

SMART ATTENDANCE SYSTEM USING QR CODE

YONG WEI LUN

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

**Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman**

April 2019

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 : Yong Wei Lun

ID No. : 1605763

Date : 26 April 2019

APPROVAL FOR SUBMISSION

I certify that this project report entitled **SMART ATTENDANCE SYSTEM USING QR CODE** was prepared by **YONG WEI LUN** 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 : Dr. Hoo Meei Hao

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.

© 2019, Yong Wei Lun. 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, Dr. Hoo Meei Hao for her invaluable advice, guidance and her enormous patience throughout the development of the research.

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

ABSTRACT

Recording attendance is something repetitive and time consuming. The process of attendance taking is the same and repeat for every day. However, the attendance system today is not automated. It requires a lot of manual workforce to accomplish it. An automated attendance system can save human labours, and increase efficient of attendance taking. This will directly help lecturers to save time, and spend more time on academic, rather than attendance records. This project intended to automate the attendance recording. The adopted development methodology is evolutionary prototyping to cater for constantly user feedbacks and improvements. The project had conducted testing on 2 classes, for one lecture class and one practical class. The results were accurate and eliminated the needs of signing attendance on attendance sheets, the manual efforts to transfer data on attendance sheets to computer system. Some improvements have to be made before it is fully functional, for instance, the camera is not zoomable at the moment. This caused inconvenience on students who sit at behind or with poor smartphone camera quality. In conclusion, the project had achieved the objectives, which ultimately save lecturers' time in managing attendance, bring convenience to students on attendance registration, and reduce the likelihood of fake attendance records.

TABLE OF CONTENTS

DECLARATION		ii
APPROVAL FOR SUBMISSION		iii
ACKNOWLEDGEMENTS		v
ABSTRACT		vi
TABLE OF CONTENTS		vii
LIST OF TABLES		xi
LIST OF FIGURES		xii
LIST OF ABBREVIATIONS		xv
LIST OF APPENDICES		xvi
 CHAPTER		
1	INTRODUCTION	16
	1.1 Introduction	16
	1.2 Background	16
	1.3 Problem Statement	16
	1.4 Goal and Objectives	18
	1.4.1 Goal	18
	1.4.2 Objectives	18
	1.5 Proposed Solution	18
	1.6 Proposed Approach	20
	1.7 Project Scope	20
	1.7.1 Target Users	20
	1.7.2 Platform	20
	1.7.3 Modules	21
	1.7.3.1 Mobile Module	21
	1.7.3.2 Web Module	21
	1.7.3.3 Backend Service Module	21
	1.7.4 Modules Not Covered	21

2	LITERATURE REVIEW	22
2.1	Introduction	22
2.2	Review on Existing System	23
2.2.1	Related Approach	23
2.3	Review on Attendance Tracking Technologies	23
2.4	Technologies Review	25
2.4.1	Single Page Application	25
2.4.2	Public Cloud Services	26
2.5	Development Methodology Review	26
3	PROJECT METHODOLOGY AND PLANNING	29
3.1	Introduction	29
3.2	Software Development Methodology	29
3.2.1	Evolutionary Prototype	29
3.2.2	Prototype Iteration	30
3.3	Research Method	30
3.3.1	Respondent's Regions	31
3.3.2	Summary of Survey	31
3.4	Project Plan	32
3.4.1	Work Breakdown Structure	32
3.4.2	Gantt Chart	32
3.5	Development Tools	32
3.5.1	Tools and IDEs	32
3.5.2	Services	33
3.5.3	Programming Languages	34
3.5.4	Frameworks	34
3.5.5	Database	35
4	PROJECT SPECIFICATION	36
4.1	Introduction	36
4.2	System Flow	36
4.3	Use Case	37
4.3.1	Use Case Diagram	37

4.3.2	Use Case Description	37
4.4	Software Requirements	39
4.4.1	Functional Requirements	39
4.4.2	Non-Functional Requirements	40
5	SYSTEM DESIGN	41
5.1	Introduction	41
5.2	Architecture Design	41
5.3	Database Design	43
5.4	Software Design	43
5.4.1	Reactive	43
5.4.2	Functional	44
5.5	User Interface	44
6	PROJECT IMPLEMENTATION	53
6.1	Introduction	53
6.2	Routes Hierarchy	53
6.3	Web API Endpoints	54
6.4	Web Socket	56
6.5	Code Generator	57
6.6	Amazon Web Service	59
6.6.1	Cognito	59
6.6.2	Elastic Beanstalk	60
6.6.3	API Gateway	62
6.6.4	Simple Storage Service	63
6.7	Firebase	63
6.7.1	Firebase Hosting	63
6.8	MongoDB Atlas	64
6.9	Process Flow Diagram	65
6.9.1	User Authentication	65
6.9.2	API Calls	66
6.9.3	QR Code Generation	66

7	PROJECT TESTING	67
7.1	Introduction	67
7.2	Unit Testing	67
7.3	User Interface Testing	69
7.4	Static Code Analysis	70
7.5	Code Quality Review	70
7.6	Application Audit	71
7.7	User Acceptance Testing	72
8	CONCLUSION AND DISCUSSION	74
8.1	Introduction	74
8.2	Limitation	74
8.3	Improvement and Recommendation	74
8.4	Challenges	74
8.5	Conclusion	75
	REFERENCES	76
	APPENDICES	78

LIST OF TABLES

TABLE	TITLE	PAGE
2.1	Comparison of Attendance Tracking Technologies	24
2.2	Comparison between SPA and Traditional Website	25
2.3	Comparison of Software Development Methodologies	27
3.1	Description of Phases for Prototype Model	30
4.1	Use case description of lecturer manage class	37
4.2	Use case description of lecturer manage attendance	38
4.3	Use case description of lecturer generate QR code	38
4.4	Use case description of student scan QR code	38
4.5	Use case description of student check attendance records	39
6.1	List of web API endpoints	54
7.1	UI test case	69

LIST OF FIGURES

FIGURE	TITLE	PAGE
1.1	The Proposed Solution Overview	19
1.2	Prototype Model	20
2.1	Attendance System at University Technology Petronas (UTP)	22
2.2	Public Cloud Adoption	26
3.1	Prototyping Model	29
3.2	Proposed Solution Architecture Design	35
4.1	IPO Diagram of Generating QR Code	36
4.2	IPO Diagram of Scanning QR Code	36
4.3	Use Case Diagram	37
5.1	System architecture design	42
5.2	MongoDB schema design	43
5.3	Login	44
5.4	Dashboard	44
5.5	Current class	45
5.6	QR code generation	45
5.7	Attendance list for current class	45
5.8	Timetable page to manage class slot	46
5.9	Add slot to a subject	46
5.10	Subject's overview	46
5.11	Past attendance records based on week	47
5.12	Bar list	47
5.13	Login	48
5.14	Dashboard of a student	49
5.15	QR code scanning for taking attendance	50
5.16	Timetable	51
5.17	Attendance records	52

6.1	Routes hierarchy for lecturer view web app	53
6.2	Routes hierarchy for student view web app	54
6.3	Simple diagram of a Web Socket connection over client and server	57
6.4	Code generator design	58
6.5	Cognito configuration for lecturer view authentication	59
6.6	Cognito configuration for student view authentication	60
6.7	The activity of Elastic Beanstalk that serves for web API	60
6.8	Elastic Beanstalk instance configuration with load balancer and auto scaling capacity	61
6.9	The activity of Elastic Beanstalk that shows target response time of web socket connection	61
6.10	The activity of Elastic Beanstalk that serves for web socket connection	61
6.11	Elastic Load Balancing configuration	62
6.12	Dashboard for API Gateway that shows activities	63
6.13	The storage used in S3	63
6.14	Usage of the Firebase Hosting	64
6.15	Overview of MongoDB running on 3 nodes, which are primary node and 2 secondary nodes	64
6.16	Overview of activities on primary node	65
6.17	User authentication process flow	65
6.18	Web API calls process flow	66
6.19	QR code generation process flow	66
7.1	Test results on browser	67
7.2	Test results on console	68
7.3	Test results on browser	68
7.4	Test results on console	68
7.5	Unit test result for backend repository	69
7.6	Code review scoring for lecturer view repository	70

7.7	Code review scoring for student view repository	71
7.8	Code review scoring for backend repository	71
7.9	Lighthouse audit report for lecturer view web app	71
7.10	Lighthouse audit report for student view web app	72
7.11	Conducting UAT on a practical class	73

LIST OF SYMBOLS / ABBREVIATIONS

<i>IDE</i>	Integrated Development Environment
<i>IPO</i>	Input Output Diagram
<i>SPA</i>	Single Page Application
<i>AWS</i>	Amazon Web Service
<i>UAT</i>	User Acceptance Testing

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
A	Survey Question	78
B	Results of Survey	79
C	Work Breakdown Structure	80
D	Gantt Chart	81
E	Lighthouse Audit	82
F	User Satisfaction Survey	83
G	Supervisor's comments	84

CHAPTER 1

INTRODUCTION

1.1 Introduction

This chapter discusses about the background, problem statement, goal and objective, proposed solution, proposed approach and project scope.

1.2 Background

The popularity of smartphone has increases over the year. This could change and speed up the attendance taking process in universities. The current attendance system is time consuming and required manual workload. Lecturers will let the students to sign on attendance list. Then, he or she will key in to the university's portal to record the attendance, each for every class. This will consume lecturers valuable time. Also, it will introduce human errors during the transferring from paper attendance to digitalised attendance records. Besides, students can easily cheat on attendance by asking flavours from their friends, to sign on the attendance sheet on their behalf. This is hard to avoid when the attendance sheets were given to students. The proposed solution is based on QR code to record students' attendance. The system will able to verify students' identity and prevent false registration. All of the attendance records will be recorded to the system and available to students and lecturers instantly. This will eventually reduce the human efforts on attendance registration.

1.3 Problem Statement

The process of taking students' attendance in higher education is time consuming and inefficient. The process become harder to manage when the class size is big. Prevention cheating on attendance is even harder to control. After each class, lecturers have to record the attendance to related web-based system according to the attendance sheets. The whole attendance taking process is consuming the valuable time of lecturers. With all the problem mentioned above, universities are not

changing the way they record attendance. Refer to Appendix A, 85% of the universities still recording attendance by signing on attendance sheets.

Mobile cellular penetration in Malaysia has reached 131.8% while smartphone penetration stood at 70% in the third quarter of last year (TheStar). With the widespread of smartphones among students in university, and the problem of wasting lecturers time in attendance taking process, proposes a solution that offer to simplify the attendance taking process. The proposed solution proposes QR code for students to scan with their smartphone. The attendance will be confirmed with the identity from the smartphone. This will save time and effort to record attendance at the same time, reduce unwanted paper usage.

The proposed solution identifies unauthorised attendance registration using multi-factor authentication. The details of authentication factors will be explained in following chapters.

1.4 Goal and Objectives

This section discusses about goal and objectives.

1.4.1 Goal

Reduce administrative and lecturers labour manually and minimise paper usage.

1.4.2 Objectives

- a. To simplified current attendance taking process by automating the process using QR Code scanning.
- b. Increase the transparency of attendance records by enabling students and lecturers to view attendance records anytime.
- c. Reduce cheating on attendance by enforcing cheat detection.
- d. Minimise paper usage by eliminating the needs of attendance sheets.

1.5 Proposed Solution

The proposed solution is by taking attendance using QR Code. 37.5% of respondents think that QR Code solution is the most cost effective.

The reason of using QR Code is because it provides low initial cost for implementing the system. By using the tools that everyone student has, smartphone to take attendance.

The solution has 3 modules, mobile module, web module and backend service module. Web module displays QR code that generated by backend service module. The QR code updates every 10 seconds to prevent cheating. This can prevent students who are not in class and wanted to scan the QR code. The mobile module enables students to scan QR code to confirm their attendance. The request is then sent to backend service module for verification. Once the attendance is verified, the backend service module will update database on attendance records. The whole process of attendance taking should be less than 5 minutes, which is much faster compared with traditional process.

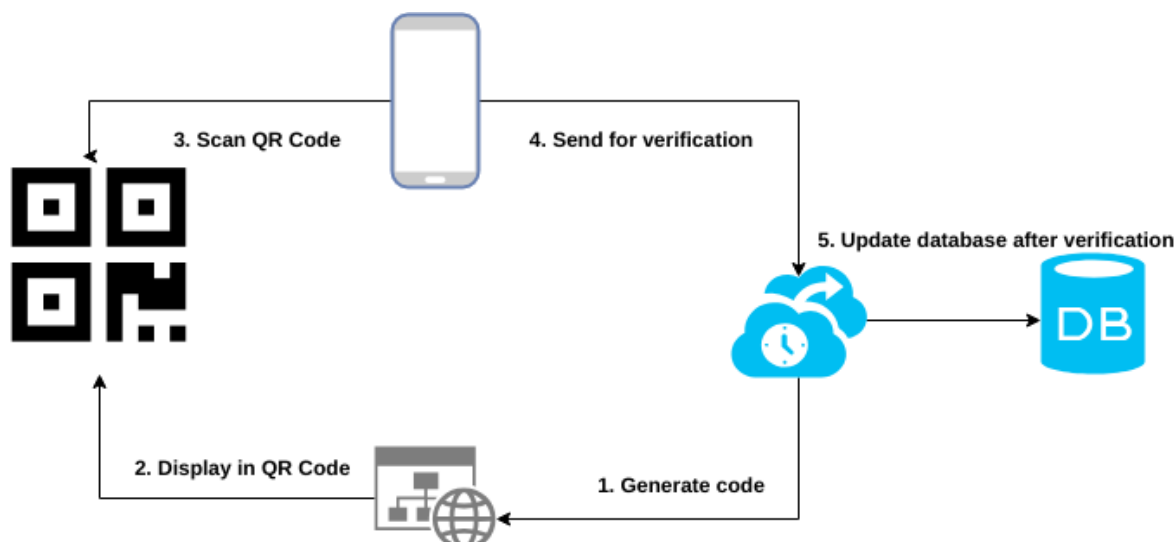


Figure 1.1: The Proposed Solution Overview

Students can view their attendance records for enrolled subjects, this can increase the transparency of attendance system. The system also issue notification if the attendance percentage is below than certain level. Moreover, lecturers also can manage students records with web module. Lecturers can update students' attendance records in case they have valid reason to absent class. Lecturers can manage class as well. New class can be created in case there is any class replacement. Existing classes can be modified or deleted by lecturers.

Administrators can manage users with web module. They can add new lecturers or students, update their details and delete existing users.

1.6 Proposed Approach

The proposed methodology is Evolutionary Prototyping Methodology.

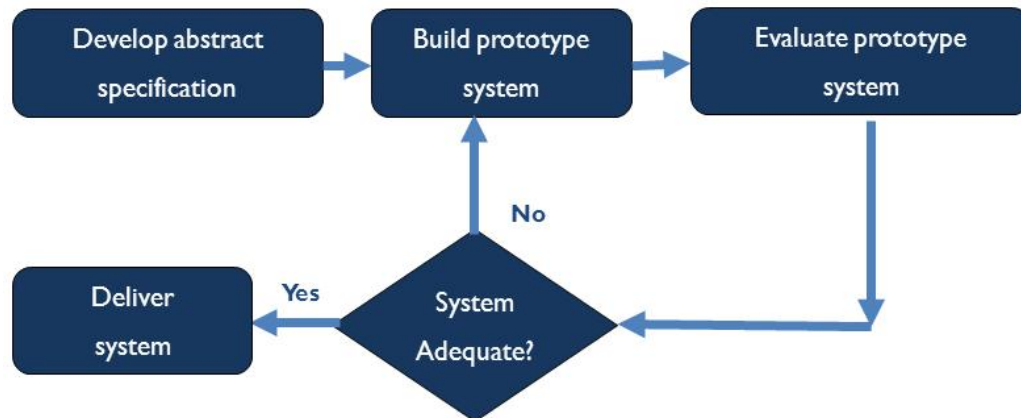


Figure 1.2: Prototype Model

Prototype model involves users in the development stage. This can have clearer understanding of requirements based on users' feedback. Errors can be detected in the early stage; missing module can be identified easily. An attendance system should be reliable and easy to use. Thus, it is good to involve users in the development stage.

1.7 Project Scope

The section below explains the target users, platform and modules of the project.

1.7.1 Target Users

- i. UTAR students
 - To records their attendance for every class.
- ii. UTAR Lecturers
 - To show the QR Code to students during class.
 - View and manage student's attendance records.
- iii. UTAR administrator staffs
 - Manage timetable, classes and users (students and lecturers).

1.7.2 Platform

The system consists 2 platforms, web and mobile platform. Web is for lecturers. Students can be accessing from mobile platform.

1.7.3 Modules

The section below explains modules in mobile, web and backend service.

1.7.3.1 Mobile Module

- i. Record attendance by scanning QR Code.
- ii. View attendance records.
- iii. Issue notification if students' attendance lower than certain limit.

1.7.3.2 Web Module

- i. View, add, modify, classes.
- ii. View, modify attendance records.
- iii. View overall attendance records in charts.
- iv. Display QR Code for each class.

1.7.3.3 Backend service Module

- i. Prevent unauthorised registration of attendance with multi-factor authentication.
- ii. Provides login authentication on both mobile and web.
- iii. Generate code for QR code.
- iv. Run data analytics operation to generate charts for attendance records.

1.7.4 Modules that are not covered

- a. Tutorials on how-to-use on both mobile and web modules.
- b. Web module for administrator to manage user pools and timetable data.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter reviews on existing system, attendance tracking technologies, technologies and development methodologies.

2.2 Review on Existing System

According to appendix A, most of the universities and colleges still record attendance on attendance sheets. Attendance system is clearly missing out the evolution.

According to Hsarena (2017), University Malaysia Sarawak (UNIMAS) was the first in Malaysia to use QR Code to record attendance in October 2017 followed by University Technology Petronas (UTP) in January 2018.



Figure 2.1: Attendance System at University Technology Petronas (UTP)

Similar system has implemented on University Technology Petronas and Multimedia University as well. Lecturers will sometimes give a pop quiz to prevent cheating. No further prevention is done.

2.2.1 Related Approach

Masalha (2014) proposes that using QR code to record attendance with additional security to prevent unauthorized attendance registration using multi-factor authentication, which are username and password with facial identification check. To ensure that the students is present in the class while scanning the QR code, the location will be check as well.

Saraswat (2010) proposes that using fingerprint scanner to records attendance. Fingerprint is something we all have uniquely and accessible all the time. By using fingerprint scanner, it will prevent cheating on attendance and provide convenient yet secure way to record attendance.

Balcoh, et al. (2012) proposes an algorithm for efficient attendance management. With face recognition, the effort of recording attendance is reduced to minimum. The process is done behind the scene. The algorithm able to identify multiple faces from a single image. This will ultimately prevent cheating on attendance.

We noticed that proposal of automated attendance systems is not something new. Different regions have different preferences on attendance taking system. Ultimately, the system will reduce labour work and prevent attendance cheating.

2.3 Review on Attendance Tracking Technologies

The table below shows some of the examples of attendance tracking technologies (The Hans India, 2018). Some of the low-cost setup are using polling, geolocation, and QR code.

Polling required students to enter the answer that asked by lecturers during the class or enter a special unique code. This however is easy to cheat because students can send out the code or answer to their friends who are not in the class. Next, tracking attendance by using geolocation. When students enter into class, the system will detect and record attendance. However, GPS often suffer with accuracy in indoor condition. This will make the attendance not accurate. Thirdly, QR code. Students required to scan the QR code that displayed during class to confirm

attendance. This method also can be cheat easily, by sending the QR code to friends who are not in the class. We could enhance the security by adding few mechanisms. QR code should be refreshed at fast as 10 seconds, so that when the QR code is sent to students who are not in the class, the code will be invalid. The system also needs to allow one device to login at student account at a time. This will prevent cheating on attendance as well.

Table 2.1: Comparison of Attendance Tracking Technologies

Tracking technologies	Description	Required equipment	Easy to fake attendance	Cost
Polling	Required students to answer a question or enter code to devices	Laptop or smartphone	Easy	low
Geolocation	Automatically records attendance when the location of students matches with a particular class	Smartphone	Easy	Low
Card Reader	Required students to scan their student ID on card readers	Student ID, card reader	Medium	Medium
Fingerprint Reader	Required students to scan their fingerprint on fingerprint reader to register attendance	Fingerprint reader	Hard	High
QR Code	Required students to scan QR code that displayed in the class	Smartphone	Medium	Low
Bluetooth	Automatically register attendance when within the Bluetooth coverage area	Smartphone with Bluetooth turned on	Medium	Medium

2.4 Technologies Review

This section discusses about technologies that are going to be used.

2.4.1 Single Page Application

Mikowski and Pollen (2013) said that Single Page Application (SPA) has been around for long time. SPA is an application that run on the browser and does not reload the page when using. SPA can render like a desktop application as it redraws the part of the user interface that needs to be update when needed. Unlike traditional website, it redraws on every request as well as user action. The website fully depending on web server. In the case of the server is overloaded or busy, or the internet connection is unable, the page will not response to user action as the user might think that the page is not responding. A page refresh will cause the request to be resent. This deliver horrible experience to the user.

SPA can show its state just like native application. When a SPA is loading, it can render progress bar or indicator dynamically. This will create better user experience, user does not have to keep on refresh the page to check its state, compared to traditional website.

SPA is universal and cross-platform. It works with any platform as long as have access to modern web browsers. It also can be update instantly and distribute.

Moreover, SPA can easily port to Progressive Web App, mobile application and Desktop application with the aids of current technologies.

Table 2.2: Comparison between SPA and Traditional Website

	Single Page Application	Traditional website
Reload	Does not reload on user action	Reload on every user action
Response	Nearly immediate response	Solely relay on servers and internet connection
Show its state	Able to show notify users by showing progress bar or indicator dynamically	Not able to notify users of its state

2.4.2 Public Cloud Services

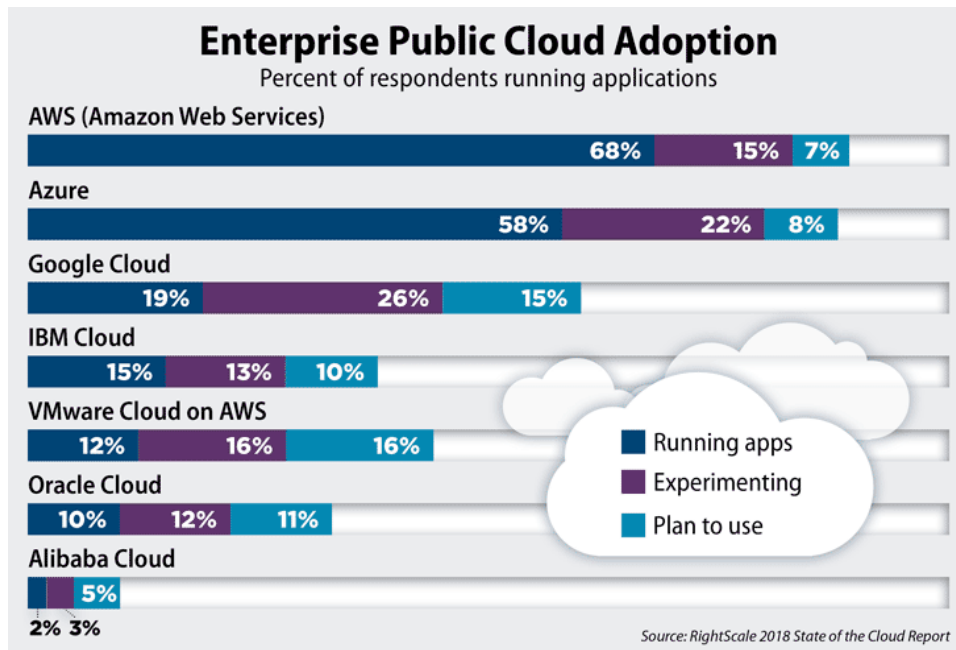


Figure 2.2 Public Cloud Adoption

Source taken from (Cynthia & Andy, 2018)

According to Cynthia and Andy (2019), Amazon Web Service (AWS) is dominating in public cloud market shares. AWS has been in the market share leader in cloud IaaS for over 10 years. Although Microsoft Azure and Google Cloud Platform (GCP) are catching up in recent years. AWS offers the most services compared with Azure and GCP. Therefore, in the project AWS is chosen for the cloud services.

2.5 Development Methodology Review

In software project, adopting a rigid or flexible approach can influence on the project outcome. With the face pace in software industry, a suitable methodology will definitely help. Code reduction, reusing and security should be neglect too.

Clear requirements mean the software requirements are clear. Clear solutions mean the proposed solutions are clear. Feedback on work done means that the feedback will be given during development. Frequent enhancements are the enhancements that need to be made after feedbacks. Cost of delay means that the cost when the project is delayed. Experience on projects is based on the software engineer's experience on similar solution.

Waterfall methodology is suitable when the requirements and solutions are clear, less feedback and enhancement during development, low cost of delay and software engineer have experience on similar project.

Kanban methodology is suitable when the requirements are clear, solution are unclear, required constant feedback but few changes during development, low cost of delay and software engineer may have experience on similar project.

Scrum methodology is suitable when the requirements are clear, solutions are unclear, required constant feedback but few changes during development, high cost of delay and software engineer may have experience on similar project. Similar with Kanban methodology, just the difference in cost of delay.

Extreme programming is suitable when the requirements and solutions are unclear, required constant feedback and frequent enhancements, high cost of delay, and software engineer has less to no experience.

Evolutionary prototyping methodology is suitable when requirements and solutions are clear, required constant feedback and frequent enhancements, low cost of delay and software engineer may have experience on similar projects.

Table 2.3: Comparison of Software Development Methodologies

Methodologies	Waterfall	Kanban	Scrum	Extreme Programming	Evolutionary Prototyping
Clear requirements	Clear	Clear	Clear	Unclear	Clear
Clear Solutions	Clear	Unclear	Unclear	Unclear	Clear
Feedback on work done	Minimum	Constant feedback	Constant feedback	Constant feedback	Constant feedback
Frequent enhancements	No changes	Few changes	Frequent changes	Frequent changes	Frequent changes
Cost of delay	Minimum	Minimum	Huge	Huge	Minimum
Experience on projects	Have experience	May have experience	May have experience	Less to no experience	May have experience

The proposed solution has clear requirements and solution, need constant feedback and frequent changes and low cost of delay. Thus, Evolutionary Prototype methodology will be the best fit.

CHAPTER 3

PROJECT METHODOLOGY AND PLANNING

3.1 Introduction

This chapter explains about software development methodology and research method that are chosen and project planning.

3.2 Software Development Methodology

This section discusses about development methodology.

3.2.1 Evolutionary Prototype

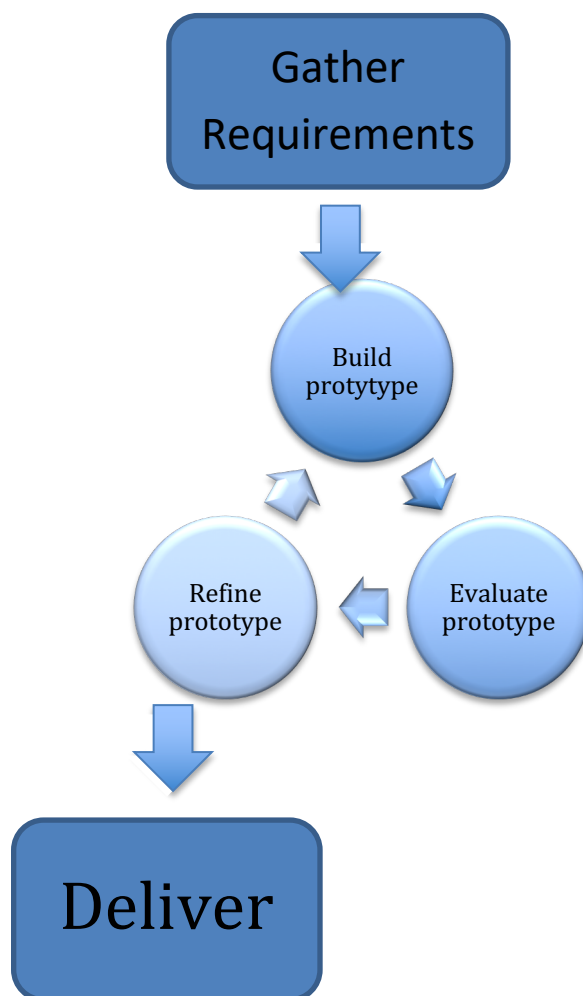


Figure 3.1: Prototyping Model

Evolutionary prototyping involves the building of prototype software which incrementally evolves into the final system (Carter, 2001). It has 5 phases of development.

Table 3.1 Description of Phases for Prototype Model

Phase	Description
Gather requirements	Gather all the requirements. The requirements were gathered from the lecturer and produce requirement specifications.
Build prototype	Build the prototype accordingly. The prototype was built incrementally based on user feedbacks.
Evaluate prototype	Evaluate prototype accordingly. The prototype was evaluated by users, and feedbacks have been collected for further evaluations.
Refine prototype	Refine prototype based on feedback. The prototype was refined based on the feedbacks and prepared for the next iteration of prototyping.
Deliver	Deliver the system to end user

3.2.2 Prototype Iteration

For the solution have 2 iterations of prototype. During each iteration, evaluation had conducted and refine at next iteration.

3.3 Research Method

This section discusses the research method for the project including respondents' selection and summary of the research. The research is conducted through online survey.

3.3.1 Respondents' Regions

There are total 103 higher education centre in Malaysia, 20 public universities, 43 private universities, 31 university colleges and 9 foreign university branches (StudyMalaysia). A survey was conducted with 20 universities as list below with one representative from each university.

Overseas

- i. Loma Linda University
- ii. London School of Economics
- iii. Newcastle University
- iv. Nanyang Technological University

Local

- i. University Science Malaysia
- ii. Health Campus University Science Malaysia
- iii. Multimedia University
- iv. Sunway College University
- v. The One Academy
- vi. University Malaysia Terengganu
- vii. University Utara Malaysia
- viii. UCSI University
- ix. Nilai University
- x. University of Malaya
- xi. National University of Malaysia
- xii. Infrastructure University Kuala Lumpur
- xiii. Tunku Abdul Rahman University College
- xiv. Asia Pacific University of Technology and Innovation
- xv. HELP University
- xvi. YPC University

3.3.2 Summary of Survey

Based on Appendix B, 90% of the respondents' universities still recording attendance by signing on attendance sheet, 5% using QR code attendance system and

5% using Barcode scanner on Student ID. This shows that universities are not aware of the attendance systems are outdated, with only a few exceptions.

75% of the respondents have no issue with current attendance system and 75% of the respondents think that cheating on attendance are easy. This shows that current attendance systems don't have much impact on them, from students' perspective. However, automated attendance systems will save lecturers' time and reduce human error when recording attendance to particular web system. 75% of respondents agreed with that and 60% of respondents think attendance system should be automated. Automation needs to be done in order to save lecturers' time from repetitive tasks.

40% of the respondents think that using biometrics scanner is the best to prevent cheating on attendance. While 20% of them think using geolocation and Student ID reader and 15% of them think using QR code. However, while biometrics scanner might be the most secure to prevent cheating on attendance, it might have some drawbacks such as high cost. 35% of the respondents think that QR code is the most cost-effective attendance tracking technology, while 25% of them think polling, 20% of them think Student ID reader and 10% of them think geolocation and biometrics scanner.

3.4 Project Plan

This section discusses about Work Breakdown Structure and Gantt chart.

3.4.1 Work Breakdown Structure

Work Breakdown structure is attached as appendix C.

3.4.2 Gantt Chart

Gantt chart is attached as appendix D.

3.5 Development Tools

This section discusses about tools and integrated development environment (IDE), services, programming languages, frameworks and database.

3.5.1 Tools and IDEs

- i. Git

Git is be used as version control. It is widely used by software development. It is a distributed version control system. It is used to track the software version to ensure that the current changes would not break the software. If it does, we can roll back to the previous version.

ii. MongoDB Compass

MongoDB Compass is a visualisation tool for MongoDB. It eased the development with MongoDB with the handy visualisation feature.

iii. JetBrains IDEs

JetBrains IDEs that using are be Webstorm and GoLand. Webstorm is popular with frontend web development and GoLand in designed for Go programming language. These IDEs were chosen in order to maximise productivity due to familiarity and the features offered in each IDEs.

3.5.2 Services

i. Amazon Web Service (AWS)

Amazon Web Service is a public cloud service that offer variety of services. It supports worldwide with different regions. AWS is used for cloud-native architecture design.

i) Amazon Elastic Beanstalk

Amazon Elastic Beanstalk is a Platform-as-a-Service that provides managed environment for servers to execute. This service is suitable for all kind of server deployments

ii) Amazon Cognito

Amazon Cognito is service that provides user authentication and access on clients. This eased the processing of development user authentication.

iii) Amazon API Gateway

Amazon API Gateway is service for API deployment. It well integrated with AWS services and provide secure APIs at any scale. This service provides a HTTPS connection.

iv) Amazon S3

Amazon S3 is a managed cloud object storage service. It can be used to store a variety of files, as well as web hosting. It is highly available and accessible anytime.

ii. MongoDB Atlas

MongoDB Atlas is a database as-a-service offered by MongoDB. It runs on 3 public clouds, Amazon Web Service, Google Cloud Platform and Microsoft Azure. By using this service, we do not need to manage our database instances.

iii. Firebase

Firebase is a mobile and web development platform. It offers a variety of services that suit for mobile and web development.

i) Firebase Hosting

Firebase hosting is a static web hosting service that provides HTTPS. It is used for secure web hosting because only a secure connection can access a mobile's camera.

iv. GitHub

GitHub is a development platform that offers web hosting for a version control system. It supports Git out of the box. GitHub is widely used by commercial and open source software development.

3.5.3 Programming Languages

i. TypeScript

TypeScript is a superset of JavaScript that was developed and maintained by Microsoft. It adds optional types and transpiles to high-performance JavaScript which is suitable for large-based applications.

ii. Go

Go is a programming language that was developed and maintained by Google. It makes building simple, reliable and effective software easy. It is good in performance and multi-threading tasks with a light weight thread called Goroutine.

3.5.4 Frameworks

i. Angular

Angular is an open source framework for front-end web application and based on TypeScript which is designed and maintained by Angular Team at Google.

ii. Ionic

Ionic is an open source framework for building hybrid mobile application and progressive web app. Ionic is heavily influenced by Angular, and the knowledge can be transferred from Angular to Ionic.

3.5.5 Database

i. MongoDB

MongoDB is an open source document-stored NoSQL database. It is well integrated with popular programming languages. It is used due to the performance and flexibility of the database structure.

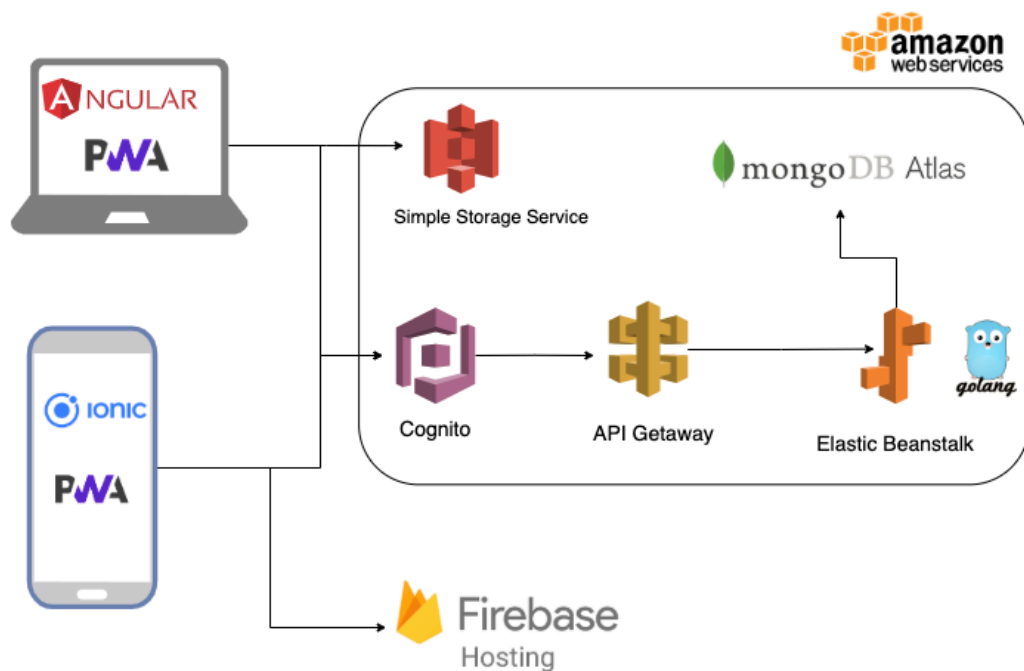


Figure 3.2: Proposed Solution Architecture Design

CHAPTER 4

PROJECT SPECIFICATION

4.1 Introduction

This chapter discusses about system flow, use case and project requirements.

4.2 System Flow

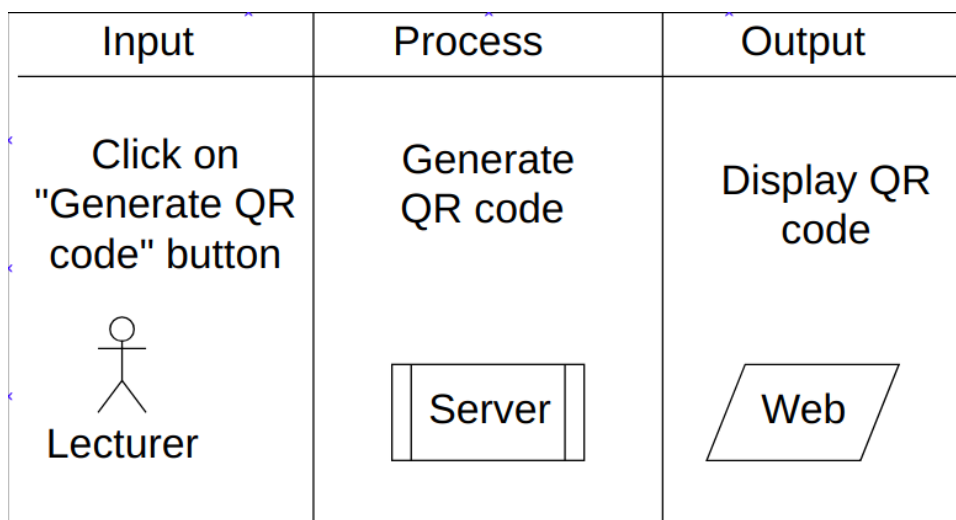


Figure 4.1: *IPO* Diagram of Generating QR Code

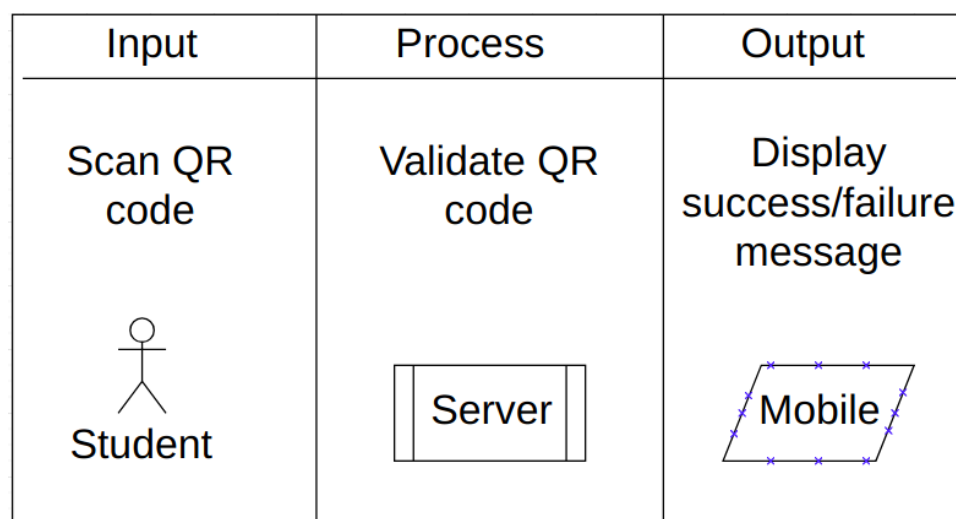


Figure 4.2: *IPO* Diagram of Scanning QR Code

4.3 Use Case

This section discusses about use cases.

4.3.1 Use Case Diagram

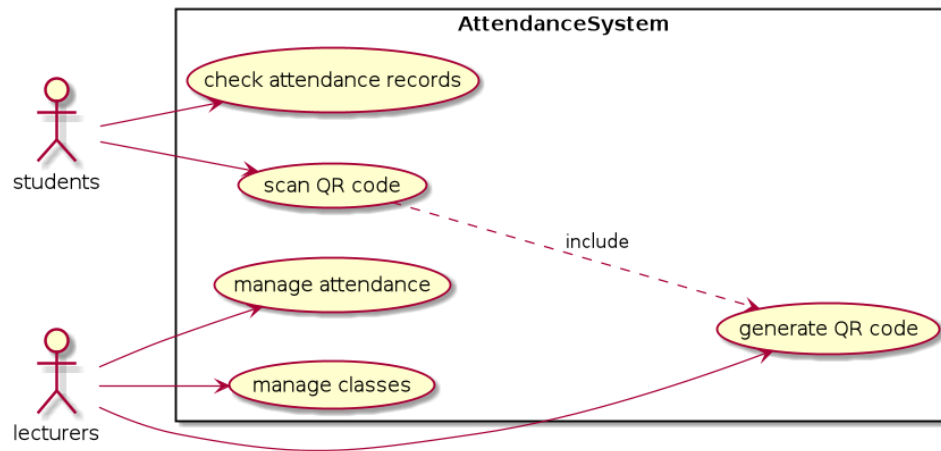


Figure 4.3: Use Case Diagram

4.3.2 Use Case Description

Table 4.1: Use case description of lecturer manage class

Use case name	Manage class
Actor	Lecturer
Flow of Events	
<p>1. Create class</p> <ol style="list-style-type: none"> i. Lecturer create class. ii. Lecturer fill in class name, class type and duration. iii. System sends the request to server and saves to database. 	
<p>2. View class</p> <ol style="list-style-type: none"> i. Lecturer selects class. ii. System displays class name, class type, duration. 	
<p>3. Update class</p> <ol style="list-style-type: none"> i. Lecturer selects class. ii. Lecturer update the field that desired to change. iii. System sends the updated records to server and saves to database. 	

4. Delete class

- i. Lecturer selects class.
- ii. Lecturer deletes class.
- iii. System sends the request to server and deletes from database.

Table 4.2: Use case description of lecturer manage attendance

Use case name	Manage attendance
Actor	Lecturers
Flow of Events	
<p>1. View attendance</p> <ol style="list-style-type: none"> i. Lecturer selects student from a class. ii. System displays the attendance records for the student of the class. <p>2. Update attendance</p> <ol style="list-style-type: none"> i. Lecturer selects student from a class. ii. Lecturer update the attendance status. iii. System sends the updated records to server and saves to database. 	

Table 4.3: Use case description of lecturer generate QR code

Use case name	Generate QR code
Actor	Lecturers
Flow of Events	
<ol style="list-style-type: none"> i. Lecturer clicks on the “Generate QR code” button to generate QR code. ii. System sends request to server and retrieve code for the class. iii. System displays the QR code. 	

Table 4.4: Use case description of student scan QR code

Use case name	Scan QR code
Actor	Students
Flow of Events	
<ol style="list-style-type: none"> i. Student clicks on the “Scan QR code” button to scan QR code ii. System sends the request to server for verification iii. System displays success or failure message 	

Table 4.5: Use case description of student check attendance records

Use case name	Check attendance records
Actor	Students
<p>Flow of Events</p> <p>1. Check for attendance status for a class</p> <ol style="list-style-type: none"> i. Student clicks on a class. ii. System displays the attendance status of the class. <p>2. Check for overall attendance records for a subject</p> <ol style="list-style-type: none"> i. System shows the percentage of the attendance records for a subject on home page. 	

4.4 Software Requirements

This section discusses about functional and non-functional requirements.

4.4.1 Functional Requirements

Lecturers

- 1) The system shall be able to generate QR code
- 2) The system shall be able to manage attendance records
 - i. The system shall be able to view attendance records
 - ii. The system shall be able to update attendance records
 - iii. The system shall be able to view bar list
- 3) The system shall be able to manage classes
 - i. The system shall be able to create classes
 - ii. The system shall be able to view classes
 - iii. The system shall be able to update classes
 - iv. The system shall be able to delete classes

Students

- 1) The system shall be able to scan QR code to record attendance
- 2) The system shall be able to check attendance records
 - i. The system shall be able to check attendance records for all classes
 - ii. The system shall be able to view students' attendance percentage
 - iii. The system shall be able to view class details with time and venue

4.4.2 Non-Functional Requirements

- i. The system shall prevent cheating attendance by verify the request.
- ii. The system shall prevent unauthorised login on users' account.
- iii. The system shall be error-tolerant.
- iv. The system shall be fast and responsive.

CHAPTER 5

SYSTEM DESIGN

5.1 Introduction

This chapter discuss about software modelling and design, database design and process modelling.

5.2 Architecture Design

The system architecture design is practicing cloud-native architecture and microservices. Cloud-native architecture is the design practice that specially designed to run on cloud, to take the advantages of cloud computing. It is taking account of the characteristic and nature of the cloud services, and resulting the application fully take advantage of the cloud. This enabled the application can be scale horizontally effectively when needed. Also, practicing cloud-native architecture, we do not have to manage the infrastructure ourselves most of the time. The time can be reduced for setting up servers on-premise.

Besides, the architecture design also has microservices in mind. Microservices is a design practice that split the application into multiple pieces, which are self-container and standalone services. Microservices increases the scalability and maintainability of the application. Compared with Monolith application, microservices required extra steps to set up and get connected to each service. However, microservices is the preferred architecture style in modern design.

By combining the two, cloud-native and microservices architecture design, the application can fully take the advantages of the cloud services. Several cloud services are leverage due to the application are splitting into separate services. Due to the nature of the cloud, each service can scale horizontally and independently effectively when needed. This offers better pricing compared with on-premise servers' setup. With cloud, scale up and down can be done easily, create and terminate instance at any time.

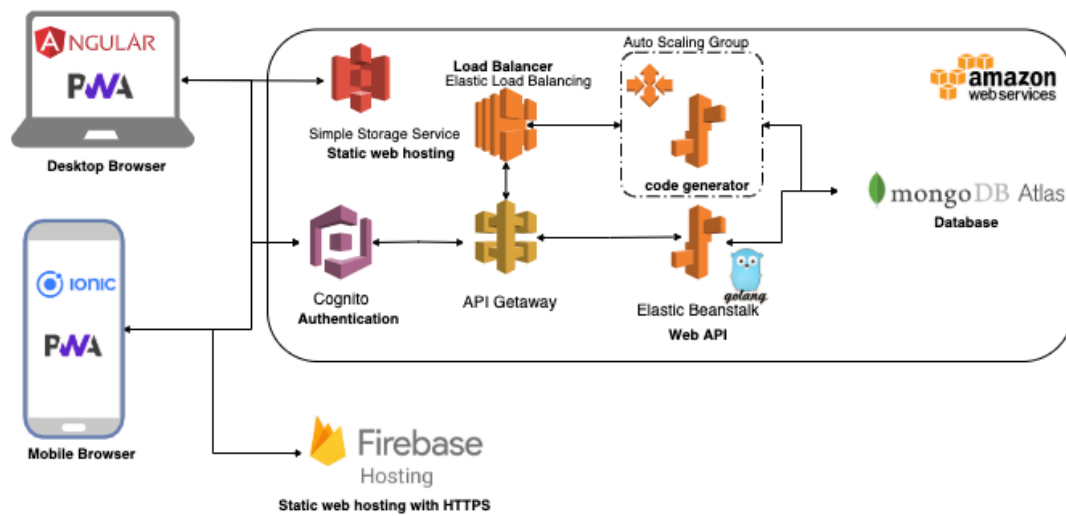


Figure 5.1: System architecture design

The diagram above shown the overall system architecture design for the application. Most of the services run on Amazon Web Service some of them on Google Firebase. We have two instances of Elastic Beanstalk. One for the REST API web service, the other for code generating service. These services have different workloads during throughout the day, it is better to split them into separate services. All the web services can be accessed from a same gateway, which is the API Gateway. The API Gateway mapped to the related services based on the requests and response accordingly. It as a unified gateway for all the web services.

The database also has been split into different service. This is to ensure that, when one of the web services is faulty, it would not affect other services. Database is important for data persistence; it is good to separate it from other services.

Moreover, the user authentication service, Cognito has been separated from the web services that we mentioned earlier. User authentication workload can be unpredictable. Users may have logged in to the system, but does not use it. By separate the authentication service with other services to ensure that it does not break other services when there is a lot of users logging in. Also, only authenticated users can access the other web services. It also acts as a secure door for all the services.

Lastly, the static web hosting. Web hosting also has been separated as a standalone service. This is to ensure that anything goes wrong during the web hosting process would not break other services.

Cloud-native architecture and microservices can be applied on any types of application, to improve the scalability and maintainability.

5.3 Database Design

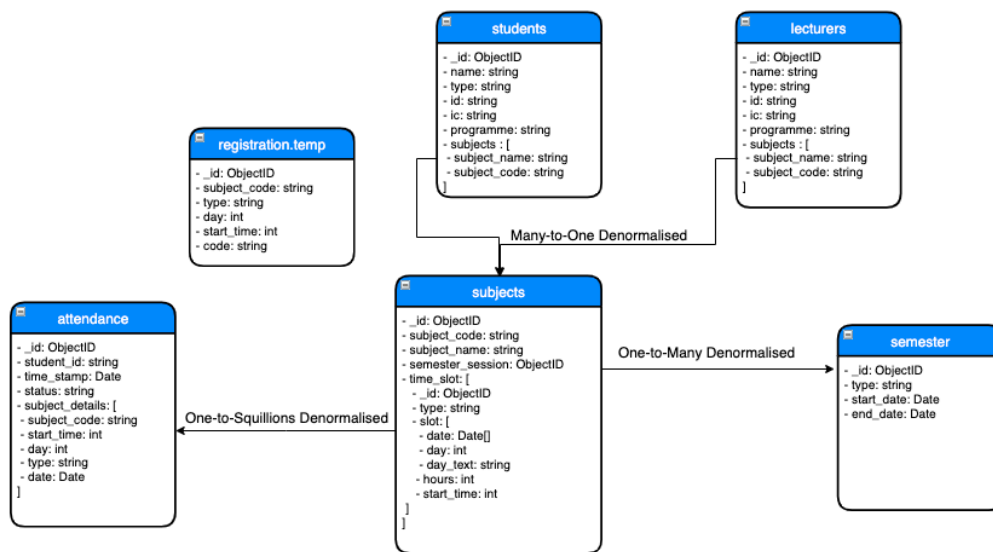


Figure 5.2: MongoDB scheme design

5.4 Software Design

The application is designed with service-oriented, reactive and functional.

5.4.1 Reactive

Reactive programming is an asynchronous programming paradigm that concerned with propagation change and data stream. RxJS (Reactive Extension for JavaScript) is a library for reactive programming. It uses observables to make it easier to compose asynchronous, or callback-based code. RxJS is heavily used in the frontend application in Angular and Ionic.

5.4.2 Functional

Functional programming is a programming paradigm that treats computation as the evaluation of pure mathematics functions, and at the same time, avoids mutable data and changing-state. Functional programming promotes clean code and referential transparency. Functional programming principle are heavily used in both TypeScript and Go programming languages.

5.5 User Interface

Two web application were developed, one is for lecturer, the other is for student.

Lecturer view web application

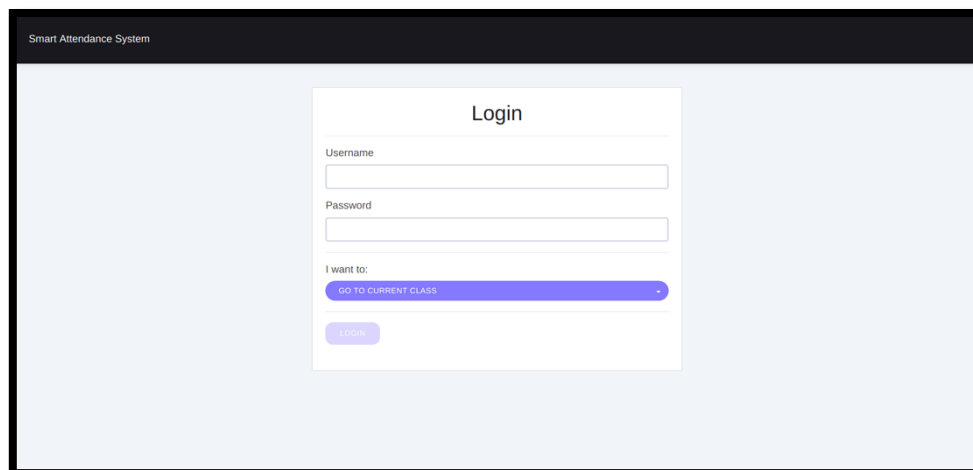


Figure 5.3: Login

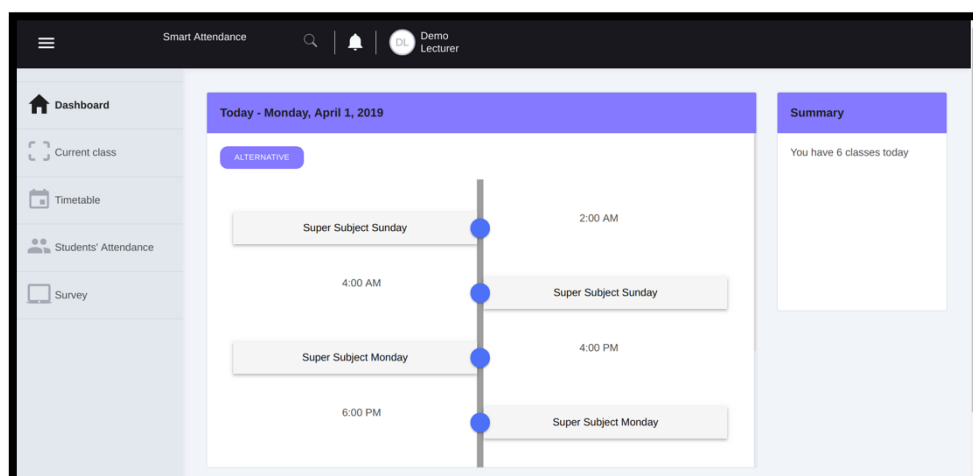


Figure 5.4: Dashboard

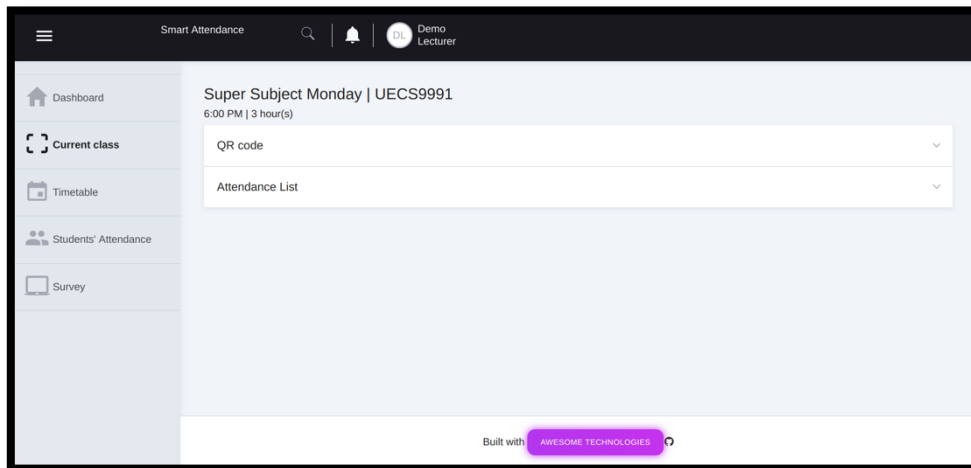


Figure 5.5: Current class

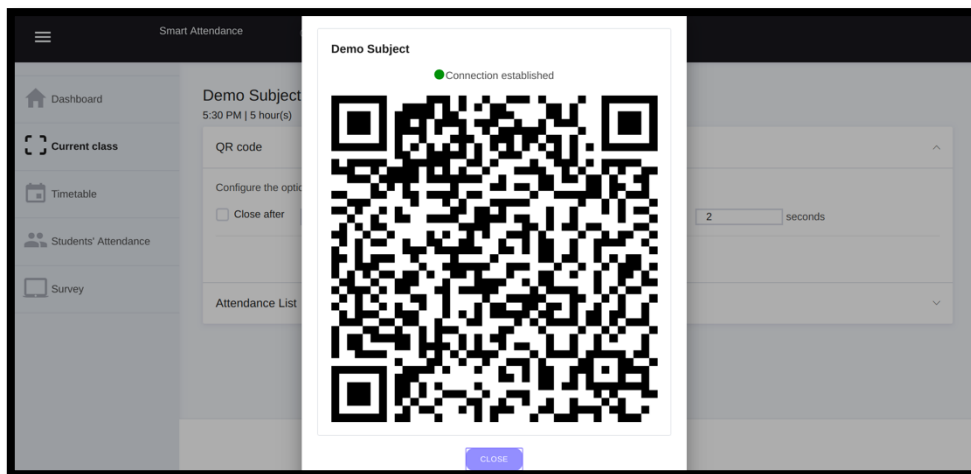


Figure 5.6: QR code generation

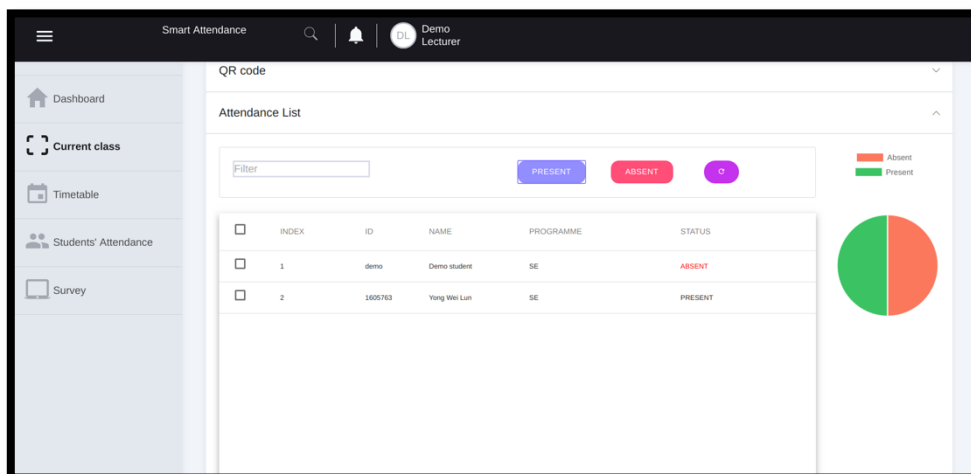


Figure 5.7: Attendance list for current class

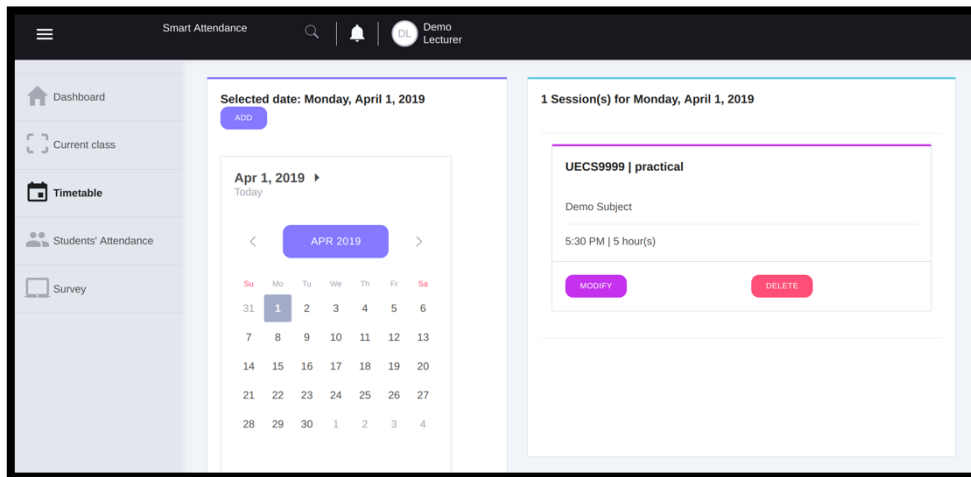


Figure 5.8: Timetable page to manage class slot

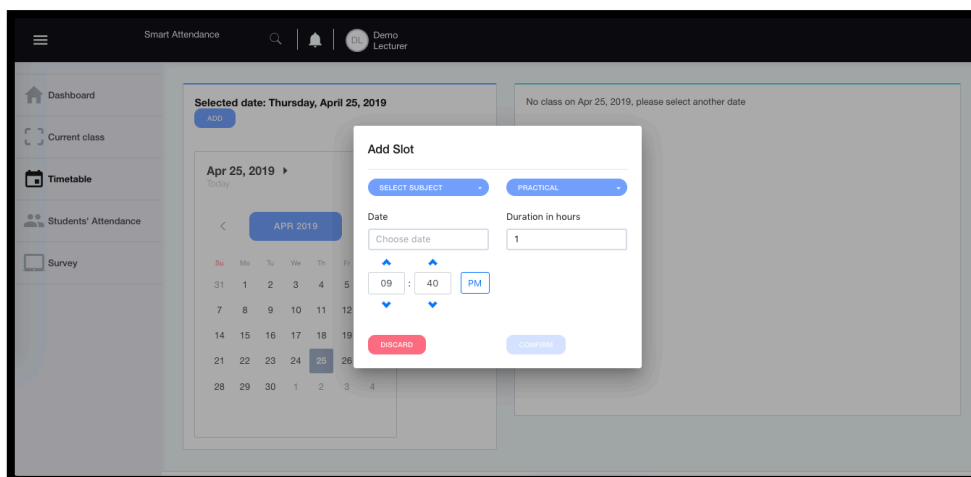


Figure 5.9: Add slot to a subject

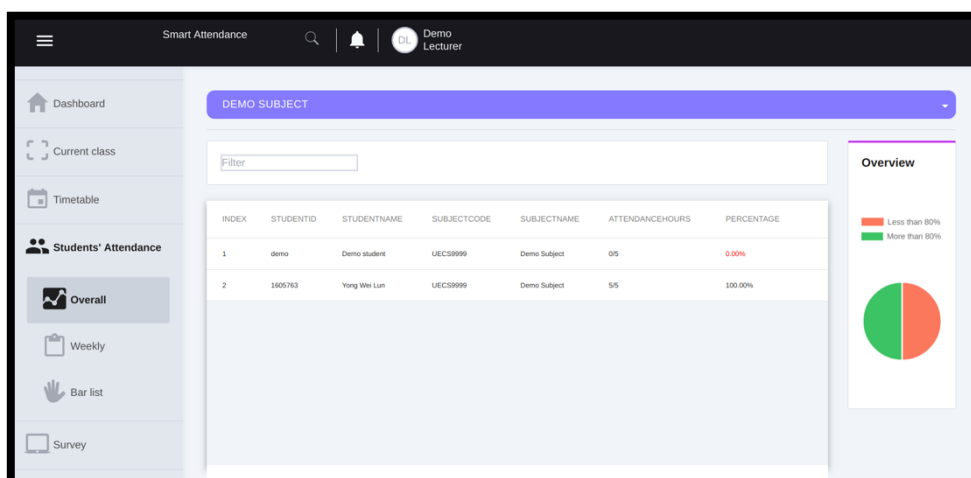


Figure 5.10: Subject's overview

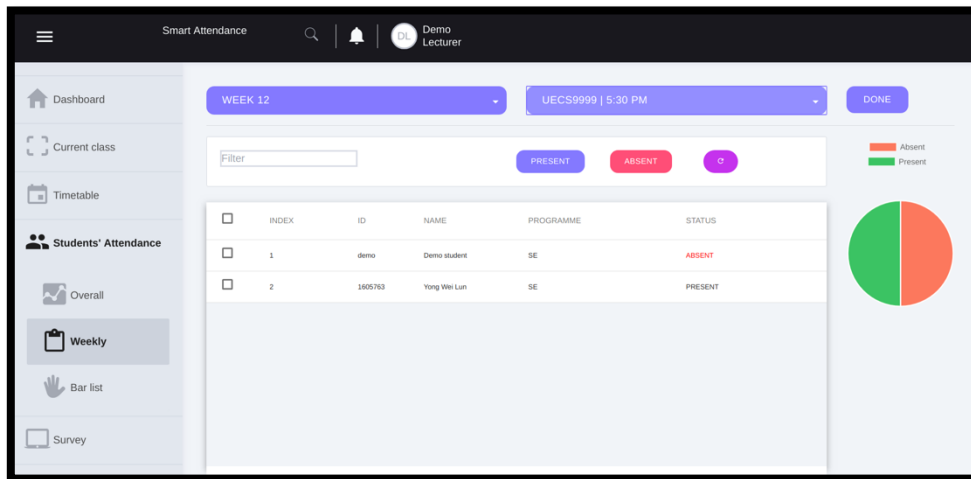


Figure 5.11: Past attendance records based on week

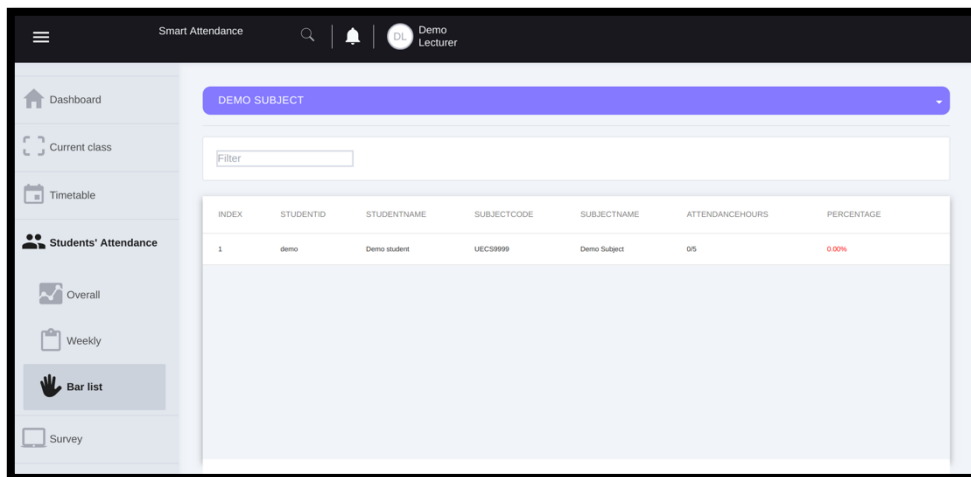
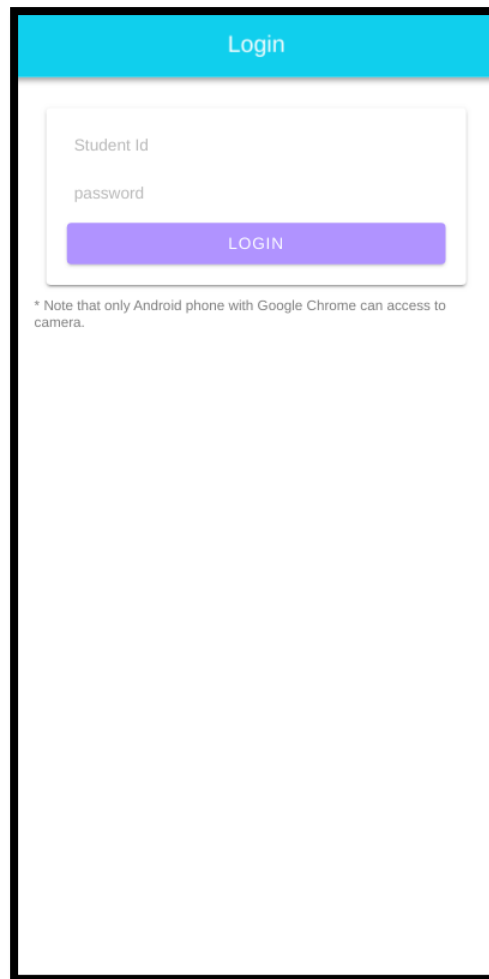


Figure 5.12: Bar list

Student view web application



The image shows a mobile application interface for a login screen. At the top, there is a blue header bar with the word "Login" in white text. Below the header is a white rectangular form containing two input fields: "Student Id" and "password". Below these fields is a purple button with the text "LOGIN" in white. Underneath the form, there is a small note: "* Note that only Android phone with Google Chrome can access to camera." The entire interface is enclosed in a black border.

Figure 5.13: Login

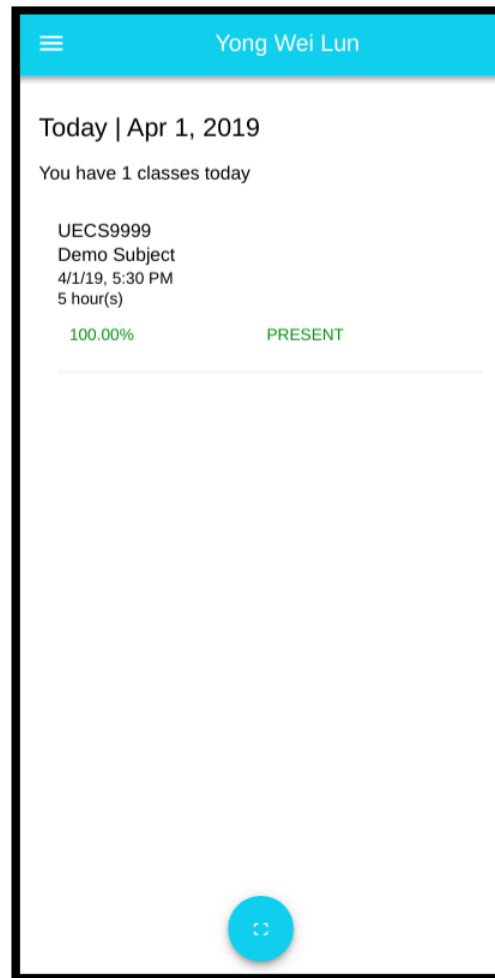


Figure 5.14: Dashboard of a student

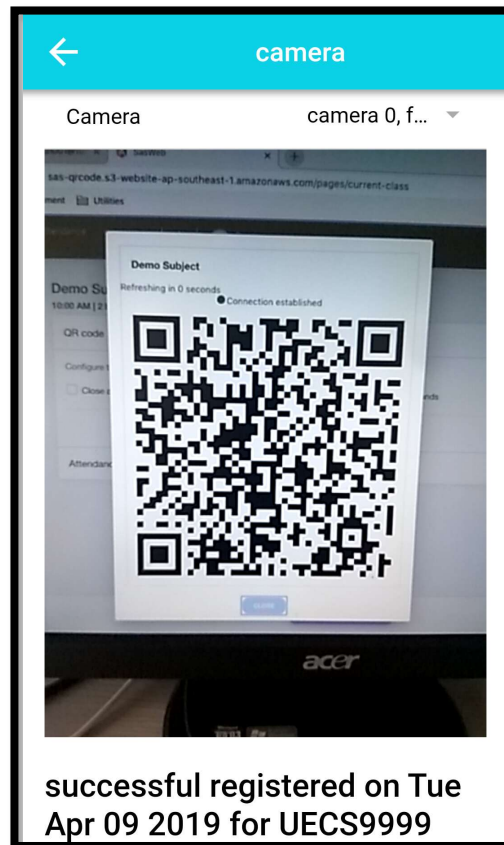


Figure 5.15: QR code scanning for taking attendance

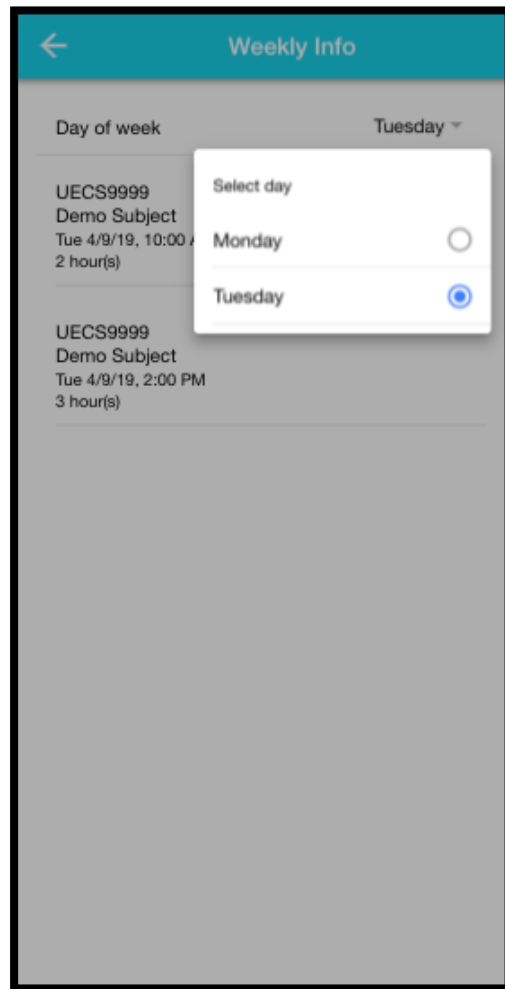


Figure 5.16: Timetable

The image shows a mobile application screen titled "Attendance Info". At the top, there is a blue header with a white back arrow on the left and the text "Attendance Info" in white. Below the header, the text "Current percentage: 100.00%" is displayed in green. Underneath, it says "1 records found for UECS2344". The next section lists the details of the record: "Tue 3/26/19, 11:00 AM", "practical", and "3 hour(s)". Below this, the status "PRESENT" is shown in green, followed by the timestamp "30/3/19 4:08:22 PM". A horizontal line is positioned below the record details.

Attendance Info	
Current percentage: 100.00%	
1 records found for UECS2344	
Tue 3/26/19, 11:00 AM practical 3 hour(s)	
PRESENT	30/3/19 4:08:22 PM

Figure 5.17: Attendance records

CHAPTER 6

PROJECT IMPLEMENTATION

6.1 Introduction

This chapter discuss about project implementation on routes hierarchy, web API endpoints, web socket, code generator, Amazon Web Service, Firebase, MongoDB Atlas and process diagram.

6.2 Routes Hierarchy

For both web app has different routes to reach the certain page. The routes that marked with lazy are lazy-loaded. In another words, a lazy-loaded page would only be loaded when the user route to it. This increases the initial load time as not all pages are downloaded.

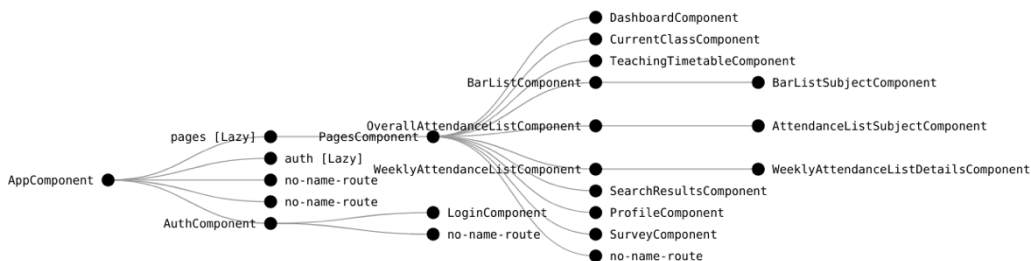


Figure 6.1: Routes hierarchy for lecturer view web app

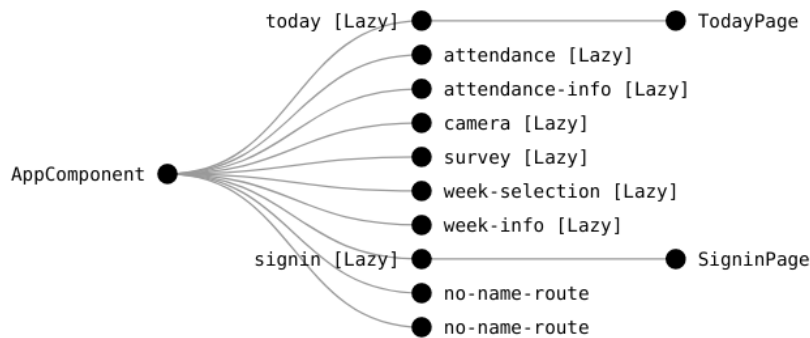


Figure 6.2: Routes hierarchy for student view web app

6.3 Web API Endpoints

The total count of API endpoints are 24 endpoints. Each endpoint serves different purposes.

Table 6.1: List of web API endpoints

Type: Student, Method: GET	
Route	Description
/students	Get all students
/students/{id}	Get a student with ID
/students/{id}/subjects	Get all the subjects of the student
/students/{id}/today	Get the student's today classes
/students/{id}/current	Get the student's current class
/students/{id}/subjects/week/{weekno}	Get the student's subjects by week number
Type: Lecturer, Method: GET	
Route	Description
/lecturers	Get all lecturers
/lecturers/{id}	Get a lecturer with ID
/lecturers/{id}/subjects	Get all the subjects of the lecturer
/lecturers/{id}/today	Get the lecturer's today classes
/lecturers/{id}/current	Get the lecturer's current class
/lecturers/{id}/subjects/week/{weekno}	Get the lecturer's subjects by week number
/lecturers/{id}/timeslot?slotDate={value}&endDate={value}	Get the lecturer's timeslot between two dates
Type: Subjects, Method: GET, POST, PUT, DELETE	
Route	Description
/subjects	Get all subjects
/subjects/{code}	Get a specific subject with subject code
/subjects/{code}/weeks	Get subject info like start and end date, total week, and current week
/subjects/{code}/list?startTime={value}&day={value}&type={value}&date={value}	Get the student list of the subjects between two dates
/subjects/{code}/overview	Get the overview student list of the subjects which includes students'

	attendance percentage so far
/subjects/{code}/barlist	Get the student list who get barred or attendance percentage less than 80% so far
POST /subjects/{code}/slot	Add a new timeslot for the subject
PUT /subjects/{code}/slot	Update a timeslot for the subject
DELETE /subjects/{code}/slot	Delete a timeslot for the subject
Type: Attendance, Method: GET	
Route	Description
/attendance	Get all attendance records
Type: Registration, Method: POST	
Route	Description
/registration?mode=auto	Automatic attendance registration by scanning QR code
/registration?mode=manual	Manual attendance alter/registration

6.4 Web Socket

Web Socket is computer communication protocol, which provides full-duplex connection channel over a TCP connection. Web Socket often used for maintaining stateful connection between client and server. Once the connection between both parties are established, the server can send data to the client and vice versa. This eliminates the needs of unnecessary of calling the web service to get updated data. Web Socket connection is used for the application for sending random code from the server and render as a QR code on the client. Due to the code is generated from the server in short interval, it is better to keep a stateful connection between client and server. When the client issues a close request, the server closes the Web Socket connection.

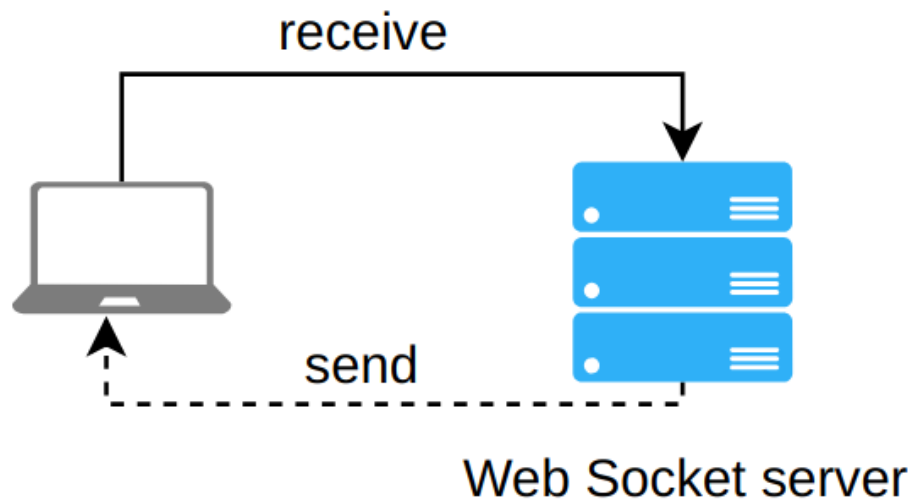


Figure 6.3: Simple diagram of a Web Socket connection over client and server

6.5 Code Generator

Code generator server is used for random and unique code generation and send to client to displayed as QR code.

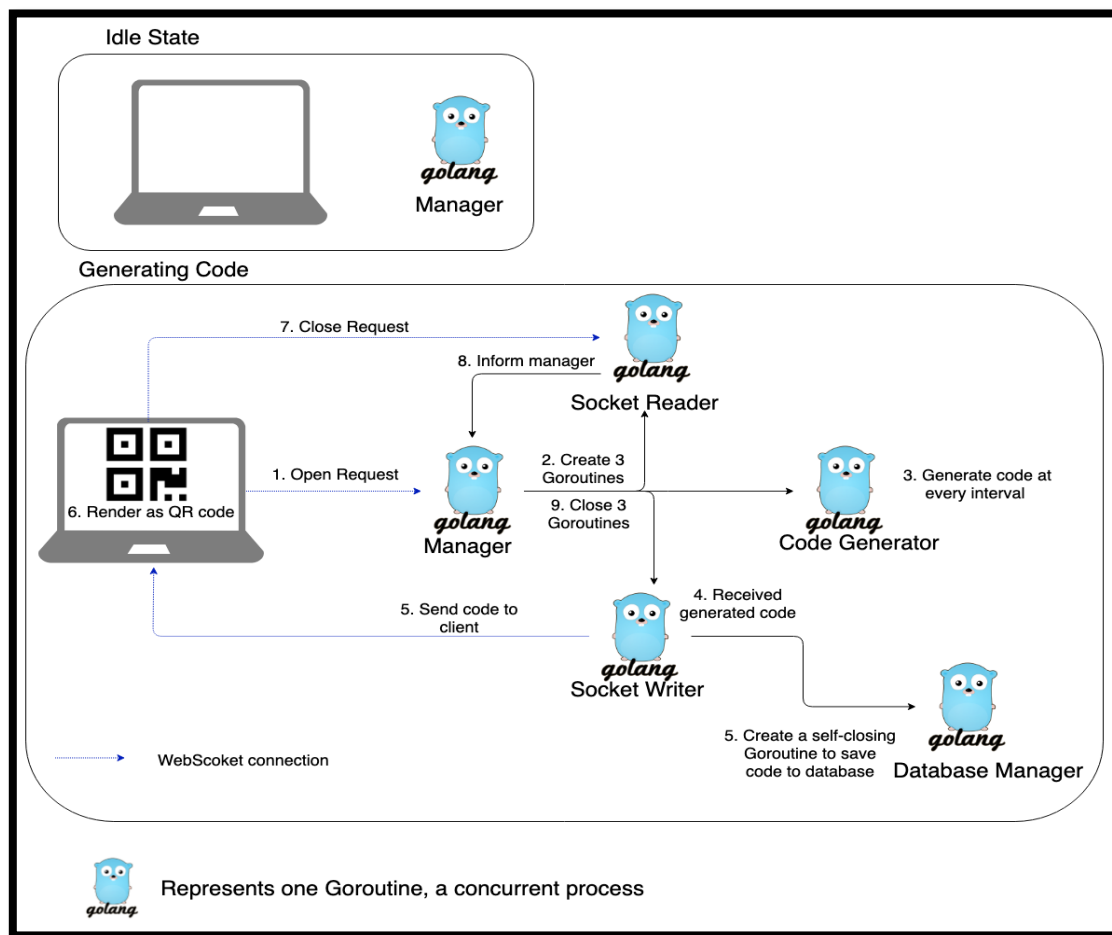


Figure 6.4: Code generator design

Scenario

1. In idle state, only one Goroutine, Manager is on ready state.
2. Client issues an open request to generate QR code
3. The open request is sent to Manager
4. Manager spawns additional 3 Goroutines concurrently, which are Socket Reader, Socket Writer and Code Generator. Each Goroutine is in ready state. Socket Writer waiting input from Code Generator meanwhile Socket Reader waiting input from the client.
5. Code Generator keeps generate code based on the time interval set. The generated code will store in a Channel.
6. Socket Writer reads the input from Channel, and send the generated code to client, and at the same time, it spawns an additional self-closing Goroutine, which is Database Manager, to save the generated code to database. When this operation is completed, the Data Manager will close itself.

7. The steps 5 and 6 keeps on repeating based on the time interval set from client.
8. When the client issues a close request, the Socket Reader will receive the request and inform Manager.
9. Once the Manager has received the close request, it closes all Goroutines, which are Socket Writer, Socket Reader and Code Generator.
10. It backs to only one Goroutine left, which is the Manager, same as the idle state.

6.6 Amazon Web Service

Four services are used on Amazon Web Service. These services serve different purposes.

6.6.1 Cognito

Cognito is used for user authentication on both lecturer and student. Both lecturer and student are in a separate user pool on Cognito.

Currently have 3 active users who are holding lecturer role.

Estimated number of users	3
Required attributes	name
Alias attributes	none
Username attributes	none
Custom attributes	Choose custom attributes...
Minimum password length	7
Password policy	no requirements
User sign ups allowed?	Only administrators can create users

Figure 6.5: Cognito configuration for lecturer view authentication

Currently have 83 active users who are holding student role.

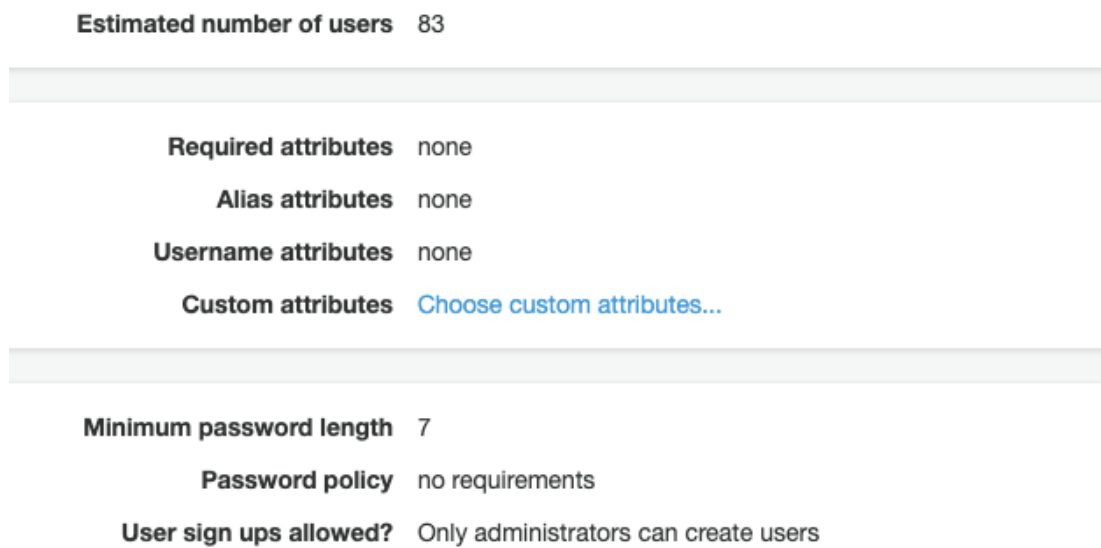


Figure 6.6: Cognito configuration for student view authentication

6.6.2 Elastic Beanstalk

Two Elastic Beanstalk instance are running, one for the web API, the other for the code generator. The code generator requires TCP connection; therefore, it needs an application load balancer. The load balancer opens up a TCP connection and enables auto scaling when the network is max out.

Web API

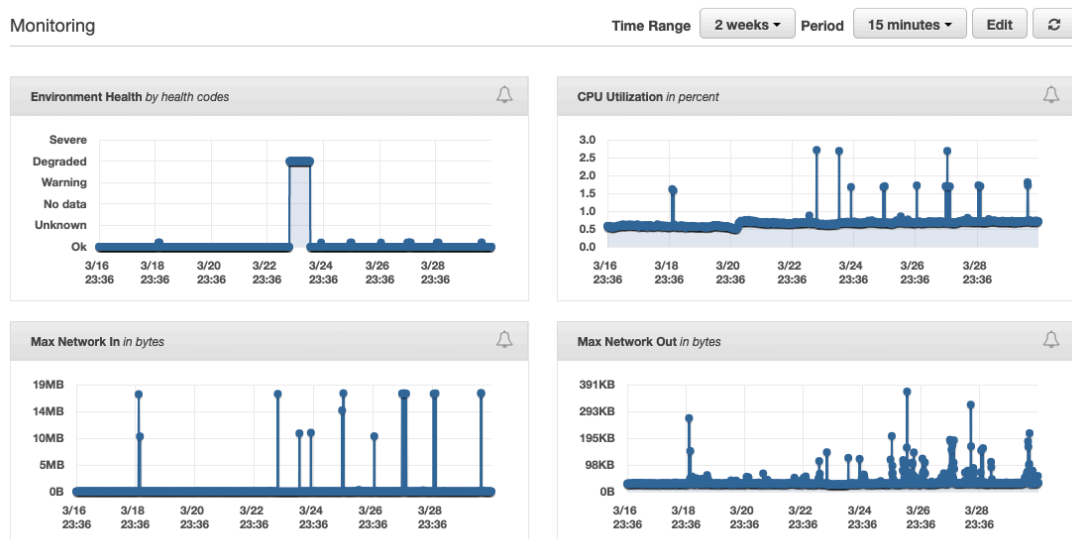


Figure 6.7: The activity of Elastic Beanstalk that serve for web API

Web Socket

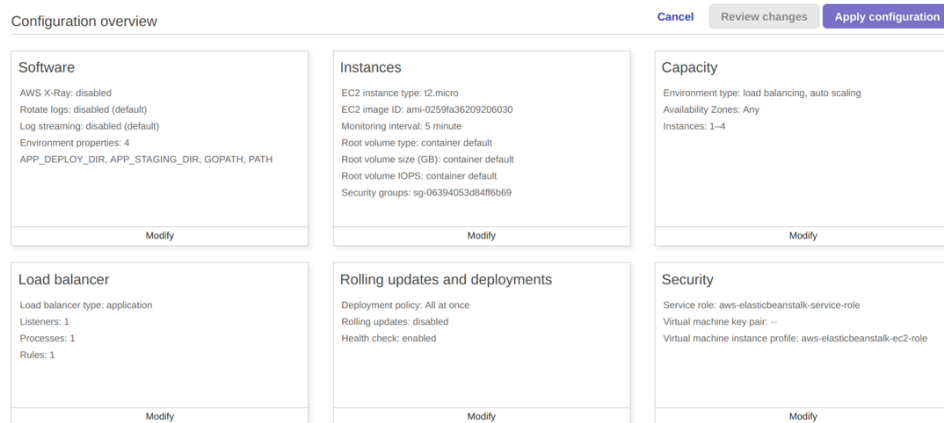


Figure 6.8: Elastic Beanstalk instance configuration with load balancer and auto scaling capacity

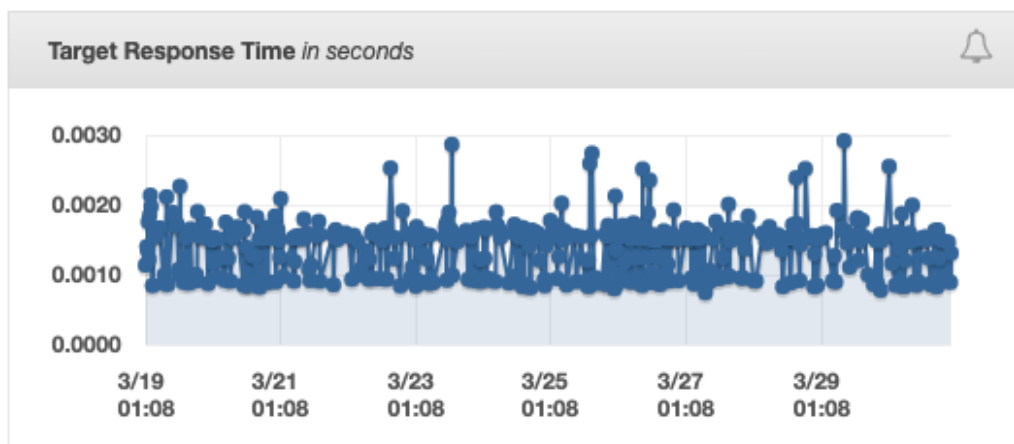


Figure 6.9: The activity of Elastic Beanstalk that show target response time of web socket connection

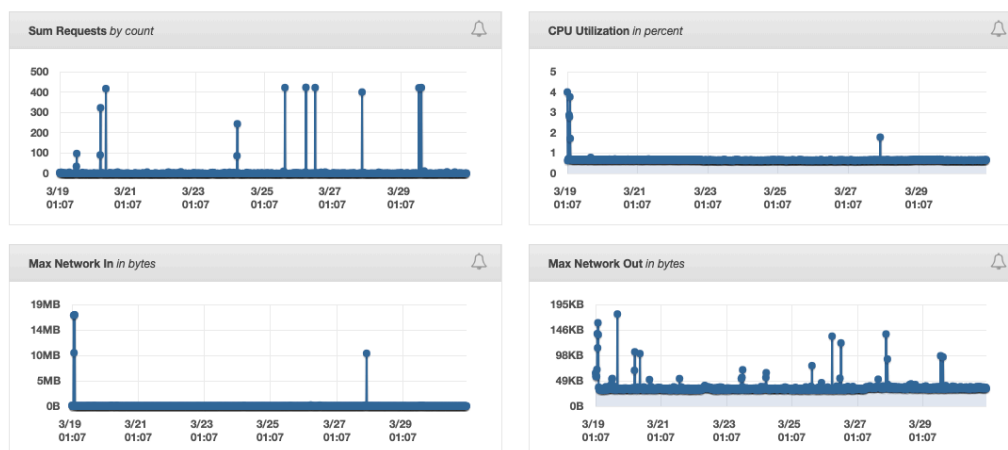


Figure 6.10: The activity of Elastic Beanstalk that serve for web socket connection

Auto Scaling Group

Configure the compute capacity of your environment and Auto Scaling settings to optimize the number of instances used.

Environment type	<input type="text" value="Load balanced"/>
Instances	Min <input type="text" value="1"/> Max <input type="text" value="4"/>
Availability Zones	<input type="text" value="Any"/>
	Number of Availability Zones (AZs) to use.
Placement	<input type="text" value="ap-southeast-1a"/> <input type="text" value="ap-southeast-1b"/> <input type="text" value="ap-southeast-1c"/>
	Specify Availability Zones (AZs) to use.
Scaling cooldown	<input type="text" value="360"/> <input type="text" value="seconds"/>

Scaling triggers

Metric	<input type="text" value="NetworkOut"/>
	Change the metric that is monitored to determine if the environment's capacity is too low or too high.
Statistic	<input type="text" value="Average"/>
	Choose how the metric is interpreted.

Figure 6.11: Elastic Load Balancing configuration

6.6.3 API Gateway

API Gateway provides secure connection to clients and act as a centralised gateway for all the web services.

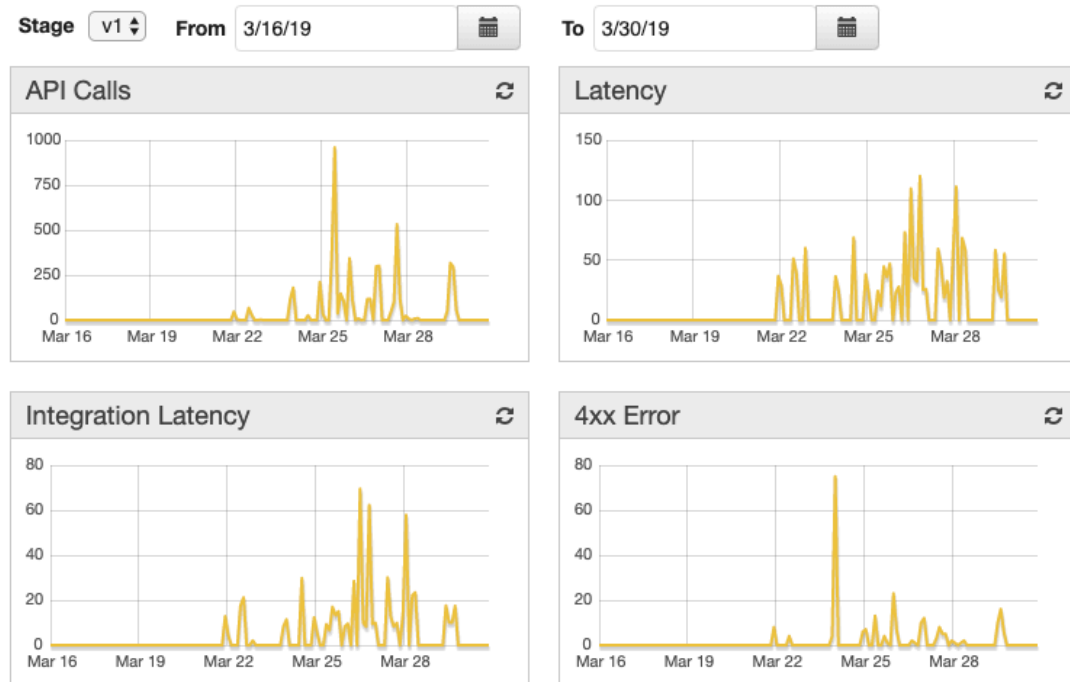


Figure 6.12: Dashboard of API Gateway that show activities

6.6.4 Simple Storage Service

Simple Storage Service (S3) provides reliable static web hosting.

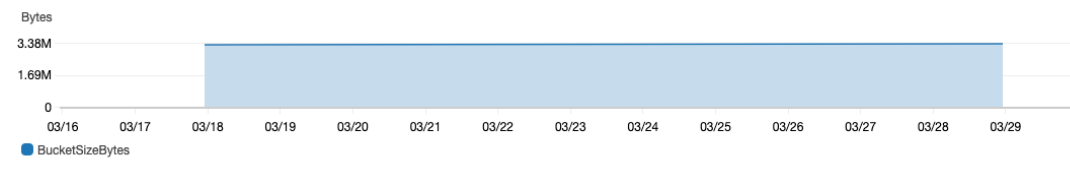


Figure 6.13: The storage used in S3

6.7 Firebase

Firebase provides various of services, but only Hosting service is used.

6.7.1 Firebase Hosting

Firebase Hosting provides static web hosting over HTTPS connection. Secure connection is a need for mobile device to access camera through web browser.

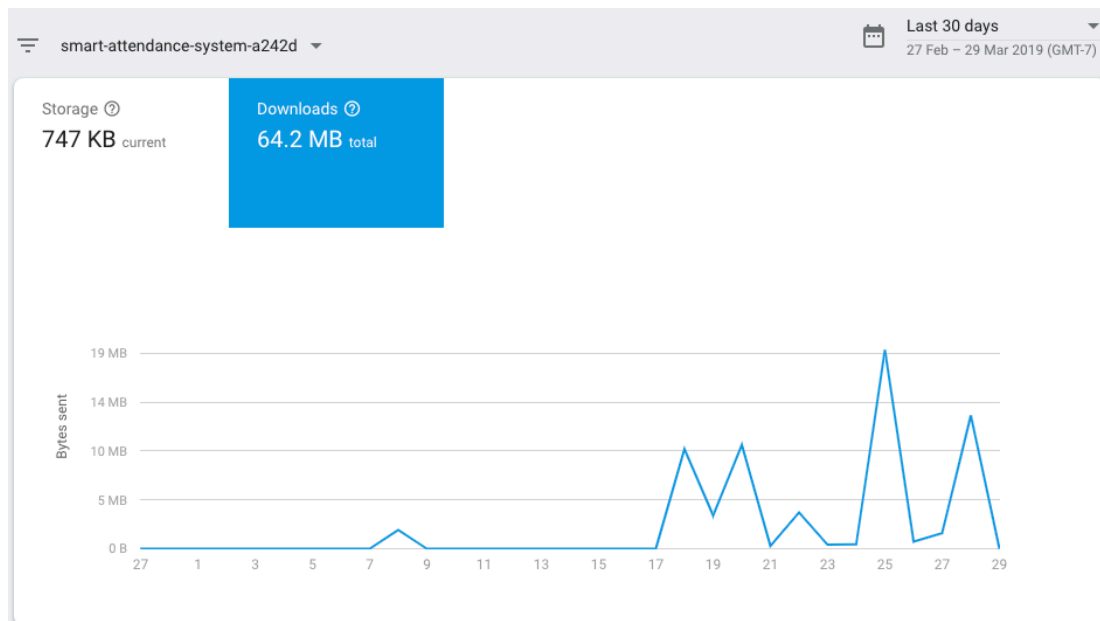


Figure 6.14: Usage of the Firebase Hosting

6.8 MongoDB Atlas

MongoDB Atlas provides Database-as-a-Service, we can use MongoDB without setting up a database server, and run on public clouds.

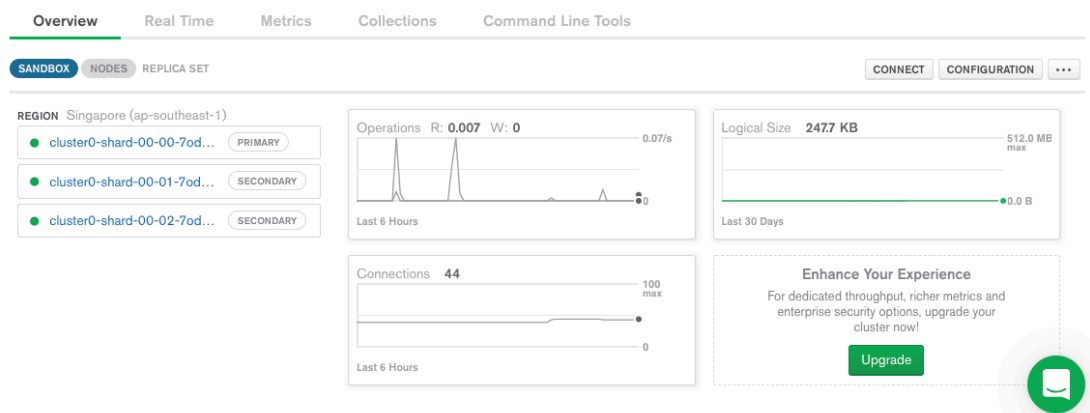


Figure 6.15: Overview of MongoDB running on 3 nodes, which are 1 primary node and 2 secondary nodes.

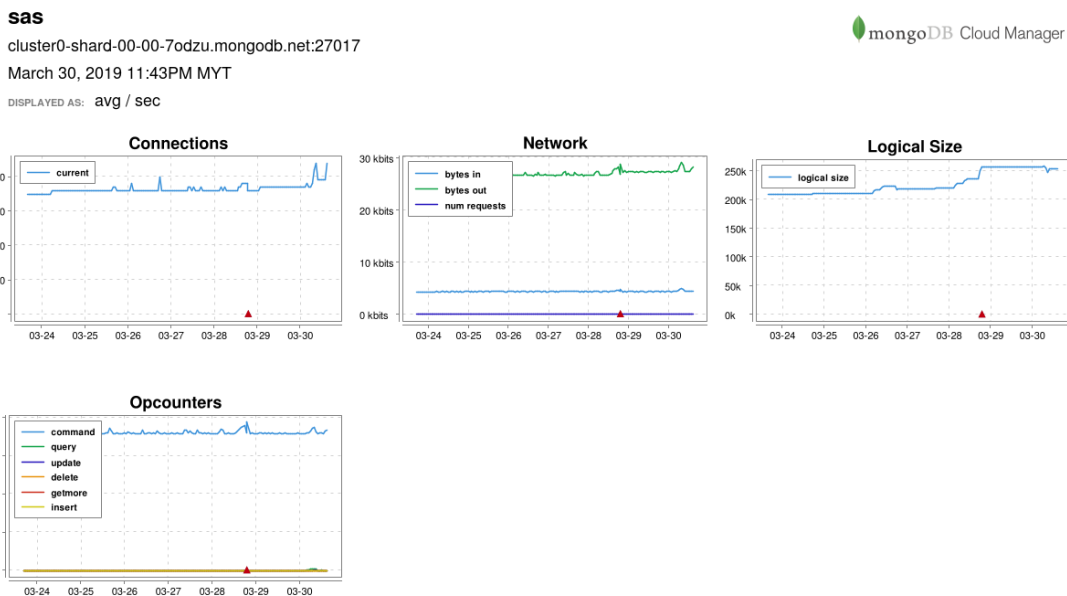


Figure 6.16: Overview of activities on primary node

6.9 Process Flow Diagram

Process flow diagram shows the flow of services involved during a process.

6.9.1 User authentication

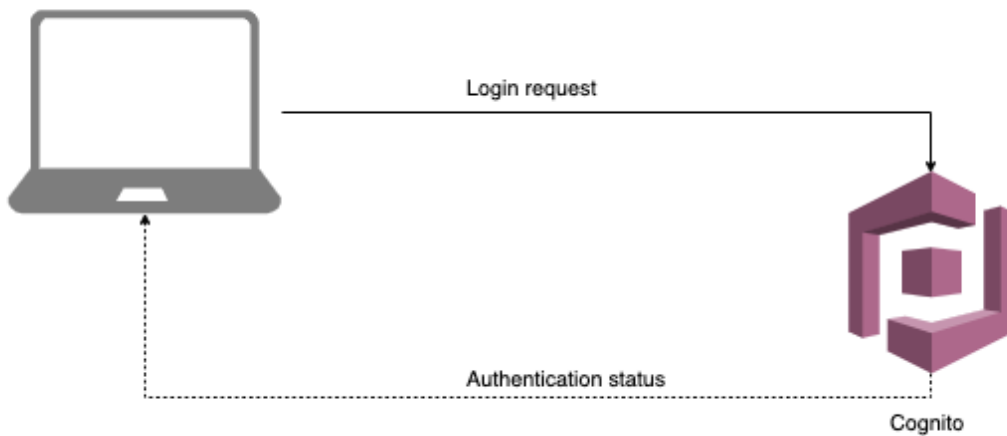


Figure 6.17: User authentication process flow

6.9.2 API Calls

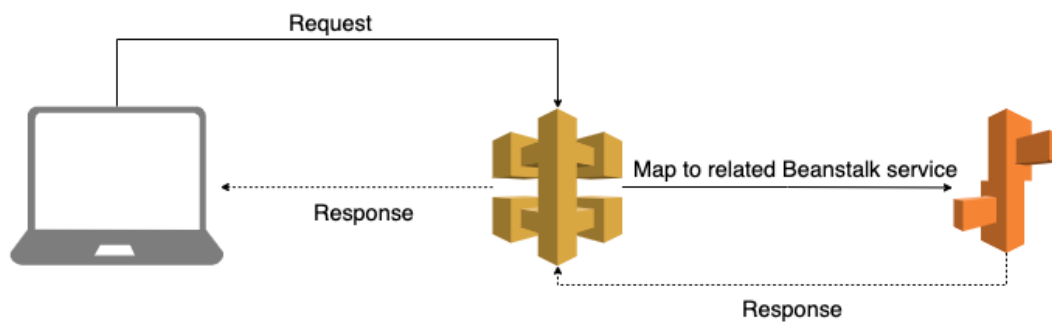


Figure 6.18: Web API calls process flow

6.9.3 QR Code Generation

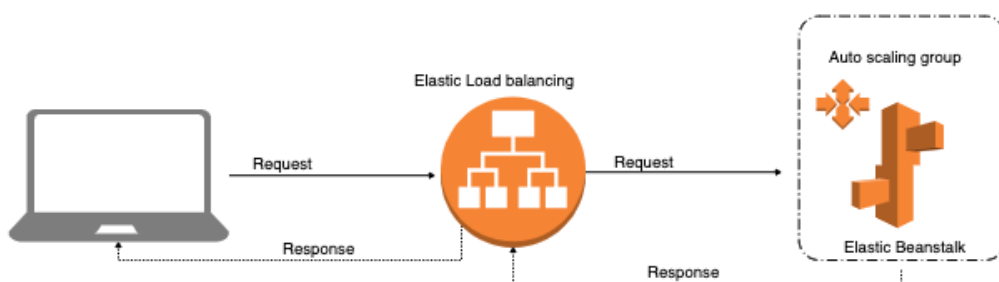


Figure 6.19: QR code generation process flow

CHAPTER 7

PROJECT TESTING

7.1 Introduction

This chapter discuss about unit testing, static code analysis, code quality review, application audit and user acceptance testing.

7.2 Unit Testing

Unit testing has been conducted with all repositories. Jasmine is used as a testing framework for frontend application. It is a JavaScript testing framework that does not required a DOM. Karma is a testing tool which spawns a web server that executes source code against test code for each browser connected. Jasmine and Karma are used together for testing for Angular and Ionic.

WebUI repository

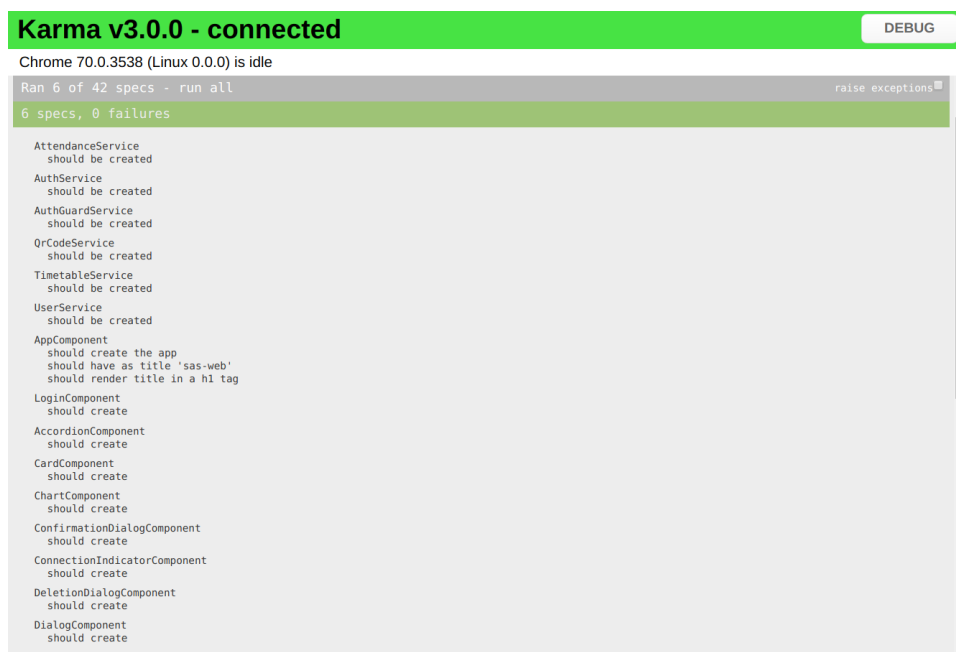


Figure 7.1: Test results on browser

```

Chrome 70.0.3538 (Linux 0.0.0): Executed 6 of 42 (skipped 36) SUCCESS (0.15 secs / 0.151 secs)
TOTAL: 6 SUCCESS

```

Figure 7.2: Test results on console

Mobile repository

The screenshot shows the Karma v3.1.4 interface in a browser. At the top, it says "Karma v3.1.4 - connected" and "Chrome 70.0.3538 (Linux 0.0.0) is idle". Below that, it indicates "Jasmine 2.99.0" and "finished in 0.116s". A progress bar shows 5 out of 19 specs completed. The main area displays "Ran 5 of 19 specs - run all" and "5 specs, 0 failures". A list of test cases follows, including AuthService, ClassService, AuthGuardService, RegistrationService, UserService, AppComponent, AttendanceInfoPage, AttendancePage, CameraPage, ClassListComponent, SigninPage, SurveyPage, and TodayPage, each with a "should create" or "should have" assertion.

Figure 7.3: Test results on browser

```

Chrome 70.0.3538 (Linux 0.0.0): Executed 5 of 19 (skipped 14) SUCCESS (0.087 secs / 0.098 secs)
TOTAL: 5 SUCCESS
TOTAL: 5 SUCCESS

```

Figure 7.4: Test results on console

Backend repository

```

=== RUN   TestGetAllStudents
--- PASS: TestGetAllStudents (0.00s)
=== RUN   TestGetStudentByID
--- PASS: TestGetStudentByID (0.00s)
=== RUN   TestGetStudentSubjects
--- PASS: TestGetStudentSubjects (0.00s)
=== RUN   TestGetAllLecturer
--- PASS: TestGetAllLecturer (0.00s)
=== RUN   TestGetLecturerByID
--- PASS: TestGetLecturerByID (0.00s)
=== RUN   TestGetLecturerSubjects
--- PASS: TestGetLecturerSubjects (0.00s)
=== RUN   TestGetLecturerSubjectsByWeek
--- PASS: TestGetLecturerSubjectsByWeek (0.00s)
=== RUN   TestGetSubjectOverviewList
--- PASS: TestGetSubjectOverviewList (0.00s)
=== RUN   TestGetSubjectBarList
--- PASS: TestGetSubjectBarList (0.00s)
=== RUN   TestGetAllSubjects
--- PASS: TestGetAllSubjects (0.01s)
=== RUN   TestGetSubjectByCode
--- PASS: TestGetSubjectByCode (0.00s)
=== RUN   TestGetSubjectWithWeek
--- PASS: TestGetSubjectWithWeek (0.00s)
=== RUN   TestGetAllAttendance
--- PASS: TestGetAllAttendance (0.00s)
PASS
ok      smart-attendance-backend/api  0.039s

```

Figure 7.5: Unit test result for backend repository

7.3 User Interface Testing

Table 7.1: UI test case

Behaviour	Expected	Actual
Type URL to enter a page without login	System should redirect to login page	System redirects user to login page
Invalid credentials to login	System should show error message	System shows error message
Valid credentials to login	System should redirect to authenticated page	System directs user authenticated page
Current class page only shows the current class	Only at exact time, the current class page should	The current class page shows the current class

information on the exact time of the class slot	only show the current class information	information during the exact time of the class
QR code configuration with auto closing dialog and refresh rate	System should close the dialog after 1 minutes with 5 seconds of QR code refresh rate	System closed the dialog after 1 minutes with and shows QR code with 5 seconds of refresh rate

7.4 Static Code Analysis

Static code analysis had been conducted throughout the whole development process. TSLint is used for TypeScript project for both lecturer and student view repositories.

TSLint is constantly and automatically analyse the code and prompt errors if the code written does not comply to the rules or style. This ensures that the code follows the best practice of the rules or style.

7.5 Code Quality Review

Code quality review had done automatically for all the codebase with Codebeat during the development. Full GPA is 4.0. The figures below shown the results for each repository.

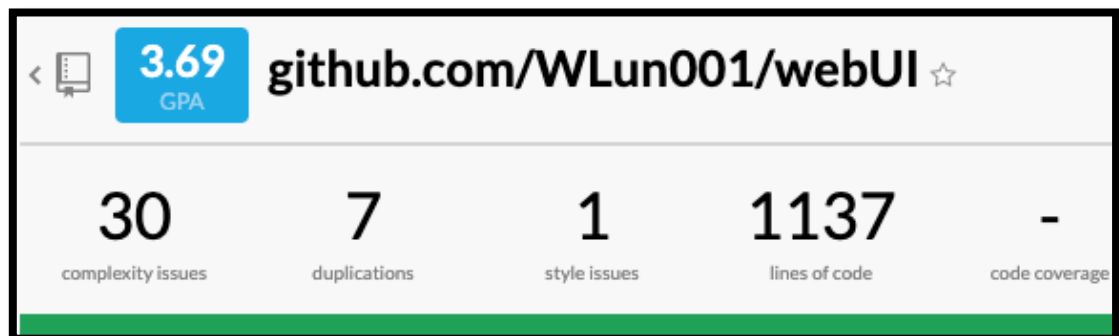


Figure 7.6: Code review scoring for lecturer view repository

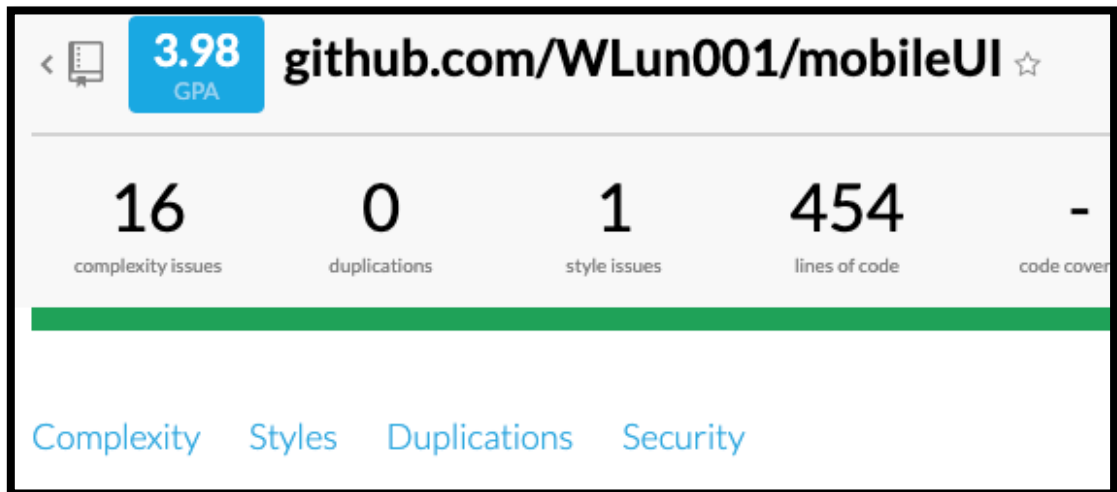


Figure 7.7: Code review scoring for student view repository

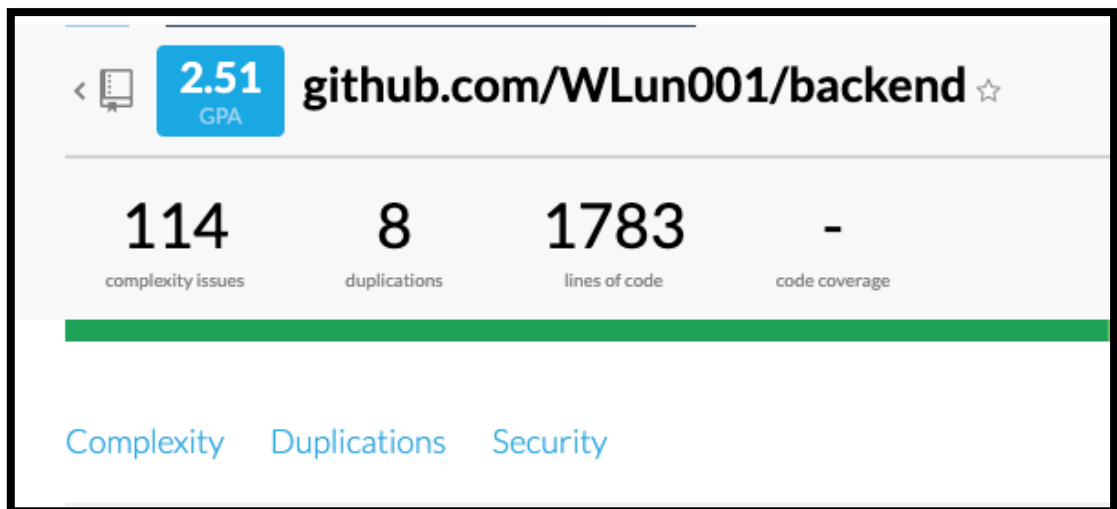


Figure 7.8: Code review scoring for backend repository

7.6 Application Audit

Google Lighthouse is an open-source tool for audit web pages. It is used for audit web app for both lecturer and student view

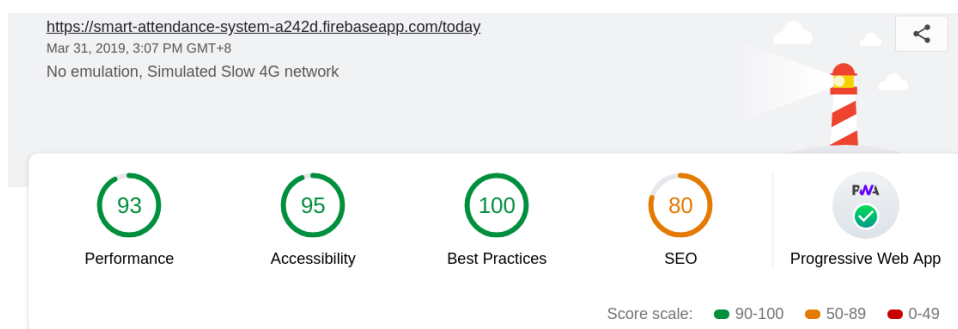


Figure 7.9: Lighthouse audit report for lecturer view web app

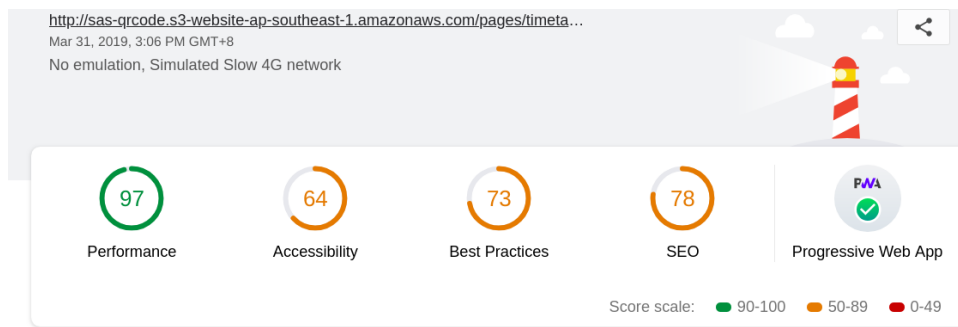


Figure 7.10: Lighthouse audit report for student view web app

Full lighthouse reports are attached on Appendix E

7.7 User Acceptance Testing

User Acceptance Testing had been conducted for two classes, practical and lecture class. Total number students involved in the testing are 62 students, however only 26 of them response to the survey. Students are giving positive feedback on the system user interface, flow, meanwhile negative feedback on the completeness of the application. Due to the camera is not zoomable, some of the students found it difficult to scan from far from the projected screen.



Figure 7.11: Conducting UAT on a practical class

Besides, 1 lecturer was involved and received 1 response to the survey. The respondent pointed out the convenience and accuracy of the system, with some unnecessary selection of inputs on UI. Further improvement can be made to the system.

The survey overall statistic is attached at appendix F.

CHAPTER 8

CONCLUSION AND DISCUSSION

8.1 Introduction

This chapter discuss about limitation of the project, future improvement and recommendation and conclusion.

8.2 Limitation

The project comes with few limitations as stated in the following

- No secure connection on code generator server as secure connection requires a verified domain to get an SSL license.
- No HTTPS connection on lecturer web app as web service that the web app consume are not secure.

8.3 Improvement and Recommendation

There are several improvements and recommendations to be considered in the future development.

- Administrator view for manage timetable, users, and view resources consumed.
- Zoomable camera on student view web app.
- Develop as full-fledged progressive web app on all web apps.
- Develop in-deep data analysis feature to further analyse students' attendance records.

8.4 Challenges

The challenges when developing the projects are listed as follows:

- Required a lot of self-learning to pick up libraries, frameworks and cloud services.
- Less similar implementations or solutions to refer to when designing the application.

8.5 Conclusion

In conclusion, taking attendance with QR code is the cheapest and adaptable options among all of the solutions. It does not require infrastructure changes to adapt it. With the popularity of the smartphone and internet accessibility, it can widely use in universities. With the automatically refresh QR code, cheating on attendance is become even difficult. Besides, it eliminates a lot of lecturers' effort on managing students' attendance records. The project objectives were achieved.

References

- Balcoh, N.K., Yousaf, M.H., Ahmad, W. and Baig, M.I., 2012. *Algorithm for efficient attendance management: Face recognition based approach*. [online]. Available at: <https://s3.amazonaws.com/academia.edu.documents/43434909/Algorithm_for_Efficient_Attendance_Manag20160306-14773-ibratp.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1534243864&Signature=SMAusM2H1GvO01umwplgFCm3v0U%3D&response-content-disposition=inline%3B%20filename%3DAlgorithm_for_Efficient_Attendance_Manag.pdf> [Accessed 10 July 2018]
- Carter, R.A., Antón, A.I., Dagnino, A. and Williams, L., 2001. *Evolving beyond requirements creep: a risk-based evolutionary prototyping model*. [online]. Available at: <<https://ieeexplore.ieee.org/abstract/document/948548>> [Accessed 13 July 2018]
- Cynthia, H. and Andy P. 2018. *AWS vs. Azure vs. Google: Cloud Comparison*. [online] Available at <<https://www.datamation.com/cloud-computing/aws-vs-azure-vs-google-cloud-comparison.html>> [Accessed 3 April 2019]
- Elankovan, V. 2018. *Malaysian University Revealed to Have Epic New Way of Taking Attendance*. [online] WORLD OF BUZZ. Available at: <<https://www.worldofbuzz.com/malaysian-university-revealed-epic-new-way-taking-attendance>> [Accessed 23 Jun. 2018].
- Hsarena. 2017. *Unimas first in the country to use Quick Response Code*. [online] Available at: <<http://www.news.unimas.my/in-the-news/eventdetail/4094/-/unimas-first-in-the-country-to-use-quick-response-code>> [Accessed 27 Jun. 2018].
- Masalha, F. and Hirzallah, N., 2014. *A students attendance system using QR code*. [online]. Available at: <<https://pdfs.semanticscholar.org/288f/0459675d41e2d3bbb8b6b65bc927ffe57262.pdf>> [Accessed 17 July 2018]
- Mikowski, M.S. and Powell, J.C., 2013. *Single page web applications*. [online]. Available at: <<http://deals.manningpublications.com/spa.pdf>> [Accessed 19 July 2018]
- Koukia, A. 2018. *Going Single Page App or Traditional Web App*. [online] Available at: <<https://koukia.ca/going-single-page-app-or-traditional-web-app-cedb10041b50>> [Accessed 23 Jun. 2018].
- Saraswat, C. and Kumar, A., 2010. *An efficient automatic attendance system using fingerprint verification technique*. [online]. Available at: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.163.4880&rep=rep1&type=pdf>> [Accessed at 15 July 2018]

StudyMalaysia. 2016. *List of Universities in Malaysia*. [online]. Available at: <<https://www.studymalaysia.com/education/top-stories/list-of-universities-in-malaysia>>[Accessed 27 July. 2018].

The Hans India. 2018. *Bluetooth based Attendance Tracking System Designed By MVJ College of Engineering Students* [online] Available at: <<http://www.thehansindia.com/posts/index/Technology/2018-03-19/Bluetooth-based-attendance-tracking-system-designed-by-MVJ-College-of-Engineering-students--/367620>> [Accessed 29 July 2018]

TheStar. 2018. *Mobile cellular penetration reaches 131.8%*. [online] Available at: <<https://www.thestar.com.my/business/business-news/2018/02/14/mobile-cellular-penetration-reaches-1318/>>[Accessed 27 July 2018]

Wood, S., Call, R., Bowyer, C. and Hennah, S. 2018. *Skill Up 2018*. Packt

APPENDICES

APPENDIX A: Survey Question

Student Attendance System Survey

Thank you for agreeing to take part in this important survey. I am Yong Wei Lun, Software Engineering student from University Tunku Abdul Rahman (Sungai Long campus). I am currently working on my Final Year Project which is developing smart attendance system using QR Code.. I would like to conduct this survey with you to obtain more information about the existing and similar system in your University or College. This survey will only take less than 3 minutes. For your information, the answer you provided will be taken into consideration for my project and the information you provided might be included in my project report. Your help is much appreciated and try your best to complete this survey form. Thank you!

* Required

1. **What is the name of your University/College (full name) ***

2. **Which attendance system does your University/College using? ***

Mark only one oval.

- Automated attendance system
- Sign attendance on attendance sheets
- Other: _____

3. **What do you think about current attendance system? ***

Mark only one oval.

- Cumbersome
- Is fine
- Other: _____

4. **Do you think it's easy to cheat on attendance? ***

Mark only one oval.

- Yes
- No
- Maybe

5. **Do you think attendance system should be automated? ***

Mark only one oval.

- Yes
- No
- Maybe

6. Do you think an automated attendance system will increase the productivity of lecturers and reduce human error? *

Mark only one oval.

- Yes
- No
- Maybe
- Other: _____

7. Do you think which tracking technology is the best to prevent fake attendance? *

Mark only one oval.

- Using QR Code in class, will be reset after short interval
- Polling registers students as present in class by requiring them to answer a polling question or enter a special code on their laptops or phones.
- Using geolocation
- Using Biometric
- Using student ID reader
- Other: _____

8. Do you think which attendance tracking technology is the most cost effective? *

Mark only one oval.

- Using QR Code
- Polling
- Geolocation
- Biometric reader
- Student ID reader
- Other: _____

APPENDIX B: Results of Survey

Student Attendance System Survey

20 responses

What is the name of your University/College (full name)

20 responses

Universiti Sains Malaysia

London School of Economics

Kampus Kesihatan Universiti Sains Malaysia

Multimedia University

Sunway College University

The One Academy

University Malaysia Terengganu

University Utara Malaysia

UCSI UNIVERISTY

Nilai Univeristy

Nanyang Technological University

Universiti Malaya

Loma Linda University

University Kebangsaan Malaysia

Infrastructure University Kuala Lumpur

Tunku Abdul Rahman University College

Asia Pacific University

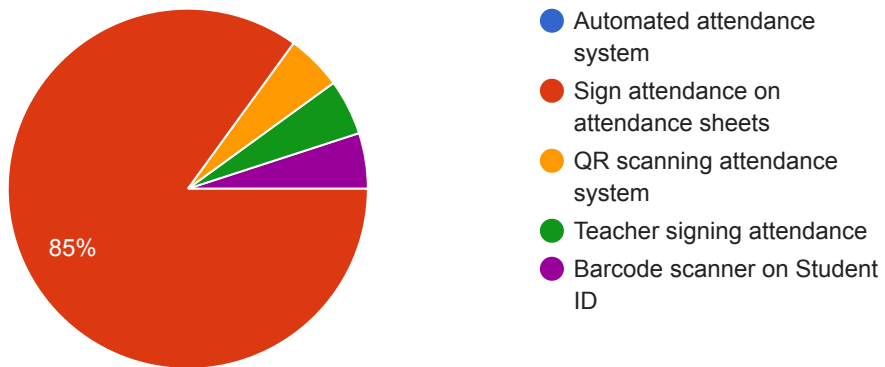
Newcastle University

HELP University

YPC International College

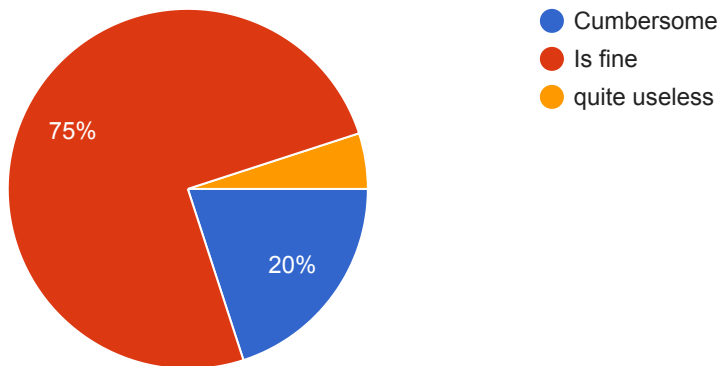
Which attendance system does your University/College using?

20 responses



What do you think about current attendance system?

20 responses



Do you think it's easy to cheat on attendance?

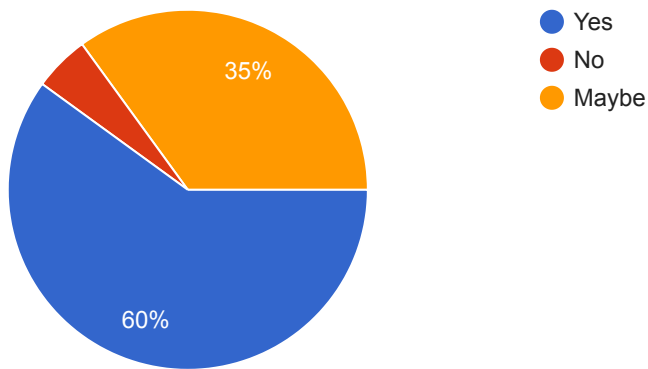
20 responses



● Yes

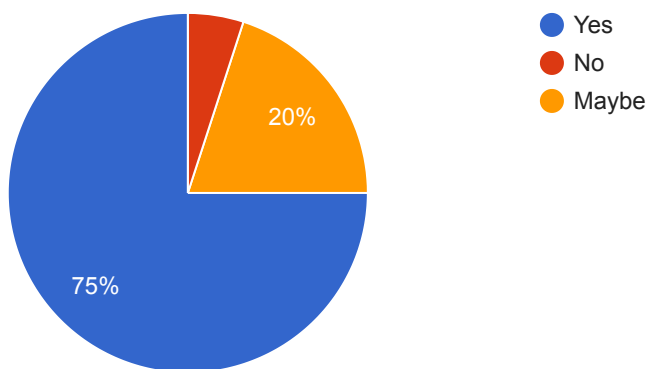
Do you think attendance system should be automated?

20 responses



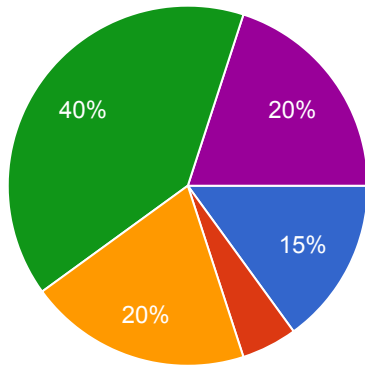
Do you think an automated attendance system will increase the productivity of lecturers and reduce human error?

20 responses



Do you think which tracking technology is the best to prevent fake attendance?

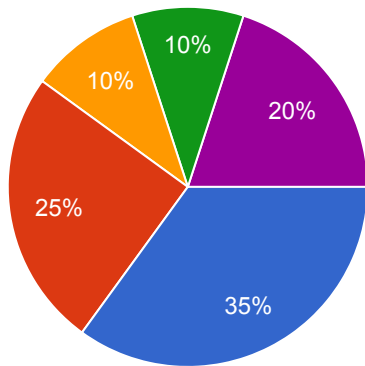
20 responses



- Using QR Code in class, will be reset after short interval
- Polling registers students as present in class by requiring them to answer a polling question or enter a special code on their laptops or ph...
- Using geolocation
- Using Biometric
- Using student ID reader

Do you think which attendance tracking technology is the most cost effective?

20 responses



- Using QR Code
- Polling
- Geolocation
- Biometric reader
- Student ID reader

This content is neither created nor endorsed by Google. Report Abuse - Terms of Service - Additional Terms

Google Forms

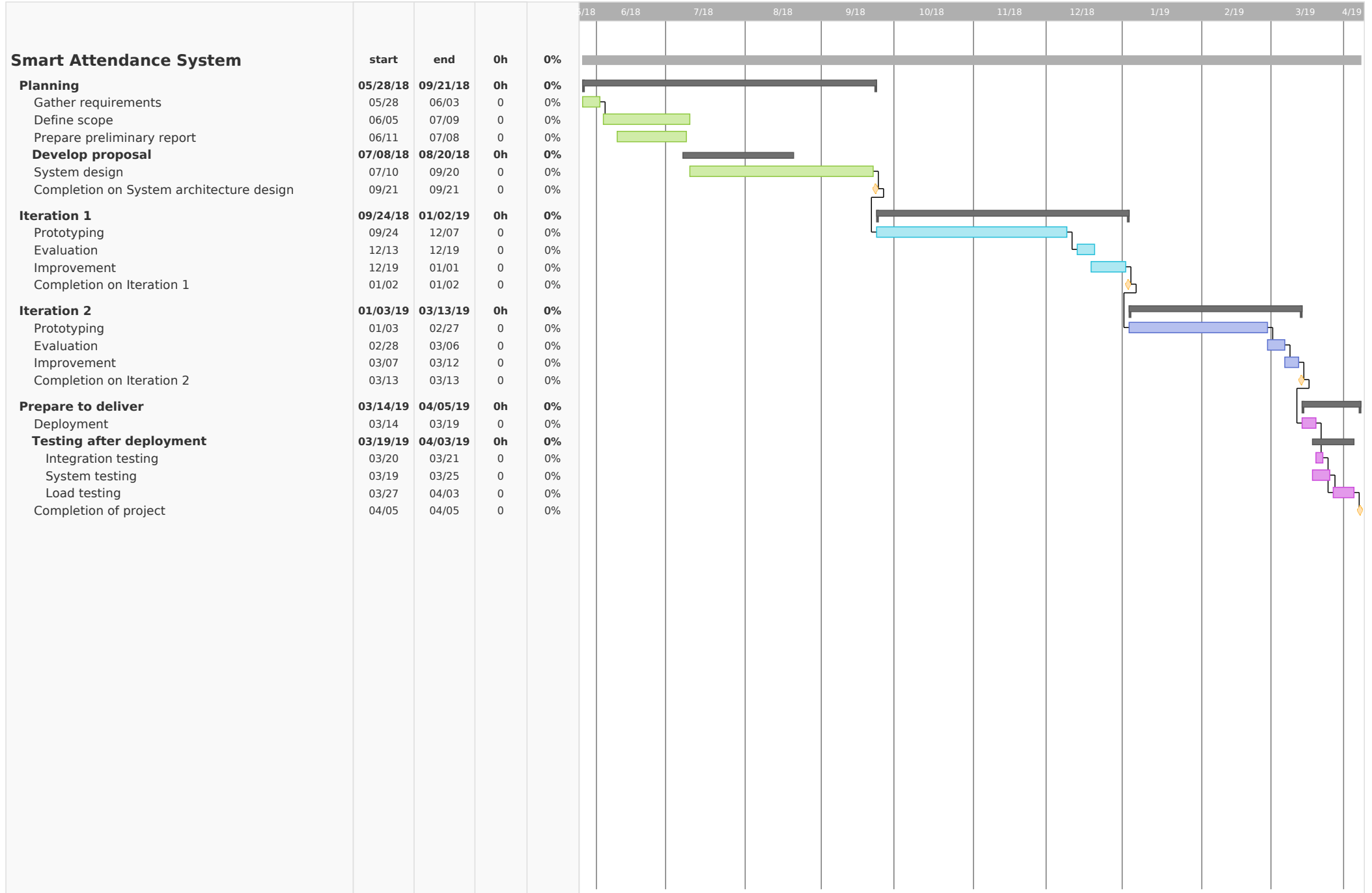
APPENDIX C: Work Breakdown Structure

Smart Attendance System

Planning	0%		Start	Due	Assigned
Gather requirements	<input type="text" value="0%"/>	<input type="text"/>	May 28, 2018	Jun 3, 2018	
Define scope	<input type="text" value="0%"/>	<input type="text"/>	Jun 5, 2018	Jul 9, 2018	
Prepare preliminary report	<input type="text" value="0%"/>	<input type="text"/>	Jun 11, 2018	Jul 8, 2018	
Develop proposal	0%				
System design	<input type="text" value="0%"/>	<input type="text"/>	Jul 10, 2018	Sep 20, 2018	
Completion on System architecture design			Sep 21, 2018	Sep 21, 2018	
Iteration 1	0%		Start	Due	Assigned
Prototyping	<input type="text" value="0%"/>	<input type="text"/>	Sep 24, 2018	Dec 7, 2018	
Evaluation	<input type="text" value="0%"/>	<input type="text"/>	Dec 13, 2018	Dec 19, 2018	
Improvement	<input type="text" value="0%"/>	<input type="text"/>	Dec 19, 2018	Jan 1, 2019	
Completion on Iteration 1			Jan 2, 2019	Jan 2, 2019	
Iteration 2	0%		Start	Due	Assigned
Prototyping	<input type="text" value="0%"/>	<input type="text"/>	Jan 3, 2019	Feb 27, 2019	
Evaluation	<input type="text" value="0%"/>	<input type="text"/>	Feb 28, 2019	Mar 6, 2019	
Improvement	<input type="text" value="0%"/>	<input type="text"/>	Mar 7, 2019	Mar 12, 2019	
Completion on Iteration 2			Mar 13, 2019	Mar 13, 2019	
Prepare to deliver	0%		Start	Due	Assigned
Deployment	<input type="text" value="0%"/>	<input type="text"/>	Mar 14, 2019	Mar 19, 2019	
Testing after deployment	0%				
Integration testing	<input type="text" value="0%"/>	<input type="text"/>	Mar 20, 2019	Mar 21, 2019	
System testing	<input type="text" value="0%"/>	<input type="text"/>	Mar 19, 2019	Mar 25, 2019	
Load testing	<input type="text" value="0%"/>	<input type="text"/>	Mar 27, 2019	Apr 3, 2019	
Completion of project			Apr 5, 2019	Apr 5, 2019	

[View in Gantt](#)

APPENDIX D: Gantt Chart



APPENDIX E: Lighthouse Audit



Performance



Accessibility



Best Practices



SEO



Progressive Web App

Score scale: 90-100 50-89 0-49

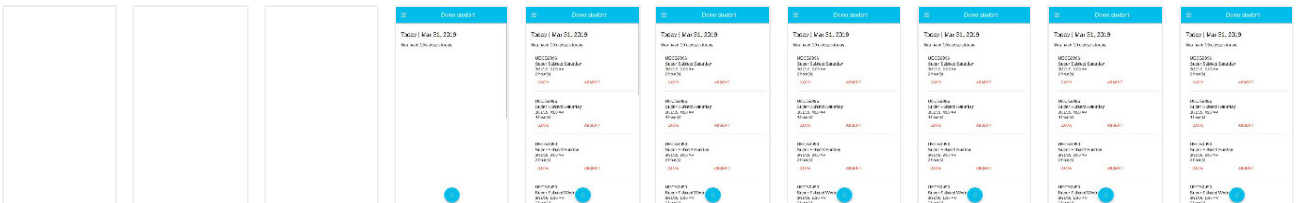
Performance



Metrics

First Contentful Paint	1,920 ms	✓	First Meaningful Paint	1,960 ms	✓
Speed Index	2,770 ms	✓	First CPU Idle	3,520 ms	✓
Time to Interactive	3,760 ms	✓	Estimated Input Latency	262 ms	⚠

Values are estimated and may vary.



Diagnostics

More information about the performance of your application.

- 1 Has significant main thread work 3,740 ms ⓘ ▾
 - 2 JavaScript boot-up time is too high 2,280 ms ⓘ ▾
 - 3 Critical Request Chains 5 chains found ▾
 - 4 User Timing marks and measures 30 user timings ▾
- ✓ Passed audits 18 audits ▾

Progressive Web App

These checks validate the aspects of a Progressive Web App, as specified by the baseline [PWA Checklist](#).



1	Page load is fast enough on 3G	✓	▼
2	Does not respond with a 200 when offline	⚠	▼
3	User will not be prompted to Install the Web App Failures: No manifest was fetched, Site does not register a service worker.	⚠	▼
4	Uses HTTPS	✓	▼
5	Redirects HTTP traffic to HTTPS	✓	▼
6	Has a <code><meta name="viewport"></code> tag with width or initial-scale	✓	▼
7	Does not register a service worker	⚠	▼
8	Does not provide fallback content when JavaScript is not available The page body should render some content if its scripts are not available.	⚠	▼
9	Is not configured for a custom splash screen Failures: No manifest was fetched.	⚠	▼
10	Address bar does not match brand colors Failures: No manifest was fetched, No <code><meta name="theme-color"></code> tag found.	⚠	▼
11	Content is sized correctly for the viewport	✓	▼
12	The short_name won't be truncated on the homescreen	✓	▼

🔍 **Additional items to manually check** 3 audits ▼

Accessibility

These checks highlight opportunities to [improve the accessibility of your web app](#). Only a subset of accessibility issues can be automatically detected so manual testing is also encouraged.



Meta Tags Used Properly

These are opportunities to improve the user experience of your site.

1	<code>[user-scalable="no"]</code> is used in the <code><meta name="viewport"></code> element or the <code>[maximum-scale]</code> attribute is less than 5.	⚠	▼
---	--	---	---

🔍 **Additional items to manually check** 10 audits ▼

✓ **Passed audits** 13 audits ▼

⊖ **Not applicable** 21 audits ▼

Best Practices



✓ Passed audits

15 audits ▾

SEO

These checks ensure that your page is optimized for search engine results ranking. There are additional factors Lighthouse does not check that may affect your search ranking. [Learn more](#).



Content Best Practices

Format your HTML in a way that enables crawlers to better understand your app's content.

1 Document does not have a meta description



Crawling and Indexing

To appear in search results, crawlers need access to your app.

2 robots.txt is not valid

29 errors found ▾

🔍 Additional items to manually check

2 audits ▾

✓ Passed audits

8 audits ▾

⊖ Not applicable

1 audits ▾

Runtime settings

- **URL:** `https://smart-attendance-system-a242d.firebaseio.com/today`
- **Fetch time:** Mar 31, 2019, 3:07 PM GMT+8
- **Device:** No emulation
- **Network throttling:** 150 ms TCP RTT, 1,638.4 Kbps throughput (Simulated)
- **CPU throttling:** 4x slowdown (Simulated)
- **User agent (host):** Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.102 Safari/537.36



Performance



Accessibility



Best Practices



SEO



Progressive Web App

Score scale: 90-100 50-89 0-49

Performance



Metrics

First Contentful Paint	650 ms	✓	First Meaningful Paint	930 ms	✓
Speed Index	1,690 ms	✓	First CPU Idle	3,160 ms	✓
Time to Interactive	3,230 ms	✓	Estimated Input Latency	248 ms	⚠

Values are estimated and may vary.



Diagnostics

More information about the performance of your application.

- Has significant main thread work 3,700 ms ⓘ ▾
- JavaScript boot-up time is too high 2,600 ms ⓘ ▾
- Critical Request Chains 2 chains found ▾
- User Timing marks and measures 26 user timings ▾

✓ Passed audits 18 audits ▾

Progressive Web App

These checks validate the aspects of a Progressive Web App, as specified by the baseline [PWA Checklist](#).



1	Page load is fast enough on 3G	-	
2	Does not respond with a 200 when offline	⚠	▼
3	User will not be prompted to Install the Web App	⚠	▼
	Failures: No manifest was fetched, Site does not register a service worker.		
4	Does not use HTTPS	9 insecure requests found	⚠ ▼
5	Does not redirect HTTP traffic to HTTPS	⚠	▼
6	Has a <meta name="viewport"> tag with width or initial-scale	✅	▼
7	Does not register a service worker	⚠	▼
8	Does not provide fallback content when JavaScript is not available	⚠	▼
	The page body should render some content if its scripts are not available.		
9	Is not configured for a custom splash screen	⚠	▼
	Failures: No manifest was fetched.		
10	Address bar does not match brand colors	⚠	▼
	Failures: No manifest was fetched, No <meta name="theme-color"> tag found.		
11	Content is not sized correctly for the viewport	⚠	▼
	The viewport size is 958px, whereas the window size is 1477px.		
12	The short_name won't be truncated on the homescreen	✅	▼

🔍 **Additional items to manually check** 3 audits ▼

Accessibility

These checks highlight opportunities to [improve the accessibility of your web app](#). Only a subset of accessibility issues can be automatically detected so manual testing is also encouraged.



Elements Have Discernible Names

These are opportunities to improve the semantics of the controls in your application. This may enhance the experience for users of assistive technology, like a screen reader.

1	Buttons do not have an accessible name	⚠	▼
2	Links do not have a discernible name	⚠	▼

Color Contrast Is Satisfactory

These are opportunities to improve the legibility of your content.

3	Background and foreground colors do not have a sufficient contrast ratio.	⚠	▼
---	---	---	---



- 🔍 **Additional items to manually check** 10 audits
- ✓ **Passed audits** 12 audits ▾
- ⊖ **Not applicable** 20 audits ▾

Best Practices



- 1 Does not use HTTPS 9 insecure requests found ⚠️ ▾
- 2 Does not use HTTP/2 for all of its resources 9 requests not served via HTTP/2 ⚠️ ▾
- 3 Links to cross-origin destinations are unsafe ⚠️ ▾
- 4 Browser errors were logged to the console ⚠️ ▾

- ✓ **Passed audits** 11 audits ▾

SEO

These checks ensure that your page is optimized for search engine results ranking. There are additional factors Lighthouse does not check that may affect your search ranking. [Learn more](#).



Content Best Practices

Format your HTML in a way that enables crawlers to better understand your app's content.

- 1 Document does not have a meta description ⚠️ ▾

Crawling and Indexing

To appear in search results, crawlers need access to your app.

- 2 Page has unsuccessful HTTP status code 404 ⚠️ ▾

- 🔍 **Additional items to manually check** 2 audits ▾
- ✓ **Passed audits** 7 audits ▾
- ⊖ **Not applicable** 2 audits ▾

Runtime settings

- **URL:** <http://sas-qr-code.s3-website-ap-southeast-1.amazonaws.com/pages/timetable>

- **Fetch time:** Mar 31, 2019, 3:06 PM GMT+8
- **Device:** No emulation
- **Network throttling:** 150 ms TCP RTT, 1,638.4 Kbps throughput (Simulated)
- **CPU throttling:** 4x slowdown (Simulated)
- **User agent (host):** Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.102 Safari/537.36

Generated by **Lighthouse** 3.0.3 | [File an issue](#)

APPENDIX F: User Satisfaction Survey

User Satisfaction Survey for Attendance System (Student)

My name is Yong Wei Lun, final year student in Software Engineering. Thanks for testing out my final year project, Smart Attendance System using QR code. Kindly fill in the survey form. Your co-operation is high appreciated.

* Required

1. I think that I would like to use this web app for attendance registration *

Mark only one oval.

- Yes
 No
 Maybe

2. I found the web app unnecessarily complex. *

Mark only one oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

3. I thought the web app was easy to use *

Mark only one oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

4. I think that I would need the support of a technical person to be able to use this web app.

Mark only one oval.

- Yes
 No
 Maybe

5. I found this web app was easily moved through without a lot of backtracking or data re-entry. *

Mark only one oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

6. I thought there was too much inconsistency in this web app *

Mark only one oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

7. I would imagine that most people would learn to use this web app very quickly. *

Mark only one oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

8. I found the web app very awkward to use. *

Mark only one oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

9. I felt very confident using the web app *

Mark only one oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strong Agree

10. I needed to learn a lot of things before I could get going with this web app *

Mark only one oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

11. What did you like best about the web app?

12. What did you like least about the web app?

13. If you were to describe this web app to a colleague in a sentence or two, what would you say?

14. Do you have any other final comments or questions?

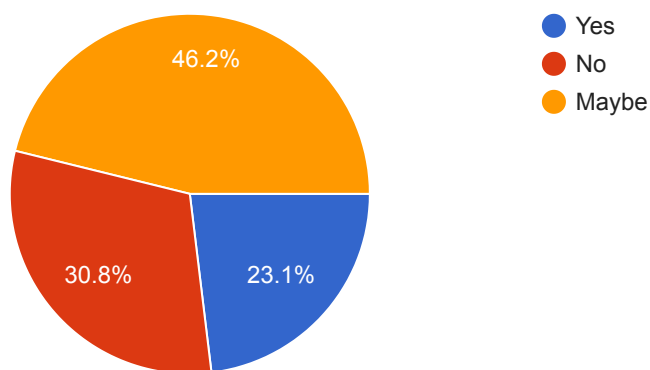


User Satisfaction Survey for Attendance System (Student)

26 responses

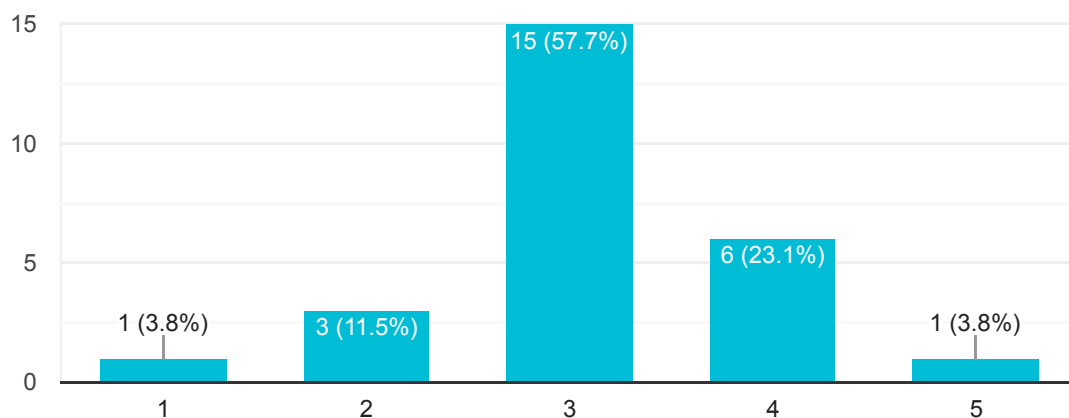
I think that I would like to use this web app for attendance registration

26 responses



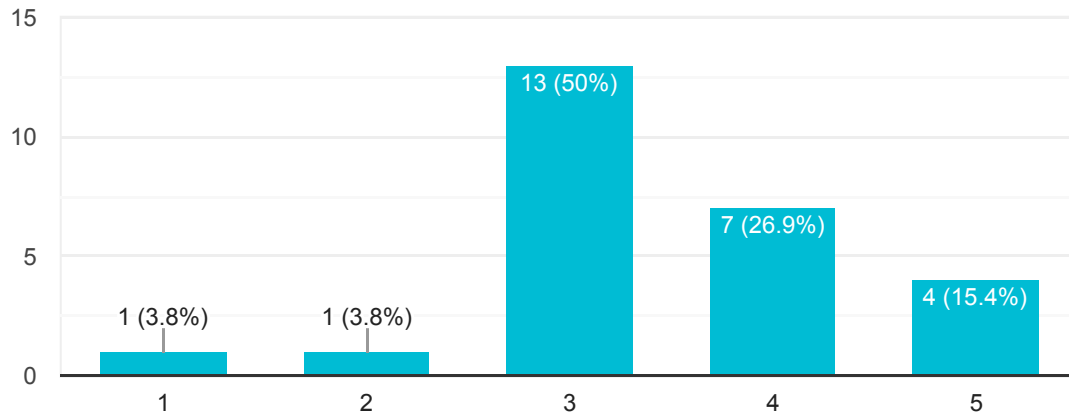
I found the web app unnecessarily complex.

26 responses



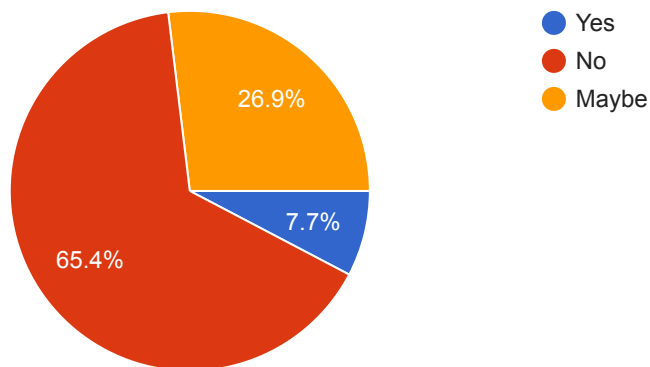
I thought the web app was easy to use

26 responses



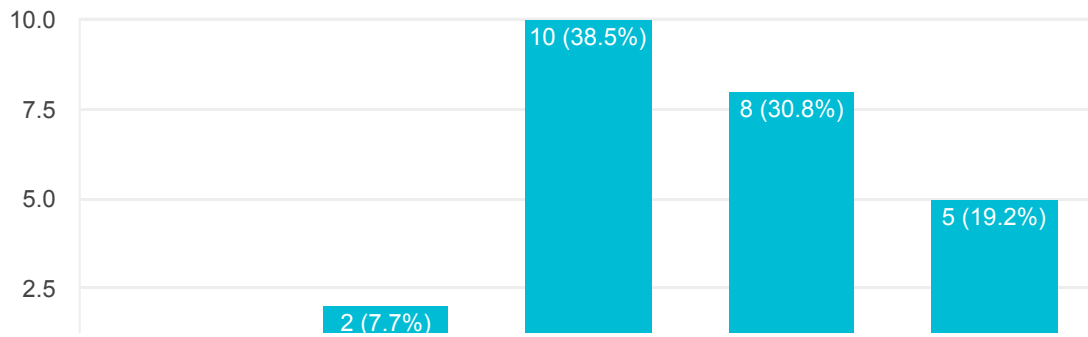
I think that I would need the support of a technical person to be able to use this web app.

26 responses



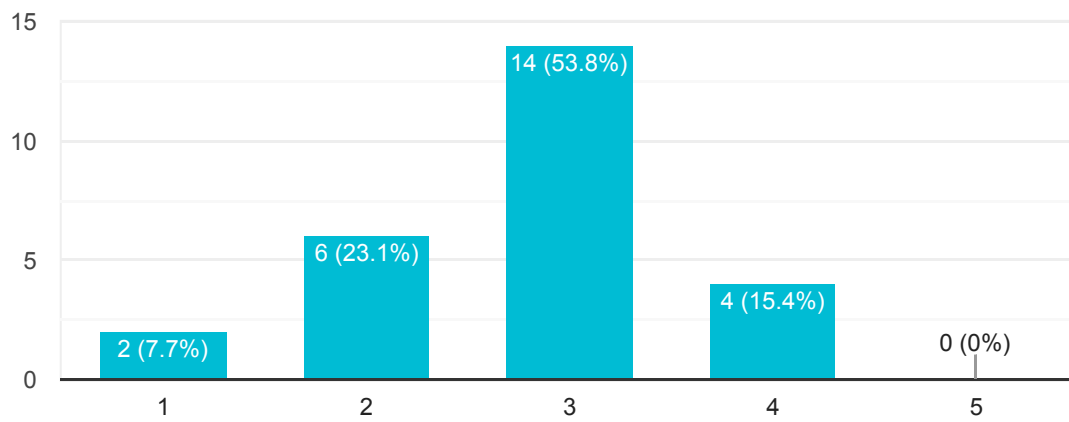
I found this web app was easily moved through without a lot of backtracking or data re-entry.

26 responses



I thought there was too much inconsistency in this web app

26 responses



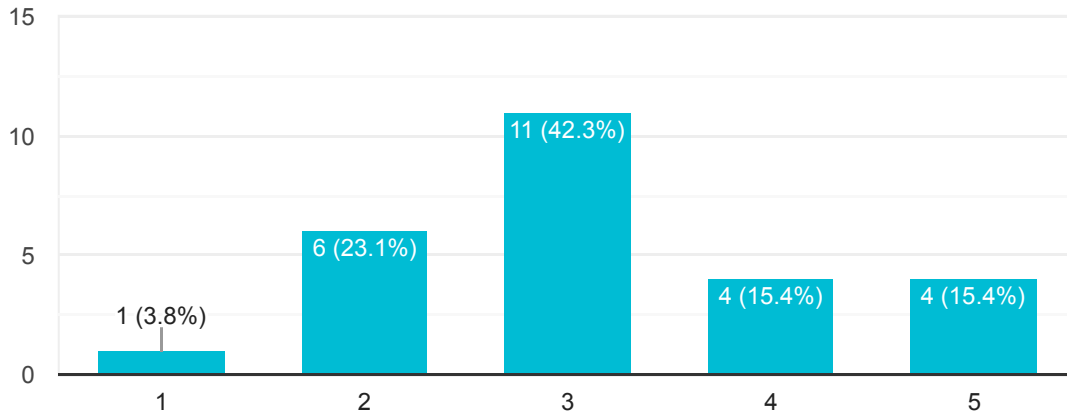
I would imagine that most people would learn to use this web app very quickly.

26 responses

15

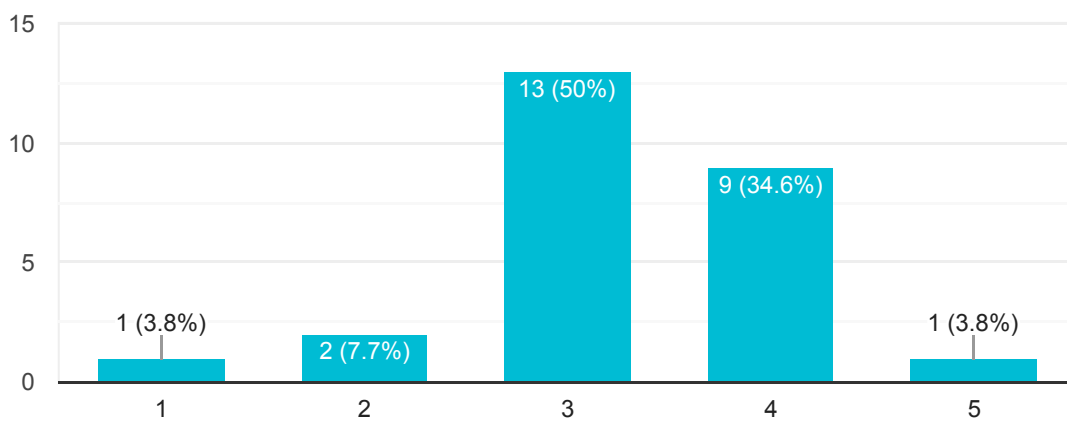
I found the web app very awkward to use.

26 responses



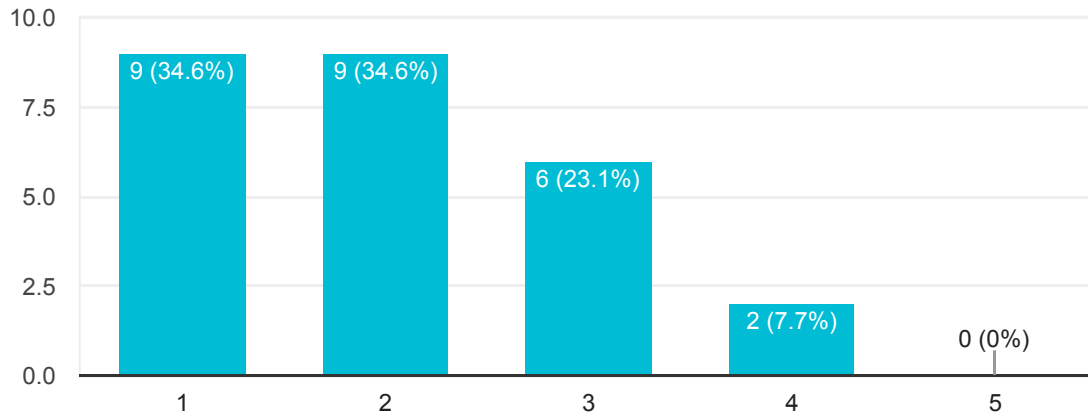
I felt very confident using the web app

26 responses



I needed to learn a lot of things before I could get going with this web app

26 responses



What did you like best about the web app?

19 responses

Simple

Nothing at all

Nothing

Convenience on signing attendance

Easy to use

Environment friendly as students need not sign on paper, reducing paper wastage

Easy

Nice user interface.

The fact that we do not have to sign for attendance

High tech

Ease of usage.

User friendly and easy to use.

Using camera to scan QR code for attendance

Qr code

Replace manual attendsnde

Scan barcode for attendance

Qr

User interface

What did you like least about the web app?

21 responses

No

Everything there is to it

Weird position of the header

Need internet connection for signing

Default setting of front camera

Cannot help friends sign

Camera cannot zoom, students sit far from the screen will have to move nearer to the screen just to get the qr code. Also preferable if can fix a qr code since the qr changes once its being capture, causing the increase of time consumption

Camera

Refresh rate of qr code

The default camera is the front camera. The circle should include on the middle bottom should include an icon to indicate camera. If invalid, the message should close automatically after a period of time.

The QR code expires after one person scans it

Not so convenience

Troublesome to scan qr code.

Camare

Try to allow usera for scanning QRcode to take attendace, instead of asking user to enter URL

Attendance page need to be refreshed after scanning, address too long

User interface can be improved (after scan ckde, the done cannot be seen)

Simple interface

Need Internet, need to type in the page

User interface

If you were to describe this web app to a colleague in a sentence or two, what would you say?

15 responses

Easy to use

Nothing good

Well thought but implementation can be done better

Nice website

Convenient

Still need improvement

Easy , efficient and effective to record attendees

Good.

Just try it

Attendance system

Good for record attendance.

It's fine

Nice app

Do you have any other final comments or questions?

14 responses

No

No thanks ^^

Instead of retying manually, maybe retry it automatically and after few fail attempts only require retry manually

Nope

Well done

Camera cannot zoom cannot focus and a bit hard to scan

Can be used with same ip adress, thus still able to help others

I do not think this can solve the problem where students are 'signing' for their friends as one person can easily log in to multiple accounts using one device.

The problem of scanning the qr code must be solved before proper use.

Maybe cannot help people to sign attendance if this system is success

Hope that the camera can have enlarge function.

Got few pages back button missing got function but no icon

Qr code refreshing interval is too short. Imagining it keeps on refreshing and not every phone has high specification. 1.Lecturer tend to spend time on giving students to scan.

2.What if people came late or have to leave early, what would happened? It wouldn't be fair to everyone. Barred chance will be increasing than usual.

This content is neither created nor endorsed by Google. [Report Abuse](#) - [Terms of Service](#)

Google Forms

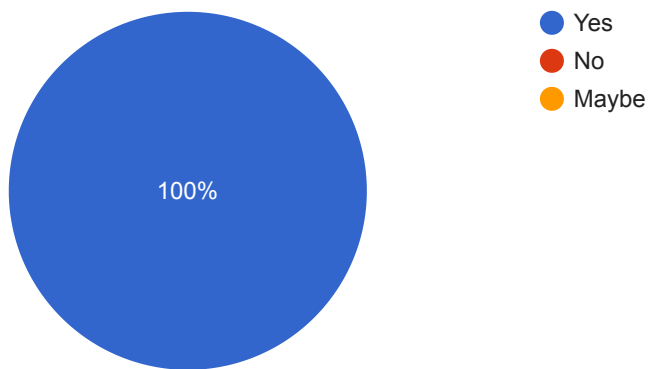


User Satisfaction Survey for Attendance System (Lecturer)

1 response

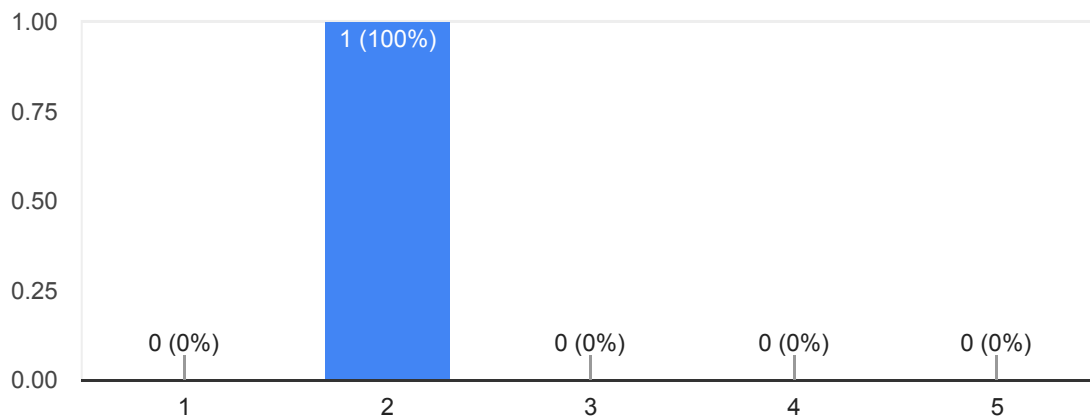
I think that I would like to use this web app for attendance registration

1 response



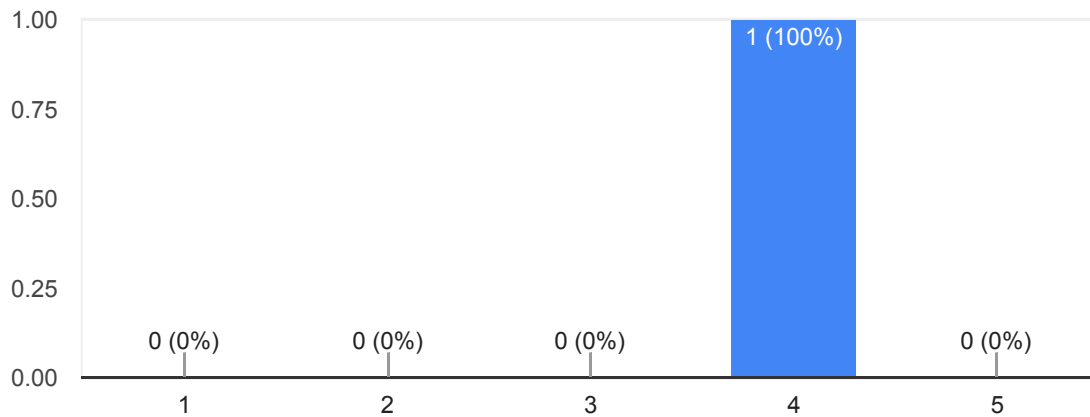
I found the web app unnecessarily complex.

1 response



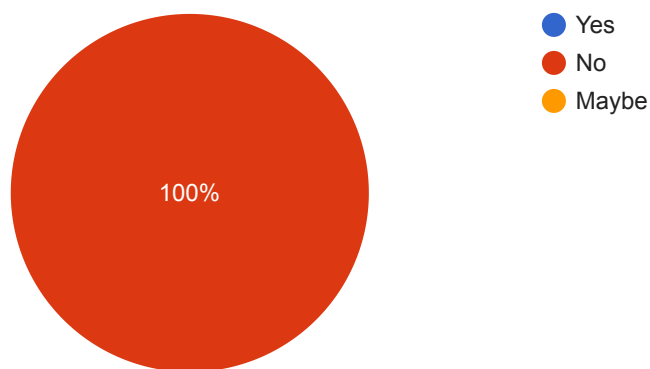
I thought the web app was easy to use

1 response



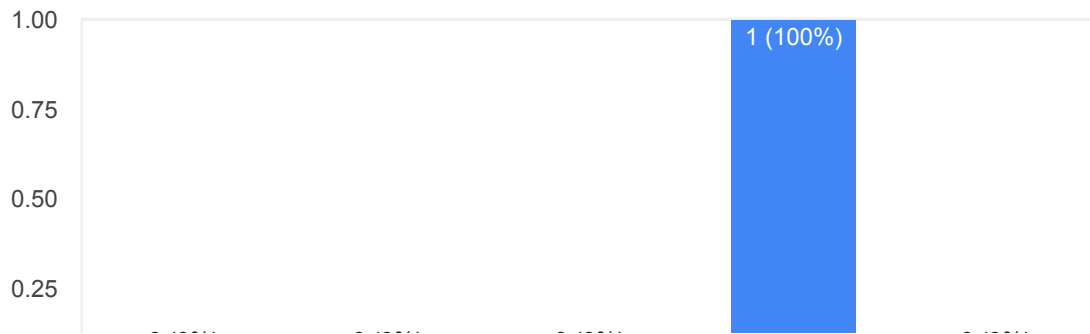
I think that I would need the support of a technical person to be able to use this web app.

1 response



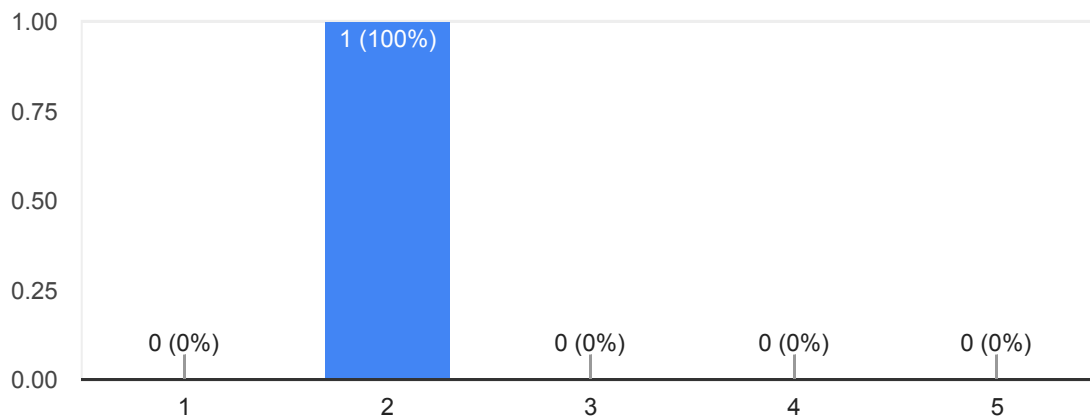
I found this web app was easily moved through without a lot of backtracking or data re-entry.

1 response



I thought there was too much inconsistency in this web app

1 response



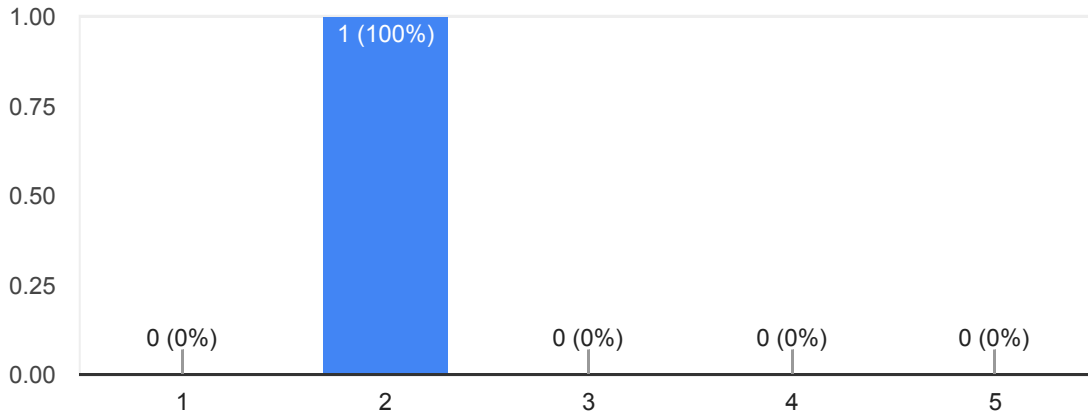
I would imagine that most people would learn to use this web app very quickly.

1 response



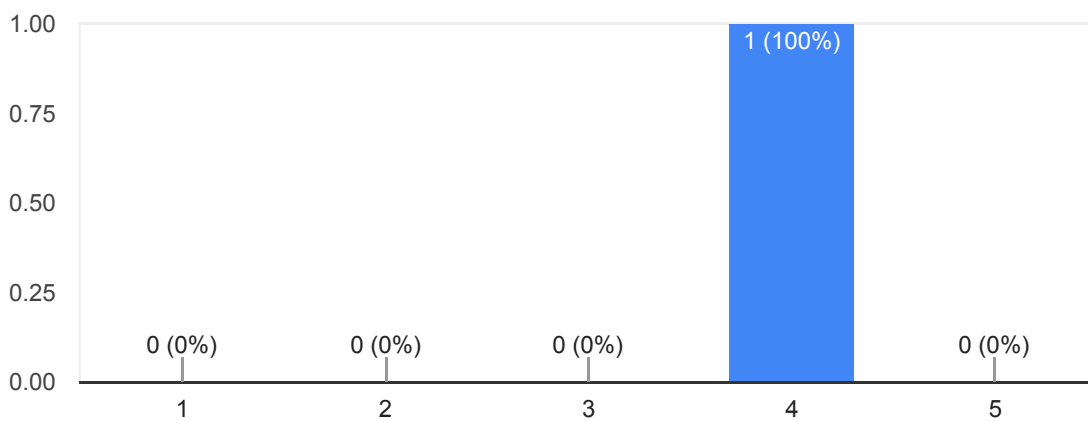
I found the web app very awkward to use.

1 response



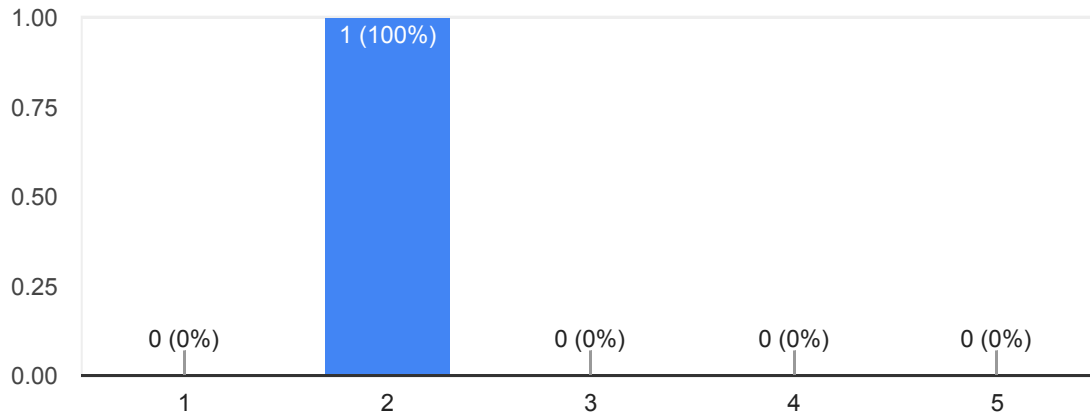
I felt very confident using the web app

1 response



I needed to learn a lot of things before I could get going with this web app

1 response



What did you like best about the web app?

1 response

convenient, accurate

What did you like least about the web app?

1 response

some unnecessary selection of input widget (eg checkbox and scroll value in the configuration)

If you were to describe this web app to a colleague in a sentence or two, what would you say?

1 response

good to have, cheapest option

Do you have any other final comments or questions?

1 response

further analysis of student attendance , eg list of students, achieved < 80% attendance to be alerted and able to see the attendance pattern in other subjects

This content is neither created nor endorsed by Google. [Report Abuse](#) - [Terms of Service](#)

Google Forms

Appendix G: Supervisor's comments

Project Title:	Smart Attendance System using QR code
Student Name:	YONG WEI LUN
Supervisor Name:	Hoo Meei Hao
Moderator:	Ooi Ean Huat

Key Assessment for Project Proposal	Supervisor Comments/Remarks	Moderator Comments/Remarks
Project Description - Is the problem or need to be addressed clearly presented? - Is the proposed approach or solution clearly presented and justified?	Include the short survey (observation of current practices) and its result in the problem statement. Proposed solution: show the process in the current practices is automated by the implemented solution. Proposed approach: cite/ relate some similar works/ projects that use evolution prototyping approach in development.	
Project Scope and Objectives - Is the scope of the project clearly defined? - Are the objectives of the project clearly specified? - Are the project scope and objectives appropriate for a final year project?	ok	
Literature Review / Fact Finding for Benchmarking / Verification of Project - Are sources for literature review / fact finding appropriate? - Is information from literature review / fact finding relevant and adequate? - Is information from literature review / fact finding clearly presented and discussed?	Technologies review are too short. Combined with the tools reviewed. Simplify the writing about the development methodologies review.	Improve on the review of technologies used.
Research/Development Methodology and Development Tools - Is the methodology for the project clearly described and discussed? - Are the required development tools clearly described and discussed? - Are the stated methodology and development tools appropriate?	too simplify. The diagram is not complete. Give some explanations of planned activities in each phase prototyping. Identify the outcomes/deliverables in each iteration. Tools used: have explained each tool used; lacking of analysis to help in deciding the tools used.	
Project Plan - Are the phases and tasks of the project properly defined and planned? - Are the phases and tasks consistent with the methodology of the project?	Should show work breakdown structures in hierarchical decomposition of the work. To understand the overall plan, show the important milestones/ phases with its start date, completion date and resources to be used in a table.	
Initial Deliverables - Are deliverables (e.g. use case diagrams and descriptions) of initial phases of the project plan included in the report?	- show the current process (based on the observation and document reviews made) and the proposed process workflow. This is to meet the objective #1.	
Report Structure and References - Is the report organised in a logical structure? - Are references listed in accordance to Harvard format?	All diagrams, tables and appendices must refer in the report content. Inconsistency of font type and size. Content about survey and its result should move to Chapter 1 as it is mainly to help to formulate problems.	
Language and Clarity of Writing - Are the sentences concise and understandable? - Are there spelling and grammar issues?	contain some incomplete sentences. Need to use different variety of conjunction to join the explanation, argument etc. Contains a lot of similar structure of languages.	