



Faculty of Computer Science and Information Technology

Kuching Historical Tour Guide Mobile Application (“LelengauKch”)

Wong Qin Yong

(81515)

Bachelor of **Software Engineering** with Honours

Ijazah Sarjana Muda Kejuruteraan Perisian dengan Kepujian

2025

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Madam Nurfaeza Binti Jali, for her guidance, support, and encouragement throughout the development of this project. Her expertise and insights have been instrumental in shaping this work. I would also like to extend my appreciation to Miss Wee Bui Lin for being supportive and providing constructive feedback on my report, which has been invaluable in refining the quality of my work. I am also extremely grateful to my family and friends for their unwavering support and understanding throughout this journey. Their encouragement has inspired me to persevere and give my best to this project. Finally, I am grateful for the opportunity to work on this project, which has allowed me to gain valuable experience and enhance my skills as a software engineering student.

Table of Contents

Declaration.....	i
Acknowledgements.....	ii
Table of Contents.....	iii
List of Figures.....	vii
List of Tables.....	x
List of Abbreviations.....	xi
Abstract.....	xii
Abstrak.....	xiii
CHAPTER 1: INTRODUCTION.....	1
1.1 Introduction.....	1
1.2 Problem Statement.....	1
1.3 Objectives.....	2
1.4 Brief Methodology.....	2
1.5 Scopes.....	5
1.6 Significance of Project.....	5
1.7 Project Schedule.....	6
1.8 Expected Outcome.....	6
1.9 Chapter Summary.....	7
1.9.1 Chapter 1 Introduction.....	7
1.9.2 Chapter 2 Literature Review.....	8
1.9.3 Chapter 3 Requirement Analysis and Design.....	8
1.9.4 Chapter 4 Implementation.....	9
1.9.5 Chapter 5 Testing.....	9
1.9.6 Chapter 6 Conclusion and Future Work.....	10
CHAPTER 2: LITERATURE REVIEW.....	11
2.1 Overview.....	11
2.2 Background Study.....	11
2.3 Review of Existing Similar Applications.....	12
2.3.1 SmartGuide.....	13
2.3.2 Sarawak Travel.....	18
2.3.3 Travel Malaysia.....	22
2.4 Comparison of the Mobile Applications.....	26
2.5 Review on Technology and Tools.....	27

2.5.1 Visual Studio Code	27
2.2.5.2 Flutter.....	28
2.5.3 Dart	30
2.5.4 Firebase.....	31
2.5.5 Google Maps.....	32
2.6 Hardware.....	33
2.7 Summary	34
CHAPTER 3: METHODOLOGY	35
3.1 Introduction.....	35
3.2 Methodology	35
3.2.1 Extreme Programming (XP)	35
3.2.1.1 Planning	36
3.2.1.2 Design	36
3.2.1.3 Coding.....	36
3.2.1.4 Testing.....	37
3.2.1.5 Listening	37
3.2.2 Kanban	38
3.3 Requirement Analysis.....	39
3.3.1 User Requirement	39
3.3.2 Use Case Diagram.....	60
3.3.3 Use Case Description.....	61
3.4 Design	73
3.4.1 Class Diagram.....	73
3.4.2 Database design	75
3.4.3 Data Dictionary.....	76
3.4.4 Architecture Design	77
3.4.5 Activity Diagram	78
3.4.6 Sequence Diagram	79
3.5 Wireframes.....	82
3.6 Summary	89
CHAPTER 4: IMPLEMENTATION	90
4.1 Introduction.....	90
4.2 Installation and Configuration of System's Components	90
4.2.1 Visual Studio Code	90
4.2.2 Firebase.....	92

4.2.4 Firebase Vertex Ai	94
4.3. Implementation of the Proposed System	95
4.3.1 Introduction of User Roles.....	96
4.3.2 Workflow for system's Application.....	97
4.4 Modules of system's Application	100
4.4.1 Account Registration	100
4.4.2 Account Login	102
4.4.3 View Historical Sites List	104
4.4.4 Search for historical site in map.....	106
4.4.5 Using quick action filter in map.....	108
4.4.6 Initiate Chat.....	110
4.4.7 View News List.....	112
4.4.8 Search for News	114
4.4.9 News and Events Sharing	115
4.4.10 Participate in Quiz.....	117
4.4.10 View Quiz Explanation and History	119
4.4.11 View Account Details	120
4.4.12 View About Us Page.....	122
4.4.13 Manage Newsletter Email Subscription.....	124
4.4.14 Manage Historical Sites (Admin)	126
4.4.15 Manage News and Events (Admin)	130
4.4.16 View Profile (Admin)	133
4.4.17 User Account Management (Admin).....	134
4.4.18 Summary of Features	136
4.5 Summary	138
CHAPTER 5: TESTING.....	139
5.1 Introduction.....	139
5.2 Functional Testing	139
5.2.1 Test Cases	139
5.2.2 System Testing.....	153
5.2.2.1 Introduction and Objectives.....	153
5.2.2.2 Testing Approach and Scope	154
5.2.2.3 Test Execution and Results.....	155
5.3 Non-Functional Testing	157
5.3.1. Usability Testing.....	157

5.3.1.1 Application Usability Evaluation.....	158
5.3.1.2 Feature-Specific Feedback.....	165
5.3.1.3 Overall Satisfaction Feedback	166
5.4 Summary	168
CHAPTER 6: CONCLUSION AND FUTURE WORKS.....	169
6.1 Introduction.....	169
6.2 Objective Achievements	169
6.3 Project Limitations.....	170
6.4 Future Works	171
6.5 Conclusion	173
References.....	175
Appendix A.....	178
Appendix B	184
Appendix C	193
Appendix D.....	194

List of Figures

Figure 1. Kanban board (Kissflow, 2023)	4
Figure 2. Extreme Programming (XP) (GeeksforGeeks, 2024)	4
Figure 3. Final Year Project 1 Gantt Chart timeline	6
Figure 4. Final Year Project 2 Gantt Chart timeline	6
Figure 5. SmartGuide Logo (SmartGuide, 2024)	13
Figure 6. SmartGuide Language Selection Page	15
Figure 7. SmartGuide homepage showing content related to user's location	16
Figure 8. Relevant top tourist destination display based on selected country or region.....	16
Figure 9. Card displaying play audio feature with the tour itinerary.....	17
Figure 10. Map overview displaying current tour itinerary	17
Figure 11. User can add locations to Favourites list.....	17
Figure 12. Sarawak Travel mobile application logo	18
Figure 13. Sarawak Travel homepage	20
Figure 14. Sarawak Travel loading screen.....	20
Figure 15. “Nearby” function that display surrounding tourist attractions.....	21
Figure 16. List of cities/towns and map overview	21
Figure 17. Travel Malaysia mobile application logo in promotional poster.....	22
Figure 18. Travel Malaysia home screen attractions	24
Figure 19. Explore Points of Interest from the curated list of touris	24
Figure 20. User browsing point of interest can add to favourite list, get navigational direction and share to other platforms.....	25
Figure 21: Visual Studio Code allows customising your development environment with a vast array of extensions.	27
Figure 22: Landing page of the Flutter official website, highlighting its cross-platform development capabilities. (Google. (n.d.)).....	28
Figure 23: Official Dart website, showcasing key feature.....	30
Figure 24: The landing page of the Firebase Realtime Database website highlighting the real-time database feature.....	31
Figure 25: Official website of Google Maps platform.....	32
Figure 26. Kanban board.....	38
Figure 27. Contunuation of Kanban board	39
Figure 28. Respondent gender	40
Figure 29. Respondent age group distribution	40
Figure 30. Respondent nationality distribution.....	41
Figure 31. Occupation distribution of respondents.....	41
Figure 32. Education level distribution of respondents	42
Figure 33. Distribution of respondents based on familiarity with Kuching, Sarawak.....	43
Figure 34. Distribution of respondents based on familiarity with historical landmarks in Kuching, Sarawak	43
Figure 35. Frequency of respondents visits to historical sites	45
Figure 36. Preferences of respondents for modes of historical tours.....	46
Figure 37. Usage of mobile applications for tour guidance among respondents.....	47
Figure 38. Key features valued by respondents in a tour guide mobile application.....	48

Figure 39. Motivations of respondents for visiting historical sites.....	49
Figure 40. Challenges faced by respondents when visiting historical sites	50
Figure 41. . Usage of historical tour guide mobile apps among respondents	51
Figure 42. Mobile application features enhancing historical tour experience	51
Figure 43. Mobile application features enhancing historical tour experience	53
Figure 44. Importance of interactive maps for navigational directions during historical tours	54
Figure 45. Importance of detailed information in historical tour guide apps	55
Figure 46. Importance of quizzes and interactive content for engagement	56
Figure 47. Importance of chatbot feature in historical tour guide app among respondents.....	57
Figure 48. Importance of sharing news and promotions in a historical tour guide app.....	58
Figure 49. Additional feedback on a historical tour guide mobile application.....	59
Figure 50. Use case diagram of LelengauKch	60
Figure 51. Class diagram for LelengauKch	74
Figure 52. Database design for proposed project.....	75
Figure 53. High-level system architecture of proposed application	77
Figure 54. Activity diagram for LelengauKch.....	78
Figure 55. Sequence diagram for general user.....	79
Figure 56. Sequence diagram for admin	80
Figure 57. Visual Studio Code installing Flutter extension	90
Figure 58. Firebase Console to setup and configure project.....	92
Figure 59. Google Maps API component overview	93
Figure 60. Vertex AI Gemini API configuration	94
Figure 61. Public user interface	97
Figure 62. Admin interface	98
Figure 63. User account registration.....	100
Figure 64. Account login for users for LelengauKch	102
Figure 65. View historical site attraction list user interface	104
Figure 66. Search for historical site user interface	106
Figure 67. Locate and navigating to historical sites using quick action filter	108
Figure 68. User interface for chat module allows user to query details in conversational format	110
Figure 69. View list of news feed and explore other news.....	112
Figure 70. User interface for searching and viewing news.....	114
Figure 71. User interface demonstrating the workflow for sharing a news article to a third-party application.....	115
Figure 72. User participate in quiz module in LelengauKch	117
Figure 73. User can view quiz explanation and history	119
Figure 74. Features for Account Page.....	120
Figure 75. The "About Us" page accessible from the Account section provides users with information about the application and its developer.....	122
Figure 76. Sequence illustrating the newsletter subscription flow and event notification via email alerts.....	124
Figure 77. Workflow for admin adding a new historical site within the management module	126

Figure 78. Sequence illustrating admin editing an existing historical site within the application.	128
Figure 79. Workflow for admin deleting a historical site from the database	129
Figure 80. Workflow for admin adding a new news and events item within the application	130
Figure 81. Illustration of the news and events item editing and update workflow by admin	131
Figure 82. Depiction of the news and events item deletion process, including confirmation	132
Figure 83. Admin account interface showcasing verified login credentials and password recovery utilities.	133
Figure 84. Administrative user management workflow, demonstrating the processes of user account search, filtering by account status, status modification, and confirmation within the application.	134
Figure 85. User feedback on app installation and launch	159
Figure 86. User feedback on page navigation.	159
Figure 87. User feedback on historical sites information.	160
Figure 88. User feedback on accessing quiz module	160
Figure 89. User feedback on quiz questions	161
Figure 90. User feedback on quiz user experience	161
Figure 91. User feedback on navigation	162
Figure 92. User feedback on chatbot assistant.	162
Figure 93. User feedback on content of news and events section	163
Figure 94. User feedback on usage of News and Events section.	163
Figure 95. User feedback on bug experience during usage	164
Figure 96. User feedback on app experience	164
Figure 97. User feedback for features used or experienced.	165
Figure 98. User feedback on usefulness of chatbot assistant in responding to queries	165
Figure 99. . User feedback on overall satisfaction with the app	166
Figure 100. User feedback on recommending the application to other users	166

List of Tables

Table 1. Use case specification for register account.....	61
Table 2. Use case specification for login account.....	62
Table 3. Use case specification for manage historical site listings.....	63
Table 4. Use case specification for manage quiz.....	64
Table 5. Use case specification for manage news and events.....	65
Table 6. Use case specification for user to view and browse historical sites	66
Table 7. Use case specification for use navigation.....	68
Table 8. Use case specification for interact with chatbot	69
Table 9. Use case specification for get news and events	70
Table 10. Use case specification for participate in quiz	71
Table 11. Data Dictionary for User.....	76
Table 12. Data Dictionary for tourist.....	76
Table 13. Data Dictionary for admin	76
Table 14. Data Dictionary for historical site.....	76
Table 15. Data Dictionary for quiz	77
Table 16. Login interface of LelengauKch	82
Table 17. Sign up interface of LelengauKch	83
Table 18. Explore interface of LelengauKch.....	84
Table 19. View historical site details interface of LelengauKch	85
Table 20. Navigation interface of LelengauKch.....	86
Table 21. Simple chatbot interface of LelengauKch	87
Table 22. News and promotion interface of LelengauKch	88
Table 23. List of features under respective modules for users for LelengauKch	136
Table 24. List of features under respective modules for admin for LelengauKch	137
Table 25. User - Account Registration Test Cases	140
Table 26. User and Admin - Account Login Test Cases	142
Table 27. User - View Historical Sites Test Cases	144
Table 28. User – Access Map Navigation Test Cases	146
Table 29. User – Interact with Chat Assistant Test Cases	148
Table 30. User - Access News and Events Test Cases	150
Table 31. User - Access Quiz Test Cases	152
Table 32. System Testing Test Cases Summary (High-Level).....	156
Table 33. Project Objectives and Achievements	169

List of Abbreviations

Abbreviation	Word/phrase
XP	Extreme Programming
TDD	Test-Driven Development
POI	points of interest
WIP	work in progress
UML	Unified Modeling Language
UNIMAS	Universiti Malaysia Sarawak
CLI	Command-Line Interface
API	Application Programming Interface
CRUD	Create, Read, Update, and Delete
API	Application Programming Interface

Abstract

Kuching, the capital of Sarawak, Malaysia, is known for its rich cultural and historical heritage. Tourists, on the other hand, frequently struggle to navigate and fully appreciate the city's historical landmarks due to a lack of interactive and educational resources. To address this gap, this project introduces the "Kuching Historical Tour Guide Mobile Application (LelengauKch)", a digital solution designed to enhance the historical tourism experience. The mobile application, built with the Flutter framework and Dart programming language, includes features like location-based navigation powered by the Google Maps API, a chatbot for historical enquiries, interactive quizzes, and a platform for sharing news and events. These features are intended to provide an engaging, educational, and user-friendly interface for discovering Kuching's historical heritage. The project uses an Agile methodology, which combines Extreme Programming (XP) and Kanban practices, to ensure flexibility, continuous feedback, and high-quality output during development. This study conducts evaluation of 3 existing tour guide mobile applications to identify strengths and areas for improvement. Survey feedback helps to refine the application's features so that they meet user expectations. The application's usability and functionality have been evaluated by 31 respondents, with results indicating a positive and promising improvement in the overall user experience. Overall, "LelengauKch" aims to preserve Kuching's historical significance while promoting local tourism, bridging the gap between tradition and modern technology.

Abstrak

Kuching, ibu negeri Sarawak, Malaysia, terkenal dengan warisan budaya dan sejarah yang kaya. Walau bagaimanapun, pelancong sering menghadapi kesukaran untuk meneroka dan menghayati sepenuhnya mercu tanda sejarah bandar ini akibat kekurangan sumber interaktif dan pendidikan. Bagi menangani jurang ini, projek ini memperkenalkan "Aplikasi Mudah Alih Panduan Pelancongan Sejarah Kuching (LelengauKch)", iaitu penyelesaian digital yang direka untuk meningkatkan pengalaman pelancongan sejarah. Aplikasi mudah alih ini, dibangunkan menggunakan rangka kerja Flutter dan bahasa pengaturcaraan Dart, merangkumi ciri seperti navigasi berasaskan lokasi (dikuasakan *API Google Maps*), bot sembang untuk pertanyaan sejarah, kuiz interaktif, serta platform perkongsian berita dan acara. Ciri-ciri ini bertujuan menyediakan reka bentuk yang menarik, mendidik dan mesra pengguna bagi menerokai warisan sejarah Kuching. Projek ini menggunakan metodologi *Agile* yang menggabungkan amalan *Extreme Programming* (XP) dan *Kanban* bagi memastikan kelenturan, maklum balas berterusan serta hasil berkualiti tinggi semasa pembangunan. Kajian ini menilai 3 aplikasi panduan pelancongan sedia ada untuk mengenal pasti kekuatan dan ruang penambahbaikan. Maklum balas tinjauan membantu memperhalusi ciri aplikasi agar selari dengan jangkaan pengguna. Aplikasi ini telah dinilai dari segi kebolegunaan dan fungsi oleh 31 responden, dengan keputusan menunjukkan peningkatan pengalaman pengguna yang positif. Secara keseluruhan, LelengauKch bertujuan memelihara kepentingan sejarah Kuching sekaligus mempromosikan pelancongan tempatan, merapatkan jurang antara tradisi dan teknologi moden.

CHAPTER 1: INTRODUCTION

1.1 Introduction

Kuching, the capital of Sarawak, Malaysia, boasts a rich history and cultural heritage, attracting many tourists annually. However, visitors often find it challenging to navigate and appreciate these sites due to a lack of interactive, comprehensive, and easily accessible information. Traditional tour guides, while informative, may not always be available or affordable, and they can miss new openings or closures. This results in missed experiences for visitors (Braga, 2024). In the digital age, mobile applications can significantly improve tourism experiences by offering interactive and real-time information, making the experience more engaging and personalized. Liu (2020) found that digital display technologies positively impact heritage visitors by encouraging further education and exploration, enhancing visualization of heritage richness, and creating new ways to interact with history.

The proposed "Kuching Historical Tour Guide Mobile Application" named "LelengauKch", aims to provide a user-friendly, interactive platform for tourists to explore Kuching's historical landmarks. Leveraging smart technologies such as artificial intelligence, cloud computing, IoTs, and mobile communication, the app will serve as a digital tour guide, offering interactive information, location-based navigation, and engaging quizzes. This approach not only captures users' attention but also enhances overall engagement through real-time, personalized content. Cordis (2016) noted that utilizing mobile features like geolocation and real-time interaction leverages the potential of smartphones to deliver tailored cultural heritage content, facilitating more active cultural interaction.

1.2 Problem Statement

Tourists visiting Kuching often struggle to find interactive and engaging historical information about the city's landmarks, causing many to miss out on the rich stories behind these sites. This project seeks to address this gap by creating a mobile application that guides users through

historical sites and enhances their experience with interactive features, offering a more immersive historical tour experience.

Visitors to Kuching often face static and non-engaging information about historical landmarks, diminishing their overall experience and understanding of the city's heritage. Existing resources like guidebooks and static websites offer limited interactivity, leading to a less engaging experience. The lack of interactive content that encourages participation further exacerbates this issue. Additionally, traditional tour guides, while informative, are expensive and not always available, especially outside regular hours. This limits many visitors' accesses to valuable insights towards Kuching's historical landmarks, reducing their appreciation of the historical and cultural context.

Navigation difficulties also pose a significant challenge. Poor signage leaves tourists unsure of where to start their tours, and inconsistent directions between landmarks cause confusion. Consequently, tourists spend more time figuring out routes than enjoying the sites. The primary concern for individual tourists without guidance is the accessibility of these points, as noted by Mohareb et al. (2008). These navigation issues detract from the overall experience, making it hard for visitors to fully immerse themselves in Kuching's historical richness.

1.3 Objectives

The objectives of this project are:

1. To study and analyse existing mobile tour guide applications for Kuching historical sites.
2. To design and implement a Kuching Historical Tour Guide mobile application.
3. To test and evaluate the mobile application's usability and functionality to ensure it enhances Kuching's tourist experience.

1.4 Brief Methodology

The mobile application will be developed using an Agile approach that combines the Kanban method with Extreme Programming (XP) practices. Kanban is chosen for its flexibility in work

management, whereas XP improves code quality through defined practices. The combined methodology allows for the visualisation of work progress, ensuring that features are delivered incrementally.

Extreme Programming calls for five core values which are simplicity, communication, feedback, respect, and courage. Adherence to these values is crucial for a successful development process that serves the agile core values. XP practices including daily stand-ups, Test-Driven Development (TDD), simple design, small iterations and stakeholder review embrace the agile core values by promoting collaboration, delivering working software, engaging stakeholder, and supporting flexible responses to change while ensuring code quality and continuous improvement, while Kanban's emphasis on workflow management is useful as it reduces time spent on less important tasks and allows for a structured approach to meeting key milestones. The Kanban board incorporates TDD into its workflow by including phases such as Write Test, Develop Code, and Refactor ensuring that each task follows the TDD cycle for disciplined development. Using a Kanban board, such as Trello, can help to manage tasks by visualising workflow and tracking progress, columns are organised into "To Do," "In Progress," "Testing," and "Done" categories. The project will be developed in two iterations to progressively build, test, and refine the application, ensuring continuous improvement and agile responsiveness.

Starting each task with TDD ensures that unit tests are written before coding starts. Software engineers concentrate on writing code that can perform the required function because XP uses the test-driven development technique (TDD), which involves writing unit tests before the code itself. Each piece of code must pass the test in order to be released (Editor, 2021). As each piece of functionality is completed, taking the time to refactor the code will improve its structure while preserving behaviour, keeping the codebase clean and manageable.



Furthermore, focussing on small iterations is consistent with XP's iterative philosophy, allowing for continuous value delivery and incremental product improvement.

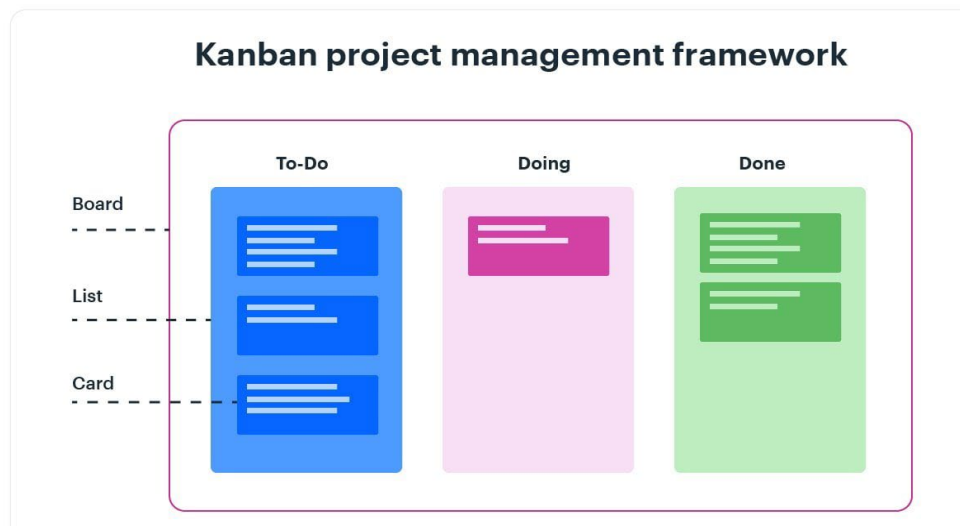


Figure 1. Kanban board (Kissflow, 2023)

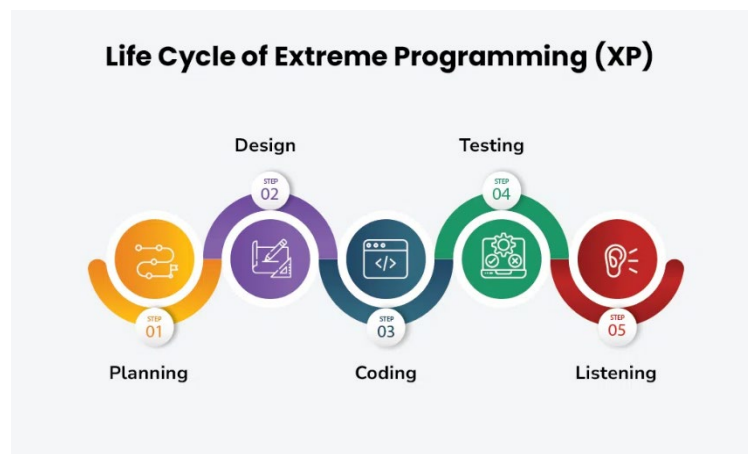


Figure 2. Extreme Programming (XP) (GeeksforGeeks, 2024)

To implement this approach, the workflow will be visualised on a Kanban board. This board will make the workflow more visible, assist in identifying bottlenecks, and improve communication. Limiting Work in Progress (WIP) will prevent overwork and ensure a smooth workflow by allowing one to reduce multitasking, and making bottlenecks easier to identify.

Combining Kanban's visual efficiency with XP's disciplined coding practices results in a productive workflow that maintains quality and adaptability throughout the project.

1.5 Scopes

The project is a mobile application called “LelengauKch”, which will provide an alternative in assisting tourists in exploring historical sites in Kuching, Malaysia. The primary users of this mobile application will be tourists wanting to visit historical landmarks in Kuching while seeking interactive information. The mobile application will include features such as location-based services, interactive quizzes, and chatbot features to provide an engaging and educational experience about Kuching's historical landmarks. For the application to function as intended, it will require an internet connection, and it will be developed using Flutter. Hence, the mobile application can act as a virtual tour guide, responding to user queries regarding the history of the historical sites or landmarks via the chatbot, provide engaging quizzes related to each site that are designed to enhance learning and retention for users, and include interactive maps that guide users to historical landmarks in Kuching, utilizing Global Positioning System (GPS) to provide real-time navigation.

1.6 Significance of Project

The project bridges gaps in accessibility, interactivity, and preservation of Kuching's historical heritage for tourists and locals. By incorporating features like chatbots, location-based navigation, and interactive quizzes, it allows users to explore Kuching's history at their own pace and convenience. This application preserves Kuching's heritage through modern technology, providing interactive knowledge of the city's historical sites and fostering a deeper connection to local culture. It promotes local tourism by encouraging independent exploration and boosting traffic to historical sites through interactive experiences hence, Kuching's history is more accessible and engaging. This approach fosters cultural appreciation and contributes to an interactive heritage tourism experience.

1.7 Project Schedule

The project will span over two semesters: Semester 1 (2024/2025) and Semester 2 (2024/2025).

1. Semester 1 (FYP1): October 7, 2024 to February 23, 2025.
2. Semester 2 (FYP2): February 17, 2025 to July 27, 2025.

The schedule of the final year project is as follows:

Figure 3. Final Year Project 1 Gantt Chart timeline

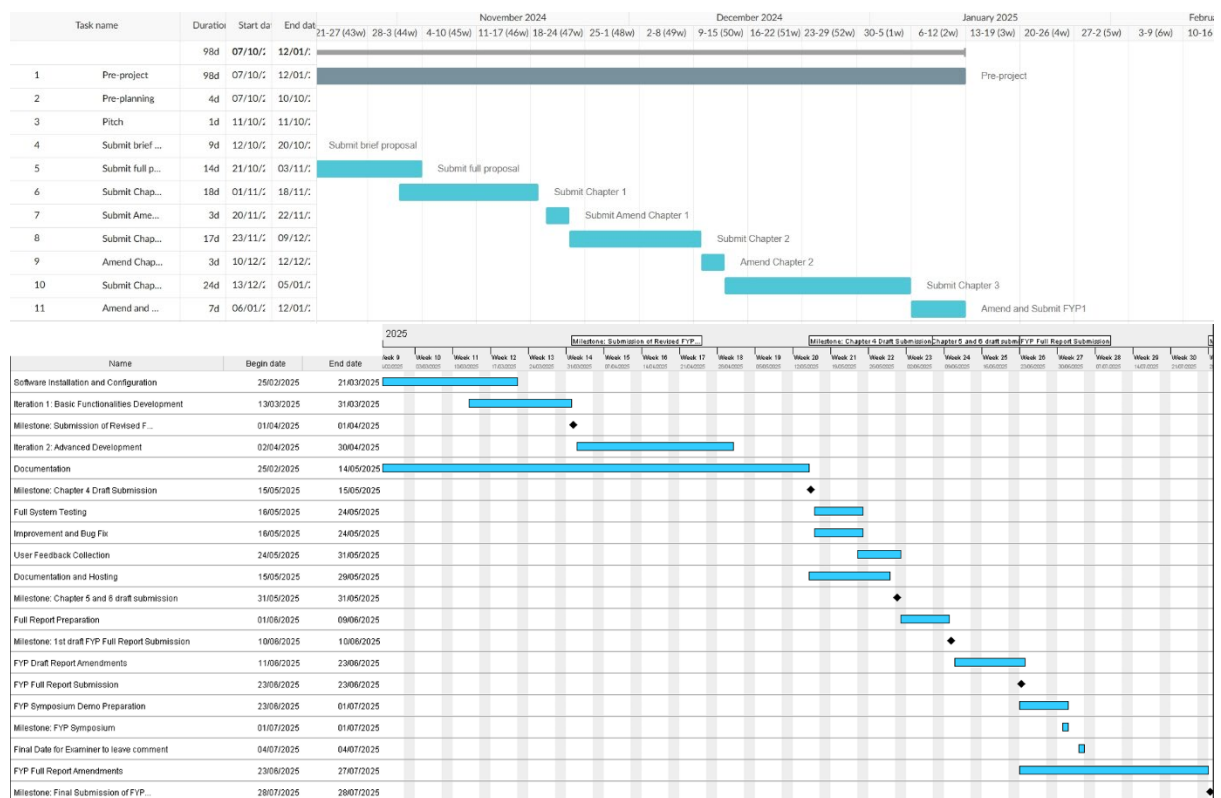


Figure 4. Final Year Project 2 Gantt Chart timeline

1.8 Expected Outcome

The expected outcome is a mobile application called "LelengauKch", designed to help tourists engage with Kuching's historical landmarks beyond static content. This app aims to provide an engaging and educational experience as users explore Malaysia's rich heritage. Key features include a chatbot for historical inquiries, location-based navigation, and interactive

quizzes. The chatbot offers real-time answers and context, enhancing users' understanding of landmarks, while location-based navigation directs users to significant sites for an interactive experience without physical guides as an alternative solution to the absence of a tour guide. In addition, the interactive quiz feature promotes engagement by allowing users to test their knowledge of Kuching's history, making learning enjoyable and memorable. Overall, the project aims to make historical exploration more accessible, interactive, and shareable for both locals and tourists.

1.9 Chapter Summary

1.9.1 Chapter 1 Introduction

The first chapter of this project discusses the significance of Kuching's historical heritage, as well as the challenges that tourists face in accessing interactive and engaging information about its landmarks. Traditional resources, such as guidebooks and static websites, lack interactivity, and tour guides may not always be available or affordable. Navigational challenges exacerbate the overall tourist experience, emphasising the need for a more innovative and immersive solution. The chapter outlines the project's primary objectives, which include researching existing mobile tour guide applications, designing and implementing a user-friendly app, and testing its usability and functionality.

The proposed mobile application, "LelengauKch," seeks to bridge the gap by offering interactive features like location-based navigation, chatbots for real-time queries, and engaging quizzes to improve the user experience. The chapter also emphasises the importance of this project in promoting cultural appreciation, preserving Kuching's historical heritage, and encouraging interactive tourism. The project schedule portrays the project timelines and important dates which serves as a guideline to complete the project within the schedule. By employing modern technology, the application provides tourists with a more engaging, accessible, and educational way to explore Kuching's historical richness.

1.9.2 Chapter 2 Literature Review

In Chapter 2, a literature review will be conducted to explore various aspects applicable in developing the mobile application 'LelengauKch.' This review will examine current historical tourism mobile applications to determine their benefits, drawbacks, and functional gaps. To assess the rationale behind this project and show how it enhances current solutions, a comparative analysis will also be carried out. In addition, this chapter will examine the technologies and tools required to develop the mobile application, with an emphasis on platforms based on Android. The suitability of tools, like Flutter for cross-platform compatibility and the Google Maps API, in accomplishing the project's goals will be evaluated. The theoretical underpinnings and technological insights needed to create an intuitive, interactive application specifically for Kuching's historical tourism are provided by this review of the literature.

1.9.3 Chapter 3 Requirement Analysis and Design

Chapter 3 describes the fundamental components required for developing the mobile application. The mobile application requirements are collected based on the methodology model chosen, and they will be analysed to ensure they align with the project objectives and user needs. The requirements are divided into two categories: functional requirements, such as the implementation of location-based navigation, simple chatbot functionality, interactive quizzes, and non-functional requirements, such as usability standards. To gather requirements for this project, a questionnaire will be distributed to tourists in Kuching.

This chapter also covers the use of diagrams and models to help visualize and organize the application's design using Unified Modeling Language (UML). Use case diagrams help to define user interactions and system features, whereas class diagrams depict the system's structure by detailing its classes, attributes, and relationships. Activity diagrams depict the application's workflow and processes, whereas sequence diagrams show the progression of

interactions between objects over time. These diagrams offer a comprehensive understanding of the system's design and functionality.

1.9.4 Chapter 4 Implementation

In Chapter 4, this chapter describes how the Kuching Historical Tour Guide Mobile Application will be implemented. The application's interface layout and structure, which were designed in Chapter 3, are realised through coding and the integration of various tools and frameworks. This chapter details the implementation process for each main feature. For example, the application uses location-based navigation via the Google Maps API to direct users to historical landmarks in Kuching. As a result, the steps for integrating the Google Maps API, such as creating API keys, customising map views, and adding location markers, are described in detail.

1.9.5 Chapter 5 Testing

Chapter 5 concludes the project by describing the comprehensive testing process for the Kuching Historical Tour Guide Mobile Application's functionality, usability, and efficiency. This chapter evaluates and tests each feature or module to ensure that it satisfies the requirements and design specifications outlined in Chapter 3. This includes thorough testing of critical features like location-based navigation, a chatbot for historical site queries, interactive quizzes, and social media integration. Test cases for each function are thoroughly documented, and they cover a wide range of scenarios to validate functionality, performance, and edge cases.

Furthermore, usability testing involves distributing questionnaires to gather user feedback on various aspects such as ease of navigation, information clarity, and overall user experience. This chapter also explains how the testing results were analysed, which provides insights into how well the application meets user needs and expectations. This analysis assesses the application's usability and draws conclusions about its effectiveness and user satisfaction. This extensive testing and analysis ensure that the application is dependable, usable, and functional.

1.9.6 Chapter 6 Conclusion and Future Work

Chapter 6 wraps up the Kuching Historical Tour Guide Mobile Application project by discussing its limitations, proposing future work, and summarising the overall results. This chapter discusses the limitations encountered during the development process, such as potential difficulties in maintaining the features. The chapter then outlines areas for future work, such as adding more languages, expanding the database to include more historical sites, and improving interactive features such as quizzes and adding social media sharing. Finally, the chapter provides a comprehensive project conclusion, reflecting on the objectives met, the effectiveness of the methodologies used, and the application's potential impact on promoting and preserving Kuching's historical heritage. It emphasises the project's importance in providing an informative and engaging tool for both locals and tourists, as well as its potential to serve as a foundation for future developments in similar applications.

CHAPTER 2: LITERATURE REVIEW

2.1 Overview

The purpose of this chapter is to provide an overview of the existing literature on the development of mobile applications for historical tour guides, focusing on the context of mobile tourism. According to Zarmou et al. (2013), particular emphasis has been devoted to creating mobile travel guides that help travellers plan their own tours or provide customised on-site tourism information about points of interest (POIs). The literature review seeks to examine and evaluate previous research, applications, technologies relevant to the development of "LelengauKch," a proposed mobile application designed to enhance the historical tour experience in Kuching.

Hence, Chapter 2 reviews three existing mobile applications that aim to improve the tour guide experience. Additionally, a comparison of three similar apps was conducted to assess their functionalities and advantages. The proposed mobile application will be developed by analysing the main functions of three apps in comparison. Other than that, a review of the selected software tools and development platform is also included in this chapter, along with the suggested features of the mobile application "LelengauKch," from which conclusions will be drawn after a comparison of the three current systems.

2.2 Background Study

Tourists visit attractions for a variety of reasons, including an interest in history and its impact on the present. Information technology has significantly developed within the travel and tourism sectors, enhancing tourist experiences (Pawar et al., 2016). Information limitations can be a major issue for tourist convenience, particularly when information should be easily obtainable and deliverable. According to Smirnov et al. (2013), the rapid advancement of information and communication technology has facilitated travellers' ability to plan their trips and access interesting information. Additionally, Yu (2014) reported that certain types of

tourism, particularly those involving smaller groups with specific interests and shorter stays, have increased, while mass tourism has decreased.

The advancement of information technology has resulted in a phenomenon that transformed how tourism services are delivered and consumed. Following the phenomenon of mobile technology, the nature of tourism is becoming more mobile, enabling tourists to obtain relevant information on the go using mobile phones. The term "mobile tourism" refers to the practice of using smartphones and other mobile devices to access information, navigate, book services, and manage travel plans while on vacation. Using their smartphones and tablets, travellers can now experience cities and tourist destinations thanks to a new travel trend called mobile tourism (Ojoo, 2017). Mobile tourism is growing due to mobile technology advancements that provide tourists with immediate access to location-specific and personalized information, thereby improving their convenience and travel experiences. Kenteris et al. (2007) indicate that a new trend in the travel industry is mobile tourism, which leverages mobile devices as electronic guides. Mobile tourism is gaining popularity because mobile apps within this sector are interactive, keeping tourists engaged and offering a more dynamic and personalized travel experience. This emphasizes the role of interactivity in driving the growth of mobile tourism.

2.3 Review of Existing Similar Applications

In this section, we examine three existing mobile applications which are SmartGuide, Sarawak Travel, and Travel Malaysia that share similarities with the proposed Kuching Historical Tour Guide Mobile Application ("LelengauKch"). These applications were selected based on their relevance to the project's goals and their functionalities that align with enhancing tourist experiences through digital means. Analysing these applications allows us to find best practices and potential areas for development, ensuring that our proposed project leverages effective

techniques and fills potential gaps to provide a more engaging historical digital tour guide experience for Kuching.

2.3.1 SmartGuide



SmartGuide

Figure 5. SmartGuide Logo (SmartGuide, 2024)

Figure 4 shows the logo of SmartGuide, which is a mobile application developed by the SmartGuide, Czech company that has developed tourist audio guides for railway operators such as the German Deutsche Bahn in 2016. The mobile application is available on the Google Play Store for Android smartphones, as well as the Apple App Store for iOS and iPadOS devices. SmartGuide is a mobile app that is free to download that offers several advanced features designed to enhance tourist's travel experience to their Point of Interest (POI).

Utilizing smart technology to turn users' smartphones into personal tour guides, the SmartGuide mobile application is an innovative platform that offers personalized and interactive tour guidance to improve the travel experience. It has many features that meet a variety of travel needs, including an audio guide that provides in-depth descriptions of tourist attractions, historical landmarks, and cultural sites, making the trip more enjoyable. Furthermore, based on Figure 4, there is a navigational icon at the centre of the logo, indicating guidance and direction. SmartGuide leverages geolocation services to provide real-time, location-based information. This feature ensures that users receive relevant content tailored to

their specific location, which improves their contextual understanding of the websites they visit.

Another notable feature is the offline mode, which allows tourists to access content without an internet connection, catering to those with limited connectivity. SmartGuide also includes interactive maps and itineraries to help users plan their trips more efficiently, as well as route and point-of-interest suggestions based on user preferences. The sharing feature also enables users to share their experiences and connect with other travellers, fostering a sense of connectivity. Furthermore, the app offers user-generated material, allowing travellers to share their own ideas and reviews, increasing the information pool with firsthand knowledge. SmartGuide's comprehensive and user-friendly design emphasises creating a seamless and enriching travel experience by merging current technologies with conventional tour assistance (SmartGuide, 2024).

In terms of usability, SmartGuide is consistent with several key principles from Nielsen's Usability Heuristics. The application ensures system status visibility by providing real-time navigation updates, and its intuitive interface and familiar language make it easy to use. Features like interactive maps, itinerary suggestions, and audio guides reduce cognitive load, allowing users to recognise information rather than recall it. Furthermore, the offline mode eliminates errors caused by connectivity issues, ensuring consistent access to content.

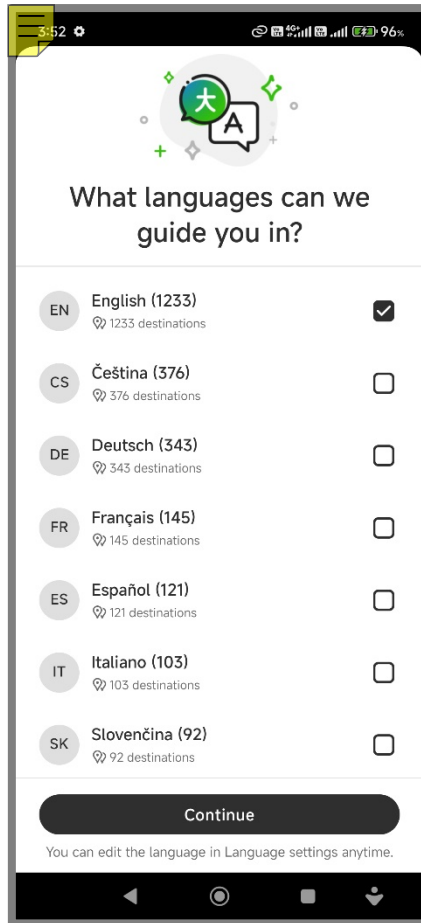


Figure 6. SmartGuide Language Selection Page

First, after installing SmartGuide, the mobile application allows users to select languages that are familiar to them as shown in Figure 5. The selected language also displays the number of tourist destinations available in the mobile app. After selecting a language, users can personalise their experience by logging in with their Google account. Next, the user can enable the geolocation feature, which allows the application to display nearby destinations around the current user, or the user can search for other locations. If the user selects to show nearby destinations, the home page will display relevant tourism information and destinations as a brief introduction or interaction based on the current user's location, as illustrated in Figure 7. The taskbar has four quick access sections: "Explore", "Map", "Favourites", and "Profile".

When the user taps on a destination card to learn more about the chosen tourist hotspot, the card expands to reveal additional cards displaying the top destinations available in the

selected region. From there, the user can browse the destination catalogue and finally select a self-guided tour or a paid-for tourism experience for the point of interest. The self-guided tour is free and includes an audio feature for users to listen to while following the tour, as well as a map showing the tour's travel distance as shown in the Figure 9.

Additionally, by clicking on the "love" icon, users can add specific locations to their "Favourites," which is a list of their current favourites. Users can also subscribe to the Premium plan to access extra features like ability to download offline guides and premium guides. Users can use the "Share feedback" feature under "Profile" to share feedback if they find a bug, have an improvement to suggest, or have questions.

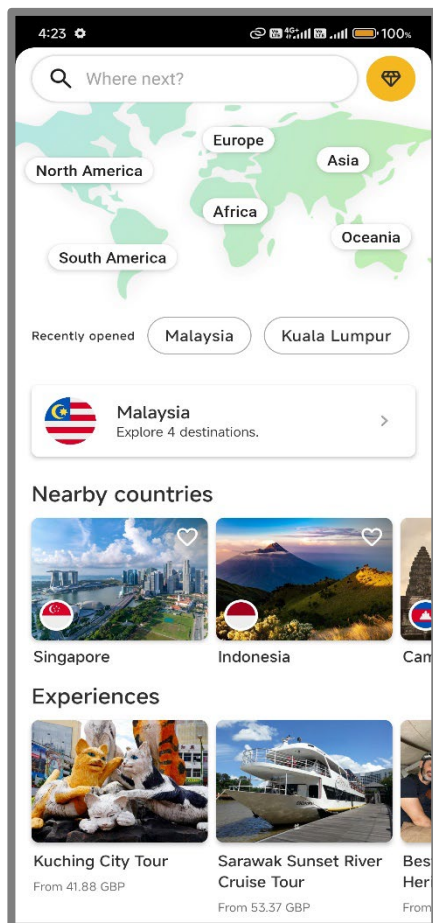


Figure 7. SmartGuide homepage showing content related to user's location

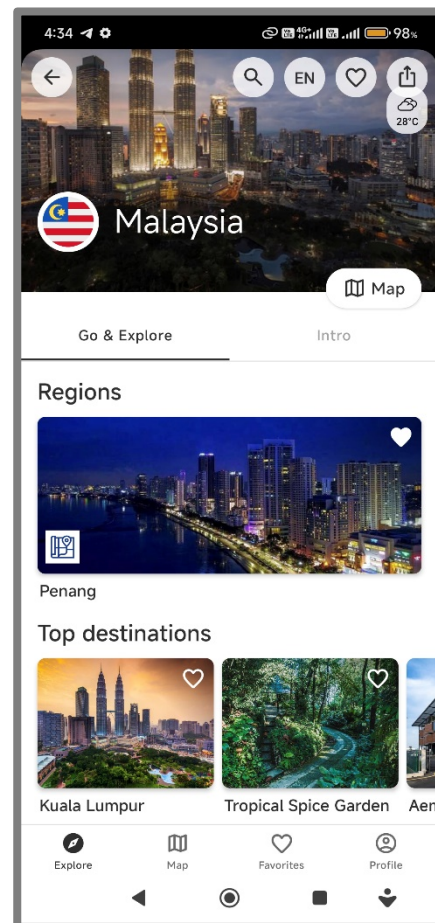


Figure 8. Relevant top tourist destination display based on selected country or region

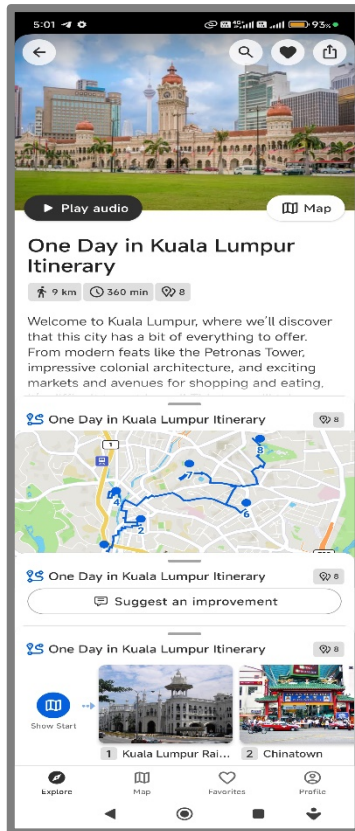


Figure 9. Card displaying play audio feature with the tour itinerary

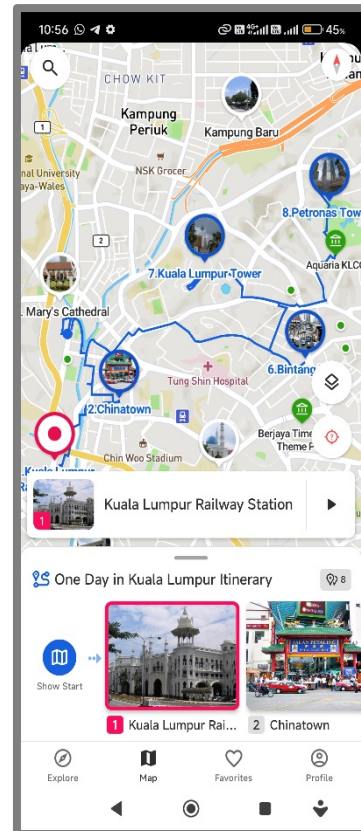


Figure 10. Map overview displaying current tour itinerary

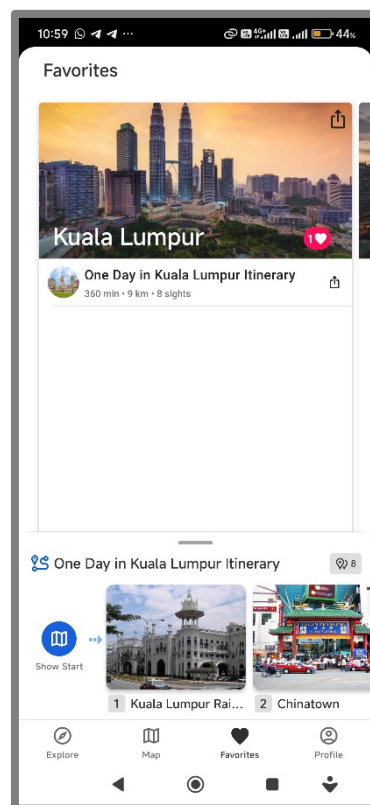


Figure 11. User can add locations to Favourite list

2.3.2 Sarawak Travel



Figure 12. Sarawak Travel mobile application logo

Sarawak Travel is a mobile application developed by the Sarawak Government and launched by the Sarawak Tourism Board (STB) in 2022. The mobile application is available on the Google Play Store for Android smartphones, as well as the Apple App Store for iOS and iPadOS devices. Sarawak Travel is a free mobile app that offers several advanced features designed to enhance tourists' travel experience in Sarawak. According to Sim (2022), to promote tourist destinations and travel-related goods throughout the Sarawak state, the Sarawak Tourism Board (STB), which is fully committed to the digitalization of the Sarawak tourism industry, unveiled the updated Sarawak Travel Portal and App whereby the mobile app is currently known as Sarawak Travel. This initiative fits seamlessly into the larger trend of mobile tourism, which uses technology to enrich and streamline the travel experience for users.

After user initially installed the mobile application, a home screen will be displayed with a bottom navigation bar with 5 modules including “Explore”, “Package”, “Home”, “Login”, and “More” as illustrated in Figure 15. Two functions which are “City” and “Explore” shall be highlighted. The "Home" module includes elements such as City and Heritage. Based on Figure 13, the "City" function offers users a full map overview of Sarawak. This static map has been enriched with icons and labels, each pointing to a major spot within the state. These markers indicate renowned tourist attractions, making it easy for users to find and learn about valuable sites.

Users can zoom in and out of the map, as well as pan across different places, to get a better look at specific regions. The map and list views are intended to be static, allowing users to view the map. Under the “Heritage” section, there are three items listed which are Old Kuching Heritage Buildings and Monuments, Kampung Heritage and Old Kuching Smart Heritage and each item aims to offer a thorough guide to the historical and cultural aspects of Sarawak. Overall, the function offers users with background information about heritage sites, allowing them to learn about their historical significance.

Another feature to highlight is the "Nearby" tab, which displays the user's present position as well as surrounding tourist spots in Sarawak as presented in Figure 12. The Sarawak Travel app's "Nearby" function provides users with a geolocation-based experience for discovering Sarawak's local attractions. This feature uses GPS data to display the user's nearest tourist attractions and provide crucial information about them, such as historical context, relevance, and driving instructions. Users can visit places like the Kuching Waterfront, the Chinese History Museum, the Brooke Gallery, and the Borneo Cultures Museum as they journey. The program aims to serve as a digital tour guide, making the travel experience more entertaining and educational.

 In terms of usability, Sarawak Travel follows Nielsen's Usability Heuristics. The clear bottom navigation bar and intuitively labelled modules, such as 'Explore' and 'Home', allow users to quickly perceive the system status and control their interaction with the app. The static map with icons and labels adheres to real-world conventions, reducing the user's cognitive load by emphasising recognition over recall. Zooming, panning, and the 'Nearby' geolocation function provide contextually relevant feedback, which improves the application's efficiency and error prevention.

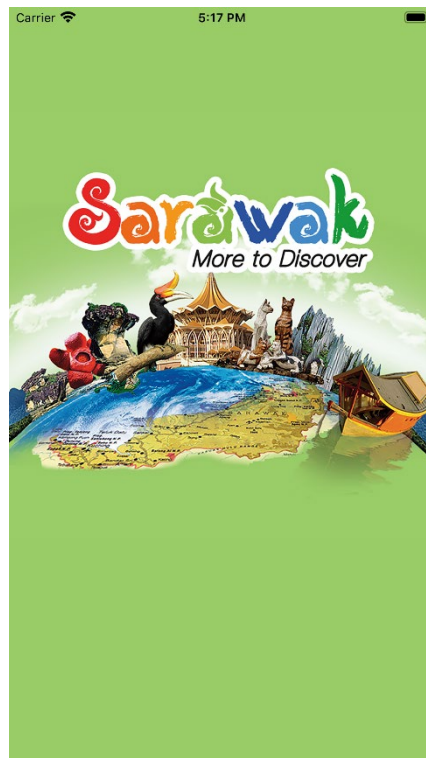


Figure 14. Sarawak Travel loading screen

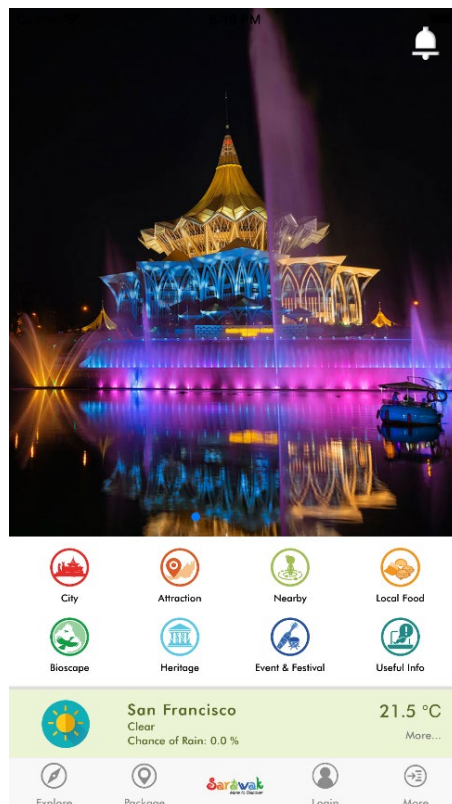


Figure 13. Sarawak Travel homepage



Figure 16. List of cities/towns and map overview

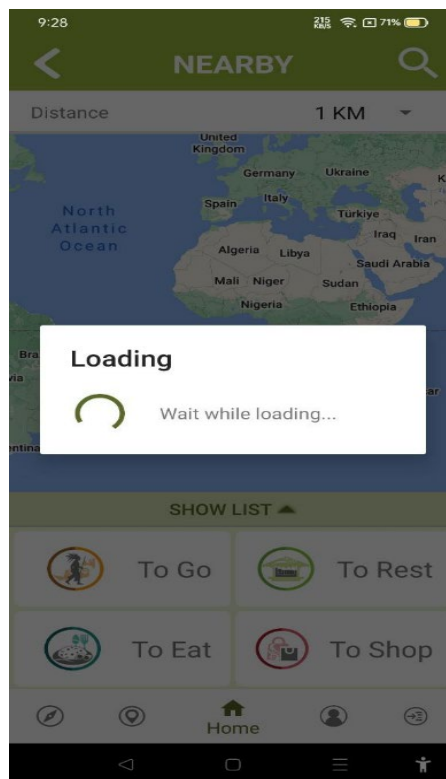


Figure 15. “Nearby” function that display surrounding tourist attractions

2.3.3 Travel Malaysia

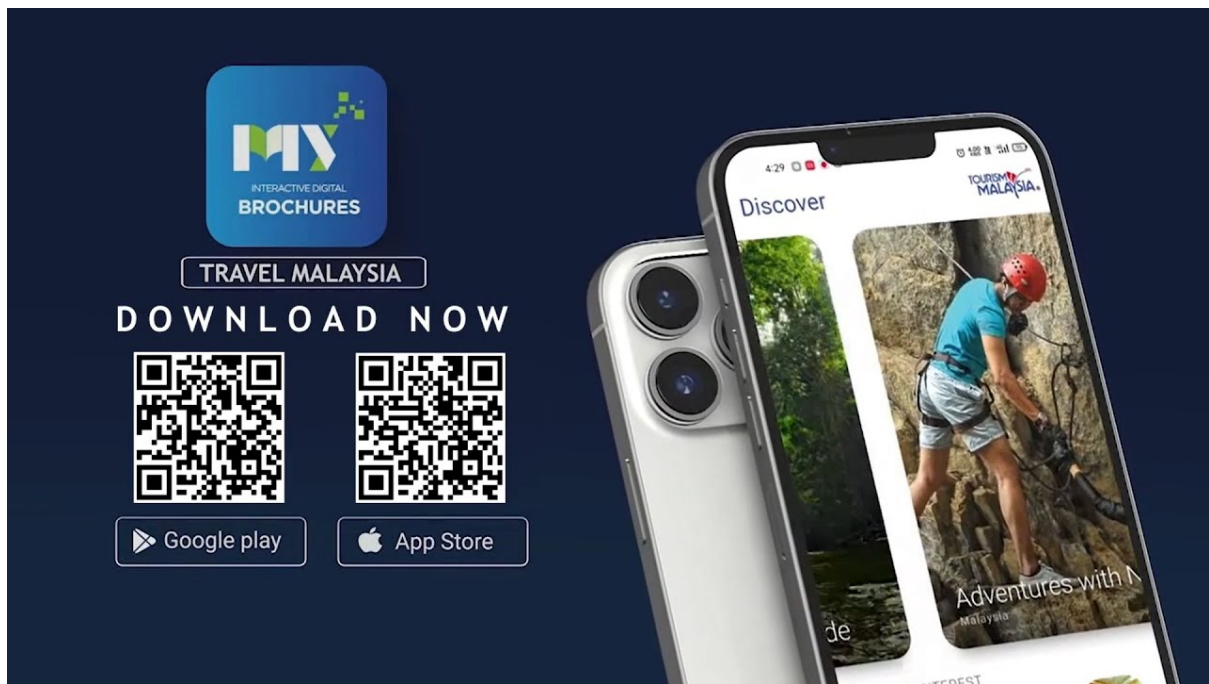


Figure 17. Travel Malaysia mobile application logo in promotional poster

The government organisation, Tourism Malaysia created the Travel Malaysia mobile application, which was made available in 2022. Both the Apple App Store for iOS and iPadOS devices and the Google Play Store for Android mobile devices offers the mobile application. On January 11, 2022, the mobile application was made available on the Google Play Store. In addition to improving the travel experience for visitors to the country, the mobile application seeks to deliver thorough and easily available information to make travel simpler and more pleasurable. The mobile application key features include curated lists of the best sights and activities, such as well-known monuments, beaches, and natural attractions. The app includes digital brochures such as the Malaysia Travel Guide, Island and Beaches, Destination Guide, and Adventure with Nature, which give extensive and diverse travel information.

After opening the mobile application, the user will be greeted with a home screen with a taskbar at the bottom displaying four quick access actions: "Home", "Explore", "What's New", and "Profile" as pictured in Figure 17. The "Home" page serves as an introductory portal to the

application, providing an overview of points of interest to the user. According to Figure 18, users can easily navigate the handpicked list of destinations which includes a filter and search  to help them find and filter relevant results. Referring to Figure 19, in the individual card showing the tourist destinations, each destination's image will be accompanied by text that provides information about the location, such as its history and significance. When the user clicks on the navigation icon, the app directs them to Google Maps, where the selected point of interest is designated as the destination, providing step-by-step directions and estimated travel times. It also keeps users up to date on major events and promotions in Malaysia, as well as providing navigation assistance such as directions and the best routes through cities.

The 10 general principles for interaction design proposed by Jakob Nielsen are referred to as "heuristics" because they are not precise usability guidelines but rather general rules of thumb (Nielsen, 2024). In terms of usability, Travel Malaysia applies several key heuristics from Jakob Nielsen's Usability Heuristics. It has system status visibility, which indicates that the application is connected to the Internet and operational, reduces user cognitive load through recognition-based navigation, and provides intuitive filtering for users when searching for tourist destinations. The search and filtering features increase efficiency, while the navigation integration provides easy user control. Error-prevention measures, such as appropriate messages for invalid searches, contribute to a positive user experience. Overall, it serves as a promotional tool, highlighting Malaysia's diverse attractions, cultural heritage, and events to a global audience, encouraging more tourists to visit various parts of the country.

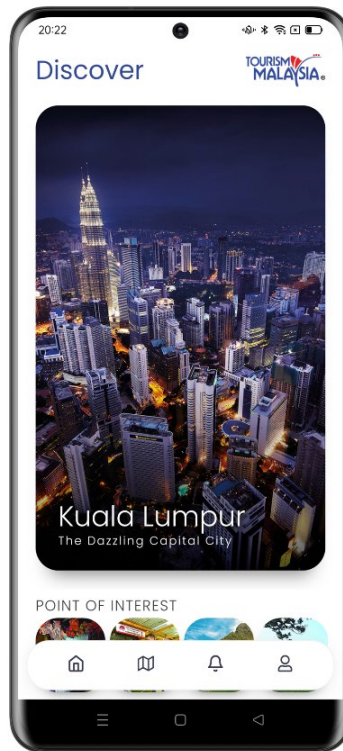


Figure 18. Travel Malaysia home screen attractions

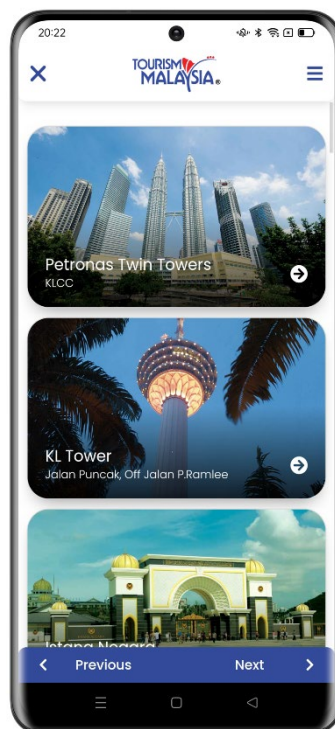


Figure 19. Explore Points of Interest from the curated list of touris

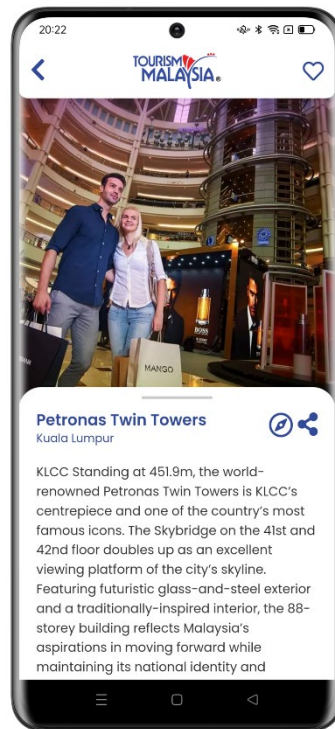


Figure 20. User browsing point of interest can add to favourite list, get navigational direction and share to other platforms.

2.4 Comparison of the Mobile Applications

	System			
	SmartGuide	Sarawak Travel	Travel Malaysia	Proposed Project
Platform	Android, iOS	Android, iOS	Android, iOS	Android
 Language	English, Chinese	English	English	English
Interactive Quiz	Absent	Absent	Absent	Present
Navigation	Present	Present	Present	Present
Chatbot Assistant	Absent	Absent	Absent	Present
Audio Tour	Present	Absent	Absent	Absent
Interactive Games	Absent	Present	Absent	Absent
Tourist Attraction List/ Historical Sites List	Present	Present	Present	Present
News and Promotion	Absent	Present	Present	Present

2.5 Review on Technology and Tools

2.5.1 Visual Studio Code

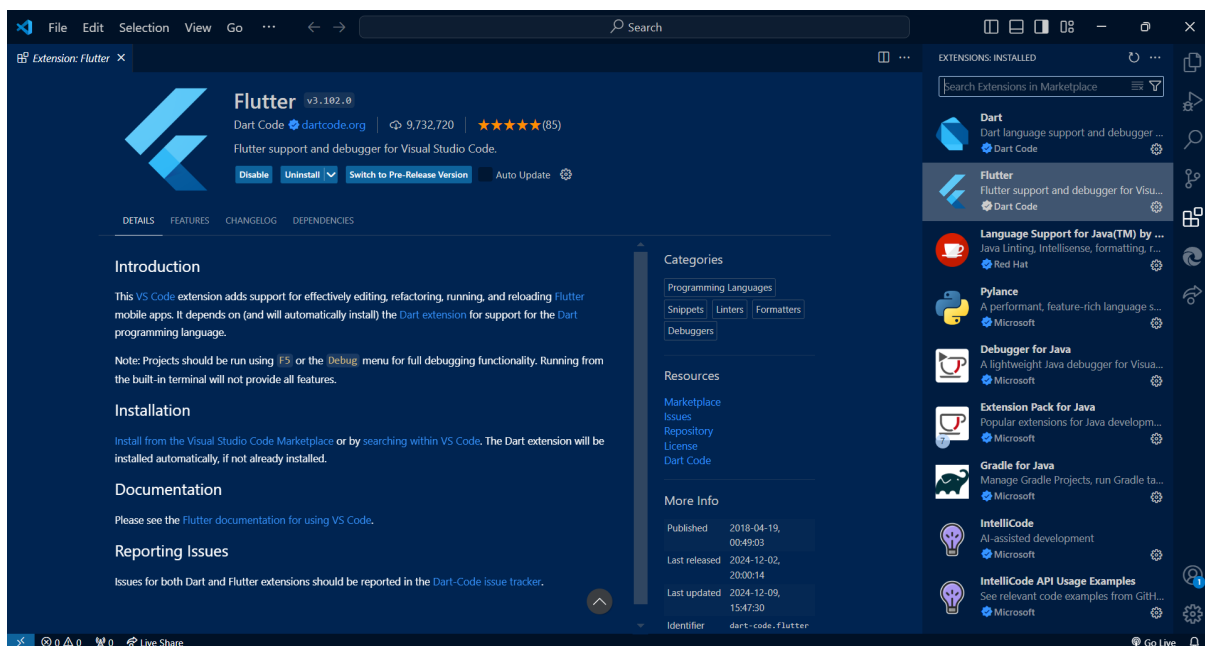


Figure 21: Visual Studio Code allows customising your development environment with a vast array of extensions.

Visual Studio Code (VS Code) is a well-known source-code editor produced by Microsoft, distinguished by its distinctive blue logo. VS Code, which was released in 2015, quickly gained traction among developers thanks to its lightweight yet strong capabilities. It is intended to be a versatile editor, allowing development in different programming languages and providing a variety of productivity-enhancing features. One of VS Code's main assets is its extensive support for extensions, which allows developers to tailor the editor to their own needs. These extensions address a variety of development topics, including language support, debugging tools, and version control integration.

VS Code is built on the Electron framework, allowing it to run on a variety of platforms, including Windows, macOS, and Linux. This cross-platform interoperability makes it an excellent alternative for a variety of development settings. The editor also has features like IntelliSense, which provides smart code completions based on variable types, function

definitions, and imported modules, and Git integration, which allows for seamless version control operations within the editor.

Furthermore, VS Code's user-friendly design and strong collection of built-in tools help to streamline the development workflow. Features such as an integrated terminal, task runner, and Docker support increase its utility in modern software development processes. VS Code also offers live sharing, which enables developers to collaborate in real time. This is especially useful for pair programming and distant team interactions. In conclusion, Visual Studio Code stands out as a strong, versatile, and user-friendly code editor that dramatically improves developer productivity and cooperation, making it a popular choice among many in the software development community.

2.2.5.2 Flutter

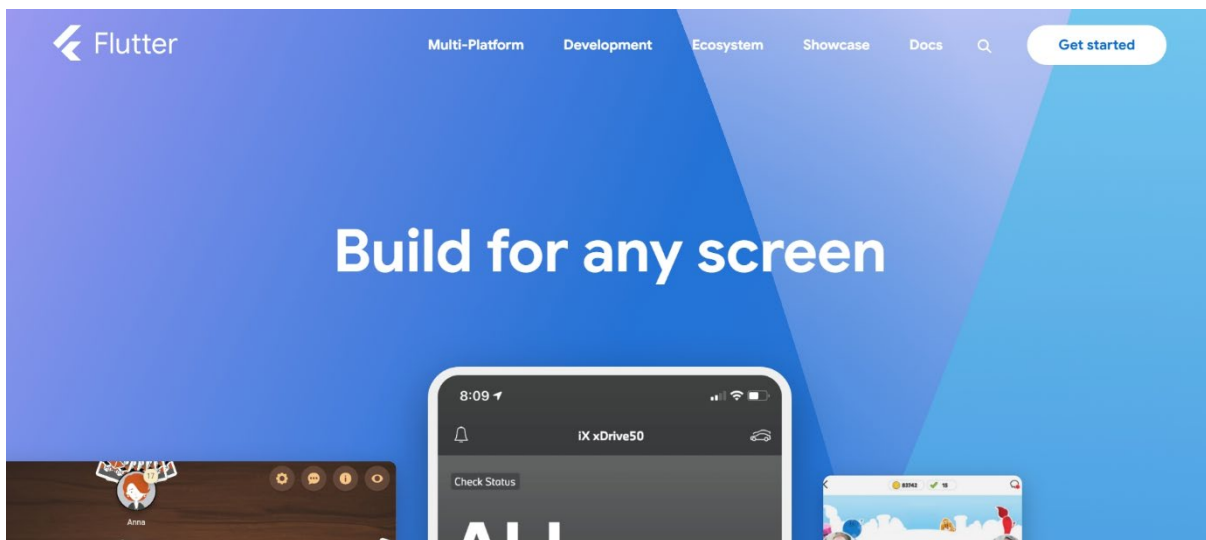


Figure 22: Landing page of the Flutter official website, highlighting its cross-platform development capabilities. (Google. (n.d.))

Flutter, an open-source framework developed by Google, facilitates the creation of natively compiled applications for mobile, web, desktop, and embedded devices from a single codebase. Flutter is distinguished by its high-performance capability achieved through ARM or Intel

machine code compilation, ensuring efficient execution and responsiveness. Its development experience is significantly enhanced by features like Hot Reload, which allows for real-time updates and rapid iteration. The framework's robust tooling, extensive libraries, and seamless integration with Google services all help to streamline the development process. Flutter is an excellent choice for multi-platform application development, with a global community and regular updates from Google. It saves significant time and resources during development. Flutter's first release was an alpha version, v0.0.6, on May 12, 2017. Flutter's stable version, 1.0, was released on December 4, 2018.

Flutter was chosen for mobile development due to several compelling advantages. For starters, Flutter's ability to support both Android and iOS platforms from a single codebase significantly reduces development and maintenance time, ensuring consistency across devices. Furthermore, with its extensive set of pre-designed widgets and rich libraries, developers can create visually appealing and highly interactive user interfaces. Flutter's integration with Firebase provides robust backend support, such as analytics, databases, and authentication. The thriving Flutter community and extensive documentation provide invaluable resources and support, allowing for rapid problem solving and continuous learning. As a result, Flutter's efficiency, versatility, and robust community support make it an excellent choice for modern mobile application development.

2.5.3 Dart

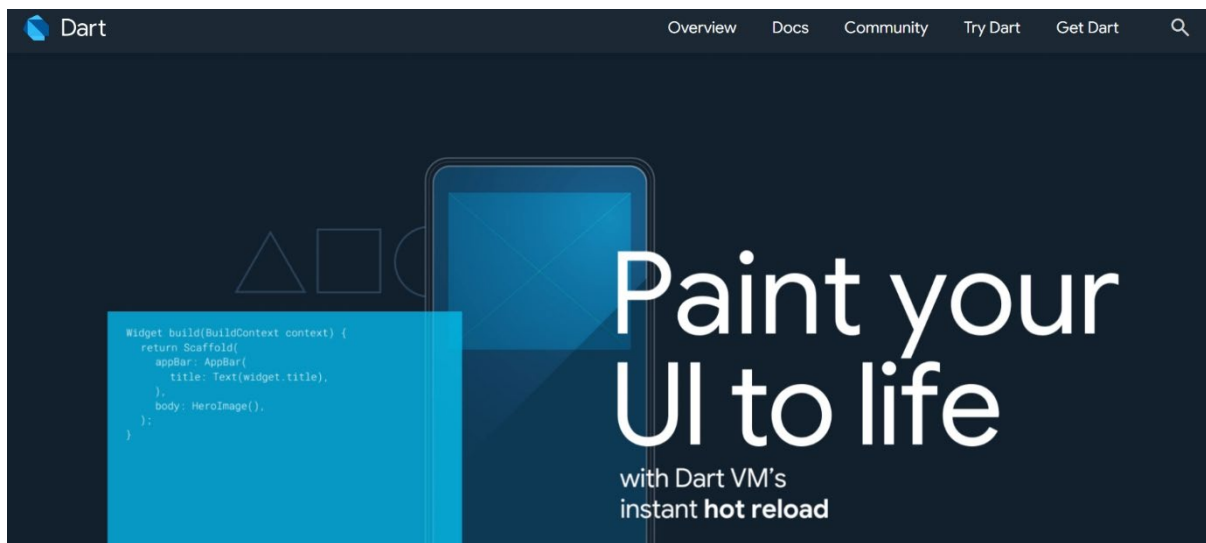


Figure 23: Official Dart website, showcasing key feature

Dart is the programming language that powers Flutter. It was also created and developed by Google. Designed to provide the most productive programming language for multi-platform development, along with a flexible execution runtime platform for app frameworks, Dart is a client-optimized language for creating quick apps on any platform (Google, n.d.-a). Dart is designed for speed and productivity, allowing developers to construct fast and dependable apps across multiple platforms. Dart's design prioritises ease of use and developer productivity, with a syntax that is simple to learn and read. It supports both just-in-time (JIT) and ahead-of-time (AOT) compilation, making it suitable for both the development and deployment stages.

Dart also works well with modern development tools and processes, providing extensive libraries and tools for simplifying typical programming chores. These include capabilities for code formatting, analysis, and testing, which aid in code quality and development workflow efficiency. The language and runtimes that drive Flutter apps are provided by Dart, which also serves as the foundation for Flutter. In addition, Dart facilitates a few fundamental developer tasks, such as formatting, code analysis, and testing. Overall, Dart's features and tools lay a

solid basis for constructing high-performance applications using Flutter, allowing developers to create robust and scalable software solutions quickly.

2.5.4 Firebase

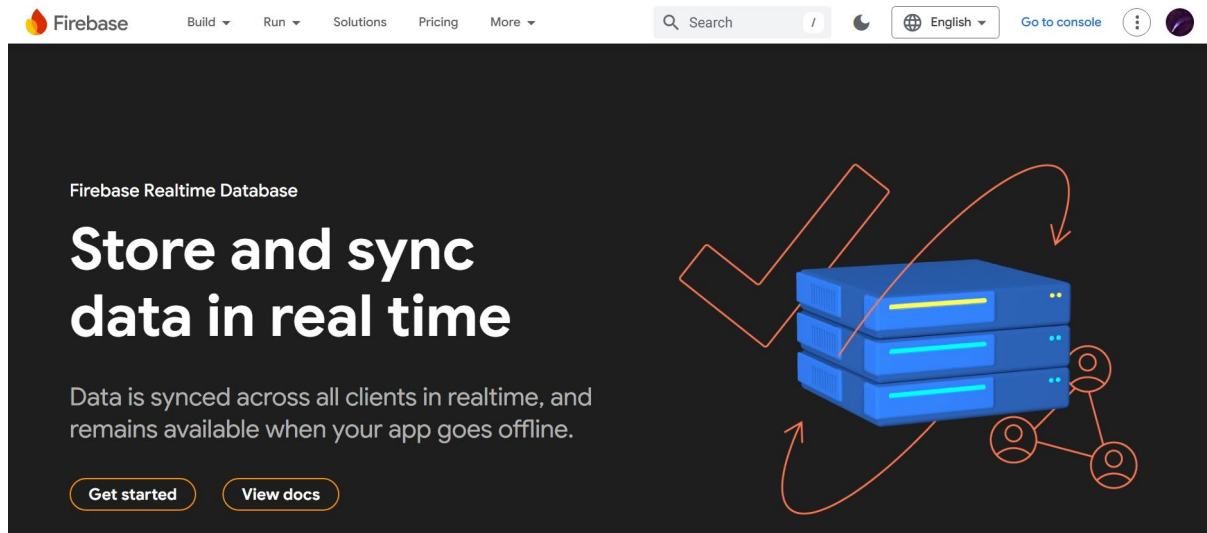


Figure 24: The landing page of the Firebase Realtime Database website highlighting the real-time database feature.

Firebase is a comprehensive platform created by Google to assist developers in creating and managing web and mobile applications. According to Google, it offers a suite of tools and services that streamline backend development, freeing developers to concentrate on creating compelling user experiences. Firebase is popular in the development of mobile apps, especially when combined with frameworks like Flutter, because it easily integrates with a variety of features like cloud storage, real-time databases, authentication, and more.

One of Firebase's core features is its Realtime Database and Cloud Firestore, which enable real-time data syncing between devices, making it ideal for apps that demand live updates, such as messaging or social media platforms. Firebase also offers Firebase Authentication, which allows you to easily create secure user authentication with different sign-in options (email, Google, Facebook, etc.). Additionally, Firebase Cloud Messaging supports push notifications,

allowing developers to engage users directly. Firebase Analytics provides detailed insights into user behaviour, allowing for data-driven decisions to improve app performance. Firebase Hosting also provides fast, secure, and scalable web hosting for static content.

When teamed with Flutter, Firebase provides a strong platform for developing feature-rich, scalable mobile applications. Its seamless connection with Flutter ensures a smooth development process with little configuration, making it an excellent alternative for developers looking to create high-quality mobile apps quickly. Firebase's cross-platform features also ensure that the same backend infrastructure can serve iOS and Android applications simultaneously.

2.5.5 Google Maps

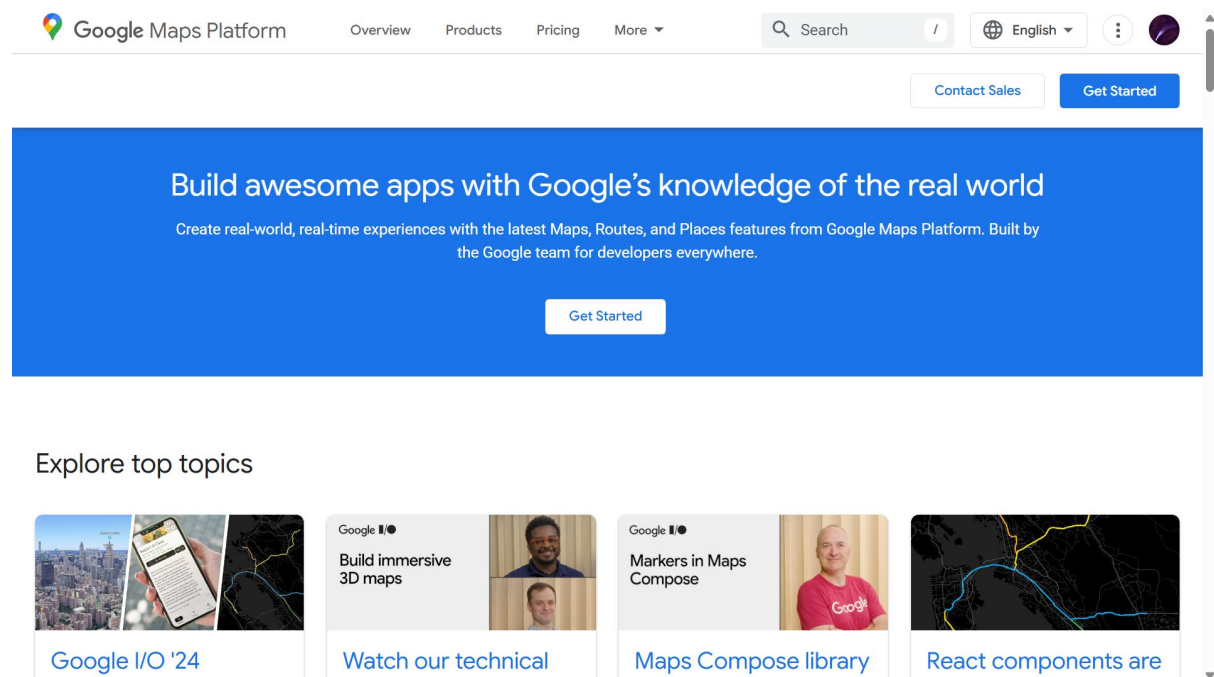


Figure 25: Official website of Google Maps platform

Google Maps is a comprehensive platform built by Google to let developers integrate geolocation and mapping services into web and mobile apps. It provides a set of tools and services that simplify the integration of maps, geocoding, and location-based features, allowing developers to focus on improving user experiences. Because of its seamless integration with

capabilities such as real-time navigation, place search, and route planning, the Google Maps API is highly popular in mobile app development, particularly when combined with frameworks like Flutter.

One of Google Maps' most important features is the Maps SDK, which allows developers to include customisable maps with a choice of functions and visual styles. The Geocoding API converts addresses into geographic coordinates and vice versa, whereas the Directions API offers full route planning across multiple modes of transportation. The Places API provides detailed information about millions of locations globally, including capabilities such as place search and autocomplete. Furthermore, the Distance Matrix API facilitates the calculation of trip time and distance for numerous origins and destinations, hence optimising logistics and planning.

When paired with Flutter, Google Maps provides a solid foundation to develop location-based mobile applications. Its seamless connection with Flutter provides an efficient development process with little configuration, making it an excellent alternative for developers who want to create high-quality, feature-rich mobile apps quickly. Google Maps' cross-platform features also ensure that the same geolocation services may be used across iOS and Android applications, resulting in a uniform user experience.

2.6 Hardware

Here are the recommended hardware requirements for running the latest version of a Flutter mobile application on Android device to ensure that users have a smooth and efficient experience with the mobile application.

For Android device:

- Operating System: Android 7.0 (Nougat) or higher.
- Processor: x86_64 or ARM64 processor or higher.

- RAM: At least 3 GB of RAM.
- Storage: Minimum 500 MB of free storage space for the app installation.
- Screen Resolution: HD (720p) or higher resolution display.
- GPU: Support for OpenGL ES 3.1 or higher.
- Internet Connectivity: Stable Wi-Fi or cellular data connection for app updates and real-time features.

2.7 Summary

This chapter examines existing literature relevant to the development of the "LelengauKch" mobile app, a Kuching Historical Tour Guide. It begins by investigating the changing role of information technology in the tourism industry, emphasising the shift towards mobile tourism due to advances in mobile technologies. The concept of mobile tourism is discussed, focussing on the convenience and enhanced travel experiences provided by mobile applications. The review then focusses on three current mobile applications: Sarawak Travel, Visit Kuching, and Explore Malaysia. Through this comparison, the chapter identifies key strengths and gaps in these applications, laying the groundwork for the proposed "LelengauKch" app. The proposed app intends to integrate immersive technologies, navigational features, and interactive elements to create an interactive and engaging historical tour guide for Kuching.

CHAPTER 3: METHODOLOGY

3.1 Introduction

This chapter will focus primarily on the analysis and design of requirements for the proposed project. The proposed application was developed using a combination of Kanban and Extreme Programming (XP) methodologies. All requirements for the proposed application will be listed and discussed in detail. Finally, this chapter will go over the design phase of the proposed application. The design scope includes user interface, database, and application architecture.

3.2 Methodology

This project will be completed using an Agile approach, combining the Kanban method with XP practices. Extreme Programming, an iterative software development methodology, that allows the production of high-quality software that meets user requirements through iterative software development.

3.2.1 Extreme Programming (XP)

Extreme Programming (XP) involves five main phases which are planning, designing, coding, testing, and listening. This chapter will further elaborate on the first phase of XP, which is planning, as well as discuss the second phase, which is designing. Certain aspects of the planning phase, such as the project completion schedule, have already been addressed in Chapter 1, while parts of the design phase were discussed in Chapter 2. In this chapter, the focus will shift to another aspect of the planning phase, specifically requirement analysis. Additionally, the second phase of XP, the design phase, will also be explored. The method employed for the design phase is Unified Modeling Language (UML), a visual modelling language used in object-oriented software development. Furthermore, Chapter 2 addressed another component of the design phase, namely the identification of the most critical features to be included in the proposed application. Meanwhile, the remaining three phases of XP which are coding, testing, and listening will be covered in Chapter 4.

3.2.1.1 Planning

The Extreme Programming (XP) methodology's first phase is planning. This includes detailing the initial setup and implementation needed to finish the project, as well as identifying the tasks required to guarantee the program meets the specified quality requirements. This phase also includes the project's requirement analysis. To acquire these needs, the target users are consulted, and their feedback is obtained. The suggested application's needs were gathered from the intended consumers via an online questionnaire. Several respondents were interviewed to get feedback on the proposed application. Furthermore, the survey responses will be evaluated, and all of the obtained data will be used to create a list of specifications for the proposed project.

3.2.1.2 Design

The second phase of the Extreme Programming (XP) methodology is the design phase. During this stage, the primary features that must be included in the software are designed. The design is developed based on the requirements gathered during the planning phase. This marks the beginning of the iterative process, where the design is refined continuously until an optimal version is achieved. This approach helps minimise potential issues during the coding phase. For the proposed application, the design of the features has been discussed in Chapter 2, drawing on the review of existing systems. The application design for the proposed project will utilise the Unified Modeling Language (UML). Various logical and UML diagrams will be employed, including use case diagrams, activity diagrams, sequence diagrams, and class diagrams. Each diagram will serve a distinct design purpose.

3.2.1.3 Coding

The third phase of the Extreme Programming (XP) methodology is the coding phase, a crucial step in the process as it involves transforming the requirements and designs into a functional software product. This phase is carried out iteratively to ensure the code remains as

simple as possible while retaining its functionality. Key tasks in this phase include programming the software, reviewing the code, and refactoring it. Refactoring involves restructuring the code to ensure it effectively delivers the features and meets the requirements outlined in the first two phases. Furthermore, the features of the proposed application will be broken down into smaller components, with each component to be completed within a specific timeframe. Consequently, each component will involve a distinct coding process, and various features of the proposed application will be developed during this phase.

3.2.1.4 Testing

The fourth phase of the Extreme Programming (XP) methodology is the testing phase. In this phase, testing is conducted concurrently with each coding process rather than waiting until the software is fully developed. This approach integrates testing as a core part of the software development process. Each part of the code is tested to ensure the absence of bugs and issues in the final product. Additionally, this phase involves users as part of the development team by conducting user acceptance testing, where features developed during the coding phase are tested with them. For the proposed application, each completed coding segment will be tested with the target users to gather their feedback on whether the produced feature meets their expectations. The target user, primarily tourists, will play a key role in completing this phase.

3.2.1.5 Listening

Listening is the final element of the Extreme Programming (XP) approach. The most significant aspect of this phase involves feedback from the target users. This is done to guarantee that the end result of the proposed application satisfies the needs of the end users. The users' feedback will be utilised to revise the requirements of the proposed application. If any changes are required, an iteration of design, coding, testing, and listening will be performed. If no adjustments are necessary from the users, the iteration will conclude at this stage.

3.2.2 Kanban

The Kanban method is a visual project management system that employs a board to represent tasks and their status, giving a simple and intuitive way to monitor progress. Tasks are organised into columns that indicate distinct stages of development, such as backlog, to do, doing, and completed. This method prioritises continuous delivery, adaptability, and efficiency while reducing work in progress (WIP). The Kanban approach is very useful for this project since it provides task organisation while also fostering openness and accountability. It enables the development member to dynamically prioritise work and respond to changes or issues that arise during the development process. By visualising the workflow, the Kanban approach aids in monitoring progress and identifying bottlenecks, ensuring that the proposed application is structured and efficient. Kanban board usage for this project is illustrated in Figure 25 and 26.

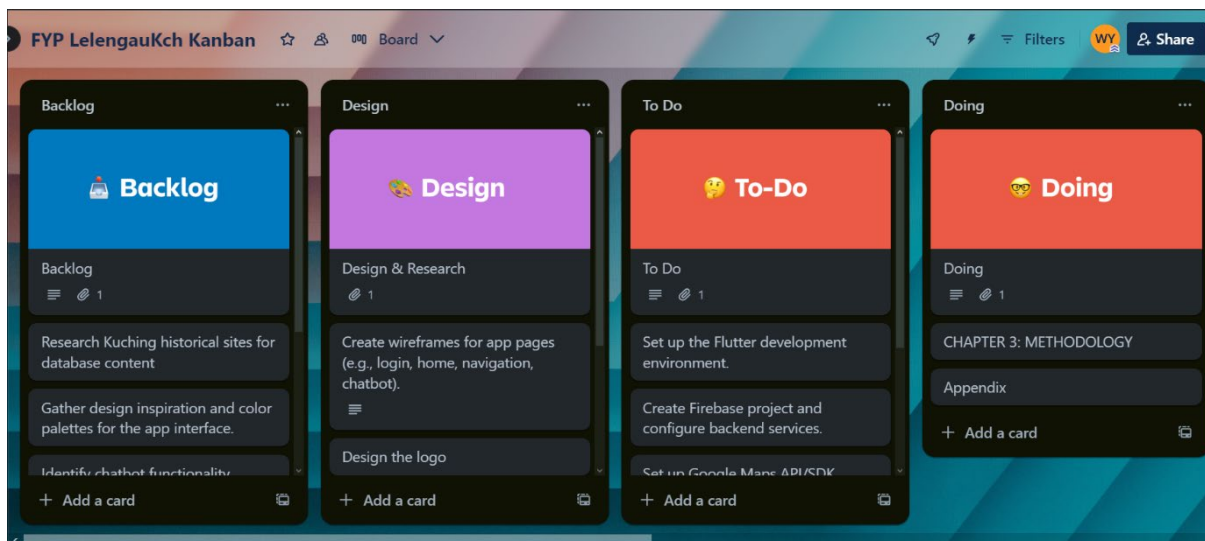


Figure 26. Kanban board

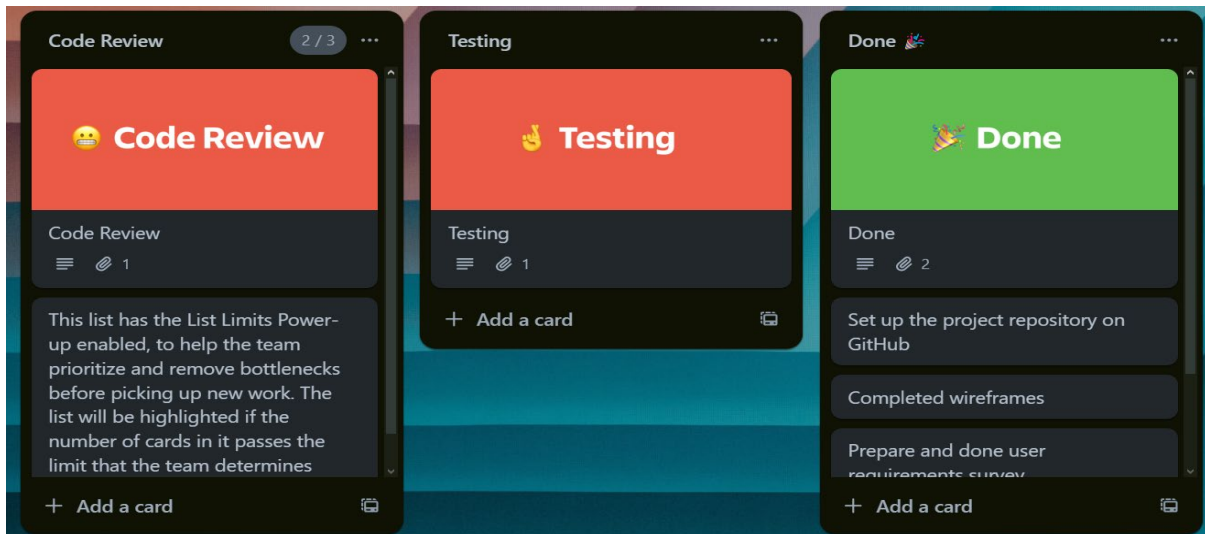


Figure 27. Continuation of Kanban board

3.3 Requirement Analysis

The requirements for the proposed application were determined using a structured approach, which included collecting user feedback via an online questionnaire. This resulted in a clear understanding of user needs and expectations, laying the groundwork for the application's development. The results of this process are detailed in the following sections.

3.3.1 User Requirement

To gather user requirements, an online questionnaire was conducted to obtain respondents' views and opinions regarding the proposed application. Google Forms was utilised as the platform to collect and organise feedback efficiently. This approach ensured a broad reach, allowing a diverse range of participants to contribute valuable insights for shaping the application's development. There are a total number of 50 respondents who answered the online survey and the results from the questionnaires will be visualised in this section.

Gender
50 responses

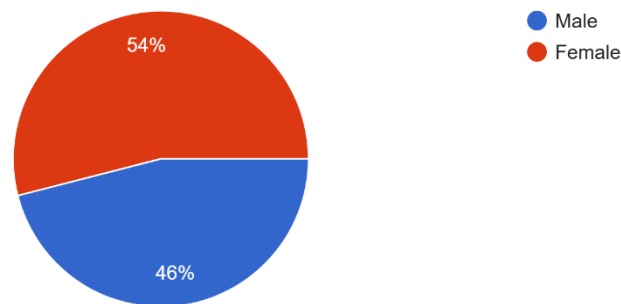


Figure 28. Respondent gender

According to Figure 28, 54% of respondents is male while the balance of 46% is female respondent.

Age Group
50 responses

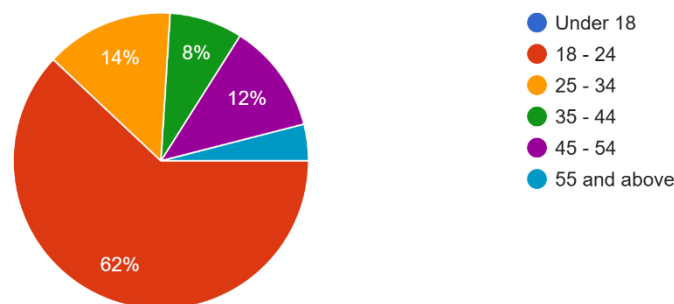


Figure 29. Respondent age group distribution

According to Figure 29, the majority of respondents fall within the age group of 18–24 years, comprising 62% of the total responses. The second largest group is 25–34 years, representing 14% of the respondents. Other age groups include 35–44 years (8%), 45–54 years (12%), and 55 and above (4%), which account for smaller proportions. No respondents are in the under 18 categories, indicating that the primary target audience is older individuals.

Nationality
50 responses

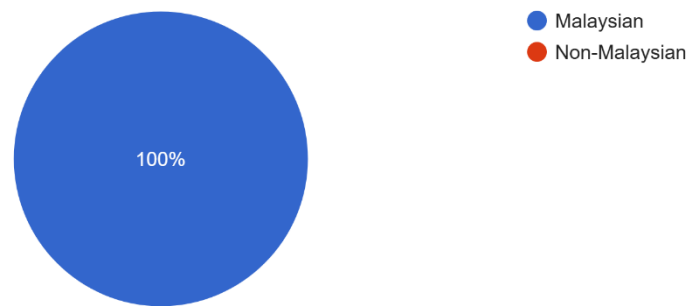


Figure 30. Respondent nationality distribution

The pie chart (Figure 29) shows the nationality distribution of the respondents. All 50 respondents, accounting for 100% of the total, are Malaysians. There are no responses from non-Malaysians, indicating that the survey represents the perspectives of Malaysian individuals.

Occupation
50 responses

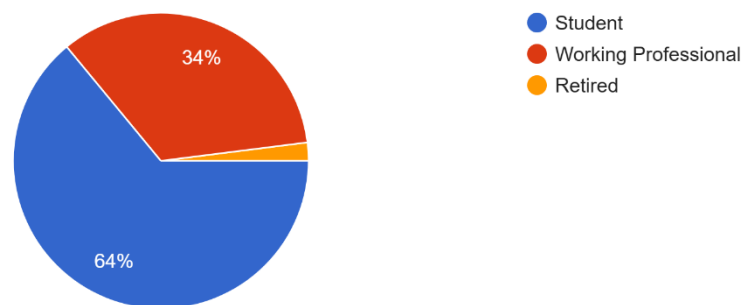


Figure 31. Occupation distribution of respondents

Figure 30's pie chart depicts the respondents' occupational distribution. According to the data, the majority of respondents are students, accounting for 64% of the total. Working professionals account for 34%, while retired individuals make up only 2%. In essence, the chart

shows that the sample consists primarily of students, with a significant presence of working professionals and a small number of retired individuals.

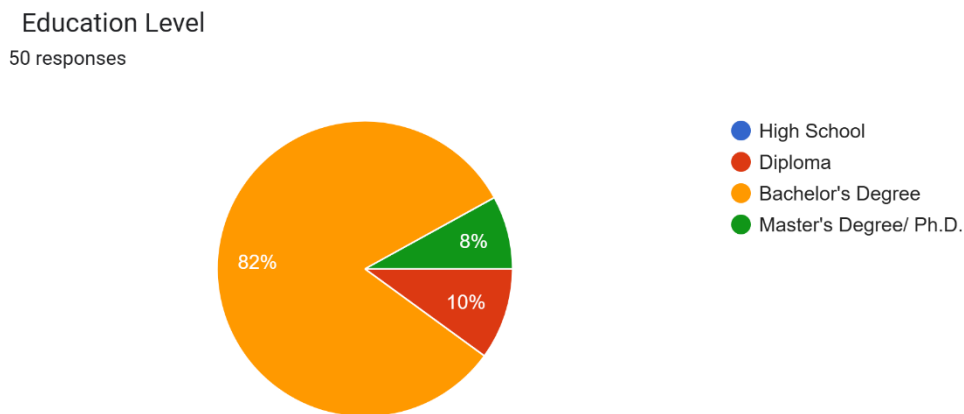


Figure 32. Education level distribution of respondents

Figure 31 portrays the distribution of educational levels among respondents. According to the data, the majority of respondents have a bachelor's degree, accounting for an impressive 82% of the total. Diploma holders make up 10% of the sample, while those with a Master's or Ph.D. account for 8%. In essence, the chart demonstrates that the sample is predominantly composed of individuals with bachelor's degrees, with a significant presence of Diploma holders and a smaller proportion of respondents with higher postgraduate degrees.

General Familiarity with Kuching, Sarawak
50 responses

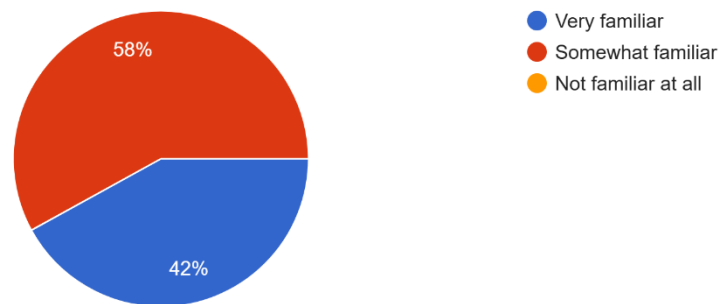


Figure 33. Distribution of respondents based on familiarity with Kuching, Sarawak

According to Figure 32, the level of familiarity respondents have with Kuching, Sarawak is illustrated. The data reveals that a significant proportion of respondents, 58%, indicated being "Very familiar" with the city, while 42% reported being "Somewhat familiar." Notably, no respondents claimed to be "Not familiar at all." This suggests a strong level of awareness and recognition of Kuching among the surveyed population.

Familiarity with Historical Landmarks in Kuching, Sarawak
50 responses

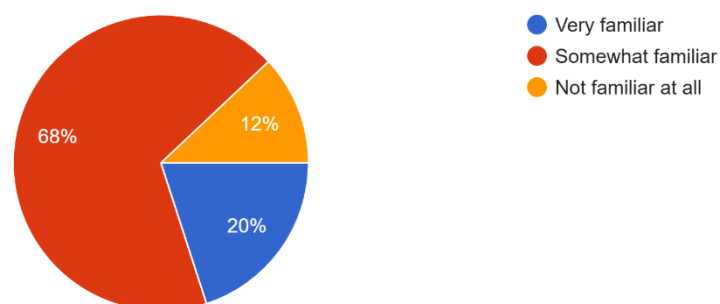


Figure 34. Distribution of respondents based on familiarity with historical landmarks in Kuching, Sarawak

Figure 33 shows a pie chart of 50 respondents' familiarity with historical landmarks in Kuching, Sarawak. The chart is divided into three segments, each representing a different degree of familiarity. The largest segment, in red, indicates that 68% of respondents are "somewhat familiar" with historical landmarks. The blue segment indicates that 20% of respondents are "very familiar" with these landmarks. The smallest segment, shown in orange, represents 12% of respondents who are "not familiar at all" with historical landmarks in Kuching, Sarawak. This data is relevant because it provides insights into the awareness and knowledge of historical landmarks among the population, implying that the surveyed population is moderately aware of and recognises Kuching's historical landmarks.

How often do you travel to historical sites?
50 responses

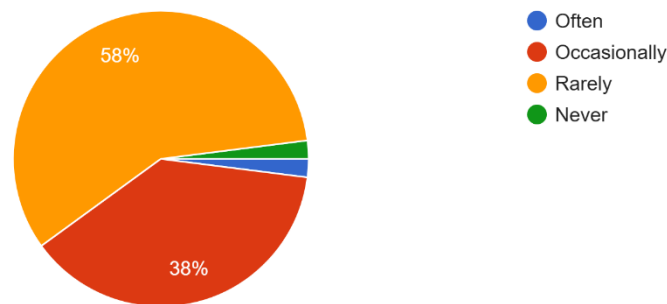


Figure 35. Frequency of respondents visits to historical sites

Figure 34 illustrates how frequently individuals visit historical sites, based on responses from a survey of 50 participants. The data is divided into four categories: "Often," "Occasionally," "Rarely," and "Never," each depicted in a distinct colour on the pie chart. The majority of respondents, represented by the orange segment, reported visiting historical sites rarely, accounting for 58% of the total. Occasional visits, shown in red, make up 38%, indicating a moderate level of interest among this group. Those who frequently visit historical sites are represented by a small blue segment, comprising just 2% of the respondents. Similarly, 2% of participants, shown in green, stated that they never visit historical sites. This distribution highlights varying levels of interest and engagement with historical sites, offering useful insights into the general attitudes of the surveyed group towards historical exploration.

What mode of historical tours do you prefer?

50 responses

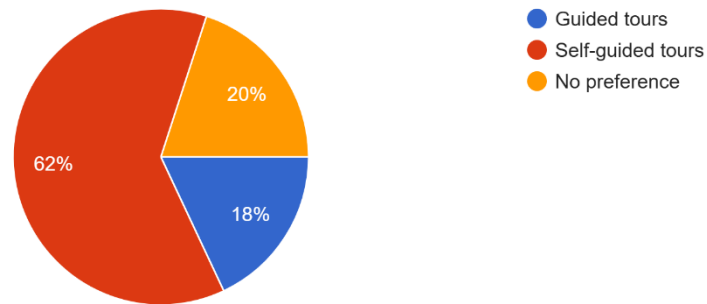


Figure 36. Preferences of respondents for modes of historical tours

The pie chart presented in Figure 35 depicts the preferences of respondents regarding different modes of historical tours. A total of 50 responses were collected for this survey. The chart is divided into three segments, corresponding to guided tours, self-guided tours, and no preference. Self-guided tours, highlighted in red, are the most popular choice, with 62% of respondents indicating their preference for exploring historical sites independently. This demonstrates a clear majority favouring self-directed exploration. In contrast, guided tours, represented in blue, are chosen by 18% of respondents, suggesting that fewer participants prefer a guided experience with explanations and context provided during the tour. Meanwhile, the orange segment, which accounts for 20% of respondents, reflects those who have no specific preference for either mode of touring. These findings offer valuable insights into how individuals prefer to engage with historical sites.

Do you use mobile application for guiding the tour?
50 responses

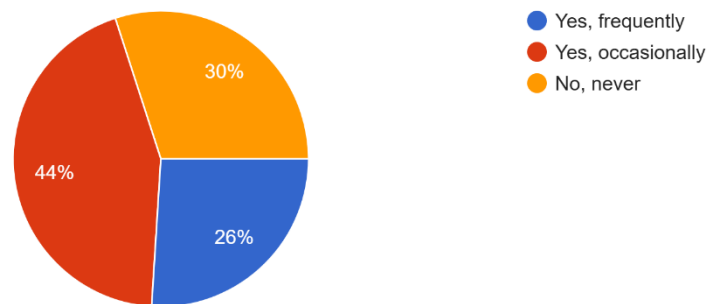


Figure 37. Usage of mobile applications for tour guidance among respondents

Figure 36 illustrates the frequency at which mobile applications are used for tour guidance, based on a survey of 50 participants. The data is represented in three segments, with each segment showing a different level of usage. The red segment, which makes up 44 percent of the chart, represents respondents who occasionally use mobile applications for tour guidance. Following this, the orange segment shows that 30 percent of participants reported never using such applications. The blue segment, comprising 26 percent of the chart, reflects respondents who frequently rely on mobile applications during their tours. These findings highlight varying degrees of dependence on mobile technology for tours, with occasional use being the most prevalent. The results suggest there is potential to enhance the adoption of mobile applications and increase engagement in the tourism sector.

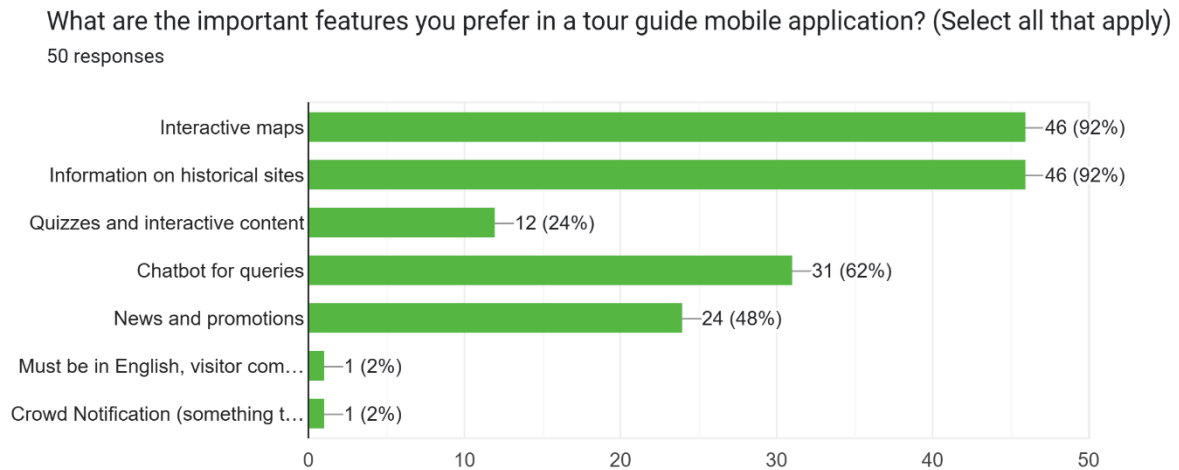


Figure 38. Key features valued by respondents in a tour guide mobile application

The bar chart in Figure 37 highlights the features respondents find most important in a tour guide mobile application, based on a survey of 50 participants. The data reveals clear preferences among the surveyed group. Interactive maps and historical site information emerge as the most valued features, with 92 percent of respondents expressing a preference for them. This demonstrates a strong demand for tools that provide detailed navigation and accessible content. Similarly, 62 percent of participants consider a chatbot for queries to be essential, reflecting the importance of instant support and information. Features such as news and promotions also hold significance, with 48 percent of respondents showing interest, suggesting the appeal of timely updates and offers. Quizzes and interactive content, however, garner a moderate interest of 24 percent, indicating some appreciation for engaging educational elements. The least popular features include the application being in English and crowd notifications, each selected by only 2 percent of participants. This possibly indicates a minimal demand for these aspects within the surveyed group. These findings provide valuable guidance for developers, helping them focus on features that are most likely to enhance user experience and meet the expectations of their audience.

What motivates you to visit historical sites?

50 responses

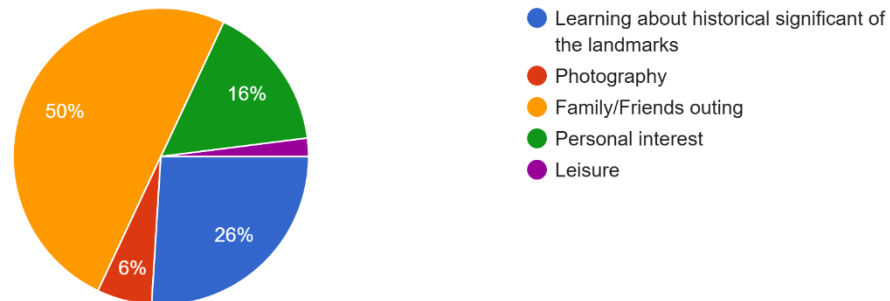


Figure 39. Motivations of respondents for visiting historical sites

Figure 38 depicts a pie chart that summarises the reasons why respondents visit historical sites. The survey data, gathered from 50 people, reveals a wide variety of reasons for such visits. The majority of respondents, 50%, say that outings with family or friends are their primary motivation. Following this, 26% of respondents visit historical sites because they are personally interested in history. Learning about the historical significance of landmarks motivates 16% of respondents, while photography drives 6%. Finally, only 2% of visitors to historical sites do so for leisure. This distribution emphasises the social aspect of visiting historical sites, with family and friend gatherings emerging as the most common reason.

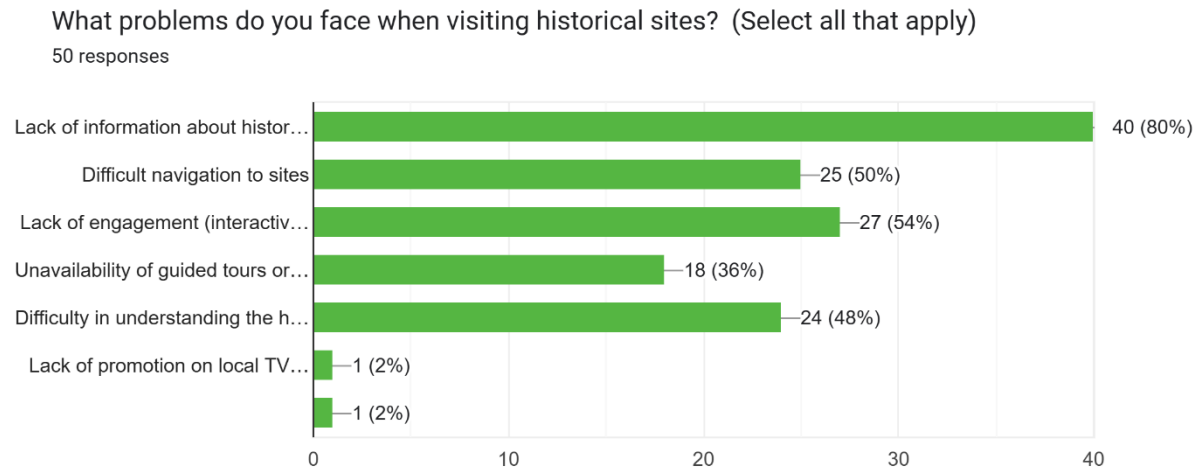


Figure 40. Challenges faced by respondents when visiting historical sites

Figure 39 presents a bar chart that identifies the challenges respondents encounter while visiting historical sites, based on feedback from a survey of 50 participants. The results point to several key areas of concern that hinder the visitor experience. The most commonly reported challenge, cited by 80 percent of respondents, is the lack of adequate information about historical sites. This is closely followed by the absence of engaging or interactive elements, which is an issue for 54 percent of participants. Navigation difficulties also stand out, with 50 percent highlighting problems in finding their way to the sites. Understanding the historical context proves challenging for 48 percent of respondents, suggesting a gap in the delivery of contextual knowledge. Furthermore, 36 percent of participants indicate that the unavailability of guided tours or audio guides negatively impacts their experience. A smaller issue, noted by only 2 percent of respondents, is the lack of promotional efforts through local TV or radio. These insights shed light on significant areas for improvement in enhancing the appeal and accessibility of historical sites. By addressing these issues, there is the possibility to foster greater visitor satisfaction and a deeper connection with cultural heritage.

Do you currently use any historical tour guide mobile app?
50 responses

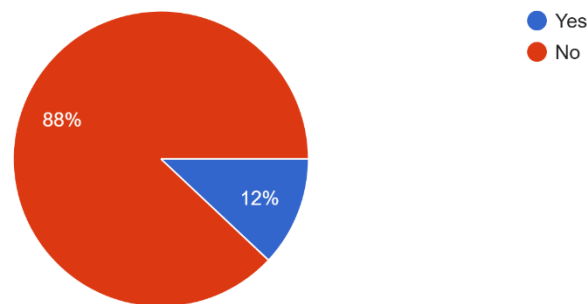


Figure 41. . Usage of historical tour guide mobile apps among respondents

Figure 40 illustrates the extent to which historical tour guide mobile applications are utilised by the surveyed group of 50 participants. The majority of respondents, accounting for 88 percent, stated that they do not use any such applications. In contrast, only 12 percent reported that they actively use historical tour guide apps. This low adoption rate highlights a significant gap in engagement with this type of technology among the sampled group. For developers and stakeholders, these findings underscore the importance of identifying barriers to usage and exploring strategies to promote the adoption of historical tour guide mobile applications.

How can a mobile application enhance your historical tour experience? (Select all that apply)
50 responses

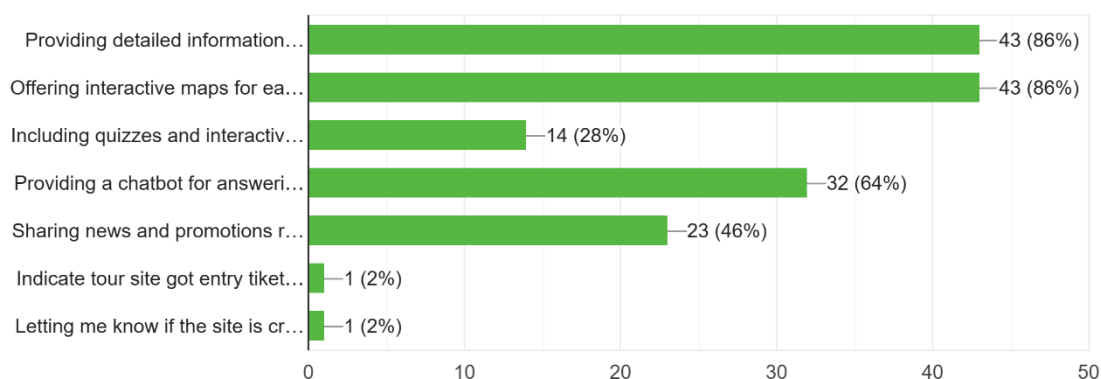


Figure 42. Mobile application features enhancing historical tour experience

Figure 41 presents a bar chart showcasing the features that respondents believe would enhance their experience during historical tours. Based on input from 50 participants, the chart offers valuable insights into user preferences, which can inform the design and functionality of a mobile application tailored to this project. The two most popular features, selected by 86 percent of respondents, are the inclusion of detailed information and interactive maps for navigation. These elements are deemed essential as they offer users both comprehensive knowledge and ease of movement while exploring historical sites. A chatbot for answering questions was highlighted by 64 percent of participants, reflecting a strong interest in real-time assistance. Additionally, 46 percent of respondents expressed a preference for features that provide news and promotions related to historical sites, suggesting moderate interest in staying informed about relevant updates and offers. Features such as quizzes and interactive activities received 28 percent support, indicating some interest in educational and engaging content. However, the least preferred features were those indicating the availability of entry tickets and crowd levels at tour sites, each chosen by only 2 percent of respondents. These findings are instrumental in prioritising the most valued functionalities, enabling developers to focus on features that significantly improve the user experience for historical tours.

What specific features would you like to see in a historical tour guide mobile app? (Select all that apply)

50 responses

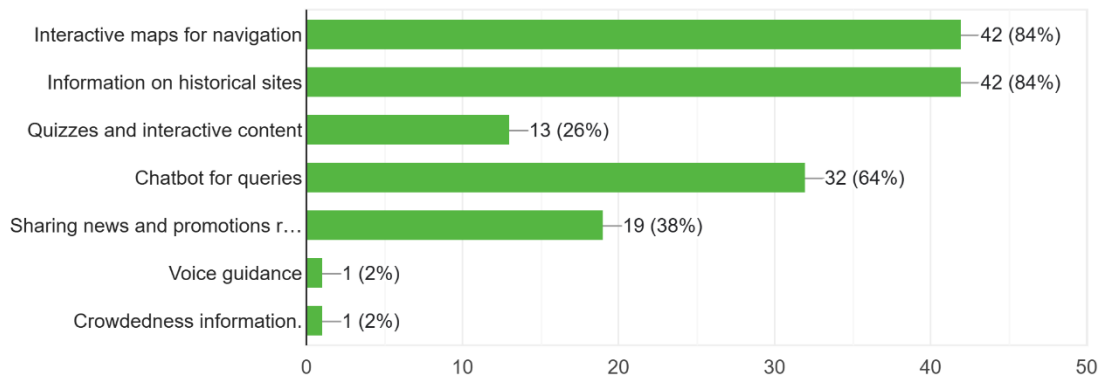


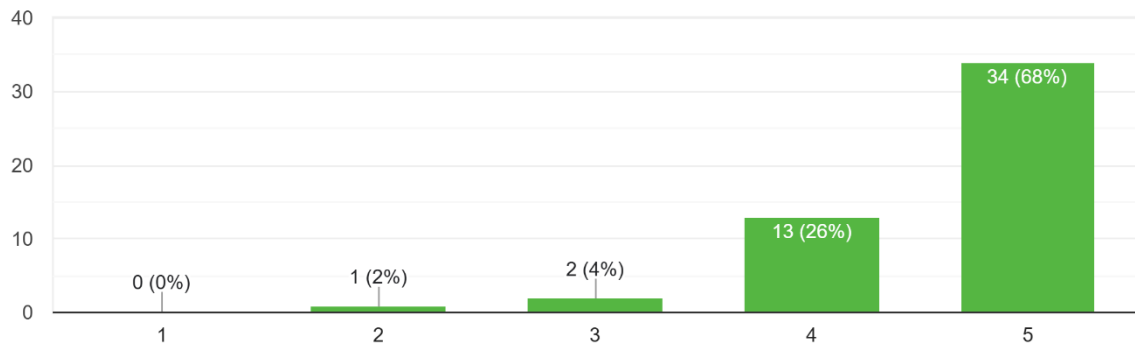
Figure 43. Mobile application features enhancing historical tour experience

The bar chart in Figure 42 presents the specific features that respondents consider most valuable in a historical tour guide mobile application. Data from a survey of 50 participants highlights key preferences that are essential for shaping the development of this final year software engineering project. Interactive maps for navigation and detailed information about historical sites emerged as the top features, with 84% of respondents (42 participants) selecting them. These features are crucial for improving the user experience by offering both seamless navigation and in-depth historical context. A chatbot for handling queries was also highly preferred, with 64% of respondents (32 participants) indicating its importance. This reflects a clear demand for real-time assistance and interactive elements within the app. News and promotions related to historical sites were of moderate interest, with 38% (19 respondents) highlighting this feature. Quizzes and interactive content received 26% of the votes (13 respondents), showing a notable, though lesser, interest in engaging and educational features. The least valued features were voice guidance and information about crowd levels, both chosen by only 2% of participants (1 respondent each). This suggests that these functionalities are not as essential for most users.



How important would you find feature of interactive maps providing navigational direction for historical sites during historical tour?

50 responses



Legend:

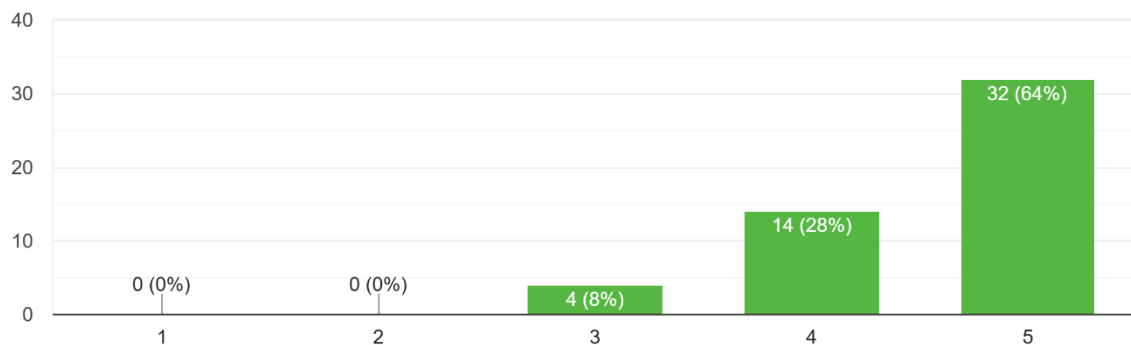
1 - Not important, 2 - Slightly important, 3 - Moderately important, 4 - Very important, 5 - Extremely important

Figure 44. Importance of interactive maps for navigational directions during historical tours

Figure 43 presents data on the perceived significance of interactive maps that provide navigational directions for historical sites, based on responses from 50 participants. The findings show that a large majority, 68% of respondents, consider this feature to be highly important, while 26% rate it as important. Only a small proportion, 4%, view it as moderately important, and an even smaller group, 2%, deem it slightly important. Importantly, none of the respondents expressed that the feature was unimportant. These results are particularly valuable for the project, as they emphasise the strong demand for interactive maps. Integrating this feature into the mobile application could greatly improve its functionality and user appeal, ensuring a more seamless and engaging experience for users during historical tours.

How important is it for a historical tour guide app to provide detailed information on historical sites?

50 responses



Legend:

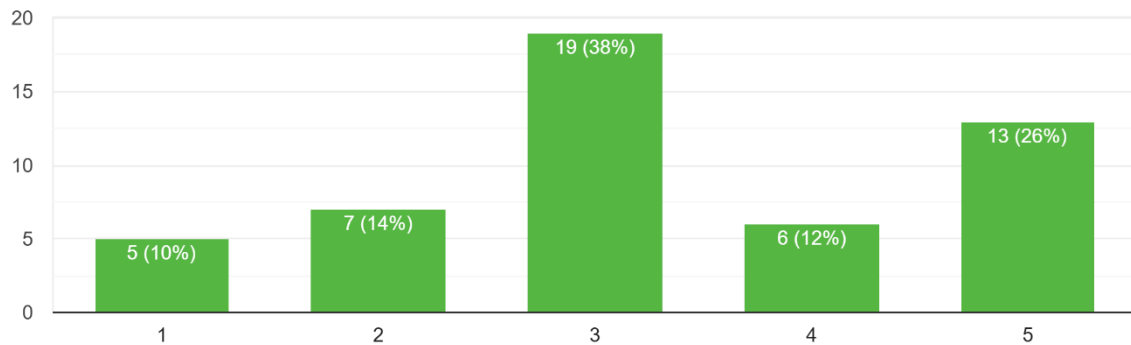
1 - Not important, 2 - Slightly important, 3 - Moderately important, 4 - Very important, 5 - Extremely important

Figure 45. Importance of detailed information in historical tour guide apps

The bar chart in Figure 44 illustrates the importance respondents attribute to having detailed information about historical sites in a tour guide app. Data from a survey of 50 participants reveals that the majority view this feature as crucial. A significant 64% rated it at the highest level of importance, while 28% assigned it a rating of 4. Only 8% of respondents considered it moderately important, giving it a rating of 3, and none rated it as unimportant (levels 1 or 2). This feedback highlights the critical role comprehensive information plays in meeting user expectations. Incorporating this feature into the app can enhance its value and usability, hence for this project, prioritising detailed and thorough content could greatly contribute to user engagement and satisfaction.

How important is it for a historical tour guide app to include quizzes and interactive content to keep users engaged?

50 responses



Legend:

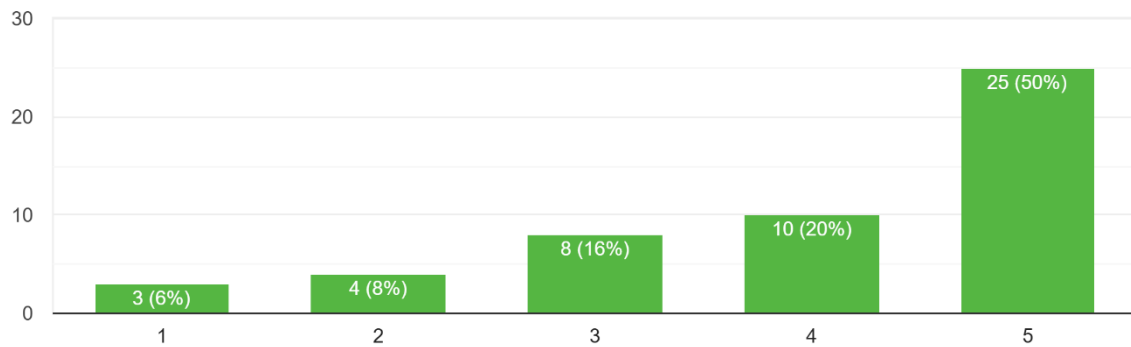
1 - Not important, 2 - Slightly important, 3 - Moderately important, 4 - Very important, 5 - Extremely important

Figure 46. Importance of quizzes and interactive content for engagement

The bar chart (Figure 45) depicts respondents' views on the importance of incorporating quizzes and interactive features in a historical tour guide application to sustain user engagement. Based on feedback from 50 participants, the results display a spectrum of opinions. A significant portion, 38%, assigned a level 3 rating, reflecting a neutral position. Meanwhile, 26% of respondents gave a level 5 rating, showing a strong endorsement of the need for such features. Additionally, 12% of participants rated the importance at level 4, indicating moderate agreement. Conversely, 14% rated it at level 2, and 10% at level 1, showing less enthusiasm for these elements. Overall, the responses indicate a mixed perspective, but a notable fraction of participants recognised the potential benefits of quizzes and interactive content in enhancing user engagement.

How important is it for a historical tour guide app to provide chatbot for queries?

50 responses



Legend:

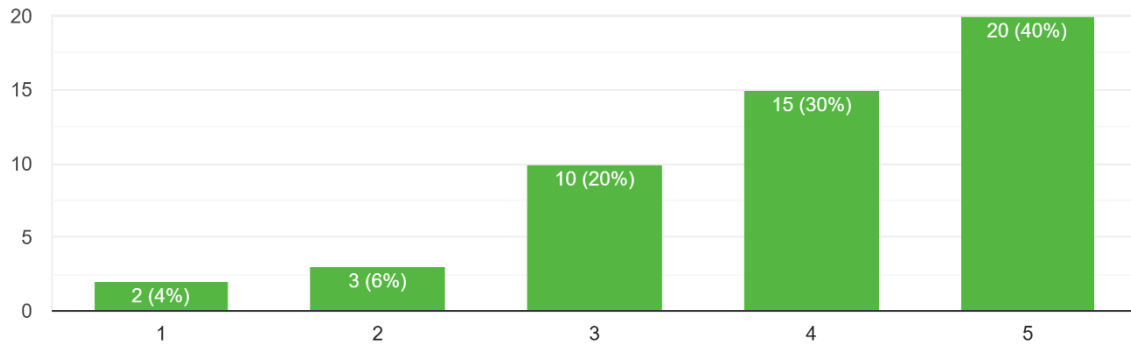
1 - Not important, 2 - Slightly important, 3 - Moderately important, 4 - Very important, 5 - Extremely important

Figure 47. Importance of chatbot feature in historical tour guide app among respondents

The bar chart in Figure 46 provides insights into how respondents value the inclusion of a chatbot feature for addressing queries in a historical tour guide mobile application. Out of 50 participants, half rated this feature with the highest importance level (level 5), reflecting a strong preference among 25 individuals. Additionally, 20%, or 10 respondents, assigned it a rating of 4, while 16% (8 participants) considered it moderately important with a rating of 3. On the lower end, 8% (4 respondents) rated the chatbot at level 2, and 6% (3 participants) deemed it the least important by assigning a level 1 rating. This information is relevant to the development of a Kuching historical tour guide mobile app because it points out the high value users place on having a chatbot feature to answer their questions.

How important is it for a historical tour guide app to provide sharing of news and promotions related to historical sites?

50 responses



Legend:

1 - Not important, 2 - Slightly important, 3 - Moderately important, 4 - Very important, 5 - Extremely important

Figure 48. Importance of sharing news and promotions in a historical tour guide app

Figure 47 presents a bar chart summarising how 50 participants perceive the importance of including news and promotions related to historical sites in a tour guide app. The results reveal that 40% of respondents, or 20 individuals, consider this feature highly important by assigning it the highest rating of 5. An additional 30%, comprising 15 participants, rated it as level 4, reflecting significant interest. A further 20% (10 individuals) viewed it as moderately important, giving it a rating of 3. At the lower end of the scale, 6% (3 respondents) rated the feature as level 2, while just 4% (2 participants) deemed it the least important with a rating of 1. These findings highlight the value users place on receiving updates and promotions within the app. Integrating this feature could enhance user engagement and align the application with user expectations, making it a relevant consideration during development.



Figure 49. Additional feedback on a historical tour guide mobile application

According to Figure 48, the bar chart expressing the feedback received on this project shows that respondents place a high value on usability and ease of navigation. While the inclusion of user reviews and ratings drew some attention, it is clear that a simple, intuitive interface is essential for a good user experience.

3.3.2 Use Case Diagram

Use case diagrams, which describe the high-level functions and scope of a system and identify the interactions between the system and its actors, are a useful tool for modelling the behaviour of a system and helping to capture its requirements. The actors and use cases in a use-case diagram describe what the system does and how the actors use it, but not how the system functions internally (IBM Corporation, 2021). The proposed application's modules are designed using a use case diagram, as illustrated in Figure 49. Each module has a unique set of actors. Furthermore, there will be a discussion for each module, which will be conducted using the use case specification table.

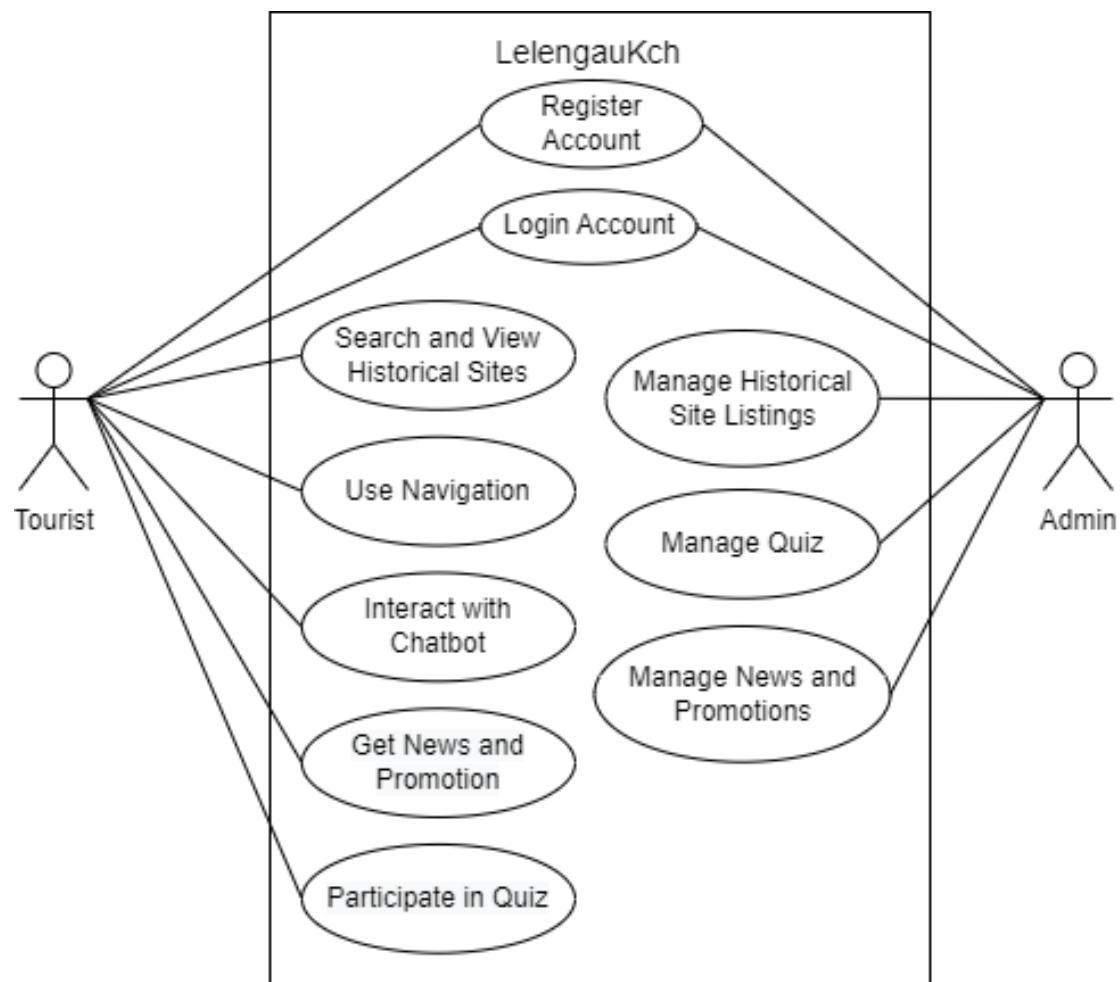


Figure 50. Use case diagram of LelengauKch

3.3.3 Use Case Description

Table 1. Use case specification for register account

Use case ID	UC1
Use case name	Register Account
Description	This use case allows a new user or admin to create an account in the system by providing required information such as email and password. The system registers the user and grants access based on their role.
Actor(s)	Tourist, Admin
Pre-conditions	• The application is installed and launched on the user's device. • The device is connected to the internet. • The user is not registered in the system.
Post-conditions	• If successful, the user account is created, and the user is redirected to the login page. • The system saves the user's information in the database.
	If unsuccessful, an error message is displayed, and the account is not created.
Main flow	1. The user launches the application and navigate to register page. 2. The system displays the registration form, prompting the user to input required personal details needed for account creation. 3. The user fills in the form and submits it. 4. The system validates the user's data. 5. If validation succeeds: a. The system creates a new account for the user. b. A confirmation message is displayed to the user. c. The user is redirected to the home page.
Alternative flow	A1: Email Already Registered 1. The user enters an email that is already registered. 2. The system displays an error message. 3. The user is redirected back to the registration form to try again.
Exception flow	E1: Missing or Invalid Input Fields 1. The user leaves one or more mandatory fields empty or enters invalid data.

	2. The system displays an error message highlighting the issue. 3. The user is prompted to correct the errors and re-submit the form.
--	--

Table 2. Use case specification for login account

Use case ID	UC2
Use case name	Login Account
Description	This use case enables a user or administrator to authenticate and login into the application by entering their credentials (email and password).
Actor(s)	Tourist, Admin
Pre-conditions	• The application is installed and launched on the device. • The user (tourist or admin) has registered an account and has valid login credentials. • The device is connected to the internet.
Post-conditions	If successful, the user is authenticated and redirected to the appropriate dashboard or homepage (tourist or admin view).
	If unsuccessful, an error message is displayed, and the user is prompted to re-enter their credentials.
Main flow	1. The user launches the application. 2. User navigate to the section where login screen is displayed, prompting the user to enter their email and password. 3. The user enters their credentials. 4. The system validates the credentials. 5. If the credentials are valid: a. The user is authenticated. b. The system displays the home page (for tourist) or admin dashboard (for admin).
Alternative flow	None
Exception flow	E1: Invalid Credentials 1. The user enters incorrect or invalid credentials (e.g., wrong password or unregistered email). 2. The system displays an error message, and the user is prompted to enter their credentials again.

Table 3. Use case specification for manage historical site listings

Use case ID	UC3
Use case name	Manage Historical Site Listings
Description	This use case allows the admin to manage the listings of Kuching historical sites. The admin can add, edit, or delete historical site information such as name, description and images. These updates are reflected in the application for tourists to view.
Actor(s)	Admin
Pre-conditions	• The admin is logged into the application. • The system is connected to the database to store or retrieve data.
Post-conditions	If successful, the historical site listings are updated in the database and reflected in the application.
	If unsuccessful, the system displays an error message, and no changes are made to the listings.
Main flow	1. The admin logs in and navigates to the section to manage the historical sites listing. 2. The system displays the current list of historical sites. 3. The admin selects an option to: • Add a new historical site whereby the admin inputs site details (e.g., name, description, location, and images) and submits. • Edit an existing historical site whereby the admin selects a site, modifies the details, and saves the changes. • Delete a historical site whereby the admin selects a site and confirms the deletion. 4. The system validates the input (e.g., mandatory fields are filled). 5. The system updates the database and displays a confirmation message.
Alternative flow	A1: Adding a Duplicate Historical Site 1. The admin attempts to add a historical site that already exists (e.g., same name and location). 2. The system notifies the admin via error message.

	3. The admin is redirected to the form to modify the details or cancel the operation.
Exception flow	E1: Invalid Input 1. The admin enters invalid data such as a blank name for the site. 2. The system highlights the errors and displays an error message. 3. The admin is prompted to resubmit the form with correct details.

Table 4. Use case specification for manage quiz

Use case ID	UC4
Use case name	Manage Quiz
Description	This use case allows the admin to manage quizzes within the application. The admin can add, edit, or delete quiz questions and answers related to Kuching historical sites. These quizzes are intended to engage users and enhance their knowledge of historical landmarks.
Actor(s)	Admin
Pre-conditions	• The admin is logged into the application. • The system is connected to the Firebase database for storing and retrieving quiz data.
Post-conditions	If successful, the quiz database is updated, and changes are reflected in the application for tourists.
	If unsuccessful, the system displays an error message, and no changes are made to the quiz content.
Main flow	1. The admin logs in and navigates to a section to manage the quiz. 2. The system displays the current list of quiz questions. 3. The admin selects an option to: • Add a new quiz question whereby the admin enters the quiz question, multiple-choice answers, and the correct answer, then submits. • Edit an existing quiz question whereby the admin selects a quiz, modifies the content (e.g., question or answers), and saves the changes.

	- Delete a quiz question whereby the admin selects a quiz question and confirms the deletion. - The system validates the input (e.g., question is not blank). - The system updates the database and displays a confirmation message.
Alternative flow	A1: Adding a Duplicate Quiz Question - The admin attempts to add a quiz question that already exists. - The system notifies the admin via an error message. - The admin is redirected to modify the question or cancel the operation.
Exception flow	E1: Invalid Input - The admin enters invalid data such as blank question. - The system highlights the errors and displays an error message. - The admin is prompted to resubmit the question after correcting the details.

Table 5. Use case specification for manage news and events

Use case ID	UC5
Use case name	Manage News and Events
Description	This use case allows the admin to manage news and promotional content related to Kuching. The admin can add, edit, or delete news articles and promotional events to keep tourists informed and engaged.
Actor(s)	Admin
Pre-conditions	- The admin is logged into the application. - The system is connected to the database for storing and retrieving news and promotional data.
Post-conditions	If successful, the news and events database is updated, and the changes are reflected in the application for tourists.
	If unsuccessful, the system displays an error message, and no changes are made to the news or events.
Main flow	- The admin logs in and navigates to a section to manage news and events. - The system displays the current list of news and events.

	3. The admin selects an option to: • Add new news or events whereby the admin enters the title, description, images, and any relevant dates, then submits. • Edit existing news or events whereby the admin selects a specific item, modifies the content, and saves the changes. • Delete news or events whereby the admin selects an item and confirms the deletion. 4. The system validates the input (e.g., title is not blank, image cover is uploaded and valid). 5. The system updates the database and displays a confirmation message.
Alternative flow	A1: Adding Duplicate News or Promotions 1. The admin attempts to add content with the same title or details as an existing item. 2. The system notifies the admin via an error message. 3. The admin is redirected to modify the content or cancel the operation.
Exception flow	E1: Invalid Input 1. The admin enters invalid data such as a blank title for the content. 2. The system highlights the errors and displays an error message. 3. The admin is prompted to resubmit the addition with correct details.

Table 6. Use case specification for user to view and browse historical sites

Use case ID	UC6
Use case name	View Historical Sites
Description	This use case allows tourist to browse for historical sites in Kuching and view detailed information, including descriptions and images.
Actor(s)	Tourist

Pre-conditions	• The tourist has successfully registered and logged into the application. • The system is connected to the database containing historical site data. • Internet connection is available.
Post-conditions	If successful, the tourist views the details of the selected historical site. If unsuccessful, the system displays an error message, and no historical site information is displayed.
Main flow	1. The tourist launches the application and navigates to the “Explore” section. 2. The system displays a welcome card and a list of historical sites. 3. The tourist proceeds to <i>either enters a keyword or filters by category thus selects a historical site from the search results or directly select any available historical sites from the default list.</i> 4. The system retrieves the information about the selected historical site from the Firebase database. 5. The system displays the following information to the tourist including name of the historical site, description and history and images. 6. The tourist has the option to save the site to their favourites or navigate to it using location-based services.
Alternative flow	A1: No Results Found 1. The Tourist enters a keyword that does not match any historical site in the database. 2. The system displays an error message to the user. 3. The tourist is redirected back to the search page to search again.
Exception flow	E1: Invalid Input in Search Bar 1. The tourist enters invalid input such as empty search. 2. The system displays an error message. 3. The tourist is prompted to search again.

Table 7. Use case specification for use navigation

Use case ID	UC7
Use case name	Use Navigation
Description	The user (tourist) utilizes the location-based navigation feature within the mobile app to get directions to a historical site or landmark in Kuching.
Actor(s)	Tourist
Pre-conditions	• The tourist has installed and launch the application. • The tourist has enabled location services on their mobile device. • The tourist has selected a historical site or landmark to navigate to within the application.
Post-conditions	• The navigation feature provides turn-by-turn directions using an integrated map system such as Google Maps. • The tourist is successfully navigated to the selected historical site or landmark.
Main flow	1. The tourist opens the mobile application. 2. The tourist selects a historical site or landmark they wish to visit. 3. The app retrieves the location data for the selected site. 4. The app provides navigation options through integrated map functionality such as Google Maps. 5. The tourist initiates the navigation and turn-by-turn directions are displayed. 6. The tourist follows the directions to reach the historical site.
Alternative flow	A1: Tourist Changes Destination 1. The tourist decides to visit another historical site instead of the initially selected one. 2. The tourist selects the new historical site from the app. 3. The system retrieves location data for the new site and updates the navigation route. 4. The tourist follows the new directions to the selected historical site.
Exception flow	E1: Location Services Disabled

	1. The tourist tries to use the navigation feature while location services are disabled on their device. 2. The system detects that location services are turned off. 3. The system displays a message prompting the tourist to enable location services for navigation to function properly. 4. The tourist may experience navigation issues until location services are enabled.
--	---

Table 8. Use case specification for interact with chatbot

Use case ID	UC8
Use case name	Interact with Chatbot
Description	The tourist interacts with the chatbot within the mobile application to get information about historical landmarks.
Actor(s)	Tourist
Pre-conditions	• The tourist has installed and opened the mobile application. • The tourist has access to the chatbot feature in the mobile application.
Post-conditions	• The chatbot provides relevant information regarding Kuching historical landmarks. • The tourist receives responses from the chatbot based on their queries.
Main flow	1. The tourist opens the mobile application. 2. The tourist navigates to the chatbot feature within the application. 3. The tourist either inputs a query or selects a preset question related to historical sites or landmarks in Kuching. 4. The mobile application processes the query and the chatbot responds with the requested information. 5. The tourist continues the interaction by asking additional questions or seeking more details. 6. The tourist exits the chatbot when they are finished.
Alternative flow	A1: Tourist Changes Query

	1. The tourist decides to ask a different question or inquire about another historical site. 2. The tourist prompts a new query into the chatbot. 3. The chatbot processes the new query and provides relevant information. 4. The tourist continues to interact with the chatbot or exits.
Exception flow	E1: Chatbot Cannot Understand Query 1. The tourist enters a query that the chatbot cannot understand. 2. The chatbot displays an error message to the user. 3. The tourist is prompted to rephrase their question or ask a different one.

Table 9. Use case specification for get news and events

Use case ID	UC9
Use case name	Get News and Promotion
Description	The tourist accesses the “News” section in the mobile application to view updates and events related to historical landmarks in Kuching.
Actor(s)	Tourist
Pre-conditions	• The tourist has installed and opened the mobile application. • News and promotions are available in the mobile application's database.
Post-conditions	• The tourist views relevant news, events, or promotions. • The tourist is informed about activities or offers related to historical sites or landmarks.
Main flow	1. The tourist opens the mobile application. 2. The tourist navigates to the "News" section. 3. The system retrieves and displays the latest news, updates, and promotions. 4. The tourist browses through the content and selects an item of interest. 5. The system provides information about the selected news or promotion.

	6. The tourist exits the section when they are finished.
Alternative flow	None
Exception flow	E1: No News or Promotions Available 1. The tourist accesses the "News and Promotion" section when no content is available. 2. The system displays a message saying, "No news or promotions available at this time. Please check back later." 3. The tourist is redirected back to the main page.

Table 10. Use case specification for participate in quiz

Use case ID	UC10
Use case name	Participate in Quiz
Description	This use case allows tourist to participate in an interactive quiz related to Kuching historical sites. The quiz is designed to enhance the tourist's knowledge and engagement with the historical heritage of the city.
Actor(s)	Tourist
Pre-conditions	• The tourist has successfully registered and logged into the application. • The system has quiz content available in the Firebase database. • Internet connection is available.
Post-conditions	The tourist completes the quiz, and their score is displayed.
Main flow	1. The tourist navigates to the "Quiz" section from the navigation bar. 2. The system displays the welcome screen to start the quiz. 3. The tourist click on the start button to begin the quiz. 4. The system retrieves the quiz questions and displays the first question along with multiple-choice answers. 5. The tourist selects an answer and submits it. 6. The system provides feedback (e.g., correct or incorrect) and displays the next question.

	7. Steps 4–6 repeat until all questions are answered. 8. The tourist has the option to retake the quiz or return to the home page.
Alternative flow	A1: Quitting the Quiz 1. The tourist chooses to exit the quiz mid-way. 2. Upon detecting the tourist's action to exit the quiz such as clicking the quit button, the system displays a confirmation prompt. 3. The tourist click on the quit button and exits the quiz.
Exception flow	E1: Quiz Content Loading Failure 1. The system fails to retrieve quiz questions from the system. 2. An error message is displayed to the user and prompts user to try again. 3. The tourist is redirected to the quiz's welcome screen.

3.4 Design



3.4.1 Class Diagram

A class diagram is a static structure diagram that describes the relationships between various aspects of LelengauKch's proposed project application. Furthermore, this diagram provides an explicit visualisation and detailed description by displaying the system's attributes, classes, relationships, and methods among its objects. Aside from that, the constraints between classes on the application were displayed, as illustrated in Figure 50. Class diagrams show the static structure of classifiers within a system. They serve as foundational notation for other structural diagrams defined by the Unified Modeling Language (UML). These diagrams are useful not only for developers, but also for other team members working on a project. Furthermore, business analysts can use class diagrams to model systems from a business standpoint, making them useful tools in both technical and business domains (Visual Paradigm, n.d.).

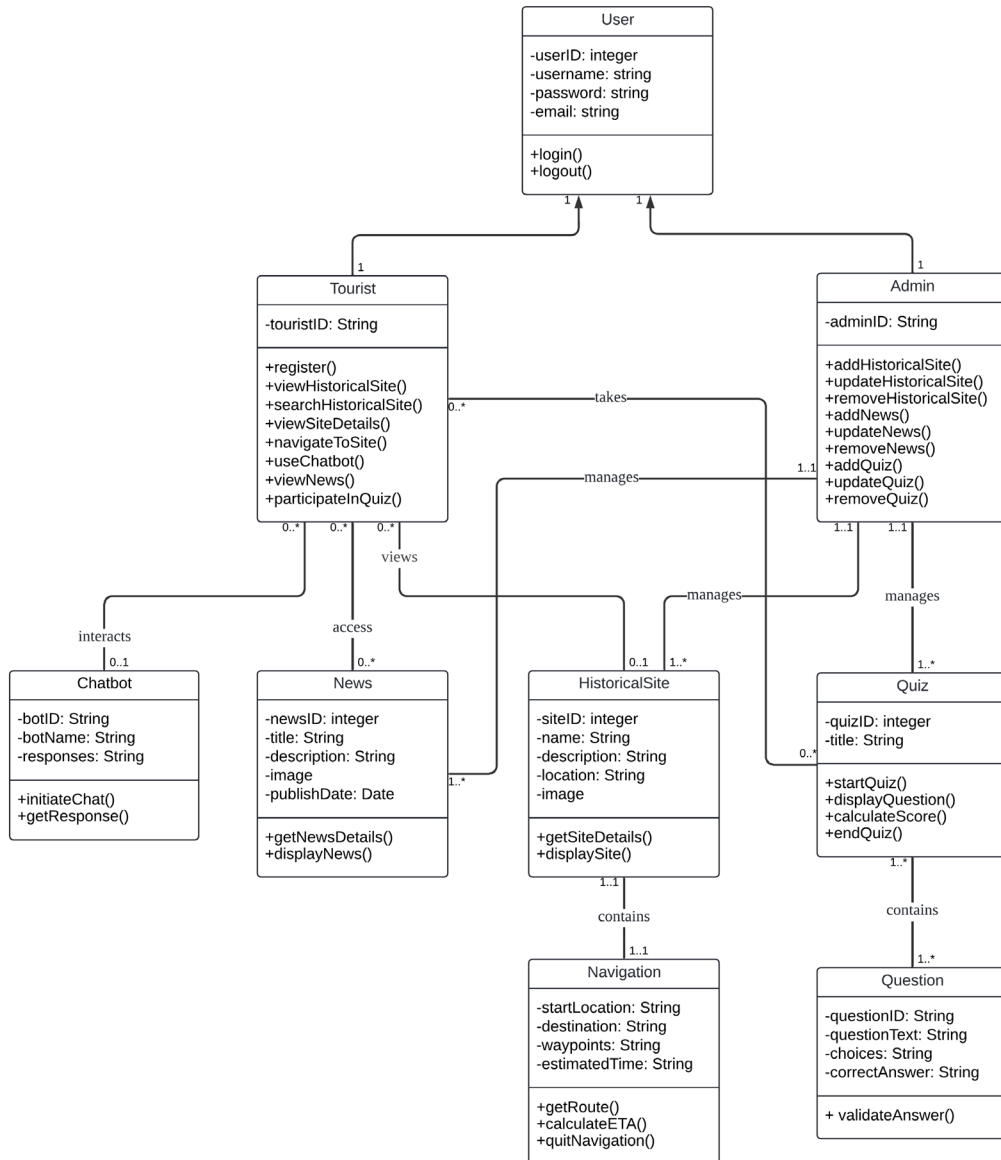


Figure 51. Class diagram for LelengauKch

3.4.2 Database design

A database design is a visual representation of a database's structure, demonstrating how data is organised and how various entities within the system are related. These diagrams are useful tools in both the planning and documentation phases of database development. Figure 51 depicts the proposed project's database design.

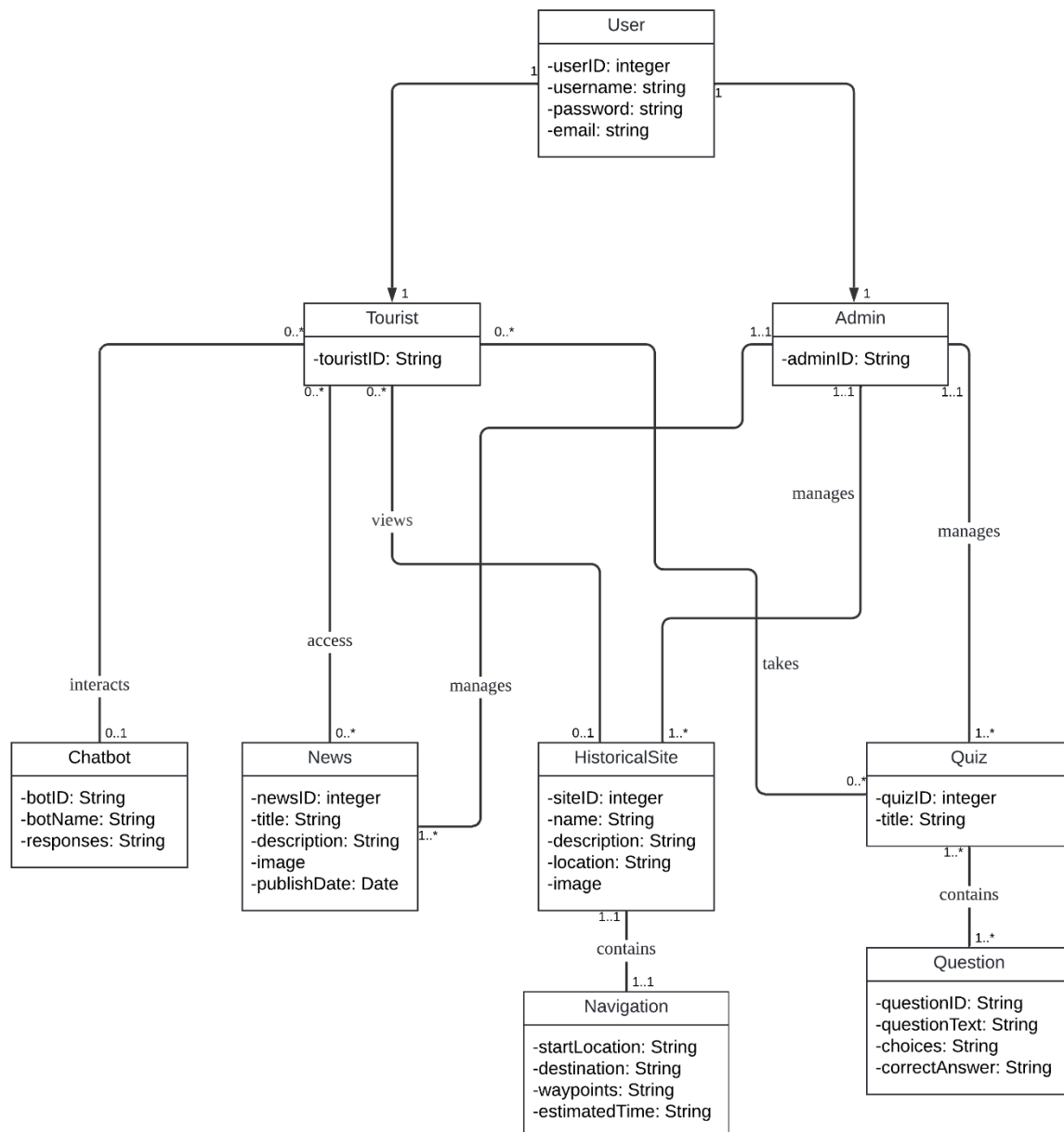


Figure 52. Database design for proposed project

3.4.3 Data Dictionary

The Data Dictionary describes data objects stored in a database. The data dictionary provides detailed information about database objects, including variable names, field types, value ranges, and constraints. The following tables show the data dictionary for this proposed mobile application.

Table 11. Data Dictionary for User

Field	Type	Constraints
userID	INT	PRIMARY KEY
username	VARCHAR(255)	NOT NULL, UNIQUE
password	VARCHAR(255)	NOT NULL
email	VARCHAR(100)	NOT NULL, UNIQUE

Table 12. Data Dictionary for tourist

Field	Type	Constraints
touristID	INT	PRIMARY KEY
userID	INT	FOREIGN KEY → User(userID), NOT NULL

Table 13. Data Dictionary for admin

Field	Type	Constraints
adminID	INT	PRIMARY KEY
userID	INT	FOREIGN KEY → User(userID), NOT NULL

Table 14. Data Dictionary for historical site

Field	Type	Constraints
siteID	INT	PRIMARY KEY
siteName	VARCHAR(255)	NOT NULL
description	TEXT	NOT NULL
photos	VARCHAR(255)	NOT NULL

Table 15. Data Dictionary for quiz

Field	Type	Constraints
questionID	INT	PRIMARY KEY
question	TEXT	NOT NULL
choices	TEXT	NOT NULL
correctAnswer	TEXT	NOT NULL

3.4.4 Architecture Design

An Architecture Design Diagram is a visual representation of a software system's overall structure that shows how its various components interact with one another. It presents a high-level overview of the system's structure. Figure 52 shows a logical diagram of the system architecture for the proposed application.

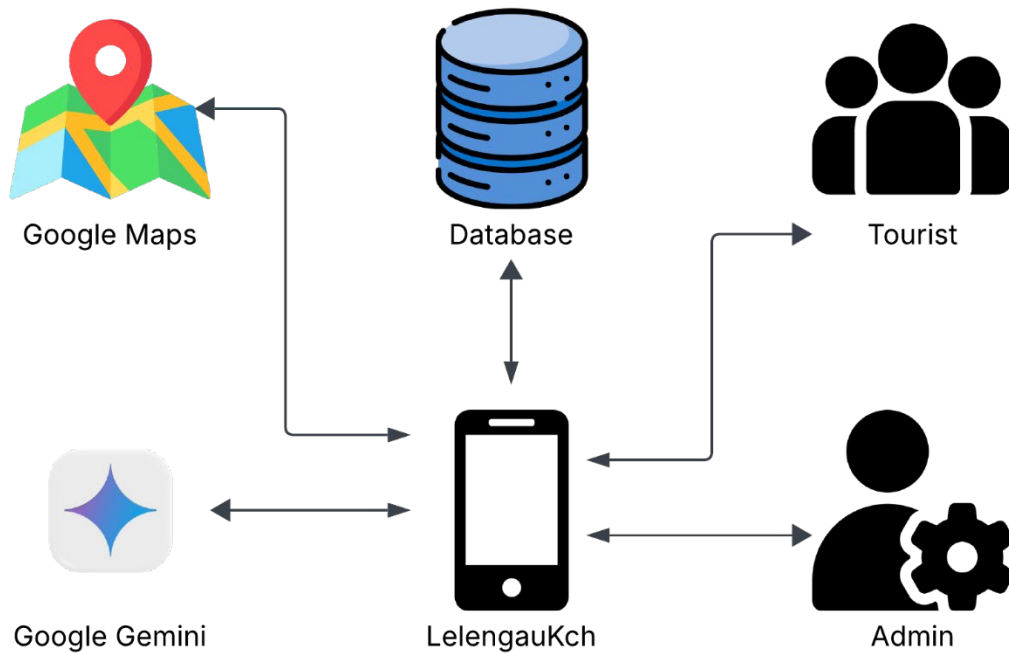


Figure 53. High-level system architecture of proposed application

3.4.5 Activity Diagram

In order to visualize workflows, processes, or activities within a system, activity diagrams are a crucial component of the Unified Modeling Language (UML). They show how various actions are connected and how a system transitions between states. Activity diagrams provide a clear picture of both simple and complex workflows, which helps stakeholders and developers better understand how different elements interact in a system (GeeksforGeeks, 2025). Activity diagrams display the steps involved in the operation of a system, offering insight into the flow of control. They illustrate the sequence in which activities occur, whether sequentially or concurrently. These diagrams also show what causes certain actions or events within a system. An activity diagram starts with an initial point and ends with a final point, emphasising several decision pathways along the way. They are often used in business and process modelling to reflect the behaviour of a system over time. The following Figure 51 shows the activity diagram of the proposed system.

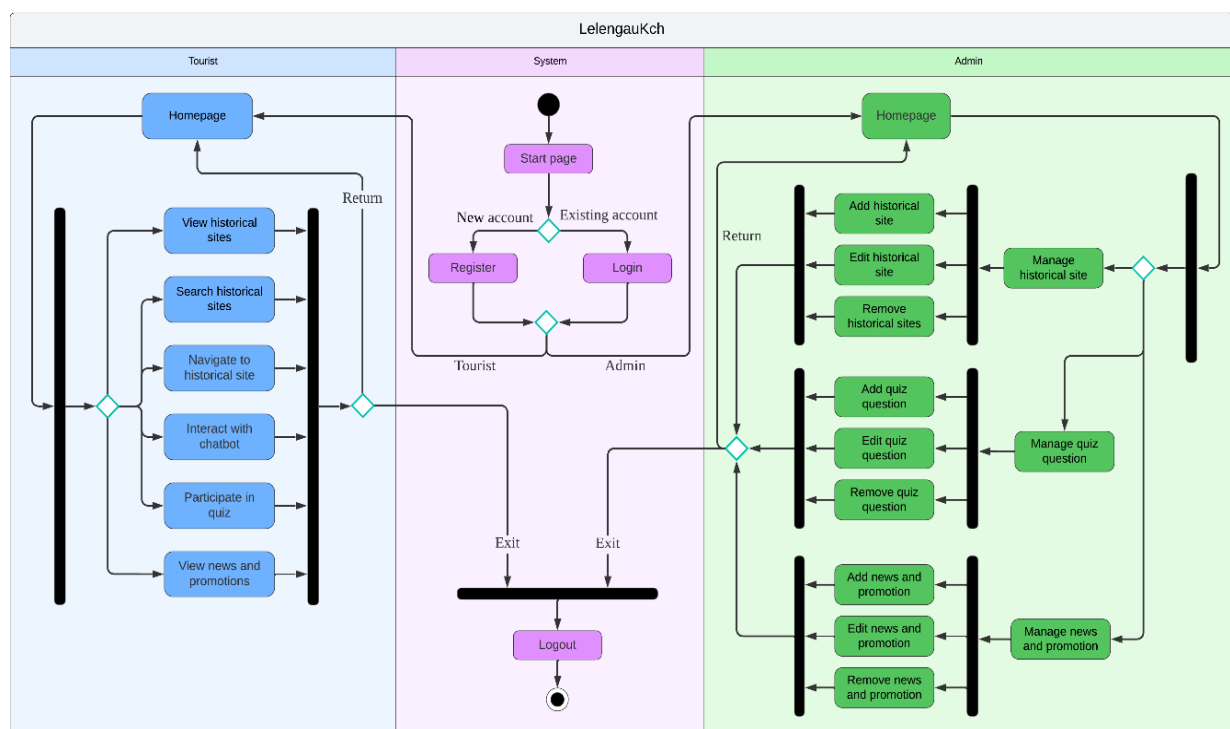


Figure 54. Activity diagram for LelengauKch

3.4.6 Sequence Diagram

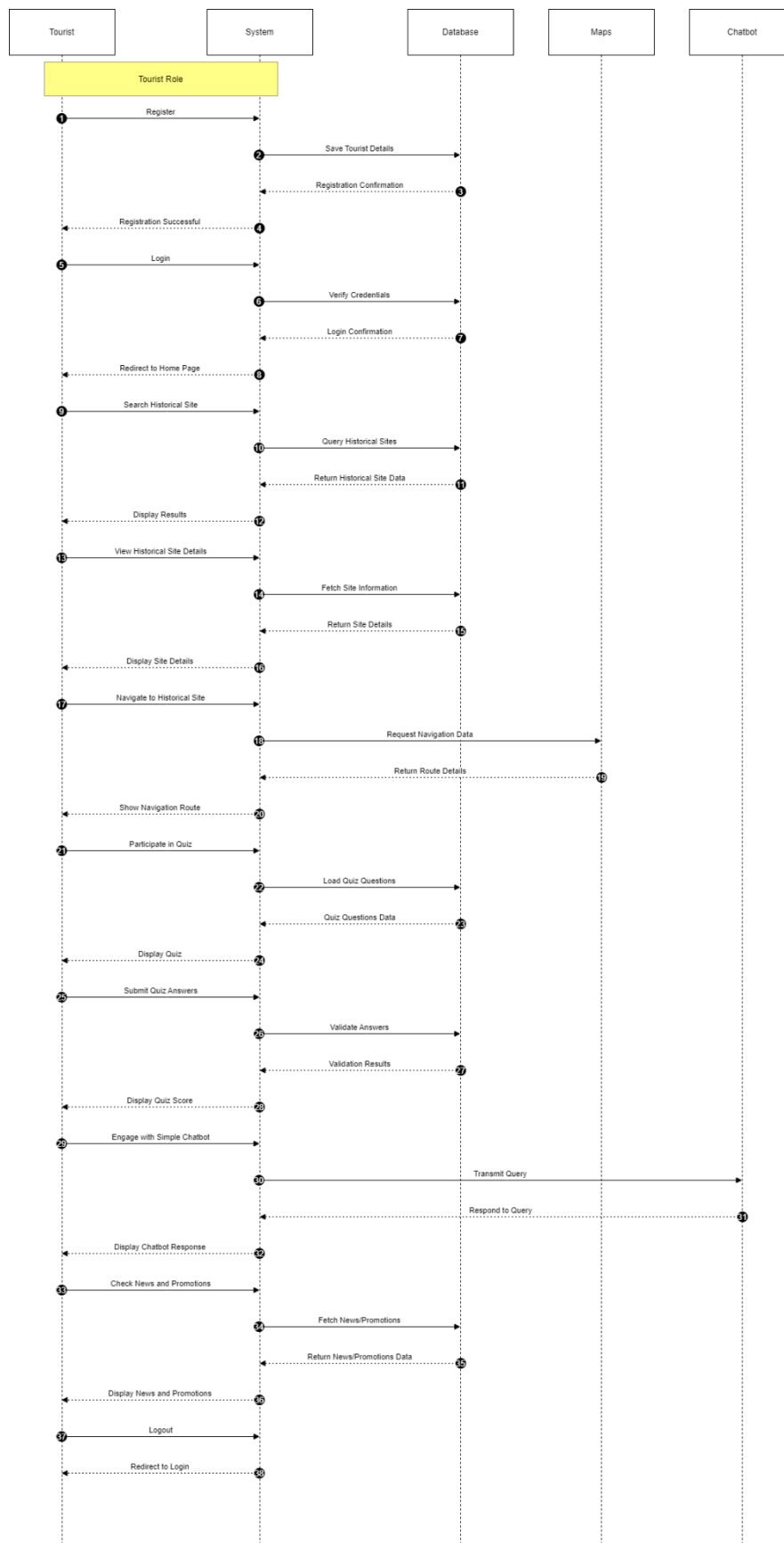


Figure 55. Sequence diagram for general user

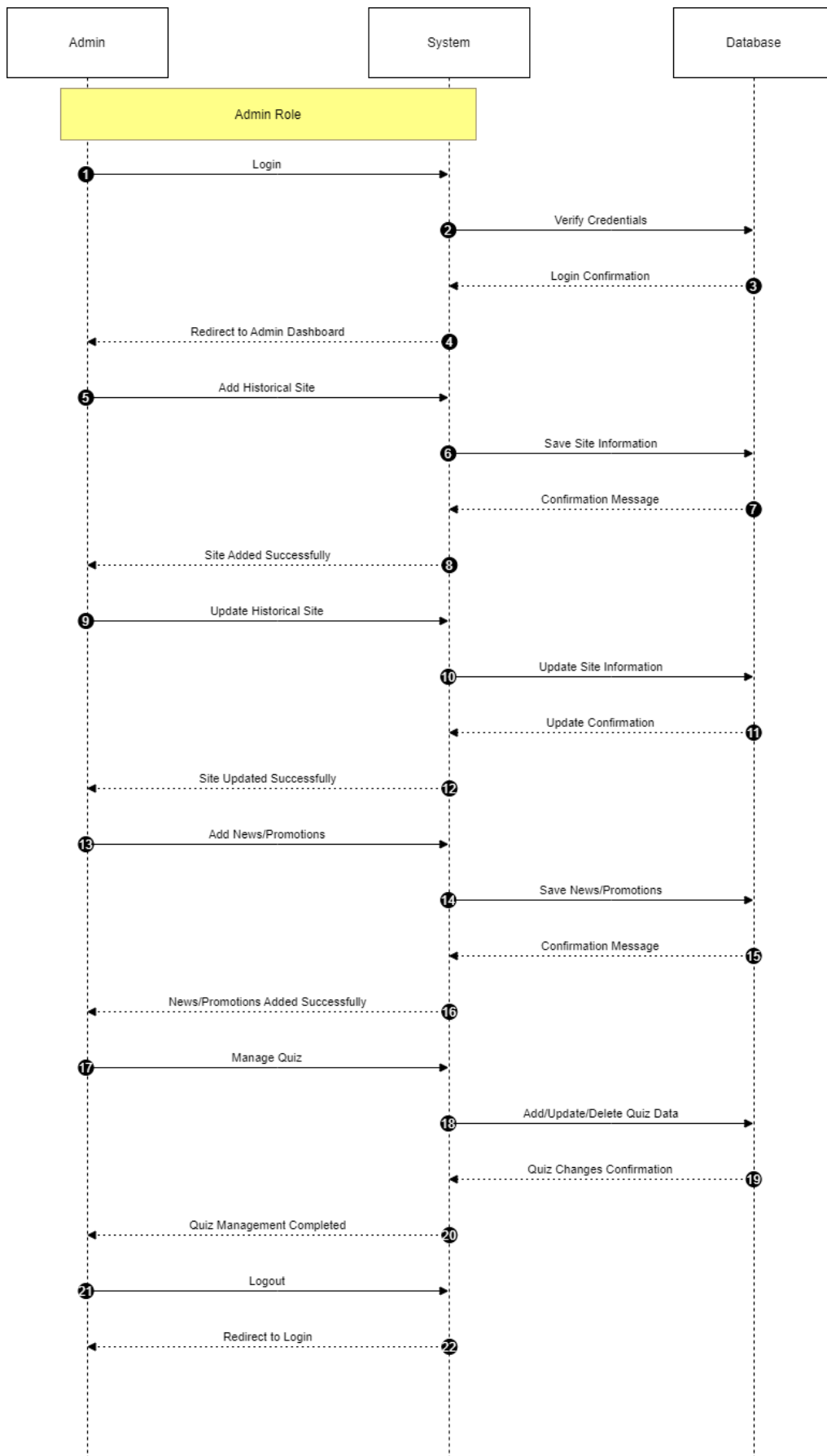


Figure 56. Sequence diagram for admin

Sequence Diagrams are interaction diagrams that show how operations are performed. They document the interaction of objects in the context of a collaboration. Sequence Diagrams are time-focused and visually depict the order of the interaction by using the diagram's vertical axis to represent time, what and when processes are initiated. Figures 52 and 53 show the interaction between tourists, administrators, and the system in a sequential order. It focusses on how objects interact with one another over time, making it an important tool for modelling dynamic behaviour in a system.

3.5 Wireframes

Table 16. Login interface of LelengauKch

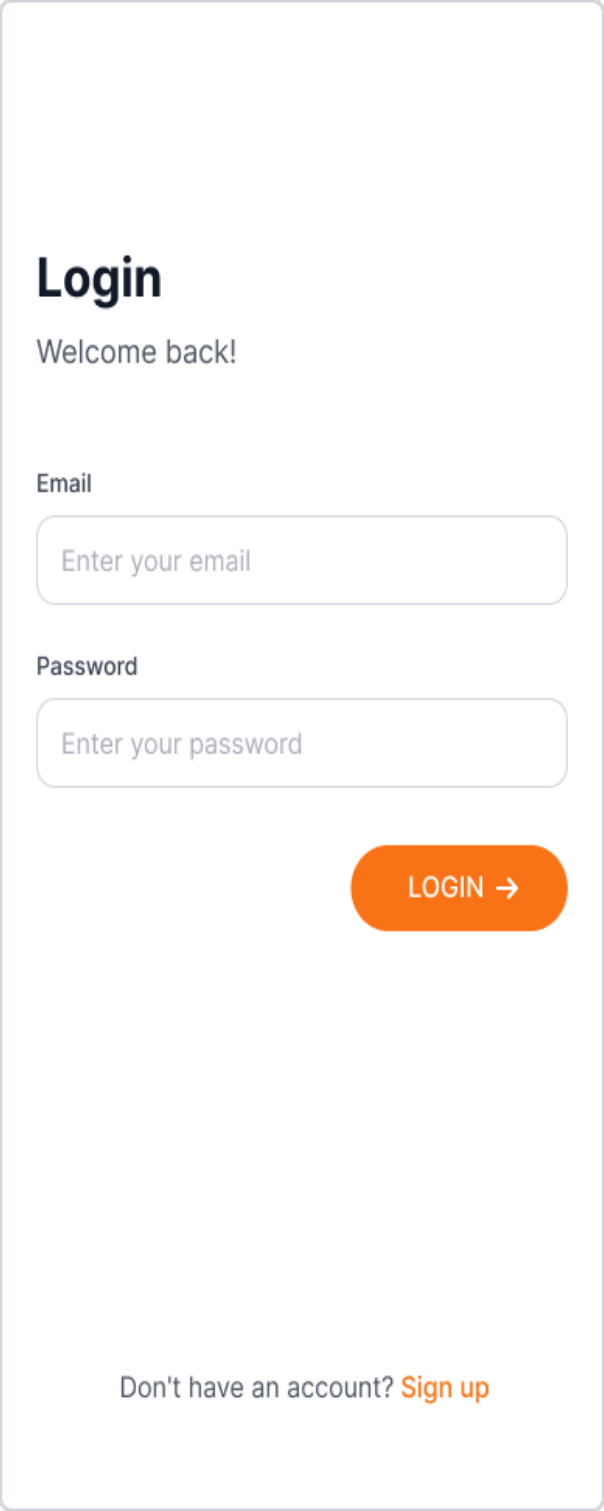
	Scene: Login page Description: This is an entry page that requires user authentication. Users need to enter their registered email and password. After verification, users can access the LelengauKch homepage if their email and password are valid. When invalid data is detected, an error message is displayed. Functions: • Login button: Runs user authentication. If user account is verified, system redirects to corresponding homepage. • Sign up hyperlink: Navigates to the sign up page for account creation.
--	--

Table 17. Sign up interface of LelengauKch

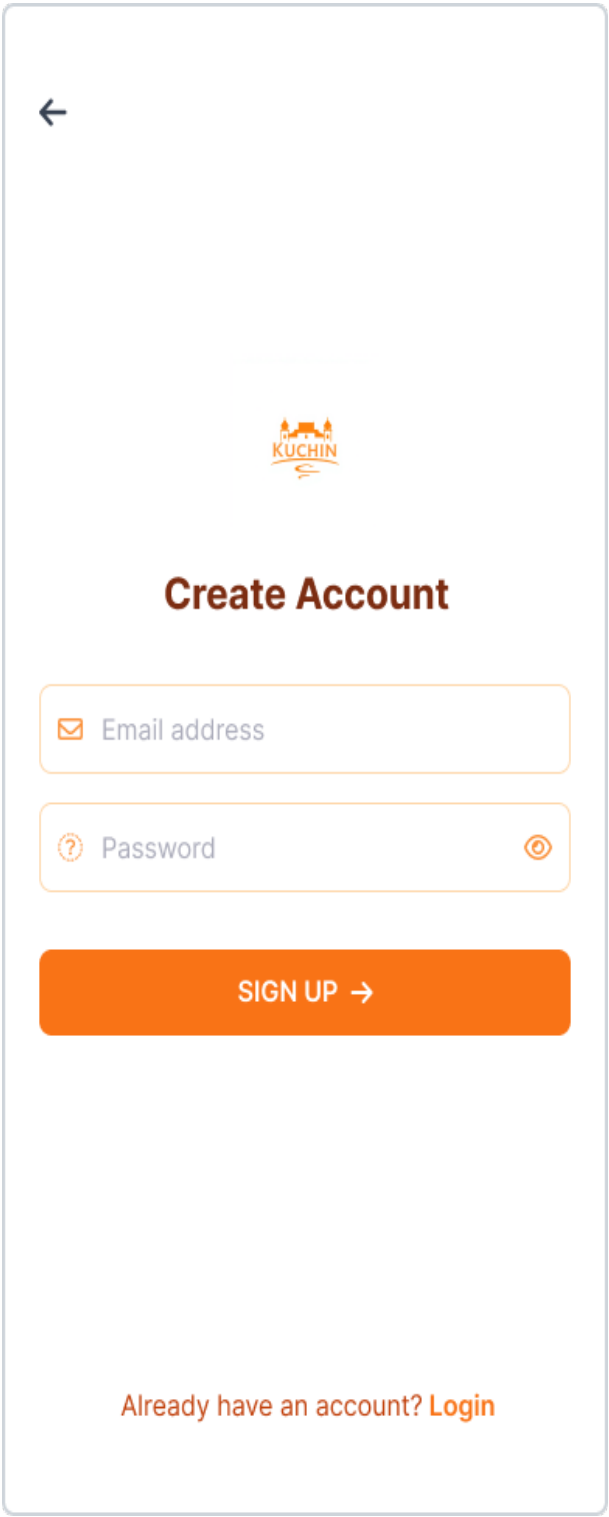
	Scene: Sign up page Description: This interface displayed an account registration form, which users have to fill out to create a new account. Users are required to enter their email address and password. After submitting, users can access the LelengauKch homepage if there are no empty fields. If the registration form is incomplete, an error message is displayed, prompting the user to try again. Functions: • Eye icon: Toggle the visibility of the password in password field. • Sign up button: Register a new user account. If there is no empty field, the system redirects to the corresponding home page. • Login hyperlink: Navigates to the login page for account access.
--	--

Table 18. Explore interface of LelengauKch

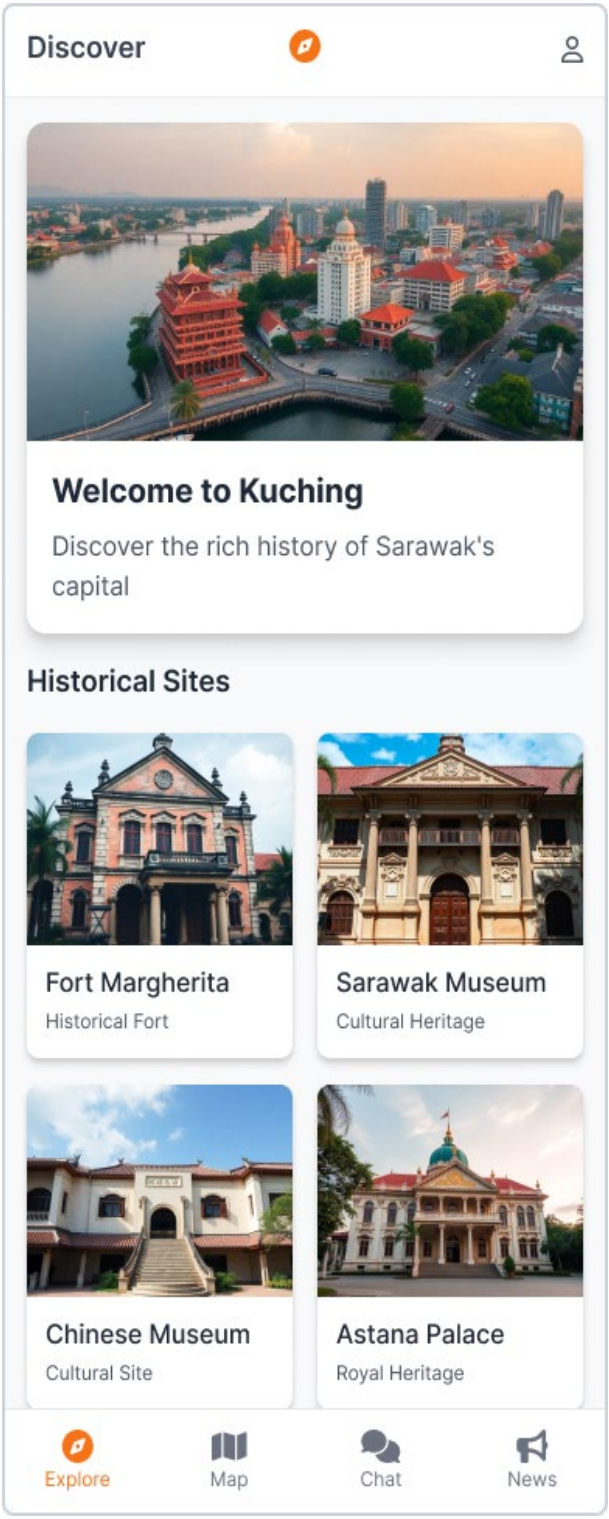
	Scene: Homepage Description: This interface shows the homepage for tourist account. User can tap on the one of the cards to view more details for the selected historical site. Functions: • User icon: View account details and has option to log out of current account. • Footer navigation bar: Explore section that has the list of Kuching historical site, Map section that display navigation to selected historical site, Chat section to interact with a simple chatbot regarding queries related to historical site, and News section to get news and promotions related to historical sites.
--	---

Table 19. View historical site details interface of LelengauKch

 Fort Margherita Built in 1879, Fort Margherita stands as an iconic landmark along the Sarawak River. Named after Charles Brooke's wife, Rane Margaret, this historic fortification served as a defensive structure to protect Kuching from pirate attacks. The fort's architecture showcases a blend of English and local influences, featuring thick walls, narrow windows, and distinctive turrets. Today, it houses the Brooke Gallery, showcasing the history of Sarawak's White Rajahs and the region's rich cultural heritage.	Scene: View site details Description: This interface shows the view historical site details page for tourist account. This page lets users view more details for the selected historical site. There is a compass icon that will redirect user to the map feature for navigation to the viewed site. Functions: • Back arrow icon: Exits the view historical site details page and returns to the list of historical sites. • Compass icon: Navigates user to the navigation map
--	---

Table 20. Navigation interface of LelengauKch

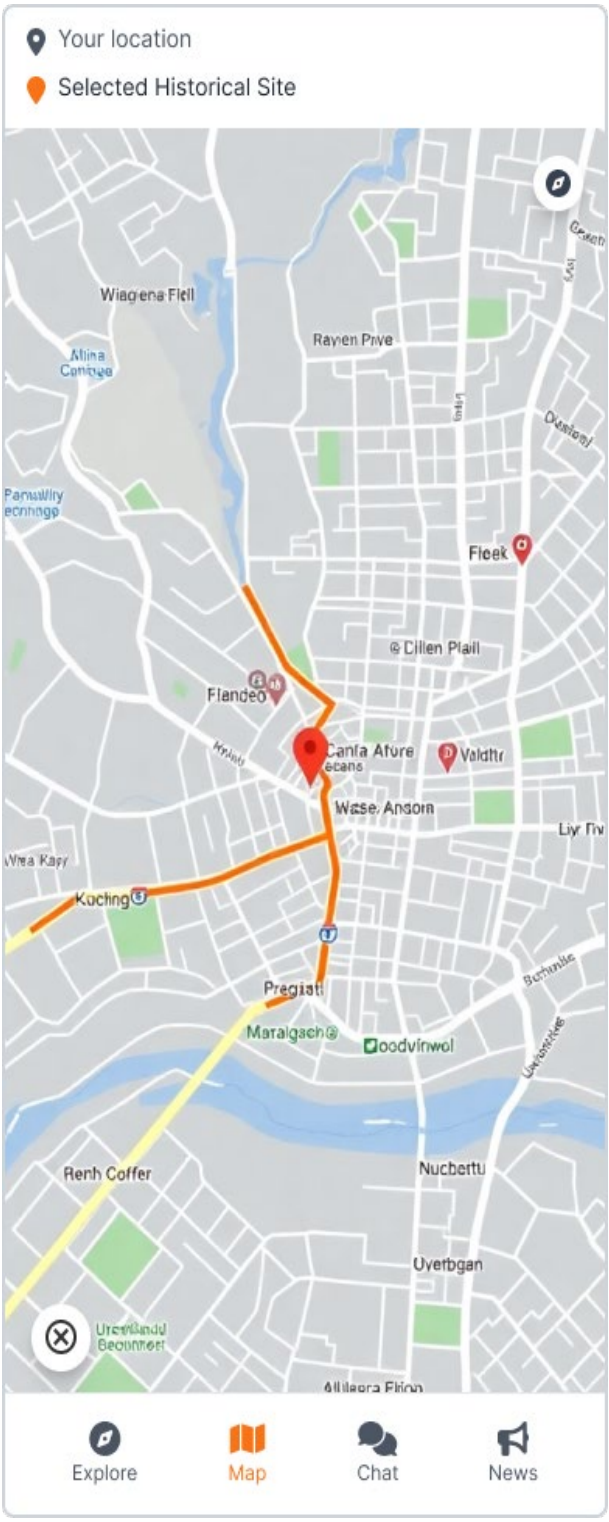
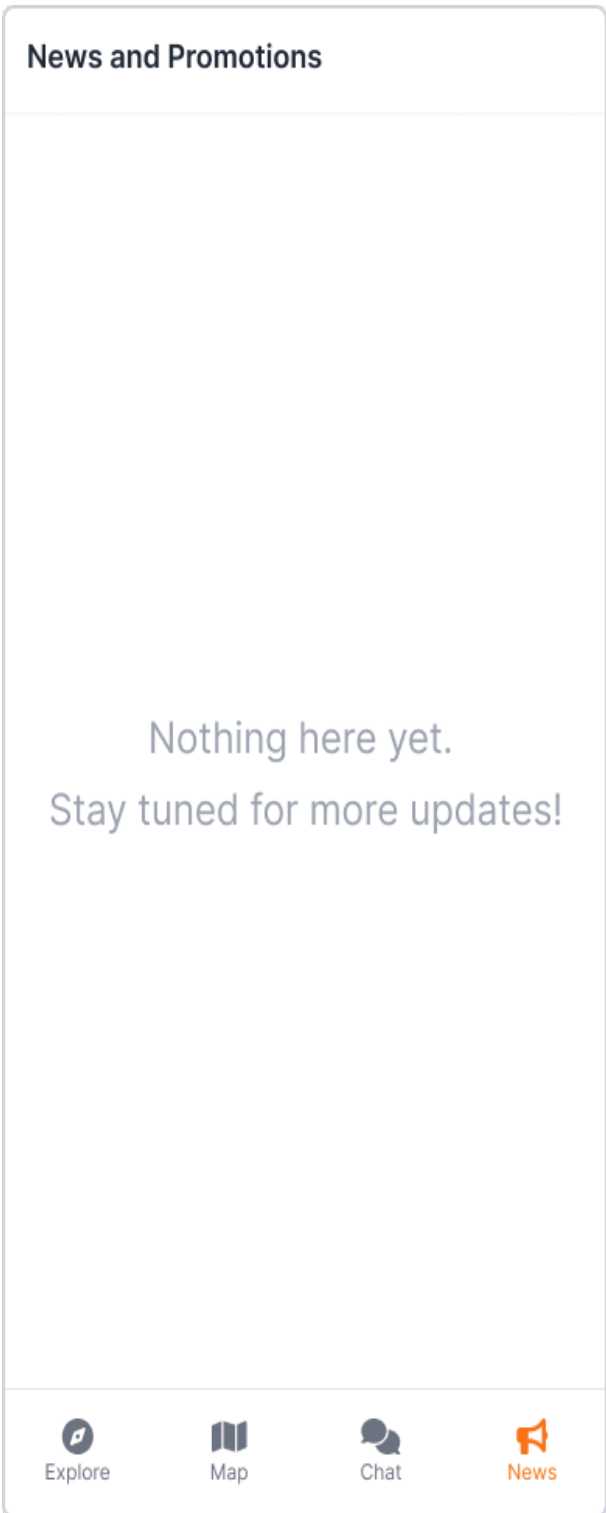
	Scene: Map navigation interface Description: This interface shows the map navigation for tourist account. User can view the route in the map that leads to the selected historical site. Functions: • Compass icon: Reset map orientation to face North up. • Cancel icon: Quits the current navigation function and clears the map route.
--	---

Table 21. Simple chatbot interface of LelengauKch

	Scene: Interact with simple chatbot interface Description: This interface allows tourists to interact with a chatbot for information about Kuching's historical sites. The chatbot provides responses on the history of a particular historical site. The interface includes a text input box where users can type questions and receive responses in a conversational format. Functions: • Text field: Allows user to input their question on historical sites. • Send icon: Sends the text question to be chatbot to be processed and in respond gets a response.
--	--

Table 22. News and promotion interface of LelengauKch

	Scene: News and promotion interface Description: This interface displays the most recent news and promotions for Kuching's historical sites. Users can see a list of articles or promotions, each with a title, short description, and thumbnail image. Selecting an item opens a detailed view with comprehensive information, images, and links. The interface is designed for easy navigation. Functions: • Card: Tapping on a news card will expand to reveal more information on the selected news item.
--	---

3.6 Summary

This chapter has examined the requirements analysis and application design for the proposed LelengauKch mobile application. It begins by detailing the methodology adopted for the project's development, which integrates the Extreme Programming (XP) methodology with the Kanban method. A comprehensive description of each phase within this methodology is provided, along with the specific tasks that must be accomplished in each phase. Furthermore, all requirements for the proposed application are outlined, including user requirements. The findings from the questionnaires are also presented visually in this chapter. Finally, the application design is discussed, encompassing various design types such as module design and system flow design. The logical design is articulated using the Unified Modeling Language (UML) method, which includes an activity diagram illustrating the application flow, a class diagram representing the database design, a use case diagram for module design, and a sequence diagram capturing interactions between system and user.

CHAPTER 4: IMPLEMENTATION

4.1 Introduction

The development and implementation of the mobile application suggested in the preceding chapter are detailed in this chapter. The design presented in Chapter 3 serves as the foundation for the development of "LelengauKch."

4.2 Installation and Configuration of System's Components

4.2.1 Visual Studio Code

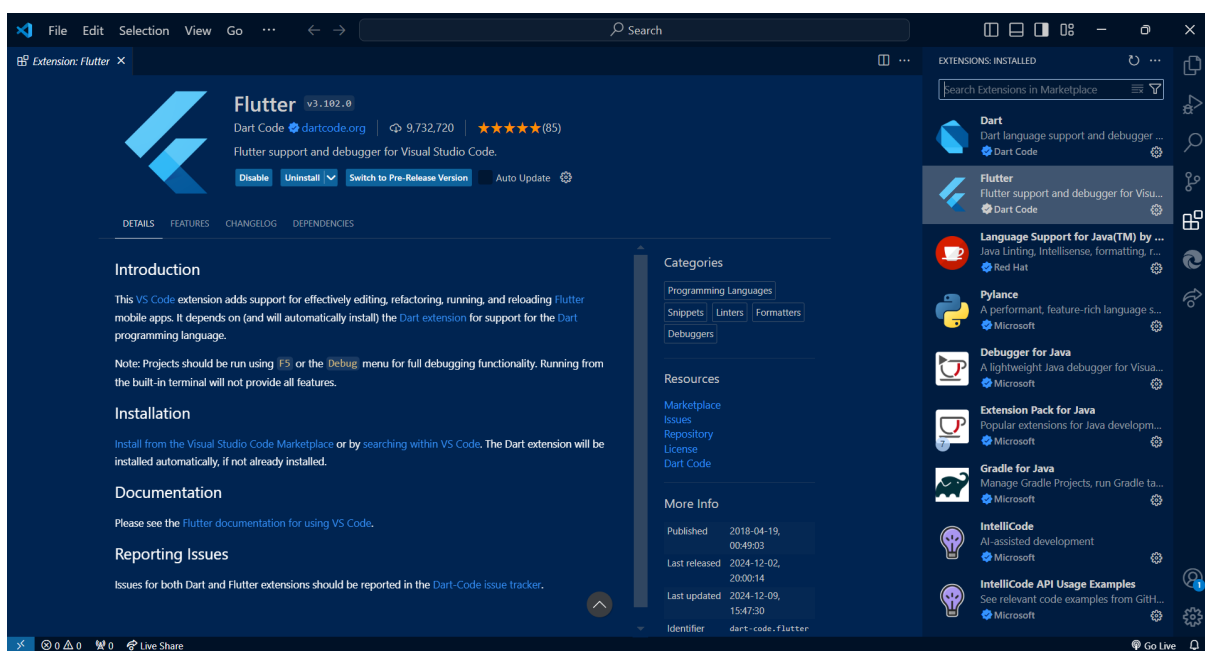


Figure 57. Visual Studio Code installing Flutter extension

The mobile application "LelengauKch" is developed using Version 1.101.0 of Visual Studio Code (commonly referred to as VS Code). The integrated Flutter and Dart extensions furnish comprehensive support for code completion, real-time error checking, widget inspection and hot-reload functionality, significantly accelerating both implementation and debugging. VS Code's extensible architecture facilitates seamless interaction between Dart classes, Flutter widgets, UI layouts, while its built-in terminal and task runner streamline version control and build automation. Furthermore, the editor's lightweight footprint and customisable interface enhance developer productivity and uphold best practices in defensive coding—enabling strict

linting rules, robust refactoring tools and thorough code analysis. Overall, Visual Studio Code 1.100.0 delivers a unified, high-performance environment that underpins the design of “LelengauKch”.

To install Visual Studio Code, begin by downloading the installer appropriate for your operating system from the official VS Code website at <https://code.visualstudio.com/>. Once the download is complete, run the executable file and proceed with the installation by following the on-screen instructions. During this process, you may customize the setup by selecting your preferred installation location and opting to add VS Code to the system’s environment variable PATH.

After installation, launch Visual Studio Code and access the Extensions pane by clicking the corresponding icon in the sidebar or by pressing the keyboard shortcut Ctrl+Shift+X. Within the Extensions view, search for and install the "Flutter" and "Dart" extensions to enable Flutter development support. Once these extensions are installed, VS Code may prompt you to reload the application to activate them fully.

To confirm that the installation was successful, initiate the creation of a new Flutter project by opening the command palette using Ctrl+Shift+P and selecting the "Flutter: New Project" option. This step ensures that both the editor and necessary extensions are functioning correctly.

4.2.2 Firebase

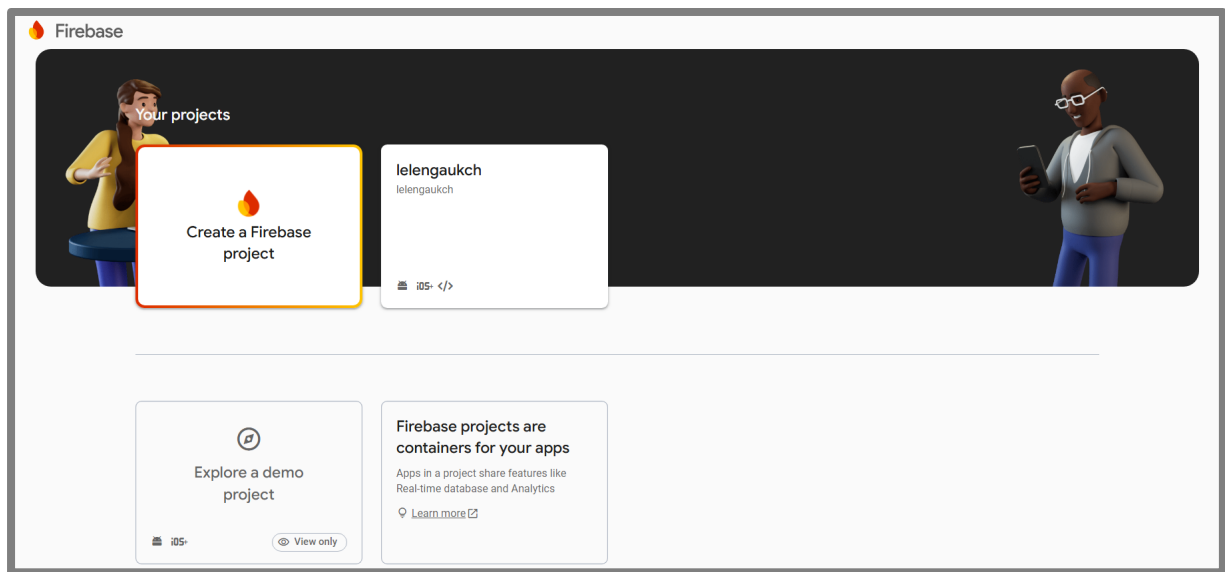


Figure 58. Firebase Console to setup and configure project

With a comprehensive range of services supporting mobile capabilities, Firebase provides the central backend platform for “LelengauKch”. In this project, Cloud Firestore has been adopted to store and query landmark data with real-time synchronisation across all clients, ensuring that any updates including new site entries or updated content, propagate instantly without requiring troublesome refreshes. Meanwhile, Firebase Authentication manages secure user sign-in via email/password provider, enforcing robust access control and session management. Additional services such as Cloud Functions, or run tasks to return AI chat responses when new queries are sent about historical sites. Overall, Firebase’s integrated toolset accelerates development, ensures data consistency, and scales seamlessly with the application’s potential growth in user base.

To integrate Firebase into the project, begin by creating a Firebase project. This involves signing in to the Firebase Console, selecting “Add project,” and entering the desired project name. You will then be guided through configuration settings, which require accepting the terms of service as part of the setup process.

Once the project is set up, proceed to add Firebase to your Flutter application. Select the Flutter icon to begin the integration. The process is divided into three main phases: preparing the workspace, installing and running the FlutterFire CLI, and initializing Firebase along with the necessary plug-ins.

During the first phase, “Prepare your workspace,” install the Firebase CLI and authenticate your account by running `firebase login`. At this point, you may either create a new Flutter project or use an existing one.

In the second phase, “Install and run the FlutterFire CLI,” install the FlutterFire CLI tool globally by executing the initial command provided in the Firebase setup guide. Then, navigate to the root directory of your Flutter project and run the second command as instructed.

Finally, in the third phase, “Initialize Firebase and add plug-ins,” follow the detailed instructions provided on-screen to complete the initialization process. This final step ensures Firebase is correctly linked to your Flutter project and the required plug-ins are installed, enabling a seamless backend integration.

4.2.3 Google Maps API

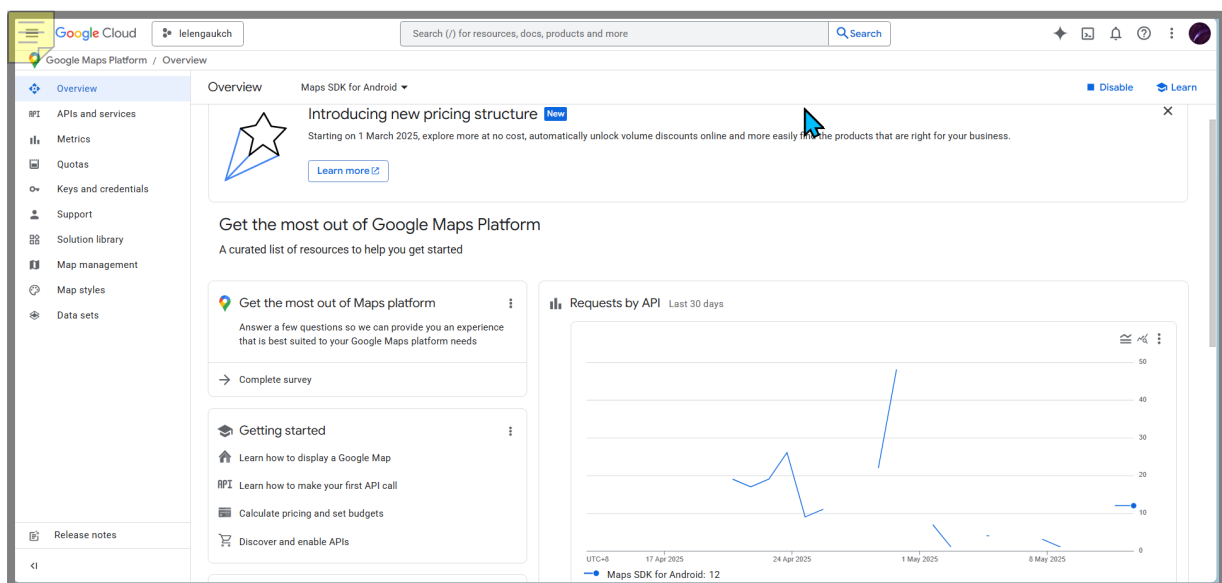


Figure 59. Google Maps API component overview

The LelengauKch mobile application incorporates location-based features to enable its primary task of navigating users to heritage locations. To do this, the Google Maps API was used to embed an interactive map within the app that displays a curated list of markers for culturally significant locations in and about Kuching. When a marker is clicked, the app sends the user to the native Google Maps app, initiating real-time navigation. To receive an individual API key for approved usage and access of the Google Maps API, one must register with the Google Cloud Platform. Spatial context and user experience are maximised through this integration by leveraging pre-existing navigation software to provide an efficient and intuitive routing experience.

4.2.4 Firebase Vertex Ai

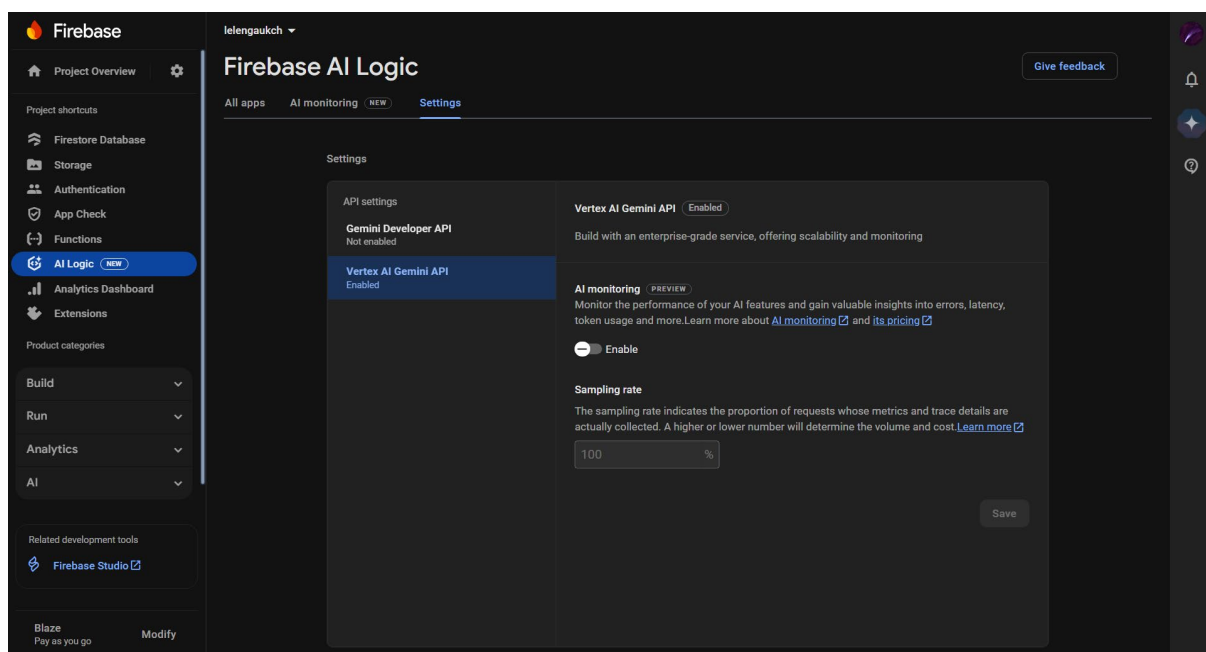


Figure 60. Vertex AI Gemini API configuration

Firebase AI Logic is a feature provided under Firebase that allows developers to integrate Google's generative AI capabilities into client applications. In this project, the AI component is used to provide intelligent content generation features (such as interactive responses) through the Vertex AI Gemini API, targeting Android devices via a Flutter frontend.

The integration enables the app to directly invoke Google's Gemini model using Firebase AI Logic SDK, eliminating the need for a custom backend or third-party intermediary. The implementation follows a client-driven approach, where prompt requests are sent from the mobile app, and responses are generated in near real-time.

To configure and use the Vertex AI Gemini API in the Flutter project, several setup steps are required in both Firebase Console and the Flutter environment. Begin by accessing the Firebase Console and selecting the existing project. From the menu, navigate to the AI Logic section and locate the Firebase AI Logic option. Upon clicking "Get started," select the Vertex AI Gemini API backend. This selection will enable the Vertex AI API on the associated Google Cloud project and require an upgrade to the Blaze billing plan.

With the cloud configuration in place, proceed to integrate the Firebase SDK into the Flutter application. Open the `pubspec.yaml` file and include the necessary dependencies, specifically `firebase_core` and `firebase_ai`, ensuring that the latest versions are used. After specifying the dependencies, execute `flutter pub get` to fetch and install the required packages.

Next, initialize Firebase in the application's main entry point. Begin by ensuring that Flutter binding is initialized, then call `Firebase.initializeApp()` before running the application using `runApp(MyApp())`. This sequence establishes Firebase connectivity at startup.

The integration continues with configuring Firebase for Android. Download the `google-services.json` file from the Firebase Console and place it within the `android/app/` directory of the Flutter project. Finally, proceed to create and utilize the Vertex AI Gemini model as part of your application's AI logic. This step builds upon the foundational configuration and enables advanced features powered by Google's Vertex AI platform.

4.3. Implementation of the Proposed System

Implementation of the mobile application LelengauKch as suggested translates the provided design artefacts into a functional mobile application through the implementation of several

modules such as data access, user interface, navigation integration and AI chat model by way of API call request. In this phase, the Firebase back end is established, Cloud Firestore schemas are determined for landmark storage, the Google Maps API is integrated for map rendering and marker management, and the Flutter front end is constructed with a feature-first folder structure. It is an iterative process, with each sprint deploying a vertical slice of functionality, from data CRUD operations to marker-tap navigation hand-off.

4.3.1 Introduction of User Roles

The mobile application will have two categories of users which are administrators and users. Each of these user classes will have different privileges in the system. The table below shows the user privileges in the system.

User type	Administrator	User
Account privileges	• Log in with email and password • Request password reset via email • Verify email address • Log out from the application • Manage historical sites • Manage news and events items • Manage user accounts	• Create a new account through signup • Log in with email and password • Request password reset via email • Verify email address • Log out from the application • View list of historical sites and its information • View site details on map • View news items • Mark news as read • Chat functionality

4.3.2 Workflow for system's Application

This section outlines the workflow of the "LelengauKch" mobile application. The workflow illustrates the sequence of user interactions and system processes that collectively support the application's core features.

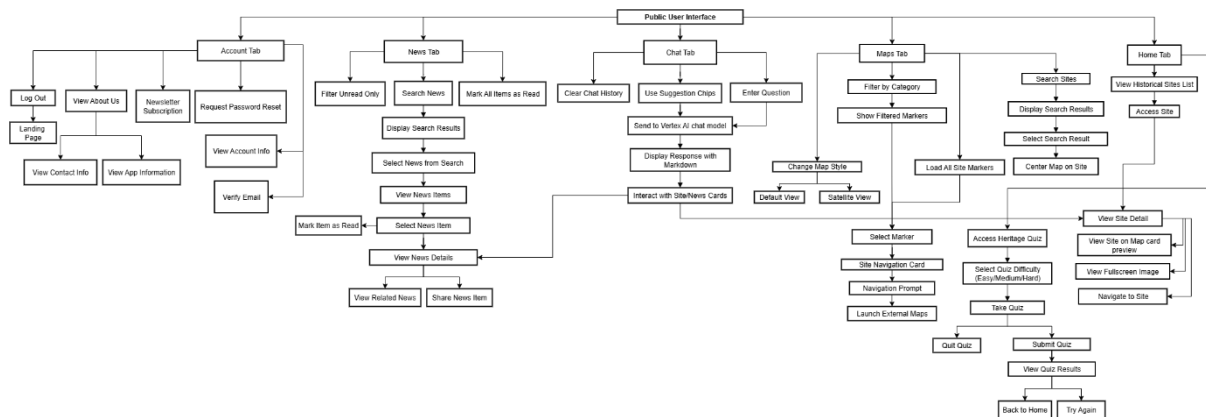


Figure 61. Public user interface

Figure 61 presents a generalized workflow structure for the LelengauKch Flutter application ecosystem. The diagram is structured as a directed graph showing user pathways, interface elements, and system functionality. The structure is designed with a hierarchical pattern of navigation based on a tabbed interface.

In terms of architectural structure, the application's architecture adopts a hub-and-spoke model, where the Public User Interface functions as the central hub, orchestrating access to the application's core features. From this centralized interface, users are directed to five primary navigation nodes. Each represents a distinct functional module and designed for separation of concerns and maintainability. The Home tab enables users to explore and interact with historical site discovery content, emphasising educational and historical enrichment. The Maps tab integrates geospatial visualization tools, allowing users to navigate and interact with location-based data, supporting feature such as site geolocation. The Chat tab features an AI-driven conversational interface, allowing for natural language interactions to provide assistance,

guidance, or retrieve contextual information. The News tab serves as an information dissemination module, curating updates, articles, and historical news relevant to the application's domain, thereby supporting content tracking and passive learning. Finally, the Account tab manages user-related functionalities, including authentication and session control. This hub-and-spoke architectural pattern promotes modularity, facilitates scalability, and aligns with best practices in user-centric mobile application design by providing a cohesive yet decoupled navigation structure.

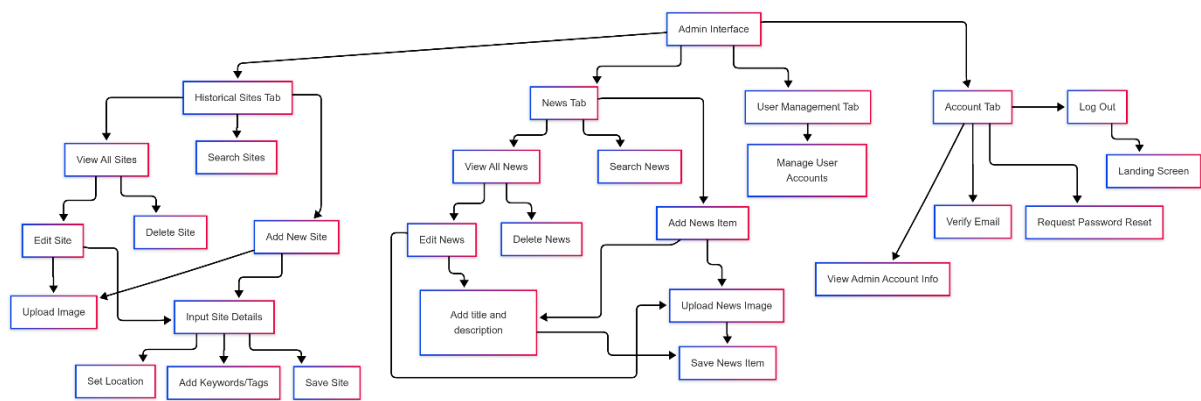


Figure 62. Admin interface

For the administrator role as shown in Figure 62, the design is constructed around a single Admin Interface node leading directly to four top-level functional domains including Historical Sites, News, User Management, and Account. In addition, there is a global log out function that returns the administrator to the application's Landing Screen. The top-level decomposition enforces separation of concerns which demonstrates that each tab has a uniform set of workflows, minimizing cross-module dependencies and maintainability.

Within the Historical Sites tab, there are two entry points, namely view all sites and search sites, which both provide read access. By default, all sites are displayed to the admin. The system displays existing records and then enables two subsequent actions, namely Edit Site and Delete Site. The Edit Site operation first invokes an Upload Image subroutine and then

presents an Input Site Details form. This form requires two metadata fields which are Set Location and Add Keywords before finally implement Save Site function. Moreover, the Add New Site pathway bypasses the listing view but reutilises the same details-entry pipeline, thereby ensuring consistency in both client-side validation and persistence logic.

The News Tab follows an analogous CRUD structure to the Historical Sites workflow. First, it offers a "View All News" function, which consequently allows either the "Edit News" operation, leading the user into an "Add Title and Description" form, or the "Delete News" operation, in addition to a separate "Search News" function. Furthermore, an independent Add News Item action guides the user through an Upload News Image step followed by Save News Item, thus ensuring that each news entry incorporates both textual and multimedia content prior to Cloud Firestore database commit.

The User Management Tab is solely focused on Manage User Accounts, implying role-based access controls and audit logging behind the scenes. The Account Tab handles administrator self-service specifically Verify Email, Request Password Reset, and View Admin Account Info, each of which enforces secure, feedback-driven flows. Finally, invoking Log Out terminates the session and redirects to the Landing Screen, terminating all ongoing workflows in a controlled fashion. This tab-based, modular design supports a familiar navigation flow, stable data validation, and clear audit trails, all of which are features of robust Flutter admin interfaces.

4.4 Modules of system's Application

4.4.1 Account Registration

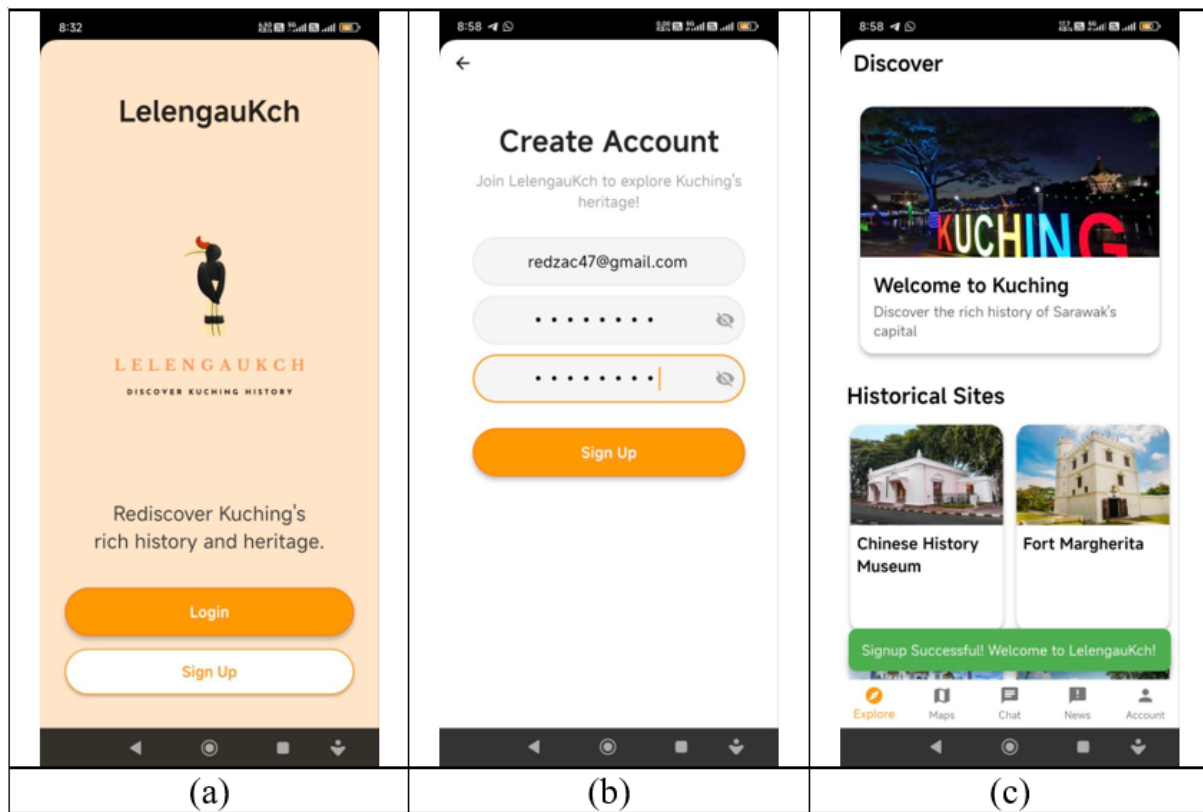


Figure 63. User account registration

User registration in the LelengauKch application is designed to be straightforward, supported by secure authentication and robust role management. Based on Figure 63(a), users begin their journey on the landing screen, which presents a welcoming interface with two prominent options: "Login" for returning users and "Sign Up" for new users. When first-time users select the "Sign Up" button, they are directed to the registration interface as depicted in Figure 63(b) where they must provide their email address, create a password, and confirm that password. The system incorporates validation checks that ensure passwords match, contain at least six characters, and that the email follows a valid format. Upon clicking the "Sign Up" button, the application communicates with Firebase Authentication to create the user account while simultaneously generating a user document in Cloud Firestore with default attributes: role set

to "user," status set to "active," and a timestamp of creation. After successful registration, the system automatically displays a success message. The user is then redirected back to the home page where they can access other modules as shown in Figure 63(c). Once logged in, users can access their account page to check their email verification status, request for sending the verification email, or request a password reset. The authentication system also performs security checks during login to prevent banned or inactive accounts from accessing the application, thereby maintaining system integrity and security.

4.4.2 Account Login

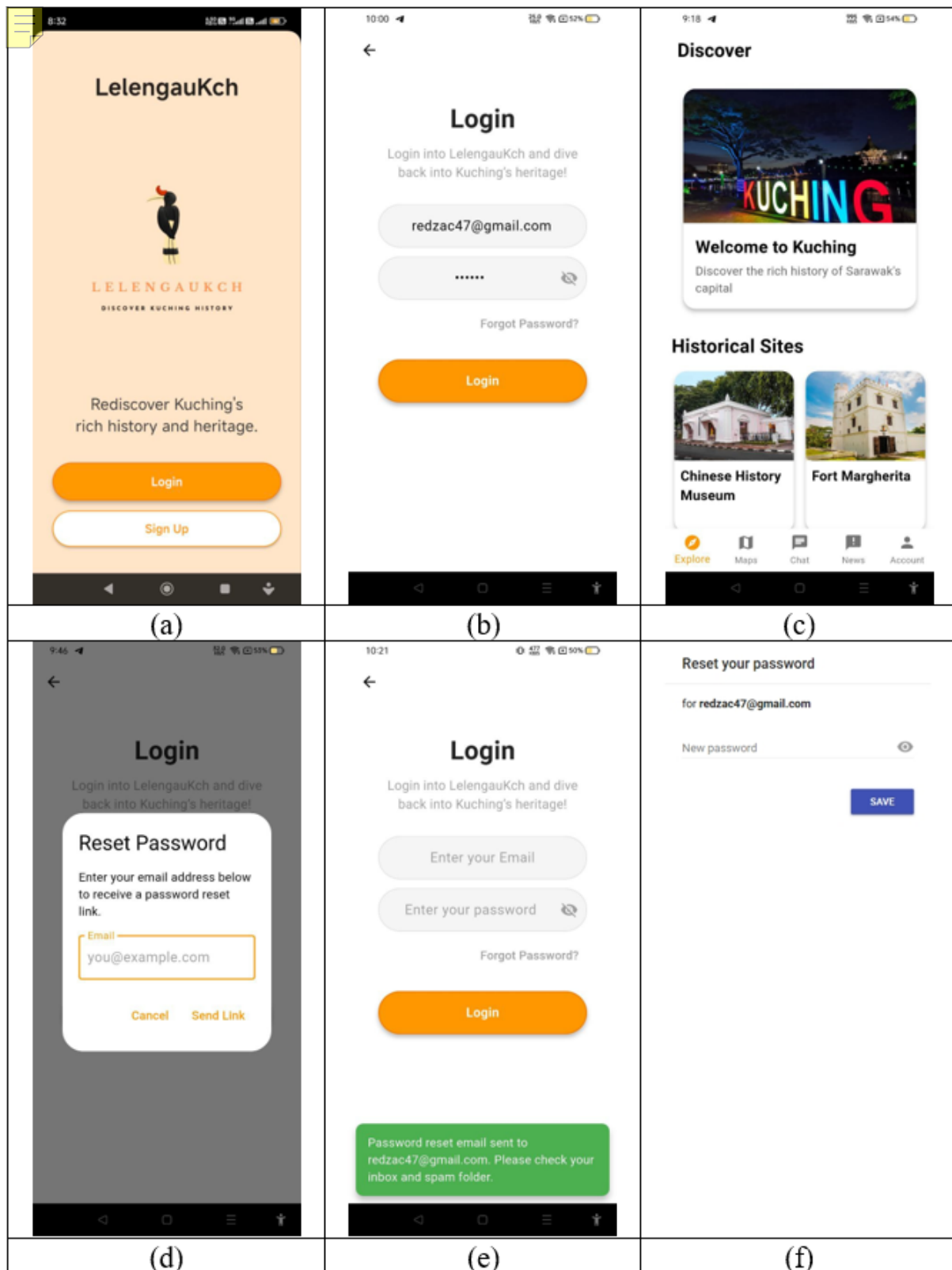


Figure 64. Account login for users for LelengauKch

The LelengauKch application implements a comprehensive user login process with secure authentication and role-based access control. Users begin on the landing screen illustrated in Figure 64(a), which presents a welcoming interface with two distinct options: "Login" for returning users and "Sign Up" for first-time users. When users select the "Login" button, they are directed to the login interface in Figure 64(b) where they must provide their registered email address and password. The system incorporates multiple validation checks to ensure fields are not empty, the password contains at least six characters, and the email follows a valid format using regular expression pattern matching. Upon clicking the "Login" button, the application connects to Firebase Authentication to verify the credentials while displaying a loading indicator to inform users that the process is underway. After successful authentication, the system performs additional security checks by querying the user's document in Cloud Firestore to verify their account status. If the account status is "banned" or "inactive," access is denied with appropriate feedback messages. For valid active accounts, regular users are directed to the main application homepage as presented in Figure 64(c) with access to historical sites, maps, chat, and news features. Additionally, according to Figure 64(d), the login interface provides a "Forgot Password" feature that allows users to request a password reset email by entering their registered email address. After requesting the password reset email, a notification will be displayed confirming the request as illustrated in Figure 64(e). Users can proceed to check for the password reset email in inbox or spam and follow the link to reset the account password based on Figure 64(f). This comprehensive login system ensures secure access while providing appropriate user feedback at each step of the authentication process.

4.4.3 View Historical Sites List

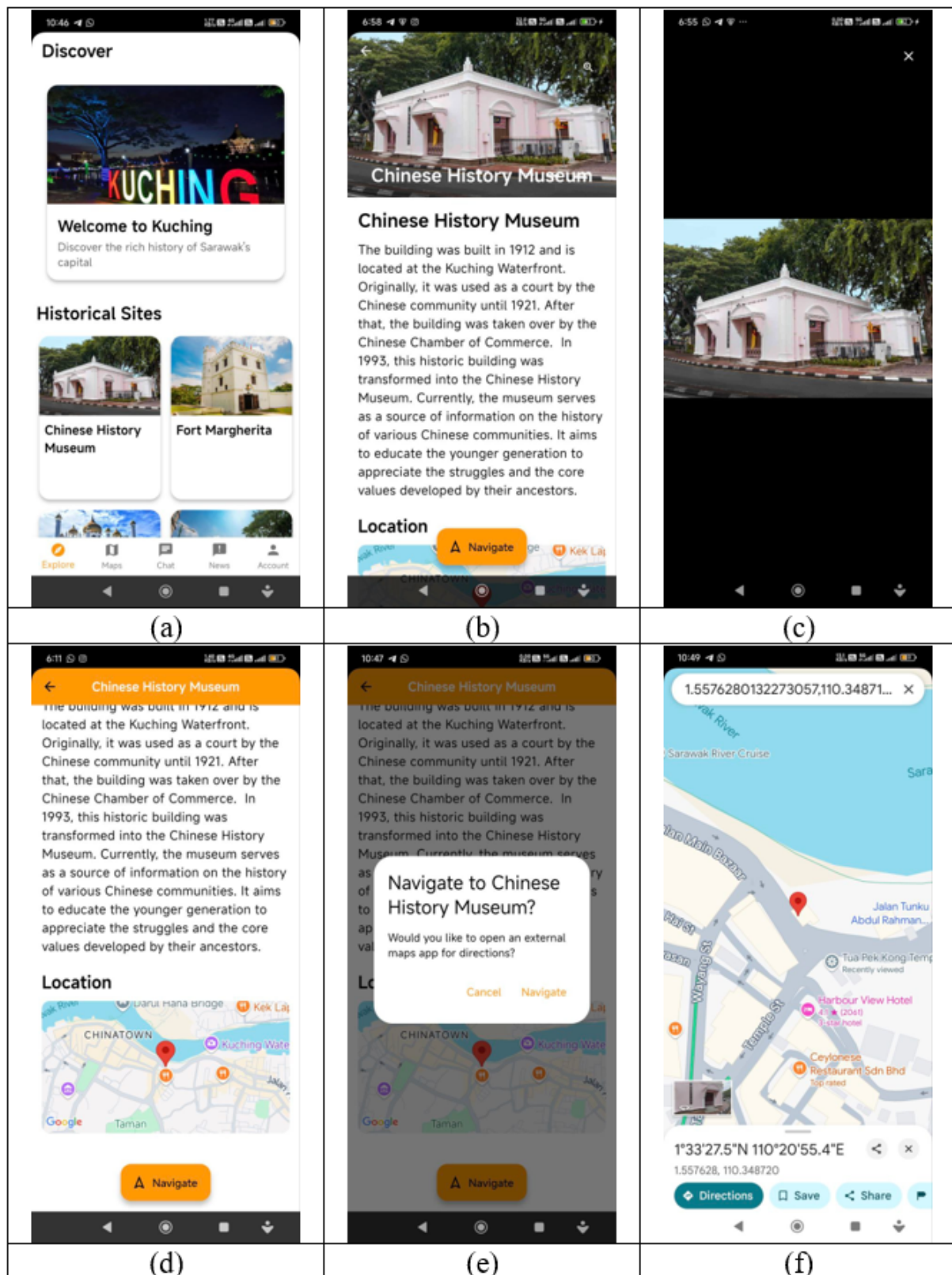


Figure 65. View historical site attraction list user interface

Users of LelengauKch application can easily explore and navigate to the selected historical site. Upon logging in, users are presented with a list of historical sites displayed as interactive cards organised in grid view on the home page shown in Figure 65(a). Each card features the name and image of a site, such as the Chinese History Museum. When a user taps on a site card, the application transitions to a detailed screen dedicated to that site as observed in Figure 65(b). This detail screen presents comprehensive information, including an extended description, a gallery image of the historical site, and an embedded map highlighting the exact location of the site. Users can tap on the image to enter fullscreen image viewer that allow them to zoom onto the image for clear observation as depicted in Figure 65(c). To facilitate real-world exploration, a floating action button labeled "Navigate" is prominently displayed. Based on Figure 65(d), when the user taps this button, the application prompts for confirmation in Figure 65(e) and, upon approval, seamlessly launches the native Google Maps application with the destination coordinates pre-filled as presented in Figure 65(f). This enables users to receive turn-by-turn navigation from their current location directly to the selected historical site, enhancing both the educational and experiential aspects of heritage exploration.

4.4.4 Search for historical site in map

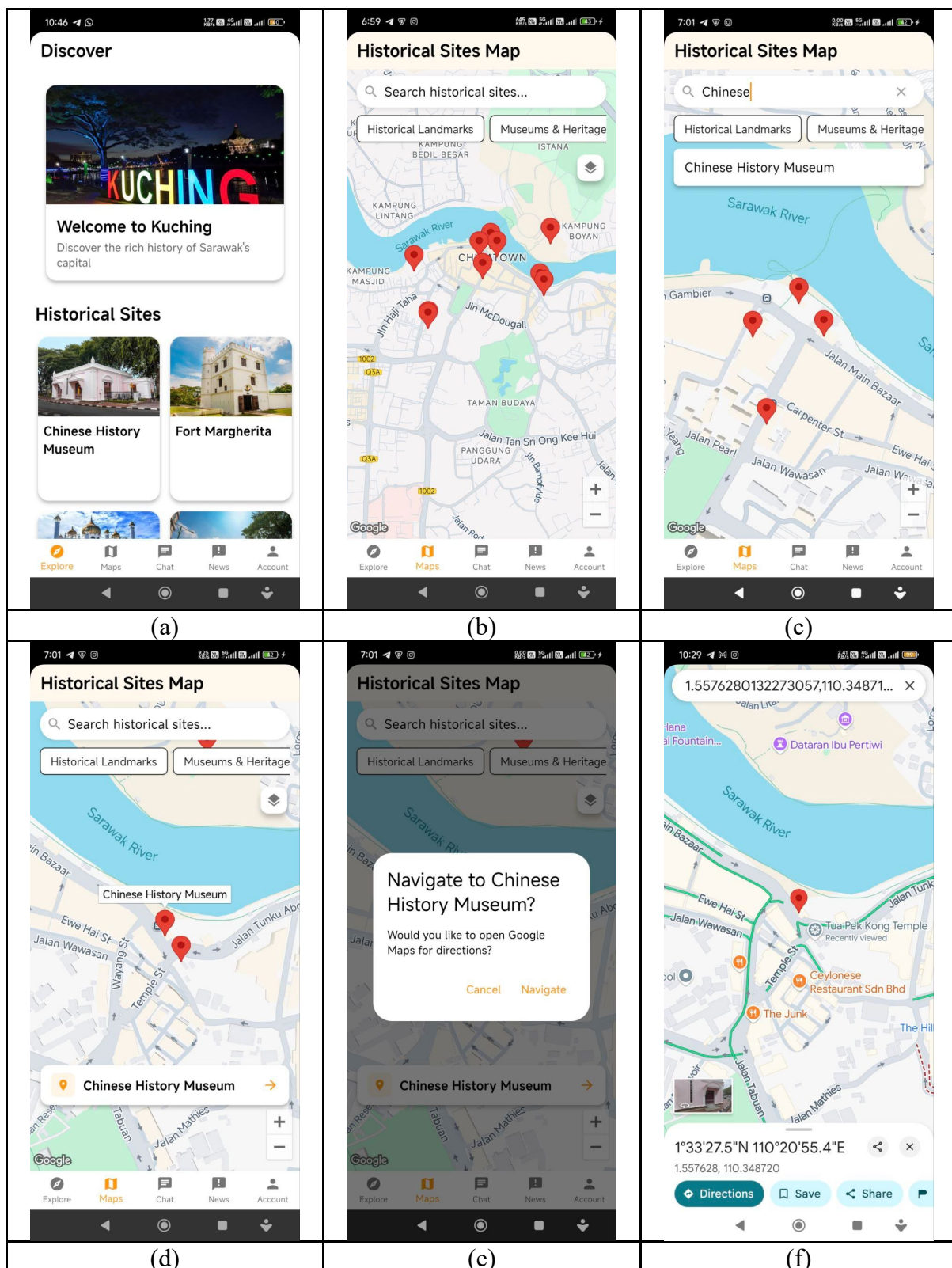


Figure 66. Search for historical site user interface

The efficient identification of historical sites in Kuching by general users is supported through an interactive mapping system integrated into the LelengauKch application. Upon accessing the map interface, users are provided with a search bar positioned at the top of the screen, enabling them to input keywords associated with the name or category of a specific historical site as observed in Figure 66(b). As users enter search terms, the application dynamically refines the available options, displaying relevant results in a list beneath the search bar according to Figure 66(c). Based on Figure 66(d), upon selecting a site from the search results, the map automatically centers on the corresponding location, marks the site, and presents a bottom card containing the site's name and a navigation feature. Tapping the introductory card triggers a navigation prompt, requesting user confirmation to proceed. Upon confirmation, Google Maps is launched externally for navigation, as illustrated in Figures 66(e) and 66(f). This integration of search functionality with spatial visualization enhances user accessibility and interaction, thereby streamlining the process of locating, identifying, and navigating to historical sites directly from the map interface.

4.4.5 Using quick action filter in map

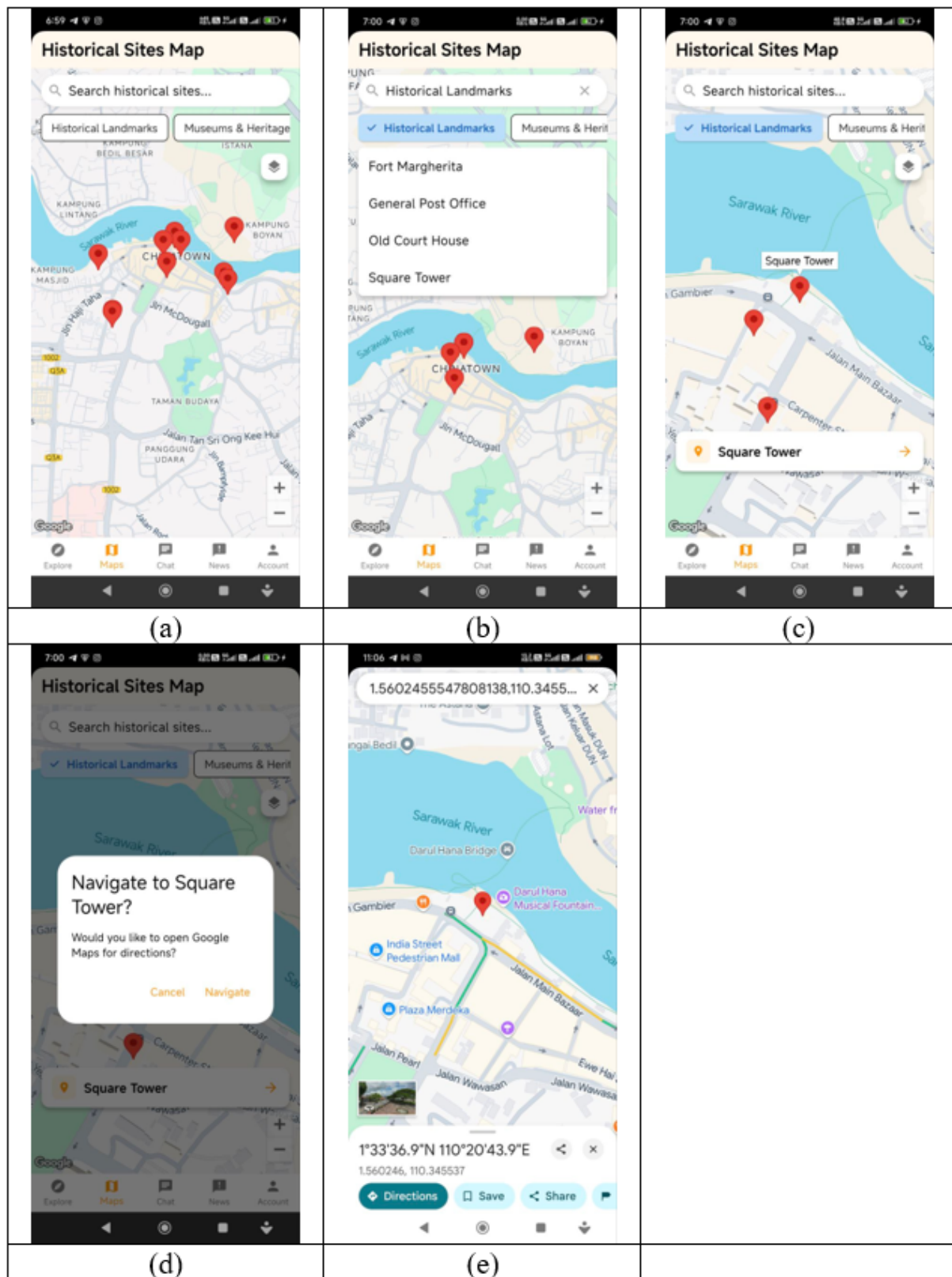


Figure 67. Locate and navigating to historical sites using quick action filter

LelengauKch mobile application integrates a convenient filtering mechanism to facilitate efficient site discovery within its map interface. This system enables public users to locate specific historical sites, such as the Fort Margherita, by utilising category-based quick action filters. Referencing to Figure 67(a), positioned below the search bar, a horizontally scrollable row of predefined filter chips provides distinct site classifications, including "Historical Landmarks," "Museums & Heritage," and "Architectural & Cultural Sites.". As shown in Figure 67(b), upon selecting the "Historical Landmarks" filter chip, the application automatically refines the displayed search results by populating the search bar with the corresponding category and retrieving relevant site entries. Simultaneously, the map dynamically adjusts to display only markers corresponding to the filtered category. In Figure 67(c), Fort Margherita is appropriately classified as a historical landmark, and its marker becomes visible on the map and appears in the search results list. Users can interact with these filtered results by either selecting a site from the list or tapping its marker to access site name and navigation options via the bottom information card. Expanding on this functionality, Figures 66(d) and 66(e) illustrate that tapping the introductory card triggers a navigation prompt requesting user confirmation. Upon confirmation, Google Maps is launched externally for navigation. This seamless integration of filtering and mapping functionalities enhances user accessibility, streamlining the process of site identification and exploration.

4.4.6 Initiate Chat

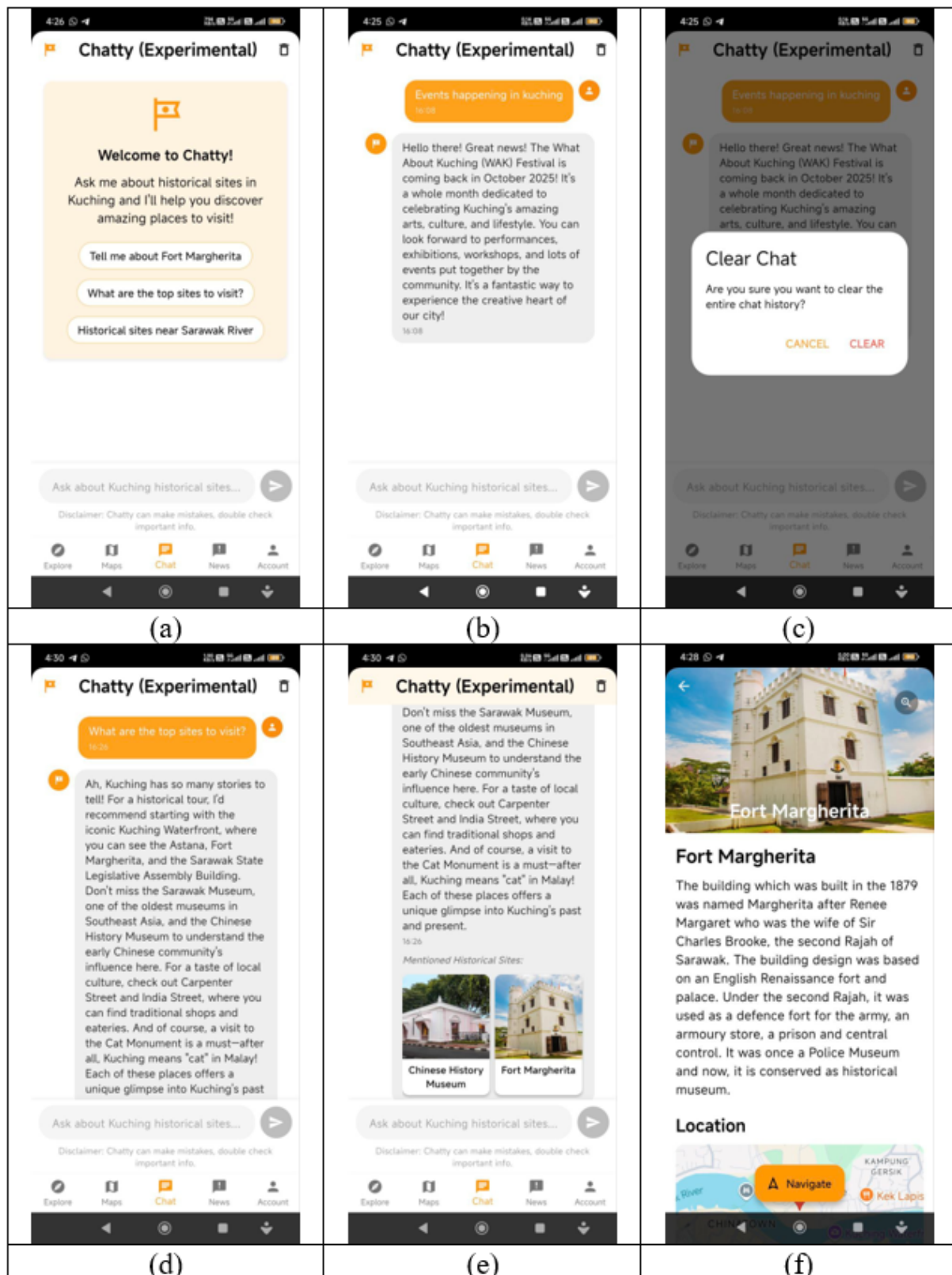


Figure 68. User interface for chat module allows user to query details in conversational format

An interactive chat module, designed to facilitate user engagement with an AI assistant named "Chatty" is available in the application as illustrated in Figure 68(a). Upon the user's initial entry into the chat interface, the system presents a welcoming environment, featuring an introductory message alongside a set of predefined conversational prompts. These prompts, displayed as interactive options, enable the user to easily initiate a conversation by selecting a topic of potential interest. These initial suggestions are designed to guide users and enhance their experience by providing a smooth introduction to the AI's capabilities as presented in Figure 68(d) and Figure 68(e). If the AI assistant references content relevant to historical sites or news items available within the LelengauKch application, it generates a response featuring interactive cards, as illustrated in Figure 68(e) and Figure 68(f). These cards include image previews and titles of the referenced sites or news, enabling users to access a more detailed view by tapping on the respective card. This design enhances user engagement by seamlessly integrating visual content with interactive characteristic.

In addition to the suggested prompts, users are also provided with the opportunity to start a conversation by directly typing into a dedicated input area shown in Figure 68(b). The input area encourages users to enter custom queries, with an accompanying disclaimer notice reminding them that "Chatty" may make occasional errors, hence users should verify the responses. Resetting a conversation can improve usability, especially for users who wish to start a new discussion without interference from previous messages as seen in Figure 68(c). Once the user submits their query, the system processes the input and generates a relevant response from the AI assistant. This two-pronged approach—predefined suggestions and custom queries—ensures that users can engage with the application in a manner that suits their preference, whether they seek a guided experience or wish to explore specific topics on their own.

4.4.7 View News List

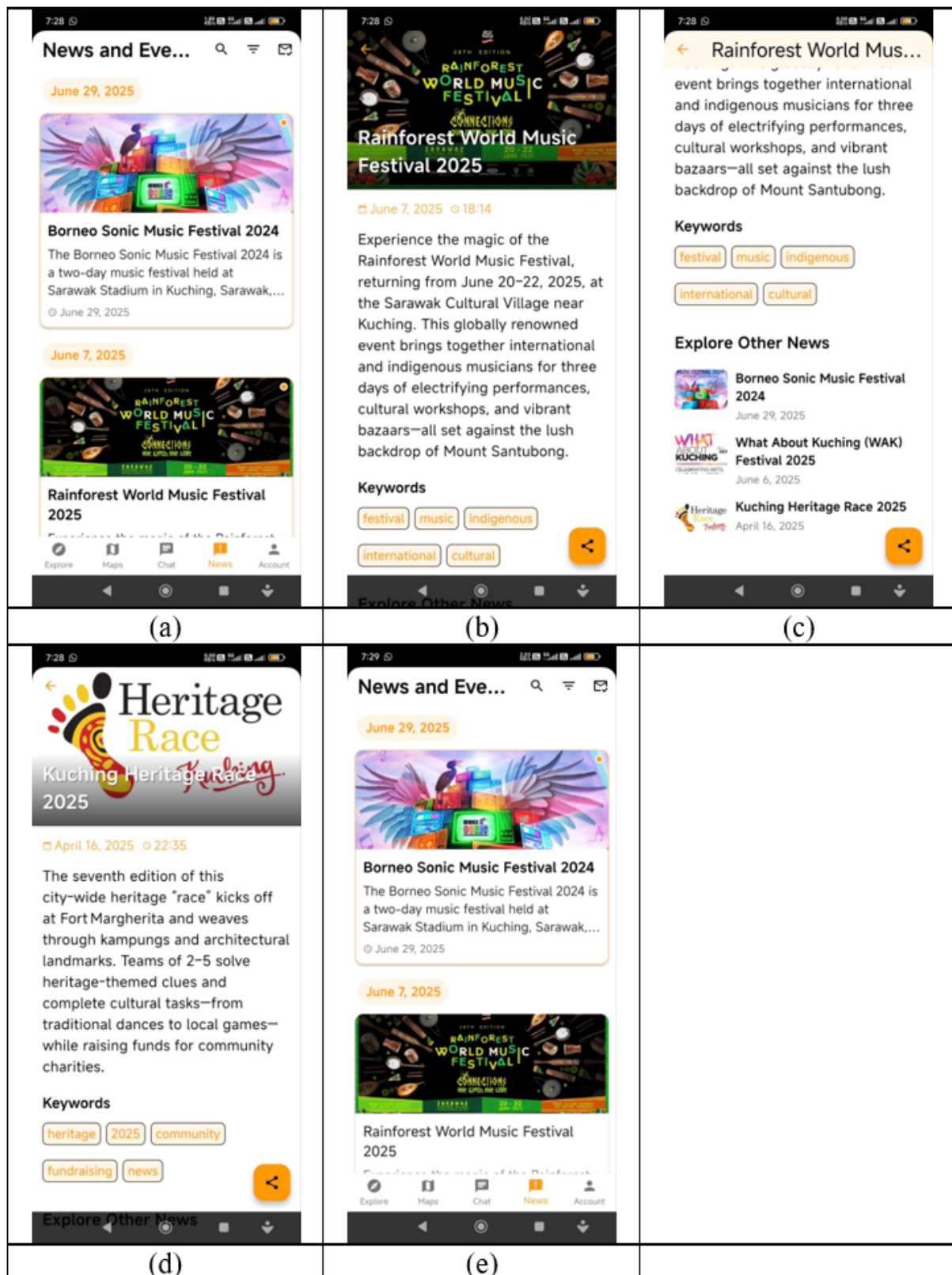


Figure 69. View list of news feed and explore other news

The LelengauKch app provides a smooth and efficient way to view list of news and events, enabling users to browse articles and uncover related content effortlessly. On the primary news interface shown in Figure 69(a), users are presented with a list of news and events stories, including one titled "Rainforest World Music Festival 2025," displayed in engaging interactive tiles. By selecting an article, users are directed to a comprehensive view as shown in Figure 69(b) that showcases the title, description, image, and publication date. As users access the detailed article view, the application automatically marks the article as read. This change in status is visually indicated, such as by altering the tile's border colour and eliminating any unread marker as indicated in an orange dot at top right of the tile according to Figure 69(e). This clearly shows the user that they have engaged with the content. Within the detailed view, there is a section labelled "Explore Other News" as depicted in Figure 69(c), which features a selection of three other articles. If a user chooses another piece, such as "Kuching Heritage Run 2025" illustrated in Figure 69(d), the app promptly loads the complete details of the new article. The system also updates the read status of this newly accessed article, ensuring that the user's reading history is accurately recorded and displayed. This fluid interaction between articles allows users to navigate seamlessly through interconnected content, all while enjoying a well-organized and tailored news browsing experience.

4.4.8 Search for News

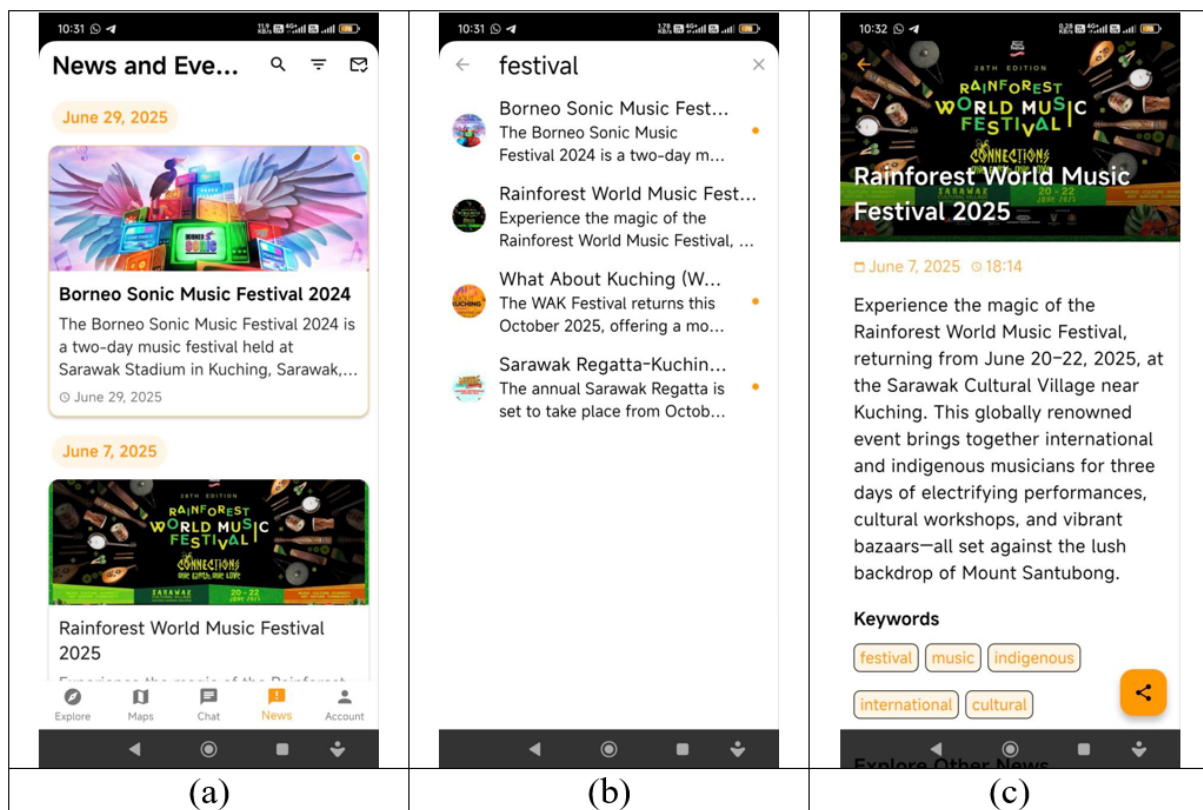


Figure 70. User interface for searching and viewing news.

From the main news interface visualised in Figure 70(a), users can initiate a search function via an accessible search icon, which presents an input field for entering keywords (for instance, "festival"). Based on Figure 70(b), as the user types, the application dynamically filters the list of available articles based on the query, comparing it against both titles and descriptions. For instance, the article "Rainforest World Music Festival 2025" matches the entered term, and it is included in the search results. When the user selects a specific article from the results, the application transitions to a detailed view of the selected content as observed in Figure 70(c). If the article has not been read previously, the system updates its status accordingly to reflect that it has now been accessed. This functionality supports real-time tracking of the user's reading activity. The combination of intuitive search, real-time feedback, and content presentation contributes to a streamlined user experience within the application.

4.4.9 News and Events Sharing

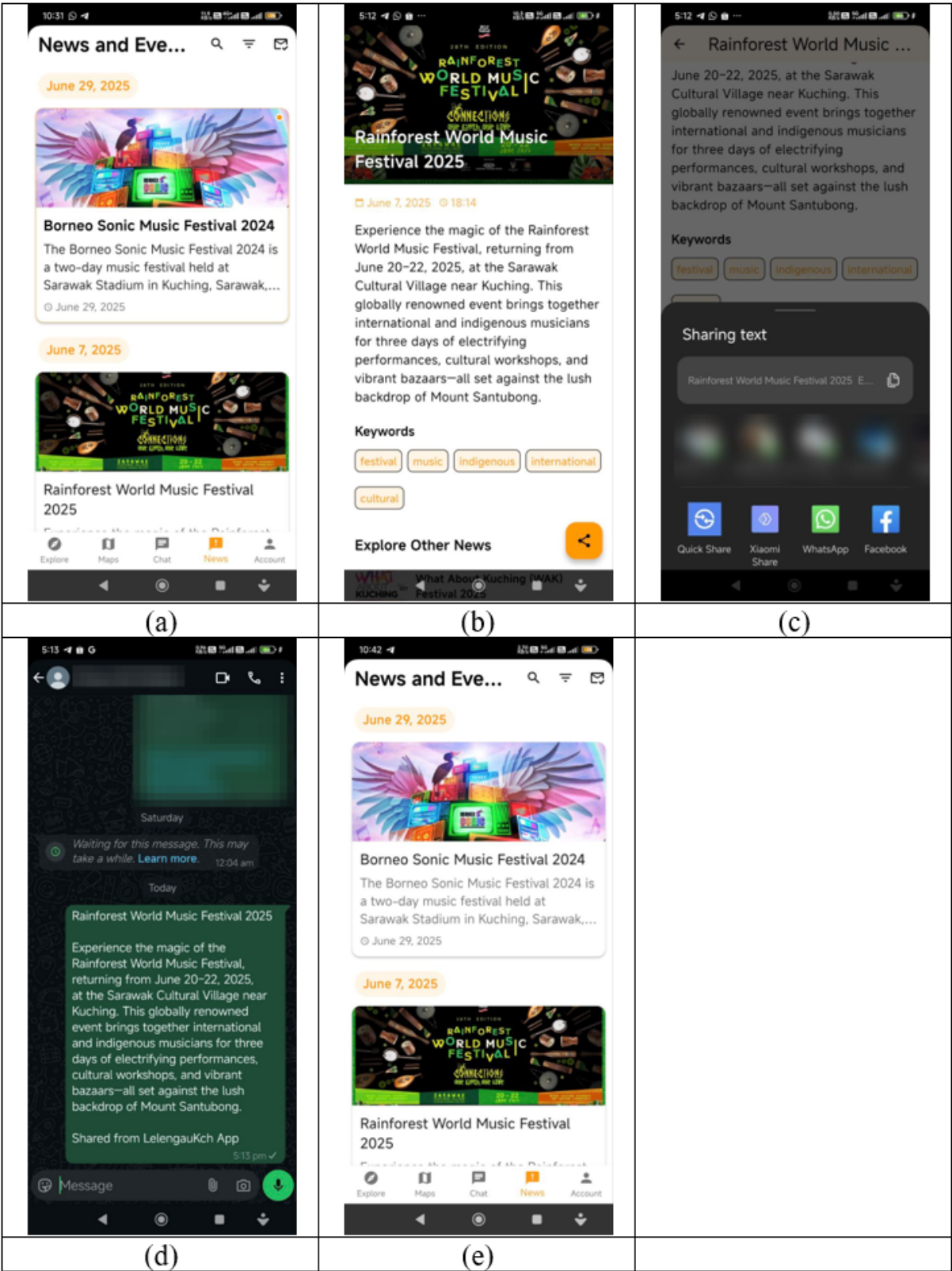


Figure 71. User interface demonstrating the workflow for sharing a news article to a third-party application.

Within the LelengauKch application, users can access a curated list of news articles from the main news interface displayed in Figure 71(a). Each item, such as "Rainforest World Music Festival 2025," is displayed in a structured format that encourages browsing and engagement. Selecting an article navigates the user to a detailed view, where the full content—including the title, accompanying image (if available), description, and publication date—is presented in an organized and reader-friendly layout as represented in Figure 71(b). The detailed view also features a persistent floating action button that enables users to share articles directly from the application. When triggered, this feature launches the device's native sharing interface, allowing users to select from various installed applications capable of handling text content, such as messaging services, email clients, or social media platforms. For example, as observed in Figure 71(c) and Figure 71(d), if the user chooses to share via a messaging application such as WhatsApp, a pre-composed message is generated containing the article's title, description, and an attribution line such as "Shared from the LelengauKch app." The user can then complete the sharing process by selecting a recipient and sending the message, effectively disseminating the content beyond the application environment. Following this, the application updates its status to reflect that it has been accessed and viewed by the user, as illustrated in Figure 71(e). This sharing functionality enhances user engagement by facilitating the distribution of cultural and community-related information through familiar digital channels, thereby extending the reach of the content provided by the LelengauKch platform.

4.4.10 Participate in Quiz

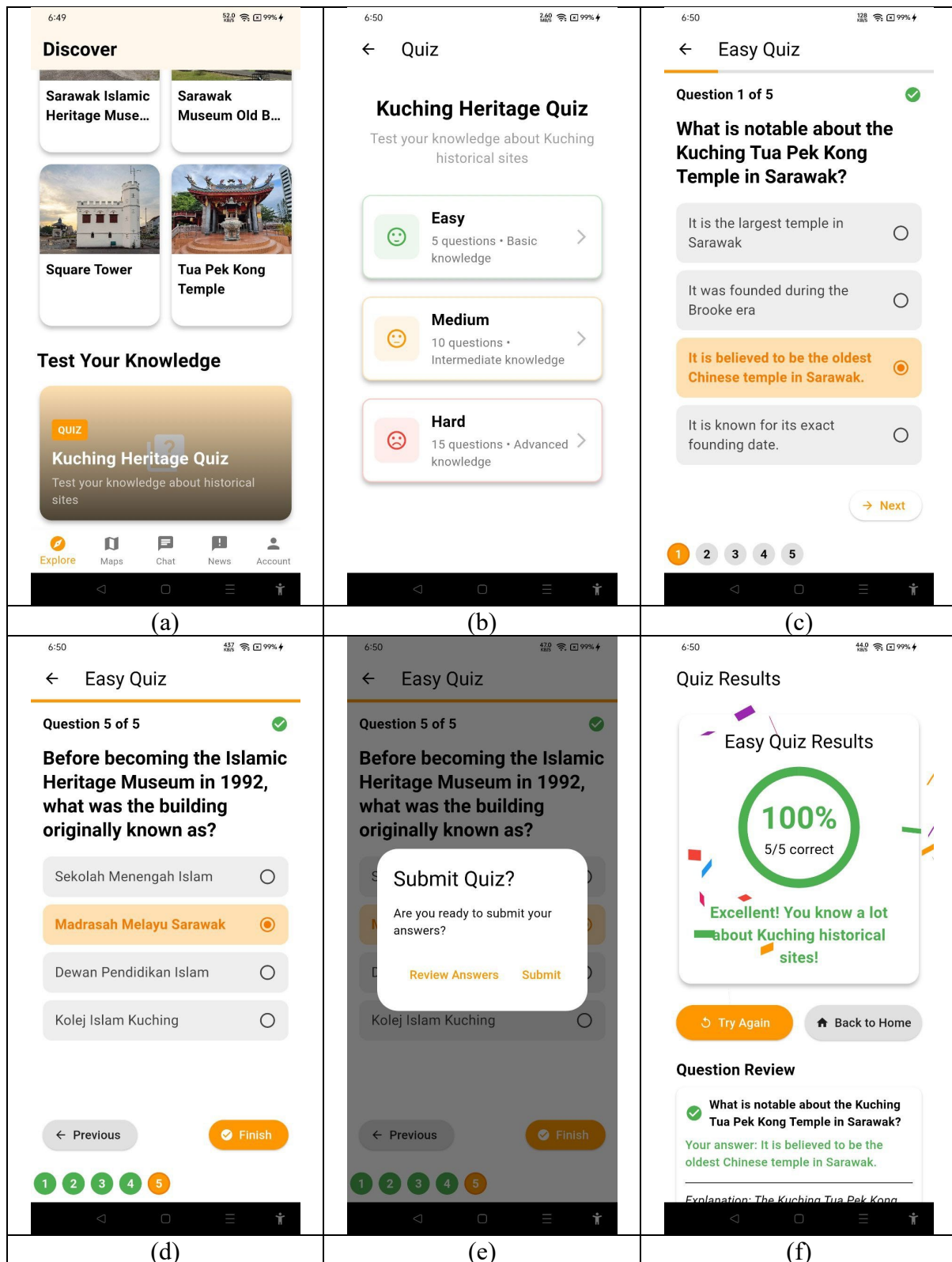


Figure 72. User participate in quiz module in LelengauKch

To make learning both engaging and enjoyable, the LelengauKch application incorporates an interactive quiz module that is accessible from Explore section as shown in Figure 72(a). Figure 72(b) illustrates that the module begins by presenting users with three distinct difficulty levels including "Easy", "Medium", and "Hard." These levels are clearly defined, with each offering a set of challenges that vary in terms of the total number of questions presented which are 5, 10 and 15 respectively. Upon selecting a difficulty level, for instance easy difficulty, users are guided through a series of questions that match the corresponding chosen level of difficulty, enhancing the personalized nature of the quiz experience.

The quiz interface is designed to be user-friendly and intuitive. Once a user selects their desired difficulty level, they are presented with a sequence of questions as presented in Figure 72(c), which are accompanied by multiple-choice answer options. As depicted in Figure 72(d), a progress bar and a question counter are prominently displayed to help users track their progress. The questions themselves are presented in clear, readable text, often supplemented with images to enhance user engagement and understanding.

As users work through the questions, their answers are recorded, and they have the ability to navigate through the quiz by moving forward or backward between questions. As observed in Figure 72(e), upon reaching the final question, the user is prompted to submit their answers. The submission process includes a confirmation step, ensuring that users are confident in their responses before finalizing their results.

As shown in Figure 72(f), After the quiz is submitted, users are presented with a summary of their performance. This includes a score displayed as a percentage, with visual feedback indicating their performance level. High achievers receive positive reinforcement through contextual feedback and celebratory animations. Additionally, users are given the opportunity to review their responses, compare them with the correct answers, and reflect on any explanations provided, thus reinforcing the learning experience. Overall, the quiz module

serves not only as a means of testing users' knowledge but also as an interactive tool for learning and improvement, providing both feedback and opportunities for further engagement.

4.4.10 View Quiz Explanation and History

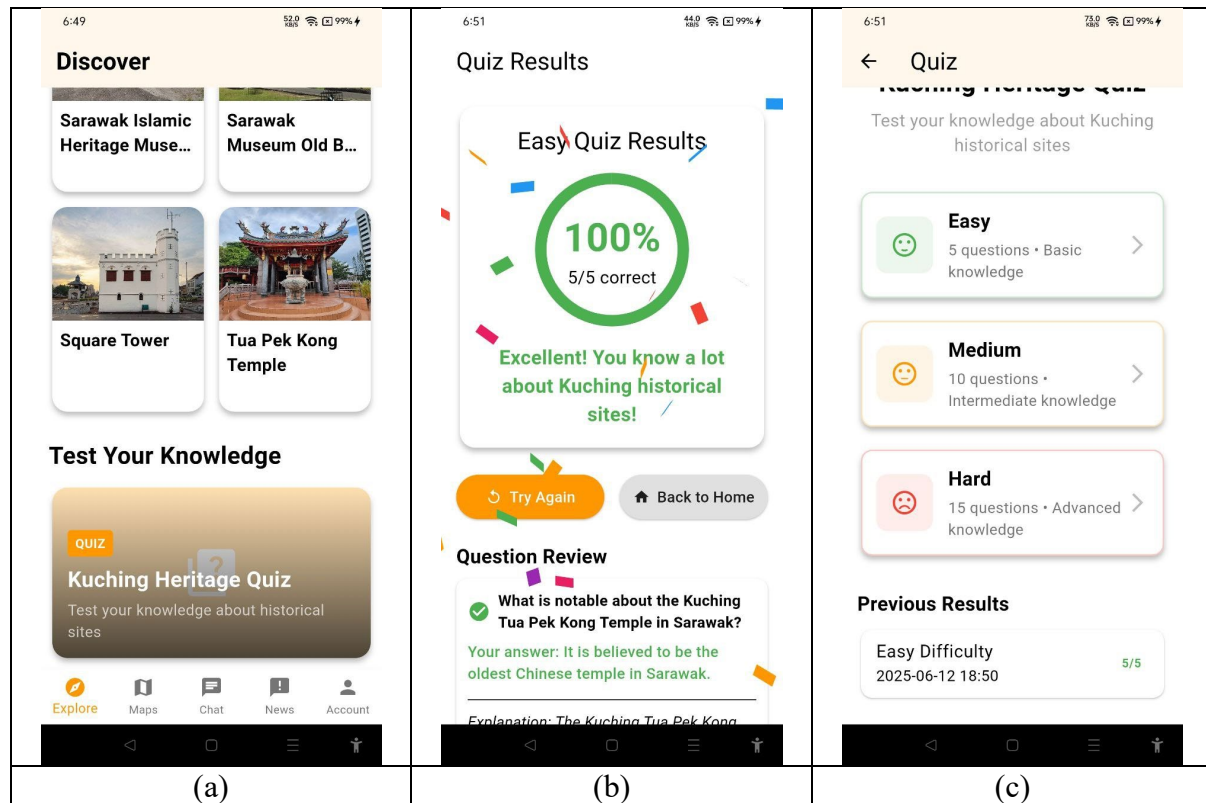


Figure 73. User can view quiz explanation and history

Figure 73 demonstrates the implementation of the quiz explanation and history viewing features within the LelengauKch mobile application. As illustrated in subfigure (a), users can access the "Kuching Heritage Quiz" from the home interface under the "Test Your Knowledge" section, which is designed to assess and reinforce users' knowledge of historical sites in Kuching. Upon completion of a quiz, the application presents the results interface as shown in subfigure (b), where the user's performance is displayed through a percentage score, accompanied by a feedback message to encourage continued engagement. Moreover, the application provides a detailed review of each quiz question, indicating the selected answer along with an explanatory note to enhance the user's understanding. Subfigure (c) depicts the quiz level selection interface, offering three difficulty tiers: Easy, Medium, and Hard.

Additionally, users are able to view their quiz history, including the date, time, and score of each attempt. This functionality supports iterative learning and allows users to monitor their progress over time.

4.4.11 View Account Details

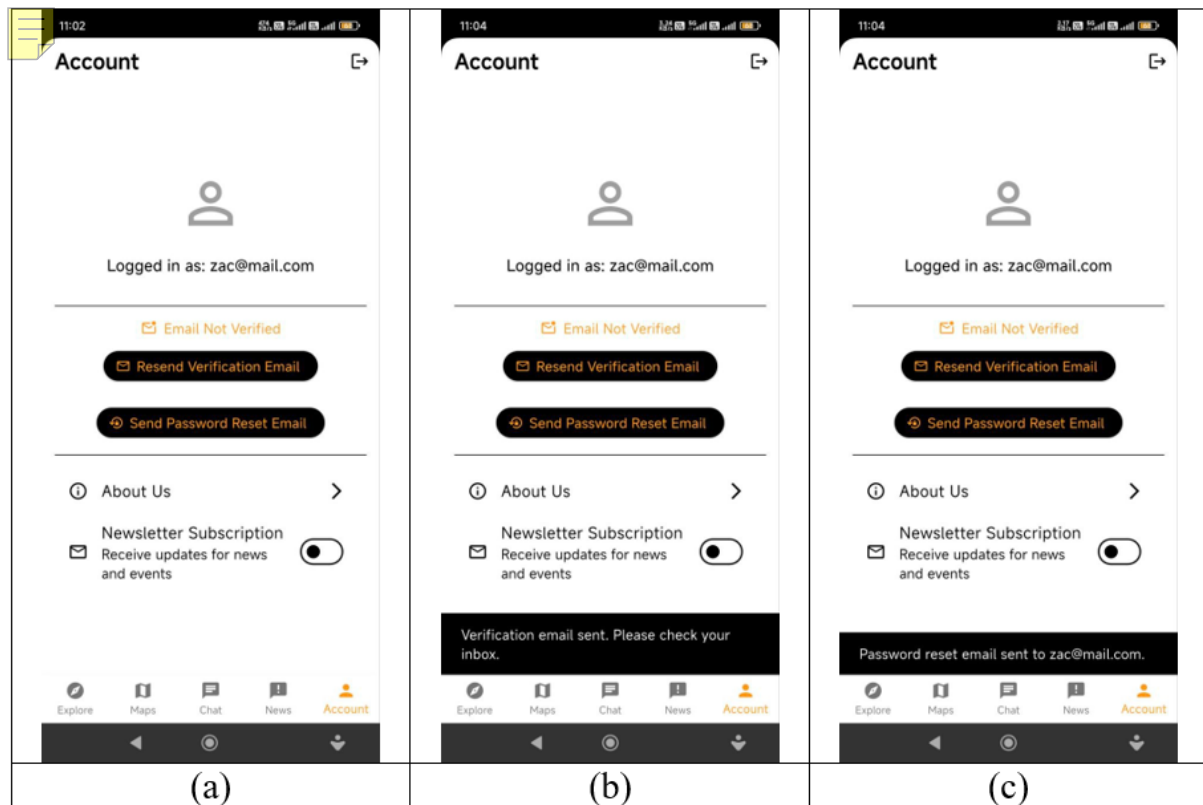


Figure 74. Features for Account Page

Figure 74 shows the dynamic user interface (UI) behavior of the “Account” section within the LelengauKch mobile application. The figure is segmented into three subfigures (a–c), each representing a distinct system state resulting from user interaction. Each screenshot displays a different interaction state related to email verification and password reset functionalities. In subfigure (a), the initial state of the interface is shown, where the user is logged in with the email address of current account. A notification indicates that the email is not verified using coloured warning label, and two interactive buttons are provided which are “Resend Verification Email” and “Send Password Reset Email”. This design adheres to usability best practices by clearly presenting critical account status information and offering immediate

action options to the user. Below these buttons, additional features such as “About Us” and “Newsletter Subscription” are available.

Figure (b) depicts the interface state after the user triggers the "Resend Verification Email" function. After a brief loading period indicated by the spinning circle in the buttons after click, a confirmation message appears at the bottom of the screen, reading "Verification email sent. Please check your inbox." This message provides user feedback indicating that the email verification process was successfully initiated. This feedback mechanism plays a vital role in improving user confidence and experience by affirming that the requested action has been successfully executed. From a human-computer interaction (HCI) perspective, the inclusion of real-time confirmation messages aligns with Jakob Nielsen's Heuristics for User Interface Design, specifically addressing the principle of visibility of system status.

Subfigure (c) displays the state after selecting the "Send Password Reset Email" button. A similar confirmation message appears at the bottom, indicating that a password reset email has been sent to the user's email address. The consistent use of positional feedback (bottom popup messages) across different functions ensures a uniform user experience and reinforces learnability. Overall, this UI sequence effectively demonstrates the feedback mechanisms incorporated into the system to guide users through essential account management tasks. By integrating these feedback components into essential functions like email verification and password recovery, the system minimises user confusion.

4.4.12 View About Us Page

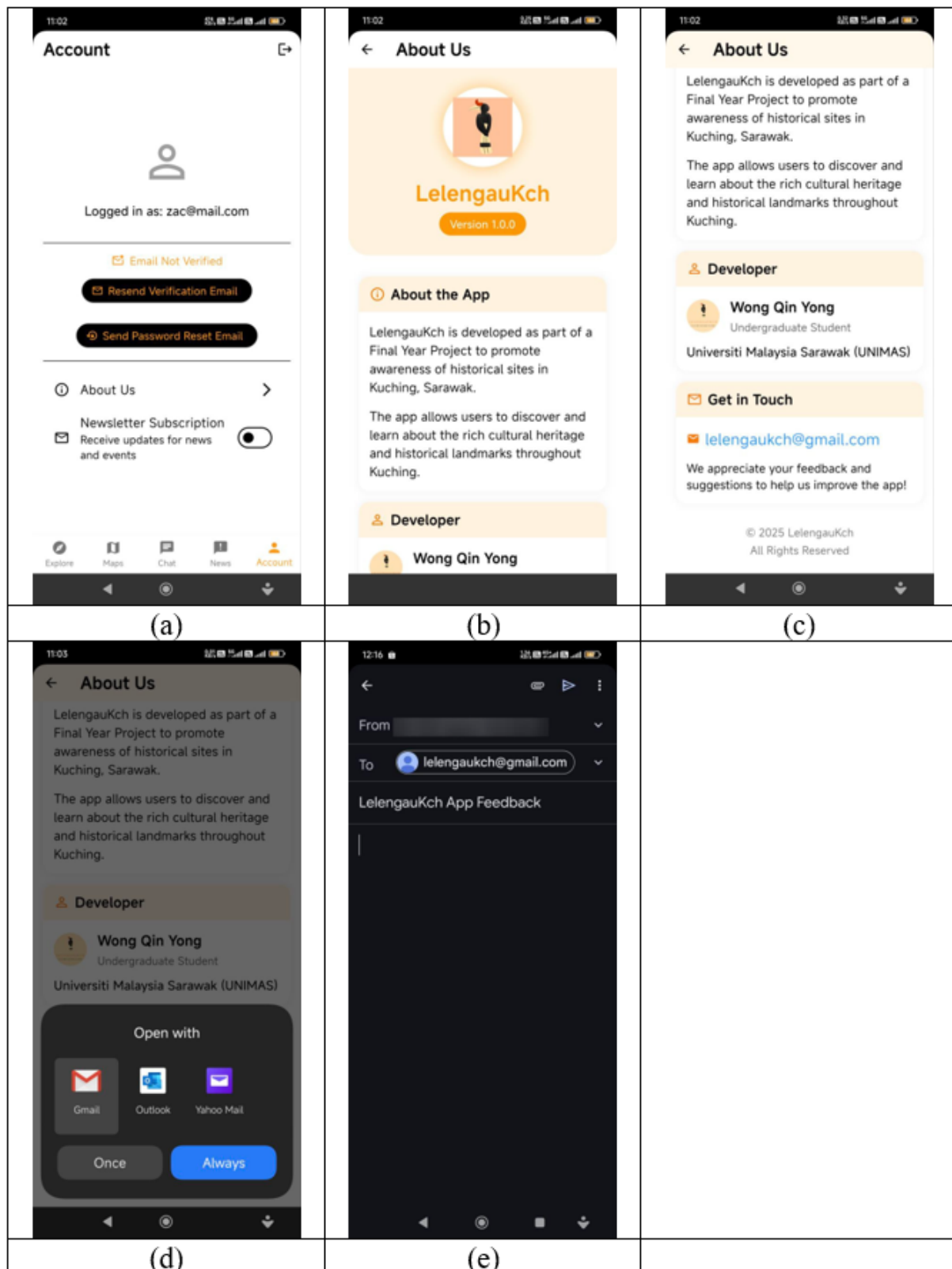


Figure 75. The "About Us" page accessible from the Account section provides users with information about the application and its developer.

Figure 75 presents a sequential workflow depicting the navigation process of the 'About Us' page. Each subfigure (a–d) illustrates a distinct stage in the user’s interaction, beginning with the account page and progressing through to the final step, which prompts the user to submit feedback via an external emailing application.

In subfigure (a), the user interface displays the “Account” page, where a button labeled “About Us” is available. When the user selects this option, the application navigates to the dedicated “About Us” section, as shown in subfigure (b). This screen prominently features the application’s icon and version number, followed by a brief introduction about the app. The descriptive text informs users that LelengauKch is a Final Year Project aimed at promoting awareness of historical sites in Kuching, Sarawak. Additionally, a labeled section highlights the developer’s identity. Subfigure (c) builds upon the information displayed in the preceding screen by showing a “Get in Touch” section is included, which provides the contact email address (lelengaukch@gmail.com) to encourage user feedback and suggestions. At the bottom of the page, a copyright notice asserts that all rights are reserved by the application.

Subfigure (d) illustrates the interface state following user interaction with the contact email link. This action initiates an operating system–level prompt that presents several email client options that user had installed on the device such as Gmail, Outlook, and Yahoo Mail, thereby enabling the user to select their preferred communication platform as observed in subfigure (e). This integration with external email services not only facilitates feedback submission but also enhances overall user engagement and system responsiveness.

4.4.13 Manage Newsletter Email Subscription

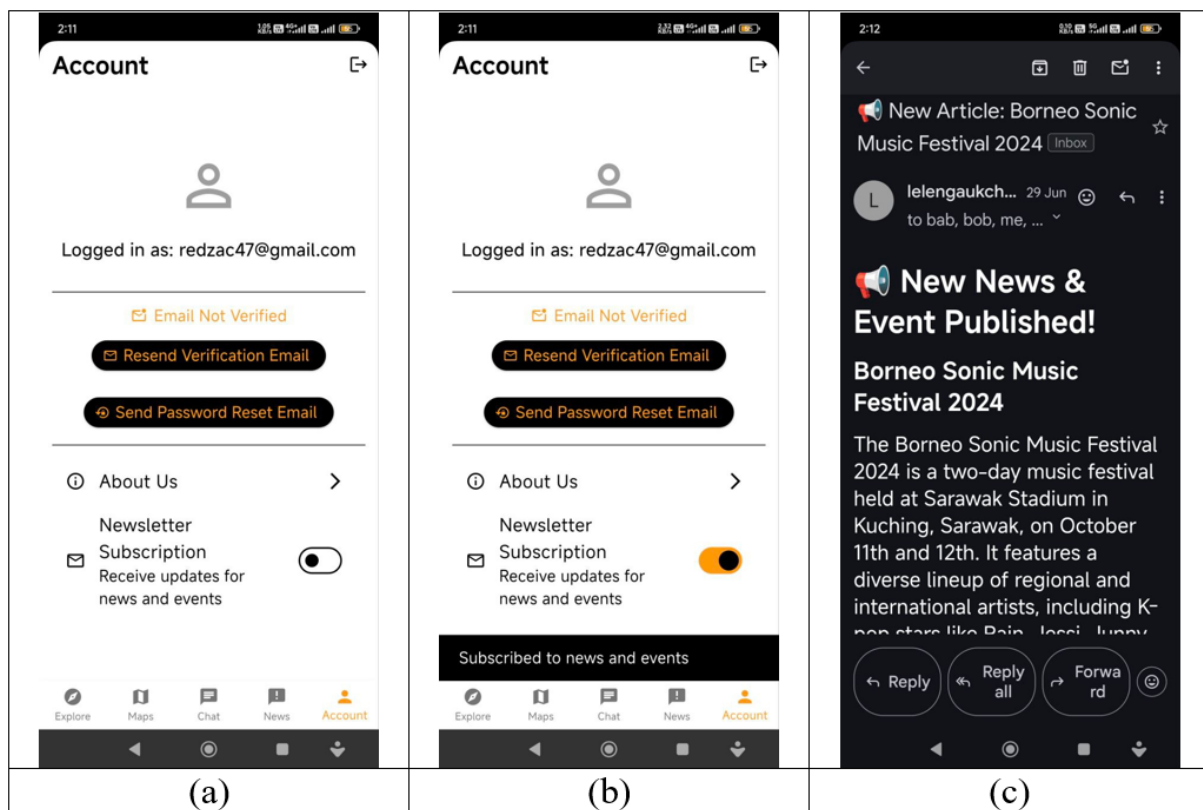


Figure 76. Sequence illustrating the newsletter subscription flow and event notification via email alerts

Figure 76 illustrates the end-to-end process by which a user subscribes to the in-app newsletter and subsequently receives an news and event notification via e-mail. In subfigure (a), the “Account” page displays the newsletter subscription control in its default, inactive state: the toggle is positioned to the left and accompanied by the label “Receive updates for news and events.” At this stage, no subscription confirmation is present, indicating that the user has not yet opted in. Upon tapping the toggle, subfigure (b) shows the switch moved to the “on” position and a transient banner message of “Subscribed to news and events” is rendered at the bottom of the screen to acknowledge the successful registration of the user’s interest. Finally, subfigure (c) captures the external outcome of this interaction which is the user’s e-mail client displaying a newly delivered message entitled “New News & Event Published! Borneo Sonic Music Festival 2024.” The body of the e-mail provides concise event details (dates, venue, and

lineup), thereby closing the loop between the in-app subscription mechanism and the out-of-app notification channel. This sequence demonstrates the system's ability to transform a simple UI toggle into a tangible informational alert, reinforcing user engagement through timely communication.

4.4.14 Manage Historical Sites (Admin)

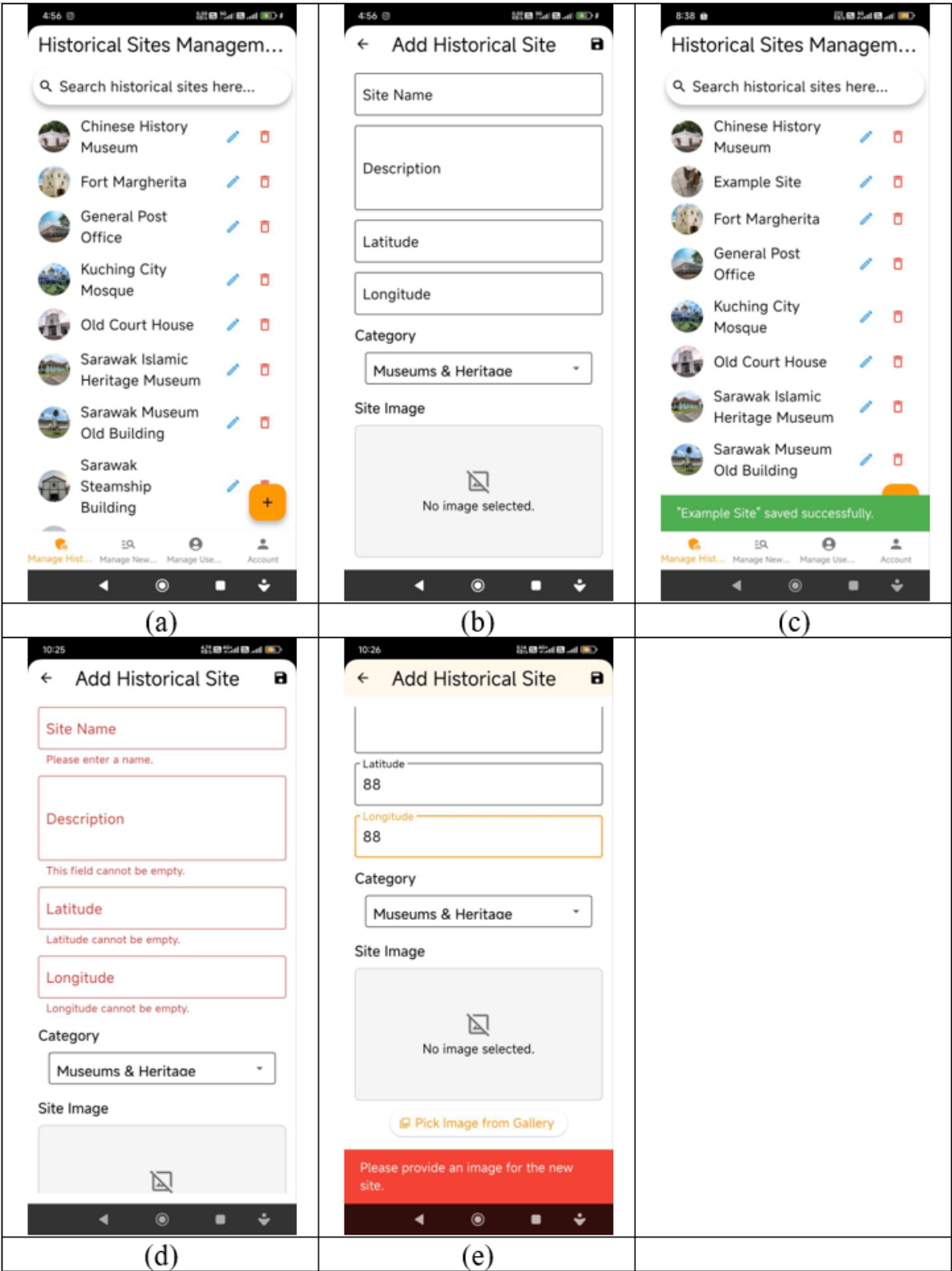


Figure 77. Workflow for admin adding a new historical site within the management module

Figure 77 illustrates the complete process for adding a new historical site, from the initial list view to error handling during data input. Subfigure (a) displays the main "Historical Sites Management" page, showing a list of existing sites with options to search, view details, or edit/delete entries. This serves as the starting point for adding new data. Subfigure (b) presents the "Add Historical Site" form, which is accessed by tapping the '+' button in subfigure (a). This form requires input for site name, description, latitude, longitude, and category, and allows for an image to be selected. Subfigure (c) shows the successful addition of a new site, indicated by a "Example site saved successfully" message shown at the bottom of the screen, and the updated list now includes the newly added entry. Subfigures (d) and (e) demonstrate the system's defensive approach to input validation.

Subfigure (d) highlights validation errors when required fields like "Site Name," "Description," "Latitude," and "Longitude" are left empty, prompting the user to provide the necessary information. Subfigure (e) further illustrates error handling for the image upload, indicating that a valid image must be provided before the site can be saved. These subfigures collectively demonstrate a robust and user-friendly mechanism for data entry, ensuring data integrity through clear validation messages, which aligns with defensive coding principles.

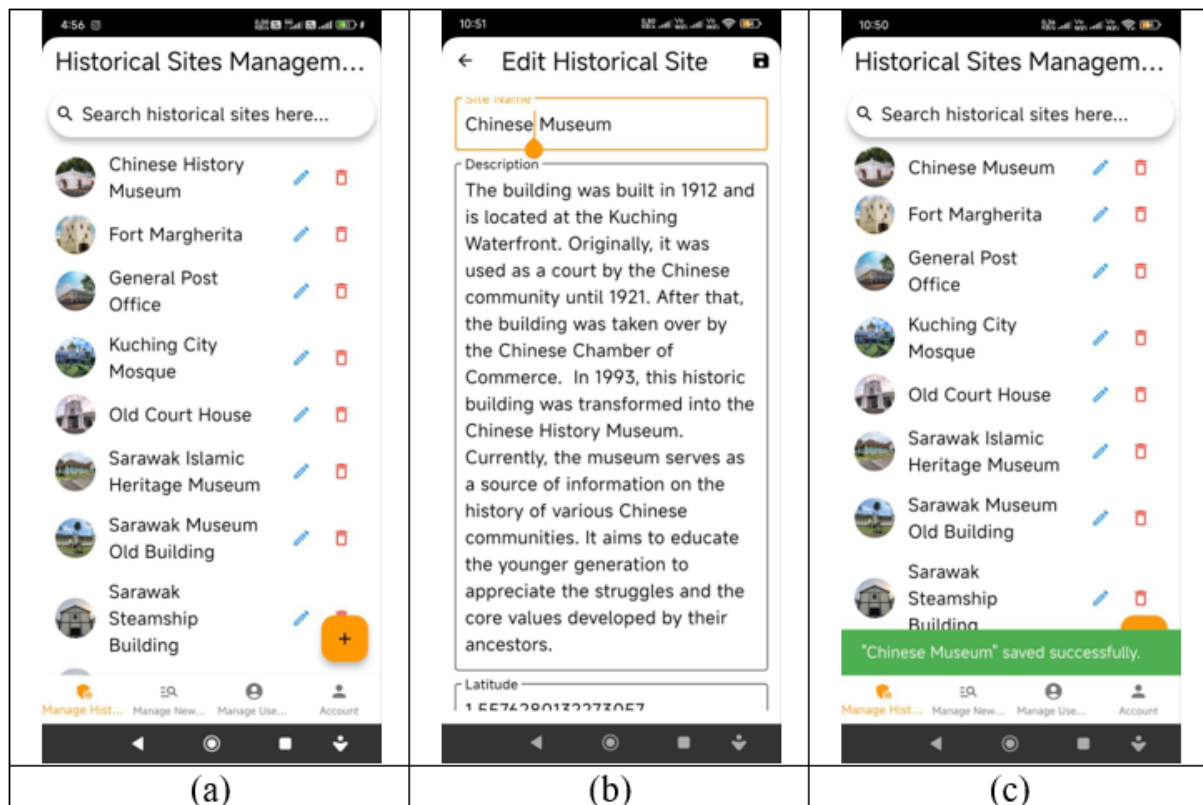


Figure 78. Sequence illustrating admin editing an existing historical site within the application.

Figure 78 illustrates the process of editing an existing historical site within the application, demonstrating the steps from selecting a site to successfully saving the changes. Subfigure (a) displays the "Historical Sites Management" page, which presents a list of historical sites. Each listed site includes an icon to view or edit details (represented by a pencil) and another to delete the entry (represented by a trash can). To initiate the editing process, the user taps the pencil icon associated with a specific site, in this case, "Chinese Museum." Subfigure (b) shows the "Edit Historical Site" form, populated with the current details of the "Chinese Museum." The user can modify any of the existing fields, such as the site name, description, latitude, and longitude. Once the necessary changes are made, the user saves them. Subfigure (c) confirms the successful update of the historical site, indicated by a green banner displaying "Chinese Museum saved successfully." This sequence demonstrates the system's ability to allow the administrator to efficiently update site information, maintaining data accuracy and ensuring a consistent user experience.

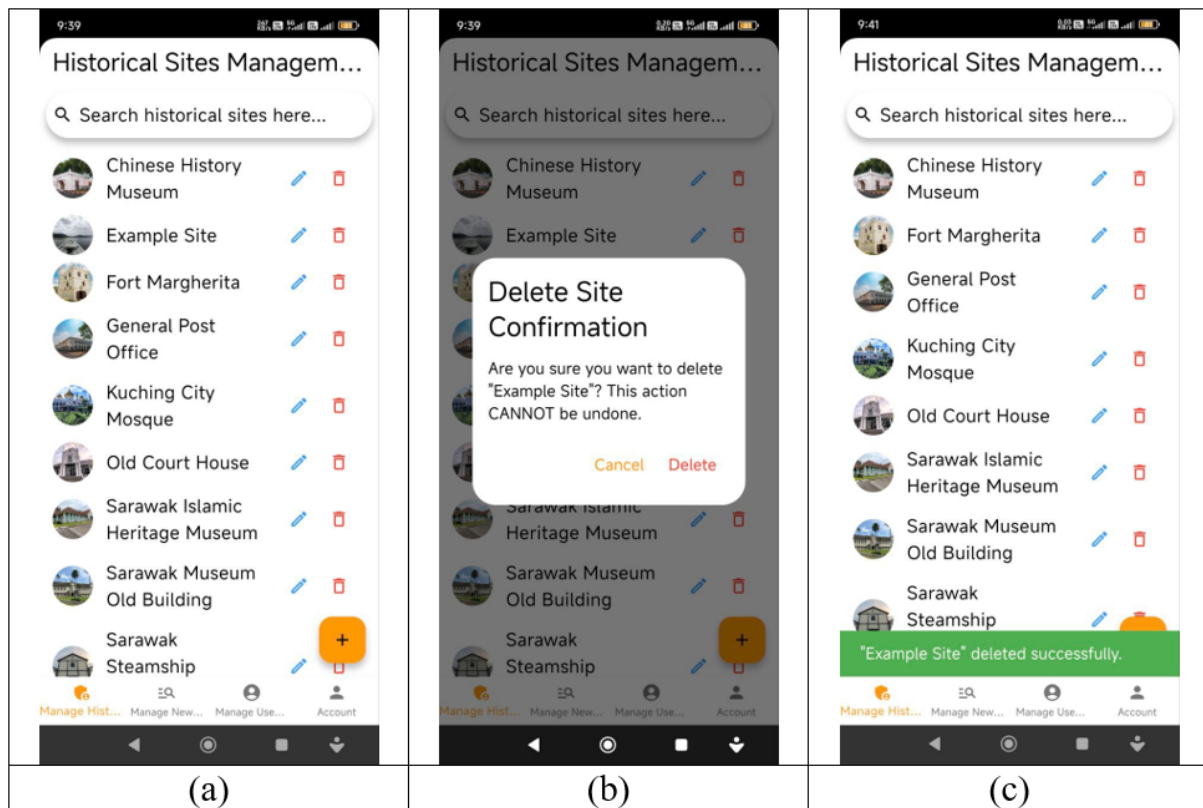


Figure 79. Workflow for admin deleting a historical site from the database

Figure 79 illustrates the process of deleting a historical site within the application, emphasising the confirmation and outcome of this irreversible action. Subfigure (a) displays the "Historical Sites Management" page, presenting a list of existing historical sites. Each listed site includes a delete icon (represented by a trash can) to initiate the removal process. To delete a site, the user taps the trash can icon next to the desired entry, for example, "Example Site." Subfigure (b) shows a critical verification measure which is a "Delete Site Confirmation" dialog box. This pop-up explicitly asks the user, "Are you sure you want to delete 'Example Site'? This action CANNOT be undone," providing a clear warning about the finality of the action and offering "Cancel" or "Delete" options. This confirmation step is crucial for preventing accidental data loss. Subfigure (c) confirms the successful deletion of the "Example Site" with a green banner message stating, "Example Site deleted successfully." The updated list no longer displays the deleted site, reinforcing that the action has been completed. This sequence demonstrates the

system's careful handling of data deletion, prioritizing user confirmation to prevent unintended data removal.

4.4.15 Manage News and Events (Admin)

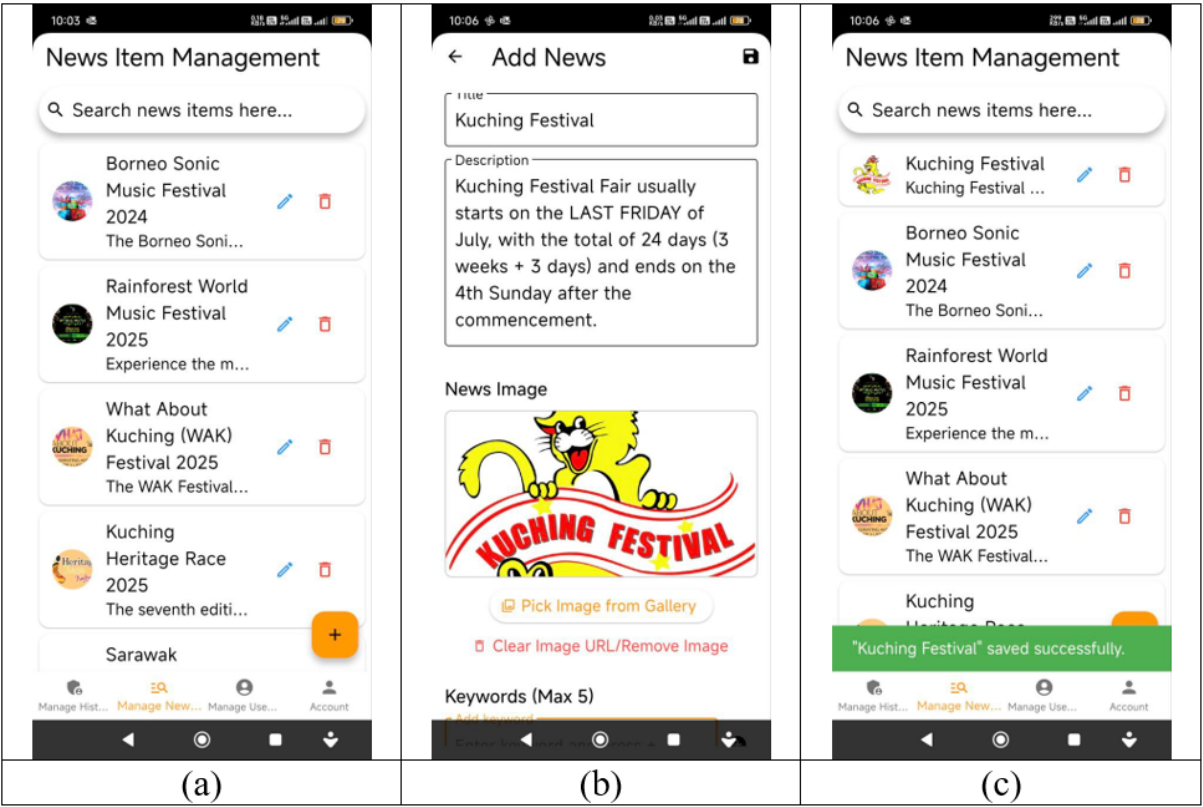


Figure 80. Workflow for admin adding a new news and events item within the application

Figure 80 illustrates the process of adding a new news item to the application's news management system. Subfigure (a) displays the "News Item Management" page, which lists existing news items and provides a search bar for filtering. This serves as the initial view from which a user can choose to add a new entry by tapping the '+' button. Subfigure (b) presents the "Add News" form, which is shown after the user initiates the addition process. This form requires the user to input the news item's title and description. It also provides options to upload a relevant image by selecting one from the gallery and to add up to five keywords for better categorization and searchability. Subfigure (c) shows the successful addition of the new news item, "Kuching Festival," confirmed by a green banner displaying the message "'Kuching Festival" saved successfully."

Festival' saved successfully." The updated "News Item Management" list now includes the newly added news item, demonstrating the seamless integration of new content into the system.

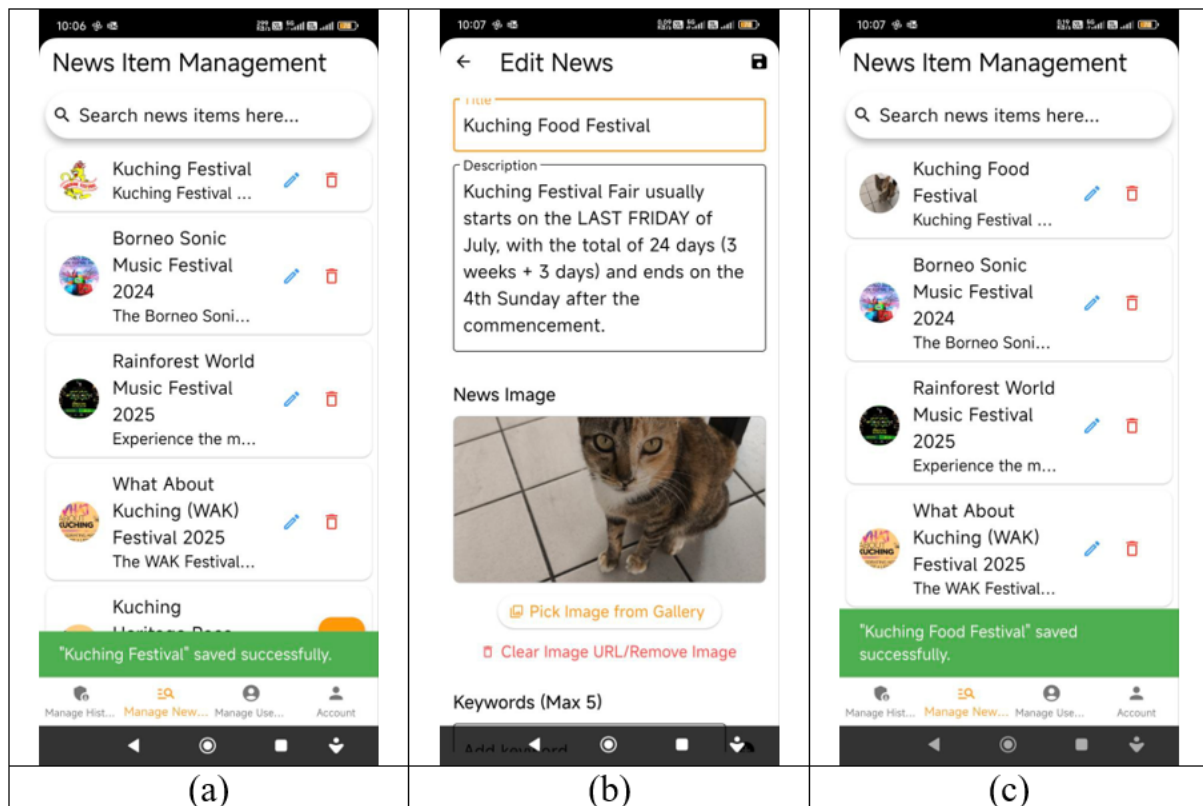


Figure 81. Illustration of the news and events item editing and update workflow by admin

Figure 81 outlines the process by which an administrator can modify an existing news and events item within the application's management interface. Subfigure (a) presents the "News Item Management" screen, displaying an existing list of news entries. Each entry is accompanied by an edit icon (depicted as a pencil) and a delete icon (a trash can). To initiate an edit, the admin can tap on the corresponding entry row or select the pencil icon corresponding to the desired news item, in this instance, "Kuching Food Festival." Subfigure (b) transitions to the "Edit News" form, pre-populated with the current details of the "Kuching Food Festival" entry. Here, the user has the flexibility to revise various fields, including the title, description, and news image. The interface also allows for updating associated keywords, ensuring the news item remains relevant and discoverable. Upon completion of the modifications, the user saves the changes. Subfigure (c) confirms the successful update with a

prominent green banner displaying the message "'Kuching Food Festival' saved successfully."

The updated list of news items now reflects the modifications, seamlessly integrating the revised content. This sequence demonstrates the system's robust capability for content maintenance, enabling administrators to keep information current and accurate through an intuitive editing mechanism.

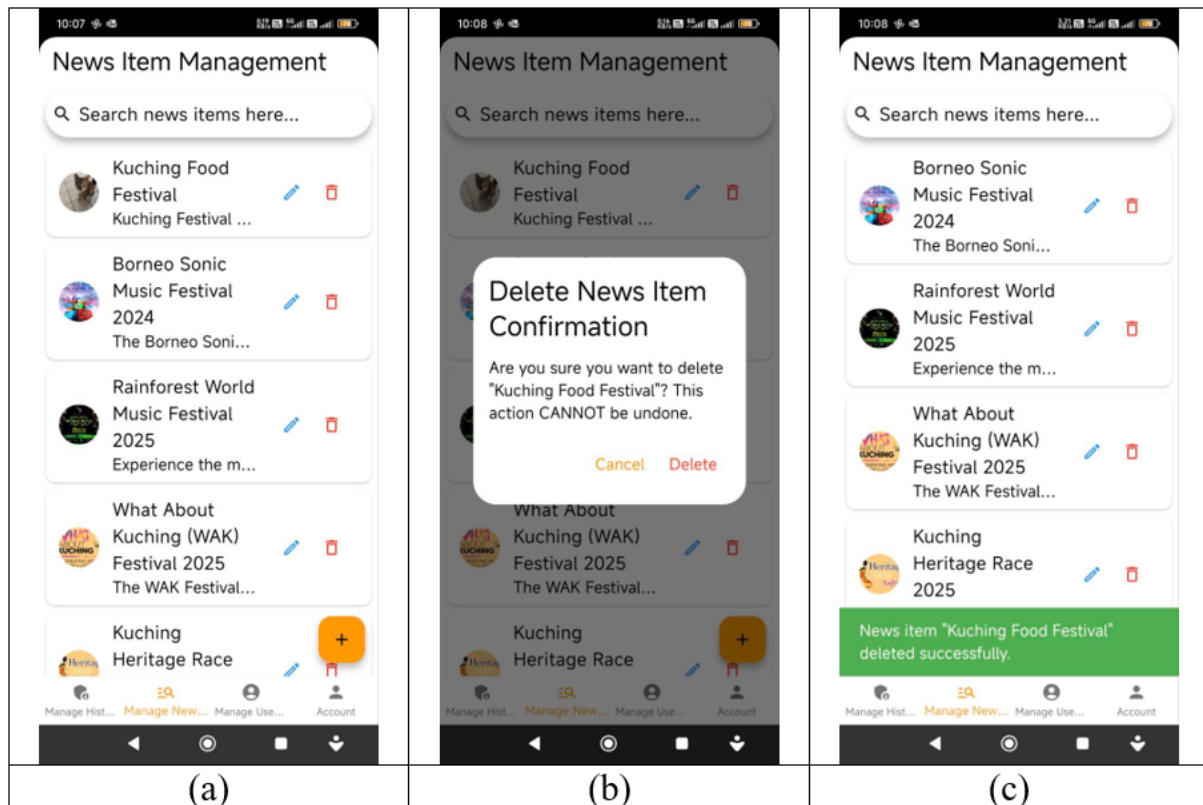


Figure 82. Depiction of the news and events item deletion process, including confirmation

Figure 82 illustrates the systematic procedure for deleting a news and events item deletion process, including confirmation item from the application's management interface, highlighting the crucial confirmation step to prevent unintended data loss. Subfigure (a) displays the "News Item Management" screen, which presents the list of all published news articles. Each entry is equipped with a trash can icon, signaling the option to remove the corresponding item. To initiate the deletion of a specific news item, such as "Kuching Food Festival," the user taps this icon. Subfigure (b) reveals a "Delete News Item Confirmation" dialog box. This essential mechanism prompts the user with the question: "Are you sure you want to delete 'Kuching Food Festival'? This action CANNOT be undone." This explicit warning serves to confirm the

user's intent and provides an opportunity to cancel the operation, thereby mitigating the risk of accidental permanent deletion. Subfigure (c) confirms the successful removal of the news item with a green banner message stating, "News item 'Kuching Food Festival' deleted successfully." The updated list no longer displays the deleted article, signifying the completion of the action. This sequence underscores the system's commitment to data integrity and user control, employing a clear confirmation step for irreversible actions.

4.4.16 View Profile (Admin)

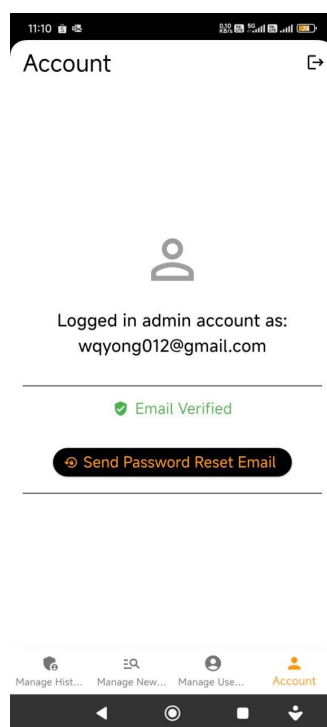


Figure 83. Admin account interface showcasing verified login credentials and password recovery utilities

Figure 83 provides a focused view of the Account section for a logged-in admin user, emphasising credential verification and account recovery features. The displayed email address confirms the user's authenticated session, while the adjacent green shield icon and “Email Verified” status reinforce the account's security integrity. Positioned beneath these identifiers is a functional control labeled “Send Password Reset Email,” enabling authorized password recovery directly through the application interface. This self-service mechanism improves admin autonomy and supports secure credential management.

4.4.17 User Account Management (Admin)

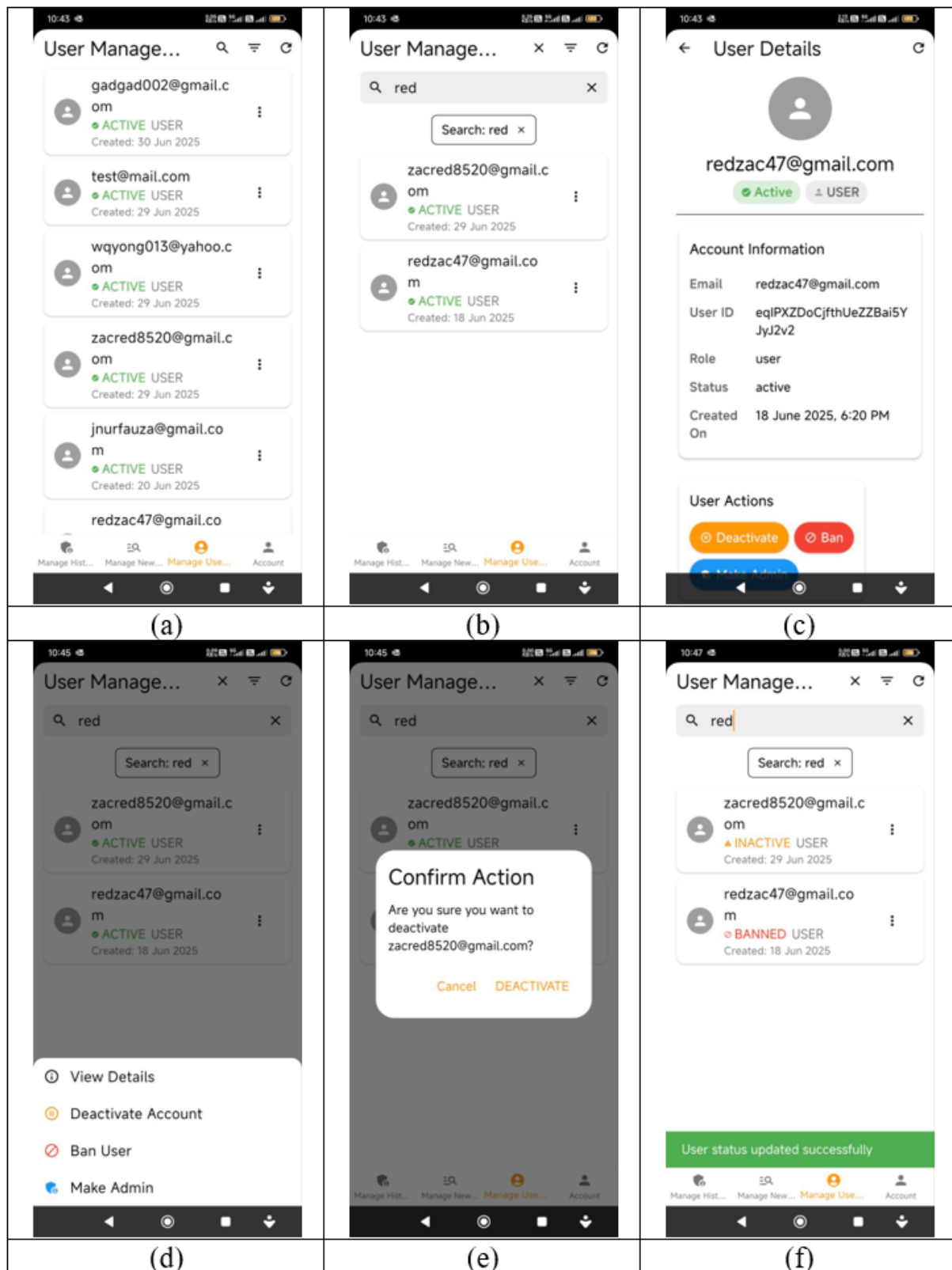


Figure 84. Administrative user management workflow, demonstrating the processes of user account search, filtering by account status, status modification, and confirmation within the application

Figure 84 meticulously illustrates the user management capabilities of the application, encompassing user searching, status modification (deactivation and banning), and the necessary confirmation steps. Subfigure (a) presents the "User Management" screen, displaying a comprehensive list of registered users, each with their email, status (e.g., "Active User"), and creation date. This serves as the primary interface for administrative oversight. Subfigure (b) demonstrates the search functionality, where an administrator can input keywords, such as "red," to filter the user list and quickly locate specific accounts. This enhances efficiency in managing a potentially large user base. Subfigure (c) delves into the "User Details" view, accessed by selecting a user from the list. This screen provides detailed account information, including email, unique user ID, role, status, and creation timestamp. Crucially, it also presents "User Actions" such as "Deactivate" and "Ban," allowing administrators to control user access. Subfigure (d) shows the action menu that appears when the ellipsis icon next to a user is tapped, offering options like "View Details," "Deactivate Account," "Ban User," and "Make Admin." This provides a quick way to perform common management tasks. Subfigure (e) highlights a critical validation practice, the "Confirm Action" dialog. When an administrator attempts to deactivate an account, this pop-up requests confirmation, stating, "Are you sure you want to deactivate zacred8520@gmail.com?" with "Cancel" and "DEACTIVATE" options. This prevents accidental changes to user status, aligning with best practices for data integrity. Finally, subfigure (f) shows the updated user list after an action has been successfully performed, along with a "User status updated successfully" message. The "zacred8520@gmail.com" entry now displays "INACTIVE USER," and "redzac47@gmail.com" is shown as "BANNED USER," reflecting the changes made. This sequence demonstrates a robust, secure, and user-friendly system for managing user accounts, emphasizing clarity and confirmation for critical administrative actions.

4.4.18 Summary of Features

This section presents a consolidated overview of the core features available in the LelengauKch application, organised by functional modules and user roles. Tables 23 and 24 describe the functionalities available to general and administrative users respectively, and highlight the scope of interaction within each module. The summary clarifies system capabilities, ensures feature completeness, and enables further evaluation of usability and coverage for all user types.

Table 23. List of features under respective modules for users for LelengauKch

Module	User Features
Authentication Module	• User registration • Login with email and password • View account email • View email verification status • Email verification • Password reset • Access “About Us” page • Log out of account • Manage subscription to news and event emails
Historical Sites Explorer Module	• View list of historical sites • View detailed information for each site • View zoomable images of the site • Open site location in external maps app (Google Maps)
Map Module	• View an interactive map with markers for historical sites • Search for a specific historical site • Filter historical sites by category using quick-action filters • Tap on a marker to view the site name and access a prompt to open in external maps app • Get directions to a site via an external maps application (Google Maps) • Switch between different map view types (default, satellite)
Chat Assistant Module	• Interact with the AI chat assistant for information about Kuching’s historical sites, culture, news, and heritage souvenirs • View related historical sites or news and events mentioned by the chatbot • Initiate a chat with the AI chat assistant using the default suggestion prompts • Clear chat history

News & Events Module	• View a list of news and events • View news and events information • Track read and unread status of news and events items • Search for particular news and events • Share news and event details (title and description) in plain text format to external platforms (e.g. WhatsApp)
Quiz Module	• Take quizzes about Kuching historical sites • Select quiz difficulty • View current quiz score and quiz answers • Track personal quiz history

Table 24. List of features under respective modules for admin for LelengauKch

Module	Admin Features
Authentication Module	• Login with email and password • Email verification • Password reset • View email • View account status Access to admin dashboard
Historical Sites Explorer Module	• Add new historical sites • Manage existing sites
News & Events Module	• Add new news items • Manage existing news content
User Management Module	• View a list of all registered users • Search and filter users by status (active, inactive, banned) and role (user, admin) • View detailed information for each user • Manage existing users • View own admin account information

4.5 Summary

The public and administrative user interface is developed in Flutter using Dart and follows a tab-driven navigation model that neatly separates the four main domains. For public users, five main domains are Historical Sites, Maps, Chat Assistant, News and Events and Account while for administrative side are Historical Sites, News, User Management, and Account. Both share a global logout feature. Every domain exists as an independent module with its own view layer and logic layer, thereby enforcing the separation of concerns and allowing for maintainability. Implementation began with environment setup which includes setting up Flutter SDK, Dart analysis rules for defensive programming, and version control via GitHub. State management is achieved through the Business Logic Component (BLoC) pattern to realize unidirectional data flow, testability, and clean error handling. Historical Sites and News modules both feature CRUD workflows. Overall, the implementation demonstrates a sound, modular, and secure Flutter-based interface that follows software engineering best practices.

CHAPTER 5: TESTING

5.1 Introduction

Software testing is an essential step that identifies defects or software anomalies while validating and verifying that the system functions as intended in accordance with the specifications. To ensure comprehensive software evaluation, testing efforts are broadly categorized into functional and non-functional approaches. Functional and non-functional testing are two distinct but equally important methods for ensuring software quality that are covered in this chapter. Verifying the software's functionality and making sure each feature operates as intended are the main goals of functional testing. Non-functional testing, on the other hand, assesses the software's performance by evaluating attributes such as usability, security, and performance to ensure a reliable system and an optimal user experience.

5.2 Functional Testing

Functional testing constitutes a critical aspect of software quality assurance, focusing on the meticulous verification of the software's operational capabilities against specified requirements. This category of testing endeavors to ascertain that each feature and function of the system performs as designed, ensuring adherence to intended behaviors and outcomes. By employing various techniques to validate input-output relationships and system interactions, functional testing plays a pivotal role in confirming the software's utility and correctness from an end-user perspective. The subsequent sections of this report will delve into the diverse methodologies and significance of different types of functional testing.

5.2.1 Test Cases

Test cases serve as a fundamental component of functional testing, comprising predefined procedures and input variables utilized to evaluate the application's behavior. These test cases provide a systematic framework that testers follow to verify whether the application operates in accordance with the specified requirements. By executing these procedures, testers can

assess the extent to which the application's implementation aligns with its intended functionality. The specific functions examined in this testing phase are presented in the following Table 25 to Table 31.

Table 25. User - Account Registration Test Cases

Module		Authentication - Account Registration (User)				
Test objective		Verify that new users can successfully register with valid credentials.				
Test ID	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Status
U-REG-1	New user registration with valid data	1. Launch app 2. Navigate to "Sign Up" 3. Enter valid email and password 4. Submit form by clicking "Sign Up"	Email: redzac47@gmail.com Password: Test210	- Account is created - User is logged in and redirected to home screen	- Account is created - User is logged in and redirected to home screen	Pass
U-REG-2	Registration with invalid email format	1. Launch app 2. Navigate to "Sign Up" 3. Enter invalid email format 4. Enter valid password 5. Click "Sign Up" button	Email: redzac47gmail.com Password: Test210	- Error message: "Please enter a valid email address" - User remains on sign up screen	- Error message: "Please enter a valid email address" - User remains on sign up screen	Pass
U-REG-3	Registration with password mismatch	1. Launch app 2. Navigate to "Sign Up" 3. Enter valid email 4. Enter password 5. Enter different confirm password	Email: redzac47@gmail.com Password: - Test210 - Test123	- Error message: "Passwords do not match" - User remains on sign up screen	- Error message: "Passwords do not match" - User remains on sign up screen	Pass

		6. Click "Sign Up" button				
U-REG-4	Registration with existing email	1. Launch app 2. Navigate to "Sign Up" 3. Enter existing email 4. Submit form by clicking "Sign Up"	Email: redzac47@gmail.com Password: Test210	• Error message: "The email address is already in use by another account" • User remains in sign up screen	• Error message: "The email address is already in use by another account" • User remains in sign up screen	Pass
U-REG-5	Registration with insufficient password length	1. Launch app 2. Navigate to "Sign Up" 3. Enter valid email and password length of less than 6 characters 4. Submit form by clicking "Sign Up"	Email: redzac47@gmail.com Password: Test	• Error message: "Password must be at least 6 characters long" • Sign Up process halted	• Error message: "Password must be at least 6 characters long" • Sign Up process halted	Pass

Table 26. User and Admin - Account Login Test Cases

Module		Authentication - Account Login (User and Admin)				
Test objective		Verify that existing users can successfully login into the app with valid credentials.				
Test ID	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Status
UA-LOG-1	Login with valid credentials	1. Launch app 2. Navigate to "Login" 3. Enter valid email and password 4. Click "Login" button	Email: redzac47@gmail.com Password: Test210	• User successfully logged in • Directed to home screen	• User successfully logged in • Directed to home screen	Pass
UA-LOG-2	Login with invalid email	1. Launch app 2. Navigate to "Login" 3. Enter invalid email and valid password 4. Click "Login" button	Email: redzac47@gmail.com Password: Test210	• Error message: "Please enter a valid email address" • User remains on login screen	• Error message: "Please enter a valid email address" • User remains on login screen	Pass
UA-LOG-3	Login with incorrect password	1. Launch app 2. Navigate to "Login" 3. Enter valid email and incorrect password 4. Click "Login" button	Email: redzac47@gmail.com Password: Test123	• Error message: "The supplied auth credential is incorrect, malformed or has expired" • User remains on login screen	• Error message: "The supplied auth credential is incorrect, malformed or has expired" • User remains on login screen	Pass
UA-LOG-4	Login with	1. Launch app	-	• Error message: "Please fill	• Error message: "Please fill	Pass

	empty fields	2. Navigate to "Login" 3. Leave both email and password fields empty 4. Click "Login" button		in all fields." • User remains on login screen	in all fields." • User remains on login screen	
UA-LOG-5	Reset account password via email	1. Launch app 2. Navigate to "Login" 3. Click "Forgot Password?" 4. Enter email address to receive password reset link. 5. Click "Send Link" to complete request	Email: redzac47@gmail.com	• Feedback message: "Password reset email sent to your email. Please check your inbox and spam folder." • Password reset window closes • User remains on login screen	• Feedback message: "Password reset email sent to your email. Please check your inbox and spam folder." • Password reset window closes • User remains on login screen	Pass

Table 27. User - View Historical Sites Test Cases

Module		Historical Sites Explorer - View List of Historical Sites (User)				
Test objective		Verify that users can successfully view a list of historical sites through the Historical Sites Explorer interface.				
Test ID	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Status
U-HS-VHS-1	User views list of historical sites in home page	1. Launch app 2. Navigate to “Explore” 3. Wait for the list of historical sites to load 4. Scroll through the list of total of 10 sites loaded with image displayed in grid format	-	• The home page successfully displays the list of historical sites under the “Explore” section. • A total of 10 historical sites are loaded dynamically from the database. • Each historical site appears in a grid layout with its image and name. • All UI elements (images, text) are properly aligned and load without error.	• The list of historical sites is displayed correctly on the home page. • 10 historical sites are successfully retrieved and shown. • Each site is rendered in a grid format with its image and name. • No loading issues or UI misalignments were observed during testing.	Pass
U-HS-VHS-2	Database returns empty list	1. Clear the historical site entries in the database 2. Launch the app	-	• App displays an empty state message such as “No historical sites available at	• App shows a proper empty state message with no issue.	Pass

		3. Observe the "Explore" page		the moment."		
U-HS-VHS-3	User taps on a historical site card	1. Navigate to "Explore" page 2. Tap "Fort Margherita" 3. View site detail page	- Site: Fort Margherita	• Site detail page loads • Site name, description, history displayed • Image displayed correctly	• Site detail page loads • Site name, description, history displayed • Image displayed correctly	Pass
U-HS-VHS-4	Users taps on image in site detail page	1. Navigate to "Explore" page 2. Tap "Fort Margherita" 3. Tap on image thumbnail 4. Perform gestures to zoom	- Site: Fort Margherita	• Site detail page loads • Image displayed correctly • Image expands to full screen and supports zoom functionality	• Site detail page loads • Image displayed correctly • Image expands to full screen and supports zoom functionality	Pass
U-HS-VHS-5	Users taps on "Navigate" button in site detail page	1. Navigate to "Explore" page 2. Tap "Fort Margherita" 3. Tap on floating button "Navigate"	- Site: Fort Margherita	• Site detail page loads • Floating "Navigate" button correctly positioned • A prompt is displayed requesting user confirmation before redirecting to Google Maps	• Site detail page loads • Floating "Navigate" button correctly positioned • A prompt is displayed requesting user confirmation before redirecting to Google Maps	Pass
U-HS-VHS-6	Image fails to load for a site	1. Navigate to "Explore" page 2. Tap "Test case" 3. Observe image thumbnail	- Site: Test case	• Site detail page loads • A placeholder icon is shown instead of site image	• Site detail page loads • A placeholder icon is shown instead of site image	Pass

Table 28. User – Access Map Navigation Test Cases

Module		Historical Sites Explorer – Access Map For Navigation (User)				
Test objective		Ensure that users can effectively interact with the map interface to discover, filter, and navigate to historical sites.				
Test ID	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Status
U-MAP-UM-1	User views interactive map with site markers	1. Launch app 2. Navigate to “Map” 3. Wait for map to load 4. Check for markers indicated on the interactive map	-	• Map loads correctly • All site markers displayed • Map controls function properly	• Map loads correctly • All site markers displayed • Map controls function properly	Pass
U-MAP-UM-2	User searches for specific historical site	1. Navigate to "Map" section 2. Tap search bar 3.Type "history" 4. Submit search	- Search: history	• Search results displayed • Map focused on searched site after tapping on search result • Info panel for “Chinese History Museum” shown	• Search results displayed • Map focused on searched site after tapping on search result • Info panel for “Chinese History Museum” shown	Pass
U-MAP-UM-3	User filters sites by category	1. Navigate to "Map" section 2. Tap filter button “Historical Landmarks”	- Filter: Historical Landmarks	• Only historical landmarks category markers displayed • Other sites hidden • Relevant search results shown • Filter indicator shown	• Only historical landmarks category markers displayed • Other sites hidden • Relevant search results shown • Filter indicator shown	Pass
U-MAP-UM-4	User switches map view types	1. Navigate to "Map" section 2. Locate the floating view type switch	- View type: Satellite	• Map switches to satellite view	• Map switches to satellite view	Pass

		3. Tap to switch to "Satellite" 4. Tap again to switch back to default		• Satellite imagery visible • Markers remain visible • Switching back works correctly	• Satellite imagery visible • Markers remain visible • Switching back works correctly	
U-MAP-UM-5	User taps on a marker to initiate navigation using the external Google Maps app	1. Navigate to "Map" section 2. Locate a site marker 3. Tap to initiate info panel 4. Tap again to prompt launch external Google Maps app for navigation	- Site: Kuching City Mosque	• The info panel for the selected site (Kuching City Mosque) is displayed after the first tap on the marker. • Upon tapping the info panel, the system prompts to open the Google Maps app. • Google Maps launches successfully, showing the correct location of Kuching City Mosque. • Navigation mode in Google Maps is initiated with the selected site set as the destination.	• The info panel for the selected site (Kuching City Mosque) is displayed after the first tap on the marker. • Upon tapping the info panel, the system prompts to open the Google Maps app. • Google Maps launches successfully, showing the correct location of Kuching City Mosque. • Navigation mode in Google Maps is initiated with the selected site set as the destination.	Pass

Table 29. User – Interact with Chat Assistant Test Cases

Module		Chat Assistant – Interact with Chat Assistant (User)				
Test objective		Verify that users can successfully interact with the Chat Assistant, including sending messages, receiving responses, and accessing relevant suggestions provided by the assistant.				
Test ID	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Status
U-CA-IC-1	User sends message to chat assistant	1. Launch app 2. Navigate to “Chat” section 3. Type a query in the text field 4. Send message 5. Wait for response	- Input: How to get to Old Court House from Hilton Hotel	• Message appears in chat • Loading indicator shown • Chatbot responds with relevant information about ways to reach Old Court House from Hilton Hotel	• Message appears in chat • Loading indicator shown • Chatbot responds with relevant information about ways to reach Old Court House from Hilton Hotel	Pass
U-CA-IC-2	User uses suggestion prompt	1. Navigate to "Chat" section 2. View suggestion queries 3. Tap on "Tell me about Fort Margherita" 4. Wait for response	- Input: Tell me about Fort Margherita	• Suggestion sent as message • Loading indicator shown • AI responds with relevant information about Fort Margherita	• Suggestion sent as message • Loading indicator shown • AI responds with relevant information about Fort Margherita	Pass
U-CA-IC-3	User views relevant mentioned historical site or news and events item	1. Navigate to "Chat" section 2. View suggestion queries 3. Tap on "Tell me about Fort Margherita" 4. Wait for response	- Input: Tell me about Fort Margherita	• Message appears in chat • Loading indicator shown • Chatbot responds with relevant information about Fort Margherita • User can tap on the card below the respond to	• Message appears in chat • Loading indicator shown • Chatbot responds with relevant information about Fort Margherita • User can tap on the card below the respond to	Pass

				navigate into site detail page	navigate into site detail page	
U-CA-IC-4	User views relevant news and events item	1. Navigate to "Chat" section 2. View suggestion queries 3. Tap on "Tell me about news" 4. Wait for response	- Input: Tell me about news	• Message appears in chat • Loading indicator shown • Chatbot responds with relevant information about news and events • User can tap on the card below the respond to navigate into news detail page	• Message appears in chat • Loading indicator shown • Chatbot responds with relevant information about news and events • User can tap on the card below the respond to navigate into news detail page	Pass
U-CA-IC-5	User clears chat history	1. Navigate to "Chat" section with existing messages 2. Tap delete icon at top right 3. Select "Clear" to confirm action	- Data: existing chat history	• Confirmation dialog appears • After confirmation, chat history cleared • Welcome message displayed	• Confirmation dialog appears • After confirmation, chat history cleared • Welcome message displayed	Pass

Table 30. User - Access News and Events Test Cases

Module		News and Events – Access News and Events (User)				
Test objective		Verify that users can successfully access, view, and read the latest news articles and upcoming event details through the News and Events module.				
Test ID	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Status
U-NAE-ACS-1	User views list of news and events	1. Navigate to "News & Events" section 2. Check initial news listing 3. Scroll through list	-	- News items displayed in descending chronological order - Thumbnails and titles visible - Read and unread status indicated	- News items displayed in descending chronological order - Thumbnails and titles visible - Read and unread status indicated	Pass
U-NAE-ACS-2	User views news and events item details	1. Navigate to "News & Events" section 2. Select "Kuching Heritage Race 2025" 3. Check detailed view	- item: Kuching Heritage Race 2025	- Full article displayed - Image displayed correctly - Date and source information visible	- Full article displayed - Image displayed correctly - Date and source information visible	Pass
U-NAE-ACS-3	User shares news and events item	1. Navigate to "News & Events" section 2. Select "Rainforest World Music Festival 2025" 3. Tap floating share icon 4. Select WhatsApp platform	- item: Rainforest World Music Festival 2025 - platform: WhatsApp	- Full article displayed - Share dialog opens - WhatsApp launches - Article title and description included in share text	- Full article displayed - Share dialog opens - WhatsApp launches - Article title and description included in share text	Pass
U-NAE-ACS-4	User searches for news and events item	1. Navigate to "News & Events" section 2. Tap search icon 3. Type "festival" 4. Check search results	- input: festival	- Only news containing "festival" displayed - Search term highlighted in results	- Only news containing "festival" displayed - Search term highlighted in results	Pass

U- NAE- ACS-5	User filters out unread news and events item	1. Navigate to "News & Events" section 2. Tap filter icon at top right 3. Check filtered results	- filter: unread	• Only unread news and events item displayed • Screen displays 'No unread news available' when all news and event items have been marked as read.	• Only unread news and events item displayed • Screen displays 'No unread news available' when all news and event items have been marked as read.	Pass
---------------------	---	--	---------------------	--	--	------

Table 31. User - Access Quiz Test Cases

Module		News and Events – Access News and Events (User)				
Test objective		Verify that users can successfully access, view, and read the latest news articles and upcoming event details through the News and Events module.				
Test ID	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Status
U-Q-AQ-1	User participates in easy difficulty quiz	1. Navigate to "Quiz" section 2. Select "Easy" difficulty 3. Start quiz 4. Answer all questions 5. Submit quiz	- Difficulty: Easy - All answers selected	• 5 questions displayed sequentially • Answers recorded correctly • Final score calculated and displayed • Correct answers shown • Celebration confetti shown when total score is at least 80%	• 5 questions displayed sequentially • Answers recorded correctly • Final score calculated and displayed • Correct answers shown • Celebration confetti shown when total score is at least 80%	Pass
U-Q-AQ-2	User utilises quiz navigation	1. Navigate to "Quiz" section 2. Start medium difficulty quiz 3. Answer first question 4. Navigate to next question 5. Navigate back to previous question 6. Change answer 7. Complete quiz	- Difficulty: Medium - Changed answer on question 1	• Navigation between questions works • Changed answer saved • Final score reflects changed answer	• Navigation between questions works • Changed answer saved • Final score reflects changed answer	Pass
U-Q-AQ-3	User views quiz history	1. Complete at least one quiz 2. Navigate to "Previous Results" 3. Check history entries	- Completed quiz data	• Quiz history displayed • Dates and scores visible	• Quiz history displayed • Dates and scores visible	Pass

5.2.2 System Testing

5.2.2.1 Introduction and Objectives

System testing is a critical phase in the software testing lifecycle in which a fully integrated software application is evaluated to ensure that it meets the defined requirements. System testing for the "LelengauKch" application was carried out in a systematic manner to validate the entire system's functionality from start to finish. This phase served as a comprehensive evaluation of whether all individual application components work together to meet the functional specifications established during the initial design and requirement gathering phases.

The testing process focused on ensuring that each functional requirement, spanning all application modules, was correctly implemented. It entailed evaluating the interactions and data flow between the Flutter-developed front-end and Firebase's back-end services, which included authentication, firestore, and storage. Attention was directed to the integration of these components to ensure that user actions on the interface correctly triggered and processed backend operations with no errors or data inconsistencies.

Furthermore, system testing aimed to identify and resolve any defects within the fully integrated application. Emphasis was placed on user workflows, examining how different modules interact under various usage scenarios to expose potential edge cases and unexpected behaviors. This facilitated the detection of system-level issues that might not surface during earlier unit or integration testing phases.

In addition, the application was tested on the targeted Android platform to determine its performance, stability, and reliability from the perspective of the end user. This ensured that the software not only worked properly in a controlled development environment, but also exhibited consistent behaviour and responsiveness on real devices under normal usage conditions. The system testing results provided critical insights and served as the final validation before the application was considered ready for deployment.

5.2.2.2 Testing Approach and Scope

A black-box testing methodology was employed during the system testing phase of the "LelengauKch" application. This approach emphasised validating the application's external behavior and functional correctness without consideration for its internal code structure or implementation details. Test cases were systematically developed based on a comprehensive breakdown of application features across all modules. These cases accounted for a wide range of scenarios, including standard user flows (happy paths), error-handling conditions, and boundary value tests to ensure robustness under edge-case inputs.

The scope of system testing is extensive, encompassing all major modules of the application. For the Authentication Module, testing covered user registration, login functionality, password reset operations, and account management features. Attention was given to role-based access control mechanisms distinguishing standard users from administrators, as well as handling status checks for 'banned' or 'inactive' accounts—logic that is centrally managed in the main application entry point.

The Historical Sites Explorer Module was tested to confirm the correct retrieval and display of historical site listings, support for detailed views including high-resolution image zooming, and integration with third-party map applications for enhanced navigation. These functionalities were validated to ensure accurate data representation and usability from an end-user standpoint.

Testing of the Map Module focused on validating the dynamic behavior of the interactive map interface. This included verifying that site markers were rendered correctly, that search and filtering functions returned appropriate results, and that navigation features—enabled via external redirection using the `'url_launcher'` package—functioned without failure across different device configurations.

In the Chat Assistant Module, the testing process validated user interactions with the integrated AI chatbot. This involved verifying prompt suggestions, assessing the responsiveness of the chat interface, and ensuring that users could clear chat histories without residual data conflicts. These tests were designed to evaluate the consistency and reliability of conversational features under varied user input.

The News and Events Module underwent thorough verification to ensure that news listings, detailed content views, search functionality, and content-sharing capabilities worked as intended. The sharing mechanism, implemented using the `'share_plus'` package, was specifically tested across different platforms to guarantee compatibility and responsiveness.

Finally, the Quiz Module was rigorously tested to validate the end-to-end quiz-taking experience. This included checking the logic for difficulty selection, ensuring that the correct number of questions were retrieved according to the selected difficulty, and verifying the transition between answering questions and viewing results. Functionality was assessed based on the behavior of the `'QuizProvider'` and its ability to manage state and logic throughout the quiz lifecycle.

The adoption of black-box testing across all modules ensured a user-centric evaluation of the application, verifying that the system functioned as specified and delivered a seamless and coherent experience across all features.

5.2.2.3 Test Execution and Results

System testing was performed manually by executing a comprehensive suite of test cases, such as those detailed in the test case tables under section 5.2.1 Test Cases. Each test case included specific steps, test data, and expected results, which were compared against the actual results to determine a Pass or Fail status. The following Table 32 provides a high-level summary of the test execution for a sample of key functionalities.

Table 32. System Testing Test Cases Summary (High-Level)

Test ID	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Status
U-LOG-1	Login with valid credentials	1. Launch app 2. Navigate to "Login" 3. Enter valid email and password 4. Click "Login"	- Email: user@gmail.com - Password: Pass123!	- User successfully logged in - Directed to home screen	- User successfully logged in - Directed to home screen	Pass
 U-REG-3	Registration with password mismatch	1. Launch app 2. Navigate to "Sign Up" 3. Enter valid email 4. Enter mismatched passwords 5. Click "Sign Up"	- Email: test@gmail.com - Password: Pass123! - Confirm: Pass456!	- Error message: "Passwords do not match" - Form not submitted	- Error message: "Passwords do not match" - Form not submitted	Pass
U-HS-VHS-3	View detailed site information	1. Navigate to "Historical Sites" 2. Select "Fort Margherita" 3. View details page	- Site: Fort Margherita	Detailed page loads with correct name, description, and image.	Detailed page loads with correct name, description, and image.	Pass
U-MAP-UM5	Get directions to site	1. Open map screen 2. Tap on a marker 3. Tap info panel 4. Tap "Navigate"	- Site: Kuching City Mosque	External maps application opens with a route to the selected site.	External maps application opens with a route to the selected site.	Pass
U-Q-AQ-1	Take easy difficulty quiz	1. Navigate to "Quiz" 2. Select "Easy" difficulty 3. Answer all questions	- Difficulty: Easy	- Quiz loads with 5 questions. - Final score is calculated and	- Quiz loads with 5 questions. - Final score is calculated and	Pass

		4. Submit quiz		displayed correctly.	displayed correctly.	
U-NAE - ACS -4	Search for specific news	1. Open news section 2. Use search feature 3. Enter search term	- Search: "Festival"	• Only news items containing the search term are displayed.	• Only news items containing the search term are displayed.	Pass

The system testing phase of the "LelengauKch" application was completed successfully. The results confirm that all modules are properly integrated, and the application as a whole meets the requirements. The application is stable and reliable, with a consistent user experience across all of its core features. The successful completion of system testing validates the application's readiness for the final stage of testing.

5.3 Non-Functional Testing

Complementary to the evaluation of software functionality, non-functional testing addresses the critical quality attributes that define the user experience and system efficiency. This domain of testing encompasses the assessment of how well the software performs, examining crucial characteristics such as performance, security, usability, reliability, and maintainability. By quantifying these attributes, non-functional testing ensures the software not only operates correctly but also does so in a manner that is secure, efficient, and user-friendly. The subsequent discussion will elaborate on the type of non-functional testing and the importance in achieving a robust and high-quality software product.

5.3.1. Usability Testing

The LelengauKch mobile application was tested with users to assess its usability, functionality, and overall user experience. Data was gathered using a structured questionnaire distributed virtually through Google Forms. Participants were asked to interact with the application and then fill out the form based on their actual usage, perceptions, and level of satisfaction with the

system's features and functionality. Appendix B contain usability testing questionnaires for reference. The results of respondents' responses are presented using bar and pie charts for efficient analysis.

5.3.1.1 Application Usability Evaluation

Figures 85 to 96 present user responses from a usability survey conducted with 31 participants who tested the LelengauKch mobile application. The evaluation focused on key interaction points and feature sets within the application, rated on a 5-point Likert scale (1 indicates Strongly Disagree, and 5 represents Strongly Agree).

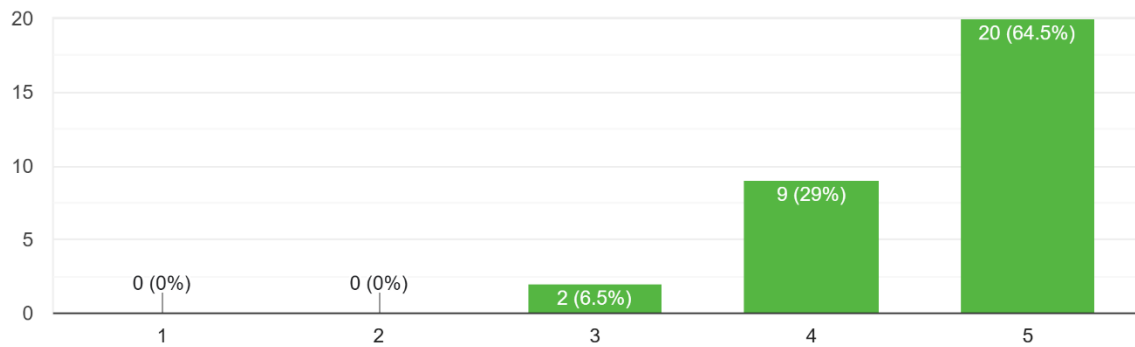
The majority of participants reported a positive experience during installation and launch of the application (Figure 85), with a large portion selecting 4 or 5, indicating minimal issues encountered during initial access. Similar strong agreement was observed in user feedback regarding page navigation (Figure 86), reflecting a well-structured and intuitive layout.

Historical site information was also well received (Figure 87), with users stating that it was clear and useful. Figures 88 to 90 show that most users had little trouble accessing the feature, understood the questions, and were generally pleased with the quiz user experience. These ratings reaffirm the module's educational value and engagement potential.

Navigation via the application and external Google Maps (Figure 91) was deemed efficient and user-friendly, bolstering the app's intuitive design. User feedback for the chatbot assistant (Figure 92) was moderately positive, with some ratings of 3 and 4 indicating that responses could be more responsive or relevant. Figure 93 and 94 also shows overall positive user experience with the “News” section. Importantly, Figure 95 shows that most users did not encounter significant bugs during use, as evidenced by the high number of ratings of 4 and 5. This demonstrates a stable and robust application build. Finally, the general app experience (Figure 96) received high ratings, indicating that the overall user journey was well received.

The app is easy to install and open.

31 responses



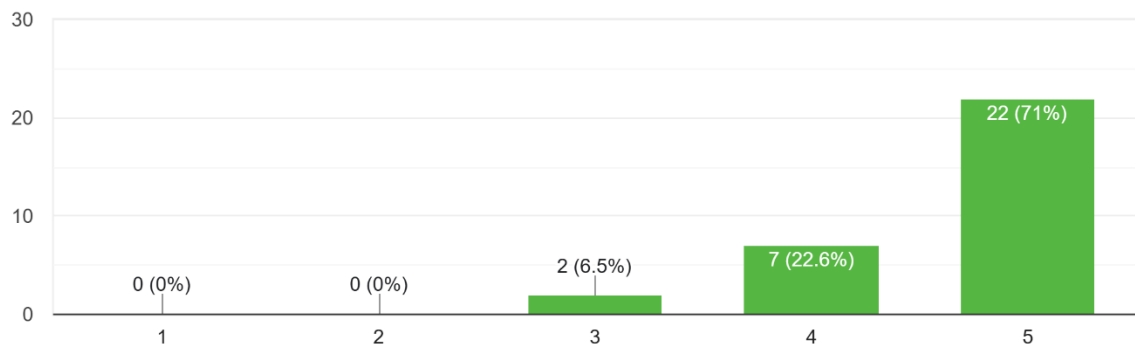
Legend: 1 - Strongly Disagree, 2 – Disagree, 3 – Neutral, 4 – Agree, 5 - Strongly Agree

Figure 85. User feedback on app installation and launch



The navigation between pages is intuitive.

31 responses

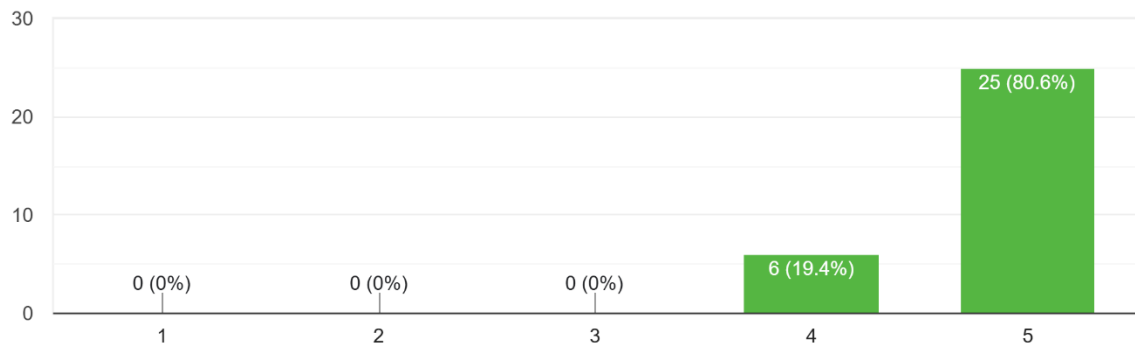


Legend: 1 - Strongly Disagree, 2 – Disagree, 3 – Neutral, 4 – Agree, 5 - Strongly Agree

Figure 86. User feedback on page navigation

The information on historical sites is clear and informative.

31 responses

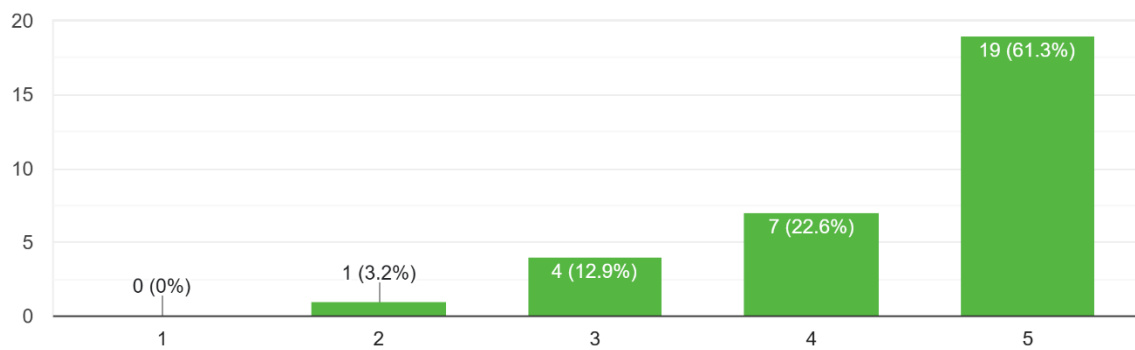


Legend: 1 - Strongly Disagree, 2 – Disagree, 3 – Neutral, 4 – Agree, 5 - Strongly Agree

Figure 87. User feedback on historical sites information.

The quiz module is easy to find and access from the main menu.

31 responses

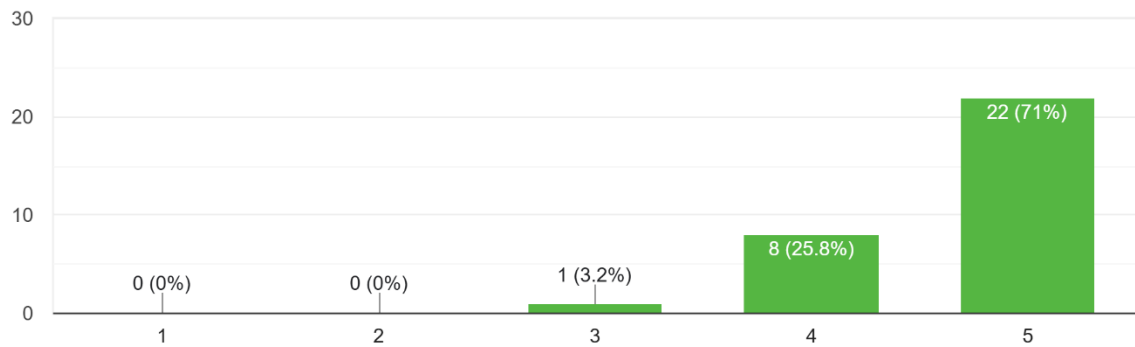


Legend: 1 - Strongly Disagree, 2 – Disagree, 3 – Neutral, 4 – Agree, 5 - Strongly Agree

Figure 88. User feedback on accessing quiz module

The questions provided are relevant to the historical content featured in the app.

31 responses

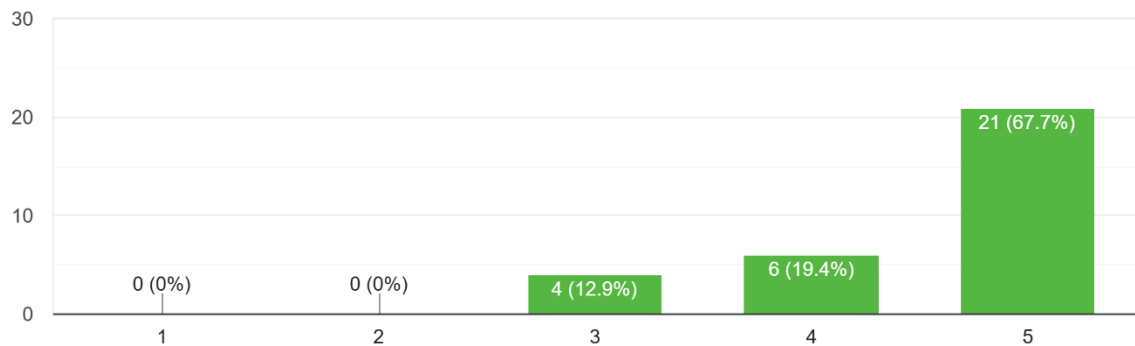


Legend: 1 - Strongly Disagree, 2 – Disagree, 3 – Neutral, 4 – Agree, 5 - Strongly Agree

Figure 89. User feedback on quiz questions

I enjoyed interacting with the quiz and found it engaging.

31 responses

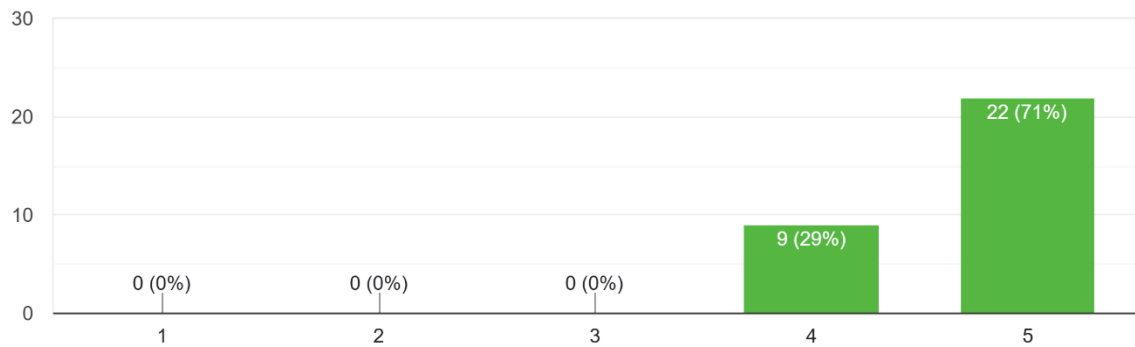


Legend: 1 - Strongly Disagree, 2 – Disagree, 3 – Neutral, 4 – Agree, 5 - Strongly Agree

Figure 90. User feedback on quiz user experience

The Google Maps navigation works correctly and helps in route guidance.

31 responses

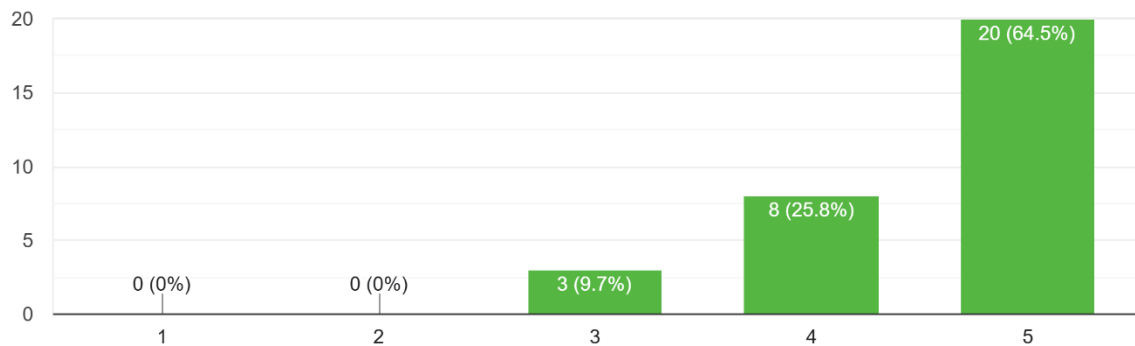


Legend: 1 - Strongly Disagree, 2 – Disagree, 3 – Neutral, 4 – Agree, 5 - Strongly Agree

Figure 91. User feedback on navigation

The chatbot feature provides helpful responses.

31 responses

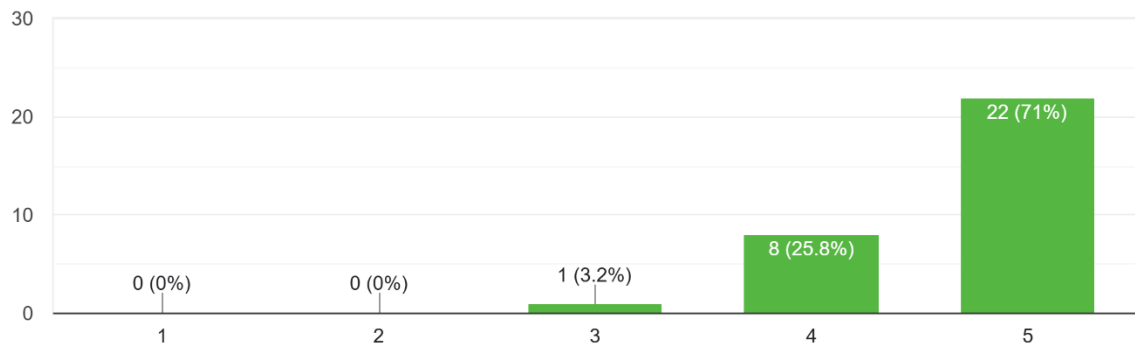


Legend: 1 - Strongly Disagree, 2 – Disagree, 3 – Neutral, 4 – Agree, 5 - Strongly Agree

Figure 92. User feedback on chatbot assistant

The content displayed in the News and Events section is relevant and informative.

31 responses

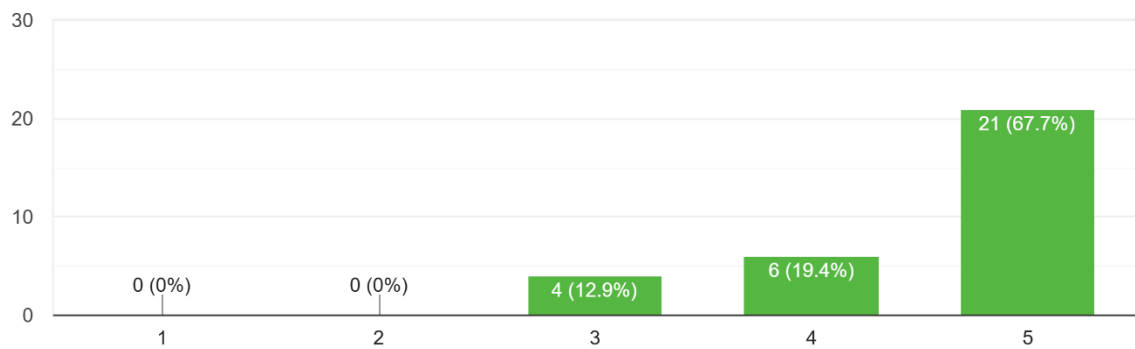


Legend: 1 - Strongly Disagree, 2 – Disagree, 3 – Neutral, 4 – Agree, 5 - Strongly Agree

Figure 93. User feedback on content of news and events section

I feel more engaged with the app because of the potential regular updated content in News and Events section.

31 responses

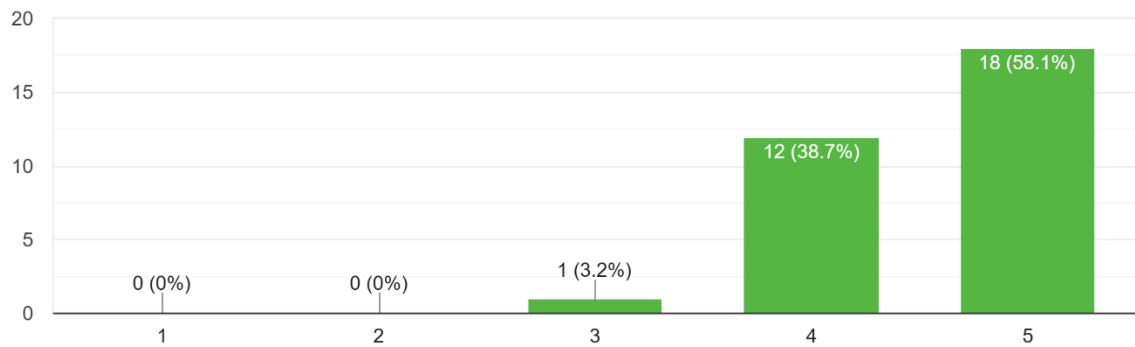


Legend: 1 - Strongly Disagree, 2 – Disagree, 3 – Neutral, 4 – Agree, 5 - Strongly Agree

Figure 94. User feedback on usage of News and Events section

I did not encounter technical errors or bugs during usage.

31 responses

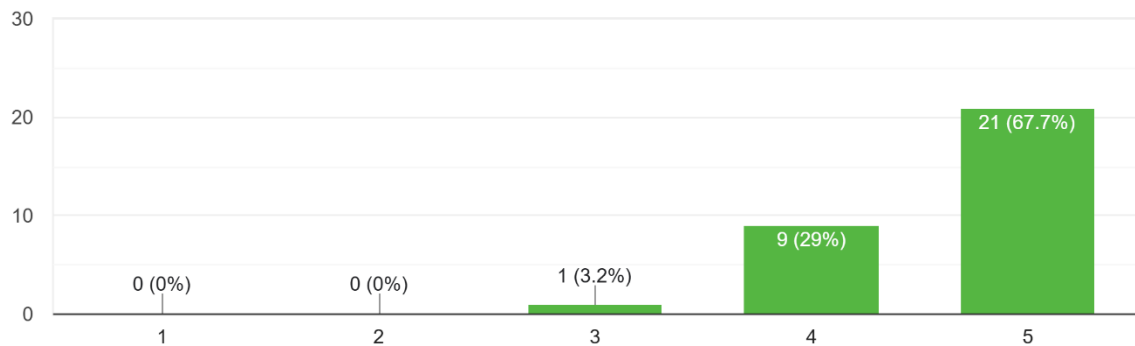


Legend: 1 - Strongly Disagree, 2 – Disagree, 3 – Neutral, 4 – Agree, 5 - Strongly Agree

Figure 95. User feedback on bug experience during usage

The app performs smoothly without lag.

31 responses



Legend: 1 - Strongly Disagree, 2 – Disagree, 3 – Neutral, 4 – Agree, 5 - Strongly Agree

Figure 96. User feedback on app experience

5.3.1.2 Feature-Specific Feedback

Feature usage and perceived usefulness were further explored in Figures 97 and 98. Figure 97 demonstrates broad engagement across key features, with especially high usage of navigation, historical site pages, and the quiz module. Figure 98 specifically evaluates the chatbot's effectiveness, where the majority of users found it helpful, though some participants offered neutral feedback. This suggests the chatbot is functional but could be enhanced for improved clarity, speed, and context handling.

Which features did you use or experience?

31 responses

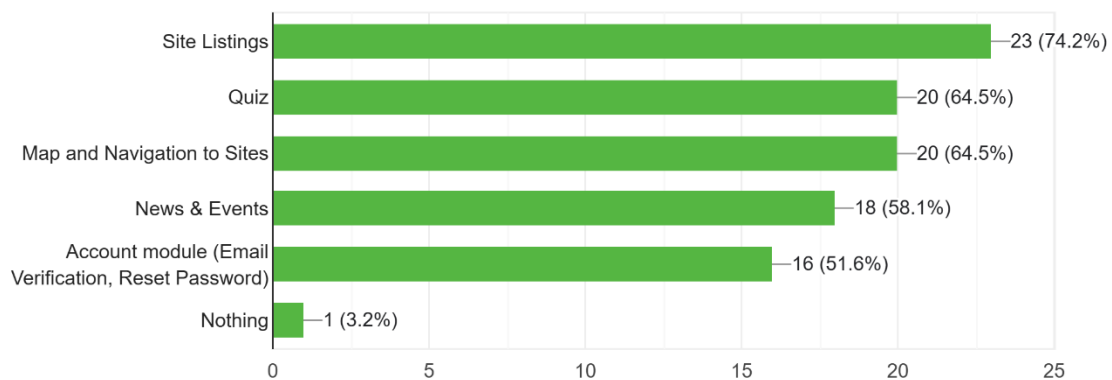


Figure 97. User feedback for features used or experienced

How useful did you find the chatbot in answering your queries?

31 responses

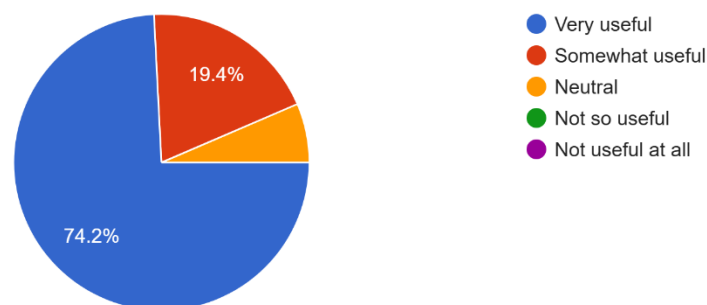


Figure 98. User feedback on usefulness of chatbot assistant in responding to queries

5.3.1.3 Overall Satisfaction Feedback

Rate your overall satisfaction with the application.

Linear scale (1–5)

(1 – Very Dissatisfied, 5 – Very Satisfied)

31 responses

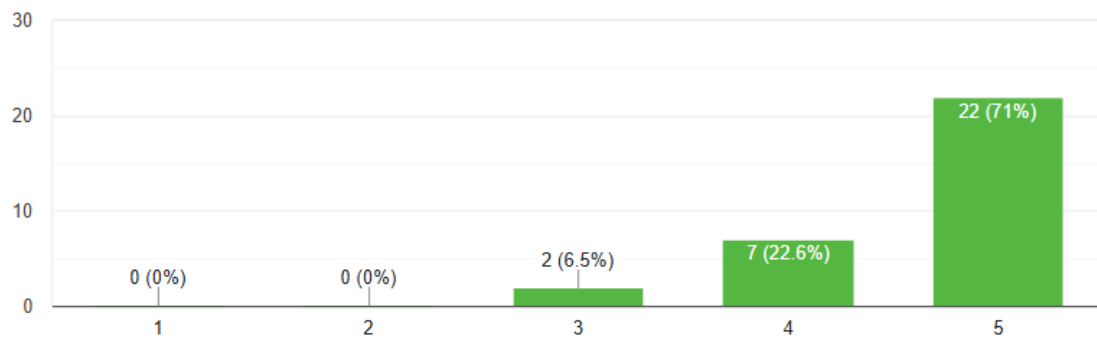


Figure 99. . User feedback on overall satisfaction with the app

The survey depicted in Figure 99 on overall application satisfaction, based on 31 responses, reveals predominantly positive user feedback. A combined total of 93.6% of users reported high satisfaction, with 22.6% rating their experience as "Satisfied" (4) and 71% as "Very Satisfied" (5). The remaining 6.5% of respondents indicated moderate satisfaction (3), with no users reporting dissatisfaction (1 or 2). These findings suggest a strong overall positive reception of the application among its users.

Would you recommend this application to others?

31 responses

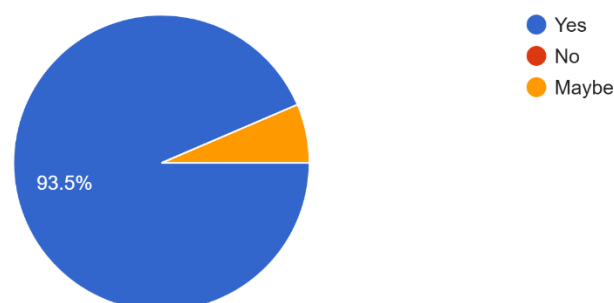


Figure 100. User feedback on recommending the application to other users

The pie chart as observed in Figure 100 illustrates user willingness to recommend the application to others, based on 31 responses. A substantial majority, 93.5% of respondents, indicated they "Yes" would recommend the application. The remaining 6.5% selected "Maybe," suggesting potential but not definitive recommendation. Notably, no respondents selected "No," indicating a complete absence of users unwilling to recommend the application. These results strongly suggest a high level of user satisfaction translating into a strong likelihood of positive word-of-mouth promotion for the application.

Lastly, user feedback regarding potential improvements for the application highlighted several areas for development. Recurring themes included enhancing the quiz feature through increased question variety, the addition of a leaderboard, and better integration or highlighting within the application. Users also suggested improvements to the chatbot, specifically concerning its speed. Furthermore, requests were made for accessibility enhancements, such as larger font sizes and better contrast for elderly users, and the inclusion of multilingual support. Other suggestions encompassed technical improvements like the development of an iOS version, and functional additions such as audio narration for sites, tour package suggestions, a "favourites" or bookmark feature, and broader inclusion of cultural events. A notable portion of respondents indicated no specific suggestions for improvement. However, there is potential for improvement in the quiz module content, such that instead of portraying the quiz in a classic exam style, it could incorporate visual-based questions such as displaying an image and prompting the user to identify the historical site implicitly depicted thereby enhancing engagement and promoting visual learning.

5.4 Summary

This chapter has presented the testing process carried out for the LelengauKch mobile application, covering both functional and non-functional aspects. Functional testing was conducted using detailed test cases to validate that each implemented feature—such as user authentication, historical site browsing, interactive maps, AI chatbot assistance, news and events, and quizzes—meets the specified requirements and behaves as expected. Each test case was designed to verify core functionalities and user interactions to ensure a seamless and reliable user experience.

In addition to functional tests, non-functional testing was performed through user feedback collected via structured questionnaires. These responses provided insight into usability, interface satisfaction, performance, and overall user acceptance of the app. The feedback highlighted that the app meets its intended goals while also identifying potential areas for future enhancement.

In conclusion, the testing results demonstrate that the LelengauKch application performs successfully according to the defined requirements. While the current version is functional and usable, there is still room for further refinement to enhance user experience and expand its capabilities based on the insights gathered during testing.

CHAPTER 6: CONCLUSION AND FUTURE WORKS

6.1 Introduction

This chapter summarizes the key achievements of the project in relation to the initial objectives, highlighting how the goals were met and the overall contribution of the work. It also outlines the limitations and challenges encountered during the development process, providing critical reflection on aspects that could be improved. Finally, it proposes potential directions for future work, suggesting enhancements or alternative approaches that could help address the identified limitations and expand the scope of the project.

6.2 Objective Achievements

The objectives of this project have been defined in Chapter 1 of this report and successfully achieved. Each objective was systematically addressed through the design, development, and testing phases of the mobile application. The implementation of key features demonstrates the practical realisation of the project's goals. The following Table 33 shows the achievement of each objective.

Table 33. Project Objectives and Achievements

Objectives	Achievements
To study and analyse existing mobile tour guide applications for Kuching historical sites.	Conducted a comparative analysis of other 3 tourism apps (including SmartGuide, Sarawak Travel, Travel Malaysia), identifying key features (interactive maps, quizzes, multimedia content, and chat assistance) and produce a feature-set and UX guidelines to inform the app design.
To design and implement a Kuching Historical Tour Guide mobile application.	Produced UI wireframes and developed a functional Flutter-based app, integrating

	Firebase Authentication, Firestore, Cloud Storage, Google Maps SDK, Provider state management, and a Vertex AI-powered chatbot. Delivered fully functional public and admin modules with CRUD capabilities.
To test and evaluate the mobile application's usability and functionality to ensure it enhances Kuching's tourist experience.	Executed functional tests on all modules (site listings, quizzes, maps, news feed, chatbot, account) and carried out a usability study with 31 participants. Collected qualitative feedback demonstrating improved engagement and clarity.

6.3 Project Limitations

Although the project has successfully achieved its stated objectives, several limitations were identified upon its completion. One of the application's main limitations is its reliance on a stable internet connection and external APIs. Users without internet access will be unable to use the app's core features because it retrieves dynamic content from Firebase Firestore, such as historical site listings, news and event items, and quizzes. This reduces the app's usability during periods of poor connectivity. Additionally, the application leverages third-party services including Google Maps and Vertex AI for functionalities such as location mapping and chatbot responses. These services are subject to usage quotas and pricing constraints. In scenarios involving high traffic or excessive API calls, the application may experience poor performance or unexpected operational costs.

Another notable limitation of the application is a lack of multilingual support. The current implementation primarily uses one language, English, which may not adequately cater to the diverse linguistic backgrounds of users, particularly international tourists visiting Kuching.

This restriction may limit the application's accessibility and usability to non-native speakers, reducing its effectiveness as a cultural and historical tour guide. In a multicultural and tourism-oriented environment, the lack of language localisation may impede user engagement and inclusion.

In addition, another limitation of the application is its platform exclusivity, as it is developed solely for the Android operating system. This restricts accessibility for users of other platforms, particularly iOS, which comprises a sizable portion of the mobile device market. The lack of cross-platform compatibility reduces the app's potential for adoption among both local and international tourists. Consequently, confining the app to a single platform may hinder its effectiveness as a widely accessible digital tourism tool.

Furthermore, there is also the limitation concern about the application's current onboarding process, which requires users to register or log in before they can access key features. While this approach may facilitate user tracking and personalised experiences, it can also serve as a barrier to entry, particularly for casual tourists looking for quick access to historical information. Mandatory login during the initial stage may discourage spontaneous exploration and result in user drop-off. To increase user engagement and lower entry barriers, future design improvements should consider providing guest access or delayed login options, allowing users to browse content before deciding to create an account.

6.4 Future Works

To enhance the functionality, accessibility, and user experience of the mobile application, several future improvements are proposed to address the limitations identified in the current version of the project. First, future development should include offline support capabilities to lessen the reliance on a consistent internet connection. This can be achieved by implementing local caching mechanisms that allow users to download and access essential data such as historical site descriptions, maps, and news and events articles without requiring real-time

internet access. Additionally, optimizing the app's data fetching strategies and integrating background data synchronization will help ensure that users can continue exploring the content even in low-connectivity environments.

Second, future versions should enable several languages in order to expand the accessibility and inclusiveness of the application. This can be implemented through internationalisation frameworks available in Flutter, allowing the user interface and content to be translated into other commonly used languages such as Malay and Mandarin. Incorporating a language selection feature would make the app more suitable for non-English-speaking users, thereby improving engagement and ensuring the application serves a wider and more diverse audience. Moreover, future efforts should try to expand the application to support iOS devices in order to overcome the present platform limitation. Since the application is built using Flutter which is a cross-platform framework, this enhancement can be achieved with relatively minimal changes to the codebase. Releasing the app on both the Android and iOS platforms would significantly expand its reach and improve its adoption among local and international users, contributing to a more inclusive digital tourism experience.

Finally, to improve the onboarding process and lower the barrier to entry, the application can include a guest access option. This would allow new users to explore the application's core features without requiring immediate registration, reducing friction and encouraging spontaneous use. A delayed login approach, in which users are prompted to create an account only when accessing dedicated features like participating in quizzes or news subscriptions, has the potential to increase engagement and reduce early drop-offs. Implementing this feature would improve the application's usability, particularly for casual or short-term users who are unlikely to commit to immediate registration.

6.5 Conclusion

In conclusion, this Final Year Project focused on the design and development of LelengauKch mobile application, aimed at enhancing historical exploration through digital means. The project was carried out over two semesters using the Agile software development methodology, with a particular emphasis on iterative design and continuous integration. The application was developed using the Flutter framework, targeting Android devices, and integrated with Firebase services to manage authentication, real-time data, and media storage. Chapter 1 outlined the project's background, objectives, scope, and the significance of promoting local historical tourism through technology.

Chapter 2 presented a review of existing tourism-related mobile applications, highlighting their core features, user engagement strategies, and the use of modern technologies such as geolocation, and multimedia integration. The findings from this analysis helped justify the inclusion of key features in the proposed system, such as historical site listings, integrated map module, quizzes, and news and events feed.

Chapter 3 described the system analysis and design phase, including use case modeling, activity diagrams, information architecture, and user interface prototyping. The system's architecture was based on a modular structure, incorporating both public user and admin modules. The application also adopted a state management approach using the Provider package in Flutter to maintain UI responsiveness and data consistency.

Chapter 4 detailed the implementation of each core module. This includes the Historical Site Explorer module (displaying site data fetched from Firebase), the News & Events module (allowing for real-time updates), and the Admin module (supporting CRUD operations for content management). Additional features, such as Google Maps integration for navigation and a chatbot powered by Vertex AI for user assistance, were also implemented. The project

leveraged Firebase Authentication, Firestore, and Cloud Storage, with Firebase App Check providing an added layer of security.

Chapter 5 covered the testing and evaluation of the system. Functional testing was conducted on all modules to ensure expected behaviour, while usability testing involved feedback from target users. The findings revealed that the application effectively met user needs in terms of usability, content accessibility, and interface clarity.

Collectively, the chapters demonstrate a comprehensive and structured development process that successfully produced a functional and informative historical tour guide application. The project not only met its intended objectives but also laid the groundwork for future enhancements such as iOS compatibility, offline access, multilingual support, and push notification features for improved user engagement.

References

- Braga, B. (2024, January 16). *Are travel guides obsolete? The Pros and Cons of Guided Adventure*. Travel Noire. Retrieved October 25, 2024, from <https://travelnoire.com/are-travel-guides-obsolete-pros-cons>
- Cordis, C. (2016, January 15). *App aims to make cultural heritage interesting and interactive*. CORDIS | European Commission. <https://cordis.europa.eu/article/id/118605-app-aims-to-make-cultural-heritage-interesting-and-interactive>
- Editor. (2021, January 18). Extreme programming: values, principles, and practices. *AltexSoft*. <https://www.altexsoft.com/blog/extreme-programming-values-principles-and-practices/>
- Firebase. (n.d.). *Firebase*. <https://firebase.google.com/>
- GeeksforGeeks. (2024, May 29). *What is Extreme Programming (XP)?* GeeksforGeeks. <https://www.geeksforgeeks.org/software-engineering-extreme-programming-xp/>
- GeeksforGeeks. (2025, January 3). *Activity Diagrams Unified Modeling Language (UML)*. GeeksforGeeks. <https://www.geeksforgeeks.org/unified-modeling-language-uml-activity-diagrams/>
- Google. (n.d.-a). *Dart Overview*. Dart. Retrieved December 8, 2024, from <https://dart.dev/overview#platform>
- Google. (n.d.-b). *Flutter*. Flutter - Build Apps for Any Screen. Retrieved December 8, 2024, from <https://flutter.dev/>
- IBM Corporation. (2021, April). *Use-case diagrams*. Retrieved January 14, 2025, from <https://www.ibm.com/docs/en/rational-soft-arch/9.6.1?topic=diagrams-use-case>
- Jeong, M., & Shin, H. H. (2019). Tourists' Experiences with Smart Tourism Technology at Smart Destinations and Their Behavior Intentions. *Journal of Travel Research*, 59(8), 1464–1477. <https://doi.org/10.1177/0047287519883034>

- Kenteris, M., Gavalas, D., & Economou, D. (2007). An innovative mobile electronic tourist guide application. *Personal and Ubiquitous Computing*, 13(2), 103–118. <https://doi.org/10.1007/s00779-007-0191-y>
- Kissflow. (2023, November). *What is Kanban*. <https://kissflow.com/project/agile/kanban-methodology/#What-is-Kanban-Methodology>
- Lau, R. (2018, July 13). *Sarawak's tourism mix attractive to European tourists*. Borneo Post Online. Retrieved October 23, 2024, from <https://www.theborneopost.com/2018/07/13/sarawaks-tourism-mix-attractive-to-european-tourists/>
- Liu, Y. (2020). Evaluating visitor experience of digital interpretation and presentation technologies at cultural heritage sites: a case study of the old town, Zuoying. *Built Heritage*, 4(1). <https://doi.org/10.1186/s43238-020-00016-4>
- Martins, J. (2024, January 19). What is Kanban? A Beginner's Guide for agile teams [2024] • Asana. *Asana*. <https://asana.com/resources/what-is-kanban>
- McDonald, K. (2023, October 18). *What is Extreme Programming (XP)? | Agile Alliance*. Agile Alliance |. <https://www.agilealliance.org/glossary/xp/#:~:text=given%20your%20context.-,Values,described%20in%20more%20detail%20below.>
- Mohareb, N. I., ElAdawi, A. G., & Ayad, H. M. (2008). Tourist navigation through historical sites: the spatial analysis approach. *WIT Transactions on Ecology and the Environment*. <https://doi.org/10.2495/st080061>
- Nielsen, J. (2024, February 20). *10 Usability heuristics for user interface design*. Nielsen Norman Group. <https://www.nngroup.com/articles/ten-usability-heuristics/>

- Ojoo. (2017, November 23). 5 challenges and opportunities for mobile tourism - OJOO - Medium. *Medium*. <https://medium.com/@ojoo/5-challenges-and-opportunities-for-mobile-tourism-e3ca858f1919>
- Pawar, P. S. S., Chavhan, P., Lohar, A., Kadam, A., & Ranjane, P. (2016). Android Based Tourist Guide System. *Imperial Journal of Interdisciplinary Research (IJIR)*, 2(3), 568–570. <https://www.scribd.com/document/428819723/100>
- Sim, A. (2022, December 6). *Interactive Travel App, Sarawak Travel Portal and app updated with VR, AR*. DayakDaily. <https://dayakdaily.com/interactive-travel-app-sarawak-travel-portal-and-app-updated-with-vr-ar/#:~:text=According%20to%20Sarawak%20Minister%20for,improve%20travellers%20engagement%20with%20Sarawak.>
- Smirnov, A. V., Kashevnik, A. M., Balandin, S. I., & Laizane, S. (2013). Intelligent Mobile Tourist Guide - Context-Based Approach and Implementation. *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, 8121 LNCS, 94–106. <https://dblp.uni-trier.de/db/conf/new2an/new2an2013.html#SmirnovKBL13>
- Visual Paradigm. (n.d.). *What is Class Diagram*. Retrieved January 13, 2025, from <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/>
- Yu, Y. H. (2014). Study on Intelligent Augmented Reality Tourist Guide Application based on Android Smart Phone. *Applied Mechanics and Materials*, 668–669, 1399–1402. <https://doi.org/10.4028/www.scientific.net/amm.668-669.1399>
- Zarpou, T., Drosopoulou, C., & Vlachopoulou, M. (2013). *Mapping the tourism mobile applications* (Vol. 3, pp. 206–212). Association for Computing Machinery. <https://doi.org/10.1145/2490257.2490295>

Appendix A

Questionnaire for Final Year Project (FYP) "LelengauKch" : A Kuching Historical Tour Guide Mobile Application

👋 Dear Respondent,

Thank you for taking time to participate in this questionnaire 📄.

My name is Wong Qin Yong, I am a final year Bachelor of Software Engineering (Honours) undergraduate at Faculty of Computer Science and Information Technology (FCSIT), Universiti Malaysia Sarawak (UNIMAS). I am currently working on my Final Year Project titled: "Kuching Historical Tour Guide Mobile Application. By completing this questionnaire, you acknowledge to complete this voluntarily.

The goal of this project is to create a mobile application that provides interactive and engaging Kuching historical content, which can be used as an alternative to a historical tour guide on a smartphone. "LelengauKch" aims to enhance historical exploration in Kuching by making it more interactive and engaging for locals and tourists.

The survey consists of three sections:

- **Section 1:** Demographic Information 🧑🧒
- **Section 2:** Travel and Tour Preferences 🗺️
- **Section 3:** Feedback on Proposed Mobile Application 📱

This survey complies with Malaysia's Personal Data Protection Act 2010 (PDPA). All personal information collected will be **used solely for academic purposes** and **kept confidential in accordance with the PDPA**, and the survey should take about **10 minutes** to complete. Your feedback is highly valued and will contribute to the success of this project.

If you have any questions or concerns, please contact me at 81515@siswa.unimas.my.

FYP Supervisor:

Ts. Nurfaeza bt Jali,

Senior Lecturer at FCSIT, Universiti Malaysia Sarawak (UNIMAS).

Email: jnurfaeza@unimas.my.

wqyong012@gmail.com [Switch accounts](#)



📧 Not shared

* Indicates required question

Consent form *

☐ I have clearly read and comprehended the information presented above. I voluntarily consent and offer to participate in this survey. By checking the box on this consent form, I certify that all of the information I have provided is true and correct to the best of my knowledge and will be used solely for academic purposes.

Section 1: Demographic Information 🧑🏻

This section consists of information regarding about yourself and your background.

Gender *

- ☐ Male
- ☐ Female

Age Group *

- ☐ Under 18
- ☐ 18 - 24
- ☐ 25 - 34
- ☐ 35 - 44
- ☐ 45 - 54
- ☐ 55 and above

Nationality *

- ☐ Malaysian
- ☐ Non-Malaysian

Occupation *

- ☐ Student
- ☐ Working Professional
- ☐ Retired

Education Level *

- ☐ High School
- ☐ Diploma
- ☐ Bachelor's Degree
- ☐ Master's Degree/ Ph.D.

General Familiarity with Kuching, Sarawak *

- ☐ Very familiar
- ☐ Somewhat familiar
- ☐ Not familiar at all

Familiarity with Historical Landmarks in Kuching, Sarawak *

- ☐ Very familiar
- ☐ Somewhat familiar
- ☐ Not familiar at all

Section 2: Travel and Tour Preferences

This section gathers insights into your travel habits and preferences, especially regarding historical tours and mobile applications.

How often do you travel to historical sites? *

- ☐ Often
- ☐ Occasionally
- ☐ Rarely
- ☐ Never

What mode of historical tours do you prefer? *

- ☐ Guided tours
- ☐ Self-guided tours
- ☐ No preference

Do you use mobile application for guiding the tour? *

- ☐ Yes, frequently
- ☐ Yes, occasionally
- ☐ No, never

What are the important features you prefer in a tour guide mobile application? (Select all that apply) *

- ☐ Interactive maps
- ☐ Information on historical sites
- ☐ Quizzes and interactive content
- ☐ Chatbot for queries
- ☐ News and promotions
- ☐ Other: _____

What motivates you to visit historical sites? *

☐ Learning about historical significant of the landmarks

☐ Photography

☐ Family/Friends outing

☐ Personal interest

☐ Other: _____

What problems do you face when visiting historical sites? (Select all that apply) *

☐ Lack of information about historical sites

☐ Difficult navigation to sites

☐ Lack of engagement (interactive or engaging experiences)

☐ Unavailability of guided tours or tour guides

☐ Difficulty in understanding the historical context

☐ Other: _____

Section 3: Preferences and Expectations on Proposed Mobile Application

This section focus on user needs, problems faced, and suggestions.

Do you currently use any historical tour guide mobile app? *

- ☐ Yes
☐ No

How can a mobile application enhance your historical tour experience? (Select all that apply) *

- ☐ Providing detailed information on historical sites
☐ Offering interactive maps for easy navigation
☐ Including quizzes and interactive content for engagement
☐ Providing a chatbot for answering questions
☐ Sharing news and promotions related to historical sites
☐ Other: _____

What specific features would you like to see in a historical tour guide mobile app? *
(Select all that apply)

- ☐ Interactive maps for navigation
☐ Information on historical sites
☐ Quizzes and interactive content
☐ Chatbot for queries
☐ Sharing news and promotions related to historical sites
☐ Other: _____

How important would you find feature of **interactive maps providing navigational direction** for historical sites during historical tour? *

- 1 2 3 4 5
not important ☐ ☐ ☐ ☐ ☐ very important

How important is it for a historical tour guide app to provide **detailed information on historical sites**? *

- 1 2 3 4 5
not important ☐ ☐ ☐ ☐ ☐ very important

How important is it for a historical tour guide app to **include quizzes and interactive content** to keep users engaged? *

- 1 2 3 4 5
not important ☐ ☐ ☐ ☐ ☐ very important

How important is it for a historical tour guide app to provide **chatbot for queries**? *

- 1 2 3 4 5
not important ☐ ☐ ☐ ☐ ☐ very important

How important is it for a historical tour guide app to provide **sharing of news and promotions** related to historical sites? *

- 1 2 3 4 5
not important ☐ ☐ ☐ ☐ ☐ very important

Do you have any additional feedback or suggestions for a historical tour guide mobile application?

Your answer _____

[Back](#)

[Next](#)

[Clear form](#)

You have reached the end of the survey 📌

Happy New Year 2025 ! May this year bless us with success and health! 🥳

Thank you! Have a nice day! All the best! 🙌



[Back](#)

[Submit](#)

[Clear form](#)

LelengauKch: A Kuching Historical Tour Guide Mobile Application – User Experience Feedback Form

👋 Dear Respondent,

Thank you for taking the time to participate in this User Acceptance Testing (UAT) survey!

I am Wong Qin Yong, a final-year undergraduate student pursuing a Bachelor of Software Engineering (Honours) at the Faculty of Computer Science and Information Technology (FCSIT), Universiti Malaysia Sarawak (UNIMAS). This survey is part of my Final Year Project titled:

“Kuching Historical Tour Guide Mobile Application”.

The Android mobile application – **LelengauKch** – is developed to promote historical tourism in Kuching by offering historical site listings, an AI-powered chatbot, in-app navigation, and cultural-themed user interface on Android devices.

🎯 Purpose of this Survey:

This User Acceptance Testing aims to gather feedback from users to assess the application’s usability, performance, feature effectiveness, and overall satisfaction. Your responses will help evaluate whether the system meets user expectations and identify potential improvements.

🌱 Survey Structure:

- Section 1: Participant Demographics
- Section 2: Application Usability & Features
- Section 3: Overall Satisfaction & Suggestions

🕒 Estimated time to complete: **Approximately 10 minutes**

 This survey complies with the Malaysian Personal Data Protection Act 2010 (PDPA). All responses are anonymous and will be used strictly for academic purposes.

 For inquiries, please contact me at: **81515@siswa.unimas.my**

FYP Supervisor: Ts. Nurfaufa bt Jali

Senior Lecturer, FCSIT, UNIMAS

Email: **jnurfaufa@unimas.my**

wqyong012@gmail.com [Switch accounts](#)



 Not shared

* Indicates required question

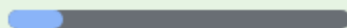
Consent form:

*

I have clearly read and comprehended the information presented above. I voluntarily consent and offer to participate in this survey. By agreeing to this consent form, I certify that all of the information I have provided is true and correct to the best of my knowledge and will be used solely for academic purposes.

☐ Agree

[Next](#)



Page 1 of 6

[Clear form](#)

Section 1: Demographic Information

This section consists of information regarding about yourself and your background.

Age Group *

- ☐ Under 18
- ☐ 18 - 24
- ☐ 25 - 34
- ☐ 35 - 44
- ☐ 45 - 54
- ☐ 55 and above

Nationality *

- ☐ Malaysian
- ☐ Non-Malaysian

Familiarity with Historical Landmarks in Kuching, Sarawak *

- ☐ Very familiar
- ☐ Somewhat familiar
- ☐ Not familiar at all

Section 2: Application Usability Evaluation 📱

This section gathers your feedback on the usability, functionality, and design of the LelengauKch mobile application. Your responses will help assess how effectively the app supports user interaction, navigation, content delivery, and overall user experience.

Please rate the following statements based on your experience using the LelengauKch mobile application. Use the scale below:

- 1 - Strongly Disagree
- 2 - Disagree
- 3 - Neutral
- 4 - Agree
- 5 - Strongly Agree

The app is easy to install and open. *

1	2	3	4	5
☐	☐	☐	☐	☐

The navigation between pages is intuitive. *

1	2	3	4	5
☐	☐	☐	☐	☐

The navigation between pages is intuitive. *

1	2	3	4	5
☐	☐	☐	☐	☐

The information on historical sites is clear and informative. *

1	2	3	4	5
☐	☐	☐	☐	☐

The quiz module is easy to find and access from the main menu. *

1	2	3	4	5
☐	☐	☐	☐	☐

The questions provided are relevant to the historical content featured in the app. *

1	2	3	4	5
☐	☐	☐	☐	☐

I enjoyed interacting with the quiz and found it engaging. *

1	2	3	4	5
☐	☐	☐	☐	☐

The Google Maps navigation works correctly and helps in route guidance. *

1

☐

2

☐

3

☐

4

☐

5

☐

The chatbot feature provides helpful responses. *

1

☐

2

☐

3

☐

4

☐

5

☐

The content displayed in the News and Events section is relevant and informative. *

1

☐

2

☐

3

☐

4

☐

5

☐

I feel more engaged with the app because of the regularly updated content in this section. *

1

☐

2

☐

3

☐

4

☐

5

☐

I did not encounter technical errors or bugs during usage. *

1

☐

2

☐

3

☐

4

☐

5

☐

The app performs smoothly without lag. *

1

☐

2

☐

3

☐

4

☐

5

☐

Section 3: Feature-Specific Feedback

This section focuses on your interaction with key features of the LelengauKch mobile application, such as the historical site listings, chatbot, navigation, news feed, and quiz module. Your responses will help identify which features are most effective, which require improvement, and how well the app meets its intended goals.

Which features did you use or experience? *

- ☐ Site Listings
- ☐ Quiz
- ☐ Map and Navigation to Sites
- ☐ News & Events
- ☐ Account module (Email Verification, Reset Password)
- ☐ Other: _____

Were there any features that did not work as expected? *

Your answer _____

How useful did you find the chatbot in answering your queries? *

- ☐ Very useful
- ☐ Somewhat useful
- ☐ Neutral
- ☐ Not so useful
- ☐ Not useful at all

Section 4: Overall Satisfaction

This section aims to evaluate your overall impression of the LelengauKch mobile application. Your feedback will help determine the general acceptance, perceived usefulness, and likelihood of continued use or recommendation to others. The responses collected here are critical in assessing whether the application fulfills its objectives and meets user expectations.

Rate your overall satisfaction with the application. *

Linear scale (1–5)

(1 – Very Dissatisfied, 5 – Very Satisfied)

1	2	3	4	5
☐	☐	☐	☐	☐

Would you recommend this application to others? *

- ☐ Yes
- ☐ No
- ☐ Maybe

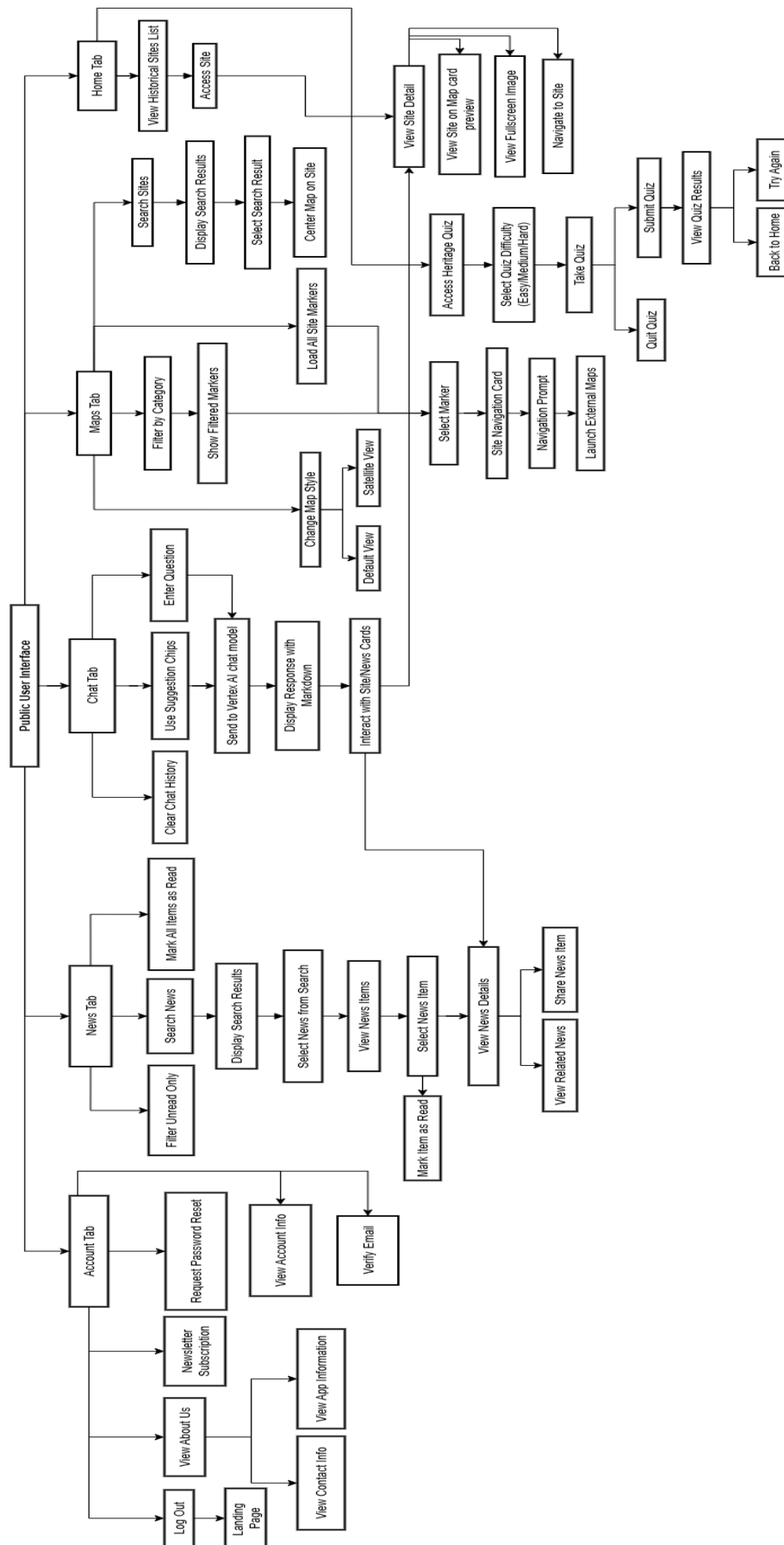
Suggestions for improvements if any?

Your answer

Any other comments or feedback?

Your answer

Appendix C



Appendix D

