

**COMPUTER SCIENCE AND ENGINEERING**

***PROJECT TITLE***

A project/thesis submitted in partial fulfillment of

the requirements for the Degree of

Bachelor of Computer Science and Engineering

from

VNUK Institute for Research & Executive Education

by

***Full name***

***Student ID***

***Full name***

***Student ID***

**SUPERVISOR**

***Name 1, Name 2***

© Copyright 2019 by

[Student Name]

ALL RIGHTS RESERVED

DECLARATION

I/We, [Student Name (s)]\*, declare that this thesis/project titled, Thesis/Project Title and the work presented in it are my own. I confirm that:

* This work was done wholly or mainly while in candidature for a [MSc/ BSc]\* degree at VNUK Institute for Research and Executive Education.
* Where any part of this thesis has previously been submitted for a degree or any other qualification at VNUK Institute for Research and Executive Education or any other institution, this has been clearly stated.
* Where I have consulted the published work of others, this is always clearly attributed.
* Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
* I have acknowledged all main sources of help.
* Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

*[Name, Student ID and Department of the student 1]*

*[Name, Student ID and Department of the student 2]*

CERTIFICATE

I do hereby declare that the research works embodied in this thesis/project entitled “**Thesis or Project Title**” is the outcome of an original work carried out by [Student Name (s)]\* under my supervision.

I further certify that the dissertation meets the requirements and the standard for the degree of [MSc/ BSc] in Computer Science and Engineering.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Name and designation of Supervisor*

*Dedicated to*

*My Parents and my Family*ACKNOWLEDGEMENT

*Here you can thank to the family, your project advisor, friends, people or institutions that helped.*

Da Nang, 2019 *Student name 1*

*Student name 2*

ABSTRACT

*You can give some brief description about your project and motivation of doing this project. This section must have between 200 and 600 word length. Keywords are ideas and topics that define what your content is about. It can be a word or a phrase.*

**TABLE OF CONTENTS**

[DECLARATION 3](#_Toc14479269)

[CERTIFICATE 4](#_Toc14479270)

[*Dedicated to* 5](#_Toc14479271)

[*My Parents and my Family*ACKNOWLEDGEMENT 5](#_Toc14479272)

[ABSTRACT 7](#_Toc14479273)

[LIST OF FIGURES 10](#_Toc14479274)

[LIST OF TABLES 11](#_Toc14479275)

[SYMBOLS 12](#_Toc14479276)

[ABBREVIATIONS 13](#_Toc14479277)

[CHAPTER 1: INTRODUCTION 14](#_Toc14479278)

[1.1 Purpose of the system 14](#_Toc14479279)

[1.2 Scope of the system 14](#_Toc14479280)

[1.3 Objectives and success criteria of the project 14](#_Toc14479281)

[1.4 Definitions, acronyms, and abbreviations 14](#_Toc14479282)

[1.5 Project plan 14](#_Toc14479283)

[1.5. 1 Gantt chart 15](#_Toc14479284)

[1.5.2 Project Phase Distribution 17](#_Toc14479285)

[1.5.3 Other Tables 18](#_Toc14479286)

[1.6 References 18](#_Toc14479287)

[1.7 Overview 18](#_Toc14479288)

[CHAPTER 2: METHODOLOGY AND THEORETICAL BACKGROUND 19](#_Toc14479289)

[2.1 METHODOLOGY 19](#_Toc14479290)

[2.2 THEORICAL BACKGROUND 20](#_Toc14479291)

[CHAPTER 3: SYSTEM ANALYSIS AND DESIGN 22](#_Toc14479292)

[3.1 CURRENT SYSTEM 22](#_Toc14479293)

[3.2 PROPOSED SYSTEM 22](#_Toc14479294)

[3.2.1 Overview 22](#_Toc14479295)

[3.2.2 Functional Requirements 22](#_Toc14479296)

[3.2.3 Nonfunctional Requirements 22](#_Toc14479297)

[3.2.4 System Models 22](#_Toc14479298)

[3.2.4.1 Scenarios 23](#_Toc14479299)

[3.2.4.2 User case model 23](#_Toc14479300)

[3.2.4.2.1 User case diagram 23](#_Toc14479301)

[3.2.4.2.2 User case table 26](#_Toc14479302)

[3.2.4.3 Object Model 29](#_Toc14479303)

[UML Activity Diagram Example 29](#_Toc14479304)

[3.2.4.4 Dynamic Model 32](#_Toc14479305)

[Scheduled Workflow 32](#_Toc14479306)

[UML Information Flow Diagram Example 32](#_Toc14479307)

[3.2.4.5 User Interface – Navigational Paths and Screen Mock-Ups 34](#_Toc14479308)

[3.2.4.5.1 Website sitemap 34](#_Toc14479309)

[3.2.4.5.2 Details 34](#_Toc14479310)

[3.2.5 GLOSSARY 39](#_Toc14479311)

[CHAPTER 4: IMPLEMENTATION AND RESULT 40](#_Toc14479312)

[4.1 Accessibility 40](#_Toc14479313)

[4.2 User Manual 40](#_Toc14479314)

[4.3 Test Plan 40](#_Toc14479315)

[4.4 Maintenance Plan 40](#_Toc14479316)

[4.5 Setup and Installation 40](#_Toc14479317)

[CONCLUSION AND DISCUSSION 41](#_Toc14479318)

[REFERENCES 42](#_Toc14479319)

[APPENDIXES 43](#_Toc14479320)

[BIOGRAPHY 44](#_Toc14479321)

[44](#_Toc14479322)

LIST OF FIGURES

[Figure 5.1 Gantt chart sample 1 15](#_Toc11165858)

[Figure 5.2 Gantt chart sample 2 made with excel 15](#_Toc11165859)

LIST OF TABLES

[Table 5.1 Project budget table 16](#_Toc11226580)

[Table 5.2 Project phase distribution 16](#_Toc11226581)

SYMBOLS

β Sample Test Phase

ABBREVIATIONS

VS Visual Studio

CHAPTER 1: INTRODUCTION

*The first section of the RAD is an Introduction. Its purpose is to provide a brief overview of the function of the system and the reasons for its development, its scope, and references to the development context (e.g., reference to the problem statement written by the client, references to existing systems, feasibility studies). The introduction also includes the objectives and success criteria of the project.*

## Purpose of the system

|  |
| --- |
|  |

## Scope of the system

|  |
| --- |
|  |

## Objectives and success criteria of the project

|  |
| --- |
|  |

## Definitions, acronyms, and abbreviations

|  |
| --- |
|  |

## Project plan

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *A project plan is one of the most important ingredients for a successful project. It is used to document and communicate expectations, to control schedule and delivery, and to calculate and manage risks. In general, it acts as a roadmap for everyone involved in the project.* 1 Gantt chart Những chart rộng quá thì nên xoay ngang để dễ nhìn      https://www.smartsheet.com/sites/default/files/complete_Timeline.png  Figure 5.2 Gantt chart sample 2 1.5.2 Project Phase Distribution In this section  Table 5.2 Project phase distribution   |  |  |  |  | | --- | --- | --- | --- | | **Planning & Analysis** | **Design** | **Implementation** | **Testing** | | %40 | %30 | 20 | %10 |  1.5.3 Other Tables You can extend this section as 5.3 Log tables, 5.4 Advisor meeting tables, or other tables or diagrams as you wish. |

|  |
| --- |
|  |
|  |

|  |
| --- |
| ReferencesOverview |
|  |

CHAPTER 2: METHODOLOGY AND THEORETICAL BACKGROUND

## METHODOLOGY

Bên dưới chỉ là ví dụ, tìm lý thuyết tương ứng copy vào

This study implements qualitative methods in the research stage of the hour tracking application development. The applied methods in this case study are interview, analysis, and benchmarking. The results from the qualitative methods have great impact on development of well designed and functioning mobile application. In the following subsections are described the methods used during the research stage of the thesis, and the design and development of the mobile application

* + 1. **Software Development Life Cycle**

The process of structuring and building the mobile application is based on the Software Development Life Cycle (SDLC). SDLC is the process used in the software industry for assuring design and development of excellent quality software (Justin 2013). In Figure 2: Software Development Life Cycle is displayed the five stages of SDL: • Planning, defining and analyzing the user requirements; • Designing the product; ……

tìm trên mạng copy lý thuyết và hình ảnh vào

* + 1. **Agile Development Method**

The SDLC Model chosen for this case study is the Agile Model. Agile SDLC focuses on quick delivery of functioning software product by adaptable process and constant user feedback. In Figure 3 Agile Model is illustrating the series of steps during the agile development process. The Agile Model was the most suitable model because it delivers functioning software fast, it receives user feedback constantly and it gives freedom for th ….

tìm trên mạng copy lý thuyết và hình ảnh vào

* + 1. **Wireframe and prototype**

The wireframe is used for graphically illustrate the visual layout and the functions of the mobile application. It is also used for representing the vision of the product to the company and receiving feedback on the design and functional ideas. In this case study the wireframe was established to demonstrate the ideas for design and functionalities of the mobile application in early stage of the project. The wireframe was presented to Evolvit Oy for receiving a feedback, suggestions, or change requests.

The prototype is a mock-up example, which development aims to represent the visual representation and behaviour of the end product. The prototype allows you to create interactive wireframe, and visualize the design better before development of the product (Banfield 2009). 14 In this case study the prototype was developed for better visual re-creation of the mobile application. The prototype allowed to test the design and see it on a real mobile device. The prototype represented the wireframe to look and feel like a real mobile application, which provided more accurate feedback from the company Evolvit Oy.

tìm trên mạng copy lý thuyết và hình ảnh vào

* + 1. **Source code control**

Dùng lý thuyết gì để quản lý source code, copy lý thuyết và hình ảnh vào

## THEORICAL BACKGROUND

Phần này viết về lý thuyết công nghệ, có công nghệ gì thì cứ copy vào từng mục nhỏ

To have better understanding of how the mobile applications work with SOAP and WSDL, one must have better knowledge of what is web services and how they work. The following subsections will introduce the web services, SOAP and other technologies as Web Service Description Language (WSDL) and XML.

**2.2.1. Overview of web service technology**

**2.2.2. Web service architechture**

Nowadays, there many architectures for building a powerful web application. Beside known stacks like MAMP, MEAN, etc. MERN

Our Web Application was built by MERN (\*) Stack:

* NodeJS (v10.16.0)
* ExpressJS: (v4.16.4)
* For client side:
  + React (v16.8.6)
  + Redux: (v4.0.1)
* MongoDB

More detail about MERN stack will be explained in glossary section.

In development, all both the client side and backend side will hosted on one server with two different ports and using proxy to forward client’s requests to backend server. This architecture provide flexible, reasonable for maintain.

**2.2.3. Understanding XML**

**2.2.4 Understanding face recognition algorithms**

The important part of our system is the face recognition phase but we are still in the entry of Machine Learning and Image processing, so we take advantage of our case study when training in AI Course of our advisor. With support of our advisor, it’s used for the face recognition process.   
 The face recognition application was built by Python and OpenCV2 with some helper libraries. It majorly used *dlib* for face detection and facial landmark detection. CNN based face detector is the main algorithms.

CHAPTER 3: SYSTEM ANALYSIS AND DESIGN

## CURRENT SYSTEM

*The Current system describes the current state of affairs. If the new system will replace an existing system, this section describes the functionality and the problems of the current system. Otherwise, this section describes how the tasks supported by the new system are accomplished now.*

## PROPOSED SYSTEM

*The third section documents the requirements elicitation and the analysis model of the new system.*

## Overview

*The overview presents a functional overview of the system.*

## Functional Requirements

* The VNUK-AMS mobile application must be able to capture student’s face
* The VNUK-AMS system must be able to export student’s ID with a picture of student’s face from input
* The VNUK-AMS system must be able to connect with existed database on Sociss System
* The VNUK-AMS system must be able to import events from Google Calendar
* The VNUK-AMS system must be able to show schedule of all lessons, classes by using information collected above
* The VNUK-AMS system must be able to show attendance status of students in each class.
* User must be able to manually update attendance status beside automatically attendance by face recognition.
* User must be able to import events from specific Google Calendar account.

## Nonfunctional Requirements

* VNUK-AMS mobile app must installed on Android mobile devices.
* VNUK-AMS mobile app must show camera to capture student’s picture
* Students must be able to see their attendance status right on the mobile application.
* VNUK-AMS system will show the entire schedule of classes
* VNUK-AMS system must have a guideline process for importing event from Google Calendar.
* VNUK-AMS system must show attendance status and the last checked time.
* VNUK-AMS system must show all class events fetched from Google Calendar to local calendar

## System Models

*Describes the scenarios, use cases, object model, and dynamic models for the system. This section contains the complete functional specification, including mock-ups illustrating the user interface of the system and navigational paths representing the sequence of screens. The subsections Object model and Dynamic model are written during the Analysis activity.*

## Scenarios

## User case model

## User case diagram

**In the first case we tend to solve of this automation check-in attendance system is we need to import the events from a 3rd calendar service which is Google Calendar because it’s using for planning and scheduling the timetable of our university in current system. For the upgrading this existed system, the import events progress must be implemented to handle this problem. Google Calendar APIs already provided endpoints to get data from calendars of associated account.**

**The UML Diagram below will be describing the import events from Google Calendar progress.**



Figure 1 VNUK-AMS import events use case

**This action will be handled by end users with a visual guide from web application. Firstly, starting with the login to google account step and then request access to google calendar of that account. Besides, they need to allow permissions for that. After that, they are ready to fetch events and customize them, import them to database. The VNUK-AMS API server will connect with Google Calendar APIs and interact with them to fetch needed data. Finally, it will analyzing and parse events to model objecs which will be saved to collections in MongoDB database.**

## User case table

|  |  |
| --- | --- |
| **Name:** | Capture Picture |
| **Actor:** | STUDENT |
| **Entry**  **Conditions:** | Mobile application is running. |
| **Flow of**  **Events:** | 1. STUDENT capture a picture with capture button  2. MOBILE APP save with name of this class event.  3. MOBILE APP push that picture to VNUK-AMS server |
| **Exit**  **Conditions:** | Student’s picture captured now uploaded to server for handling the face recognition process. |

|  |  |
| --- | --- |
| **Name:** | Forward Picture to Face Recognition App |
| **Actor:** | SYSTEM |
| **Entry**  **Conditions:** | VNUK-AMS server is running. |
| **Flow of**  **Events:** | 1. VNUK-AMS server receive pictures from MOBILE APP through upload APIs.  2. VNUK-AMS update time check in field and rename picture file format.  3. VNUK-AMS send pictures to VNUK – Face recognition AI app at folder /raw |
| **Exit**  **Conditions:** | Successfully forward student’s pictures from Mobile App to Face recognition AI app |

|  |  |
| --- | --- |
| **Name:** | Picture Processing |
| **Actor:** | SYSTEM |
| **Entry**  **Conditions:** | VNUK Face recognition AI app is running as watching status |
| **Flow of**  **Events:** | 1. From pictures in /raw folder uploaded from VNUK-AMS server  2. Image Recognition Process, input student picture and output with StudentID matched with picture.  3. Face recognition AI app will update student attendance status with StudentID collected from student’s picture. |
| **Exit**  **Conditions:** | Updated attendance status of student |

|  |  |
| --- | --- |
| **Name:** | Login to VNUK-AMS system |
| **Actor:** | ADMIN |
| **Entry**  **Conditions:** | VNUK-AMS system is running |
| **Flow of**  **Events:** | 1. Click button “Login” to navigate to Login Page  2. Login with provided admin account |
| **Exit**  **Conditions:** | ADMIN logged in VNUK-AMS system |

|  |  |
| --- | --- |
| **Name:** | Logout of VNUK-AMS system |
| **Actor:** | ADMIN |
| **Entry**  **Conditions:** | VNUK-AMS system is running and logged in with admin account |
| **Flow of**  **Events:** | 1. Click button “Logout of [Admin email]” to log out  2. Redirect to Home Page |
| **Exit**  **Conditions:** | ADMIN logged out of VNUK-AMS system |

|  |  |
| --- | --- |
| **Name:** | View calendar of events |
| **Actor:** | ADMIN |
| **Entry**  **Conditions:** | VNUK-AMS system is running and logged in with admin account |
| **Flow of**  **Events:** | 1. Navigate to Home Page  2. Homepage will show the calendar contain all events at specific time. |
| **Exit**  **Conditions:** | ADMIN saw events calendar of VNUK-AMS system |

|  |  |
| --- | --- |
| **Name:** | Interact with Calendar |
| **Actor:** | ADMIN |
| **Entry**  **Conditions:** | VNUK-AMS system is running and logged in with admin account and Home Page showed with Calendar. |
| **Flow of**  **Events:** | 1. Navigate dates, months, on calendar interface.  2. Calendar filtered and showed with above selected field |
| **Exit**  **Conditions:** | ADMIN can interact with events calendar of VNUK-AMS system |

|  |  |
| --- | --- |
| **Name:** | View Event Attendance Status |
| **Actor:** | ADMIN |
| **Entry**  **Conditions:** | VNUK-AMS system is running and logged in with admin account |
| **Flow of**  **Events:** | 1. Select specific event from schedule  2. View event detail and list of students come with checkbox present uncheck and checked status |
| **Exit**  **Conditions:** | VNUK-AMS system displayed students attendance status of event |

|  |  |
| --- | --- |
| **Name:** | Manually set attendance status |
| **Actor:** | ADMIN |
| **Entry**  **Conditions:** | VNUK-AMS system is running and logged in with admin account |
| **Flow of**  **Events:** | 1. Select specific event from schedule  2. View event detail and list of students come with checkbox present uncheck and checked status  3. Trigger checkbox for check and uncheck status of student |
| **Exit**  **Conditions:** | VNUK-AMS system saved students attendance status of event |

|  |  |
| --- | --- |
| **Name:** | Show Class Detail |
| **Actor:** | ADMIN |
| **Entry**  **Conditions:** | VNUK-AMS system is running and logged in with admin account |
| **Flow of**  **Events:** | 1. Select Dashboard tab at sidebar  2. View class detail and table of students come with the number of event checked, unchecked and absent. |
| **Exit**  **Conditions:** | VNUK-AMS system show students attendance status of class in detail |

|  |  |
| --- | --- |
| **Name:** | Fetch event data from API Server to Mobile App |
| **Actor:** | MOBILE APP |
| **Entry**  **Conditions:** | Mobile application is running. |
| **Flow of**  **Events:** | 1. MOBILE APP will automatically perform fetching tasks.  2. MOBILE APP show details of event fetched. Analyzing and display absent status for student. |
| **Exit Conditions:** | Student’s picture captured now uploaded to server for handling the face recognition process. |

## Object Model

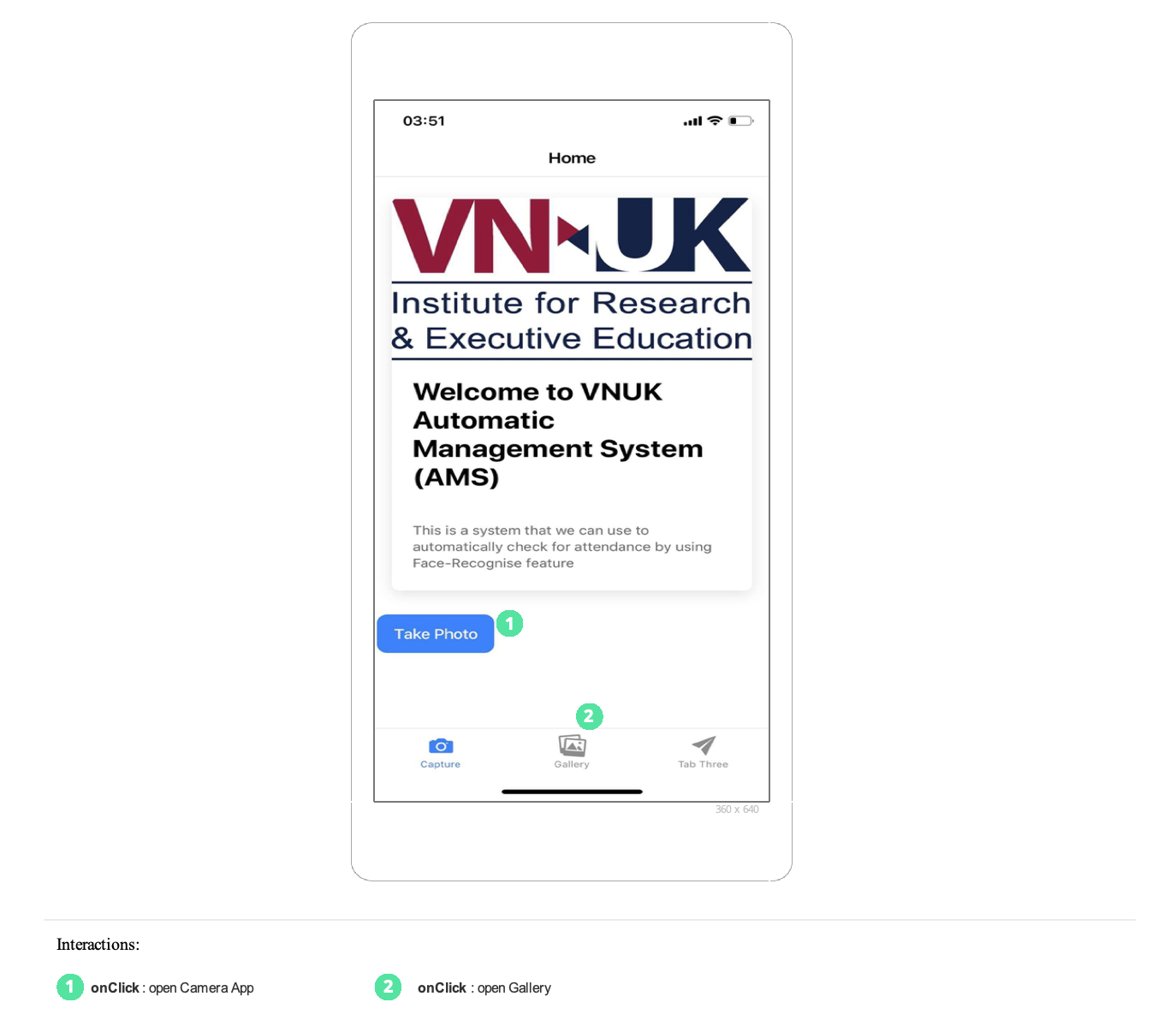
|  |
| --- |
| In this section, the activities and model class of our system will be figured out by two UML diagrams.  Activity is the whole progress of the main problem which our system wants to solve. Activity is started by Students who need to check-in when joining the classes. They will do a required action which is take a picture of their face by mobile devices in front of class rooms. The VNUK-AMS mobile app will be installed in devices and opened for actions related to check-in attendance status like taking a photo, sending it to API server for further processing and fetching updated attendance status from API send back. Take a look at the behavior activity diagram for understanding the workflow of our system.    Figure 2 Check-in attendance behaviour use cases diagram  The second UML diagram shows a domain model for VNUK-AMS system. The purpose of the diagram is to introduce some common terms, "dictionary" for our system - User, Student, Event, TimeTable, Class, and relationships between. It could be used as a common ground between business analysts and software developers.  Each student has unique id and is linked to exactly one user. Events owns a list of students come with *isChecked* boolean property. Event will linked with Class by through *classCode* property. The big TimeTable one will be hold these events of specific class for display calendar and analyzing purpose.  The left side of diagram above is describing the functions, properties of Controllers and Boundaries.    Figure 3 VNUK-AMS system domain UML class diagram |

## User Interface – Navigational Paths and Screen Mock-Ups

## VNUK-AMS client side sitemap

## Details

|  |
| --- |
|  |
|  |



## GLOSSARY

*A glossary of important terms, to ensure consistency in the specification and to ensure that we use the client’s terms. A precursor to the****Data Dictionary.***

* MERN Stack

The MERN stack is a popular stack of technologies for building a modern single-page application. In this tutorial, you will learn the MERN stack by building an exercise tracker.

The MERN stack consists of the following technologies:

* + MongoDB: A document-based open source database.
  + Express: A web application framework for Node.js.
  + React: A JavaScript front-end library for building user interfaces.
  + Node.js: JavaScript run-time environment that executes JavaScript code outside of a browser (such as a server).
* UML

The Unified Modeling Language™ (UML®) is a standard visual modeling language intended to be used for

* modeling business and similar processes,
* analysis, design, and implementation of software-based systems

UML is a common language for business analysts, software architects and developers used to describe, specify, design, and document existing or new business processes, structure and behavior of artifacts of software systems.

UML can be applied to diverse application domains (e.g., banking, finance, internet, aerospace, healthcare, etc.) It can be used with all major object and component software development methods and for various implementation platforms (e.g., J2EE, .NET).

* Google Calendar APIs

To interact with google calendar, we are using *googleapis* nodejs client libary which provided various APIs for our fetching events purpose.

NPM package: **google-api-nodejs-client**

Github: <https://github.com/googleapis/google-api-nodejs-client>

CHAPTER 4: IMPLEMENTATION AND RESULT

In this chapter, you can write anything about your project. User manual, screenshot, essential part of your project, whatever you want...

You can extend this chapter as 6.6, 6.7 and so on.

|  |
| --- |
| Accessibility |
| Base on the moderate ground knowledge of programming, we found it not too difficult to build the system, not to mention the help from Dr. Vu. It cost us approximately 2 weeks to complete the project outline then we started digging into documents about technologies and programming languages. |

|  |
| --- |
| User Manual |

1. VNUK AMS Website

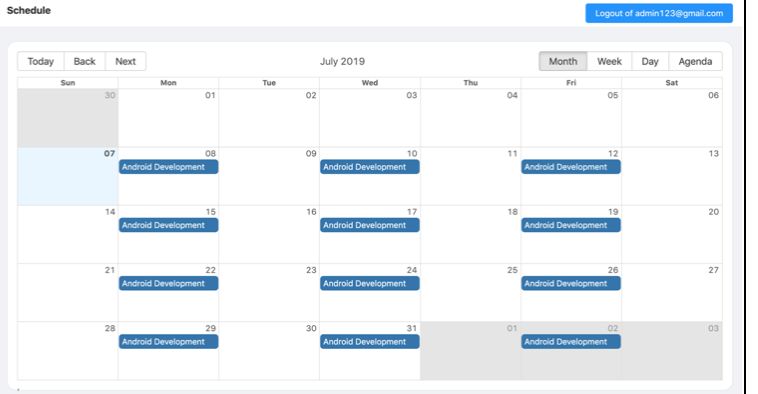
* By following the installation guides in 6.5 Setup & Installation section below, systems are running and end up with locating to Landing Page of VNUK AMS client and also a test admin account.
* Navigate to login page by pressing button “Login”



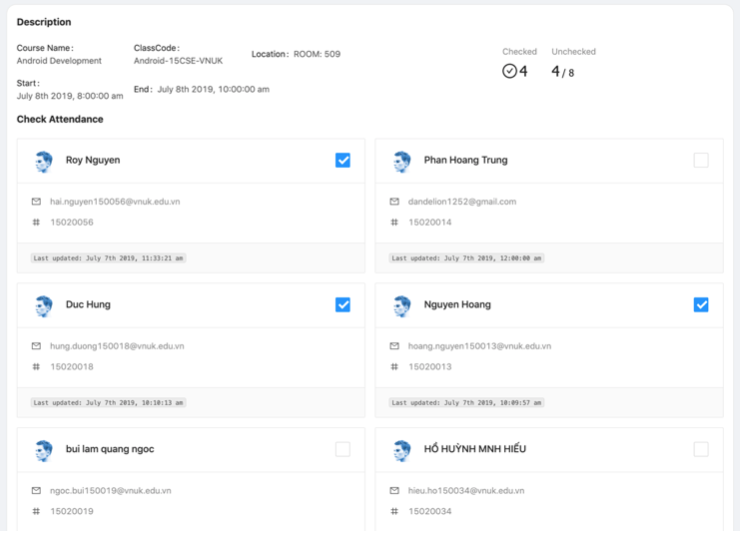
* Use test account provided to login to system



* After login successfully, all events will be show on the calendar of Home Page.



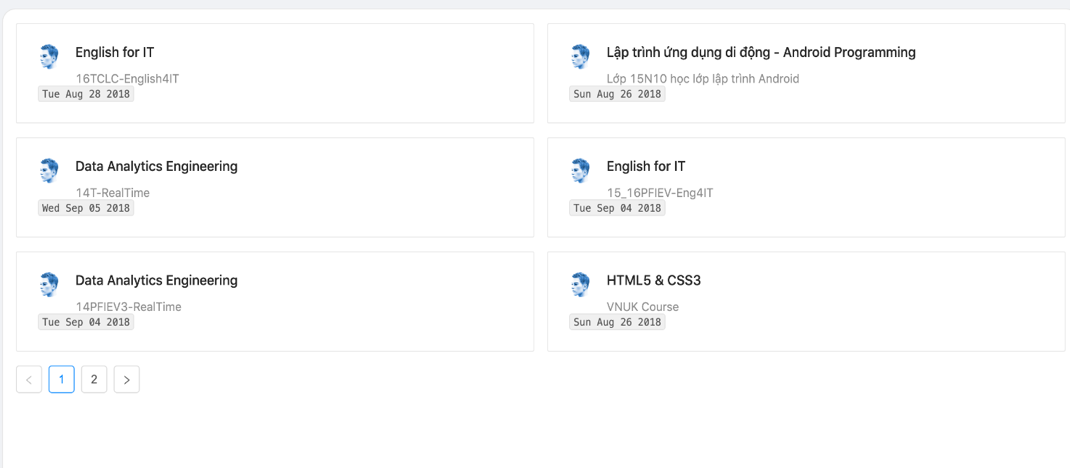
* By click on event blue box, Event Detail page will be navigated and show the data of that event (including students of class associated with that event)



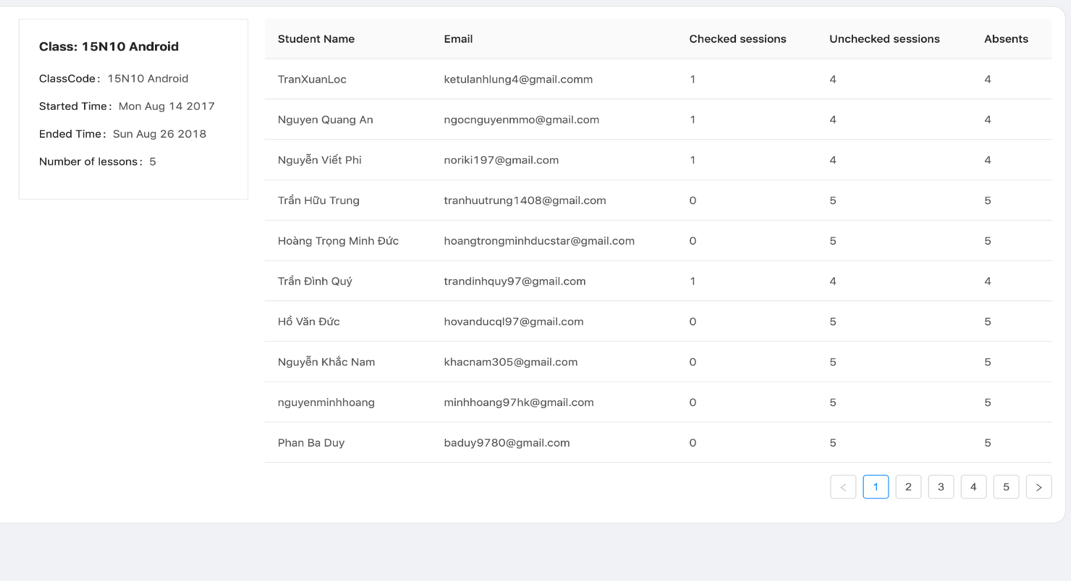
* Checkbox here for identify the check-in status of students fetched from result of AI Face Recognition App. To manually update check-in status, simply check on checkboxes.
* To view the statistics of student’s check-in, absent status, navigate to dashboard page by click *“Dashboard”* Tab on the right side-bar



* Dashboard page will show the list of classes, select a class to view details of classes.



* Class Detail will show up the check-in, absent statistic of class.



1. VNUK-AMS Mobile Application

* After launching the application, Information about the classroom is displayed on the screen right above the “Take Photo” button. The information includes: Classroom number, Timeline, Course, Class Code, etc …
* Press “Take Photo” button to open the device’s camera. Take a picture with the white round button. Afterwards, You are in.

|  |
| --- |
| Test Plan |
| 15-04: Analysing the product   * Mapping out risk potentials   10-05: Evaluating Testing criteria   * Defining successful criteria   29-05: WebView Testing after connected to Google APIs   * Checking the data displayed on website   01-06: 1st Website Testing Session (Smoke test)   * Running randomly testing on the website (UI, UX)   10-06: Mobile Application 1st build attempt on physical device (Failed)   * Trying to build the app on real device but the result is not optimistic due to errors while installing libraries (IPhone)   14-06: 2nd Website Testing Session   * Successfully processing imported images (locally)   20-06: Mobile Application 2nd build attempt on physical device (Succeeded)   * Completely deploying the app on physical device (Samsung)   25-06: Mobile Application Testing Phase 1 (Primary flow of the app finished)   * The mobile app is now able to take a picture and send to server to analyse   30-06: 3rd Website Testing Session (85%)   * After handling the image, the website can now receive the response and update the viewing on the user interface   06-07: Mobile Application Testing Phase 2 (Improving workflow)   * Improving the process of taking picture and sending phase   10-07: 4th Website Testing Session (100%)   * Successfully finalizing the workflow (Receive image => AI Analysing => Sending response => Update UI)   15-07: Mobile Application Testing Phase 3 (Final)  The app can now provide the classroom information and taking pictures of students and sending it to the server afterwards. |

|  |
| --- |
| Maintenance Plan |
| The maintenance phase is very important, that is the key factor to keep your customers stay. And as a service provider, there is no doubt about the responsibility we have to VNUK. |

|  |
| --- |
| Setup and Installation |
| Following the instruction below to install applications in VNUK-AMS system including website, mobile app and AI Face Recognition app.  This is installation instruction for development purpose, not production builds.   * **VNUK AMS Website**   Including VNUK-AMS client and API server.   * Setup and config required resources   All external required resources will be located in **resources** folder which is will be provided beside source code including:   * + *vnuk-ams-database.zip*: unzip this folder for import database in next step * Import and config database   This step is for importing *vnuk-ams-database* to local database.   * Install MongoDB   This system using MongoDB, the installation instruction provided in Appendix.   * Config mongoURL to connect database:   + Open keys.js located at ./config   + Replace value of mongoURL field with new correct url: default is “ mongodb://localhost:27017/vnuk-ams”and your will be “mongodb://localhost:27017/*[your database name imported]”* * Install project technologies   VNUK-AMS website is based on NodeJS/npm so your local system must have Node installed   * Install Node/npm environment ( Installation instruction located on Appendix ) * Following next steps with opened terminal/CMD and located to project *root* directory * Install dependencies for server and client:   + Server packages: From the project *root* directory, run command *yarn install or npm install*   + Client packages: Go to *client* folder: cd client and then run command *yarn install or npm install* * Run application   + From root directory, run command *npm run dev* to start server and client app ( this command will concurrently running both client and API server )   + Your browser will be navigate to [http://localhost:3000](http://localhost:3000/) as client application running which built with React and Redux   + Also, you will endup with another port running is api server [http://localhost:5000](http://localhost:5000/) * Open browser and located to <http://localhost:3000>, the application is running. * Test account: located at */database/test-account.json* * Important: The client application was already proxy configured to forward APIs requests from client with port 3000 to port 5000 of API server.   + *For example:* instead of *http://localhost:5000/api/\**, the actual api address is *http://localhost:3000/api/\*.* All request following api/\* from port 5000 will be fowarded to port 3000 * Test account: located at /database/test-account.json * **AI Face Recognition App**   The source code will be located inside the VNUK-AMS website root directory at *./face-attendance*   * Install python3, OpenCV and essential packages (the installation instruction located at Appendix) * Setup and config required resources   All external required resources will be located in **resources** folder which is will be provided beside source code including:   * + *shape\_predictor\_68\_face\_landmarks.dat*: Copy this file to source code folder with path: **./face-attendance/models/** * Run application:   + AI Face recognition App will be run as the second service besides VNUK-AMS Website, so it will be running concurrently with VNUK-AMS website.   + Open a new Terminal/CMD and located to .*/face-attendace*directory   + Run python3 face\_watcher.py   + Application will turn to watching mode.   After the whole installation, the VNUK-AMS system will be end up with:   * VNUK-AMS Website with client app and API server up and running * AI Face-recognition app are in watching picture mode * VNUK-AMS Mobile… * VNUK-AMS MOBILE APPLICATION * Cloning project from this link <https://bitbucket.org/hai_dev/final/src/master/> * Download the project from a remote repository * Installing Visual Studio Code (Installation instructions located in Appendixes) * By using VSCode we can compile the project to make it readable * Opening the project by Visual Studio Code * Open VSCode and Select “Open a folder”, Select “Final” folder you have just downloaded * Connecting with a mobile device * Connecting the device with the laptop by using USB cable * Installing Node Package Manager (Installation instructions located in Appendixes) * It is the World’s Largest Software Registry (Library). By installing it, we can access to tools to help developing the app * Installing Android Debug Bridge (ADB) (Installation instructions located in Appendixes) * It is versatile command-line tool that lets us communicate with the device that we have connected * Opening Project’s Debug Console (Instructions located in Appendixes) * The console is a place where we can execute commands * Checking if the device is connected by typing “adb devices” in the Debug Console (Name of the device will be displayed right under the command) * Typing “sudo npm i -g ionic cordova” to install the tools (Type your laptop’s password the provide the access) * Typing “npm i” to install the packages within the app * Typing “ionic cordova run android -l” to build the app to connected device. |

CONCLUSION AND DISCUSSION

*Write your final words about your great project. Discuss the result and point out how this project is reported as successful. Its length is at least 200 words.*

We would not have come this far if we do not have the necessary support from teachers. At first, the idea is just an idea, it is more likely we shared the same thought. We did not know how to make it, how to analyze an image then return data, etc … But coincidentally, VNUK scheduled us an AI course which we just realized how helpful it is.

We were having a lot of troubles as other teams and it is no easy to turn an idea into a product that we are presenting today. Fortunately, we figured out the solutions for our problem by researching tons of articles, books and even videos. Without research skills we have learnt in the past 4 years at VNUK, we do not think we can make it out alive.

Until today, the system is basically finished with primary feature: Face-Recognition. Students can now check in using their face as FaceID. On the management side, staffs can check the status regularly and observe the data by using our website which is friendly to users.

In the future, we will develop more feature as well as improving the qualification of the system. Test plans and maintenance schedules will always be update.

REFERENCES

*Includes all references: articles, media facts, books, reports, regulations, internet articles, papers that you referenced from the text.*

*Some examples are given below.*

*[1] P. Kessler, “Ericsson IMS client platform,” Ericsson Review, vol. 2, pp. 50–59, 2007.*

*[2] R. Levenshteyn, and I. Fikouras, “Mobile services interworking for IMS and XML Web Services,” IEEE Communications Magazine, pp. 80–87, Sep. 2006.*

*[3] D. Lozano, L.A. Galindo, and L. Garcia, “WIMS 2.0: Converging IMS and Web 2.0. Designing REST APIs for the exposure of session-based IMS capabilities,” in Proc. of The Second International Conference on Next Generation Mobile Applications, Services, and Technologies, 2008, pp. 18–24.*

APPENDIXES

*Type or paste your appendices here. Appendices are a place to organize and include all of the “extra” material that is important to your research work but that is too detailed for the main text.  Examples can include: specific analytical methods, computer code, spreadsheets of data, details of statistical analyses, etc. But, these materials do not speak for themselves. There should be a reference to these materials from the main chapters (complete details included in Appendix A) and there should be some text at the beginning of each appendix to briefly explain what the information is and means that is included in that appendix.*

BIOGRAPHY



Student name 1



Student name 2