** MINISTRY OF EDUCATION AND TRAINING**

**FPT UNIVERSITY**

Capstone Project Document

**Laptop Reviews**

|  |  |
| --- | --- |
| **Group 1** | |
| **Group members** | Vo Thi Minh Chau – Team Leader – SE60931  Nguyen Van Hon – Team Member - 60390  Dinh Huu Toan – Team Member - SE60871  Nguyen Manh Khuong – Team Member - 60455 |
| **Supervisor** | Mr. Kieu Trong Khanh |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** | LRA |

-Ho Chi Minh City, ***05/2015***-

*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](#_Toc419932189)

[List of Tables 5](#_Toc419932190)

[List of Figures 6](#_Toc419932191)

[Definitions, Acronyms, and Abbreviations 6](#_Toc419932192)

[Report No.2 Software Project Management Plan 7](#_Toc419932193)

[1. Problem Definition 7](#_Toc419932194)

[1.1 Name of this Capstone Project 7](#_Toc419932195)

[1.2 Problem Abstract 7](#_Toc419932196)

[1.3 Project Overview 7](#_Toc419932197)

[2. Project organization 8](#_Toc419932198)

[2.1 Software Process Model 8](#_Toc419932199)

[2.2 Roles and responsibilities 11](#_Toc419932200)

[2.3 Tools and Techniques 12](#_Toc419932201)

[3. Project Management Plan 12](#_Toc419932202)

[3.1 Software Development Life Cycle 12](#_Toc419932203)

[3.2 Phase Detail 14](#_Toc419932204)

[3.3 All Meeting Minutes 17](#_Toc419932205)

[4. Coding Convention 17](#_Toc419932206)

# List of Tables

[Table 1: Hardware Requirement for Server 8](#_Toc419932207)

[Table 2: Compare Plan-driven and Agile development 10](#_Toc419932208)

[Table 3: Roles and Responsibilities Details 12](#_Toc419932209)

[Table 4: Software Development Life Cycle Detail 14](#_Toc419932210)

[Table 5: Phase 1: Outline Planning and Architectural Design 14](#_Toc419932211)

[Table 6: Phase 2: Sprint Cycle – Parser Management 14](#_Toc419932212)

[Table 7: Phase 3: Sprint Cycle – Training Machine 15](#_Toc419932213)

[Table 8: Phase 3: Sprint Cycle – Dictionary Management 15](#_Toc419932214)

[Table 9: Phase 5: Sprint Cycle – Analyzing Comment Algorithm 16](#_Toc419932215)

[Table 10: Phase 6: Sprint Cycle – Main User’s Functions 16](#_Toc419932216)

[Table 11: Phase 7: Sprint Cycle – User Account Management 17](#_Toc419932217)

[Table 12: Phase 8: Project Closure 17](#_Toc419932218)

# List of Figures

[Figure 1: The Scrum Process 11](#_Toc419932219)

# Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| LRA | Laptop Reviews |
| UI | User interface |

# Report No.2 Software Project Management Plan

## Problem Definition

### Name of this Capstone Project

Laptop Reviews (LRA)

### Problem Abstract

Today, with the growth of the internet, when a customer want to buy a laptop, they can search for its specifications and other information through technology websites. But there are many reviews from experience users about a laptop that will take the customer a lot of times and may make them confused. Our system will help user to make a good decision when buying a laptop by gather reviews from trusted websites, classify and show them to users.

### Project Overview

#### Current Buying Habits

In Vietnam, people tend to choose laptops base on what they hear from sellers at electronic markets or what they read on some technical forums, websites. These activities have limitations. Sellers’ advices may be not accurate, some reviews may be non-sense and are not classified. Moreover, it will take lots of time for people to come to electronic markets to have laptops’ information or read reviews on many forums, websites. After searching on Google, we found this page: [www.buydig.com](http://www.buydig.com). They offer classified reviews for laptops but not all laptops. Therefore, customers will be upset when they can’t find what they need. Our solution will do a better job. We will gather and classify the reviews from trusted websites so that customers can make the best decision. Moreover, customers can claim for laptops which they can’t see on our system and get notification when those laptops’ information is updated.

#### The Proposed System

The system is intended for use by those with a smart phone or a laptop/computer with Internet connection. The system will have the following functions:

##### Web

* Admins can manage accounts.
* System can parse product, classify review and store to database.
* Staff can configure system, manage parser, train machine, manual update dictionary and check feedbacks from users.
* Users can search laptop’s information and give feedbacks.

#### Boundaries of the System

* The system can be used by every people with a smart phone or a laptop/computer with Internet connection.
* 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 | Windows Server 2008 R2 | Windows Server 2008 R2 |
| Computer Processor | Intel® Core 2 Duo 1.4 GHz | Intel® Core(TM) i7 CPU , M 460 @ 2.53GHz |
| Computer Memory | 512 MB RAM | 2GB or more RAM |

Table 1: Hardware Requirement for Server

##### Software requirements

* SQL Server 2008 Express R2: used to create and manage the database for system.
* StarUML 5.0: used to create models and diagrams.
* Skype: used for communication and meeting.
* Visual Studio 2013: used to implement website and web service.
* Github.com & TortoiseSVN: used for source control.

## Project organization

### Software Process Model

To determine the right process model for the software process, we answer ten guidelines questions (Sommerville, Software Engineering, 2011)

| No. | Question | Answer | Plan-Driven | Agile |
| --- | --- | --- | --- | --- |
| 1. | Is it important to have a very detailed specification and design before moving to implementation? If so, you probably need to use a plan-driven approach. | We need a correct specification and design to code but a very detailed one is not necessary. | 0.5 | 0.5 |
| 2. | Is an incremental delivery strategy, where you deliver the software to customers and get rapid feedback from them, realistic? If so, consider using agile methods. | Yes. Our product need rapid feedback from customers so that we can modify to satisfy them. | 0 | 1 |
| 3. | How large is the system that is being developed? Agile methods are most effective when the system can be developed with a small co-located team who can communicate informally. This may not be possible for large systems that require larger development teams so a plan-driven approach may have to be used. | This system can be developed with a 4-member-team and we are at the same age so we can communicate informally. | 0 | 1 |
| 4. | What type of system is being developed? Systems that require a lot of analysis before implementation (e.g., real-time system with complex timing requirements) usually need a fairly detailed design to carry out this analysis. A plan-driven approach may be best in those circumstances. | This is an ecommerce system. We need some careful analysis but not too many. We don’t have many complex timing requirements | 0.5 | 0.5 |
| 5. | What is the expected system lifetime? Long-lifetime systems may require more design documentation to communicate the original intentions of the system developers to the support team. However, supporters of agile methods rightly argue that documentation is frequently not kept up to date and it is not of much use for long-term system maintenance. | This system may operate in two or three years. | 1 | 0 |
| 6. | What technologies are available to support system development? Agile methods often rely on good tools to keep track of an evolving design. If you are developing a system using an IDE that does not have good tools for program visualization and analysis, then more design documentation may be required. | We have a good tool to track changes. The IDE support tracking phase. | 0.5 | 0.5 |
| 7. | How is the development team organized? If the development team is distributed or if part of the development is being outsourced, then you may need to develop design documents to communicate across the development teams. You may need to plan in advance what these are. | We all live in Ho Chi Minh City so that we can have meetings every day | 0 | 1 |
| 8. | Are there cultural issues that may affect the system development? Traditional engineering organizations have a culture of plan-based development, as this is the norm in engineering. This usually requires extensive design documentation, rather than the informal knowledge used in agile processes. | We are all young people and we share common software development knowledge. Moreover, our own cultures are similar to each other | 0 | 1 |
| 9. | How good are the designers and programmers in the development team? It is sometimes argued that agile methods require higher skill levels than plan-based approaches in which programmers simply translate a detailed design into code. If you have a team with relatively low skill levels, you may need to use the best people to develop the design, with others responsible for programming. | Although we are confident about software development skill. We are still student and we have no or little experience. | 1 | 0 |
| 10. | Is the system subject to external regulation? If a system has to be approved by an external regulator (e.g., the Federal Aviation Authority [FAA] approve software that is critical to the operation of an aircraft) then you will probably be required to produce detailed documentation as part of the system safety case. | No | 0 | 0 |
| Total | | | 3.5 | 5.5 |

Table 2: Compare Plan-driven and Agile development

After carefully considering the two approaches, we can see that agile methods are more suitable for our project. We decide to choose the Scrum approach which is a general agile method.

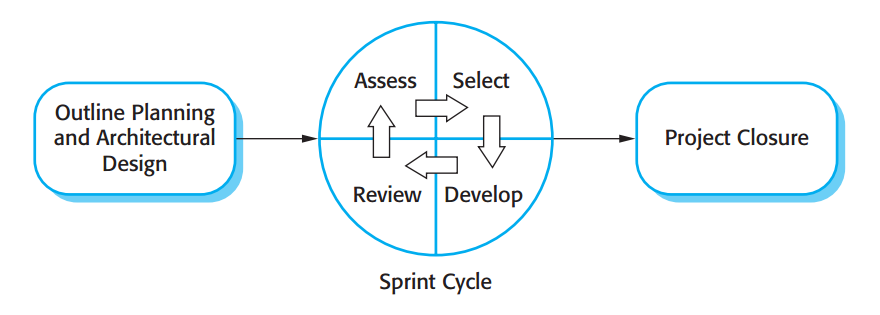


Figure 1: The Scrum Process

For more information: Sommerville, Software Engineering, 2011

### 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** | Vo Thi Minh Chau | Team Leader, BA, DEV, Tester | * Managing process * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **3** | Nguyen Van Hon | Team Member, BA, DEV, Tester | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **4** | Dinh Huu Toan | Team Member, BA, DEV, Tester | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **5** | Tran Manh Khuong | 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: HTML5, CSS3, JavaScript, jQuery 1.10, AJAX.

- Back-end: Website: ASP.NET MVC5 + Entity Framework 5.

- Web Server: Microsoft IIS 7.

- Database Management System: MS SQL Server 2008 Express R2.

## Project Management Plan

### Software Development Life Cycle

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Phase**  **/Iteration** | **Description** | **Deliverables** | **Resource needed** | **Dependencies and Constrains** | **Risks** |
| **Outline Planning and Architectural Design** | - Study similar existing systems.  -Identify and clarify requirements for the system in general. | - Introduction of proposed system.  - Project task plan.  - Software requirement specification.  - Prototypes. | 20 man-days | N/A | - Missing  requirement  - Unclear  scope of  project  - Lack of  member share  of understand |
| **Sprint Cycle – Parser Management** | - Collect data from websites.  - Manage parser. | - Collected data.  - Parser management service. | 30 man-days | N/A | - Lack of experience.  - The implemented parsers are not the best.  - Lack of test data. |
| **Sprint Cycle – Training Machine** | -Teach the system how to synchronize products’ names. | N/A | 20 man-days | Depends on “Data management”. | - Not have a clear understanding about business process. |
| **Sprint Cycle – Dictionary Management** | - Input data manually.  - Import data from excel files.  - Find synonyms and antonyms form dictionary websites. | - Dictionary management service. | 30 man-days | N/A | - Lack of experience.  - The dictionary is not variety. |
| **Sprint Cycle – Analyzing Comment Algorithm** | - Build algorithm to analyze comments then classify them into 3 groups: positive, negative and neutral. | - Classified comment system. | 30 man-days | Depend on “Data management” and “Dictionary management”. | - The implemented algorithm is not the best.  - Lack of test data.  - Lack of experience on analyzing sentence’s meaning. |
| **Sprint Cycle – Main User’s Functions** | - User can search a product.  - Let user see classified comments of each product.  - Member (staff and admin) can manage profile. | - Main user’s functions on web. | 7 man-days | Depend on “Data management”, “Dictionary management” and “Analyzing Comment Algorithm”. | - Lack of experience.  - Not have a clear understanding about business process. |
| **Sprint Cycle – User Account**  **Management** | - Manage user accounts in the system. | - Account management system. | 5 man-days | N/A | - Lack of experience.  - Not have a clear understanding about business process. |
| **Project Closure** | - Documentation. | - Installation guide.  - User manual. | 3 man-days | Depend on “Sprint Cycle” | - Lack of experience. |

Table 4: Software Development Life Cycle Detail

### Phase Detail

#### Phase 1: Outline Planning and Architectural Design

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Identifying and studying existing systems** | - Find which systems currently provide similar service, their strengths and weakness. | ChauVTM, HonNV, ToanDH, KhuongNM |
| **2. Identifying and clarifying main functions.** | - Define which main functions system should provide. | ChauVTM |
| **3. Introduction.** | - Complete Introduction Report. | ToanDH, KhuongNM |
| **4. Project Management**  **Plan.** | - Prepare Project  - Management Plan. | ChauVTM |
| **5. Website Prototype.** | - Build a prototype of proposed system (Website). | ToanDH, HonNV |
| **6. Design Entity Relationship diagram** | - Design Entity Relationship diagram. | ChauVTM, HonNV, ToanDH, KhuongNM |

Table 5: Phase 1: Outline Planning and Architectural Design

#### Phase 2: Sprint Cycle – Parser Management

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Assess** | - Determine the list of work to be done. | ChauVTM |
| **2. Select** | - Which feature this function should have and how to implement. | ChauVTM |
| **3. Develop** | - Add SRS, SDD.  - Create appropriate parsers to parse data from many websites.  - Create the interface for user.  - Test system behavior and performance.  - Test user behavior and performance. | ChauVTM, |
| **4. Review** | - Review and present to stakeholders. | ChauVTM, HonNV, ToanDH, KhuongNM |

Table 6: Phase 2: Sprint Cycle – Parser Management

#### Phase 3: Sprint Cycle – Training Machine

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Assess** | - Determine the list of work to be done. | ChauVTM |
| **2. Select** | - Which feature this function should have and how to implement. | ToanDH |
| **3. Develop** | - Add SRS, SDD.  - Compare many string comparison algorithms and choose the best one.  - Implement algorithm.  - Implement merging and splitting products’ names function.  - Create the interface for user.  - Test system behavior and performance.  - Test user behavior and performance. | ToanDH |
| **4. Review** | - Review and present to stakeholders. | ChauVTM, HonNV, ToanDH, KhuongNM |

Table 7: Phase 3: Sprint Cycle – Training Machine

#### Phase 4: Sprint Cycle – Dictionary Management

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Assess** | - Determine the list of work to be done. | ChauVTM |
| **2. Select** | - Which feature this function should have and how to implement. | ChauVTM |
| **3. Develop** | - Add SRS, SDD.  - Build a function which let user input data by manually inputting and importing excel file.  - Build a function which let system find synonyms and antonyms of existed words in dictionary from dictionary websites automatically.  - Create the interface for user.  - Test system behavior and performance.  - Test user behavior and performance. | ChauVTM, ToanDH, KhuongNM |
| **4. Review** | - Review and present to stakeholders. | ChauVTM, HonNV, ToanDH, KhuongNM |

Table 8: Phase 3: Sprint Cycle – Dictionary Management

#### Phase 5: Sprint Cycle – Analyzing Comment Algorithm

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Assess** | - Determine the list of work to be done. | ChauVTM |
| **2. Select** | - Which feature this function should have and how to implement. | ChauVTM |
| **3. Develop** | - Add SRS, SDD.  - Compare many algorithms and choose the best one.  - Implement the chosen algorithm to classify the comments into 3 groups: positive, negative and neutral.  - Test system behavior and performance. | ChauVTM, HonNV |
| **4. Review** | - Review and present to stakeholders. | ChauVTM, HonNV, ToanDH, KhuongNM |

Table 9: Phase 5: Sprint Cycle – Analyzing Comment Algorithm

#### Phase 6: Sprint Cycle – Main User’s Functions

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Assess** | - Determine the list of work to be done. | ChauVTM |
| **2. Select** | - Which feature this function should have and how to implement. | HonNV |
| **3. Develop** | - Add SRS, SDD.  - Allow staff to manage user accounts.  - Allow user view details of the product.  - Allow user to search product.  - Test system behavior and  Performance.  - Test user behavior and  Performance. | HonNV |
| **4. Review** | - Review and present to stakeholders. | ChauVTM, HonNV, ToanDH, KhuongNM |

Table 10: Phase 6: Sprint Cycle – Main User’s Functions

#### Phase 7: Sprint Cycle – User Account Management

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Assess** | - Determine the list of work to be done. | ChauVTM |
| **2. Select** | - Which feature this function should have and how to implement. | KhuongNM |
| **3. Develop** | - Add SRS, SDD.  - Staff can manage accounts in the system.  - Test system behavior and  Performance.  - Test user behavior and  Performance. | KhuongNM |
| **4. Review** | - Review and present to stakeholders. | ChauVTM, HonNV, ToanDH, KhuongNM |

Table 11: Phase 7: Sprint Cycle – User Account Management

#### Phase 8: Project Closure

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Documentation** | - Complete Installation Guide, Manual Guide. | ChauVTM, HonNV, ToanDH, KhuongNM |
| **2. Assess** | - Assess the lessons learned from the project. | ChauVTM, HonNV, ToanDH, KhuongNM |

Table 12: Phase 8: Project Closure

### All Meeting Minutes

Refer to Meeting Minutes folder.

## Coding Convention

C#: Using to develop website.

Summary:

* Naming Convention:
  + For variable’s name, use camel case. Eg: minValue, maxValue,…
  + For function name, class name, use pascal case. Eg: SearchProduct, ImportFile,…
* Layout Convention:
  + Write only one statement/declaration per line.
  + Indent continuation 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.
* Language Guidelines:

Using C# Code Convention From:

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