Find Your Way: An IOS-Based Travel Planning Application With Route Optimization Using Agile Methodology

.

ABSTRACT

|  |
| --- |
| **Aims:** This study aimed to develop Find Your Way, an iOS-based travel planning application that addressed user needs through the Agile development approach.  **Study design:** The study applied Agile methodology, consisting of six iterative phase such as planning, design, development, testing, deployment, and review, executed over two development cycles to refine the application based on user feedback.  **Place and Duration of Study:** The research was conducted in Indonesia over a five-month period, from December 2024 to April 2025.  **Methodology:** Researchers conducted interviews with four participants aged 18–35 to gather functional requirements. The data were analyzed using thematic analysis to identify key user needs. These needs guided the design and implementation of the application using Swift and MVVM Clean Architecture. The core features developed included route optimization using Particle Swarm Optimization (PSO), destination recommendations, saving and managing travel routes, and customizable map settings. Testing was conducted using blackbox and whitebox methods. The app was deployed via TestFlight using CI/CD with Xcode Cloud. In the review phase, 30 users aged 18–25 completed a User Acceptance Test (UAT) to evaluate the app's usability and performance.  **Results:** The study produced Find Your Way, a travel planning application tailored to user-identified functional needs. Five key features were implemented, including destination information, location and route recommendations, as well as storage and map display settings. The application achieved a UAT score of 97.33%, indicating a high level of user satisfaction. Testing confirmed the application's functionality and program logic through blackbox and whitebox methods.  **Conclusion:** The Find Your Way application was successfully developed and met user functional requirements. The high UAT score of 97.33% demonstrated that the app effectively fulfilled its intended purpose, validating the Agile development approach used in this study. |

*Keywords: Find Your Way, Particle Swarm Optimization, Agile Methodology, Optimal Route, Navigation Application, User Acceptance Test*

1. INTRODUCTION

Generation Z, the generation born between the mid-1990s and early 2010s, is known for its high curiosity, adventurous spirit, and passion for hobbies such as traveling. For Gen Z, traveling is more than just a leisure activity—it serves as a means of seeking new experiences, exploring cultures, and expressing themselves (Medcom, 2017). Recent studies show a rising trend in Gen Z’s interest in travel, with 76% of them admitting to being more enthusiastic about traveling than before. Additionally, 63% of them reported that the primary motivation for vacationing was for “healing” or stress relief, underlining the role of travel in supporting mental health and emotional well-being (Sabri, 2025).

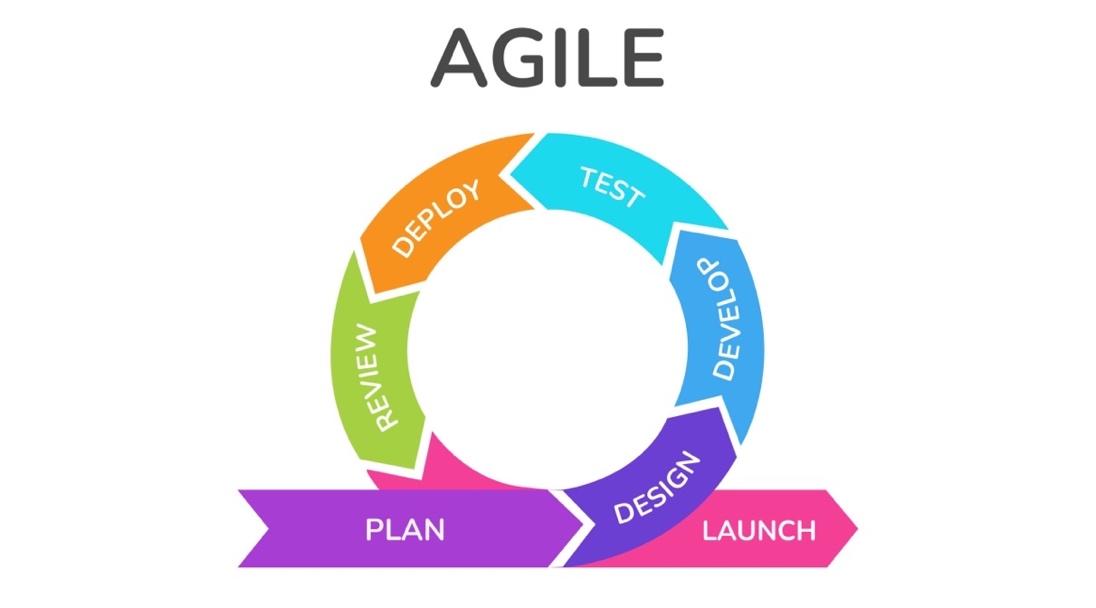
Despite this enthusiasm, Gen Z often faces financial limitations. Due to their relatively young age and limited fixed income, they are generally more mindful of their expenses—including vacation-related costs. As a result, Gen Z travelers seek cost-effective solutions, particularly in determining travel routes and choosing tourist destinations (Medcom, 2017).

Interviews conducted with four Gen Z individuals revealed several common challenges in travel planning. One major issue was time inefficiency and mismatched distances between destinations, which often led to unplanned changes or cancellations in their itineraries. They also expressed difficulty in identifying suitable tourist destinations near their location, resulting in time-consuming planning processes.

In response to these challenges, this study aims to develop the Find Your Way application as an innovative solution that caters to the functional needs of Gen Z travelers. Specifically, this study has three main objectives: (1) to identify the functional features required by users through qualitative interviews analyzed thematically, (2) to develop the application based on the findings from these interviews, and (3) to assess the application's usability and user satisfaction using a quantitative User Acceptance Testing (UAT) framework involving 30 respondents. The final application is designed to recommend optimal travel routes with estimated travel times and nearby tourist destinations tailored to user preferences, thereby improving the efficiency and overall experience of travel planning.

2. methodology

The Find Your Way application development method uses the SDLC (Software Development Life Cycle) approach by implementing the Agile methodology. The Agile methodology is the framework adopted to ensure smoothness and flexibility in the development of this system. The Agile methodology offers an iterative and collaborative approach, allowing developers to respond to changing needs and ensure the quality of the resulting software (Abrahamsson et al., 2017). The Agile methodology will be carried out 2 iterations with the first iteration focusing on developing applications according to user functional needs and the second iteration focusing on developing applications based on user feedback. The agile methodology flow diagram can be seen in Fig. 1



**Fig. 1 Agile Methodology Flow Diagram On Each Iteration**

**2.1 Planning**

The Planning Stage in Agile Methodology serves as the foundation for understanding user needs and preparing a development plan for future iterations (Parsons & MacCallum, 2019). In the development of the Find Your Way application, a qualitative approach was used in this stage to explore user needs in depth. This involved conducting semi-structured interviews to collect detailed insights, which were then interpreted using thematic analysis to identify recurring patterns and themes that would inform the application’s features.

The first iteration involved in-depth interviews with four respondents aged 18–25 years, who represent the target demographic of the application. These respondents were selected using purposive sampling to ensure relevance to the application's intended user base which young people who frequently engage in traveling. The selection aimed to include individuals with varying travel habits and planning preferences to gather diverse insights. The interview questions, as shown in Table 1, were designed to explore how users currently plan their travel, the challenges they face, and their expectations from a travel planning application.

**Table 1 Stage 1 Interview Question**

|  |  |
| --- | --- |
| **Code** | **Question** |
| Q1 | How do you plan your travel itinerary? |
| Q2 | Is it important for you to know the shortest route on your trip? |
| Q3 | Do you take into account travel time between destinations when planning your travel? |
| Q4 | Is it important for you to know the amount of time you can spend at each tourist spot? |
| Q5 | Have you ever been confused in determining a tourist destination? |
| Q6 | If yes, what factors do you usually consider in determining tourist destination recommendations? |
| Q7 | Have you ever used a travel planning app before? |
| Q8 | If yes, what do you think about the application? |

The responses were analyzed using thematic analysis, which revealed several recurring needs and pain points among users. First, in terms of travel planning behavior, users generally plan their trips manually by using maps and estimating travel time, highlighting the need for an automated system that can suggest optimal travel routes based on distance and time efficiency to help save time and money (Q1–Q4). Second, users often felt confused when choosing tourist destinations, typically considering factors such as proximity and the type of tourist spot, indicating the importance of a system that can recommend destinations tailored to user preferences (Q5–Q6). Lastly, most respondents reported never having used a dedicated travel planning application, instead relying on tools like Google Maps and social media, suggesting a lack of existing solutions that fully meet their needs (Q7–Q8). These findings directly informed the key features required in the application: route optimization based on distance and travel time, personalized destination recommendations according to user interests and location, and integrated destination information such as attraction type, pricing, and images.

**2.2 Design**

At the design stage, the interface design of the Find Your Way application uses the iOS Design System, namely the iOS UI Kit which is officially provided by Apple. The selection of this Design System aims to adjust to the characteristics and standards of the iOS user environment, thus providing a consistent, intuitive user experience that is in accordance with platform design principles (Inc, 2025). The design process begins with the creation of a wireframe or low fidelity design as an initial framework that describes the structure and navigation flow of the application. Furthermore, the design is further developed into a high fidelity design with more detailed and realistic visual and interactive elements, as a reference in the user interface implementation stage in the application development process (Abrahamsson et al., 2017).

**2.3 Development**

At the development stage, the application design implementation process will be carried out in the form of codes (Parsons & MacCallum, 2019). This implementation process will focus on developing iOS applications using Swift and several tech stacks that can be seen in Table 2

**Table 2 Tech Stack Application**

|  |  |
| --- | --- |
| **Type** | **Name** |
| Framework | iOS Native Development |
| Platform | iOS |
| Databases | Swift Data |
| Packages | Google Maps iOS SDK, Google Maps Place iOS SDK, Google Maps Navigation API, SwiftUI, UIKit, SwiftData, Combine, and Foundation |
| Architecture Patter | MVVM (Model-View-View Model) |
| Design Pattern | Repository Pattern, Delegates Pattern, and Dependency Injection |
| VCS | Git (Local) and Github (Remote) |
| Design | Figma |
| Tools | XCode, Simulator, and iOS SDK |
| CI/CD | XCode Cloud |
| Deployment | App Store Connect |

In addition, in the development phase, especially in the route optimization feature, the PSO algorithm will be used. This algorithm was chosen because of its more economical computational cost efficiency compared to other methods with the same efficiency such as Glowworm Swarm Optimization (GSO) (Azmi et al., 2019)

**2.4 Testing**

At the testing stage, a quantitative approach is applied using secondary data to evaluate the application’s performance and ensure that both its functionality and internal logic operate as expected. The testing process combines two main techniques: blackbox testing and whitebox testing. In blackbox testing, the Equivalence Partitioning technique is used to group inputs into equivalent classes, enabling the testing of the application's responses to a variety of input conditions without requiring knowledge of the internal code structure (Wijaya & Astuti, 2021). This helps assess how the system handles different user scenarios from an external perspective. Meanwhile, in whitebox testing, the Basis Path Testing method is employed to examine the internal logic flow of the program. This technique allows for the identification and testing of each independent logical path within the code, ensuring that all algorithmic components—including the Particle Swarm Optimization (PSO) implementation—function accurately and efficiently (Londjo, 2021). The combination of these testing methods ensures a comprehensive evaluation of the application's reliability from both functional and structural perspectives.

**2.5 Deploy**

At the deployment stage, the process of implementing the Find Your Way application into the production environment is carried out by utilizing the CI/CD (Continuous Integration/Continuous Deployment) service provided by Xcode Cloud. The use of Xcode Cloud enables automation in the build, testing, and distribution processes of applications to test devices and the App Store. With this CI/CD integration, every change to the source code sent to the repository will be immediately tested and built automatically, minimizing the risk of manual errors and accelerating the application release cycle. In addition, Xcode Cloud also supports efficient team collaboration by providing real-time build reports and test results, which helps developers to immediately detect and fix bugs before the application is launched to the public (Inc, 2025).

**2.6 Review**

The Find Your Way application underwent a User Acceptance Testing (UAT) process that utilized a quantitative, descriptive approach with mean average analysis to evaluate user satisfaction and system usability. The UAT was distributed to 30 respondents aged 18–25 years, the target demographic of the application. This testing phase was conducted twice: first, after the implementation of user needs identified in iteration 1, and again after the implementation of iteration 2. The purpose of this stage was to collect structured feedback from users regarding their experience with the application’s interface, functionality, and features. The results and average scores from each question were used to determine the overall level of user acceptance and identify areas for improvement in the subsequent iteration. The questions used in this stage are presented in Table 3.

**Table 3 UAT Question**

|  |  |
| --- | --- |
| **Code** | **Question** |
| Q1 | Is the application user interface (UI) of this application easy to understand and user friendly? |
| Q2 | Did you find that the application was responsive to user input and performed its tasks smoothly without any hiccups or errors? |
| Q3 | Did you find it easy and did you experience few technical difficulties while using this application? |
| Q4 | Do you find the optimal route determination feature of this app easy to use? |
| Q5 | Are the nearby tourist recommendations on the app relevant and helpful in finding interesting tourist destinations? |
| Q6 | Do you feel that this application is reliable in providing travel recommendations and information related to travel routes? |

**2.7 Launch**

At the launch stage, the Find Your Way application is released to the public through Apple's official platform, namely App Store Connect. Through this service, applications that have been developed and tested are then packaged and sent for review by Apple. App Store Connect allows the development team to manage application metadata, such as descriptions, icons, screenshots, and application categories. After the application passes the Apple review process and is declared to meet all the provisions of the App Store Guidelines, the Find Your Way application is officially launched to the public through the App Store, so that it can be downloaded and used by iOS device users (Inc, 2025).

3. results and discussion

**3.1 Planning**

Determination of user functional needs was conducted by interviewing 4 people with an age range of 18-25 years. The summary of the interview results can be seen in table 1

**Table 4 Summary of User Functional Needs Interview Results**

|  |  |  |
| --- | --- | --- |
| **Codes** | **Question** | **Summary Result** |
| Q1 | How do you plan your travel itinerary? | Generally, the resource person will search for the tourist destinations they want to visit first, either through social media, destination wishlists, or suggestions from friends/family. Then, some resource persons will plan the trip either by determining the location or travel time from each destination. |
| Q2 | Is it important for you to know the shortest route on your trip? | All sources answered that it was important because it could save time and several sources also said that the shortest route could save travel costs, especially fuel costs. |
| Q3 | Do you take into account travel time between destinations when planning your travel? | All sources took into account travel time between destinations, because according to several sources, travel time between destinations is needed to find out how long it will take us to travel or arrive at our destination, and one source said that he wanted to visit... |
| Q4 | Is it important for you to know the amount of time you can spend at each tourist spot? | All sources answered that it was important because it was used to find out the duration of the tour and estimate how many places could be visited in one day. In addition, one source argued that travel time could also be used as a reference in knowing when the source would continue the tour to the next tourist destination. |
| Q5 | Have you ever been confused in determining a tourist destination? | All sources answered that they had felt confused in determining tourist destinations. |
| Q6 | If yes, what factors do you usually consider in determining tourist destination recommendations? | All sources answered that the location of the destination and type (restaurant, mall, zoo, etc.) were the main references for determining tourist destinations. Several sources also argued that the price and photos of the destination were also references in choosing a destination. |
| Q7 | Have you ever used a travel planning app before? | All of the interviewees answered that they had never used a travel planning application, but several interviewees said that they tended to use assistive applications such as Google Maps and social media such as TikTok and Instagram, to help plan routes and provide recommendations for places. |
| Q8 | If yes, what do you think about the application? | Several sources said that the Google Maps application is very useful in finding locations, routes and travel times from destinations you want to visit, and the Social Media application is used to provide recommendations for tourist destinations. |

Based on the interview results in Table 1, users still plan their travel manually by considering the location and travel time between destinations based on Q1. They consider the importance of determining the shortest route because it can save time and money based on Q2, and always consider the travel time and duration of visits to each tourist spot based on Q3 and Q4. This shows the need for a system that can recommend optimal travel routes based on distance and time.

In Q5 and Q6, all respondents admitted that they had been confused about choosing a destination and considering the location and type of tourist spot. This indicates the need for a system that recommends destinations according to user preferences, especially the type of tourism and the closest distance.

In addition, from Q7 and Q8, it is known that users have never used a special travel planning application, but rely on Google Maps and social media. Therefore, the application needed must be able to help with travel planning while providing complete information such as types of tourism, locations, prices, and images.

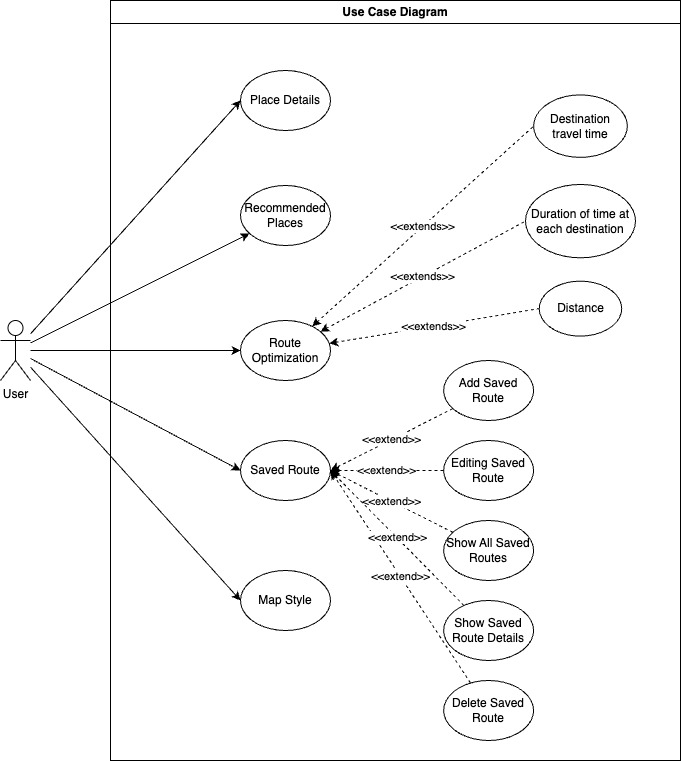
In addition, at the review stage in iteration 1, there was some feedback from users based on the UAT results in iteration 1. The UAT results stated that users wanted to be able to save routes that had been inputted into the application so that they could be used later. In addition, there was also feedback where users also wanted to be able to see several other types of maps according to user needs such as satellite and terrain maps. This feedback will be processed into user needs to be implemented in the application.

The conclusion of functional requirements is described in Table 5

**Table 5 User Functional Requirements**

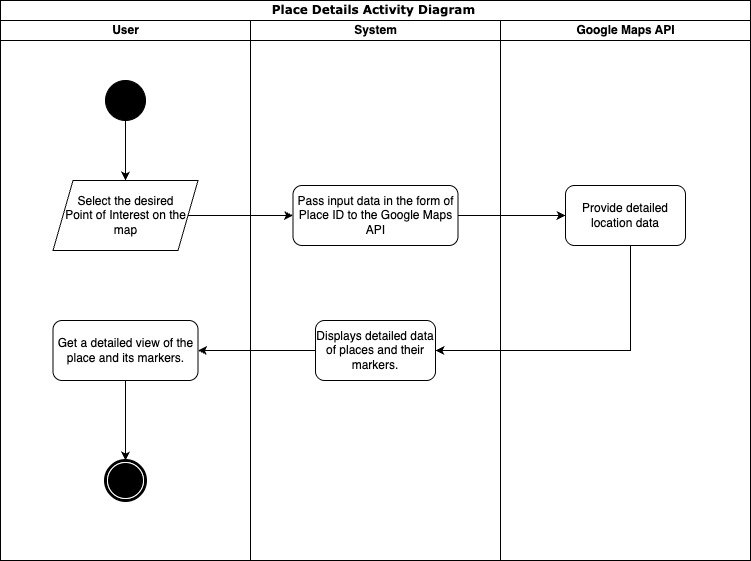
|  |  |
| --- | --- |
| **Codes** | **User Functional Requirement** |
| KF-01 | Users need a system that can be used to view tourist destination information and plan travel practically in the same application. |
| KF-02 | Users need a system that can recommend tourist destinations according to their wishes based on the closest distance. |
| KF-03 | Users need a system that can recommend travel routes from several destinations based on the shortest distance, estimated travel time between locations, and the duration of time spent at each destination. |
| KF-04 | Users need a system that allows them to save the inputted travel routes so that they can be accessed and reused at a later date. |
| KF-05 | Users need a system that allows them to change the type of map display according to their preferences, such as standard, satellite, or terrain maps. |

Based on the user's needs, a use case diagram of the application can be prepared as in Figure 2.



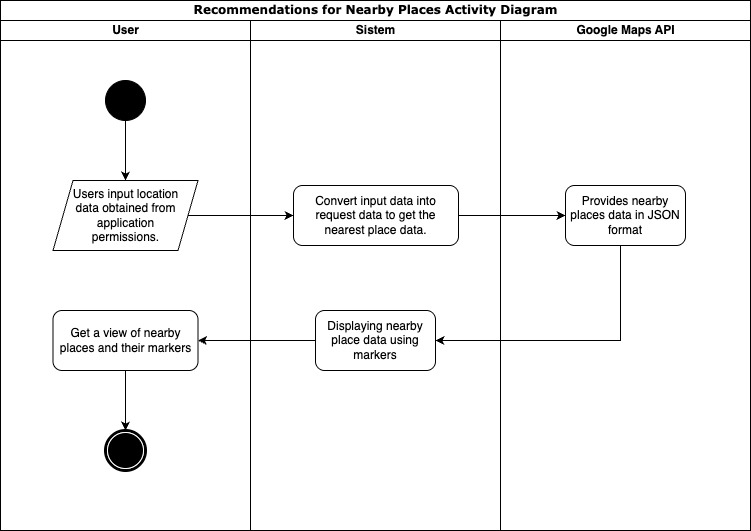
**Fig. 2 Use Case Diagram**

In the use case diagram in Figure 2, there is 1 actor, namely the user of the Find Your Way application and it has 5 main features based on the user's functional needs, namely recommendations for nearby tourist attractions based on categories entered by the user which refer to KF-01, detailed tourist information which refers to KF-02, route optimization from several tourist destinations which refer to KF-03, saved route features which refer to KF-04, and map style features which refer to KF-05.



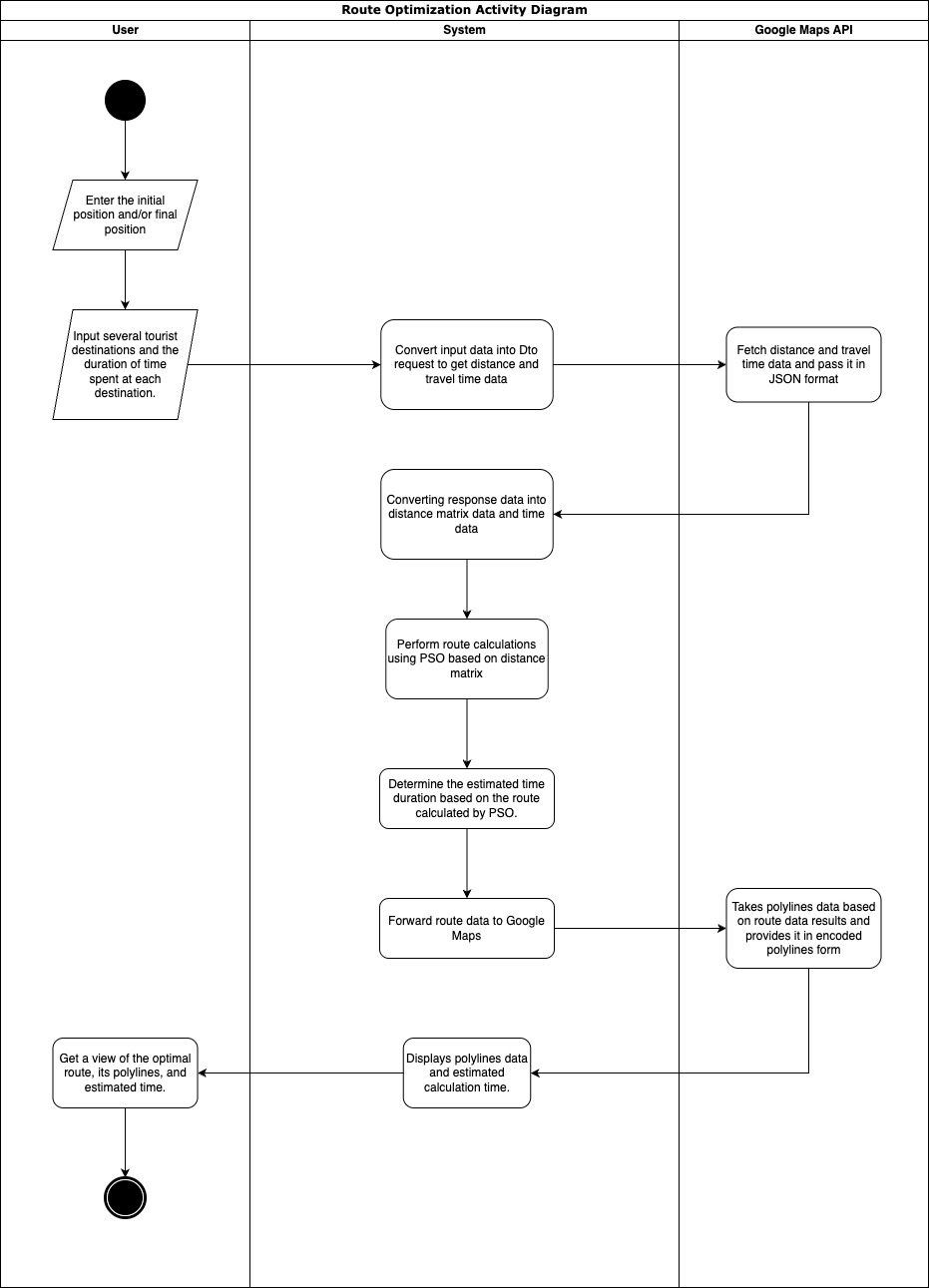
**Fig. 3 Place Details Activity Diagram**

The Place Detail Activity Diagram at Figure 3 illustrates the flow when a user wants to view details of a place or Point of Interest (POI) in the application. The process begins when the user opens the interactive map view and explores the available POIs. After selecting a POI by tapping on an icon or marker, the system retrieves the PlaceID, a unique identifier from Google Maps, and sends it to the Google Maps API to request details of the place. The API responds in JSON format containing information such as the place name, full address, operating hours, ratings, number of reviews, and photos (if any). This data is then parsed and displayed in an easy-to-read interface. The POI marker is also displayed back on the map as visual context for the user.



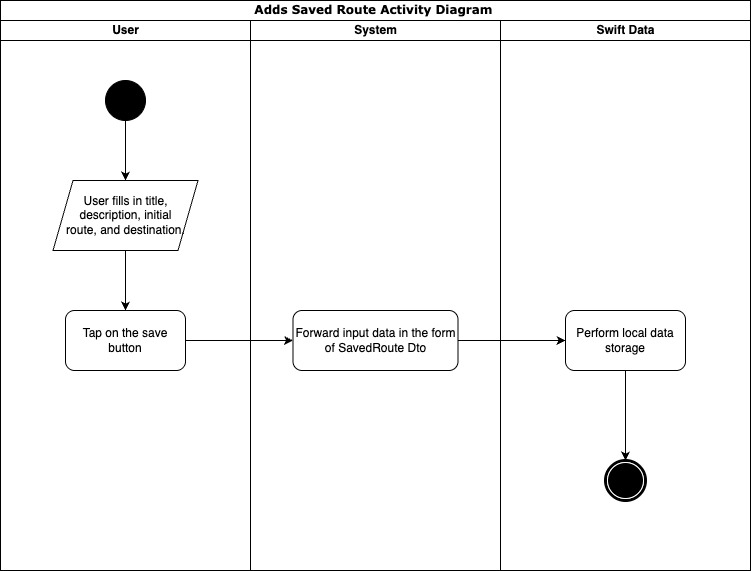
**Fig. 4 Recommendations for Nearby Place Activity Diagram**

Based on the Activity Diagram in Figure 4, the process of finding the nearest place begins when the user grants location access permission to the application. After obtaining the user's location through the permission given, the system will convert the location data into a format suitable for sending as a request to the Google Maps API. This request aims to obtain information about the nearest places from the user's current position. Next, the Google Maps API will respond by returning data in JSON format containing a list of the nearest places along with their details. The system then processes the data and displays it to the user via a map display by adding markers to each nearest location found. With this display, users can easily see and select places that are relevant to their needs directly from the application map.



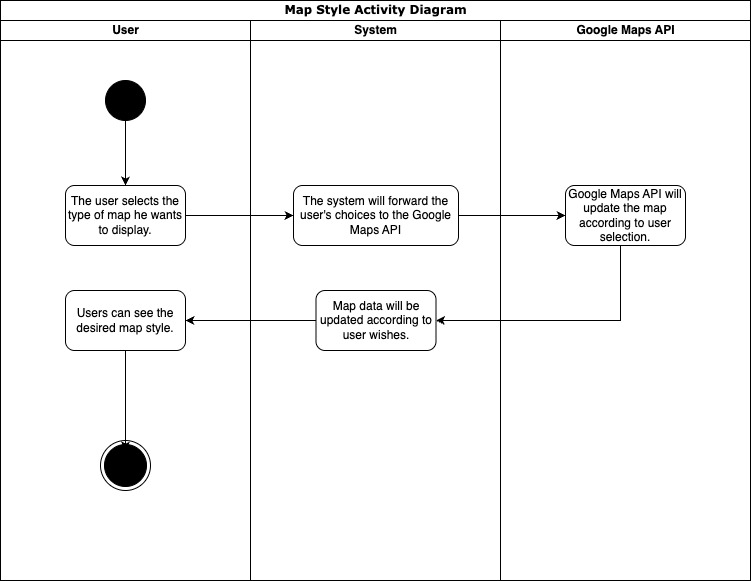
**Fig. 5 Route Optimization Activity Diagram**

The Activity Diagram in Figure 5 explains the workflow of the travel route optimization feature in the system. The process begins when the user accesses this feature and enters data such as starting point, destination point, list of tourist destinations, and estimated duration of visit at each location. After the input is complete, the system forms a request to the Google Maps API to obtain distance information and estimated travel time between destinations. The API will respond in JSON format containing distance and travel time matrix data, which is then parsed by the system. Distance data is used in the route optimization process using the Particle Swarm Optimization (PSO) algorithm to determine the most efficient visit sequence. Meanwhile, travel time and visit duration data are used to calculate the estimated total travel time. Finally, the calculation results are displayed as a route visualization in the form of polylines on the map, depicting the complete travel path from start to finish.



**Fig. 6 Saved Route Activity Diagram**

In Figure 6, the user is asked to fill in all the required data before saving the route, such as title, description, initial position, destination, duration time at each destination, and/or final position of the route to be saved. After all the information is filled in completely, the user will press the "Save" button to start the saving process. Next, the data that has been input will be processed by the system and sent to local storage using Swift Data, which acts as a data persistence medium in the application. By using this approach, the system ensures that the data entered by the user is stored securely

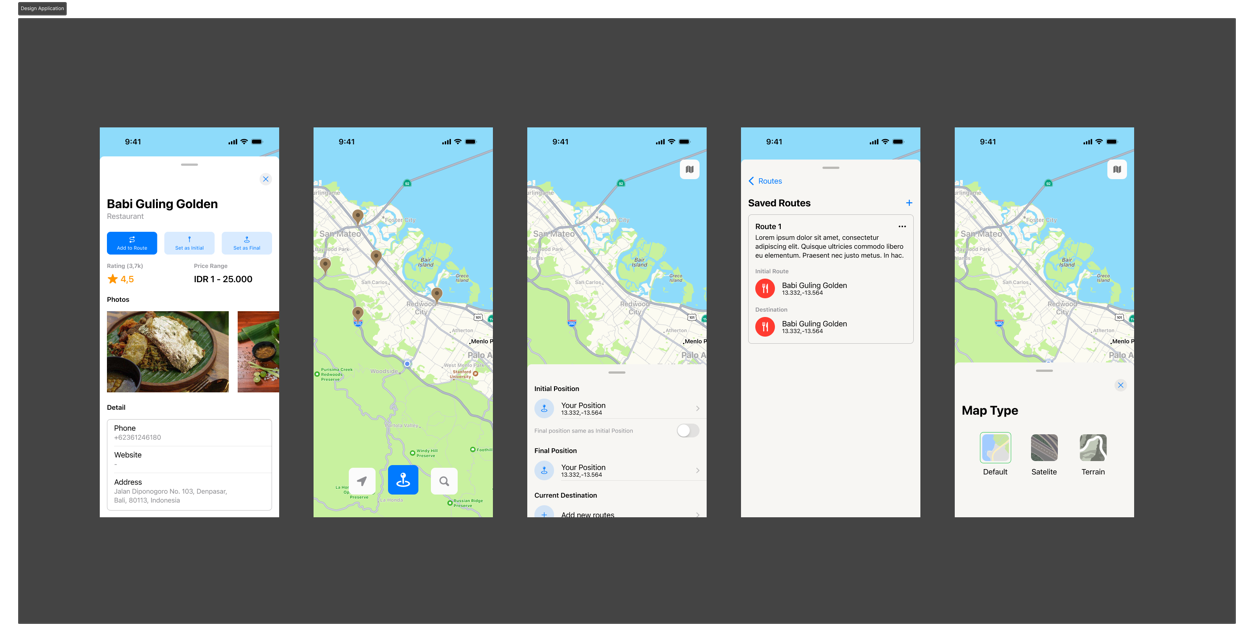


**Fig. 7 Map Style Activity Diagram**

Based on the Activity Diagram in Figure 7, the map style change process begins when the user selects the desired map type through the application interface. This choice can be various types of maps such as standard maps, satellite, or terrain available in the settings options. After the user makes a selection, the system will capture the input and forward it as a request to the Google Maps API. Google Maps API then processes the request by changing the map style according to the user's choice. After the map style change is complete, the API will send map update data back to the system. The system then displays the map with the new style in real-time on the application screen, so that users can immediately see the map display that has been adjusted to their preferences. This process ensures responsive interaction and a more personalized user experience in using the application. In addition, the system also saves the map style preferences selected by the user so that they can be applied automatically for subsequent use. Thus, users do not need to reset every time they open the application, thus providing convenience and comfort in long-term use. This feature also increases application efficiency by reducing the burden of repetitive data processing.

**3.2 Design**

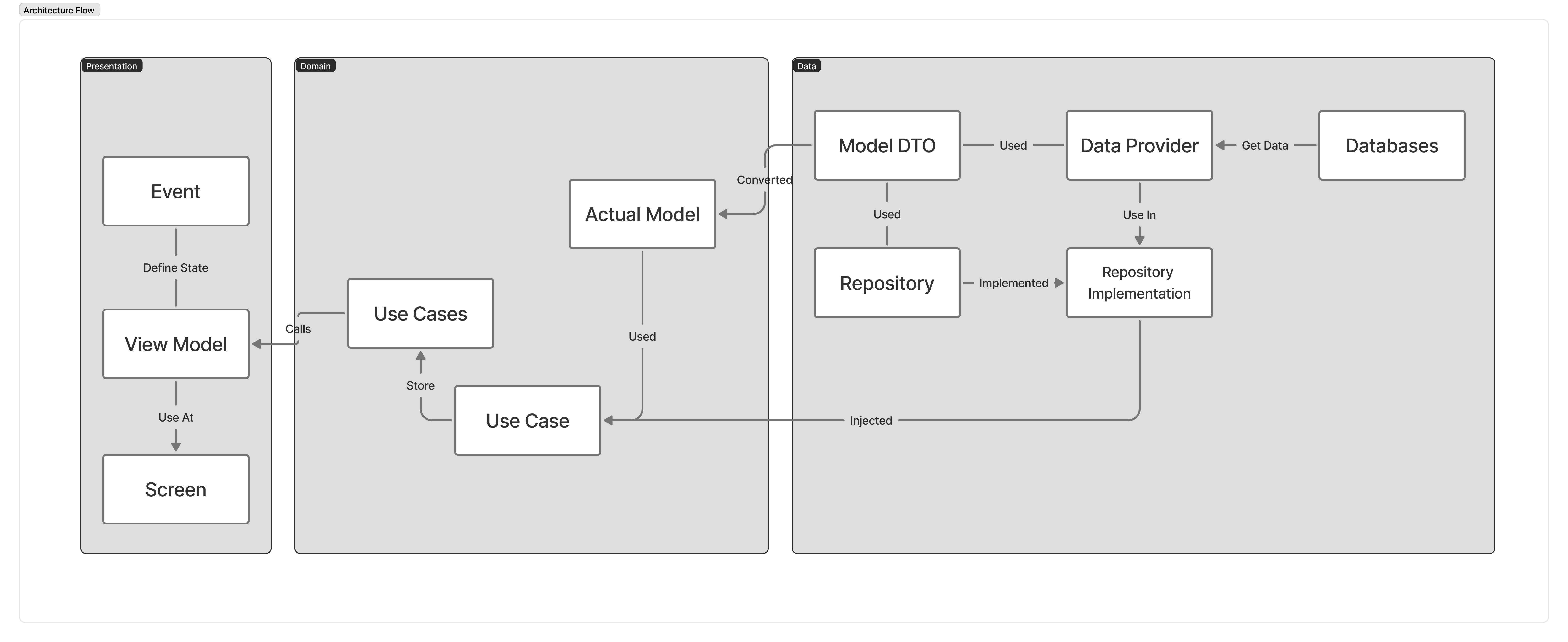
For the application design, it can be seen in Figure 8, where the application design focuses on a single view application to make it easier for users to navigate. Users will always have full access to the map and some features will be tucked away using modal sheets, so that users can focus on the map utility.



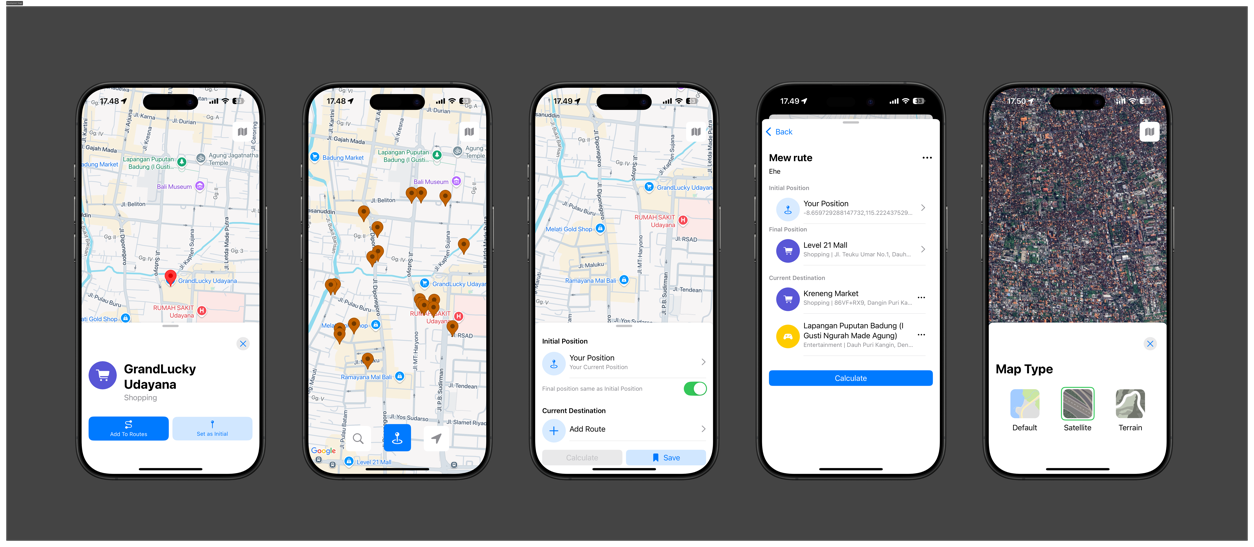
**Fig. 8 Design Application**

**3.3 Development**

Application development stage, using MVVM Clean Architecture architecture which is divided into 3 main folders, namely presentation, domain, and data. The presentation folder is used to store view related code such as screen, viewModel, event, and widgets. The domain folder is used to store protocol files such as repository protocol, actual model, and also application business logic. And data is used to store files related to data providers such as Dto, repository implementation, and data providers. The architecture flow application can be seen on Fig. 9 and for the implementation can be seen on Fig. 10



**Fig. 9 Architecture Flow Application**



**Fig. 10 Application Implementation**

In addition, there is a main feature developed in this application, namely travel route optimization using the Particle Swarm Optimization (PSO) algorithm. This algorithm works by initializing a number of particles as potential solutions from a sequence of cities visited, with particle positions arranged randomly and initial speed set to zero. Each particle's total distance is calculated using the calculateDistance function, then the pBest (personal best) and gBest (global best) values are determined. During iteration, the particle's speed and position are updated based on the pBest and gBest values by considering social constants (c1), cognitive (c2), and inertia factors (w). This process is repeated until the shortest route is obtained. The optimal route is then used to calculate the estimated travel time and visualized in the form of polylines on the map in the application.

**3.4 Testing**

The implemented Find Your Way application was tested using blackbox and whitebox testing to test the application's functionality. Blackbox testing of the application can be seen in Table 6 -10 which focuses on providing validation based on the input and output provided to the application. The results of blackbox testing show that all test scenarios provided are acceptable as they should be and meet the application's functionality.

**Table 6 Result Detail Place Test**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Code** | **Test Case** | **Expected Result** | **Actual Result** | **Summary** |
| T1 | User clicks on one of the Points of Interest | Markers and detail modals will appear and display data from that point. | Markers and detail modals appear as expected. | Accepted |
| T2 | While the marker and detail modal are still visible, the user clicks any point on the map. | The marker will disappear and the detail modal will close. | The marker disappears and the detail modal closes. | Accepted |
| T3 | While the marker and modal details are still visible, the user closes the modal. | The marker will disappear and the detail modal will close. | The marker disappears and the detail modal closes as expected. | Accepted |

**Table 7 Search Nearby Test Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Code** | **Test Case** | **Expected Result** | **Actual Result** | **Summary** |
| T1 | User clicks search and looks for the nearest search category. | A marker will appear and the location will appear on the map. | The marker appears and the location appears on the map as expected. | Accepted |
| T2 | User clicks on Point of Interest marker | The recommendation marker will disappear and the detail modal will appear. | The recommendation marker is gone and the detail modal appears as expected. | Accepted |

**Table 8 Route Optimization Test Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Code** | **Test Case** | **Expected Result** | **Actual Result** | **Summary** |
| Q1 | The user adds the destination, initial position, and/or final position and clicks the calculate button. | Route calculation and time estimate will be visible and the result modal will appear. | Route calculation and time estimation are visible and the result modal will appear. | Accepted |
| Q2 | User did not add any destination and did not add any final position at all. | The calculate button will be disabled | Calculation button disabled as expected | Accepted |

**Table 9 Saved Route Test Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Code** | **Test Case** | **Expected Result** | **Actual Result** | **Summary** |
| T1 | The user fills in the title, description, initial position, and destination and clicks save. | The route will be saved and displayed in the SavedRoute modal. | The route is saved and appears in the SavedRoute modal as expected. | Accepted |
| T2 | User did not fill in any of the fields | The capital will be closed and a pop up will appear if all the mandatory fields are filled in. | The capital is closed and a pop up appears if all the mandatory fields are filled in as expected. | Accepted |

**Table 10 Map Style Test Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Code** | **Test Case** | **Expected Result** | **Actual Result** | **Summary** |
| T1 | User selects the type of map desired | Maps will be changed to user's choice | Maps are changed to user's choice as expected | Accepted |

The test cases presented across Tables 6 to 10 reflect a comprehensive evaluation of the app’s core functionalities, ensuring that each feature performs as intended. The Place Test cases verify user interactions with Points of Interest (POI), ensuring markers and detail modals respond correctly to user inputs such as clicks and modal closures. The Search Nearby Test ensures that the application accurately identifies and displays nearby locations based on user-selected categories. The Calculate Route Test validates the route planning feature, confirming that routes and time estimates appear only when the necessary inputs are provided. The Saved Route Test evaluates the route-saving mechanism, checking that valid data is correctly saved and displayed, while incomplete input is appropriately handled with prompts. Finally, the Map Style Test confirms that users can customize the visual presentation of the map according to their preferences. Overall, these test cases align closely with the app’s goal of providing an interactive and user-friendly map-based navigation experience.

Meanwhile, the results of whitebox testing carried out using the basis path technique are presented in Table 11, and show that all logical paths in the program have been tested and produce appropriate output. Based on the results of both types of testing, it can be concluded that the route recommendation feature functions well and has met the acceptance criteria.

**Table 11 White Box Test Result**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Region** | **Condition** | **Input** | **Expected Output** | **Actual Output** | **Summary** |
| 1. | No iteration (⁠maxIter = 0⁠) | w: 1.0, c1: 1.0, c2: 1.0, numParticles: 2, maxIter: 0, distanceMatrix: [[0, 10], [10, 0]] | [0, 1, 0] | [0, 1, 0] | Accepted |
| 2. | All conditions are false | w: 0.0, c1: 0.0, c2: 0.0, numParticles: 2, maxIter: 1, distanceMatrix: [[0, 10, 20], [10, 0, 5], [20, 5, 0]] | [0, 1, 0] | [0, 1, 0] | Accepted |
| 3. | Only pbest update | w: 0.5, c1: 2.0, c2: 0.0, numParticles: 2, maxIter: 1, distanceMatrix: [[0, 1, 5], [1, 0, 1], [5, 1, 0]] | [0, 1, 0] | [0, 1, 0] | Accepted |
| 4. | Only gbest update | w: 0.5, c1: 0.0, c2: 2.0, numParticles: 2, maxIter: 1, distanceMatrix: [[0, 4, 3], [4, 0, 2], [3, 2, 0]] | [0, 1, 2, 0] | [0, 1, 2, 0] | Accepted |
| 5. | pbest and gbest update | w: 0.5, c1: 2.0, c2: 2.0, numParticles: 2, maxIter: 1, distanceMatrix: [[0, 1, 3, 4], [1, 0, 1, 2], [3, 1, 0, 1], [4, 2, 1, 0]] | [0, 1, 2, 3, 0] | [0, 1, 2, 3, 0] | Accepted |
| 6. | Iteration > 1 with combination of conditions | w: 0.5, c1: 2.0, c2: 2.0, numParticles: 3, maxIter: 5, distanceMatrix: [[0, 2, 9, 10], [2, 0, 6, 4], [9, 6, 0, 3], [10, 4, 3, 0]] | [0, 1, 3, 2, 0] | [0, 1, 3, 2, 0] | Accepted |

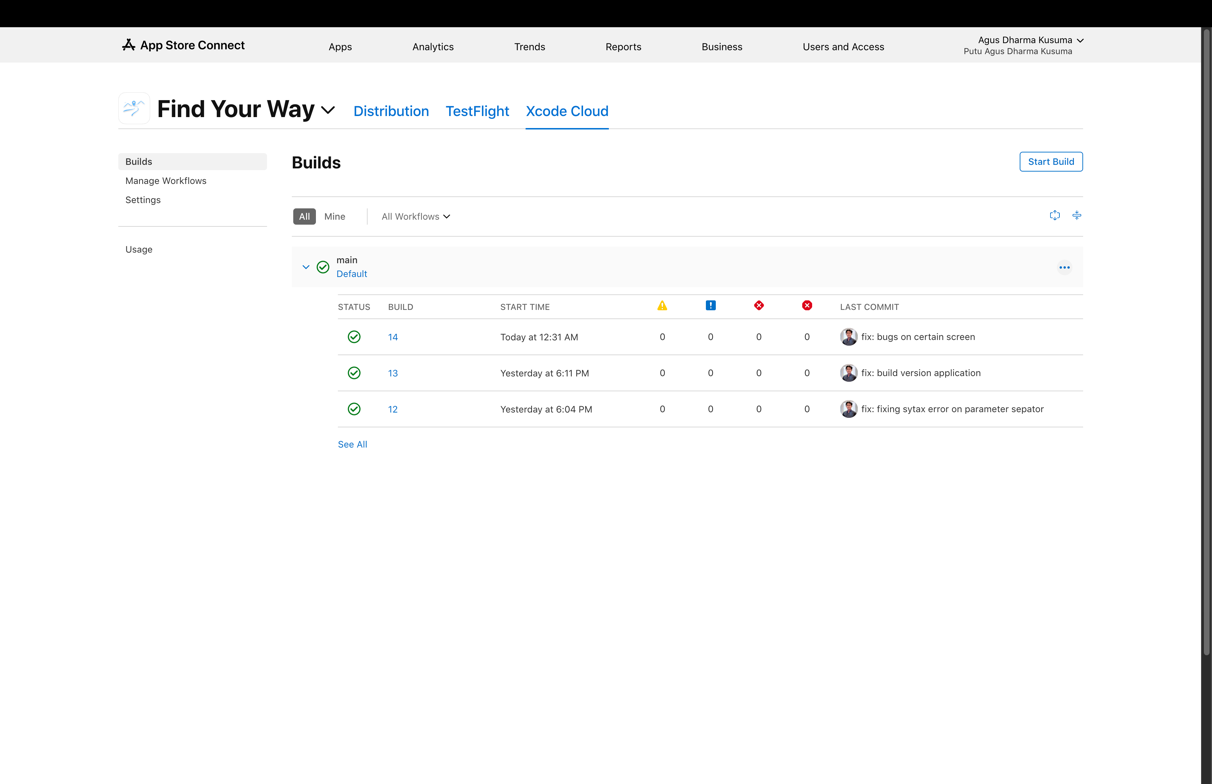
The white box test cases in Table 11 are designed to ensure that all possible execution paths and logical conditions within the route optimization algorithm function correctly and do not produce bugs or incorrect results. By covering scenarios such as no iterations, all conditions being false, updates to only personal best (pbest) or global best (gbest), and combinations of those updates across multiple iterations, the tests comprehensively validate the internal workings of the algorithm.

This testing approach ensures that regardless of input variations or parameter configurations, the algorithm consistently returns correct and reliable results. It plays a critical role in guaranteeing the stability and robustness of the app’s calculation feature, preventing potential errors that could affect route accuracy or app behavior.

**3.5 Deploy**

The application deployment process is carried out by utilizing the TestFlight feature available in the App Store Connect. TestFlight functions as an application distribution platform in the testing phase, allowing developers to send application builds to test users before the application is officially launched to the public. In this way, feedback from user testing can be collected to improve and refine the application before the final release.

In this process, the application build is created and deployed automatically using the CI/CD pipeline integrated with Xcode Cloud, as shown in Figure 6. The use of Xcode Cloud enables automation in building, testing, and distributing applications more efficiently and consistently. This ensures that any code changes can be immediately tested by the team and user testers through TestFlight, so that the development and testing process runs faster and more structured.



**Fig. 11 XCode Cloud CI/CD**

**3.6 Review**

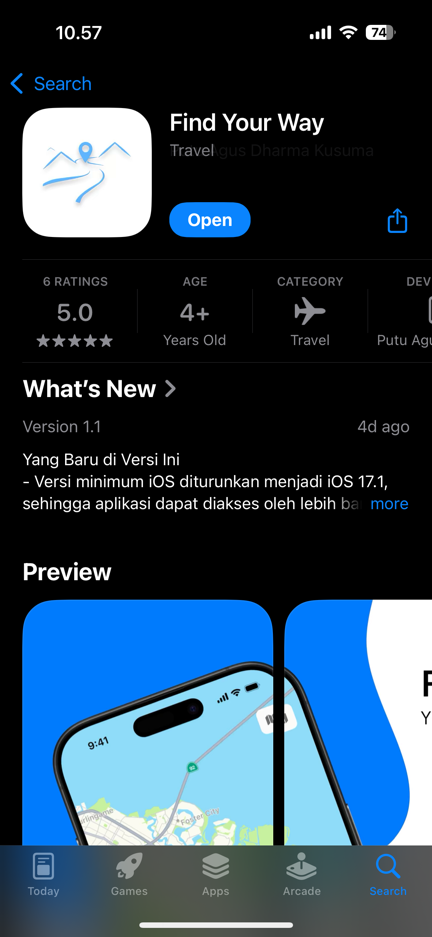
Based on the lastest results of the User Acceptance Test (UAT) at Table 12, that conducted on 30 respondents aged 18–25 years, the final score was 97.33%. This score indicates that the majority of respondents feel that the Find Your Way application has met their needs in planning a practical and efficient travel trip. Achieving this value is a strong indicator that the developed system is able to provide a user experience that is in accordance with the expectations and functional needs that have been identified at the planning stage. Thus, it can be concluded that this application has succeeded in answering user problems.

**Table 12 UAT Result**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Question** | **Variable** | **Bobot** | **Respondents Total** | **Actual Score** |
| Is the application user interface (UI) of this application easy to understand and user friendly? | Strongly Agree | 5 | 23 | 115 |
| Agree | 4 | 7 | 28 |
| Netral | 3 | 0 | 0 |
| Disagree Less | 2 | 0 | 0 |
| Disagree | 1 | 0 | 0 |
| **Total** | | | **30** | **143** |
| Did you find that the application was responsive to user input and performed its tasks smoothly without any hiccups or errors? | Strongly Agree | 5 | 26 | 130 |
| Agree | 4 | 4 | 16 |
| Netral | 3 | 0 | 0 |
| Disagree Less | 2 | 0 | 0 |
| Disagree | 1 | 0 | 0 |
| **Total** | | | **30** | **146** |
| Did you find it easy and did you experience few technical difficulties while using this application? | Strongly Agree | 5 | 25 | 125 |
| Agree | 4 | 5 | 20 |
| Netral | 3 | 0 | 0 |
| Disagree Less | 2 | 0 | 0 |
| Disagree | 1 | 0 | 0 |
| **Total** | | | **30** | **145** |
| Do you find the optimal route determination feature of this app easy to use? | Strongly Agree | 5 | 26 | 130 |
| Agree | 4 | 4 | 16 |
| Netral | 3 | 0 | 0 |
| Disagree Less | 2 | 0 | 0 |
| Disagree | 1 | 0 | 0 |
| **Total** | | | **30** | **146** |
| Are the nearby tourist recommendations on the app relevant and helpful in finding interesting tourist destinations? | Strongly Agree | 5 | 27 | 135 |
| Agree | 4 | 3 | 12 |
| Netral | 3 | 0 | 0 |
| Disagree Less | 2 | 0 | 0 |
| Disagree | 1 | 0 | 0 |
| **Total** | | | **30** | **147** |
| Do you feel that this application is reliable in providing travel recommendations and information related to travel routes? | Strongly Agree | 5 | 29 | 145 |
| Agree | 4 | 1 | 4 |
| Netral | 3 | 0 | 0 |
| Disagree Less | 2 | 0 | 0 |
| Disagree | 1 | 0 | 0 |
| **Total** | | | **30** | **149** |
| **% Total Score** | | | | **97,33%** |

**3.7 Launch**

After that the application can be published on the App Store platform. The application that has been developed will be reviewed again by Apple and if it is still lacking, Apple will provide feedback on the application and if it is safe, the application can be directly accessed to the public. The application can be downloaded in <https://apps.apple.com/id/app/find-your-way/id6744858864>



**Fig. 12 App Store Launch Application**

4. Conclusion

Based on the results of testing and evaluation, it can be concluded that the Find Your Way application has been successfully implemented and performs optimally. The application passed both functional testing using the blackbox method and program logic validation through the whitebox method. Additionally, it achieved a high User Acceptance Test (UAT) score of 97.33% from 30 respondents aged 18–25 years, indicating that the application effectively meets the functional needs of its target users.

These results demonstrate the application's strong potential to enhance the travel planning experience for young adults. However, further studies could explore long-term user engagement, usability improvements, and the integration of additional features such as real-time collaboration or AI-based itinerary suggestions. Future development could also consider expanding testing to a broader demographic to ensure the application's adaptability and effectiveness across diverse user groups.

**Disclaimer (Artificial intelligence)**

Author(s) hereby declare that NO generative AI technologies such as Large Language Models (ChatGPT, COPILOT, etc.) and text-to-image generators have been used during the writing or editing of this manuscript.

References

Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2017). *Agile Software Development Methods: Review and Analysis* (Version 1). arXiv. https://doi.org/10.48550/ARXIV.1709.08439

Azmi, A. U., Hidayat, R., & Arif, M. Z. (2019). PERBANDINGAN ALGORITMA PARTICLE SWARM OPTIMIZATION (PSO) DAN ALGORITMA GLOWWORM SWARM OPTIMIZATION (GSO) DALAM PENYELESAIAN SISTEM PERSAMAAN NON LINIER. *Majalah Ilmiah Matematika Dan Statistika*, *19*(1), 29. https://doi.org/10.19184/mims.v19i1.17263

Inc, A. (2025). *Apple Developer*. Apple Developer. https://developer.apple.com/

Londjo, M. F. (2021). IMPLEMENTASI WHITE BOX TESTING DENGAN TEKNIK BASIS PATH PADA PENGUJIAN FORM LOGIN. *Jurnal Siliwangi Seri Sains dan Teknologi*, *7*(2), Article 2. https://doi.org/10.37058/jssainstek.v7i2.4086

Medcom, id. (2017, October 6). *Ini Perbedaan Preferensi Travelling Tiap Generasi*. medcom.id. https://www.medcom.id/rona/wisata-kuliner/3NOEl4yk-ini-perbedaan-preferensi-travelling-tiap-generasi

Parsons, D. T., & MacCallum, K. (Eds.). (2019). *Agile and lean concepts for teaching and learning: Bringing methodologies from industry to the classroom*. Springer.

Sabri, H. A. (2025). *30+ Statistik dan Tren Perjalanan Gen Z [Pembaruan 2025]—Bisniswisata*. https://bisniswisata.co.id/30-statistik-dan-tren-perjalanan-gen-z-pembaruan-2025/

Wijaya, Y. D., & Astuti, M. W. (2021). PENGUJIAN BLACKBOX SISTEM INFORMASI PENILAIAN KINERJA KARYAWAN PT INKA (PERSERO) BERBASIS EQUIVALENCE PARTITIONS. *Jurnal Digital Teknologi Informasi*, *4*(1), Article 1. https://doi.org/10.32502/digital.v4i1.3163