**MINISTRY OF EDUCATION AND TRAINING**

**FPT UNIVERSITY**

Capstone Project Document

**Vietnamese Sign Language Recognition**

|  |  |
| --- | --- |
| **Group 05** | |
| **Group members** | Nguyễn Hữu Kỳ Long – Team leader – SE60984  Nguyễn Đình Tân – Team member – SE61115  Nguyễn Xuân Ý – Team member – SE60869  Lê Phương Bình – Team member – SE61049 |
| **Supervisor** | Mr. Đỗ Đức Minh Quân |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** | VSLR |

-Ho Chi Minh City, 24/05/2015-

*This page is intentionally left blank*

# Table of Contents

[Table of Contents 3](#_Toc424722492)

[List of Tables 3](#_Toc424722493)

[Definitions, Acronyms, and Abbreviations 4](#_Toc424722494)

[A. Report No.2 Software Project Management Plan 5](#_Toc424722495)

[1. Problem Definition 5](#_Toc424722496)

[1.1 Name of this Capstone Project 5](#_Toc424722498)

[1.2 Problem Abstract 5](#_Toc424722499)

[1.3 Project Overview 5](#_Toc424722500)

[2. Project organization 9](#_Toc424722501)

[2.1 Software Process Model 9](#_Toc424722502)

[2.2 Roles and responsibilities 9](#_Toc424722503)

[2.3 Tools and Techniques 11](#_Toc424722504)

[3. Project Management Plan 11](#_Toc424722505)

[2.4 Product Backlog 11](#_Toc424722506)

[2.5 Sprint Backlog 11](#_Toc424722507)

[2.6 All Meeting Minutes 11](#_Toc424722508)

[3. Coding Convention 11](#_Toc424722509)

# List of Tables

[Table 1: Hardware Requirement for Server 7](#_Toc417269080)

[Table 2: Hardware Requirement for Mobile 7](#_Toc417269081)

[Table 3: Roles and Responsibilities Details 8](#_Toc417269082)

[Table 4: Product Backlog Detail 11](#_Toc417269083)

# Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| **Name** | **Definition** |
| VSLR | Vietnamese Sign Language Recognition |
| LCD | Liquid crystal display |
|  |  |
|  |  |

# Report No.2 Software Project Management Plan

# Problem Definition



### Name of this Capstone Project

* Vietnamese Sign Language Recognition

### Problem Abstract

As we know, in the daily life, there is a lot of ways people can understand others such as speech, expression of act, gesture or feelings, etc. However, it is better to express oneself in speech. At the same time, it is an actual matter to mute people to get other people and in the opposite way. The current solution for them is sign language but that means it requires everyone to know sign language of mute people or need someone play as a translator. But these solutions just solve the problem at that time, these are not a long-term strategy. It expects a long time and high cost for preparation from them to solve the problem. In additional, there still are some temporary solutions such as handwriting or using familiar signs, but these way will not produce the desired effect and requires lots of time or effort.

To solve those problems mentioned above, we propose a solution which can help dump person to express themselves in speech or text. That is a device playing a translator and act as intermediary role.

### Project Overview

#### Current Situation and Disadvantages

Below are some current behaviors of user:

* Handwriting:
* People will use something can write on as vehicle for communication.
* They can write out exactly what they want to say to the recipient.
* The recipient can receive and read the content immediately.
* Familiar signs:
* Speakers will describe the word which they want say through action, describe the shape, body language.
* Listeners observe the speaker's actions. They predict information that the speaker shown.
* Interpreters:
* Act as intermediary to translate the content of communication.
* Speakers express words by their language, the interpreter receive information from the speaker and then convey that information by the language of the listener.
* Degree of accuracy of translated content is quite high for both two sides.

Below are the disadvantages of current situation:

* Hand-writing :
* Users must use an intermediary for communication such as paper, pens. However, these things are not always available.
* Users spend more time to write out all their wishes and read them.
* User can meet difficulties about different languages.
* The error can be caused by user handwriting.
* Using familiar signs :
* Maybe be misleading because the symbols are not standardized.
* It is trending towards personally identifiable user.
* It is difficult to show all wishes of communicator.
* Time consuming for understanding the content is long.
* Translator :
* Hiring a translator must be costly.
* Translator who work only in the fixed time, thus not always can meet user's demands.
* Translator must be a experienced person.
* Number of translator is limited.

Analyzing image is the most common way to solve many problems in the real life. One of those problems is recognition. Today, with growth of support analyzing image library and algorithms provided to process image is widespread, tracking and recognition can be performed more easily. Our project is taking into consideration about it to recognize hand signs to help people can communicate with another people.

* Advantages:
* Can be implemented on many different platforms.
* Operating costs less expensive.
* Implemented quickly by many image processing algorithm diversity.
* Disadvantages:
* Analyzing image still remains restriction on process environment, point of view.
* Recognition have still not covered every cases yet. Within weird characterizes, the result maybe not high accurate.
* Currently, analyzing image and recognition just detect and recognize hand signs without motion.
* To get high degree of accuracy, it requires some accessories from users.

#### The Proposed System

Exploiting the development of embedded technology and the growing of image processing, we put forward a system which can recognize hand sign language to help dumb people can communicate. This system includes a camera which captures hand signs from user, a raspberry board plays role as central processing unit which analyzes these captures, processes some algorithms to recognize them and performs some different functions in the system, and a LCD which shows interfaces of the system and recognition result. Besides that, the system still provides some electronic devices to user can control battery, or devices.

##### Controlling System

* Users can turn on/off the system by a switch button.
* Users can monitor the battery capacity.
* Users use hand gestures to select the functions and move between functions.

##### Hand Sign Language Recognize

* Users express hand gestures which describes the desired content, then they can receive the hand sign recognition result.
* Users can see your hand gestures on LCD.
* Users can check the result of the current hand sign.
* Users can edit the current translated content.
* Users receive the recognition result via text or sound shown from LCD.

##### Learning Hand Sign

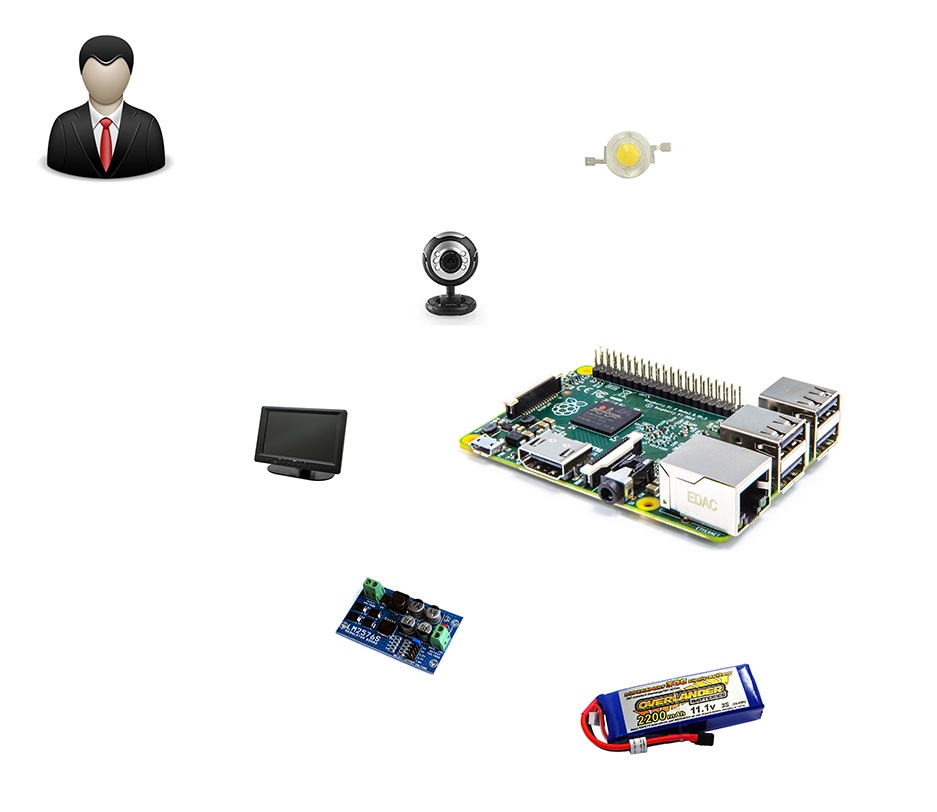
* Users can choose words that they want to learn which existed in the system.
* Users can see images which express the hand gesture.
* User's hand signs can be practiced and checked by following some steps of the system.
* Users receive the current recognized result of the hand sign via text or sound.

#### Boundaries of the System

##### The restrictions

* The system language is Vietnamese.
* Hand sign language the system supports is Vietnamese sign language.
* The system requires users must use supported accessories.
* The system requires users must provide a stable environment in room with sufficient light and a background is not complex on color, especially, no color close to skin color.
* The system must be fixed during the working process.

##### The components of the system:



**Figure 1: Boundaries of the System**

#### Development Environment

##### Hardware requirements

* 4 laptops is used for development the system. These are setup Ubuntu 14.04 operating system.
* Raspberry Pi B2 is used to process as central processing unit.
* Cable is connection between laptop and raspberry pi 2.
* Keyboard, mouse, and usb wifi are used to setup operating system and necessary environments for raspberry pi 2.
* Backup flash memory: a backup solution when problems with operating system. This memory must be setup similiar to main flash memory.
* LIPO battery (12V – 3A): power for the system can works.
* Camera module of raspberry kit: is used to capture images.
* LCD 7 inch is used to show the inteface of functions and the recognized results.
* 2 Led (1W): is used to balance light.

##### Software requirements

* Linux: operating system and platform for deploy
* Remote Desktop: application for remoting to work on raspberry
* QT Creator: is to develop c++ application and Linux GUI
* OpenCV 2.4.9 library: supporting image processing.
* SQLite: software creates and manages the system database.
* Software Ideas Modeler: application for creating models and diagrams.
* Microsoft Office 2010: is used to write documents and assign tasks.
* Githup and TortoiseSVN and Rabbit VCS: used for source control
* Skype: used for communication and meeting

## Project organization

### Software Process Model

*Mô tả sơ*

*Hình mô hình*

*Lí do chọn*

Project is developed under scrum model. We choose this model because the scope of the project is not fixed when the requirement changes day by day. Products are created quickly. Therefore ,the development team can easy to change if the wrong direction. Degree of cooperation between the members is set to high.

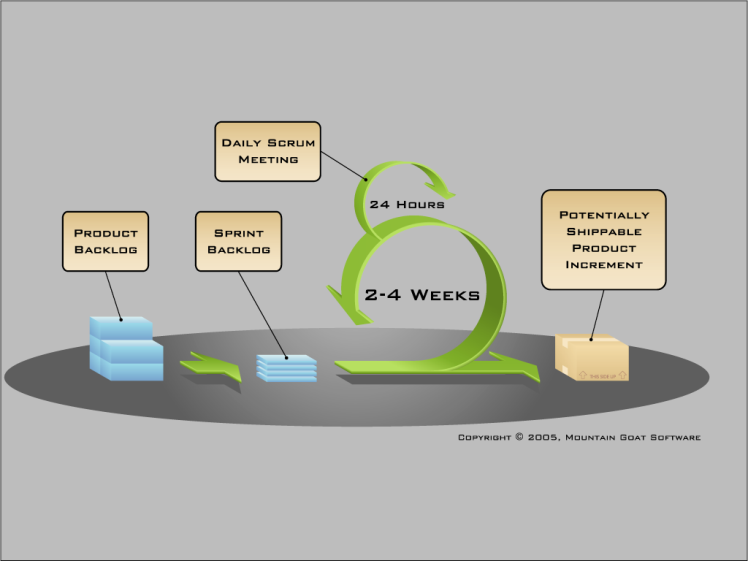


Figure 1: Scrum Development Model

For more information: <http://www.mountaingoatsoftware.com/agile/scrum>

### Roles and responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full name** | **Role in Group** | **Responsibilities** |
| **1** | Đỗ Đức Minh Quân | Scrum Master/Product Owner | * Defining user requirements * Specifying business * Control the development process * Give advices on techniques, solutions and business analysis support |
| **2** | Nguyễn Hữu Kỳ Long | Team Leader, BA, DEV, Tester | * Managing process * Clarifying requirements * Researching solutions and techniques * Assigning task for members * Reviewing the result of task of members. * Editing documents and reports * Reviewing documents and reports * Developing the system software * Reviewing the system hardware * Coding * Creating test plan. * Testing |
| **3** | Nguyễn Đình Tân | Team Member, BA, DEV, Tester | * Clarifying requirements * Researching solutions and techniques * Designing database * Preparing documents and reports * Reviewing documents and reports * Developing the system software * Reviewing the system hardware * Coding * Testing |
| **4** | Lê Phương Bình | Team Member, BA, DEV, Tester | * Clarifying requirements * Preparing documents and reports * Reviewing documents and reports * Developing the system hardware * Reviewing the system software * Coding * Testing |
| **5** | Nguyễn Xuân Ý | Team Member, BA, DEV, Tester | * Clarifying requirements * Editing documents and reports * Reviewing documents and reports * Developing the system hardware * Coding * Testing |

Table 3: Roles and Responsibilities Details

### Tools and Techniques

- Front-end and back-end IDE: QT 5.4 Creator

- Front-end technology: QT Linux GUI

- Back-end library:

* OPENCV 2.4.9 library
* LIBSVM library

- Managing database: SQLite 3

- Connecting to Raspberry PI 2: Remote Desktop Connection of Ubuntu 14.04

- Managing the project:

* SVNtortoise
* Rabbit VCS

- Managing documents, reports, models and diagrams:

* Software Ideas Modeler
* Microsoft Office 2010

## Project Management Plan

### Product Backlog

### Sprint Backlog

### All Meeting Minutes

## Coding Convention

*General view of C++ Programming Style put into practice in the project*

* Naming Conventions
* Variable names must be in mixed case starting with lower case.
* Named constants must be all uppercase using underscore to separate words.
* Names representing methods or functions must be verbs and written in mixed case starting with lower case.
* Plural form should be used on names representing a collection of objects
* The prefix is should be used for boolean variables and methods
* Include Files and Include Statements
* Header files must contain an include guard
* Include statements should be sorted and grouped
* Include statements must be located at the top of a file only
* Variables
* Class variables should never be declared public
* C++ pointers and references should have their reference symbol next to the type rather than to the name
* Conditionals
* Complex conditional expressions must be avoided
* The conditional should be put on a separate line
* Executable statements in conditionals must be avoided
* Comments
* Use // for all comments, including multi-line comments
* Comments should be included relative to their position in the code
* Class and method header comments should follow the JavaDoc conventions

*References*

C++ Programming Style Guidelines, Version 4.9, January 2011, Geotechnical Software Services, Copyright © 1996 – 2011

<http://geosoft.no/development/cppstyle.html>