** MINISTRY OF EDUCATION AND TRAINING**

**FPT UNIVERSITY**

Capstone Project Document

Capstone Project Management

|  |  |
| --- | --- |
| **Group 13** | |
| **Group members** | Trần Nguyễn Đăng Khoa – Team Leader – SE60680  Đặng Ngọc Huy – Team Member – SE60913  Nguyễn Hoàng Tân – Team Member - SE60819 |
| **Supervisor** | Mr. Kiều Trọng Khánh |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** | MSSC |

-Ho Chi Minh City, 09/2014-

*This page is intentionally left blank*

# Table of Contents

[Table of Contents 4](#_Toc398517987)

[List of Tables 5](#_Toc398517988)

[List of Figures 6](#_Toc398517989)

[Definitions, Acronyms, and Abbreviations 6](#_Toc398517990)

[B. Report No.2 Software Project Management Plan 7](#_Toc398517991)

[1. Problem Definition 7](#_Toc398517992)

[1.1 Name of this Capstone Project 7](#_Toc398517993)

[1.2 Problem Abstract 7](#_Toc398517994)

[1.3 Project Overview 7](#_Toc398517995)

[2. Project organization 9](#_Toc398517996)

[2.1 Software Process Model 9](#_Toc398517997)

[2.2 Roles and responsibilities 10](#_Toc398517998)

[2.3 Tools and Techniques 10](#_Toc398517999)

[3. Project Management Plan 12](#_Toc398518000)

[3.1 Iteration 12](#_Toc398518001)

[3.2 Iteration Detail 13](#_Toc398518002)

[3.3 All Meeting Minutes 14](#_Toc398518003)

[4. Coding Convention 15](#_Toc398518004)

# List of Tables

[Table 1: Hardware Requirement for Server 9](#_Toc398557029)

[Table 2: Hardware Requirement for Mobile 9](#_Toc398557030)

[Table 3: Roles and Responsibilities Details 10](#_Toc398557031)

[Table 4: Software Development Life Cycle Detail 11](#_Toc398557032)

[Table 5: Phase 1: Requirement Analysis 12](#_Toc398557033)

[Table 6: Phase 2: Design 12](#_Toc398557034)

[Table 7: Phase 3: Implementation 12](#_Toc398557035)

[Table 8: Phase 4: Testing 12](#_Toc398557036)

# List of Figures

[Figure 1: Modified Waterfall Development Model 9](#_Toc398556983)

# Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| **Name** | **Definition** |
| MSSC | Multi Services Card |
| App | Application |
| OS | Operating System |
| Admin | Administrator |
| API | Application Programming Interface |
| HTTP | Hyper Text Transfer Protocol |
| Big 4 | Cloud, Mobile, Social, Data Analytics |
| 3G | Third generation of mobile telecommunications |
| Business Card | The NFC card with contain contact information |
| Contact Card | The NFC card with contain contact information |
| Event Ticket | The NFC card with contain event ticket information |

# Report No.2 Software Project Management Plan

## Problem Definition

### Name of this Capstone Project

* M-Services Card (MSSC)

### Problem Abstract

Most of the information system purposes are to provide effective ways in storing, analyzing, and processing data. After collecting information, many transition rule and knowledge learning processes are applied to mining the data. However, the correctness of data depends on input process. If the input process is taken by human, the correctness of data can’t be guarantee always true. Mistakes can happen easily due to various reasons: visual impairments, noise distraction, and fat finger error… in both objective and subjective ways.

Furthermore, exchange data is no longer the case that only happened between different data systems. For considering, information exchange between humans has long history of using communication channels such as voice, text, and image… Accompanied with channels are barriers that can cause misunderstand, misconstrue, and inaccurate…

These two problems are addressed in this project as our purpose is to answer the question: “Can we apply NFC technology to overcome these problems in order to enhance the process of recognize data by eyes and accelerate the exchange information speed?”

To emphasis on our narrow issues addressed in this project scope, we want to explore the NFC feature of smart phone device on Android OS, to support users manage their business cards and provide an e-Ticket for event attendance.

### Project Overview

#### Current Situation and Disadvantages

Below are some current behaviors of user:

* Name card exchange:
  + User makes a name card which personal information is written on
  + User gives his/her name card to business partner. The receiver needs to read the card and save the information usually on cellphone (or memory, or phonebook) for later reference.
  + When user changes information, they throw their old card and make new one. They inform the changes to their partner by their new card or via some communication channels (SMS, email, voice message…). Partner must update the information manually.
* Event ticket:
  + User purchases an event ticket on online website or at retailer store.
  + At event time, user brings his/her ticket to event gate, waits in line for check-in procedure. An event’s staff will collect ticket from user, use his/her eyes to check the ticket format and does other check-in required procedures.
  + In case of paper ticket, staffs will have to spend time counting to get statistic information.
  + In case of QR code ticket, staffs will have to use scanner device to decode the picture.

Below are the disadvantages of current situation:

* The process of recognize data by human eye and transfer it to information is take a lot of time and may be affected by human errors.
* No centralized approach on the case of changing information
* For event ticket, remembering a code may obstruct user.
* Actual paper ticket may be damaged on delivery or carry
* No real time statics can be count for event providers
* The process of scanning QR code, bar code sometime cause bad performance and depend on the quality of the image and the camera.

Our project is also taking into consideration about:

- Technology Feasibility:

- NFC applied in Android implementation [Ref:]

- NFC compare with Bluetooth [Ref:]

- NFC specification [Ref:]

- Social study report:

- “Vietnam ranks third in the world for the number of first-time smartphone users” [Ref:]

- “Smartphone orders occupy more than 35% density on e-commerce websites” [Ref:]

#### The Proposed System

We propose a new system to take advantages of NFC feature on Android smart phone to try to solve those disadvantages. Our system includes a web application and an Android application to help user manage name card information and sanitate event attendance check-in process.

##### Web Site

* Admins can manage the system, manage accounts.
* Staffs can approve change request from partners.
* Partners can sell event tickets online.
* Partners can replies to a comment made by member about their events.
* Partners can view statics information of their events.
* Partners can change some basic information of their profiles and their events but will have to be approved by Admins.
* Users can purchase event tickets online.
* Users can comment on an event or add an event to their wish list.
* Users can change their profile information and notification will be sent to their friends and otherwise.
* System provides user suggested events and suggested relationships.

##### Mobile Application

* Users can touch NFC business card to store contacts into their online phonebooks.
* Users can write their own business card.
* Users can write their own ticket to an NFC card.
* Partners can read NFC tickets and do check-in process. Also, they can view statics information of their events.

#### Boundaries of the System

* The system can be used by every people with a smart phone which enable NFC feature and a laptop/computer with Internet connection.
* The language of the system is Vietnamese.
* The complete product includes:

+ The website, for admins, partners and users.

+ The mobile applications, for partners and users.

+ All the process document involved.

#### Development Environment

##### Hardware requirements

**For server**

|  |  |  |
| --- | --- | --- |
| Windows | Minimum Requirements | Recommended |
| Internet Connection | Cable, Wi-Fi (4 Mbps) | Cable, Wi-Fi (8 Mbps) |
| Operating System | Window Server 2008 | Window Server 2008 |
| 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 Server

**For Mobile**

|  |  |  |
| --- | --- | --- |
| Mobile | Minimum Requirements | Recommended |
| Internet Connection | 2 Mbps | 4 Mbps |
| Operating System | Android 4.0 | Android 4.4.2 |
| Hardware | NFC supported | NFC supported |
| Memory | 512MB | 1GB or more |

Table 2: Hardware Requirement for Mobile

##### Software requirements

* Window Server 2008: 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.
* Google Code & TortoiseSVN: used for source control.
* StarUML: used to create models and diagrams.
* Skype: used for communication and meeting.

## Project organization

### Software Process Model

Project is developed under modified waterfall model (SASHIMI).

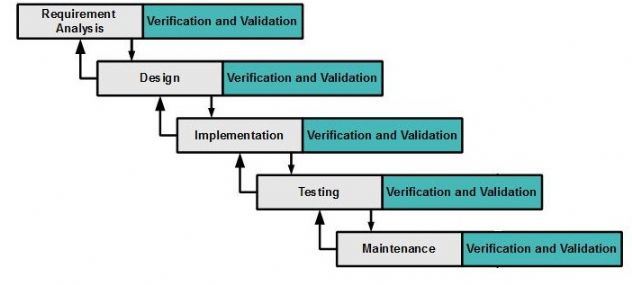


Figure 1: Modified Waterfall Development Model

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

### Roles and responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full name** | **Role in Group** | **Responsibilities** |
| **1** | Kiều Trọng Khánh | Project manager | * Specify user requirement * Control the development process * Give out technique and business analysis support |
| **2** | Trần Nguyễn Đăng Khoa | Team Leader, BA, DEV, Tester | * Managing process * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **3** | Đặng Ngọc Huy | Team Member, BA, DEV, Tester | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **4** | Nguyễn Hoàng Tân | 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, AJAX.

- Back-end technologies:

+ Website: ASP.NET MVC5 + Entity Framework 6.

+ Web service: ASP.NET Web API 2

- Mobile: Android KitKat 4.4.2 – Java 7

- Web Server: Microsoft IIS 7.5 with .Net Framework 4.5.1 enable.

- Database Management System: MSSQL Server 2008 R2 Enterprise.

## Project Management Plan

### Software development life cycle

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Phase** | **Description** | **Deliverables** | **Resource needed** | **Dependencies and Constrains** | **Risks** |
| **Requirement Analysis** | - Collect requirements from customer.  -Identify and clarify requirements for the system in general. | -Introduction of proposed system.  -Software requirement specification.  -Project Task Plan.  - Prototypes | 20 man-days | N/A | - Missing requirement  - Unclear scope of project  - Lack of member share of understand |
| **Design** | - Architecture design for the system  - Detail design using top-down break down  - Choose Architecture style | - Software Design Document  - Base code structure  - Technology notes | 20 man-days | Depend on “Requirement Analysis” | - Lack of experience.  - Not fulfil requirement. |
| **Implementation** | - Coding system core functions and other feature with GUI  - Unit test | - Main user’s functions on web and mobile.  - Unit test document | 50 man-days | Depend on “Design”. | - Lack of experience and knowledge.  - Human mistake. |
| **Testing** | - Integration test the system  - Alpha test  - Correct bugs  - Beta test  - Acceptance test | - Test document  - Defect log | 20 man-days | Depend on “Implementation” | - Lack of experience  - Missing test case |
| **Maintenance** | - Deploy on sever and mobile | - 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** | Find which systems currently provide similar service, their strengths and weakness. | KhoaTND, HuyDN, TanNH |
| **2. Identify and clarify main functions.** | Define which main functions system should provide. | KhoaTND, HuyDN, TanNH |
| **3. Create System Introduction.** | Complete Introduction Report. | KhoaTND |
| **4. Software Project Management Plan.** | Prepare Project Management Plan. | KhoaTND |
| **5. Website Prototype.** | Build a prototype of proposed system (Website/Mobile). | KhoaTND, HuyDN, TanNH |
| **6. SRS** | Create SRS document. | KhoaTND, HuyDN, TanNH |

Table 5: Phase 1: Requirement Analysis

#### Phase 2: Design

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Architecture Design** | Implement function import and breakdown data from docx files. | KhoaTND, HuyDN, TanNH |
| **2. Detailed Design** | Compare new document with existed documents of system. | KhoaTND, HuyDN, TanNH |
| **3. Database Design** | Get jobs from other server to recommendation. | KhoaTND, HuyDN, TanNH |
| **4. Technology research** | Create search engine for basic search and advance search. | KhoaTND, HuyDN, TanNH |
| **5. Design Document** | Create software design document | KhoaTND, HuyDN, TanNH |

Table 6: Phase 2: Design

#### Phase 3: Implementation

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Front-end web functions** | Implement front-end functions on web | KhoaTND, HuyDN, TanNH |
| **2. Back-end web functions** | Implement back-end functions on web | KhoaTND, HuyDN, TanNH |
| **3. Mobile functions** | Implement mobile application | HuyDN, TanNH |
| **4. Suggestion algorithms** | Research and implement suggestion algorithms | KhoaTND |
| **5. Unit testing** | Write test case and testing for web functions | KhoaTND, HuyDN, TanNH |
| Write test case and testing for mobile functions | HuyDN, TanNH |

Table 7: Phase 3: Implementation

#### Phase 4: Testing

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Integration testing** | Write test case and testing system | KhoaTND, HuyDN, TanNH |
| **2. Alpha testing** | Do alpha test with customer | KhoaTND, HuyDN, TanNH |

Table 8: Phase 4: Testing

#### Phase 5: Maintenance

|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Author** |
| **1. Installation guide** | Write installation guide | KhoaTND |
| **2. User Manual** | Write user manual | KhoaTND, HuyDN, TanNH |

Table 9: Phase 4: User Interface

### 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: 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>

Java: Using to develop mobile application.

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>