

MYSTATION MOBILE APPLICATION

NIK MUHAMMAD ASYRAF BIN NIK ISMAIL

206630

Faculty of Computer Science and Information Technology

Universiti Putra Malaysia

Semester 2023/2024

MYSTATION MOBILE APPLICATION

By

NIK MUHAMMAD ASYRAF BIN NIK ISMAIL

206630

SUPERVISOR

**ASSOCIATE PROFESSOR Ts. DR. MAS RINA BINTI
MUSTAFFA**

This thesis is submitted as the partial fulfilment of requirement for the degree of the Bachelor of Computer Science (Multimedia) with Honours

**Faculty of Computer Science and Information Technology
Universiti Putra Malaysia**

Semester 2023/2024

DECLARATION

I declare that the work in this project is my own except for all the summaries or phrases which have been quoted or mentioned to the sources. The project was not submitted earlier for any degree at Universiti Putra Malaysia (UPM) or any other institutions.

Signature:



Name: Nik Muhammad Asyraf Bin Nik Ismail

Matric Number: 206630

Date: 16 February 2024

Endorsed by: *Mas Rina*

Supervisor Name: Associate Professor Ts. Dr. Mas Rina Binti Mustaffa

17 February 2024

ACKNOWLEDGEMENT

I am profoundly grateful to all those who have played an instrumental role in the completion of this thesis, marking a significant milestone in my academic journey. I extend my heartfelt appreciation to my dedicated supervisor, Assoc. Professor Ts. Dr. Mas Rina Binti Mustaffa, for her unwavering support, invaluable guidance, and patience. Her expertise and encouragement were indispensable throughout the preparation and execution of this project. I am truly fortunate to have had such a mentor who believed in my potential and pushed me to excel. My deepest gratitude goes to my family and friends for their constant love, encouragement, and unwavering support. Their belief in my abilities and the countless moments of encouragement were the driving force that propelled me forward, especially during challenging times. This journey has been both demanding and rewarding, and I want to acknowledge my own determination and perseverance. Despite facing various challenges and obstacles, I remained steadfast in my commitment to seeing this project through to completion. I am immensely proud of the resilience I exhibited and the lessons learned along the way.

ABSTRACT

The evolution of technology has revolutionised information access and daily mobility, emphasising the need for reliable and up-to-date data on railway stations, amenities, and services in public transportation. However, contemporary commuters face challenges in locating essential facilities, compounded by the limited functionalities of existing mobile applications and the absence of integrated platforms for station information and user feedback. This thesis addresses these challenges by proposing the development of a comprehensive station information mobile application, aiming to enhance user decision-making, facilitate robust reporting mechanisms, and foster community engagement. Leveraging Android Studio as the primary development environment and Google Firebase for backend functionalities, the application integrates various Flutter packages to optimise functionality and user experience. MyStation mobile apps uses methods such as DatabaseServiceTrial for handling interactions with Firebase Firestore and Firebase Storage. It includes methods to submit and display sorted reports, submit and retrieve replies to each report, increment and decrement like and dislike counts for feedbacks, edit feedback and reply text, and delete feedback and replies. The app provides extra features to admins for filtering reports and managing images within the app by utilising the Firebase Storage functions. These allows admin to upload and store image URLs in Firestore, retrieve image URLs, and delete images from Firebase Storage. The User Acceptance Test (UAT) phase validates the application's usability, functionality, and overall user experience, gathering insights from a representative user base. Results from the UAT highlight strengths and areas for improvement, guiding iterative enhancements to ensure the application effectively meets user needs. Overall, this thesis contributes to the advancement of commuter experience by providing a valuable tool for accessing essential station information and fostering community-driven enhancements in public transportation services.

ABSTRAK

Perkembangan teknologi telah merevolusi akses maklumat dan mobiliti harian, menekankan keperluan untuk data yang boleh dipercayai dan terkini mengenai stesen kereta api, kemudahan, dan perkhidmatan dalam pengangkutan awam. Walau bagaimanapun, pendatang kontemporari menghadapi cabaran dalam lokasi kemudahan penting, ditambah dengan fungsi terhad aplikasi mudah alih yang sedia ada dan kurangnya platform bersepadu untuk maklumat stesen dan maklum balas pengguna. Tesis ini menangani cabaran ini dengan mencadangkan pembangunan aplikasi mudah alih maklumat stesen yang komprehensif, bertujuan untuk meningkatkan pengambilan keputusan pengguna, memudahkan mekanisme pelaporan yang kukuh, dan menggalakkan keterlibatan masyarakat. Menggunakan Android Studio sebagai persekitaran pembangunan utama dan Google Firebase untuk fungsi backend, aplikasi ini mengintegrasikan pelbagai pakej Flutter untuk mengoptimumkan fungsi dan pengalaman pengguna. Aplikasi mudah alih MyStation menggunakan kaedah seperti DatabaseServiceTrial untuk menangani interaksi dengan Firebase Firestore dan Storage. Ia termasuk kaedah untuk menghantar dan memaparkan laporan yang disyorkan, menghantarkan dan mendapatkan jawapan kepada setiap laporan, menambah dan mengurangkan jumlah like dan dislike untuk maklum balas, mengedit maklumbalas dan teks jawapan, dan memadamkan maklum balasan dan jawapan. Aplikasi ini menyediakan ciri tambahan kepada pentadbir untuk menapis laporan dan menguruskan imej dalam aplikasi dengan menggunakan fungsi Penyimpanan Firebase. Ini membolehkan pentadbir untuk memuat naik dan menyimpan URL imej dalam Firestore, mendapatkan URL gambar, dan memadam imej dari Penyimpanan Firebase. Tahap UAT (User Acceptance Test) mengesahkan kegunaan, fungsi dan pengalaman pengguna keseluruhan aplikasi, mengumpul wawasan daripada pangkalan pengguna yang mewakili. Hasil daripada UAT menonjolkan kekuatan dan kawasan untuk peningkatan, membimbing peningkatan iteratif untuk memastikan aplikasi secara berkesan memenuhi keperluan pengguna. Secara keseluruhan, tesis ini menyumbang kepada kemajuan pengalaman pendatang dengan menyediakan alat yang berharga untuk mengakses maklumat stesen penting dan menggalakkan peningkatan yang dipandu oleh masyarakat dalam perkhidmatan pengangkutan awam.

TABLE OF CONTENTS

DECLARATION	III
ACKNOWLEDGEMENT	IV
ABSTRACT	V
ABSTRAK.....	VI
TABLE OF CONTENTS	VII
LIST OF FIGURES.....	XI
LIST OF TABLES.....	XIV
LIST OF ABBREVIATIONS.....	XV
CHAPTER 1 INTRODUCTION	2
1.1 Introduction	2
1.2 Problem Statement	3
1.3 Project Objectives.....	4
1.4 Project Scope.....	5
1.4.1 User Scope.....	5
1.4.2 System Scope	5
1.5 Expected Output.....	6
1.6 Thesis Outline	6
CHAPTER 2: LITERATURE REVIEW	7
2.1 Introduction	7
2.2 Related Term	7
2.2.1 Added Value of a Customised Transit App for Metropolitan Bus Trips.....	7
2.2.2 Citizenapp: Yet Another Social Media Platform for Public Facility Monitoring in Sidoarjo	8
2.2.3 Emergency Incident Detection from Crowdsourced Waze Data Using Bayesian Information Fusion	9

2.2.4	Exploring Intercity Passengers' Attitudes and Loyalty to Intercity Passenger Rail: Evidence from An On-Board Survey	10
2.2.5	Influence Of Public Transportation Applications for Foreign Tourists.....	11
2.2.6	Offline Mobile Application for Train Time Prediction	13
2.2.7	Tip Express: An Android School Navigation Application	13
2.3	Existing System.....	15
2.3.1	Malaysia MRT/LRT (Offline).....	15
2.3.2	Moovit: Bus & Train Schedules	17
2.3.3	MyRapid Pulse	19
2.4	Conclusion.....	23
CHAPTER 3: METHODOLOGY		24
3.1	Introduction	24
3.2	Agile Software Development Life Cycle	25
3.2.1	Planning.....	25
3.2.1.1	Preliminary Study based on Train Station Information Mobile Application	25
3.2.1.2	Conclusion for Preliminary Survey	28
3.2.2	Design.....	29
3.2.3	Coding	30
3.2.4	Testing	30
3.3	Tools and Languages.....	30
3.3.1	Hardware	31
3.3.2	Software Tools	31
3.3.3	Framework.....	31
3.3.4	Programming Languages.....	32
3.3.5	Server.....	32
3.3.6	Features	32
3.4	Project Timeline	33
CHAPTER 4: SYSTEM ANALYSIS AND DESIGN.....		34

4.1 Requirement Specification.....	34
4.1.1 User Requirements	34
4.1.2 Functional Requirement	35
4.1.3 Non – Functional Requirements.....	35
4.2 System Design	35
4.2.1 User Flowchart	36
4.2.2 Admin Flowchart.....	38
4.3 Data Flow Diagram	40
4.4 Sequence Diagram.....	41
4.5 Use Case Diagram	43
4.6 Entity Relationship Diagram.....	44
4.7 User Interface	45
4.7.1 User Site	45
4.7.2 Admin Site.....	48
CHAPTER 5: IMPLEMENTATION AND DISCUSSION.....	49
5.1 Development.....	49
5.1.1 Android Studio	50
5.1.2 Google Firebase.....	55
5.1.3 Application Programming Interface (API).....	57
5.2 Completed Application Process.....	62
5.2.1 User Site	62
5.2.1.2 Selection of Train Stations	65
5.2.1.3 Overview Page	66
5.2.1.4 Services Page.....	67
5.2.1.5 Feedback Function.....	68
5.2.1.6 Station Reports	69
5.2.1.7 Profile Page	70
5.2.2 Admin Site.....	71

5.2.2.1	Upload and Delete Image	72
5.2.2.2	Delete Irrelevant Feedbacks and Reports	73
5.3	User Acceptance Test	74
5.4	Result.....	79
5.4.1	User Interface (UI) and Design	80
5.4.2	Performance.....	82
5.4.3	Functionality.....	82
5.4.4	User Feedback	85
5.5	Discussion.....	88
CHAPTER 6 CONCLUSION AND FUTURE WORK		92
6.1	Conclusion.....	92
6.2	Limitations	92
6.3	Future Works.....	93
6.4	Personal Reflections	93
REFERENCES		94

LIST OF FIGURES

Figure 2.1	Screenshot of Malaysia MRT/LRT (Offline) Mobile Application Part 1	16
Figure 2.2	Screenshot of Malaysia MRT/LRT (Offline) Mobile Application Part 2	17
Figure 2.3	Screenshot of Moovit: Bus & Train Schedules Mobile Application Part 1	18
Figure 2.4	Screenshot of Moovit: Bus & Train Schedules Mobile Application Part 2	19
Figure 2.5	Screenshot of MyRapid PULSE Mobile Application Part 1	20
Figure 2.6	Screenshot of MyRapid PULSE Mobile Application Part 2	21
Figure 3.1	Phases in Agile Development Lifecycle	24
Figure 3.2	Response for Question 3	26
Figure 3.3	Response for Question 5	26
Figure 3.4	Response for Question 6	27
Figure 3.5	Response for Question 8	28
Figure 3.6	Basic Mock-up of the Mobile Application (i)	29
Figure 3.7	Basic Mock-up of the Mobile Application (ii)	29
Figure 3.8	Gantt Chart FYP A	33
Figure 3.9	Gantt Chart FYP B	33
Figure 4.1	User Flowchart	37
Figure 4.2	Admin Flowchart	39
Figure 4.3	Data Flow Diagram	41
Figure 4.4	Sequence Diagram	42
Figure 4.5	Use Case Diagram	43
Figure 4.6	Entity Relationship Diagram	44
Figure 4.7	User Interface Part 1	45
Figure 4.8	User Interface Part 2	46
Figure 4.9	User Interface Part 3	46
Figure 4.10	User Interface Part 4	47
Figure 4.11	User Interface Part 5	47

Figure 4.12	Admin Interface	48
Figure 5.1	Android Studio and Emulator	50
Figure 5.2	Flutter Firebase package used in Android Studio	51
Figure 5.3	Authentication Package used in Android Studio	52
Figure 5.4	ImagePicker package used in Android Studio	53
Figure 5.5	Customisable Rating Slider Using the Slider Widget	54
Figure 5.6	Firebase Authentication	55
Figure 5.7	Firestore Database	56
Figure 5.8	Firebase Storage	56
Figure 5.9	Method to Retrieve User's Username	57
Figure 5.10	Method Allowing Users to Edit Username	58
Figure 5.11	Method to Update User's Username in the Database	59
Figure 5.12	"users" Collection in the Firebase	60
Figure 5.13	Syntax to Pass the Username Parameter	60
Figure 5.14	Method to Access the Data from Firebase Collection	61
Figure 5.15	How the Username is Displayed in the App	61
Figure 5.16	Register and Sign in Interfaces	63
Figure 5.17	User Main Menu	64
Figure 5.18	Putrajaya Line	65
Figure 5.19	Overview Page	66
Figure 5.20	Services Page	67
Figure 5.21	Feedback Function	68
Figure 5.22	Station Reports	69
Figure 5.23	User Profile Page	70
Figure 5.24	Upload and Delete Image	72
Figure 5.25	Delete Irrelevant Feedbacks and Reports	73
Figure 5.26	Questions in the User Interface Section	75
Figure 5.27	Questions in the Performance Section	76
Figure 5.28	Questions in the Functionality Section Part 1	77
Figure 5.29	Questions in the Functionality Section Part 2	77
Figure 5.30	Questions in the User Feedback Section	78

Figure 5.31	Type of Trains Used by Respondents	79
Figure 5.32	The Frequency of Respondents Using Trains	79
Figure 5.33	Users' rating for the overall visual design	80
Figure 5.34	Users' Rating on Clarity and Aesthetic Appeal of Visual Elements	81
Figure 5.35	Users' Rating on Ease of Navigation	81
Figure 5.36	Users' Responses on Challenges Faced	81
Figure 5.37	Users' Responses on the Performance of the App	82
Figure 5.38	Users' Responses on Functionality of the App	83
Figure 5.39	Users' Comments on the Functionality of the App	83
Figure 5.40	Users' Rating on the Features of the App	83
Figure 5.41	Users' Rating on Functionality Compared to Users' Initial Expectation	84
Figure 5.42	Users' Rating on the Successfulness in Completing the Tasks	84
Figure 5.43	Users' Rating on the Effectiveness in Meeting User Needs	85
Figure 5.44	Users' Suggestions on the Features to be Added or Improved (i)	85
Figure 5.45	Users' Suggestions on the Features to be Added or Improved (ii)	86
Figure 5.46	Users' Suggestions on the Features to be Added or Improved (iii)	86
Figure 5.47	Users' Suggestions on the Features to be Added or Improved (iv)	87
Figure 5.48	Users' Choice on the Likelihood of Recommending the App to Others	87

LIST OF TABLES

Table 2.1	Existing Application Comparison	18
Table 3.1	Hardware Specifications	28
Table 5.1	Summary of User Feedback on Application User Interface	83
Table 5.2	Summary of User Feedback on Application Performance	83
Table 5.3	Summary of User Feedback on Application Functionality	84
Table 5.4	Users' Feedback Summary	84

LIST OF ABBREVIATIONS

RTI	-	Real-Time Information
MAM	-	Multi-attribute Attitude Model
UAT	-	User Acceptance Testing
IDE	-	Integrated Development Environment
DFD	-	Data Flow Diagram
ERD	-	Entity Relationship Diagram
API	-	Application Programming Interface
UID	-	User ID
UI	-	User Interface

CHAPTER 1 INTRODUCTION

1.1 Introduction

The rapid development of technology has fundamentally changed how we obtain information and move through our daily lives. It has become crucial for commuters to have access to reliable and recent information on railway stations, amenities, and services in the world of public transport. Various amenities are frequently available at train stations to improve the commuter experience. These might consist of cheque-in desks, waiting spaces, bathrooms, stores, dining establishments, information desks, and more. Within the station, various amenities may be distributed differently and in different layouts. Users can more easily locate these facilities without needless searching or misunderstanding if they utilise a smartphone app that provides information on their location. We suggest creating a station information mobile application to meet this purpose.

Crowd control is essential because of the large number of passengers and the complexity of railway stations. For both safety and effectiveness, a smooth passenger flow and congestion avoidance are crucial. To assist passengers in navigating the station more efficiently and avoiding congested sections, a mobile app can offer real-time information on crowd levels, busy locations, or other routes.

Real-Time Information (RTI), is the abbreviation for current and accurate data or updates sent in real-time. The development of real-time information processes by public transport operators has enabled them to provide users basic information on their smartphones [1]. RTI can offer data on platform or train capacity, as well as the density of people. This enables railway station applications to provide information about busy times or peak hours, assisting passengers in making the most effective travel plans. Users can choose trains or stations that are likely to be less congested by knowing this information in advance.

RTI enables railway station applications to give consumers quick notifications in situations where there are facility disruptions, such as renovations on the lavatory at the station or temporary closures of some food outlets. Commuters can be informed of these changes, giving them the opportunity to make alternate travel plans or select alternative routes to get to their destinations.

This smartphone app seeks to give users thorough and easy-to-use information about various railway stations, allowing them to move about the stations more quickly and effectively. The app will give customers access to station maps, images, descriptions of the amenities, and services offered at each station by utilising mobile technology. It will also include real-time information on train timetables and any service interruptions, guaranteeing a smooth and informed commute.

1.2 Problem Statement

Commuters navigating through contemporary railway stations, particularly in vast and intricate transportation hubs, face substantial challenges that hinder their travel experience. The array of facilities, amenities, and services offered by train stations to meet travellers' demands often leaves users struggling due to the lack of a centralised platform providing comprehensive information. This challenge is particularly pronounced for those unfamiliar with the intricate layouts of these stations. Without a mobile application offering detailed insights into station facilities and services, commuters find it difficult to efficiently navigate the station and locate the essential facilities they need, resulting in suboptimal travel experiences.

Moreover, the existing mobile applications tailored for train station information may lack robust functionalities in user reporting and feedback. This limitation poses constraints on the timely addressing of real-time issues and concerns faced by commuters. The deficiency in a comprehensive reporting mechanism not only restricts the users' ability to contribute valuable insights but also hampers the efficiency of addressing emergent issues within the station environment. This hinders the potential for creating an environment where users

actively engage in sharing experiences and contributing to the overall improvement of the station services.

Furthermore, the overarching issue lies in the absence of a unified platform that seamlessly integrates essential station information with user-centric reporting features. This absence contributes to a pervasive lack of convenience and satisfaction for train station users. As users navigate through stations, the disjointed nature of information availability and reporting mechanisms impedes their ability to make informed decisions. A centralised platform that combines vital station information with real-time user reporting features is crucial to providing users with a holistic, convenient, and satisfying experience. The lack of such integration diminishes the overall quality of service and convenience that users expect during their journey.

1.3 Project Objectives

With the intention of improving the train station experience, this project sets out to achieve the following objectives:

- To enhance user decision-making by incorporating visual representations, including photos and detailed descriptions of facilities and services, to offer a preview of the station environment.
- To implement robust user reporting and feedback functionalities, allowing commuters to contribute real-time information, share experiences, and address issues within the station environment.
- To develop a comprehensive mobile application that not only serves as a repository of train station information but also fosters a sense of community to drive improvements in the quality of services.

1.4 Project Scope

The development and deployment of a complete mobile application for information about train stations is included in the project's scope. The app will act as a user-friendly platform with several features and capabilities to improve the commuter experience. The main emphasis will be on finding solutions to the problems consumers have when navigating within railway stations, finding pertinent information about amenities and services, and keeping up with real-time changes.

1.4.1 User Scope

The mobile application targets commuters and travellers who utilise train stations regularly or occasionally for their daily commute or travel purposes. It caters to both experienced users who are familiar with the stations and newcomers who may be unfamiliar with the station layout and available services. Users can engage with the app through their mobile devices, utilising a range of features including station images, detailed facility descriptions, real-time updates, and service locating functionalities. They can search for specific facilities, access visual representations through photos, read comprehensive descriptions and share feedback in the app.

1.4.2 System Scope

The mobile app is specifically designed for Android platforms, offering compatibility and accessibility for users on mobile devices. The system includes a robust data management component, encompassing a comprehensive database that stores vital information about various train stations. This database includes details about station facilities, accompanied by relevant photos, enabling users to gain visual insights. Additionally, the system facilitates the collection and storage of user feedback and ratings, allowing for analysis and future enhancements based on user experiences.

1.5 Expected Output

The expected output from the MyStation app project would be a functional and user-friendly mobile application that provides users with comprehensive information about different train stations. The app should fulfil the following expectations:

- The app should have an intuitive and visually appealing user interface, making it easy for users to navigate and access station information.
- The app should provide users with accurate and up-to-date information about various train stations. This includes station maps, photos, and descriptions of facilities and services available at each station.
- The app should provide real-time updates on station information, including any changes in feedbacks, station reports or other relevant announcements. This ensures that users have the most current and reliable information.

1.6 Thesis Outline

There are six chapters in this thesis. Chapter 1 is basically an introduction of a mobile-based application named as MyStation along with the problem statements, objectives, project scope, and expected output. Chapter 2 is a review of literatures and existing applications that are implemented mobile apps in multiple systems. Chapter 3 includes the methodology and its stages to be used in this project, the tools and programming language, and the project planning. Next, Chapter 4 presents the system architecture including the user interface design and the database design. Chapter 5 explains the implementation of the project, full process of the application, UAT, and discussion. Lastly, Chapter 6 will conclude the project as well as with the limitations and future work of this project.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

The literature review explores the existing research and studies related to train station information mobile applications, aiming to provide a comprehensive understanding of the current state of the field.

2.2 Related Term

The related term will be defined in further detail below.

2.2.1 Added Value of a Customised Transit App for Metropolitan Bus Trips

The article explores the benefits of a customised transit app specifically designed for metropolitan bus trips. As per the authors' account, a transport application was created as a component of the European HARMONY project, aiming to augment the information accessible to passengers regarding Metropolitan buses [1].

The article adopts a qualitative research approach, employing a combination of user surveys, interviews, and data analysis. The researchers conducted surveys among metropolitan bus users to gather insights on their preferences, needs, and experiences with existing transit apps. They also conducted interviews with transportation authorities and app developers to gain a deeper understanding of the challenges and opportunities in developing a customised app. The collected data was analysed to identify key features and functionalities that can enhance the value of the app for users.

The results of the research indicate that a customised transit app for metropolitan bus trips offers significant added value to users. One key finding is the importance of real-time updates in the application. Public transport operators have leveraged the advancements in real-time information systems to furnish smartphone users with essential details [1]. Users expressed a strong desire for accurate and up-to-date information regarding bus schedules,

delays, and route changes. They emphasised that the availability of real-time updates greatly enhances their travel experience by allowing them to plan their journeys effectively and make informed decisions. This aligns with the objective of this project, which aims to develop a train station information mobile application complete with user feedback as crowdsource data. Real-time updates are essential in providing users with accurate and up-to-date information about train schedules, delays, and platform changes.

In relation to the project, which focuses on developing a train station information mobile application, the insights from this article can be valuable in informing the app's design and features. By incorporating real-time updates, personalised features, and user-friendly interfaces, we can enhance the value and usability of the app, aligning with the objectives of providing comprehensive train station information and improving the overall travel experience for users.

2.2.2 Citizenapp: Yet Another Social Media Platform for Public Facility Monitoring in Sidoarjo

The authors suggest that to facilitate the reporting of damages to public facilities in Sidoarjo, a platform has been established whereby users can scan the QR code present on each public facility to document the relevant data. The purpose of this work was to explore the implementation and effectiveness of a social media platform called CITIZENAPP for monitoring public facilities in the region of Sidoarjo.

The article employs a case study approach, focusing on the implementation of the CITIZENAPP platform in Sidoarjo. The researchers conducted interviews and surveys with residents, government officials, and app users to gather insights on their experiences and perceptions of using CITIZENAPP. They also collected data on the usage patterns, user engagement, and impact of the platform on public facility monitoring. The collected data was analysed to evaluate the effectiveness of the app in improving public facility management and engaging citizens in monitoring activities. The app provides a platform for citizens to report issues and problems related to public facilities such as roads, parks, waste management, and infrastructure. Users can upload photos, provide descriptions, and

geotag the location of the issues, allowing authorities to identify and address them promptly.

The research findings indicate that CITIZENAPP has led to increased citizen engagement and participation in public facility monitoring [2]. Users reported a sense of empowerment and ownership in contributing to the improvement of their community. The platform also facilitated better communication and collaboration between citizens and local government authorities, enabling timely responses to reported issues. Furthermore, the study highlights the role of social media features in the app, such as likes, comments, and sharing functionalities. These features fostered a sense of community among users and encouraged them to actively participate in monitoring activities. The social media aspect of the app also enabled users to raise awareness about public facility issues and mobilise support from other community members.

For the proposed project, we can consider implementing features that allow users to report issues or provide feedback on train station facilities and services. This can foster a sense of ownership among users and enable them to actively contribute to the improvement of train station amenities. In addition, exploring the use of social media features to promote community engagement and awareness can enhance the user experience and encourage user participation in providing feedback and monitoring train station facilities.

2.2.3 Emergency Incident Detection from Crowdsourced Waze Data Using Bayesian Information Fusion

The objective of the paper is to propose a methodology for detecting emergency incidents, such as accidents or road hazards, using crowdsourced data from the Waze platform. The study aims to leverage Bayesian information fusion techniques to enhance the accuracy and reliability of emergency incident detection based on the collected Waze data.

The paper adopts a data-driven approach, utilising crowdsourced data from the Waze platform to detect emergency incidents. The researchers gather real-time data from Waze users, which includes incident reports, traffic flow information, and location data. Bayesian

information fusion techniques are applied to integrate and analyse the collected data, combining multiple sources of information to improve the accuracy of incident detection.

The research results demonstrate that the integration of Waze data and the application of Bayesian information fusion techniques enable the timely detection of emergency incidents. The improved accuracy in incident detection has the potential to facilitate faster response times from emergency services and aid in the management of traffic congestion around the incidents. The paper highlights the value of crowdsourced data in emergency incident detection and emphasises the importance of utilising Bayesian information fusion techniques to combine and analyse diverse data sources [3]. By leveraging the collective contributions of Waze users, authorities can benefit from a broader and more up-to-date understanding of the current traffic and incident conditions, ultimately improving emergency response and overall public safety.

We are considering crowdsourced data gathered from the users' feedback to provide real time update. By incorporating incident detection features into the app and utilising Bayesian information fusion or similar techniques, we can enhance the accuracy and reliability of incident reporting within train stations, improving the overall safety and user experience for commuters and travellers.

2.2.4 Exploring Intercity Passengers' Attitudes and Loyalty to Intercity Passenger Rail: Evidence from An On-Board Survey

The authors conducted research to determine what motivated people to keep using rail services. In order to better understand passengers' perspectives, preferences, and variables impacting their loyalty to intercity train travel, the researchers attempt to collect empirical information via an on-board survey. The simplest way to measure a user's loyalty to a particular method of public transport is to gauge how likely they are to continue using it [4].

The study uses a quantitative research methodology and conducts an on-board survey of intercity rail passengers to gather data. To investigate the attitudes of passengers towards

passenger train services and potential competing modes along the HST route, the multi-attribute attitude (MAM) model was selected [4]. A sample of passengers were given the survey by the researchers while they were travelling, allowing them to record their attitudes, levels of satisfaction, opinions of the quality of the services, and the variables influencing their loyalty to intercity rail travel. To pinpoint important elements and patterns relating to passengers' opinions and loyalty, the gathered data is analysed.

The study's conclusions emphasise the significance of customer satisfaction and service excellence in determining passengers' opinions and loyalty. Higher levels of customer satisfaction and a higher chance of returning to a destination are influenced by positive experiences, dependable service, and effective customer service. The survey also suggests areas for development to raise customer happiness and loyalty, giving train service providers useful information. Operators of intercity trains should concentrate on the aspects that passengers gave the train good marks for in order to market the train and look for other ways to enhance the aspects that received low marks [4].

This project may design features for the app that cater to the interests of our users by considering the elements that affect customer loyalty, such as service quality and convenience. For instance, offering accurate platform information, real-time updates on train station services, and user-friendly navigation tools can improve passengers' travel experiences and possibly increase station loyalty.

2.2.5 Influence Of Public Transportation Applications for Foreign Tourists

In order to improve the level of service in railway stations, the newest technical infrastructure with intelligence systems that process information in real time and information exchange must be available [5]. This research aims to investigate how public transit applications affect international visitors. Besides, the relationship on between these applications and foreign visitors' travel experiences, decision-making, and general happiness when utilising public transit in foreign cities must be studied.

The researchers claimed that the study uses a mixed-methods approach to data collecting, integrating qualitative and quantitative techniques. To acquire information from foreign visitors who have utilised public transport applications while travelling, surveys, interviews, and observations are undertaken. The qualitative data offers perceptions on the advantages, difficulties, and experiences related to utilising these programmes. To find statistical patterns and connections relating to the impact of public transit applications on foreign visitors, the quantitative data is analysed.

Based on the article, the researchers found out that those applications are essential for improving accessibility, efficiency, and comfort when using foreign public transit systems. This mobile app makes it easy to see routes that are travelled by buses, trains, and other forms of public transit [5]. Features like real-time information, route planning, multilingual support, and payment system integration are valuable to travellers from other countries.

Other than that, apps for public transport have a favourable influence on foreign visitors' decision-making by giving them trustworthy information, lowering uncertainty, and boosting their confidence in utilising public transport. These tools enable autonomous traveller exploration of unfamiliar locations, enhance feelings of safety and security, and promote cross-cultural encounters.

However, the study also outlines several problems and opportunities for development in relation to apps for public transit. Language obstacles, technological difficulties, the insufficient coverage of some locations, and the requirement for ongoing updates and enhancements to guarantee accuracy and usefulness are a few of these.

We can address these problems by considering about including multilingual help, simple navigation, and real-time updates. Our app may enhance the good travel experiences of foreign visitors and promote the usage of public transport by offering a user-friendly and educational experience.

2.2.6 Offline Mobile Application for Train Time Prediction

The paper's goal is to create and recommend an offline mobile application that offers precise train time forecasts. By enabling users to obtain train timetables and time projections without relying on real-time Internet data, the study seeks to solve the difficulties experienced by users in locations with poor or no internet connectivity. Google's map services are entirely dependant on the availability of Internet services, such as 3G and higher networks, which are unreliable during train rides and use a significant amount of battery power and packet data [6].

To collect historical train schedule data, analyse it, and develop prediction models, the authors use a variety of approaches, algorithms, and data analytic methods. Even in offline mode, the application is intended to use these prediction algorithms to deliver precise train time forecasts. The study's findings show how well the offline mobile application for predicting train times works. With the help of the application, users can obtain the information they want even in locations with spotty or no internet connectivity since it effectively uses historical train schedule data and prediction algorithms to deliver precise and trustworthy train time forecasts.

Relating our project with the journal paper, our app may serve users in places with spotty or inconsistent internet connectivity by offering offline access to train timetables, station information, and other pertinent information. Regardless of network connectivity, this function makes sure customers can still get vital information and efficiently plan their train travels.

2.2.7 Tip Express: An Android School Navigation Application

The objective of the research paper is to create and showcase an Android-based navigation application that is especially made for travelling a school campus. The study's goal is to give students, employees, and visitors a simple and effective method to find their way about the campus, find buildings, courses, and amenities, and access important information.

A mobile application that creates the route transfer from one building to another is the goal of this project [7]. The application was designed and developed by the authors using Android development tools and methods. Data collecting, campus mapping, and the incorporation of features like building and room search, real-time position tracking, route planning, and information retrieval are all part of the development process.

When a user signs in or registers with the system, the application requests the user's credentials for authentication, which starts the process. The user creates an account upon registration by providing personal information such name, address, age, and birthdate, which is then recorded in the database. After successfully completing registration, the user may log in and take advantage of the mobile application's features. The app integrates several features, including as location-based services, Text-to-Speech, Bluetooth technology, and notification systems. The mobile application uses automated beginning point detection and lets users enter numerous destinations for navigational reasons. The smartphone application uses a fuzzy logic technique to provide text-to-speech capabilities that gives users navigation directions. Additionally, the app uses a channel selection algorithm and Bluetooth technologies to deliver details about active users. Finally, the app uses MD5 encryption to guarantee the confidentiality of data supplied by users while creating accounts.

Users may thereby streamline their movement across the campus of the school, saving time and reducing the likelihood of getting lost. The mobile app also improves the entire campus experience by giving users access to pertinent data about the many facilities, activities, and services that are offered there.

The knowledge from this article might motivate us to think about include navigation and wayfinding capabilities particular to railway stations in our project. Our app may help users navigate railway stations more effectively by identifying platforms, amenities, and services by offering them extensive maps, search features, and real-time advice. Users at railway stations will benefit from a smooth and improved travel experience as a result of this.

2.3 Existing System

This section provides an overview of the current landscape of train station information mobile applications. It examines and evaluates the existing apps that are available in the market or have been implemented in similar contexts. This section aims to identify the strengths, limitations, and gaps in the existing solutions, providing insights for the proposed development of a new and improved mobile application. Through a comprehensive analysis, it assesses the functionalities, features, user interfaces, and user experiences offered by these existing apps. Additionally, it examines the extent to which these apps address the challenges faced by users in navigating train stations, accessing facility information, receiving real-time updates, and engaging with user feedback. By examining the existing apps, this section serves as a foundation for identifying opportunities for innovation and improvement in the development of the proposed train station information mobile application.

2.3.1 Malaysia MRT/LRT (Offline)

The app's primary function is to provide offline train station information for the MRT and LRT networks in Malaysia. Offline maps, station details, train timetables, and route planning are some of the things it provides. The app has an easy-to-use UI with simple navigation. Access to various elements, such as maps, stations, timetables, and route planning, is made simple from the main screen. Finding the needed information is simple thanks to the user-friendly app navigation.

The availability of offline maps is one of this app's unique features. Without an internet connection, users may browse maps of the MRT and LRT networks, which is useful for travellers who might experience spotty connections while travelling. The maps show the locations of the stations, interchanges, and local landmarks in a straightforward and understandable manner.

There is need for improvement even if the software offers helpful offline train station information. Real-time alerts on train delays, interruptions, or modifications are one thing to think about because they would make the app much more helpful. The app may be more

valuable if it included features like fare information, ticketing choices, and user feedback and ratings. The interface of the Malaysia MRT/LRT (Offline) system is shown in Figure 2.1 and Figure 2.2.

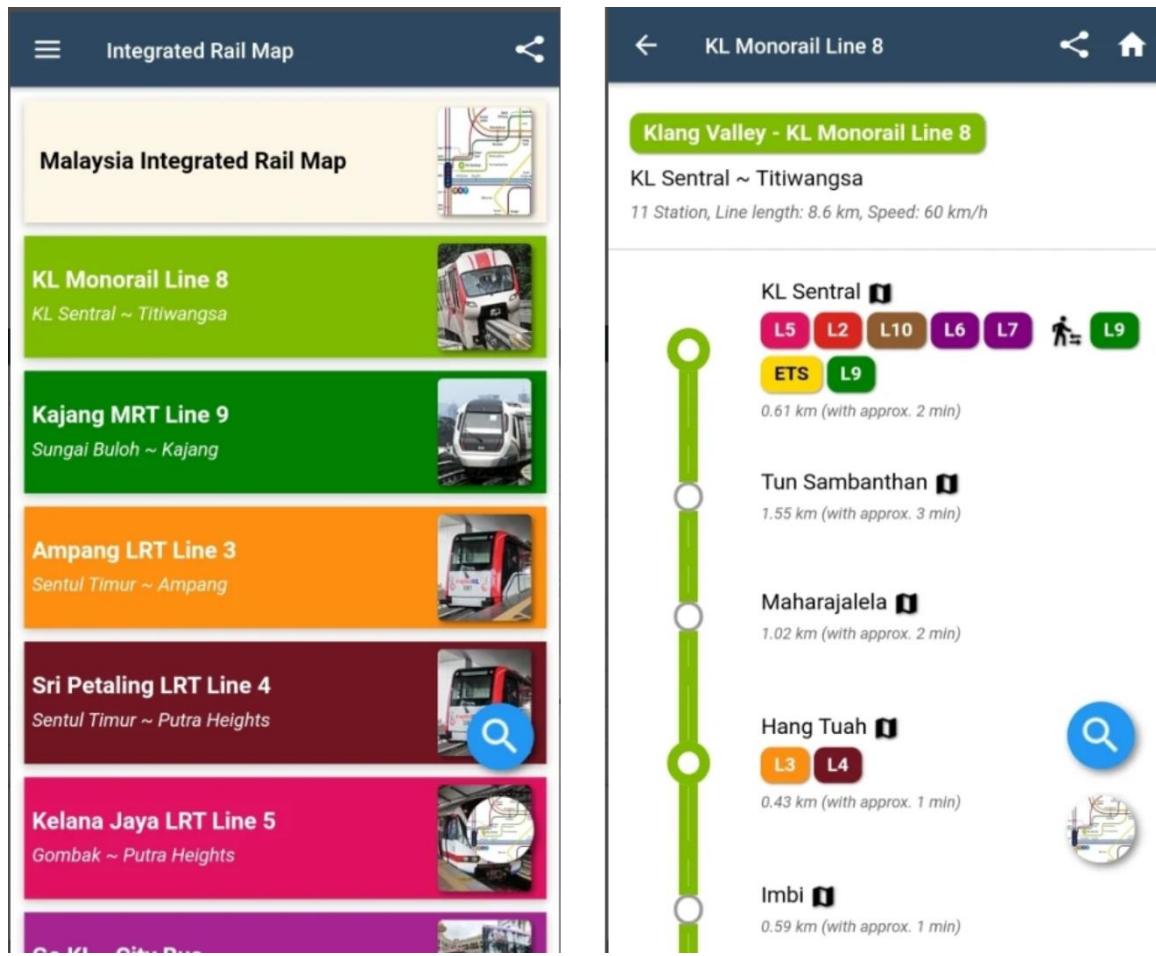


Figure 2.1 Screenshot of Malaysia MRT/LRT (Offline) Mobile Application Part I

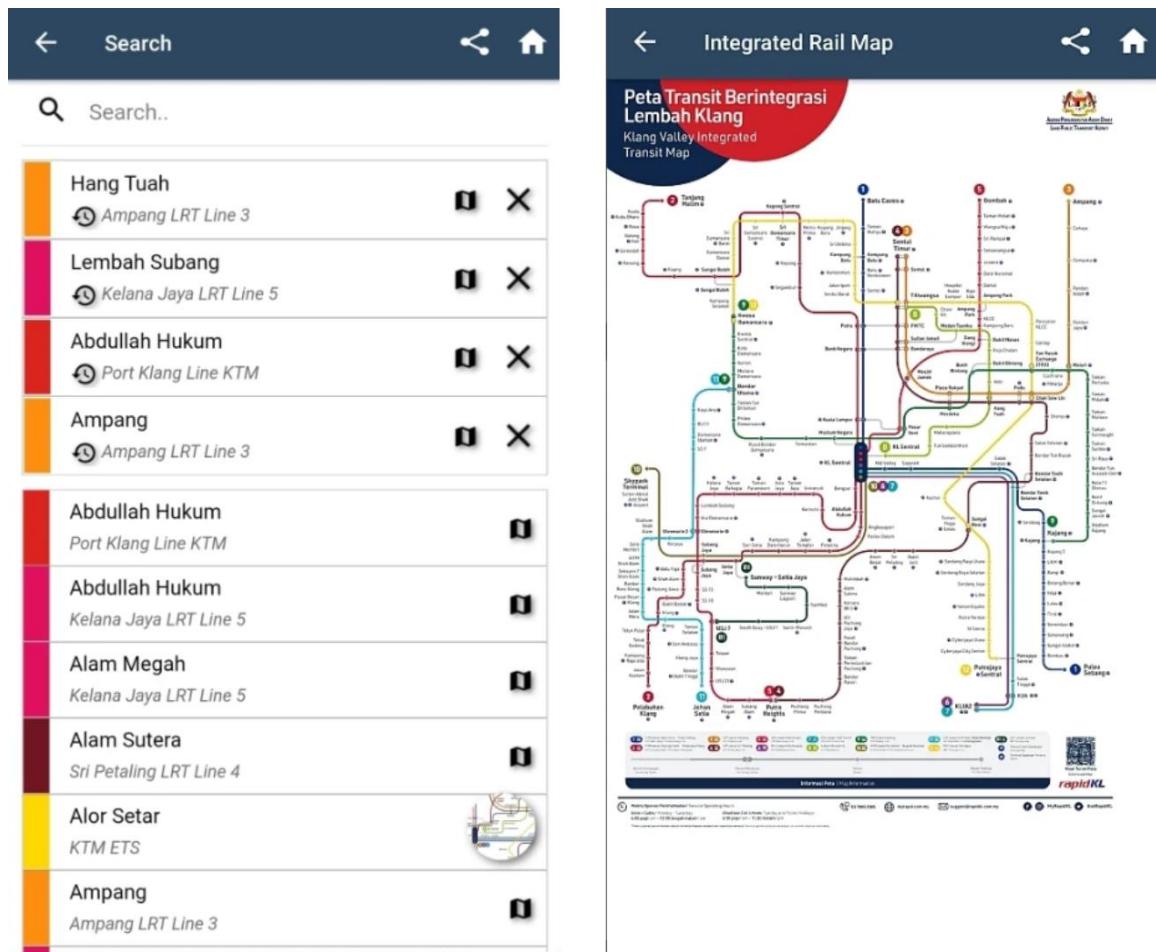


Figure 2.2 Screenshot of Malaysia MRT/LRT (Offline) Mobile Application Part 2

2.3.2 Moovit: Bus & Train Schedules

The mobile application under review is called "Moovit: Bus & Train Schedules" and it was created by Moovit. Real-time bus and rail timetables, route planning tools, and navigation support are all included in the app. It has functions including real-time updates, detailed instructions, and multimodal journey planning.

Real-time updates are one of Moovit's best qualities. Users may monitor the current status of buses and trains, including any delays, arrivals, or departures, using the app's live departures feature. By providing the most recent information, this function guarantees that customers may efficiently plan their travels.

Next, users may follow the app's precise, step-by-step guidance while travelling. Users are assisted in navigating via various stops, interchanges, and transfers by clear directions and visual cues. To improve the overall user experience, Moovit also delivers warnings and messages about impending stops, transfers, and any disruptions or modifications to the chosen route.

When it comes to providing multimodal travel planning tools, Moovit excels. The software allows users to enter their desired origin and destination, and it then offers a variety of route possibilities that combine different transportation options including buses, trains, trams, and more. This function offers thorough journey planning and enables easy switching between various forms of public transport. Figure 2.3 and Figure 2.4 displays the Moovit: Bus & Train Schedules system's user interface.

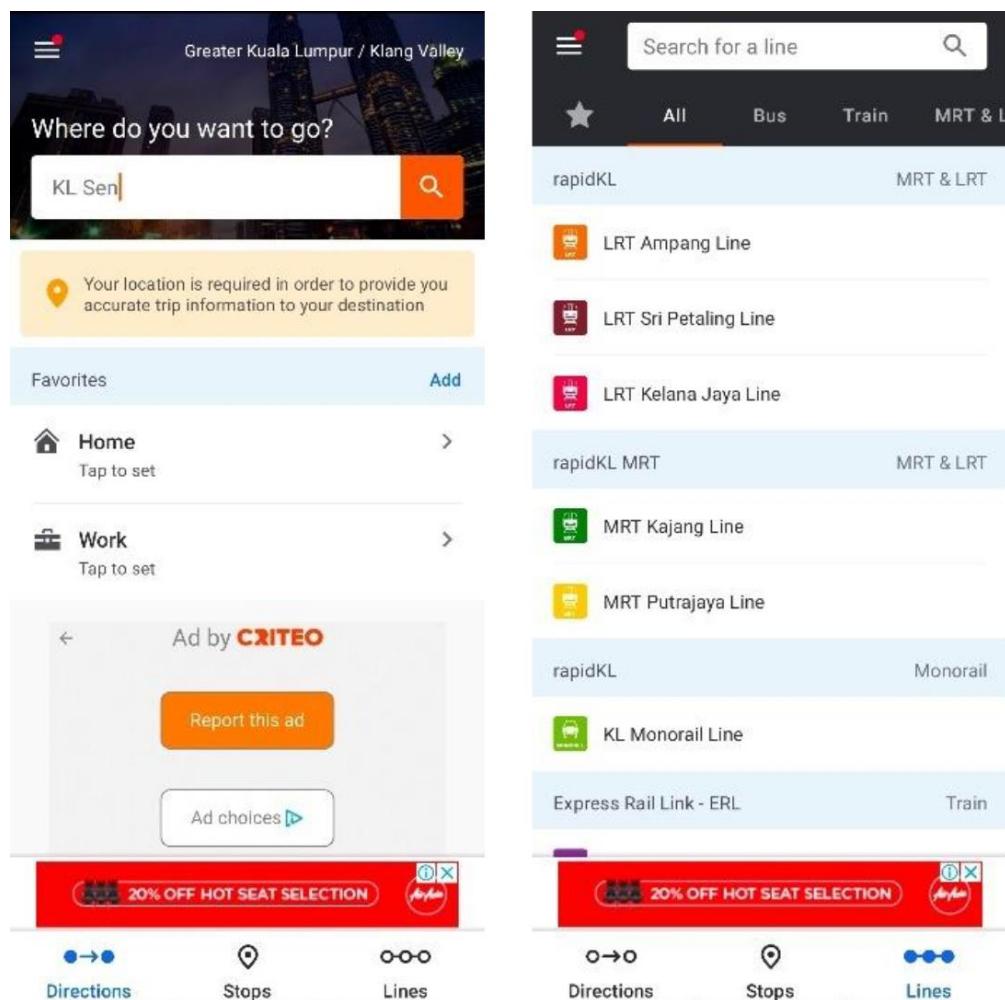


Figure 2.3 Screenshot of Moovit: Bus & Train Schedules Mobile Application Part 1

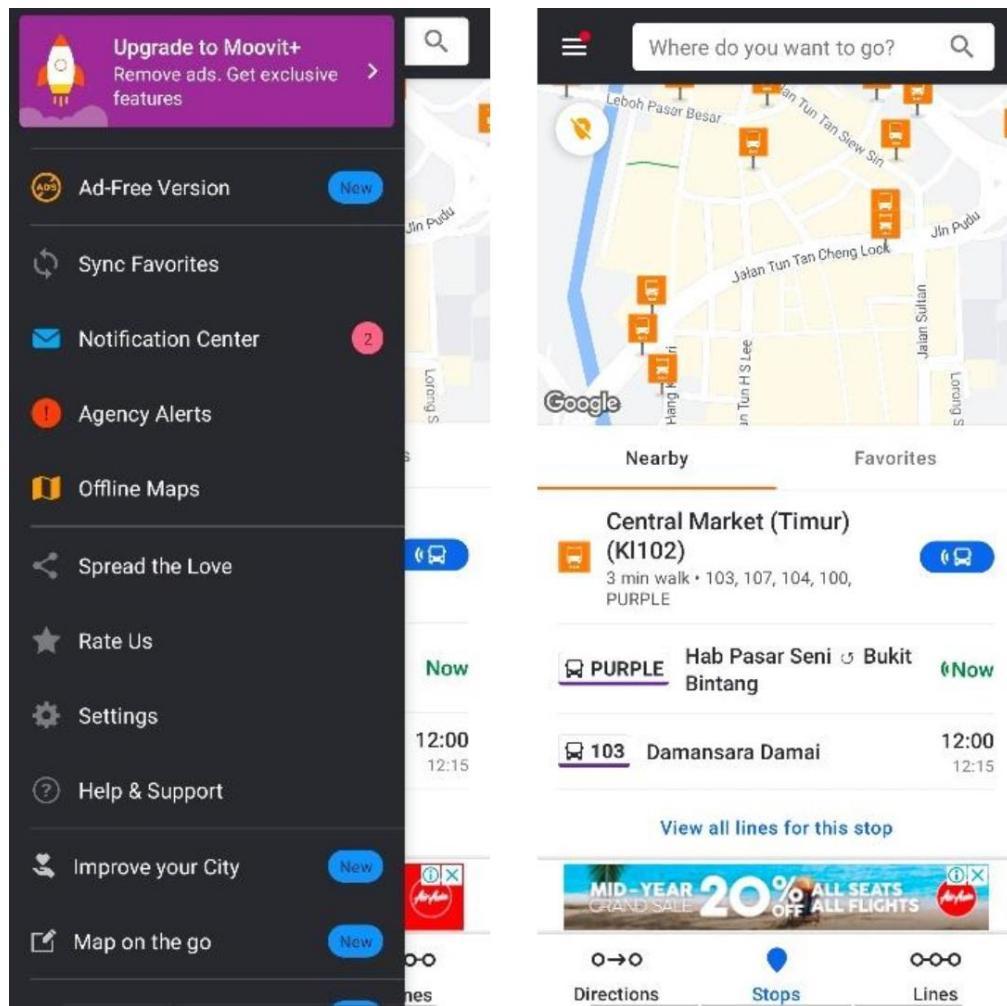


Figure 2.4 Screenshot of Moovit: Bus & Train Schedules Mobile Application Part 2

2.3.3 MyRapid Pulse

The smartphone application was created by Prasarana Malaysia Berhad. The app's users will have access to in-depth data and services on Malaysia's public transportation system, including information on trains, buses, and other modes of transit. Real-time information, travel planning, ticket booking, and several transit-related services are among the features it provides.

The application features an easy-to-use UI with simple navigation. A variety of choices are available on the home page, including local stations, routes, and service updates. It is easier for users to plan their trips and use the public transport system since a variety of features and information are available and easy to access.

The app offers real-time data on railway and bus arrivals, departures, and any delays or service interruptions. Users may use this tool to monitor the progress of their excursions and make changes as needed. MyRapid PULSE provides extensive travel planning features. The app allows users to input their starting point and ending point, and it offers a variety of route alternatives that take advantage of trains, buses, and other forms of transportation. Users may more easily comprehend and select the best choice thanks to the route mapping tool, which presents clear visual representations of the proposed routes. The interface of the MyRapid PULSE system is shown in Figure 2.5 and Figure 2.6.

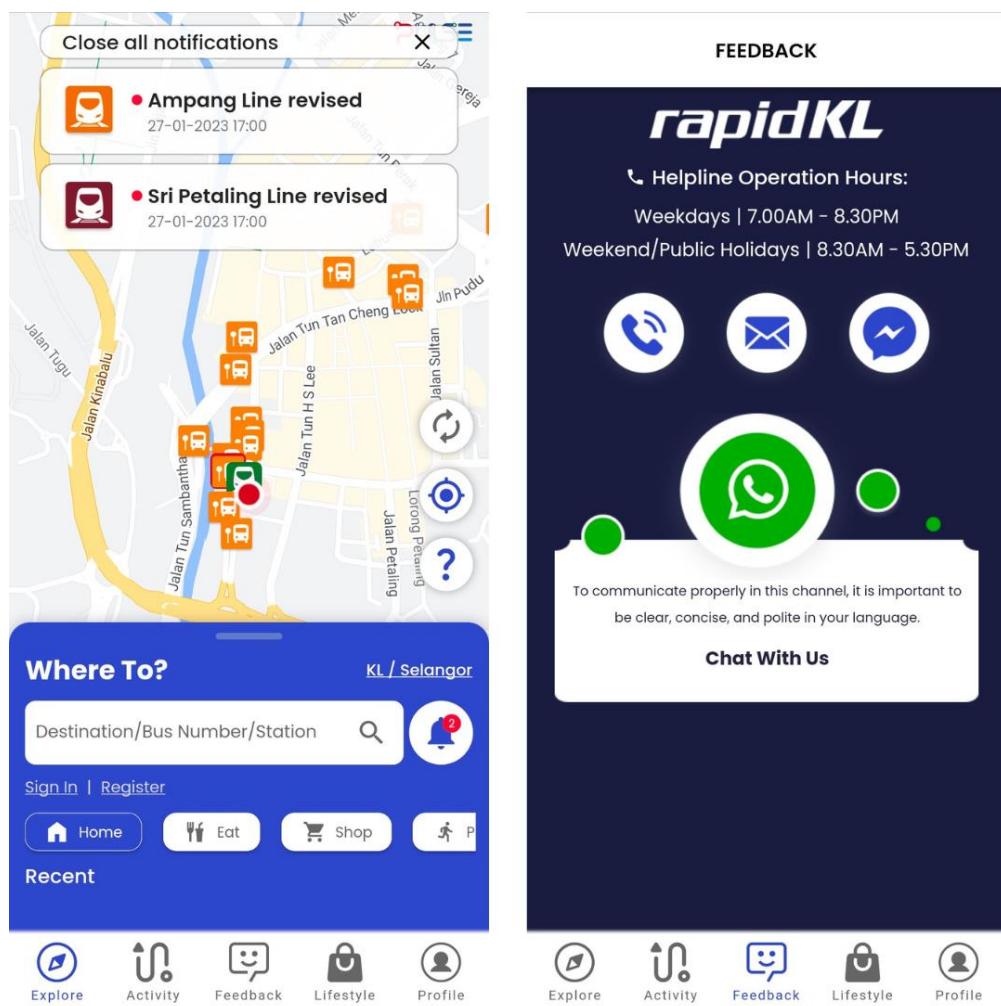


Figure 2.5 Screenshot of MyRapid PULSE Mobile Application Part 1

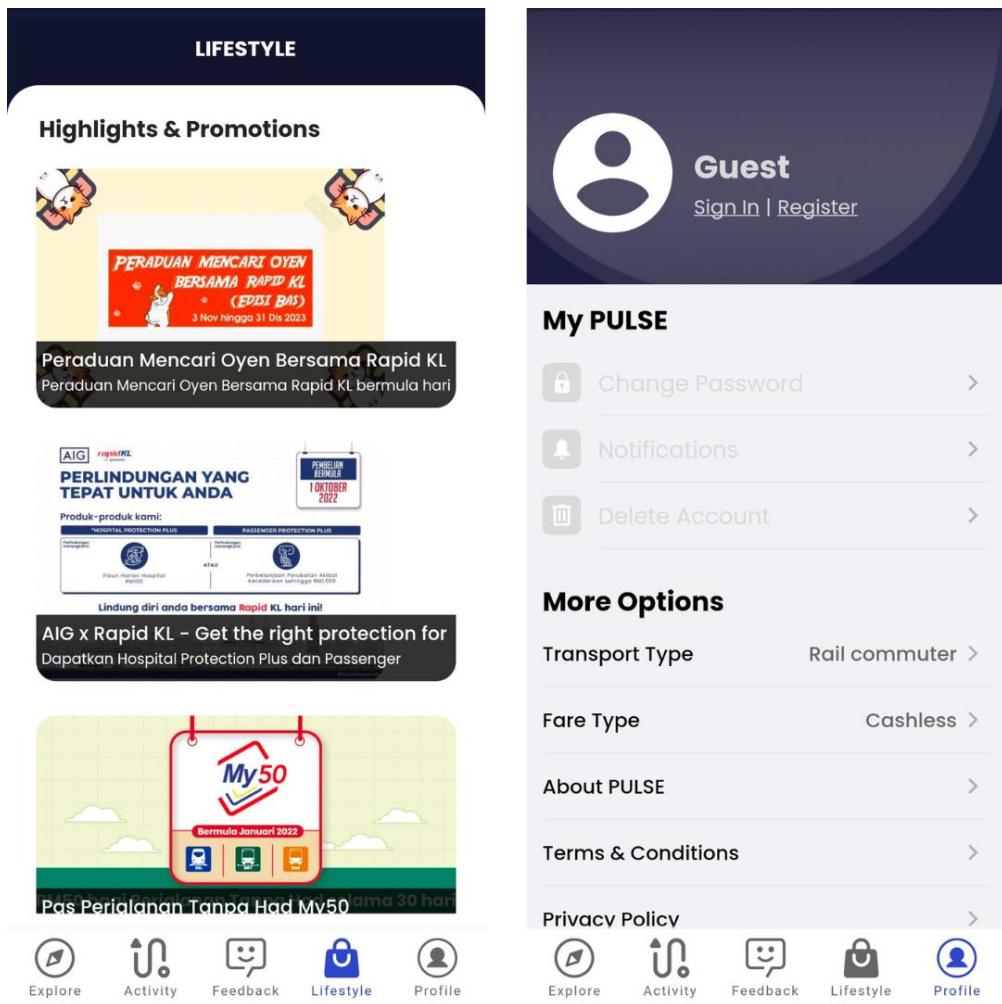


Figure 2.6 Screenshot of MyRapid PULSE Mobile Application Part 2

Table 2.1 Existing Application Comparison

Name	Malaysia MRT/LRT (Offline)	Moovit: Bus & Train Schedules	MyRapid PULSE
References	PB Solutions. (2020). Malaysia MRT/LRT (Offline) (Version 4.3) [Mobile application software]. Retrieved from https://play.google.com/store/apps/details?id=pb.mrt.malaysiarailmap	Moovit (2012). Moovit: Bus & Train Schedules (Version 5.119.0.581) [Mobile application software]. Retrieved from https://play.google.com/store/apps/details?id=com.tranzmate	Prasarana Malaysia Berhad (2020). MyRapid PULSE (Version 3.0.4) [Mobile application software]. Retrieved from https://play.google.com/store/apps/details?id=com.prasarana.pulse
Advantages	1) Maps and station information are accessible offline even without an internet connection. 2) Offers route planning and rail timetables. 3) Simple navigation and an intuitive user interface.	1) Current bus and rail timetable updates. 2) Planning a multimodal journey with a variety of route alternatives. 3) Detailed instructions and real-time departures. 4) Service notifications and user-generated content.	1) Up-to-date arrival, departure, and service interruption information for trains and buses. 2) Careful route planning with a variety of possibilities. 3) Integrated transit services as well as supplementary elements like Park and Ride lots.
Disadvantages	1) There are no real-time updates on modifications or delays to trains. 2) Limited extra features beyond the bare minimum schedules and station information.	1) Because of the app's extensive feature set, some users could find it overwhelming. 2) Infrequent errors in real-time updates.	1) Occasional problems and technical issues reported by users. 2) Some users might think the UI of the app is a little crowded.

2.4 Conclusion

In conclusion, commuters and tourists can benefit from the examined mobile applications for train station information. Each app offers special benefits and features to improve the user experience and offer easy access to information about public transit.

Due to its offline capability, which enables users to view maps and station information without an internet connection, the "Malaysia MRT/LRT (Offline)" app stands out. It does not, however, provide real-time information on modifications or delays to trains.

When it comes to real-time updates, multimodal travel planning, and detailed directions, "Moovit: Bus & Train Schedules" excels. It provides a wide range of functions, such as service notifications and user-generated content. Due of the app's broad features, some users could find it overwhelming, and there occasionally seem to be inconsistencies with real-time updates.

The real-time updates, route planning and integrated transportation services of "MyRapid PULSE" are impressive. The software offers smooth service integration along with other features like Park and Ride options. Technical issues have been noted by some users, and the app's user interface could seem a little crowded.

CHAPTER 3: METHODOLOGY

3.1 Introduction

Agile Software Development has been selected for this project to direct the system prototype and entire development process through each stage, as illustrated in Figure 3.1. The iterative and flexible agile methodology places a strong emphasis on teamwork, adaptation, and continual progress. It is a reaction to the constraints of conventional, linear project management approaches, which frequently find it difficult to adapt to changing requirements and changing client wants. Agile's iterative methodology enables ongoing input and modifications. The developer does a retrospective at the end of each sprint to evaluate the procedure and pinpoint areas that might use improvement in following iterations. Developers are better able to adapt to changing requirements, deal with problems as they arise, and constantly improve the product to satisfy stakeholders' shifting expectations thanks to this continuous feedback loop. The agile method seeks to support software development practises where modifications are permitted at any time and unavoidable biological process alterations may be detected [8].

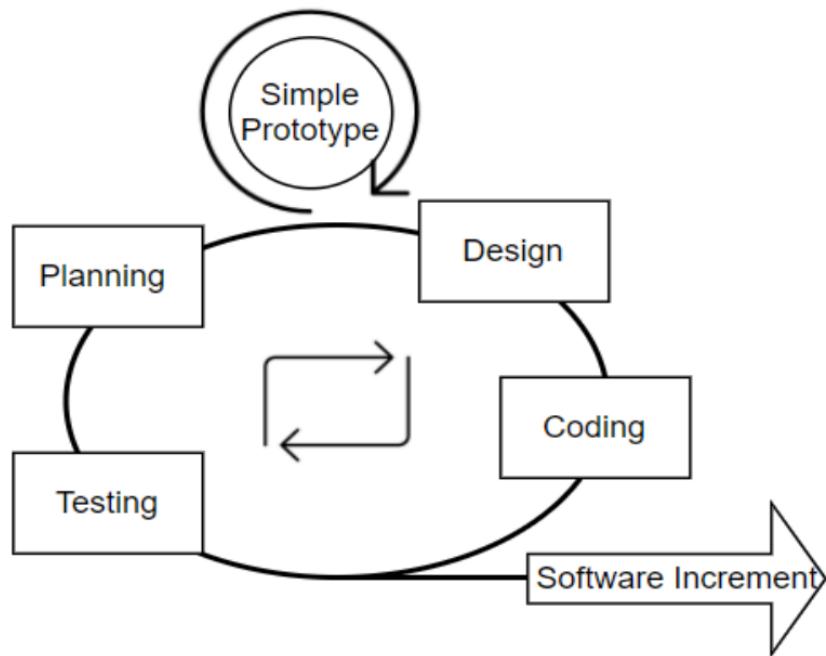


Figure 3.1 Phases in Agile Development Lifecycle

The framework's emphasis on providing clients with high-quality software while expediting the development process has made it well-known in agile [9].

3.2 Agile Software Development Life Cycle

3.2.1 Planning

The first phase in the Agile Methodology is planning, which entails establishing and understanding the requirements and expectations of the stakeholders and users. This stage is essential for determining the project's course and making sure the finished product corresponds to the planned results.

The planning stage is akin to the preliminary study while creating a mobile application for railway station information. It entails doing in-depth research and analysis to get a complete knowledge of the current infrastructure, services, and information requirements of commuters and travellers at train stations.

3.2.1.1 Preliminary Study based on Train Station Information Mobile Application

In order to get input on the mobile application for railway station information, a survey has been created using Google Form. This survey is intended to aid in the completion of the project, and the information gathered will be analysed to produce the mobile application. 28 people participated in this survey in total. The chart below provides a statistical breakdown of their reaction.

There were two components to the survey. While Section B presents some satisfaction scale questions, Section A focuses mostly on demographic questions. Starting with Section B, Question 3 as shown in the Figure 3.2, reveals that 35.7% of respondents is dissatisfied with the information available about the facilities in the train stations.

How satisfied are you with the information about the facilities within train stations? (e.g. information about types of food vendors, ATMs and directory)

28 responses

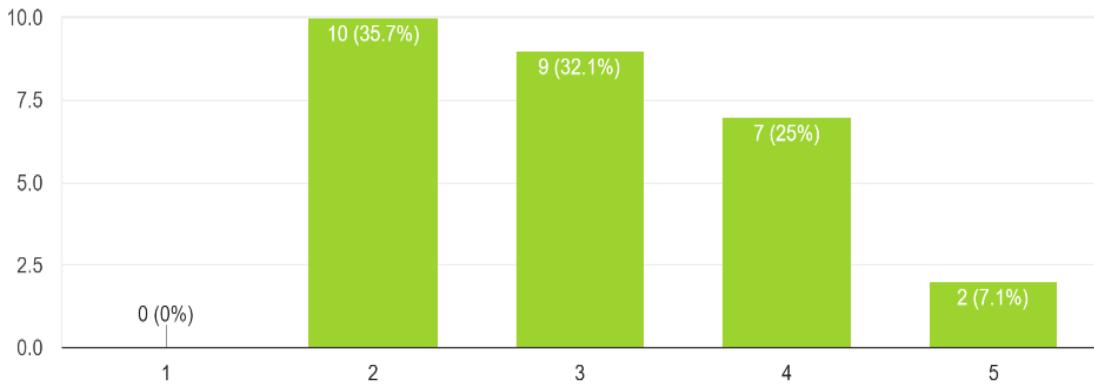


Figure 3.2 Response for Question 3

Moving on to Question 5, Figure 3.3 demonstrates that 78.6% of all respondents think a mobile app for railway stations is significant. This is due to the fact that the great majority of individuals regularly access information through their mobile phones.

Do you think it is important to have a mobile app that provides information about the facilities and services available at each train station? (e.g. info... about types of food vendors, ATMs, and directory)

28 responses

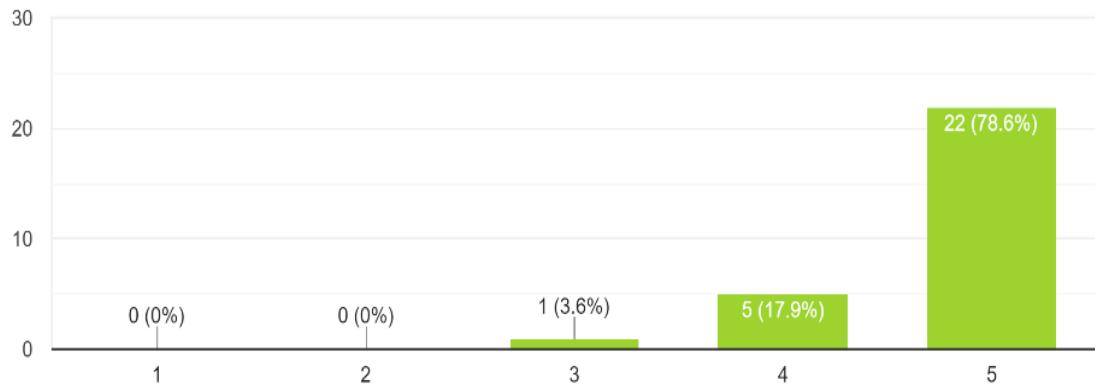


Figure 3.3 Response for Question 5

Train routes, platform information, and directories ranked as the top 3 most desired features by respondents when asked about the specific aspects they would love to explore in the suggested applications. These elements are consistent with the goals of the proposed application. This can be viewed in Figure 3.4 below.

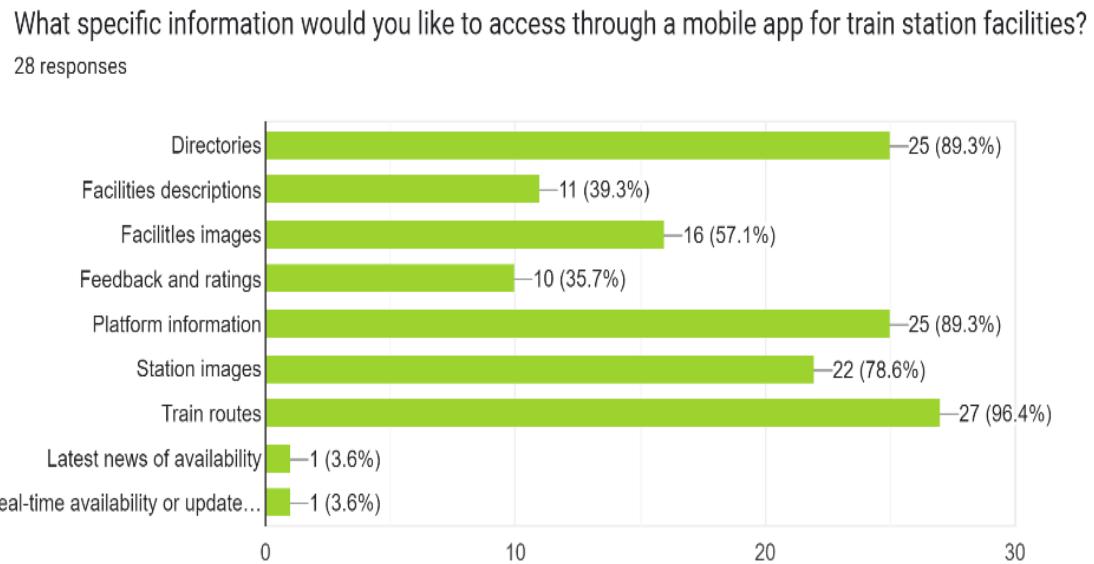


Figure 3.4 Response for Question 6

Figure 3.5 below indicates that 53.6% of respondents concur that a feedback system in the app is necessary to obtain user input. The operators of train stations can improve the quality of their services by using this function.

How important it is to have a feedback system in the train station information mobile app?
28 responses

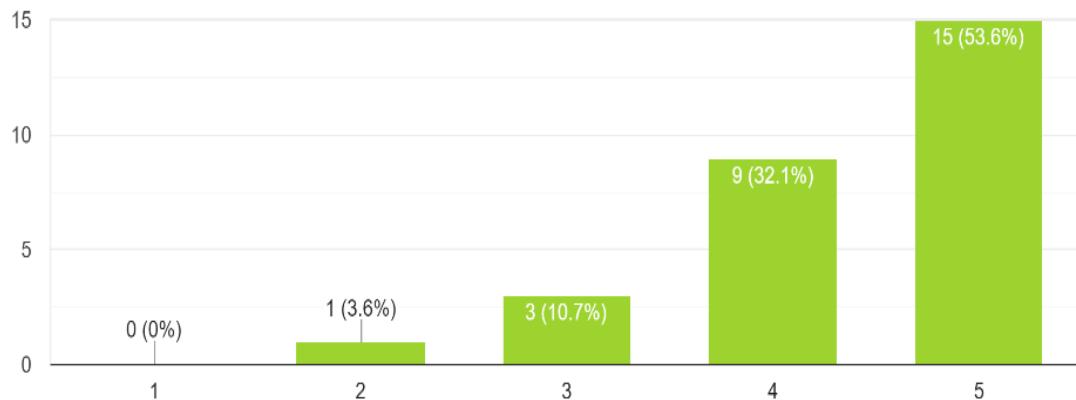


Figure 3.5 Response for Question 8

In the survey, an open-ended question was included to gather respondents' input on any additional features they would like to see in the train station information mobile app. This question encourages participants to share their creative ideas and suggestions beyond the predefined options, allowing for the exploration of innovative features that may enhance the overall user experience. According to the findings in the responses, some of the suggestions made by the respondents include a feature to notify them right once if there are any construction projects affecting the routes or the most recent information about availability or unavailability routes. Along with live updates on the position of the train, they also proposed adding a function that would let users know whether it was possible to top up their Touch 'n Go cards at a certain stop.

3.2.1.2 Conclusion for Preliminary Survey

By conducting a thorough requirement analysis, the project can gather valuable insights and establish a solid foundation for the development of the train station information mobile application. This phase helps to identify the key features, functionalities, and user experience requirements that should be prioritised during the subsequent stages of development. It ensures that the final product addresses the specific needs and expectations

of the stakeholders and users, leading to a more successful and user-centric mobile application.

3.2.2 Design

The design phase of the Agile approach is concerned with converting the obtained requirements into a clear system architecture and user interface design. To visualise the suggested solution and promote communication among the developer and stakeholders, design artefacts and prototypes are created. The basic mock-ups for the Train Station Information Mobile Application are displayed in Figure 3.6 and Figure 3.7 below.

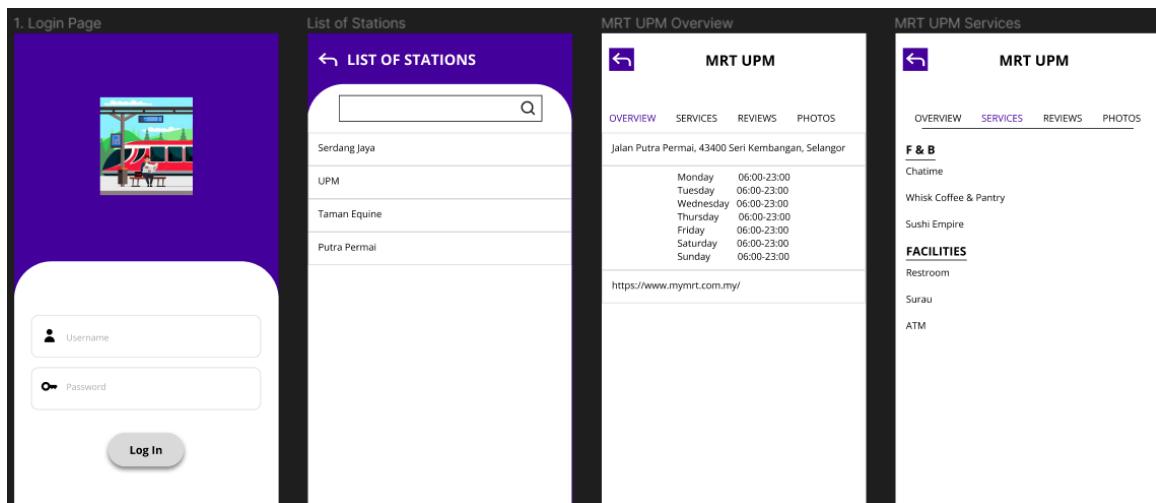


Figure 3.6 Basic Mock-up of the Mobile Application (i)

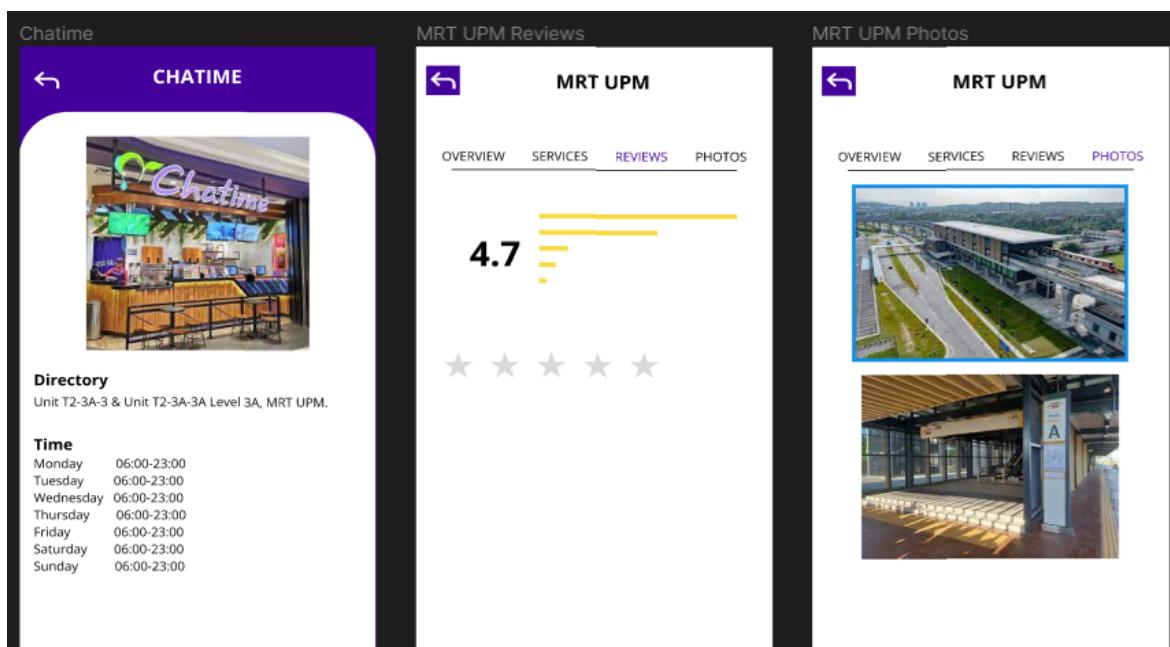


Figure 3.7 Basic Mock-up of the Mobile Application (ii)

3.2.3 Coding

In the Coding stage represents a crucial phase where the envisioned features and functionalities start taking shape through both front-end and back-end development processes. The front-end development focuses on crafting the user interface and ensuring a seamless, intuitive experience for the end-users. This involves the design and implementation of the visual elements, ensuring the application's user-centricity. Concurrently, the back-end development involves creating the underlying infrastructure, databases, and server-side logic to support the application's functionality. Both aspects of coding work in tandem, with continuous collaboration and iteration, aligning with Agile principles to deliver incremental and iterative progress throughout the development lifecycle. This approach allows for adaptability, quick feedback loops, and ensures that the evolving software meets the changing needs and expectations of stakeholders.

3.2.4 Testing

For the Testing stage encompasses a comprehensive approach involving design testing, development testing, and user acceptance testing (UAT). Design testing ensures that the initial design specifications align with the intended functionalities, identifying potential issues early in the development process. Development testing, on the other hand, involves thorough examination of the code to detect and rectify bugs, ensuring the reliability and stability of the software. Finally, User Acceptance Testing serves as the final validation phase, allowing end-users to interact with the system and verify that it meets their expectations and requirements. This iterative testing approach, integrated throughout the development cycle, ensures the early identification and resolution of issues, promoting a more resilient and user-approved end product.

3.3 Tools and Languages

This subsection details the array of tools, software, services, and programming languages leveraged in the development of the Train Station Information Mobile Application.

3.3.1 Hardware

The project was executed on a computer with specifications outlined in Table 3.1 below. The hardware specifications played a crucial role in supporting the development environment and ensuring optimal performance during the implementation phase.

Table 3.1 Hardware Specifications

Hardware Specification	Model
Processor	Intel(R) Core (TM) i5-10300H CPU @ 2.50GHz
RAM	8.00 GB
GPU	Intel(R) UHD Graphics and NVIDIA GeForce GTX 1650 Ti
SSD	WDC PC SN530 SDBPNPZ-512G-1014

3.3.2 Software Tools

- Android Studio**

The development process of MyStation involves the use of Android Studio as the chosen Integrated Development Environment (IDE) for native mobile app development. Android Studio provides a comprehensive platform equipped with tools and features specifically tailored for Android app creation. Its robust capabilities and compatibility with the Android operating system make it an ideal choice for ensuring seamless development and optimal performance of the mobile application.

3.3.3 Framework

- Flutter**

Flutter, developed by Google, is a UI toolkit that facilitates the creation of natively compiled applications for mobile, web, and desktop from a single codebase. Its distinctive feature lies in its ability to provide a consistent and visually appealing user interface across different platforms. By leveraging the Flutter framework, the application ensures a streamlined development process, enhanced user interface design, and broader accessibility for Android users.

3.3.4 Programming Languages

- **Dart**

The programming language employed for the cross-platform development is Dart, integrated with the Flutter framework. Dart's suitability for building cross-platform applications, coupled with its object-oriented features and strong support for reactive programming, aligns seamlessly with the requirements of the project. The synergy between Dart and Flutter enables efficient coding, offering a robust solution for creating a responsive and dynamic mobile application that meets the diverse needs of Android users.

3.3.5 Server

- **Firebase**

For the backend functionalities, Firebase services will be employed. Firebase, a comprehensive mobile development platform provided by Google, offers a range of features crucial for the app's seamless operation. This includes real-time data storage, user authentication, and dynamic updates. By utilising Firebase as the server for the application, the development process is streamlined, ensuring efficient data management, secure user authentication, and the integration of real-time updates to enhance the user experience.

3.3.6 Features

User registration and authentication mechanisms will be implemented to ensure secure and personalised interactions within the app. The application will provide comprehensive station information, including details about facilities, services, and real-time updates sourced from crowd-generated data. This approach not only empowers users with accurate and up-to-date information but also fosters a community-driven platform where commuters can contribute to the collective knowledge and improvement of the train station environment.

3.4 Project Timeline

The Gantt Chart, illustrated in Figure 3.8 and Figure 3.9 below, outlines the schedule for the various stages and tasks involved in the development and completion of MyStation Mobile Application.

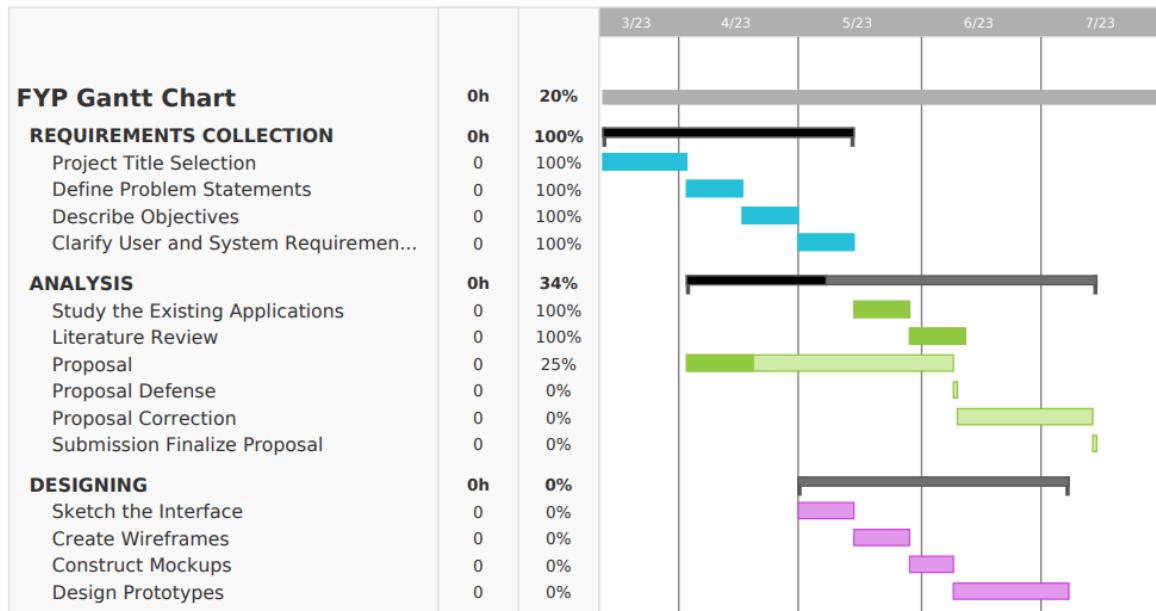


Figure 3.8 Gantt Chart FYP A

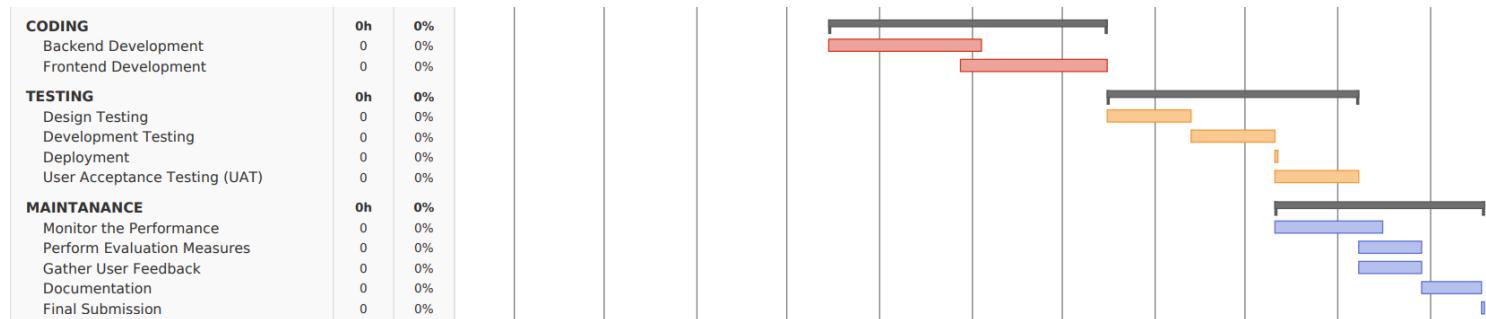


Figure 3.9 Gantt Chart FYP B

CHAPTER 4: SYSTEM ANALYSIS AND DESIGN

In this chapter, we delve into the intricacies of system analysis and design for the Train Station Information Mobile Application. The comprehensive analysis encompasses the overall system framework, interface design, flowcharts, data flow, sequence diagrams, use cases, and entity-relationship diagrams. The overall framework elucidates the general processes inherent in the system, providing a bird's-eye view of its architecture. Interface design outlines the strategic placement of graphical elements within the system interface, ensuring an intuitive and user-friendly experience. The flowcharts delineate the step-by-step process flow of the system, offering a systematic understanding of its operational dynamics. This chapter serves as a crucial bridge between conceptualisation and implementation, laying the groundwork for the subsequent phases of development.

4.1 Requirement Specification

This section serves to articulate the blueprint for constructing the application, encompassing the conceptualisation, design, and flow of the system. The project requirements were systematically gathered through a preliminary study conducted via a Google Form, engaging 28 participants to ensure a comprehensive understanding of the user expectations and needs.

4.1.1 User Requirements

The user requirements for the proposed MyStation mobile application revolve around three key aspects. Firstly, the app should feature a robust reporting and feedback system, enabling users to actively contribute real-time information, share experiences, and address concerns within the station environment, fostering a dynamic and community-driven platform. Secondly, the application must provide users with access to a rich visual representation of train station facilities and services, incorporating photos and detailed descriptions to empower users in making informed travel decisions. Lastly, the user interface should be intuitively designed to facilitate easy navigation, ensuring users can swiftly locate the desired information and contribute to a seamless and convenient user experience. In essence, these user requirements aim to create a user-centric app that not

only addresses current challenges but also enhances overall user satisfaction and engagement.

4.1.2 Functional Requirement

The functional requirements focus on the functionality and tasks that the system is expected to perform to fulfil user needs and meet business objectives. The functional requirements of Train Station Mobile Application are:

- The app should provide a rich interface for accessing station information.
- The app should integrate with data sources to fetch and display station details, including maps, photos, and descriptions
- The app should provide real-time updates on station information, such as changes in user reports or feedbacks.

4.1.3 Non – Functional Requirements

Non-functional requirements, also known as quality attributes or system qualities, specify the desired characteristics of a system that are not directly related to its functionality. The non - functional requirements of Train Station Mobile Application are stated below:

- The app should be responsive and provide fast loading times to ensure a smooth user experience.
- The app should have an intuitive and user-friendly interface, making it easy for users to navigate and find the desired information.
- The app should be compatible with Android mobile platforms and different screen sizes and resolutions.

4.2 System Design

The system design phase delineates the anticipated flow of the application. This involves the creation of two comprehensive flowcharts, one for user interactions and another for administrative processes. These flowcharts collectively depict the entirety of application features and operations from both user and admin perspectives.

4.2.1 User Flowchart

The user flowchart, illustrated in Figure 4.1 below, delineates four primary options available within the main menu of the application.

The first option, "Overview," enables users to access and review comprehensive details pertaining to the selected station details. This page displays the images of the station and some general information such as the operating hours and the status if it is open or close, some buttons that navigate to calling the station's operator, Google Map and the station's website. The second option directs users to the "Services" page, where all of the outlets available at the station is displayed, encompassing details regarding each outlets' images, directories, operating hours and feedbacks.

The third option, "Profile," facilitates the presentation of user's profile page. This page shows the user's name, email, role and user ID. The final option leads users to the "Station Reports" page, initiating a multi-step process. Here, users are prompted to view all of the reports submitted by other users about that station. If they find those reports to be relevant, they can like the reports and dislike if otherwise. They will see an icon button that allows them to upload their own reports to be displayed. Upon clicking the button, a bottom sheet will appear prompting the user to choose the type of report, rating and report details. The intricacies of each process within the flowchart are further expounded upon in the following sections.

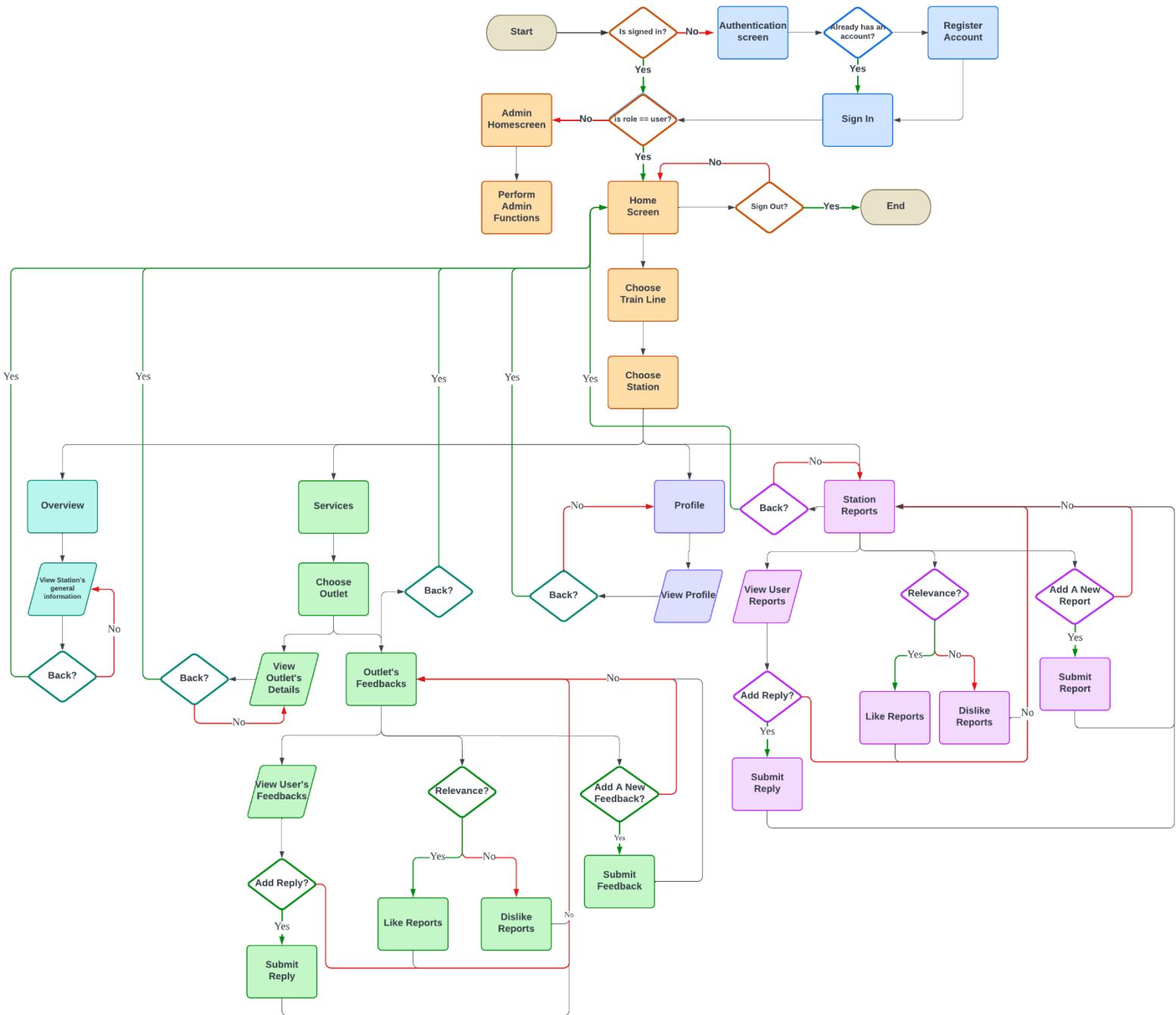


Figure 4.1 User Flowchart

4.2.2 Admin Flowchart

The admin flowchart, depicted in Figure 4.2 below, mirrors the structure of the user interface but introduces distinct functionalities tailored to administrative responsibilities.

The first option, "Overview," provides administrators with an insight into station details. This page allows admins to upload and remove the station's images in the database directly from the app. The second option navigates administrators to the "Services" page, a comprehensive display of all outlets within the station. This section provides essential function for admins to add and remove every outlet's image and filter out any feedbacks irrelevant to those outlets from the app.

The third option, "Profile," leads administrators to their dedicated profile page. Here, administrators can review and manage personal details such as their name, email, role, and user ID. The final option directs administrators to the "Station Reports" page, where they can perform the same functions as other users with additional feature to remove any irrelevant comments or reports from the page.

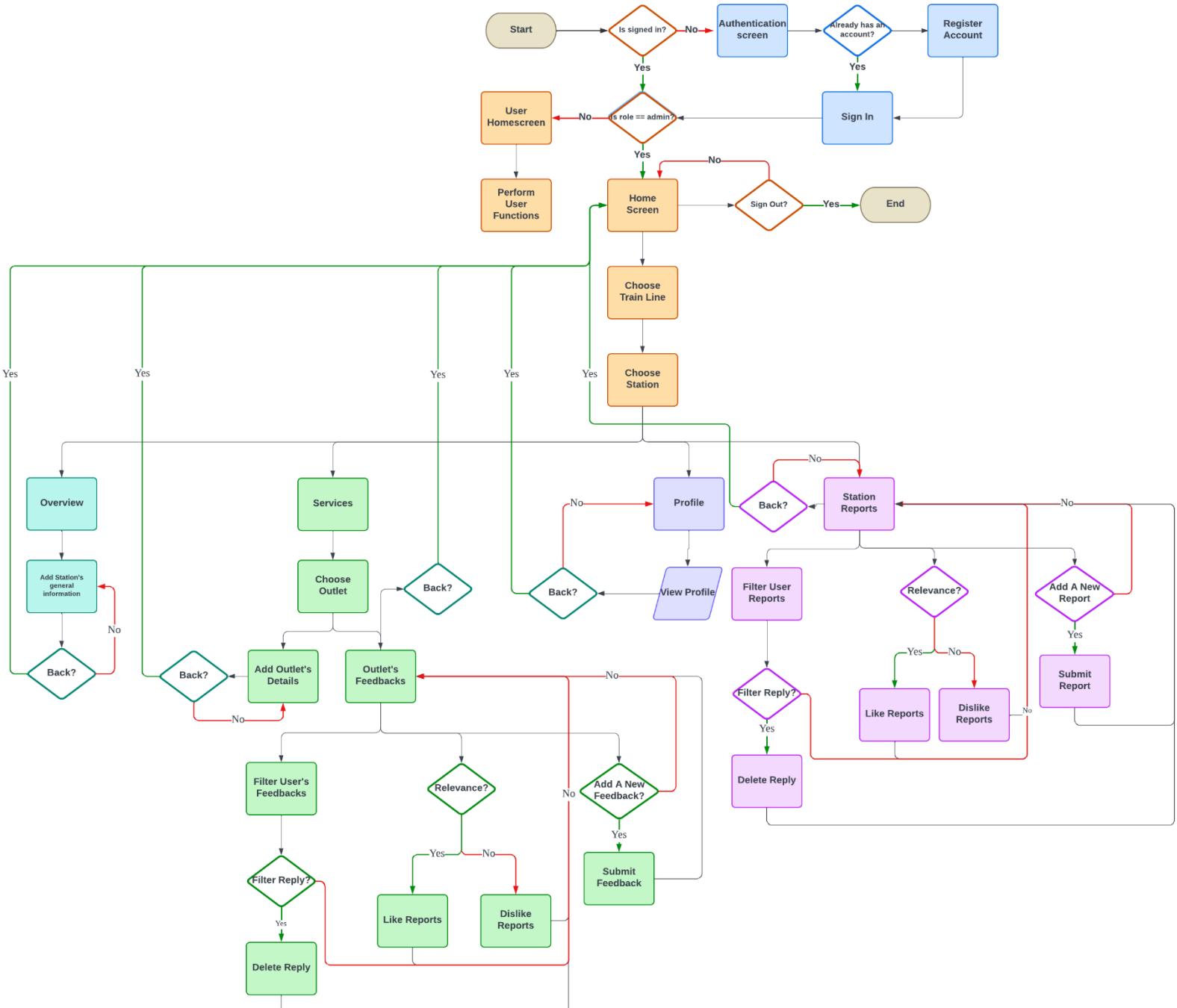


Figure 4.2 Admin Flowchart

4.3 Data Flow Diagram

The Data Flow Diagram (DFD) serves as a graphical representation of the information flow within the MyStation application, employing the Yourdon and Coad system of symbols. This diagram encapsulates the key processes, data stores, and external entities, elucidating the system's information architecture.

The external entity, depicted as the "User Interface," symbolises the point of interaction between the application and its users. This external entity is crucial for user input and output interactions, facilitating seamless user engagement. The central data store employed in the system is Firebase, a cloud-based database service. Firebase acts as the repository for crucial data, storing information pertaining to user profiles, station details, feedback, and reports. The processes within the DFD encapsulate distinct functionalities crucial to the application's operation. The processes include "Authenticate Users", "Manage Train Station Information", "Retrieve Train Station Information", "Manage User Feedback and Reports" handles the bidirectional flow of user-generated content. In parallel, the bidirectional data flow between "Manage User Feedback and Reports" and "Store User Feedback and Reports" illustrates the interaction between the application's internal processes and Firebase.

The "Admin Functions" process encompasses functionalities exclusive to administrators. Bidirectional data flows exist for processes like "Delete User Reports," "Create New Services," and "Upload Images." These features empower administrators to interact with and modify the system's data in real-time.

In essence, the DFD offers a holistic visualisation of the information flow, showcasing the symbiotic relationship between the user interface, internal processes, and the Firebase data store within the MyStation application. Figure 4.3 below shows the data flow diagram of the application according to the requirements.

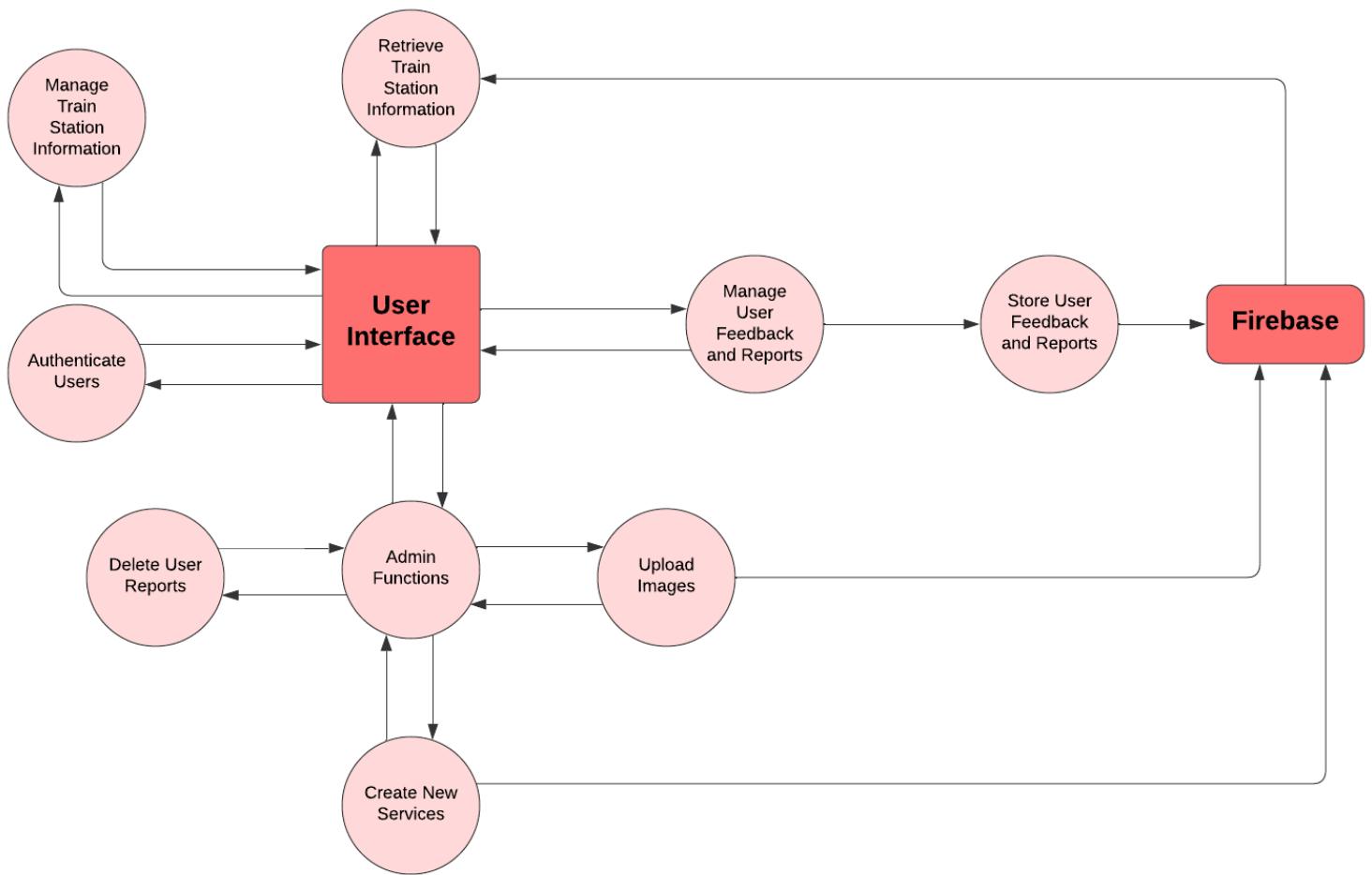


Figure 4.3 Data Flow Diagram

4.4 Sequence Diagram

The Sequence Diagram provides a dynamic representation of the interactions and order of events within the MyStation application, offering insights into the flow of control between objects and processes. In this depiction of system behaviour, the interactions between various components are elucidated chronologically.

Based on the figure 4.4 below, the diagram commences with the “User” as the initiating actor, symbolising the point of entry for user interactions. Upon initiating the application, the sequence unfolds as “User” initiating a request to “Authenticate Users,” which collaborates with “Firebase” to verify credentials, granting access upon successful authentication. For retrieving station information, the user prompts the “Train Station

App,” leading to interaction with the “Retrieve Station Information” process and Firebase, presenting real-time details on the interface. Feedback submission follows a user-triggered process via the “User,” engaging the “Save User Reports” process to store feedback in Firebase and confirm the submission. Admin functionalities, such as deleting reports or uploading images, involve bidirectional communication between the “Admin” actor and “Firebase”, ensuring seamless real-time updates facilitated by the “Train Station App.”

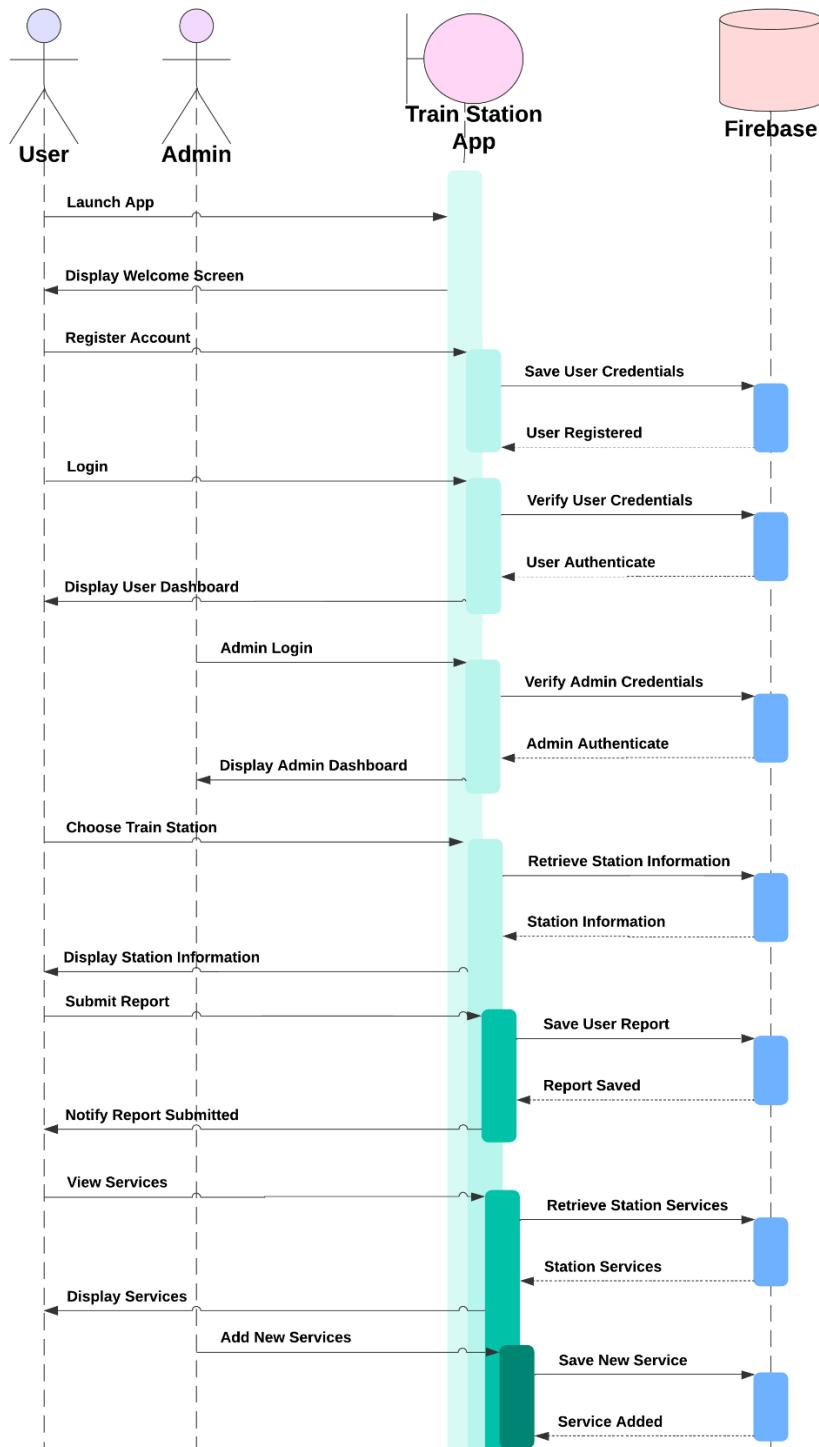


Figure 4.4 Sequence Diagram

4.5 Use Case Diagram

Figure 4.5 below indicates how the use case diagram for the MyStation application delineates the interactions between key actors, including "User" and "Admin," and the system itself. On the left side, users can engage in various use cases such as authentication, view images, reports, profile, like/dislike reports, add report and add reply. Admins, on the other hand, have specific functionalities like “add and delete reports” and “add and delete images.” These actors interact with the system, represented on the right side, which encapsulates essential functionalities like authentications, view data and add data, and admin functions. The diagram illustrates the distinct roles and functionalities associated with both users and admins, showcasing the system's capabilities in facilitating a comprehensive station information experience.

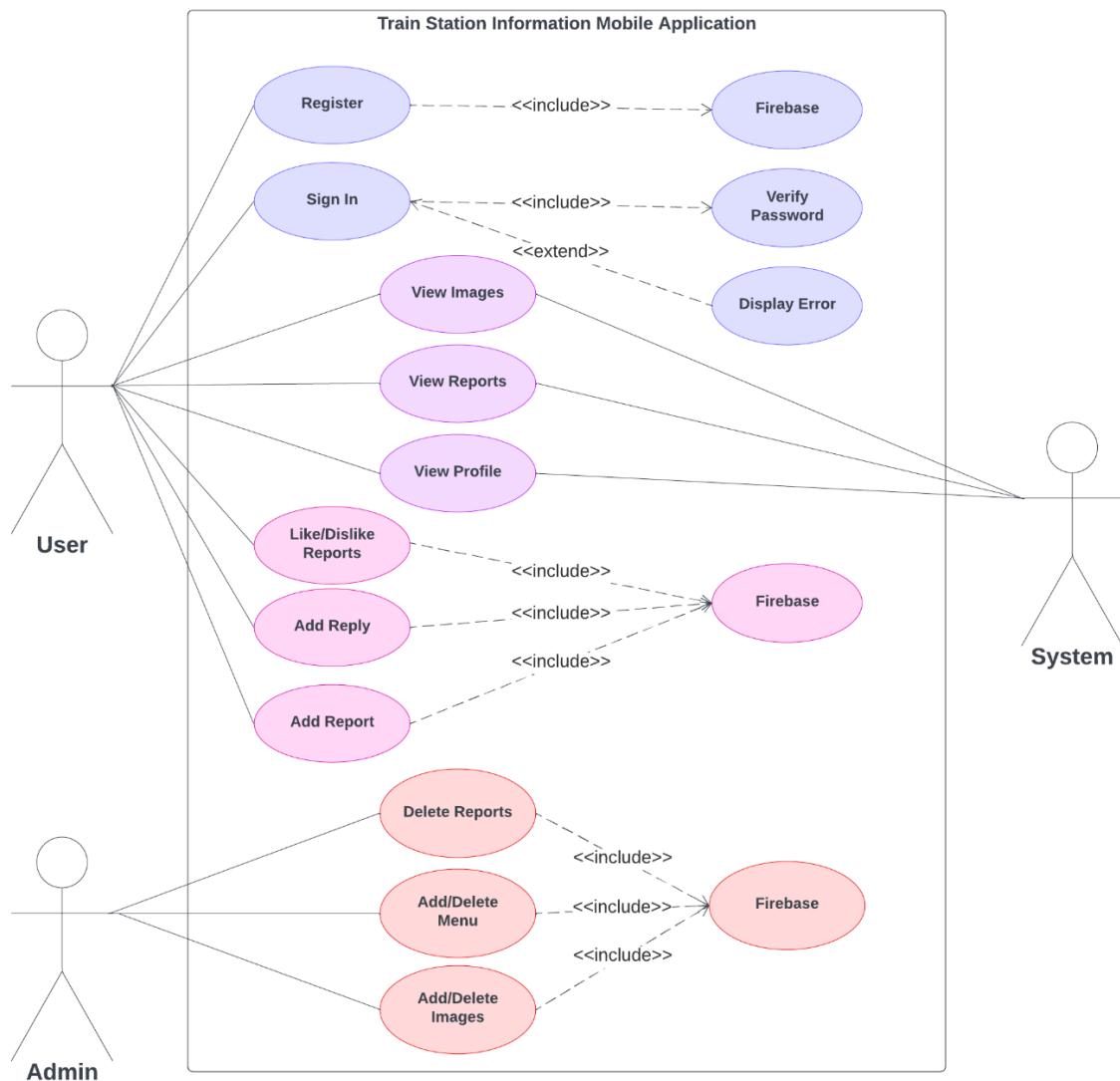


Figure 4.5 Use Case Diagram

4.6 Entity Relationship Diagram

The Entity Relationship Diagram (ERD) employed in the MyStation application serves as a pivotal tool for visualising the database structure and illustrating the interrelationships among various entities. Figure 4.6 showcases seven distinct entities within the database design: Report, User, TrainStation, Services, Feedback, Image and Admin. These entities are interconnected by relationships that signify how the attributes are related. For instance, the User entity links with Report and Feedback entities through userId, indicating the relationship between user information, user-generated report, and uploaded feedbacks. The TrainStation entity, representing detailed information about each station, is associated with the Services entity through stationId, encapsulating the relationship between stations and the services they offer. Admin, serving as a separate entity, is positioned to manage the overall system, underscoring the organisational structure of the database within the application. This ERD offers a comprehensive overview of the data model, ensuring efficient and organised data storage and retrieval.

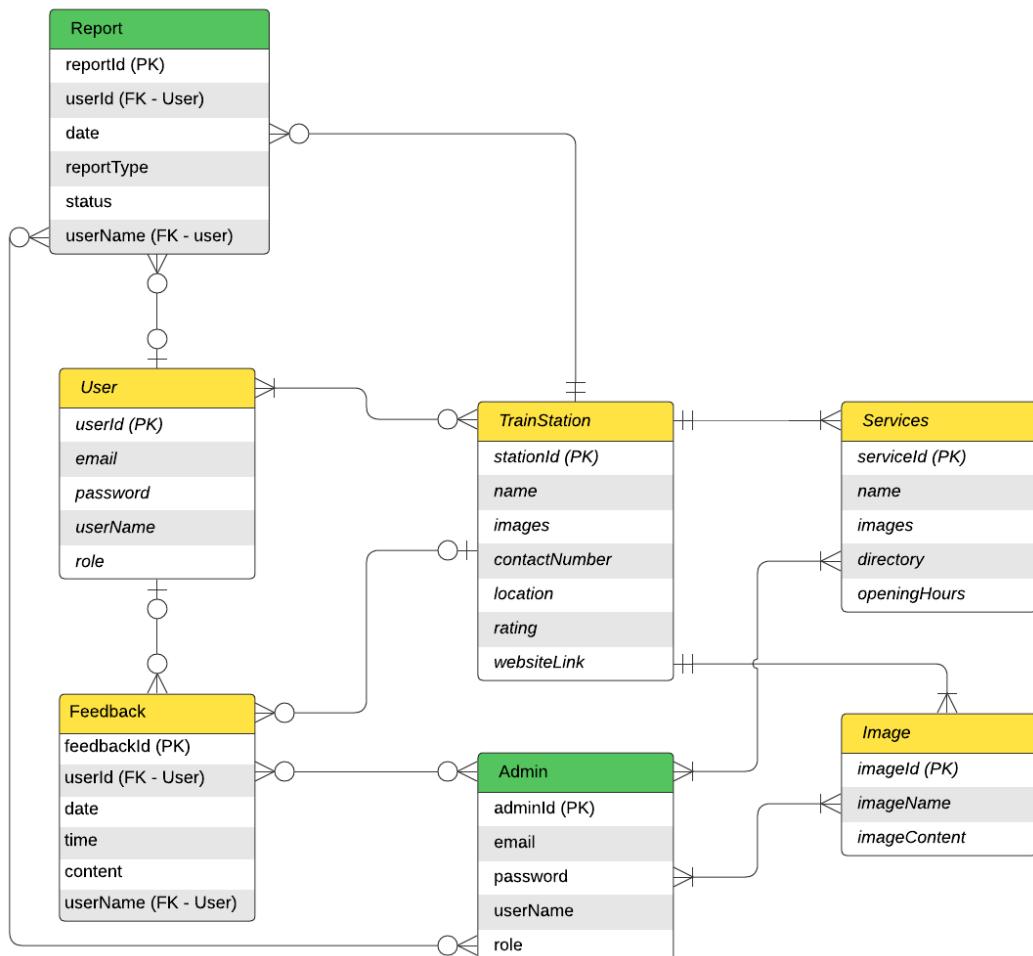


Figure 4.6 Entity Relationship Diagram

4.7 User Interface

4.7.1 User Site

Commencing with the authentication page, the user interface of the MyStation mobile application presents a streamlined and intuitive design, prioritising user experience. Upon successful authentication, users are greeted with a homepage housing a list of available train lines in Malaysia. When users tap on any of the listed train lines, the app will navigate them to a page that lists all the stations available for that line. By tapping on any of those stations, the app will navigate the user into a page with a bottom navigation bar consisting of four tabs. The tabs include “Overview,” allowing users to see the general information of that station. “Services” tab lets users to navigate and explore detailed station information, “Station Reports” providing access to user-generated reports and ratings for stations, and ‘Profile,’ enabling users to personalise their experience and view historical interactions. The navigation bar facilitates seamless movement between these sections, ensuring a user-friendly experience. Figure 4.7 shows the application's user interface adheres to high-quality standards, emphasising usability, simplicity, and consistency in design elements, fostering an engaging and efficient interaction for commuters and travellers alike.

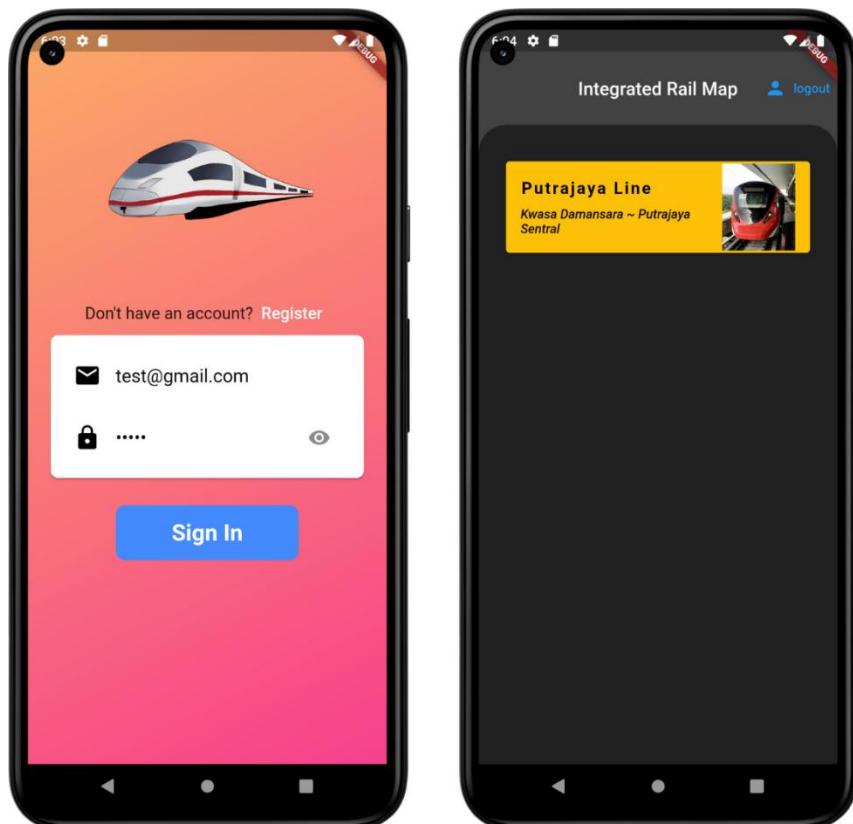


Figure 4.7 User Interface Part 1

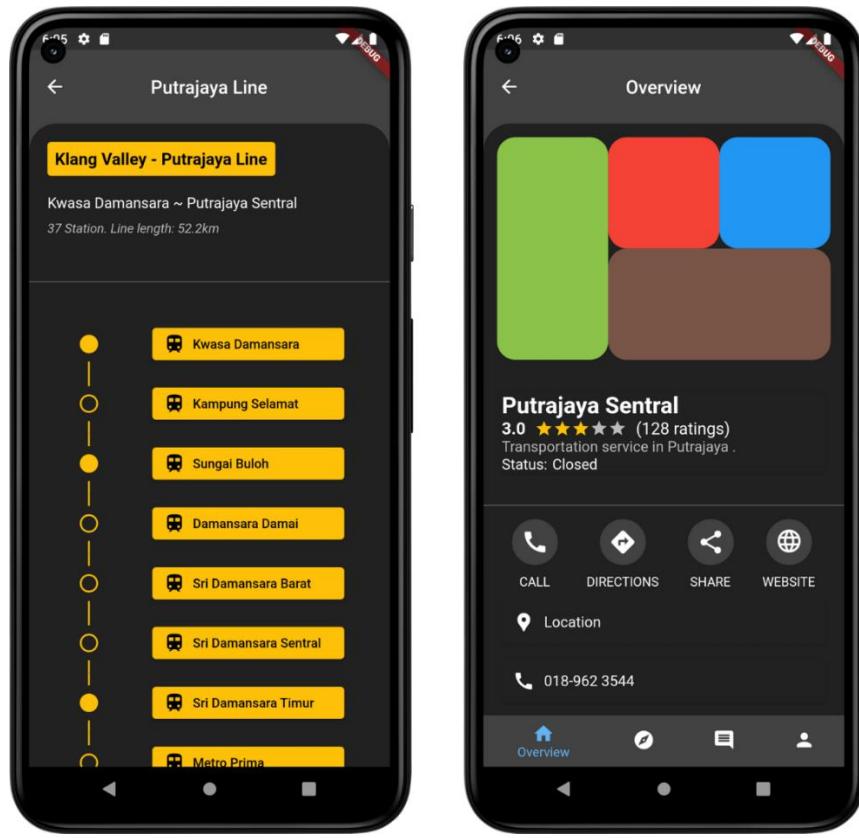


Figure 4.8 User Interface Part 2

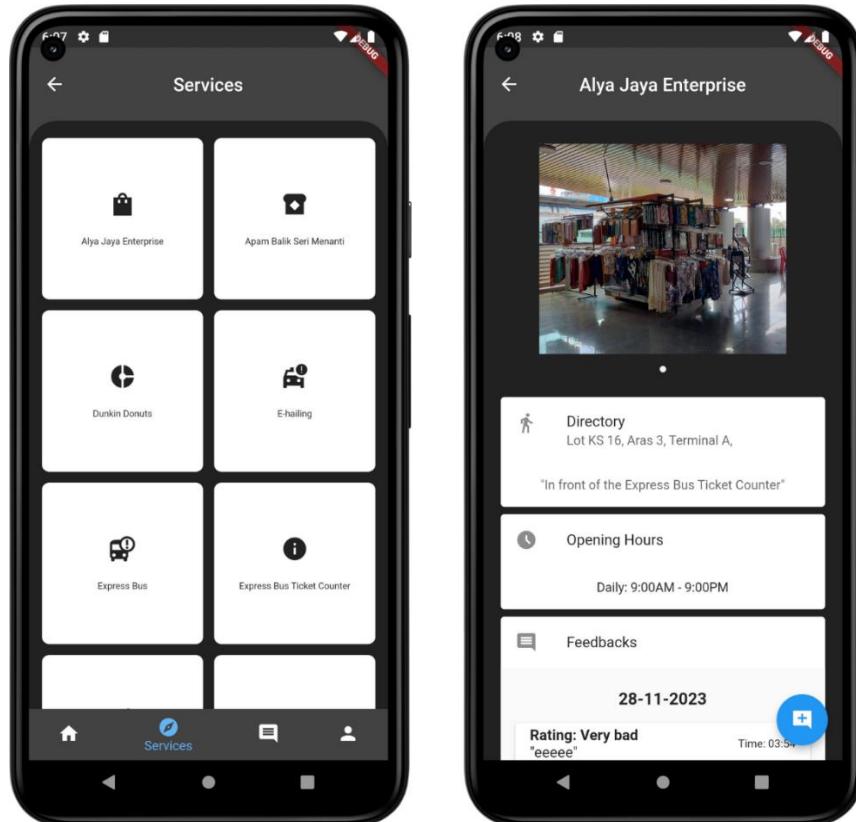


Figure 4.9 User Interface Part 3

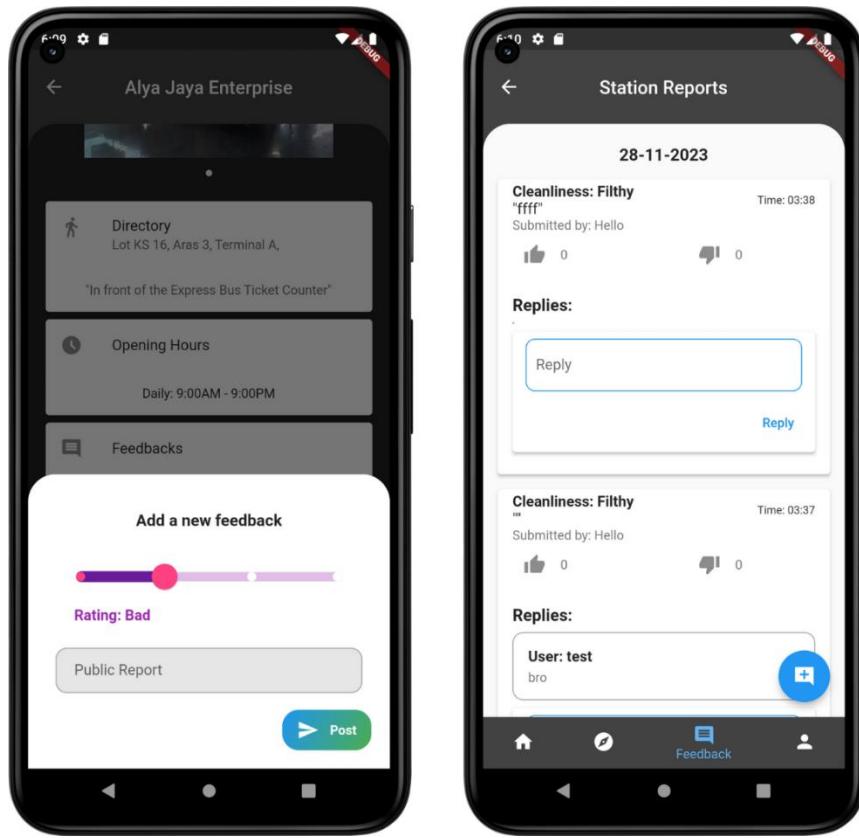


Figure 4.10 User Interface Part 4

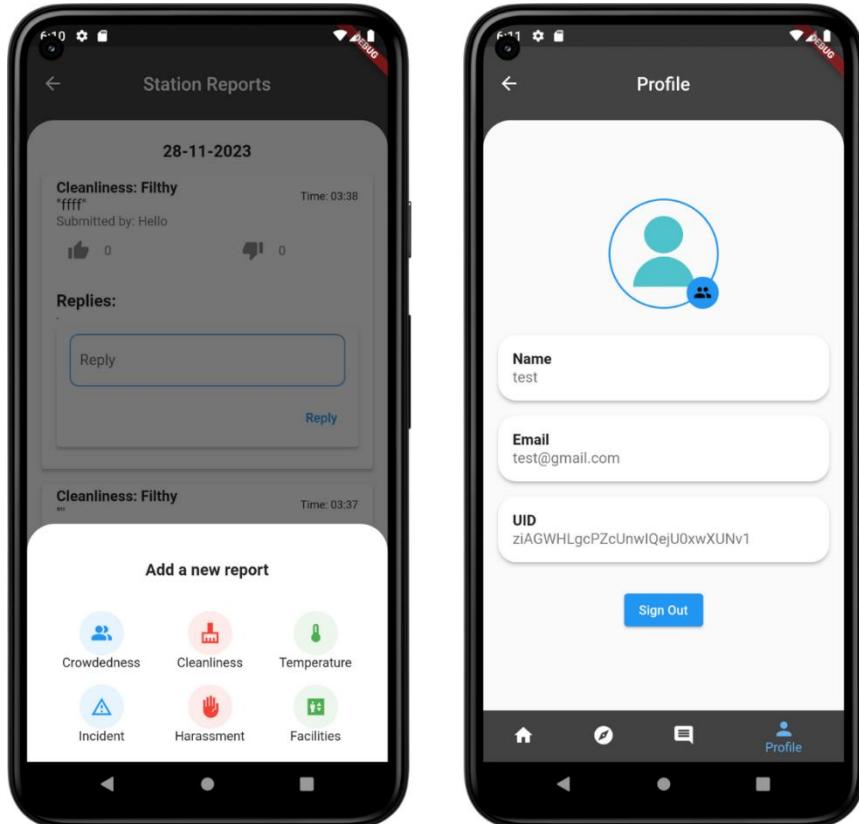


Figure 4.11 User Interface Part 5

4.7.2 Admin Site

The administrative interface of the MyStation application mirrors the user site, preserving its intuitive design for a consistent and seamless experience. The primary features, including “Overview,” “Services,” “Station Reports” and “Profile,” remain intact. However, the administrative dashboard incorporates additional functionalities, notably within the “Station Reports” tab. Admins have access to delete buttons, enhancing the platform's capacity to manage and filter user-generated comments effectively as depicted in Figure 4.8. This addition allows administrators to maintain a relevant and informative feedback section by removing irrelevant or unnecessary comments. The uniformity in design elements between the user and admin interfaces ensures familiarity, making it easy for administrators to navigate and utilise the extended functionalities seamlessly. This strategic integration promotes efficiency and coherence in the overall user and administrative interactions within the MyStation application.

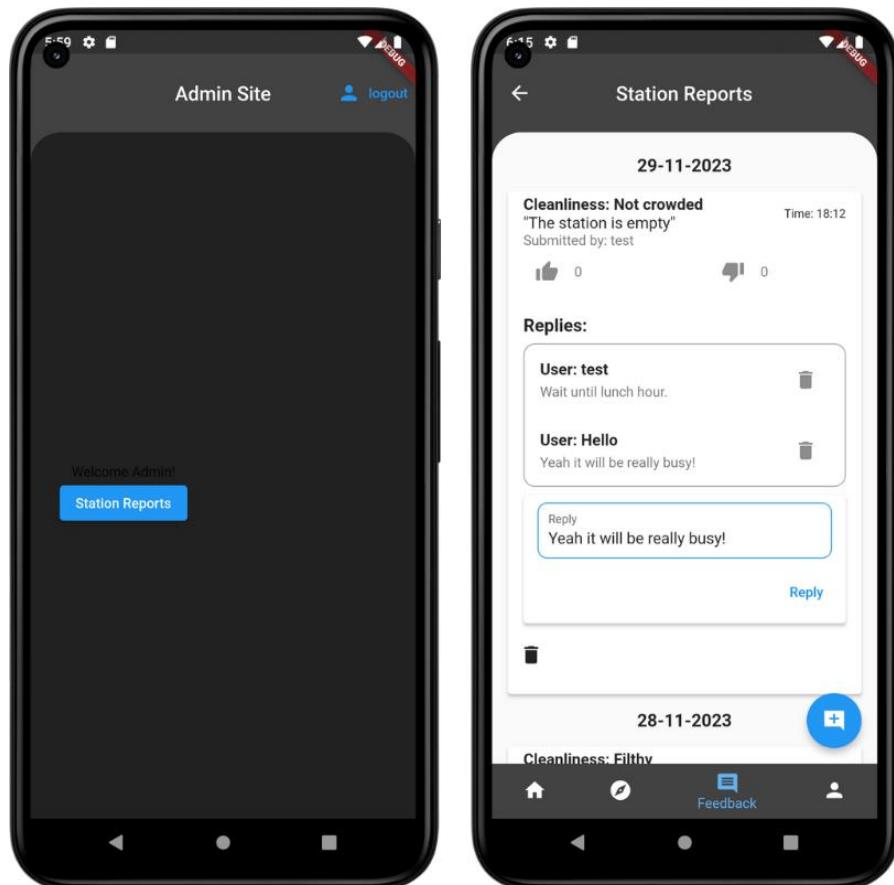


Figure 4.12 Admin Interface

CHAPTER 5: IMPLEMENTATION AND DISCUSSION

5.1 Development

The implementation phase of the MyStation mobile application involved leveraging powerful development tools and platforms to bring the project to fruition. In crafting the app's functionalities and user interface, the development process predominantly relied on the versatile capabilities of Android Studio, coupled with the Flutter framework for cross-platform mobile app development. Google Firebase emerged as a pivotal component, serving as the cloud-based database solution for storing and managing user information securely.

5.1.1 Android Studio

Android Studio serves as the primary development environment for crafting the MyStation mobile application. Tailored specifically for Android app development, Android Studio offers a comprehensive suite of tools and features to streamline the development process. With its built-in emulator, developers can preview their applications in real-time across various Android devices. Moreover, Android Studio provides robust syntax highlighting and autocomplete capabilities, powered by IntelliSense, to facilitate error detection and code completion. Throughout the development process, various Flutter packages were integrated into the project via the pubspec.yaml file, including essential packages such as Firebase for backend functionalities, Firebase authentication services allowing users to sign in with email/password, Image Picker to select images from the device's gallery or capture images using the camera, and rating bar that offers a customisable star rating bar widget. These packages enhance the application's functionality and user experience. Figure 5.1 below depicts the Android Studio IDE, showcasing the development environment utilised in building the MyStation app.

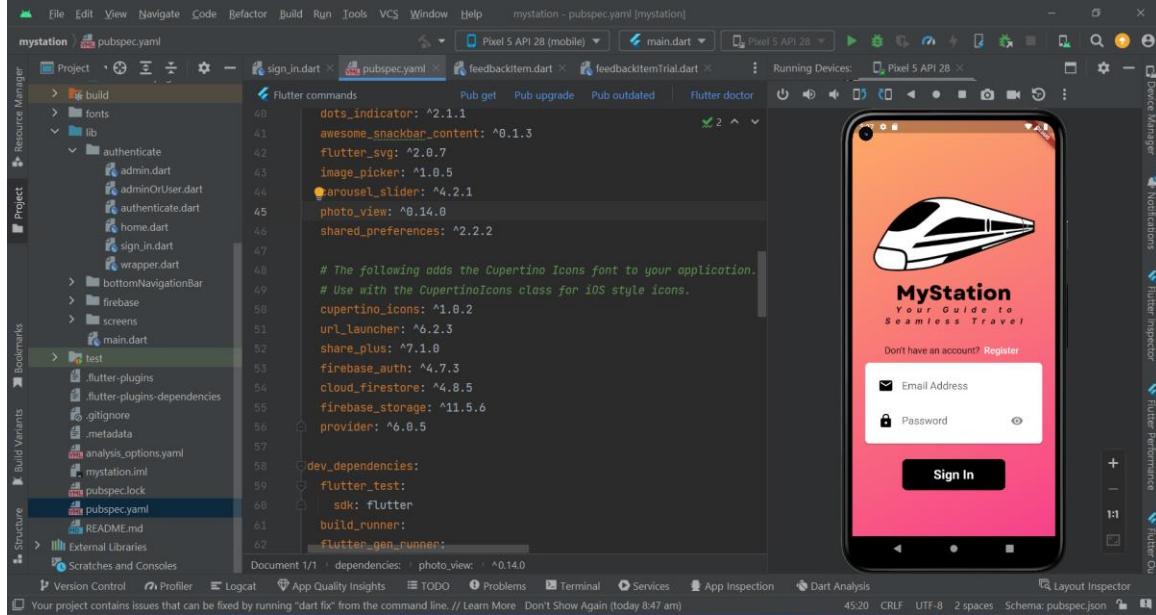
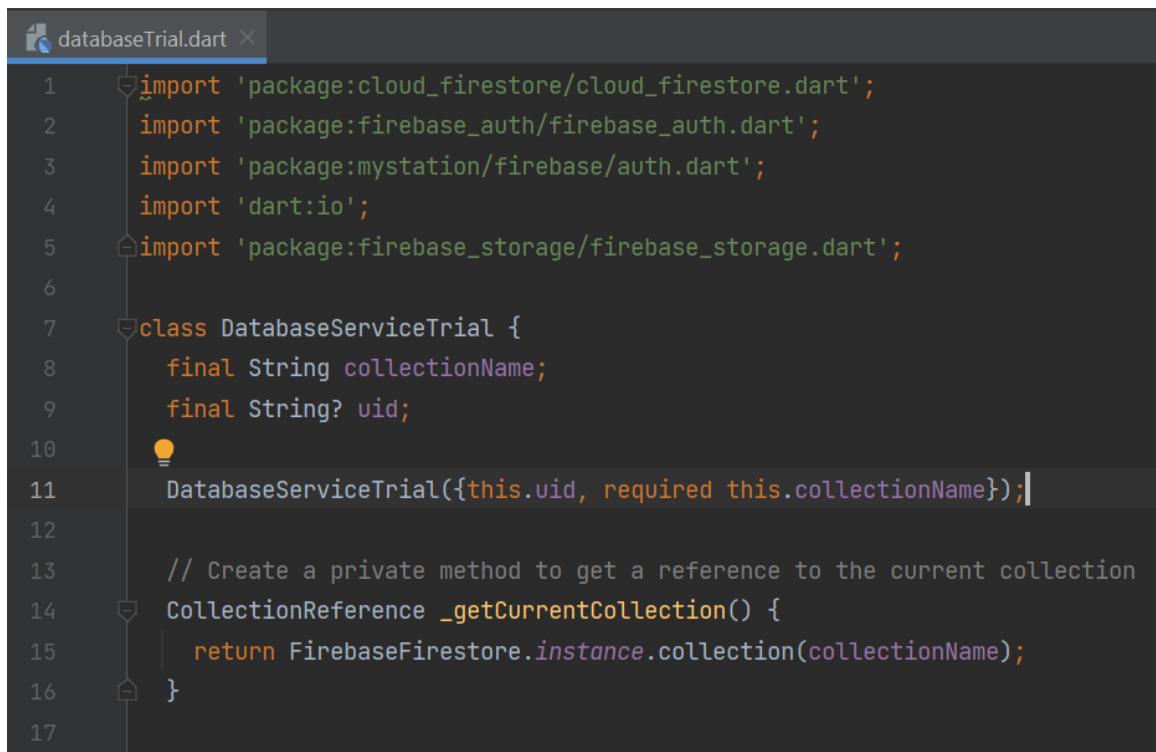


Figure 5.1 Android Studio and Emulator

Illustrated in Figure 5.2 is the code that defines a **DatabaseServiceTrial** class responsible for handling interactions with Firebase Firestore and Firebase Storage. It includes methods to submit feedback, submit reports, submit replies to feedback, retrieve replies for a given feedback, retrieve sorted feedbacks, increment and decrement like and dislike counts for feedbacks, retrieve feedback details, edit feedback and reply text, delete feedback and replies, upload images to Firebase Storage, store image URLs in Firestore, retrieve image URLs, and delete images from Firebase Storage and Firestore. The class utilises Firebase services for authentication, database storage, and cloud storage, allowing users to interact with the application by submitting feedback, reports, and replies, as well as managing images. Additionally, it implements user roles to enforce permissions for deleting feedback and replies, ensuring appropriate data management and security within the application.

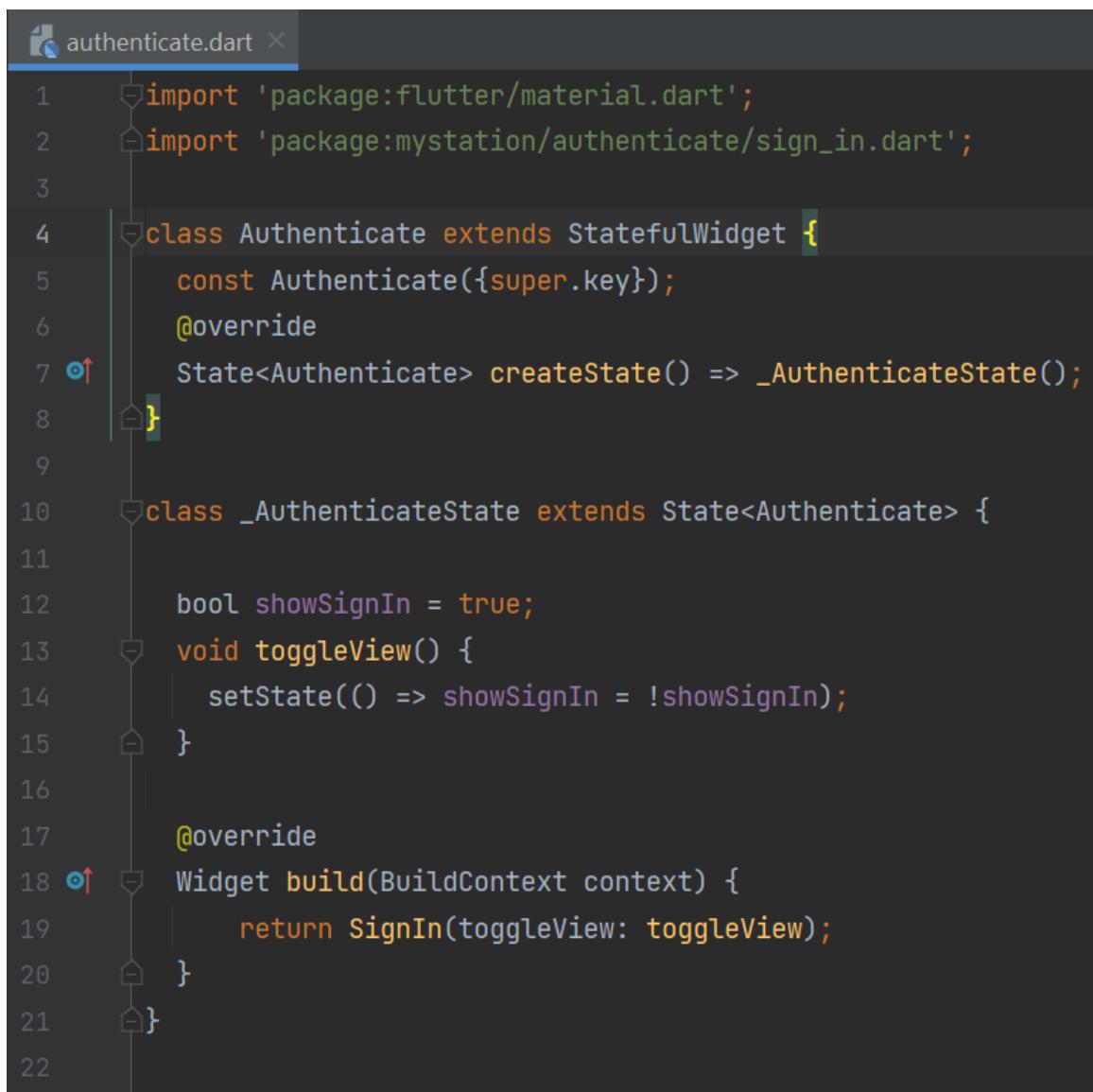


The screenshot shows the code editor in Android Studio with the file `databaseTrial.dart` open. The code defines a class `DatabaseServiceTrial` with imports for `cloud_firestore`, `firebase_auth`, `mystation/firebase/auth`, `dart:io`, and `firebase_storage`. The class has fields `collectionName` and `uid`, and a constructor that takes `uid` and `collectionName`. It also contains a private method `_getCurrentCollection` that returns a `CollectionReference` for the specified collection name.

```
1 import 'package:cloud_firestore/cloud_firestore.dart';
2 import 'package:firebase_auth/firebase_auth.dart';
3 import 'package:mystation/firebase/auth.dart';
4 import 'dart:io';
5 import 'package:firebase_storage/firebase_storage.dart';
6
7 class DatabaseServiceTrial {
8     final String collectionName;
9     final String? uid;
10
11     DatabaseServiceTrial({this.uid, required this.collectionName});
12
13     // Create a private method to get a reference to the current collection
14     CollectionReference _getCurrentCollection() {
15         return FirebaseFirestore.instance.collection(collectionName);
16     }
17 }
```

Figure 5.2 Flutter Firebase package used in Android Studio

Referencing Figure 5.3, this code defines a widget called **Authenticate** that manages the authentication flow of the application. It contains a **showSignIn** Boolean variable to toggle between showing the sign-in and registration screens. The **toggleView** method updates the state to switch between the two screens. The **build** method returns a **SignIn** widget, passing the **toggleView** method as a parameter, allowing users to switch between signing in and registering. This widget encapsulates the authentication process, facilitating seamless navigation between the sign-in and registration screens within the application.

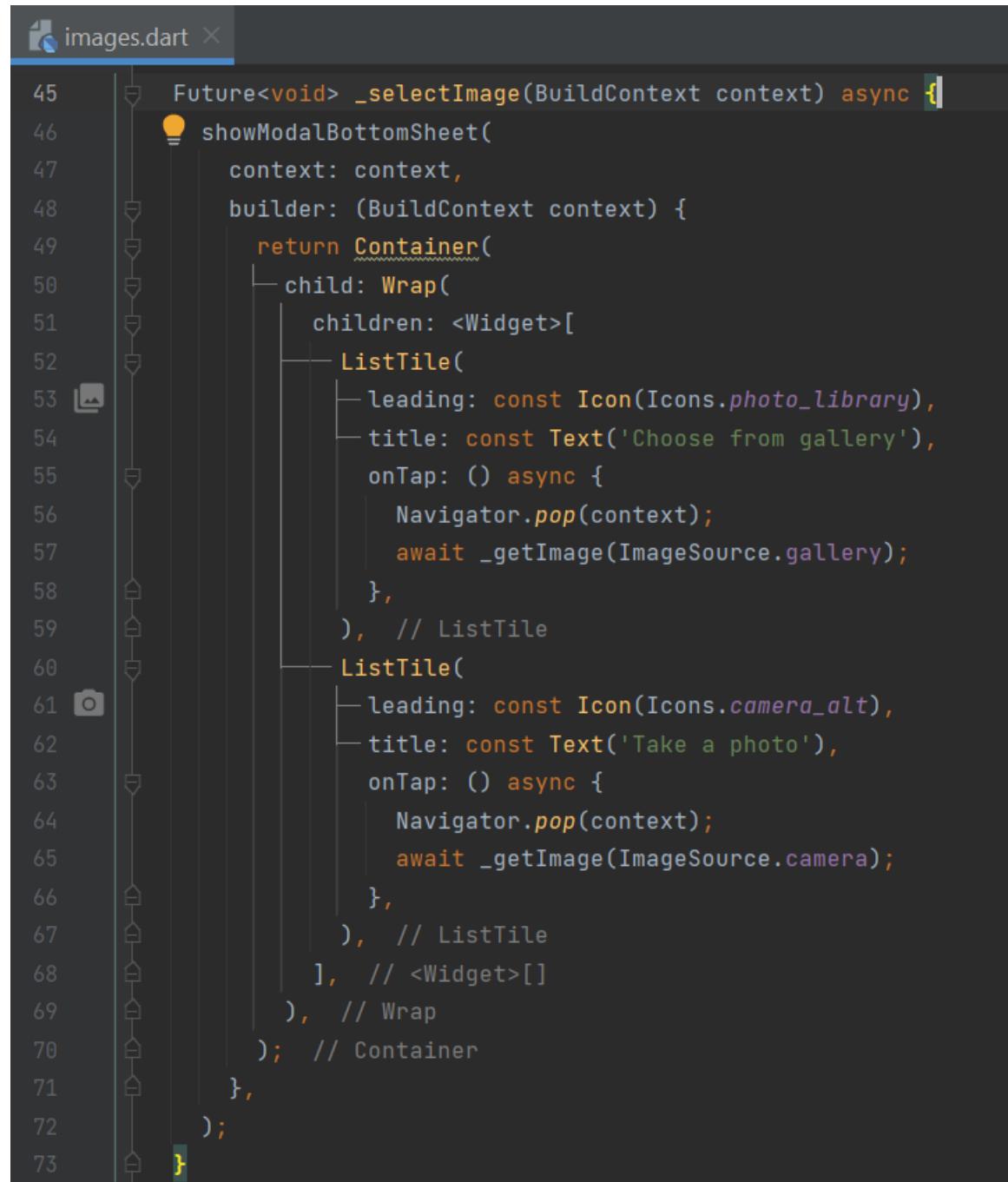


The screenshot shows the code for the `authenticate.dart` file in the `mystation/authenticate/sign_in.dart` package. The code defines a `Authenticate` class that extends `StatefulWidget`. It has a `const` constructor and overrides the `createState` and `build` methods. The `toggleView` method toggles the `showSignIn` state. The `build` method returns a `SignIn` widget with the `toggleView` parameter set to the current value of `showSignIn`.

```
1 import 'package:flutter/material.dart';
2 import 'package:mystation/authenticate/sign_in.dart';
3
4 class Authenticate extends StatefulWidget {
5     const Authenticate({super.key});
6     @override
7     State<Authenticate> createState() => _AuthenticateState();
8 }
9
10 class _AuthenticateState extends State<Authenticate> {
11
12     bool showSignIn = true;
13     void toggleView() {
14         setState(() => showSignIn = !showSignIn);
15     }
16
17     @override
18     Widget build(BuildContext context) {
19         return SignIn(toggleView: toggleView);
20     }
21 }
22
```

Figure 5.3 Authentication Package used in Android Studio

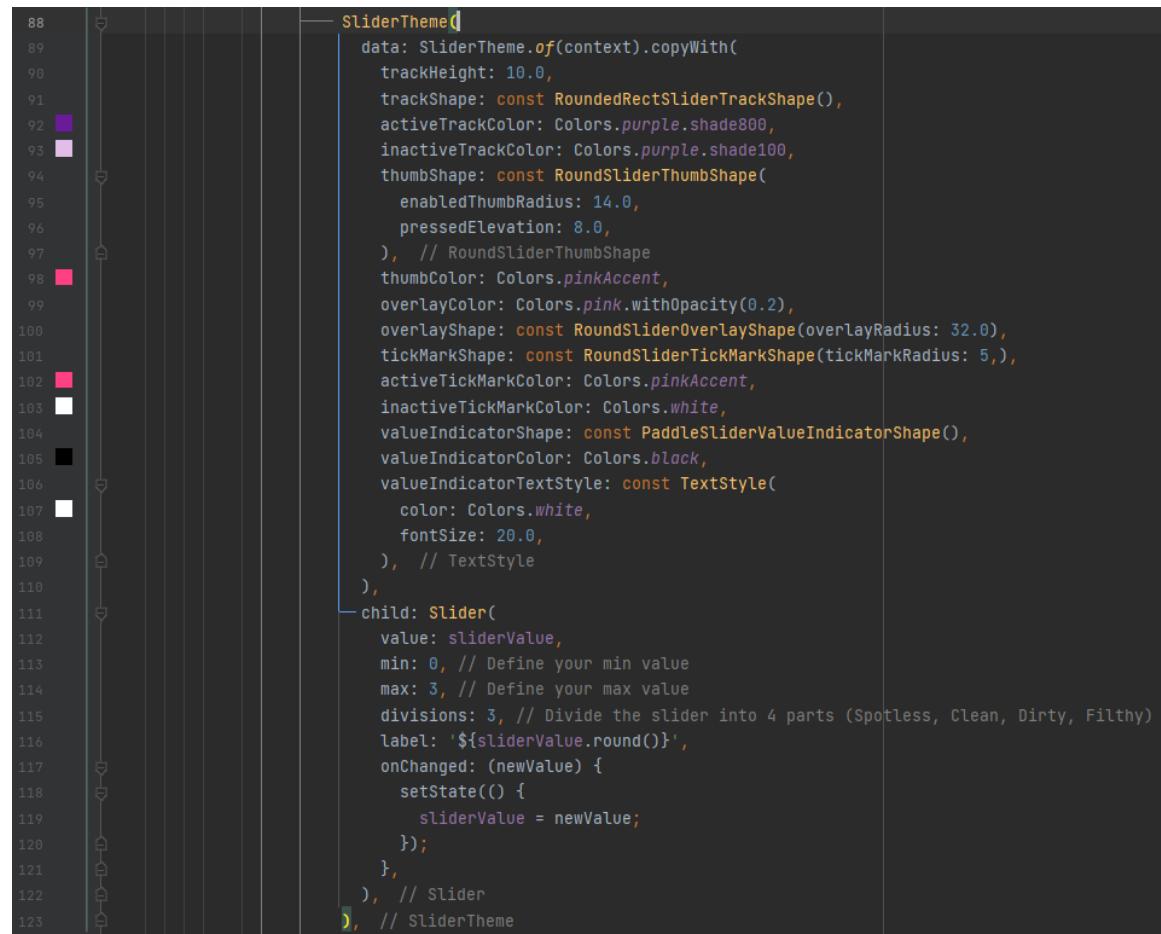
This code defines a widget called **Images** that allows users to select images from the gallery or take a photo using the device's camera. It utilises the **ImagePicker** package to provide this functionality. The `_selectImage` method displays a modal bottom sheet with two options: "Choose from gallery" and "Take a photo." Overall, as indicated by Figure 5.4, this widget facilitates the process of adding images to the application's database, enhancing its functionality and user experience.



```
1 import 'package:flutter/material.dart';
2 import 'package:image_picker/image_picker.dart';
3
4 class Images extends StatelessWidget {
5   @override
6   Widget build(BuildContext context) {
7     return Scaffold(
8       appBar: AppBar(
9         title: Text('Image Picker Example'),
10        ),
11        body: Center(
12          child: ElevatedButton(
13            onPressed: () {
14              _selectImage(context);
15            },
16            child: Text('Select Image'),
17          ),
18        ),
19      );
20    }
21  }
22
23 Future<void> _selectImage(BuildContext context) async {
24   await showModalBottomSheet(
25     context: context,
26     builder: (BuildContext context) {
27       return Container(
28         child: Wrap(
29           children: <Widget>[
30             ListTile(
31               leading: const Icon(Icons.photo_library),
32               title: const Text('Choose from gallery'),
33               onTap: () async {
34                 Navigator.pop(context);
35                 await _getImage(ImageSource.gallery);
36               },
37             ), // ListTile
38             ListTile(
39               leading: const Icon(Icons.camera_alt),
40               title: const Text('Take a photo'),
41               onTap: () async {
42                 Navigator.pop(context);
43                 await _getImage(ImageSource.camera);
44               },
45             ), // ListTile
46           ], // <Widget>[]
47         ), // Wrap
48       ); // Container
49     },
50   );
51 }
```

Figure 5.4 ImagePicker package used in Android Studio

This portion of the code defines a customisable rating slider using the **Slider** widget wrapped in a **SliderTheme**. The **SliderTheme** allows customisation of various visual aspects of the slider, such as track height, track shape, thumb shape, colours, and overlay. For instance, **trackHeight** sets the height of the slider track, **activeTrackColor** and **inactiveTrackColor** specify the colours of the active and inactive portions of the track, respectively. The **thumbShape** defines the appearance of the slider thumb, while **overlayColor** and **overlayShape** control the appearance of the overlay that appears when the thumb is pressed. Additionally, **tickMarkShape** and **activeTickMarkColor** allow customisation of tick marks along the slider track. The **Slider** widget itself defines the behaviour of the slider, including its current value (**value**), minimum and maximum values (**min** and **max**), divisions, and label display. Overall, as evidenced by Figure 5.5 below, this customisable rating slider provides a visually appealing and interactive way for users to select a value within a defined range.



```

88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123

```

```

  data: SliderTheme.of(context).copyWith(
    trackHeight: 10.0,
    trackShape: const RoundedRectSliderTrackShape(),
    activeTrackColor: Colors.purple.shade800,
    inactiveTrackColor: Colors.purple.shade100,
    thumbShape: const RoundSliderThumbShape(
      enabledThumbRadius: 14.0,
      pressedElevation: 8.0,
    ), // RoundSliderThumbShape
    thumbColor: Colors.pinkAccent,
    overlayColor: Colors.pink.withOpacity(0.2),
    overlayShape: const RoundSliderOverlayShape(overlayRadius: 32.0),
    tickMarkShape: const RoundSliderTickMarkShape(tickMarkRadius: 5.),
    activeTickMarkColor: Colors.pinkAccent,
    inactiveTickMarkColor: Colors.white,
    valueIndicatorShape: const PaddleSliderValueIndicatorShape(),
    valueIndicatorColor: Colors.black,
    valueIndicatorTextStyle: const TextStyle(
      color: Colors.white,
      fontSize: 20.0,
    ), // TextStyle
  ),
  child: Slider(
    value: sliderValue,
    min: 0, // Define your min value
    max: 3, // Define your max value
    divisions: 3, // Divide the slider into 4 parts (Spotless, Clean, Dirty, Filthy)
    label: '${sliderValue.round()}',
    onChanged: (newValue) {
      setState(() {
        sliderValue = newValue;
      });
    },
  ), // Slider
), // SliderTheme

```

Figure 5.5 Customisable Rating Slider Using the Slider Widget

5.1.2 Google Firebase

In this project, Google Firebase is utilised as a cloud-based backend service to store and manage various data related to the application. Firebase Authentication is employed for user authentication, enabling secure access to the application's features. Firestore, Firebase's NoSQL cloud database, is leveraged to store user information, feedback, reports, and image URLs. Real-time updates and data synchronisation are facilitated through Firestore's capabilities, ensuring seamless communication between the client-side application and the server-side database. Additionally, Firebase Storage is utilised to store and manage image files uploaded by users, providing efficient and scalable storage solutions. As depicted graphically by Figure 5.6, 5.7 and 5.8, Google Firebase serves as a robust backend infrastructure, empowering the application with features like authentication, data storage, and real-time synchronisation, essential for delivering a seamless and responsive user experience.

The screenshot shows the Firebase Authentication interface. At the top, there are tabs for 'Users', 'Sign-in method', 'Templates', 'Usage', 'Settings', and 'Extensions'. Below the tabs is a search bar with placeholder text 'Search by email address, phone number, or user UID' and a blue 'Add user' button. A three-dot menu icon is also present. The main area displays a table of user data with columns: Identifier, Providers, Created, Signed In, and User UID. The data includes:

Identifier	Providers	Created	Signed In	User UID
newuser@gmail.com	✉️	Jan 24, 2024	Jan 24, 2024	ke0vmgBqLAfHJUuqfvCGn9p...
test55@gmail.com	✉️	Jan 24, 2024	Jan 24, 2024	BhRkxeKFBZWH0XICQCTy0gh...
test44@gmail.com	✉️	Jan 22, 2024	Feb 9, 2024	HRsLI59SiKcj1qSOJgqxt6bq7...
test4@gmail.com	✉️	Jan 21, 2024	Jan 21, 2024	SPMNDvqQ0yWJsRuedl5Qqk0...
sitianakazam2404@gm...	✉️	Jan 21, 2024	Jan 21, 2024	Dc0pBVymQIN2GTZie5qjeSUG...
test3@gmail.com	✉️	Jan 21, 2024	Jan 21, 2024	5u9gq401o9VpDt6CPxv7jXep...
danyizatul01@gmail.com	✉️	Jan 21, 2024	Jan 21, 2024	3UU8LI3fTYcQ0MloJiFRwsjxk...
test2@gmail.com	✉️	Jan 21, 2024	Jan 21, 2024	KzxwrSqWSnSnx6rvE6nB2wsC...
mijapatini@gmail.com	✉️	Jan 20, 2024	Jan 20, 2024	gZ0en5KBBMWePvMYbUcnV4...

Figure 5.6 Firebase Authentication

The screenshot shows the Cloud Firestore interface. At the top, there are tabs for Data, Rules, Indexes, Usage, and Extensions. Below the tabs, a breadcrumb navigation shows the path: Home > Dunkin Donuts > CLuLj1Tzy42Zw... . On the right, there are buttons for Panel view and Query builder. The main area displays a document structure under the collection 'Dunkin Donuts'. The document has fields like 'condition', 'feedback', 'label', 'likes', 'timestamp', 'uid', and 'userName'. A sidebar on the left lists other collections: (default), Start collection, Alya Jaya Enterprise, Dunkin Donuts, E-hailing, Express Bus Ticket Counter, I Love Yoo!, Images: Alya Jaya Enterprise, Images: Apam Balik Seri Menanti, Images: Dunkin Donuts, Images: E-hailing, and Images: Express Bus.

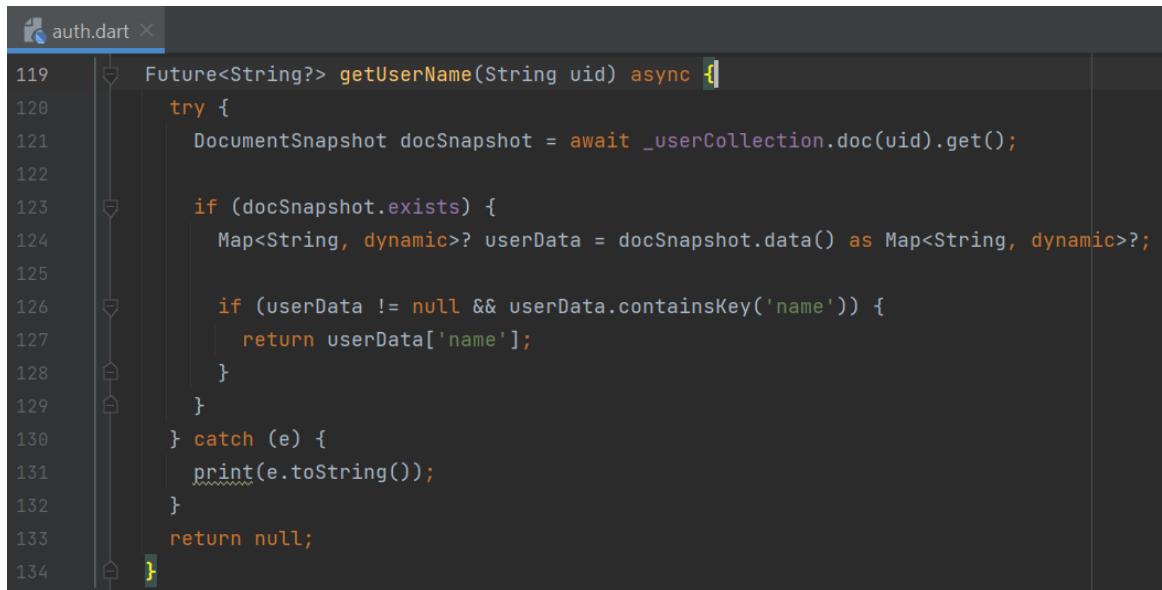
Figure 5.7 Firestore Database

The screenshot shows the Firebase Storage interface. At the top, there are tabs for Files, Rules, Usage, and Extensions. Below the tabs, a URL gs://mystation-1010b.appspot.com is shown. On the right, there are buttons for Upload file, Create folder, and More options. The main area displays a list of uploaded folders. The list includes: Images: Apam Balik Seri Menanti/, Images: Dunkin Donuts/, Images: E-hailing/, Images: Express Bus Ticket Counter/, Images: Express Bus/, Images: Hot & Roll/, Images: I Love Yoo/, Images: KLIA Transit/, and Images: Kedai K1/.

Figure 5.8 Firebase Storage

5.1.3 Application Programming Interface (API)

One of the Application Programming Interfaces (API) that have been implemented in the MyStation is updating user information. In the **auth.dart** file, the **getUserName** method retrieves the user's name from the Firebase Firestore database based on their unique user ID (**uid**). Upon invocation, it queries the **_userCollection** for the user document corresponding to the provided ID. If the document exists and contains a field named 'name', it returns the user's name. Otherwise, it returns null. As showcased by Figure 5.9, this method facilitates fetching user information from the database to display in the application interface.



```
auth.dart x
119 Future<String?> getUserName(String uid) async {
120   try {
121     DocumentSnapshot docSnapshot = await _userCollection.doc(uid).get();
122
123     if (docSnapshot.exists) {
124       Map<String, dynamic>? userData = docSnapshot.data() as Map<String, dynamic>?;
125
126       if (userData != null && userData.containsKey('name')) {
127         return userData['name'];
128       }
129     }
130   } catch (e) {
131     print(e.toString());
132   }
133   return null;
134 }
```

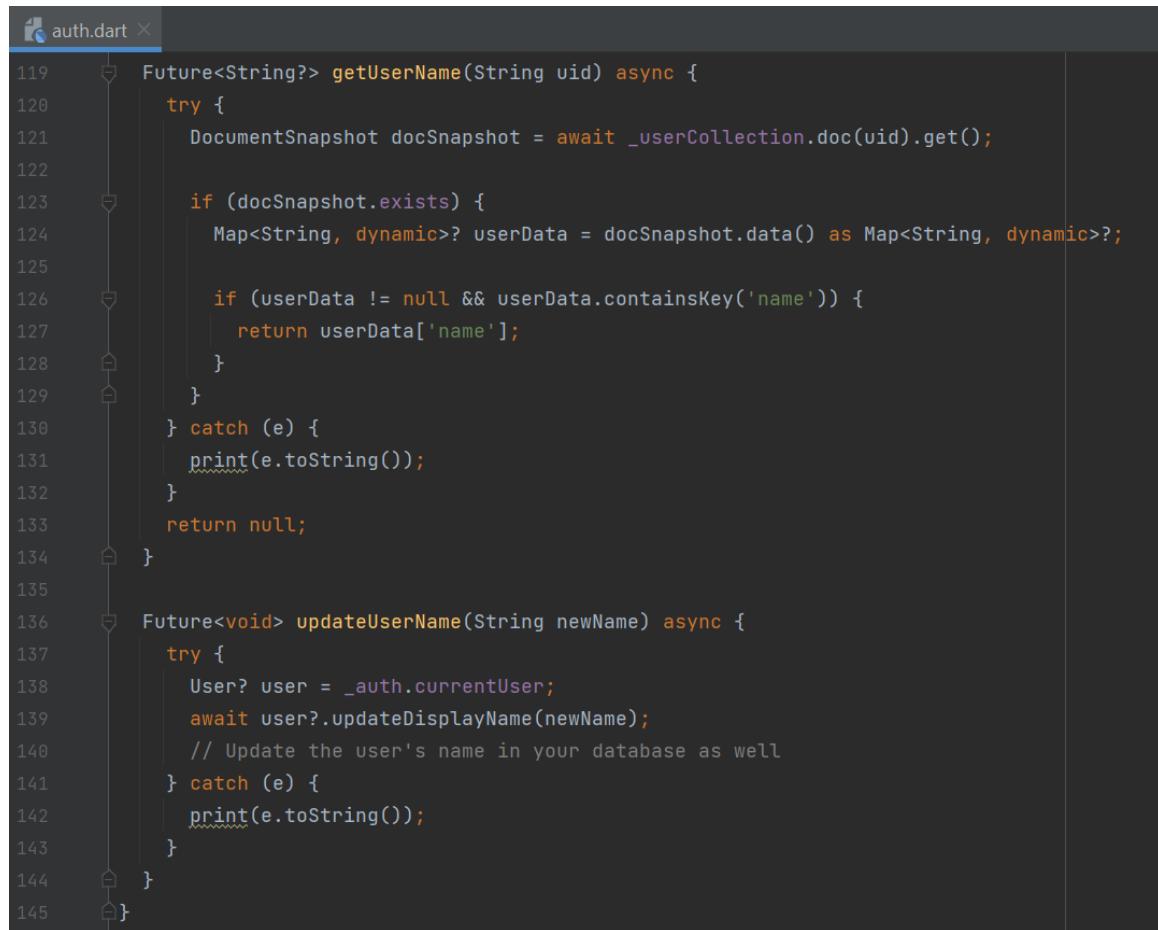
Figure 5.9 Method to Retrieve User's Username

The **UserProfileScreen** class encapsulates functionality to edit a user's name. Upon invoking the **_showEditName** method, an **AlertDialog** is displayed, allowing users to input a new name. The current name, retrieved as a parameter, is pre-filled in the text field. Users can then make edits and choose to save or cancel their changes. If the save button is pressed, the **_auth.updateUserName** method is called, which asynchronously updates the user's name in the backend. Subsequently, the dialogue is closed, and the updated name is reflected in the application. As outlined in the visual representation of Figure 5.10, this mechanism provides users with a convenient and intuitive interface to modify their profile information seamlessly.

```
3 import 'package:mystation/firebase/auth.dart';
4 import 'package:mystation/firebase/user.dart';
5
6 class UserProfileScreen extends StatelessWidget {
7   final AuthService _auth = AuthService();
8
9   UserProfileScreen({super.key});
10
11   void _showEditName(BuildContext context, String? currentName) {
12     TextEditingController editController = TextEditingController(
13       text: currentName);
14
15     showDialog(
16       context: context,
17       builder: (context) =>
18         AlertDialog(
19           title: const Text('Edit Name'),
20           content: TextField(
21             controller: editController,
22             decoration: const InputDecoration(labelText: 'Edit your name'),
23           ), // TextField
24           actions: [
25             TextButton(
26               onPressed: () {
27                 // Close the dialog
28                 Navigator.pop(context);
29               },
30               child: const Text('Cancel'),
31             ), // TextButton
32             TextButton(
33               onPressed: () async {
34                 // Implement edit functionality
35                 String updatedName = editController.text;
36                 await _auth.updateUserName(updatedName);
37                 // Close the dialog
38                 Navigator.pop(context);
39               },
40               child: const Text('Save'),
41             ), // TextButton
42           ],
43         ), // AlertDialog
44       );
45   }
46 }
```

Figure 5.10 Method Allowing Users to Edit Username

The **updateUserName** allows users to update their name. It takes a new name (**newName**) as input and attempts to update the display name of the current user authenticated through Firebase. Subsequently, it may also update the user's name in the database if necessary. In alignment with the data presented in Figure 5.11, this function ensures that changes made by users are reflected both locally and, in the database, ensuring consistency across the application. Any errors encountered during the process are logged for debugging purposes.



```

auth.dart ×
119   Future<String?> getUserName(String uid) async {
120     try {
121       DocumentSnapshot docSnapshot = await _userCollection.doc(uid).get();
122
123       if (docSnapshot.exists) {
124         Map<String, dynamic>? userData = docSnapshot.data() as Map<String, dynamic>?;
125
126         if (userData != null && userData.containsKey('name')) {
127           return userData['name'];
128         }
129       }
130     } catch (e) {
131       print(e.toString());
132     }
133     return null;
134   }
135
136   Future<void> updateUserName(String newName) async {
137     try {
138       User? user = _auth.currentUser;
139       await user?.updateDisplayName(newName);
140       // Update the user's name in your database as well
141     } catch (e) {
142       print(e.toString());
143     }
144   }
145 }
```

Figure 5.11 Method to Update User's Username in the Database

The user data will be updated and displayed in Firebase. Figure 5.12 below shows the collection of ‘users’ and its content.

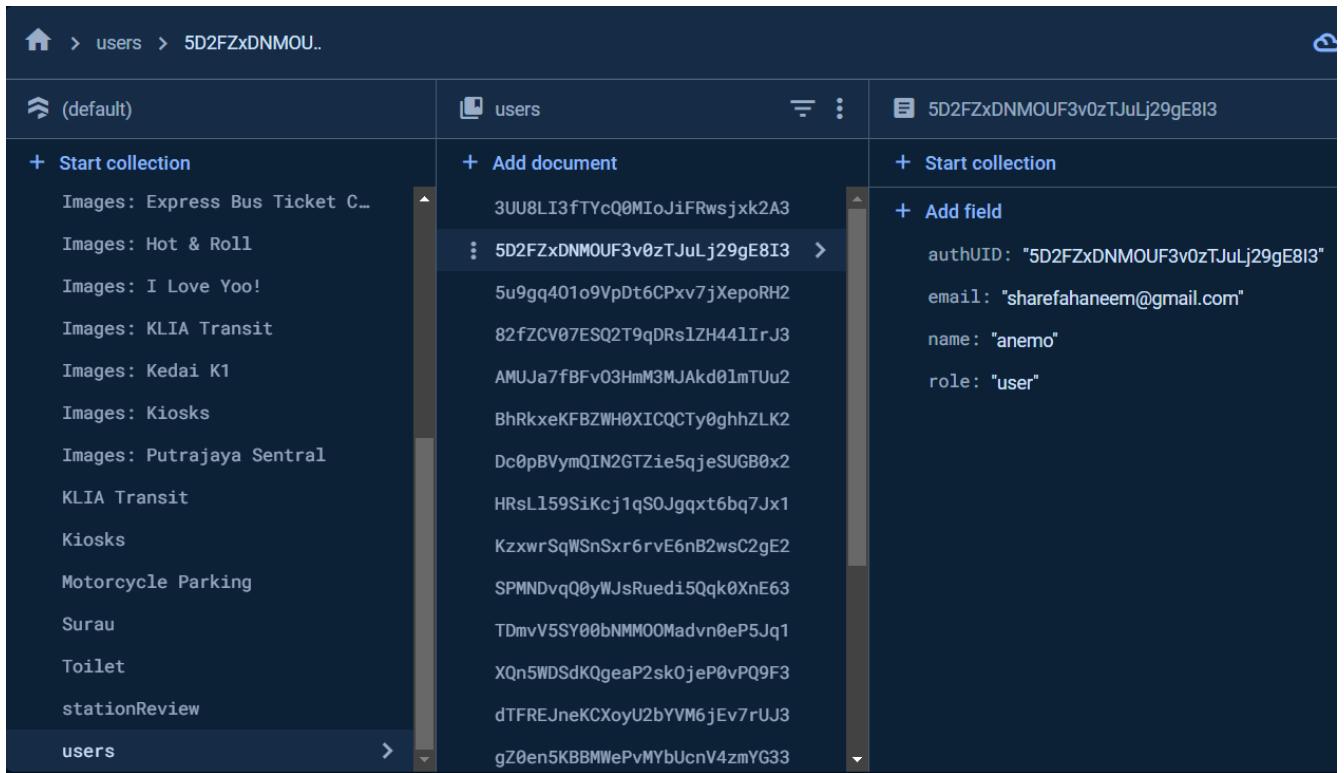


Figure 5.12 "users" Collection in the Firebase

Furthermore, we also write the code of calling user data from the Firebase to display on the MyStation application. the **FeedbackList** widget is initialised with two parameters: **userName** and **collectionName** as depicted in Figure 5.13 below. The **userName** parameter represents the current username, which is passed to the widget when it is instantiated. This username can be obtained from the authentication service (**AuthService**) using methods such as **getCurrentUserName** or similar functions that retrieve the username associated with the currently authenticated user. By passing this username to the **FeedbackList** widget, the system can utilise it within the widget's state to display user-specific information or perform user-related actions within the feedback list interface.

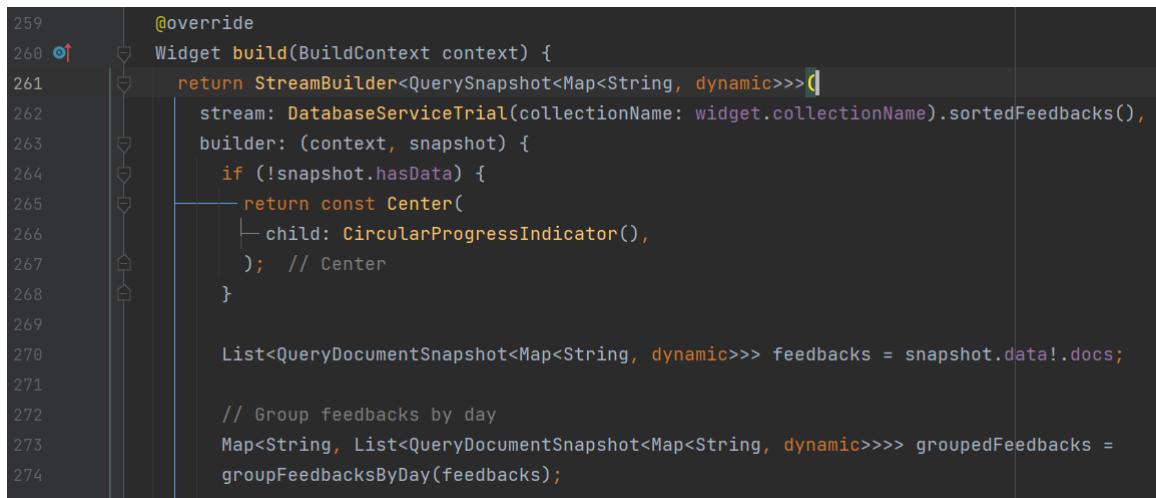
```

15     class FeedbackList extends StatefulWidget {
16       final String? userName;
17       final String collectionName;
18       const FeedbackList({Key? key, required this.userName, required this.collectionName}) : super(key: key);
19
20       @override
21       State<FeedbackList> createState() => _FeedbackListState();
22     }
23
24     class _FeedbackListState extends State<FeedbackList> {
25       AuthService auth = AuthService();

```

Figure 5.13 Syntax to Pass the Username Parameter

In the build method of the **FeedbackList** widget, a **StreamBuilder** is used to listen for changes in the sorted feedbacks stream obtained from the **DatabaseServiceTrial**. Within the **ListView.builder**, each feedback item is displayed along with its associated information such as the username of the user who submitted the feedback. The username is retrieved from the **feedbackData** map, which contains information about each feedback item. If the username is available, it is displayed as "Submitted by: username", otherwise, it defaults to "Anonymous". This ensures that the username associated with each feedback is appropriately displayed in the feedback list. Illustrated in Figure 5.14 is the method to access the data from Firebase Collection while Figure 5.15 shows how the username is displayed.

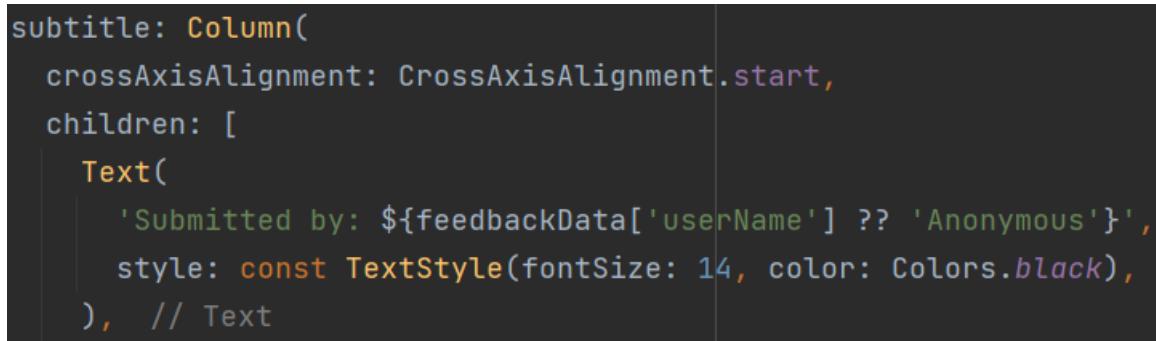


```

259     @override
260     Widget build(BuildContext context) {
261       return StreamBuilder<QuerySnapshot<Map<String, dynamic>>>(
262         stream: DatabaseServiceTrial(collectionName: widget.collectionName).sortedFeedbacks(),
263         builder: (context, snapshot) {
264           if (!snapshot.hasData) {
265             return const Center(
266               child: CircularProgressIndicator(),
267             ); // Center
268           }
269
270           List<QueryDocumentSnapshot<Map<String, dynamic>>> feedbacks = snapshot.data!.docs;
271
272           // Group feedbacks by day
273           Map<String, List<QueryDocumentSnapshot<Map<String, dynamic>>>> groupedFeedbacks =
274             groupFeedbacksByDay(feedbacks);

```

Figure 5.14 Method to Access the Data from Firebase Collection.



```

subtitle: Column(
  mainAxisAlignment: MainAxisAlignment.start,
  children: [
    Text(
      'Submitted by: ${feedbackData['userName'] ?? 'Anonymous'}',
      style: const TextStyle(fontSize: 14, color: Colors.black),
    ), // Text

```

Figure 5.15 How the Username is Displayed in the App

5.2 Completed Application Process

In the MyStation application, there are 8 primary processes designed for users and 2 main processes tailored for administrators. Upon accessing the application, users are directed to the Sign In page, where registered users and administrators with specific credentials can log in. The application's functionalities are divided into distinct user and admin sides, ensuring that each user type can access relevant features seamlessly. From submitting feedback and viewing station information to managing user reports and uploading images, the MyStation application provides a comprehensive experience for both users and administrators, facilitating efficient interaction with station-related data and functionalities.

5.2.1 User Site

The user side of the MyStation application encompasses several key processes aimed at providing a seamless experience for users interacting with the station-related functionalities. This includes registration and sign-in procedures, allowing users to create and access their accounts securely. Additionally, users can edit their profiles to update personal information as needed. The MyStation application enables users to view general information of certain train station efficiently. Users can also view station images, services or shops available, give feedbacks and view or add station reports. Finally, users have the option to sign out their accounts if necessary, ensuring flexibility and control over their user experience within the MyStation application.

5.2.1.1 Register and Sign In

Upon launching the app, users are presented with a welcoming screen adorned with the MyStation logo, featuring vibrant gradients that evoke a sense of dynamism and modernity. The initial screen prompts users to either sign in or register, offering a seamless transition between the two authentication processes. By tapping on the "Sign In" button, users are directed to a form where they can input their registered email address and password. The form is thoughtfully designed with visual cues such as icons and text fields to enhance user experience and facilitate input. Alternatively, users can navigate to the registration form by selecting the "Register" option, which reveals additional fields for creating a new account, including name, email, and password. Throughout the process, error messages are displayed in real-time to provide immediate feedback and guide users towards successful authentication. Upon successful sign-in or registration, users are seamlessly redirected to the appropriate interface based on their role, whether it be a regular user or an administrator, ensuring a tailored and intuitive experience from the outset. The Figure 5.16 provided offers a visual representation of the data discussed.

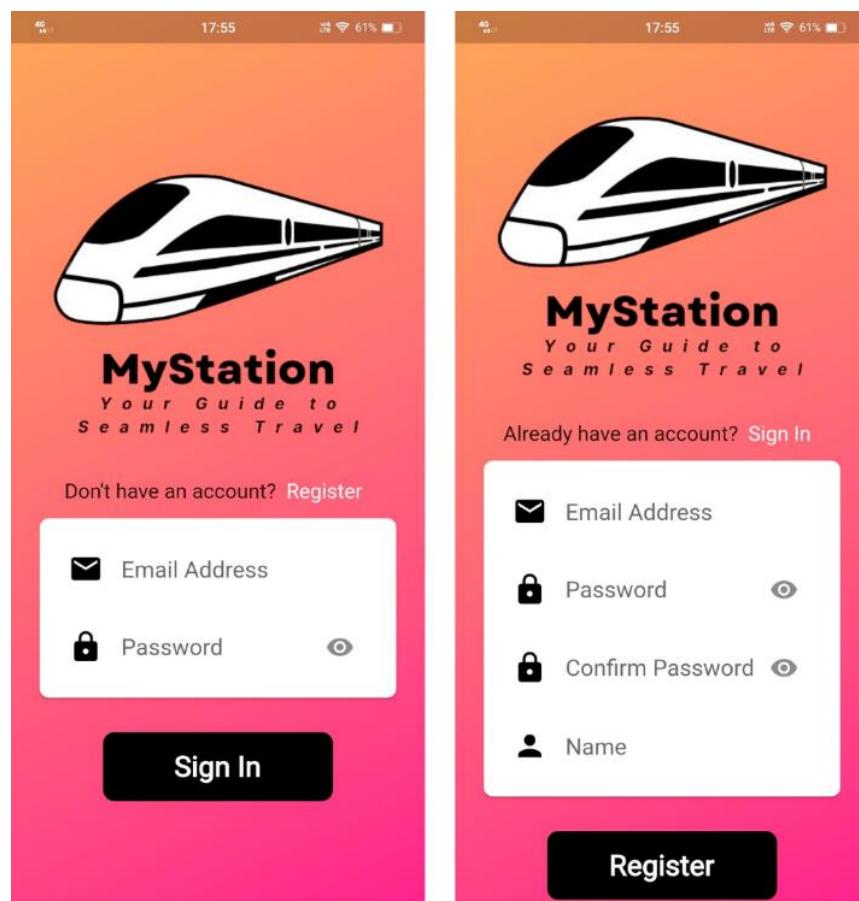


Figure 5.16 Register and Sign in Interfaces

5.2.1.1 Main Menu

Upon registering and signing into the train station application, users will be welcomed to a dynamic and intuitive main menu, featuring vibrant buttons showcasing an array of train lines with names, routes, and captivating images. With a tap, users can delve into the intricacies of each line, immersing themselves in the city's rich tapestry of routes. The backdrop's supplementary content offers insights and tips, while intuitive navigation options beckon users to explore further, from checking station facilities to planning trips using interactive maps. It's not just accessing a menu; it's embarking on a journey—an exploration of the city's heart and soul, all from the palm of their hand. For a clearer illustration, the following Figure 5.17 displays the concepts described.

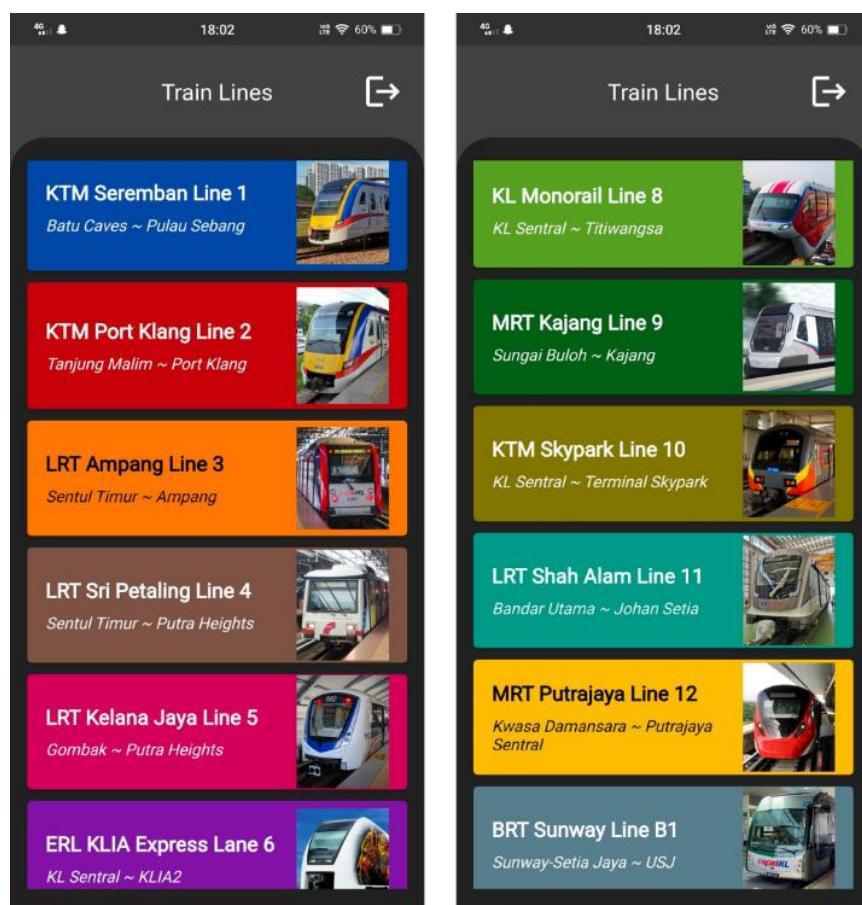


Figure 5.17 User Main Menu

5.2.1.2 Selection of Train Stations

Upon selecting the Putrajaya Line from the main menu, users will be presented with a comprehensive list of stations along this route. The interface boasts a sleek design, with a vibrant amber colour scheme setting the tone for exploration. Each station is meticulously listed, accompanied by icons indicating whether transfers to other lines are available. Users can effortlessly navigate through the stations, with intuitive buttons inviting interaction and providing essential information at a glance. As users delve deeper into the list, they're empowered to plan their journey with ease, fostering a seamless and engaging travel experience. To provide a visual aid, the subsequent Figure 5.18 elucidates the points made.

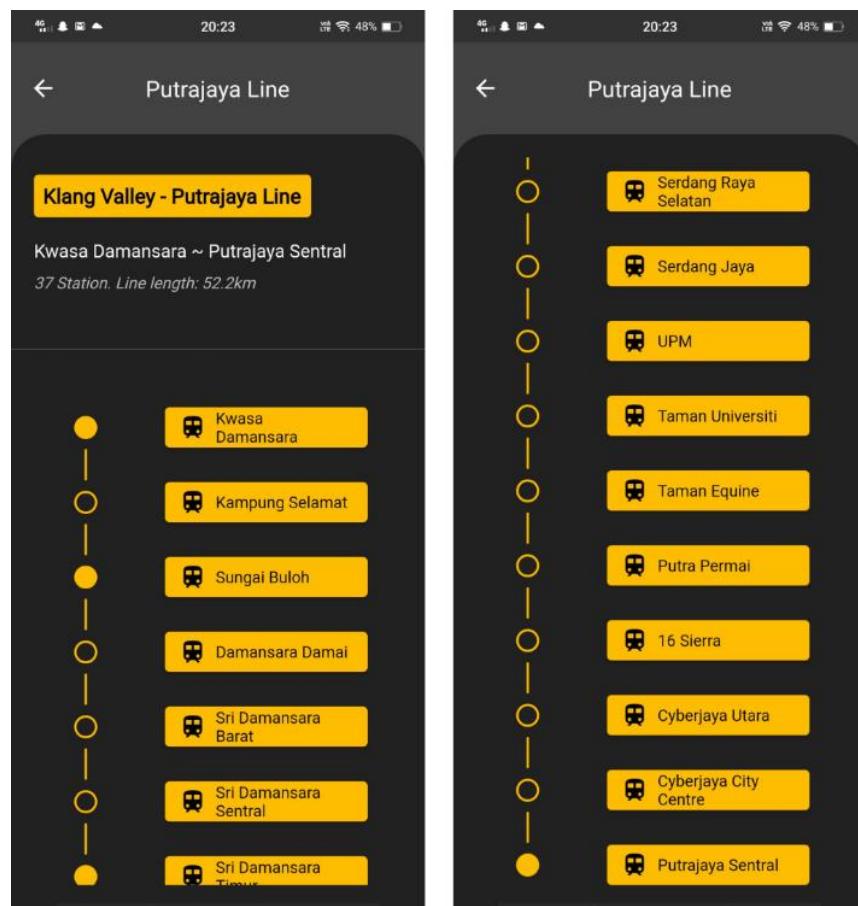


Figure 5.18 Putrajaya Line

5.2.1.3 Overview Page

Upon selecting a specific station, users are directed to a dedicated page tailored to their needs. Users can explore various sections, including an overview of the station, available services, station reports, and their profile details. Each section is accessible via a user-friendly bottom navigation bar, allowing for effortless switching between different functionalities. The overview page displays comprehensive information about the selected station, including transportation services, operational status, and contact details. Additionally, users can view captivating images of the station and access essential functions such as calling, directions, and sharing via intuitive icons, enhancing their interaction with the app. Figure 5.19 illustrates the layout of the Overview page, showcasing various components such as a carousel slider for images, operational status, contact information, and interactive buttons for calling, directions, and sharing.

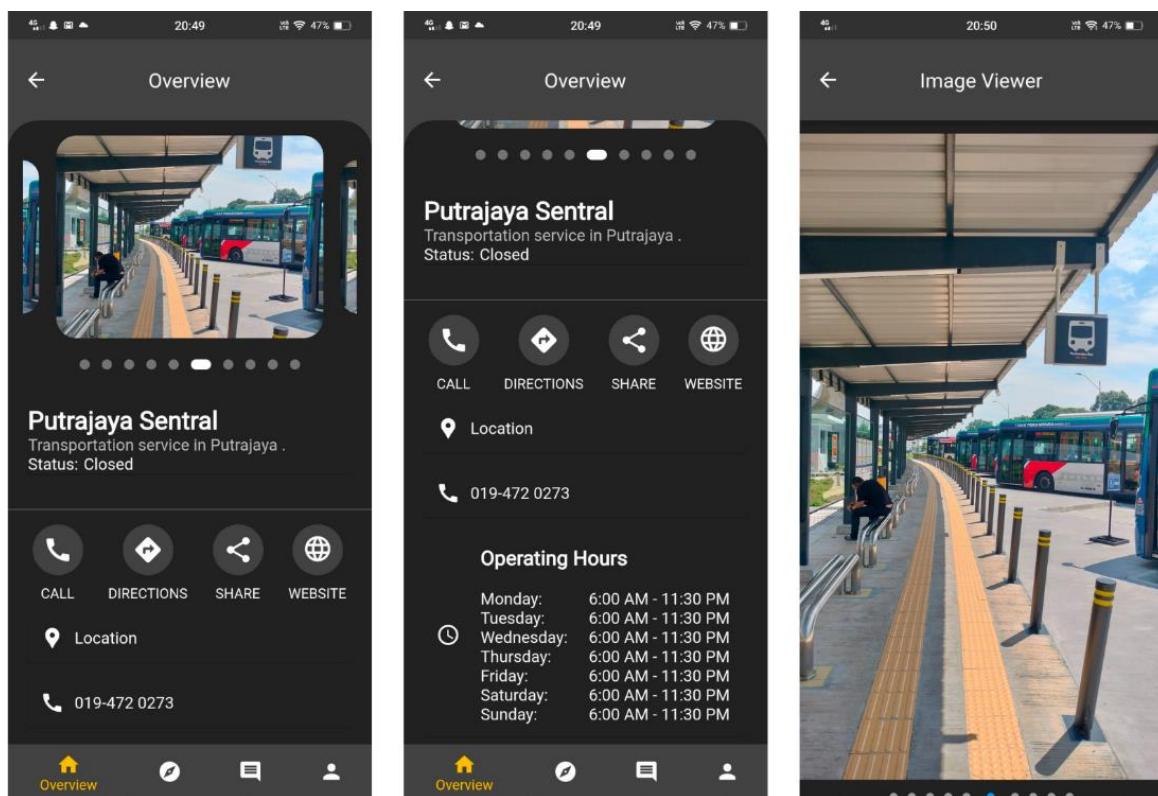


Figure 5.19 Overview Page

5.2.1.4 Services Page

Upon tapping on the Services section in the bottom navigation bar, users are directed to the Menu Detail page, as depicted in Figure 5.20. This page showcases a grid layout of cards representing various available services within the station. Each card displays an icon representing the service category along with its title. Users can explore different services by tapping on the respective card, which leads to detailed information about each service.

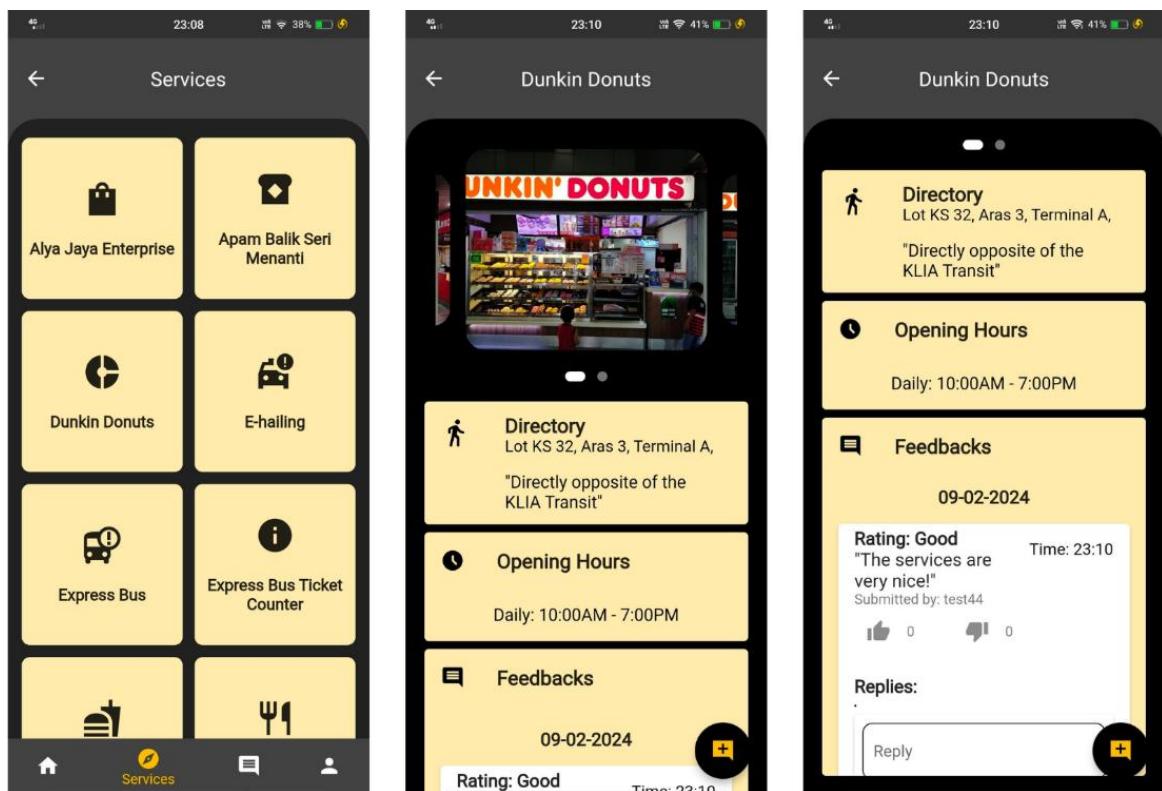


Figure 5.20 Services Page

5.2.1.5 Feedback Function

The Figure 5.21 below shows a comprehensive Flutter implementation for a feedback system, where users can rate facilities, provide comments, and manage feedback entries and replies. It incorporates features such as a grouped list of feedback items by date, with options for users to edit or delete their feedback and replies. Users can also like or dislike feedback items, and administrators have additional capabilities for managing all feedback. The interface includes sliders for rating selection and text fields for comments, with confirmation messages upon submission. Integrated user authentication ensures secure data handling throughout the feedback process.

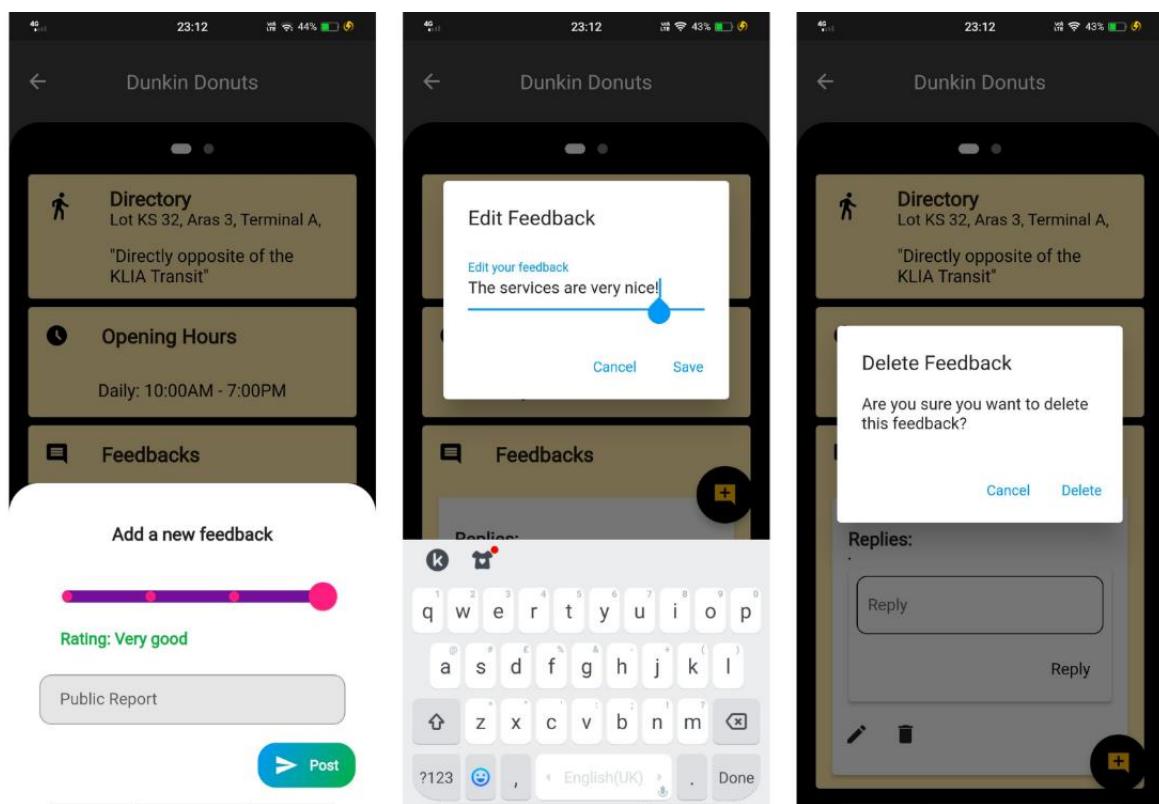


Figure 5.21 Feedback Function

5.2.1.6 Station Reports

The figure below shows the user interface for the "Station Reports" section in the bottom navigation bar, where users can submit reports on various aspects of station conditions. Upon tapping the section, users are presented with a page that displays existing reports sorted by day, allowing users to view and interact with individual report items. Each report item includes details such as the user's submission, timestamp, and options to like or dislike. Besides, other feature that exists in this page is the submission of new reports through a modal bottom sheet, where users can select the type of report they want to submit, ranging from crowdedness to cleanliness, each accompanied by an appropriate form for detailed report submission. Finally, a detailed form for submitting cleanliness reports, featuring a slider for rating with different levels and a text field for additional comments is provided after user chooses their report's type. After submitting a report, users receive a confirmation message, ensuring successful submission. Overall, this feature enhances user engagement by enabling comprehensive report submission and interaction with station conditions. The accompanying Figure 5.22 provides a graphical representation of the information outlined.

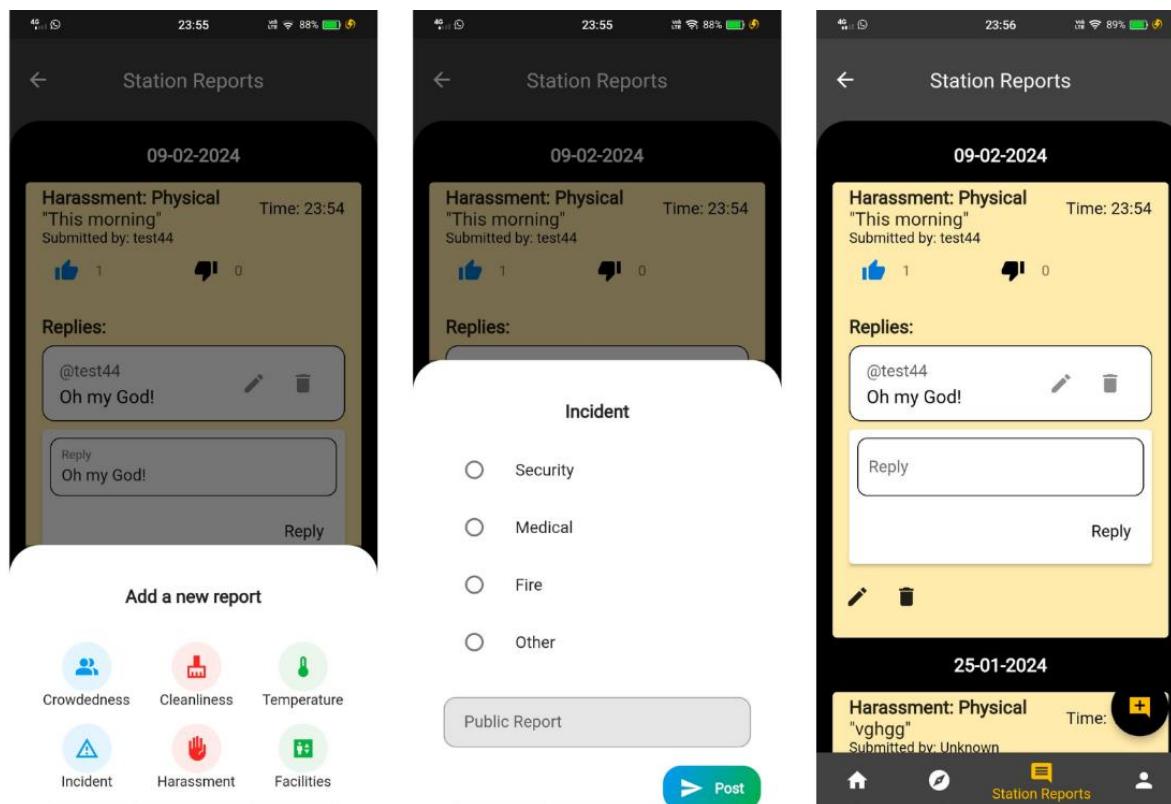


Figure 5.22 Station Reports

5.2.1.7 Profile Page

The figure depicts the user interface for the **UserProfileScreen** class, showcasing a user profile screen within the application. The screen features a black background with user information displayed in a centred layout. The user's profile picture, name, email, role, and UID are presented in individual cards, each adorned with a subtle amber colour. A "Sign Out" button at the bottom allows users to log out of their accounts. Notably, the "Name" card includes an edit icon, enabling users to edit their displayed name. Tapping the edit icon triggers a dialogue prompting users to input a new name, which upon saving, updates the displayed name accordingly. This functionality enhances user control over their profile information within the application. For a visual summary, refer to the Figure 5.23 below.

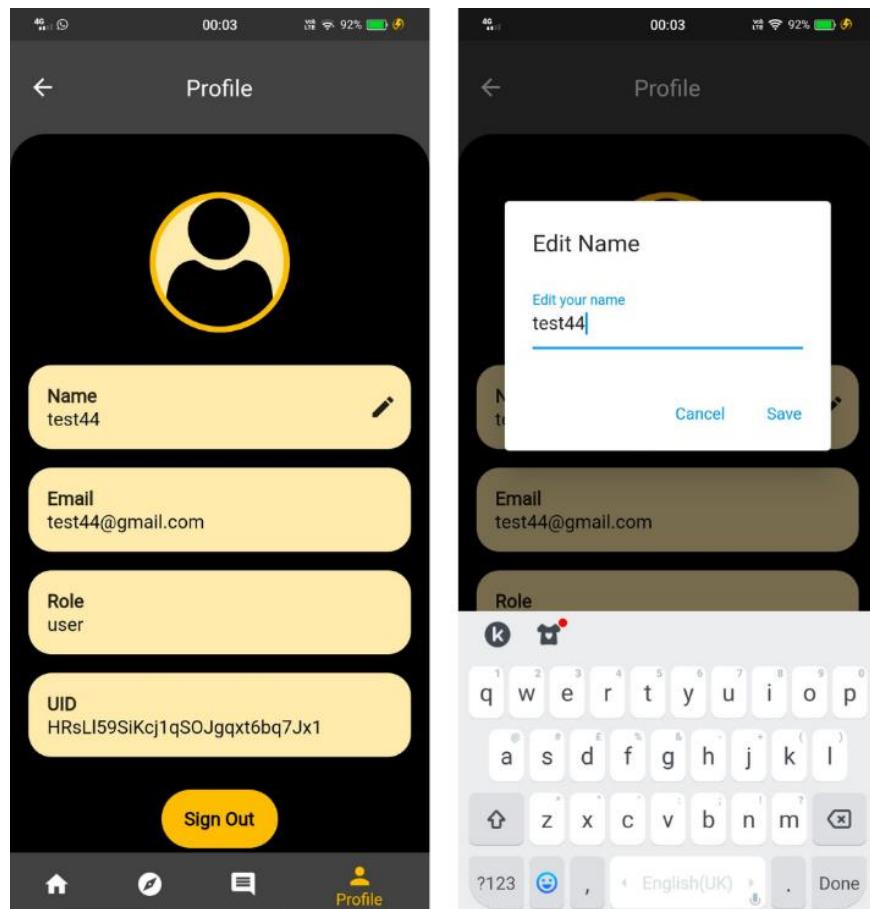


Figure 5.23 User Profile Page

In summary, the eight processes outlined encapsulate the core functions of the MyStation application, serving to fulfil the project's objectives from the user's perspective.

5.2.2 Admin Site

In the MyStation application, the admin site plays a crucial role in overseeing and managing user-generated content and feedback. Remarkably, the admin site interface mirrors that of the user site, maintaining consistency and familiarity for administrators. However, administrators are endowed with additional control and functionalities tailored to their roles. One significant capability is the ability to filter and review all user feedback and reports, empowering administrators to delete any irrelevant or inappropriate content as necessary, thus ensuring the quality and integrity of the platform's interactions. Moreover, administrators possess the authority to upload and delete images that are to be displayed or removed within the application, allowing for seamless content management and customisation. This amalgamation of familiar user interface elements with enhanced administrative functionalities underscores the versatility and user-centric design philosophy of the MyStation application ecosystem.

5.2.2.1 Upload and Delete Image

In the Overview page of the MyStation admin site, administrators are granted comprehensive control over user-generated content and the overall presentation of the application. One prominent feature available to administrators is the ability to manage images seamlessly. Administrators can add new images to enhance the visual appeal of the application, ensuring that it remains engaging and visually appealing to users. This feature allows administrators to upload relevant images such as promotional banners, event posters, or any visual elements that enrich the user experience. Additionally, administrators have the authority to delete images that are no longer relevant or necessary, maintaining the relevance and currency of visual content within the application. This capability empowers administrators to curate and customise the visual identity of the MyStation application, aligning it with the evolving needs and preferences of its user base. To complement the discussion, a graphical depiction is presented in the Figure 5.24 below.

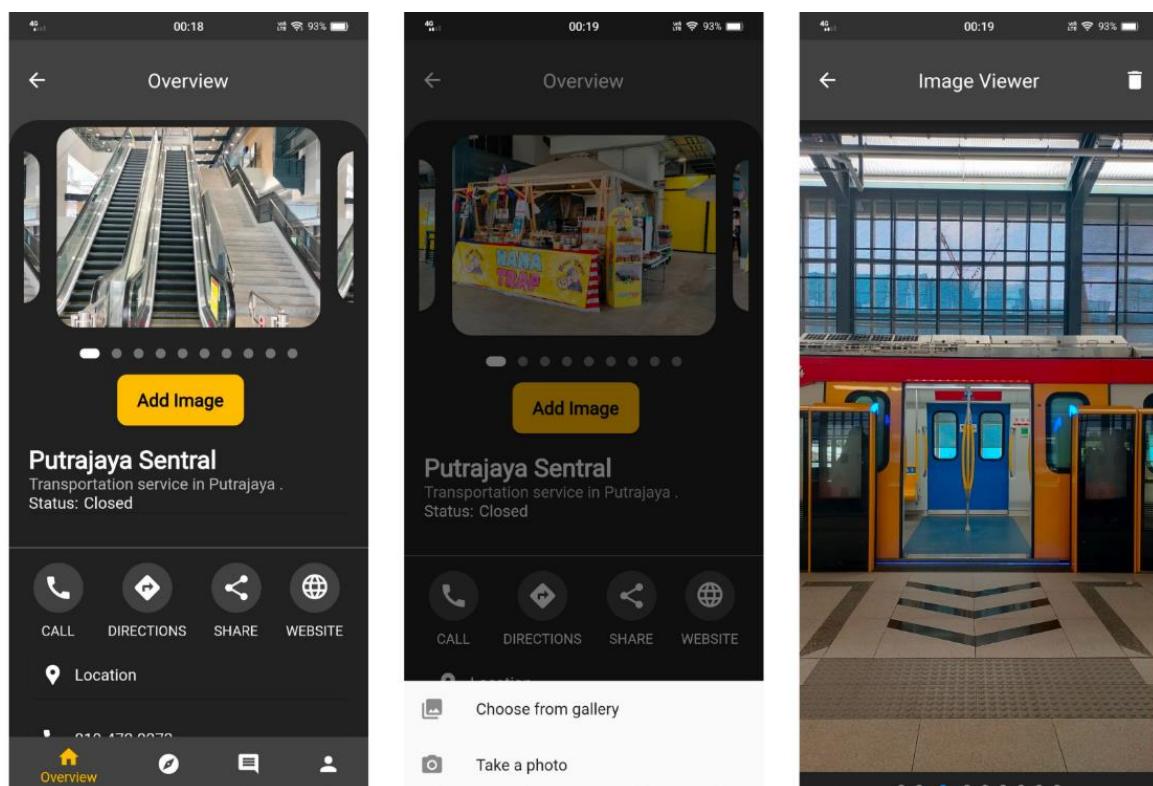


Figure 5.24 Upload and Delete Image

5.2.2.2 Delete Irrelevant Feedbacks and Reports

In the Station Reports page of the MyStation admin site, administrators wield critical authority to maintain the integrity and quality of user-generated feedback and reports. Their primary role involves meticulously reviewing and assessing the submitted feedback and reports to ensure they align with the app's standards and objectives. Administrators have the essential function of identifying and removing any irrelevant or inappropriate feedbacks and reports from the application. This involves scrutinising the content provided by users and taking appropriate action to delete any submissions that do not meet the app's guidelines or may potentially compromise the user experience. By actively monitoring and curating the feedback and reports, administrators uphold the credibility and reliability of the information presented within the application, fostering a trustworthy and user-centric environment for all users. This proactive approach underscores the importance of the administrator role in maintaining the overall quality and relevance of content within the MyStation app. For a more detailed overview, consult the Figure 5.25 provided.

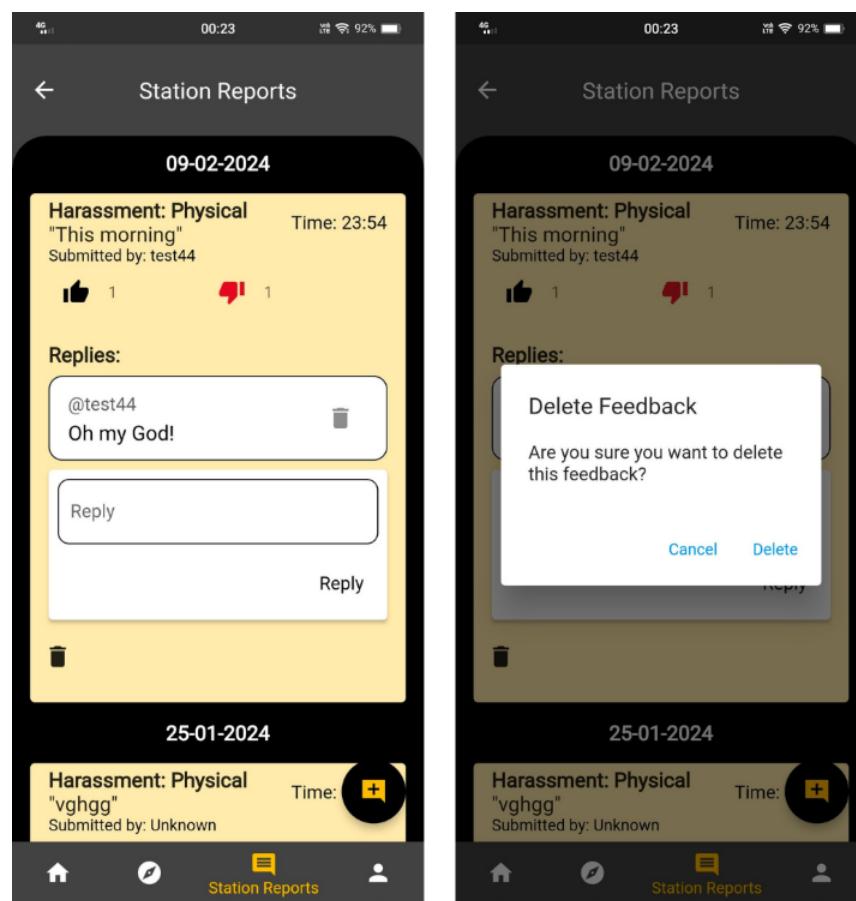


Figure 5.25 Delete Irrelevant Feedbacks and Reports

Similarly, these two processes delineate the primary functions of the MyStation application from the administrative standpoint, aligning with the project's objectives to empower administrators with comprehensive control and oversight capabilities.

5.3 User Acceptance Test

The User Acceptance Test (UAT) stands as a crucial milestone in the development journey of the MyStation mobile application, tasked with assessing its usability, functionality, and overall user experience. This phase aims to collect insights from a representative user base to evaluate the application's efficacy, pinpoint areas for enhancement, and validate implemented features. Utilising a blend of qualitative and quantitative research methodologies, including usability testing and task-based assessments, the UAT process selects participants based on their familiarity with train transportation, ensuring diverse perspectives and comprehensive feedback. Through structured evaluation sessions, participants engage with the application across real-world scenarios, offering invaluable insights into interaction patterns and overall satisfaction levels. The evaluation form is structured into four distinct sections: user interface (UI) and design, performance, functionality, and user feedback. These segments respectively scrutinise the application's design, layout, navigation, stability, responsiveness, core features, task completion, security measures, and user satisfaction levels. Developed on Google Forms, the evaluation form facilitates comprehensive assessment, guiding participants through a series of targeted questions. Figures 5.26-5.29 illustrate the four main sections of the evaluation alongside their respective enquiries, delineating the comprehensive approach undertaken during the UAT phase.

Section B : User Interface (UI) and Design

This section focus on the overall design, layout and navigation of the app including questions about the ease of the use, visual design and any missing features.

How would you rate the overall visual design of the app? *

1 2 3 4

Very unattractive

Very attractive

Were the interface and visual elements clear and aesthetically pleasing? *

Yes

No

Were you able to easily navigate through the app's features? *

Yes

No

If no, please describe the challenges you faced.

Your answer

Figure 5.26 Questions in the User Interface Section

Section C : Performance

This section focus on MyStation's speed, responsiveness and stability.

Please rate the performance of the MyStation app based on the following aspects: *

	Very poor	Poor	Good	Very good
Speed and responsiveness	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Smoothness	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
App loading times	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Overall performance	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Figure 5.27 Questions in the Performance Section

Section D: Functionality

This section focus on the MyStation's core features and how well it work including questions about MyStation's ability to complete specific tasks, how well the features meet the user's needs, and any missing features.

Did all the features and functionalities work as expected? *

- Yes
- No

If no, please specify which features had issues.

Your answer

Please rate the functionality of the MyStation app based on the following features: *

	Not Functional	Somewhat Functional	Functional	Very Functional
Station Information	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Real-time Updates	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
User Reports and Feedbacks	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Figure 5.28 Questions in the Functionality Section Part 1

How well does the app's functionality compare to your initial expectations? *

1 2 3 4

Poor Excellent

How successful were you in completing the tasks you intended to do within the app? *

1 2 3 4

Not at all Completely

Figure 5.29 Questions in the Functionality Section Part 2

Section F: User Feedback

This section focus on the users' overall satisfaction with the app including questions about your experience and if you have any suggestions for improvements.

How well did the app meet your needs? *

1 2 3 4

Poor

Excellent

Are there any features or functionality that you found particularly useful or innovative? *

Your answer

Are there any specific features you would like to see added or improved? *

Your answer

Would you recommend this app to others? *

Yes

No

Figure 5.30 Questions in the User Feedback Section

5.4 Result

In the User Acceptance Test (UAT) of the MyStation mobile application, 18 responses were gathered from frequent train users. The feedback revealed insights into various aspects of the app's usability, functionality, and user experience. Participants provided valuable input on the application's interface design, navigation flow, performance, and core features. Their responses highlighted areas of strength and areas needing improvement, guiding iterative enhancements to ensure MyStation meets the needs and expectations of its target audience effectively. Figure 5.30 and Figure 5.31 below illustrates the percentage breakdown of the types of trains primarily used by participants for their daily commute or travel.

Which type of train do you primarily use for your commute or travel?

18 responses

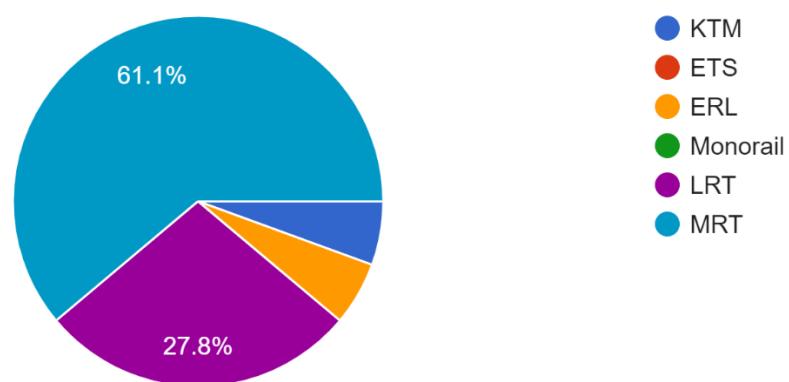


Figure 5.31 Type of Trains Used by Respondents

How often do you use trains (KTM, ETS, ERL, Monorail, LRT or MRT) for your commute or travel?

18 responses

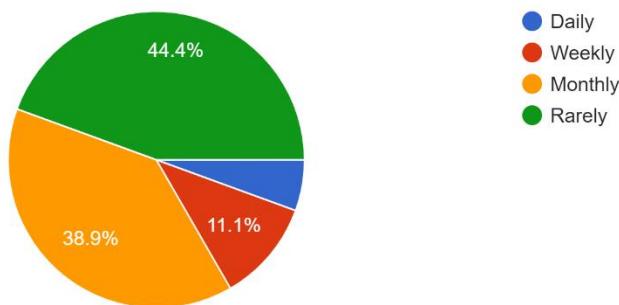


Figure 5.32 The Frequency of Respondents Using Trains

The outcome of the User Acceptance Testing (UAT) is segmented into four distinct categories: user interface (UI) and design, performance, functionality, and user feedback. Various questions within these sections are structured using a Likert scale, while others prompt participants to provide brief textual responses. In the Likert Scale format, respondents rate aspects on a scale from 1, indicating extremely poor, to 4, representing extremely excellent.

5.4.1 User Interface (UI) and Design

The feedback gathering process for the interface revolves around four primary aspects: the overall visual design, clarity and aesthetic appeal of visual elements, ease of navigation, and encountered user challenges. Figures 5.32-5.35 depict the outcomes of the UAT pertaining to the user interface segment.

How would you rate the overall visual design of the app?

18 responses

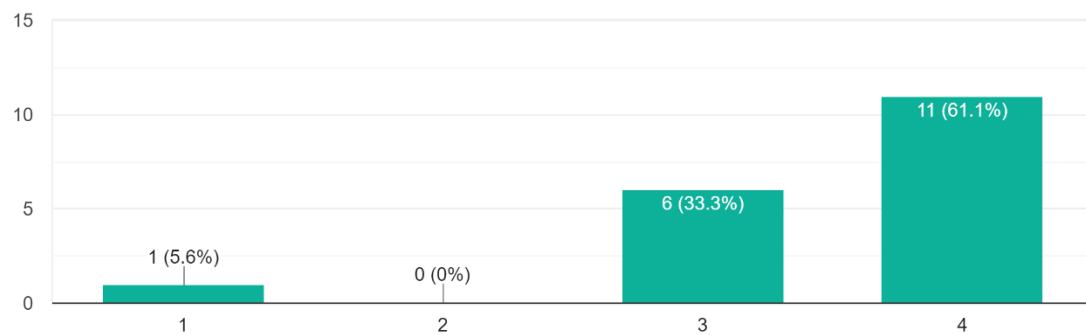


Figure 5.33 Users' rating for the overall visual design

Were the interface and visual elements clear and aesthetically pleasing?
18 responses

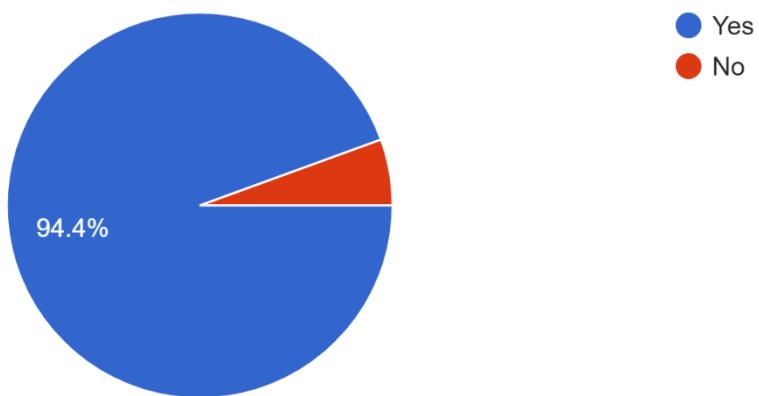


Figure 5.34 Users' Rating on Clarity and Aesthetic Appeal of Visual Elements

Were you able to easily navigate through the app's features?
18 responses

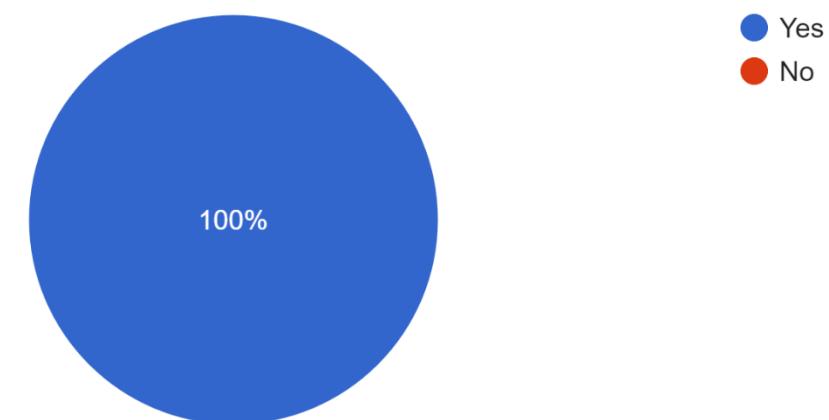


Figure 5.35 Users' Rating on Ease of Navigation

If no, please describe the challenges you faced.

0 responses

No responses yet for this question.

Figure 5.36 Users' Responses on Challenges Faced

5.4.2 Performance

This section delves into four key areas evaluating the performance of the MyStation application, including its speed and responsiveness, smoothness of operation, app loading times, and overall performance. Figures 5.36 showcase the findings derived from the UAT concerning the application's performance.

Please rate the performance of the MyStation app based on the following aspects:



Figure 5.37 Users' Responses on the Performance of the App

5.4.3 Functionality

This section encompasses five main areas of focus, assessing the functionality of the MyStation app. These include enquiries into whether all features and functionalities operated as anticipated, identification of any features encountering issues if applicable, rating the functionality across key features such as Station Information, Real-time Updates, and User Reports and Feedbacks, evaluating the alignment of app functionality with initial expectations, and gauging the success rate of completing intended tasks within the app. Figures 5.37-5.41 detail the outcomes pertaining to the application's functionalities concerning its primary objectives.

Did all the features and functionalities work as expected?

18 responses

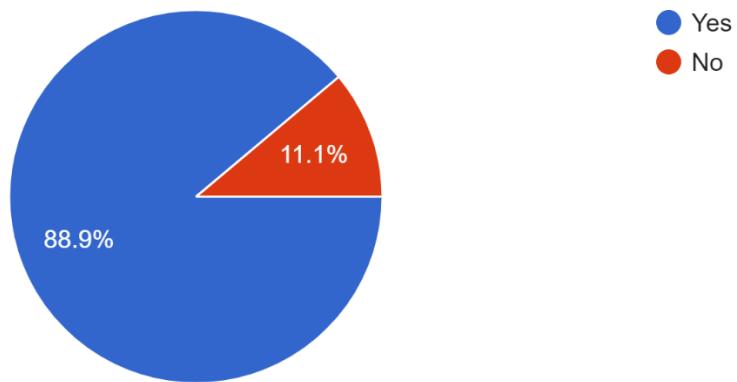


Figure 5.38 Users' Responses on Functionality of the App

If no, please specify which features had issues.

2 responses

Not all button functional but understandable as the project is quite difficult to execute

some of the button cant be click, need to improve the key in words section

Figure 5.39 Users' Comments on the Functionality of the App

Please rate the functionality of the MyStation app based on the following features:

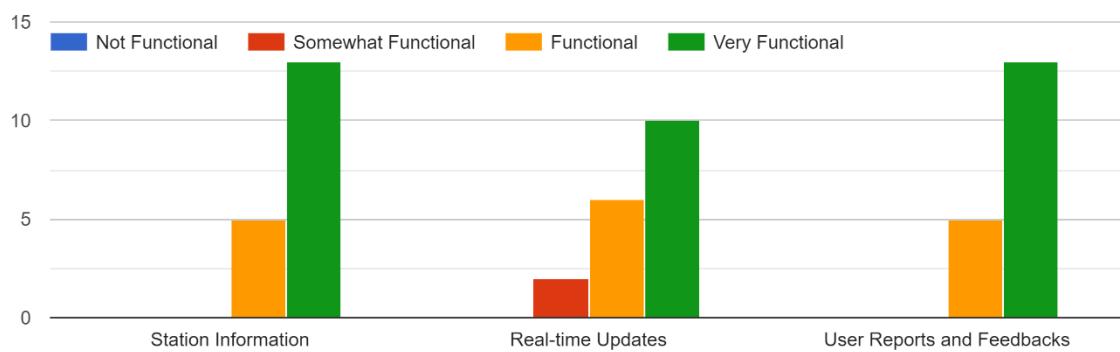


Figure 5.40 Users' Rating on the Features of the App

How well does the app's functionality compare to your initial expectations?

18 responses

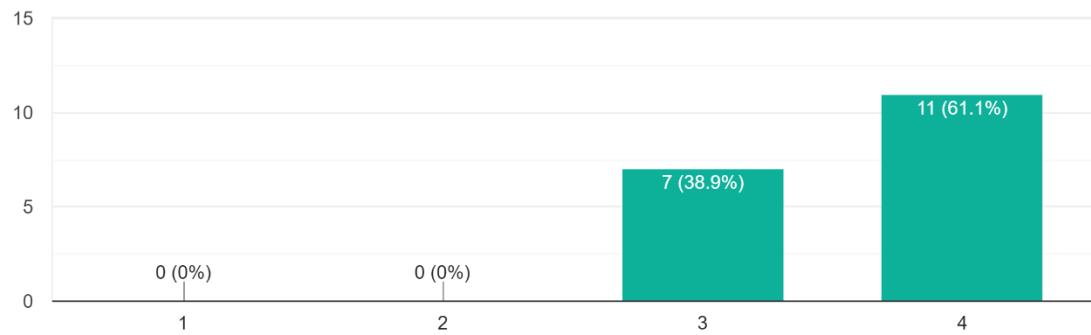


Figure 5.41 Users' Rating on Functionality Compared to Users' Initial Expectation

How successful were you in completing the tasks you intended to do within the app?

18 responses

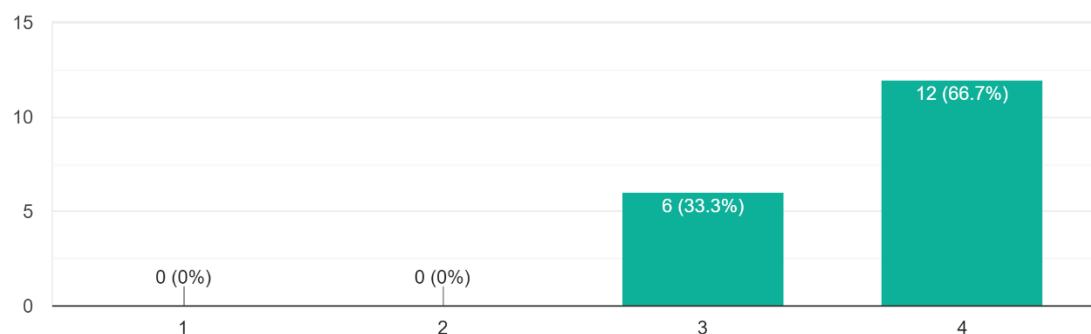


Figure 5.42 Users' Rating on the Successfulness in Completing the Tasks

5.4.4 User Feedback

In the user feedback sections, our attention centered on the holistic user experience throughout application usage. The corresponding findings regarding user feedback are illustrated in Figures 5.42 and 5.43 provided below.

How well did the app meet your needs?

18 responses

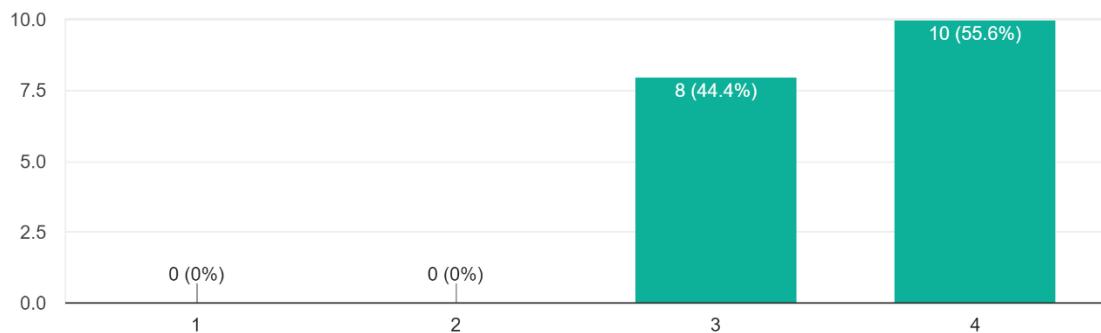


Figure 5.43 Users' Rating on the Effectiveness in Meeting User Needs

Are there any features or functionality that you found particularly useful or innovative?

18 responses

The information for all station in one app which can be accessible to users

The reporting section where other user sees the overall report

Whoever can report and give their opinion about the station situation and environment

No

Yes. The report features

complaint/report function

yes

the services near the station is very useful

The report page

Figure 5.44 Users' Suggestions on the Features to be Added or Improved (i)

Are there any features or functionality that you found particularly useful or innovative?

18 responses

Rating of the each service

directory of shops and toilet etc

Makesure all the buttons are function

the report part is useful

Services available at the station

the real time updates

The report features

maybe more info for every station

Feature that gives notification about the train arrival, that would be great

Figure 5.45 Users' Suggestions on the Features to be Added or Improved (ii)

Are there any specific features you would like to see added or improved?

18 responses

no

The train time

No. Everything's okayy

No

Maybe can add navigation assist so that there will be guidance for someone who is not familiar with any transportation app

already nice

a more fresh UI, and add more stations

Faster loading time

Would be nice if we can just touch our phones on the scanner to pay for train tickets

Figure 5.46 Users' Suggestions on the Features to be Added or Improved (iii)

Are there any specific features you would like to see added or improved?

18 responses

Would be nice if we can just touch our phones on the scanner to pay for train tickets

-

add a user custom feedback

I think its a good apps

Feedbacks for other inquiries

train arrival time

No for now

maybe gps?

The loading image of the station images should be improve

Figure 5.47 Users' Suggestions on the Features to be Added or Improved (iv)

Would you recommend this app to others?

18 responses

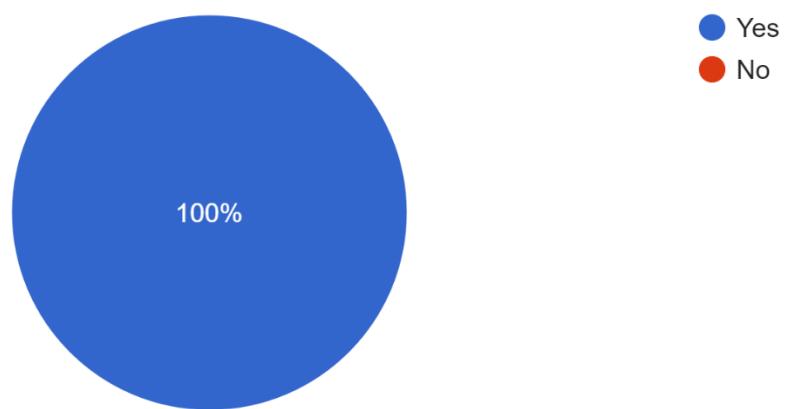


Figure 5.48 Users' Choice on the Likelihood of Recommending the App to Others

5.5 Discussion

The UAT results shed light on the strengths and weaknesses of the MyStation mobile application. In evaluating the user interface, it is evident that users generally expressed satisfaction with the overall visual design, ease of navigation, and the functionality of icons and buttons. Table 5.1 provides a detailed breakdown of the feedback across each aspect of the user interface. the overall visual design, clarity and aesthetic appeal of visual elements, ease of navigation, and encountered user challenges.

Table 5.1 Summary of User Feedback on Application User Interface

USER INTERFACE	
Focus	Rating (%)
Overall visual design	87.5
Clarity and aesthetic appeal of visual elements	94.4
Ease of navigation	100

For the performance section, the evaluation revealed favourable ratings across various metrics. The application demonstrated strong performance in terms of speed and responsiveness, garnering a rating of 88.89%. Users reported high satisfaction with the smoothness of interactions, with a rating of 93.06%. Additionally, the app loading time received a respectable rating of 84.72%. Overall, the application's performance was well-received, achieving a rating of 91.67%. These results indicate a robust performance that contributes to a positive user experience. To enhance understanding, the following Table 5.2 visually represents the information conveyed.

Table 5.2 Summary of User Feedback on Application Performance

PERFORMANCE	
Focus	Rating (%)
Speed and responsiveness	88.89
Smoothness	93.06
App loading time	84.72
Overall performance	91.67

In the functionality section, the application aligns with user requirements, as indicated by 88.9% of respondents reporting that all features and functionalities operate as anticipated. Moreover, the application demonstrates a satisfactory performance regarding bug issues, accounting for 11.1% of responses. The breakdown of points for the functionality section is presented in Table 5.3.

Table 5.3 Summary of User Feedback on Application Functionality

FUNCTIONALITY	
Features	Rating (%)
Station information	93.06
Real-time updates	86.11
User reports and feedbacks	93.06
Functionality compared to users' initial expectation	90.28
Successfulness in completing the tasks	91.67

Lastly for the user feedback section, the result is unexpectedly high. The responses to the question regarding particularly useful or innovative features or functionalities varied among participants. Some highlighted the convenience of accessing information for all stations within a single app, while others appreciated the reporting section's transparency and community engagement aspect. Features such as the ability for any user to report and provide feedback on station conditions were also noted positively. However, one respondent indicated no specific feature, while others mentioned the usefulness of the report function, access to nearby services, real-time updates, and the directory of shops and amenities. Suggestions for improvement included providing more comprehensive station information and implementing features like train arrival notifications for enhanced convenience. Overall, respondents appreciated features that streamlined access to information and facilitated interaction with the app's reporting functionalities. Table 5.4 below shows the overall user experience and other feedback.

Table 5.4 Users' Feedback Summary

USER FEEDBACK	
Focus	Rating (%)
Effectiveness in meeting user needs	88.89
The likelihood of recommending the app to others	100
Any features to be added or improved	<ol style="list-style-type: none"> 1. The train times 2. Maybe can add navigation assist so that there will be guidance for someone who is not familiar with any transportation app 3. a fresher UI, and add more stations 4. Faster loading time 5. Would be nice if we can just touch our phones on the scanner to pay for train tickets 6. add a user custom feedback 7. Feedbacks for other enquiries 8. train arrival time 9. maybe GPS? 10. The loading image of the station images should be improved

Based on the comprehensive examination of each section's results, it is evident that the application offers a high level of convenience and utility to its users. The core functionalities, including accessing station information, receiving real-time updates, and submitting/viewing station reports and feedback, are well-implemented and effectively fulfil users' primary needs. Additionally, the administrative capabilities, such as managing station images, filtering user feedback and reports, further enhance the application's functionality. However, certain areas for improvement have been identified, notably the need for more extensive details regarding train arrival and departure times, the absence of a user manual for individuals unfamiliar with transportation apps, and the limitation to the Android platform, potentially excluding users of other operating systems.

CHAPTER 6 CONCLUSION AND FUTURE WORK

6.1 Conclusion

In conclusion, the MyStation Train Station Information Mobile Application has unequivocally met its objectives, providing users with a robust platform to access essential information, navigate station facilities, and actively participate in community-driven enhancements. By prioritising visual representations, real-time updates, and user feedback, the application has cultivated a user-friendly environment conducive to collaboration among commuters. Moreover, the integration of Firebase as the cloud database has facilitated efficient data management, underpinned the application's seamless functionality and enhanced the overall user experience. Through these concerted efforts, MyStation has successfully established itself as a valuable tool for commuters seeking convenience, engagement, and efficiency in their daily travels.

6.2 Limitations

While MyStation Train Station Information Mobile Application has demonstrated notable success in achieving its objectives and fostering user engagement, certain limitations persist. Specifically, the application currently lacks comprehensive information regarding train arrival and departure times, potentially hindering users seeking precise scheduling details. Moreover, the absence of a user manual may pose challenges for individuals unfamiliar with transportation apps, potentially impacting usability and accessibility. Additionally, the application's restriction to the Android platform presents a barrier for users of other operating systems, limiting its reach and potential user base. Addressing these limitations could further enhance the application's functionality, accessibility, and overall user experience, ensuring broader utility and appeal among commuters.

6.3 Future Works

In future iterations, several enhancements could elevate the functionality and user experience of the MyStation Train Station Information Mobile Application. Firstly, prioritising the provision of comprehensive and real-time train schedule information would address a key user need and enhance the application's utility. Additionally, improving navigation assistance features could benefit users, particularly those less familiar with transportation apps, by providing clearer guidance and enhancing usability. Furthermore, exploring the development of versions compatible with iOS and other operating systems would expand the application's reach and accessibility, ensuring it caters to a broader user base and maximises its impact within the commuting community. These future works present exciting opportunities to further refine and evolve the MyStation app, solidifying its position as a valuable resource for commuters seeking efficient and reliable train station information.

6.4 Personal Reflections

Embarking on this project has been a transformative experience, marked by both hurdles and personal development. It has offered invaluable insights into the intricacies of mobile application development and the importance of prioritising user-centric design principles. Integrating visual elements and real-time feedback mechanisms has highlighted the paramount role of empathy and iterative design in crafting applications that resonate with users. Moreover, this endeavour has served as a platform for enhancing technical proficiencies, from mastering Dart within Android Studio to adeptly managing cloud databases. Overall, this project has not only broadened my skill set but also deepened my appreciation for the iterative nature of software development and the profound impact of user-focused design choices.

REFERENCES

- [1] C. Romero, A. Monzón, A. Alonso and R. Julio, "Added Value Of A Customised Transit App For Metropolitan Bus Trips," *Transportation Research Procedia*, vol. 47, pp. 513-520, 2020.
- [2] J. Y. Awali, I. A. Kautsar and A. S. Fitran, "Citizenapp: Yet Another Social Media Platform For Public Facility Monitoring In Sidoarjo," *Jurnal Teknoinfo* , vol. 16, no. 2, pp. 408-419, 2022.
- [3] Y. Senarath, S. Nannapaneni, H. Purohit and A. Dubey, "Emergency Incident Detection From Crowdsourced Waze Data Using Bayesian Information Fusion," *2020 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, pp. 187 - 194, 2020.
- [4] L. L. Losada-Rojas, C. Gkartzonikas, V. D. Pyrialakou and K. Gkritza, "Exploring Intercity Passengers' Attitudes And Loyalty To Intercity Passenger Rail: Evidence From An On-Board Survey," *Transport Policy*, vol. 73, pp. 71-83, 2019.
- [5] G. A. Widoseno and T. Hidayatullah2, "Influence Of Public Transportation Applications For Foreign Tourists," *Proceedings of the 1st International Conference on Informatics, Engineering, Science and Technology, INCITEST 2019*, 18 July 2019, Bandung, Indonesia, 2019.
- [6] R. Ramesh, G. Divya, A. Jayan, S. V. Maju and R. Varghese, "Offline Mobile Application For Train Time Prediction," *2019 International Conference on Communication and Electronics Systems (ICCES)*, pp. 1159-1163, 2019.
- [7] R. E. Castillo, P. J. M. Castro, . M. C. R. Aragon and H. C. Macugay, "Tip Express: An Android School Navigation Application," *International Journal of Computer and Communication Engineering*, vol. 7, no. 3, pp. 98-106, 2018.
- [8] B. Mathur and S. M. Satapathy, "An Analytical Comparison Of Mobile Application," *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, pp. 1147-1152, 2019.

- [9] K. Singh, "Agile Methodology for Product Development: A Conceptual Study," *Int. J. Recent Technol. Eng.*, vol. 10, no. 1, pp. 209-215, 2021.
- [10] P. M. Berhad, "MyRapid Pulse (Version 3.0.4)," [Mobile Application Software], 2020. [Online]. Available: <https://play.google.com/store/apps/details?id=com.prasarana.pulse>.
- [11] Moovit, "Moovit: Bus & Train Schedules (Version 5.119.0.581)," [Mobile Application Software], 2012. [Online]. Available: <https://play.google.com/store/apps/details?id=com.tranzmate>.
- [12] P. Solutions, "Malaysia MRT/LRT (Offline) (Version 4.3)," [Mobile Application Software], 2020. [Online]. Available: <https://play.google.com/store/apps/details?id=pb.mrt.malaysiarailmap>.