MINISTRY OF EDUCATION AND TRAINING

FPT UNIVERSITY

Capstone Project Document

The Roll System using Mobile Device

Group 1	
Group member	Pham Huy Hoang – Team Leader – SE60740 Nguyen Thanh Binh – Team Member - 60321 Nguyen Quoc Huy – Team Member - 60551 Do Minh Dat – Team Member - 60545
Supervisor	Mr. Kieu Trong Khanh
Ext. Supervisor	N/A
Capstone Project code	RSM

-Ho Chi Minh City, 09/2013-

This page is intentionally left blank

ACKNOWLEDGEMENTS

We wish to thank various people for their contribution to this project: Our teachers for their advices and participant in the final review, our friend for the valuable technical support.

Special thanks should be given to Mr.Kiều Trọng Khánh, our research supervisor for his professional guidance and the useful, constructive recommendations throughout the course of this project.

Table of Contents

Table of Contents 3
List of Tables 6
List of Figures 8
Definitions, Acronyms, and Abbreviations 12
A. Introduction 13
1. Project Information 13
2. Introduction 13
3. Current Roll Call System 13
4. Problem Definition 13
5. Proposed Solution 14
5.1 Feature functions 14
5.2 Advantage and disadvantage 14
6. Functional Requirement 15
6.1 Attendance Checking 15
6.2 Attendance Report 15
6.3 Roll Call Management 15
6.4 Calendar Syncing 15
6.5 Student Management 15
6.6 Create Account 16
6.7 Instructor Management 16
6.8 Class Management 16
6.9 Subject Management 16
6.10 Semester Management 16
7. Role and Responsibility 16
B. Software Project Management Plan 17
1. Problem Definition 17
1.1 Name of this Capstone Project 17
1.2 Problem Abstract 17
1.3 Project Overview 17
2. Project organization 19
2.1 Software Process Model 19
2.2 Roles and responsibilities 20
2.3 Tools and Techniques 21
3. Project Management Plan 21
3.1 Iteration 22
3.2 Iteration Detail 23
3.3 All Meeting Minutes 27
4. Coding Convention 27
C. Software Requirement Specification 29
1. User Requirement Specification 29

1.1 Instructor Requirement 29
1.2 Admin Requirement 29
1.3 Staff Requirement 29
1.4 System Requirement 29
1.5 Student Requirement 29
1.6 Guest Requirement 29
2. System Requirement Specification 30
2.1 External Interface Requirement 30
2.2 System Overview Use Case 31
2.3 List of Use Case 31
3. Software System Attribute 114
3.1 Usability 114
3.2 Reliability 114
3.3 Availability 114
3.4 Security 114
3.5 Maintainability 114
3.6 Portability 114
3.7 Performance 114
4. Conceptual Diagram 115
Data Dictionary 116
D. Software Design Description 117
1. Design Overview 117
2. System Architectural Design 118
3. Component Diagram 119
4. Detailed Description of Components 120
4.1 Class Diagram 120
4.2 Class Diagram Explanation 121
4.3 Sequence Diagram 129
5. User Interface Design 141
5.1 Instructor Interface Design 141
5.2 Admin Interface Design 147
5.3 Staff Interface Design 151
5.4 Mobile Interface Design 155
6. Database Design 161
6.1 Entity Relationship Diagram 161
6.2 Data Dictionary 162
7. Algorithms 166
7.1 Face Detection 166
7.2 Face Recognition 169
E. System Implementation & Test 172
1. Introduction 172
1.1 System Overview 172

1.2 Test Approach 172
2. Database Relationship Diagram 173
3. Performance measures 174
3.1 Face Detection Performance 174
3.2 Face recognition performance 174
4. Test Plan 176
4.1 Features to be tested 176
4.2 Features not to be tested 176
5. System Testing Test Case 177
6. Module Test Case 180
6.1 Manage RollCall 180
6.2 Manage Student Image 182
6.3 Check Attendance 185
6.4 Attendance Report 188
6.5 Manage Teaching Calendar 189
6.6 Manage Study Session 190
6.7 Check own Attendance Rate 192
F. Software User's Manual 194
1. Installation Guide 194
1.1 Setting up environment at server site 194
1.2 Deployment at server side 194
1.3 Setting up the environment at client side 200
2. User Guide 202
2.1 Login 202
2.2 Admin – Add Image for student 202
2.3 Staff – Create Roll Call 203
2.4 Instructor – Check Attendance (Mobile) 207
2.5 Instructor – Check Report 210
3. User Notice 212
3.1 Image Data Notice 212
3.2 Position & Angle Notice 213
3.3 Other Notice 214
G. Appendix 215

List of Tables

Table 1: Roles and Responsibility 16
Table 2: Hardware Requirement for Server 19
Table 3: Hardware Requirement for Mobile App 19
Table 4: Roles and Responsibility Details 21
Table 5: Iteration 23
Table 6: Phase 1: Preliminary Investigation or Analysis 23
Table 7: Phase 2: Face Detect & Recognize 24
Table 8: Phase 3: Student Management 25
Table 9: Phase 4: Roll Call Management 26
Table 10: Phase 5: Web Service Implement 26
Table 11: Phase 6: Attendance Checking 27
Table 12: Phase 7: Study Session Management 27
Table 13: <Admin> Add Image for Single Student 35
Table 14: <Admin> Add Image for Multi Students 37
Table 15: <Admin> Remove Student Images 39
Table 16: <Admin>Get Image From Log Image 41
Table 17: <Admin>Create Request 42
Table 18: <Admin>Accept Request 44
Table 19: <Admin>Deny Request 46
Table 20: <Admin> Create Account 48
Table 21: <Admin> Active Account 49
Table 22: <Admin> Inactive Account 51
Table 23: <Admin> Config System 52
Table 24: <System> Face Detection 54
Table 25: <System> Recognize Student 56
Table 26: <System> Auto Free Storage Space 57
Table 27: <System> Auto Active Roll Call 58
Table 28: <System> Auto Inactive Roll Call 59
Table 29: <Staff>Report Roll Call Attendance Rate 61
Table 30: <Staff>Edit Roll Call 73
Table 31: <Staff> Add Subject 75
Table 32: <Staff> Edit Subject 77
Table 33: <Staff> Active Subject 79
Table 34: <Staff> Inactive Subject 81
Table 35: <Staff> Add Class 91
Table 36<Staff> Edit Class 93
Table 37: <Instructor> Check Attendance by Taking Picture 96
Table 38: <Instructor> Check Attendance Manually 98
Table 39: <Instructor> View Roll Call List 100
Table 40: <Instructor> Report Attendance by Class 102

Table 41: <Student> CheckPresent Rate 104
Table 42: <Student> Check Present Rate By Learned Course 106
Table 43: <Student> Check Present Rate by Learning Course 107
Table 44: <Student>Upload Images as Request 108
Table 45: <Guest> Login 110
Table 46: < Authorized User> Change Password 112
Table 47: <Authorized User> Logout 113
Table 48: Entity Data Dictionary 162
Table 49: Attribute Data Dictionary 165
Table 50: Core Workflow Test Case 180
Table 51: Manage Roll Call Test Case 182
Table 52: Manage Student Image Test Case 185
Table 53: Check Attendance Test Case 188
Table 54: Attendance Report Test Case 189
Table 55: Manage Teaching Calendar Test Case 190
Table 56: Manage Study Session Test Case 192
Table 57: Check own Attendance Rate Test Case 193
Table 58: Login Step 202
Table 59: View Student Image Step 202
Table 60: Upload Image Step 203
Table 61: Select Student Image Step 203
Table 62: View Roll Call List Step 204
Table 63: Create Roll Call Step 205
Table 64: Change Student List Step 206
Table 65: Change Schedule Step 206
Table 66: Login Step 207
Table 67: View Roll Call List Step 207
Table 68: Check Attendance Step 208
Table 69: View Roll Call Detail Step 208
Table 70: Take Attendance Manual Step 209
Table 71: View Teaching Roll Call Step 210
Table 72: View Roll Call Report Step 210

List of Figures

Figure 1: Agile Development Model 20
Figure 2: System Overview Use Case 31
Figure 3:<Admin> Overview Use Case 32
Figure 4: <Admin> Add Image for Single Student 32
Figure 5: <Admin> Add Image for Multi Student 35
Figure 6: <Admin> Remove Student Image 37
Figure 7: <Admin>Get Image From Log Images 39
Figure 8: <Admin>Create Request 41
Figure 9: <Admin>Accept Request 43
Figure 10: <Admin>Deny Request 44
Figure 11: <Admin> Create Account 46
Figure 12: <Admin> Active Account 48
Figure 13: <Admin> Inactive Account 50
Figure 14: <Admin> Config System 51
Figure 15: <System> Overview Use Case 53
Figure 16: <System> Face Detection 53
Figure 17: <System> Recognize Student 55
Figure 18: <System> Auto Free Storage Space 56
Figure 19: <System> Auto Active Roll Call 57
Figure 20: <System> Auto Inactive Roll Call 58
Figure 21: <Staff> Overview Use Case 59
Figure 22: <Staff> Report Roll Call Attendance Rate 60
Figure 23: <Staff> Report Student List for Final Exam 61
Figure 24: <Staff>Add Roll Call 63
Figure 25: <Staff>Add Session 65
Figure 26: <Staff>Change Session Time 67
Figure 27: <Staff>Change Session Instructor 69
Figure 28: <Staff>Edit Roll Call 72
Figure 29: <Staff> Add Subject 74
Figure 30: <Staff> Edit Subject 76
Figure 31: <Staff>Add Instructor 86
Figure 32: <Staff>Edit Instructor 88
Figure 33: <Staff> Add Class 90
Figure 34: <Staff> Edit Class 92
Figure 35: <Instructor> Overview Use Case 93
Figure 36: <Instructor>Capture and Show result attendance 94
Figure 37: <Instructor> Check Attendance Manually 96
Figure 38: <Instructor> View Roll Call Detail 99
Figure 39: <Instructor> Report Attendance by Class 101
Figure 40: <Student> Overview Use Case 102

Figure 41: <Student>CheckPresentPresent Rate 103
Figure 42: <Student>Check Present Rate by Learned Course 104
Figure 43: <Student>Check Present Rate by Learning Course 106
Figure 44: <Student> Upload Image as Request 107
Figure 45: <Guest> Overview Use Case 109
Figure 46: <Guest> Login 109
Figure 47: < Authorized User> Overview Use Case 110
Figure 48: < Authorized User> Change Password 110
Figure 49: < Authorized User> Logout 112
Figure 50: Conceptual Diagram 115
Figure 51: Multi-tier Architecture (N-tier Architecture) 118
Figure 52: Component Diagram 119
Figure 53: Class Diagram 120
Figure 54: Add Image for Singe Student Sequence Diagram 129
Figure 55: Add Image for Many Students Sequence Diagram 129
Figure 56: Delete Student Image Sequence Diagram 130
Figure 57: Create Account Sequence Diagram 130
Figure 58: Configure System Sequence Diagram 131
Figure 59: Face Detection Sequence Diagram 131
Figure 60: Face Recognition Sequence Diagram 132
Figure 61: Auto Free Storage Space Sequence Diagram 132
Figure 62: Auto Active Roll Call Sequence Diagram 133
Figure 63: Add Roll Call Sequence Diagram 133
Figure 64: Edit Roll Call Sequence Diagram 134
Figure 65: Import Student List Sequence Diagram 134
Figure 66: Export Report Sequence Diagram 135
Figure 67: Add Student Sequence Diagram 135
Figure 68: Add Class Sequence Diagram 136
Figure 69:Edit Class Sequence Diagram 136
Figure 70: Add Subject Sequence Diagram 137
Figure 71: Edit Subject Sequence Diagram 137
Figure 72: Take Attendance Auto Sequence Diagram 138
Figure 73: Take Attendance Manual Sequence Diagram 138
Figure 74: Take Attendance Offline & Sync Data Sequence Diagram 139
Figure 75: View Roll Call Detail Sequence Diagram 139
Figure 76: Report Attendance by Class Sequence Diagram 140
Figure 77: Check Present Rate Sequence Diagram 140
Figure 78: Instructor Home Page 141
Figure 79: Instructor – Teaching Calendar 142
Figure 80: Instructor – Present Roll Call 143
Figure 81: Instructor – Roll Call Detail 144
Figure 82: Instructor – Check Attendance Manual 146

Figure 83: Admin Home Page 147
Figure 84: Admin – Student List 148
Figure 85: Admin – Single Student Info 149
Figure 86: Admin – Select Images for Students 150
Figure 87: Staff - Roll Call List 151
Figure 88: Staff - Create Roll Call 153
Figure 89: Staff - Check Attendance 154
Figure 90: Login Activity 155
Figure 91: Course Activity 156
Figure 92: Camera Activity 157
Figure 93: Roll Call Activity 158
Figure 94: Check Attendance Activity 159
Figure 95: Entity Relationship Diagram 161
Figure 96: Example of Haar Future 166
Figure 97: Apply Haar future to sub-window 167
Figure 98: Classifier Cascade 167
Figure 99: Viola-Jones method flowchart 168
Figure 100: Face Recognition Process 169
Figure 101: Face Recognition Algorithm Flowchart 171
Figure 102: Database Relationship Diagram 173
Figure 103: Face Recognizer Algorithm Accuracy 174
Figure 104: Number of images required 175
Figure 105: Full Recognize Percent 175
Figure 106: Core Workflow 177
Figure 107: Prepare deployment package 195
Figure 108: Deploy database – 1 195
Figure 109: Deploy database - 2 196
Figure 110: Deploy database - 3 196
Figure 111: Configure Server before deploy – 1 196
Figure 112: Configure Server before deploy – 2 197
Figure 113: Configure Server before deploy - 3 197
Figure 114: Deploy web application on server - 1 198
Figure 115: Deploy web application on server – 2 198
Figure 116: Deploy web application on server – 3 198
Figure 117: Deploy web application on server – 4 199
Figure 118: Deploy web application on server – 5 199
Figure 119: Change Settings 200
Figure 120: Download app 201
Figure 121: Install App 201
Figure 122: Login Page 202
Figure 123: Student List Page 202
Figure 124: Upload Image Page 203

Figure 125: Select Student Image Page 203
Figure 126: Roll Call List Page 204
Figure 127: Create Roll Call Page 205
Figure 128: Roll Call Student List Page 206
Figure 129: Change Schedule Page 206
Figure 130: Login Screen 207
Figure 131: Roll Call List Screen 207
Figure 132: Camera Screen 208
Figure 133: Roll Call Detail Screen 208
Figure 134: Take Attendance Screen 209
Figure 135: Instructor Home Page 210
Figure 136: Roll Call Detail Page 210
Figure 137: Report - Sheet 1 211
Figure 138: Report - Sheet 2 211

Definitions, Acronyms, and Abbreviations

RSM	Roll System Mobile
Face Detection	**Face detection** is a computer technology that determines the locations and sizes of human faces in arbitrary (digital) images. It detects facial features and ignores anything else, such as buildings, trees and bodies. http://en.wikipedia.org/wiki/Face_detection
Face Recognition	Face recognition is the task of identifying an already detected object as a **KNOWN or UNKNOWN face**, and in more advanced cases, telling **EXACTLY WHO'S face it is***!*
Student List for final exam	A list contains all the students who can take part in the final exam of a roll call (Student with **absent rate > 20%** can't take part in final exam).

A. Introduction

1. Project Information

- Project name: **The roll system using mobile device**
- Project Code: **RSM**
- Product Type: **Website, Phone Application**
- Start Date: **September 9th, 2013**
- End Date: **December14th, 2013**

2. Introduction

Roll system was known with HPLite32, SimplePass of HP fingerpint system; roll system with ID card using by almost corporation or company around the world; the system face identify by Uniqul – Finland publish on 7/15/2013 or LogonSmart by Asus. Today, the trend machines replace humans in the hard work or the work does not require high intelligence became popular. So roll system face recognition becomes ever more necessary. The number of students in a university as well as the number of employees in large corporations in Vietnam is increasing dramatically, which means that the system should have professional roll with accuracy high.

3. Current Roll Call System

Below are some current roll call systems:

- By manual: This system is widely used in school, university. The instructor will call the name of each student, check the absent, then submit the result to log system.
- Using ID card: This system is usually used in corporations. Each employee has a card. The card will be read by a card reader to check the attendance of the employee.
- Using fingerprint: This system is currently used in FPT University. 15 minutes before and after a studying session, the student must show his fingerprint to a machine to take attendance.
- Using camera + face recognition: This system is just in experimental stage. Each classroom has a camera. At the beginning of the studying session, the instructor uses the computer, connect to the camera and take picture of classroom. The system will recognize the student in the picture, write to roll call log system.

4. Problem Definition

Below are the advantage and disadvantage of current roll call systems:

- By manual
  + Advantage: Simple to implement, cheap.
  + Disadvantage: The roll call take 3-5 minutes, take effort of instructor, wrong roll call.
- Using ID card
  + Advantage: High accuracy, not take much effort to check attendance.
  + Disadvantage: High cost (ID Card, Card Reader). Risk of ID card lending, missing.
- Using fingerprint
  + Advantage: Quickly. No effort must be made from instructor.

+ Disadvantage:Cost of fingerprint reading machine. Attendance Data is lost in case of network error.

- Using camera + face recognition
  + Advantage: Quicklycompare to traidional method (10-30 seconds). The entire class picture can be stored as log.
  + Disadvantage: High cost (Camera cost). Face recognition is not accuracy.

5. Proposed Solution

The system is intended for used in only school or university, where the pupils/students sit in a classroom/examine room (A small room, from 20-30 students). The system must to manage the course, teacher, check attendance…. In detail, the system will enable following function:

5.1 Feature functions

- The system will provide a method for admin to upload the student's images, select the students in the images to make training data.
- The admin can select image from log files, or make request for student to upload images to make training data.
- The staff can manage information about semester, course, class, instructor, students.
- The system will make a roll call list (Contains: Course, Class, Time, Instructor, Student List) for each class, based on the input information.
- The system will provide a method to assign instructor and student to course. The system must check the availability of the instructor and class before assigning.
- The system must support a method to change instructor of a course (When instructor is sick or busy), change or add new study session for a course. The system will send a mail to notify the instructor about schedule changing.
- The system can output attendance report of a roll call, or the student list for final exam.
- The instructor/examiner can view info about: The course they are teaching (Contains class, subject, student list of that course).
- The instructor can view his teaching calendar, or sync this calendar to his/her Google Calendar account.
- The instructor will use the mobile application. At the beginning of a studying session, a picture of entire class will be taken for checking attendance.Base on the student list, the system will recognize the students from the picture. The mobile app will notify attendance result to instructor.
- In case of network error or wrong recognition, the instructor can check attendance manually.
- The student can view info about what course they're studying.
- The student can view their own attendance of the course they participated in.

5.2 Advantage and disadvantage

The advantage and disadvantage of the proposed solution:

- Advantage: Small cost or no cost (If the instructor has phone). Face Recognition accuracy is about 70~80%, will reach 100% with confirm from instructor. The attendance check is quickly (10~30 seconds). The taken picture can be stored as log.
- Disadvantage: Complex to implement. The accuracy can vary depend on: Noise, brightness, number of people, image solution… Need student information to setup (Each student must have at least 10 images as image data).

6. Functional Requirement

Function requirement of the system are listed as below:

6.1 Attendance Checking

- At the beginning of each studying session, the instructor takes a picture of entire class for attendance checking.The mobile app will notify the result to the instructor.
- In case of wrong recognition, the instructor can re-check attendance manually. The system must support instructor to check attendance offline when the server or network is not available.
- The instructor can view/re-check the attendance log of the course they are teaching.
- The student can view the attendance log of the course they are studying.

6.2 Attendance Report

- The system will make attendance report, which shows the attendance, absent of student.
- The system can export report about the student list for final exam.
- The report is only available to staff and instructor.

6.3 Roll Call Management

- The staff can add/edit/active/inactive roll call.
- The staff must input info of a roll call: The course, the instructor who teaches the course, the student list, the time of the course (Begin, End Date).
- The staff can change schedule, change instructor of a study session in roll call. The staff can also create add new study session.
- When the schedule is changed, an email will be sent to instructor.

6.4 Calendar Syncing

- The instructor can see his teaching calendar: What class, subject he will teach at time.
- This calendar can be synced with the instructor's Google Calendar account.

6.5 Student Management

- The staff can add/edit/active/inactive student.
- Each student must has a training set, contains 8-20 images for face recognizing. The admin can upload images, select the faces of the students to add. The admincancan also remove images from training set.
- The admin can use the log images to get student images, or send upload request to student.

6.6 Create Account

- The account of staff, instructors and students create by admin.
- Each user can change their password.

6.7 Instructor Management

- The staff can add/edit/active/inactive instructor.

6.8 Class Management

- The staff can add/edit/active/inactive class.

6.9 Subject Management

- The staff can add/edit/active/inactive subject.

6.10 Semester Management

- The staff can add/edit/active/inactive semester.

7. Role and Responsibility

No	Full Name	Role	Position	Contact
1	Kiều Trọng Khánh	Project Manager	Instructor	khanhkt@fpt.edu.vn
2	Phạm Huy Hoàng	Developer	Team Leader	hoangphse60740@fpt.edu.vn
3	Nguyễn Thanh Bình	Developer	Team Member	binhnt60321@fpt.edu.vn
4	Nguyễn Quốc Huy	Developer	Team Member	huynq60551@fpt.edu.vn
5	Đỗ Minh Đạt	Developer	Team Member	datdm60545@fpt.edu.vn

Table 1: Roles and Responsibility

B. Software Project Management Plan

1. Problem Definition

1.1 Name of this Capstone Project

The roll system using mobile device (RSM)

1.2 Problem Abstract

Roll system was known with HPLite32, SimplePass of HP fingerpint system; roll system with ID card using by almost corporation or company around the world; the system face identify by Uniqul – Finland publish on 7/15/2013 or LogonSmart by Asus. Today, the trend machines replace humans in the hard work or the work does not require high intelligence became popular. So roll system face recognition becomes ever more necessary. The number of students in a university as well as the number of employees in large corporations in Vietnam is increasing dramatically, which means that the system should have professional roll with accuracy high.

1.3 Project Overview

1.3.1 The Current System

Below are some current roll call systems:

- By manual: This system is widely used in school, university. The instructor will call the name of each student, check the absent, then submit the result to log system.
  + Advantage: Simple to implement, cheap.
  + Disadvantage: The roll call take 3-5 minutes, take effort of instructor, wrong roll call.
- Using ID card: This system is usually used in corporations. Each employee has a card. The card will be read by a card reader to check the attendance of the employee.
  + Advantage: High accuracy, not take much effort to check attendance.
  + Disadvantage: High cost (ID Card, Card Reader). Risk of ID card lending, missing.
- Using fingerprint: This system is currently used in FPT University. 15 minutes before and after a studying session, the student must show his fingerprint to a machine to take attendance.
  + Advantage: Quickly. No effort must be made from instructor.
  + Disadvantage: Cost of fingerprint reading machine. Attendance Data is lost in case of network error.
- Using camera + face recognition: This system is just in experimental stage. Each classroom has a camera. At the beginning of the studying session, the instructor uses the computer, connect to the camera and take picture of classroom. The system will recognize the student in the picture; write to roll call log system.
  + Advantage: Quickly compare to traidional method (10-30 seconds). The entire class picture can be stored as log.
  + Disadvantage: High cost (Camera cost). Face recognition is not accuracy.

1.3.2 The Proposed System

The system is intended for used in only school or university, where the pupils/students sit in a classroom. The system must to manage the course, teacher, check attendance…. In detail, the system will enable following function:

1.3.2.1 Web Application

- The staff can manage information about course, class, instructor, students. The system provide a method for staff to import student, class list from Excel file.
- The system will provide a method for admin to upload the student's images, select the students in the images to make training data. The admin can also select images for log, or send image upload request to student.
- The system will make a roll call list (Contains: Course, Class, Time, Instructor, Student List) for each class, based on the input information.
- The system will provide a method to assign instructor and student to course. The system must check the availability of the instructor and class before assigning.
- The system must support a method to change instructor of a course (When instructor is sick or busy). The staff can also create study session, or change teaching time of existed sesssion.
- The system will provide a method for instructor to view their teaching calendar, and synchronize this calendar to their Google Calendar account
- The system can output report about the attendance, absent rate of roll calls, or the student list for final exam of that roll call.
- The student can view info about what course they're studying, their own attendance of the course they participated in.

1.3.2.2 Mobile Application

- The instructor can view info about the course they are teaching (Contains class, subject, student list of that course).
- The instructor will use the mobile application to take a picture.
- The mobile shows the list of student present in class, notify back to the instructor.
- In case of network error or wrong recognition, the instructor can re-check attendance manually.

1.3.3 Boundaries of the System

- The system is intended for using university, with small classroom, or for the examine room (The testing site will be FPT University).
- The maximum number of a room is 30 people. The room size is about: 5m x 8m
- The system is not intended for managing these aspect:
  + Auto-arranging the teaching calendar of instructor, teaching calendar of semester.
  + Managing instructor qualification, salary info.
  + Managing the testing, mark of student of each class.
- The language of the system is English.
- The complete product includes:

+ The website, for admin, staff and students. Instructor can also use the website to change profile, view roll call info, check attendance.
+ Mobile Application for instructor to check attendance.
+ All the process involved document.

1.3.4 Development Environment

1.3.4.1 Hardware requirements

For Server

Windows	Minimum Requirements	Recommended
Internet Connection	Cable, Wifi (4 Mbps)	Cable, Wifi (8 Mbps)
Operating System	XP, Vista, 7, 8	XP, Vista, 7, 8
Computer Processor	Intel® Core 2 Duo	Intel® Core(TM) i5 CPU , M 460 @ 2.53GHz
Computer Memory	1GB RAM	3GB or more

Table 2: Hardware Requirement for Server

For Mobile Application

Mobile	Minimum Requirements	Recommended
Internet Connection	Wifi (2Mbps)	Wifi (4Mbps)
Operating System	Android 4.0 or later version	Android 4.4
Hardware	Touchscreen, Camera 2.0 MP or above	Touchscreen, Camera 4.0 MP or above
Memory	512 MB or more	1 GB or more

Table 3: Hardware Requirement for Mobile App

1.3.4.2 Software requirements

- Microsoft Windows 7 Service Pack 1: operating system and platform for development.
- SQL Server 2008 Express:used to create and manage the database for system.
- StarUML: used to create models and diagrams
- Skype: used for communication and meeting
- Visual Studio 2010: used to implement website and web service.
- Eclipse Juno 4.4, Android SDK 22.0.5, ADT 22.0.5 & JDK 7u25: used to implement mobile application.
- Google Code & TortoiseSVN: used for source control.

2. Project organization

2.1 Software Process Model

Project is developed under agile model.

Figure 1: Agile Development Model

For more information: http://www.indicthreads.com/1439/quick-introduction-to-agile-software-development/[1]

(Owner: IndicThreads.com. Online Software Developer Magazine and Conferences)

2.2 Roles and responsibilities

No	Full name	Role in Group	Responsibilities
1	Kiều Trọng Khánh	Project manager	• Specify user requirement • Control the development process • Give out technique and business analysis support
2	Phạm Huy Hoàng	Team Leader, BA, DEV, Tester	• Managing process • Designing database • Clarifying requirements • Prepare documents • GUI Design • Create test plan • Coding

			• Testing
3	Nguyễn Thanh Bình	Team Member, BA, DEV, Tester	• Designing database • Clarifying requirements • Prepare documents • GUI Design • Create test plan • Coding • Testing
4	Nguyễn Quốc Huy	Team Member, BA, DEV, Tester	• Designing database • Clarifying requirements • Prepare documents • GUI Design • Create test plan • Coding • Testing
5	Đỗ Minh Đạt	Team Member, BA, DEV, Tester	• Designing database • Clarifying requirements • Prepare documents • GUI Design • Create test plan • Coding • Testing

Table 4: Roles and Responsibility Details

2.3 Tools and Techniques

- Front-end technologies: HTML5, CSS3, JavaScript, jQuery, AJAX.
- Back-end: Website: ASP.NET MVC3 + Entity Framework.
- Mobile App: Android SDK 22.0.5 – Java JDK 7u25.
- Web Server: Microsoft IIS 7.0.
- Database Management System: MS SQL Server 2008 Express.

3. Project Management Plan

3.1 Iteration

Phase /Iteration	Description	Deliverables	Resource needed	Dependencies and Constrains	Risks
Preliminary Investigation or Analysis	- Study similar existing systems. -Identify and clarify requirement for the system in general.	-Introduction of proposed system. -Main functions. -Project Iteration Plan.	30 man-days	N/A	Project may not be feasible for developing because lack of technologies and/or data
Face Detect & Recognize	-Studying face detect, face recognize algorithm & library. -Find a solution, optimize for higher performance and accuracy.	-Face Detect and Recognize System (On Web Site and Web Service)	25 man-days	N/A	Lack of experience. The implemented algorithm is not the best. Lack of test data
Student management	-Manage subject, student images/information.	-Student management function	20 man-days	Depend on "**Face Detect & Recognize**"	
Roll call management	-Create the roll call list, based on the information of student, instructor, class and course .	-Roll call management function	30 man-days	N/A	Lack of experience. Not have a clear understanding about business process.
Web Service Implement	-Create and deploy the web service on server. Provide RestFul service.	-Running web service	20 man-days	Depend on "**Roll call management**"	Lack of experience on making and deploying web service.
Attendance	-Instructor	-Android App	25 man-	Depend on	Lack of

checking	use mobile app to take picture, use picture for attendance checking. -Instructor and admin can view reports.	with attendance checking function	days	"**Web Service Implement**"	experience on Android development, transfer file between Android and web service. No Android device available for testing.
Study session management	- Staff can add session, chanage instructor, teaching time of existed session	- Study session management function.	20 man-days	N/A	Lack of experience of making report. Not have a clear understanding about business process.

Table 5: Iteration

3.2 Iteration Detail

3.2.1 Phase 1: Preliminary Investigation or Analysis

Task	Description	Author
1. Identifying and studying existing systems	Find which systems currently provide similar service, their strengths and weakness.	HuyNQ, HoangPH, BinhNT
2. Identifying and clarifying main functions	Define which main functions system should provide.	HuyNQ, HoangPH, BinhNT
3. Introduction	Complete Introduction Report.	HoangPH
4. Project Management Plan	Prepare Project Management Plan.	HoangPH
5. Website Prototype	Build a prototype of proposed system (Website).	HuyNQ, HoangPH
6. Mobile Prototype	Build a prototype of proposed system (Mobile App).	BinhNT, HoangPH
7. Design ER diagram	Design ER diagram.	HoangPH, HuyNQ, BinhNT

Table 6: Phase 1: Preliminary Investigation or Analysis

3.2.2 Phase 2: Face Detect & Recognize

Task	Description	Author
1. Identifying Requirement and Planning	Which feature this function should have and how to implement.	HoangPH

2. Studying Face Detection & Recognition Algorithm	Studying algorithm, implement by using library EmguCV.	HoangPH
3. Extract Face from Image	Find the faces in images, extract them for later use	HoangPH
4. Recognize Face	From the input face, find out who the face belong to	HoangPH
5. Optimize	Optimize the implement for more performance and accuracy	HoangPH
6. Implement GUI	Create the interface for extracting and storing face	HoangPH
7. Testing	Test system behavior and performance Test user behavior and performance	HoangPH
8. Document	Adding SRS, SDD, Installation Guide, Manual Guide	HoangPH, HuyNQ, BinhNT

Table 7: Phase 2: Face Detect & Recognize

3.2.3 Phase 3: Student Management

Task	Description	Author
1. Identifying Requirement and Planning	Which feature this function should have and how to implement.	HoangPH, HuyNQ, BinhNT
2. Design ER Diagram	Design ER Diagram	HoangPH, HuyNQ, BinhNT
3. Manage Student	Allow staff to add/edit/active/inactive student Allow staff to import student list from excel file.	HuyNQ
4. Manage Student Face	Each student has a training set, which contains 8-20 faces. Allow admin to add, edit images in this training set.	HoangPH, HuyNQ
5. Use Log Image for Recognizer Training	Admin can select image from log to train the recognizer, for higher accuracy	HoangPH
6. Send Image Upload Request to Student	If the student image isn't enough, admin can send request for student to upload image	HuyNQ, HoangPH
7. Implement GUI	Create the interface for managing student info	HuyNQ, HoangNQ, BinhNT
8. Testing	Test system behavior and	HoangPH, HuyNQ

	performance Test user behavior and performance	
9. Document	Adding SRS, SDD, Installation Guide, Manual Guide	HoangPH, HuyNQ

Table 8: Phase 3: Student Management

3.2.4 Phase 4: Roll Call Management

Task	Description	Author
1. Identifying Requirement and Planning	Which feature this function should have and how to implement.	HoangPH, HuyNQ, BinhNT
2. Manage Instructor	Allow staff to add/edit/active/inactive instructor	HuyNQ
3. Manage Course	Allow staff to add/edit/active/inactive course	HuyNQ
4. Manage Class	Allow staff to add/edit/active/inactive course. Assign student to class	HuyNQ
5. Manage semester	Allow staff to add/edit/active/inactive semester	HuyNQ
6. Manage Roll Call	Allow staff to create/edit/delete roll call. Each roll call contain info: The instructor, the student list, the course, time, begin – end date Assign, remove student to/from roll call	HuyNQ, HoangPH
7. Export Excel Report	Export the attendance report (Contains attendance, absent rate of student in a roll call). Export the student list for final exam of a roll call.	HoangPH, HuyNQ
8. Implement GUI	Create the interface for managing roll call	HuyNQ, BinhNT
9. Testing	Test system behavior and performance Test user behavior and performance	HoangPH, HuyNQ, BinhNT
10. Document	Adding SRS, SDD, Installation Guide, Manual	HoangPH, HuyNQ, BinhNT

	Guide	

Table 9: Phase 4: Roll Call Management

3.2.5 Phase 5: Web Service Implement

Task	Description	Author
1. Identifying Requirement and Planning	Which feature this function should have and how to implement.	HoangPH, HuyNQ, BinhNT
2. Studying WebService	Studying the create and using of web serivce	HuyNQ, HoangPH
3. Instructor Login	Check the input id and password from mobile to login instructor	HuyNQ
4. Get Instructor Info, Roll Call List	Based on the instructor's id, show to current roll call list	HuyNQ, HoangPH
5. Face Recognize From Android Camera	Studying about transferring image files between Android and webserivce Transfer the result back to Mobile for showing	HuyNQ, BinhNT
6. Check Attendance Manually	Allow instructor to confirm and re-check attendance	HoangPH, HuyNQ, BinhNT
7. Testing	Test system behavior and performance Test user behavior and performance	HoangPH, HuyNQ, BinhNT
8. Document	Adding SRS, SDD, Installation Guide, Manual Guide	HoangPH, HuyNQ, BinhNT

Table 10: Phase 5: Web Service Implement

3.2.6 Phase 6: Attendance Checking

Task	Description	Author
1. Identifying Requirement and Planning	Which feature this function should have and how to implement.	HoangPH, HuyNQ, BinhNT
2. Getting Image From Mobile, sending to Web Service	Allow instructor to take picture of class, use picture for checking attendance	BinhNT
3. Confirm result, re-check attendance manually	Allow instructor to confirm and re-check attendance	BinhNT
4. Offline Attendance Checking	Store some info into mobile for checking when offline	BinhNT
5. Implement Mobile App	Create the Android App for instructor to User	BinhNT, HuyNQ, HoangNQ
6. Testing	Test system behavior and performance	HoangPH, HuyNQ, BinhNT

	Test user behavior and performance	
7. Document	Adding SRS, SDD, Installation Guide, Manual Guide	HoangPH, HuyNQ, BinhNT

Table 11: Phase 6: Attendance Checking

3.2.7 Phase 7: Study session management

Task	Description	Author
1. Identifying Requirement and Planning.	Which feature this function should have and how to implement.	HoangPH, HuyNQ, BinhNT
2. Display Study Session	Process all the study session of a roll call, display as event in calendar	HoangPH, HuyNQ
3. Manage Study Session	Create new study session, edit the teaching time, change session instructor. Send mail to instructor when change teaching time	HoangPH, HuyNQ
4. Teaching Calendar	Display the teaching calendar of an instructor. The instructor can sync this calendar to his/her Google Calendar Account	HoangPH, HuyNQ
5. Testing	Test system behavior and performance Test user behavior and performance	HoangPH, HuyNQ, BinhNT
6. Document	Adding SRS, SDD, Installation Guide, Manual Guide	HoangPH, HuyNQ, BinhNT

Table 12: Phase 7: Study Session Management

3.3 All Meeting Minutes

Refer to Meeting Minutes folder.

4. Coding Convention

Java: Using to develop Android App.

Summary:

- Naming Convention.
- Indentation.
- Declaration.
- Code Examples

Follow "Code Conventions for the Java TM Programming Language, by Sun Microsystems, rev April 20, 1999".

C#: Using to develop website and web service.

Summary:

- Naming Convention.
- Layout Convention.
- Commenting Convention.
- Language Guidelines

Using C# Code Convention From:

http://msdn.microsoft.com/en-us/library/vstudio/ff926074.aspx [2]

C. Software Requirement Specification

1. User Requirement Specification

1.1 Instructor Requirement

Instructor is teacher of the university. Instructor's account is created by admin.Instructor can use mobile app or website, that can do functions:

- Check Attendance Auto/Manually
- View/Sync Teaching Calendar
- Get Attendance Report

1.2 Admin Requirement

Admin is the one who maintained and config the system, and manage the image data of student. Admin can do functions:

- Manage Student Image: Add/Delete Student's Image
- Use log image to train the face recognizer
- Manage Image Upload Request: Create/Accept/Deny
- Manage Account: Create/Edit/Active/Inactive Account
- Configure System

1.3 Staff Requirement

Staff is the employee of the university office. Staff's account is created by admin. Staff can use website with these functions:

- Manage Roll Call: Add/Edit Roll Call. Change Roll Call Student List
- Manage Study Session: Add/Change Time/Instrctuor of Study Session
- Report Attendance Rate: Report by Class, Block, Student. Export report
- Manage Subject: Add/ Edit/Active/ Inactive Subject
- Manage Class: Add/Edit /Active/ Inactive Class
- Manage Instructor: Add/Edit/Active/InactiveInstructorInstructor
- Manage Semester: Add/Edit/Active/InactiveSemesterSemester
- Manage Student: Add/Edit Student. Import Student List
- Check Attendance of existed Roll Call

1.4 System Requirement

System is also an actor, run in the background to keep the system working. System can do functions:

- Auto free storage space
- Face Detection
- Face Recognition: Recognition Student/Store Result
- Manage Roll Call: Auto Active/Inactive Roll Call

1.5 Student Requirement

Student is the student of the university. Student's account is created by admin. . Staff can use website with function:

- Check own present rate

1.6 Guest Requirement

Guest is a person who doesn't have access to the system. To use system functions, guest must Login.

2. System Requirement Specification

2.1 External Interface Requirement

2.1.1 User Interface

- The interface of website is clear, do not annoy customer.
- The interface of mobile app must be clear, compatibale with touch screen. The size of controls must be big enough to touch on smartphone.
- The error, warning messages must be make clear, easy to understand. Error warning does not disturb customer.

2.1.2 Hardware Interface

- The system will use the standard hardware and data communications resources of a standard computer.

2.1.3 Software Interface

- Firefox Browser, Chromes with Resolution (1024*768) or bigger and support JavaScript and HTML5
- Smartphone with Android 4.0 or above. Screen size (70 x 120 mm) or bigger.

2.1.4 Communication Protocol

- Website using HTTP protocol for communication between the web browser and the web server.
- Mobile app using HTTP protocol for communicating between app and web service.

2.2 System Overview Use Case

Figure 2: System Overview Use Case

2.3 List of Use Case

2.3.1 <Admin>Overview Use Case

Figure 3:<Admin> Overview Use Case

2.3.1.1 <Admin> Add Image for Single Student Use Case Diagram

Figure 4: <Admin> Add Image for Single Student

Use Case Specification

USE CASE – AD001			
Use Case No.	AD001	**Use Case Version**	2.0
Use Case Name	Add Image for Single Student		
Author	Pham Huy Hoang		
Date	20/09/2013	**Priority**	Normal

Actor: Admin

Summary:

This user case allow admin add image data for a student.

Goal:

Image of a students will be stored on server, the links to images is stored in database

Triggers:

- For face recognition, our system must store images of students face. With more images, we can have better recognition accuracy.
- To add new image for a single student, admin must:
  - Click menu item"Manage Student Image" -> "Student List".
  - On the Student List Page, click Detail to view image data of a student.
  - Upload images. System will detect human faces from uploaded images.
  - Select the student faces from images.
  - Press "Submit Result" to save the images.

Preconditions:

- User must logged in the system with the role is admin.
- There must be at lease one student in the student list.
- The admin has images contain the student's face.

Post Conditions:

- **Success:** The images of a student will be stored on server and database
- **Fail:**No image added

Main Success Scenario:

Step	Actor Action	System Response
1	- Click menu item"Manage Student Image" - Click choose "Student List" menu. [Alternative 1]	Redirect to Student List Page, contains : • Table with 5 columns, record of 30 lastest students: - Student Code: label - Name: label - Email: label - Image Data: "… images" button - Request" "Request Upload" button. • Class: Dropdownlist, data source: All active class in database.
2	Click on "… images" button.	Redirect to Student Detail Page, contains detail of Student Image Data, include: • Student Current Image: A list of images

		• File Upload Control • Upload: button
2	Select the student's images from computer.	The selected images will be displayed as thumbnails. [Exception 1]
3	Click "Upload" button.	System will detect student faces from uploaded images, then display image full size with white **face regions** (See Use Case SY001) [Exception 2, 3]
4	Select student's face from face region. Click "Submit Result" button.	The selected face region will be cropped from original picture, stored into server. Face's owner and link will be stored in database. [Exception 4] The pop-up is closed. The Student List Page is auto refresh.

Alternative Scenario:

No	Actor Action	System Response
1	[Alternative 1] - Select a class from Class drop down list. - Press "Select" button.	A table show the records of all students in selected class. Table with 5 columns: - Student Code: label - Name: label - Email: label - Image Data: "… images" button - Request" "Request Upload" button.

Exceptions:

No	Actor Action	System Response
1	Select more than 20 images	Display error message: " Too much images. Only process first 20 images"
2	Not select image, click "Upload" button	Display error message: "Please choose images for uploading"
3	- Upload images don't contain human faces. Or -No faces detected from uploaded images	Display error message: " No face found. Select another images"
4	The selected face already in	Display message: "Face XXX existed in

	the database	database”

Relationships: Login,Manage Student Image
Business Rules:

- For saving storage space, only student’s face is stored. The face is saved as jpeg, resolution 100 x 100.
- The recommend number of image for a student is larger than 10 for acceptable accuracy.
- The original uploaded images willbe stored on Temp folder on server. This Temp folder will be emptied at time interval (See Use Case SY001).

Table 13: <Admin> Add Image for Single Student

2.3.1.2 <Admin> Add Image for Multi Student

Use Case Diagram

Figure 5: <Admin> Add Image for Multi Student

Use Case Specification

USE CASE – AD002			
Use Case No.	AD002	**Use Case Version**	2.0
Use Case Name	Add Image for Multi Student		
Author	Pham Huy Hoang		
Date	20/09/2013	**Priority**	Normal

Actor: Admin
Summary:
This user case allow admin to add image data for many students.
Goal:
Image of students will be stored on server, the links to images is stored in database
Triggers:

- For face recognition, our system must store images of students face. With more images, we can have better recognition accuracy.
- To add new image for a single student, admin must:
  - Click menu item“Manage Student Image” -> “Add Image for Students”.
  - Upload images. System will detect student faces from uploaded images.
  - Select the student faces from images.
  - Select the each face’s owner.
  - Press “Submit Result” to save the images and students.

Preconditions:

- User must logged in the system with the role is admin.
- There must be at lease one student in the student list.
- The admin has images contain the student’s face.

Post Conditions:

- **Success:** The images of students will be stored on server and database
- **Fail:**No image added

Main Success Scenario:

Step	Actor Action	System Response
1	- Click menu item"Manage Student Image" - Click choose "Add Image for Students".	Redirect to Add Image for Students Page, contains: • Upload Students Images: text label • File Upload Control • Upload: button
2	Select the student's images from computer	The selected images will be displayed as thumbnails. [Exception 1]
3	Click "Upload" button	System will detect student faces from uploaded images, then display image full size with white **face regions** (See Use Case SY001) [Exception 2, 3]
4	Select student's face from face region, select the face owner [Alternative 1]	When user hover into a face region, a suggestion textbox is displayed. (This suggestion textbox help admin to choose the face's owner.) The textbox data source is all students in database. A suggestion record includes: • Student Face: image • Student ID: label • Full name: label • Class: label
5	Click "Submit Result" button.	The selected face region will be cropped from original picture, stored into server. Face's owner and link will be stored in database. [Exception 4]

Alternative Scenario:

No	Actor Action	System Response
1	Admin doesn't choose the face's owner.	The suggestion textbox is cleared.

Exceptions:

No	Actor Action	System Response
1	Select more than 20 images	Display error message: " Too much images. Only process first 20 images"
2	Not select image, click "Upload" button	Display error message: "Please choose images for uploading"
3	Upload images don't contain human faces. No faces detected from uploaded images	Display error message: " No face found. Select another images"
4	The select face already in the database	Display message: "Face XXX existed in database"

Relationships: Login,Manage Student Image
Business Rules:

- The same as Use Case AD005.
- The auto-complete textbox must show the student's face, name, ID and class, as the same as facebook.

Table 14: <Admin> Add Image for Multi Students

2.3.1.3 <Admin> Remove Student Image

Use Case Diagram

Figure 6: <Admin> Remove Student Image

Use Case Specification

USE CASE – AD003			
Use Case No.	AD003	**Use Case Version**	2.0
Use Case Name	Remove Student Image		
Author	Pham Huy Hoang		
Date	20/09/2013	**Priority**	Normal

Actor: Admin
Summary:
This user case allow admin to remove image of a student.
Goal:
Remove an/many images from a student Image Data
Triggers:

- For face recognition, our system must store images of students face. Sometimes, the admin can add the wrong face of students, which lower the accuracy. So the admin must be able to remove image of a student.
- To remove image of a single student, admin must:

- Click menu item“Manage Student Image” -> “Student List”.
- On the Student List Page, click Detail to view image data of a student.
- Click on the “X” button at the top-right corner of the image.
- Click “Ok”

Preconditions:

- User must logged in the system with the role is admin.
- There must be at lease one student in the student list. This student must has at least 1 face image.

Post Conditions:

- **Success:** The image of a student will be removed from server and database
- **Fail:**No image removed

Main Success Scenario:

Step	Actor Action	System Response
1	Click menu item“Manage Student Image” -> “Student List”. [Alternative 1]	Redirect to Student List Page, contains : • Table with 5 columns, record of 30 lastest students: - Student Code: label - Name: label - Email: label - Image Data: “… images” button - Request” “Request Upload” button. • Class: dropdownlist, data source: All active class in database.
2	Click on “Detail” button	Redirect to Student Detail Page, contains detail of Student Image Data, include: • Student Current Image: A list of images • File Upload Control • Upload: button
3	Click on the “X” button at the top-right corner of the image	Show confirms windows with message: “Are you sure to delete this image?”
4	Click “Ok” button. [Alternative 2]	The selected image will be removed from server and database. An message is display: “Image XXX removed”.

Alternative Scenario:

No	Actor Action	System Response
1	[Alternative 1] - Select a class from Class	

	drop down list. - Press “Select” button.	A table show the records of all students in selected class. Table with 4 columns: - Student Code: label - Name: label - Email: label - Image Data: “Detail” button
3	[Alternative 2] Click “Cancel”	The image is not removed.

Exceptions: N/A
Relationships: Login,Manage Student Image
Business Rules: N/A

Table 15: <Admin> Remove Student Images

2.3.1.4 <Admin> Get Image From Log Images

Use Case Diagram

Figure 7: <Admin>Get Image From Log Images

Use Case Specification

USE CASE – AD004			
Use Case No.	AD004	**Use Case Version**	2.0
Use Case Name	Get Image From Log Images		
Author	Pham Huy Hoang		
Date	20/09/2013	**Priority**	Normal

Actor: Admin
Summary:
This user case allow admin to select student image from log images (Images stored as log for attendance checking).
Goal:
Select student image from log images, to train the face recognizer for higher accuracy.
Triggers:

- When the instructor take attendance, the images of class will be stored ad log images. We can use these log images to add image data for student, without uploading new image.
- To get image from log images, admin must:
  - Click menu item“Manage Student Image” -> “Log Images”.

- Select the begin date, end date, class to get log.
- Select the log images to used.
- Select the face owner from log images.
- Presss “Submit Result” button.

Preconditions:

- User must logged in the system with the role is admin.
- There must be at lease one log images in database.

Post Conditions:

- **Success:** The selected image will be stored to database.
- **Fail:**No image removed

Main Success Scenario:

Step	Actor Action	System Response
1	Click menu item“Manage Student Image” -> “Log Images”.	Redirect to Log Image Page, contains : • Start Date: datepicker • End Date: datepicker • Class: dropdownlist, data source: All active class in database
2	Click on “Find” button	Show the log image of selected class, from start date to end date. Date and class of the images are shown when user hover the mouse on images.
3	Click the image to select. Press “Use selected images” button.	System will detect student faces from uploaded images, then display image full size with white **face regions** (See Use Case SY002)
4	Select student’s face from face region, select the face owner [Alternative 1]	When user hover into a face region, the face owner is shown. If user click the region, a suggesion textbox will be shown. (This suggestion textbox help admin to choose the face’s owner.) The textbox data source is all students in database. A suggestion record includes: • Student Face: image • Student ID: label • Full name: label • Class: label
5	Press “Save” button	The selected face region will be cropped

		from original picture, stored into server. Face's owner and link will be stored in database. [Exception 1]

Alternative Scenario:

No	Actor Action	System Response
1	Admin doesn't choose the face's owner.	The suggestion textbox is cleared.

Exceptions:

No	Actor Action	System Response
1	The select face already in the database	Display message: "Face XXX existed in database"

Relationships: Login,Manage Student Image
Business Rules: N/A

Table 16: <Admin>Get Image From Log Image

2.3.1.5 <Admin> Create Request

Use Case Diagram

Figure 8: <Admin>Create Request

Use Case Specification

USE CASE – AD005			
Use Case No.	AD005	**Use Case Version**	2.0
Use Case Name	Create Request		
Author	Pham Huy Hoang		
Date	20/09/2013	**Priority**	Normal

Actor: Admin
Summary:
This user case allow admin to create an image uploading request
Goal:
An image uploading request is created
Triggers:

- For face recognition, our system must store images of students face. If the student images isn't enough, the admin can create a request for the student to upload his/her own images.
- To create a request, admin must:
  - Click menu item"Manage Student Image" -> "Student List".
  - On the Student List Page, click "Request Upload" to send request to a student.
  - Enter the request content.

- Press “Send Request” content.

Preconditions:

- User must logged in the system with the role is admin.
- There must be at least 1 inactive account in database.

Post Conditions:

- **Success:**A request will be created.
- **Fail:** No request created.

Main Success Scenario:

Step	Actor Action	System Response
1	- Click menu item“Manage Student Image” - Click choose “Student List” menu.	Redirect to Student List Page, contains : • Table with 5 columns, record of 30 lastest students: - Student Code: label - Name: label - Email: label - Image Data: “… images” button - Request” “Request Upload” button. • Class: Dropdownlist, data source: All active class in database.
2	Click “Request Upload” button.	Redirect to Create Request Page, contains: • Request to: textbox (uneditable) • Content: textbox (max length: 50, required). • Send Request: button
3	Click “Send Request” button.	Redirect back to the Request List Page, contains: • Table with 4 columns, record of pending request: - No: label - Student Code: label - Full Name: label - Images: “… images” button

Alternative Scenario: N/A

Exceptions:N/A

Relationships: Login,Manage Image Upload Request

Business Rules:

- Admin can only send request to student with less than 10 images.

Table 17: <Admin>Create Request

2.3.1.6 <Admin> Accept Request

Use Case Diagram

Figure 9: <Admin>Accept Request

Use Case Specification

USE CASE – AD006			
Use Case No.	AD006	**Use Case Version**	2.0
Use Case Name	Accept Request		
Author	Pham Huy Hoang		
Date	20/09/2013	**Priority**	Normal

Actor: Admin

Summary:

This user case allow admin to accept an image uploading request

Goal:

An image uploading request from student is accepted

Triggers:

- When student login, they can see the request from admin. If the student upload images as request, the request will be viewed by admin. If admin accept the request, all the images sent by request will be saved as student image data
- To accept a request, admin must:
  - Click menu item"Manage Request" -> "Pending Request".
  - On the Request List Page, click "… images" button to view request detail.
  - Press "Accept" button.

Preconditions:

- User must logged in the system with the role is admin.
- There must be at least 1 pending request from student.

Post Conditions:

- **Success:** The request is accepted. Images in the request are saved.
- **Fail:** No request accepted.

Main Success Scenario:

Step	Actor Action	System Response
1	- Click menu item"Manage Request" - Click choose "Pending Request" menu.	Redirect back to the Request List Page, contains: • Table with 4 columns, record of pending request:

		- No: label - Student Code: label - Full Name: label - Images: "… images" button
2	Click "… images" button.	Redirect to Request Detail Page, contains: • Student name: label • Student Code: label • Class: label • Request Image: images • Accept: button • Deny. Send request again: button
3	Click "Accept" button. [Alternative 1]	All the images in the request will be saved as student image data. Redirect back to the Request List Page, contains: • Table with 4 columns, record of pending request: - No: label - Student Code: label - Full Name: label - Images: "… images" button

Alternative Scenario:

No	Actor Action	System Response
1	Click the "X" button as the top right corner of request image.	The image is removed from request.

Exceptions: N/A
Relationships: Login,Manage Image Upload Request, Create Request
Business Rules: N/A

Table 18: <Admin>Accept Request

2.3.1.7 <Admin> Deny Request

Use Case Diagram

Figure 10: <Admin>Deny Request

Use Case Specification

USE CASE – AD007			
Use Case No.	AD007	**Use Case Version**	2.0
Use Case Name	Deny Request		
Author	Pham Huy Hoang		
Date	20/09/2013	**Priority**	Normal

Actor: Admin

Summary:

This user case allow admin to deny an image uploading request

Goal:

An image uploading request from student is denied

Triggers:

- When student login, they can see the request from admin. If the student upload images as request, the request will be viewed by admin. If admin denied the request, student must upload image again.
- To accept a request, admin must:
  - Click menu item"Manage Request" -> "Pending Request".
  - On the Request List Page, click "… images" button to view request detail.
  - Press "Deny. Send request again" button.

Preconditions:

- User must logged in the system with the role is admin.
- There must be at least 1 pending request from student.

Post Conditions:

- **Success:** The request is accepted. Images in the request are saved.
- **Fail:** No request accepted.

Main Success Scenario:

Step	Actor Action	System Response
1	- Click menu item"Manage Request" - Click choose "Pending Request" menu.	Redirect back to the Request List Page, contains: • Table with 4 columns, record of pending request: - No: label - Student Code: label - Full Name: label - Images: "… images" button
2	Click "… images" button.	Redirect to Request Detail Page, contains: • Student name: label • Student Code: label • Class: label • Request Image: images • Accept: button • Deny. Send request again: button
3	Click "Deny. Send request	

	again" button. [Alternative 1]	Redirect to Create Request Page, contains: • Request to: textbox (uneditable) • Content: textbox (max length: 50, required). • Send Request: button

Alternative Scenario:

No	Actor Action	System Response
1	Click the "X" button as the top right corner of request image.	The image is removed from request.

Exceptions: N/A
Relationships: Login,Manage Image Upload Request, Create Request
Business Rules: N/A

Table 19: <Admin>Deny Request

2.3.1.8 <Admin> Create Account

Use Case Diagram

Figure 11: <Admin> Create Account

Use Case Specification

USE CASE – AD008			
Use Case No.	AD008	**Use Case Version**	2.0
Use Case Name	Create Account		
Author	Pham Huy Hoang		
Date	20/09/2013	**Priority**	Normal

Actor: Admin
Summary:
This user case allow admin to create an account for login the system
Goal:
A new account (For student, instructor or staff) will be added to database
Triggers:

- In our system, the account of a student or instructor is auto created when staff create new instructor/new student. This function is used to create staff, student or instructor account, in case of backup.
- To create a new account, admin must:
  - From the admin Home Page, click the menu item "Manage Account".
  - In the Account Manager Page, click on the Tab "Staff Accounts".
  - Input new account information, click "Create" to finish.

Preconditions:

- User must logged in the system with the role is admin

Post Conditions:

- **Success:** A new account will be created. The acocunt will be displayed on account list.
- **Fail:**No acocunt created.Transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click menu item "Account Manager".	Redirect to Account Manager Page, includes a panel with 2 tabs: • Account List • Create Account Current tab is Account List. This tab includes: • Table with 5 column: - Account ID: label - Username: label - Email: label - Role: label - Select: checkbox • Action: dropdownlist, values: Inactive, Active. • Process: button
2	Click Tab "Create Account".	Display a tab panel includes: • Username: textbox (min length:5, max length: 30, required). • Password: textbox (password box, min length,:5 max length: 30, required). • Password: textbox (password box, min length:5, max length: 30, required). • Roles : dropdownlist, values: Admin, Staff, Instructor, Student. • Create: button • Reset: button
3	Input new account information. Click "Create" button. [Alternative 1]	System will add the account to database. Show the tab panel "Account List" [Exception 1,2,3,4]

Alternative Scenario:

No	Actor Action	System Response
1	Click on "Reset" button	The form is emptied.

Exceptions:

No	Actor Action	System Response
1	Not enter enough require information.	Display error message: "Please enter all require information."

2	Username/password is not in range [6,30]	Display error message: "Username/Password must in range [6,30]."
3	Confirm password is not equal password	Display error message: "Password and confirm not match."
4	Username or email existed in database	Display error message: "Username/Email already exist."

Relationships: Login,Manage Account

Business Rules:

- Each account is active when added to database.
- The "confirm password" value must match "password" value.

Table 20: <Admin> Create Account

2.3.1.9 <Admin> Active Account

Use Case Diagram

Figure 12: <Admin> Active Account

Use Case Specification

USE CASE – AD009			
Use Case No.	AD009	**Use Case Version**	2.0
Use Case Name	Active Account		
Author	Pham Huy Hoang		
Date	20/09/2013	**Priority**	Normal

Actor: Admin

Summary:

This user case allow admin to active an accout

Goal:

A inactive account will be activated

Triggers:

- In our system, when we do not want an user to login, we inactive his/her account (See Use Case – AD003). We can active the inactivated account so they can login again.
- To active an account, admin must:
  - From the admin Home Page, click the menu item "Account Manager".
  - Check the Select checkbox of the account record, select "Active" on the drop down list, and press "Process" button.

Preconditions:

- User must logged in the system with the role is admin.
- There must be at least 1 inactive account in database.

Post Conditions:

- **Success:** The selected account(s) will be activated.

- **Fail:**No acocunt activated.Transfer to error page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click menu item“Account Manager”.	Redirect to Account Manager Page, includes a panel with 2 tabs: • Account List • Create Account Current tab is Account List. This tab includes: • Table with 5 column: - Account ID: label - Username: label - Email: label - Role: label - Select: checkbox • Action: dropdownlist, values: Inactive, Active. • Process: button
2	Check the checkboxs of the account(s), select “Active” on the drop down list, and press “Process” button.	The selected account(s) will be activated [Exception 1,2]

Alternative Scenario: N/A

Exceptions:

No	Actor Action	System Response
1	The account has already been activated by another admin	Nothing happen, the account(s) status is active
2	Check no checkbox, press “Process” button	Display error message: “Please select account to activated.”

Relationships: Login,Manage Account

Business Rules:

- Admin can’t active/inactive account of themselves or other admin.

Table 21: <Admin> Active Account

2.3.1.10 <Admin> Inactive Account

Use Case Diagram

Figure 13: <Admin> Inactive Account

Use Case Specification

USE CASE – AD010			
Use Case No.	AD010	**Use Case Version**	2.0
Use Case Name	Inactive Account		
Author	Pham Huy Hoang		
Date	20/09/2013	**Priority**	Normal

Actor: Admin

Summary:

This user case allow admin to active an accout

Goal:

A active account will be inactivated

Triggers:

- In our system, when we do not want an user to login, we inactive his/her account. The account can be activated later.
- To active an account, admin must:
  - From the admin Home Page, click the menu item "Account Manager".
  - Check the Select checkbox of the account record, select "Active" on the drop down list, and press "Process" button.

Preconditions:

- User must logged in the system with the role is admin
- There must be at least 1 active account in database.

Post Conditions:

- **Success:** The selected account(s) will be inactivated.
- **Fail:**No acocunt inactivated.Transfer to error page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click menu item"Account Manager".	Redirect to Account Manager Page, includes a panel with 2 tabs: • Account List • Create Account Current tab is Account List. This tab includes: • Table with 5 column: - Account ID: label - Username: label - Email: label - Role: label

		- Select: checkbox • Action: dropdownlist, values: Inactive, Active. • Process: button
2	Check the checkboxs of the account, select "Inactive" on the drop down list, and press "Process" button.	The selected account(s) will be inactivated. [Exception 1,2]

Alternative Scenario: N/A

Exceptions:

No	Actor Action	System Response
1	The account has already been inactivated by another admin.	Nothing happen, the account(s) status is inactive
2	Check no checkbox, press "Process" button.	Display error message: "Please select account to inactivated."

Relationships: Login,Manage Account

Business Rules:

- Admin can't active/inactive account of themselves or other admin.

Table 22: <Admin> Inactive Account

2.3.1.11 <Admin> Config System

Use Case Diagram

Figure 14: <Admin> Config System

Use Case Specification

USE CASE – AD011			
Use Case No.	AD011	**Use Case Version**	2.0
Use Case Name	Configure System		
Author	Pham Huy Hoang		
Date	20/09/2013	**Priority**	Normal

Actor: Admin

Summary:

This user case allow admin to configure system's properties.

Goal:

Change current properties of the system

Triggers:

- For some reason, administrator needs to re-configure system's properties.

- To config the system, admin must:
  - Click menu item“Configuration”.
  - On the configuration page, edit properties and click on “Update” button corresponding with each properties in list.

Preconditions:

- User must logged in the system with the role is admin.

Post Conditions:

- **Success:** System’s properties is updated to new value.
- **Fail:**Nothing changed.

Main Success Scenario:

Step	Actor Action	System Response
1	Click menu item “Configuration”.	Redirect to System Configuration Page, with list of system’s properties include: • Name: label • Value: textbox • Update: button
2	Editproperty value and click “Update” button.	Show confirms windows with message: “Are you sure to change this property?”
3	Click “Ok” to confirm. [Alternative 1]	Update property value to new value.

Alternative Scenario:

No	Actor Action	System Response
1	Click “Cancel”.	Nothing happen, the value is unchanged.

Exceptions: N/A

Relationships: Login

Business Rules:

- Configure system’s properties will affect to all function using these properties..

Table 23: <Admin> Config System

2.3.2 <System> Overview Use Case

Figure 15: <System> Overview Use Case

2.3.2.1 <System> Face Detection Use Case Diagram

Figure 16: <System> Face Detection

Use Case Specification

USE CASE – SY001			
Use Case No.	SY001	**Use Case Version**	2.0
Use Case Name	Face Detection		
Author	Pham Huy Hoang		
Date	21/09/2013	**Priority**	High
Actor: System. **Summary:** This Use Case is used to detect the face regions in an image after it is uploaded successful. **Goal:** Detect the face regions in an image **Triggers:**			

- When admin upload images for Manage Student Image (See Use Case AD005, AD006), or instructor send images for Checking Attendance (See Use Case IU001), system will detect the face regions in the upload images.
- The detected region will be displayed at client, or used to recognize the students in the images.

Preconditions:

- The images are uploaded to server successful.

Post Conditions:

- **Success:** The face regions is detected. Send result to front end for displaying.
- **Fail:**No face detected

Main Success Scenario:

Step	Actor Action	System Response
1	- Admin upload image for Manage Student Image. Or - Instructor send image for Checking Attendance.	Detect the face in the images, create a list of face regions. A face region's properties: • Top position • Left position • Height • Width The face region can be stored as Image Data for Students, or for Student Recognition (See Use Case SY002) [Exception 1,2]

Alternative Scenario: N/A

Exceptions:

No	Actor Action	System Response
1	The upload image is corrupt	Show error message: "The image XXX is corupted".
2	Upload images don't contain human faces. No faces detected from uploaded images.	Display error message: " No face found. Select another images"

Relationships: Login

Business Rules:

- The accuracy of face detection is larger than 80%. Picture with higher brightness and resolution will has higher accuracy.
- If the uploaded images are too big, the image will be resized for faster processing.

Table 24: <System> Face Detection

2.3.2.2 <System> Recognize Student

Use Case Diagram

Figure 17: <System> Recognize Student

Use Case Specification

USE CASE – SY002			
Use Case No.	SY002	**Use Case Version**	2.0
Use Case Name	Recognize Student		
Author	Pham Huy Hoang		
Date	21/09/2013	**Priority**	Normal

Actor: System.

Summary:

This Use Case is used to detect the face regions in an image (for checking attendance)

Goal:

Detect the face regions in an image

Triggers:

- When instructor send images for Checking Attendance (See Use Case IU001), system will detect the face regions in the upload images. After that, the system will recognize the student present in the images.
- The recognized student list will be stored as attendance log in database.

Preconditions:

- The images are uploaded to server successful.
- The Image Data of the students must existed in database (At least 10 images). If a student doesn't have image in database, that student will be recognized as "Unknow"

Post Conditions:

- **Success:** The students are recognized from images. Recognize result will be stored.
- **Fail:**No student recognized. No log stored.

Main Success Scenario:

Step	Actor Action	System Response
1	Instructor send image for Checking Attendance.	The same as step 1 , Use Case SY002. • Detect face regionsin the picture. • Create a face matching list. • Each face will be matched to find the face's owner. • After recognizing, show the result and store the result to database.

		[Exception 1,2,3]

Alternative Scenario: N/A

Exceptions:

No	Actor Action	System Response
1	The number of student detected is too small (fewer than 5)	Show error message: "Only some students detected. Please check".
2	Find unknow people in the image	Alert: "Stranger Detected. Please check".

Relationships: Login,Face Detection, Face Recogntion, Store Recognize Result

Business Rules:

- The accuracy of face recognition is 60~80%, depend on the number of image in training data.
- The recommend number of image for a student is larger than10 for acceptable accuracy.
- When stranger is detected, a log will be stored for later checking.

Table 25: <System> Recognize Student

2.3.2.3 <System> Auto Free Storage Space

Use Case Diagram

Figure 18: <System> Auto Free Storage Space

Use Case Specification

USE CASE – SY003			
Use Case No.	SY003	**Use Case Version**	2.0
Use Case Name	Auto Free Storage Space		
Author	Pham Huy Hoang		
Date	21/09/2013	**Priority**	Normal

Actor: System.

Summary:

This Use Case is used to auto free space on server

Goal:

Delete the image in the Temp folder to free space on server.

Triggers:

- When admin upload images for Managed Student Images (See Use Case AD005), those images will be stored in Temp folder.
- At 0.00am, or when the size of Temp folder is bigger than 500MB, the Temp folder will be emptied for free space.

Preconditions: N/A.

Post Conditions:

- **Success:** The Temp folder is emptied.
- **Fail:**No images deleted from temp folder

Main Success Scenario:

Step	Actor Action	System Response
1	- Time hit 0.00am - Size of Temp folder is bigger than 500MB	Delete all the images in Temp Folder

Alternative Scenario: N/A
Exceptions: N/A
Relationships: N/A
Business Rules: N/A

Table 26: <System> Auto Free Storage Space

2.3.2.4 <System> Auto Active Roll Call

Use Case Diagram

Figure 19: <System> Auto Active Roll Call

Use Case Specification

USE CASE – SY004			
Use Case No.	SY004	**Use Case Version**	2.0
Use Case Name	Auto Active Roll Call		
Author	Pham Huy Hoang		
Date	21/09/2013	**Priority**	Normal

Actor: System.
Summary:
This Use Case is used to auto active a roll call
Goal:
The roll call on roll call list will be activated when begin date is current date
Triggers:

- System will auto active a roll call when begin date is current date.
- When a roll call is activated, its status in Roll Call List will be "Active".
- The instructor can only take attendance of an active roll call list.

Preconditions: N/A.
Post Conditions:

- **Success:** System will auto active a roll call.
- **Fail:**No roll call activated

Main Success Scenario:

Step	Actor Action	System Response
1	Roll call begin date is current date. The system auto activate a roll call.	The roll call status become "Active".

		The roll call is show on instructor's phone. The instructor can take attendance of the roll call.

Alternative Scenario: N/A
Exceptions: N/A
Relationships: Manage Roll Call
Business Rules:

- A roll call has 3 status: Created, Active, Inactive.
- Staff manage the info of a roll call, but staff can't active/inactive rollcall manually.
- When a roll call is Active, the staff can only Change the Instructor, or Add/Remove Student.

Table 27: <System> Auto Active Roll Call

2.3.2.5 <System> Auto Inactive Roll Call

Use Case Diagram

Figure 20: <System> Auto Inactive Roll Call

Use Case Specification

USE CASE – SY005			
Use Case No.	SY005	**Use Case Version**	2.0
Use Case Name	Auto Active Roll Call		
Author	Pham Huy Hoang		
Date	21/09/2013	**Priority**	Normal

Actor: System.
Summary:
This Use Case is used to auto inactive a roll call
Goal:
The roll call on roll call list will be inactivated when end date expired.
Triggers:

- System will auto inactive a roll call when end date expired.
- When a roll call is inactivated, its status in Roll Call List will be "Inactive".
- The instructor can't take attendance of an active roll call list. The staff can't edit info of an inactive roll call.

Preconditions: N/A.
Post Conditions:

- **Success:** System will auto inactive a roll call.
- **Fail:**No roll call activated

Main Success Scenario:

Step	Actor Action	System Response
1	Roll call end date expired. The system auto inactivate a	

	roll call.	The roll call status become “Inactive”. The roll call not show instructor’s phone.

Alternative Scenario: N/A
Exceptions: N/A
Relationships: Manage Roll Call
Business Rules:

- A roll call has 3 status: Created, Active, Inactive.
- Staff manage the info of a roll call, but staff can’t active/inactive rollcall manually.
- When a roll call is Inactive, the staff can’t edit the roll call info. The instructor can’t take attendance of an inactive roll call.

Table 28: <System> Auto Inactive Roll Call

2.3.3 <Staff >Overview Use Case

Figure 21: <Staff> Overview Use Case

2.3.3.1 <Staff>Report Roll Call Attendance Rate Use Case Diagram

Figure 22: <Staff> Report Roll Call Attendance Rate

Use Case Specification

USE CASE – SU001			
Use Case No.	SU001	**Use Case Version**	2.0
Use Case Name	Report Roll Call Attendance Rate.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	High

Actor: Staff.
Summary:
Staff can report all roll call attendance rates.
Goal:
A file excel includes: Roll call details, student list with attendance by date.
Triggers:

- Staff can report attendance rate to control student, instructor required...
- Staff must do these step to report attendance rate:
  - On Home page, click on link "Manage Roll Call" in menu sidebar and Manage Roll Call page will be showed.
  - Then, select list roll call by search or select date.
  - Click on button "Roll Call Reports".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: A file excel will auto save in staff computer.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Roll Call" in menu bar.	Redirect to Roll Call List page , include: - Create new roll call: link. - Semester: Drop down list (value: All semester, default value: current roll call). - From –to: date picker (default value: begin date, end date of current semester). - Roll Call Report: button. - Final Exam Student List: button. - Table roll call list, for each roll call:

		• No. : label • Class: label • Subject: label • Instructor: label • Date: label • Time: label • Schedule: link • Student list : link • Edit: link • Check Attendance: link
2	Click "Roll Call Reports" button.	The excel report will auto save in staff computer.

Alternative Scenario: N/A
Exceptions: N/A
Relationships: Login,Manage Roll Call.
Business Rules: N/A

Table 29: <Staff>Report Roll Call Attendance Rate

2.3.3.2 <Staff>Report Student List for Final Exam

Use Case Diagram

Figure 23: <Staff> Report Student List for Final Exam

Use Case Specification

USE CASE – SU002			
Use Case No.	SU002	**Use Case Version**	2.0
Use Case Name	Report Student List for Final Exam.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	High

Actor: Staff.
Summary:
Staff can report all roll call attendance rates.
Goal:
A file excel includes: Roll call details, student list, and student detail with total attendance.
Triggers:

- Staff can report attendance rate for prepare final exam...
- Staff must do these step to report final exam:
  - On Home page, click on link "Manage Roll Call" in menu sidebar and Manage Roll Call page will be showed.

- Then, select list roll call by search or select date.
- Click on button "Roll Call Reports".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: A file excel will auto save in staff computer.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Roll Call" in menu bar.	Redirect to Roll Call List page , include: - Create new roll call: link. - Semester: Drop down list (value: All semester, default value: current roll call). - From –to: date picker (default value: begin date, end date of current semester). - Roll Call Report: button. - Final Exam Student List: button. - Table roll call list, for each roll call: • No. : label • Class: label • Subject: label • Instructor: label • Date: label • Time: label • Schedule: link • Student list : link • Edit: link • Check Attendance: link
2	Click "Final Exam Student Lists" button.	The excel report will auto save in staff computer.

Alternative Scenario: N/A
Exceptions: N/A
Relationships: Login,Manage Roll Call.
Business Rules: N/A

Table 30: <Staff>Report Student List for Final Exam

2.3.3.3 <Staff>Add Roll Call

Use Case Diagram

Figure 24: <Staff>Add Roll Call

Use Case Specification

USE CASE – SU003			
Use Case No.	SU003	**Use Case Version**	2.0
Use Case Name	Add Roll Call.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	High

Actor: Staff.

Summary:

Staff can create a new Roll Call.

Goal:

A new roll call includes: instructor, subject, class, date, time will be added to database.

Triggers:

- Staff can create new roll call for every semester
- Staff must do these step to create new roll call:
  - On Home page, click on link "Manage Roll Call" in menu sidebar and Manage Roll Call page will be showed.
  - Then, choose "Create new Roll Call" link.
  - Then, choose the instructor, subject, class, student, date, time… for class.
  - Click on button "Add".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: A new roll call will be created. The roll call will be displayed on roll call list.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Roll Call" in menu bar.	Redirect to Roll Call List page , include: - Create new roll call: link. - Semester: Drop down list (value: All semester, default value: current roll call). - From –to: date picker (default value: begin date, end date of

		current semester). - Roll Call Report: button. - Final Exam Student List: button. - Table roll call list, for each roll call: • No. : label • Class: label • Subject: label • Instructor: label • Date: label • Time: label • Schedule: link • Student list : link • Edit: link • Check Attendance: link
2	Click "Create new roll call" link.	Display "Create new Roll Call" page with : • Semester: Drop down list (value: All semester from now) • Date picker "Begin date". • Major: Drop down list (value: all active major in database) • Class: Drop down list (value: all active class in database has major selected). • Subject: Drop down list (value: all active subjects in database have major selected.) • Time: Drop down list(value: free time base on subject, class and start date). • Instructor: Drop down list (value: all active instructors in database same type with subject and free in start time, date.) • Button "Add" • Link "Back to list"
3	Input roll call information, click "Add" button.	Add new roll call to database. Redirect Roll call list page.

Alternative Scenario: N/A
Exceptions: N/A
Relationships: Login,Manage Roll Call.
Business Rules:

- Start date must be later than current date; end date must be later than start date. Start date must be Monday. - The class and subject belong to the same major. - When staff selects a class for roll call, all students in that class is assigned to the roll call. - For each semester, the class just studies a subject one times. - In time, a instructor just teaches one class.

Table 31: <Staff>Add Roll Call

2.3.3.4 <Staff>Add Session

Use Case Diagram

Figure 25: <Staff>Add Session

Use Case Specification

USE CASE – SU004			
Use Case No.	SU004	**Use Case Version**	2.0
Use Case Name	Add Session.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	Normal

Actor: Staff.

Summary:

Staff can create new session for roll call.

Goal:

A new session includes: instructor, subject, class, date, time will be added to database.

Triggers:

- Staff can create new session for roll call
- Staff must do these step to create new session:
  - On Home page, click on link "Manage Roll Call" in menu sidebar and Manage Roll Call page will be showed.
  - Then, choose "Change" link. The "Roll call schedule" page will be show.
  - Click on button "Add Session".
  - A pop-up add session will be showed. Fill the information for session.
  - Click button "Add".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: A new session will be created. The session will be displayed on roll call session list.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Roll Call" in menu	

	bar.	Redirect to Roll Call List page , include: - Create new roll call: link. - Semester: Drop down list (value: All semester, default value: current roll call). - From –to: date picker (default value: begin date, end date of current semester). - Roll Call Report: button. - Final Exam Student List: button. - Table roll call list, for each roll call: • No. : label • Class: label • Subject: label • Instructor: label • Date: label • Time: label • Schedule: link • Student list : link • Edit: link • Check Attendance: link
2	Click “Change” link.	Display “Roll Call Schedule” page with : • Subject: label. • Class: label. • Instructor: label • Date : label • Time: label. • Add Session: button • Change Instructor : button • Calendar session: calendar.
3	Click “Add session” button.	Show “Add study session” pop-up include: • Date: Date picker (default value: now). • Start Time: Dropdown list (Default value: The start time of roll call). • Instructor: Dropdown list (Default value: The start time of roll call). • Note: Text box(min length: 1,

		max length: 100, required) • Add: button. • Close: button.
4	Fill the information for new session add click "Add" button. [Alternative 1]	Close the pop-up, the calendar will show with new session.

Alternative Scenario:

Step	Actor Action	System Response
1	The staff click "Close" button.	The pop-up will be closed add no session haven't create.

Exceptions: N/A
Relationships: Login,Manage Roll Call Study Session.
Business Rules:
- The new session will not occur in sun day.
- 2 sessions haven't same date time.

Table 32: <Staff>Add Session

2.3.3.5 <Staff>Change Session Time

Use Case Diagram

Figure 26: <Staff>Change Session Time

Use Case Specification

USE CASE – SU005			
Use Case No.	SU005	**Use Case Version**	2.0
Use Case Name	Change Session Time.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	Normal

Actor: Staff.
Summary:
Staff can change a session time of roll call.
Goal:
A session includes: instructor, subject, class, date, time will be updated to database.
Triggers:
- Staff can change session time of roll call
- Staff must do these step to changesession time:
  - On Home page, click on link "Manage Roll Call" in menu sidebar and Manage Roll Call page will be showed.

- Then, choose “Change” link. The “Roll call schedule” page will be show
- Click on a session staff wants to change.
- A pop-up change session will be showed. Change the information for session.
- Click button “Save”.

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: Thesession will be update. The session will be displayed on roll call session list.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click “Manage Roll Call” in menu bar.	Redirect to Roll Call List page , include: - Create new roll call: link. - Semester: Drop down list (value: All semester, default value: current roll call). - From –to: date picker (default value: begin date, end date of current semester). - Roll Call Report: button. - Final Exam Student List: button. - Table roll call list, for each roll call: • No. : label • Class: label • Subject: label • Instructor: label • Date: label • Time: label • Schedule: link • Student list : link • Edit: link • Check Attendance: link
2	Click “Change” link.	Display “Roll Call Schedule” page with : • Subject: label. • Class: label. • Instructor: label • Date : label • Time: label. • Add Session: button • Change Instructor : button • Calendar session: calendar.

3	In session calendar, click on session you want to change.	Show "Change schedule" pop-up include: • Class: label. • Instructor: label. • Date: Date picker (default value: the date of session). • Start time: Dropdown list (Default value: the start time of session). • Note: (min length: 1, max length: 100, required). • Save: button. • Close: button.
4	Update the information for session add click "Save" button. [Alternative 1]	Close the pop-up, the calendar will show with session updated.

Alternative Scenario:

Step	Actor Action	System Response
1	The staff click "Close" button.	The pop-up will be closed add no session haven't change.

Exceptions: N/A
Relationships: Login,Manage Roll Call Study Session.
Business Rules:
- The new session will not occur in sun day.
- 2 sessions haven't same date time.

Table 33: <Staff>Change Session Time

2.3.3.6 <Staff>Change Session Instructor

Use Case Diagram

Figure 27: <Staff>Change Session Instructor

Use Case Specification

USE CASE – SU006			
Use Case No.	SU006	**Use Case Version**	2.0
Use Case Name	Change Session Instructor.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	Normal

Actor: Staff.

Summary:

Staff can change a session instructor of roll call.

Goal:

Change instructor of roll call.

Triggers:

- Staff can change instructor of roll call.
- Staff must do these step to changesession time:
  - On Home page, click on link "Manage Roll Call" in menu sidebar and Manage Roll Call page will be showed.
  - Then, choose "Change" link. The "Roll call schedule" page will be show
  - Click on "Change Instructor" button.
  - A pop-up change instructor will be showed. Change the information for session.
  - Click button "Save".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: Thesession with new instructor will be update. The session will be displayed on roll call session list.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Roll Call" in menu bar.	Redirect to Roll Call List page , include: - Create new roll call: link. - Semester: Drop down list (value: All semester, default value: current roll call). - From –to: date picker (default value: begin date, end date of current semester). - Roll Call Report: button. - Final Exam Student List: button. - Table roll call list, for each roll call: • No. : label • Class: label • Subject: label • Instructor: label • Date: label • Time: label • Schedule: link

		• Student list : link • Edit: link • Check Attendance: link
2	Click “Change” link.	Display “Roll Call Schedule” page with : • Subject: label. • Class: label. • Instructor: label • Date : label • Time: label. • Add Session: button • Change Instructor : button • Calendar session: calendar.
3	Click “Change Instructor” button.	Show “Change Instructor” pop-up include: • Class: label. • Time: label. • From: Date picker (default value: The next date of now). • To: Date picker (default value: The next date of now). • Instructor: Dropdown list (Default value: instructor of session). • Note: (min length: 1, max length: 100, required). • Save: button. • Close: button.
4	Update the information for session add click “Save” button. [Alternative 1]	Close the pop-up, the calendar will show with session updated.

Alternative Scenario:

Step	Actor Action	System Response
1	The staff click “Close” button.	The pop-up will be closed add no session haven’t change.

Exceptions: N/A
Relationships: Login,Manage Roll Call Study Session.
Business Rules:
- The new session will not occur in sun day.
- 2 sessions haven’t same date time.

Table 34: <Staff>Change Session Instructor

2.3.3.7 <Staff>Edit Roll Call

Use Case Diagram

Figure 28: <Staff>Edit Roll Call

Use Case Specification

USE CASE – SU007			
Use Case No.	SU007	**Use Case Version**	2.0
Use Case Name	Edit Roll Call.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	Normal

Actor: Staff.

Summary:

Staff can use this case to edit a roll call.

Goal:

Update new information roll call includes: instructor, subject, class, date, time…

Triggers:

- Staff can update new information for roll call to database.
- Staff must do these step to edit roll call:
  - In Home Page, click on link "Manage roll call".
  - Click on button "Edit" in row, which staff wants to edit.
  - In "Edit roll call" page, edit information.
  - Click on button "Save".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: A roll call will be updated. The system will transfer to roll call list page.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Roll call" in menu bar.	Redirect to Roll Call List page , include: - Create new roll call: link. - Semester: Drop down list (value: All semester, default value: current roll call).

		- From –to: date picker (default value: begin date, end date of current semester). - Roll Call Report: button. - Final Exam Student List: button. - Table roll call list, for each roll call: • No. : label • Class: label • Subject: label • Instructor: label • Date: label • Time: label • Schedule: link • Student list : link • Edit: link • Check Attendance: link
2	Click link "Edit" in row, which staff wants to edit.	The system will show "Edit roll call" page: • Date picker "Begin date". • Instructor: Drop down list (value: all active instructors in database same type withsubject and free in start time, date.) • Time: Drop down list(value: free time base on subject, class and start date). • Button "Save" • Link "Back to list"
3	Edit roll call information, click "Save" button.	The system will update roll call to database. The system transfer to roll call list page.

Alternative Scenario: N/A
Exceptions: N/A
Relationships: Login,Manage Roll Call.
Business Rules:

- If the roll call status is "Created", the staff can edit all roll call info.
- If the roll call status is "Active" with start less than 2 days from now, the staff can only change the student of roll call.
- If the roll call finishes, the staff cannot edit.

Table 30: <Staff>Edit Roll Call

2.3.3.8 <Staff> Add Subject

Use Case Diagram

Figure 29: <Staff> Add Subject

Use Case Specification

USE CASE – SU008			
Use Case No.	SU008	**Use Case Version**	2.0
Use Case Name	Add Subject.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	Normal

Actor: Staff.

Summary:

Staff can create a new Subject.

Goal:

A new Subject include: major, name, slot will be added to database.

Triggers:

- Staff can create new subject with all information: name, major , slot …
- Staff must do these step to create new subject:
  - On Home page, click on link "Manage Subject" in menu sidebar and Subject list page will be showed.
  - Then, choose "Create new subject" link.
  - Then, choose the subject type, input name, number of session, short name for subject.
  - Click on button "Create".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: A new Subject will be created. The Subject will be displayed on Subject list.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Subject" in menu bar.	Redirect to Subject List page. its include table subject ,for each row: • No. : label • Short name: label • Full name: label • Number of session:label • Number of slot: label

		• Active/Inactive: button • Edit/Delete: link
2	Click “Create new subject” link.	Display “Create new subject” page with : • Short name: textbox (min length: 1, max length: 6, required). • Full name: textbox (min length:1, max length: 30, required). • Number of session (/semester): textbox (min length:1, max length: 30, required). • Number of slot (/day): drop down list (value: 1, 2). • Button “Create”. • Link “Back to list”
3	Input Subject information, click “Add” button.	Create new Subject to database. Redirect Subject list tab. [Exception 1,2]

Alternative Scenario: N/A

Exceptions:

Step	Actor Action	System Response
1	Short name’s length is not in range [1,6]	Display “Short name’s length must be from 1 to 6 words”.
2	Full name’s length is not in range [1,30]	Display “Full name’s length must be from 1 to 30 words”.

Relationships: Login,Manage Subject.

Business Rules:N/A

Table 31: <Staff> Add Subject

2.3.3.9 <Staff> Edit Subject

Use Case Diagram

Figure 30: <Staff> Edit Subject

Use Case Specification

USE CASE – SU009			
Use Case No.	SU009	**Use Case Version**	2.0
Use Case Name	Edit Subject.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	Normal

Actor: Staff.

Summary:

Staff can use this case to edit a Subject.

Goal:

Update new information Subject includes: major, name, slot…

Triggers:

- User can update new information for subject to database.
- Staff must do these step to Edit subject:
  - In Home Page, click on link "Manage subject".
  - Click on button "Edit" in row, which staff wants to edit.
  - In "Edit subject" page, edit information.
  - Click on button "Save".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: A Subject will be updated. The system will transfer to Subject list page.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Subject" in menu bar.	Redirect to Subject List page. its include table subject ,for each row: • No. : label • Short name: label • Full name: label • Number of session: label • Number of slot: label • Active/Inactive: button • Edit/Delete: link

2	Click button “Edit” in row, which staff wants to edit.	Display “Create new subject” page with : • Short name: textbox (min length: 1, max length: 6, required). • Full name: textbox (min length:1, max length: 30, required). • Number of session (/semester): textbox (min length:1, max length: 30, required). • Number of slot (/day): drop down list (value: 1, 2). • Button “Save”. • Link “Back to list”
3	Edit Subject information, click “Save” button.	The system will update Subject to database. The system transfer to Subject list page. [Exception 1,2]

Alternative Scenario: N/A

Exceptions:

Step	Actor Action	System Response
1	Short name’s length is not in range [1,6]	Display “Short name’s length must be from 1 to 6 words”.
2	Full name’s length is not in range [1,30]	Display “Full name’s length must be from 1 to 30 words”.

Relationships: Login,Manage Subject.

Business Rules:

- In “Edit subject” page, the value of major, name, slot will be auto set by subject was chosen.

Table 32: <Staff> Edit Subject

2.3.3.10 <Staff> Active Subject Use Case Diagram

Figure 30: <Staff> Active Subject

Use Case Specification

USE CASE – SU010			
Use Case No.	SU010	**Use Case Version**	2.0
Use Case Name	Active Subject.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	Normal

Actor: Staff.
Summary:
Staff can use this case to active subject.
Goal:
Active a subject for new block.
Triggers:

- Staff can active a subject have exit.
- Staff must do these step to Active subject:
  - In Home Page, click on link "Manage subject".
  - Click on button "Inactive" in row, which staff wants to edit.
  - The system will show a message box confirm.
  - Click on button "Yes".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: A Subject will be active. The system will still in current manage subject page.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Subject" in menu bar.	Redirect to Subject List page. its include table subject ,for each row: • No. : label • Short name: label • Full name: label • Number of session: label • Number of slot: label • Active/Inactive: button • Edit/Delete: link

2	Click button “Inactive” in row, which staff wants to edit.	The system will show confirm message box: • Text: “Are you sure want to active subject:” + name subject selected. • Button “Yes” • Button “No”
3	Click on button “Yes” [Alternative 1]	The system will change status subject to active and still in current page.

Alternative Scenario:

Step	Actor Action	System Response
1	In confirm message box, staff click on button “No”	The system will still in current page with no value is changed.

Exceptions: N/A
Relationships: Login,Manage Subject.
Business Rules:
- The button active has color green and button inactive has color gray.

Table 33: <Staff> Active Subject

2.3.3.11 <Staff> Inactive Subject

Use Case Diagram

Figure 31: <Staff> Inactive Subject

Use Case Specification

USE CASE – SU011			
Use Case No.	SU011	**Use Case Version**	2.0
Use Case Name	Inactive Subject.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	Normal

Actor: Staff.
Summary:
Staff can use this case to inactive subject.
Goal:
Inactive a subject.

Triggers:

- Staff can inactive a subject have exit.
- Staff must do these step to Inactive subject:
  - In Home Page, click on link "Manage subject".
  - Click on button "Active" in row, which staff wants to edit.
  - The system will show a message box confirm.
  - Click on button "Yes".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: A Subject will be inactive. The system will still in current manage subject page.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Subject" in menu bar.	Redirect to Subject List page. its include table subject ,for each row: • No. : label • Short name: label • Full name: label • Number of session: label • Number of slot: label • Active/Inactive: button • Edit/Delete: link
2	Click button "Inactive" in row, which staff wants to edit.	The system will show confirm message box: • Text: "Are you sure want to Inactive subject:" name subject selected. • Button "Yes". • Button "No".
3	Click on button "Yes" [Alternative 1]	The system will change status subject to inactive and still in current page.

Alternative Scenario:

Step	Actor Action	System Response
1	In confirm message box, staff click on button "No"	The system will still in current page with no value is changed.

Exceptions: N/A

Relationships: Login,Manage Subject.

Business Rules:

- The button active has color green and button inactive has color gray.

Table 34: <Staff> Inactive Subject

2.3.3.12 <Staff>Add Student

Use Case Diagram

Figure 32: <Staff> Add Student

Use Case Specification

USE CASE – SU012			
Use Case No.	SU012	**Use Case Version**	2.0
Use Case Name	Add Student.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	Normal

Actor: Staff.

Summary:

Staff can create a new student.

Goal:

A new student include: name, email, student id, birthday, citizen id, address will be added to database.

Triggers:

- Staff can create new student with all information: name, email, student id, birthday, citizen id, address...
  - Staff must do these step to create new student:
    - On Home page, click on link "Manage student" in menu sidebar and Manage student page will be showed.
    - Then, choose "Create new student" link.
    - Then, choose the class, input name, id, email, citizen id, address for student.
    - Click on button "Add".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: A new student will be created. The student will be displayed on student list.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Student" in menu bar.	Redirect to Student List Page, include table student ,for each row:

		• No: label • Class: label • Full name: label • Student code: label • Active/Inactive: button • Attendance report: link. • Change class/ Edit/ Delete: link.
2	Click "Create new student" link.	Display "Create new student " page with : • Name: textbox (min length: 1 max length: 50, required). • Class: Select list (value: all active classes in database). • Birthday: date picker (value: current date.) • Address: textbox (min length: 1 max length: 50, required). • Student ID: : textbox (min length: 1 max length: 50, required). • Email: textbox (match regular expression: ^([\w\.])+@([\w])+\.(\w){2,6}(\.([\w]){2,4})*$, required). • Citizen ID: (match regular expression:^\d{8}$, required). • Button "Add" • Link "Back to list"
3	Input Student information, click "Create" button.	Create new student to database. Redirect student list page. [Exception 1,2,3,4]

Alternative Scenario:N/A

Exceptions:

No	Actor Action	System Response
1	Name's length is not in range [1,30]	Display "Name is invalid".
2	Address's length is not in range [1,50]	Display "Address is invalid".
3	Student ID's length is not in range [1,30]	Display "Student ID is invalid".
4	Email is not matched regular expression.	Display "Invalid email address".

Relationships: Login,Manage Student.

Business Rules: N/A

Table 40: <Staff> Add Student

2.3.3.13 <Staff>Edit Student

Use Case Diagram

Figure 33: <Staff> Edit Student

Use Case Specification

USE CASE – SU013			
Use Case No.	SU013	**Use Case Version**	2.0
Use Case Name	Edit Student.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	Normal

Actor: Staff.

Summary:

Staff can use this case to edit a student.

Goal:

Update new information student includes: name, birthday, citizen ID, email…

Triggers:

- Staff can update new information for student to database.
- Staff must do these step to edit student:
  - In Home Page, click on link "Manage Student".
  - Click on button "Edit" in row, which staff wants to edit.
  - In "Edit student" page, edit information.
  - Click on button "Save".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: A student will be updated. The system will transfer to student list page.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Student" in menu bar.	Redirect to Student List Page, include table student ,for each row: • No: label • Class: label • Full name: label • Student code: label

		• Active/Inactive: button • Attendance report: link. • Change class/ Edit/ Delete: link.
2	Click button "Edit" in row, which staff wants to edit.	Display "Edit student " page with : • Name: textbox (min length: 1 max length: 50, required). • Class: Select list (value: all active classes in database). • Birthday: date picker (value: current date.) • Address: textbox (min length: 1 max length: 50, required). • Student ID: : textbox (min length: 1 max length: 50, required). • Email: textbox (match regular expression: ^([\w\.])+@([\w])+\.(\w){2,6}(\.([\w]){2,4})*$, required). • Citizen ID: (match regular expression:^\d{8}$, required). • Button "Add" • Link "Back to list".
3	Edit student information, click "Save" button.	The system will update student to database. The system transfer to student list page. [Exception 1,2,3,4]

Alternative Scenario:N/A

Exceptions:

No	Actor Action	System Response
1	Name's length is not in range [1,30]	Display "Name is invalid".
2	Address's length is not in range [1,50]	Display "Address is invalid".
3	Student ID's length is not in range [1,30]	Display "Student ID is invalid".
4	Email is not matched regular expression.	Display "Invalid email address".

Relationships: Login,Manage Student.

Business Rules:

- In "Edit student" page, the value of name, current class, address, citizen id, and birthday will be auto set by student was chosen.

Table 41: <Staff> Edit Student

2.3.3.14 <Staff>Import Student List

Use Case Diagram

Figure 34: <Staff> Import Student List

Use Case Specification

USE CASE – SU014			
Use Case No.	SU014	**Use Case Version**	2.0
Use Case Name	Import student list		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	Normal

Actor: Staff.

Summary:

Staff can use this case to import student list.

Goal:

Create multiple student by import from excel file.

Triggers:

- User can import an excel file student list.
- Staff must do these step to import student list:
  - In Home Page, click on link "Import File Excel".
  - Download file template and fill the student list.
  - Then, choose the excel file in browser dialog., upload file.
  - Click button "Import".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: Student list will be added to database.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Import file excel" in menu bar.	Redirect to Import data page, include: • File direction "Choose file": file upload control. • "Here": link down load. • Button "Submit".

2	Click on textbox Import file, choose file. Click "Ok"	The system will get path file and show on text box (only allow excel file.)
3	Click on button "Import"	Import student list to database and show a message box: • Text:"Import is successful! There are [number of student in file] student and [number of class in file] class were imported to database" • Button "Ok". [Exception 1]

Alternative Scenario: N/A

Exceptions:

No	Actor Action	System Response
1	Import file get error.	Show error message, the sheet, line, row where error occur.

Relationships: Login,Manage Student.

Business Rules:

- The input excel file must follow a student list excel template.

Table 42: <Staff> Import Student List

2.3.3.15 <Staff>Add Instructor

Use Case Diagram

Figure 31: <Staff>Add Instructor

Use Case Specification

USE CASE – SU015			
Use Case No.	SU015	**Use Case Version**	2.0
Use Case Name	Add instructor.		
Author	Nguyen Quoc Huy.		

Date	19/09/2013	Priority	Normal

Actor: Staff.

Summary:

Staff can create a new instructor.

Goal:

A new instructor include: name, current subject, current class, and phone number will be added to database.

Triggers:

- Staff can create new instructor with all information: name, current subject , current class, phone number …
- Staff must do these step to create new instructor:
  - On Home page, click on link "Manage Instructor" in menu sidebar and Manage Instructor page will be showed.
  - Then, choose "Create new instructor" page.
  - Then, choose the input name, phone number, select subject type for instructor.
  - Click on button "Add".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: A new instructor will be created. The instructor will be displayed on instructor list.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Instructor" in menu bar.	Redirect to Instructor List page, include table instructor list, for each instructor: • No: label • Full name: label • User: label • Schedule: label • Active/Inactive: button. • Edit/Delete: link.
2	Click "New Instructor" tab.	Display "Create new instructor" page with : • Instructor name: Text box (min length:1 max length:30, required) • Email: textbox (match regular expression: ^([\w\.])+@([\w])+\.(\w){2,6}(\.([\w]){2,4})*$, required). • Phone number: text box (min length: 8, max length: 11, match regular expression: ^\s*\d+\s*$) • Subject type : select list (value: all subject type

		in database) • Button ”Create” • Link “Back to list”
3	Input instructor information, click “Create” button.	Create new instructor to database. Redirect instructor list tab. [Exception 1, 2, 3]

Alternative Scenario: N/A

Exceptions:

No	Actor Action	System Response
1	Name’s length is not in range [1,30]	Display “Name is invalid”.
2	Email not matched regular expression.	Display “Email is invalid”.
3	Phone number is not matched regular expression.	Display “Phone number is invalid”.

Relationships: Login,Manage instructor.

Business Rules: N/A

Table 43: <Staff>Add Instructor

2.3.3.16 <Staff>Edit Instructor

Use Case Diagram

Figure 32: <Staff>Edit Instructor

Use Case Specification

USE CASE – SU016			
Use Case No.	SU016	**Use Case Version**	2.0
Use Case Name	Edit instructor.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	Normal

Actor: Staff.

Summary:

Staff can use this case to edit an instructor.

Goal:

Update new information instructor includes: name, current subjects, current classes, phone…

Triggers:

- Staff can update new information for instructor to database.
- Staff must do these step to edit instructor:
  - In Home Page, click on link “Manage Instructor”.
  - Click on button “Edit” in row, which staff wants to edit.
  - In “Edit instructor” page, edit information.
  - Click on button “Save”.

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: An instructor will be updated. The system will transfer to instructor list page.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click “Manage Instructor” in menu bar.	Redirect to Instructor List page, include table instructor list, for each instructor: • No: label • Full name: label • User: label • Schedule: label • Active/Inactive: button. • Edit/Delete: link.
2	Click button “Detail” in row, which staff wants to edit.	Display “Edit instructor” page with : • Instructor name: Text box (min length:1 max length:30, required) • Email: textbox (match regular expression: ^([\w\.])+@([\w])+\.(\w){2,6}(\.([\w]){2,4})*$, required). • Phone number: text box (min length: 8, max length: 11, match regular expression: ^\s*\d+\s*$) • Subject type : select list (value: all subject type in database) • Button ”Save” • Link “Back to list”
3	Edit instructor information, click “Save” button. [Alternative 1]	The system will update instructor to database. The

		system transfer to instructor list page. [Exception 1, 2]

Alternative Scenario:N/A

Exceptions:

No	Actor Action	System Response
1	Name's length is not in range [1,30]	Display "Name is invalid".
2	Phone number is not matched regular exception.	Display "Phone number is invalid".

Relationships: Login,Manage Instructor.

Business Rules:

- In Edit instructor page, the value of name, current class, current subject, phone number will be auto set by instructor was chosen.

Table 44: <Staff> Edit Instructor

2.3.3.17 <Staff>Add Class

Use Case Diagram

Figure 33: <Staff> Add Class

Use Case Specification

USE CASE – SU017			
Use Case No.	SU017	**Use Case Version**	2.0
Use Case Name	Add Class.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	Normal

Actor: Staff.

Summary:

Staff can create a new Class.

Goal:

A new class includes: major, name will be added to database.

Triggers:

- Staff can create new class with for every semester…
- Staff must do these step to create new class:
  - On Home page, click on link "Manage Class" in menu sidebar and Manage subject page will be showed.
  - Then, choose "Create new class" link.
  - Then, choose the major, input name for class.
  - Click on button "Create".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: A new class will be created. The class will be displayed on Class list.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Class" in menu bar.	Redirect to Class list Page, include table class list, for each class: • No: label • Major: label • Class name: label • Student List: link • Active/Inactive: button • Edit/ Delete : link
2	Click "Create new class" link.	Display "Create new class" page with : • Major: Drop down list (value: Software Engineering, Financial Banking, Business Analysis) • Name: textbox (min length: 1 max length: 30, required). • Button: "Create" • Link: "Back to list"
3	Input class information, click "Create" button.	Create new class to database. Redirect Class list tab. [Exception 1, 2]

Alternative Scenario:N/A

Exceptions:

Step	Actor Action	System Response
1	Name's length is not in range [1,30]	Display "Name is invalid".
2	The name class has existed.	Display "The class has existed".

Relationships: Login, Manage Class.

Business Rules: N/A

Table 35: <Staff> Add Class

2.3.3.18 <Staff>Edit Class

Use Case Diagram

Figure 34: <Staff> Edit Class

Use Case Specification

USE CASE – SU018			
Use Case No.	SU018	**Use Case Version**	2.0
Use Case Name	Edit Class.		
Author	Nguyen Quoc Huy.		
Date	19/09/2013	**Priority**	Normal

Actor: Staff.
Summary:
Staff can use this case to edit a class.
Goal:
Update new information class includes: major, name …
Triggers:

- User can update new information for class to database.
- Staff must do these step to Edit class:
  - In Home Page, click on link "Manage Class".
  - Click on button "Edit" in row, which staff wants to edit.
  - In "Edit class" page, edit information.
  - Click on button "Save".

Preconditions:

- User must logged in the system with the role is staff.

Post Conditions:

- **Success**: A class will be updated. The system will transfer to class list page.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Manage Class" in menu bar.	Redirect to Class list Page, include table class list, for each class: • No: label • Major: label • Class name: label • Student List: link • Active/Inactive: button • Edit/ Delete : link
2	Click button "Edit" in row, which staff wants to edit.	

		Display “Edit class” page with : • Major: Drop down list (value: Software Engineering, Financial Banking, Business Analysis) • Name: textbox (min length: 1 max length: 30, required). • Button: “Save” • Link: “Back to list”
3	Edit Class information, click “Save” button. [Exception 1]	The system will update class to database. The system transfer to class list page.

Alternative Scenario:N/A

Exceptions:

No	Actor Action	System Response
1	Name’s length is not in range [1,30]	Display “Name is invalid”.

Relationships: Login,Manage Class.

Business Rules:

- In Edit class page, the value of major, name will be auto set by class was chosen.

Table 36<Staff> Edit Class

2.3.4 <Instructor>Overview Use Case

Figure 35: <Instructor> Overview Use Case

2.3.4.1 <Instructor>Capture and Show result attendance

Figure 36: <Instructor>Capture and Show result attendance

USE CASE – IU001			
Use Case No.	IU001	**Use Case Version**	2.0
Use Case Name	Capture and Show result attendance		
Author	Nguyen Thanh Binh		
Date	23/09/2013	**Priority**	High

Actor: Instructor.

Summary:

Instructor uses this case to check attendance.

Goal:

Allow instructorcapturecapture image, send image to web service and check attendance.

Triggers:

- Instructor can use mobile to take picture. This picture will be used for attendance checking ... To check attendance using picture, instructor must do:
- Instructor login application and click button "Take Attendance" in Course Activity. System transfer to Camera activity.
- On Camera activity, Instructor capture image, then send image to web service for detected face of member.
- Instructor receive result from Web service. Instructor view result and check attendance for members.

Preconditions:

- User must logged in the system with the role is Instructor
- Instructor is assign to class

Post Conditions:

- **Success**: Instructor can check attendance for student. The attendance log will be stored.
- **Fail:** System will alert error.

Main Success Scenario:

Step	Actor Action	System Response
1	Instructor open mobile app. Instructor type username, password, press "Login" to login into application.	

		System will show the Course activity with information includes: • Subject: textview • Class name: textview • Class Time: textview • Class date: textview **Main menu:** • Take Attendance: Button
2	Instructor click button "Take Attendance" to capture picture. [Alternative 1]	System transfer to Camera activity with: • Camera screen view: Frame layout • Capture: Button
3	Instructor look at screen and click "Capture" button to capture image and send image to web service.	System will save image on real time when instructor click button "Capture". Image will show on screen with two button: Retake & Ok. [Exception 1,2]
4	Instructor click "OK" button to return to Roll Call activity. [Alternative 2]	System will notify result to Instructor. Result contains the students detected from image. [Exception 3, 4, 5]

Alternative Scenario:

Step	Actor Action	System Response
1	Instructor click one Roll Call to check attendance manual with Use Case "IU002".	System will transfer to Roll Call activity.
2	Instructor click Retake "button" to retake an image.	System will transfer to Camera activity without saving image.

Exceptions:

Step	Actor Action	System Response
1	Error with camera (No camera, can't connect to camera)	System will show popup "Camera is not available".
2	Can't send image to	System will show a popup "Failed to connect

	service	server, please check you internet connection and try it again".Instructor click "OK" button to retake an image.
3	No respond from service	System will show popup "Request timeout. Please try again".
4	Only some student detected (<30% number of class students)	System will vibrate, soundandand notify "Only X student detect. You can take more picture, or check attendance manually (See Use Case IU002)".
5	Stranger detected	System will vibrate, sound and notify "Stranger Detected".

Relationships: Login

Business Rules:

- Image : min 1500px * 1000px max 3552px * 2000px or size smaller than 3mb; resolution 72dpi (recommend)
- Cable: 3mb ~ 6mb, Wifi: 3mb ~ 6mb
- The average time for attendance taking is 10 seconds.
- In success case, the app will notify normally, without sound or vibration. In exception case, the notification is with sound and vibration.

Table 37: <Instructor> Check Attendance by Taking Picture

2.3.4.2 <Instructor> Check Attendance Manually

Figure 37: <Instructor> Check Attendance Manually

USE CASE – IU002			
Use Case No.	IU002	**Use Case Version**	2.0
Use Case Name	Check Attendance Manually		
Author	Nguyen Thanh Binh		
Date	23/09/2013	**Priority**	High

Actor: Instructor.

Summary:

Instructor uses this case to check attendance manually.

Goal:

Allow instructor check attendance again for somestudent have special reasons.

Triggers:

- Sometimes, the auto attendance checking is not accuracy, the instructor must check attendance manually.
- To check attendance again manually, instructor must do:

- Instructor login application and choose one roll call in “Course”activity. System transfer to “Roll Call” activity.
- Instructor review all members in List student and click button “Log Attendance” transfer to Log activity.
- Instructor review click on Log that want to review and check attendance again. The system will show the popup have information about log.
- Instructor tick choose the member to check attendance.

Preconditions:

- User must logged in the system with the role is Instructor.
- Instructor is assign to class.

Post Conditions:

- **Success**: Instructor can check attendance for member on this day or previous day.
- **Fail:**System will alert error.

Main Success Scenario:

Step	Actor Action	System Response
1	Instructor open mobile app. Instructor type username, password, press “Login” to login into application.	System will show the Course activity with information includes: • Subject: textview • Class name: textview • Class Time: textview • Class date: textview **Main menu:** • Take Attendance: Button
2	Instructor click one “Roll Call” row to view detail of that Roll Call. [Alternative 1]	System will transfer to Roll Call activity with: • Class Name: textview • Subject: textview • Date & Time: textview • Student List: (Name (textview), Code (textview))
3	Instructor click “Log Attendance” button with correlative date.	System will show Log detail page on correlative with : • Date: textview

		• Present: textview
4	Instructor click one "Log" row to view detail of that Log.	System will show popup with: • Date: textview • Student Name: textview • Checkbox: checkbox
5	Instructor check the student to check attendance. [Alternative 2] Click button "Ok" send request to check attendance for present member. [Alternative 3]	System will send request check attendance for present student to server.

Alternative Scenario:

Step	Actor Action	System Response
1	Instructor click button "Take Attendance" to check attendance with Use Case "IU001".	System will transfer to Camera activity.
2	Instructor uncheck student.	System will set absent for correlative student.
3	Instructor click "Cancel" button	System will close the popup without saving result.

Exceptions:

Step	Actor Action	System Response
1	Can't connection to service	System will show a popup "Failed to connect server, please check you internet connection and try it again".Instructor click "OK" button to transfer Roll Call page.
2	No respond from service	System will show popup "Request timeout. Please try again".

Relationships: Login

Business Rules:

- Instructor can check present member again for student have special reason.

Table 38: <Instructor> Check Attendance Manually

2.3.4.3 <Instructor> View Roll Call Detail

Figure 38: <Instructor> View Roll Call Detail

USE CASE – IU003			
Use Case No.	IU003	**Use Case Version**	2.0
Use Case Name	View Roll Call Detail		
Author	Nguyen Thanh Binh		
Date	23/09/2013	**Priority**	Normal

Actor: Instructor.

Summary:

Instructor uses this case to view all roll call detail.

Goal:

Allow instructorviewview roll call detail with selected roll call, class.

Triggers:

- Instructor login application and choose one roll call in "Course"activity. System transfer to "Roll Call" activity.
- Instructor review all members with course, class and selected day.

Preconditions:

- User must logged in the system with the role is Instructor.
- Instructor is assign to class.

Post Conditions:

- **Success**: Instructor can view list of students with selected roll call, class.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Instructor open mobile app. Instructor type username, password, press "Login" to login into application.	System will show the Course activity with information includes: • Subject: textview • Class name: textview

		• Class Time: textview • Class date: textview **Main menu:** • Take Attendance: Button
2	Instructor click one "Roll Call" row to view detail of that Roll Call. [Alternative 1]	System will transfer to Roll Call activity with: • Class Name: textview • Subject: textview • Date & Time: textview • Student List: (Name (textview), Code (textview))

Alternative Scenario:

Step	Actor Action	System Response
1	Instructor click button "Take Attendance" to check attendance with Use Case "IU001".	System will transfer to Camera activity.

Exceptions:

Step	Actor Action	System Response
1	Can't connection to service	System will show a popup "Failed to connect server, please check you internet connection and try it again".Instructor click "OK" button to transfer Course page.

Relationships: Login,Check Attendance Manually

Business Rules:

- Instructor can view all roll call detail, help to review status of student and check attendance again for students have special reason.

Table 39: <Instructor> View Roll Call List

2.3.4.4 <Instructor> Report Attendance by Class

Figure 39: <Instructor> Report Attendance by Class

USE CASE – IU004			
Use Case No.	IU004	**Use Case Version**	2.0
Use Case Name	Report Attendance by Class		
Author	Nguyen Thanh Binh.		
Date	23/09/2013	**Priority**	Normal

Actor: Instructor.

Summary:

This Use Case allows Instructor report attendance of all students by class.

Goal:

Instructor can report attendance of class.

Triggers:

- Instructor can report information of class: subject, student, attendance…
- Instructor must do these step to view report result:
- On Course page, click on link "Report" in menu bar and Report page will be showed.
- Then, in select box "Report by" choose type: Class.
- Then, in drop down list "Name" chose the class name.
- Click on button "Statistic".

Preconditions:

- User must logged in the system with the role is Instructor.

Post Conditions:

- **Success**: A report attendance by class will be showed on screen.
- **Fail:** System will not return any value. Transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Click "Report" link in menu bar.	Redirect to Report Page, include 4 part: • Drop down list "Report by" (value: Class, Student, Block, and Semester). • Drop down list "Name"

		(value: auto set based on drop down list Report by) • Button “Statistic” • Button “Export”.
2	Choose “Class” in drop down list Report by.	System will auto set value based on type Class.
3	Chose the class in drop down list “Name”. Click button “Statistic”.	System will show result attendance by class on screen. System still in current “Report” page.

Alternative Scenario: N/A
Exceptions: N/A
Relationships: Login,Manage Report Attendance.
Business Rules:
- System will show attendance result in chosen conditions.
- Default value when manager clicks on tab “Report” is: Class.

Table 40: <Instructor> Report Attendance by Class

2.3.5 <Student>Overview Use Case

Figure 40: <Student> Overview Use Case

2.3.5.1 <Student>Check Present Rate by Searched Course

Figure 41: <Student>CheckPresentPresent Rate

USE CASE – STU001			
Use Case No.	STU001	**Use Case Version**	2.0
Use Case Name	Check Present Rate by Searched Course		
Author	Nguyen Thanh Binh		
Date	23/09/2013	**Priority**	Normal

Actor: Student.

Summary:

Student uses this case to search learned/learning course.

Goal:

Allow student views learned/learning course info. The search condition is course name.

Triggers:

- Student can search course name on list course page. The searching results follow course name.
- On list courses page, student choose search condition and clicks search button. System will show list search in list course table at current page.
- On list search page, student choose the course. System will show information detail about that course.

Preconditions:

- User must logged in the system with the role is Student.

Post Conditions:

- **Success**: The search course will be found. System will show list search in lists course page.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	At lists course page, student choose search condition.	System will show the search condition include: Course Name: text box
2	After chose search condition, student clicks on search button (submit	

	button) finish.	System will show list search of courses that student searched by condition in List Course table. List Courses table includes 6 column: No (number), Subject, Class, Instructor, Time, Attendance.
3	Ai list course table, student choose the course they want to view and click on that course name.	System will show information about that course in the popup on current page. Information course includes: Student Name, Roll Call, Subject, Class, Total Session, Completed Session, Absent, Absent Rate and Attendance Table. Attendance table with 4 column: No, Date, Present, Note.

Alternative Scenario: N/A

Exceptions:

Step	Actor Action	System Response
1	Search not found	System will show "No course was found!"

Relationships: Login,Check Present Rate

Business Rules:

- The courses will be search by course name.
- Student can use search course to view their course easily.

Table 41: <Student> CheckPresent Rate

2.3.5.2 <Student>Check Present Rate by Learned Course

Figure 42: <Student>Check Present Rate by Learned Course

USE CASE – STU002			
Use Case No.	STU002	**Use Case Version**	2.0
Use Case Name	Check Present Rate by Learned Course		
Author	Nguyen Thanh Binh		

Date	23/09/2013	Priority	Normal

Actor: Student.

Summary:

Student uses this case to search learned course.

Goal:

Allow student views learned course info. The search condition is semester.

Triggers:

- On list learned courses page, student choose search condition. System will show list search in list course table at current page.
- On list search page, student choose the course. System will show information detail about that course.

Preconditions:

- User must logged in the system with the role is Student.

Post Conditions:

- **Success**: The search course will be found. System will show list search in lists course page.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	At lists course page, student choose search condition.	System will show the search condition include: Semester: drop down list
2	After chose search condition, student view list learned course will be show on current page.	System will show list search of courses that student searched by condition in List Course table. List Courses table includes 6 column: No (number), Subject, Class, Instructor, Time, Attendance.
3	Ai list course table, student choose the course they want to view and click on that course name.	System will show information about that course in the popup on current page. Information course includes: Student Name, Roll Call, Subject, Class, Total Session, Completed Session, Absent, Absent Rate and Attendance Table. Attendance table with 4 column: No, Date, Present, Note.

Alternative Scenario: N/A

Exceptions: N/A

Relationships: Login,Check Present Rate **Business Rules:** - The learned courses will be search by semester. - Student can use search course to view their course easily.

Table 42: <Student> Check Present Rate By Learned Course

2.3.5.3 <Student>Check Present Rate by Learning Course

Figure 43: <Student>Check Present Rate by Learning Course

USE CASE – STU003			
Use Case No.	STU003	**Use Case Version**	2.0
Use Case Name	Check Present Rate by Learning Course		
Author	Nguyen Thanh Binh		
Date	23/09/2013	**Priority**	Normal

Actor: Student.
Summary:
Student uses this case to search learning course.
Goal:
Allow student views learning course info.
Triggers:
- On list learning courses page student choose the course. System will show information detail about that course.

Preconditions:
- User must logged in the system with the role is Student.

Post Conditions:
- **Success**: The search course will be found. System will show list search in lists course page.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	Student choose view Learning Course page.	System will show list learning courses. List Courses table includes 6 column: No (number), Subject, Class, Instructor, Time, Attendance.
2	Ai list course table, student choose the course they	

	want to view and click on that course name.	System will show information about that course in the popup on current page. Information course includes: Student Name, Roll Call, Subject, Class, Total Session, Completed Session, Absent, Absent Rate and Attendance Table. Attendance table with 4 column: No, Date, Present, Note.

Alternative Scenario: N/A
Exceptions: N/A
Relationships: Login,Check Present Rate
Business Rules:
- Student can use search course to view their course easily.

Table 43: <Student> Check Present Rate by Learning Course

2.3.5.4 <Student>Upload Image as Request

Figure 44: <Student> Upload Image as Request

USE CASE – STU004			
Use Case No.	STU004	**Use Case Version**	2.0
Use Case Name	Upload Images as Request		
Author	Nguyễn Quốc Huy		
Date	23/09/2013	**Priority**	Normal

Actor: Student.
Summary:
Student uses this case to upload images to their own image data.
Goal:
Allow student to upload images as request from admin
Triggers:
- When the admin create a request for student, the student can see the request when they login.
- To upload image as request, the student must:
  - Click the "Upload Images" button under the request.
  - Upload images. System will detect human faces from uploaded images.
  - Select the own faces from images.
  - Press "Submit Result" to save the images.

Preconditions:
- User must logged in the system with the role is Student.

Post Conditions:

- **Success**: The request and images will be sent to admin.
- **Fail:**No request sent.

Main Success Scenario:

Step	Actor Action	System Response
1	From the Student Home Page, click "Upload Image" button.	Redirect to Request Page, contains: • File Upload Control • Upload: button
2	Select the own images from computer.	The selected images will be displayed as thumbnails. [Exception 1]
3	Click "Upload" button.	System will detect student faces from uploaded images, then display image full size with white **face regions** (See Use Case SY001) [Exception 2, 3]
4	Select student's face from face region. Click "Submit Result" button.	The request and images are sent. [Exception 4] Redirect back to Student Home Page.

Alternative Scenario: N/A

Exceptions:

No	Actor Action	System Response
1	Select more than 20 images	Display error message: "Too much images. Only process first 20 images"
2	Not select image, click "Upload" button	Display error message: "Please choose images for uploading"
3	- Upload images don't contain human faces. Or -No faces detected from uploaded images	Display error message: "No face found. Select another images"
4	The selected face already in the database	Display message: "Face XXX existed in database"

Relationships: Login,Manage Image Uploading Request

Business Rules:

- Student can use search course to view their course easily.

Table 44: <Student>Upload Images as Request

2.3.6 <Guest>Overview Use Case

Figure 45: <Guest> Overview Use Case

2.3.6.1 <Guest> Login

Use Case Diagram

Figure 46: <Guest> Login

Use Case Specification

USE CASE – GU001			
Use Case No.	GU001	**Use Case Version**	2.0
Use Case Name	Login		
Author	Pham Huy Hoang		
Date	19/09/2013	**Priority**	Normal

Actor: Guest.
Summary:
Guest use this case to login into system.
Goal:
Allow authentication andauthorizationauthorization of the system.
Triggers:
- Guest want to login into system.
- To login: Guest go to the login page, guest enter username and password, then click on "Log in" button to login.

Preconditions: N/A.
Post Conditions:
- **Success:** Guest is authorized, redirect to corresponding page.
- **Fail:**System will show error on current page

Main Success Scenario:

Step	Actor Action	System Response
1	Guest go to login page.	System show a login form, includes: • Username: textbox (min length: 5, max length: 30, required). • Password: textbox (password box, min length:5, max length: 30,

		required). • Login: button.
1	Guess enter username and password into textboxs. Press “Log In” button.	Guest is logged into system, authorized with corresponding role. Redirect to corresponding page.. [Exception 1,2]

Alternative Scenario: N/A

Exceptions:

No	Actor Action	System Response
1	No input in “Username” or “Password” textboxs	Show error message: “Please enter username/password” below Username/Password textbox.
2	Username/Password not in range [5,30]	Show error message: “Username/Password must be from 5 to 30 characters” textbox.
3	Input invalid “Username” and “Password”	Show error message: “Invalid username or password”.

Relationships: N/A

Business Rules:

- Each student, instructor, staff, admin has an account.
- Only active account can log in.
  - Only instructor’s account can log in into mobile app.

Table 45: <Guest> Login

2.3.7 <Authorized User> Overview Use Case

2.3.7.1 <Authorized User> Overview Use Case

Figure 47: < Authorized User> Overview Use Case

2.3.7.2 <Authorized User> Change Password

Figure 48: < Authorized User> Change Password

USE CASE – UU001			
Use Case No.	UU001	**Use Case Version**	2.0
Use Case Name	Change Password		
Author	Nguyen Thanh Binh		
Date	23/09/2013	**Priority**	Normal

Actor: User (Instructor, Staff, Student, Admin).

Summary:

- User users this case to change their password.

Goal:

- Allow users can change their password.

Triggers:

- User use this Use Case to help improve their account security.
- User login their account, click "Change Password" link. Go to "Change Password" page.

Preconditions:

- User must logged in the system with the role is User (Staff, Instructor, Student, Admin).

Post Conditions:

- **Success**: The password be changed.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	User selects the "Change Password" command in the top-right column of Home page.	System will show "Change Password" page includes: • Username – Text box • Current Password – Text box • New Password – Text box • Confirm New Password – Text box • Submit – Button • Reset - Button
2	- User enters the following information (Username, Current Password, New password, Confirm new password). - Click "Submit" button to send request.	• System changes the user's password. • System automatically re-authenticates the user with the new credentials. • System will show "Your

		password was changed successfully!”

Alternative Scenario: N/A
Exceptions:

Step	Actor Action	System Response
1	System verifies that the Username and Current Password entries are correct and correspond to a current user.	System will show “Wrong username/ password” and reset all.
2	New password same current password	System will show “New password was used! Please choose new password” and reset all.
3	New password and confirm new password does not match.	System will show “Password does not match” and reset all.

Relationships: Login
Business Rules:N/A

Table 46: < Authorized User> Change Password

2.3.7.3 <Authorized User>Logout

Figure 49: < Authorized User> Logout

USE CASE – UU002			
Use Case No.	UU002	**Use Case Version**	2.0
Use Case Name	Change Password		
Author	Nguyen Thanh Binh		
Date	23/09/2013	**Priority**	Normal

Actor: User (Instructor, Staff, Student, Admin).
Summary:

- User use this Use Case to logout system.

Goal:

- The user selects the option to log off of the HCPT website and is then redirected back to the Login page.

Triggers:

- The Use Case starts when the user selects the option to logout.
  - The user selects the option to logout.
  - The system informs the user of logging out.
  - The user confirms logging out.
  - The system redirects user back to the Login page.

Preconditions:

- User must logged in the system with the role is User (Staff, Instructor,

Student, Admin).

Post Conditions:

- **Success**: User logout system and redirect to login page.
- **Fail:** System will transfer to error page or still in current page.

Main Success Scenario:

Step	Actor Action	System Response
1	The user selects log out.	The system will show a pop-up: Do you really want to logout" • Yes: button • No: button
2	User select "Yes". [Alternative 1]	The system will clear user session and the user is logged out

Alternative Scenario:

Step	Actor Action	System Response
1	Userselects No.	User stay at current page.

Exceptions: N/A

Relationships: Login

Business Rules: N/A

Table 47: <Authorized User> Logout

3. Software System Attribute

3.1 Usability

3.1.1 Graphic User Interface

- All the text, label and image in staff, instructor and student page should be English.
- All the text, label and image in admin page should be English.

3.1.2 Usability

- Website admin, staff and instructor should need more than one week of training to be productive with the system.
- The student can use the system easily without training.

3.1.3 Installation

- The system must be easy to deploy. Customer can deploy successfully and learn to configure, maintain the system within one day of training.
- The mobile app must be easy to install. Compatible with almost Android Phone with Android 4.0 or above.
- The attached manual guide must be clear. User can read and do themselves without developer's help.

3.2 Reliability

- The accuracy of face recognizer algorithm is about 80-90%, with the image data required for each student is more than 10 images.
- These results are achieved when user follow the guideline "**User Notice**" in **Software User's Manual.**

3.3 Availability

- The instructor can take attendance offline using mobile in case of network or server unavailable.

3.4 Security

- Privacy: Each role of user has a specific permission to interact with system.
- System always checks authorization and authenticated before doing anything.

3.5 Maintainability

- The system is divided into sepearate modules.
- The code is easy to maintain and upgrade.

3.6 Portability

- N/A

3.7 Performance

- Detect Faces From an Image: 3~4 seconds/image. Resolution of image is 2000px X 3000px, size 3~4 MB. Smaller images will be processed faster.
- Recognize Faces: 20~30 seconds/100 faces. With training set of 30 students, 600 images, size of each training image is 100px X 100px.

4. Conceptual Diagram

Figure 50: Conceptual Diagram

Data Dictionary

Entity Data dictionary: describe content of all entities	
Entity Name	**Description**
Semester	Describe all semesters in school system. Each semester last 4-5 months.
Major	Describe all majors in school system. (Current major: Software Engineering, Financial Banking, Business Administrator …)
Subject	Describe all subjects in school system, A subject can be taught in many majors
SubjectType	Describe type of subject (IT, economic, language ...)
Class	Describe all class in school system. Each class belong to one major
Student	Describe all students in school system.
StudentImage	Describe all images of each student.
Instructor	Describe all instructors in school system. An instructor can teach many types of subject.
RollCall	Describe all roll calls in school system. A roll call contains info: Class, Subject, begin-end Time…..
StudySession	Describe all study session of roll call. A roll call has many study sessions.
AttendanceLog	Describe all all the attendance log of roll call. A log contains the present/absent students in a date of a roll call.
LogImage	Describe all images of attendance log.
LogType	Describe all type of attendance log (Auto, Manual).
Request	Describe all request send from admin to student.
RequestImage	Describe all images of request.
Staff	Describe all staffs in school system.
User	Describe account of user of system. User include: admin, staff, student …
Role	Describe role of user. One user has one role.

D.Software Design Description

1. Design Overview

- This document describes the technical and user interface design of The Roll System using mobile device. 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 include class diagram, class explanation, and sequence diagram to details the application functions.
    - Section 5: describe an ERD with logical diagram.

2. System Architectural Design

Figure 51: Multi-tier Architecture (N-tier Architecture)

(http://en.wikipedia.org/wiki/Multitier_architecture)[3]

3. Component Diagram

Figure 52: Component Diagram

4. Detailed Description of Components

4.1 Class Diagram

Figure 53: Class Diagram

4.2 Class Diagram Explanation

4.2.1 RollCall

Attribute

Attribute	Type	Visibility	Description
RollCallID	Int	Public	Unique id of each roll call
StartTime	Time	Public	Start time of roll call
EndTime	Time	Public	End time of roll call
BeginDate	Date	Public	Begin date of roll call
EndDate	Date	Public	End date of roll call
SubjectID	Int	Public	ID of subject of roll call
InstructorID	Int	Public	ID of the main instructor of the roll call
ClassID	Int	Public	ID of class of roll call
SemesterID	Int	Public	ID of semester of roll call
Status	Int	Public	Status of roll call

Method

Method	Return Type	Visibility	Description
getAllRollCalls	List<RollCall>	Public	Get all roll call in system
getRollCallbyID	RollCall	Public	Get roll call in system by id
getRollCallBySemester	List<RollCall>	Public	Get roll call in system by semester

4.2.2 Subject

Attribute

Attribute	Type	Visibility	Description
SubjectID	Int	Public	Unique id of each subject
FullName	String	Public	Full name of subject
ShortName	String	Public	Short name of subject
Description	String	Public	Description of subject
NumberOfSlot	Int	Public	Number of slot of subject
NumberOfSession	Int	Public	Number of slot of

			subject
IsActive	Boolean	Public	Status active of subject
IsSlotFixed	Booean	Public	Number of slot of special subject

Method

Method	Return Type	Visibility	Description
GetActiveSubjects	List<Subject>	Public	Get all active subjects
GetAllSubjects	List<Subject>	Public	Get all subjects in system
GetSubjectByID	Subject	Public	Get subject in system by id

4.2.3 SubjectType

Attribute

Attribute	Type	Visibility	Description
TypeID	Int	Public	Unique id of each subject
TypeName	String	Public	Name of subjecttypetype

Method

Method	Return Type	Visibility	Description
GetType	SubjectType	Public	Get type of subject

4.2.4 Major

Attribute

Attribute	Type	Visibility	Description
MajorID	Int	Public	Unique id of each major
ShortName	String	Public	Short name of major
FullName	String	Public	Full name of major

Method

Method	Return Type	Visibility	Description
AddMajor	Void	Public	Add new Major to database

4.2.5 Class

Attribute

Attribute	Type	Visibility	Description
ClassID	Int	Public	Unique id of each class
MajorID	Int	Public	Int of major of class
ClassName	String	Public	Name of class

isActive	Boolean	Public	Status active of class

Method

Method	Return Type	Visibility	Description
GetActiveClasses	List<Class>	Public	Get all classes are active in system
GetClassByID	Class	Public	Get class in system by id

4.2.6 StudySession

Attribute

Attribute	Type	Visibility	Description
ID	Int	Public	Unique id of each study sesion
InstructorID	Int	Public	ID of the instructor who teach study sesion
RollCallID	Int	Public	ID of roll call of instructor teaching
ClassID	Int	Public	ID of the class learn study sesion
SessionDate	Date	Public	Date of study sesion
StartTime	Time	Public	Start time of study session
EndTime	Time	Public	End date of study session

Method

Method	Return Type	Visibility	Description
GetSessionByID	StudySession	Public	Get the session in database by ID
GetInstructorSession	List<StudySession>	Public	Get all the sessions of an instructor

4.2.7 Instructor

Attribute

Attribute	Type	Visibility	Description
InstructorID	Int	Public	Unique id of each instructor
FullName	String	Public	Full name of instructor
Email	String	Public	Email of instructor
Phone	String	Public	Phone of instructor
UserID	Int	Public	ID of the user
ApiToken	String	Public	The API token of Google Calendar

			API
IsActive	Boolean	Public	Status active of instructor

Method

Method	Return Type	Visibility	Description
GetActiveInstructors	List<Instructor>	Public	Get all active instructors in system
GetAllInstructors	List<Instructor>	Public	Get all instructors in system
GetInstructorByID	Instructor	Public	Get instructor in system by id

4.2.8 Semester

Attribute

Attribute	Type	Visibility	Description
SemesterID	Int	Public	Unique id of each semester
SemesterName	String	Public	Name of semester
BeginDate	Date	Public	Begin date of semester
EndDate	Date	Public	End date of semester

Method

Method	Return Type	Visibility	Description
GetAllSemesters	List<Semester>	Public	Get all semesters in system
GetSemesterByID	Semester	Public	Get semester in system by id

4.2.9 Student

Attribute

Attribute	Type	Visibility	Description
StudentID	Int	Public	Unique id of each student
ClassID	Int	Public	ID of the class of student
FullName	String	Public	Full name of student
Birthday	Date	Public	Birthday of student
Address	String	Public	Address of student
CitizenID	String	Public	Citizen id of student
Email	String	Public	Email of student
UserID	User	Public	User id of student
IsActive	Boolean	Public	Status active of

			student
StudentCode	String	Public	Code of student

Method

Method	Return Type	Visibility	Description
GetActiveStudents	List<Student>	Public	Get all active students in system
GetAllStudents	List<Student>	Public	Get all students in system
GetStudentByID	Student	Public	Get student in system by id

4.2.10 Role

Attribute

Attribute	Type	Visibility	Description
RoleID	Int	Public	Unique id of each role
RoleName	String	Public	Name of role

Method

Method	Return Type	Visibility	Description
GetRole	Role	Public	Get role in system.

4.2.11 Staff

Attribute

Attribute	Type	Visibility	Description
StaffID	Int	Public	Unique id of each staff
FullName	String	Public	Full name of staff
Email	String	Public	Email of staff
Phone	String	Public	Phone number of staff
UserID	Int	Public	ID of the user
Isactive	Boolean	Public	Status active of staff

Method

Method	Return Type	Visibility	Description
GetActiveStaffs	List<Staff>	Public	Get all active staffs in system
GetAllStaffs	List<Staff>	Public	Get all staff in system
GetStaffByID	Staff	Public	Get staff in system by id

4.2.12 User

Attribute

Attribute	Type	Visibility	Description

UserID	Int	Public	Unique id of each user
UserName	String	Public	Username of user
Password	String	Public	Password of user
RoleID	Int	Public	ID of the roll of system
IsActive	Boolean	Public	Status active of user

Method

Method	Return Type	Visibility	Description
GetActiveUsers	List<User>	Public	Get all active users in system
GetAllUsers	List<User>	Public	Get all users in system
GetUserByID	User	Public	Get user in system by id

4.2.13 StudentImage

Attribute

Attribute	Type	Visibility	Description
ImageID	Int	Public	Unique id of each image
StudentID	Int	Public	ID of the image's student
ImageLink	String	Public	Link of image

Method

Method	Return Type	Visibility	Description
GetStudentImages	List<StudentImage>	Public	Get images of student in system

4.2.14 StudentAttendance

Attribute

Attribute	Type	Visibility	Description
LogID	Int	Public	Log of student attendance
StudentID	Int	Public	ID of student of attendance
IsPresent	Boolean	Public	Status present of student
Note	String	Public	Note of student attendance

Method

Method	Return Type	Visibility	Description
AddStudentAttendance	Void	Public	Add new student attendance

4.2.15 AttendanceLog

Attribute

Attribute	Type	Visibility	Description
LogID	Int	Private	Unique id of each log
RolllCallID	Int	Public	ID of roll call of attendance log
LogDate	Date	Public	Date of log
TypeID	Type	Public	Type of log

Method

Method	Return Type	Visibility	Description
GetLogAtDate	Attendance log	Public	Get log in system by date

4.2.16 LogType

Attribute

Attribute	Type	Visibility	Description
TypeID	Int	Public	Unique id of each log type
TypeName	String	Public	Name of log type

Method

Method	Return Type	Visibility	Description
GetType	LogType	Public	Get type of log

4.2.17 LogImage

Attribute

Attribute	Type	Visibility	Description
ImageID	Int	Public	Unique id of each image
LogID	Int	Public	ID of the image's log
ImageLink	String	Public	Link of image

Method

Method	Return Type	Visibility	Description

GetLogImages	List<LogImage>	Public	Get all the images of a log

4.2.18 Request

Attribute

Attribute	Type	Visibility	Description
RequestID	Int	Public	Unique id of each log type
CreatedAdminID	Int	Public	ID of the admin who created the request
CheckedAdminID	Int	Public	ID of the admin who accept/deny the request
StudentID	Int	Public	ID of the student receive the request
Context	String	Public	Content of the request
IsResponse	Boolean	Public	The request is responese or not

Method

Method	Return Type	Visibility	Description
GetRequestByID	Request	Public	Get Request in system by ID
GetStudentRequest	List<Request>	Public	Get all requests of a student

4.2.19 RequestImage

Attribute

Attribute	Type	Visibility	Description
ImageID	Int	Public	Unique id of each image
RequestID	Int	Public	ID of the image's request
ImageLink	String	Public	Link of image

Method

Method	Return Type	Visibility	Description
GetRequestImages	List<LogImage>	Public	Get all the images of a request

4.3 Sequence Diagram

4.3.1 Add Image for Singe Student

Figure 54: Add Image for Singe Student Sequence Diagram

4.3.2 Add Image for Many Students

Figure 55: Add Image for Many Students Sequence Diagram

4.3.3 Delete Student Image

Figure 56: Delete Student Image Sequence Diagram

4.3.4 Create Account

Figure 57: Create Account Sequence Diagram

4.3.5 Configure System

Figure 58: Configure System Sequence Diagram

4.3.6 Face Detection

Figure 59: Face Detection Sequence Diagram

4.3.7 Face Recognition

Figure 60: Face Recognition Sequence Diagram

4.3.8 Auto Free Storage Space

Figure 61: Auto Free Storage Space Sequence Diagram

4.3.9 Auto Active Roll Call

Figure 62: Auto Active Roll Call Sequence Diagram

4.3.10 Add Roll Call

Figure 63: Add Roll Call Sequence Diagram

4.3.11 Edit Roll Call

Figure 64: Edit Roll Call Sequence Diagram

4.3.12 Import Student List

Figure 65: Import Student List Sequence Diagram

4.3.13 Export Report

Figure 66: Export Report Sequence Diagram

4.3.14 Add Student

Figure 67: Add Student Sequence Diagram

4.3.15 Add Class

Figure 68: Add Class Sequence Diagram

4.3.16 Edit Class

Figure 69:Edit Class Sequence Diagram

4.3.17 Add Subject

Figure 70: Add Subject Sequence Diagram

4.3.18 Edit Subject

Figure 71: Edit Subject Sequence Diagram

4.3.19 Take Attendance Auto

Figure 72: Take Attendance Auto Sequence Diagram

4.3.20 Take Attendance Manual

Figure 73: Take Attendance Manual Sequence Diagram

4.3.21 Take Attendance Offline & Sync Data

Figure 74: Take Attendance Offline & Sync Data Sequence Diagram

4.3.22 View Roll Call Detail

Figure 75: View Roll Call Detail Sequence Diagram

4.3.23 Report RollCall by Class

Figure 76: Report Attendance by Class Sequence Diagram

4.3.24 Check Present Rate

Figure 77: Check Present Rate Sequence Diagram

5. User Interface Design

5.1 Instructor Interface Design

5.1.1 Instructor Home Page

Figure 78: Instructor Home Page

Fields

No	Field Name	Description	Read only	Mand -atory	Control Type	Data Type	Length
6	Roll Call Info	Info of a roll call: Subject, class, time….	Y	Y	Label	N/A	N/A

Button/ Hyperlink

No	Function	Description	Validation	Outcome
1	Account Info	Username of user	No	Account Info
2	Home Page	Click to go to home page	No	Home Page
3	Instructor Home	Click to go to home page	No	Home Page
4	Teaching Calendar	Click to View Teaching Calendar	No	Teaching Calendar Page
5	Present Roll Call	Click to view the roll call at present	No	Present Roll Call Page

5.1.2 Instructor – Teaching Calendar

Figure 79: Instructor – Teaching Calendar

Fields

No	Field Name	Description	Read only	Mand -atory	Control Type	Data Type	Length
1	Traching Calendar	The teaching calendar of instructor	Y	Y	Calenda r	N/A	N/A
4	Past event	Event of the days before today	Y	N	Label	N/A	N/A
5	Future Event	Event of the days after today	Y	N	Label	N/A	N/A

Button/ Hyperlink

No	Function	Description	Validation	Outcome
2	Renew Token	Click to get a new token from Google Calendar	No	Get Token from Google
3	Request Calendar Sync now	Click to go to sync the calendar to Google Calendar	No	Calendar is synced

5.1.3 Instructor – Present Roll Call

Figure 80: Instructor – Present Roll Call

Fields

No	Field Name	Description	Read only	Mand -atory	Control Type	Data Type	Length
1	Current Roll Call	The info of current roll call	Y	Y	N/A	N/A	N/A
2	Student List Table	The list of student and their attendance of current roll call	N	Y	Table	N/A	N/A

Current Roll Call includes

Field Name	Description	Read only	Control Type	Data Type	Length
Roll Call	ID of the roll call	Y	Label	Interger	N/A
Subject	Subject of the roll call	Y	Label	String	N/A
Instructor	Instructor of the roll call	Y	Label	String	N/A
Class	Class of the roll call	Y	Label	String	N/A
Date	Begin date and end date of the roll call	Y	Label	String	N/A
Time	Begin time and end time and end time	Y	Label	String	N/A
Select Images	Upload image for check attendance	N	File Upload	N/A	N/A
Upload	Upload button	N	Button	N/A	N/A

Table Student List includes

Field Name	Description	Read only	Control Type	Data Type	Length
No	No	Y	Label	Interger	N/A
Student Name	Full Name of the student	Y	Label	String	N/A
Present	Check if the student is present	N	Radio Button	N/A	N/A
Absent	Check if the student is absent	N	Radio Button	N/A	N/A
Note	Note of instructor	N	Textbox	String	[0, 50]
Submit Attendance	Click to submit the re-check attendance	N	Button	N/A	N/A

5.1.4 Instructor – Roll Call Detail

Figure 81: Instructor – Roll Call Detail

Fields

No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
1	Current Roll Call	The info of the roll call	Y	Y	N/A	N/A	N/A
2	Student List Table	The list of student and their attendance of the roll call	Y	Y	Table	N/A	N/A
3	Attendance Log Table	Every log of the roll call	N	Y	Table	N/A	N/A

Table Current Roll Call includes

Field Name	Description	Read only	Control Type	Data Type	Length
Roll Call	ID of the roll call	Y	Label	Interger	N/A
Subject	Subject of the roll call	Y	Label	String	N/A
Instructor	Instructor of the roll call	Y	Label	String	N/A
Class	Class of the roll call	Y	Label	String	N/A
Date	Begin date and end date of the roll call	Y	Label	String	N/A
Time	Begin time and end time and end time	Y	Label	String	N/A

Table Student List includes

Field Name	Description	Read only	Control Type	Data Type	Length
No	No	Y	Label	Integer	N/A
Student Code	Code of the student	Y	Label	String	N/A
Student Name	Full Name of the student	Y	Label	String	N/A
Absent Rate	The absent rate of the student	Y	Label	Integer	N/A

Table Attendance Log includes

Field Name	Description	Read only	Control Type	Data Type	Length
No	No	Y	Label	Integer	N/A
Date	The date of attendance log	Y	Label	Datetime	N/A
Present Rate	The present rate of attendance lig	Y	Label	String	N/A

Log Detail	Click to view the detail, or check attendance manually	Y	Button	N/A	N/A

5.1.5 Instructor – Check Attendance Manual

Figure 82: Instructor – Check Attendance Manual

Fields

No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
1	Roll Call Info	The info of the roll call	Y	Y	N/A	N/A	N/A
2	Log Image	The image of the log	Y	Y	Image	N/A	N/A
3	Attendance	Attendance of student in the log	N	Y	Table	N/A	N/A

Button/ Hyperlink

No	Function	Description	Validation	Outcome
4	Submit Attendance	Click to submit the attendance	No	Attendance Submitted
5	Close	Close the log detail pop-up	No	Pop-up is closed

Roll Call Info includes

Field Name	Description	Read only	Control Type	Data Type	Length
Roll Call	ID of the roll call	Y	Label	Integer	N/A

Subject	Subject of the roll call	Y	Label	String	N/A
Class	Class of the roll call	Y	Label	String	N/A
Date	The date of the log	Y	Label	String	N/A

Table Student List includes

Field Name	Description	Read only	Control Type	Data Type	Length
No	No	Y	Label	Integer	N/A
Student Name	Full Name of the student	Y	Label	String	N/A
Present	Check if the student is present	N	Radio Button	N/A	N/A
Absent	Check if the student is absent	N	Radio Button	N/A	N/A
Note	Note of instructor	N	Textbox	String	[0,50]

5.2 Admin Interface Design

5.2.1 Admin Home Page

Figure 83: Admin Home Page

Button/ Hyperlink

No	Function	Description	Validation	Outcome
1	Add Staff	Click to add new staff	No	Add Staff
2	Admin Dashboard	Click to go to admin home page	No	Home Page
3	Staff Accounts	Click to go to staff accounts page	No	Staff Accounts Page
4	Instructor Accounts	Click to go to instructor accounts page	No	Instructor Accounts Page

5	Student Accounts	Click to go to student accounts page	No	Student Accounts Page
6	Student List	Click to go to student list page	No	Student List Page
7	Upload Images	Click to select images for students	No	Image Multi Upload Page
8	Configure System	Click to configure the system	No	Configure Page
9	Log Images	Click to show log images	No	Log Images
10	Accounts Info	Click to log out account	No	Login Page
11	Pending Request	Click to view student request	No	Image Request
12	Create Request	Click to create request student	No	Create Request

5.2.2 Admin – Student List

Figure 84: Admin – Student List

Fields

No	Field Name	Description	Read only	Mand -atory	Control Type	Data Type	Length
1	Class	Filter student by class	N	Y	Dropdo-wnlist	String	N/A
3	Number of record	Number of record per page	N	Y	Dropdo-wnlist	Integer	N/A
4	Search	Search textbox to find student	N	Y	Textbox	String	[0, 50]
5	Student List	Table showing	Y	Y	Table	N/A	N/A

		info of students					

Button/ Hyperlink

No	Function	Description	Validation	Outcome
2	Select	Select the class to filter	No	Table show students of class
6	Paging	Paging area of the student table	No	Home Page
7	Request Upload	Click create image request to student	No	Create Request

Table Student List includes

Field Name	Description	Read only	Control Type	Data Type	Length
No	No	Y	Label	Integer	N/A
Student Code	Code of the student	Y	Label	String	7
Student Name	Full Name of the student	Y	Label	String	N/A
Email	Email of the student	Y	Label	String	N/A
Class	Class of the sutdent	Y	Label	String	N/A
Detail	Click to view image data of a student	N	Button	N/A	N/A

5.2.3 Admin – Single Student Info

Figure 85: Admin – Single Student Info

Fields

No	Field Name	Description	Read only	Mand -atory	Control Type	Data Type	Length
1	Image Date	Image Data of a student. Click the X button to delete a image	N	N	N/A	N/A	N/A
2	Select Student Image	Select the image of the student	N	Y	File Upload	Image	N/A

Button/ Hyperlink

No	Function	Description	Validation	Outcome
3	Upload Button	Click to upload the images	No	Show Upload Result

5.2.4 Admin – Select Images for Students

Figure 86: Admin – Select Images for Students

Fields

No	Field Name	Description	Read only	Mand -atory	Control Type	Data Type	Length
1	Face Region	Face Region in a	Y	Y	N/A	N/A	N/A

		picture. When hover, a dropdownlist will be show					
2	Student List	A dropdownlist to select face's owner. Contain student image, code and full name	N	Y	Dropdo-wnlist	N/A	N/A

Button/ Hyperlink

No	Function	Description	Validation	Outcome
3	Submit Result	Click to submit and save the result	No	Back to Image Data Page

5.3 Staff Interface Design

5.3.1 Staff – Roll Call List

Figure 87: Staff - Roll Call List

Fields

No	Field Name	Description	Read only	Mand -atory	Control Type	Data Type	Length
2	Date	Filter Roll call by date.	N	Y	Dropdo-wnlist	String	N/A
6	Search	Search textbox to find roll calls	N	Y	Textbox	String	[0, 50]
7	Roll call List	Table showing info of roll calls	Y	Y	Table	Integer	N/A

Button/ Hyperlink

No	Function	Description	Validation	Outcome
1	Create new roll call	Create a new roll call	No	Create roll call page.
3	Select	Select the duration date to filter (or base on semester).	Yes	Table show students of class
4	Roll call report	Report attendance per day of roll call	No	Auto download file to user computer.
5	Final exam report.	Report summary of roll call.	No	Auto download file to user computer.
8	Change	Change schedule of roll call.	No	Change schedule page.
9	Edit	Edit the info of roll call.	No	Edit roll call page.
10	Delete	Delete the roll call	No	Message confirms.
11	Student list	Show and edit student list of roll call.	No	Student list page.
12	Check Attendance	Check attendance of roll call.	No	Check attendance page.
13	View attendance	View attendance of roll call.	No	View attendance page.
14	Paging	Paging area of the roll call table	No	Home Page

5.3.2 Staff – Create New Roll Call

Figure 88: Staff - Create Roll Call

Fields

No	Field Name	Description	Read only	Mand -atory	Control Type	Data Type	Length
1	Semester	The semester of roll call.	N	Y	Dropdo-wnlist	String	N/A
2	Begin Date	The begin date of roll call.	Y	Y	Dropdo-wnlist	String	N/A
3	Major	The major of roll call.	N	Y	Dropdo-wnlist	String	N/A
4	Class	The class of roll call.	N	Y	Dropdo-wnlist	String	N/A
5	Subject	The subject of roll call.	N	Y	Dropdo-wnlist	String	N/A
6	Start time	The start time of roll call.	N	Y	Dropdo-wnlist	String	N/A
7	Instructor	The instructor of roll call.	N	Y	Dropdo-wnlist	String	N/A

Button/ Hyperlink

No	Function	Description	Validation	Outcome
8	Create	Click to create a new roll call.	No	Roll call list page.
9	Back to list	Back to roll call list.	No	Roll call list page.

5.3.3 Staff – Check Attendance

Figure 89: Staff - Check Attendance

Fields

No	Field Name	Description	Read only	Mand -atory	Control Type	Data Type	Length
1	Roll call detail	The info detail of roll call.	Y	Y	Label	String	N/A
2	Attendance Log	The attendance log of roll call by date.	Y	Y	Label	String	N/A

Button/ Hyperlink

No	Function	Description	Validation	Outcome
3	Report	Download the attendance log in select date.	No	Excel file.
4	Log detail	Detail of attendance log in select date.	No	Log detail page.

5.4 Mobile Interface Design

5.4.1 Login Activity

Figure 90: Login Activity

Fields

No	Field Name	Description	Read only	Mand -atory	Control Type	Data Type	Length
1	Username	Fill user's username	N	N	Textbox	String	[5, 30]
2	Password	Fill user's password	N	N	Textbox	String	[5, 30]

Button/ Hyperlink

No	Function	Description	Validation	Outcome
3	Login	Click to login the system	No	Transfer to Course activity

5.4.2 Course Activity

Figure 91: Course Activity

Fields

No	Field Name	Description	Read only	Mand -atory	Control Type	Data Type	Length
1	Roll Call out of class time	Show information about Roll Call out of class time	Y	Y	N/A	N/A	N/A
2	Roll Call on class time	Show information about Roll Call on class time	Y	Y	N/A	N/A	N/A

Button/ Hyperlink

No	Function	Description	Validation	Outcome
3	Take Attendance	Click transfer to camera activity	No	Transfer to Camera activity

RollCall out of class time includes

Field	Description	Read	Control	Color	Data	Length

Name		only	Type		Type	
Subject	Subject of the roll call	Y	Textview	#ffffff	String	N/A
Class	Class of the roll call	Y	Textview	#7dccd3	String	N/A
Time	The time of the roll call	Y	Textview	#7dccd3	String	N/A
Date	The date of the roll call	Y	Textview	#7dccd3	String	N/A

RollCall on class time includes

Field Name	Description	Read only	Control Type	Color	Data Type	Length
Subject	Subject of the roll call	Y	Textview	#ff6f6f	String	N/A
Class	Class of the roll call	Y	Textview	#5ffdff	String	N/A
Time	The time of the roll call	Y	Textview	#5ffdff	String	N/A
Date	The date of the roll call	Y	Textview	#5ffdff	String	N/A

5.4.3 Camera Activity

Figure 92: Camera Activity

Fields

No	Field Name	Description	Read only	Mandatory	Control Type	Data Type	Length
1	Screen view	Show screen of camera on real time	Y	Y	Frame layout	N/A	N/A

Button/ Hyperlink

No	Function	Description	Validation	Outcome
2	Capture	Click to capture image	No	Image on real time send to web service and transfer to Course activity

5.4.4 Roll Call Activity

Figure 93: Roll Call Activity

Fields

No	Field Name	Description	Read only	Mand -atory	Control Type	Date Type	Length
1	Roll Call Information	Show information about roll call	Y	Y	N/A	N/A	N/A
2	Student List	Show list student of roll call	Y	Y	List View	N/A	N/A

Button/ Hyperlink

No	Function	Description	Validation	Outcome
3	Log List	Click one log to view log detail	No	Popup log detail

Student List includes

Field Name	Description	Read only	Control Type	Color	Data Type	Length
Student Code	Code of student	Y	Textview	#7dccd3	String	N/A
Student Name	Name of student	Y	Textview	#7dccd3	String	N/A

Log List includes

Field Name	Description	Read only	Control Type	Color	Data Type	Length
Date	Show log date	Y	Textview	#7dccd3	String	N/A
Present	Show log present	Y	Textview	#7dccd3	String	N/A

5.4.5 Check Attendance Activity

Figure 94: Check Attendance Activity

Fields

No	Field Name	Description	Read only	Mand atory	Control Type	Date Type	Length
1	Student List of Log	Show list student of log	Y	Y	List View	N/A	N/A

Button/ Hyperlink

No	Function	Description	Validation	Outcome
2	Cancel	Click to close log without saving result	No	Transfer to Log activity
3	Submit	Click to submit result to server	No	Transfer to Log activity

Student List of Log includes

Field Name	Description	Read only	Control Type	Data Type	Length
Date	Date of Log	Y	Textview	String	N/A
Student Name	Name of student	Y	Textview	String	N/A
Check box	Check to present or absent student	N	Checkbox	N/A	N/A

6. Database Design

6.1 Entity Relationship Diagram

Figure 95: Entity Relationship Diagram

6.2 Data Dictionary

Entity Data dictionary: describe content of all entities	
Entity Name	**Description**
Semester	Describe all semesters in school system. Each semester last 4-5 months.
Major	Describe all majors in school system. (Current major: Software Engineering, Financial Banking, Business Administrator …)
SubjectMajorMapping	Describe relationship between table subject and table major.
Subject	Describe all subjects in school system, A subject can be taught in many majors
SubjectType	Describe type of subject (IT, economic, language ...)
Class	Describe all class in school system. Each class belong to one major
Student	Describe all students in school system.
StudentImage	Describe all images of each student.
Instructor	Describe all instructors in school system. An instructor can teach many types of subject.
InstructorTeaching	Describe all the subject type the instructor can teach.
RollCall	Describe all roll calls in school system. A roll call contains info: Class, Subject, begin-end Time…..
StudySession	Describe all study session of roll call. A roll call has many study sessions.
StudentInRollCall	Describe student list of roll call.
StudentAttendance	Describe attendance of student in a log. The student can be present or absent.
AttendanceLog	Describe all all the attendance log of roll call. A log contains the present/absent students in a date of a roll call.
LogImage	Describe all images of attendance log.
LogType	Describe all type of attendance log (Auto, Manual).
Request	Describe all request send from admin to student.
RequestImage	Describe all images of request.
Staff	Describe all staffs in school system.
User	Describe account of user of system. User include: admin, staff, student …
Role	Describe role of user. One user has one role.

Table 48: Entity Data Dictionary

Entity name	Attributes	Description	Domain	Nulls
Semester	SemesterID{PK} SemesterName BeginDate EndDate	Uniquely identifies a semester, auto increment. Name of semester. The begin date of semester. The end date of semester.	Int 15 nvar Datetime Datetime	No No No No
Major	MajorID{PK} ShortName FullName	Uniquely identifies a major, auto increment. Short name of major. Full name of major.	Int 2 nvar 30 nvar	No Yes No
SubjectMajorMapping	MajorID {PK} SubjectID {PK}	Uniquely identifies of a major, copy from table Major Uniquely identifies of a subject, copy from table Subject	Int Int	No No
Subject	SubjectID{PK} ShortName FullName NumberOfSlot NumberOfSession IsSlotFixed Description IsActive	Uniquely identifies a subject, auto increment. Short name of subject. Full name of subject. Number of slot of subject (per day). Number of session of subject. This value shows that time of slot can be fixed or not. Description of subject This value shows that the subject is active or not.	Int 10 nvar 50 nvar Int Int Bit 10 char Bit	No Yes No No No No Yes No
SubjectType	TypeID {PK} TypeName	Uniquely identifies a type, auto increment. Name of type.	Int 50 nvar	No No
Class	ClassID {PK} ClassName IsActive	Uniquely identifies a class, auto increment. Name of class This value shows that the class is active or not.	Int 6 char Bit	No No No
Student	StudentID {PK} FullName Birthdate CitizenID Address Email IsActive	Uniquely identifies a student, auto increment. Full name of student. Birthdate of student. CitizenID of student Address of student Email of student. This value shows that student is active or not.	Int 50 nvar Datetime 8 char 100 nvar 50 nvar Bit	No No No No Yes Yes No

	StudentCode	Student code of student.	7 char	No
StudentImage	ImageID {PK} ImageLink	Uniquely indentifies of image, auto increment. The link of image.	Int 100 var	No No
Instructor	InstructorID {PK} FullName Email Phone ApiToken IsActive	Uniquely identifies of instructor, auto increment. Full name of instructor. Email of instructor. Phone of instructor. Token used for sync Google Calendar This value shows the instructor is active or not.	Int 50 nvar 50 nvar 12 nvar 100 nvar Bit	No No Yes Yes Yes No
InstructorTeaching	InstructorID {PK } SubjectTypeID{PK}	Uniquely identifies of an instructor, copy from table Instructor. Uniquely identifies of a subject type, copy from table SubjectType.	Int Int	No No
StudySession	SessionID {PK} SessionDate StartTime EndTime Note	Uniquely identifies of session, auto increment. The date of session. The time session start. The time session end. Note of session (Reason for change instructor, change time ….).	Int Date 7 time 7 time 50 nvar	No No No No Yes
RollCall	RollCallID {PK} StartTime EndTime BeginDate EndDate Status	Uniquely identifies of session, auto increment. The time roll call start. The time roll call end. The date roll call begins. The date roll call end. Status of roll call. (0: created. 1: active. 2: unactive)	Int 7 time 7 time Date Date Int	No No No No No No
StudentInRollCall	StudentID {PK} RollCallID {PK}	Uniquely identifies of a student, copy from table Student. Uniquely identifies of a roll call, copy from table Roll Call.	Int Int	No No
StudentAttendance	LogID {PK}	Uniquely identifies of an attendance log, copy from	Int	No

		table Attendance Log.		
	StudentID {PK}	Uniquely identifies of a student, copy from table Student.	Int	No
	IsPresent	This value shows that student is present or not.	Bit	No
	Note	Note of student attendance.	50 nvar	Yes
AttendanceLog	LogID {PK}	Uniquely identifies of attendance, auto increment.	Int	No
	LogDate	The date of log.	Date	No
LogImage	ImageID {PK}	Uniquely identifies of image, auto increment.	Int	No
	ImageLink	The link of image.	100 var	No
LogType	TypeID {PK}	Uniquely identifies of log type, auto increment.	Int	No
	TypeName	Name of type.	10 nvar	Yes
Request	RequestID {PK}	Uniquely identifies of request, auto increment.	Int	No
	Context	Content of the request.	100 nvar	Yes
	IsResponse	The request is responsed or not	Bit	No
RequestImage	ImageID {PK}	Uniquely identifies of image, auto increment.	Int	No
	ImageLink	The link of image.	100 var	No
Staff	StaffID {PK}	Uniquely identifies of staff, auto increment.	Int	No
	FullName	Full name of staff.	50 nvar	No
	Email	Email of staff.	50 nvar	Yes
	Phone	Phone of staff.	12 nvar	Yes
	IsActive	This value shows that staff is active or not.	Bit	No
User	UserID {PK}	Uniquely identifies of user, auto increment.	Int	No
	UserName	Username of user.	30 nvar	No
	Password	Password of user.	30 nvar	No
	IsActive	This value shows that user is active or not.	Bit	No
Role	RoleID {PK}	Uniquely identifies of role, auto increment.	Int	No
	RoleName	Name of role.	10 nvar	No

Table 49: Attribute Data Dictionary

7. Algorithms

7.1 Face Detection

7.1.1 Definition

Face detection is a computer technology that determines the locations and sizes of human faces in arbitrary (digital) images. It detects facial features and ignores anything else, such as buildings, trees and bodies.
References: http://en.wikipedia.org/wiki/Face_detection[1]

7.1.2 Different method for face detection

There are many ways to detect a face in a scene - easier and harder ones. Here is a list of the most common approaches in face detection:

- Finding faces in images with controlled background
- Finding faces by color
- Finding faces by motion
- Using a mixture of the above
- Finding faces in unconstrained scenes:
    - Neural Nets using statistical cluster information
    - Neural Net approach
    - Weak classifier cascades
    - Model-based Face Tracking

References:
http://www.facedetection.com/facedetection/techniques.htm[2]

7.1.3 The method we chose

We chose the Viola-Jones method. Reason:

- Simple, fast and good accuracy.
- Already implemented in EmguCV.

Method Introduction

This approach to detecting objects in images combines four key concepts:

- Simple rectangular features, called Haar features
- An Integral Image for rapid feature detection
- The AdaBoost machine-learning method
- A cascaded classifier to combine many features efficiently

Figure 96: Example of Haar Future

The presence of a Haar feature is determined by subtracting the average dark-region pixel value from the average light-region pixel value. If the difference is above a threshold (set during learning), that feature is said to be present.

Figure 97: Apply Haar future to sub-window

To select the specific Haar features to use, and to set threshold levels, Viola and Jones use a machine-learning method called AdaBoost. AdaBoost combines many "weak" classifiers to create one "strong" classifier.

Figure 98: Classifier Cascade

During use, if any one of these filters fails to pass an image region, that region is immediately classified as "Not Face." When a filter passes an image region, it goes to the next filter in the chain. Image regions that pass through all filters in the chain are classified as "Face." Viola and Jones dubbed this filtering chain a cascade.
Reference:
http://cognotics.com/opencv/servo_2007_series/part_2/sidebar.html[3]

The basic principle of the Viola-Jones face detection algorithm is to scan the detector many times through the same image – each time with a new size. Pseudo code for the algorithm:

1) Opens an image and transforms to greyscale if needed.
2) Runs a sub-window through the image.
2a) Rescales the content of the sub-window to 24*24pixels (if needed). Check the Haar feature in each window. Move the sub-window if Haar feauture missing.

2b) Enlarges the sub-window by a given factor a goes back to 2).
2c) Continues until the sub-window size is equal to the least dimension of the image.

Figure 99: Viola-Jones method flowchart

7.2 Face Recognition

7.2.1 Definition

- Face recognition is the task of identifying an already detected object as a KNOWN or UNKNOWN face, and in more advanced cases, telling EXACTLY WHO'S face it is!
- One of the ways to do this is by comparing selected facial features from the image and a facial database.

Reference: http://en.wikipedia.org/wiki/Facial_recognition_system[4]

7.2.2 Difference between Face Recognition and Face Detection

- Face Detection: identify an object as a "face" and locate it in the *input image*.
- Face Recognition: decide if this "face" is someone KNOWN, or UNKNOWN, basing on the database of faces it uses to validate this *input face.*
- Face detection's output(the face) is in fact recognition's input. And face recognition's output is the final decision: face known/ face unknown!

Figure 100: Face Recognition Process

7.2.3 Method for face recognition

Recognition algorithms can be divided into two main approaches:

- Geometric: which looks at distinguishing features.
- Photometric: which is a statistical approach that distill an image into values and comparing the values with templates to eliminate variances.

Popular recognition algorithms include:

1. Principal Component Analysis using Eigenfaces
2. Linear Discriminate Analysis
3. Elastic Bunch Graph Matching
4. TheHidden Markov model
5. The neuronal motivated dynamic link matching.

References: http://www.biometricscatalog.org/NSTCSubcommittee[5]

7.2.4 The Method we choose

We chose Fisherface method, reason:

- Already implemented in openCV.
- Has the best result, according to report. (http://asp.eurasipjournals.com/content/2012/1/92[6])
- Has the best result in our automation testing, with our current face database.

Method Introduction

Linear Discriminant Analysis (LDA), derived from an idea suggested by R.A. Fisher in 1936. When LDA is used to find the subspace representation of a set of face images, the resulting basis vectors defining that space are known as Fisherfaces.

Algorithm description:

- From the training image (Resolution NxN), transfrom the image to a vector with size N^2. .
- Calculate the average vector.
- Find the normalized vector, by subtracting the original vector by the average vector.
- Represent the original vector a linear combination of normalized vector.
- Find the weighted vector (the eigenface representation of ith face).
- The Training Phase is completed.
- From the input image, transform it to a vector with size N^2.
- Repeat above step, find the weighted vector.
- Find the distance between the input weighted vector and each weighted vector in the training set. Get the minimum distance.
- If the distance is higher than threehold, the image is unknow. If the distance is lower than threehold, find the nearest weighted vector, get the face owner.

Reference: http://www.scholarpedia.org/article/Fisherfaces[7], http://www.bytefish.de/blog/fisherfaces/[8]

Figure 101: Face Recognition Algorithm Flowchart

E. System Implementation & Test

1. Introduction

1.1 System Overview

This section provides in detail all necessary information about test plans, test cases, test result, test environments, pass/fail criteria and risks estimations as well as a checklist to cover all possible cases of RSM.

1.2 Test Approach

- Goal: To test the whole system based on the core workflow.
- Method:System Testing, Black-box Testing.

2. Database Relationship Diagram

Figure 102: Database Relationship Diagram

3. Performance measures

3.1 Face Detection Performance

- Average time to detect face in 100 images(Resolution: 1100 x 750px): 116.5s
- Best face detected result: 206/222 faces. Percent: 92.8%
- Worst face detected result: 173/222 faces. Percent 77.9%

3.2 Face recognition performance

- Maximum time to load the face recognizer: 12.56s
- Average time to load the face recognizer: 1.22s
- Recognition accuracy:

Figure 103: Face Recognizer Algorithm Accuracy

- The result in real world testing:

Figure 104: Number of images required

Figure 105: Full Recognize Percent

About testing progress and detail report, read **FaceRecognizerTestingReport.docx.**

4. Test Plan

The purpose of this section is to verify and ensure that RSM meets its design specification and other requirements from user. The following part will describe which features to be tested and which will not.

4.1 Features to be tested

We will test the entire system based on the following core workflow (Reference: **Figure 101: Core Workflow**).

Based on the workflow, all following features will be tested, which can include one or more functions. These features will be focused and tested thoroughly during the test phase.

- Manage Student Image: Add Images for Single Student/Many Students, Remove Student Images, Select Image From Log Images.
- Manage RollCall: Add/Edit Roll Call, Add/Remove Student in Roll Call.
- Face Detection & Recognition: Detect Face From Image, Recognize Face From Image (Read **FaceRecognizerTestingReport.docx**).
- Check Attendance: Show Current Roll Call, Check Attendance Auto (Mobile), Check Attendance Manual (Mobile, Web).

These remaining feature will be tested as individual module:

- Attendance Report: View Attendance Log, Get Excel Report
- ManageTeaching Calendar: View Teaching Calendar, Sync with Google Calendar
- Manage Study Session: Add RollCall Study Session, Change Session Date/Time/Instructor.
- Check own Attendance Rate

4.2 Features not to be tested

- Login, Logout
- Change Password
- Auto Free Storage Space
- Auto Google Calendar Sync
- Auto Active/Inactive Roll Call
- Configure System
- Admin - Manage Account
- Staff – Manage Semester, Manage Instructor, Manage Subject

5. System Testing Test Case

Figure 106: Core Workflow

ID	Test Case Description	Test Case Procedure	Excpected output	Result	Test Date	Note
ST01	Admin add images info for a student. (In actual use, admin must add images for all student in class)	1 – Login into the system using an admin account 2 – On the admin home page, click "Student Images", then click "Student List". 3 – Click a button in cell "Images" of a student. 4 – Click "Select student's images",	- After step 3, the thumbnails of selected image will be shown below the "Select student's images" button. - After step 4, display the uploaded images. The images contains white	Passed	30/10/2013	

		select student's image from computer. 5 – Press "Upload" button. 6 – Click the face region in images to select student face. 7 – Press "Submit Result" button.	regions on face detected. - After step 6, redirect back to Single Student Page.			
ST02	Staff create new Roll Call. (Class of this Roll Call contains the student in test case ST01).	1 – Login into the system using a staff account 2 – On the staff home page, click "Create Roll Call" link. 3 – Fill in Roll Call Info: 1. Semester 2. Begin Date 3. Major 4. Class 5. Subject 6. Begin Time 7. Instructor 4 – Press "Create" button.	- On the Create Roll Call Page, all the dropdownlist are disabled except semester. - When staff select a value from dropdowlist, the following dropdownlist will be enable, with filtered info. - After step 4, the new roll call is created.	Passed	30/10/2013	
ST03	Instructor login into Mobile App.	1 – Touch the app icon from mobile. 2 – On login screen, input username and password of the instructor selected in test case STO2. 3 – Touch "Login" button.	- A message is displayed: "Hello, [teacher name]." - All the instructor teaching roll call will be displayed with label contains teaching time. - The "Take Attendance" button is disabled.	Passed	30/10/2013	
ST04	Instructor check attendance by taking picture	1 – Set the server time to begin time of the roll call in ST02. 2 – Login again into the mobile app. 3- Touch "Take Attendance" button.	- After step 2, the "Take Attendance" button is enabled. - After step 6: A notification	Passed	30/10/2013	

		4 – Take a picture of class. 5 – Press Back button. 6 - Wait for the notification, slide from top to view. 7 – Touch the notification message.	with sound and vibration will appear. A picture is saved as log on server. - After step 7: Back to the home screen.			
ST05	Instructor check attendance by taking picture again	After step 7 in test case ST04 1- Touch "Take Attendance" button. 2 – Take a picture of class. 3 – Press Back button.	- A notification with sound and vibration will appear. - - One more picture is saved as log on server. - The attendance is update with more student.	Passed	30/10/2013	
ST06	Instructor view Roll Call Log	1 – On the home screen after login, touch an roll call item. 2- Press "Attendance Log" button. 3 – Touch a log date.	- After Step 2, attendance log of today and yesterday (If exist) will be shown. - After Step 3, a select list of present/absent will be shown	Passed	30/10/2013	
ST07	Instructor check attendance manually	1 – On the home screen after login, touch an roll call item. 2- Press "Attendance Log" button. 3 – Touch a log date. 4 – Recheck the attendance. 5 – Presss "OK" button	- The checked attendance will be submit, the screen is refresh.	Passed	30/10/2013	
ST08	Instructor view Roll Call Log using Web. (In case of mobile problem)	1 – Login into the system using the instructor account in test case ST03. 2 – On the instructor home page, click on "Detail" link of a Roll Call 3 – On the "Log Attendance" table, click "Log Detail" button of a log (Log date must be before yesterday)	- After step 2, the Roll Call Info page will be shown, contains roll call info and student list and a list of attendance log. - After step 3, a pop-up will be show, contains log images, the present/absent	Passed	30/10/2013	

			students.			
ST09	Instructor check attendance manually using Web. (In case of mobile problem)	1 - Repeat step 1-3 of test case ST08. 2 – Recheck the attendance. 3 – Click "Submit Attendance" button.	- The attendance is submitted.	Passed	30/10/2013	
ST10	Instructor get attendance report	1 – Login into the system using the instructor account in test case ST03. 2 – On the instructor home page, click on "Detail" link of a Roll Call. 3 – Click on the "Report" link in Log Attendance table.	- An excel file is downloaded. The file contains two sheets: Roll Call Book, Final Examl Student List.	Passed	30/10/2013	

Table 50: Core Workflow Test Case

6. Module Test Case

6.1 Manage RollCall

ID	Test Case Description	Test Case Procedure	Excpected output	Inter-test Case Dependence	Result	Test Date	Note
MR01	Staff view roll call list.	1 – Login into the system using a staff account. 2 – Click the Manage Roll Call tab in menu sidebar.	- The Roll call list is showed base on current semester (from begins date to end date).		Passed	15/10/2013	
MR02	Create new roll call.	1 – On the Manage Roll Call Page, click on "Create new roll call" link. 2 – Select information of roll call. Click on "Create" button.	- Dropdown list will show option is filter. -Transfer to Manage Roll Call page after step 2.	MR01	Passed	15/10/2013	
MR03	Staff filters the roll call by	1 – On Manage Roll Call page, staff select the semester in	_All roll call in semester selected or from begin	MR01	Passed	15/10/2013	

	semester or date.	dropdown list or select a new begin date – end date. 2 – Click on “Select” button.	date to end date will be show in page.				
MR04	View attendance	1 – On Manage Roll Call page, select an old semester. Click on “Select” button 2 – Click on “View attendance” link.	_The “View attendance” page will be show include: student list, attendance, absent...	MR01	Passed	16/10/2013	
MR05	Edit roll call	1 – On Manage Roll Call page. Click on “Edit” link. 2 – Click on “Save” button.	_The Edit page will be showed. _After step 2, transfer to Manage roll call with new roll call, which have been update.	MR01	Passed	17/10/2013	
MR06	Delete roll call	1 – On Manage Roll Call page, select a new semester. Click on “Select” button. 2 – Click on “Delete” link. Select “Yes”.	_The confirm message will be showed. _After step 2, transfer to Manage Roll Call page.	MR01	Passed	17/10/2013	
MR07	Change schedule roll call.	1– On Manage Roll Call page, click on “Change” link. 2 – Click on one date. 3 – Fill the new value and click on “Change” button.	_The calendar of roll call will be showed. _After step 2, a pop-up will be showed. _After step 3, the pop-up hide, and show calendar with new value.	MR01	Passed	18/10/2013	
MR08	View student list.	1 – On Manage Roll Call page, select a new semester. Click on “Select” button. 2 – Click on “Student list” link.	_The student list of class of roll call will be showed.	MR01	Passed	18/10/2013	
MR09	Add new students	1 – On Student list page,	_The student list will be	MR01,MR08	Passed	18/10/2013	

	to roll call	Type or select student on select box. 2 – Click on “Add” button.	showed with new students.				
MR10	Remove a student from roll call	1 – On Student list page, click on “Remove” link. 2 – Select “Yes” button.	_After step 1, a confirm message will be show. _After step 2, the student list will be show with no student has been remove.	MR01, MR08	Passed	19/10/2013	

Table 51: Manage Roll Call Test Case

6.2 Manage Student Image

ID	Test Case Description	Test Case Procedure	Excpected output	Inter-test Case Dependence	Result	Test Date	Note
MS01	Admin view the list of all students	1 – On the admin home page, click “Student Images”, then click “Student List”.	- All active students will be displayed, order descending by StudentID		Passed	22/09/2013	
MS02	Admin filter the student by class	1 – On the admin home page, click “Student Images”, then click “Student List”. 2 – Select a class from the dropdownlist. 3 – Press “Select” button.	- All students of the selected class will be filtered and displayed	MS01	Passed	22/09/2013	
MS03	Admin view student images data	1 – On the admin home page, click “Student Images”, then click “Student List”. 2 – Click a button in cell “Images” of a student.	- Display all images of the selected student	MS01	Passed	24/09/2013	
MS04	Admin add images for a	1 – On the admin home page, click “Student Images”, then	- After step 3, the thumbnails of selected	MS03	Passed	24/09/2013	

	student	click “Student List”. 2 – Click a button in cell “Images” of a student. 3 – Click “Select student's images”, select student’s image from computer. 4 – Press “Upload” button. 5 – Click the face region in images to select student face. 6 – Press “Submit Result” button.	image will be shown below the “Select student’s images” button. - After step 4, display the uploaded images. The images contains white regions on face detected. - After step 6, redirect back to Single Student Page.				
MS05	Admin add images for a student (Image already in database)	1 - Repeat Step 1-6 of case MS04. 2 – Click “Select student's images”, select the old image again. 4 – Press “Upload” button. 5 – Click the same region as first time. 6 – Press “Submit Result” button.	- Redirect back to Single Student Page. A warning show “Image already in database”.	MS03, MS04	Passed	24/09/2013	
MS06	Admin Delete Images of a student	1 – On the admin home page, click “Student Images”, then click “Student List. 2 – Click a button in cell “Images” of a student. 3 – Click the “X” mark on the top-right corner of an image. 4 – A confirm box is shown, click “Yes”.	- The image is deleted from database. A message is shown then hide.	MS03	Passed	25/09/2013	
MS07	Admin add	1 – On the admin home page,	- After step 2, the		Passed	27/09/2013	

	images for many students	click “Student Images”, then click “Upload Images”. 2 – Click “Select student's images”, select students’s image from computer. 3 – Press “Upload” button. 4 – Hover the mouse over the face region, click the dropdownlist. 5 – Select the face’s owner from dropdownlist. 6 – Click “Submit Result” button	thumbnails of selected image will be shown below the “Select student’s images” button. - In step 3, the dropdownlist contains: image, student code, student name of all active students. - After step 6, redirect back to Add Images Page, show the submitted result.				
MS08	Admin view the attendance log images	1 – On the admin home page, click “Student Images”, then click “Log Images”. 2 – Select the begin date to find. 3 – Select the end date to find. 4 – Click the “Find” button.	- On Log Images page, the default begin date is today, the default end date is set to tomorrow. - After step 4, all log images between begin date and end date will be shown as thumbnail.		Passed	05/10/2013	
MS09	Admin view the attendance log images filtered by class	1 – On the admin home page, click “Student Images”, then click “Log Images”. 2 – Select the begin date to find. 3 – Select the end date to find. 4 – Select a class from dropdownlist. 5 – Click the “Find” button.	- On Log Images page, the default begin date is today, the default end date is set to tomorrow. - After step 5, all log images between begin date and end date of the selected class will be shown as thumbnail.	MS08	Passed	05/10/2013	

MS10	Admin select learning data from log images	1 – Repeat step 1-4 of test case MS08. 2 – Select the log images by clicking them. 3 – Click "Use selected images" button. 4 – Hover the mouse on face region to see student name. 5 – If the name is wrong, click the face region. 6 – Select the face's owner using the dropdownlist. 7 – Press "Save Result" button.	- After step 5, a dropdownlist contains: image, student code, student name of all active students will be shown. - After step 7, redirect to Log Images page, show the submited result	MS08	Passed	05/10/2013	

Table 52: Manage Student Image Test Case

6.3 Check Attendance

ID	Test Case Description	Test Case Procedure	Excpected output	Inter-test Case Dependence	Result	Test Date	Note
CA01	Instructor view teaching session (Web)	1 – Create some roll calls for an instructor. 2 – Login into the system with that instructor account	- All the instructor teaching roll call will be displayed with label contains teaching time.	MR01	Passed	01/10/2013	
CA02	Instructor view teaching session on Sunday (Web)	1 – Create some roll calls for an instructor. 2 – Change server time to Sunday. 3 – Login into the system with that instructor account	- All the instructor teaching roll call will be displayed with red label contains "No Teaching Slot".	MR01, CA01	Passed	01/10/2013	
CA03	Instructor view	1 – On the instructor home	- The Roll Call Info page		Passed	01/10/2013	

	Present Roll Call (Web)	page, click on “Present Roll Call” link	will be shown, contains roll call info and student list				
CA04	Instructor view Roll Call Log (Web)	1 – On the instructor home page, click on “Detail” link of a roll call 2 – On the “Log Attendance” table, click “Log Detail” button of a log (Log date must be before yesterday)	- After step 1, the Roll Call Info page will be shown, contains roll call info and student list and a list of attendance log. - After step 2, a pop-up will be show, contains log images, the present/absent students.	CA01	Passed	04/10/2013	
CA05	Instructor check Attendance Manually (Web)	1 – On the instructor home page, click on “Detail” link of a roll call 2 – On the “Log Attendance” table, click “Log Detail” button of a log (Log date must be yesterday or today) 3 – Recheck the attendance. 4 – Click “Submit Attendance” button.	The attendance is submitted.	CA01, CA04	Passed	06/10/2013	
CA06	Instructor view teaching session (Mobile)	1 – Create some roll calls for an instructor. 2 – Login into the system with that instructor account using mobile	- All the instructor teaching roll call will be displayed. - The present roll call will be bigger, with red and orange text.	MR01	Passed	10/10/2013	
CA07	Instructor view teaching session	1 – Create some roll calls for an instructor.	- All the instructor teaching roll call will be	MR01, CA06	Passed	10/10/2013	

	without present Roll Call (Mobile)	2 – Change server time to 11:00 pm 2 – Login into the system with that instructor account using mobile	displayed. - The "Take Attendance" button will be disable				
CA08	Instructor check attendance by taking picture	1 – On the home screen after login, touch "Take Attendance" button. 2 – Take a picture of class. 3 - Wait for the notification, slide from top to view. 4 – Touch the notification message. 5 – Continue to take a picture. 6 – Press Back button.	- After step 2: A notification with sound and vibration will appear. A picture is saved as log on server. - After step 4: Show the home screen. - After step 5: The notification appears agagin, 1 more picture is stored as log.		Passed	13/10/2013	
CA08	Instructor view Roll Call Detail (Mobile)	1 – On the home screen after login, touch an roll call.	- All roll call info, include the student list will be shown at a new screen	CA06	Passed	13/10/2013	
CA09	Instructor view Roll Call Log (Mobile)	1 – On the home screen after login, touch an roll call item. 2- Press "Attendance Log" button. 3 – Touch a log date.	- After Step 2, attendance log of today and yesterday will be shown. - After Step 3, a select list of present/absent will be shown	CA06, CA08	Passed	13/10/2013	
CA10	Instructor check attendance manually	1 – On the home screen after login, touch an roll call item. 2- Press "Attendance Log" button. 3 – Touch a log date.	- The checked attendance will be submit, the screen is refresh.	CA06, CA09	Passed	15/10/2013	

		4 – Recheck the attendance. 5 – Presss “OK” button					

Table 53: Check Attendance Test Case

6.4 Attendance Report

ID	Test Case Description	Test Case Procedure	Excpected output	Inter-test Case Dependence	Result	Test Date	Note
AR01	Instructor get Roll Call attendance report	1 – Login into the system using an instructor account. 2 – On the instructor home page, click on “Detail” link of a Roll Call. 3 – Click on the “Report” link in Log Attendance table.	- An excel file is downloaded. The file contains two sheets: Roll Call Book, Final Exam Student List.		Passed	25/10/2013	
AR02	Staff get attendance report of Roll Calls	1 – Login into the system using an instructor account. 2 – On the Roll Call Page, filter the Roll Call by select start date and end date 3 – Click “Get Attendance Rate” button.	- An excel file is downloaded. Each sheet contains the Roll Call Book with attendance rate of each Roll Call.		Passed	25/10/2013	
AR03	Staff get final exam’s student list of Roll Calls	1 – Login into the system using an instructor account. 2 – On the Roll Call Page, filter the Roll Call by select start date and end date 3 – Click “Get Final Exam Student List” button.	- An excel file is downloaded. Each sheet contains the Final Exam Student List of each Roll Call.		Passed	25/10/2013	

Table 54: Attendance Report Test Case

6.5 Manage Teaching Calendar

ID	Test Case Description	Test Case Procedure	Excpected output	Inter-test Case Dependence	Result	Test Date	Note
MT01	Instructor view teaching calendar	1 – Create some roll calls for an instructor. 2 – Login into the system with that instructor account 3 - On the instructor home page, click on "Teaching Calendar" link	- A calendar will be shown, with the events are instructor teaching session.	MR01	Passed	04/11/2013	
MT02	Instructor request Google API token	1 – Repeat step 1-3 of case MT01 2- Click button "Sync with Google Calendar". 3 – On the new page, click "Accept" button	- Redirect back to Calendar Page, with a message "Token received. Your calendar will be sync to Google Calendar later." - On the Calendar Page, there're two button "Renew Token" and "Request Calendar Sync Now"	MT01	Passed	04/11/2013	
MT03	Instructor request Google Calendar sync	1 – Login with the instructor account used in MT02. 2 - On the instructor home page, click on "Teaching Calendar" link. 3 – Click "Request Calendar Sync Now" button.	- Redirect back to Calendar Page, with a message "Your calendar was synced to Google Calendar at …". - The events are synced to Google Calendar	MT02	Passed	04/11/2013	

		4 – Login into Google Calendar account to check.					
MT04	Instructor request new token	1 – Login with a new Google Calendar Account. 2 – Login with the instructor account used in MT02. 3 – Click "Renew Token" button. 4 – Click "Request Calendar Sync Now" button. 5 – Login into Google Calendar account to check.	- After step 3, redirect back to Calendar Page, with a message "Token received. Your calendar will be sync to Google Calendar later." - The events are synced to new Google Calendar account.	MT03	Passed	04/11/2013	
MT05	Auto sync Google Calendar	1 – At 02:00AM of everyday	- All the events begin from today will be synced for instructors who received Google API Token.	MT02	Passed	04/11/2013	

Table 55: Manage Teaching Calendar Test Case

6.6 Manage Study Session

ID	Test Case Description	Test Case Procedure	Excpected output	Inter-test Case Dependence	Result	Test Date	Note
MS01	Staff add more study session for roll call	1 – Login into the system using a staff account. 2 – Click on "Roll Call" link. 3 – In the "Schedule" column, click the "Change" link. 4 – Click "Add new Session" button. 5 – Fill in information:	- After step 3, a calendar is shown, contains all study session of the roll call. - After step 4, a pop-up is shown. - After step 6, the created session is added		Passed	25/10/2013	

		1. Date 2. Time 3. Instructor 4. Note 6 – Press “Add” button.	to the calendar.				
MS02	Staff change study session time (In past)	1 – Login into the system using a staff account. 2 – Click on “Roll Call” link. 3 – In the “Schedule” column, click the “Change” link. 4 – Click on an gray event on calendar (Event in the past).	- The pop-up is not shown		Passed	25/10/2013	
MS03	Staff change study session time (In future)	1 – Login into the system using a staff account. 2 – Click on “Roll Call” link. 3 – In the “Schedule” column, click the “Change” link. 4 – Click on an green event on calendar (Event in future). 5 – Select the date to change, 6 – Select time to changes. 7 – Enter reason into “Note”. 8 – Press “Save” button.	- After step 4, a pop-up is shown. - After step 5, the time dropdownlist will be filled with avaible time. - After step 8, the time of selected event is changed.		Passed	25/10/2013	
MS04	Staff change study session time without enter Note	1 – Login into the system using a staff account. 2 – Click on “Roll Call” link. 3 – In the “Schedule” column, click the “Change” link. 4 – Click on an green event on calendar (Event in future). 5 – Press “Save” button.	- Alert “Enter reason for changing date”.		Passed	26/10/2013	

MS05	Staff change study session time with Note longer than 50 characters	1 – Login into the system using a staff account. 2 – Click on "Roll Call" link. 3 – In the "Schedule" column, click the "Change" link. 4 – Click on an green event on calendar (Event in future). 5 – Enter note, longer than 50 characters 5 – Press "Save" button.	- Alert "Reason too long. Must be shorter than 50 characters".		Passed	26/10/2013	
MS06	Staff change instructor of study session	1 – Login into the system using a staff account. 2 – Click on "Roll Call" link. 3 – Click the "Change" link. 4 – Click "Change Instructor". 5 – Choose the start date. 6 – Choose the end date. 7 – Choose instructor. 8 – Enter reason into "Note". 9 – Press "Save" button.	- After step 4, a pop-up is shown. - After step 6, the instructor dropdownlist will be filled with avaiable instructor. - The instructor of selected study sessions will be changed.		Passed	27/10/2013	

Table 56: Manage Study Session Test Case

6.7 Check own Attendance Rate

ID	Test Case Description	Test Case Procedure	Excpected output	Inter-test Case Dependence	Result	Test Date	Note
CA01	New student check own attendance	1 – Using a staff account, create a new student 2 – Using an admin account, create the student account. 3 – Login into system, using	- The Student Home Page is shown, with "No Learning Course" text.		Passed	5/10/2013	

		the newly created student account.					
CA02	Student check own attendance Rate	1 – Login into the system using a student account. 2 – On the Roll Call Page, all the learning and learned course are shown. 3 – Click on the "Attendance" button of a course.	- A pop-up is shown, contains the absent rate, the log of present/absent of the student.		Passed	6/10/2013	

Table 57: Check own Attendance Rate Test Case

F. Software User's Manual

1. Installation Guide

1.1 Setting up environment at server site

The following software must be installed into the server machine:

1.1.1 Hardware requirements

Personal computers for developing with the minimum configuration:

- CPU Core 2 Duo 2.0GHz.
- 4GB of RAM.
- 300GB of hard disk.
- Wifi Internet (> 2Mbps).

1.1.2 Software requirements

- Web Server: Internet Information System – IIS 7.5
- Microsoft Windows 7: operating system and platform for development
- SQL Server 2008 R2: used to create and manage the database for web application
- Other library:
  - EmguCV x86 2.4.2 (http://sourceforge.net/projects/emgucv/files/emgucv/2.4.2/libemgucv-windows-x86-gpu-2.4.2.1777.exe/download)
  - Microsoft Visual C++ 2010 SP1 Redistributable Package (x86) (http://www.microsoft.com/en-us/download/details.aspx?id=8328)

1.1.3 Mobile Requirement

- Android 4.0 or above
- Camera 3.0 Megapixed or above

1.2 Deployment at server side

1.2.1 Prepare deployment package

- Extract the deployment package to a folder on the server.
- For example: D:\Capstone Project

Figure 107: Prepare deployment package

1.2.2 Deploy database

- Open Microsoft SQL Server Management Studio

Figure 108: Deploy database – 1

- Open the script.sql file

Figure 109: Deploy database - 2

- Execute the script to finish the process

Figure 110: Deploy database - 3

1.2.3 Configure Server before deploy

- Open the Internet Information System Manager
- Change the .NET Framework version to 4.0

Figure 111: Configure Server before deploy – 1

- Click the Application Pool

Figure 112: Configure Server before deploy – 2

- Right click the ASP.NET 4.0 Pool, select "Advance Settings".
- Set "Enable 32-bit Applications" to True.

Figure 113: Configure Server before deploy - 3

1.2.4 Deploy web application on server

- Change the connection string in Web.config
- Replace the "integrated security" with your own username and password

Figure 114: Deploy web application on server - 1

Figure 115: Deploy web application on server – 2

- Right click as Sites. Choose “Add Web Site”.

Figure 116: Deploy web application on server – 3

- Enter site name. Choose Application Pool as ASP.NET 4.0
- Select the path as the folder in step 1. Enter port other than 80.

Figure 117: Deploy web application on server – 4

- Press ok to finish the process

Figure 118: Deploy web application on server – 5

1.3 Setting up the environment at client side

1.3.1 Setting up for computer

- The client devices need to have one of the following browsers to access the website:
  - Google Chrome.
  - Firefox.

1.3.2 Setting up for Android devices

- Change setting to install unknown source apk.
- Connect to the same network as the server

Figure 119: Change Settings

- Use browser and goto address 192.168.218.1:2987, download the apk file.

Figure 120: Download app

Click Install to install the app.

Figure 121: Install App

2. User Guide

The user guide is based on the normal workflow of the system

2.1 Login

Figure 122: Login Page

Step	Description
1	Fill Username and Password (Sample admin account: Username: admin1. Password: 123456)
2	Click button Login

Table 58: Login Step

2.2 Admin – Add Image for student

2.2.1 View Student Image

Figure 123: Student List Page

Step	Description
1	Click Tab Student Images
2	Click link Student List
3	Click button "…. Images" to view, add more image for students
4	Click button "Request Upload" to send image upload request to students

Table 59: View Student Image Step

2.2.2 Upload Image

Figure 124: Upload Image Page

Step	Description
1	Click to delete an image
2	Click button “Select student’s images” to select image for upload
3	Display thumbnail of selected image
4	Click button “Upload” to upload the selected images

Table 60: Upload Image Step

2.2.3 Select Student Image

Figure 125: Select Student Image Page

Step	Description
1	Click to select the face of students
2	Click button “Submit result” to save the selected face

Table 61: Select Student Image Step

2.3 Staff – Create Roll Call

2.3.1 View Roll Call List

Figure 126: Roll Call List Page

Step	Description
1	Click on link "Create new roll call" to create a roll call.
2	Select the semester, duration date includes roll calls you want to show in list.
3	Click on "Select" button to show result of step 2.
4	Click to report attendance of roll call.
5	Click to report attendance summary of roll call.
6	Click to search information you find.
7	Clink on link "Check Attendance" to check attendance for roll call.
8	Click on link "Change" to change schedule of roll call. The old roll call will not have this link.
11	Click on link "Student list" to view, remove or add student for roll call. The old roll call or roll call start over 2 day will not have this link.

Table 62: View Roll Call List Step

2.3.2 Create Roll Call

Figure 127: Create Roll Call Page

Step	Description
1	Select the semester of roll call.
2	Select the start date of roll call.
3	Select the major of roll call.
4	Select the class of roll call.
5	Select the subject of roll call
6	Select the start time of roll call.
7	Select the instructor of roll call.
8	Click on button "Create" to create a roll call.

Table 63: Create Roll Call Step

2.3.3 Change Roll Call Student List

Figure 128: Roll Call Student List Page

Step	Description
1	Click to remove student from roll call.
2	Type or select students to add to roll call.

Table 64: Change Student List Step

2.3.4 Change Roll Call Schedule

Figure 129: Change Schedule Page

Step	Description
1	Click to add a session or change instructor of roll call.
2	Click to change session of roll call.

Table 65: Change Schedule Step

2.4 Instructor – Check Attendance (Mobile)

2.4.1 Login

Figure 130: Login Screen

Step	Description
1	Fill Username and Password (Sample instructor account: Username: phuonglhk. Password: 123456)
2	Click button Login

Table 66: Login Step

2.4.2 View Roll Call List

Figure 131: Roll Call List Screen

Step	Description
1	View Roll Call list and make sure about the roll call that you want taking attendance
2	Click button Take Attendance

Table 67: View Roll Call List Step

2.4.3 Take Attendance Auto

Figure 132: Camera Screen

Step	Description
1	View camera screen and choose the best position and angle
2	Click button to capture image

Table 68: Check Attendance Step

2.4.4 View Roll Call Detail

Figure 133: Roll Call Detail Screen

Step	Description
1	View Roll Call info (class, subject, date, time, student list and log list)
2	Click one log you want to taking attendance manual

Table 69: View Roll Call Detail Step

2.4.5 Take Attendance Manual

Figure 134: Take Attendance Screen

Step	Description
1	View student list in roll call and check or uncheck any student.
2	Click button OK to taking attendance manual

Table 70: Take Attendance Manual Step

2.5 Instructor – Check Report

2.5.1 View Teaching Roll Call

Figure 135: Instructor Home Page

Step	Description
1	Click link "Attendance" to view and check attendance of teaching class
2	Click link "Detail" to view attendance detail of the roll call

Table 71: View Teaching Roll Call Step

2.5.2 View Roll Call Report

Figure 136: Roll Call Detail Page

Step	Description
1	Click link "Report" to download the attendance report
2	Click link "Log Detail" to view the log detail of selected date

Table 72: View Roll Call Report Step

2.5.3 Report Sample

Roll Call Book

Semester: Fall 2013
Class: SE0999
Subject: HCI
Session: 15
Time: 08:45 - 10:15
Date: 07-10-2013 to 27-10-2013
Instructor: Lâm Hữu Khánh Phương

X = Present

No.	Student Code	Student Name	Week 1					Week 2					Week 3					Percent
1	SE60740	Pham Huy Hoàng	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	100.00%
2	SE60741	Trịnh Hoàng Minh	X		X	X	X	X	X	X	X	X	X	X	X		X	86.67%
3	SE60742	Tran Quang	X	X	X	X	X	X	X	X	X	X				X	X	80.00%
4	SE60730	Trần Quyền	X	X	X	X	X	X	X	X	X	X			X	X	X	86.67%
5	60749	Lê Anh Bảo	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	100.00%
6	SE94059	Kiều Trọng Khánh	X	X	X	X	X	X	X	X	X	X	X		X	X	X	93.33%
7	SE50964	Nguyễn Quang Huy	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	100.00%
8	48721	Nguyễn Thanh Bình	X	X	X	X			X	X	X	X	X	X	X	X	X	86.67%
9	26989	Hà Minh Thanh	X		X				X	X	X	X		X	X		X	60.00%
10	04943	Phan Ngọc Trầm	X	X	X					X	X		X	X		X	X	60.00%
11																		
12																		
13																		
14																		

SE0999_HCI | SE0999_HCI - Final Exam

Figure 137: Report - Sheet 1

Final Exam Student List

Semester: Fall 2013
Class: SE0999
Subject: HCI

No.	Student Code	Student Name	Birthdate	Class	Note
1	SE60740	Pham Huy Hoàng	03/04/1992	SE0670	
2	SE60741	Trịnh Hoàng Minh	12/03/1992	SE0670	
3	SE60742	Tran Quang	01/01/1992	SE0670	
4	SE60730	Trần Quyền	01/02/1992	SE0670	
5	60749	Lê Anh Bảo	12/02/1921	SE0670	
6	SE94059	Kiều Trọng Khánh	11/11/1970	SE0999	
7	SE50964	Nguyễn Quang Huy	05/05/1991	SE0999	
8	48721	Nguyễn Thanh Bình	02/04/1992	SE0999	
~~9~~	~~26989~~	~~Hà Minh Thanh~~	~~01/01/1990~~	~~SE0999~~	Absent Rate: 40%
~~10~~	~~04943~~	~~Phan Ngọc Trầm~~	~~01/04/1991~~	~~SE0692~~	Absent Rate: 40%

Figure 138: Report - Sheet 2

3. User Notice

3.1 Image Data Notice

- Each students must has at least 8 images as training images, for acceptable accuracy.
- The image should be clear without too much noise.
- Two eyes should be as the center.
- The face must look front, or tilt ~5-10 degree.
- Gray image is also okay

Example of good image data:

Sample data of a student (From 8-20 images), should contains:

- 70-80% of frontal face, with different brightness, facial expression....
- 20% of side face, with different brightness.

Example of a good sample data set:

Avoid using images like these:

3.2 Position & Angle Notice

The below position and angle are recommend for best result.

For small class room

For bigger classroom

- According to test result, with class of 15-20 students, the number of required image is 4-6 images.
- With bigger class, the number of required image is 6-10 images.

3.3 Other Notice

- The classroom should has good brightness.
- The student should look to front, look at the camera.

Example:

In some situation, the student on the front cover the face of back students, this lead to lower recognition rate.

Solution for better recognition rate:

* Take more images in different angles.
* Tell front student to lower their head.
* Use the “learning” modules.

G. Appendix

1. Introduce to Aglie Software Development: http://www.indicthreads.com/1439/quick-introduction-to-agile-software-development/
2. C# Coding Convention: http://msdn.microsoft.com/en-us/library/vstudio/ff926074.aspx
3. Multi-tier Architecture: http://en.wikipedia.org/wiki/Multitier_architecture
4. Face Detection Wikipedia: http://en.wikipedia.org/wiki/Face_detection
5. Face Detection Technicques: http://www.facedetection.com/facedetection/techniques.htm
6. Face Detection Viola-Johnes method: http://cognotics.com/opencv/servo_2007_series/part_2/sidebar.html
7. Face Recognition System Wiki: http://en.wikipedia.org/wiki/Facial_recognition_system
8. Face Recognition Technique: http://www.biometricscatalog.org/NSTCSubcommittee
9. Test Report: Eigenfaces, Fisherfaces error rates: http://asp.eurasipjournals.com/content/2012/1/92
10. Fisherfaces algorithm: http://www.scholarpedia.org/article/Fisherfaces
11. Fisherfaces algorithm: http://www.bytefish.de/blog/fisherfaces/