**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](#_Toc417273911)

[List of Tables 3](#_Toc417273912)

[Definitions, Acronyms, and Abbreviations 4](#_Toc417273913)

[A. Report No.2 Software Project Management Plan 5](#_Toc417273914)

[1. Problem Definition 5](#_Toc417273915)

[1.1 Name of this Capstone Project 5](#_Toc417273917)

[1.2 Problem Abstract 5](#_Toc417273918)

[1.3 Project Overview 5](#_Toc417273919)

[2. Project organization 7](#_Toc417273920)

[2.1 Software Process Model 7](#_Toc417273921)

[2.2 Roles and responsibilities 8](#_Toc417273922)

[2.3 Tools and Techniques 8](#_Toc417273923)

[3. Project Management Plan 10](#_Toc417273924)

[3.1 Product Backlog 10](#_Toc417273925)

[3.2 Sprint Backlog 11](#_Toc417273926)

[3.3 All Meeting Minutes 11](#_Toc417273927)

[4. Coding Convention 11](#_Toc417273928)

# 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 |
|  |  |
|  |  |
|  |  |

# 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.

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

##### Advantages

* Can be applied in many fields.
* Convenient, social acceptability and inexpensive technique of identification.

##### Disadvantages

* The relative angle of the target’s hand influences the recognition score profoundly. When a hand is enrolled in the recognition software, usually multiple angles are used (profile, frontal and 45-degree are common). Anything less than a frontal view affects the algorithm’s capability to generate a template for the hand.
* Other conditions where hand recognition does not work well include poor lighting, handicap, injured, or so much color background.

#### The Proposed System

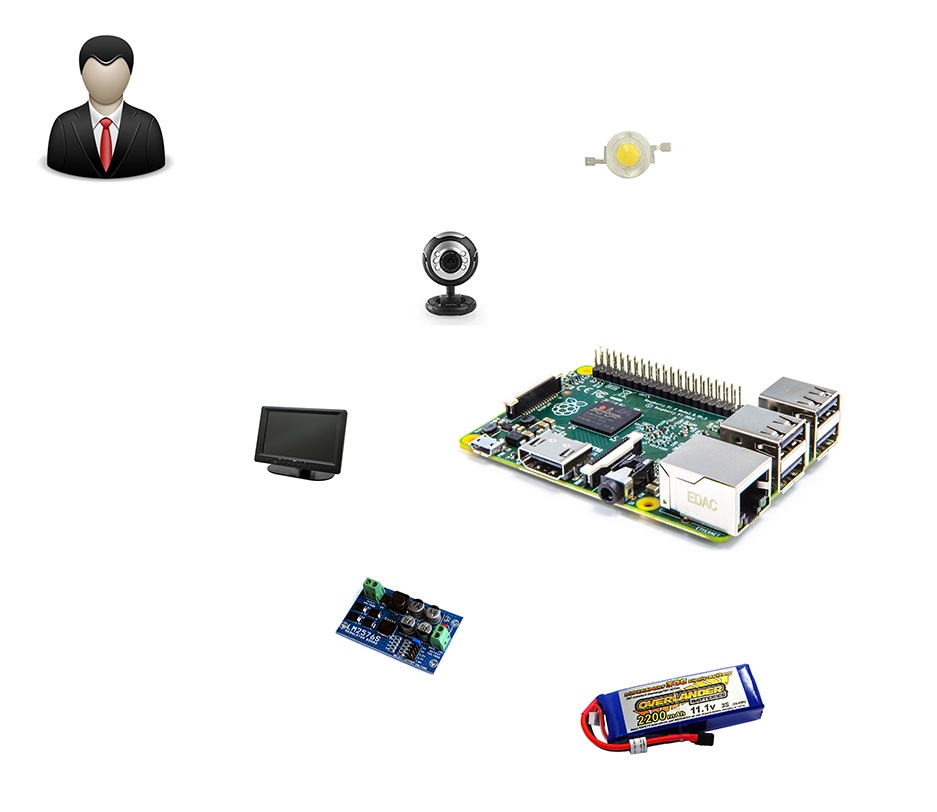
Exploiting the development of embedded technology and the growing of image analysis, we put forward a system to solve the problem. This system includes a camera capturing hand signs and a board to analyze these captures and process some different functions.

##### Hand Tracking

##### Hand Features Taking

##### SVM

#### Boundaries of the System



**Figure 1: Boundaries of the System**

#### Development Environment

##### Hardware requirements

* Raspberry Pi B2
* Camera module of raspberry kit
* LCD 7 inch
* LM2576ADJ-Board
* Lipo Battery
* Led 1W

##### Software requirements

* Linux: operating system and platform for deploy
* Gnu compiler collection: g++
* Remote Desktop: application for remoting to work on raspberry
* QT Creator: is to develop c++ application and Linux GUI
* Githup and TortoiseSVN and Rabbit VCS: used for source control
* Skype: used for communication and meeting

## Project organization

### Software Process Model

Project is developed under scrum model. We choose this model because the unpredictable properties of the project, when the requirement changes day by day and technology need to be research to adapt the scope of the project.

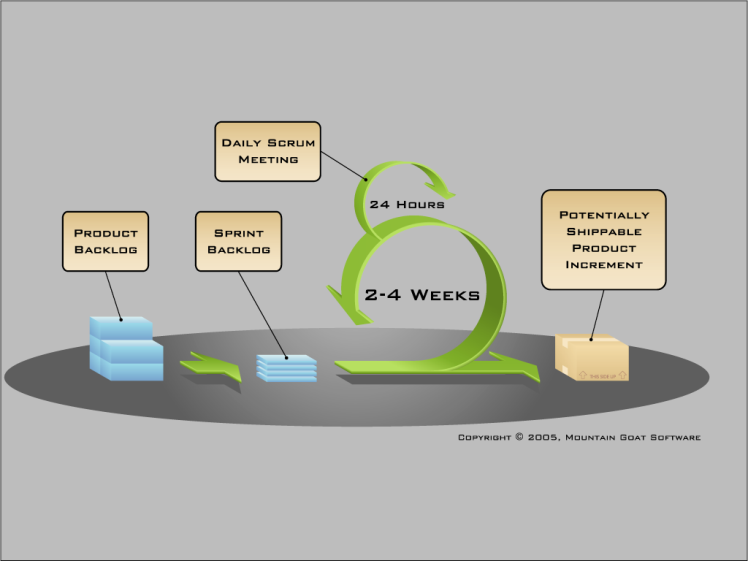


Figure 1: Scrum Development Model

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 | * Specify user requirement * Control the development process * Give out technique and business analysis support |
| **2** | Nguyễn Hữu Kỳ Long | Team Leader, BA, DEV, Tester | * Managing process * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **3** | Nguyễn Xuân Ý | Team Member, BA, DEV, Tester | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| **4** | Lê Phương Bình | 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 technology: Linux GUI.

- Back-end technologies:

* C++ application
* OPENCV library
* LIBSVM library

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

## Project Management Plan

### Product Backlog

### Sprint Backlog

### All Meeting Minutes

## Coding Convention

Use GCC C++ coding convention to develop website, web service and mobile app.

Summary:

* Naming Convention.
  + Use camel case for both variable and function name.
  + Use Pascal case for class name.
* Indentation.
  + Avoid lines longer than 80 characters, since they are 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 https://gcc.gnu.org/wiki/CppConventions