** MINISTRY OF EDUCATION AND TRAINING**

**FPT UNIVERSITY**

Capstone Project Document

**Library Automation using RFID**

|  |  |
| --- | --- |
| **Group 04** | |
| **Group members** | Vũ Trần Hậu – Team Leader – SE60881  Hồ ĐắcVĩnh Ân – Team Member –60406  Trần Nhật Duật – Team Member –SE61033  Trần TrungThiên – Team Member – SE60656  Nguyễn Xuân Ý – Team Member – SE60869 |
| **Supervisor** | MSc. Đỗ Đức Minh Quân |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** | LAR |

-Ho Chi Minh City, 01/2015-

*This page is intentionally left blank*

# Table of Contents

[Table of Contents 3](#_Toc409471390)

[List of Tables 4](#_Toc409471391)

[List of Figures 5](#_Toc409471392)

[Definitions, Acronyms, and Abbreviations 5](#_Toc409471393)

[B. Report No.2 Software Project Management Plan 6](#_Toc409471394)

[1. Problem Definition 6](#_Toc409471395)

[1.1 Name of this Capstone Project 6](#_Toc409471396)

[1.2 Problem Abstract 6](#_Toc409471397)

[1.3 Project Overview 6](#_Toc409471398)

[2. Project organization 8](#_Toc409471399)

[2.1 Software Process Model 8](#_Toc409471400)

[2.2 Roles and responsibilities 8](#_Toc409471401)

[2.3 Tools and Techniques 9](#_Toc409471402)

[3. Project Management Plan 9](#_Toc409471403)

[3.1 Software development life cycle 9](#_Toc409471404)

[3.2 Phase Detail 10](#_Toc409471405)

[3.3 All Meeting Minutes 11](#_Toc409471406)

[4. Coding Convention 11](#_Toc409471407)

# List of Tables

[Table 1: Hardware Requirement for Application 8](#_Toc409471730)

[Table 2: Hardware Requirement for Components 8](#_Toc409471731)

[Table 3: Roles and Responsibilities Details 9](#_Toc409471732)

[Table 4: Software Development Life Cycle Detail 10](#_Toc409471733)

[Table 5: Phase 1: Requirement Analysis 10](#_Toc409471734)

[Table 6: Phase 2: Design 11](#_Toc409471735)

[Table 7: Phase 3: Implementation 11](#_Toc409471736)

[Table 8: Phase 4: Testing 11](#_Toc409471737)

[Table 9: Phase 5: Maintenance 11](#_Toc409471737)

# List of Figures

[Figure 1: Modified Waterfall Development Model 8](#_Toc398556983)

# Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| **Name** | **Definition** |
|  |  |

# Report No.2 Software Project Management Plan

## Problem Definition

### Name of this Capstone Project

* Library Automation using RFID (LAR)

### Problem Abstract

Library is something familiar with each of us. Library is the collection of books, newspapers, magazines, documents and some kind of pictures, music or event painting. They contain huge amounts of information and knowledge. So the management library is not simple thing. Students who wish to borrow and return books must undergo a relatively complex process. The librarian event much more, they have to manage all borrow-return book information with a lot of thing to do. Our system will help them by Automation the return-borrowed books phase with a simple interface and easy to use. This will become much simpler.

In addition, the system also has security system. The book is not borrowed that accidentally or intentionally taken can be detected immediately. It means that the theft of the library will be stopped completely. The book will be protected effectively.

### Project Overview

#### Current Situation and Disadvantages

There are some types of library management as below:

* Manual traditional library management
  + All activities that relate to borrowing and returning books are manual.
  + Users select books then come to librarian position to fill a form. To borrow a book, users have to fill information of that book such as: ID, title, writer, etc…
  + Librarians have to check that information. Then he/she decides which books users can borrow.
* Library management using barcode
  + Each book has a barcode which is used to identify a book. All barcodes are different with each other.
  + When users want to borrow book, librarians will use a scanner to read the barcode. Book’s information is showed on computer screen.

However, current library management methods have some disadvantages:

* In manual traditional method
  + Users have to do many steps to borrow books. They have to fill all required information of books:
  + The time for each user to complete the borrowing process is so long. It wastes both of users and librarians a lot of time.
* In method using barcode:
  + The barcode needs to be seen clearly by scanner. If the barcode is not clear or any else problems with reading process, book’s information will not be read or will be wrong.

#### The Proposed System

To improve quality and control effective, reducing the process of borrowing and returning the books. We have an idea about the system control library uses RFID technology (Radio Frequency Identification).

##### Desktop application

* + For Librarians:
    - Librarians can manage books (add, update, delete).
    - Librarians can manage student’s information.
    - Can define students who borrow books and number of books which he/she had borrowed.
    - Can define the books which students want to borrow (return) and update to book’s deport, auto update the book’s number which students are borrowing.
  + For students:
    - They can finish borrow-return books process right at borrow area.
    - They can check how much books they are borrowing.

##### Hardware

* + The system can read and write information of the card on books and student’s card. Then, we can know which students are borrowing which books.
  + Transmit data between reader and microcontroller to handle.

#### Boundaries of the System

* The system can only used by the librarian in library with application on PC.
* The system can work fully function with Internet connection.
* The complete product will includes :
  + An application, for librarian only.
  + Components which use to write and read RFID tags.
  + Components which use to scan RFID tags as security gate form.
  + Wires, batteries, frames and other necessary.

#### Development Environment

##### Hardware requirements

**For PC application**

|  |  |  |
| --- | --- | --- |
|  | Minimum Requirements | Recommended |
| Internet Connection | Cable, Wi-Fi (4 Mbps) | Cable, Wi-Fi (8 Mbps) |
| Operating System | Window XP | Window XP |
| Computer Processor | Intel® Xeon ® 1.4GHz | Intel® Xeon ® Quad Core (12M Cache, 2.50 GHz) |
| Computer Memory | 1GB RAM | 2GB or more |

Table 1: Hardware Requirement for Application

**For Components**

|  |  |
| --- | --- |
|  | Requirements |
| Communication | Wi-Fi, UART, cable, SPI |
| Development languages | C or assembly |
| Power | 5V / 12A(battery or supplier) |
| Memory | 512MB |

Table 2: Hardware Requirement for Components

##### Software requirements

* Window XP: operating system and platform for development.
* SQL Server 2008 Enterprise R2: used to create and manage the database for system.
* Visual Studio 2012: used to implement website and web service.
* Code Vision: used to develop and embedded code to device.
* Google Code & TortoiseSVN: used for source control.
* StarUML: used to create models and diagrams.
* Skype: used for communication and meeting.
* Arduino IDE: for C development.

## Project organization

### Software Process Model

Project is developed under modified waterfall model (SASHIMI).

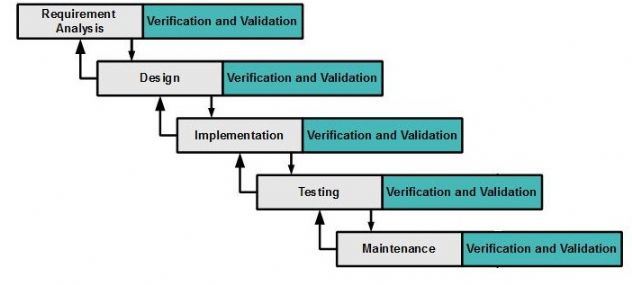


Figure 1: Modified Waterfall Development Model

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

### Roles and responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full name** | **Role in Group** | **Responsibilities** |
| **1** | Đỗ Đức Minh Quân | Supervisor, Project Manager | * Specify user requirement * Control the development process * Give out technique and business analysis support |
| **2** | Vũ Trần Hậu | Team Leader, Developer, Tester | * Managing process * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **3** | Hồ Đắc Vĩnh Ân | Team Member, Developer, Tester | * Designing database * Clarifying requirements * Prepare documents * Coding * Testing |
| **4** | Trần Nhật Duật | Team Member, Developer, Tester | * Designing database * Clarifying requirements * Prepare documents * Coding * Testing |
| **5** | Trần Trung Thiên | Team Member, Developer, Tester | * Designing database * Clarifying requirements * Prepare documents * Coding * Testing |
| **6** | Nguyễn Xuân Ý | Team Member, Developer, Tester | * Designing database * Clarifying requirements * Prepare documents * Coding * Testing |

Table 3: Roles and Responsibilities Details

### Tools and Techniques

## Project Management Plan

### Software development life cycle

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Phase** | **Description** | **Deliverables** | **Resource needed** | **Dependencies and Constrains** | **Risks** |
| **Requirement Analysis** | - Collect requirements.  -Identify and clarify requirements for the system in general. | -Introduction of proposed system.  -Software requirement specification.  -Project Task Plan. | 15 man-days | N/A | - Missing requirement  - Unclear scope of project  - Lack of member share of understand |
| **Design** | - Architecture design for the system  - Detail design  - Database design | - Software Design Document | 20 man-days | Depend on “Requirement Analysis” | - Lack of experience.  - Not fulfil requirement. |
| **Implementation** | - Coding system core functions  - Develop desktop application  - Unit test | - Main functions of the system  - Desktop application  - Unit test document | 50 man-days | Depend on “Design”. | - Lack of experience and knowledge.  - Human mistake. |
| **Testing** | - Integration test the system  - Correct bugs | - Test document  - Defect log | 20 man-days | Depend on “Implementation” | - Lack of experience  - Missing test case |
| **Maintenance** | - Integrated installation | - Installation guide  - User Manual | 10 man-days | Depend on “Testing” | - Lack of experience. |

Table 4: Software Development Life Cycle Detail

### Phase Detail

#### Phase 1: Requirement Analysis

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Collect requirements** | List out current similar systems and their strengths and weaknesses. | HauVT, ThienTT, YNX |
| **2. Identify and clarify main functions.** | Define which main functions system should provide. | DuatTN, AnHDV |
| **3. Create System Introduction.** | Complete Introduction Report. | HauVT, DuatTN, AnHDV, ThienTT, YNX |
| **4. Software Project Management Plan.** | Prepare Project Management Plan. | HauVT |
| **5. SRS** | Create SRS document. | HauVT, DuatTN, AnHDV, ThienTT, YNX |

Table 5: Phase 1: Requirement Analysis

#### Phase 2: Design

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Architecture Design** |  | HauVT, DuatTN, AnHDV, ThienTT, YNX |
| **2. Detailed Design** |  | HauVT, DuatTN, AnHDV, ThienTT, YNX |
| **3. Database Design** |  | HauVT, DuatTN, AnHDV, ThienTT, YNX |

Table 6: Phase 2: Design

#### Phase 3: Implementation

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
|  |  |  |
|  |  |  |
| **4. Suggestion algorithms** |  |  |
| **5. Unit testing** |  | HauVT, DuatTN, AnHDV, ThienTT, YNX |

Table 7: Phase 3: Implementation

#### Phase 4: Testing

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Integration testing** | Write test case and testing system | HauVT, DuatTN, AnHDV, ThienTT, YNX |
| **2. Correct bugs** | Correct defects, bugs appear while doing test | HauVT, DuatTN, AnHDV, ThienTT, YNX |

Table 8: Phase 4: Testing

#### Phase 5: Maintenance

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Installation guide** | Write installation guide | HauVT |
| **2. User Manual** | Write user manual | DuatTN, AnHDV |

Table 9: Phase 5: Maintenance

### All Meeting Minutes

Refer to Meeting Minutes folder.

## Coding Convention

C#: Using to develop desktop application.

Summary:

* Naming Convention:
  + Use camel case for variable’s name. Eg: minValue, maxValue…
  + For function name, class name, use Pascal case. Eg: SearchEvent, GetRecommendEvent…
* 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.
* 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.
  + Do not create formatted blocks of asterisks around comments.
* Language Guidelines:

Using C# Code Convention From:

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

C/C++: Using to develop hardware

Summary:

* Naming convention:
  + Use camel case for variable’s name. Eg: minValue, maxValue…
  + For function name, class name, use Pascal case. Eg: SearchEvent, GetRecommendEvent…

Using C/C++ code convention from:

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