UNIVERSITY MALAYSIA TERENGGANU

FACULTY OF OCEAN ENGINEERING TECHNOLOGY & INFORMATICS

[CSM3114]

FRAMEWORK-BASED MOBILE APPLICATION DEVELOPMENT

INDIVIDUAL PROJECT 2 FRAMEWORK

(SMART TRAVEL PLANNER)

PREPARED BY :

FARIS ISKANDAR BIN ABD RAHMAN (S62371)

PREPARED FOR :

DR. MOHAMAD NOR HASSAN

[BACHELOR OF SCIENCE COMPUTER(MOBILE COMPUTING)]

SEMESTER I 2023/2024

Table of Contents

1 EXECUTIVE SUMMARY

ABC Technologies recognizes the growing importance of mobile applications in the contemporary IT business landscape and has outlined a strategic initiative to develop a series of innovative solutions. These mobile applications aim to enhance corporate operations, providing remote access for customers and transforming the accessibility and personalization of business services. The proposed mobile applications, Smart Travel Planner app. This app is designed to offer a comprehensive set of features, such as user authentication through login and registration, itinerary creation, cost estimation, and a dedicated Q&A section. The Smart Travel Planner app aligns with the company's vision of leveraging mobile technology to streamline and enrich the travel planning experience for users.

In conclusion, ABC Technologies' focus on developing innovative mobile applications underscores its dedication to staying ahead in the IT industry. The Smart Travel Planner app, with its diverse functionalities, is poised to bring about a significant positive change in how users plan and manage their travel experiences. This strategic move aligns with the broader goal of making business services more accessible, easy to use, and personalized for an enhanced customer experience.

2 USE CASE FOR SMART TRAVEL PLANNER APP

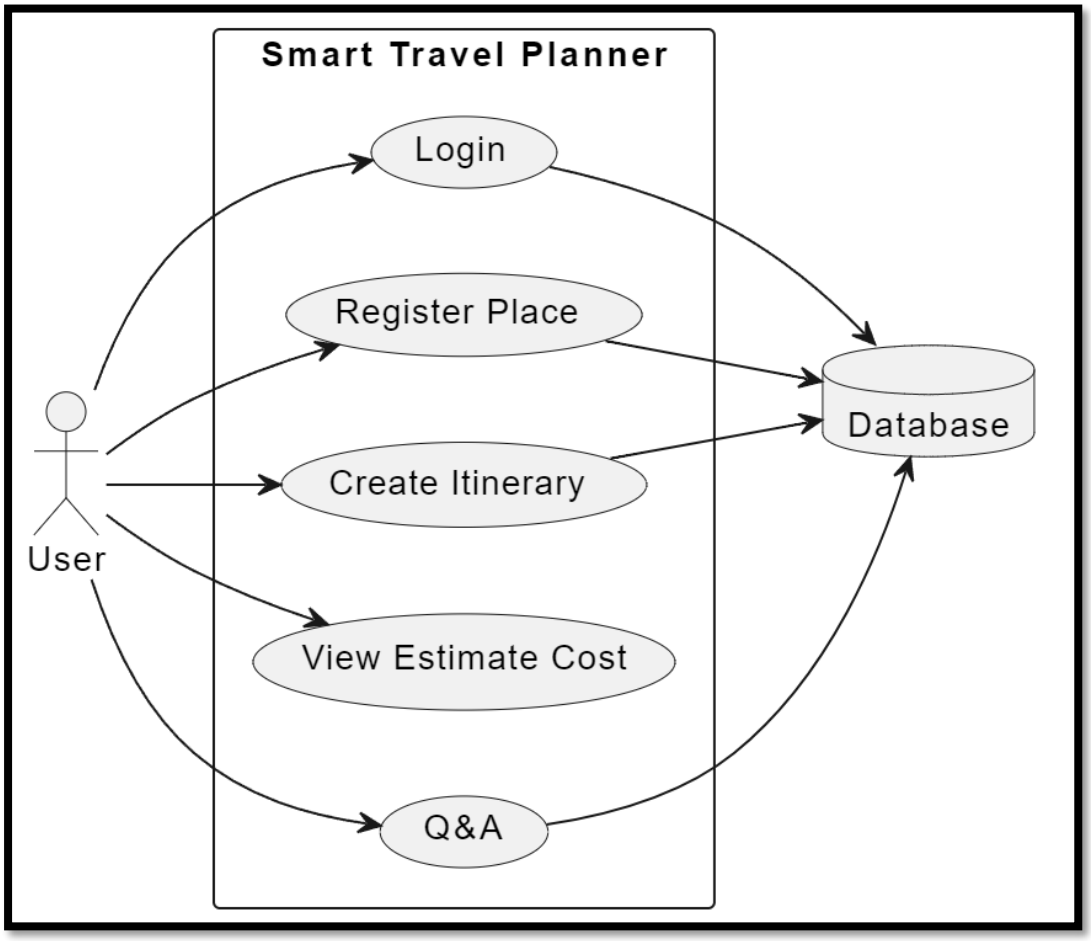

Figure 1 : Smart Travel Planner Use Case

The user starts by logging in or registering. Once logged in, they can create an itinerary by entering their desired destinations and dates of travel. The planner then uses a database of travel information to generate an itinerary that includes transportation, accommodation, and activities. The user can view the estimated cost of the trip and ask questions about the itinerary using the Q&A feature.

3 THE COMMON STRUCTURE TREE WIDGET

Figure 3: Widget Tree (Register Screen)

4 FLUTTER WIDGETS AND FEATURES ADOPTED IN APPLICATION

1. **Scaffold**: Provides a basic structure for the visual interface of your app.
2. **AppBar**: Displays the app bar at the top of the screen.
3. **Text**: Displays text on the screen.
4. **Icon**: Represents icons in the app.
5. **SafeArea**: Ensures that the child is safely within the visual area.
6. **Padding**: Adds padding around its child.
7. **SingleChildScrollView**: Enables scrolling if the content overflows the screen.
8. **ListView**.builder: Creates a scrollable, linear array of widgets.
9. **InkWell**: Recognizes taps by the user.
10. **Container**: A box model that allows you to decorate its child with various visual properties.
11. **DecorationImage**: Defines a background image decoration.
12. **Column**: A widget that arranges its children vertically.
13. **Row:** A widget that arranges its children horizontally.
14. **DropdownButton**: Creates a dropdown button to select a value from a list.
15. **DatePickerWidget**: A custom widget, presumably for selecting dates.
16. **GestureDetector**: A widget that detects gestures made by the user.
17. **SnackBar**: A temporary, informative message that appears at the bottom of the screen.
18. **PageRouteBuilder**: Builds the route for navigation.
19. **MaterialPageRoute**: A modal page route that uses a material page transition.
20. **Drawer**: A slide-in menu from the left side of the screen.
21. **BoxDecoration**: Allows for the decoration of the container.
22. **TextButton**: A button with plain text.
23. **Theme**: A widget that holds the current theme data.
24. **TextStyle**: Defines text styles.
25. **AlertDialog**: A dialog with a title and body.
26. **showDialog**: A function to show dialog boxes.
27. **BoxShadow**: Adds a shadow to the box.
28. **Border**: Adds borders to the container.
29. **CircularProgressIndicator**: Displays a circular loading indicator.
30. **DataTable**: Displays information in Table format
31. **TabController**: Manages tab selection between TabBar and TabBarView

5 SAMPLE INTERFACE AND EXPLANATION

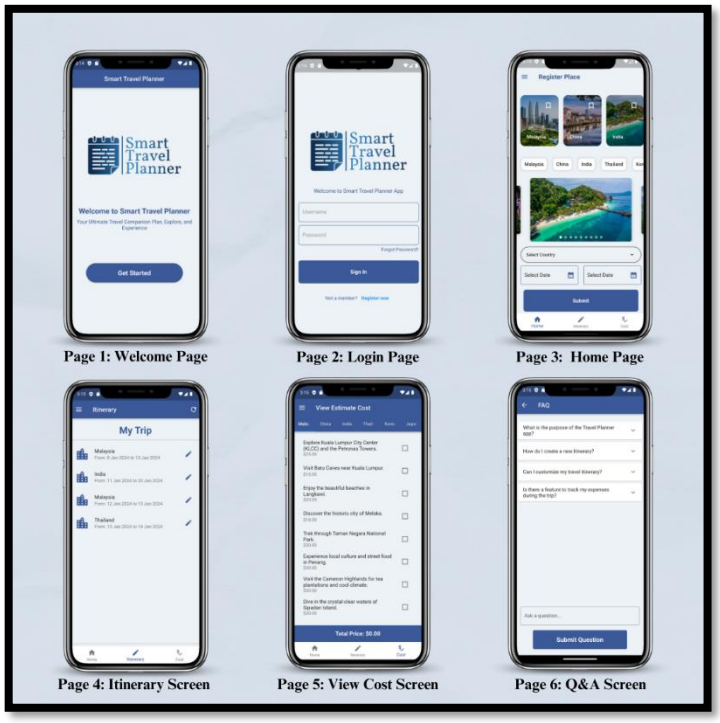

Figure 1: Interface of Application

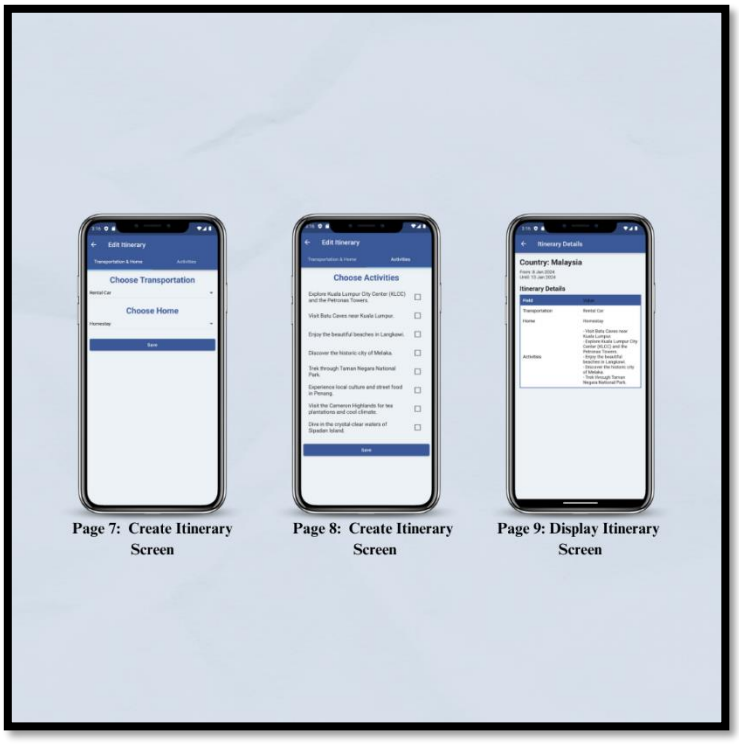

Figure 2: Interface of Application 2

Explanation (Figure 1)

Page 1 (Welcome Page) : The first page of the Smart Travel Planner Application is Welcome page, the app will welcoming with the app logo and "**Get Started**" button to Login the app.

Page 2 (Login Page) : The second page is the Login Page, allow user to login the app using existing account or create new account for new user by clicking 'Register now' text at the bottom.

Page 3 (Home Page) : The interface for home page of Smart Travel Planner Image using image container to enhance the view of UI to give best first impression for user, this page allows users to register place by select country and travel date to create itinerary for their journey.

Page 4 (Itinerary Screen): After the users register their chosen place, user can create itinerary for their journey by go to the Itinerary Screen, Itinerary Screen will show the list of place that register by user, at the list have icon edit that navigate users to the new page for create or update their itinerary. When users click the list, it will display the details of itinerary that their choose.

Page 5 (View Cost Screen) : The next page is the view cost screen, which is users can view estimate cost for their journey before creating itinerary. The page show various activity based on the country.

Page 6 (Q&A Screen) : The next page is question and answer page, which is, it display all the frequently asked question and the for the apps, users also can ask a question by write it inside the box and submit the question, so admin can answer it as soon as possible.

Explanation (Figure 2)

Page 7 & 8 (Create Itinerary Screen) : When user click the edit icon in the list at itinerary screen, it will navigate users to new screen to create an itinerary. Users can choose the transportation and home to put in their itinerary, the next tab allow users to choose an activities to put in their itinerary.

Page 9 (Display Itinerary Screen) : After user save the itinerary, users can display it back by clicking the list at the Itinerary screen, it will display all the information about their vacation place.

6 CONCLUSION

In summary, making a Smart Travel Planner app would be a great tool for travelers. The app is easy to use and has lots of helpful features for planning, exploring, and enjoying trips, making it a convenient all-in-one solution. With its focus on personalization and suggestions, the app can make travel experiences more enjoyable and unique. Overall, a Smart Travel Planner app could change the game in the travel industry, making it simpler and more fun for people to explore the world. In the broader context, a Smart Travel Planner app has the potential to revolutionize the travel industry. Its ability to simplify and enhance the entire journey from conceptualization to execution makes it a game-changer. By providing users with a comprehensive and cohesive platform, the app aims to redefine how people perceive and engage with travel, making the exploration of the world a more enjoyable, accessible, and personalized endeavour.

7 REFERENCE

Napoli, M. L. (2019). *Beginning flutter: A Hands On Guide to App Development*. John Wiley & Sons.

carousel_slider | Flutter Package. (n.d.). Dart Packages. https://pub.dev/packages/carousel_slider/example

image_card | Flutter Package. (n.d.). Dart Packages. https://pub.dev/packages/image_card

BottomNavigationBar class - material library - Dart API. (n.d.). https://api.flutter.dev/flutter/material/BottomNavigationBar-class.html

TabController class - material library - Dart API. (n.d.). https://api.flutter.dev/flutter/material/TabController-class.html

ExpansionTile class - material library - Dart API. (n.d.). https://api.flutter.dev/flutter/material/ExpansionTile-class.html

https://firebase.google.com/docs/reference/rest/database

https://chat.openai.com/c/40b9ce41-abb4-481d-be96-270637bc7de8

Darji, P. (2024, January 1). *Change AppBar color in flutter – the RIGHT way*. FlutterBeads. https://www.flutterbeads.com/change-appbar-color-in-flutter/

8 GITHUB LINK

https://github.com/ismifariezz/CSM3114