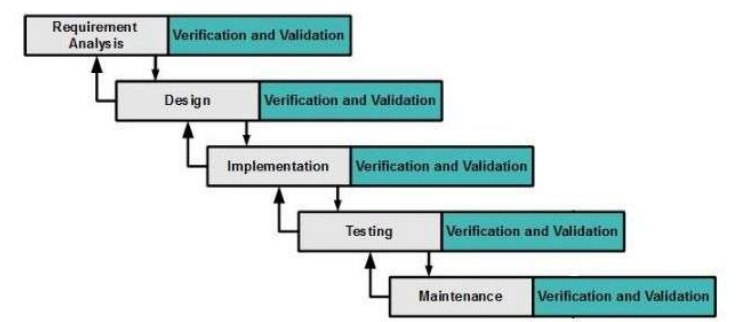


Figure 1: Sashimi Waterfall Model

For more information: <http://www.waterfall-model.com/sashimi-waterfall-model/>

The purpose of applying Sashimi Waterfall model for IMWHC is because of:

* Requirements of the project may be made clearly and difficult to change.
* In current phrase, if the error which in previous phrase or the update occur, we can return and fix it.

### Roles and responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full name** | **Role in Group** | **Responsibilities** |
| **1** | Ngo Dang Ha An | Project manager | * Specify user requirement * Control the development process * Give out technique and business analysis support |
| **2** | Tran Thi Dieu Thu | Team Leader, BA, DEV, Tester | * Managing process * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **3** | Le Tai Nhat Minh | Team Member, BA, DEV, Tester | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **4** | Nguyen Van Ngoc Truong | Team Member, BA, DEV, Tester | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **5** | Vo Van | Team Member, BA, DEV, Tester | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |

Table 4: Roles and Responsibilities Details

### Tools and Techniques

* Front-end technologies: HTML5, CSS3, JavaScript, jQuery, AJAX.
* Application is built on .NET platform.
* Web Server: Microsoft IIS version 7.
* Database Management System: MS SQL Server 2008 .

## Project Management Plan

### Software development life cycle

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Phase** | **Description** | **Deliverables** | **Resource needed** | **Dependencies and Constrains** | **Risks** |
| **Requirement Analysis** | - Collect requirements base on user’s behaviours  - Identify and clarify requirements for the system in general | - Introduction of project  - Software requirement specification  - Project task plan  - Prototypes | 40 man-days | N/A | - Missing requirement  - Unclear scope  - Not have a clear understanding about business process |
| **Design** | - Design system Architecture  - Design component Diagram, conceptual diagram  - Detail design | - Software design description  - Technology notes | 60 man-days | Depend on  “Requirement  Analysis” | - Lack of experience  - Not full fill requirements |
| **Implementation** | - Implement physical database  - Coding  system core  functions and  other feature  with GUI  - Unit test | - Physical database diagram  - Main user’s  functions on  website | 80 man-days | Depend on  “Design” | - Lack of experience  - Human mistake |
| **Testing** | - Write test case  - Do Integration  test the system test  - Do Alpha test  - Correct bugs  - Beta test  - Acceptance  test | - Test  document | 40 man-days | Depend on  “Implementation” | - Lack of experience  - Missing  test case |
| **Maintenance** | - Deploy on  sever | - Installation guide  - User Manual | 20 man-days | Depend on  “Testing” | - Lack of  experience |

Table 5: Software Development Life Cycle Detail

### Phase Detail

#### Phase 1: Requirement Analysis

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| 1. Collect requirements | - Analysis real user’s behaviours  - Find which systems currently  provide similar service, their  strengths and weaknesses | ThuTTD, MinhLTN, TruongNVN, VanV |
| 2. Identify and clarify  main functions | - Define which main functions system  should provide | ThuTTD, MinhLTN, TruongNVN, VanV |
| 3. Create System  Introduction | - Create Introduction Report. | ThuTTD, MinhLTN, TruongNVN, VanV |
| 4. Software Project  Management Plan | - Create Project Management Plan | ThuTTD |
| 5. Prototype | - Build a prototype of propose  website | MinhLTN, VanV |
| 6. SRS | - Create SRS document | ThuTTD, MinhLTN, TruongNVN, VanV |

Table 6: Phase 1 - Requirement Analysis

#### Phase 2: Design

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| 1.Design system Architecture | - Design system architecture  - Design component diagram | ThuTTD, MinhLTN, TruongNVN, VanV |
| 2. Database Design | - Based on parsed data to  recommendation  - Based on other needs to  recommendation | ThuTTD, MinhLTN, TruongNVN, VanV |
| 3.Detail design | - Design for each function | ThuTTD, MinhLTN, TruongNVN, VanV |
| 4. Technology  research | - Note some technology will be applied in project | TruongNVN |
| 5. Design Document | - Create software design document | ThuTTD, MinhLTN, TruongNVN, VanV |

Table 7: Phase 2 - Design

#### Phase 3: Implementation

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| 1. Physical database | - Implement physical database | ThuTTD |
| 2. Front-end web  functions | - Implement front-end functions  on web | ThuTTD, MinhLTN, TruongNVN, VanV |
| 3. Back-end web  functions | - Implement back-end functions on  web | ThuTTD, MinhLTN, TruongNVN, VanV |
| 4. Unit testing | - Write test case and testing for  web functions | ThuTTD, MinhLTN, TruongNVN, VanV |

Table 8: Phase 3 - Implementation

#### Phase 4: Testing

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| 1. Integration testing | - Write test case and testing  modules | ThuTTD, MinhLTN, TruongNVN, VanV |
| 2. System testing | - Write test case and testing  system | ThuTTD, MinhLTN, TruongNVN, VanV |
| 2. Alpha testing | - Do alpha test with customer | ThuTTD, MinhLTN, TruongNVN, VanV |

Table 9: Phase 4 - Testing

#### Phase 5: Maintenance

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| 1. Installation guide | - Write installation guide | ThuTTD, MinhLTN, TruongNVN, VanV |
| 2. User manual | - Write user manual | ThuTTD, MinhLTN, TruongNVN, VanV |

Table 10: Phase 5 - Maintenance

### All Meeting Minutes

Refer to Meeting Minutes folder in CD.

## Coding Convention

Using C# language to develop website:

* Naming Convention:

+Using Camel Case for method arguments and local variables.

+Using Pascal Case for class names and method names.

* Layout Convention:

+ Write only one statement per line.

+ Write only one declaration per line.

+ If continuation lines are not indented automatically, indent them one tab stop (four spaces).

+ Add at least one blank line between method definitions and property definitions.

+ Use parentheses to make clauses in an expression apparent, as shown in the following code.

* Commenting Convention:

+ Place the comment on a separate line, not at the end of a line of code.

+ Begin comment text with an uppercase letter.

+ End comment text with a period.

+ Insert one space between the comment delimiter (//) and the comment text, as shown in the following example.

Ex: // Here is your comment.

+ Do not create formatted blocks of asterisks around comments.

* Languages Guideline:

Using C# Coding Convention from:

<https://msdn.microsoft.com/en-us/library/ff926074.aspx>