WANDERWISE – TRAVEL COMPANION MOBILE APPLICATION

Nur Athira Binti Azman

(78368)

Bachelor of Software Engineering with Honours

2024

UNIVERSITI MALAYSIA SARAWAK

THESIS STATUS ENDORSEMENT FORM

TITLE WANDERWISE – TRAVEL COMPANION MOBILE APPLICATION

ACADEMIC SESSION: 2024/2025

NUR ATHIRA BINTI AZMAN
(CAPITAL LETTERS)

hereby agree that this Thesis* shall be kept at the Centre for Academic Information Services, Universiti Malaysia Sarawak, subject to the following terms and conditions:

1. The Thesis is solely owned by Universiti Malaysia Sarawak
2. The Centre for Academic Information Services is given full rights to produce copies for educational purposes only
3. The Centre for Academic Information Services is given full rights to do digitization in order to develop local content database
4. The Centre for Academic Information Services is given full rights to produce copies of this Thesis as part of its exchange item program between Higher Learning Institutions [or for the purpose of interlibrary loan between HLI]
5. ** Please tick (√)

☑	CONFIDENTIAL	(Contains classified information bounded by the OFFICIAL SECRETS ACT 1972)
☑	RESTRICTED	(Contains restricted information as dictated by the body or organization where the research was conducted)
☑	UNRESTRICTED	

(AUTHOR'S SIGNATURE)

Permanent Address
32, Persiaran Perdana 13,
Taman Pinji Perdana,
31500, Lahat,
Perak

Date: 27/07/2025

Validated by

(SUPERVISOR'S SIGNATURE)

Ts. Nurfauza Binti Jali
Senior Lecturer
Faculty of Computer Science and Information Technology
Universiti Malaysia Sarawak

Date: 27 July 2025

Note * Thesis refers to PhD, Master, and Bachelor Degree
** For Confidential or Restricted materials, please attach relevant documents from relevant organizations / authorities

ACKNOWLEDGEMENT

I would like to extend my deepest gratitude to my supervisor, Ts. Nurfauza binti Jali, for her invaluable guidance, constructive feedback, and unwavering support throughout the development of WanderWise, my Final Year Project. Her expertise and dedication have been instrumental in shaping this project and bringing it to completion. My sincere appreciation also goes to the Final Year Project coordinator, Professor Dr Wang Yin Chai, for providing clear briefings, comprehensive guidelines, and constant support during the course. His structured guidance created a solid foundation for the project's progression and ensured that we remained on track throughout the semester.

I would also like to express my sincerest gratitude to Dr Tan Ping Ping as my examiner for her constructive feedback and guidance that has also shaped WanderWise to what it is today. Special thanks to all 31 usability testing participants who took the time to download, explore, and provide feedback on the WanderWise app. Their insights were crucial in refining the system's functionality and enhancing user experience.

To my family and friends, your encouragement, understanding, and patience have meant the world to me. To Bonnay, thank you for being my pillar of emotional support during moments of doubt and exhaustion. Lastly, I am thankful for the challenges, late nights, and breakthroughs that this project has brought. This journey has not only improved my technical capabilities but also deepened my resilience and determination.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Travelling has become an integral part of modern lifestyles, with travellers often relying on multiple applications to manage various aspects of their journeys, such as planning itineraries, converting currencies, and seeking local travel tips. This fragmentation leads to inefficiencies and a less enjoyable travel experience. To address this challenge, WanderWise, a comprehensive travel companion application, is introduced. Designed as an all-in-one platform, WanderWise integrates essential travel tools, including a currency converter, itinerary planner, and an AI-powered travel assistant, along with a minimal social travel forum for sharing and downloading itineraries. The app aims to streamline travel planning and execution by reducing the need to juggle multiple apps, enabling users to focus on enjoying their journeys. The development of WanderWise follows an Agile methodology, with iterative improvements based on user feedback. Usability testing and feature evaluations ensure that the application meets the needs of modern travellers, providing convenience and enhancing the overall travel experience. Through the development of WanderWise, travellers can experience a unified and efficient approach to travel planning.

CHAPTER 1: INTRODUCTION

1.1.Introduction

In 2021, individuals in Malaysia downloaded approximately 1.33 billion mobile apps (Statista, 2024), highlighting a growing demand for smartphone usage in recent years. With the rapid development of mobile applications, Mobile Travel Apps (MTA) have also seen a positive impact. There is a shift towards this field, especially since tech-savvy travellers often depend on wearable devices, smartphones, and tablets during their holidays. This culture of bring-your-own-devices (BYOD) shows that travellers often utilise these tools to learn about their holiday destination, explore the location, and ease their holiday experience (Sia et al., 2022).

However, even with the high demand for mobile applications, there is still a lack of unified mobile applications that have all the essential needs a traveller needs during a holiday. Therefore, a more comprehensive travel companion application is proposed. The WanderWise mobile application is designed to transform the modern travel experience by consolidating essential tools and resources into one convenient platform. In today's fast-paced world, travellers are often burdened with organising multiple travel components separately, from managing currency exchanges to planning itineraries and seeking travel tips. WanderWise addresses this need by offering a range of streamlined features tailored to simplify and personalise every journey, whether for seasoned globetrotters or occasional explorers.

WanderWise combines functionality and accessibility by providing tools like currency conversion, an intuitive itinerary planner, and a social travel forum that allows users to share and download itineraries and travel tips. The application also integrates AI-powered travel assistance to offer recommendations and support, ensuring that each user enjoys a smooth and

well-guided travel experience. With its minimalistic social component, WanderWise caters to travellers looking for insights without the clutter of traditional social platforms, encouraging a focused and productive sharing of travel advice and itineraries.

1.2.Problem Statement

While the rapid development of mobile technology has created a surge of applications to enhance the travel experience, there remains a lack of a comprehensive solution that addresses all essential travel needs in one place. Moreover, tourism research has highlighted the significant role and impact of information and communication technologies (ICTs) on the industry, emphasising their positive influence (Sustacha et al., 2022). Travellers are increasingly tech-savvy and accustomed to using smartphones and other devices to access information quickly and conveniently. Despite this, the absence of a single, unified platform often leaves them burdened with the need to switch between apps, complicating what should be a seamless experience.

In developing an MTA, there are a number of features that should be focused on. However, AI-based travel assistance, integrated personalised recommendations, and integrated destination videos for better engagement should be the highlights for the development of a travel planner application (Nimbalkar, 2020). WanderWise aims to bridge this gap by offering a consolidated travel companion app that integrates key tools for planning and enhancing journeys. With features such as an intuitive itinerary planner, currency conversion, and a social travel forum for sharing and downloading itineraries and tips, WanderWise is designed to simplify travel planning. Additionally, its minimal social component fosters a focused environment for exchanging insights, free from the distractions of conventional social platforms.

1.3.Scope

The WanderWise project is a mobile application designed to simplify and improve the travel experience for budget-conscious travellers, group travellers, and solo adventurers. By combining essential tools such as a currency converter, itinerary planner, travel forum, and AI-powered chatbot into one accessible platform, WanderWise addresses the need for a unified travel solution. The app allows users to manage trip planning efficiently and ensures data security throughout their journey. With real-time assistance and personalised recommendations from the AI chatbot, users gain confidence while exploring unfamiliar places. Additionally, the app fosters a community forum where travellers can share itineraries and tips, creating a supportive, connected environment for a more enriching travel experience. However, there is a limitation to this project whereby there is a need for internet access at all times. The application is not available for offline settings due to its AI-powered chatbot nature.

1.4.Aims and Objective

The main reason for developing this project is to understand the gaps of the fragmented tools and applications in relation to a good travelling experience. Furthermore, this project serves as an opportunity to enhance the travelling experience of each traveller and introduce vital features and key trends for an MTA that appeal to them. The aims and objectives of this project are as follows:

a. To study and analyse existing systems of travel planning and itinerary management applications.
b. To design and implement a comprehensive mobile application that integrates essential travel tools.
c. To test and evaluate the usability and functionality of the WanderWise app through user feedback and usability testing.

1.5.Brief Methodology (Agile)

The methodology that was chosen for this project is Agile, in particular Condensed Agile.

Figure 1. Agile Methodology Cycles (Laoyan, 2024)

In particular, the condensed agile methodology is chosen for this project rather than other methodology models. Although Rapid Application Development (RAD) comes close to the Agile concepts, there were a few rationales for the chosen methodology. As the project requires more than just simple user interfaces and functionalities, Agile fits best for its complex features. Furthermore, Agile is a software development methodology that emphasises iterative development and adaptability to change (Andruszków, 2024) and is more focused on breaking down the development cycle into smaller 'sprints.' Condensed Agile, on the other hand, has a faster sprint timeline than the typical sprint in an Agile Methodology.

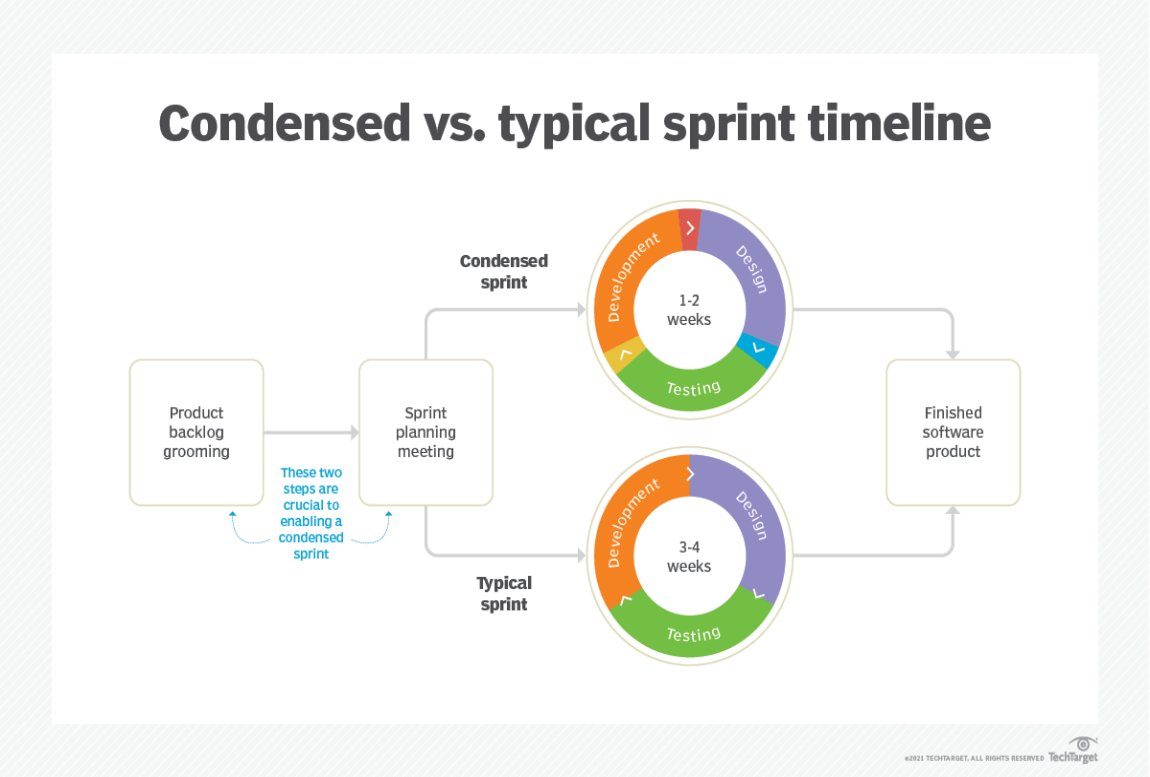


Figure 2. Condensed Agile Methodology's Timeline (Reichert, 2021)

1.5.1 Phases in Condensed Agile

1.5.1.1. Plan

In this first phase of the development cycle, the project's scope, limitations, and requirements are decided and planned out. Moreover, the sprint is done by planning the tasks and objectives for the upcoming sprints as well as the sprint backlog.

1.5.1.2. Design

During this phase, the overall design of the project is outlined to ensure that the idea of the project is well-aligned with all stakeholders involved. This would also help out in the development phase as a reference.

1.5.1.3. Develop

This phase is the longest phase, as the product is being developed. All the requirements of the project are being implemented, and it goes under several reviews and refinements until finalised.

1.5.1.4. Test

In this phase, the final working product is ready to be tested, and user testing is conducted. Different from waterfall testing, agile testing is executed alongside the development, and this allows for flexibility (GeeksforGeeks, 2024). This phase ensures that the project meets the requirements as well as helps to detect any bugs before the deployment phase.

1.5.1.5. Deploy

The final product is ready to be deployed. In this phase, there will be regular releases and improvements through user feedback.

1.5.1.6. Review

In this final phase, the product is being maintained after deployment. Users will be provided with regular support to ensure that the product is running smoothly without any bugs.

1.6.Significance of Project

The significance of WanderWise addresses a common challenge faced by modern travellers, who face issues with managing multiple applications and fragmented tools for various travel needs. This mobile application can offer a unified platform wherein it simplifies the travel experience, enabling users to access essential features such as currency conversion,

itinerary planning, and social travel forums, and introduce an AI-powered chatbot to ensure a good experience during travelling.

This integrated approach not only enhances user convenience but also promotes a more efficient and cohesive travel experience. This reduces the need for multiple apps and ensures smoother transitions between different aspects of their journey.

1.7. Project Schedule

Figures below show the overall project schedule as well as a Gantt Chart that depicts the timeline of the whole project.

Tasks	Start	Due	Start on Day*	Duration*
FYP1				
Submission of Approved Brief Proposal	15/10/2024	20/10/2024	0	5
Supervisors and Examiners Reviews	21/10/2024	28/10/2024	6	7
Submission of Approved Full Proposal	29/10/2024	14/11/2024	14	16
Submission of Chapter 1	15/11/2024	21/11/2024	31	6
Submission of Chapter 2	22/11/2024	13/12/2024	38	21
Submission of Chapter 3	14/12/2024	05/01/2025	60	22
Submission of FYP 1 Final report & Paper	06/01/2025	12/01/2025	83	6
FYP2				
Planning and Requirement Gathering	13/01/2025	19/01/2025	90	6
System Design & Initial Prototype	20/01/2025	26/01/2025	97	6
DEVELOPMENT AND IMPLEMENTATIONS				
Sprint 1: Developing AI powered chatbot	27/01/2025	10/02/2025	104	14
Sprint 2: Homepage/Basic UI Implementations	11/02/2025	25/02/2025	119	14
Sprint 3: Itinerary Planner Feature Development	26/02/2025	12/03/2025	134	14
Sprint 4: Minimal Travel Forum Feature Development	13/03/2025	27/03/2025	149	14
Sprint 5: Currency Conversion Feature Development	28/03/2025	10/04/2025	164	13
Sprint 6: Refine Features Based on Cumulative Feedback	11/04/2025	24/05/2025	178	43
User Testing & Final Adjustments	25/05/2025	04/06/2025	222	10
Documentations and Final Submission	07/06/2025	16/06/2025	235	9
Final Refinements and Submission of FYP2 Final Report and Presentation	17/06/2025	29/06/2025	245	12

Figure 3. Project Schedule

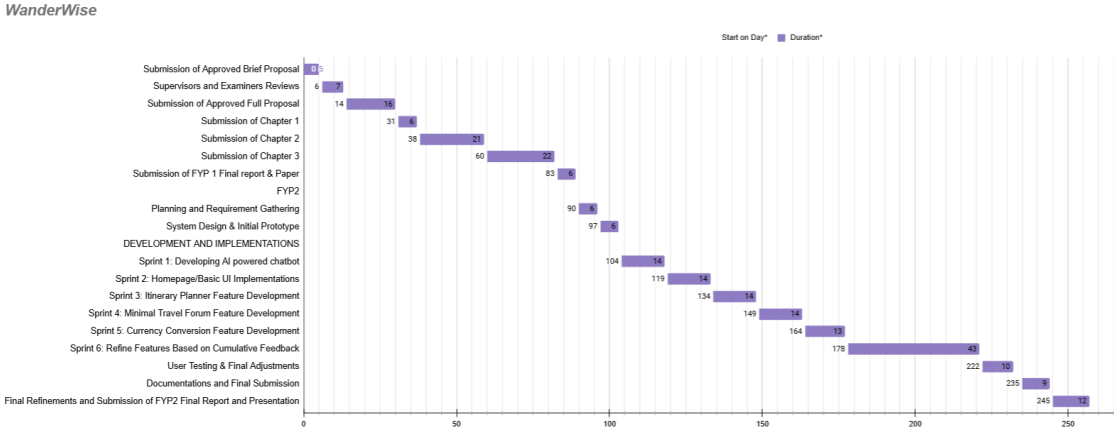


Figure 4. Gantt Chart

1.8.Expected Outcome

The successful implementation of the WanderWise app create a fully integrated travel companion that elevates user experience, streamlines trip planning, and nurtures a community of travellers sharing valuable insights and resources. Through this application, users will gain access to essential travel tools all in one place, including a real-time currency converter, a customisable itinerary planner with collaboration options, and a travel forum designed for minimal but meaningful interactions. This forum will enable travellers to exchange practical tips and insights, enhancing each journey with firsthand knowledge from other WanderWise users.

Additionally, the AI-powered chatbot feature will empower users to generate tailored itineraries, access destination-specific information, and discover must-visit attractions and experiences. With this feature, WanderWise users can enjoy a seamless travel experience with 24/7 assistance at their fingertips, offering a sense of security and ease—especially valuable when exploring unfamiliar destinations. Personalised recommendations from the AI chatbot will cater directly to each user's preferences, which they can seamlessly incorporate into their travel plans.

By bringing together these features in a single platform, WanderWise will offer travellers a unified and streamlined approach to managing all their travel needs while enabling personalised recommendations. This combination will not only simplify travel management but also enrich each journey with curated insights, fostering a more connected and efficient travel experience.

1.9.Project Outline

In this report, each chapter clearly explains respective aspects of the project.

i. Chapter 1

This chapter discusses the project title, introduction, problem statement, objectives, methodology used, definition of the scope, significance of the project, project schedule, and expected outcome. It summarises the overall view of the project.

ii. Chapter 2

Chapter 2 describes the literature review as well as the comparison and review of similar existing systems. This part also highlights the tools and technology.

iii. Chapter 3

In this chapter, the project's methodology is explained in detail as compared to the brief explanation in Chapter 1. Unified Modelling Language (UML) diagrams such as use case diagrams, activity diagrams, and other diagrams will be explained to give a better understanding of the conceptual framework of the project. Furthermore, this chapter also demonstrates a prototype cycle.

iv. Chapter 4

Chapter 4 describes the installation and configuration used in building this project. A detailed description of the application and database used is provided, as well as the workflow, and the modules of the system's application are explained.

v. Chapter 5

In this chapter, the functional testing and non-functional testing are discussed with explanations on system testing, unit testing, and usability testing. The system functionality and user interface design are also reviewed.

vi. Chapter 6

This chapter describes the objectives, achievements, project limitations, conclusion, and future works. The overall progress of the project, its achievements, and any work done are reviewed and concluded.

1.10. Summary

This chapter gives a brief overview of the project's details. The problem statement, objectives, methodology, scope, and significance of the project are outlined. A clear visualisation of the project's schedule is given, and the expected outcome of the project is defined.

CHAPTER 2: LITERATURE REVIEW

2.1. Introduction

For this chapter, three existing mobile applications that have similar features and functionality as the proposed application, WanderWise, will be introduced. All of which will be carefully compared and reviewed to better understand the strengths and weaknesses as well as the existing gaps that lie for each of the mobile applications. Additionally, there will be a comprehensive review on the tools and technology, including the software and hardware that will be used.

2.2. Overview of Objectives

The primary goal of this project is to identify and address the strengths, shortcomings, and opportunities of current existing fragmented travel tools and applications to enhance the overall travel experience. Most travel companion applications lack the capabilities for real-time multi-user planning and updates, which causes disruption in streamlining everyone's efforts in planning trips either individually or by groups (Wong & Wahid, 2024). By analysing current travel planning and itinerary management systems, WanderWise aims to design and implement an all-encompassing mobile application, which integrates essential travel tools. Furthermore, the project seeks to evaluate and refine the app's usability and functionality through extensive user feedback and usability testing, ensuring that WanderWise meets the needs and expectations of travellers by incorporating vital features and key trends.

2.3. Review of Similar Existing Systems

This section reviews the three existing mobile applications that provide similar functions, which are Google Trip, TripIt, and Wanderlog. The mobile applications' overall functionalities and features offered will be analysed to better understand the strengths and weaknesses. This

analysis will also identify any existing gaps within these mobile applications, offering insights that can be leveraged to enhance the proposed application, WanderWise.

2.3.1. Google Trips

Google Trips was a travel planning mobile application developed by Google to assist users in organising their trips more efficiently (Barker, 2024). The app, released in 2016, was designed as an all-in-one travel planner and itinerary management tool. Although it was discontinued in 2019, now being integrated into Google Travel, it remains a benchmark for understanding what a comprehensive travel assistant app can offer. Its primary appeal lay in its ability to automatically gather travel details, such as flight bookings, hotel reservations, and car rentals, directly from a user's Gmail account. This automation eliminated manual entry and provided a seamless way to consolidate travel information in one place. The app also excelled in day-to-day trip management, offering curated daily itineraries tailored to specific destinations, which included recommendations for top attractions, restaurants, and activities. These itineraries were customisable, allowing users to add, remove, or rearrange activities based on their preferences. Additionally, Google Trips provided offline access to all trip details, including maps and travel plans, making it especially useful for international travellers or those visiting areas with unreliable internet connectivity. Another key feature was its integration with Google Maps, which enabled accurate navigation and location tracking, further enhancing the travel experience, as shown in Figure 5.

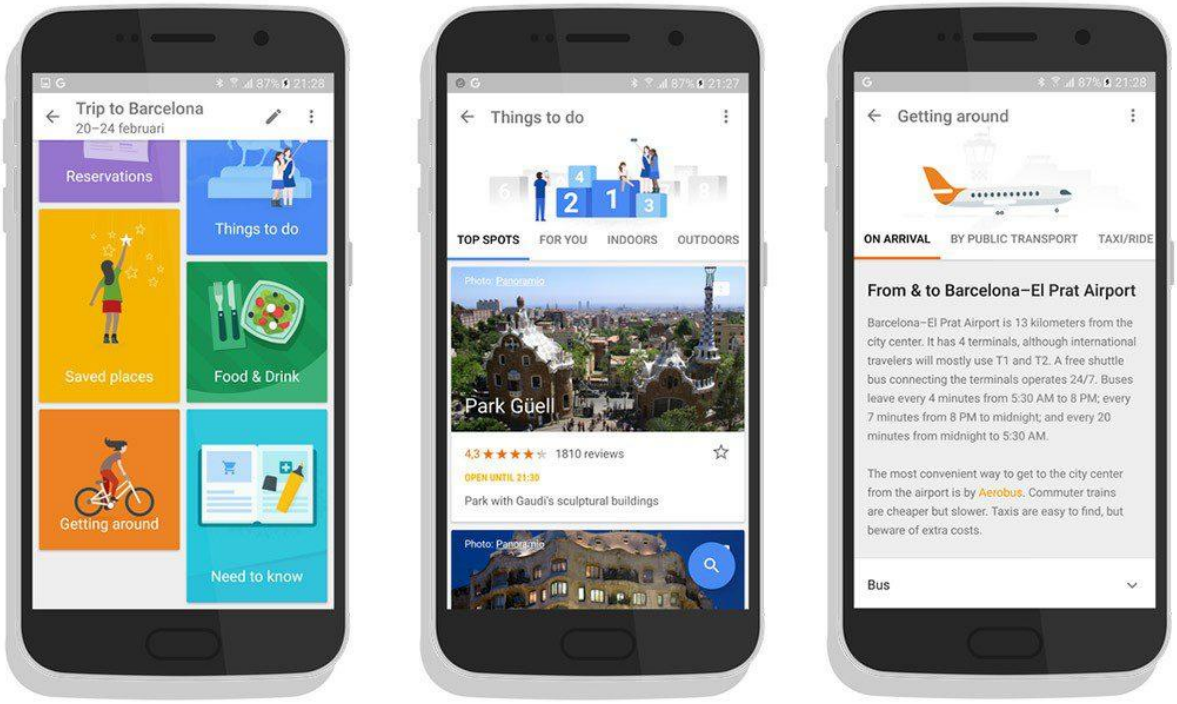


Figure 5. Google Trips features

However, in 2019, Google decided to discontinue Google Trips and integrate its functionalities into a web-based platform called Google Travel. Accessible through Google Search, Google Travel inherits many of the hallmark features of Google Trips while expanding on them to create a more comprehensive and accessible service. Like its predecessor, Google Travel consolidates flight, hotel, and activity bookings into a centralised platform, allowing users to manage their trips effortlessly. It builds upon Google Trips' day planning by providing real-time updates on travel schedules and recommendations tailored to user preferences. The platform also leverages Google's advanced search capabilities to offer extensive flight options, accommodations, and activities, complete with pricing comparisons and detailed reviews. While primarily designed for online use, Google Travel integrates seamlessly with other Google services such as Google Maps and Google Calendar, ensuring a holistic and connected travel planning experience.

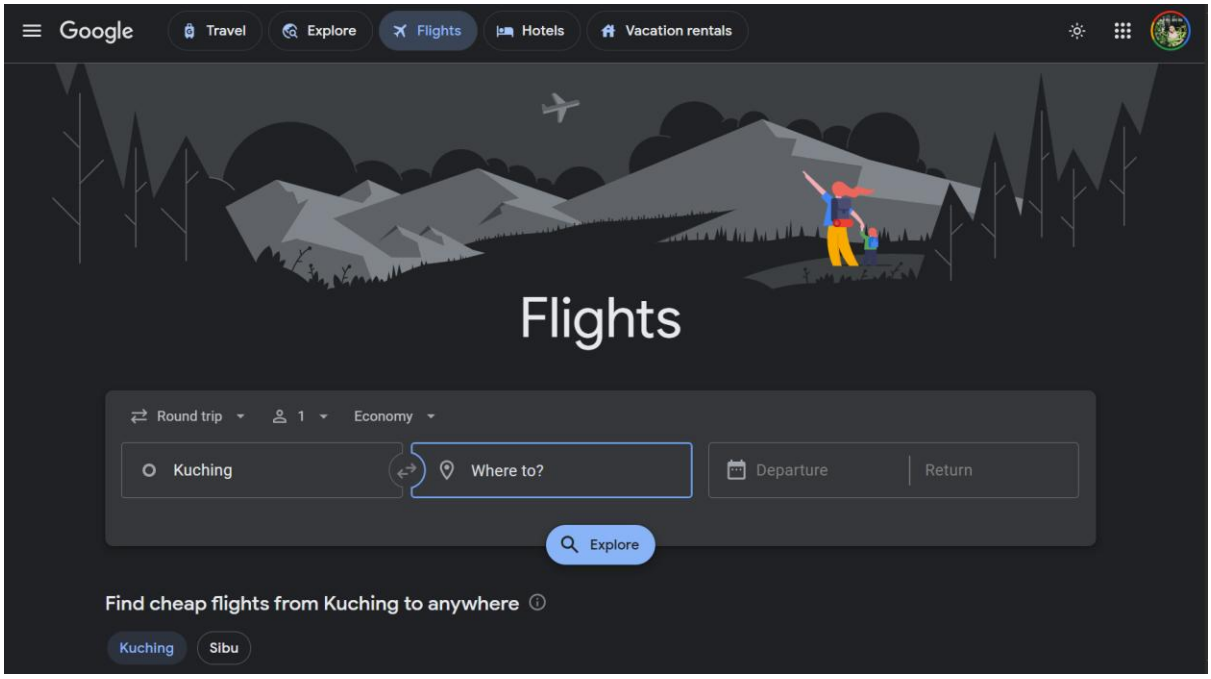


Figure 6. Google Travel's (Integration of Google Trips) general functionalities

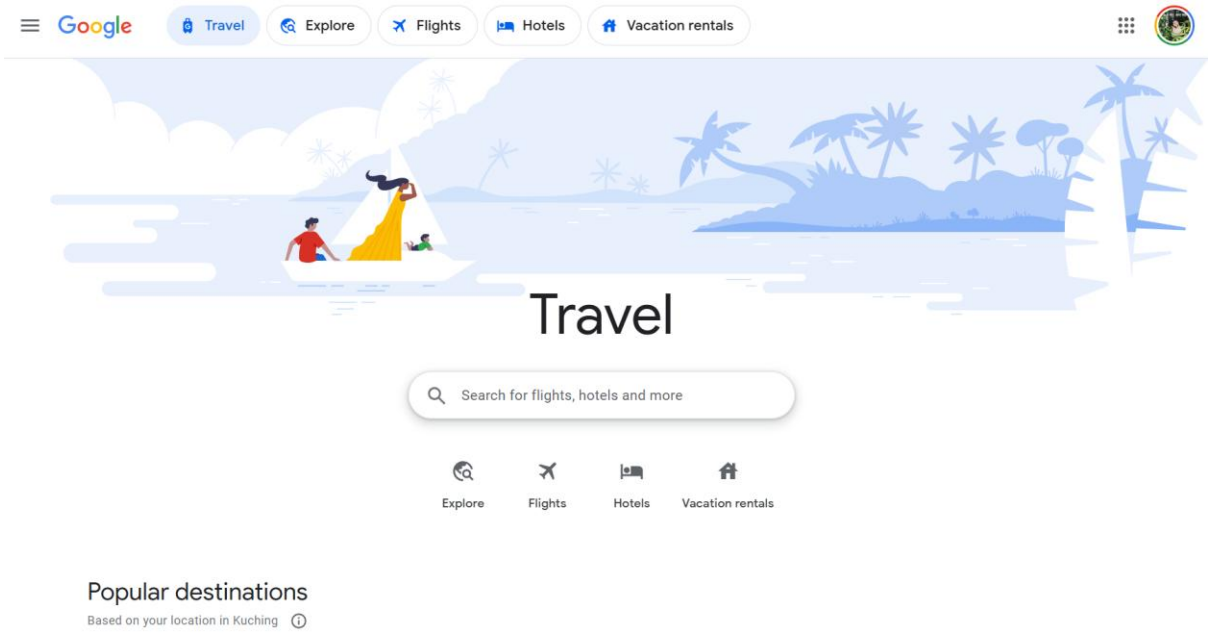


Figure 7. Google Travel's travel functionalities, which include booking flights and hotels and features for search functions

2.3.2. TripIt

TripIt is a highly rated travel planning and itinerary management application with features that allow users to keep track of past travels (Olson, 2024), the application is designed to simplify the travel experience. By forwarding your travel confirmation emails to TripIt, the app automatically organises all your reservations, including flights, hotels, car rentals, and activities, into a comprehensive itinerary. This feature ensures that all your travel details are easily accessible in one place, reducing the stress of managing multiple bookings.

TripIt offers real-time updates and alerts on flight statuses, delays, and gate changes, helping travelers stay informed and make necessary adjustments on the go. The app also provides detailed maps of airports, showing terminals, gates, and amenities, which aids in navigating unfamiliar airports. For international travelers, TripIt includes country-specific travel information and reminders for important tasks like passport renewals.

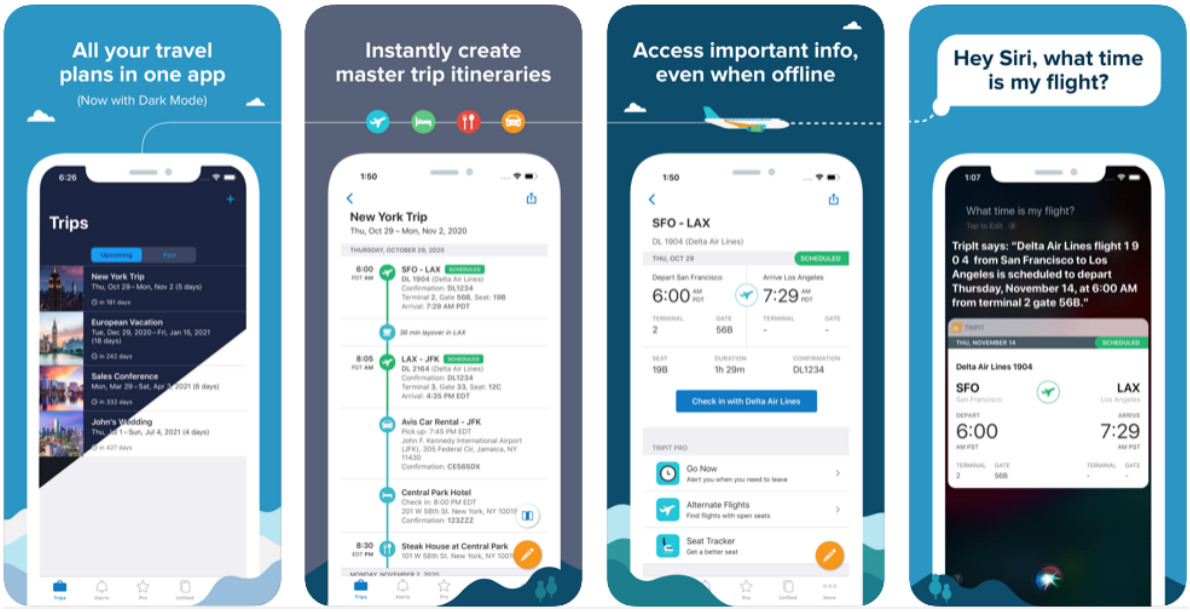


Figure 8. TripIt's General Functionalities

TripIt Pro, the premium version, includes additional features such as security wait times, neighbourhood safety scores, and travel stats that track the total distance travelled, number of

trips, and countries visited. It also offers personalised reminders for check-in times, gate departures, and even walking directions to gates and baggage claim areas. The app integrates with other services like Google Maps and supports offline access, ensuring travellers have access to their itineraries even without an internet connection.

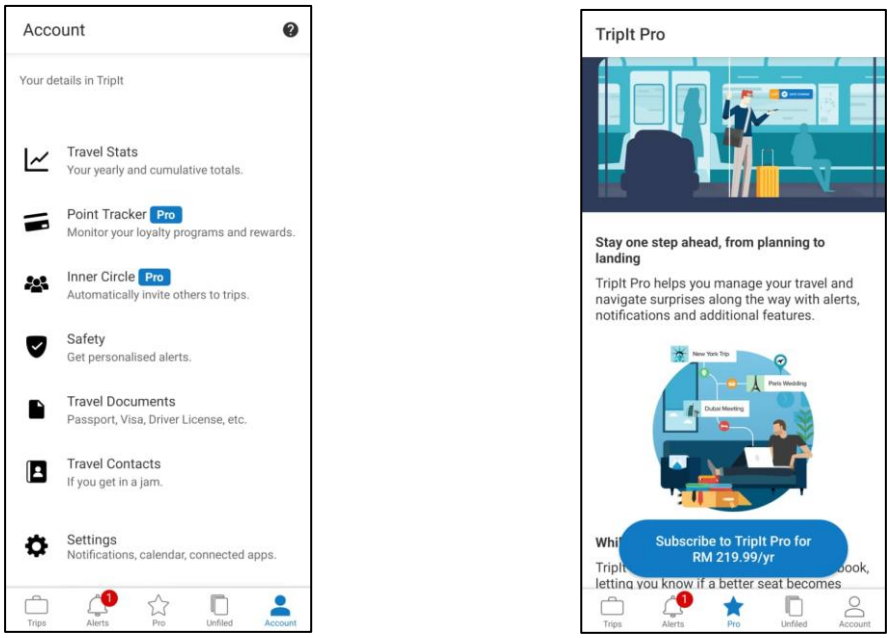


Figure 9. Pro features in TripIt

Overall, TripIt aims to provide a seamless travel planning experience by consolidating travel information, offering real-time updates, and providing useful tools and insights to enhance the journey.

2.3.3. Wanderlog

Wanderlog is a versatile travel planning and itinerary management application designed to streamline the travel experience (Harris, 2024). By connecting to your email account, Wanderlog automatically organises all your travel reservations, including flights, hotels, car rentals, and activities, into a comprehensive itinerary. This feature ensures that all your travel details are easily accessible in one place, reducing the stress of managing multiple bookings.

The app allows users to create detailed itineraries by adding activities, attractions, and places to visit, with personalised recommendations based on the user's location and preferences. Wanderlog integrates with Google Maps, providing a visual representation of your travel plans and optimising routes to save time and effort. The app also supports offline access, allowing travellers to download their trip plans and guides for offline use.

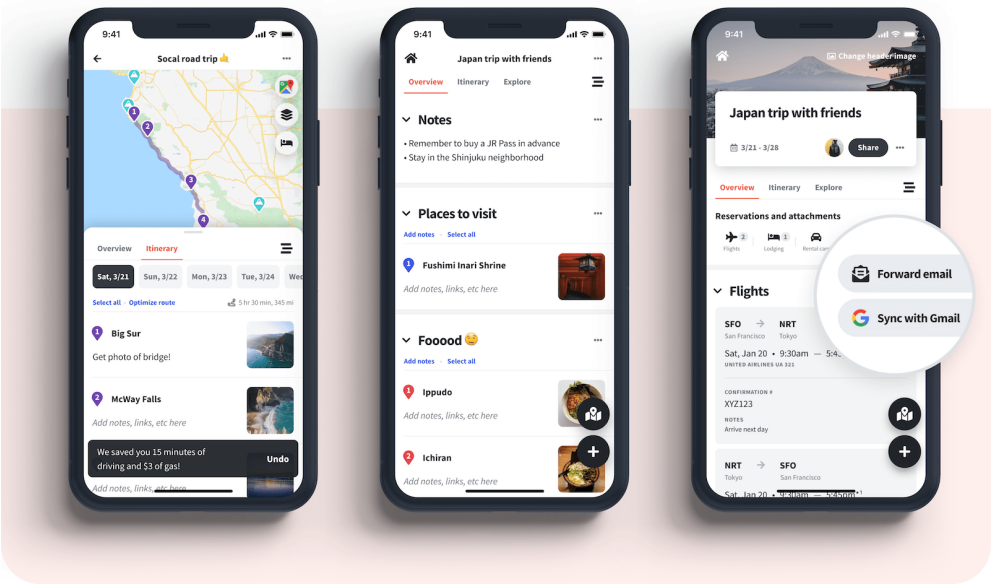


Figure 10. Wanderlog's General Itinerary/Trip Planner Function

Wanderlog offers a budgeting feature, enabling users to track their expenses and manage their travel budget effectively. Additionally, the app allows for collaboration, letting users share their itineraries with friends and family, making it easier to plan group trips. With its user-friendly interface and comprehensive features, Wanderlog aims to provide a seamless travel planning experience, ensuring that travellers are well-prepared and informed for their journeys (Wanderlog, n.d.).

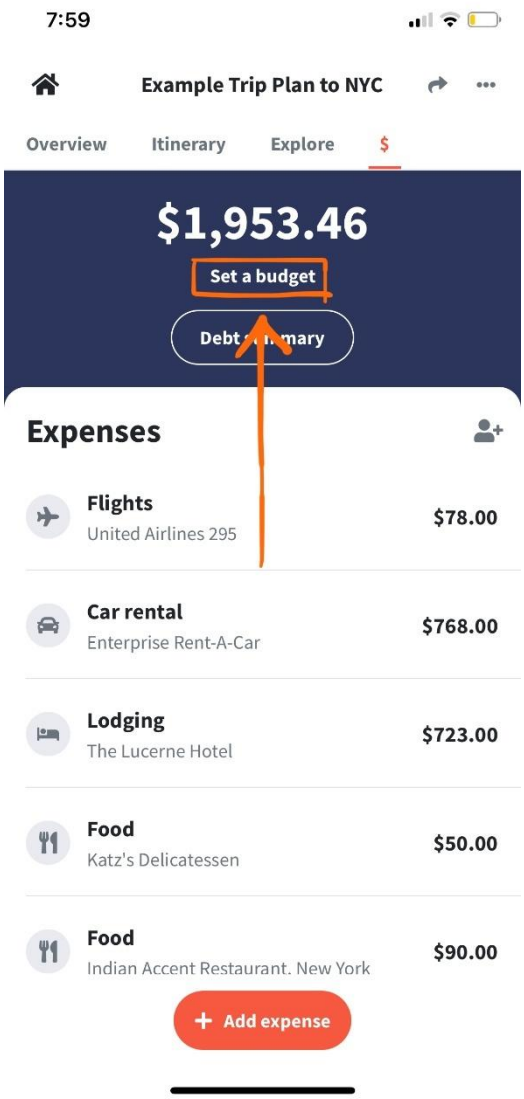


Figure 11. Wanderlog's Budgeting and Expense Features

2.3.4. Comparison between the Systems

From the analysis and explanation of all existing systems, this section will compare all the functionalities to evaluate the strengths, weaknesses, and opportunities to identify the best solution for the proposed application, WanderWise. The Table 1 below shows the comparison.

Table 1. Comparison of the existing systems and proposed application, WanderWise

Aspects	Existing Systems			
	Google Trips	TripIt	Wanderlog	WanderWise
Platform	Mobile app (discontinued, features now in Google Travel)	Mobile app and web-based platform	Mobile app and web-based platform	Mobile app
Travel Data Integration	Automated from Gmail	Email forwarding for travel confirmations	Manual entry or import from linked accounts	Manual entry or import from linked accounts
Itinerary Management	Customisable day plans with preloaded recommendations	Centralised itinerary with manual customisation	Collaborative itinerary with multi-user access	Dynamic itineraries with the ability to embed itineraries shared in the forum
Offline Functionality	Full offline access for all trip details	Limited offline access	Offline access to saved itineraries	Offline access for saved itineraries and AI travel chatbot
Community Features	None	None	Shared itineraries, user	Minimal community engagement

			comments, and tips	through itinerary sharing and liking
AI Assistance	None	None	AI-powered itinerary planner	AI-powered travel assistant for personalised recommendations and travel queries
Mapping and Navigation	Google Maps integration for navigation	Basic mapping for travel routes	Integrated maps for collaborative planning	None
Cost Tracking	None	None	Cost tracking for budget management	Expense tracker with currency conversion feature

Based on the comparison of existing systems which are Google Trips, TripIt, and Wanderlog, the proposed application WanderWise aims to integrate the strengths of these platforms while addressing their limitations to provide a superior travel companion experience. WanderWise stands out with its unique combination of features, including automated travel data integration from multiple email providers, customisable itineraries, and offline access to essential trip details. Unlike its predecessors, WanderWise incorporates an AI-powered travel assistant to offer personalised recommendations and resolve travel queries, making it highly adaptable to user needs.

While Google Trips and TripIt focus on itinerary management, they lack community engagement and robust collaborative tools. Wanderlog, on the other hand, emphasises user collaboration but can overwhelm users seeking minimal interaction. WanderWise balances this by offering minimal yet meaningful community features, such as itinerary sharing and the ability to like or download plans, catering to users who value simplicity and utility. Additionally, WanderWise introduces an expense tracker with currency conversion capabilities, addressing a significant gap in cost management across all analysed platforms.

The proposed solution leverages the advantages of modern technology, such as AI integration, enhanced offline functionality, and intuitive user interface design, to create a comprehensive travel tool that meets the evolving demands of modern travellers. By addressing the limitations of existing systems while introducing innovative features, WanderWise establishes itself as a reliable, user-focused solution tailored for efficiency, convenience, and accessibility.

2.4. Review of Tools and Technology

This section primarily reviews on tools and technologies which include the software and hardware used to develop the project, WanderWise.

2.4.1. Software

The section below discusses the software required to build the project.

2.4.1.1. Visual Studio Code (VSC)

Visual Studio Code (VSC) is a lightweight yet powerful source-code editor developed by Microsoft, designed to support various programming languages and development frameworks (Microsoft, n.d.). Its intuitive interface and extensive features, such as integrated debugging, intelligent code completion, and syntax highlighting, make it an ideal tool for software

development. VSC is particularly popular among developers for its vast marketplace of extensions, enabling customisation for specific needs, such as Flutter and Dart extensions for mobile app development. It also includes version control integration through Git and GitHub, which eases the process of keeping track of changes as well as branching out different versions (Microsoft, 2021). Additionally, VSC's cross-platform compatibility ensures accessibility across Windows, macOS, and Linux systems, while its performance efficiency allows it to run smoothly even on less powerful hardware. These features make Visual Studio Code a valuable tool in developing the WanderWise application, offering a robust and adaptable environment for coding and debugging.

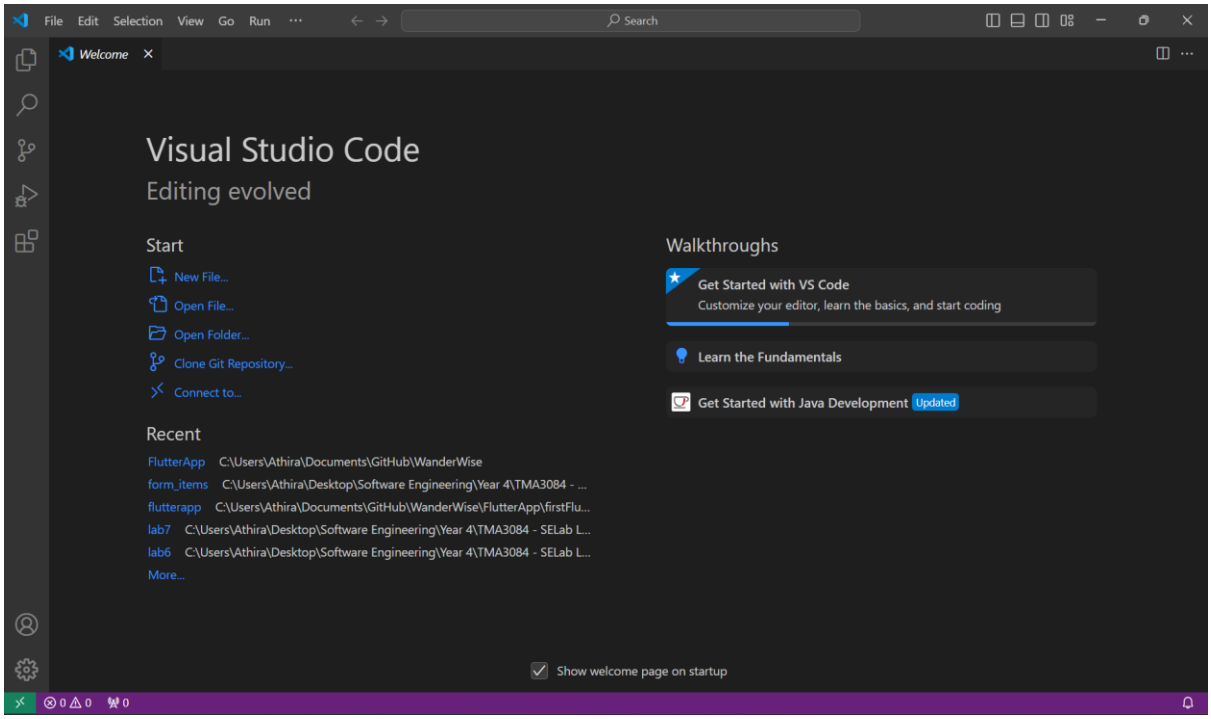


Figure 12. Visual Studio Code (VSC)'s Interface

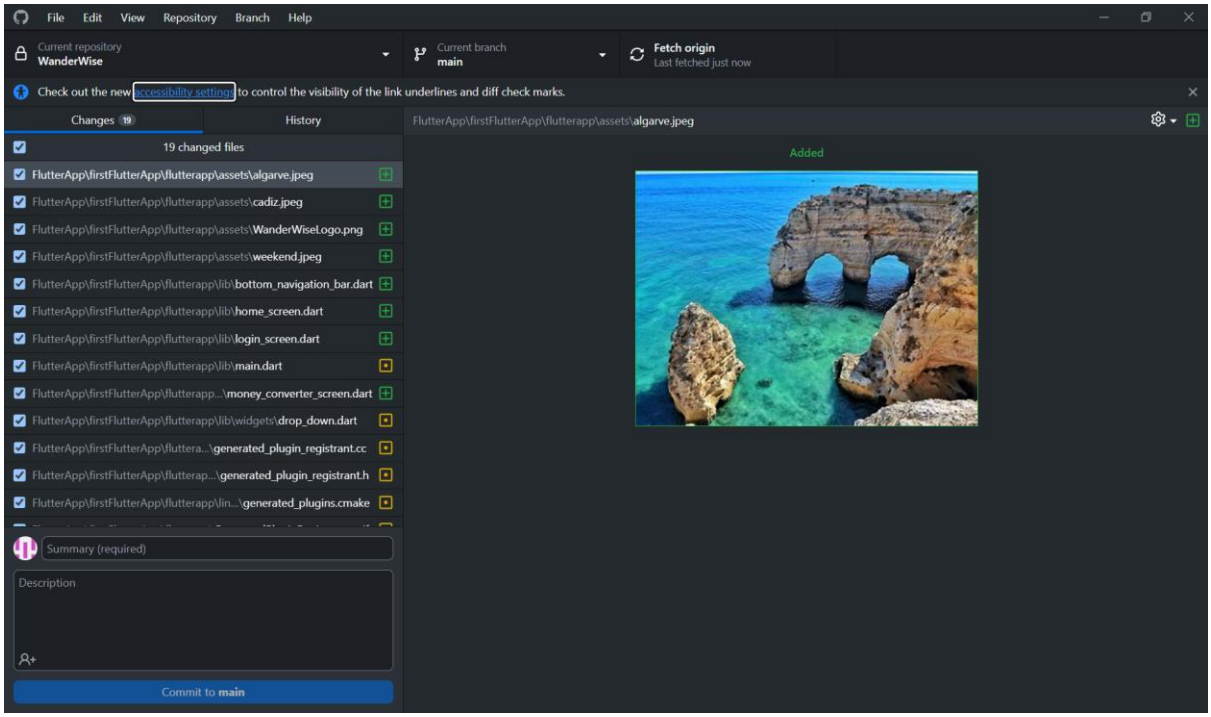


Figure 13. Integration of GitHub with Visual Studio Code (VSC).

2.4.1.2. Flutter Framework and Dart Language

Flutter is an open-source UI development framework developed by Google, designed for creating natively compiled applications across mobile, web, and desktop platforms using a single codebase (Google, 2021). It includes a comprehensive set of pre-designed widgets and tools that enable the development of responsive and visually consistent user interfaces. The framework's "hot reload" feature allows developers to apply code changes and view updates instantly without restarting the application, streamlining the development process. Flutter also supports platform-specific functionalities, ensuring high-performance applications across Android, iOS, and other platforms. Dart, on the other hand, is a programming language that complements Flutter, also developed by Google and optimised for building cross-platform applications. Dart supports both just-in-time (JIT) compilation for efficient development and ahead-of-time (AOT) compilation for optimised production builds. Its straightforward syntax and asynchronous programming capabilities make it suitable for managing complex

application workflows (Google, 2021). The combination of Flutter and Dart offers a robust and efficient framework for developing the WanderWise application, facilitating the creation of a cross-platform app with a consistent user experience and optimised performance.

Figure 14. Flutter Framework and Dart Language

2.4.2. Hardware

The section below discusses the hardware required to build the project.

2.4.2.1. Development Hardware

Below is the requirement for sufficient specifications to handle the app development and testing.

- Processor: Intel i5/i7 or AMD Ryzen 5/7 (or higher).
- RAM: Minimum 8GB (16GB recommended for multitasking).
- Storage: SSD with at least 256GB (recommended: 512GB+).
- Graphics: Integrated graphics are fine, but a dedicated GPU helps with rendering simulations.

2.4.2.2. Testing Hardware

Below are mobile devices that can be used to test the application to diversify the screen sizes, resolutions, and operating systems (OS).

- Android Devices: Test on different brands (e.g., Samsung, Xiaomi, or Pixel) and versions (e.g., Android 11–14).

2.5.Summary

This chapter provides an in-depth analysis of existing systems, tools, and technologies relevant to the development of WanderWise. It begins with a review of three comparable travel applications—Google Trips, TripIt, and Wanderlog—highlighting their features, strengths, and limitations. A comparative analysis identifies gaps in these systems, offering insights into how WanderWise can address them through unique functionalities like AI-driven assistance and a simplified social travel forum. Additionally, the chapter reviews key development tools such as Visual Studio Code and technologies like natural language processing NLP and machine learning (ML) technologies for implementing the AI chatbot. The chapter concludes with reviews for hardware and software that ensure efficient development, testing, and deployment of the application.

CHAPTER 3: ANALYSIS AND DESIGN

3.1. Introduction

For this chapter, the methodology used in the project development is further explained with breakdowns of each phase as well as clear explanations of the design phase. In this development, Condensed Agile is used as the methodology. On the other hand, several designs are used, such as user-interface design, and the overall development will be visualised through diagrams like UML, sequence diagrams, database designs, and many more.

3.2. Condensed Agile

The methodology used for this project development is Condensed Agile. Condensed Agile is an iterative and incremental software development methodology that emphasises flexibility, collaboration, and rapid delivery of functional software (Amajuoyi et al., 2024). There are six main phases in Condensed Agile: planning, designing, developing, testing, deploying, and reviewing. This chapter will focus more on the first phase of Condensed Agile, which is planning, and the second phase, which is designing.

3.2.1. Planning

The first phase of the Condensed Agile methodology is planning, where the project scope, limitations, and requirements are decided and planned out. This phase includes sprint planning, identifying tasks and objectives for the upcoming sprints, and organising them into a sprint backlog. Requirement analysis is conducted through reading and analysis of the current trends from surveys that are done internationally. Through understanding the general acceptance, bottlenecks, and challenges faced by travellers all over the world, a clear scope and solution can be made. Then a detailed project plan outlining key milestones, deliverables, and timelines, ensuring a focused and high-quality development process, is created. By carefully planning

and organising tasks, the project timeline and development stay on track and meet the desired quality standards.

3.2.1.1. Analysis on Survey

A survey was conducted among 11 respondents, which captured key demographic information, including age, gender, travel frequency, and travel reasons. The data provided valuable insights to gauge the respondents' perceptions towards travelling, travelling applications, and general issues faced. The first part of the survey assesses personal understanding of the respondents' travelling preferences and frequency of travel.

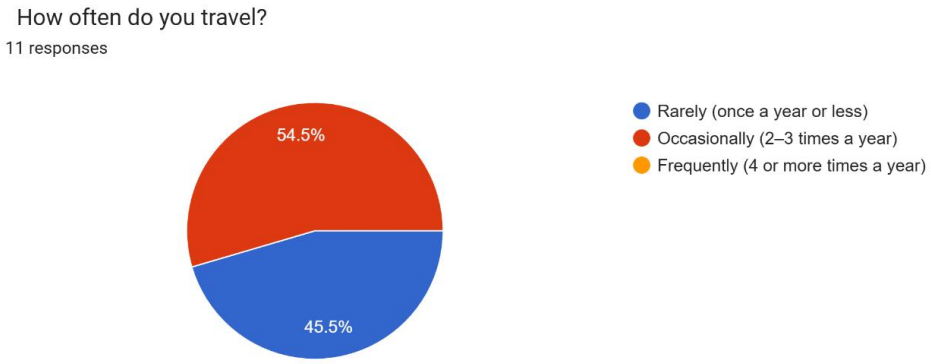


Figure 15. Pie Chart of Respondent's Travel Frequency

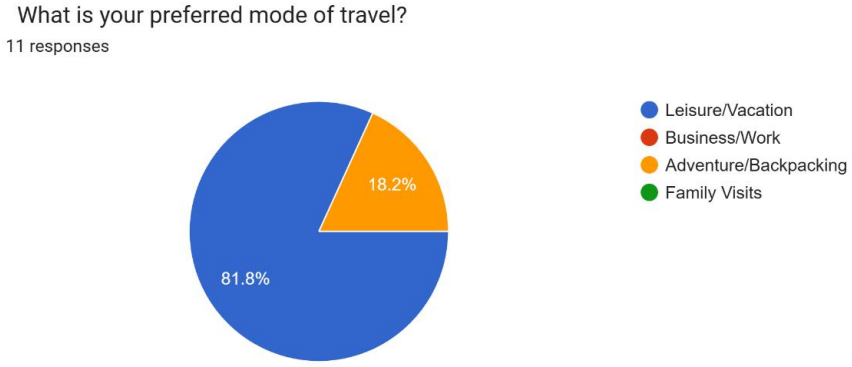


Figure 16. Pie Chart of Respondent's Preferred Mode of Travel

The survey results reveal insights into the travel frequency and preferences of respondents. As shown in Figure 15, 54.5% of respondents travel occasionally, about 2–3 times a year, while 45.5% rarely travel, indicating that most individuals travel infrequently, which may reflect constraints such as time, budget, or purpose. Figure 16 illustrates that the majority of respondents (81.8%) prefer travelling for leisure or vacation, emphasising the recreational purpose of their journeys, while 18.2% travel for business or work purposes. Other travel preferences, such as adventure, backpacking, or family visits, were not selected, indicating a strong focus on leisure-related travel among the surveyed group. These findings highlight that the application should prioritise features that cater to leisure travellers while also offering functionalities that appeal to occasional and infrequent travellers.

The next part of the survey shows the challenges faced the most by respondents during travelling as well as their preferred content to see in the travel forum.

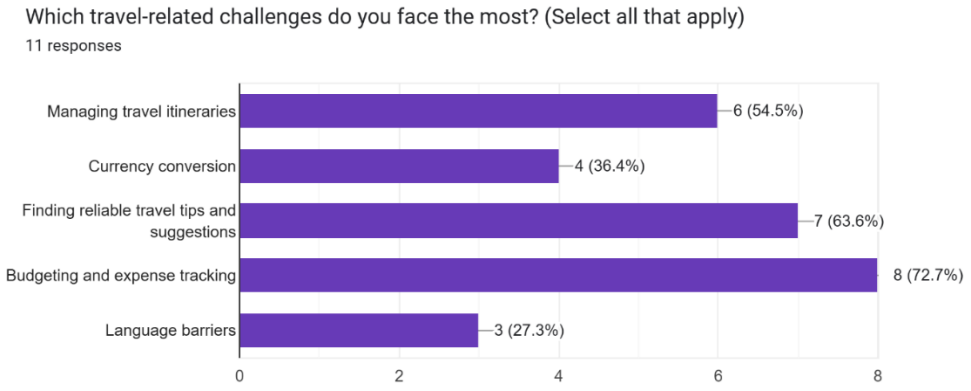


Figure 17. Bar Chart of Respondents' Travel-Related Challenges

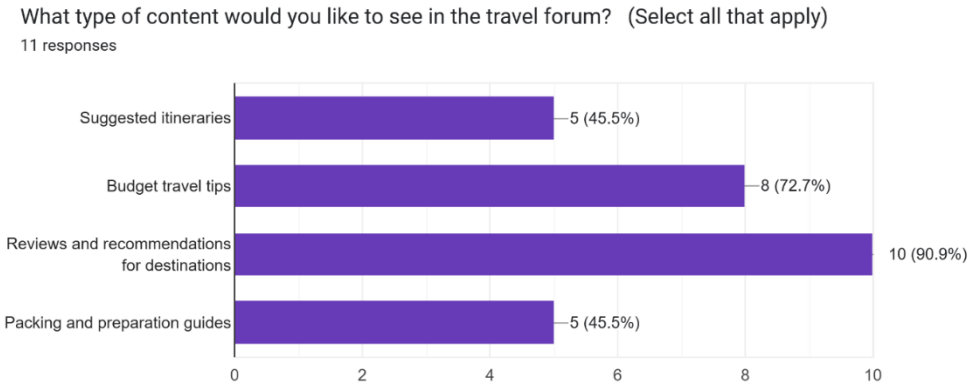


Figure 18. Bar Chart of Respondents' Preferred Content in Travel Forum

The survey results provide insights into the travel challenges faced by respondents and their preferences for content in a travel forum. As shown in the first chart, the most significant challenge is budgeting and expense tracking, affecting 72.7% of respondents, followed by finding reliable travel tips and suggestions (63.6%) and managing travel itineraries (54.5%). Currency conversion and language barriers were less common issues, with 36.4% and 27.3% of respondents highlighting them, respectively. Regarding preferred content for a travel forum, the second chart reveals a strong interest in reviews and recommendations for destinations, with 90.9% of respondents selecting this option. Budget travel tips are also highly sought after by 72.7% of respondents, while suggested itineraries and packing/preparation guides each garnered interest from 45.5%. These findings indicate a need for the application to prioritise tools and resources that assist with budgeting, travel tips, and itinerary planning while offering rich content on destination recommendations and budget-friendly advice.

The last section of the survey consists of the respondents' suggestions and concerns for WanderWise.

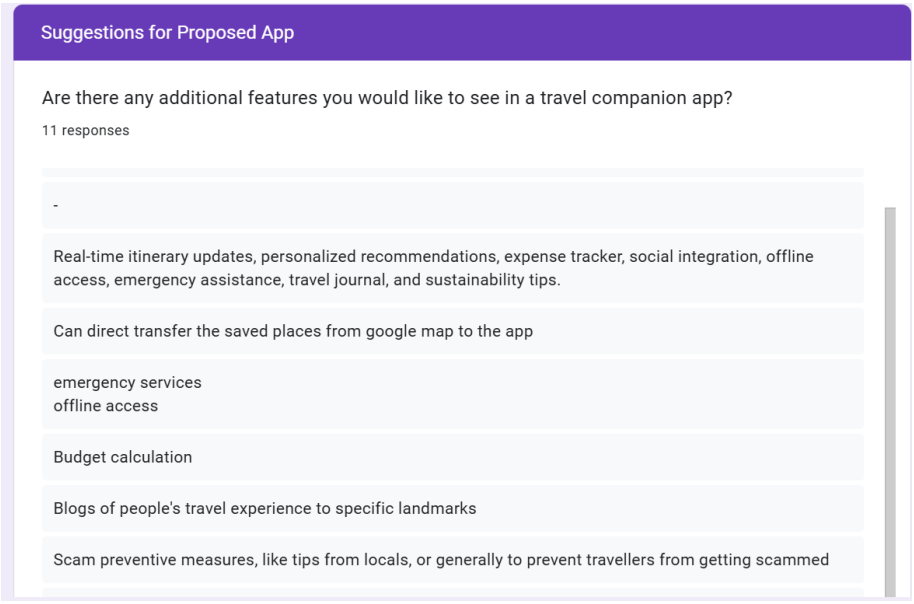


Figure 19. Open-Ended Question for Suggestions for WanderWise

The survey responses highlight several valuable suggestions for additional features in a travel companion app. Respondents emphasised the need for real-time itinerary updates, personalised recommendations, an expense tracker, and offline access. Other notable suggestions include emergency services, budget calculation tools, direct integration with saved places from Google Maps, and features like a travel journal and sustainability tips. Additionally, there is interest in blogs sharing travel experiences at specific landmarks and scam prevention tips from locals. These suggestions underscore the importance of integrating practical, safety-focused, and user-friendly features to enhance the overall travel experience.

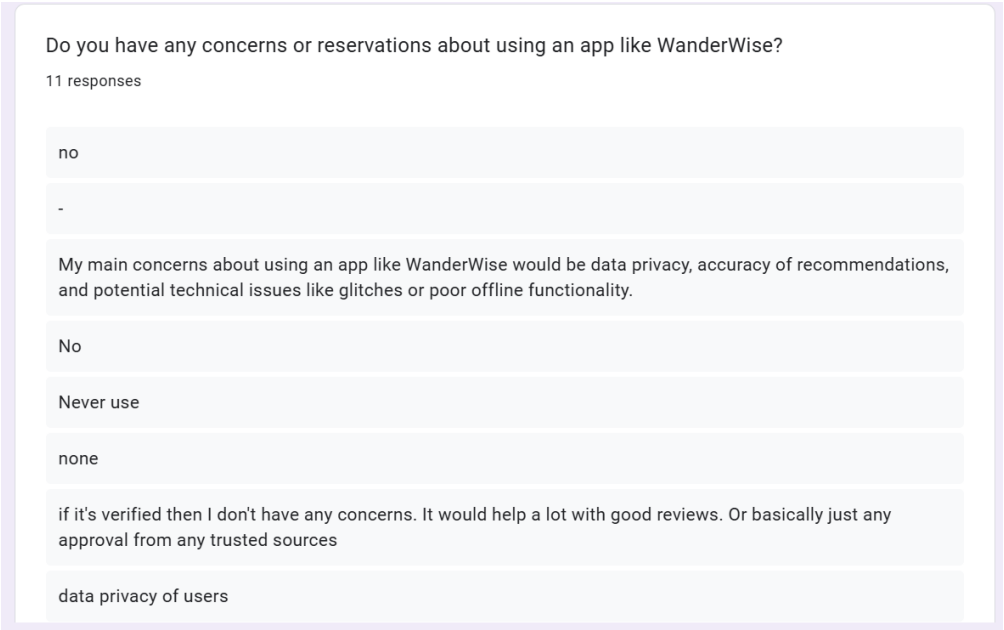


Figure 20. Open-Ended Question for Concerns for WanderWise

The survey responses indicate that while some users have no concerns about using an app like WanderWise, others highlighted potential issues. Key concerns include data privacy, the accuracy of recommendations, and potential technical problems, such as glitches or poor offline functionality. Some respondents emphasised the importance of verified reviews and approval from trusted sources to build confidence in the app. These insights highlight the need for robust data protection measures, reliable performance, and verified recommendations to address user reservations and foster trust in the application.

3.2.2. Design

The design phase in the Condensed Agile methodology is a pivotal stage where the key features of the proposed application are carefully developed to meet the requirements established during the planning phase. This iterative process allows for continuous refinement and improvement, minimising risks and addressing potential challenges before the development stage. Guided by insights from the analysis of existing systems discussed in Chapter 2, the design process employs Unified Modelling Language (UML) to ensure clarity

and structure. Various UML diagrams, including use case diagrams, activity diagrams, sequence diagrams, and class diagrams, are utilised to comprehensively represent the application's functionality, workflows, and architecture. Each diagram serves a specific purpose in detailing different system components, providing a robust and well-structured foundation for development. This approach, rooted in iterative refinement and user-centric considerations, ensures the design aligns seamlessly with project objectives while addressing user needs effectively.

3.2.2.1. Design and Prototyping (UML Diagrams)

This section introduces all the UML diagrams, ranging from use case diagrams to use case specifications to activity diagrams, sequence diagrams, and class diagrams. It aims to give a clearer picture of WanderWise's architecture; at the same time, all of the details needed will be discussed for each subsection.

3.2.2.1.1. Use Case Diagram

The use case diagram below shows roughly what the interactions between the user and the system are, while focusing on core functionalities and their relationships. Users can sign up, log in (with an option to retrieve their password), view the homepage, and explore features like viewing or posting guides, tracking expenses, and accessing live currency exchange rates. Additional functionalities include messaging, AI-powered chatbot recommendations, trip planning with options to match with travel buddies, and viewing profiles, saved guides, or trips. The system supports these actions by providing live currency rates and showing similar profiles for personalised interactions. The diagram effectively uses "extends" to indicate optional functionalities, such as retrieving a password or exploring specific travel content, and "includes" for shared actions like trip planning. This design ensures a user-centric application with flexible and comprehensive features for seamless travel planning and management.

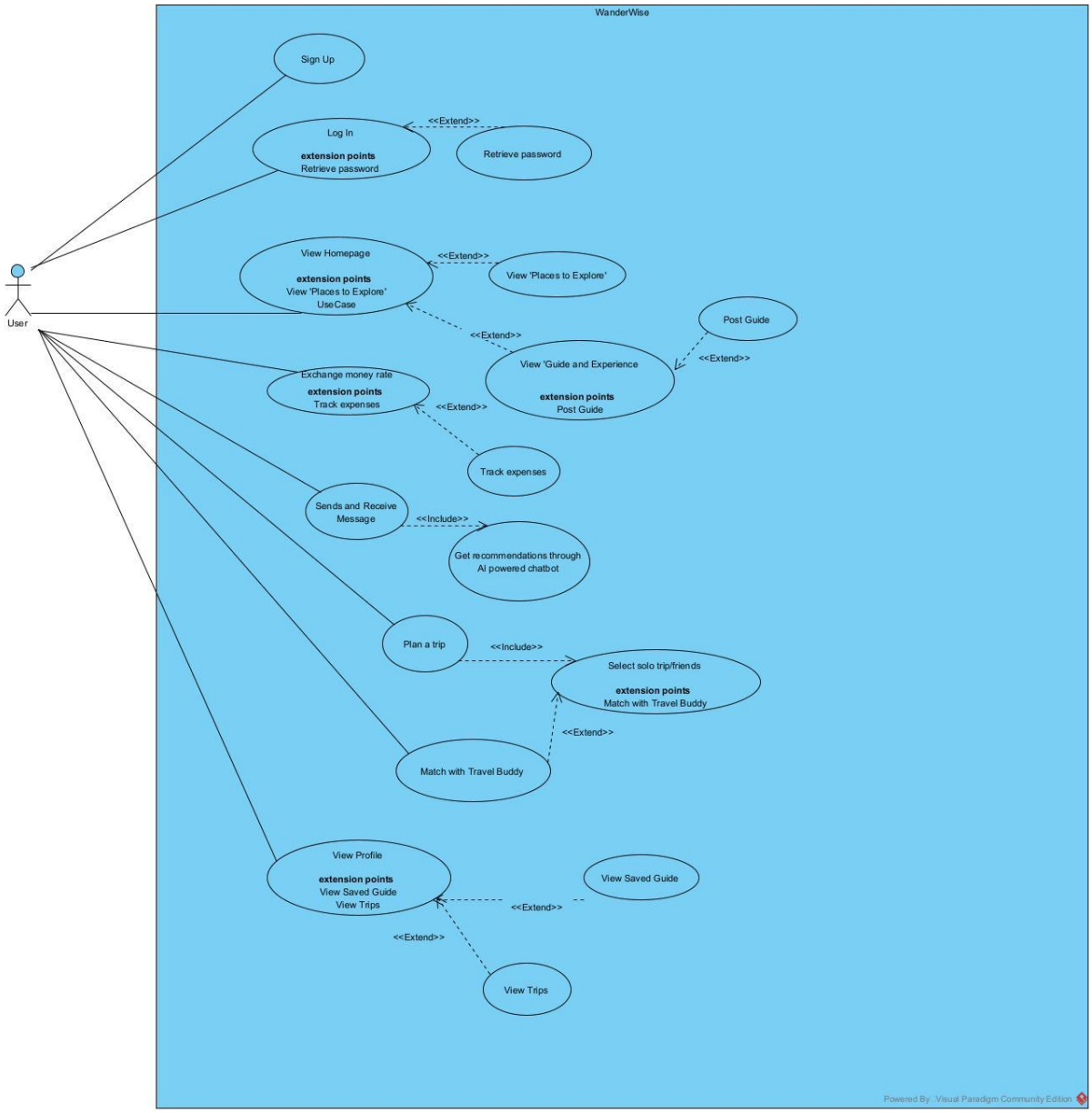


Figure 21. Use Case Diagram for WanderWise

3.2.2.1.2. Sequence Diagrams

This section focuses on sequence diagrams, which provide a visual representation of the step-by-step flow of interactions between users, the system, and other components involved in specific activities. Each diagram highlights the roles of various entities and their interactions, ensuring a clear picture of the system's functionality.

Create New Post in Guide Forum

Figure 22 and Table 2 depicts the activity of posting a guide in the forum.

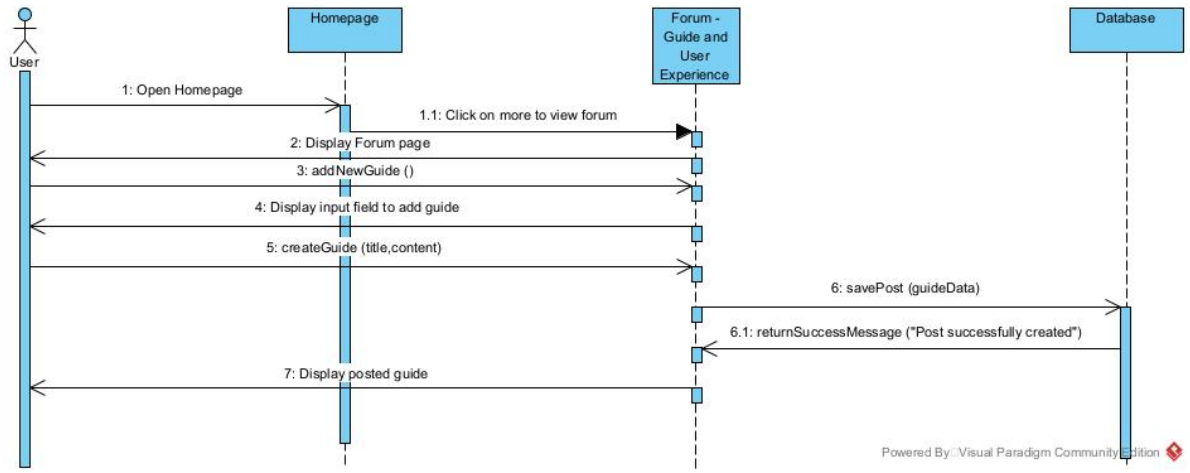


Figure 22. Sequence Diagram for Creating New Post

Table 2. Use Case Specification for Post a Guide

Use Case	Post a Guide
Actor	User
Description	This use case allows the user to create and post a guide in the 'Guide and User Experiences' part of the forum.
Precondition	• The user is logged into the application. • The user has access to the "Forum" feature via the homepage. • The application is connected to the internet.
Post-Condition	• The guide is successfully created and saved in the database. • The new guide is displayed in the forum for other users to view.
Scenario	1. The user navigates to the homepage.

	2. The user clicks on the "More" button to access the forum section. 3. The forum page is displayed. 4. The user clicks on the "+" button to add a new guide. 5. An input field is displayed for the user to enter guide content. 6. The user enters the guide details and clicks "Post." 7. The system saves the guide to the database and returns a successful post dialogue box. 8. The newly posted guide is displayed in the forum for all users to view.
Alternative Flow	None
Exception Flow	If the user fails to enter any content in the input field and clicks "Post," the system will prompt an error message indicating that content is required.

Download Itinerary from Guide

Figure 23 and Table 3 depict the activity of downloading itineraries from guides.

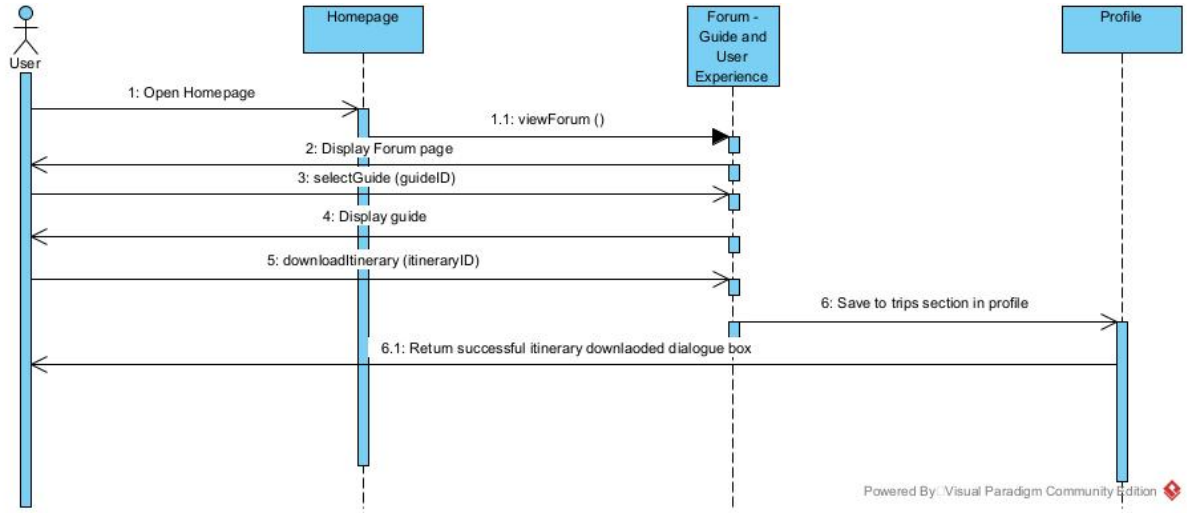


Figure 23. Sequence Diagram for Downloading Itinerary

Table 3. Use Case Specification for Downloading Itinerary

Use Case	View and Download Guide Itinerary
Actor	User
Description	This use case allows the user to view a guide shared by other users in the forum and download the associated itinerary for personal use.
Pre-Condition	• The user is logged into the application. • The user has access to the "Forum" feature via the homepage. • The application is connected to the internet.
Post-Condition	• The selected itinerary is successfully downloaded and saved in the user's profile under the trips section.

	• The system confirms the successful download with a notification.
Scenario	1. The user navigates to the homepage. 2. The user clicks on the "More" button to access the forum section. 3. The forum page is displayed. 4. The user selects a guide shared by another user. 5. The system displays the selected guide along with its details. 6. The user clicks the "Download Itinerary" button. 7. The system saves the itinerary to the "Trips" section in the user's profile. 8. The system displays a dialogue box confirming the successful download.
Alternative Flow	None
Exception Flow	If the user tries to download an itinerary without viewing a guide first, the system will prompt an error message requiring the user to select a guide.

Edit Profile

Figure 24 and Table 4 depicts the activity of editing users' profile.

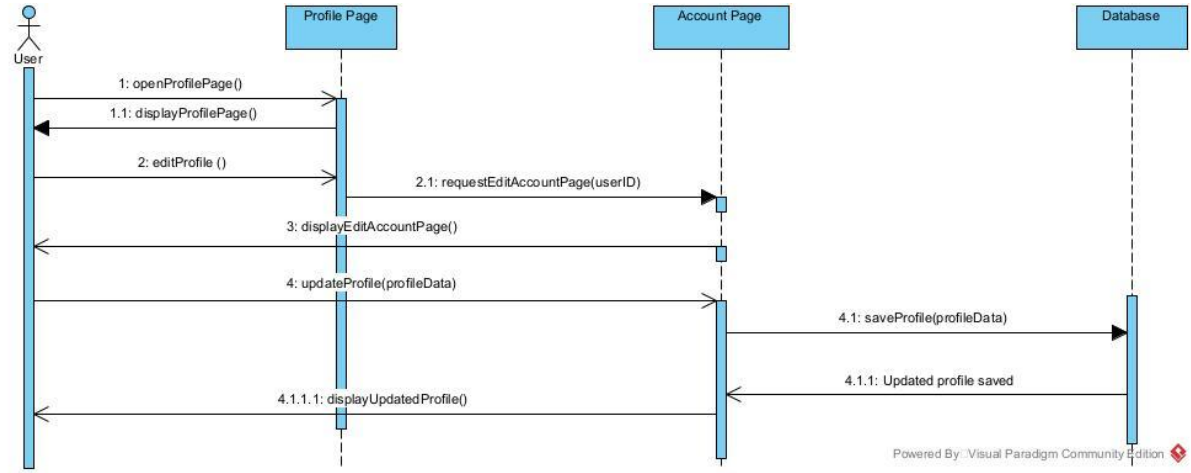


Figure 24. Sequence Diagram for editing on Account Page

Table 4. Use Case Specification for Edit Profile Information

Use Case	Edit Profile Information
Actor	User
Description	This use case allows the user to access their profile page, make changes to their profile information, and save the updates.
Pre-Condition	• The user is logged into the application. • The user has access to their profile page. • The application is connected to the internet.
Post-Condition	• The updated profile information is successfully saved in the database.

	• The profile page displays the updated information.
Scenario	1. The user navigates to the profile page from the application menu. 2. The system displays the user's profile page. 3. The user clicks on the "Edit Profile" button. 4. The system displays the "Edit Account" page. 5. The user updates the desired profile information and submits the changes. 6. The system saves the updated information in the database. 7. The updated profile information is displayed on the profile page.
Alternative Flow	Alternative Step 1: • The user clicks on the displayed contact card on homepage. • The user access profile page from a ‘profile’ button on the contact card.
Exception Flow	If the user leaves required fields blank when updating their profile, the system will prompt an error message requesting the completion of those fields.

Planning A Trip

Figure 25 and Table 5 depicts the activity of planning a trip.

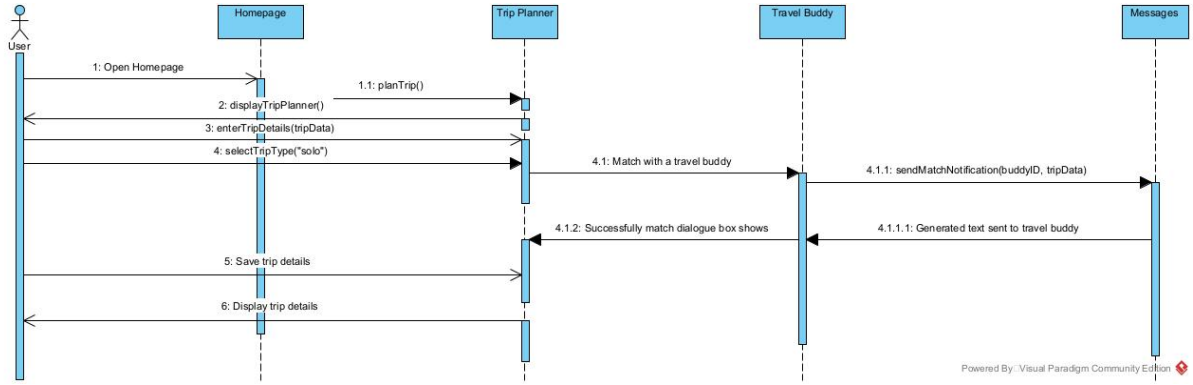


Figure 25. Sequence Diagram for Planning a Trip

Table 5. Use Case Specification for Planning A Trip

Use Case	Plan a Trip and Match with a Travel Buddy
Actor	User
Description	This use case allows the user to plan a trip, save trip details, and optionally match with a travel buddy for the trip.
Pre-Condition	• The user is logged into the application. • The "Trip Planner" feature is accessible from the homepage. • The application is connected to the internet.
Post-Condition	• The trip details are successfully saved in the system. • If the user opts to match with a travel buddy, the match is completed, and a generated text is sent to the travel buddy.

Scenario	1. The user navigates to the homepage. 2. The system displays the homepage, and the user clicks on the "Plan a Trip" option. 3. The system displays the Trip Planner page. 4. The user inputs trip details and selects the "Solo Trip" option. 5. The system saves the trip details. 6. The system displays the saved trip details to the user. 7. If the user selects the option to match with a travel buddy: • The system matches the user with a compatible travel buddy. • A dialogue box confirms the successful match. • A generated text is automatically sent to the matched travel buddy via the messaging feature.
Alternative Flow	Alternative Step 7: • The user clicks on ‘Travel Buddy’ option in the homepage.
Exception Flow	If the user does not input any trip details, the system will display a prompt asking for mandatory trip information before proceeding.

Exchanging Currency Rate and Tracking Expenses

Figure 24 and Table 4 depicts the activity of exchanging currency rate and tracking expenses.

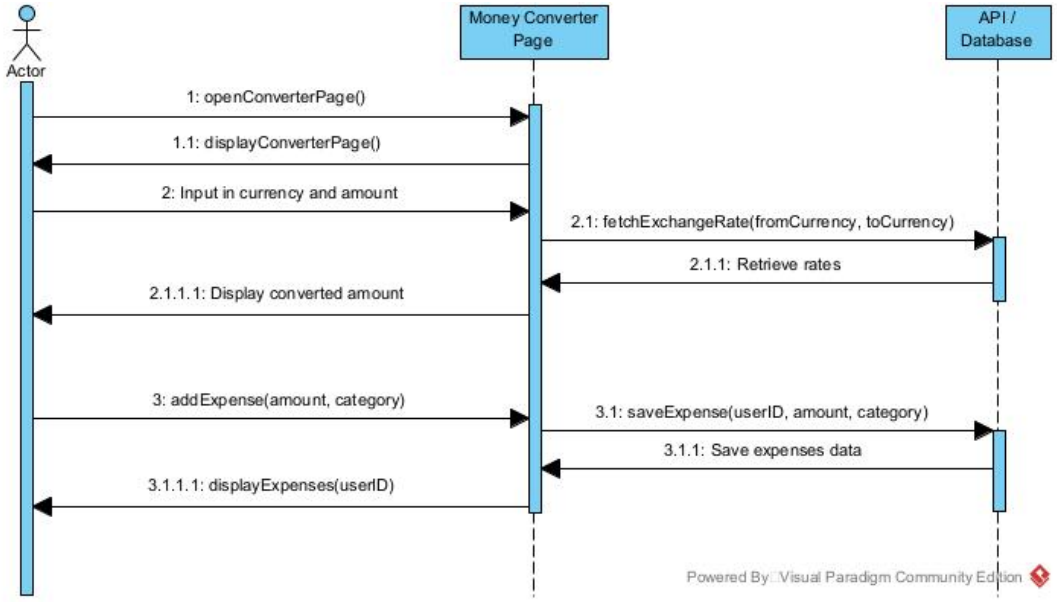


Figure 26. Sequence Diagram for Exchanging Currency Rate and Tracking Expenses

Table 6. Use Case Specification for Exchanging Currency Rate and Tracking Expenses

Use Case	Exchanging Currency Rate and Tracking Expenses
Actor	User
Description	This use case allows the user to convert currency and track expenses, with data stored in the system for future reference.
Pre-Condition	• The user is logged into the application. • The "Money Converter" feature is accessible from the homepage.

	• The application is connected to the internet.
Post-Condition	• The converted currency amount is displayed to the user. • Expense details are successfully saved in the system
Scenario	1. The user navigates to the Money Converter Page. 2. The system displays the Money Converter Page. 3. The user inputs the currency and amount to convert. 4. The system sends a request to retrieve the current exchange rates. 5. The exchange rate is retrieved from the API/Database. 6. The system calculates and displays the converted amount to the user. 7. The user inputs expense details. 8. The system sends the expense data to the database. 9. The system saves the expense data and displays the saved expenses to the user.
Alternative Flow	None
Exception Flow	• If the user does not input any currency or amount, the system will display a prompt asking for mandatory input. • If the API fails to retrieve the exchange rate, the system displays an error message and suggests trying again later.

3.2.2.1.2. Activity Diagram

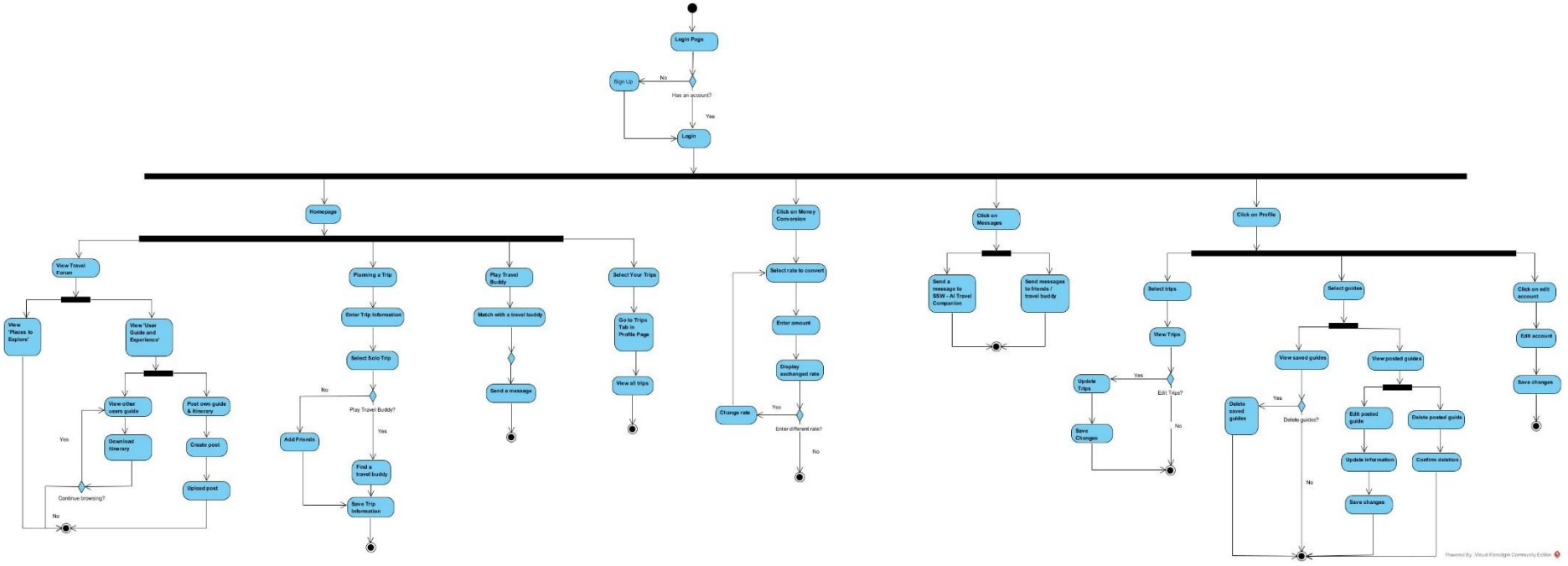


Figure 27. Activity Diagram for WanderWise

The activity diagram gives a detailed visualisation of the workflow in the WanderWise application, showing how users interact with the system to perform various tasks. It begins with the user accessing the application, which serves as the entry point to the system. From there, users can choose to engage in a range of activities, such as exploring places and guides, contributing their own guides, or planning trips. The trip planning process includes options to save trip details, choose solo trips, or match with a travel buddy, highlighting the flexibility and customisation offered to the user. Additional functionalities, such as expense management and currency conversion, are integrated to support practical travel needs, while a messaging feature allows users to communicate directly, particularly when matched with travel buddies. The diagram also incorporates decision points that guide users through different paths based on their inputs, ensuring the workflow adapts to individual preferences. Each activity leads to a logical endpoint, such as successfully saving trip details, post a guide, or sending a message, emphasising a seamless and user-centered experienc

3.2.2.1.4. Class Diagram

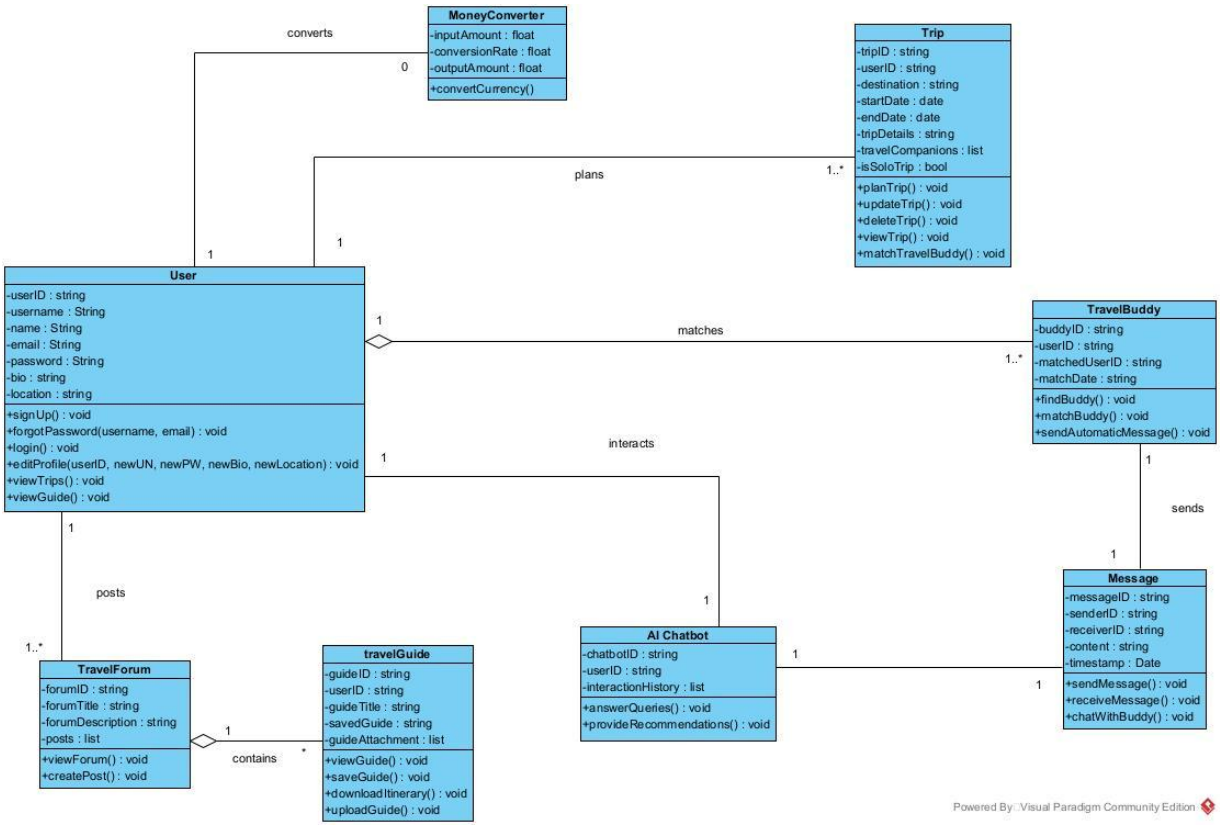


Figure 28. Class Diagram for WanderWise

The class diagram outlines the structural design of the WanderWise application, depicting the main components, their attributes, and their relationships. At its core is the User class, which represents the application's users and includes attributes such as userID, username, and email. This class enables fundamental user actions, including signing up, logging in, editing profiles, and managing trips or guides. Closely related to this is the Trip class, which manages trip planning by storing details like destination, dates, and travel companions, while providing methods to plan, update, or delete trips. For users seeking travel partners, the TravelBuddy class facilitates matching and communication with compatible travel buddies. This TravelBuddy class uses an aggregation relationship which aids for functionalities between the class with User.

Other supporting classes include the TravelForum, which handles forum interactions such as creating and viewing posts, and the TravelGuide, which allows users to upload, save, or access guides. The TravelGuide class uses an aggregation relationship with the User class, meaning that the TravelGuide can exist independently of the User but is associated with the User for functionalities like uploading and saving guides. The Message class manages user communication, storing messages and timestamps while enabling sending and receiving messages. To enhance the user experience, the AI Chatbot class provides features like answering queries and offering travel recommendations, while the MoneyConverter class simplifies currency conversion by managing rates and calculations. The diagram also defines the relationships between these classes, showing how they interact to ensure seamless functionality. Together, these components form a well-structured and interconnected system designed to meet user needs effectively.

3.2.2.2. UI Design using Wireframe

This section shows the wireframe of the proposed application made using Figma. The created mock-up aims to visualise the user interface of WanderWise.

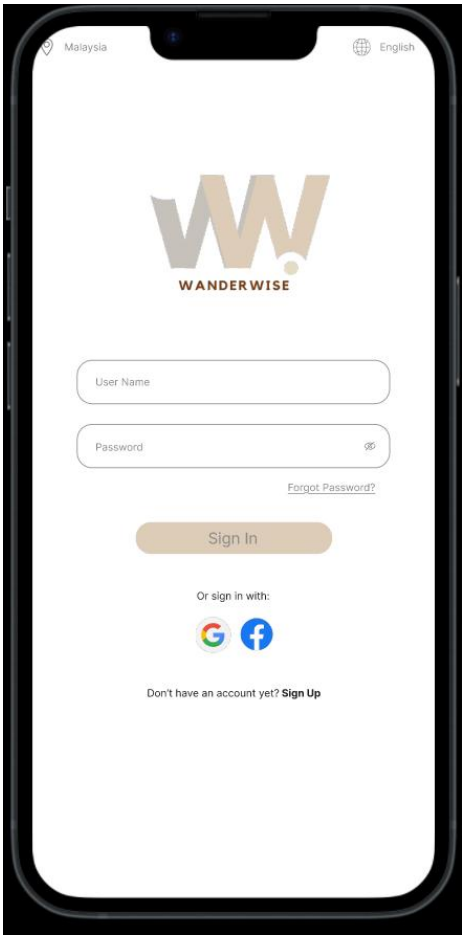


Figure 29. Login/SignUp Page for WanderWise

Figure 29 shows the login/sign up page of WanderWise. Upon launching the application, users will automatically be prompted to this page. Then, they can login into the application if they have an account or sign in through a Google account or Facebook account. If the users are not registered, they can sign up for a new account.

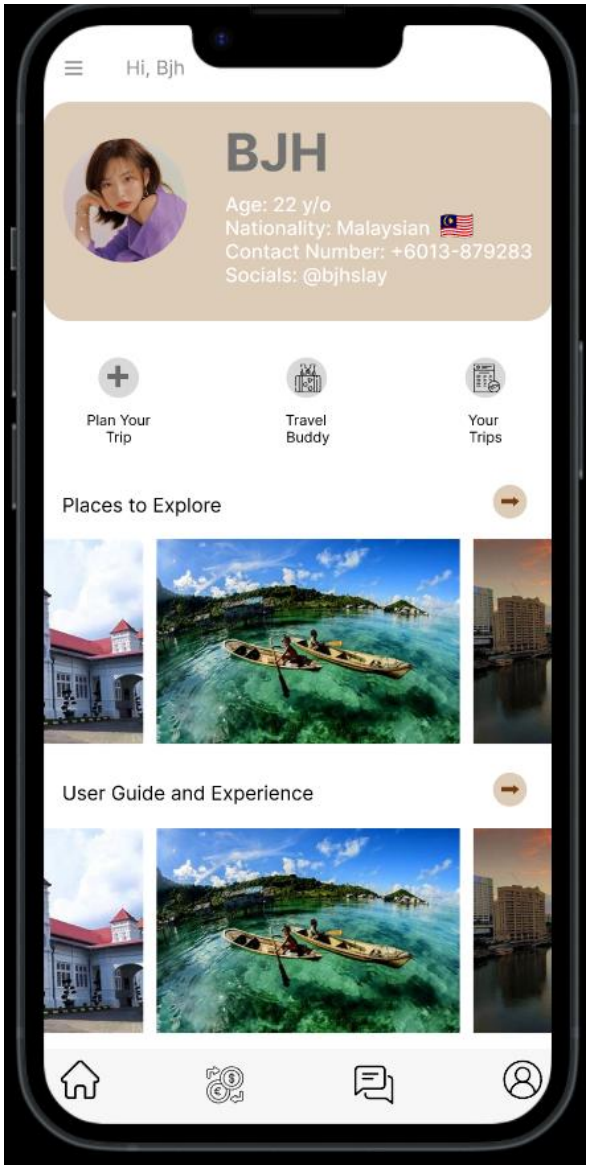


Figure 30. Homepage for WanderWise

The figure above shows the homepage of WanderWise. At the top of the page, users' brief contact cards will be displayed to help the travellers when travelling. Apart from that, from the homepage, there are quick shortcut actions such as planning a trip, viewing all trips, and accessing Travel Buddy. There is also a travel forum where users can view places to explore and user guides and experiences.

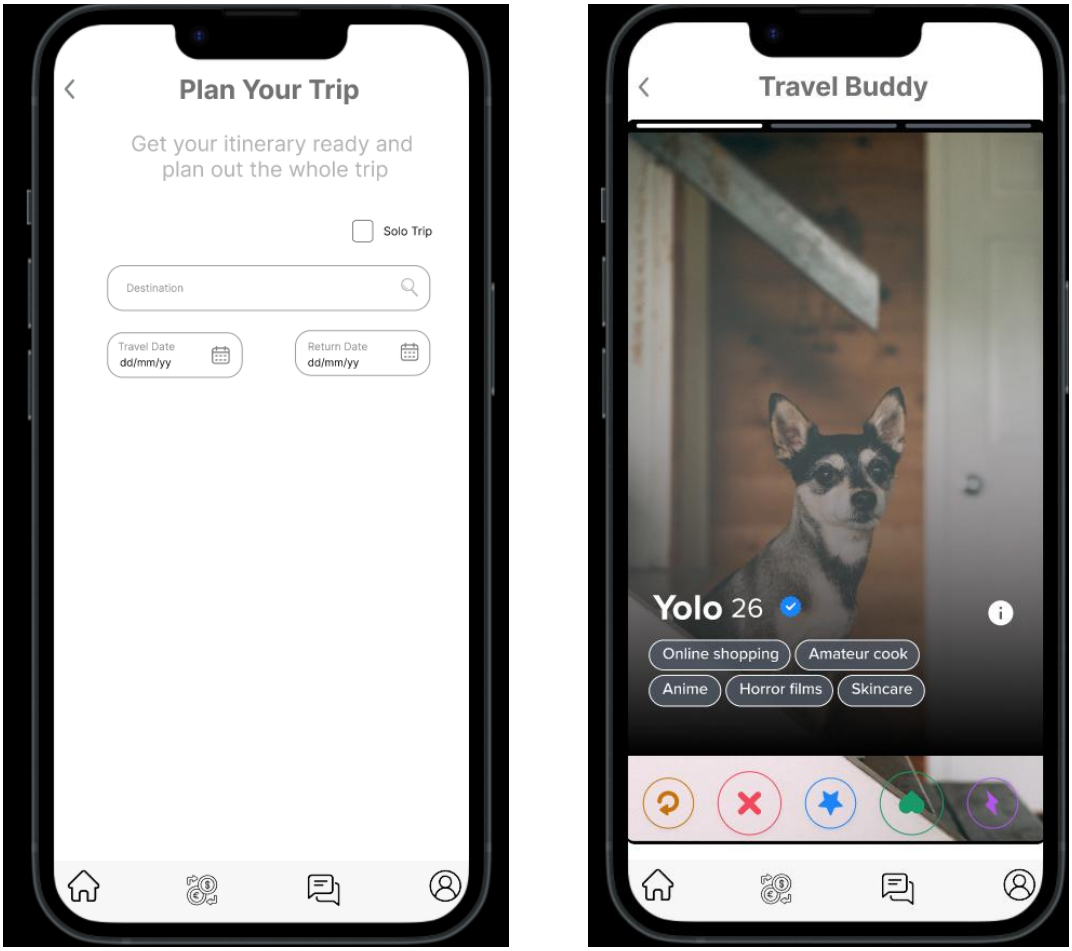


Figure 31. Plan Your Trip and Travel Buddy WanderWise Mockup

Figure 31 shows the plan a trip feature where they can input details of their travelling plans. There is a checkbox to click on for solo trips, where users will then be prompted to go to Travel Buddy. This travel buddy concept aims to let the users find travel buddies around the world that share similar interests.

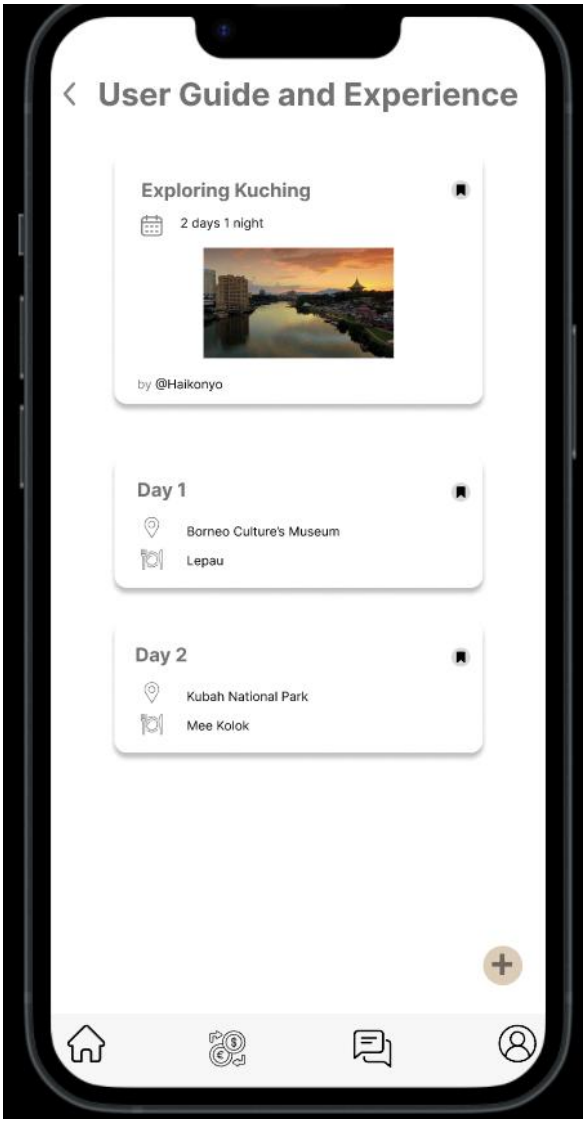


Figure 32. Guide in Travel Forum WanderWise Mockup

The figure above shows one of the guides in the travel forum. Users will then be able to either download the guide as a whole or download a specific itinerary to be embedded into their own trips. In the guide, users can also post a guide of their own.

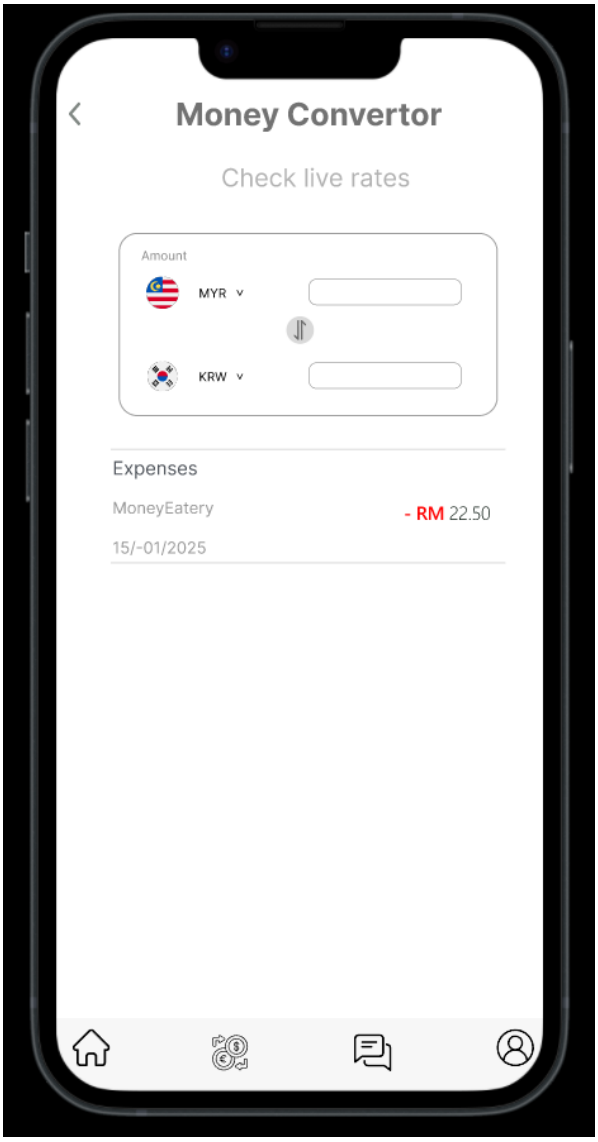


Figure 33. Money Converter for WanderWise Mockup

Figure 33 shows the money converter feature, which also tracks expenses for users. This money converter will use exchange rates with the latest currency rate to aid users, especially when doing transactions. After that, the users can input their expenses manually to keep track of their budget.

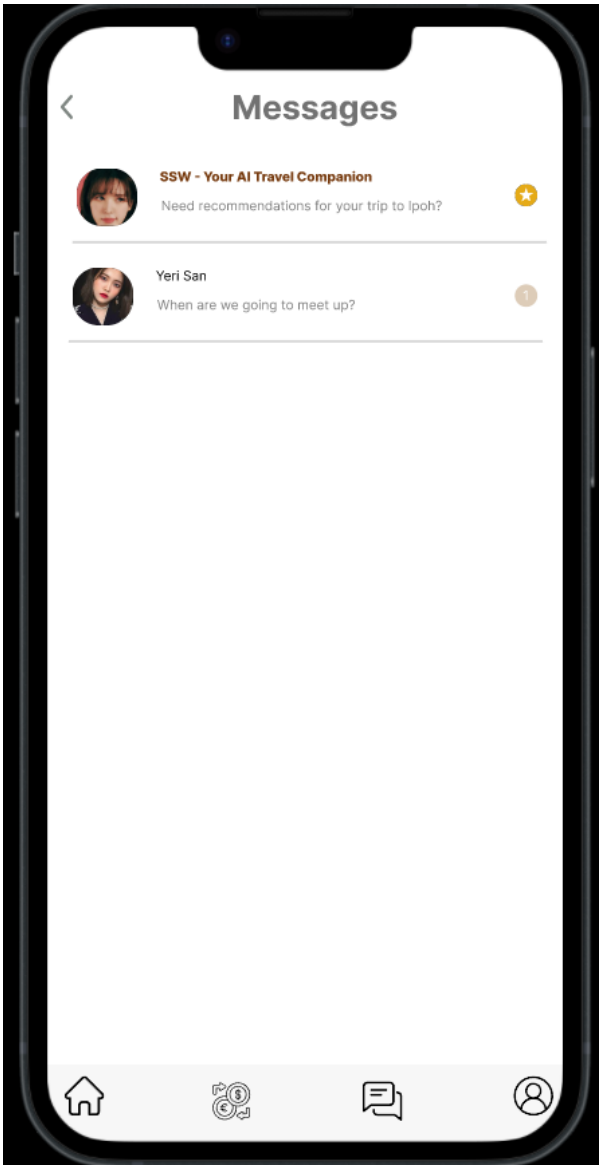


Figure 34. Messages for WanderWise Mockup

The message platform allows users to chat with their travel buddy as well as message the AI-powered travel chatbot. At the top of the list, users can directly have a conversation with the AI-powered chatbot. This message will always be pinned at the top, and the chatbot will offer personalised recommendations, including suggestions for places to explore or activities for a specific destination. Below the pinned chatbot come messages from travel buddies that are successfully matched. This allows seamless interaction between all users.

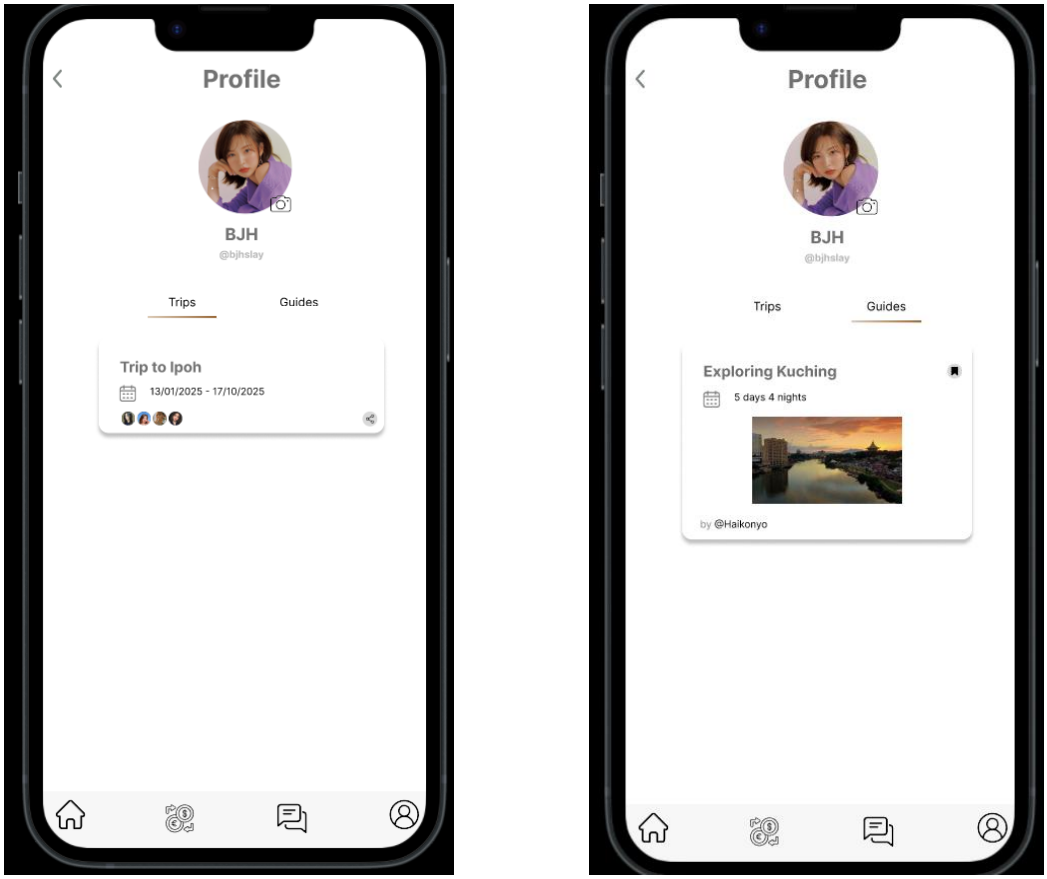


Figure 35. Profile Page for WanderWise Mockup

Figure 35 shows the profile section where users can manage and view their account information. They are able to update their profile details, including but not limited to updating their picture, bio, and more. Apart from that, in the profile page, there are 2 tabs for the users to navigate through the page, which are Trips and Guides. The Trips tab allows the users to view their planned and past trips. Each trip is displayed in card views, which show essential information for easy and convenient viewing. Next, the Guides tab shows either the posted guides or the saved guides. This allows users to gain easy access to their previously bookmarked or personalised travel content.

3.2.3. Develop

The development phase in the Condensed Agile methodology is where the proposed application transitions from design to implementation, with the planned features and system

architecture being translated into functional components. This phase builds upon the foundation established in the design phase, ensuring a structured and iterative approach to coding and implementation. The development process follows the software, hardware, and testing requirements discussed in Chapter 2, ensuring alignment with the predefined specifications. Unified Modelling Language (UML) diagrams, such as sequence and class diagrams, serve as critical references during this phase to guide the implementation of core functionalities. This approach ensures that the application evolves in alignment with project objectives and user needs while adhering to the iterative and adaptable principles of Condensed Agile.

3.2.4. Test

The testing phase in the Condensed Agile methodology is a critical step in ensuring the quality, functionality, and reliability of the proposed application. This phase rigorously evaluates the application to identify and resolve any potential issues. Various testing techniques, including unit testing, integration testing, and system testing, are employed to verify that individual components, as well as the application, function as intended. The iterative nature of Condensed Agile allows for continuous testing and refinement throughout development, ensuring that defects are identified and addressed promptly. Moreover, a test plan is written and updated for every release (Hamilton, 2024).

3.2.5. Deploy

The deployment phase in the Condensed Agile methodology focuses on releasing the application to the intended live environment while adhering to the use case specifications and UML class diagrams discussed in Chapter 3. This phase ensures that the deployment aligns with the system's functional requirements and design specifications. Incremental deployment is performed, reflecting the iterative principles of Condensed Agile, to ensure a controlled and

smooth release. Pre-deployment testing is rigorously conducted to confirm that the application is fully operational and free of critical issues. This phase also involves configuring the application for the live environment, documenting deployment procedures, and providing user training to ensure the effective adoption of the system. By aligning the deployment process with the comprehensive specifications outlined in Chapter 3, this phase ensures that the application performs reliably in its intended context.

3.2.6. Review

The review phase in Condensed Agile is a vital step to evaluate the application's performance and overall success following deployment. Feedback is collected from users, stakeholders, and team members to assess the system’s functionality, usability, and adherence to project objectives. The iterative approach of Condensed Agile facilitates the integration of this feedback into future updates, ensuring the application evolves to meet changing needs. Performance metrics and user feedback are analysed to identify areas for improvement, address outstanding issues, and enhance user satisfaction. This phase not only validates the project’s achievements but also establishes a roadmap for ongoing maintenance and iterative enhancement, ensuring the application remains relevant and effective over time.

3.2.7. Summary

This chapter detailed the requirement analysis and application design for the proposed application, WanderWise, starting with an introduction to its objectives and the adoption of the Condensed Agile methodology. Each phase starting from planning, design, develop, test, deploy, and review was thoroughly explained, explaining each of the phases. Planning phase included a survey to gather user insights on challenges and desired features, forming the foundation for the design and development process. Next, the design phase utilised wireframes and various UML diagrams, including activity, class, sequence, and use case diagrams, to

visualise the system's structure, processes, and user interactions effectively. Development phase on the other hand, translated these designs into functional components through iterative coding and testing to ensure quality and reliability. Testing focused on ensuring system usability and functionality through unit and user acceptance testing and deploy phase ensured a seamless rollout while the review phase gathered user feedback to refine and improve the application. This chapter comprehensively outlined the requirement analysis and structured design process, ensuring the proposed solution aligns with user needs and project objectives.

CHAPTER 4: IMPLEMENTATION

4.1. Introduction

This chapter explains the implementation process of the proposed application, WanderWise. It covers each step involved in setting up the development environment, configuring the necessary tools, and building the system modules. Next, this chapter will introduce the role-based access implemented in the system to ensure that users only have access to features relevant to them. A detailed workflow will also be included to give a better understanding of how the application operates from start to end. Furthermore, each module developed for the system will be explained, including how it functions and how it supports the overall objective of WanderWise. Overall, this chapter focuses on translating the design into a working mobile application as well as how each modules piece together.

4.2. Installation and Configuration of System's Components

The development of WanderWise required the installation and configuration of several essential software tools and components to support the full-stack mobile application architecture. Since WanderWise is built entirely using Flutter, which uses Dart as its programming language, the setup focused on preparing an environment that supports both front-end and back-end functionalities within a single codebase. For the development environment, Visual Studio Code (VSC) was chosen as the primary Integrated Development Environment (IDE). On the backend, the application integrates with Firebase Firestore, a cloud-based NoSQL database provided by Google. Setting up Firestore involved creating a Firebase project in the Firebase Console which links it with the Flutter app through configuration files (google-services.json). In addition, OpenRouter was used to power the AI features in WanderWise. All components were configured to work seamlessly together, allowing the app to deliver both strong performance and reliable data management. There will be further detailed elaboration in each subtopic of the chapter.

4.2.1. Visual Studio Code

Visual Studio Code (VSC) is one of the most important tools used in the development of WanderWise. It serves as the main Integrated Development Environment (IDE) where all the Dart code is written and managed. VSC is lightweight yet powerful, and it supports Flutter development through a variety of extensions, making it a suitable choice for building full-stack mobile applications (Vij, 2025). Once Visual Studio Code is downloaded through (https://code.visualstudio.com), setting of Flutter SDK and Java JDK is done next. Hence, successful completion will allow for additional extensions such as Flutter, Dart, and Firebase to be added from the Extensions Marketplace to support the development of this project.

One of the reasons VSC was chosen is due to its user-friendly interface, fast performance, and integrated Git support. It also offers useful features such as code auto-completion, real-time debugging, terminal access, and version control, all in one place. These tools helped to speed up the development process and improve productivity when building WanderWise. Furthermore, the built-in-terminal was most frequently used to run the common key Flutter commands such as *'flutter run'*, *'flutter build'*, and *'flutter pub get'*.

Figure 36 shows an example of writing Flutter code using Visual Studio Code, while Figure 37 illustrates the running of the app on a physical emulator.

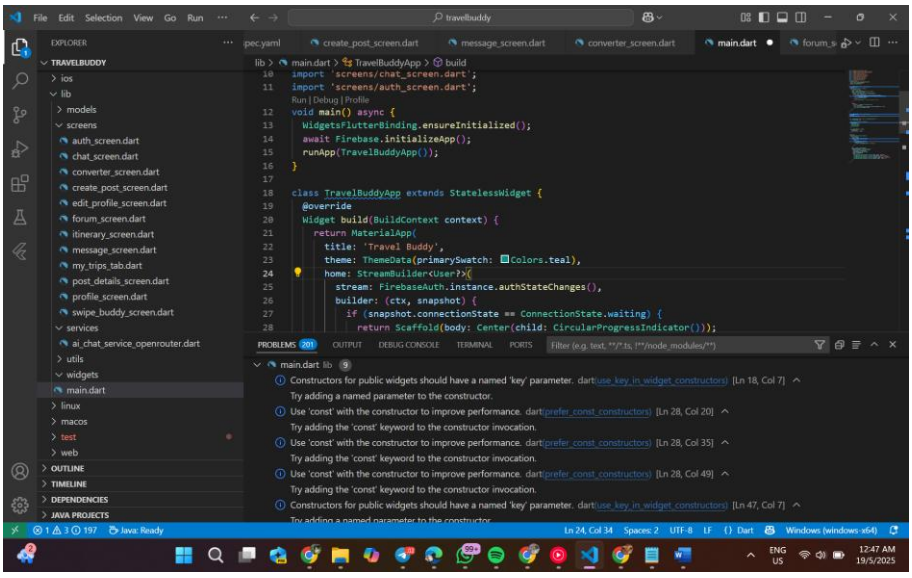


Figure 36. Usage of VSC for WanderWise development

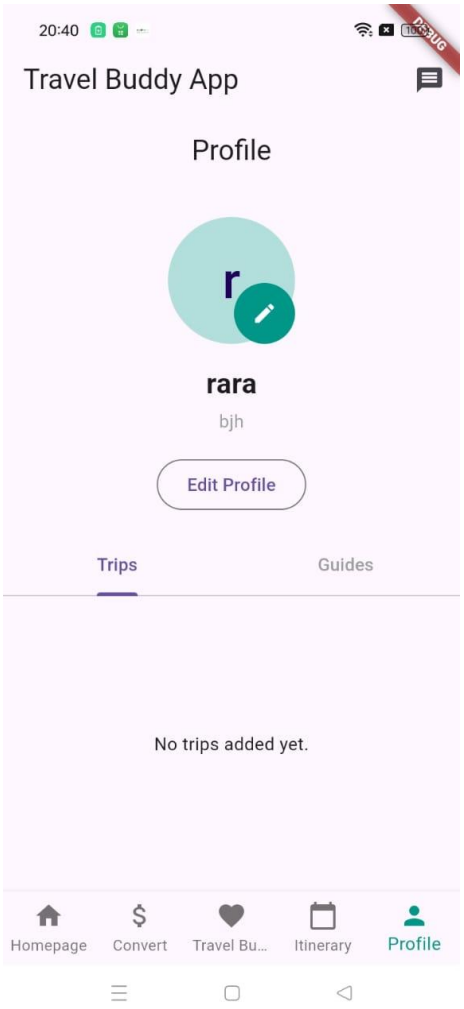


Figure 37. WanderWise running on physical emulator

4.2.2. OpenRouter

One of the key features of WanderWise is its AI-powered travel assistant, which was implemented using OpenRouter. OpenRouter acts as a unified API gateway that allows access to various large language models (LLMs), such as GPT, Mistral, Claude, and others. In this project, OpenRouter (Kumar, 2025) embedded with Mistral was used to integrate AI capabilities into the app, enabling smart features like itinerary suggestions, travel tips, and responsive user support. The configuration began by creating an account (https://openrouter.ai) and generating a personal API key, which is required to authenticate requests to the AI models. This API key was stored securely and integrated into the app using environment variables to prevent exposure in the source code. In the Flutter project, the 'http' package was added to the 'pubspec.yaml' file to handle API requests. Then, a custom service was written in Dart to send POST requests to OpenRouter's API endpoint. These requests included the user's query and

the specific model name (such as ‘openai/gpt-3.5-turbo’), along with the necessary headers for authentication.

The AI responses from OpenRouter were then parsed and displayed in the app’s chat interface, mimicking a real-time conversation. This allowed users to ask travel-related questions, get location-based recommendations, or even receive packing reminders based on their itinerary. The integration was kept lightweight and efficient, ensuring a responsive experience even on mobile devices. Figure below shows the website to gain access towards the personal token. By using OpenRouter, the development process became more flexible as it allowed switching between different LLMs depending on the required performance and cost efficiency.

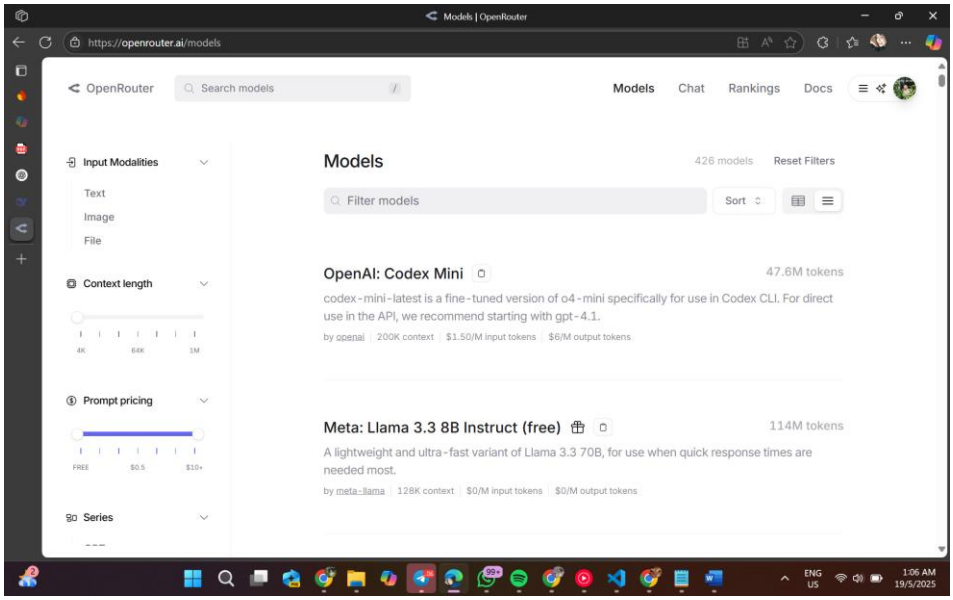


Figure 38. OpenRouter interface for AI Chatbot

4.2.3. Configuring Database Connection

For this project, Firebase Firestore was used as the main cloud database to store and manage all application data in real-time. Firestore is a flexible, scalable NoSQL cloud database provided by Google (Google, n.d.), and it integrates seamlessly with Flutter, which eases most part of the back-end handling. To configure the database connection, the first step was to create a Firebase project through the Firebase Console at (https://console.firebase.google.com). After creating the project, an Android app was registered within Firebase, where a unique *‘google-services.json’* file was generated and added to the Flutter project’s ‘android/app/build.gradle’ directory. This file contains essential configuration details such as the app’s Firebase ID and

API keys. Next, the Firebase SDK was added to the Flutter project by modifying the *'pubspec.yaml'* file mainly to include dependencies like *'firebase_core'*, *'cloud_firestore'*, and *'firebase_auth'*. These packages enable the application to initialize Firebase, perform database operations, and handle user authentication. Figure 39 shows the Firestore database while Figure 40 shows the connection of firebase towards the project.

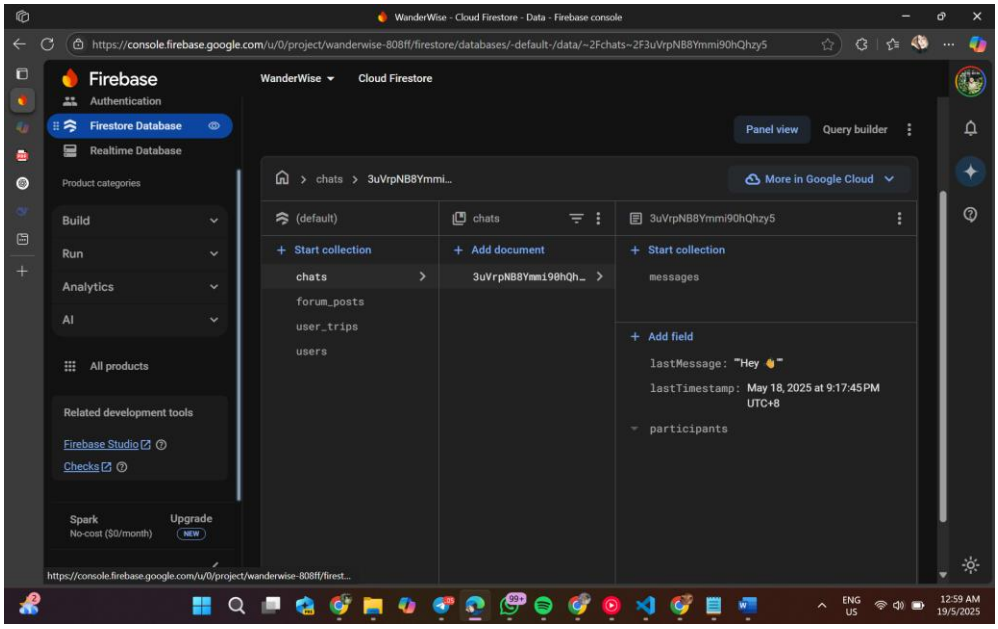


Figure 39. Firestore Database

```
12  void main() async {
13    WidgetsFlutterBinding.ensureInitialized();
14    await Firebase.initializeApp();
15    runApp(TravelBuddyApp());
16  }
```

Figure 40. Initialising firebase connection from Flutter project to Firebase Console

4.3 Introduction the Role Based Access

WanderWise is designed with a single user role in mind which is the travelers. This role encompasses all the functionalities available in the application, from itinerary planning to AI-powered travel assistance. By having just one role, the app simplifies the user experience and reduces complexity in access control. Each traveler has full access to all relevant features of the system without needing to switch between roles or manage permissions.

The key tasks and permissions available to travelers will be outlined below to show how they interact with the different modules of the application.

4.3.1. Traveler

Travelers are the only user type in WanderWise. Every registered user of the application is considered a traveler and has access to the following tasks:

- Register and log in to the app
- Plan personalized itineraries
- Browse, like, and save public itineraries from other users
- Use the AI-powered travel assistant to ask questions or get recommendations
- Convert currencies using the built-in currency converter
- Share tips and itineraries to the travel forum
- View and manage their own profile
- Save destinations or activities to their wishlist
- Match with other travelers through 'Travel Buddy'

Table 7 shows the tasks that can be performed by the traveler

Task/Role	Traveler
Register and log in	/
Create and manage itineraries	/
Browse and save itineraries	/
Use AI assistant	/
Use currency converter	/
Post on travel forum	/
Match with other travelers	/
Message with others	/

Table 7. Roles Performed by Travelers

4.4. Workflow for System's Application

This section outlines the key flow of user interaction throughout the system and how various components work together to deliver a seamless travel companion experience. The application starts with the user authentication process, where the traveler can either register for a new account or log in to their existing one. Once authenticated, the user is redirected to the home interface, where they are introduced to several core features such as itinerary

planning, travel forums, AI assistant through message, travel buddy, and currency converter. Each feature is accessible via a bottom navigation bar, allowing the traveler to switch between functionalities easily.

4.5. Modules for System's Application

In this section the key features designed in Chapter 3 is further explained for its implementation in the WanderWise application. It covers the core modules that make up the app and shows how each part functions in the actual system.

4.5.1. Register/Sign in Module

The Register / Sign In module is one of the core modules in the WanderWise application. This module allows new users to create an account and existing users to log into the app. To register, users are required to provide essential details such as their full name, email address, and password. Upon submission, the user's information is stored in Firebase Firestore, and the authentication credentials (email and password) are handled through Firebase Authentication. Once registered, users can sign in using the same email and password. This ensures secure access and personalized features for each traveler. Figure 41 shows the layout of the registration and sign-in interface in the WanderWise application.

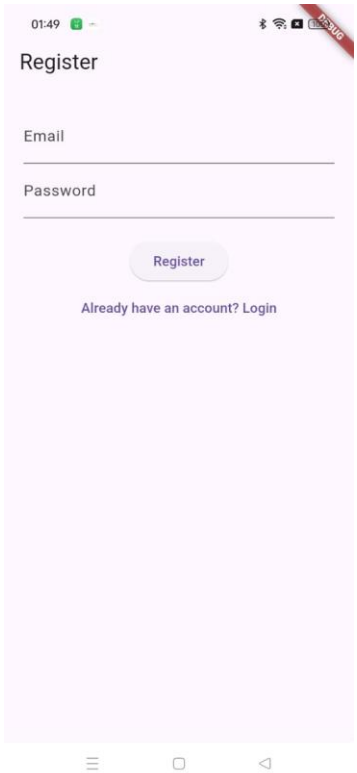


Figure 41. Register/Sign In Module

4.5.2. Forum Module

The Forum Page module in WanderWise serves as a platform where travelers can share itineraries, travel tips, and recommendations. Unlike a fully interactive forum, this page is designed to focus on minimal yet valuable community engagement. Users can view posts made by other travelers, such as travel guides or local suggestions, and interact with them by liking or saving the content. One of the key features of this module is that users can save itineraries shared by others. Users are also able to post their own content for the forum. All forum data is stored and retrieved from Firebase Firestore, ensuring real-time updates and accessibility across different devices. Figure 42 illustrates the layout of the forum page in the WanderWise application.

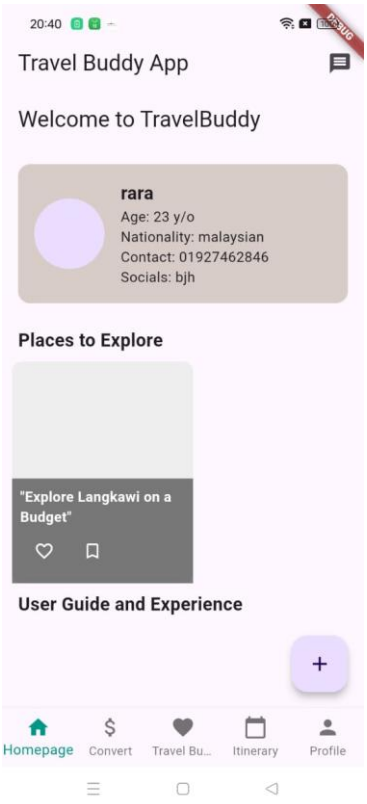


Figure 42. Home Page

4.5.3. Money Converter

The Money Converter module is designed to help travellers easily convert currencies based on real-time exchange rates. This feature is especially useful for those travelling across

different countries, allowing them to plan expenses more accurately. The module uses a third-party API to fetch the latest currency exchange rates. Users simply select the base currency and the target currency, then enter the amount they wish to convert. The converted value is displayed instantly. Users are also able to add these expenses into this feature. This tool helps improve convenience for users who might otherwise need to switch between apps or manually search for conversion rates online. The integration of this module into WanderWise allows it to be accessed quickly from within the app, making travel budgeting simpler and more efficient. Figure 43 shows the interface of the money converter feature.

Figure 43. Money Converter

4.5.4. AI Travel Chatbot/Message Module

The AI Chatbot module is an intelligent assistant embedded in WanderWise to support travellers with quick and relevant travel-related responses. This chatbot is powered by OpenRouter, which allows integration with large language models capable of understanding natural language and giving meaningful suggestions or answers. Users can ask the chatbot a variety of travel-related questions, such as recommendations for places to visit, travel tips, or even help with itinerary planning. The chatbot enhances the user experience by acting as a

virtual travel guide, available at any time during the user's journey. All queries and responses are handled through an interactive chat interface within the app, providing a seamless and user-friendly experience. Figure 44 displays the AI chatbot in action within the WanderWise application.

Meanwhile, the Messaging Platform module allows users to communicate directly with other travellers, especially those connected through the Travel Buddy within the WanderWise app. This feature is useful for connecting with people who may be travelling to the same destination or who have shared helpful itineraries in the forum. The platform supports basic chat functions, such as sending and receiving text messages, and is integrated with Firebase to ensure real-time updates and smooth message delivery. This feature encourages minimal but meaningful community interaction while maintaining the app's focus on simplicity and functionality. Figure 44 shows the messaging interface of the WanderWise app.

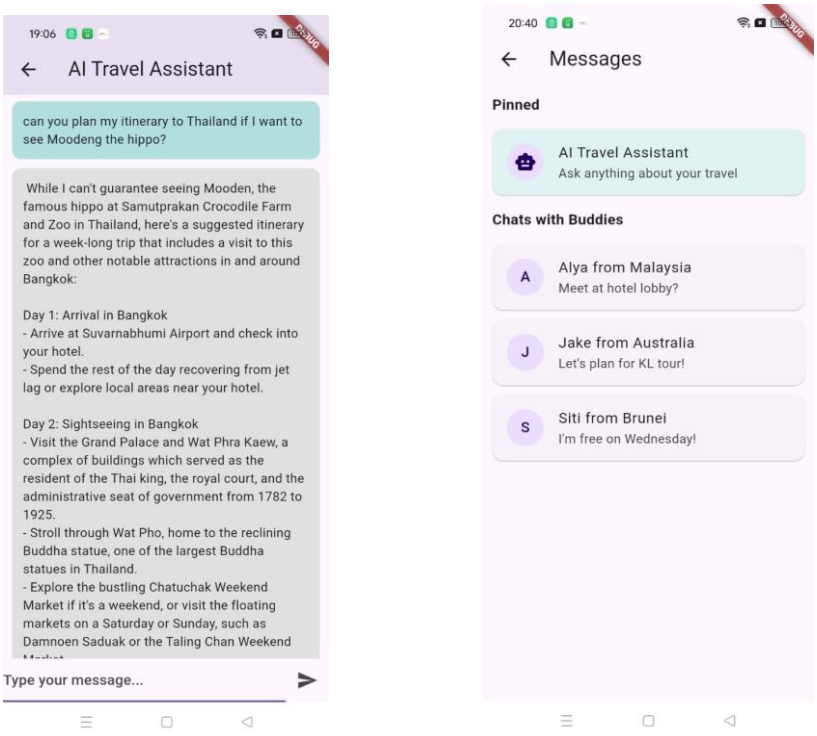


Figure 44. AI Chatbot and Messaging Feature

4.5.5. Travel Itinerary

The Travel Itinerary module is one of the core features of the WanderWise application. It allows users to plan and organise their travel schedules in a structured and efficient manner. Through this module, users can create personalised itineraries by adding locations, activities,

accommodation details, and time slots for each day of their trip. The interface is designed to be user-friendly, enabling travellers to easily input and edit details. Each itinerary is stored in Firebase Firestore, making it accessible across devices and available for updates anytime. This cloud-based storage ensures users do not lose their plans and can retrieve or modify them on the go. Furthermore, if a user wishes to go for a solo trip. The application would suggest matches for the users through Travel Buddy. Figure 45 illustrates the itinerary creation interface in WanderWise.

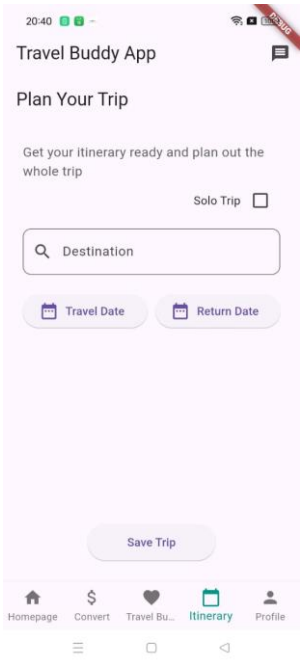


Figure 45. Itinerary Planning Feature

4.5.6. Travel Buddy

The Travel Buddy module in WanderWise is designed to help users connect with other travelers who are planning trips to the same destination or who share similar travel interests. This feature promotes social interaction and collaboration among users, making solo travel more engaging and group travel more organised. Through this module, users can search for potential travel buddies based on destination, travel dates, or trip type (e.g., backpacking, sightseeing, food trips). Once a match is found, users can view each other's profiles and initiate conversations using the in-app messaging platform. The module aims to foster safe and meaningful connections, so only verified users with completed profiles can access the feature. It is particularly useful for users who prefer to meet new people during their travels or need companions for specific activities or shared expenses. Figure 46 displays the interface of the Travel Buddy matching system within WanderWise.

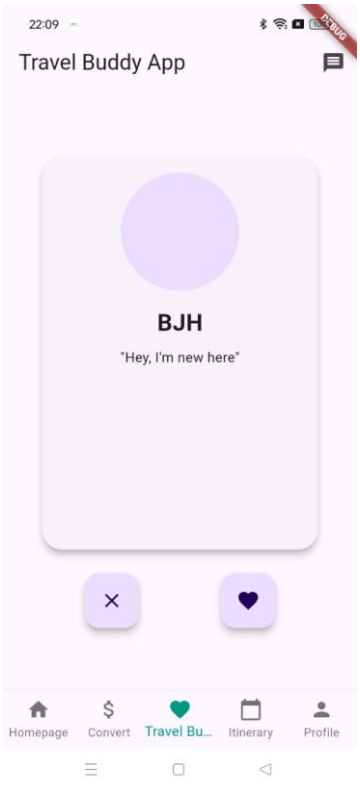


Figure 46. Travel Buddy

4.5.7. Profile Page

The Profile Page module serves as the central hub for users to view and manage their personal information within the WanderWise application. It displays essential details such as the user's name, profile photo, travel interests, saved itineraries, past trips, and preferred destinations. This module allows users to edit their profiles by updating personal information, changing their profile picture, or modifying travel preferences to improve feature recommendations (e.g., Travel Buddy matches or suggested itineraries). Only the user can modify their own profile, and sensitive information is kept secure. Other users can view a simplified public version of the profile (if made public) to assess compatibility for features like the Travel Buddy module. Figure 47 shows the layout of the Profile Page interface in WanderWise.

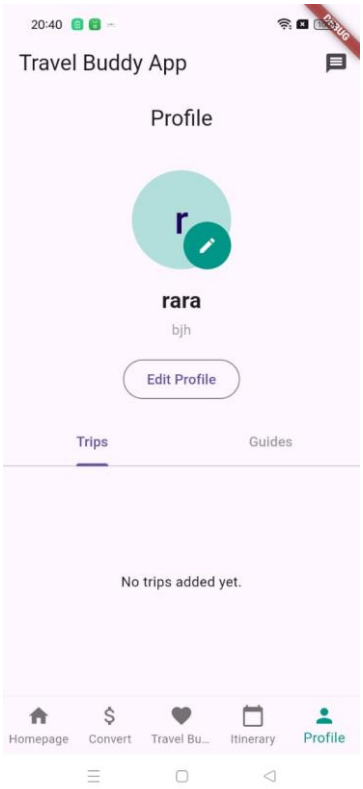


Figure 47. Profile Page

4.6. Summary

This chapter has covered the implementation of the WanderWise mobile application. Several key aspects related to the development and integration of core modules have been discussed in detail. The chapter began with an overview of the development setup and tools used, highlighting how these tools supported the efficient building and testing of the application. Additionally, the roles and access levels of different users have been clarified, focusing on the single user type which is the traveler and how they interact with the system. Various modules such as the AI Travel Assistant, Itinerary Planner, Currency Converter, Travel Buddy, Forum, and Profile Page were implemented based on the designs proposed in the earlier chapter. This chapter has demonstrated how each design component from Chapter 3 was translated into a fully functional and user-friendly application.

CHAPTER 5: TESTING

5.1.Introduction

This chapter details the testing conducted for WanderWise. Throughout the testing phase of the mobile application, there were two types that have been done which are functional testing and non-functional testing. Functional testing was conducted to verify that all core features of the app, such as user registration, travel buddy matching, messaging, and itinerary creation, perform accurately and reliably whereas the non-functional testing was carried out to evaluate the app's performance, security, scalability, and overall user experience. The aim of the two testing was to not only ensure that it meets the functional expectations of its users but also delivers an optimized, secure, and engaging experience that encourages widespread adoption.

5.2.Functional Testing

Functional testing is a crucial phase in the development of the app to ensure that all features and functions work as intended. This type of testing focuses on verifying that the app performs its core tasks correctly, such as user registration, travel buddy matching, itinerary creation, and messaging and typically conducted by the developer without the direct involvement of the target users. The success of functional testing is pivotal as it sets the foundation for more advanced testing phases, such as non-functional testing and user acceptance testing.

5.2.1. System Testing

System testing is conducted to test the overall functions of the WanderWise app. This test ensures that the system meets the requirements outlined in Chapter 3. In this section, the WanderWise app is tested as a whole, including core features such as user registration, travel buddy matching, itinerary creation, and messaging. Table 8 represents the test cases for system testing.

Table 8. System Testing Test Case

Test Objective: To test the overall functions of WanderWise				
Test ID	Test Case	Expected Result	Actual Result	Pass/Fail
ST1	User registers to use the application	User should be registered and able to access the application	User is registered and able to access the application	Pass
ST2	User logs in to the application	User should be able to log in and access the application	User able to log in and access the application	Pass
ST3	User views potential travel buddies and swipes left/right	User should be able to swipe left (dislike) or right (like)	User is able to swipe and match with a travel buddy	Pass
ST4	User creates a travel itinerary	User should be able to create an itinerary with destinations	User is able to create an itinerary with destinations	Pass
ST5	User saves a travel buddy match	The match should be saved in the user's list of matches	Match is saved to the user's match list	Pass
ST6	User sends a message to a matched travel buddy	User should be able to send and receive messages with a match	Message is sent and received successfully	Pass

5.2.2. Unit Testing

Unit testing is conducted to test each module or unit of the WanderWise application. The goal of this test is to ensure that each module of the application is working as expected. The unit testing will be conducted for both the main WanderWise app and the admin interface.

5.2.2.1. Main Application

This section will conduct unit testing for the main application of WanderWise.

Test Plan 1: Start Page

Table below shows the test case for Start Page.

Table 9. Test Case for Start Page

Test Objective: To verify the navigation to the WanderWise start page				
Test ID	Test Case	Expected Result	Actual Result	Pass/Fail
SP1	Tap on the application's logo	The application will open and display the start page	The application is opened and displayed the start page	Pass
SP2	Tap on the Register button	The user should be directed to the register page	The user is navigated to the register page	Pass
SP3	Tap on the Login button	The user should be directed to the login page	The user is navigated to the login page	Pass
SP4	Previously logged in users tap the application's logo	The user should be directed to the home page of the application	The user is navigated to the home page of the application	Pass

Test Plan 2: Registration Page

Table below shows the test case for Registration Page.

Table 10. Test Case for Registration Page

Test Objective: To validate whether the application can register a new user.				
Test ID	Test Case	Expected Result	Actual Result	Pass/Fail
RP1	Enter all required information inside the input fields	The user should be able to use the input field to enter all required information	The user is able to type in all required information using the input fields	Pass
RP2	Click on the register button with some fields empty or wrong format	The user will be alerted to enter all required information	The user is displayed with a toast message asking them to enter all required information	Pass
RP3	Click on the register button with all required information filled and in the correct format	The registration should be successful, and all the details should be inserted into the database	The registration is successful, and all information is saved into the database	Pass

Test Plan 3: Login Page

Table below shows the test case for Login Page.

Table 11. Test Case for Login Page

Test Objective: To examine the ability of the application to log the user into the application.				
Test ID	Test Case	Expected Result	Actual Result	Pass/Fail
LP1	User enters login credentials that do not exist in the	The application should alert the user	Error message is displayed stating that the email does	Pass

	database and taps on the Login button	with an error message	not exist in the application records	
LP2	User enters the correct email address but with the wrong password and taps on the Login button	The application should alert the user with an error message	Error message is displayed stating that the password is incorrect	Pass
LP3	User enters correct login credentials as registered and taps on the Login button	The user should be directed to the home page of the application	The user is directed to the home page of the application	Pass

Test Plan 4: Home Page

Table below shows the test case for Home Page.

Table 12. Test Case for Home Page

Test Objective: To view the homepage including guides, accessing AI travel buddy, and random destination picker				
Test ID	Test Case	Expected Result	Actual Result	Pass/Fail
HP1	Tap on the AI travel buddy icon on the homepage	The user should be directed to the AI Travel Assistant chat page	The user is directed to the AI Travel Assistant chat page	Pass
HP2	Tap on the "Random Destination Picker" button	The user should see a random travel destination suggestion	The user is shown a random travel destination suggestion	Pass

HP3	Tap on a guide in the homepage list	The user should be directed to the selected guide's detail page	The user is directed to the selected guide's detail page	Pass
HP4	Tap on the add button to add user's own guides	The user should be directed to create forum page	The user is directed to create forum page	Pass
HP5	Tap on the contact card	The user should be prompted with an enlarged version of the contact card	The user should be prompted with an enlarged version of the contact card	Pass

Test Plan 5: Travel Buddy Matching Page

Table below shows the test case for Travel Buddy Matching Page.

Table 13. Test Case for Travel Buddy Matching Page

Test Objective: To test if the user can match with potential travel buddies.				
Test ID	Test Case	Expected Result	Actual Result	Pass/Fail
TB1	Click on X button (dislike) on a travel buddy	The travel buddy should not appear again	The travel buddy is no longer shown	Pass
TB2	Click on the love button (like) on a travel buddy	The travel buddy should be added to the user's list of liked buddies	The travel buddy is added to the liked list	Pass
TB3	Tap on a match notification	The user should be prompted to send a text to the matched buddy and be directed to the chat	The user is prompted to send a text to the matched buddy and be directed to the chat	Pass

		screen with their match	screen with their match	
TB4	Tap on the liked profiles button	The user should be prompted with a list of profiles that they have liked	The user is prompted with a list of profiles that they have liked	Pass

Test Plan 6: Profile Page

Table below shows the test case for Profile Page.

Table 14. Test Case for Profile Page

Test Objective: To test the ability to display and update user profile information.				
Test ID	Test Case	Expected Result	Actual Result	Pass/Fail
PP1	Tap on the profile icon	The user should be directed to their profile page	The user is directed to their profile page	Pass
PP2	Tap on the "Edit Profile" button	The user should be directed to the edit profile page	The user is directed to the edit profile page	Pass
PP3	Tap on the tab bars of ‘Trips’, ‘Guides’, and ‘Liked’	The user should be able to view their ‘Trips’, ‘Guides’, and ‘Liked’ posts.	The user is able to view their ‘Trips’, ‘Guides’, and ‘Liked’ posts.	Pass
PP4	Tap on the saved button	The user should be able to view their saved posts.	The user is able to view their saved posts.	Pass

Test Plan 7: Travel Itinerary Page

Table below shows the test case for Travel Itinerary Page.

Table 15. Test Case for Travel Itinerary Page

Test Objective: To test if the user can create, view, and update their travel itinerary.				
Test ID	Test Case	Expected Result	Actual Result	Pass/Fail
IT1	Tap on the "Create Itinerary" button	The user should be directed to the itinerary creation page	The user is directed to the itinerary creation page	Pass
IT2	Add a new destination to the itinerary	The destination should be added to the itinerary	The destination is added to the itinerary successfully	Pass
IT3	Tick on the Solo Trip	The user should be prompted to the Travel Buddy page	The user is prompted to the Travel Buddy page	Pass
IT4	Tap on the "Save Itinerary" button	The itinerary should be saved successfully to the user's profile	The itinerary is saved successfully	Pass

Test Plan 8: Money Converter

Table below shows the test case for Money Converter Page.

Table 16. Test Case for Money Converter Page

Test Objective: To test if the money converter screen works as expected, including real-time conversion of currencies and the ability to fetch data from an external API.				
Test ID	Test Case	Expected Result	Actual Result	Pass/Fail
MC1	User selects a currency to convert from and a currency to convert to	The currencies selected should be displayed correctly	The currencies selected are displayed correctly	Pass
MC2	User enters a valid amount for conversion	The app should display the converted amount correctly based on real-time exchange rates	The amount is correctly converted and displayed	Pass
MC3	User enters an invalid or empty amount	The app should display an error message or prompt the user to enter a valid amount	An error message is displayed asking the user to enter a valid amount	Pass
MC4	User taps the "Convert" button with valid inputs	The app should trigger the conversion process and display the result	The result is shown as expected after conversion	Pass
MC5	User taps the "Convert" button with invalid inputs	The app should display an error message indicating a failure to fetch exchange rates	An error message is shown stating that the exchange rate could not be fetched	Pass
MC6	User changes the currency	The selected currencies should	The currencies are updated and the	Pass

	selection after entering an amount	update, and the conversion result should adjust accordingly	conversion result is recalculated	
MC7	User clicks on the "Save as Expense" button after converting the currency	The converted amount and selected currencies should be saved as an expense in the user's profile	The expense is successfully saved in the user's profile	Pass
MC8	User accesses the "Saved Reports" section	The user should be able to view their saved conversions/expenses	The saved conversions/expenses are displayed correctly in the reports section	Pass
MC9	User accesses the "Expenses" page	The user should be able to view their previous conversions and expenses	The expense history is displayed correctly	Pass

5.3.Non-Functional Testing

Non-functional testing is conducted to evaluate the non-functional aspects of the WanderWise app, including its performance, user experience, loading speed, and overall usability. In this part, three types of non-functional testing are carried out which are performance testing, portability testing, compatibility testing, and usability testing.

5.3.1. Usability Testing

Usability testing is conducted to evaluate the user experience and determine how effectively users can interact with the WanderWise app. Feedback is gathered from targeted users amounted to 31 participants after they have interacted with the app. Participants ranged from 18–45, with a mix of genders (17 females, 8 males, and 6 prefer not to say). These details were used to understand feature preferences. In this case, the testing was conducted to gather

insights into how easy and intuitive the app is to use, as well as identifying potential areas for improvement.

The usability test was conducted by distributing the APK file to participants, who installed the WanderWise app on their Android devices. After using the app and exploring all its features, they were asked to complete a Google Form questionnaire. The form contained Likert scale questions and open-ended feedback fields related to system functionality, user interface design, and overall satisfaction.

5.3.1.1.System Functionality of WanderWise

The system functionality is evaluated to ensure that the WanderWise app performs as intended in real-world conditions. This includes ensuring that key features, such as user registration, travel buddy matching, messaging, and itinerary creation, are easy to use and function as expected. System functionality testing also covers how well the app handles user inputs, interactions, and navigation. A survey was conducted to gather feedback from users. Table 17 shows the results of usability testing under system functionality.

Table 17. Results of Usability Testing under System Functionality

No.	Test Elements	1	2	3	4	5
1	Did the AI Chatbot provide useful responses			2	1	28
2	The currency converter was helpful in converting money and recording expenses				9	22
3	It was easy to plan an itinerary and embedding a guide in it				6	25
4	Travel Buddy was convenient to use, and users can see their liked profiles.				3	28
5	Viewing guides are easily navigated and can like or save the guides				8	23

5.3.1.2.User Interface Design of WanderWise

The user interface (UI) design of the WanderWise app is assessed for its ease of use, aesthetic appeal, and overall user experience. Usability testing here focuses on evaluating the UI elements, such as buttons, icons, navigation flows, and readability. The goal is to ensure that users can easily navigate through the app, find the features they need, and enjoy a seamless

experience while using the app. Feedback collected during testing will highlight areas for improvement in the UI design.

Table 18. Results of Usability Testing under User Interface Design

No.	Test Elements	1	2	3	4	5
1	How easy was it to navigate through the app?			2	15	14

5.3.2. Performance Testing

Table below shows the performance testing for WanderWise.

Table 19. Performance Testing for WanderWise

Test Objective: To test the overall performance of WanderWise.				
Test ID	Test Case	Expected Result	Actual Result	Pass/Fail
PT1	The application must respond to user actions within 5 seconds for any type of action	The application should respond to user actions within 5 seconds.	The application responds within 5 seconds under normal network conditions.	Pass
PT2	The application must handle the simultaneous actions of multiple users	The app should handle multiple actions without freezing or crashing	The app handles multiple simultaneous actions without any significant issues	Pass

5.3.3. Compatibility Testing

Table below shows the compatibility testing for WanderWise.

Table 20. Compatibility Testing for WanderWise

Test Objective: To test the compatibility of WanderWise across different devices and operating systems				
Test ID	Test Case	Expected Result	Actual Result	Pass/Fail
CT1	The application is compatible with Android version 8.0 and above.	The app should work seamlessly on Android devices with version 8.0 and above.	The app works on Android devices running version 8.0 and above	Pass

CT2	The application is compatible with iOS version 12.0 and above	The app should work seamlessly on iOS devices with version 12.0 and above.	The app works on iOS devices running version 12.0 and above.	Pass

5.3.4. Overall Evaluation

The overall evaluation by the targeted users is also collected. All 31 test participants are asked about their experience of using the application as well as providing feedback on areas to improve for WanderWise in future works. Table 21. Results of User Overall Satisfaction on App's Function shows the overall satisfaction while Table 22. Suggestions on Future Works for WanderWise shows the suggestions for improvement in future works.

Table 21. Results of User Overall Satisfaction on App's Function

No.	Test Elements	1	2	3	4	5
1	How satisfied are you with the overall app performance?			1	12	18
2	Do you feel WanderWise helped improve your travel experience?					31

Table 22. Suggestions on Future Works for WanderWise

No.	Suggestions
1	Add search or filter button at guide page
2	Add more currency conversion
3	Able to view past dislike profiles
4	Feature to chat with locals from the area
5	Forgot password feature and email authentication upon registering

From the results of user's overall evaluation, 12 participants out of 31 are very satisfied with the application. While 18 deemed to be satisfied and another one is neutral. Furthermore, all 31 participants noted that WanderWise helped to improve their travel experience. Apart from that, there were also few suggestions from the users on future updates or features that they would like to see upon the further development of WanderWise.

5.4.Summary

This chapter outlined the testing methods conducted for the WanderWise app. Two main types of testing were performed which are functional testing and non-functional testing. Functional testing includes unit testing, which tests individual components of the application, and system testing, which evaluates the overall functionality of the app. Non-functional testing, on the other hand, covers performance testing, portability testing, compatibility testing, and usability testing, ensuring the app performs well, is compatible across devices, is portable, and provides a seamless user experience.

CHAPTER 6: CONCLUSION & FUTURE WORK

6.1. Introduction

Chapter 6 concludes the development and testing process of WanderWise, a mobile application designed to enhance the travel experience by connecting solo travellers with potential travel buddies. This chapter provides an overview of the achievements during the project, highlights the limitations encountered, summarises the findings, and proposes possible future developments for the app. The purpose of this chapter is to reflect on the accomplishments of the project and outline the areas that could benefit from further enhancements.

6.2. Objective Achievement

The primary objective of the WanderWise project was to create a functional mobile application that enables users to ease their travel experience through a centralised travel companion app. Table below shows the aimed objectives with their achievements.

Table 23. Comparison of Aimed and Achieved Objectives

Aimed Objectives	Achieved Objectives
To study and analyse existing systems of travel planning and itinerary management applications.	Several existing systems were studied and analysed as part of the project.
To design and implement a comprehensive mobile application that integrates essential travel tools.	WanderWise development consists of integrated essential travel tools such as a currency converter, an itinerary planner, and user guides.
To test and evaluate the usability and functionality of the WanderWise app through user feedback and usability testing.	Usability feedback and testing were done to evaluate the functionality of WanderWise.

6.3. Project Limitations

While significant progress was made in developing WanderWise, there were several limitations encountered during the project:

- **Limited Device Compatibility:** The app was primarily tested on Android devices, and cross-platform compatibility issues with iOS were not fully addressed, limiting its usability on all devices.
- **Real-Time Data Accuracy:** The travel buddy matching feature relies on user inputs, which can sometimes lead to inaccurate matches if users do not fully complete their profiles or update their travel plans regularly.
- **Performance on Low Network Conditions:** While the app performs well under optimal network conditions, slow or intermittent connections may affect certain features, such as real-time chat and travel buddy matching.
- **Limited Destinations:** The random destination picker feature is currently limited to a predefined list of destinations, which could be expanded to offer a wider range of options.

6.4. Conclusion

In conclusion, the WanderWise app was successfully developed to meet the core objectives of connecting solo travelers with potential travel companions and facilitating trip planning. The project achieved its goals of building a secure platform for user interaction, creating intuitive features for travel buddy matching and itinerary management, and ensuring smooth user communication. Despite the limitations mentioned, the app offers valuable functionality for users seeking to connect with others for travel purposes, helping them plan their trips more effectively. Testing and user feedback have provided valuable insights into areas for improvement, ensuring a stronger foundation for future development.

6.5. Future Works

While WanderWise has made significant strides, there are several areas where improvements can be made, and additional features can be incorporated in future versions. First, the expansion for cross-platform compatibility where it supports iOS and other platforms to ensure broader reach and usability across different devices. Next, adding more destinations to the random destination picker feature to give users more options for trip planning, introducing a profile verification system to increase trust among users and improving the matching accuracy. Other than that, WanderWise could work on offline functionality where the app functions in low or no-network conditions, particularly for itinerary management and previously downloaded travel guides. Furthermore, for future development could include integrating external travel services such as flight bookings, accommodation reservations, and activity recommendations

directly within the app. Lastly, to increase better matching and optimisation use of AI, WanderWise could be integrated with machine learning algorithms to enhance the matching process by analysing user preferences and past behaviour for more accurate travel buddy recommendations.

REFERENCES

Amajuoyi, Prisca & Benjamin, Lucky & Adeusi, Kudirat. (2024). Agile methodologies: Adapting product management to rapidly changing market conditions. GSC Advanced Research and Reviews. 19. 249-267. 10.30574/gscarr.2024.19.2.0181. Wong, W. C. S., & WAHID, D. N. B. W. (2024). Trek: Real-Time Collaborative Travel Planner Mobile Application. *Applied Information Technology And Computer Science*, *5*(2), 480-499. https://penerbit.uthm.edu.my/periodicals/index.php/aitcs/article/download/16190/5698

Andruszków, K. (2024, January 29). *Rad vs. agile: Which methodology fits your project?*. Main page. https://ulansoftware.com/blog/rad-vs-agile

Barker, S. (2024, October 23). The rise and fall of google trips: Charting Google's travel ambitions. Expertbeacon. https://expertbeacon.com/the-rise-and-fall-of-google-trips-charting-googles-travel-ambitions/

GeeksforGeeks. (2024, July 12). *Difference between agile testing and waterfall testing*. https://www.geeksforgeeks.org/difference-between-agile-testing-and-waterfall-testing/

Google. (n.d.). *Firebase*. https://firebase.google.com/

Hamilton, T. (2024, April 24). *What is agile testing? Process & Life Cycle*. Guru99. https://www.guru99.com/agile-testing-a-beginner-s-guide.html

Harris, K. (2024, November 8). *7 best travel apps*. Wanderlog blog. https://wanderlog.com/blog/2024/10/23/7-best-travel-apps/

Kumar, R. (2025, January 29). *What is openrouter?*. AiOps Redefined!!! https://www.theaiops.com/what-is-openrouter/

Laoyan, S. (2024, February 2). What is agile methodology? (a beginner's guide) [2024] Asana. https://asana.com/resources/agile-methodology

Microsoft. (2021, November 3). Collaborate on GitHub. RSS. https://code.visualstudio.com/docs/sourcecontrol/github

Microsoft. (n.d.). *Visual Studio Code*. https://code.visualstudio.com/

Nimbalkar, N. (2020), "Must have features for travel planner app development",CustomerTHINK, available at: https://customerthink.com/must-have-features-for-travel-planner-appdevelopment/

Olson, G. (2023, March 13). *Review: Why Tripit is the tool every traveler needs*. Thrifty Traveler. https://thriftytraveler.com/guides/travel/tripit-review/

Reichert, A. (2021, July 9). Condensed Agile sprint timeline pros and cons: TechTarget. Search Software Quality. https://www.techtarget.com/searchsoftwarequality/tip/Condensed-Agile-sprint-timeline-pros-and-cons

Sia, P. Y.-H., Saidin, S. S., & Iskandar, Y. H. (2022). Systematic review of Mobile Travel Apps and their smart features and challenges. Journal of Hospitality and Tourism Insights, 6(5), 2115–2138. https://doi.org/10.1108/jhti-02-2022-0087

Siddharta, A. (2024, September 16). *Malaysia: Number of mobile app downloads 2023*. Statista. https://www.statista.com/statistics/1398226/malaysia-number-of-mobile-app-downloads/#:~:text=In%202022%2C%20people%20in%20Malaysia%20downloaded%20a%20total,U.S.%20dollars%20on%20mobile%20apps%20and%20in-app%20purchases.

Sustacha, I., Baños-Pino, J. F., & Del Valle, E. (2023). The role of technology in enhancing the tourism experience in smart destinations: A meta-analysis. *Journal of Destination Marketing & Management*, *30*, 100817. https://doi.org/10.1016/j.jdmm.2023.100817

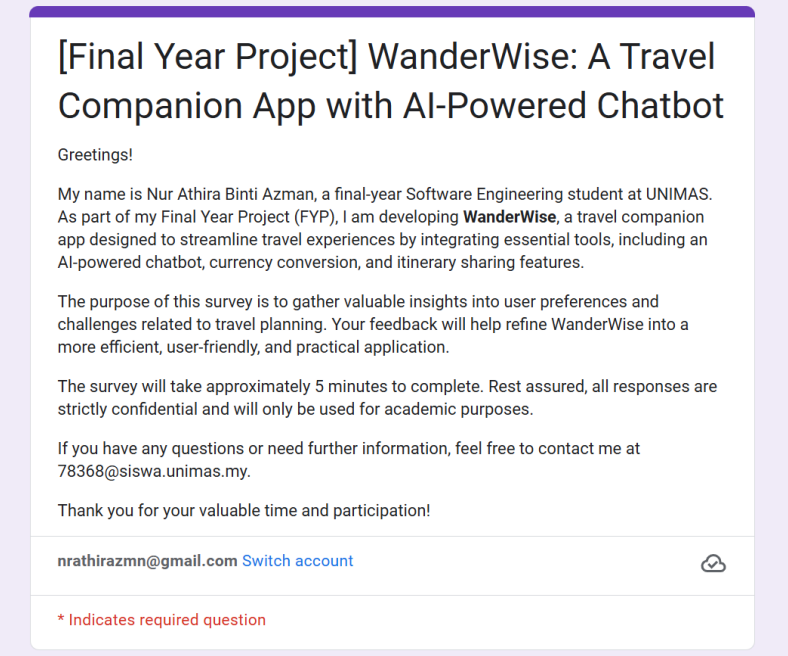
Vij, A. (2025, February 15). Visual studio code - how to download and install VSC. https://www.thewindowsclub.com/visual-studio-code-download

Wanderlog. (n.d.). *Wanderlog: Trip planner app and website*. https://wanderlog.com/

Wong, W. C. S., & WAHID, D. N. B. W. (2024). Trek: Real-Time Collaborative Travel Planner Mobile Application. *Applied Information Technology And Computer Science*, *5*(2), 480-499. https://penerbit.uthm.edu.my/periodicals/index.php/aitcs/article/download/16190/5698

APPENDIX

(FYP 1) Google Form Survey



[Final Year Project] WanderWise: A Travel Companion App with AI-Powered Chatbot

Greetings!

My name is Nur Athira Binti Azman, a final-year Software Engineering student at UNIMAS. As part of my Final Year Project (FYP), I am developing **WanderWise**, a travel companion app designed to streamline travel experiences by integrating essential tools, including an AI-powered chatbot, currency conversion, and itinerary sharing features.

The purpose of this survey is to gather valuable insights into user preferences and challenges related to travel planning. Your feedback will help refine WanderWise into a more efficient, user-friendly, and practical application.

The survey will take approximately 5 minutes to complete. Rest assured, all responses are strictly confidential and will only be used for academic purposes.

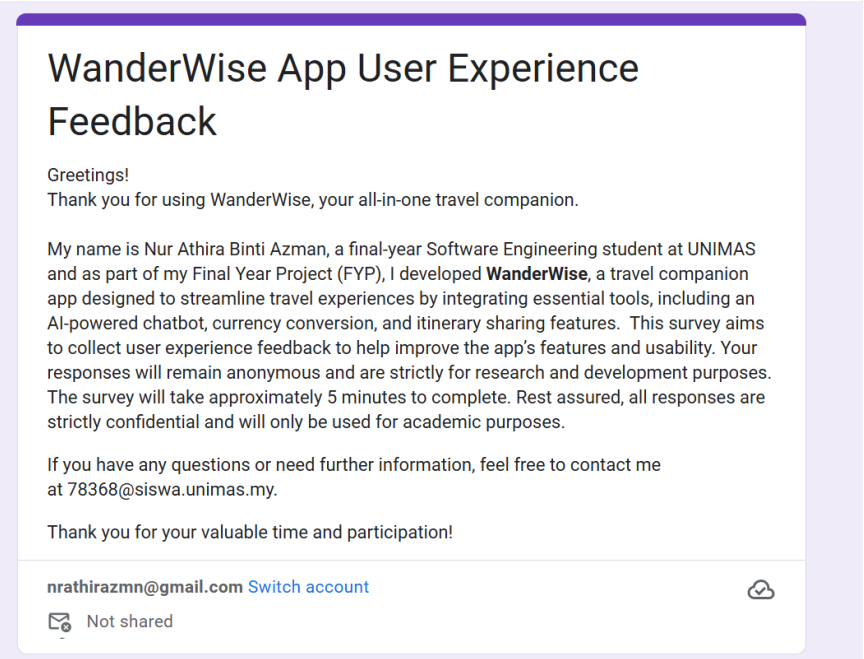
If you have any questions or need further information, feel free to contact me at 78368@siswa.unimas.my.

Thank you for your valuable time and participation!

nrathirazmn@gmail.com Switch account

* Indicates required question

(FYP 2) Google Form Survey – Usability Testing



WanderWise App User Experience Feedback

Greetings!
Thank you for using WanderWise, your all-in-one travel companion.

My name is Nur Athira Binti Azman, a final-year Software Engineering student at UNIMAS and as part of my Final Year Project (FYP), I developed **WanderWise**, a travel companion app designed to streamline travel experiences by integrating essential tools, including an AI-powered chatbot, currency conversion, and itinerary sharing features. This survey aims to collect user experience feedback to help improve the app's features and usability. Your responses will remain anonymous and are strictly for research and development purposes. The survey will take approximately 5 minutes to complete. Rest assured, all responses are strictly confidential and will only be used for academic purposes.

If you have any questions or need further information, feel free to contact me at 78368@siswa.unimas.my.

Thank you for your valuable time and participation!

nrathirazmn@gmail.com Switch account

Not shared