**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, 17/05/2015-

*This page is intentionally left blank*

# Table of Contents

[Table of Contents 3](#_Toc425170720)

[List of Tables 5](#_Toc425170721)

[List of Figure 5](#_Toc425170722)

[Definitions, Acronyms, and Abbreviations 7](#_Toc425170723)

[A. Report No. 1 Introduction 8](#_Toc425170724)

[1. Project Information 8](#_Toc425170725)

[2. Introduction 8](#_Toc425170726)

[3. Current Situation 8](#_Toc425170727)

[4. Problem Definition 8](#_Toc425170728)

[5. Proposed Solution 8](#_Toc425170729)

[5.1 Feature functions 9](#_Toc425170730)

[5.2 Advantages and disadvantages 9](#_Toc425170731)

[6. Functional Requirements 9](#_Toc425170732)

[6.1 Tracking hand 9](#_Toc425170733)

[6.2 Hand recognition 9](#_Toc425170734)

[6.3 Showing the content 9](#_Toc425170735)

[6.4 Learning hand sign 9](#_Toc425170736)

[6.5 Controlling System 9](#_Toc425170737)

[6.6 Controlling power 9](#_Toc425170738)

[7. Role and Responsibility 10](#_Toc425170739)

[B. Report No.2 Software Project Management Plan 11](#_Toc425170740)

[1. Problem Definition 11](#_Toc425170741)

[1.1 Name of this Capstone Project 11](#_Toc425170743)

[1.2 Problem Abstract 11](#_Toc425170744)

[1.3 Project Overview 11](#_Toc425170745)

[Figure 1: Components of the the system 14](#_Toc425170746)

[2. Project organization 15](#_Toc425170747)

[2.1 Software Process Model 15](#_Toc425170748)

[Figure 2 : Scrum Development Model 16](#_Toc425170749)

[2.2 Roles and responsibilities 16](#_Toc425170750)

[2.3 Tools and Techniques 18](#_Toc425170751)

[3. Project Management Plan 19](#_Toc425170752)

[3.1 Product Backlog 19](#_Toc425170754)

[3.2 Sprint Backlog 20](#_Toc425170755)

[3.3 Sprint Burndown Chart 24](#_Toc425170761)

[Figure 3: Chart of Sprint Backlog 24](#_Toc425170762)

[3.4 All Meeting Minutes 25](#_Toc425170763)

[4. Coding Convention 34](#_Toc425170764)

[C. Report No. 3 Software Requirement Specification 35](#_Toc425170765)

[1. User Requirement Specification 35](#_Toc425170766)

[2. System Requirement Specification 35](#_Toc425170767)

[2.1 External Interface Requirement 35](#_Toc425170768)

[2.2 System Overview Use Case 36](#_Toc425170769)

[Figure 4: System Overview Use Case 36](#_Toc425170770)

[2.3 List of Use Case 37](#_Toc425170775)

[Figure 5: Select Function use case diagram 37](#_Toc425170776)

[Figure 6: Recognize Hand Sign Language use case diagram 39](#_Toc425170777)

[Figure 7: Learn sign use case diagram 41](#_Toc425170778)

[Figure 8: Learn sign use case diagram 43](#_Toc425170779)

[3. System Attribute 49](#_Toc425170780)

[3.1 Usability 49](#_Toc425170781)

[3.2 Reliability 49](#_Toc425170782)

[3.3 Availability 49](#_Toc425170783)

[3.4 Security 49](#_Toc425170784)

[3.5 Maintainability 49](#_Toc425170785)

[3.6 Portability 49](#_Toc425170786)

[3.7 Performance 49](#_Toc425170787)

[D. Report No.5 System Implementation & Test 50](#_Toc425170788)

[1. Introduction 50](#_Toc425170789)

[1.1 Overview 50](#_Toc425170790)

[1.2 Test Approach 50](#_Toc425170791)

[2. Database Relationship Diagram 50](#_Toc425170792)

[2.1 Physical Diagram 50](#_Toc425170793)

[Figure 52: Physical Database Diagram 50](#_Toc425170794)

[2.2 Data Dictionary 51](#_Toc425170795)

[3. Test Plan 51](#_Toc425170796)

[3.1 Test items 51](#_Toc425170797)

[3.2 Features to be tested 51](#_Toc425170798)

[3.3 Features not to be tested 51](#_Toc425170799)

[- N/A 51](#_Toc425170800)

[3.4 Environmental needs 51](#_Toc425170801)

[3.3 Test case pass/fail criteria 51](#_Toc425170802)

[4. Integration Test Specifications 52](#_Toc425170803)

[4.1 “Background Color Subtraction” Test 52](#_Toc425170804)

[Figure 52: Components of the Background Color Subtraction 52](#_Toc425170805)

[4.2 “Selecting Function” Test 57](#_Toc425170806)

[Figure 52: Components of the Selecting Function 57](#_Toc425170807)

[4.3 “Hand Sign Language Recognition” Test 62](#_Toc425170808)

[Figure 52: Components of the Hand Sign Language Recognition 62](#_Toc425170809)

[4.4 “Learning Hand Sign Language” Test 67](#_Toc425170810)

[Figure 52: Components of the Selecting Function 67](#_Toc425170811)

[4.5 “Monitor Battery Capacity” Test 75](#_Toc425170812)

[Figure 52: Components of the ??? 75](#_Toc425170813)

[4.6 “Charging Battery” Test 78](#_Toc425170814)

[Figure 52: Components of the ??? 78](#_Toc425170815)

# List of Tables

[Table 1: Roles and Responsibilities 10](#_Toc425258027)

[Table 2: Roles and Responsibilities Details 17](#_Toc425258028)

[Table 3: Product Backlog Details 19](#_Toc425258029)

[Table 4: Sprint Backlog Details 23](#_Toc425258030)

[Table 5: Definitions, Acronyms, and Abbreviations 25](#_Toc425258031)

[Table 6: Scrum Meeting Minutes Detail 33](#_Toc425258032)

# List of Figure

[Figure 1: Components of the the system 14](#_Toc425258033)

[Figure 2 : Scrum Development Model 16](#_Toc425258034)

[Figure 3: Chart of Sprint Backlog 24](#_Toc425258035)

[Figure 4: System Overview Use Case 36](#_Toc425258036)

[Figure 5: Select Function use case diagram 37](#_Toc425258037)

[Figure 6: Recognize Hand Sign Language use case diagram 39](#_Toc425258038)

[Figure 7: Learn sign use case diagram 41](#_Toc425258039)

[Figure 8: Learn sign use case diagram 43](#_Toc425258040)

[Figure 9: Monitor Battery Capacity use case diagram 45](#_Toc425258041)

[Figure 10: Notify Battery Capacity use case diagram 47](#_Toc425258042)

[Figure 52: Physical Database Diagram 50](#_Toc425258043)

[Figure 52: Components of the Background Color Subtraction 52](#_Toc425258044)

[Figure 52: Components of the Selecting Function 57](#_Toc425258045)

[Figure 52: Components of the Hand Sign Language Recognition 62](#_Toc425258046)

[Figure 52: Components of the Learning Hand Sign Language 67](#_Toc425258047)

[Figure 52: Components of the Monitoring Battery Capacity 75](#_Toc425258048)

[Figure 52: Components of the charging battery 78](#_Toc425258049)

# Definitions, Acronyms, and Abbreviations

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

# Report No. 1 Introduction

## Project Information

* Project name: **Vietnamese Sign Language Recognition**
* Project Code: **VSLR**
* Product Type: **Embedded system**
* Start Date: **May 11th, 2015**
* End Date:

## Introduction

Nowadays, the communication is the way people can understand each other, is the way people can express their ideas, their thoughts to others. As we know, speaking is the most common way to communicate in life. However, to dumb person, they still need to communicate with others so they have a different way to expose themselves, it is called hand sign language or dumb language.

In this project, we want to develop a device that can help dumb person communicate with not only another mute but also everyone. The device can capture hand signs and then recognize them into text or sound with the same meaning.

## Current Situation

When you want to talk to a dumb person or when a mute wants to present his / her ideas, presentations in a meeting but you are not able to get their signs. Furthermore, when two dumb persons talk to each other but they are from different countries, they have distinct hand sign language, which way can they understand each other? Obviously, there are some ways, they can write out what they want or they can use some signs that are familiar to the daily life, and they can even hire a translator to interpret.

## Problem Definition

*The following disadvantages of current situation:*

* Handwritten: Time consuming to write out all content is very high.
* Using familiar signs: Without time consuming, the accuracy of the content is not high.
* Hand sign language translator can not respond the instant needs of communication. Moreover, the price for hiring a translator is very costly.

## Proposed Solution

To meet the needs of users we offer a solution based on translating hand signs into content and then show them.

Our system is a small device with a camera to capture hand signs and then translate them.

*In more detail, our system has the following functions:*

### **Feature functions**

* The system detects your hands, keeps track them and then analyzes the captured images into content.
* Showing the translated content for users on text and sound.
* Learning sign language hand for people who want to know about the language in order to better communicate with dumb people.

### Advantages and disadvantages

*The advantages and disadvantages of the proposed solution:*

* Advantages:
  + Quick and easy communicate for dumb person.
  + Train for person who don’t know about mute language.
  + Standardized for hand sign language.
  + People get used to the dumb language easily.
* Disadvantages:
  + In some cases, this solution does not work really exactly with the hands have weird characterize.
  + This solution needs stable environment (light, background) and some accessories.
  + This solution can not solve the problem about hand motion language.

## Functional Requirements

*Function requirements of the system are listed as below:*

### Tracking hand

* Allow users can move the hand in range area but the system still works correctly.

### Hand recognition

* The system analyzes the images which is captured by camera, then detects and recognizes the hand sign on these images into content.

### Showing the content

* The translated content is shown not only on text but also on sound.

### Learning hand sign

* Users select and learn words existed in the system with images express the hand gesture.

### Controlling System

* Allow users can turn on / off the system by the power button.
* Users can select functions by hand signs.
* Users can perform operations of function by hand signs.

### Controlling power

* System uses battery power gives users more flexibility in using.
* Combining with controlling the battery capacity that helps users to use the most effective.

## Role and Responsibility

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Full Name | Role | Position | Contact |
| 1 | Đỗ Đức Minh Quân | Scrum Master/Product Owner | Instructor | [m](mailto:Khanhkt@fpt.edu.vn)inhquandd@fpt.edu.vn |
| 2 | Nguyễn Hữu Kỳ Long | Developer | Leader | [longnhkse60984@fpt.edu.vn](mailto:longnhkse60984@fpt.edu.vn) |
| 3 | Nguyễn Đình Tân | Developer | Member | [tanndse61115@fpt.edu.vn](mailto:tanndse61115@fpt.edu.vn) |
| 4 | Nguyễn Xuân Ý | Developer | Member | [ynxse60896@fpt.edu.vn](mailto:ynxse60896@fpt.edu.vn) |
| 5 | Lê Phương Bình | Developer | Member | [binhlpse61049@fpt.edu.vn](mailto:binhlpse61049@fpt.edu.vn) |

Table : Roles and Responsibilities

# 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. Nevertheless, these solutions just solve the problem at that time and 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 ways 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 an 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 supported 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:
* The system can be implemented on many different platforms.
* Operating costs are less expensive.
* Recognition is implemented quickly by many image-processing algorithms.
* Disadvantages:
* Analyzing image still remains restriction on process environment, point of view.
* Recognition has still not covered every case 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 just recognizes no motion hand signs.
* 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: Components of the 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 similar 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 interface of functions and the recognized results.
    - 2 Led (1W): is used to balance light.
    - LM2576ADJ-Board: UNI Regulator Board.
    - LT084 + zener 5.1v is used to monitor battery capacity.
    - XL6009 DC-DC Voltage Boost Module is used for increasing voltage.

##### Software requirements

* + - Operating system and platform for deployment and development: Ubuntu 14.04 for laptop and Raspbian for Raspberry PI.
    - Remote Desktop: application for remoting to work on raspberry.
    - QT 5.4 Creator: is to develop C++ application and Linux GUI.
    - OpenCV 2.4.9 library: supporting image processing.
    - SQLite 3: 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

#### Overall Description

Scrum is an agile methodology that can be applied to nearly any project; however, the Scrum methodology is most commonly used in software development. The Scrum process is suited for projects with rapidly changing or highly emergent requirements. Scrum software development progresses via a series of iterations called sprints, which last from one to four weeks. In the agile Scrum world, a sprint planning meeting is described in terms of the desired outcome (a commitment to a set of features to be developed in the next sprint) instead of a set of Entry criteria, Task definitions, Validation criteria, Exit criteria. The Scrum model suggests each sprint begins with a brief planning meeting and concludes with a review. These are the basics of Scrum project management.

#### Scrum Development Model



**Figure 2 : Scrum Development Model**

#### Reasons for Choosing

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.

### 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 2: Roles and Responsibilities Details

### 

### Tools and Techniques

* + - Front-end and back-end IDE:
* QT 5.4 Creator
  + - Front-end technology:
* QT 5.4 Linux GUI
  + - Back-end library:
* OPENCV 2.4.9 library
* LIBSVM 3.20 library
* Espeak 1.48.04 library
  + - Managing database:
* SQLite 3
  + - Connecting to Raspberry PI 2:
* Remote Desktop Connection Program of Ubuntu 14.04
  + - Managing the project:
* SVNtortoise version 1.8.11
* Rabbit VCS
  + - Managing documents, reports, models and diagrams:
* Software Ideas Modeler version 7.70.5385.38708
* Microsoft Office 2010

## 

## Project Management Plan



### Product Backlog

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Theme** | **User Type** | **Wants to...** | **So that...** | **Priority** | **Sprint** |
| 1 | Detection | User | keep track their hand gesture | can see his/her hand in the screen | Very High | 1 |
| 2 | Device | User | the system is a portable system | move the device easily and use it more flexibly | High | 1 |
| 3 | Recognition | User | recognize the hand signs | express the same meaning to the partner can understand | Very High | 2 |
| 4 | Recognition | User | receive the recognition result via text and sound | express the translated content in a clearly way | High | 2 |
| 5 | Detection | User | control the system functions by hand gesture | perform and move beetween the system functions | High | 3 |
| 6 | Power | User | know remaining of battery capacity | can monitor the use of device | Medium | 3 |
| 7 | Recognition | User | increase the accuracy of the recognition result | raise the reliability of the translated content | Very High | 4 |
| 8 | Learning | User | learn the hand sign language | learn new signs or pratice his/her signs | Medium | 4 |
| 9 | Device | User | turn on/off the system | can turn on/off the device according to the demand | Medium | 4 |

Table 3: Product Backlog Details

### Sprint Backlog

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **User Story** | **Time (est)** | **Time (spent)** | **Time (left)** | **Task** | **Time (est)** | **Who's working** | **Status** | **Work Done per week per task [Days]** | | | | | | | | | | | |
|  | *Project Name: "Vietnamese Language Sign Recognition" Started: 12-05-2015* | | | | | | | | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** |
| 1 | As a user, I want to keep track my hand gesture | 15 | 15 | 0 | Setup development environment for Raspberry PI and laptops | **3** | All Team | Done | 3 |  |  |  |  |  |  |  |  |  |  |  |
| Capturing images from camera and showing them to user | **4** | TanND | Done | 1 | 3 |  |  |  |  |  |  |  |  |  |  |
| Extracting background color | **4** | LongNHK | Done | 1 | 3 |  |  |  |  |  |  |  |  |  |  |
| Extracting hand color | **4** | LongNHK | Done |  | 2 | 2 |  |  |  |  |  |  |  |  |  |
| Subtracting color to get hand binary image | **4** | LongNHK | Done |  |  |  | 1 |  |  |  |  |  |  |  |  |
| Finding the hand contours on the hand binary image | **4** | TanND | Done |  | 2 | 2 |  |  |  |  |  |  |  |  |  |
| Designing QT Linux GUI | **4** | TanND | Done |  |  | 3 | 1 |  |  |  |  |  |  |  |  |
| As a user, I want the system is a portable system | 15 | 15 | 0 | Choose type of battery | **3** | YNX | Done | 1 | 2 |  |  |  |  |  |  |  |  |  |  |
| Choose voltage regulator circuit | **3** | BinhLP | Done | 1 | 2 |  |  |  |  |  |  |  |  |  |  |
| Connecting components | **5** | BinhLP + YNX | Done |  | 3 | 2 |  |  |  |  |  |  |  |  |  |
| Adjusting the device components | **4** | BinhLP + YNX | Done |  |  | 3 | 1 |  |  |  |  |  |  |  |  |
| 2 | As a user, I want to recognize the hand signs | 13 | 13 | 0 | Creating SQLite database | **2** | TanND | Done |  |  |  | 2 |  |  |  |  |  |  |  |  |
| Training SVM | **3** | LongNHK | Done |  |  |  | 3 |  |  |  |  |  |  |  |  |
| Defining features to recognize | **3** | LongNHK | Done |  |  |  | 1 | 2 |  |  |  |  |  |  |  |
| Features extraction | **4** | LongNHK | Done |  |  |  |  | 3 | 1 |  |  |  |  |  |  |
| Recognizing the hand sign by SVM | **3** | LongNHK | Done |  |  |  |  |  | 3 |  |  |  |  |  |  |
| Defining the meaning word of the SVM result from database | **3** | TanND + YNX | Done |  |  |  | 2 | 1 |  |  |  |  |  |  |  |
| Designing QT Linux GUI | **4** | BinhLP + YNX | Done |  |  |  | 4 |  |  |  |  |  |  |  |  |
| As a user, I want to receive the recognition result via text and sound | 10 | 10 | 0 | Choosing text to speech opensource | **3** | TanND + BinhLP | Done |  |  |  |  | 3 |  |  |  |  |  |  |  |
| Implementing text to speech opensource | **3** | TanND + BinhLP | Done |  |  |  |  | 1 | 2 |  |  |  |  |  |  |
| Changing pronunciation of word | **4** | TanND + YNX | Done |  |  |  |  |  | 3 | 1 |  |  |  |  |  |
| Showing the recognition result via text | **2** | LongNHK | Done |  |  |  |  |  | 1 | 1 |  |  |  |  |  |
| Showing the recognition result via sound | **4** | BinhLP + YNX | Done |  |  |  |  |  | 3 | 1 |  |  |  |  |  |
| 3 | As a user, I want to control the system functions by hand gesture | 15 | 15 | 0 | Defining operations of function | **3** | LongNHK + TanND | Done |  |  |  |  |  |  | 3 |  |  |  |  |  |
| Implementing operations of function | **3** | LongNHK + TanND | Done |  |  |  |  |  |  | 1 | 2 |  |  |  |  |
| Designing QT Linux GUI | **4** | LongNHK + TanND | Done |  |  |  |  |  |  |  | 3 | 1 |  |  |  |
| Synchronize between hand gestures and operations of function | **5** | LongNHK + TanND | Done |  |  |  |  |  |  |  |  | 4 | 1 |  |  |
| As a user, I want to know remaining of battery capacity | 15 | 15 | 0 | Choosing chip voltage comparator | **3** | BinhLP + YNX | Done |  |  |  |  |  |  | 3 |  |  |  |  |  |
| Choosing zener | **3** | BinhLP + YNX | Done |  |  |  |  |  |  | 1 | 2 |  |  |  |  |
| Constructing circuit | **4** | BinhLP + YNX | Done |  |  |  |  |  |  |  | 3 | 1 |  |  |  |
| Adjusting the device components | **2** | BinhLP + YNX | Done |  |  |  |  |  |  |  |  | 2 |  |  |  |
| Connecting to the system | **3** | BinhLP + YNX | Done |  |  |  |  |  |  |  |  | 2 | 1 |  |  |
| 4 | As a user, I want to increase the accuracy of the recognition result | 11 | 11 | 0 | Defining more features to recognize | **3** | All Team | Done |  |  |  |  |  |  |  |  |  | 3 |  |  |
| Extracting features | **3** | LongNHK | Done |  |  |  |  |  |  |  |  |  | 1 | 2 |  |
| Providing two LEDs to balance light | 3 | BinhLP + YNX | Done |  |  |  |  |  |  |  |  |  | 1 | 2 |  |
| Improving camera setting | 2 | TanND + YNX | Done |  |  |  |  |  |  |  |  |  | 1 | 1 |  |
| Improving background - hand color subtraction | 2 | LongNHK + TanND | Done |  |  |  |  |  |  |  |  |  |  | 3 |  |
| Training SVM | **3** | LongNHK | Done |  |  |  |  |  |  |  |  |  |  |  | 2 |
| As a user, I want to learn hand sign language | 8 | 8 | 0 | Creating database | **2** | TanND | Done |  |  |  |  |  |  |  |  |  |  | 1 |  |
| Managing database | **2** | TanND | Done |  |  |  |  |  |  |  |  |  |  |  | 2 |
| Implementing the practice function of learning | **4** | LongNHK + BinhLP | Done |  |  |  |  |  |  |  |  |  |  |  | 4 |
| Designing GUI | **4** | TanND + YNX | Done |  |  |  |  |  |  |  |  |  |  |  | 4 |
| As a user, I want to turn on/off the system | 8 | 8 | 0 | Choosing switch button | **2** | BinhLP | Done |  |  |  |  |  |  |  |  |  |  | 2 |  |
| Constructing circuit | **3** | BinhLP + YNX | Done |  |  |  |  |  |  |  |  |  |  |  | 3 |
| Connecting to the system | **3** | BinhLP | Done |  |  |  |  |  |  |  |  |  |  |  | 3 |
|  | TOTAL | 60 | 60 | 0 |  |  |  |  | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
|  | Daily burnout | 0 |  |  |  |  |  |  | **5** | **5** | **5** | **5** | **5** | **5** | **5** | **5** | **5** | **5** | **5** | **5** |
|  | **Total time left (from estimate)** |  |  |  |  |  | Estimate | 60 | 55 | 50 | 45 | 40 | 35 | 30 | 25 | 20 | 15 | 10 | 5 | 0 |
|  | **Total time left (from spent)** |  |  |  |  |  | Burnout | 60 | 55 | 50 | 45 | 40 | 35 | 30 | 25 | 20 | 15 | 10 | 5 | 0 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Table 4: Sprint Backlog Details



### Sprint Burndown Chart

**Figure 3: Chart of Sprint Backlog**

### All Meeting Minutes

|  |  |
| --- | --- |
| **Name** | **Definition** |
| x | Selected Person |
| VH | Very High Priority |
| H | High Priority |
| M | Medium Priority |
| A | Approved |
| Y | Yes |
| N | No |

Table 5: Definitions, Acronyms, and Abbreviations

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Scrum Meeting Minutes** | | | | | Target Personas | | | | Status | | Approval | |
| LongNHK | TanND | BinhLP | YNX | Degree of priority | Task completion | LongNHK | Mr. QuanDDM |
|
|
|
|
|
| **Date: 12/05/2015** | | | | |  | | | |  | |  | |
| *Raspberry PI 2* | | | | |  | | | |  | |  |  |
| Development environment for Raspberry | | | | | x | x | x | x | H | Y | A | A |
| *Report 1* | | | | |  | | | |  | | | |
| Orienting the way writing report 1 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 16/05/2015** | | | | |  | | | |  | |  | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Extracting background color | | | | | x |  |  |  | VH | Y | A | A |
| Capturing images from camera and showing them to user | | | | |  | x |  |  | VH | Y | A | A |
| *Portable System* | | | | |  | | | |  | |  |  |
| Choosing type of battery | | | | |  |  |  | x | H | Y | A | A |
| Choosing voltage regulator circuit | | | | |  |  | x |  | H | Y | A | A |
| *Report 1* | | | | |  | | | |  | | | |
| Reviewing report 1 | | | | | x | x | x | x | H | Y | A | A |
| **Date: 19/05/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Demo extracting background color | | | | | x |  |  |  | H | Y | A | A |
| Demo capturing images from camera and showing them to user | | | | |  | x |  |  | H | Y | A | A |
| *Portable System* | | | | |  | | | |  | |  |  |
| Reviewing battery | | | | | x | x | x | x | M | Y | A | A |
| Reviewing voltage regulator circuit | | |  |  | x | x | x | x | M | Y | A | A |
| *Report 2* | | | | |  | | | |  | | | |
| Orienting the way writing report 2 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 23/05/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Extracting hand color | | | | | x |  |  |  | VH | Y | A | A |
| Finding the hand contours on the hand binary image | | | | |  | x |  |  | VH | Y | A | A |
| *Portable System* | | | | |  | | | |  | |  |  |
| Connecting components | | | | |  |  | x | x | H | Y | A | A |
| *Report 2* | | | | |  | | | |  | | | |
| Reviewing report 2 | | | | | x | x | x | x | H | Y | A | A |
| **Date: 26/05/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Extracting hand color | | | | | x |  |  |  | H | Y | A | A |
| Designing QT Linux GUI which of subtracting color | | | | |  | x |  |  | H | Y | A | A |
| *Portable System* | | | | |  | | | |  | |  |  |
| Demo fully made portable system | | |  |  |  |  | x | x | H | Y | A | A |
| *Report 3* | | | | |  | | | |  | | | |
| Orienting the way writing report 3 | | | | | X | x | x | x | VH | Y | A | A |
| **Date: 30/05/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Subtracting color to get hand binary image | | | | | x |  |  |  | VH | Y | A | A |
| Finding the hand contours on the hand binary image | | | | |  | x |  |  | VH | Y | A | A |
| *Portable System* | | | | |  | | | |  | |  |  |
| Demo adjusted portable system | | | | |  |  | x | x | M | Y | A | A |
| *Report 3* | | | | |  | | | |  | | | |
| Orienting the way writing report 3 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 02/06/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Training SVM | | | | | x |  |  |  | VH | Y | A | A |
| Creating SQLite database | | |  |  |  | x |  |  | VH | Y | A | A |
| Designing QT Linux GUI which of steps recognition | | | | |  |  | x | x | M | Y | A | A |
| *Report 3* | | | | |  | | | |  | | | |
| Reviewing report 3 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 06/06/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Defining features to recognize | | | | | x |  |  |  | VH | Y | A | A |
| Defining the meaning word of the SVM result from database | | | | |  | x |  |  | VH | Y | A | A |
| *Portable System* | | | | |  | | | |  | |  |  |
| Designing QT Linux GUI which of the recognition flow | | | | |  |  | x | x | M | Y | A | A |
| *Report 3* | | | | |  | | | |  | | | |
| Reviewing report 3 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 09/06/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Defining features to recognize | | | | | x |  |  |  | VH | Y | A | A |
| Defining the meaning word of the SVM result from database | | | | |  |  |  | x | M | Y | A | A |
| *Result via sound* | | | | |  | | | |  | |  |  |
| Choosing text to speech opensource | | | |  |  | x | x |  | H | Y | A | A |
| *Report 4* | | | | |  | | | |  | | | |
| Orienting the way writing report 4 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 13/06/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Features extraction | | | | | x |  |  |  | VH | Y | A | A |
| *Result via sound* | | | | |  | | | |  | |  |  |
| Changing pronunciation of word | | | | |  | x |  | x | H | Y | A | A |
| Implementing text to speech opensource | | | |  |  | x | x |  | M | Y | A | A |
| *Report 4* | | | | |  | | | |  | | | |
| Orienting the way writing report 4 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 16/06/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Recognizing the hand sign by SVM | | | | | x |  |  |  | VH | Y | A | A |
| *Result via sound* | | | | |  | | | |  | |  |  |
| Changing pronunciation of word | | | | |  | x |  | x | H | Y | A | A |
| Showing the recognition result via sound | | | |  |  |  | x |  | M | Y | A | A |
| *Report 4* | | | | |  | | | |  | | | |
| Reviewing report 4 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 20/06/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Showing the recognition result via text | | | | | x |  |  |  | H | Y | A | A |
| *Result via sound* | | | | |  | | | |  | |  |  |
| Changing pronunciation of word | | | | |  |  |  | x | H | Y | A | A |
| Showing the recognition result via sound | | | |  |  |  | x |  | M | Y | A | A |
| *Report 4* | | | | |  | | | |  | | | |
| Reviewing report 4 | | | | | x | x | x | x | H | Y | A | A |
| **Date: 30/06/2015** | | | | |  | | | |  | | | |
| *Controlling function* | | | | |  | | | |  | |  |  |
| Defining operations of function | | | | | x | x |  |  | VH | Y | A | A |
| *Monitor the battery capacity* | | | | |  | | | |  | |  |  |
| Choosing chip voltage comparator | | | | |  |  | x | x | H | Y | A | A |
| *Report 5* | | | | |  | | | |  | | | |
| Orienting the way writing report 5 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 04/07/2015** | | | | |  | | | |  | | | |
| *Controlling function* | | | | |  | | | |  | |  |  |
| Implementing operations of function | | | | | x | x |  |  | VH | Y | A | A |
| *Monitor the battery capacity* | | | | |  | | | |  | |  |  |
| Choosing zener | | | | |  |  | x | x | H | Y | A | A |
| *Report 5* | | | | |  | | | |  | | | |
| Orienting the way writing report 5 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 07/07/2015** | | | | |  | | | |  | | | |
| *Controlling function* | | | | |  | | | |  | |  |  |
| Demo implementing operations of function | | | | | x | x |  |  | H | Y | A | A |
| *Monitor the battery capacity* | | | | |  | | | |  | |  |  |
| Reviewing the chip voltage comparator and zener | | | | | x | x | x | x | VH | Y | A | A |
| *Report 5* | | | | |  | | | |  | | | |
| Orienting the way writing report 5 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 11/07/2015** | | | | |  | | | |  | | | |
| *Controlling function* | | | | |  | | | |  | |  |  |
| Designing QT Linux GUI which of the flow of operations | | | | | x | x |  |  | H | Y | A | A |
| *Monitor the battery capacity* | | | | |  | | | |  | |  |  |
| Constructing circuit | | | | |  |  | x | x | VH | Y | A | A |
| *Report 5* | | | | |  | | | |  | | | |
| Reviewing report 5 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 14/07/2015** | | | | |  | | | |  | | | |
| *Controlling function* | | | | |  | | | |  | |  |  |
| Synchronizing between hand gestures and operations of function | | | | | x | x |  |  | VH | Y | A | A |
| *Monitor the battery capacity* | | | | |  | | | |  | |  |  |
| Adjusting the device components | | | | |  |  | x | x | H | Y | A | A |
| *Report 5* | | | | |  | | | |  | | | |
| Reviewing report 5 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 18/07/2015** | | | | |  | | | |  | | | |
| *Controlling function* | | | | |  | | | |  | |  |  |
| Demo synchronizing between hand gestures and operations of function | | | | | x | x |  |  | M | Y | A | A |
| *Portable System* | | | | |  | | | |  | |  |  |
| Connecting to the system | | | | |  |  | x | x | VH | Y | A | A |
| *Report 5* | | | | |  | | | |  | | | |
| Reviewing report 5 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 21/07/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Defining more features to recognize | | | | | x | x | x | x | VH | Y | A | A |
| *Report 5* | | | | |  | | | |  | | | |
| Reviewing report 5 | | | | | x | x | x | x | H | Y | A | A |
| **Date: 25/07/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Extracting features | | | | | x |  |  |  | VH | Y | A | A |
| Improving camera setting | | |  |  |  | x |  | x | M | Y | A | A |
| *Stable Environment* | | | | |  | | | |  | |  |  |
| Providing two LEDs to balance light | | | | |  |  | x | x | H | Y | A | A |
| *Report 5* | | | | |  | | | |  | | | |
| Reviewing report 5 | | | | | x | x | x | x | H | Y | A | A |
| **Date: 28/07/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Demo extracting features | | | | | x |  |  |  | H | Y | A | A |
| Improving background - hand color subtraction | | | | | x | x |  |  | VH | Y | A | A |
| Demo improving camera setting | | | | |  | x |  | x | M | Y | A | A |
| *Stable Environment* | | | | |  | | | |  | |  |  |
| Demo balancing light | |  |  |  |  |  | x | x | M | Y | A | A |
| *Report 6* | | | | |  | | | |  | | | |
| Orienting the way writing report 6 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 01/08/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Improving background - hand color subtraction | | | | | x | x |  |  | H | Y | A | A |
| *Learning Function* | | | | |  | | | |  | |  |  |
| Creating database | |  |  |  |  | x |  |  | H | Y | A | A |
| Designing GUI | | | | |  | x |  | x | M | Y | A | A |
| Implementing the practice function of learning | | | | | x |  | x |  | M | Y | A | A |
| *Report 6* | | | | |  | | | |  | | | |
| Orienting the way writing report 6 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 04/08/2015** | | | | |  | | | |  | | | |
| *Hand Detection* | | | | |  | | | |  | |  |  |
| Training SVM | | | | | x |  |  |  | VH | Y | A | A |
| *Learning Function* | | | | |  | | | |  | |  |  |
| Managing database | |  |  |  |  | x |  |  | M | Y | A | A |
| Designing GUI | | | | |  | x |  | x | H | Y | A | A |
| *ON/OFF the system* | | | | |  | | | |  | |  |  |
| Choosing switch button | |  |  |  |  |  | x |  | H | Y | A | A |
| *Report 6* | | | | |  | | | |  | | | |
| Orienting the way writing report 6 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 08/08/2015** | | | | |  | | | |  | | | |
| *Learning Function* | | | | |  | | | |  | |  |  |
| Implementing the practice function of learning | | | | | x |  | x |  | VH | Y | A | A |
| Demo GUI |  |  |  |  |  | x |  | x | H | Y | A | A |
| *ON/OFF the system* | | | | |  | | | |  | |  |  |
| Constructing circuit | | | | |  |  | x | x | VH | Y | A | A |
| *Report 6* | | | | |  | | | |  | | | |
| Reviewing report 6 | | | | | x | x | x | x | VH | Y | A | A |
| **Date: 11/08/2015** | | | | |  | | | |  | | | |
| *Learning Function* | | | | |  | | | |  | |  |  |
| Demo learning function | | | | | x |  | x |  | H | Y | A | A |
| Demo GUI |  |  |  |  |  | x |  | x | M | Y | A | A |
| *ON/OFF the system* | | | | |  | | | |  | |  |  |
| Demo constructed circuit | | | | |  |  | x | x | H | Y | A | A |
| Connecting to the system | | |  |  |  |  | x | x | VH | Y | A | A |
| *Report 6* | | | | |  | | | |  | | | |
| Reviewing report 6 | | | | | x | x | x | x | VH | Y | A | A |

Table 6: Scrum Meeting Minutes Detail

## 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>

# Report No. 3 Software Requirement Specification

## User Requirement Specification

*The system is not only reserved for mute person but also everyone who wants to learn sign language. Therefore, we have determined the requirement from these users:*

* Recognize his or her hand signs to text and sound: users want devices that can recognize exactly their hand signs. Then, the device must show recognition results via text on screen and emit pronunciation of this word via speaker.
* Learn the way expressing hand signs: there still are a lot of hand signs that users do not know exactly, they want a device that can help them practice these signs. The system should have images which can describe clearly the way expressing hand sign for user can follow. In addition, the system should have practice function for user practice.
* Controlling the system by hand gesture: users want to perform the operations of the system through his or her hand gesture without electricity devices.
* The system is portable: Users can easily move the system. They expect the system can work at many places, and it still works during a power outage.
* System's power must be controlled: Users can know the remaining battery capacity to monitor the use of equipment. Moreover, they can charge the battery when the battery is low.
* System should be easy to use as the electricity systems people use in daily live: Users can turn on/off the system safely without prejudice to the durability of the equipment.

## System Requirement Specification

### External Interface Requirement

External interface is concerned with designing interactive products to support the way people communicate and interact in their everyday and working lives. The products must be usability means easy to learn, effective to use and provide an enjoyable experience.

#### User Interface

* The GUI should be simple, clear, intuitive, and reminiscent.
* The interface is accessible, easy to use, and efficient.
* The interface should meet some criteria such as direct manipulation, device actions, information processing approach, visual features, …
* Each screen has fully instructions of the function implementation. Besides that, it still provides error, success, or implementation notification.

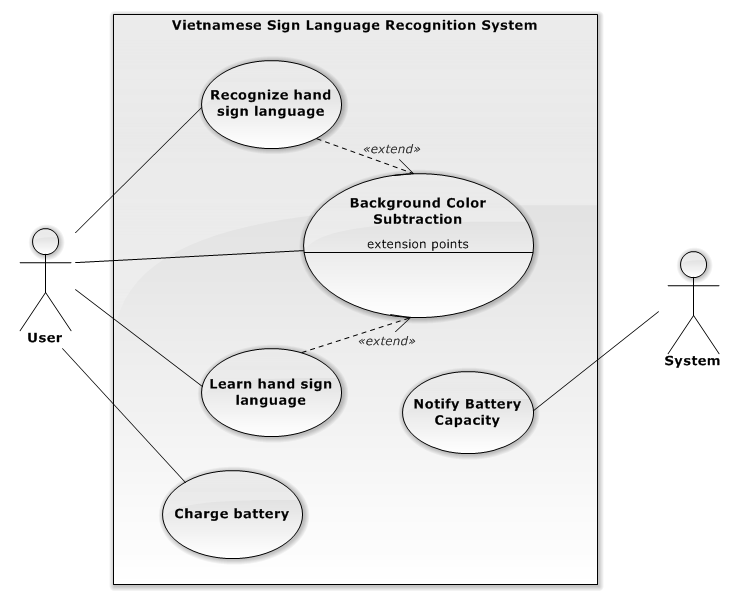
#### Hardware Interface

* The system must design hardware interface similar to the standard electricity system for anyone can use.
* Provide fully devices of a portable system.
* The system needs to be designed suitable for capturing the hands with an appropriate height, and a width for people can watch the LCD.
* The provided devices should be easy to replace.
* Electricity devices should be packaged in the safety way.

#### Software Interface

* Linux GUI for Raspbian Operating System.
* The interface must be responsive for LCD 7-inch with the resolution 1024 \* 600.

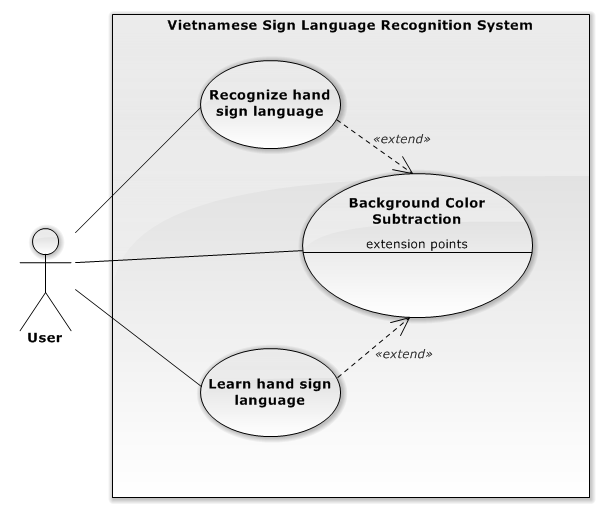
### System Overview Use Case



**Figure 4: System Overview Use Case**

### List of Use Case

#### Background Color Subtraction

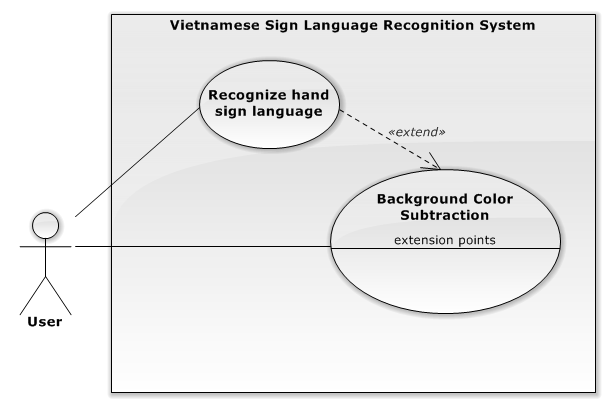


**Figure 6: Recognize Hand Sign Language use case diagram**

**Use Case Specification**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **USE CASE - 1 SPECIFICATION** | | | | | |
| **Use-case No.** | VSLR001 | **Use-case Version** | | | 2.0 |
| **Use-case Name** | Background Color Subtraction | | | | |
| **Author** | Nguyễn Hữu Kỳ Long | | | | |
| **Date** | 31/05/2015 | | **Priority** | High | |
| **Actor**   * User   **Summary**   * The use case describes the way subtracting background color and checking the result of background color subtraction.   **Goal**   * The system can detect and keep track of the hands.   **Triggers**   * User turns on the switch button on the system hardware.   **Preconditions**   * The system is turned on.   **Post Conditions**   * **On Success**: The system navigates to the interface to select function.   **Main Success Scenario**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | - User turns on the switch button on the system hardware. | - The system displays the notify requiring users move out of the camera area.  - The system shows the images captured from camera on the interface for users.  - The system shows the count down time and counts down from 5 by a second.  - The system shows the message “Đang tiến hành"  [Exception No.1] | | 2 | User keeps the background fixed and waits for the count down time is 0. | - The system displays the notify requiring users show the “testing” hand sign inside the camera area.  - The system shows the images subtracting background color on the interface for users.  - The system shows the count down time and counts down from 5 by a second.  - The system shows the message “Đang tiến hành" | | 3 | User shows the right “testing” sign inside the camera. | - The system continues counting down.  - The system shows a message “Thành Công”.  [Alternative No.1] | | 4 | User waits for count down time is 0. | - The system navigates to the interface for user selects function.  [Alternative No.2] |   **Alternative Scenario**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User shows the wrong “testing” signs or the background user selected is not good. | - The system continues counting down.  - The system shows a message “Đang tiến hành”. | | 2 | The user “testing” sign is not recognized. | - The system shows message “Thất bại”  - The system backs to the Step No.1 |   **Exceptions**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | |  |  |  |   **Relationships**   * “Background Color Subtraction” use case is conditionally extended by “Recognize Hand Sign Language” use case and “Learn Hand Sign Language” in extension points “Hand Sign Recognition” and “Learning Hand Sign” respectively.   **Business Rules**   * The background color subtraction will executes immediately when the application starts working. * The images captured from camera will shows continuously for users can follow. * The count down time informs users of the system is working and will finish in five seconds. * The recognizing “testing” hand sign step is to check whether background is good to recognize and the recognition will execute continuously in five seconds. * Users can get the “Thành công” message immediately when the “testing” hand sign recognition is successful and the recognition will stop. However, the count down time still continues down to 0 and then the system will move to the function interface. * After five seconds, if the “testing” sign recognition is unsuccessful, the system will backs to the first step in five seconds. | | | | | |

#### Recognize Hand Sign Language

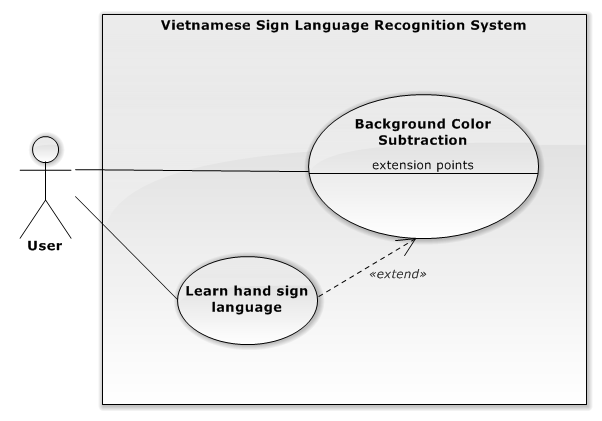
****

**Figure 6: Recognize Hand Sign Language use case diagram**

**Use Case Specification**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **USE CASE - 2 SPECIFICATION** | | | | | |
| **Use-case No.** | VSLR002 | **Use-case Version** | | | 2.0 |
| **Use-case Name** | Recognize Hand Sign Language | | | | |
| **Author** | Nguyễn Hữu Kỳ Long | | | | |
| **Date** | 31/05/2015 | | **Priority** | High | |
| **Actor**   * User   **Summary**   * The use case describes the way recognizing hand signs captured by camera.   **Goal**   * Recognize hand signs and translate them to the same meaning content with the kind of sound and text.   **Triggers**   * User shows the specific “select” hand sign on the “Nhận dạng” function area that is drawn on the images shown continuously to user.   **Preconditions**   * Background color subtraction is successful.   **Post Conditions**   * **On Success**: The translated content shows on the screen and speaker of LCD.   **Main Success Scenario**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | - User completes background color subtraction step | - The system shows a notify “Hãy chọn chức năng mong muốn bằng cách đưa ký hiệu hình bên vào vùng chức năng đó!” in the groupbox “Thông Báo”  - The analyzed images show on the interface continuously.  - Two white “Nhận Dạng” and “Học” area are drawn inside analyzed images.  - The system shows an image guiding users to select function. | | 2 | - User shows “select” hand sign inside “Nhận dạng” area. | - The system shows the hand sign recognition interface  - The analyzed images show on the interface continuously.  - A notify “Hệ thống sẽ lưu lại kết quả nhận dạng sau 3 giây” is shown in groupbox “Thông Báo”  - Countdown time is shown from 3 in groupbox “Thời Gian”  - The system shows two groupbox “Nội dung toàn bộ” and “Kết Quả Hiện Tại” with empty content. | | 3 | User shows the hand sign through camera | - Countdown time counts down by second.  - The system shows messages containing the recognition result of the current hand sign continuously in the group “Kết Quả Hiện Tại”.  [Alternative No.1] | | 4 | User waits for the counting down counts to 0. | - The entire translated content will be updated and shown in the group “Nội Dung Toàn Bộ”  - The system backs to step No.1.  [Alternative No.2]  [Alternative No.3]  [Alternative No.4] |   **Alternative Scenario**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Can not detect the hand inside the camera area. | - Countdown time counts down by second.  - The system shows a message “Không tìm thấy bàn tay!” in the group “Kết Quả Hiện Tại”. | | 2 | Detect the “end” hand sign inside the camera area. | - The system navigates to the interface at the step No.1. | | 3 | Detect the “speak” hand sign inside the camera area. | - The system reads the whole content result via LCD speaker.  - After that, the whole content will be cleared.  - The system backs to step No.2 | | 4 | No hand inside the camera area | - The system remains the whole content without updating.  - The system backs to step No.2 |   **Exceptions**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | |  |  |  |   **Relationships**   * “Recognize hand sign language” use case is an optional extension of “background color subtraction”   **Business Rules**   * The images captured from camera will be analyzed be system and then shown on the interface for users can keep track their hands. * The recognition result of the current hand sign will be shown as soon as possible for users can check the hand sign. * The last recognition result of the hand sign will update to the entire translated content every 3 seconds. * Every 3 seconds, if the hand sign is recognized with “speak” hand sign, the system will speak the whole content via speaker before clearing the content. * When the message “Không tìm thấy bàn tay!” is shown, that means the system does not detect user’s hands on the captured image. * Every 3 seconds, if the system does not detect user’s hands, it will not update the whole content. * Recognizing works continuously until the user shows “end” hand sign through camera. * The recognition result is always gotten from the system database. | | | | | |

#### Learn Hand Sign



**Figure 7: Learn sign use case diagram**

**Use Case Specification**

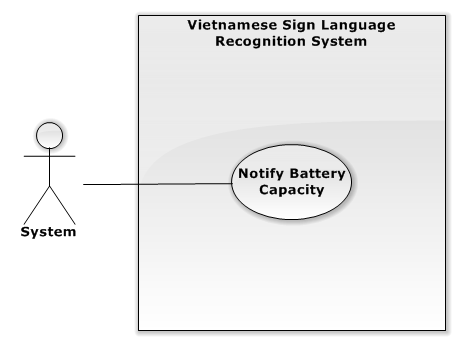
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **USE CASE - 3 SPECIFICATION** | | | | | |
| **Use-case No.** | VSLR003 | **Use-case Version** | | | 2.0 |
| **Use-case Name** | Learn Hand Sign | | | | |
| **Author** | Nguyễn Hữu Kỳ Long | | | | |
| **Date** | 31/05/2015 | | **Priority** | Medium | |
| **Actor**   * User   **Summary**   * The use case describes the way practicing a hand sign.   **Goal**   * It is to help user training his or her hand gesture more accurately.   **Triggers**   * User shows the specific “select” hand sign on the “Học” function area that is drawn on the images shown continuously to user.   **Preconditions**   * Background color subtraction is successful.   **Post Conditions**   * **On Success**: The system shows the image describing the selected word and the mean of the hand sign which is captured.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | - User completes background color subtraction step. | - The system shows a notify “Hãy chọn chức năng mong muốn bằng cách đưa ký hiệu hình bên vào vùng chức năng đó!” in the groupbox “Thông Báo”  - The analyzed images show on the interface continuously.  - Two white “Nhận Dạng” and “Học” area are drawn inside analyzed images.  - The system shows an image guiding users to select function. | | 2 | - User shows “select” hand sign inside “Học” area. | - The system shows the list of words on the interface in the group box “Hướng Dẫn”.  - The analyzed images show on the interface continuously.  - A notify “Hãy đưa kí hiệu trong hướng dẫn vào vùng mũi tên lên xuống để thay đổi từ được chọn ” is shown in the groupbox “Thông Báo”  - Two white “Lên” and “Xuống” area were drawn on these images showing on the interface. | | 3 | User shows the hand gesture through camera. | - The system returns the recognition result on text of the current hand sign continuously in the group “Kết Quả Hiện Tại”.  [Alternative No.1]  [Alternative No.2]  [Alternative No.3]  [Alternative No.4] |   **Alternative Scenario**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Detect the “select” hand sign inside the “Lên” area drawn on the interface. | - The system will move the selection up to upper word in the list of words.  - The system shows the image describing the selected word. | | 2 | Detect the “select” hand sign inside the “Xuống” area drawn on the interface. | - The system will move the selection down to lower word in the list of words.  - The system shows the image describing the selected word. | | 3 | Detect the “end” hand sign through camera. | - The system navigates to the interface at the step No.1. | | 4 | Detect no hand inside the camera area. | - The system shows a message “Không tìm thấy bàn tay!” in the group box “Kết Quả Hiện Tại” |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | |  |  |  |   **Relationships**   * “Learn hand sign language” use case is an optional extension of “background color subtraction”   **Business Rules**   * List of words is always loaded from the system database. * The word “A” will be selected first. * The image corresponding to the selected word will be loaded and shown on the interface. * Recognizing processes continuously until the “end” hand sign is recognized. * Users just select one of two functions. * When the message “Không tìm thấy bàn tay!” is shown, that means the system does not detect user’s hands on the captured image. * The “Lên” rectangle is to move selecting up one-step in the list of words. * The “Xuống” rectangle is to move selecting down one-step in the list of words. * The recognition result of the current hand sign will be shown as soon as possible for users can check the hand sign. | | | | | |

#### Charge Battery



**Figure 8: Learn sign use case diagram**

#### Notify Battery Capacity



**Figure 10: Notify Battery Capacity use case diagram**

## System Attribute

### Usability

The system should be designed for everyone can use easily in controlling and GUI operations.

#### Graphic User Interface

* The system musts show all instructions, notifications and operations in Vietnamese.

#### Usability

* User just needs to read the user manual which is enclosed with the system for using in the first time. The attached manual guide must be clear. User can read and do by themselves.

#### Hardware controlling

* User can control the device very easily as well as using any electronic device in the daily live.

### Reliability

* The database should be constructed on Vietnamese sign language.
* The system uses “Support Vector Machine” library to recognize hand sign language and OpenCV library to process image.
* The system is using Raspberry PI 2 to process which is popular board in the world.

### Availability

* The system runs continuously about 3 hours with LIPO 2700mAh battery. That means it is safe to user.

### Security

N/A

### Maintainability

* Electronic devices in the system are common so when any electronic equipment, which is attached with the system, is out of ordered, it is so easy to change or to fix at any electronic store.
* The system can be extended in the future.

### Portability

* The system supplies the LIPO battery as power source in which user can use for 3 hours without charging. In addition, the system also provides LIPO B3AC charger for users.
* The system provides a circuit monitoring LIPO battery for users.

### Performance

The system uses Raspberry PI 2 with RAM 1GB as central unit processing, so that the system can recognize one hand sign in 1 to 3 seconds and hand sign recognize can be performed continuously.

**Use Case Specification**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **USE CASE - 6 SPECIFICATION** | | | | | |
| **Use-case No.** | VSLR001 | **Use-case Version** | | | 2.0 |
| **Use-case Name** | Notify Battery Capacity | | | | |
| **Author** | Nguyễn Hữu Kỳ Long | | | | |
| **Date** | 31/05/2015 | | **Priority** | High | |
| **Actor**   * System   **Summary**   * The use case can notify user the battery capacity when system is working.   **Goal**   * Notify users of the battery capacity can use the system in an appropriate way.   **Triggers**   * Users switch on the system.   **Preconditions**   * The system is ON.   **Post Conditions**   * **On Success**: The battery capacity will shows continuously at the LEDs indicator on the system box and on the interface.   **Main Success Scenario**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | System checks the battery capacity is higher than 75%.  [Alternative No.1]  [Alternative No.2]  [Alternative No.3]  [Alternative No.4] | - System shows four bright leds on the box.  - System shows the “full” battery image on the top right of the application interface. |   **Alternative Scenario**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | System checks the battery capacity is between 75% and 50%. | - System shows three bright leds on the box.  - System shows the “75%” battery image on the interface. | | 2 | System checks the battery capacity is between 50% and 25%. | - System shows two bright leds on the box.  - System shows the “50%” battery image on the interface. | | 3 | System checks the battery capacity is under 25% | - System shows one bright leds on the box.  - System shows the “25%” battery image on the interface.  - System shows a dialog message “Bin yếu vui lòng tắt hệ thống và cắm sạc. Thông báo sẽ được tự động tắt.” within a countdown time from 5 on the system application interface in five seconds. | | 4 | System checks the battery capacity is empty. | - System shows no bright leds on the box.  - System shows the “empty” battery image on the interface.  - System shows a dialog message “Ứng dụng sẽ tắt trong vòng 5 giây. Thông báo sẽ được tự động tắt.” on the system application interface in five seconds. |   **Exceptions**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | |  |  |  |   **Relationships**   * N/A   **Business Rules**   * System will check how much the battery capacity is every five minutes. * System shows the battery capacity by the electricity devices such as LEDs on the system hardware. * The notify dialog will show every five minutes if the system checks that battery capacity is under 25% or empty. * The notify dialog closes automatically after five seconds. * A countdown time will be shown on the notify dialog. * Actually, checking battery capacity is to check the voltage level of battery. Battery Capacity Display circuit will check the voltage level of battery continuously and notify on LEDs indicator and it returns 5 levels of battery capacity to the system application every five minutes. * The battery capacity is higher than 75% if the voltage is higher than 12V. * The battery capacity is between 75% and 50% if the voltage is lower than or equal 12V and higher than 11.3V. * The battery capacity is between 50% and 25% if the voltage is lower than or equal 11.3V and higher than 10.8V. * The battery capacity is lower than 25% if the voltage is lower than or equal 10.8V and 9.9V. * The battery capacity is empty if the voltage is lower than 9.9V. | | | | | |

# Report No. 4 Software Design Description

## Design Overview

* This document describes the technical and user interface design of VSLR System. It includes the architectural design, the detailed design of common functions and business functions and the design of database model.
* The architectural design describes the overall architecture of the system and the architecture of each main component and subsystem.
* The detailed design describes static and dynamic structure for each component and functions. It includes class diagrams, class explanations and sequence diagrams for each use cases.
* The database design describes the relationships between entities and details of each entity.
* Document overview:
* Section 2: gives an overall description of the system architecture design.
* Section 3: gives component diagrams that describe the connection and integration of the system.
* Section 4: gives the detail design description, which includes class diagram, class explanation, and sequence diagram to details the application functions.
* Section 5: describe a fully attributed ERD.

## System Architectural Design

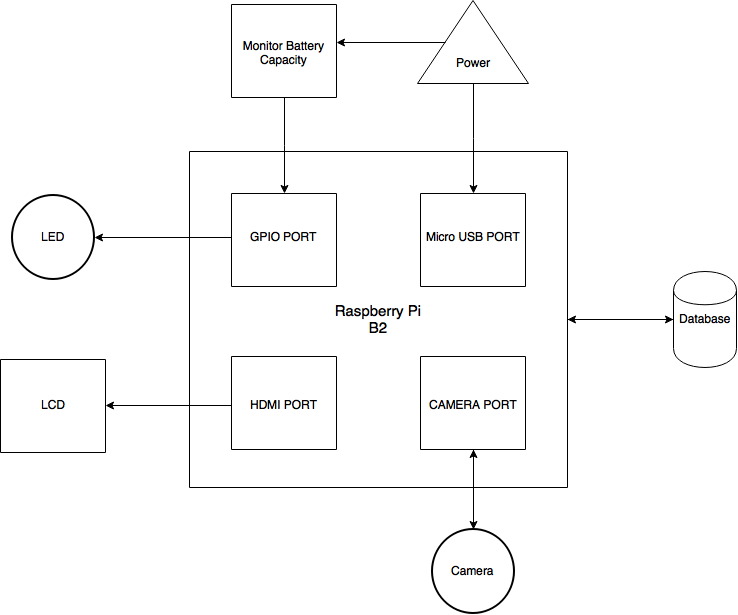


Figure 1: VSLR System Architectural



## Component Diagram

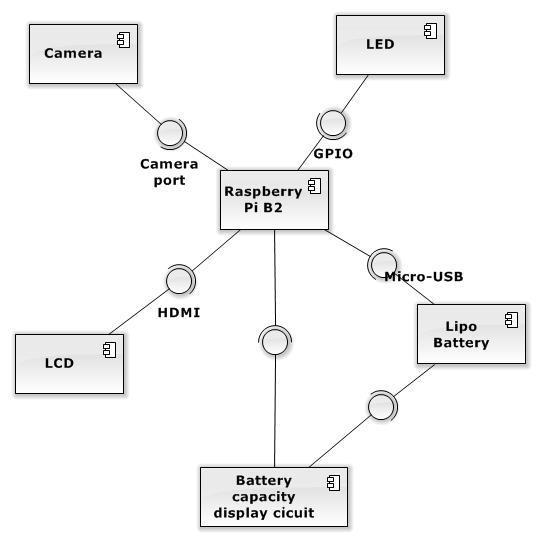


Figure 2: Component Diagram

## Detailed Description

### **Software Detailed Description**

#### Class Diagram



Figure 3: Class Diagram

|  |  |
| --- | --- |
| Class Dictionary : Class Description | |
| Class Name | Description |
| HandSignDTO | This is data transfer object class of table HandSign encapsulating HandSign attributes. |
| HandSigns | This class contains list of hand sign loaded from database and methods for getting attributes of a specific hand sign. |
| BackgroundTimerThread | This class inherits QThread class that is used for sending signals to notify, count down timer and move between steps in background color subtraction phase.  This thread will stop if the phase testing background color subtraction succeeds. |
| RetreivingFrameThread | This class inherits QThread class that is used for activing camera and then retrieving images captured from camera continuously.  This thread will run until the application stops. |
| ShowingImageThread | This class inherits QThread class that is used for getting image from RetreivingFrameThread when it finish , processing these image to subtract background color which will show on the interface.  This thread will run until the application stops. |
| ImageProcessingThread | This class inherits QThread class that receives image subtracted background color to detect hands, create the binary image containing hand features and then extract the features from these image.  This thread receives signals asking to perform image processing for testing background subtraction, learning function and recognition function.  This thread just is enable to process when it receives image from ShowingImageThread.  This thread will run until the application stops. |
| HandGesture | This class encapsulates attributes, methods to create binary images containing hand features and methods to output features related to height, hand palm and finger lines. |
| ExtractingBinaryImage | This class provides methods which outputs histogram features of a binary image. |
| SignRecognition | This class encapsulates attributes and methods to perform SVM algorithm for hand sign recognition. |
| Speech | This class encapsulates attributes and methods to initiate some parameters for using espeak library to read a content in sound via LCD speaker. |
| RecognitionContent | This class encapsulates attributes and methods to manage the whole recognition content for hand sign language recognition function. |
| RecognitionTimerThread | This class inherits QThread class that is used for implementing the real time timer to manage hand sign recognition function.  This thread will start when the application runs but it is just enable to process when the recognition function is selected.  This thread will stop until the application stops. |
| BatteryThread | This class inherits QThread class that will run during the application works to check battery capacity and send signals notifying user. |
| LowBatteryDialog | This class is used for initiating dialog interface which shows low battery announcement. |
| LowBatteryTimerThread | This class inherits QThread processes as real time timer to manage the time showing LowBatteryDialog. |
| MainWindow | This class is the main UI thread which manages the application interfaces and creates communications between thread objects. |

#### Class Diagram Explanation

##### HandSigns

Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| signsMap | QMap<int, HandSignDTO> | Private | This map contains hand sign records loaded from HandSign table. |

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| getAllSigns | QStringList | Private | Returns all records signsMap hold. |
| getLinkBySelectedIndex | QString | Private | Return record by index user select in combobox. |
| getMeanByID | QString | Private | Return record by recognition result returned from SVM. |

##### HandSignDTO

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | int | Private | Unique identifier of a word and recognition result number. |
| content | QString | Private | Content of a particular sign. |
| link | QString | Private | Path to directory contain image of hand sign. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| SetId | void | Public | Set value of attribute id |
| GetId | int | Public | Get attribute id value |
| SetLink | void | Public | Set value of attribute link |
| GetLink | QString | Public | Get attribute link value |
| SetContent | void | Public | Set value of attribute content |
| GetContent | QString | Public | Get attribute content value |

##### 

##### BackgroundTimerThread

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| STOP | bool | Public | This variable is used for to stop the thread before the application is closed. |
| mutex | QMutex | Private | It is to protect section of code so that only one thread can access at a time. |
| isEnableToCountDown | bool | Private | This variable is to check whether this thread continues counting down timer. |
| testingResult | bool | Private | This variable is to held background subtraction testing result and it is condition to stop this thread. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| run | void | Private | This is starting point for the thread and this method implements functions of the thread. |
| sendSignalChangingToHandDetectionMode | void | Private | This is signal which is sent Image Processing thread to move to hand detection step. |
| sendSignalChangingToTestingTime | void | Private | This is signal which is sent Image Processing thread to move to background subtraction testing step. |
| sendSignalFinishingColorSubtraction | void | Private | This is signal which is sent with testing result to notify main thread. |
| sendSignalGetTestingResult | bool | Private | This is signal which is sent Image Processing thread to get background subtraction testing result and stop testing step. |
| sendSignalChangingLabelNotice | void | Private | This is signal which is sent with notify content to main thread. |
| sendSignalFailTesingResult | void | Private | This is signal which is sent to notify main thread of testing result. |
| continueCountDown | void | Private | This is function that is called in response to signal which enables to continue counting down timer. |

##### 

##### SignRecognition

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| svmModel | struct svm\_model | Private | SVM will load the model file to this struct. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| getRecognitionResult | double | Public | This method receives the hand sign features, then uses SVM library to predict recognition result and return it. |

##### Speech

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| bufferLength | int | Private |  |
| options | int | Private |  |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| speak | void | Public |  |

##### RecognitionTimerThread

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| STOP | bool | Public | This variable is used for to stop the thread before the application is closed. |
| mutex | QMutex | Private | It is to protect section of code so that only one thread can access at a time. |
| isEnableWorking | bool | Private | This variable is to check whether the thread can process functions. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| enableWorking | void | Public | This is function that is called in response to signal which enables to perform functions. |
| run | void | Private | This is starting point for the thread and this method implements function counting down timer. |
| sendSignalUpdatingContent | void | Protected | This is signal which is sent to notify main thread of performing updating recognition content. |
| sendSignalCountDownRecognitionTime | void | Protected | This is signal which is sent main thread the real time timer. |

#### 

##### RetrievingFrameThread

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| STOP | bool | Public | This variable is used for to stop the thread before the application is closed. |
| frame | Mat | Private | This is object holding image captured from camera. |
| camera | VideoCapture | Private | This is object which actives camera working and retrieves image captured from camera to frame object. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| run | void | Private | This is starting point for the thread and this method implements retrieving images continuously. |
| changingFrame | void | Protected | This is signal which is sent to notify Showing Image thread that new image is retrieved. |

##### 

##### RecognitionContent

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| content | QString | Private | This is the whole recognition content will be updated every 3 seconds. |
| previousWord | QString | Private | This object holds the last updated hand sign. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| updateContent | QString | Public | This method receives new hand sign recognition result and then performs updating content function. |
| clear | void | Public | This method is used for clear the current content. |

##### LowBatteryTimerThread

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| STOP | bool | Public | This variable is used for to stop the thread before the application is closed. |
| mutex | QMutex | Private | It is to protect section of code so that only one thread can access at a time. |
| isEnableWorking | bool | Private | This variable is to check whether the thread can process functions. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| enableWorking | void | Public | This is function that is called in response to signal which enables thread can perform functions. |
| run | void | Private | This is starting point for the thread and this method implements retrieving images continuously. |
| sendSignalCountingDown | void | Protected | This is signal which is sent the real time timer to low battery dialog. |
| sendSignalClosingDialog | void | Protected | This is signal which is to notify main thread of closing current low battery dialog. |

#### 

##### BatteryThread

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| STOP | bool | Public | This variable is used for to stop the thread before the application is closed. |
| mutex | QMutex | Private | It is to protect section of code so that only one thread can access at a time. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| run | void | Private | This is starting point for the thread and this method implements retrieving images continuously. |
| sendSignalShowingBatteryCapacity | void | Protected | This is signal which is to notify main thread of current battery capacity every 5 minutes. |

#### 

##### LowBatteryDialog

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
|  |  |  |  |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| countDownTimer | void | Public | This method is used for changing the timer which is shown on the low battery dialog interface. |

##### HandGesture

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| contours | vector<vector<Point> > | Public | This holds sets of points as the contours of possible hands. |
| cMaxId | int | Public | This variable is to determines which contour is of right hand. |
| boundingRect | Rect | Public | This is rectangle bounding the contour of the right hand. |
| approxContour | vector<Point2f> | Public | This is contour of the hand in the form of set of float points after it is approximated polygonal curves. |
| biggestApporxContour | vector<Point > | Public | This is contour of the hand in the form of set of integer points after it is approximated polygonal curves. |
| hullI | vector<int> | Public | This is convex hull output. |
| defects | vector<Vec4i> | Public | This is convexity defects output. |
| correctDefects | vector<Point> | Public | This is convexity defects output after verifying which is correct. |
| inCircle | Point | Public | This is center location of hand palm. |
| inCircleRadius | double | Public | This is radius length of hand palm. |
| ratioHeightAndWidth | double | Public | This is height feature output of the hand sign. |
| isHand | bool | Public | This is output of detecting the hands. |
| nTotalLine | double | Private | This variable holds the number of finger lines. |
| totalLenLine | double | Private | This variable hold the total length of all finger lines. |
| n0To45Line | double | Private | This variable holds the number of finger lines belongs to 0 to 45 degree category. |
| totalLen0To45Line | double | Private | This variable hold the total length of all finger lines belongs to 0 to 45 degree category. |
| n46To90Line | double | Private | This variable holds the number of finger lines belongs to 46 to 90 degree category. |
| totalLen46To90Line | double | Private | This variable hold the total length of all finger lines belongs to 46 to 90 degree category. |
| n91To135Line | double | Private | This variable holds the number of finger lines belongs to 91 to 135 degree category. |
| totalLen91To135Line | double | Private | This variable hold the total length of all finger lines belongs to 91 to 135 degree category. |
| n136To180Line | double | Private | This variable holds the number of finger lines belongs to 136 to 180 degree category. |
| totalLen136To180Line | double | Private | This variable hold the total length of all finger lines belongs to 136 to 180 degree category. |
| nN1ToN45Line | double | Private | This variable holds the number of finger lines belongs to -45 to -1 degree category. |
| totalLenN1ToN45Line | double | Private | This variable hold the total length of all finger lines belongs to -45 to -1 degree category. |
| nN46ToN90Line | double | Private | This variable holds the number of finger lines belongs to -46 to -90 degree category. |
| totalLenN46ToN90Line | double | Private | This variable hold the total length of all finger lines belongs to -46 to -90 degree category. |
| nN91ToN135Line | double | Private | This variable holds the number of finger lines belongs to -91 to -135 degree category. |
| totalLenN91ToN135Line | double | Private | This variable hold the total length of all finger lines belongs to -91 to -135 degree category. |
| nN136ToN179Line | double | Private | This variable holds the number of finger lines belongs to -136 to -179 degree category. |
| totalLenN136ToN179Line | double | Private | This variable hold the total length of all finger lines belongs to -136 to -179 degree category. |
| whiteColor | Scalar | Private | White color is used to draw binary images. |
| blackColor | Scalar | Private | Black color is used to draw binary images. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| findBiggestContour | void | Public | This method is to find which contour is right. |
| detectIsHand | bool | Public | This method is to detect the biggest contour is contour of hand or not. |
| findInscribedCircle | void | Public | This method is to specify the hand palm. |
| drawPalmAndFingerLine | void | Public | This method is to draw two binary images such as hand palm and finger line. |
| getHeightFeatures | void | Public | This method returns the height feature of hand. |
| getRadiusFeatures | void | Public | This method calculates and returns radius features of hand palm. |
| getAngleFeatures | void | Public | This method calculates and returns angle features of finger lines. |
| getLinesFeatures | void | Public | This method calculate and returns classifier features of finger lines. |
| classifyLine | void | Public | This method classifies finger lines to 8 degree categories. |
| getSelectPoint | Point | Public | This method is to find the highest point of select hand sign and return it. |

#### 

##### ShowingImageThread

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| STOP | bool | Public | This variable is used for to stop the thread before the application is closed. |
| mutex | QMutex | Private | It is to protect section of code so that only one thread can access at a time. |
| BACKGROUND\_MODE | int | Private | This is a constant static variable holds value of background mode. |
| GETTING\_BACKGROUND\_MODE | int | Private | This is a constant static variable holds value of step sampling background. |
| HAND\_DETECTION\_MODE | int | Private | This is a constant static variable holds value of mode detecting hand. |
| isWorking | bool | Private | This variable is to check whether this thread is working. |
| frame | Mat | Private | This is image received from Retrieving Image thread. |
| blurMat | Mat | Private | This is image after blurring image process. |
| labMat | Mat | Private | This is image after converting BGR image into LAB image. |
| showMat | Mat | Private | This is image subtracted background and it is used for showing on the interface. |
| binMat | Mat | Private | This is binary image subtracted background. |
| mode | int | Private | This variable holds value of current mode. |
| IMAGE\_COLS | int | Private | This is constant static variable holds image’s width value. |
| IMAGE\_ROWS | int | Private | This is constant static variable holds image’s height value. |
| element | Mat | Private | This is structuring element is for morphological transformations. |
| LIGHT\_RANGE | double | Private | This is constant static variable holds range value of lightness. |
| WARM\_RANGE | double | Private | This is constant static variable holds range value of warm color-opponent. |
| COOL\_RANGE | double | Private | This is constant static variable holds range value of cool color-opponent. |
| whiteColor | Scalar | Private | White color is used to draw binary images. |
| blackColor | Scalar | Private | Black color is used to draw binary images. |
| lowerBoundArray | Scalar | Private | This array holds lower boundary of every single pixels. |
| upperBoundArray | Scalar | Private | This array holds upper boundary of every single pixels. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| setToDefaults | void | Public | This method sets this thread back to first step. |
| onChangingImage | void | Public | This is function that is called in response to signal, which receives new image captured from camera. |
| moveToHandDetectionMode | void | Public | This is function that is called in response to signal, which changes current mode to hand detection mode. |
| run | void | Private | This is starting point for the thread and this method implements processing these image to subtract background color which will show on the interface. |
| sendSignalShowingImage | void | Protected | This is signal which is to send images subtracted background color to main thread can show on the interface. |
| sendImageToExtractFeatures | void | Protected | This is signal which is to send images subtracted background color to Image Processing thread can extract features. |
| sendSignalEnableCountDown | void | Protected | This is signal which is to enable Timer thread to continue working. |

##### 

##### ExtractingBinaryImageFeatures

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
|  |  |  |  |

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Descrition** |
| getFeature1of3VerticalAreas | void | Private | This method extracts 3 vertical area features of binary image. |
| getFeature1of3HorizontalAreas | void | Private | This method extracts 3 horizontal area features of binary image. |
| getFeatures4SquareAreas | void | Private | This method extracts 4 square area features of binary image. |
| getFeature4CornerAreas | void | Private | This method extracts 4 triangle area features of binary image. |
| getFeatures4x4 | void | Private | This method extract 16 square area features of binary image. |
| getFeatures3x3 | void | Private | This method extract 9 square area features of binary image. |

##### ImageProcessingThread

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| STOP | bool | Public | This variable is used for to stop the thread before the application is closed. |
| NO\_MODE | int | Public | This is a constant static variable holds value of no mode. |
| TESTING\_MODE | int | Public | This is a constant static variable holds value of testing background subtraction mode. |
| SELECTING\_MODE | int | Public | This is a constant static variable holds value of selecting function mode. |
| RECOGNITION\_MODE | int | Public | This is a constant static variable holds value of hand sign language recognition mode. |
| LEARNING\_MODE | int | Public | This is a constant static variable holds value of hand sign language learning mode. |
| SELECT\_SIGN | int | Public | This is a constant static variable holds recognition result of “select” hand sign. |
| SELECT\_SIGN\_SECOND | int | Public | This is a constant static variable holds recognition result of “select” hand sign. |
| END\_SIGN | int | Public | This is a constant static variable holds recognition result of “end” hand sign. |
| SPEAK\_SIGN | int | Public | This is a constant static variable holds recognition result of “speak” hand sign. |
| TEST\_SIGN | int | Public | This is a constant static variable holds recognition result of “test” hand sign. |
| mode | int | Private | This variable holds value of current mode. |
| testingResult | bool | Private | This is testing background color subtraction result at testing mode. |
| binaryMat | Mat | Private | This is hand binary image received from Showing Image thread. |
| frame | Mat | Private | This is image received from Showing Image thread which is subtracted background color. |
| croppedFrame | Mat | Private | This is hand images cropped from images subtracted background. |
| croppedHand | Mat | Private | This is cropped images which contains hand after adjusting size. |
| croppedBinHand | Mat | Private | This is cropped binary images which contains hand after adjusting size. |
| croppedInnerHand | Mat | Private | This is cropped binary images which contains hand palm after adjusting size. |
| croppedFingerLines | Mat | Private | This is cropped binary images which contains finger lines after adjusting size. |
| subROI | Mat | Private | This is region of hand image is to hold hand images cropped. |
| subInnerROI | Mat | Private | This is region of hand palm image is to hold hand palm cropped. |
| subFingerROI | Mat | Private | This is region of finger lines image is to hold finger line images cropped. |
| maskForFrame | Mat | Private | This is a mask which is used for cropping hand images. |
| maskForInner | Mat | Private | This is a mask which is used for cropping hand palm images. |
| maskForFingerLine | Mat | Private | This is a mask which is used for cropping finger line images. |
| cropImageSize | Size | Private | This is common size for every cropped images. |
| handGesture | HandGesture | Private | This is object processing images to create binary images and output features related to height, hand palm and finger lines. |
| extractingBinaryImageFeatures | ExtractingBinaryImageFeatures | Private | This is object processing binary images to output histogram features. |
| signRecogntion | SignRecognition | Private | This object is used for recognize hand sign. |
| isEnbaleProcessing | bool | Private | This variable is to check whether this thread can process images. |
| whiteColor | Scalar | Private | White color is to draw binary images containing features. |
| blackColor | Scalar | Private | Black color is to draw binary images containing features. |
| recognitionResultNumber | double | Private | This is recognition result predicted by SVM. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| getRecognitionResult | double | Public | This method return hand sign recognition result. |
| recognizeSign | void | Private | This method implements steps to recognize hand sign. |
| changeToSelectingFunctionMode | void | Public | This method is to change current mode to selecting function mode. |
| changeToTestingMode | void | Public | This method is to change current mode to selecting function mode. |
| receiveBinaryImage | void | Public | This is function that is called in response to signal, which receives new image subtracted background color. |
| getTestingResult | bool | Public | This is function that is called in response to signal, which return testing background subtraction result. |
| run | void | Private | This is starting point for the thread and this method perform image processing for testing background subtraction, learning function and recognition function. |
| sendSignalToChangeLabelTestingResult | void | Protected | This is signal which is sent testing background color subtraction to main thread can update the interface. |
| sendSignalSelectingRecognition | void | Protected | This is signal which is sent notify main thread of changing current mode to recognition mode. |
| sendSignalSelectingLearning | void | Protected | This is signal which is sent notify main thread of changing current mode to learning mode. |
| sendSignalChangeToSelectingMode | void | Protected | This is signal which is sent notify main thread of changing current mode to selecting function mode. |
| sendSignalChangingRecognitionResult | void | Protected | This is signal which is sent notify main thread of outputting recognition result at recognition mode. |
| sendSignalMovingToUpperWord | void | Protected | This is signal which is sent notify main thread of “Lên” area is selected at learning mode. |
| sendSignalMovingToLowerWord | void | Protected | This is signal which is sent notify main thread of “Xuống” area is selected atlearning mode. |
| sendSignalChangingLearningResult | void | Protected | This is signal which is sent notify main thread of outputting recognition result at learning mode. |

#### 

##### MainWindow

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| backgroundTimerThread | BackgroundTimerThread | Private | This object is to manage background color subtraction phase. |
| recognitionTimerThread | RecognitionTimerThread | Private | This object is to manage timer at recognition mode. |
| handSigns | HandSign | Private | This object is to output hand signs content. |
| recognitionSpeech | Speech | Private | This object is to speak recognition content. |
| recognitionContent | RecognitionContent | Private | This object is to manage the whole recognition content at recognition mode. |
| retrievingFrameThread | RetrievingFrameThread | Private | This is thread object which retrieves images captured from camera continuously. |
| showingImageThread | ShowingImageThread | Private | This is thread object which subtracts background color. |
| imageProccessingThread | ImageProccessingThread | Private | This is thread object which processes image to recognize hand sign during the application runs. |
| batteryThread | BatteryThread | Private | This is thread object which outputs battery capacity every 5 minutes. |
| lowBatteryTimerThread | LowBatteryTimerThread | Private | This object is to manage timer of Low Battery Dialog. |
| lowBatteryDialog | LowBatteryDialog | Private | This is UI thread of Low Battery Dialog. |

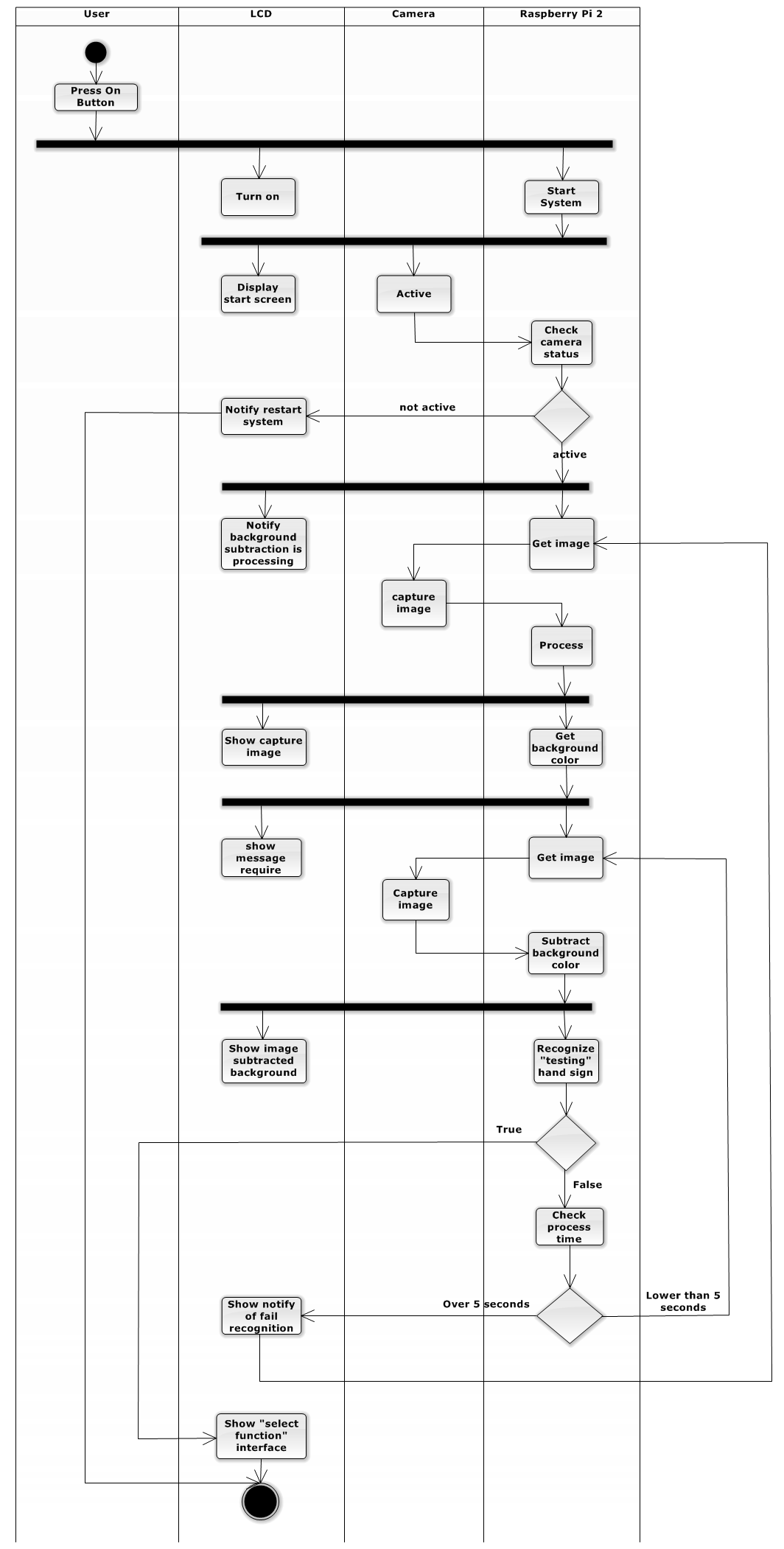
##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| onToShow | void | Private | This is function that is called in response to signal, which shows images on the interface. |
| changeLabelNotice | void | Private | This is function that is called in response to signal, which updates notify on the interface. |
| changeLabelTestingResult | void | Private | This is function that is called in response to signal, which updates testing background color subtraction on the interface. |
| countDownRecognitionTimer | void | Private | This is function that is called in response to signal, which updates real time timer on the interface at recognition mode. |
| changeRecognitionResult | void | Private | This is function that is called in response to signal, which updates recognition result on the interface at recognition mode. |
| updateRecognitionContent | void | Private | This is function that is called in response to signal, which updates new recognition result to the whole content and shows it on the interface at recognition mode. |
| changeToSelectingFunction | void | Private | This is function that is called in response to signal, which implements steps to change to selecting function mode. |
| changeToRecognitionFunciton | void | Private | This is function that is called in response to signal, which implements steps to change to recognition mode. |
| changeToLearningFunction | void | Private | This is function that is called in response to signal, which implements steps to change to learning mode. |
| onFinishingColorSubtraction | void | Private | This is function that is called in response to signal, which checks the testing background subtraction result. |
| moveToUpperWord | void | Private | This is function that is called in response to signal, which moves selection to upper word at learning mode. |
| moveToLowerWord | void | Private | This is function that is called in response to signal, which moves selection to lower word at learning mode. |
| changeLearningResult | void | Private | This is function that is called in response to signal, which updates recognition result on the interface at laerning mode. |
| showBatteryCapacity | void | Private | This is function that is called in response to signal, which shows battery capacity images on the interfaces. |
| closeLowBatteryDialog | void | Private | This is function that is called in response to signal, which closes low battery dialog. |
| initiateSelectingFunctionInterface | void | Private | This methods initiates components of selecting function mode on the interface. |
| initiateLearningInterface | void | Private | This methods initiates components of learning mode on the interface. |
| initiateRecognitionInterface | void | Private | This methods initiates components of recognition mode on the interface. |
| initiateColorSubtractionInterface | void | Private | This methods initiates components of selecting function mode on the interface. |
| changeImageByWordID | void | Private | This is function that is called in response to signal, which updates image of selected word on the interface at learning mode. |
| closeEvent | void | Protected | This is an override method which stops threads which is still working before closing application. |

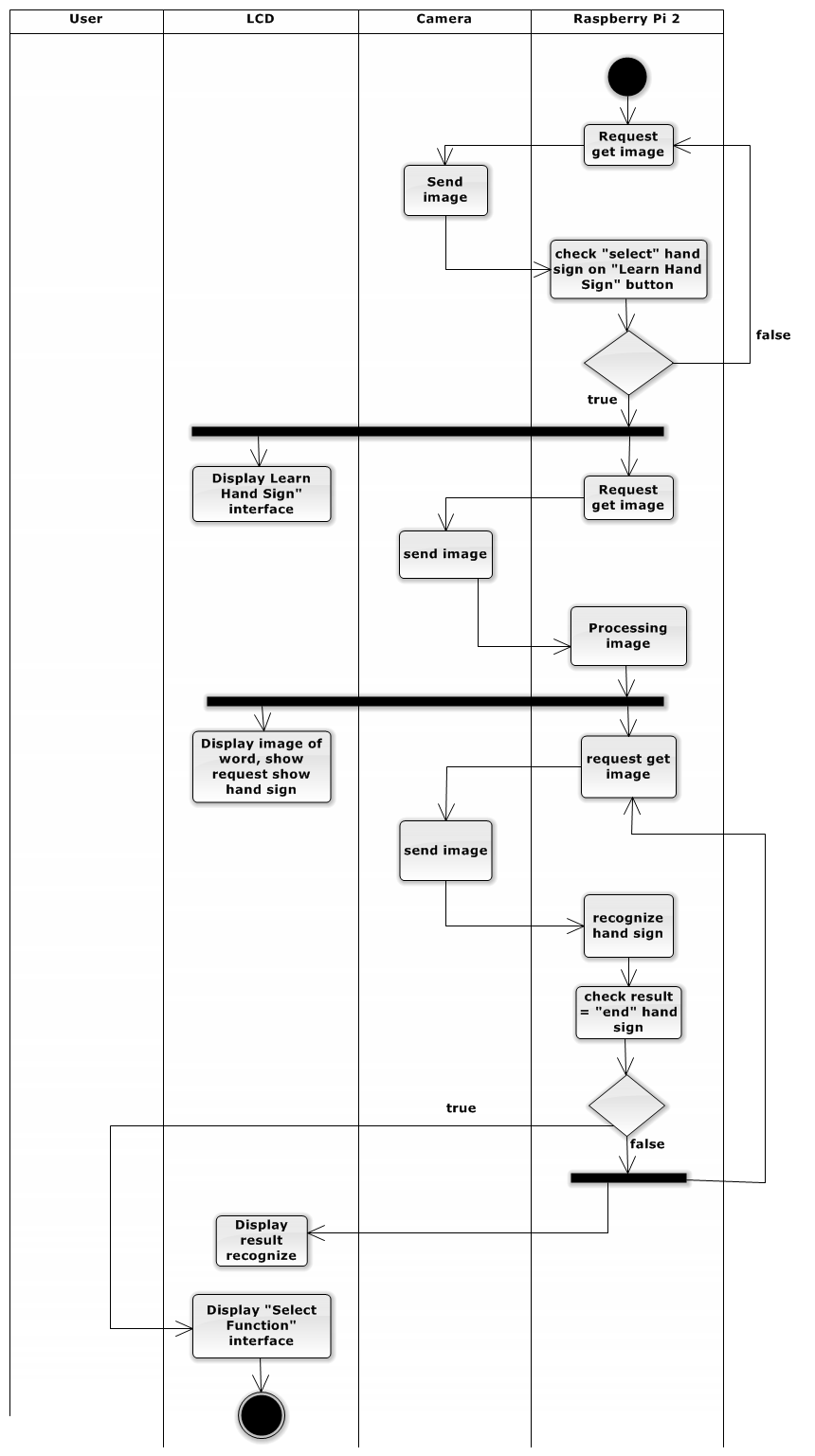
#### 

#### Activity Diagram

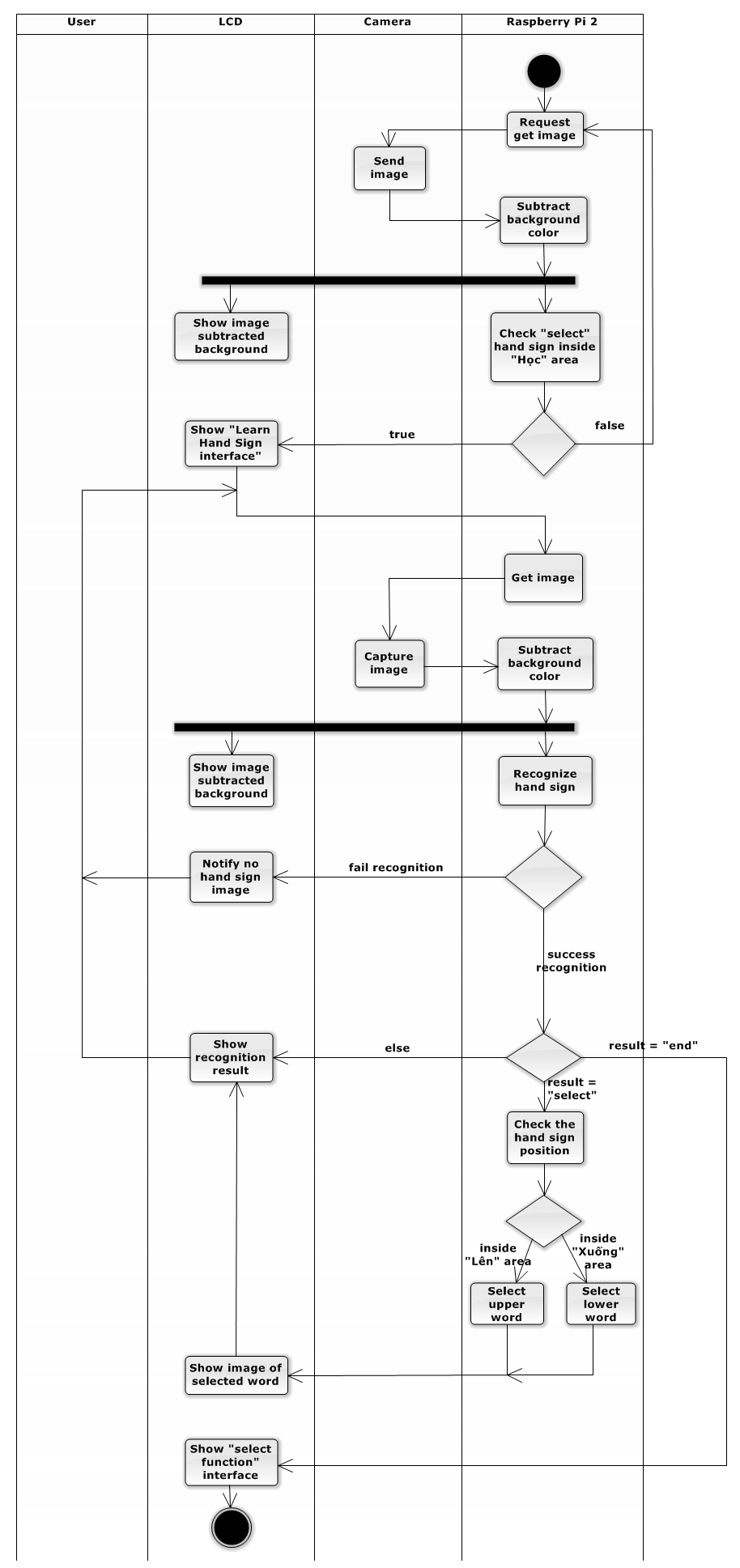
##### Background color subtraction



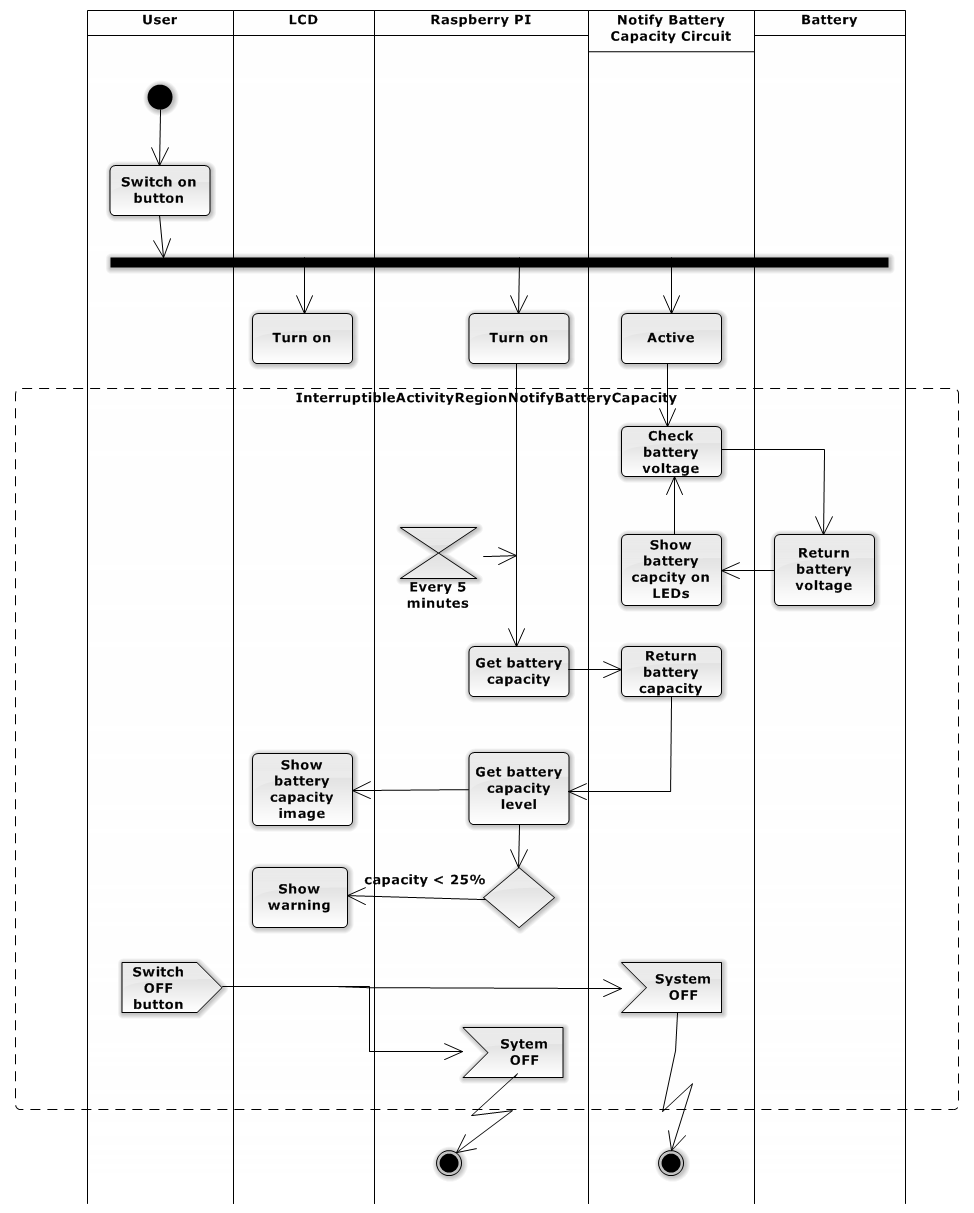
##### Recognize Hand Sign Language



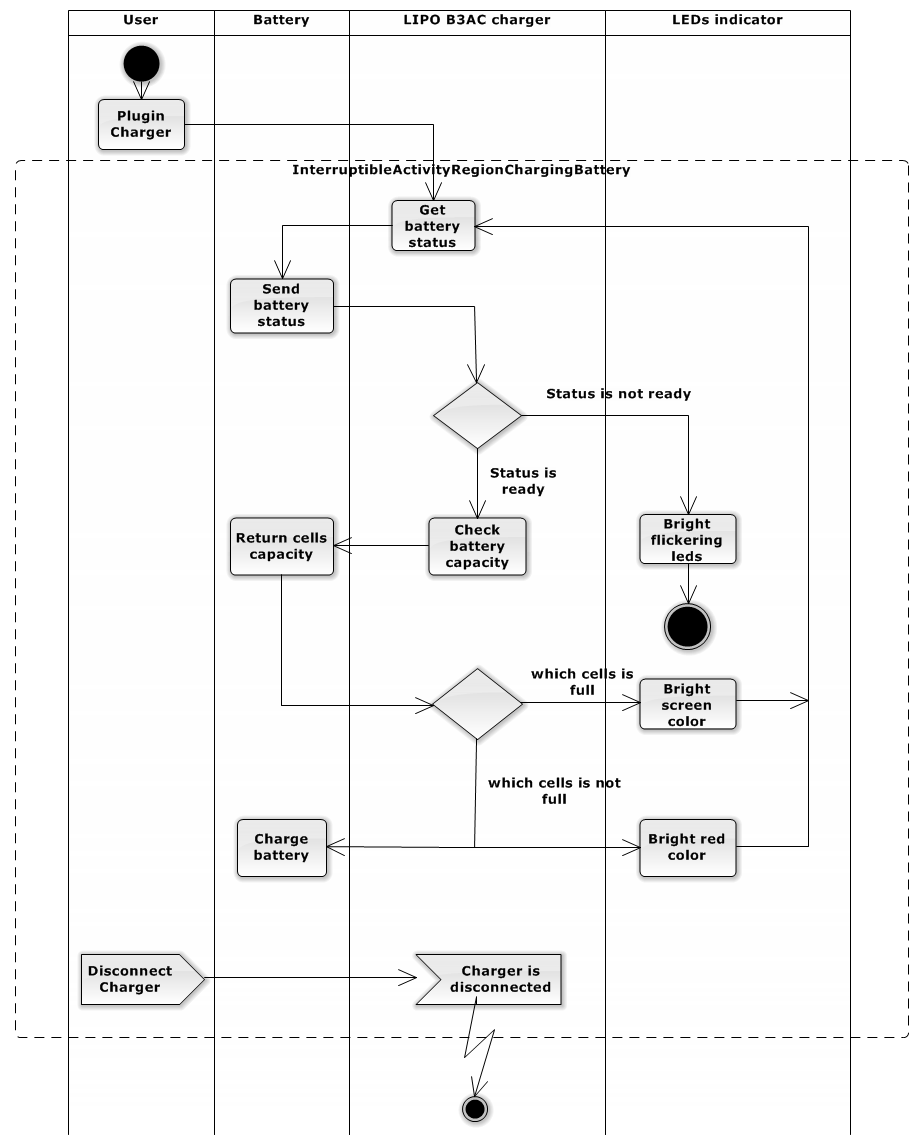
##### Learn Hand Sing Language



##### Notify Battery Capacity

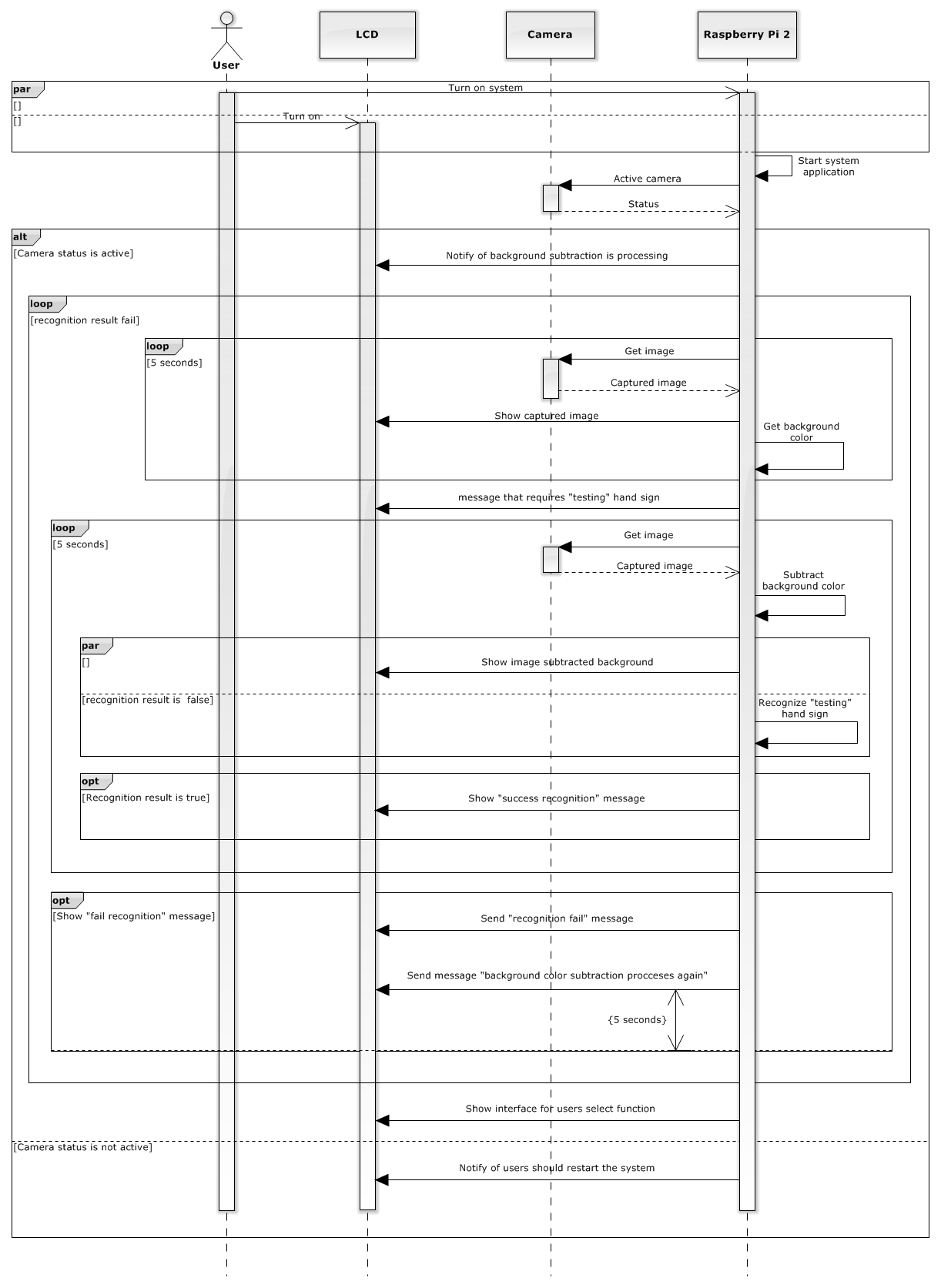


##### Charge Battery

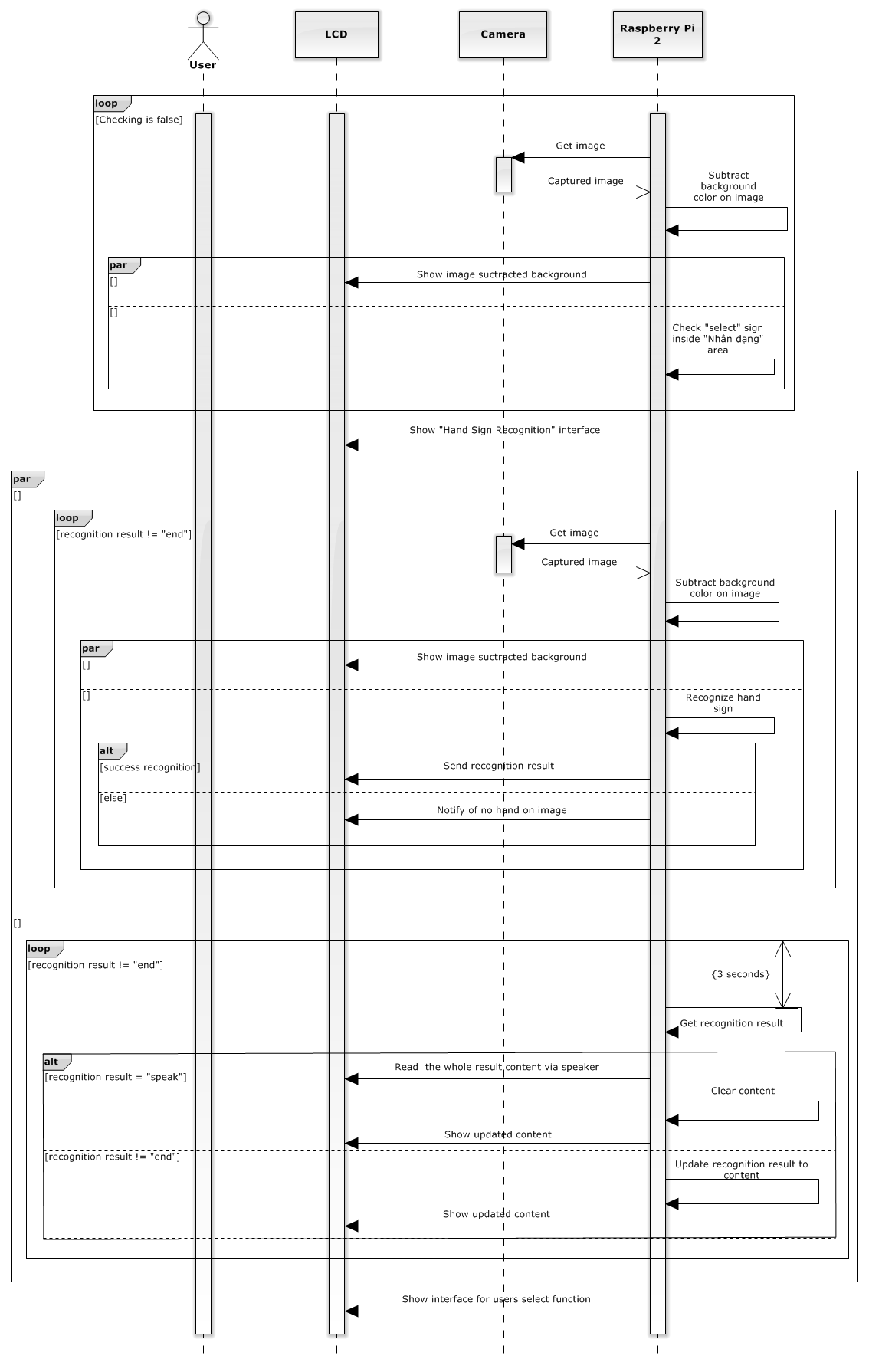


#### Sequence Diagram

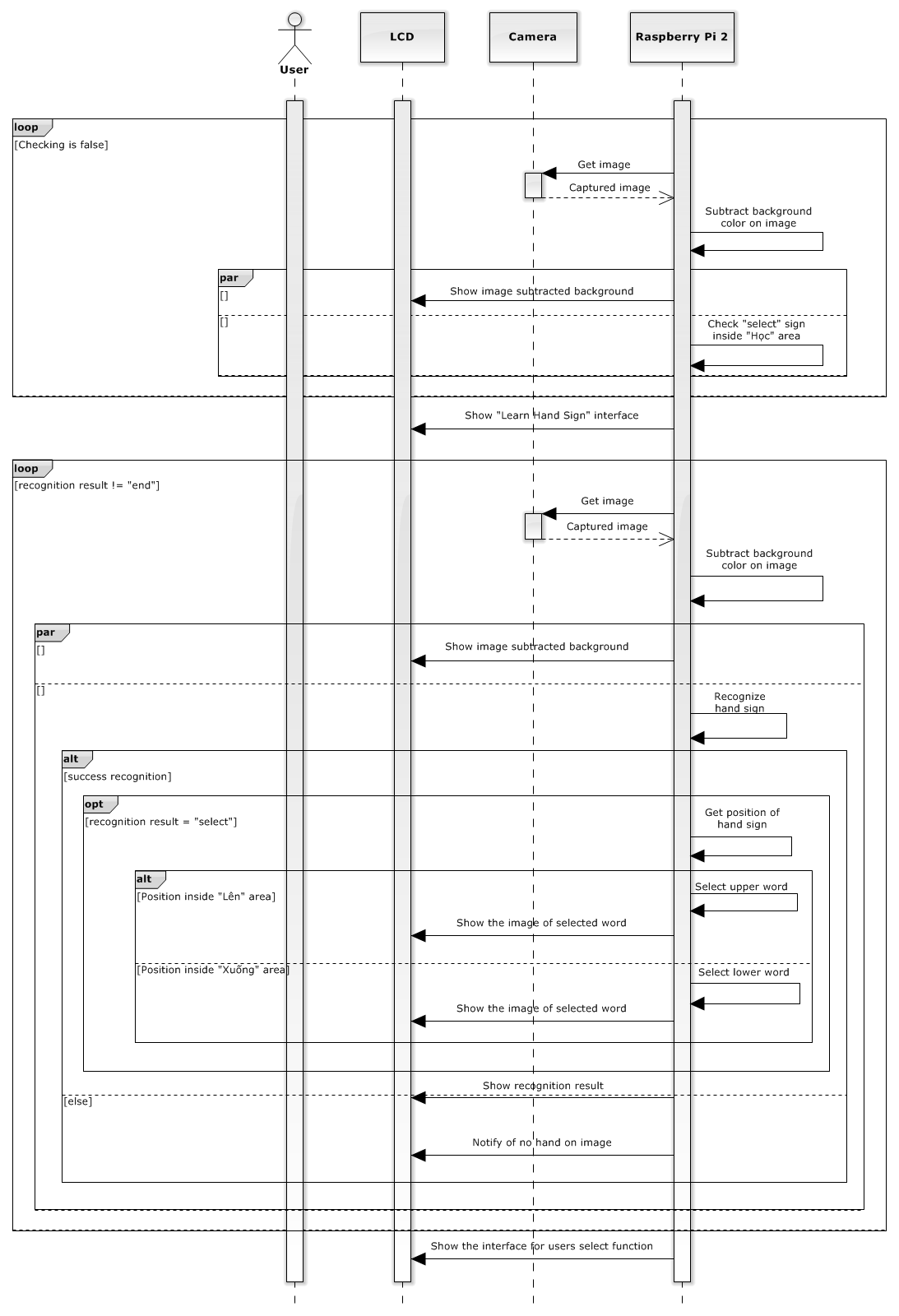
#### Subtract Background Color



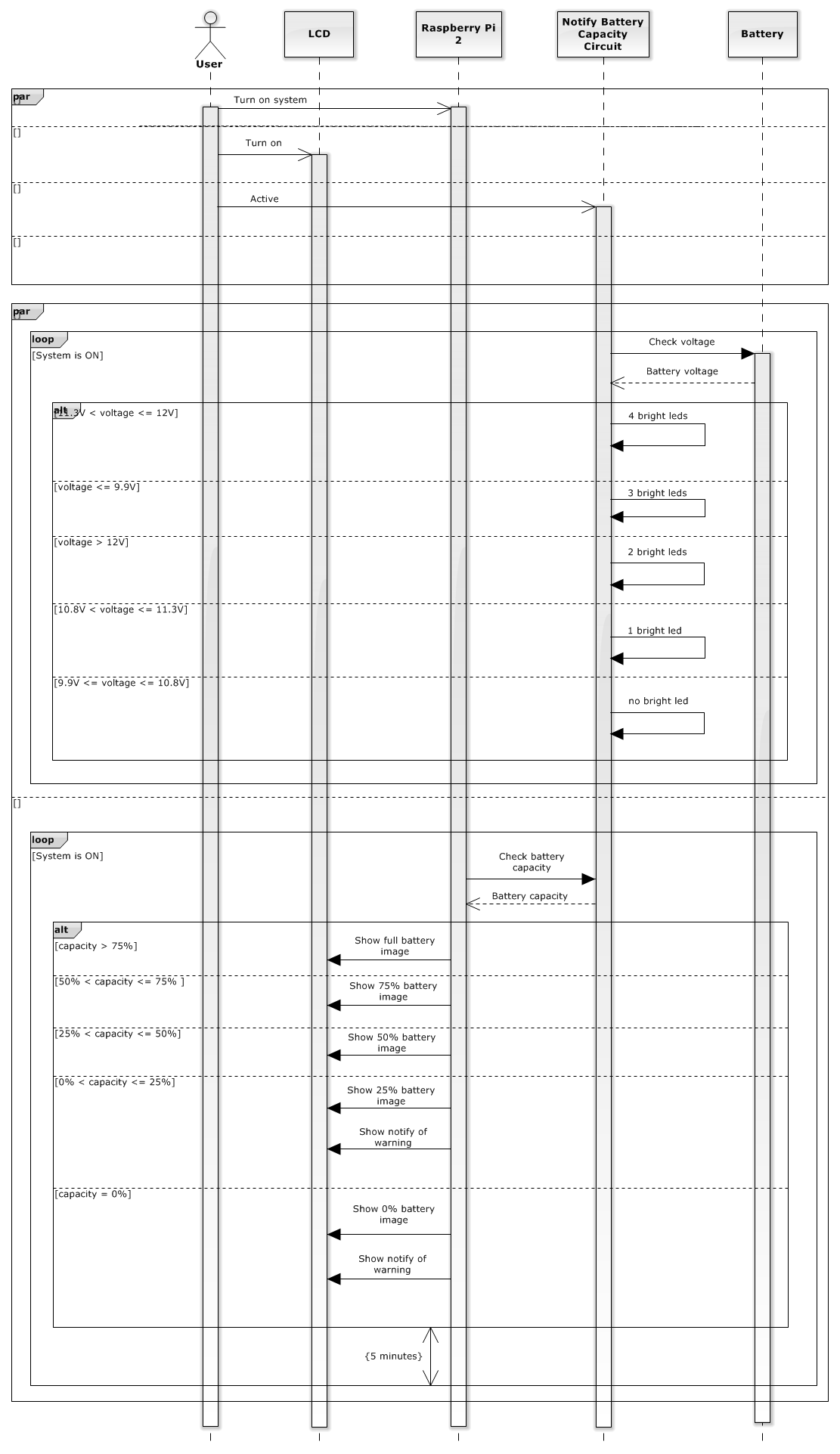
#### Recognize Hand Sign Language



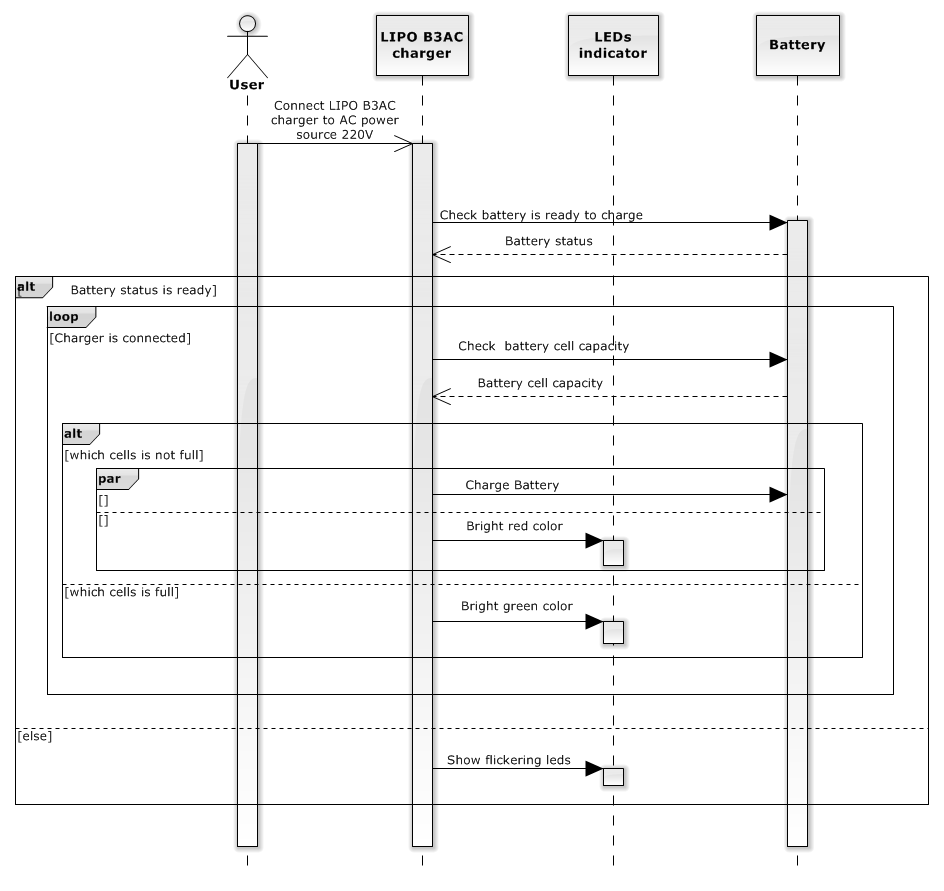
#### Learn Hand Sign Language



#### Notify Battery Capacity

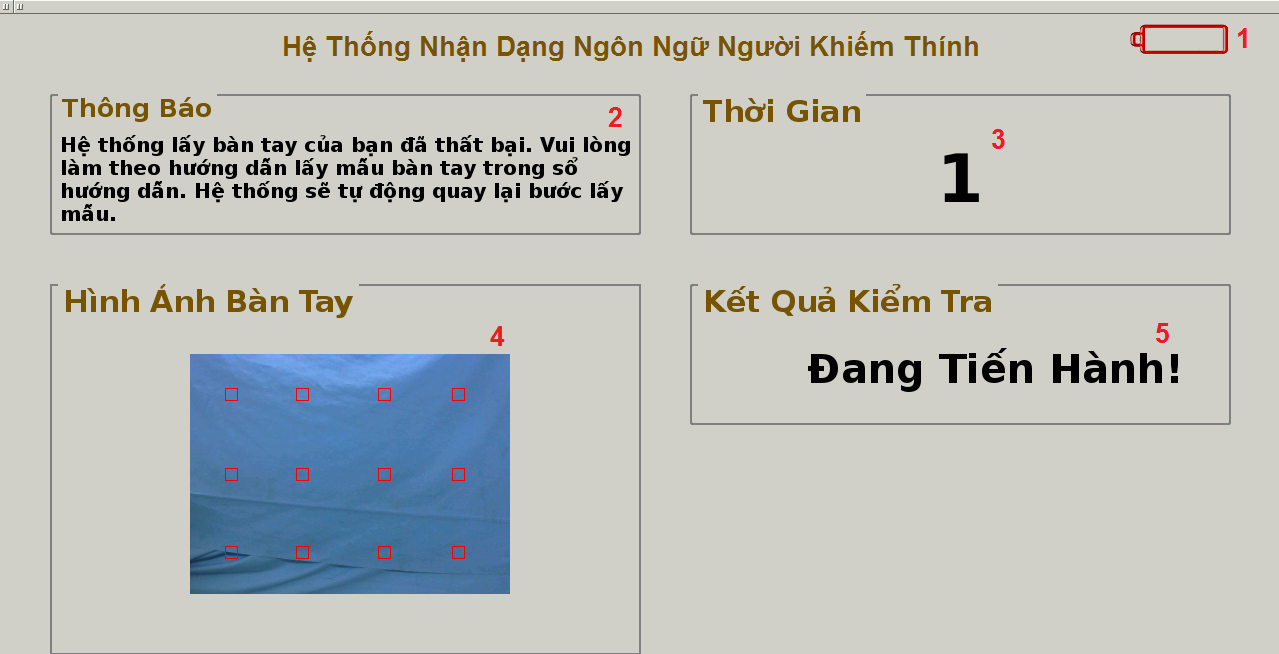


#### Charge Battery



## User Interface Design

### Subtract Background Color



**Figure 5:** Subtract Background Color

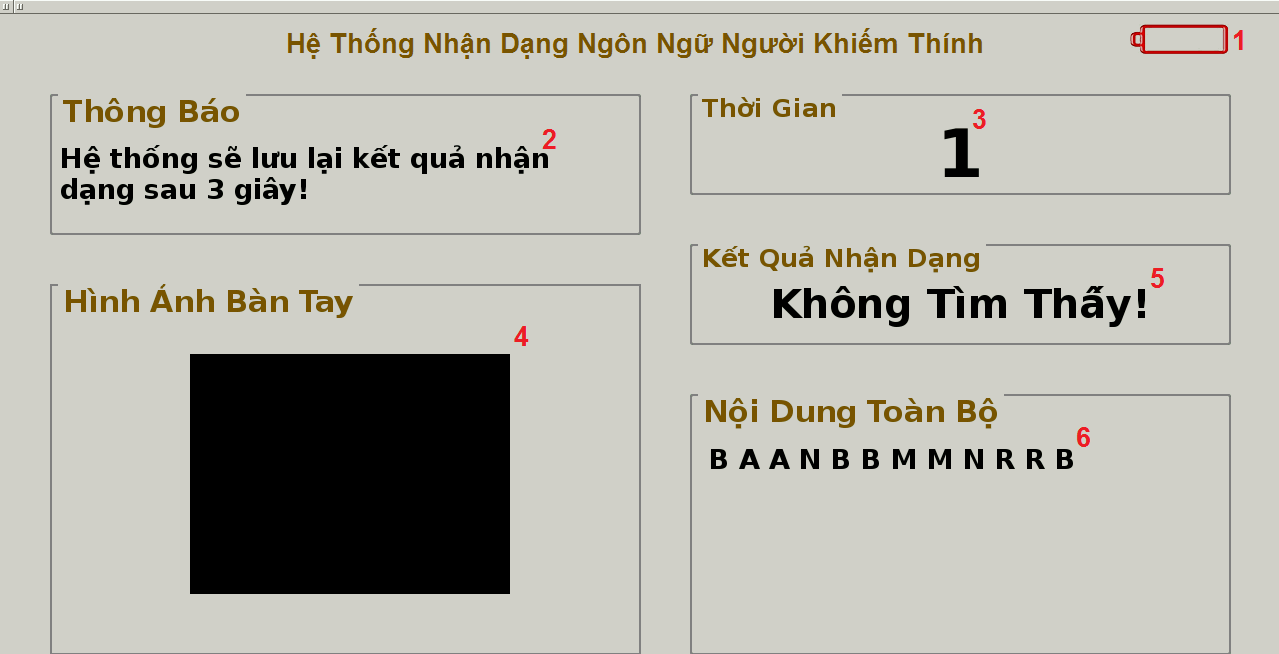
**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |
| 1 | Username | Fill user name | No | Yes | Textbox | String | N/A |
| 2 | Password | Fill password | No | Yes | Password | String | N/A |

**Buttons/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 3 | Sign in | Log-in into the system | N/A | Transfer to admin home page |

### Recognize Hand Sign Language



**Figure 5:** Recognize Hand Sign Language

**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |
| 1 | Username | Fill user name | No | Yes | Textbox | String | N/A |
| 2 | Password | Fill password | No | Yes | Password | String | N/A |

**Buttons/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 3 | Sign in | Log-in into the system | N/A | Transfer to admin home page |

### Learn Hand Sign Language



**Figure 5:** Learn Hand Sign Language

**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |
| 1 | Username | Fill user name | No | Yes | Textbox | String | N/A |
| 2 | Password | Fill password | No | Yes | Password | String | N/A |

**Buttons/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 3 | Sign in | Log-in into the system | N/A | Transfer to admin home page |

### Select



**Figure 5:** Select

**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |
| 1 | Username | Fill user name | No | Yes | Textbox | String | N/A |
| 2 | Password | Fill password | No | Yes | Password | String | N/A |

**Buttons/Hyperlinks**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 3 | Sign in | Log-in into the system | N/A | Transfer to admin home page |

### Hardware Detailed Description

#### Raspberry Pi B2



**Figure 33. Raspberry Pi B2 Kit**

The Raspberry Pi is a low cost, **credit-card sized computer** that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python. It’s capable of doing everything you’d expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and playing games.

***FOR***

* Quad-core CPU
* Backwards compatible
* More RAM
* Will fit existing cases

***AGAINST***

* Could prove intimidating for Linux newcomers
* No micro-USB adapter included

Raspberry Pi B2 specification:

SoC: Broadcom 2836  
CPU: Quad-core ARM7 800MHz  
GPU: Videocore IV 250MHz  
Memory: 1GB  
GPIO: 40 pin  
Ports: 4x USB 2.0, 100BaseT Ethernet, HDMI and MicroSD card  
Size: 85.60 × 56.5mm (about 3.2 x 2.1-inch)

#### Create a Portable System

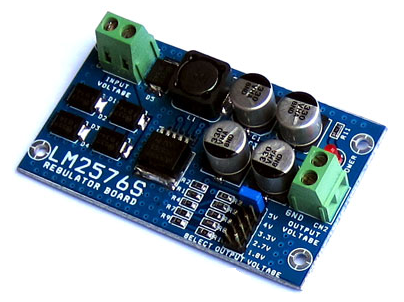
Raspberry use input is 5v and 1A so; we do not use battery AA or AAA for Raspberry. We choose lipo battery use for Raspberry Pi B2 kit because lipo battery supplies 11.1v and 1A.



But Raspberry use input is 5v and 1A so, we need a circuit transformer convert from 11.1v to 5v.

That reason why, we choose LM2576ADJ - 3A UNI REG Board.

**LM2576ADJ - 3A UNI REG Board**



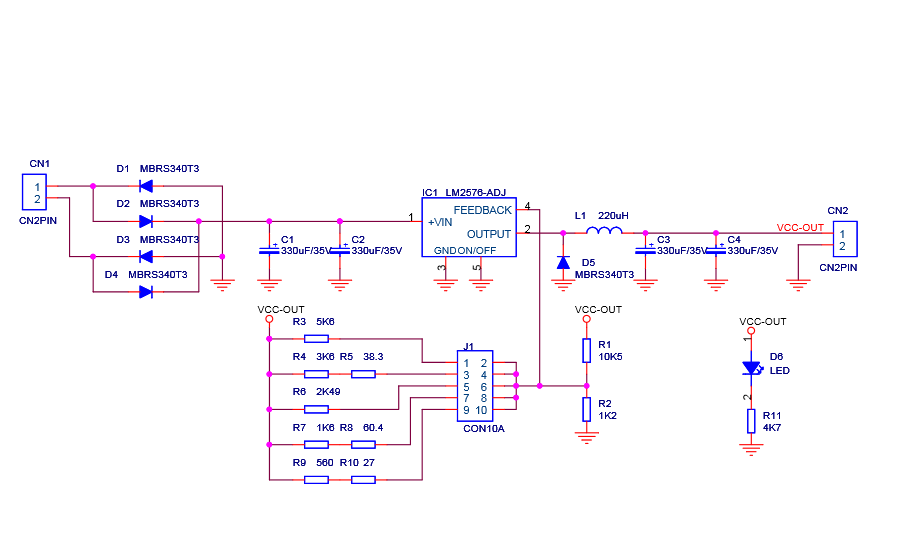
**Overview:**

* UNI-REG board allows changing voltages from 7-23V AC (or 9-32V DC) to 5V, 4V, 3.3V, 2.7V or 1.8V.
* Circuit Board using LM2576 - Step-Down Voltage Regulator.
* On-board screw-terminals are available for easy connection.

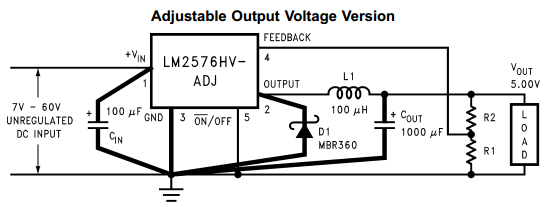
**Description:**

LM2576ADJ - UNI REG Board uses voltage regulator IC provides functional step-down (buck) switching regulator, capable of responding and changing load voltage lines are excellent. This is the ideal motherboard for projects requiring high voltage switching from lower to AC (DC).

Circuit Board accepts 7-23V AC input voltage (9-32V DC or), and stable output 5V, 4V, 3.3V, 2.7V or 1.8V DC, suitable for most electronics projects. The output voltage is selected via a jumper on the board. Compact and affordable, this board is perfect for use when switching power supplies are needed for your embedded project.



IC LM2576HV-ADJ



VOUT = VREF (1+ R2/R1)

R2 = R1(VOUT/VREF  - 1)

Where VREF = 1.23v, R1 between 1k and 5k

#### 4.2.3 The Battery Capacity Display Circuit

**Components of the circuit:**

- Battery lipo

- LT084

- 1 diode zener 5.1v (1N4733)

- 3 resistors 220 ohm

- 2 resistors 3.3k ohm

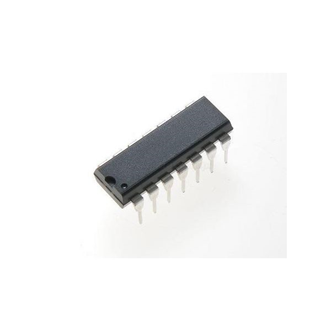
- 2 resistors 2.2k ohm

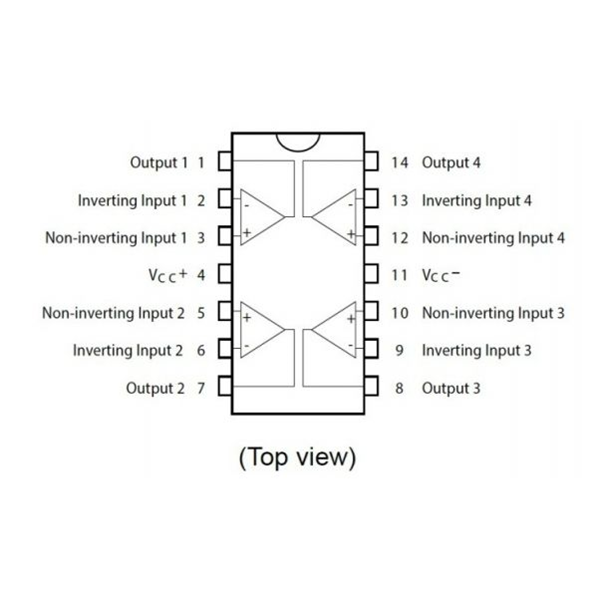
- 5 resistors 1k ohm

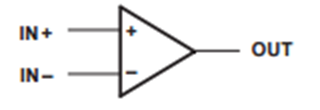
- 4 LED

**a. LT084**

The TL084 JFET-input operational amplifier family is designed to offer a wider selection than any previously developed operational amplifier family. Each of these JFET-input operational amplifiers incorporates well-matched, high-voltage JFET and bipolar transistors in a monolithic integrated circuit. The devices feature high slew rates, low input bias and offset currents, and low offset voltage temperature coefficient.



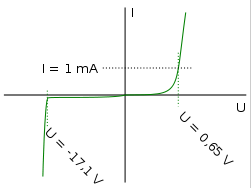




**b. Diode zener**

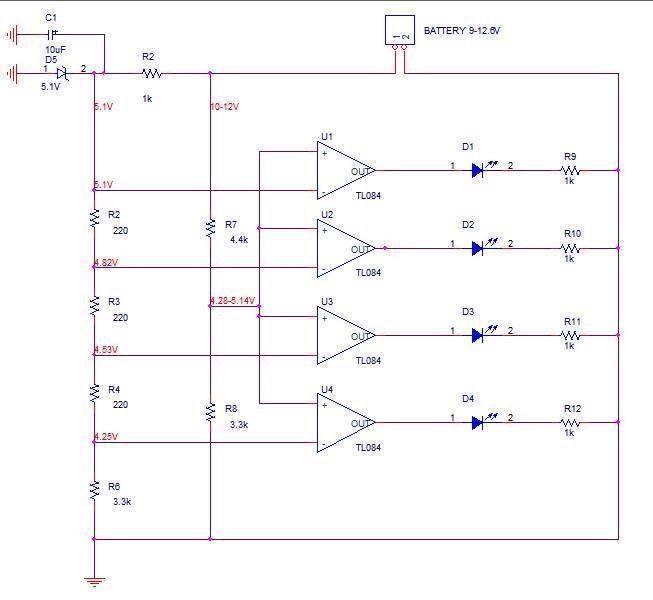
Zener diode, also known as voltage regulator diode, is a semiconductor diode work in reverse polarity mode on the breakdown voltage (breakdown). This voltage is also called Zener voltage avalanche or cascade (avalanche). At that voltage value little changed. It was built so that the reverse polarity, the Zener diodes will pin a fixed voltage level nearly equal to the value indicated on the diode, do Stabilizers of circuit.

When biased diodes Zener diodes operate like normal. When the polarity invert, at first only a small electric current through the diode truth. But if the voltage is increased to a value inversely adaptation: Vnguoc = Vz (Vz: Zener voltage), the current through diodes increase, but the voltage between the two ends of the diodes hardly change, so-called Zener effect.

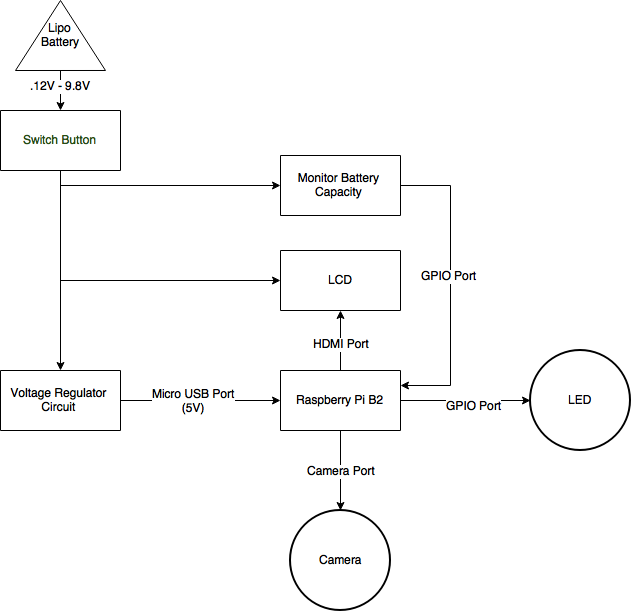


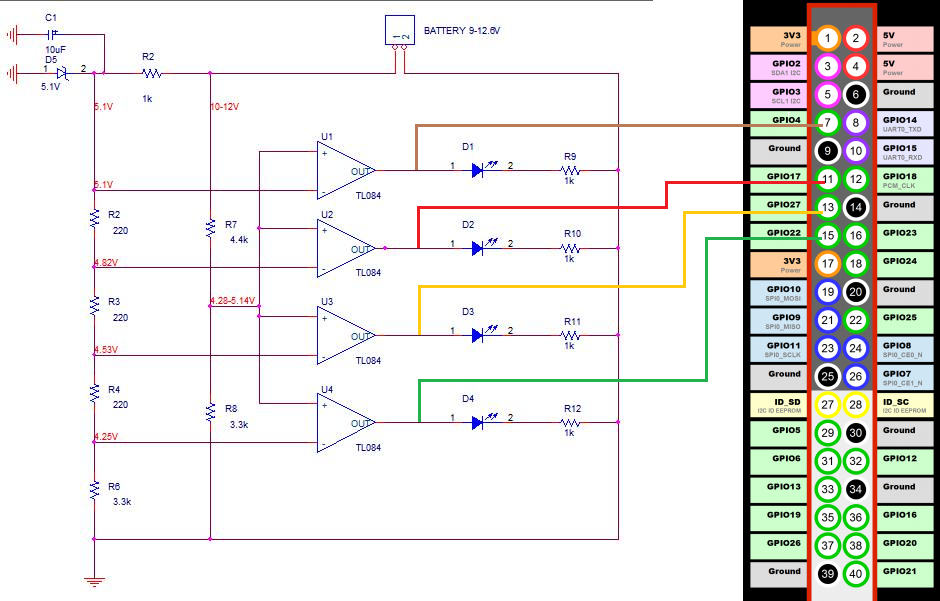


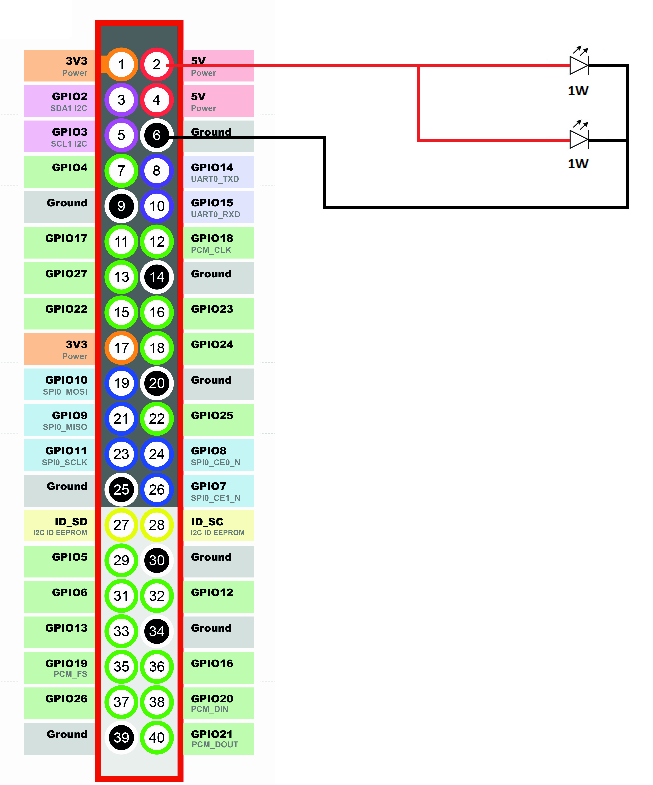
**2. Principles of circuit**



#### Connecting Components In System







## Algorithms

# Report No.5 System Implementation & Test

## Introduction

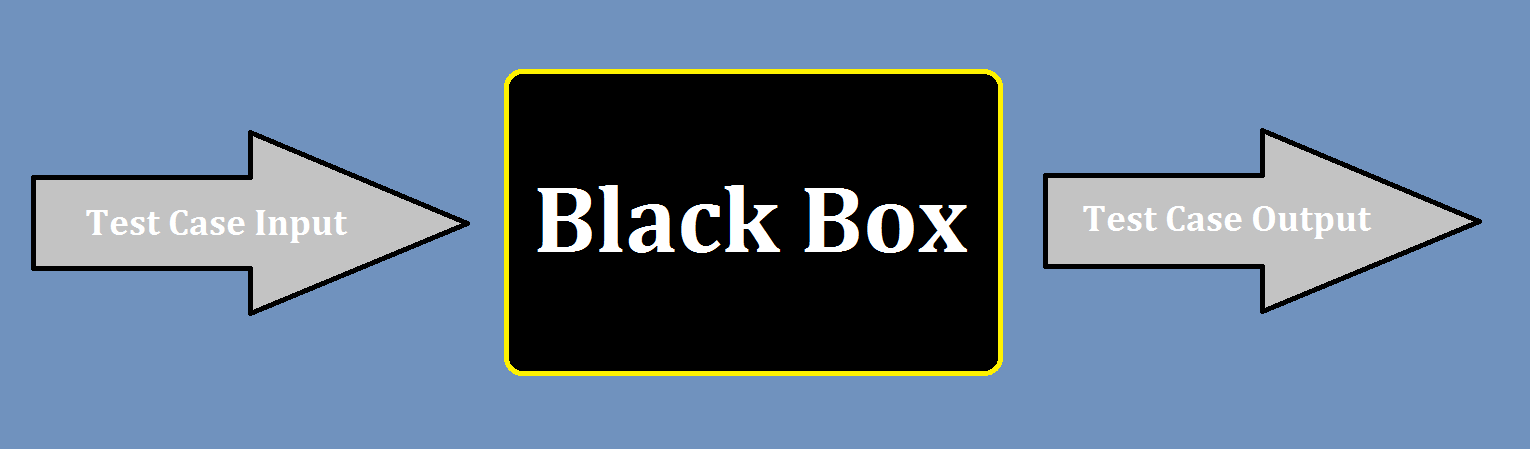
### Overview

This section provides in detail all necessary information about implementation information and testing procedure of VSLR includes test plans, test cases, test procedures and test result.

### Test Approach

#### Method

* *Black-box testing*: We examines the functionality of the system without peering into its internal structures or workings. This testing can dominate integration testing as well.

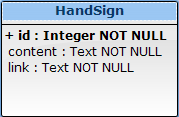


#### Goal

* To validate that the application works as the user will be operating it, then find out incorrect or missing functions, interface errors, behavior and performance errors.

## Database Relationship Diagram

### Physical Diagram



**Figure 52: Physical Database Diagram**

### Data Dictionary

|  |  |
| --- | --- |
| **Entity Data dictionary: describe content of all entities** | |
| **Entity Name** | **Description** |
| HandSign | Describe the hand sign words in the system. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Entity name** | **Attributes** | **Description** | **Domain** | **Null** |
| HandSign | id | Unique identifier of the hand sign. | Integer | No |
| content | The translated content of the handsign | Text | No |
| link | The image link lead to images describing the hand sign. | Text | No |

## Test Plan

### Test items

*We have a main test phase : Integration test:*

* Integration Testing: We test the integration of the code modules developed and interaction with hardware. The integration testing starts at the bottom level. Each component at lower hierarchy is tested individually; then the components that rely upon these are tested.

### Features to be tested

*Integration Test includes the following features:*

* Background Color Subtraction
* “Selecting Function” function
* “Hand Sign Language Recognition” function
* “Learning Hand Sign Language” function
* “Charging Battery” function
* “Monitoring Battery Capacity” function

### Features not to be tested

N/A

### Environmental needs

* A complete system with fully devices and functions.
* An environment with stable light, and background is not complex in color.

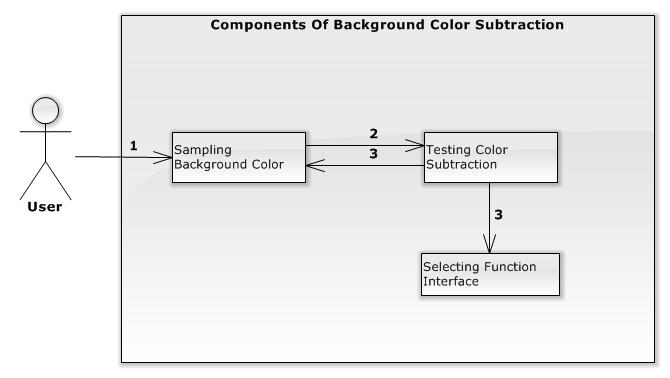
### Test case pass/fail criteria

* Every test case must describe what expected output are to pass that specific test.
* Test coverage must be at least 90%.
* All test case must pass.

## 

## Integration Test Specifications

### “Background Color Subtraction” Test



**Figure 52: Components of the Background Color Subtraction**

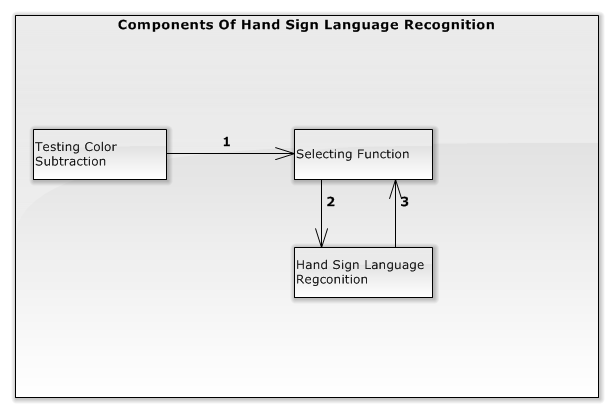
#### Integration test case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Item(s)** | **Input specification** | **Expected Output** | **Condition** |
| BCSTC01 | Display Screen to Sampling Background Color | Switch on | Displays notify “Người dùng vui lòng di chuyển ra khỏi vùng camera đang theo dõi”. | N/A |
| Shows the images captured from camera on the interface for users. |
| Shows the count down time and counts down from 5 by a second. |
| Show the message “Đang tiến hành". |
| BCSTC02 | Display Screen to Sampling Background Color | Switch on | Displays notify dialog “”. | Camera is not active. |
| BCSTC03 | Background Color -> Testing Color Subtraction | The background is fixed and user waiting for the countdown time counts to 0. | A notify “Vui lòng điều chỉnh bàn tay của bạn theo kí hiệu “kiểm tra” trong hướng dẫn” is shown. | Test case BCSTC01 is executed |
| Shows the images subtracting background color on the interface for users |
| Countdown time is shown by seconds from 5. |
| Shows the message “Đang tiến hành" |
| BCSTC04 | Background Color -> Testing Color Subtraction | The background is continuously changing and user waiting for the countdown time counts to 0. | A notify “Vui lòng điều chỉnh bàn tay của bạn theo kí hiệu “kiểm tra” trong hướng dẫn” is shown. | Test case BCSTC01 is executed |
| Shows the images subtracting background color on the interface for users |
| Countdown time is shown by seconds from 5. |
| Shows the message “Đang tiến hành" |
| BCSTC05 | Testing Color Subtraction | Showing right “testing” hand sign inside camera area | Counting down continues. | Test case BCSTC03 or BCSTC04  is executed |
| Show a message “Thành Công”. |
| BCSTC06 | Testing Color Subtraction | Showing wrong “testing” hand sign inside camera area | Counting down continues | Test case BCSTC03 or BCSTC04  is executed |
| Show a message “Đang tiến hành”. |
| BCSTC07 | Testing Color Subtraction | Don’t show hand sign | Counting down continues | Test case BCSTC03 or BCSTC04  is executed |
| Show a message “Đang tiến hành”. |
| BCSTC08 | Testing Color Subtraction | Showing right “testing” hand sign outside camera area | Counting down continues | Test case BCSTC03 or BCSTC04  is executed |
| Show a message “Đang tiến hành”. |
| BCSTC09 | Testing Color Subtraction | Showing wrong “testing” hand sign outside camera area | Counting down continues | Test case BCSTC03 or BCSTC04  is executed |
| Show a message “Đang tiến hành”. |
| BCSTC10 | Testing Color Subtraction -> Sampling Background Color | Waiting for the countdown time counts to 0. | Shows message “Thất bại” | Test case BCSTC06 or BCSTC07 or BCSTC08 or BCSTC09  is executed |
| A notify “Người dùng vui lòng di chuyển ra khỏi vùng camera đang theo dõi” is shown. |
| Show the images captured from camera on the interface for users . |
| Shows the count down time and counts down from 5 by a second |
| Show the message “Đang tiến hành". |
| BCSTC11 | Testing Color Subtraction -> Selecting Function Interface | Waiting for the countdown time counts to 0. | Selecting Function Interface is shown | Test case BCSTC05 is executed |

#### Integration test procedure TP1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Purpose** | **Procedure Steps** | **Excuted By** | **Result** | **Test Date** | **Note** |
| BCSITP01 | Test the success background color subtraction flow can work correctly | 1. Execute test case BCSTC01  2. Execute test case BCSTC03  3. Execute test case BCSTC05  4. Execute test case BCSTC11 |  | Pass |  |  |
| BCSITP02 | Test the fail background color subtraction work flow correctly: show wrong “test” hand sign inside camera area, background is fixed | 1. Execute test case BCSTC01  2. Execute test case BCSTC03  3. Execute test case BCSTC06  4. Execute test case BCSTC10 |  | Pass |  |  |
| BCSITP03 | Test the fail background color subtraction work flow correctly: show right “testing” hand sign inside camera area, background is continuously changing | 1. Execute test case BCSTC01  2. Execute test case BCSTC04  3. Execute test case BCSTC05  4. Execute test case BCSTC10 |  | Pass |  |  |
| BCSITP04 | Test the fail background color subtraction work flow correctly : background is continuously changing, show wrong “test” hand sign inside camera area | 1. Execute test case BCSTC01  2. Execute test case BCSTC04  3. Execute test case BCSTC06  4. Execute test case BCSTC10 |  | Pass |  |  |
| BCSITP05 | Test the fail background color subtraction work flow correctly: don’t show hand sign, background is fixed | 1. Execute test case BCSTC01  2. Execute test case BCSTC03  3. Execute test case BCSTC07  4. Execute test case BCSTC10 |  | Pass |  |  |
| BCSITP06 | Test the fail background color subtraction work flow correctly: don’t show hand sign, background is continuously changing | 1. Execute test case BCSTC01  2. Execute test case BCSTC04  3. Execute test case BCSTC07  4. Execute test case BCSTC10 |  | Pass |  |  |
| BCSITP07 | Test the fail background color subtraction work flow correctly: show right “testing” hand sign outside camera area, background is fixed | 1. Execute test case BCSTC01  2. Execute test case BCSTC03  3. Execute test case BCSTC08  4. Execute test case BCSTC10 |  | Pass |  |  |
| BCSITP08 | Test the fail background color subtraction work flow correctly: show right “testing” hand sign outside camera area, background is continuously changing | 1. Execute test case BCSTC01  2. Execute test case BCSTC04  3. Execute test case BCSTC08  4. Execute test case BCSTC10 |  | Pass |  |  |
| BCSITP09 | Test the fail background color subtraction work flow correctly: show wrong “testing” hand sign outside camera area, background is fixed | 1. Execute test case BCSTC01  2. Execute test case BCSTC03  3. Execute test case BCSTC09  4. Execute test case BCSTC10 |  | Pass |  |  |
| BCSITP10 | Test the fail background color subtraction work flow correctly: show wrong “testing” hand sign outside camera area, background is continuously changing | 1. Execute test case BCSTC01  2. Execute test case BCSTC04  3. Execute test case BCSTC09  4. Execute test case BCSTC10 |  | Pass |  |  |
| BCSITP11 | Test “Background Color Subtraction” can handle when camera not active. | 1. Execute test case BCSTC02 |  | Pass |  |  |

### “Hand Sign Language Recognition” Test



**Figure 52: Components of the Hand Sign Language Recognition**

#### Test Case Specification

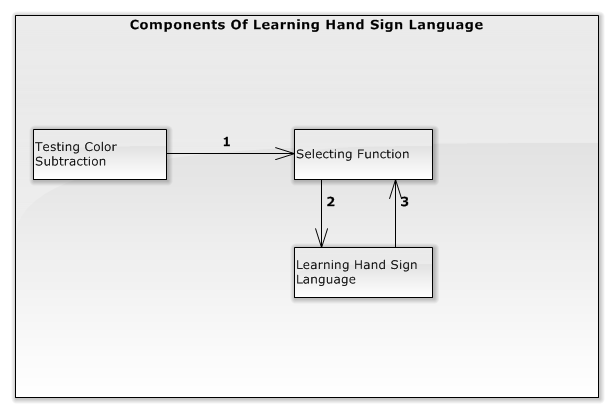
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Item(s)** | **Input Specification** | **Output Specification** | **Condition** |
| HSRTC01 | Testing Color Subtraction -> Selecting Function | N/A | Show a notify “” in the groupbox “Thông Báo” | Background Color Subtraction is executed |
| The analyzed images show on the interface continuously. |
| Two white “Nhận Dạng” and “Học” area are drawn inside analyzed images |
| System shows an image guiding users to select function |
| HSRTC02 | Selecting Function | Show “select” hand sign outside “Nhận dạng” area | Nothing change | Test case HSRTC01 is executed |
| HSRTC03 | Selecting Function | Show hand sign different “select” hand sign inside “Nhận dạng” area | Nothing change | Test case HSRTC01 is executed |
| HSRTC04 | Selecting Function | Show hand sign different “select” hand sign outside “Nhận dạng” area | Nothing change | Test case HSRTC01 is executed |
| HSRTC05 | Selecting Function -> Hand Sign Recognition | Show “select” hand sign inside “Nhận dạng” area. | The system shows the hand sign recognition interface | Test case HSRTC01 is executed |
| The analyzed images show on the interface continuously |
| A notify “Hệ thống sẽ lưu lại kết quả nhận dạng sau 3 giây” is shown in groupbox “Thông Báo” |
| Countdown time is shown from 3 in groupbox “Thời Gian” |
| The system shows two groupbox “Nội dung toàn bộ” and “Kết Quả Hiện Tại” with empty content. |
| HSRTC06 | Hand Sign Language Recognition | Showing the “A” hand sign through camera | Countdown time counts down by second. | Test case HSRTC05 is executed |
| Groupbox “ Kết Quả Hiện Tại ” is shown with the result content “A” below. |
| HSRTC07 | Hand Sign Language Recognition | Don’t show hand sign through camera | Countdown time counts down by second. | Test case HSRTC05 is executed |
| Show a message “Không tìm thấy bàn tay!” in the group “Kết Quả Hiện Tại”. |
| HSRTC08 | Hand Sign Language Recognition | User keeps the “A” hand sign through camera and waiting for the countdown time counts to 0. | Groupbox “ Nội Dung Toàn Bộ ” is shown with the result content “A” | Test case HSRTC06 is executed |
| HSRTC09 | Hand Sign Language Recognition | Showing the “C” hand sign through camera and keeps when the countdown time counts to 0. | Groupbox “ Nội Dung Toàn Bộ ” is shown with the result content “C” | Test case HSRTC06 is executed |
| HSRTC010 | Hand Sign Language Recognition | Showing the “speak” hand sign through camera and keeps when the countdown time counts to 0. | Groupbox “ Nội Dung Toàn Bộ ” is still empty | Test case HSRTC06 is executed |
| HSRTC11 | Hand Sign Language Recognition | Showing the “B” hand sign through camera | Countdown time counts down by second. | Test case HSRTC08 is executed |
| Groupbox “ Kết Quả Hiện Tại ” is shown with the result content “B” |
| HSRTC12 | Hand Sign Language Recognition | User keeps the “B” hand sign through camera and waiting for the countdown time counts to 0. | Groupbox “ Nội Dung Toàn Bộ ” is shown with the result content “AB” | Test case HSRTC10 is executed |
| HSRTC13 | Hand Sign Language Recognition | Show “speak” hand sign inside the camera area and waiting for the countdown time counts to 0. | Reads “A” via LCD speaker | Test case HSRTC08 is executed |
| Groupbox “ Nội Dung Toàn Bộ ” is clear |
| HSRTC14 | Hand Sign Language Recognition | No hand inside the camera area and waiting for the countdown time counts to 0 | Countdown time counts down by second. | Test case HSRTC06 is executed |
| Show a message “Không tìm thấy bàn tay!” in the group “Kết Quả Hiện Tại”. |
| HSRTC15 | Hand Sign Language Recognition | Show “end” hand sign then show “speak” hand sign inside the camera area and waiting for the countdown time counts to 0 | Reads “A” via LCD speaker | Test case HSRTC08 is executed |
| Groupbox “ Nội Dung Toàn Bộ ” is clear |  |
| HSRTC16 | Hand Sign Language Recognition -> Selecting Function | Show “speak” hand sign then show “end” hand sign inside the camera area and waiting for the countdown time counts to 0 | Selecting Function interface is displayed | Test case HSRTC08 is executed |
| HSRTC17 | Hand Sign Language Recognition -> Selecting Function | Showing the “end” hand sign through camera and waiting for the countdown time counts to 0. | Selecting Function interface is displayed | Test case HSRTC08 is executed |

#### Integration Test Procedure TP3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Purpose** | **Procedure Steps** | **Excuted By** | **Result** | **Test Date** | **Note** |
| HSRTP01 | Test fail “Selection Function”: “select” hand sign outside “Nhận dạng” area | 1. Execute test case HSRTC01  2. Execute test case HSRTC02 |  | **Pass** |  |  |
| HSRTP02 | Test fail “Selection Function”: hand sign different “select” hand sign inside “Nhận dạng” area | 1. Execute test case HSRTC01  2. Execute test case HSRTC03 |  | **Pass** |  |  |
| HSRTP03 | Test fail “Selection Function”: different “select” hand sign outside “Nhận dạng” area | 1. Execute test case HSRTC01  2. Execute test case HSRTC04 |  | **Pass** |  |  |
| HSRTP04 | Test the flow Hand Sign Recognition can work correctly show the “AB” content in groupbox “ Nội Dung Toàn Bộ ” | 1. Execute test case HSRTC01  2. Execute test case HSRTC05  3. Execute test case HSRTC06  4. Execute test case HSRTC08  5. Execute test case HSRTC12 |  | **Pass** |  |  |
| HSRTP05 | Test the flow Hand Sign Recognition can work correctly show and speak the “A” content | 1. Execute test case HSRTC01  2. Execute test case HSRTC05  3. Execute test case HSRTC06  4. Execute test case HSRTC08  5. Execute test case HSRTC13 |  | **Pass** |  |  |
| HSRTP06 | Test the flow Hand Sign Recognition can work correctly show and speak the “A” content | 1. Execute test case HSRTC01  2. Execute test case HSRTC05  3. Execute test case HSRTC06  4. Execute test case HSRTC08  5. Execute test case HSRTC15 |  | **Pass** |  |  |
| HSRTP07 | Test the flow Hand Sign Recognition can work correctly show the “C” content in groupbox “ Nội Dung Toàn Bộ ” | 1. Execute test case HSRTC01  2. Execute test case HSRTC05  3. Execute test case HSRTC06  4. Execute test case HSRTC09 |  | **Pass** |  |  |
| HSRTP08 | Test the flow Hand Sign Recognition can work correctly : show Selecting Function interface | 1. Execute test case HSRTC01  2. Execute test case HSRTC05  3. Execute test case HSRTC17 |  | **Pass** |  |  |
| HSRTP09 | Test the flow Hand Sign Recognition can work correctly : show Selecting Function interface | 1. Execute test case HSRTC01  2. Execute test case HSRTC05  3. Execute test case HSRTC16 |  | **Pass** |  |  |
| HSRTP10 | Test the flow Hand Sign Recognition can work correctly : show Selecting Function interface | 1. Execute test case HSRTC01  2. Execute test case HSRTC05  3. Execute test case HSRTC06  4. Execute test case HSRTC17 |  | **Pass** |  |  |
| HSRTP11 | Test the flow Hand Sign Recognition can work correctly : show Selecting Function interface | 1. Execute test case HSRTC01  2. Execute test case HSRTC05  3. Execute test case HSRTC06  4. Execute test case HSRTC16 |  | **Pass** |  |  |
| HSRTP12 | Test fail “Hand Sign Recognition” with No hand inside the camera area | 1. Execute test case HSRTC01  2. Execute test case HSRTC05  3. Execute test case HSRTC06  4. Execute test case HSRTC14 |  | **Pass** |  |  |
| HSRTP13 | Test fail “Hand Sign Recognition” with No hand inside the camera area | 1. Execute test case HSRTC01  2. Execute test case HSRTC05  3. Execute test case HSRTC07 |  | **Pass** |  |  |

### 

### “Learning Hand Sign Language” Test



**Figure 52: Components of the Learning Hand Sign Language**

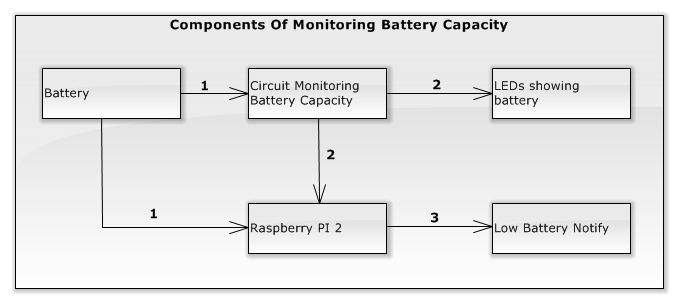
#### Test case specification

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Item(s)** | **Input Specification** | **Output Specification** | **Condition** |
| LHSTC01 | Testing Color Subtraction -> Selecting Function | N/A | Show a notify “” in the groupbox “Thông Báo” | Background Color Subtraction is executed |
| The analyzed images show on the interface continuously. |
| Two white “Nhận Dạng” and “Học” area are drawn inside analyzed images |
| System shows an image guiding users to select function |
| LHSTC02 | Selecting Function | Show “select” hand sign outside “ Học ” area | Nothing change | Test case LHSTC01is executed |
| LHSTC03 | Selecting Function | Show hand sign different “select” hand sign inside “ Học ” area | Nothing change | Test case LHSTC01is executed |
| LHSTC04 | Selecting Function | Show hand sign different “select” hand sign outside “ Học ” area | Nothing change | Test case LHSTC01is executed |
| LHSTC05 | Selecting Function -> Learning Hand Sign Language | N/A | Show the list of words on the interface in the group box “Hướng Dẫn” | Test case LHSTC04 is executed |
| The analyzed images show on the interface continuously |
| The “A” word is selected first |
| Image describing “A” hand sign is displayed |
| A notify “Hãy đưa kí hiệu trong hướng dẫn vào vùng mũi tên lên xuống để thay đổi từ được chọn ” is shown in the groupbox “Thông Báo” |
| Two white “Lên” and “Xuống” area were drawn on these images showing on the interface. |
| LHSTC06 | Learning Hand Sign Language | Move the “select” hand sign into the “Lên” square area. | The “A” word in the list is still selected | Test case LHSTC05 is executed |
| Image describing “A” hand sign is still displayed |
| Group box “Kết Quả Hiện Tại” with no result content below |
| LHSTC07 | Learning Hand Sign Language | Move the “select” hand sign into the “Xuống” square area. | The “B” word in the list is selected | Test case LHSTC05 or LHSTC06 is executed |
| Image describing “B” hand sign is displayed |
| Group box “Kết Quả Hiện Tại” with no result content below |
| LHSTC08 | Learning Hand Sign Language | Move the “select” hand sign outside the “Lên” and “Xuống” square area. | The “B” word in the list is still selected | Test case LHSTC07 is executed |
| Image describing “B” hand sign is still displayed |
| Group box “Kết Quả Hiện Tại” with result “N” content below |
| LHSTC09 | Learning Hand Sign Language | Show “E” hand sign outside the “Lên” and “Xuống” square area. | The “B” word in the list is still selected | Test case LHSTC07 is executed |
| Image describing “B” hand sign is still displayed |
| Group box “Kết Quả Hiện Tại” with result “E” content below |
| LHSTC10 | Learning Hand Sign Language | Show “E” hand sign inside the “Lên” square area. | The “B” word in the list is still selected | Test case LHSTC07 is executed |
| Image describing “B” hand sign is still displayed |
| Group box “Kết Quả Hiện Tại” with result “E” content below |
| LHSTC11 | Learning Hand Sign Language | Show “E” hand sign inside the “Xuống” square area. | The images containing only the hand on the The “B” word in the list is still selected | Test case LHSTC07 is executed |
| Image describing “B” hand sign is still displayed |
| Group box “Kết Quả Hiện Tại” with result “E” content below |
| LHSTC12 | Learning Hand Sign Language | Move the “select” hand sign into the “Lên” square area. | The “A” word in the list is selected | Test case LHSTC07 is executed |
| Image describing “A” hand sign is displayed |
| Group box “Kết Quả Hiện Tại” is empty |
| LHSTC13 | Learning Hand Sign Language | Showing the “end” hand sign outside of the two square areas “Lên ” and “Xuống” | Selecting Function interface is displayed | Test case LHSTC05 is executed |
| LHSTC14 | Learning Hand Sign Language -> Selecting Function | Showing the “end” hand sign inside of the two square areas “Lên” | Selecting Function interface is displayed | Test case LHSTC05 is executed |
| LHSTC15 | Learning Hand Sign Language -> Selecting Function | Showing the “end” hand sign inside of the two square areas “Xuống” | Selecting Function interface is displayed | Test case LHSTC05 is executed |
| LHSTC16 | Learning Hand Sign Language | Don’t show hand inside the camera area | Shows a message “Không tìm thấy bàn tay!” in the group box “Kết Quả Hiện Tại” |  |

#### Integration Test Procedure TP4

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Purpose** | **Procedure Steps** | **Executed By** | **Result** | **Test Date** | **Note** |
| LHSTP01 | Test fail “Selection Function”: “select” hand sign outside “Học” area | 1. Execute test case LHSTC01  2. Execute test case LHSTC02 |  | **Pass** |  |  |
| LHSTP02 | Test fail “Selection Function”: hand sign different “select” hand sign inside “Nhận dạng” area | 1. Execute test case LHSTC01  2. Execute test case LHSTC03 |  | **Pass** |  |  |
| LHSTP03 | Test fail “Selection Function”: different “select” hand sign outside “Nhận dạng” area | 1. Execute test case LHSTC01  2. Execute test case LHSTC04 |  | **Pass** |  |  |
| LHSTP04 | Test success Learning Hand Sign Language: select “B” word below the “A” word | 1. Execute test case LHSTC01  2. Execute test case LHSTC05  3. Execute test case LHSTC07 |  | **Pass** |  |  |
| LHSTP05 | Test success Learning Hand Sign Language: select “A” word above the “B” word | 1. Execute test case LHSTC01  2. Execute test case LHSTC05  3. Execute test case LHSTC07  4. Execute test case LHSTC12 |  | **Pass** |  |  |
| LHSTP06 | Test success Learning Hand Sign Language: select “Lên” when  “A” word is selected in the top | 1. Execute test case LHSTC01  2. Execute test case LHSTC05  3. Execute test case LHSTC06 |  | **Pass** |  |  |
| LHSTP07 | Test success Learning Hand Sign Language: checking “E” hand sign with result “E” | 1. Execute test case LHSTC01  2. Execute test case LHSTC05  3. Execute test case LHSTC07  4. Execute test case LHSTC09 |  | **Pass** |  |  |
| LHSTP08 | Test success Learning Hand Sign Language: checking “E” hand sign with result “E” | 1. Execute test case LHSTC01  2. Execute test case LHSTC05  3. Execute test case LHSTC07  4. Execute test case LHSTC10 |  | **Pass** |  |  |
| LHSTP10 | Test success Learning Hand Sign Language: checking “E” hand sign with result “E” | 1. Execute test case LHSTC01  2. Execute test case LHSTC05  3. Execute test case LHSTC07  4. Execute test case LHSTC11 |  | **Pass** |  |  |
| LHSTP11 | Test success finish Learning Hand Sign Language | 1. Execute test case LHSTC01  2. Execute test case LHSTC05  3. Execute test case LHSTC13 |  | **Pass** |  |  |
| LHSTP12 | Test success finish Learning Hand Sign Language | 1. Execute test case LHSTC01  2. Execute test case LHSTC05  3. Execute test case LHSTC07  4. Execute test case LHSTC15 |  | **Pass** |  |  |
| LHSTP13 | Test success finish Learning Hand Sign Language | 1. Execute test case LHSTC01  2. Execute test case LHSTC05  3. Execute test case LHSTC07  4. Execute test case LHSTC14 |  | **Pass** |  |  |
| LHSTP14 | Test fail Learning Hand Sign Language | 1. Execute test case LHSTC01  2. Execute test case LHSTC05  3. Execute test case LHSTC16 |  | **Pass** |  |  |

### “Monitor Battery Capacity” Test



**Figure 52: Components of the Monitoring Battery Capacity**

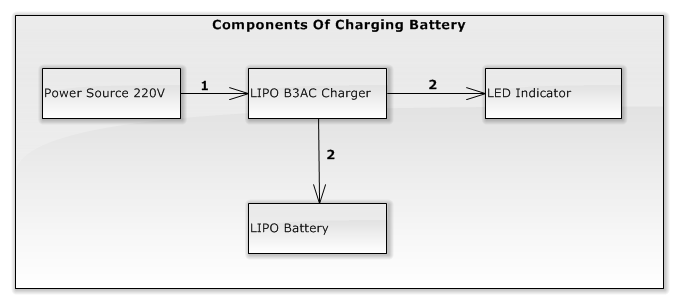
#### Integration test case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Item(s)** | **Input Specification** | **Output Specification** | **Condition** |
| MBCTC01 | Battery -> Battery Capacity Display Circuit -> LEDs showing battery | Switch on | LED on circuit is bright. | The battery voltage > 9.8V |
| MBCTC02 | Battery -> Battery Capacity Display Circuit -> LEDs showing battery | The battery voltage is higher 12V | 4 LEDs on circuit is bright. | Test case MBCTC01 is exectuted |
| MBCTC03 | Battery -> Battery Capacity Display Circuit -> LEDs showing battery | The battery voltage is between 11.3V and 12V | 3 LEDs on circuit is bright. | Test case MBCTC01 is exectuted |
| MBCTC04 | Battery -> Battery Capacity Display Circuit -> LEDs showing battery | The battery voltage is between 11.3V and 10.8V | 2 LEDs on circuit is bright. | Test case MBCTC01 is exectuted |
| MBCTC05 | Battery -> Battery Capacity Display Circuit -> LEDs showing battery | The battery voltage is between 10.8V and 9.9V | 1 LEDs on circuit is bright. | Test case MBCTC01 is exectuted |
| MBCTC06 | Battery -> Battery Capacity Display Circuit -> LEDs showing battery | The battery voltage is lower 9.9V | 4 LEDs on circuit is off. | Test case MBCTC01 is exectuted |
| MBCTC07 | Battery -> Battery Capacity Display Circuit -> Raspbbery PI 2 | N/A | LED on Raspberry is bright. | Test case MBCTC01 is executed |
| MBCTC08 | Raspbbery PI 2 -> Low Battery Notify | The battery voltage is between 9.8V and 10.5V | Low Battery Notify “Bin yếu vui lòng tắt hệ thống và cắm sạc. Thông báo sẽ được tự động tắt.” is shown on the system interfaces. | Test case MBCTC02 is executed |
| Countdown time is shown in Low Battery Notify by seconds from 3. |
| MBCTC09 | Raspbbery PI 2 -> Low Battery Notify | The battery voltage is between ..V and ..V | Low Battery Notify is not displayed. |  |
| MBCTC10 | Raspbbery PI 2 -> Low Battery Notify | Waiting for the countdown time counts to 0. | Low Battery Notify is hide. | Test case MBCTC03 is executed |

#### Integration test procedure

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Purpose** | **Procedure Steps** | **Executed By** | **Result** | **Test Date** | **Note** |
| MBCTP01 | Test the flow Monitoring Battery Capacity works correctly with low battery. | 1. Execute test case MBCTC01  2. Execute test case MBCTC07  3. Execute test case MBCTC05  4. Execute test case MBCTC08  5. Execute test case MBCTC10 |  | **Pass** |  |  |
| MBCTP02 | Test the flow Monitoring Battery Capacity works correctly with normal battery. | 1. Execute test case MBCTC01  2. Execute test case MBCTC07  3. Execute test case MBCTC04  4. Execute test case MBCTC09  5. Execute test case MBCTC10 |  | **Pass** |  |  |
| MBCTP01 | Test the displaying battery capacity on LEDs. | 1. Execute test case MBCTC01  2. Execute test case MBCTC02 |  | **Pass** |  |  |
| MBCTP01 | Test the displaying battery capacity on LEDS. | 1. Execute test case MBCTC01  2. Execute test case MBCTC03 |  | **Pass** |  |  |
| MBCTP01 | Test the displaying battery capacity on LEDs. | 1. Execute test case MBCTC01  2. Execute test case MBCTC04 |  | **Pass** |  |  |

### 4.5 “Charging Battery” Test



**Figure 52: Components of the charging battery**

#### 4.5.1 Integration test case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Test Item(s)** | **Input Specification** | **Output Specification** | **Condition** |
| CBTC01 | 220V power source -> LIPO B3AC charger -> Led Indicator | Connect LIPO B3AC charger to power source | The charger’s LEDs indicator is bright with red color. | The system is OFF. |
| CBTC02 | LIPO B3AC charger -> LIPO Battery | Connect LIPO B3AC charger to LIPO Battery has the voltage lower 9.8V. | The charger’s LEDs indicator is bright with red color. | Test case CBTC01 is executed. |
| CBTC03 | LIPO B3AC charger -> LIPO Battery | Waiting for about ?? minutes. | One Charger’s LED is bright with green color. | Test case CBTC02 is executed. |
| CBTC04 | LIPO B3AC charger -> LIPO Battery | Waiting for about ?? minutes. | Two Charger’s LEDs are bright with green color. | Test case CBTC02 is executed. |
| CBTC05 | LIPO B3AC charger -> LIPO Battery | Waiting for about ?? minutes. | Three Charger’s LEDs are bright with green color. | Test case CBTC02 is executed. |

#### Integration test procedure

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Purpose** | **Procedure Steps** | **Excuted By** | **Result** | **Test Date** | **Note** |
| CBIPT01 | Test the charging battery flow can succeed and work stability. | 1. Execute test case CBTC01  2. Execute test case CBTC02  3. Execute test case CBTC03 |  | Pass |  |  |
| CBIPT02 | Test the charging battery flow can succeed and work stability. | 1. Execute test case CBTC01  2. Execute test case CBTC02  3. Execute test case CBTC04 |  | Pass |  |  |
| CBIPT03 | Test the charging battery flow can succeed and work stability. | 1. Execute test case CBTC01  2. Execute test case CBTC02  3. Execute test case CBTC05 |  | Pass |  |  |