

Student Attendance Recording System

Woo Wing Hong

**A proposal submitted in partial fulfilment of the
requirements for the award of Bachelor of Science
(Hons.) Software Engineering**

**Faculty of Engineering and Science
Universiti Tunku Abdul Rahman**

August 2014

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at UTAR or other institutions.

Signature : _____

Name : Woo Wing Hong

ID No. : 12 UEB 07679

Date : _____

APPROVAL FOR SUBMISSION

I certify that this project report entitled “**Student Attendance Recording System**” was prepared by **Woo Wing Hong** has met the required standard for submission in partial fulfilment of the requirements for the award of Bachelor of Science (Hons.) Software Engineering at Universiti Tunku Abdul Rahman.

Approved by,

Signature : _____

Supervisor : Lim Khong Leng

Date : _____

The copyright of this report belongs to the author under the terms of the copyright Act 1987 as qualified by Intellectual Property Policy of Universiti Tunku Abdul Rahman. Due acknowledgement shall always be made of the use of any material contained in, or derived from, this report.

© 2014, Woo Wing Hong. All right reserved.

ACKNOWLEDGEMENTS

I would like to thank everyone who had contributed to the successful completion of this project. I would like to express my gratitude to my research supervisor, Mr. Lim Khong Leng for his invaluable advice, guidance and his enormous patience throughout the development of the research.

In addition, I would also like to express my gratitude to my loving parent and friends who had helped and given me encouragement throughout my studies.

TABLE OF CONTENTS

DECLARATION	ii
APPROVAL FOR SUBMISSION	iii
ACKNOWLEDGEMENTS	v
TABLE OF CONTENTS	vi
LIST OF TABLES	ix
LIST OF FIGURES	x
LIST OF SYMBOLS / ABBREVIATIONS	xii

CHAPTER

1	INTRODUCTION	1
	1.1 Summary	1
	1.2 Problem Statement	1
	1.3 Objective	2
	1.4 Project Scope	2
	1.5 Justification	3
	1.6 Technology Involved	4
2	LITERATURE REVIEW	5
	2.1 Student ID Card of UTAR	5
	2.2 Barcode	6
	2.2.1 Code 128	7
	2.3 Mobile Phones Barcode Reader	8

2.3.1	Barcode Scanner SDK Library	9
2.4	Existing System in the Market	9
2.4.1	Manual Punch Card Attendance System	9
2.4.2	Card Reader Attendance System	14
2.4.3	Fingerprint Recognition Attendance System	15
2.4.4	Face Recognition Attendance System	16
3	METHODOLOGY	18
3.1	Development Methodology	18
3.1.1	Waterfall Model	18
3.1.2	Prototype Model	19
3.1.3	Extreme Programming	20
3.1.4	Rapid Application Development	20
3.2	Adopted Development Methodology	22
3.3	Development Tools	22
3.4	Project Plan	23
3.4.1	Project Breakdown	23
3.4.2	Work Breakdown Structure	25
3.4.3	Gantt Chart	25
4	SYSTEM DESIGN	26
4.1	System Architecture	26
4.2	Use Case Diagram	27
4.3	Database Design	28
4.3.1	Entity Relationship Diagram	28
4.3.2	Data Dictionary	29
4.4	Behavioural Design	30
4.4.1	Take Attendance	30
4.4.2	Search and View Attendance	32
4.4.3	Modify Attendance	34

5	IMPLEMENTATION AND TESTING	37
5.1	Implementation	37
5.1.1	Barcode Scanner	37
5.2	System Testing	39
5.2.1	Mobile Application Testing	39
5.2.2	Website Testing	53
6	CONCLUSION AND RECOMMENDATION	59
6.1	Limitation and Future Improvement	59
6.2	Conclusion	60
	REFERENCES	61

LIST OF TABLES

TABLE	TITLE	PAGE
1.1	Technology Involved	4
2.1	ASCII Table	8
2.2	Comparison between Different Types of Cards	15
2.3	Advantages of Fingerprint System	15
4.1	Description of Entities	28

LIST OF FIGURES

FIGURE	TITLE	PAGE
2.1	Student ID Card of UTAR	5
2.2	Result of Barcode Scanner Application	6
2.3	Linear (1D) Barcode	7
2.4	Matrix (1D) Barcode	7
2.5	Punch Card Clock Machine	10
2.6	System Architecture of Card Reader System	11
2.7	Proximity Card and Reader	11
2.8	Contact Smart Card and Reader	12
2.9	Contactless Smart Card	12
2.10	Contactless Smart Card Reader	13
2.11	Magnetic Stripe Card and Reader	13
2.12	Price Tag with Barcode	13
2.13	Fingerprint Recognition Attendance Recording Machine	15
2.14	Summarized Flow of Feature Extraction	16
2.15	Process Flow of Feature Extraction	16
2.16	Face Recognition Attendance Recording Machine	17
3.1	Waterfall Model	18
3.2	Prototyping Model	19

3.3	Extreme Programming Process	20
3.4	Rapid Application Development Methodology	21
3.5	Modules of the Project	22
3.6	Work Breakdown Structure	25
3.7	Gantt Chart	25
4.1	System Architecture of the System	26
4.2	Use Case Diagram of the System	27
4.3	ERD of the System	28
4.4	Activity Diagram – Take Attendance	31
4.5	Sequence Diagram – Take Attendance	32
4.6	Activity Diagram – Search and View Attendance	33
4.7	Sequence Diagram – Search and View Attendance	34
4.8	Activity Diagram – Modify Attendance	35
4.9	Sequence Diagram – Modify Attendance	36
5.1	Section of Code – Import ZXing Project	37
5.2	Section of Code – Start Barcode Scanner	38
5.3	Section of Code – Get Scanner Result	38

LIST OF ABBREVIATIONS

UTAR	University Tunku Abdul Rahman
ID	Identification
OS	Operating System
XML	Extensible Markup Language
1D	One-Dimensional
2D	Two-Dimensional
DSP	Digital Signal Processing
SDK	Software Development Kit
RF	Radio Frequency
ATM	Automated Teller Machine
API	Application Programming Interface
SE	Software Engineering
XP	Extreme Programming
RAD	Rapid Application Development
ERD	Entity Relationship Diagram

CHAPTER 1

INTRODUCTION

1.1 Summary

This Student Attendance Recording System is developed for University Tunku Abdul Rahman (UTAR). This system will provide a mobile application for lecturers of the university to install and use it during class. The application will allow the users to record student attendance, and it will automatically update to the server. By using the application, the student attendance will be updated automatically. The purpose of this system is to facilitate the lecturer by reducing the manual works needed to record and update student attendance.

1.2 Problem Statement

In many universities, such as UTAR, student's attendance is recorded in the manual way. Lecturers record the student attendance by distributing the name list to students, and let the students to sign their signature on the name list by themselves. In this case, a student may help other student to sign. This action can be considered as cheating.

Besides that, updating the student attendance is a manual task. They refer to the name list to update every single student's attendance. If the number of student of a class is large, updating student attendance will takes a lot of time. And also, a

lecturer will be assigned to handle at least more than one class. So, a simple task of updating student attendance record has turned to a tiring and time consuming task.

Furthermore, it will be very troublesome if the lecturer did not bring the name list to the class or the name list is missing. In this case, lecturer needs to think a way to record the student attendance. By giving a real life example, if the lecturer did not bring the name list to the class, he/she will record the attendance by ordering all the student to write down their name and student ID on a paper.

1.3 Objective

The main purpose of this project is to introduce a new method for UTAR to handle the task of recording and updating student attendance. The new method has four objectives need to be achieved:

- To provide a mobile application for lecturers to record student attendance.
- To provide a mobile application for lecturers to view and modify student attendance.
- To provide a website for admin and lecturers to search, view and modify student attendance.
- To provide a mobile application for students to check their attendance record.

1.4 Project Scope

Modules covered in this project:

1) Lecturer Module:

- Lecturers able to use the mobile application to record student attendance.

- Lecturers able to use the mobile application and website to view student attendance and attendance percentage.
- Lecturers able to use the mobile application and website to modify student attendance.
- Lecturer able to use the website to search student attendance.

2) Student Module:

- Student able to use the mobile application to view their attendance of all enrolled subject.

3) Admin Module

- Admin able to use the website to search, view and modify student attendance.

4) Server Module:

- Server able to complete client's requests.
- Server able to retrieve data from database.
- Server able to update the data in database.

5) Database Module:

- Store data of staffs, students, classes, and student attendance.

Due to constraint of time and cost:

- System developed is not integrated with system of UTAR.
- Data is hardcoded into database.

1.5 Justification

The reason for developing this system is to suggest a new method for UTAR's lecturers to record and update student attendance. Besides UTAR, other universities or colleges such as TARUC (Tunku Abdul Rahman University College), Taylors

University, HELP University College, SEGI University College, UCSI University are still using the same method to record and update the student attendance.

Besides that, this system is also developed to reduce the workloads of updating the student attendance. Currently, the method used by UTAR is very time-consuming. The larger class size, the longer time needed to update the student attendance. It is because the student attendance is updated one-by-one by a single person.

1.6 Technology Involved

The deliverable of Student Attendance Recording System is an Android application which will be installed on users' mobile devices. According to Android official website, the technologies involved in development are shown in Table 1.1:

Table 1.1: Technology Involved

Hardware	<ol style="list-style-type: none"> 1) Personal computer 2) Mobile phones with Android OS 3) Camera of mobile phones
Software	<ol style="list-style-type: none"> 1) OS: Microsoft Windows, Android 2) Programming Language: Java, XML, PHP, HTML, CSS, JavaScript (JQuery) 3) Server: Apache Web Server 4) Database: MYSQL Database 5) IDE: Android Studio 6) Web browsers

CHAPTER 2

LITERATURE REVIEW

2.1 Student ID Card of UTAR

The use of ID card is mainly to verify the identity of the person who holding it. Universities issue ID card to students so they can use it as a proof to prove that he/she is a student of the university. Normally, ID card only contains the essential information such as ID number, name of the person and other identity information. Figure 2.1 shows the student ID card of UTAR.

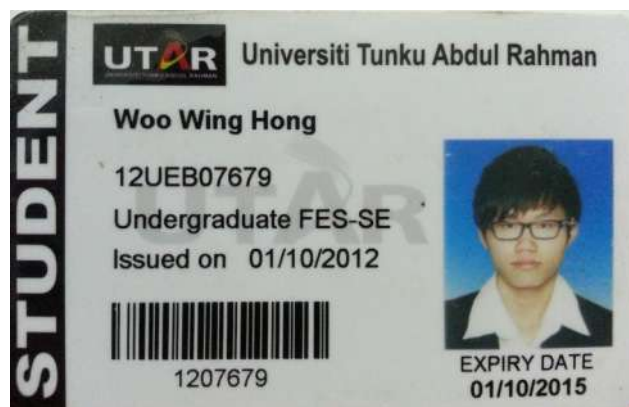


Figure 2.1: Student ID Card of UTAR

Student ID card of UTAR contains barcode of student ID number. The barcode is a 1D barcode, with format CODE 128, and the type of information set is a simple text. To identify the barcode format and type of information set, users can use existing scanner application to identify the barcode format and its encode

information. Figure 2.2 shows the screenshot of the result of scanner application (Barcode Scanner by ZXing) after scanning the barcode on UTAR's student ID card.

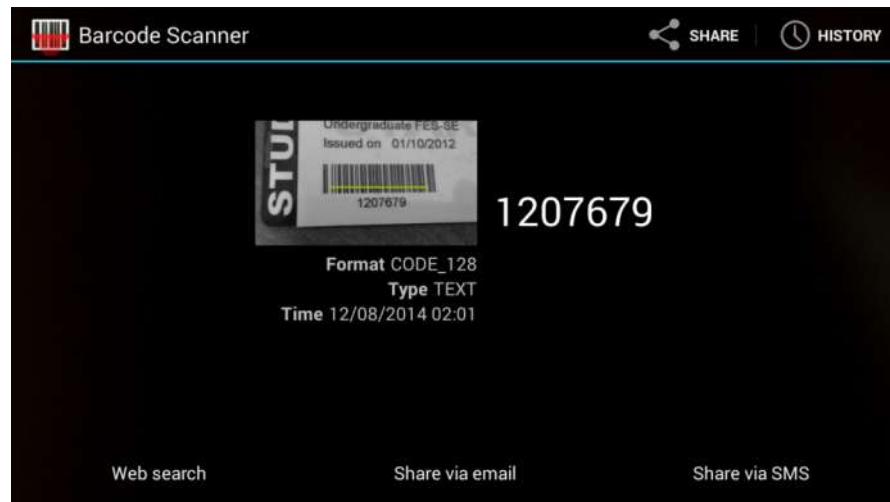


Figure 2.2: Result of Barcode Scanner Application

2.2 Barcode

A barcode is a pattern that consists of grouped dark bars and white spaces. Users encode data characters to generate barcode thus every barcode represent different data. It is a machine-readable pattern by using bar code scanner to decode it. In the current world, barcode is used to identify ID number of item such as retail sales product, books and others.

Based on community of IDAutomation Incorporation (2012), the design of barcode is to eliminate the manual data entry error rate. In the case of manually enters data by keyboard, typo error is always exist. By using barcode, error such as inaccurate or incorrect data can be reduced, at the same time shortens the working procedural.

There are many types of barcodes use, the most common used barcode are Linear (1D) Barcodes and Matrix (2D) Barcodes. There are also 3D Barcodes developed. Figure 2.3 and 2.4 show the Linear and Matrix barcodes.



Figure 2.3: Linear (1D) Barcode



Figure 2.4: Matrix (2D) Barcode

The main difference between Linear barcodes and Matrix barcodes is the Linear barcodes hold less than 85 characters, and Matrix barcodes can hold hundreds of characters.

2.2.1 Code 128

Code 128 Barcodes is one of the types of Linear Barcode. It is a very compact symbology that can encode all the 128 ASCII characters. Figure 2.5 shows the ASCII table.

Table 2.1: ASCII Table

Dec	Hx	Char	Dec	Hx	HTML	Char	Dec	Hx	HTML	Char	Dec	Hx	HTML	Char
0	0	NUL (null)	32	20	 	Space	64	40	@	@	96	60	`	`
1	1	SOH (Start of heading)	33	21	!	!	65	41	A	A	97	61	a	a
2	2	STX (Start of text)	34	22	"	"	66	42	B	B	98	62	b	b
3	3	ETX (End of text)	35	23	#	#	67	43	C	C	99	63	c	c
4	4	EOT (End of transmission)	36	24	$	\$	68	44	D	D	100	64	d	d
5	5	ENQ (Enquiry)	37	25	%	%	69	45	E	E	101	65	e	e
6	6	ACK (Acknowledge)	38	26	&	&	70	46	F	F	102	66	f	f
7	7	BEL (Bell)	39	27	'	'	71	47	G	G	103	67	g	g
8	8	BS (Backspace)	40	28	((72	48	H	H	104	68	h	h
9	9	TAB (Horizontal tab)	41	29))	73	49	I	I	105	69	i	i
10	A	LF (NL line fd, new line)	42	2A	*	+	74	4A	J	J	106	6A	j	j
11	B	VT (Vertical tab)	43	2B	+	+	75	4B	K	K	107	6B	k	k
12	C	FF (NP form fd, new page)	44	2C	,	,	76	4C	L	L	108	6C	l	l
13	D	CR (Carriage return)	45	2D	-	-	77	4D	M	M	109	6D	m	m
14	E	SO (Shift out)	46	2E	.	.	78	4E	N	N	110	6E	n	n
15	F	SI (Shift in)	47	2F	/	/	79	4F	O	O	111	6F	o	o
16	10	DLE (Data link escape)	48	30	0	0	80	50	P	P	112	70	p	p
17	11	DC1 (Device control 1)	49	31	1	1	81	51	Q	Q	113	71	q	q
18	12	DC2 (Device control 2)	50	32	2	2	82	52	R	R	114	72	r	r
19	13	DC3 (Device control 3)	51	33	3	3	83	53	S	S	115	73	s	s
20	14	DC4 (Device control 4)	52	34	4	4	84	54	T	T	116	74	t	t
21	15	NAK (Negative acknowledge)	53	35	5	5	85	55	U	U	117	75	u	u
22	16	SYN (Synchronous idle)	54	36	6	6	86	56	V	V	118	76	v	v
23	17	ETB (End of trans. block)	55	37	7	7	87	57	W	W	119	77	w	w
24	18	CAN (Cancel)	56	38	8	8	88	58	X	X	120	78	x	x
25	19	EM (End of medium)	57	39	9	9	89	59	Y	Y	121	79	y	y
26	1A	SUB (Substitute)	58	3A	:	:	90	5A	Z	Z	122	7A	z	z
27	1B	ESC (Escape)	59	3B	;	;	91	5B	[[123	7B	{	{
28	1C	FS (File separator)	60	3C	<	<	92	5C	\	\	124	7C	|	
29	1D	GS (Group separator)	61	3D	=	=	93	5D]]	125	7D	}	}
30	1E	RS (Record separator)	62	3E	>	>	94	5E	^	^	126	7E	~	~
31	1F	US (Unit separator)	63	3F	?	?	95	5F	_	_	127	7F		DEL

2.3 Mobile Phone Barcode Reader

There are several barcode reader applications available in the application store of mobile devices. Our mobile device is able to acts as a barcode scanner to decode the data of the barcode.

To achieve this, there are some processes have to be done by our mobile devices (Eisaku Ohbuchi, et al., 2004). The summarized processes include:

- Run the barcode reader application
- Capture barcode image from embedded camera of mobile device
- Do image processing with Digital Signal Processing (DSP)
- Decode the code from the processed image data
- Display the decoded result

The summarized process can be understand easily, but the actual process behind process (c) and (d) is very complex.

2.3.1 Barcode Scanner SDK Library

In development of Barcode Reader application, a Barcode Scanner SDK library is an important component.

Barcode scanner SDK library is a specific code library for implementing barcode reader application. Developers who wish to carry out barcode scanning feature in their mobile application, this SDK library must be added into their project.

There are several SDK libraries available in the market:

- 1) ZXing – Open source, supports Android
- 2) ZBar – Open source, supports only iOS
- 3) QuickMark – Free, supports Android, iOS, Mac OS and Microsoft Windows and Windows Phone
- 4) VisionSmartsBarcodeReader – Paid, supports Android, iOS and Windows Phone
- 5) RedLaser – Paid, supports Android, iOS and Windows Phone
- 6) Scandit – Paid, supports Android, iOS and Symbian

2.4 Existing System in the Market

Currently, there are three existing attendance system is available in the market. In this section, all the systems will be studied and analysed. The systems available in the market are:

2.4.1 Manual Punch Card Attendance System

In the early days of 1900s, punch card system was the primary method used by companies to store and retrieve data information. A punch card attendance system is a system where employees manually punch their card to a punch card clock machine at the arranged in-out time. The total working hours and overtime hours is recorded

on the card, then companies collect the cards and generate payroll based on the record on the card (FingerTec, 2007). Figure 2.5 shows the punch card and the machine.



Figure 2.5: Punch Card Clock Machine

The drawback of Manual Punch Card Attendance System is it takes up valuable time to collect and record time data on the card, especially to companies with large number of employees. In another situation, buddy punching is created where employees take advantage by having another person punch their card for them. This situation will lead to inaccurate or incorrect entry errors.

2.4.2 Card Reader Attendance System

A Card Reader Attendance System is an identification system that gets information from the card and perform action correlative to the card. This system can immediately captures information in the card, then send it to a central location where the data storage is located. This system typically consisting three important components:

1. Cards with information that are carried by the users
2. Card reader machine
3. Control unit and its hardware and software

A Card Reader System requires a control unit to control and manage the overall process of the system. This unit gets the information on the card and respond

to user with appropriate response. Figure 2.6 shows the system architecture of a Card Reader System.

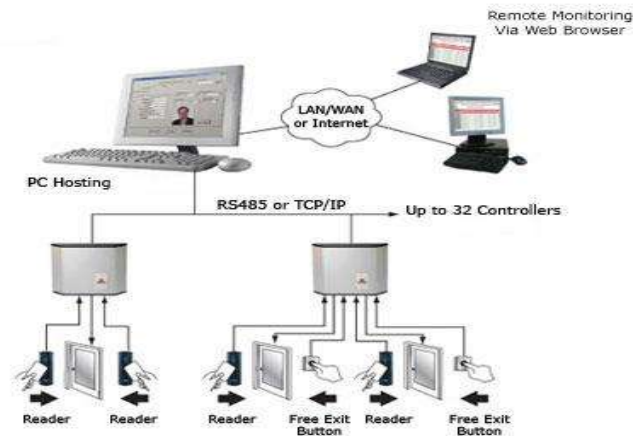


Figure 2.6: System Architecture of Card Reader System

There are three types of card available in the market. There are:

1) Proximity Card

Proximity card is a technology that embedding Radio Frequency (RF) circuit with encoded unique data. The data encoded in the card is transmitted through the radio wave to the card reader machine when two of them go near to each other. Figure 2.7 shows the proximity card and reader.



Figure 2.7: Proximity Card and Reader

2) Smart Card

Smart Card is a card that contains an embedded microprocessor. The microprocessor is a gold color chip that contains memory for data store. The

chip on the smart card acts as a contact pad, users insert the card into a smart card reader to read the data. There are two types of smart card:

a) Contact Smart Card

The embedded chip of contact smart card is visible. The transmission of data requires a physical contact point. Figure 2.8 shows the contact smart card and reader.



Figure 2.8: Contact Smart Card and Reader

b) Contactless Smart Card

Contactless smart card is similar to proximity card. The chip is embedded inside the card thus it is not visible to users. The transmission of data is also similar to the proximity card by going near to the smart card reader. Figure 2.9 shows the contactless smart card.

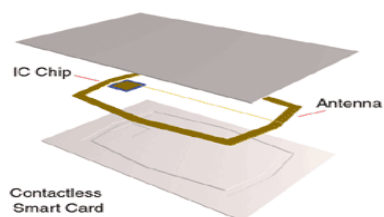


Figure 2.9: Contactless Smart Card



Figure 2.10: Contactless Smart Card Reader

3) Magnetic Stripe Card

Magnetic stripe card is also called swipe card. Users swipe past the card at a magnetic reader head and capture data on the card. Magnetic stripe card encodes data by using electromagnetic charges on the black tape. Magnetic stripe card is commonly used on card of bank such as ATM card and credit card. Figure 2.11 shows the magnetic stripe card and the magnetic card reader.



Figure 2.11: Magnetic Stripe Card and Reader

4) Barcode Card

Barcode card is the card with printed barcode that contain information. It is widely used on retail sales market. For example, the products are with a price tag that contains barcode which is storing price of the product. Figure 2.12 shows some example price tag with barcode.



Figure 2.12: Price Tag with Barcode

Every type of card has its advantages and disadvantages. In Table 2.2, it shows the comparison between these four types of cards.

Table 2.2: Comparison between Different Types of Cards

Type of Cards	Security Level	Cost	Vulnerability	Life Cycle
Proximity	High	Moderate	Virtually none	Long
Smart Card	Highest	Expensive	High	Short
Magnetic Stripe	Moderate	Inexpensive	Moderate	Moderate
Barcode	Low	Inexpensive	Easily damaged	Short

2.4.3 Fingerprint Attendance System

Fingerprint Recognition Attendance System is an attendance system which is able to record attendance data based on fingerprints. The fingerprints have to be registered and stored before using this system. Fingerprint attendance system provides better management and control of the human resources. Based on Dr.M.Ramakrishnan and Josphineleela.R (2012), using fingerprint attendance system is more efficient than using card-based attendance system. Table 2.3 shows the advantages and comparison of Fingerprint and Card-based system.

Table 2.3: Advantages of Fingerprint System

Advantages	Fingerprint System	Card-based System
Convenience in operation	Simple record attendance by using fingerprint	Have to carry card to record attendance
Better discipline control	Impossible to cheat	“Buddy punching” might occur
Lower cost	Longer lifespan, easy to maintain	Higher maintenance cost

Flexible deployment	Only consists of light weight machine	Space required for keeping cards
Reduce workloads	Automatic storing data once it is captured	Manually input data by referring to the cards
Better data management	Information is ready most of the time because it is stored immediately after captured	Data entry task have to be done first before using the data



Figure 2.13: Fingerprint Recognition Attendance Recording Machine

2.4.4 Face Recognition Attendance System

Face Recognition Attendance System is similar to Fingerprint Attendance System, both of the systems are using biometrics recognition technology to record attendance.

Face recognition technology has four main processes (Abhishek Jha):

a) Face Detection

This process is focus on separating the detected areas of the faces from the background of the image.

b) Face Alignment

This process focus on locating the position of the facial components, such as mouth, nose, eyes, ears and face outline.

c) Feature Extraction

This process is reducing the representation of an image. Figure 2.14 shows the summarized flow of feature extraction.

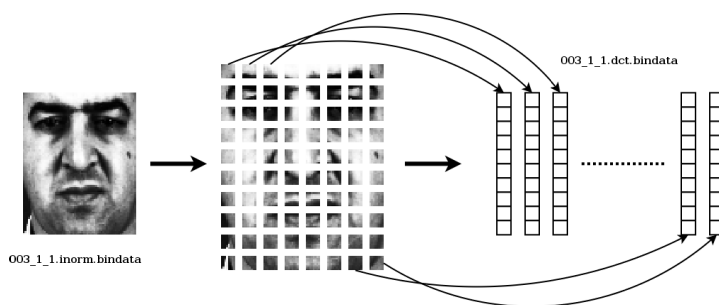


Figure 2.14: Summarized Flow of Feature Extraction

d) Face Matching

This process is matching the output of feature extraction with the data stored in database, and made decision based on the confidence in the matching process.

Figure 2.15 shows the simple process flow of face recognition.

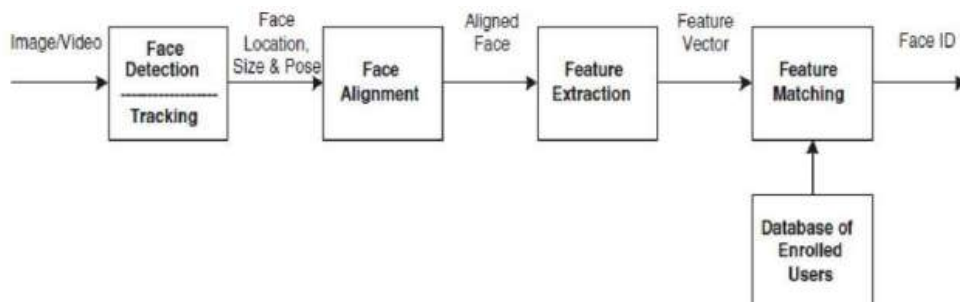


Figure 2.15: Process Flow of Face Recognition

The machine of the face recognition and fingerprint recognition machine are also quite similar based on their appearance, but the process behind face recognition is more complex. Figure 2.1 shows the face recognition attendance recording machine.



Figure 2.16 Face Recognition Attendance Recording Machine

CHAPTER 3

METHODOLOGY

3.1 Development Methodology

There are several software development methodology introduced on the field. In this section, several development models will be described, and one of the development method described will be chosen as adopted methodology of this project.

3.1.1 Waterfall Model

Waterfall model is a traditional model that provides a solid workflow. It was the first model that used in the Software Engineering (SE) field. This model provides a regularized workflow to ensure the project runs successfully. Figure 3.1 shows the stages available in waterfall model.

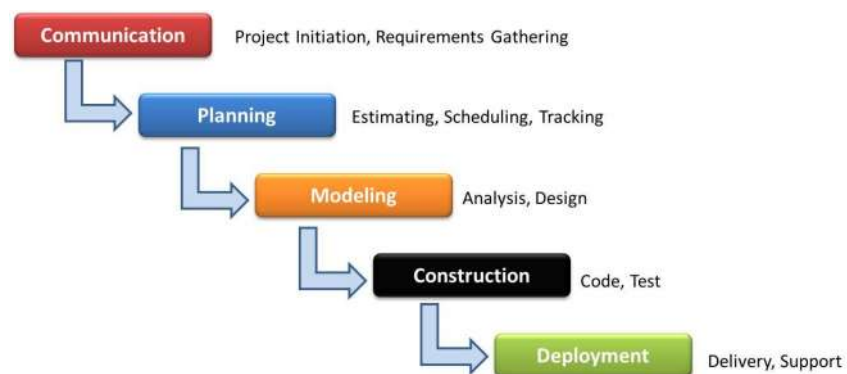


Figure 3.1: Waterfall Model

Waterfall model proposes a sequential development process which is following the rigid model shown in Figure 3.1. It has the advantage that this model is simple and straightforward and easy to understand. It is also easy to management because of the rigid workflow.

The disadvantage about waterfall model is also because of its development process. It is not suitable for software development where the change of requirement is unavoidable. Waterfall model is suitable when requirements are well planned and fixed.

3.1.2 Prototyping Model

Prototyping is an incomplete system that represents some aspect of the complete system. From the prototype, users can gather information and further elicit requirements from the incomplete system. Thus, prototyping model is a development method which refining prototype based on the collected feedback from the client. Figure 3.2 shows the life cycle of prototyping model.

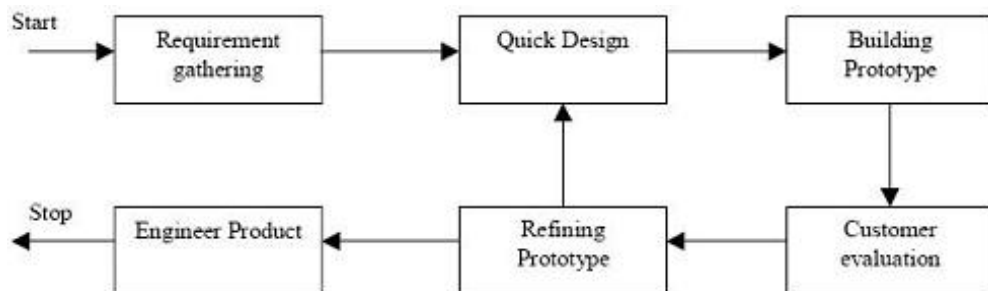


Figure 3.2: Prototyping Model

From the life cycle of prototyping model, this model requires active participation of user for requirement elicitation. It is suitable when the initial requirement of project is not clearly defined. After every user evaluation, a robust requirement set will be elicited.

The disadvantages of prototyping model are:

- a) Customer expecting fast completion because they might assume the prototype as final product
- b) High involvement of user resulting development impact

3.1.3 Extreme Programming (XP)

Extreme programming is one of the agile development methodologies. XP highly relies on user involvement. Users play an important role by writing “user stories” to describe the requirements of the software should have. The user stories act as guide, it helps developer team to estimate the time and resources to build the software.

The developments tasks of XP are divided into iterations. Before an iteration ends, users will perform acceptance tests. Thus, users satisfaction is slightly higher compared to others methodology. Figure 3.3 shows the simplified XP process.

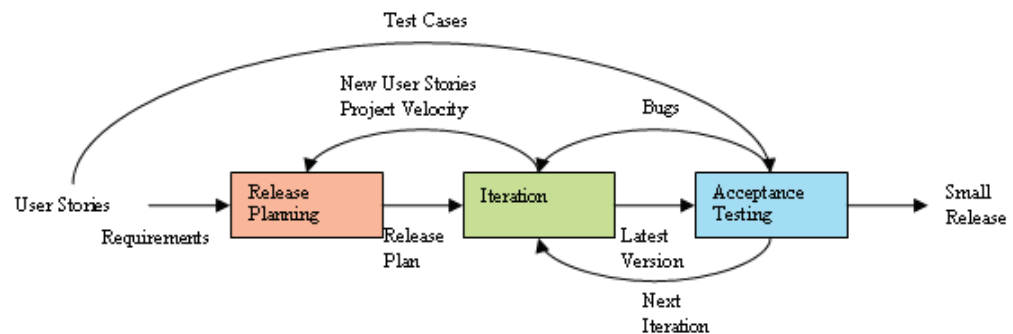


Figure 3.3: Extreme Programming Process

3.1.4 Rapid Application Development (RAD)

Rapid Application Development is a prototype-based and incremental software development methodology. It able to shortens the total development time and produces a high quality outcome.

RAD methodology aims to:

- a) Timely delivery a complete application
- b) Help customer to identify requirements properly
- c) Use prototype and iterations to deliver requirements identified by customer

RAD develops the project modules in parallel as prototypes, and then integrates them to make a complete product. It focuses on delivery working prototype iteratively and incrementally to the customer. It is quite similar compared with prototype model. Figure 3.4 shows the graphical illustration of RAD.

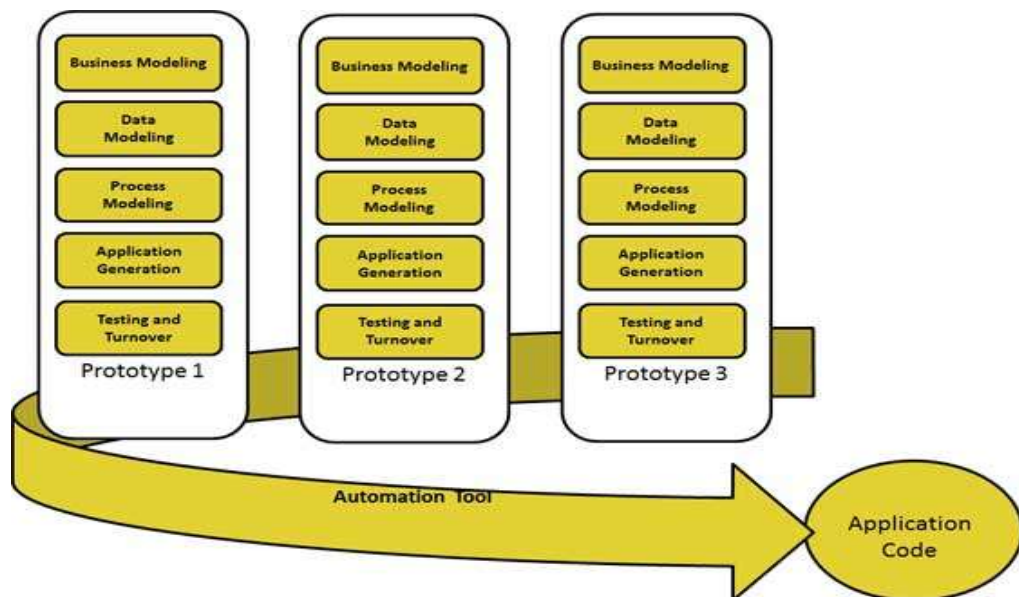


Figure 3.4: Rapid Application Development Methodology

The disadvantages of RAD are:

- a) Fast development affects product quality
- b) System must be able to modularized can be built
- c) Require skilled developer

3.2 Adopted Development Methodology

After studied and analyzed all four development methodologies, Rapid Application Development is the most suitable methodology for this project. The reasons are:

- 1) Duration for development of this project is approximately 7 months. Rapid process for project development duration has to be considered.
- 2) The foundation requirements of the project are almost clearly defined during planning and analysis stage.
- 3) Project can be modularized into few modules.

Figure 3.5 shows the modules covered in this project.

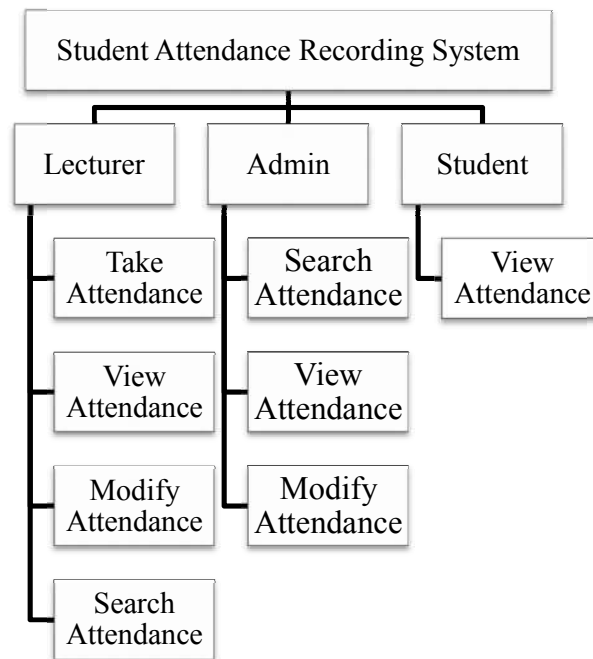


Figure 3.5: Modules of the project

3.3 Development Tools

The development tools used in this project are:

1. Android Studio

Android Studio is the official IDE for Android development. It provides everything needed to start developing an Android project. In this project, Android Studio will be used to create the mobile application.

2. Notepad++

Notepad++ is an open source code editor software. It is an advanced version of notepad. Notepad++ supports many programming languages and syntax. Notepad++ will be used to write and edit code such as PHP, HTML, CSS and JavaScript.

3. WampServer

WampServer is a web development environment for Windows. It provides a set of features to developer which is including Apache web server, MySQL database and PHP server-side scripting. By using WampServer in this project, developer able to:

- Create a localhost server with the Apache
- Manage database with phpMyAdmin

3.4 Project Plan

3.4.1 Project Breakdown

The development of the system will be divided into six main phases:

1. Planning

- Research on the problem statement and objectives
- Define problem statement of the project
- Define objective(s) of the project

- Determine scope of the system
2. Analysis
 - Literature reviewing
 - Analyzing methodologies to be used
 - Writing complete proposal of the project
 3. Design
 - Designing architecture of the system
 - Designing the functions to be implemented
 - Designing database
 - Designing user interface of the application
 - Reviewing design
 - Refining design
 4. Implementation
 - Implementing server-side coding
 - Implementing client-side application coding
 5. Testing
 - Developing test plan
 - Unit Testing
 - System Testing
 - User Acceptance Testing
 - Modify and refining code
 6. Deployment
 - Distributing applications to user

3.4.2 Work Breakdown Structure

Figure 4.1 shows the work breakdown structure of this project.

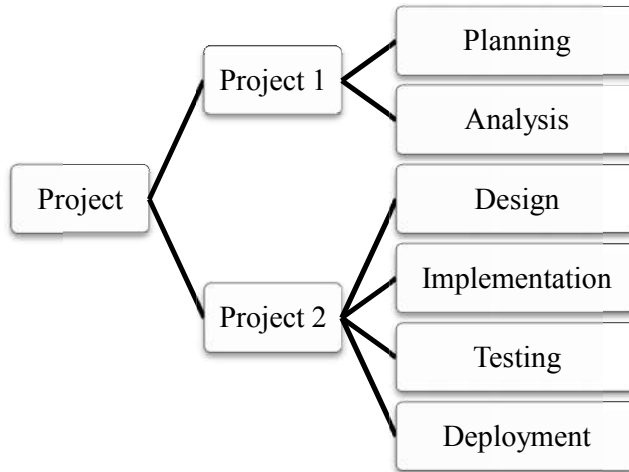


Figure 3.6: Work Breakdown Structure

3.4.3 Gantt Chart

Figure 4.2 shows the Gantt Chart of this project.

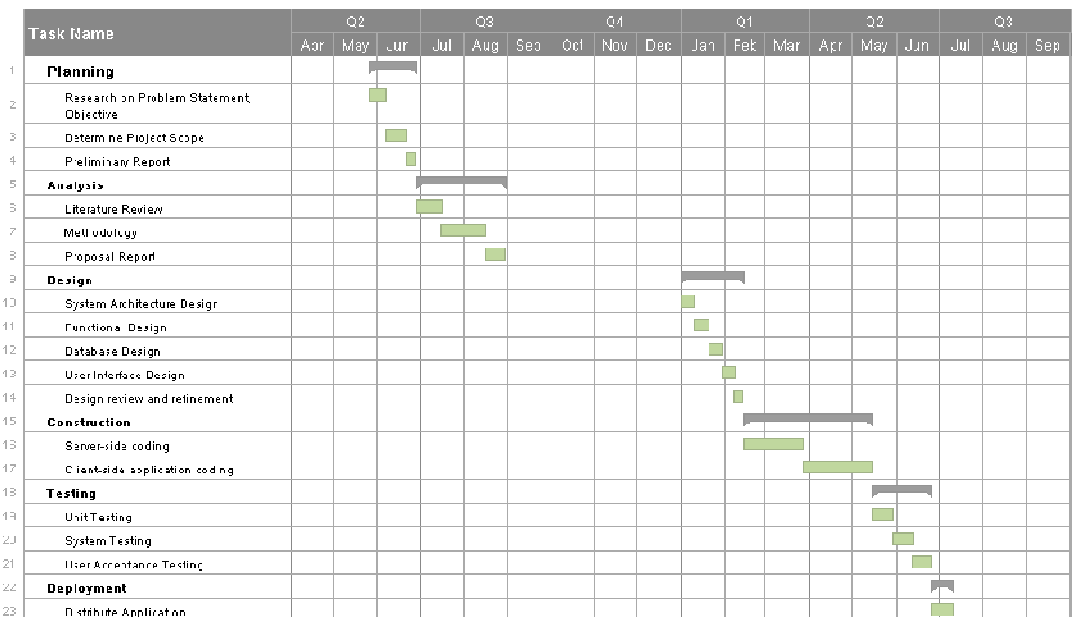


Figure 3.7: Gantt Chart

CHAPTER 4

SYSTEM DESIGN

4.1 System Architecture

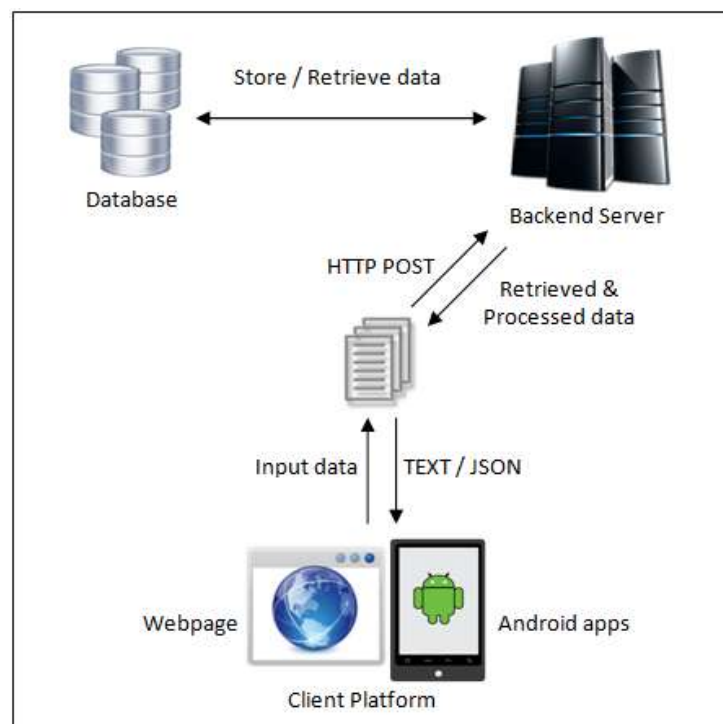


Figure 4.1: System Architecture of the System

As described in Figure 4.1, the android application and webpage is the client of the system. The clients send their input data to the backend server by using HTTP Post method. Once the server received, server will do the processing, including retrieve data, update database record and calculate data, which is based on the input of client.

After finish processing, the result will be sent back to the client platform in format of plain text or JSON.

4.2 Use Case Diagram

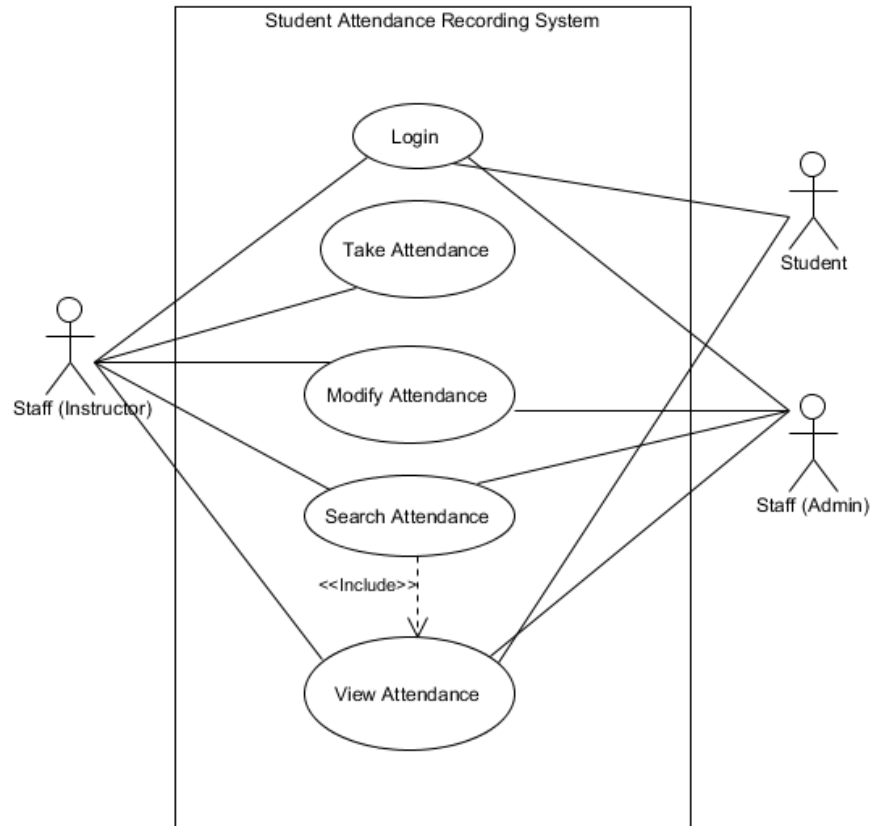


Figure 4.2: Use Case Diagram of the System

Users of the system:

1. Lecturer
2. Admin
3. Student

Actions available of the system:

1. Take attendance
2. Modify attendance
3. View attendance
4. Search attendance

4.3 Database Design

4.3.1 Entity Relationship Diagram (ERD)

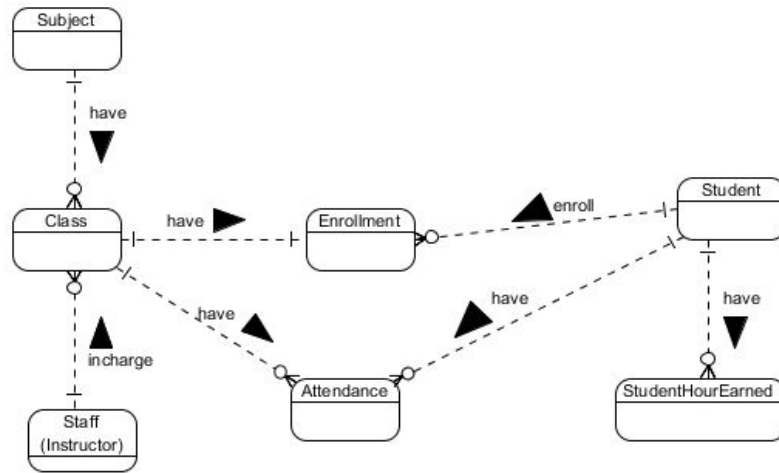


Figure 4.3: ERD of the System

The ERD of the system is involved seven entities (tables) which are Student, Staff, Subject, Class, Attendance, StudentHourEarned and Enrollment.

Table 4.1: Description of Entities

Entities (Table)	Description
Student	Contains information of all the students.
Staff	Contains information of all the staffs.
Subject	Contains information of all the subjects available.
Class	Store class information such as lecturer, time and venue.
Enrollment	Store class enrolment of every student.
Attendance	Contains student attendance record of every classes enrolled.
StudentHourEarned	Store total attended hour of students.

4.3.2 Data Dictionary

Student table

Attribute	Description	Data Type	PK/FK
StudentID	ID of student	Varchar(20)	PK
FirstName	Student's first name	Varchar(255)	-
LastName	Student's last name	Varchar(255)	-
Course	Course of student	varchar(255)	-
LoginPass	Student login password	varchar(255)	-

Staff table

Attribute	Description	Data Type	PK/FK
Student	ID of staff	Varchar(20)	PK
FirstName	Staff's first name	Varchar(255)	-
LastName	Staff's last name	Varchar(255)	-
LoginPass	Staff login password	varchar(255)	-

Subject table

Attribute	Description	Data Type	PK/FK
SubjectID	ID of subject	Varchar(20)	PK
Name	Title of the subject	Varchar(255)	-
TotalHours	Accumulate hours of class that are ended	Float(5,1)	-

Class table

Attribute	Description	Data Type	PK/FK
ClassID	ID of class	Varchar(25)	PK
SubjectID	Subject of the class	Varchar(20)	FK
StaffID	Lecturer of the class	Varchar(20)	FK
Type	Class type	Varchar(1)	-
Venue	Class venue	Varchar(10)	-
Day	Day of the class will be run	Varchar(20)	-
StartTime	The starting time of the class	Time	-
EndTime	The ending time of the class	Time	-
Hours	The period of the class in hour	Float(4,1)	-

Enrollment table

Attribute	Description	Data Type	PK/FK
EnrollmentID	ID of enrolment	Integer(11)	PK
StudentID	Student ID of this enrollment	Varchar(20)	FK
ClassID	Class ID of this enrollment	Varchar(20)	FK

Attendance table

Attribute	Description	Data Type	PK/FK
AttendanceID	ID of attendance	Integer(20)	PK
Date	Date of this attendance	Date	-
ClassID	Class ID of this attendance	Varchar(20)	FK
StudentID	Student ID of this attendance	Varchar(20)	FK
Attendance	Attendance status	Varchar(5)	-
Hours	The period of the class in hours	Float(3,1)	-

StudentHourEarned table

Attribute	Description	Data Type	PK/FK
ID	ID of this table	Integer(11)	PK
StudentID	ID of student of this record	Varchar(20)	FK
SubjectID	ID of subject of this record	Varchar(20)	FK
HoursEarned	The student's total attended hours of the subject	Float(5,1)	-
NcHours	The not counted hours of student in this subject	Float(4,1)	-

4.4 Behavioural Design**4.4.1 Take Attendance**

Lecturer can take attendance of the student by using the application provided by the system. First of all, lecturer selects and starts the class. After that, lecturer needs to

get the student ID to take attendance. To get the student ID, lecturer needs to activate the integrated scanner application and scan the barcode on the student ID card. After capture the ID, application will validate the ID then send it to the server and process it.

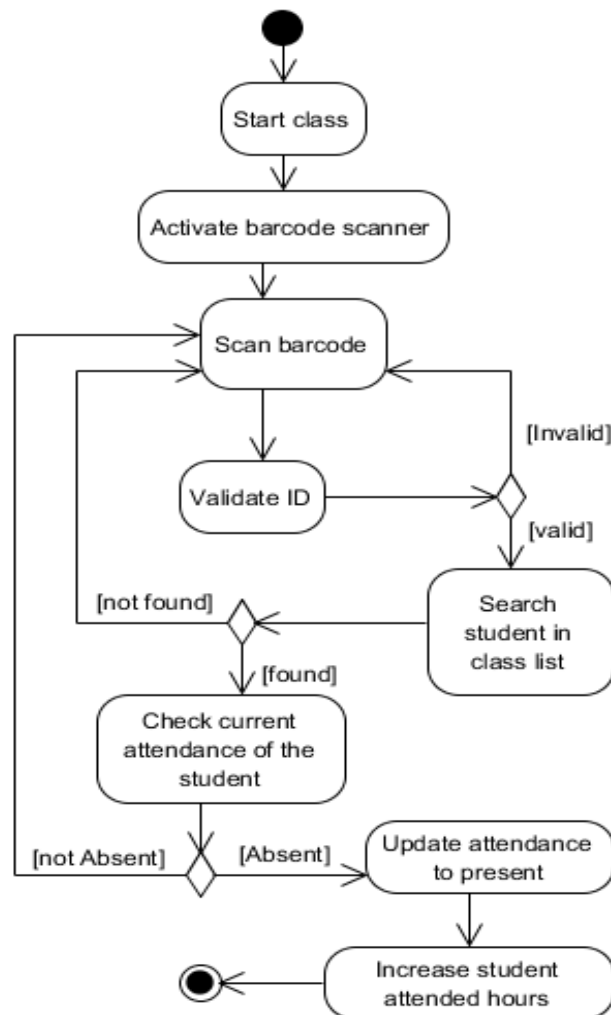


Figure 4.4: Activity Diagram – Take Attendance

Diagram above have shown the activity involved to take student attendance and its process flow. There are including few checking activity to ensure the attendance record will be updated correctly.

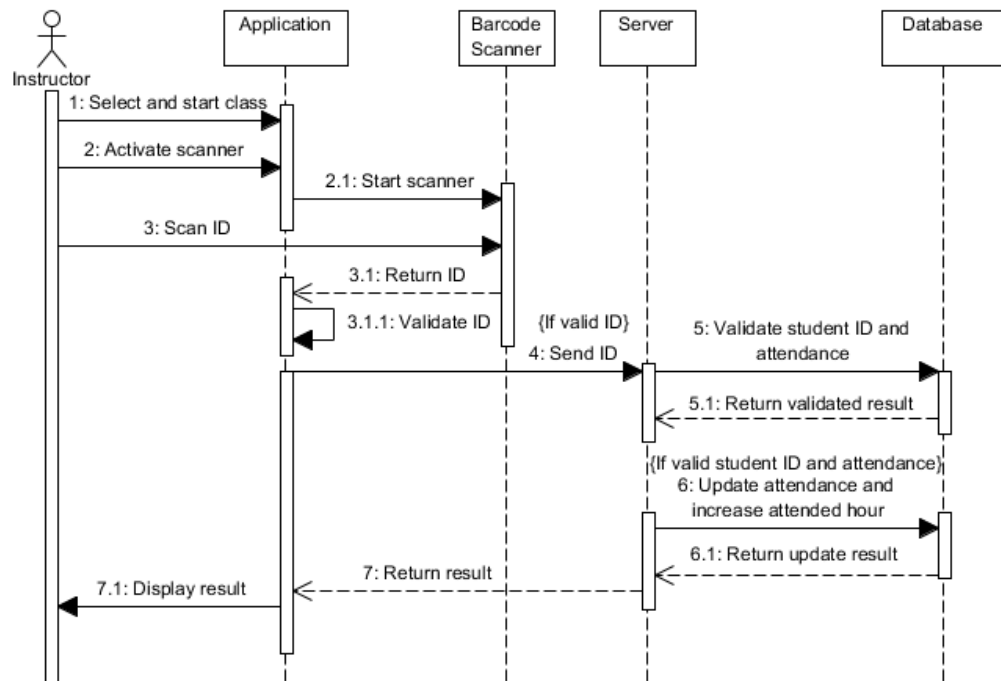


Figure 4.5: Sequence Diagram – Take Attendance

Diagram above has shown the interaction between each components and what the task done by every component to take attendance.

- 1) Lecturer uses the application to start the selected class to take attendance.
- 2) Lecturer activates the barcode scanner application and scans the ID barcode on student ID card to get their ID number.
- 3) Barcode scanner application captures the ID number and send back to the system application.
- 4) Application validates the captured ID number. If the format of ID number is valid, it will be sent to the server.
- 5) Server will search over the database to validate the student ID received from the application. If the student ID number is valid, server will update the attendance record and increase student attended hours.
- 6) Server sends the result back to the application.

4.4.2 Search and View Attendance

Search attendance is a task that every user will involve at. The search result is depends on the input that sent to the server. When user has logged in, or user has

made selection, or user entered some data, server will do searching with the input received from user, and return the result to the user machine.

There are two type of attendance result:

- 1) Attended percentage of student or subject

Required input: Either [SubjectID] OR [StudentID]

- 2) A subject attendance record of a student

Required input: Both [SubjectID] AND [StudentID]

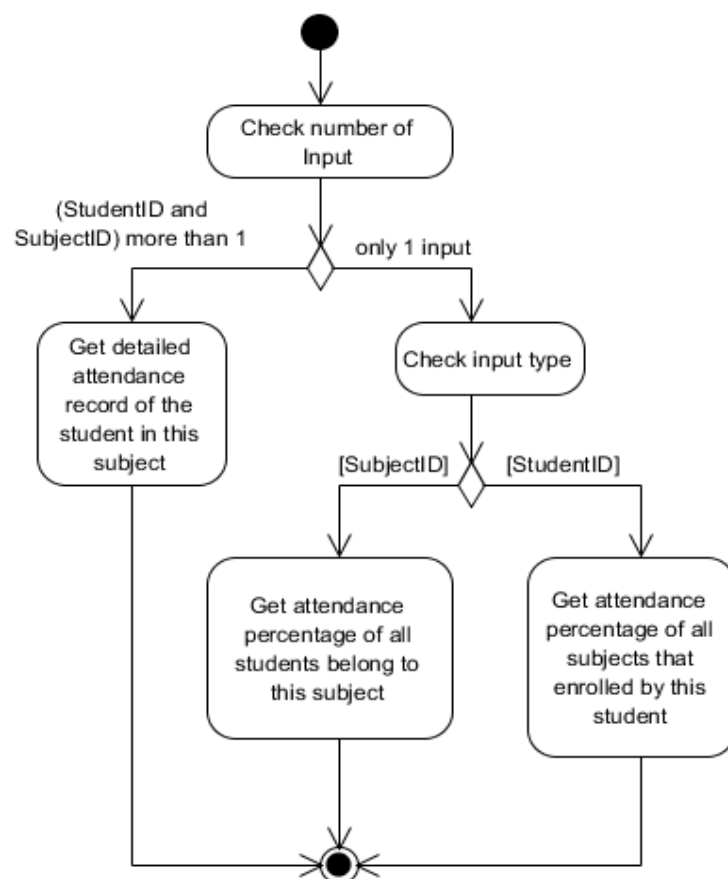


Figure 4.6: Activity Diagram – Search and View Attendance

Diagram above has shown the activity involved to search and view attendance and its process flow. The input is checked first before searching the result.

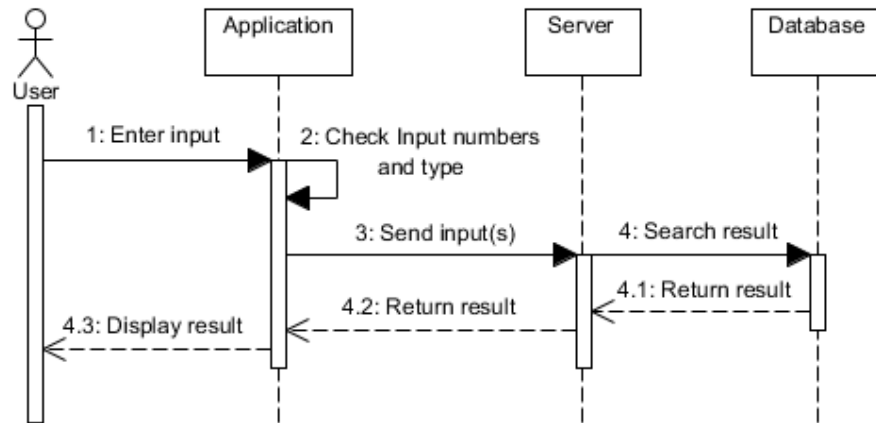


Figure 4.7: Sequence Diagram – Search and View Attendance

Diagram above has shown the interaction between each components and what the task done by every component to search and view attendance result.

- 1) Application gets the input from user.
- 2) Application checks the number of inputs and its type.
- 3) After the checking, the application sends input(s) to the server.
- 4) Server search result based on the received input(s), and then sends the result back to application.

4.4.3 Modify Attendance

Modify attendance is a task for staff (lecturer and admin) to modify or update student attendance. For example, admin have to update the attendance when the student who apply leave. To modify attendance, at first the staff has to complete the task of search and view attendance. If the result is successfully returned and displayed, the staffs are able to modify the attendance.

There are four types of attendance:

- 1) Present – When student is attended the class
- 2) Absent – When student did not attend the class
- 3) Leave – When student missed the class, and his/her application for leave is approved. The attendance is also not

- 4) Not Counted/Not Applicable – When attendance of the class is not counted; or the student is allowed not to attend the class.

To ensure the modified attendance record is correct, there are few checking has to be done. First of all, the original attendance (attendance before modified) will be checked, this is to ensure the attended hours and not counted hours are correctly modified at the same time. After that, the attendance will be updated by the server according to the received data.

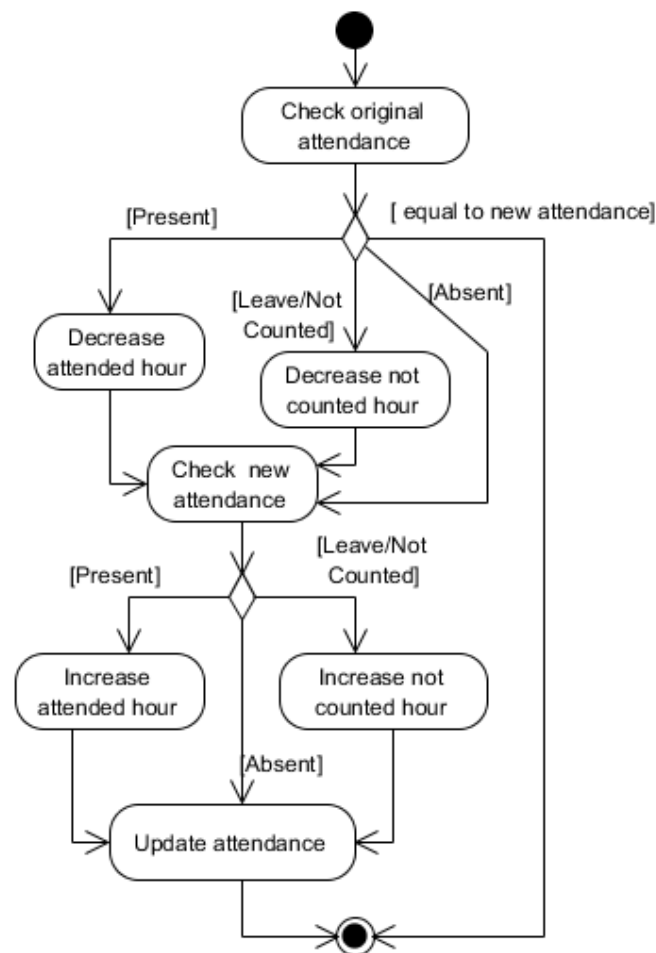


Figure 4.8: Activity Diagram – Modify Attendance

Diagram above has shown the activity involved to modify attendance and its process flow. The original attendance is checked and compared with the new attendance to decide the next activity.

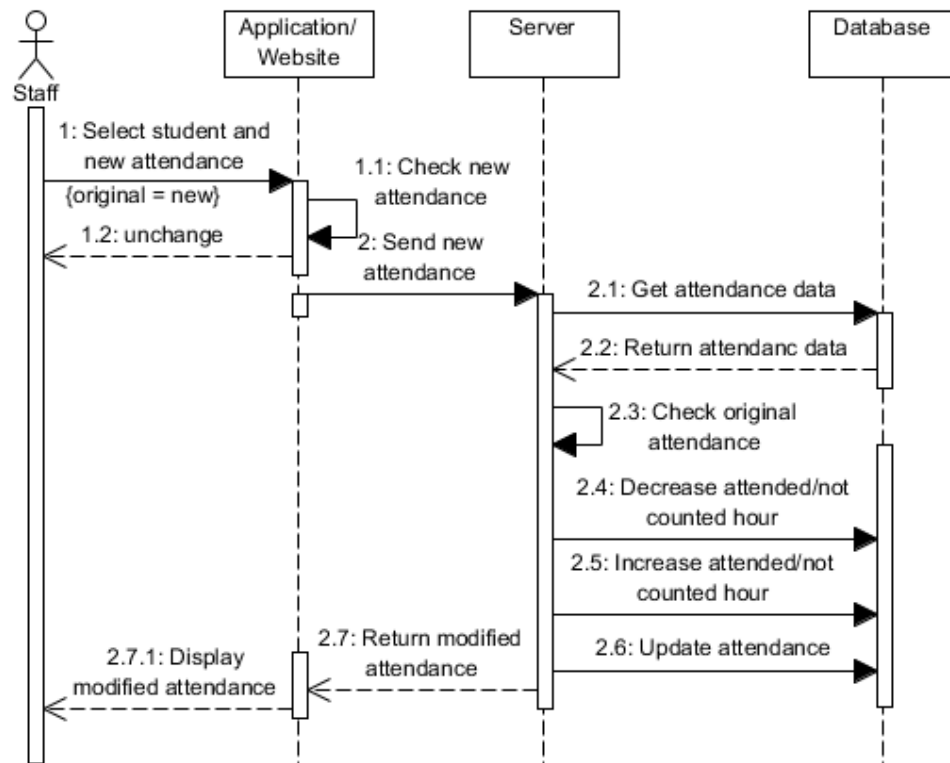


Figure 4.9: Sequence Diagram – Modify Attendance

Diagram above has shown the interaction between each components and what the task done by every component to modify attendance.

- 1) Staff selects student and his/her new attendance on the application or website.
- 2) Application will check the new attendance that selected by staff, if it is equal to original attendance, there will be no changes. Else, the new attendance will be sent to server.
- 3) Server also checks the original attendance. If the original attendance is present, the attended hour will be decreased; if the original attendance is leave or not counted, not counted hour will also be decreased.
- 4) Server will increase the attended hour or not count hour based on the received new attendance.
- 5) Server updates the attendance and return result back to the application.

CHAPTER 5

IMPLEMENTATION AND TESTING

5.1 Implementation

5.1.1 Barcode Scanner

In this system, the mobile application will need to use an integrated barcode scanner application to get input. The barcode scanner is integrated in the mobile application provided to the lecturers. Lecturer is able to start the barcode scanner and capture student ID number as an input from their ID card.

The integrated barcode scanner application is called “Barcode Scanner”, which is created with ZXing team. To integrate the barcode scanner application, the mobile application needs to import the open source ZXing project which is barcode image processing library.

```
dependencies {  
    compile fileTree(dir: 'libs', include: ['*.jar'])  
    compile 'com.android.support:appcompat-v7:21.0.3'  
  
    compile 'com.google.zxing:core:3.1.0'  
    compile 'com.google.zxing:android-integration:3.1.0'  
}
```

Figure 5.1: Section of Code – Import ZXing Project

From Figure 5.1, the highlighted part is the Gradle code to import ZXing project. The mobile application is able to use the API of ZXing to use the barcode scanner application.

```
public void scanNow(){
    IntentIntegrator ii = new IntentIntegrator(this);
    ii.addExtra("SCAN_WIDTH", 640);
    ii.addExtra("SCAN_HEIGHT", 480);
    ii.initiateScan(IntentIntegrator.ONE_D_CODE_TYPES);
}
```

Figure 5.2: Section of Code – Start Barcode Scanner

As shown in Figure 5.2, the section of code is the code for starting the barcode scanner application. The IntentIntegrator class is one of the API of ZXing. The purpose of this section of code is to pass the intents to the barcode scanner application and start it up. The scan mode is set to one dimensional mode because of the barcode on student ID card is also a Code_128 barcode.

```
IntentResult result = IntentIntegrator.parseActivityResult(requestCode, resultCode, intent);
if (result != null && requestCode == IntentIntegrator.REQUEST_CODE) {
    if(resultCode == RESULT_OK){
        scannedFormat = result.getFormatName();
        if(!scannedFormat.equals("CODE_128"))
            Toast.makeText(getBaseContext(), "Invalid code format", Toast.LENGTH_LONG).show();
        else{
            scannedID = result.getContents().substring(0,7);
            if(scannedID.matches("[0-9]+")) {
                Toast.makeText(getBaseContext(), "Student ID: " + scannedID, Toast.LENGTH_SHORT).show();
                new saveStudentAttendance().execute(scannedID);
            }
            else
                Toast.makeText(getBaseContext(), "Invalid student ID", Toast.LENGTH_LONG).show();
        }
    }
    else
        Toast.makeText(getBaseContext(), "Scanner canceled", Toast.LENGTH_LONG).show();
}
else
    Toast.makeText(getBaseContext(), "No scan result", Toast.LENGTH_LONG).show();
```

Figure 5.3: Section of Code – Get Scanner Result

The section of code in Figure 5.3 showed how to get result from scanner application. The IntentResult class is the API of ZXing to retrieve scanner result. There are few checking has to pass to ensure there is a result returned, and the returned result has the same format with student ID number, which is a plain text that only contains numeric number and its length is seven.

5.2 System Testing

System testing is conducted after the system implementation to ensure the system developed has met the requirements and it is error free.

5.2.1 Mobile Application Testing



To test the mobile application, the developed system has to be installed into a mobile device with Android OS.

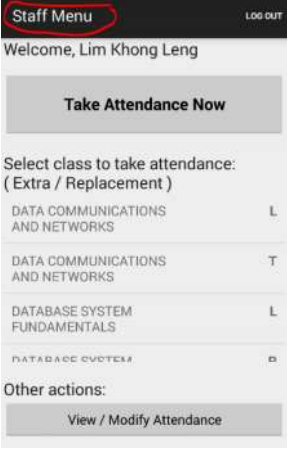
Device name: Samsung Galaxy S3

- Processor: Quad-core 1.4 GHz
- OS: Android 4.3 (Jelly Bean)
- Camera: 8 Megapixel camera

5.2.1.1 Login

Test Case 1: Staff login with correct login ID and password.

<p>Execution Steps</p>	<ol style="list-style-type: none"> 1. Start the mobile application. 2. Enter a staff's login ID and password.  3. Select "Staff" radio button.  4. Click "Login" button.
<p>Expected Result</p>	<ul style="list-style-type: none"> • No error message. • User will be redirect to Staff Menu page.

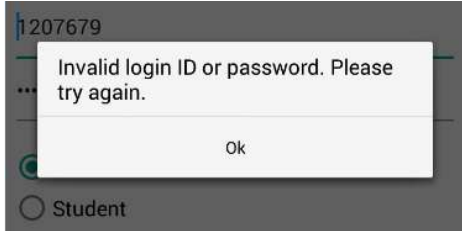
	
Actual Result	Redirected to Staff Menu page without any error message.
Pass/Fail	Pass

Test Case 2: Student login with correct login ID and password.


Execution Steps	<ol style="list-style-type: none"> 1. Start the mobile application. 2. Enter a student's login ID and password. 3. Select "Student" radio button. <div data-bbox="612 1039 954 1196" style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;"> <p>Login ID</p> <p>Password</p> <p><input type="radio"/> Staff</p> <p><input checked="" type="radio"/> Student</p> </div> 4. Click "Login" button. 						
Expected Result	<ul style="list-style-type: none"> • No error message. • User will be redirect to Attendance Percentage page. • The action bar will appear a "LOG OUT" button. <div data-bbox="612 1435 1059 1883" style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;"> <p>Attendance Percentage LOG OUT</p> <p>Select subject to view detail:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tbody> <tr> <td style="padding: 2px;">UECS1003 PROGRAMMING CONCEPTS AND DESIGN</td> <td style="text-align: right; padding: 2px;">50%</td> </tr> <tr> <td style="padding: 2px;">UECS1033 DATA COMMUNICATIONS AND NETWORKS</td> <td style="text-align: right; padding: 2px;">39%</td> </tr> <tr> <td style="padding: 2px;">UECS1203 DATABASE SYSTEM FUNDAMENTALS</td> <td style="text-align: right; padding: 2px;">91%</td> </tr> </tbody> </table> </div>	UECS1003 PROGRAMMING CONCEPTS AND DESIGN	50%	UECS1033 DATA COMMUNICATIONS AND NETWORKS	39%	UECS1203 DATABASE SYSTEM FUNDAMENTALS	91%
UECS1003 PROGRAMMING CONCEPTS AND DESIGN	50%						
UECS1033 DATA COMMUNICATIONS AND NETWORKS	39%						
UECS1203 DATABASE SYSTEM FUNDAMENTALS	91%						

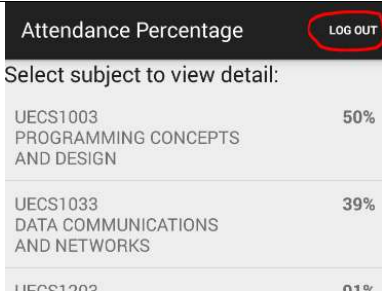
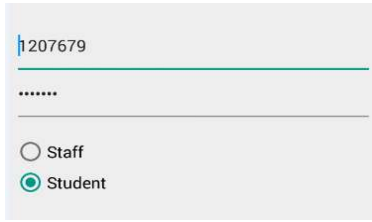
Actual Result	<ul style="list-style-type: none"> • Redirected to Attendance Percentage page without any error message. • The action bar has a “LOG OUT” button.
Pass/Fail	Pass

Test Case 3: Login with incorrect login ID and password.

Execution Steps	<ol style="list-style-type: none"> 1. Start the mobile application. 2. Enter an incorrect login ID and password. 3. Select “Staff” or “Student” radio button. 4. Click “Login” button.
Expected Result	<p>Error message will be displayed.</p> 
Actual Result	An error message is displayed.
Pass/Fail	Pass

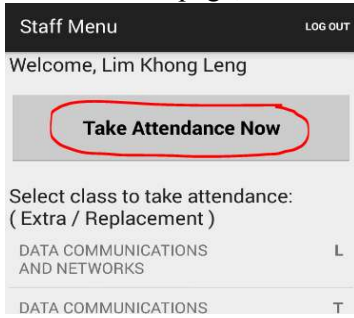
Test Case 4: Remember me checkbox feature.

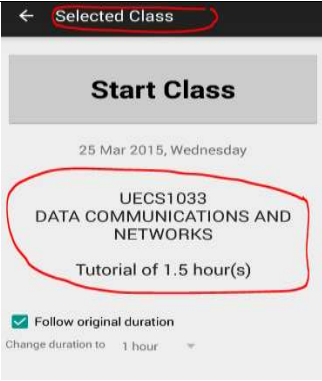
Execution Steps	<ol style="list-style-type: none"> 1. Start the mobile application. 2. Enter a student’s login ID and password correctly. 3. Select “Student” radio button. 4. Tick the “Remember Me” checkbox.  <ol style="list-style-type: none"> 5. Click “Login” button. 6. After login, click “LOG OUT” button.
------------------------	---

	 <p>Attendance Percentage</p> <p>Select subject to view detail:</p> <table border="1"> <tr> <td>UECS1003 PROGRAMMING CONCEPTS AND DESIGN</td> <td>50%</td> </tr> <tr> <td>UECS1033 DATA COMMUNICATIONS AND NETWORKS</td> <td>39%</td> </tr> <tr> <td>UECS1003</td> <td>01%</td> </tr> </table>	UECS1003 PROGRAMMING CONCEPTS AND DESIGN	50%	UECS1033 DATA COMMUNICATIONS AND NETWORKS	39%	UECS1003	01%
UECS1003 PROGRAMMING CONCEPTS AND DESIGN	50%						
UECS1033 DATA COMMUNICATIONS AND NETWORKS	39%						
UECS1003	01%						
Expected Result	<ul style="list-style-type: none"> • User will be redirected back to Login page • The text fields of login ID and password will be filled in with last logged in user detail • The radio button of last logged in user type will be auto selected. 						
Actual Result	The text fields are filled in with last logged in user detail and the radio button is auto selected.						
Pass/Fail	Pass						


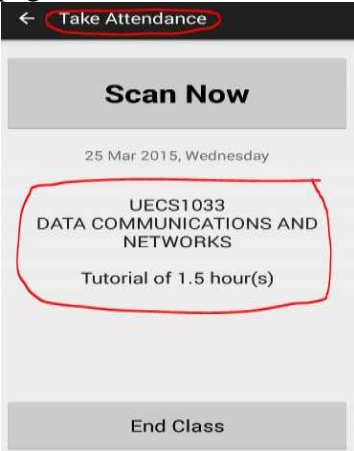
5.2.1.2 Take Attendance

Test Case 1: “Take Attendance Now” button

Execution Steps	<ol style="list-style-type: none"> 1. Start application and login with staff account. 2. In Staff Menu page, click “Take Attendance Now” button. 
Expected Result	<ul style="list-style-type: none"> • User will be redirected to Start Class page. • Start Class page will display the detail of ongoing class.

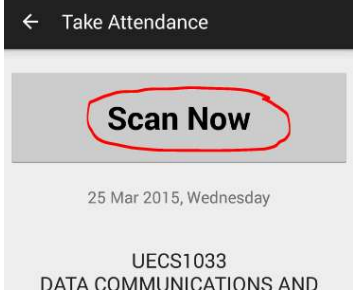

	
<p>Actual Result</p>	<p>User is redirected to Start Class page, and the page is displaying the correct ongoing class detail.</p>
<p>Pass/Fail</p>	<p>Pass</p>

Test Case 2: Start selected class.

<p>Execution Steps</p>	<ol style="list-style-type: none"> 1. Redo all the steps in test case 1. 2. In Selected Class page, click the “Start Class” button. 
<p>Expected Result</p>	<ul style="list-style-type: none"> • User will be redirected to Take Attendance page. • Class detail in Start Class page is same with Take Attendance page. 

Actual Result	User is redirected to Take Attendance page, and the displayed class detail is same with Selected Class page.
Pass/Fail	Pass

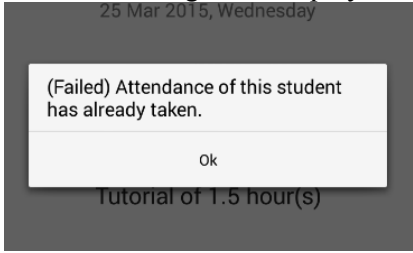
Test Case 3: Start barcode scanner application.

Execution Steps	<ol style="list-style-type: none"> 1. Redo all the steps in test case 1 and 2. 2. In Take Attendance page, click the “Scan Now” button. 
Expected Result	Barcode scanner application will be started. 
Actual Result	Barcode scanner application is started.
Pass/Fail	Pass

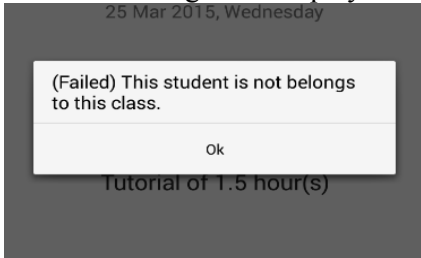
Test Case 4: Scan student ID that is belong to the started class.

Execution Steps	<ol style="list-style-type: none"> 1. Redo all the steps in test case 1, 2 and 3. 2. In barcode scanner application, scan the barcode on student ID card.
Expected Result	<ul style="list-style-type: none"> • User will be redirected back to Take Attendance page. • A toast notification with student name and ID number will display on the same page.
Actual Result	User is redirected back to Take Attendance page, and a toast notification with student name and ID number is displayed.
Pass/Fail	Pass

Test Case 5: Take the attendance again that is already taken.

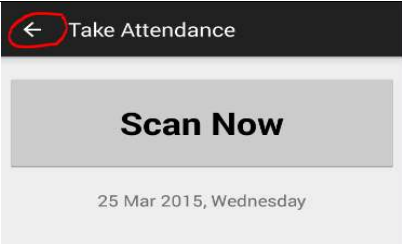
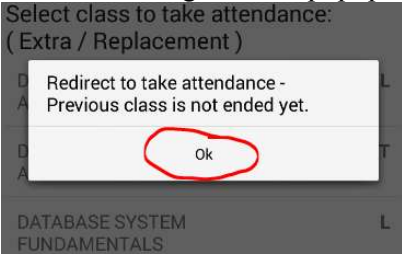
Execution Steps	<ol style="list-style-type: none"> 1. Redo all the steps in test case 1, 2 and 3. 2. In barcode scanner application, scan the ID that is already scanned.
Expected Result	<ul style="list-style-type: none"> • User will be redirected back to Take Attendance page. • An error message will display on the same page. 
Actual Result	User is redirected back to Take Attendance page, and a toast notification with success message is displayed.
Pass/Fail	Pass

Test Case 6: Scan student ID that is **NOT** belong to the started class.


Execution Steps	<ol style="list-style-type: none"> 1. Redo all the steps in test case 1, 2 and 3. 2. In barcode scanner application, scan the barcode on student ID card.
Expected Result	<ul style="list-style-type: none"> • User will be redirected back to Take Attendance page. • An error message will display on the same page. 
Actual Result	User is redirected back to Take Attendance page, and a toast notification with success message is displayed.
Pass/Fail	Pass

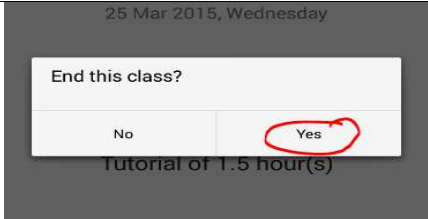
Test Case 7: Exit Take Attendance page without end the class.

Execution Steps	<ol style="list-style-type: none"> 1. Redo all the steps in test case 1 and 2. 2. Click back button.
------------------------	--

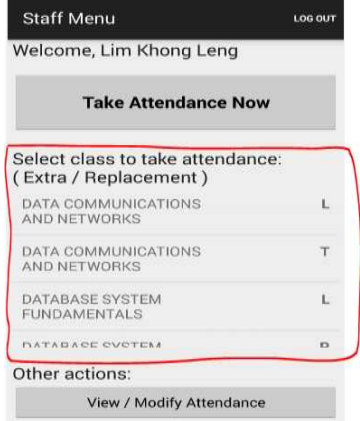
	
Expected Result	<ul style="list-style-type: none"> An error message will be popup.  After clicking “OK” button, user will be redirected back to Take Attendance page.
Actual Result	An error message is displayed. After clicked “OK”, user is redirected back to Take Attendance page.
Pass/Fail	Pass

Test Case 8: End class.

Execution Steps	<ol style="list-style-type: none"> Redo all the steps in test case 1 and 2. Click “End Class” button.  In the confirmation dialog, click “Yes”.
------------------------	---

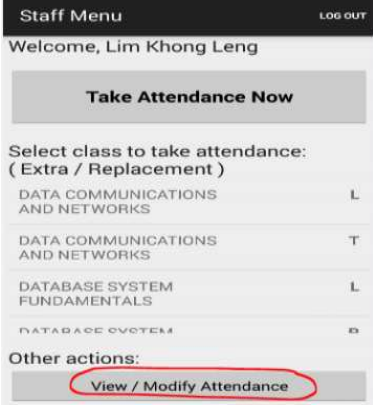
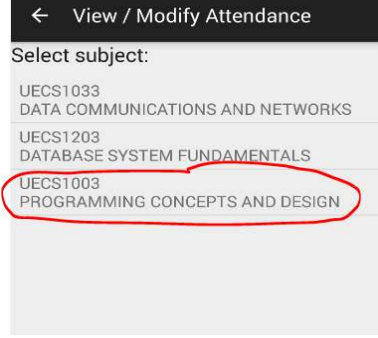
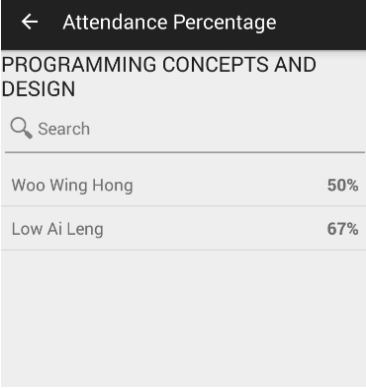
	
Expected Result	<ul style="list-style-type: none"> • If “Yes” button is clicked, user will be redirected to Staff Menu page. • If “No” button is clicked, user will remain in Take Attendance Page.
Actual Result	User is redirected back to Staff Menu page when clicked “Yes”. User is remaining in Take Attendance Page.
Pass/Fail	Pass

Test Case 8: Select class from class list in Staff Menu.

Execution Steps	<ol style="list-style-type: none"> 1. Start application and login with staff account. 2. In Staff Menu page, select any class from the class list. 
Expected Result	<ul style="list-style-type: none"> • User will be redirected to Selected Class page. • The selected class detail is displayed correctly in Selected Class page.
Actual Result	User is redirected to Selected Class page, and the selected class detail is correctly displayed in the page.
Pass/Fail	Pass

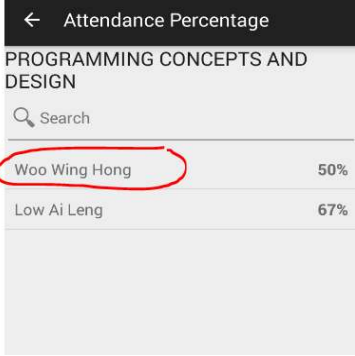
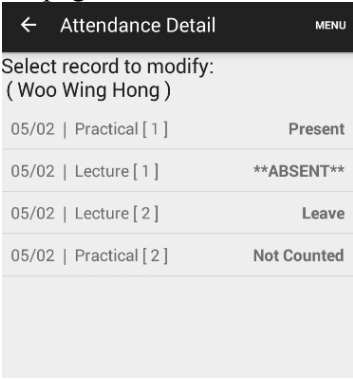
5.2.1.3 Search and View Attendance

Test Case 1: Lecturer view student attendance percentage.

<p>Execution Steps</p>	<ol style="list-style-type: none"> 1. Start application and login with staff account. 2. In Staff Menu page, click “View / Modify Attendance” button.  3. Select a subject from the list. 
<p>Expected Result</p>	<ul style="list-style-type: none"> • User will be redirected to Attendance Percentage page. • Selected subject name will display on top of the page. • A list of student attendance percentage of the subject is on the page. 
<p>Actual Result</p>	<ul style="list-style-type: none"> • User is redirected to Attendance Percentage page. • Selected subject name is correct. • The list of student attendance percentage is correctly listed out.

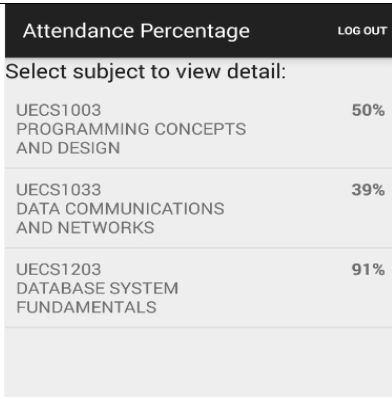
Pass/Fail	Pass
------------------	------

Test Case 2: Lecturer view student attendance in detail.

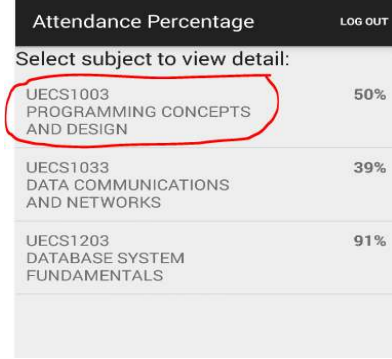
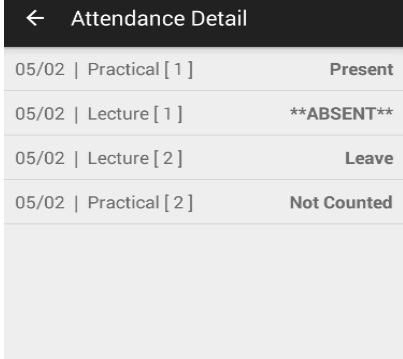
Execution Steps	<ol style="list-style-type: none"> Redo all the steps in test case 1. Select a student from the list. 
Expected Result	<ul style="list-style-type: none"> User will be redirected to Attendance Detail page. Selected student name will display on top of the page. A list of student attendance record of the subject will display on the page. 
Actual Result	<ul style="list-style-type: none"> User is redirected to Attendance Detail page. Selected student name is correct. The list of student attendance detail is correctly listed out.
Pass/Fail	Pass

Test Case 3: Students view their attendance percentage.

Execution Steps	<ol style="list-style-type: none"> Start application and login with student account.
Expected Result	<ul style="list-style-type: none"> User will be redirected to Attendance Percentage page. A list of attended percentage of enrolled subject will display on the page.

	
Actual Result	<ul style="list-style-type: none"> • User is redirected to Attendance Percentage page. • A list of attended percentage of enrolled subject is correctly display on the page.
Pass/Fail	Pass

Test Case 4: Students view their attendance in detail.

Execution Steps	<ol style="list-style-type: none"> 1. Start application and login with student account. 2. Select a subject from the list. 
Expected Result	<ul style="list-style-type: none"> • User will be redirected to Attendance Detail page. • A list of attendance record of selected subject will display on the page. 

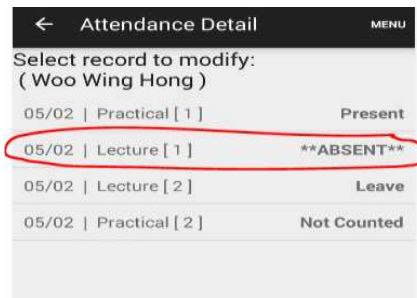
Actual Result	<ul style="list-style-type: none"> • User is redirected to Attendance Detail page. • A list of attendance record of selected subject is displayed correctly on the page.
Pass/Fail	Pass

5.2.1.4 Modify Attendance

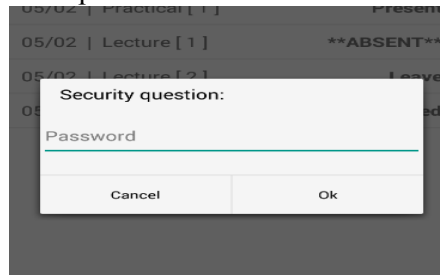
Test Case 1: Lecturer modifies attendance.

Execution Steps	<ol style="list-style-type: none"> 1. Start application and login with staff account. 2. In Staff Menu page, click “View / Modify Attendance” button. <div data-bbox="609 698 957 1120" data-label="Image"> <p>Staff Menu LOG OUT</p> <p>Welcome, Lim Khong Leng</p> <p>Take Attendance Now</p> <p>Select class to take attendance: (Extra / Replacement)</p> <table border="1"> <tr> <td>DATA COMMUNICATIONS AND NETWORKS</td> <td>L</td> </tr> <tr> <td>DATA COMMUNICATIONS AND NETWORKS</td> <td>T</td> </tr> <tr> <td>DATABASE SYSTEM FUNDAMENTALS</td> <td>L</td> </tr> <tr> <td>DATABASE SYSTEM FUNDAMENTALS</td> <td>D</td> </tr> </table> <p>Other actions:</p> <p>View / Modify Attendance</p> </div> 3. Select a subject from the list. <div data-bbox="609 1187 1021 1500" data-label="Image"> <p>← View / Modify Attendance</p> <p>Select subject:</p> <table border="1"> <tr> <td>UECS1033 DATA COMMUNICATIONS AND NETWORKS</td> </tr> <tr> <td>UECS1203 DATABASE SYSTEM FUNDAMENTALS</td> </tr> <tr> <td>UECS1003 PROGRAMMING CONCEPTS AND DESIGN</td> </tr> </table> </div> 4. Select a student from the list. <div data-bbox="609 1568 1021 1897" data-label="Image"> <p>← Attendance Percentage</p> <p>PROGRAMMING CONCEPTS AND DESIGN</p> <p>Search</p> <table border="1"> <tr> <td>Woo Wing Hong</td> <td>50%</td> </tr> <tr> <td>Low Ai Leng</td> <td>67%</td> </tr> </table> </div> 	DATA COMMUNICATIONS AND NETWORKS	L	DATA COMMUNICATIONS AND NETWORKS	T	DATABASE SYSTEM FUNDAMENTALS	L	DATABASE SYSTEM FUNDAMENTALS	D	UECS1033 DATA COMMUNICATIONS AND NETWORKS	UECS1203 DATABASE SYSTEM FUNDAMENTALS	UECS1003 PROGRAMMING CONCEPTS AND DESIGN	Woo Wing Hong	50%	Low Ai Leng	67%
DATA COMMUNICATIONS AND NETWORKS	L															
DATA COMMUNICATIONS AND NETWORKS	T															
DATABASE SYSTEM FUNDAMENTALS	L															
DATABASE SYSTEM FUNDAMENTALS	D															
UECS1033 DATA COMMUNICATIONS AND NETWORKS																
UECS1203 DATABASE SYSTEM FUNDAMENTALS																
UECS1003 PROGRAMMING CONCEPTS AND DESIGN																
Woo Wing Hong	50%															
Low Ai Leng	67%															

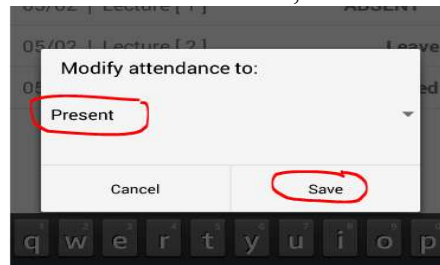
5. Select a record from the list.



6. Enter password.

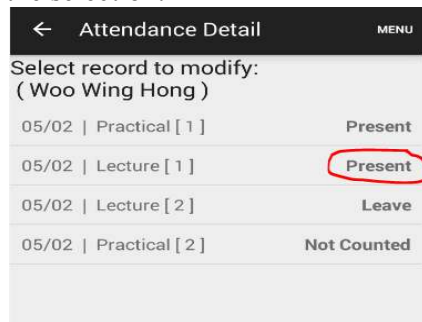


7. Select attendance status, and click "Save" button.

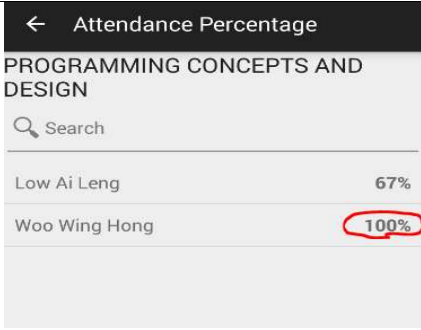


Expected Result

- Attendance Detail page will auto refresh with new record.
- The attendance status of the record will be modified based on the selection.



- Go to Attendance Percentage page by clicking back button, the page will also auto refreshed to correct percentage.

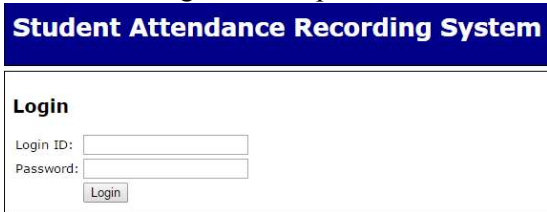

	
Actual Result	<ul style="list-style-type: none"> • Attendance Detail page is refreshed with new record correctly. • Attendance Percentage page is refreshed with new percentage correctly.
Pass/Fail	Pass

5.2.2 Website Testing

Browser name: Google Chrome (Version 41.0.2272.101 m)


5.2.2.1 Login

Test Case 1: Login with staff account.

Execution Steps	<ol style="list-style-type: none"> 1. Go to login page. 2. Enter staff's login ID and password. 
Expected Result	<p>User will be redirected to homepage.</p> 
Actual Result	User is redirected to homepage.



Pass/Fail	Pass
------------------	------

Test Case 2: Login with student account.

Execution Steps	<ol style="list-style-type: none"> Go to login page. Enter existing student's login ID and password.
Expected Result	<p>Error message will be displayed in the same page.</p> 
Actual Result	Error message is displayed on the same page.
Pass/Fail	Pass

5.2.3 Search and View Attendance

Test Case 1: Search attendance ONLY by student ID

Execution Steps	<ol style="list-style-type: none"> Go to login page and login with staff account. Enter student ID and click "Search" button. 										
Expected Result	<ul style="list-style-type: none"> Attendance of all enrolled subject of the student will be displayed in the same page.  <table border="1"> <thead> <tr> <th colspan="2">Student Woo Wing Hong (1207679)</th> </tr> <tr> <th>Subject</th> <th>Percentage(%)</th> </tr> </thead> <tbody> <tr> <td>1 (UECS1003) PROGRAMMING CONCEPTS AND DESIGN</td> <td>100 View</td> </tr> <tr> <td>2 (UECS1033) DATA COMMUNICATIONS AND NETWORKS</td> <td>39 View</td> </tr> <tr> <td>3 (UECS1203) DATABASE SYSTEM FUNDAMENTALS</td> <td>91 View</td> </tr> </tbody> </table>	Student Woo Wing Hong (1207679)		Subject	Percentage(%)	1 (UECS1003) PROGRAMMING CONCEPTS AND DESIGN	100 View	2 (UECS1033) DATA COMMUNICATIONS AND NETWORKS	39 View	3 (UECS1203) DATABASE SYSTEM FUNDAMENTALS	91 View
Student Woo Wing Hong (1207679)											
Subject	Percentage(%)										
1 (UECS1003) PROGRAMMING CONCEPTS AND DESIGN	100 View										
2 (UECS1033) DATA COMMUNICATIONS AND NETWORKS	39 View										
3 (UECS1203) DATABASE SYSTEM FUNDAMENTALS	91 View										

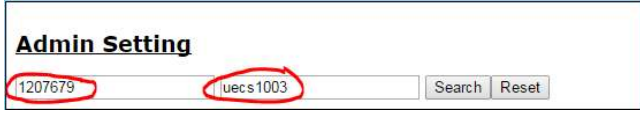
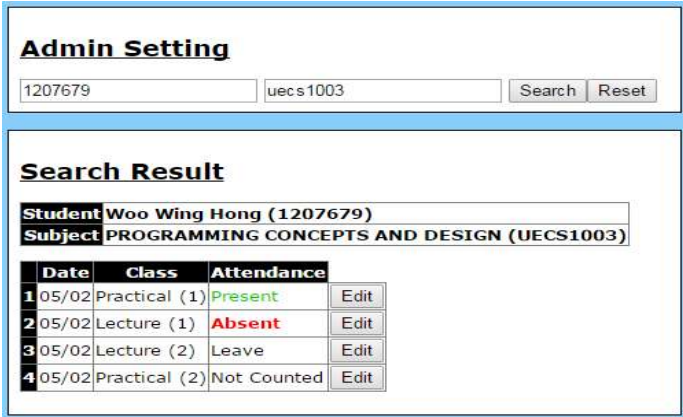
	<ul style="list-style-type: none"> If “View” button is clicked, the content of Search Result will be changed and subject attendance detail record of the student will be displayed. <div style="border: 1px solid black; padding: 5px;"> <p>Search Result</p> <p>Student Woo Wing Hong (1207679)</p> <p>Subject PROGRAMMING CONCEPTS AND DESIGN (UECS1003)</p> <table border="1"> <thead> <tr> <th>Date</th> <th>Class</th> <th>Attendance</th> <th></th> </tr> </thead> <tbody> <tr> <td>1 05/02</td> <td>Practical (1)</td> <td>Present</td> <td>Edit</td> </tr> <tr> <td>2 05/02</td> <td>Lecture (1)</td> <td>Absent</td> <td>Edit</td> </tr> <tr> <td>3 05/02</td> <td>Lecture (2)</td> <td>Leave</td> <td>Edit</td> </tr> <tr> <td>4 05/02</td> <td>Practical (2)</td> <td>Not Counted</td> <td>Edit</td> </tr> </tbody> </table> </div>	Date	Class	Attendance		1 05/02	Practical (1)	Present	Edit	2 05/02	Lecture (1)	Absent	Edit	3 05/02	Lecture (2)	Leave	Edit	4 05/02	Practical (2)	Not Counted	Edit
Date	Class	Attendance																			
1 05/02	Practical (1)	Present	Edit																		
2 05/02	Lecture (1)	Absent	Edit																		
3 05/02	Lecture (2)	Leave	Edit																		
4 05/02	Practical (2)	Not Counted	Edit																		
Actual Result	<ul style="list-style-type: none"> Attendance of all enrolled subject of the student is displayed correctly in the same page. When clicked the “View button”, the content of Search Result is changed and subject attendance detail record of the student is displayed correctly. 																				
Pass/Fail	Pass																				

Test Case 2: Search attendance ONLY by subject ID

Execution Steps	<ol style="list-style-type: none"> Go to login page and login with staff account. Enter subject ID and click “Search” button. <div style="border: 1px solid black; padding: 5px;"> <p>Admin Setting</p> <p>Student ID... <input type="text" value="uecs1003"/> Search Reset</p> </div>									
Expected Result	<ul style="list-style-type: none"> The subject attendance of all students will be displayed in the same page. <div style="border: 1px solid black; padding: 5px;"> <p>Admin Setting</p> <p>Student ID... <input type="text" value="uecs1003"/> Search Reset</p> <p>Search Result</p> <p>Subject PROGRAMMING CONCEPTS AND DESIGN (UECS1003)</p> <table border="1"> <thead> <tr> <th>Student</th> <th>Percentage(%)</th> <th></th> </tr> </thead> <tbody> <tr> <td>1 (1200001) Low Ai Leng</td> <td>67</td> <td>View</td> </tr> <tr> <td>2 (1207679) Woo Wing Hong</td> <td>100</td> <td>View</td> </tr> </tbody> </table> </div> <ul style="list-style-type: none"> If “View” button is clicked, the content of Search Result will be changed and student attendance detail record of the subject will be displayed. 	Student	Percentage(%)		1 (1200001) Low Ai Leng	67	View	2 (1207679) Woo Wing Hong	100	View
Student	Percentage(%)									
1 (1200001) Low Ai Leng	67	View								
2 (1207679) Woo Wing Hong	100	View								

	<p>Search Result</p> <p>Student Woo Wing Hong (1207679) Subject PROGRAMMING CONCEPTS AND DESIGN (UECS1003)</p> <table border="1"> <thead> <tr> <th>Date</th> <th>Class</th> <th>Attendance</th> <th></th> </tr> </thead> <tbody> <tr> <td>1 05/02</td> <td>Practical (1)</td> <td>Present</td> <td>Edit</td> </tr> <tr> <td>2 05/02</td> <td>Lecture (1)</td> <td>Absent</td> <td>Edit</td> </tr> <tr> <td>3 05/02</td> <td>Lecture (2)</td> <td>Leave</td> <td>Edit</td> </tr> <tr> <td>4 05/02</td> <td>Practical (2)</td> <td>Not Counted</td> <td>Edit</td> </tr> </tbody> </table>	Date	Class	Attendance		1 05/02	Practical (1)	Present	Edit	2 05/02	Lecture (1)	Absent	Edit	3 05/02	Lecture (2)	Leave	Edit	4 05/02	Practical (2)	Not Counted	Edit
Date	Class	Attendance																			
1 05/02	Practical (1)	Present	Edit																		
2 05/02	Lecture (1)	Absent	Edit																		
3 05/02	Lecture (2)	Leave	Edit																		
4 05/02	Practical (2)	Not Counted	Edit																		
Actual Result	<ul style="list-style-type: none"> The subject attendance of all students is displayed correctly in the same page. When clicked the “View button”, the content of Search Result is changed and student attendance detail record of the subject is displayed correctly. 																				
Pass/Fail	Pass																				

Test Case 3: Search attendance by student ID and subject ID

Execution Steps	<ol style="list-style-type: none"> Go to login page and login with staff account. Enter student ID and subject ID and click “Search” button. 
Expected Result	<p>The subject attendance detail record of the student will be displayed in the same page.</p> 
Actual Result	The subject attendance detail record of the student is displayed in the same page.
Pass/Fail	Pass

5.2.3.1 Modify Attendance

Test Case 1: Modify attendance

<p>Execution Steps</p>	<ol style="list-style-type: none"> Go to login page and login with staff account. Enter student ID and subject ID and click “Search” button. <div data-bbox="611 405 1374 533" style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <p>Admin Setting</p> <p>1207679 uecs1003 Search Reset</p> </div> After search result appeared, click “Edit” button. <div data-bbox="611 600 1318 943" style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <p>Search Result</p> <p>Student Woo Wing Hong (1207679) Subject PROGRAMMING CONCEPTS AND DESIGN (UECS1003)</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th></th> <th>Date</th> <th>Class</th> <th>Attendance</th> <th></th> </tr> </thead> <tbody> <tr> <td>1</td> <td>05/02</td> <td>Practical (1)</td> <td>Present</td> <td>Edit</td> </tr> <tr> <td>2</td> <td>05/02</td> <td>Lecture (1)</td> <td>Absent</td> <td>Edit</td> </tr> <tr> <td>3</td> <td>05/02</td> <td>Lecture (2)</td> <td>Leave</td> <td>Edit</td> </tr> <tr> <td>4</td> <td>05/02</td> <td>Practical (2)</td> <td>Not Counted</td> <td>Edit</td> </tr> </tbody> </table> </div> After “Edit Attendance” section appeared, choose attendance status from the drop down list and click “Edit” button. <div data-bbox="611 1048 1031 1346" style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <p>Edit Attendance</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>Date</td> <td>05/02/2015</td> </tr> <tr> <td>Class</td> <td>Lecture (1)</td> </tr> <tr> <td>Attendance</td> <td>Present ▼</td> </tr> </table> <p>Edit Close</p> </div> In the confirmation box, click “Ok” button. <div data-bbox="611 1406 1171 1697" style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <p>Confirm to edit attendance?</p> <p>1207679</p> <p>OK Cancel</p> <hr/> <p>Edit Attendance</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>Date</td> <td>05/02/2015</td> </tr> <tr> <td>Class</td> <td>Lecture (1)</td> </tr> <tr> <td>Attendance</td> <td>Present ▼</td> </tr> </table> <p>Edit Close</p> </div> 		Date	Class	Attendance		1	05/02	Practical (1)	Present	Edit	2	05/02	Lecture (1)	Absent	Edit	3	05/02	Lecture (2)	Leave	Edit	4	05/02	Practical (2)	Not Counted	Edit	Date	05/02/2015	Class	Lecture (1)	Attendance	Present ▼	Date	05/02/2015	Class	Lecture (1)	Attendance	Present ▼
	Date	Class	Attendance																																			
1	05/02	Practical (1)	Present	Edit																																		
2	05/02	Lecture (1)	Absent	Edit																																		
3	05/02	Lecture (2)	Leave	Edit																																		
4	05/02	Practical (2)	Not Counted	Edit																																		
Date	05/02/2015																																					
Class	Lecture (1)																																					
Attendance	Present ▼																																					
Date	05/02/2015																																					
Class	Lecture (1)																																					
Attendance	Present ▼																																					
<p>Expected Result</p>	<ul style="list-style-type: none"> The content of table of student attendance detail record will be changed automatically and correctly. A success message will appear on top of the table. 																																					

	<p>Search Result</p> <p>Student Woo Wing Hong (1207679) Subject PROGRAMMING CONCEPTS AND DESIGN (UECS1003)</p> <p>Attendance edited successfully.</p> <table border="1"> <thead> <tr> <th>Date</th> <th>Class</th> <th>Attendance</th> <th></th> </tr> </thead> <tbody> <tr> <td>1 05/02</td> <td>Practical (1)</td> <td>Present</td> <td>Edit</td> </tr> <tr> <td>2 05/02</td> <td>Lecture (1)</td> <td>Present</td> <td>Edit</td> </tr> <tr> <td>3 05/02</td> <td>Lecture (2)</td> <td>Leave</td> <td>Edit</td> </tr> <tr> <td>4 05/02</td> <td>Practical (2)</td> <td>Not Counted</td> <td>Edit</td> </tr> </tbody> </table>	Date	Class	Attendance		1 05/02	Practical (1)	Present	Edit	2 05/02	Lecture (1)	Present	Edit	3 05/02	Lecture (2)	Leave	Edit	4 05/02	Practical (2)	Not Counted	Edit
Date	Class	Attendance																			
1 05/02	Practical (1)	Present	Edit																		
2 05/02	Lecture (1)	Present	Edit																		
3 05/02	Lecture (2)	Leave	Edit																		
4 05/02	Practical (2)	Not Counted	Edit																		
Actual Result	<ul style="list-style-type: none"> The content of table of student attendance detail record is changed automatically and correctly. A success message is appeared on top of the table. 																				
Pass/Fail	Pass																				

CHAPTER 6

CONCLUSION AND RECOMMENDATION

6.1 Limitation and Future Improvement

The limitation of the system developed is the speed of getting student ID from barcode scanner application is slow. This is because the barcode scanner application needs to complete few processes which are included capturing image, image processing and analysis for the captured image. In order to finish these three processes may takes some time.

In addition, there are some future improvements suggested for the developed system:

- System developed can integrate with system of university.
- System developed should eliminate the hardcoded data in the database.
- The mobile application can add feature that allow lecturer to take attendance without Internet connection.
- The mobile application can add more methods to get student ID number.

Suggested method:

- New feature of mobile application that allow student to send their ID number to lecturer's mobile device by using connectivity technology such as Near-Field-Communication (NFC), Bluetooth and Wi-Fi P2P.
- Fully integrate the barcode scanner application to the mobile application so user can use barcode scanner feature without installing the scanner application.

6.2 Conclusion

This project aims to provide a new method to take student attendance for all the universities. Lecturers can take student attendance, view student attendance and modify student attendance with mobile device. It is a “smarter” way to complete the task.

This project is completed by referring and following the project plan planned during the planning phases. The project breakdown in the project plan also acts as a guidance of this project.

In the conclusion, the result of this project has met all the requirements gathered during the planning and analysis phases. The system developed with the selected methodology. It is also passed all the test cases.

This project can be guidance and reference for future people who are going to develop the system similar to this project. People can refer this project and come up with a more creative and innovative idea.

REFERENCES

Android (no date) *Android Developer Tools* [Online] Available at <http://developer.android.com/tools/index.html>. Accessed on 18th July 2014.

IDAutomation (2012) *Barcode Educational Guide*. IDAutomation.com, Inc. [Online]. Available at <http://www.idautomation.com/barcode-faq/barcode-educational-guide.pdf>. Accessed on 2nd August 2014.

Janaswamy Phaniteja, P Derin J Tom (no date) *Evolution of Barcode*. International Journal for Development of Computer Science & Technology. [Online] Available at <http://www.ijdcst.com/pdf/Evolution%20of%20Barcode.pdf>. Accessed on 2nd August 2014.

Bibase Software (no date) *The ASCII Table*. [Image] Available at <http://www.bibase.com/ascii.htm>. Accessed on 3th August 2014.

FingerTec (2007) *Overview of Punch Card & Card Reader System* [Online] Available at <http://www.fingertec.com/newsletter/images/PunchCard-E.pdf>. Accessed on 8th August 2014.

Dr.M.Ramakrishnan, Josphineleela.R (2012) *An Efficient Automatic Attendance System Using Fingerprint Reconstruction Technique*. International Journal of Computer Science and Information Security. [Online] Available at <http://arxiv.org/ftp/arxiv/papers/1208/1208.1672.pdf>. Accessed on 8th August 2014.

Abhishek Jha (no date) *Class Room Attendance System Using Facial Recognition System*. The International Journal of Mathematics, Science, Technology and Management. [Online] Available at http://klresearch.org/IJMSTM/papers/v2i3_2.pdf. Accessed on 8th August 2014.

Eisaku Ohbuchi, Hiroshi Hanaizumi, Lim Ah Hock (2004) *Barcode Reader using the Camera Device in Mobile Phone*. IEEE Computer Society.

