



UNIVERSITY MALAYSIA TERENGGANU
FACULTY OF OCEAN ENGINEERING TECHNOLOGY & INFORMATICS

[CSM3114]

FRAMEWORK

INDIVIDUAL PROJECT 2 REPORT

Prepared by:

DANIAL SOLEHIN B SAFIE (S62091)

Prepared for:

DR MOHAMAD NOR BIN HASSAN

[MOBILE COMPUTING]

SEMESTER I 2023/2024

TABLE OF CONTENT

SUMMARY.....	3
USE CASE.....	4
FLUTTER WIDGET.....	5-6
TREE WIDGET.....	7
INTERFACE.....	8-13
CONCLUSION.....	14
REFERENCE.....	15

SUMMARY:

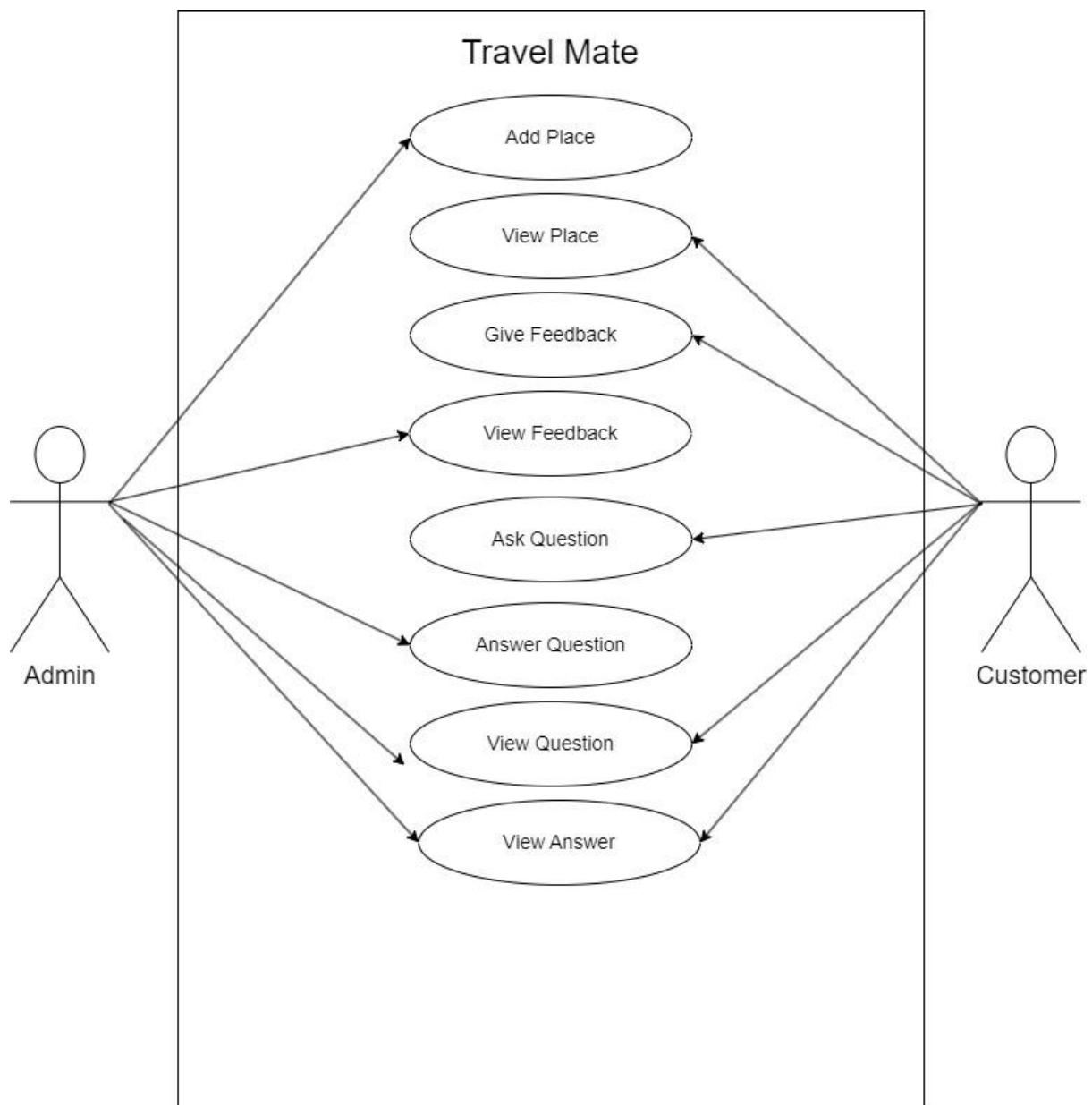
"TravelMate" is a multipurpose Flutter project that aims to provide a seamless travel experience for both customers and administrators. The platform is designed for two types of users: customers and administrators, each with their own set of features.

"TravelMate" is a multipurpose Flutter project designed to deliver a seamless travel experience for both customers and administrators. The platform is intended for two categories of users: customers and administrators, each with their own set of capabilities.

On the administrative side, TravelMate provides administrators with tools to continuously improve the platform. Admins can add new locations, increasing the number of alternatives available to clients. They can also monitor and manage input, creating a more responsive environment. Furthermore, administrators can connect directly with customers by answering their questions, ensuring effective communication and client satisfaction.

In summary, TravelMate aims to build a holistic travel ecosystem in which customers may explore, rate, and communicate, while administrators have the tools they need to improve and manage the platform for the best user experience.

Use Case



List of Widgets and properties

Widgets:

Scaffold: Represents the basic structure of your app's visual interface.

AppBar: The top app bar with the title, menu icon, and optional actions.

Drawer: A slide-in menu that provides navigation options.

ListView.builder: Dynamically builds a scrollable list of items based on an iterable.

Container: A box model that can contain other widgets and allows for styling.

TextField: Allows the user to input and edit text data.

GestureDetector: Wraps a widget and responds to gestures.

CircleAvatar: A circular avatar that can display an image or icon.

Image: Displays an image from various sources.

ElevatedButton: A Material Design raised button.

Card: A material design card with rounded corners.

ListTile: Represents a single fixed-height row.

Icon: Displays icons.

TextFormField: Allows users to input and edit questions or text data.

Visibility: Controls the visibility of a widget based on user interaction.

SingleChildScrollView: Enables scrolling for the entire content.

IconButton: A button that contains an icon.

InkWell: A rectangular area of a Material that responds to touch.

SnackBar: A temporary notification typically shown near the bottom of the screen.

Properties:

placeId: A unique identifier for a place.

placeName: The name of the place associated with feedback or information.

description: A description of the place.

imagePath: The path to the image associated with the place.

special: Special features or characteristics of the place.

mostVisitedPlaces: Information about the most visited places.

funFact: A fun fact about the place.

_isLit: A property representing the state of whether the light is turned on or off.

_feedbackRating: A property representing the user's feedback rating.

Function and Method:

loadFeedbackData(): Fetches feedback data from a remote server using HTTP requests.

_deleteFeedback(int index): Deletes a feedback entry at the specified index and updates the UI.

logout(BuildContext context): Logs out the user, clears user data, and navigates to the login page.

clearUserData(): Clears user-specific data from the Firebase Realtime Database.

loadQuesData(): Fetches question data from the Firebase database using an HTTP GET request.

loadAnswerData(): Fetches answer data from the Firebase database using an HTTP GET request.

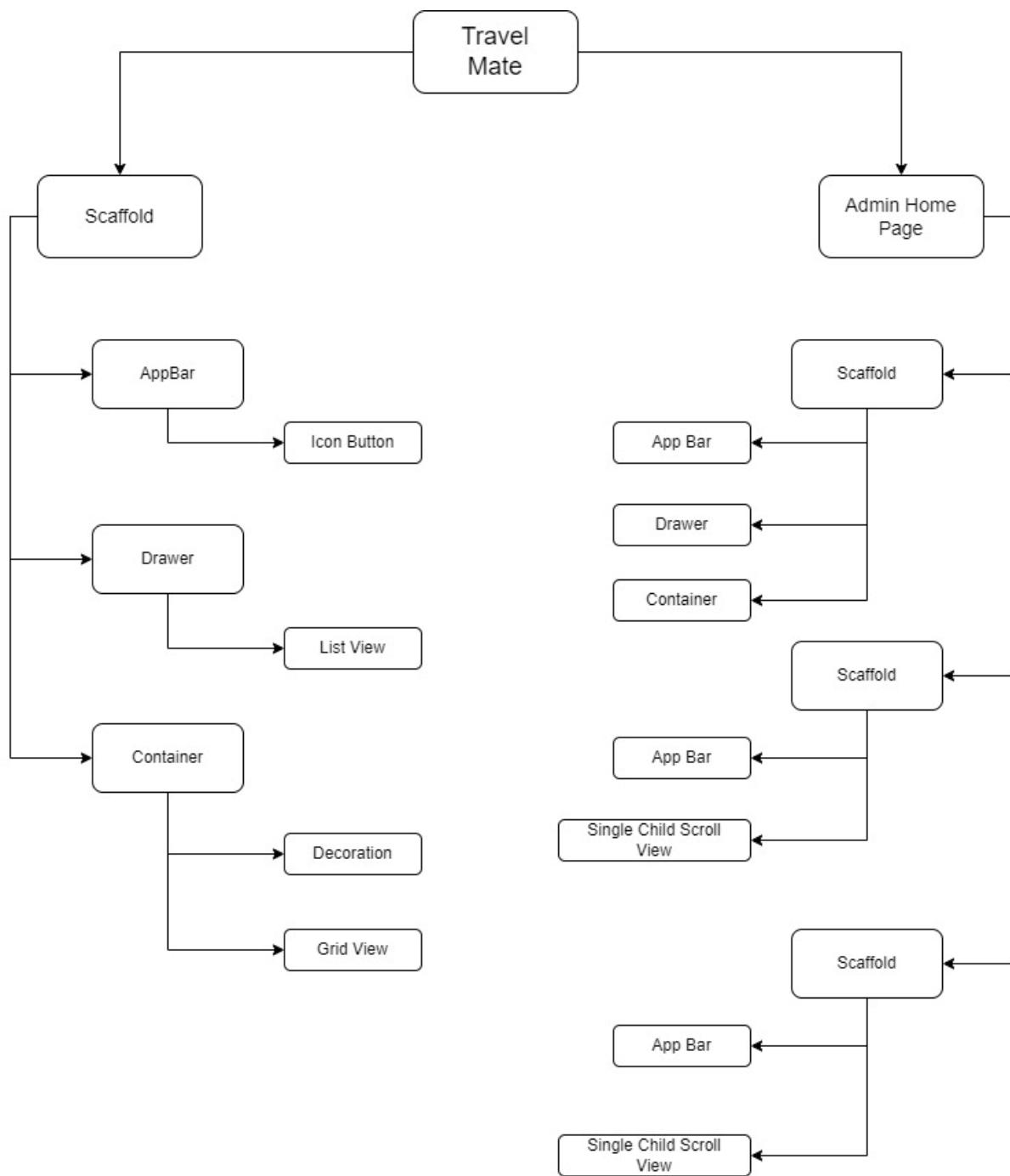
toMap(): Converts an instance of FeedbackData into a map.

_saveFeedbackToDatabase(): Asynchronously saves feedback data to the Firebase Realtime Database.

_setFeedback(): Updates the feedback rating when the user interacts with the star rating.

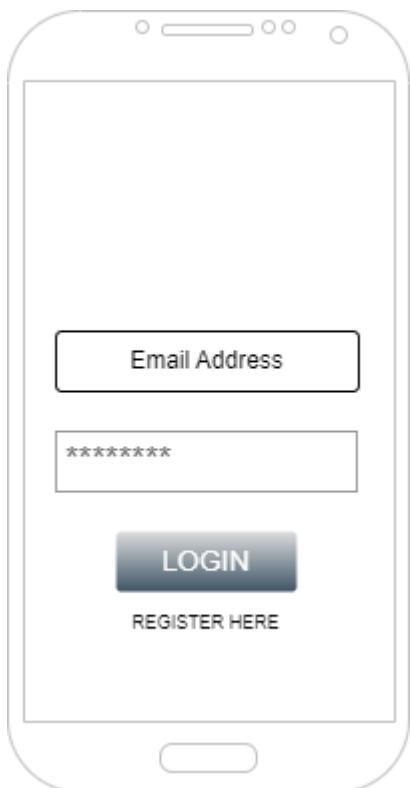
_toggleLight(): Toggles the light state and shows a Snackbar with a fun fact.

Widget Tree



Wireframe

Admin



Admin can login here by inserting their details such as, email address and password. If admin still don't have account they can register first.



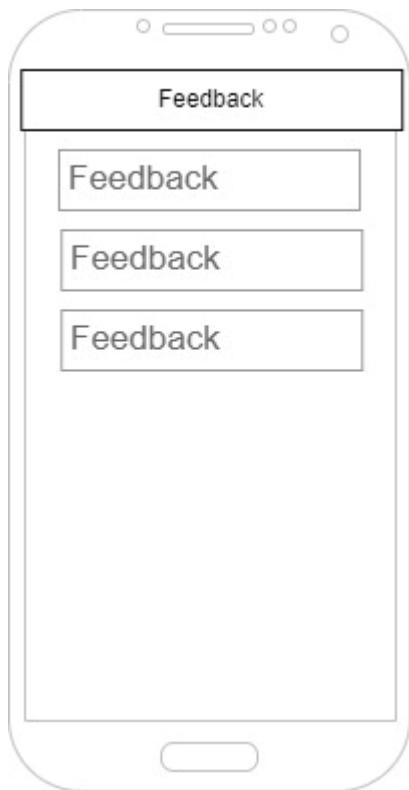
Here is the registration page where admin can register their details before continuing to login. They can register username, email address, password and choose their role whether admin or customer. After sucessfully, register they will direct to login page



This is the homepage for admin. Here, they can add details of new place by pressing the button in the bottom right corner of the screen. They can also view location or edit and delete the location that they already have.



This is the form for add a new place. Here admin can update a details of the place and save it. Upon save, it will direct to admin home page again.

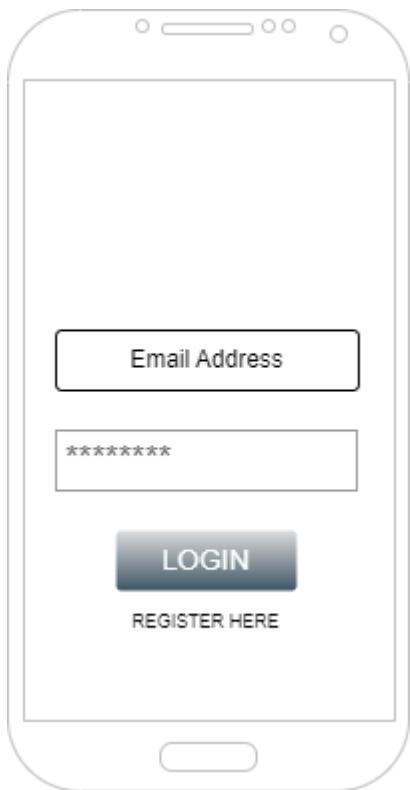


This is the feedback page. Admin can see feedback for each of place that they already add.

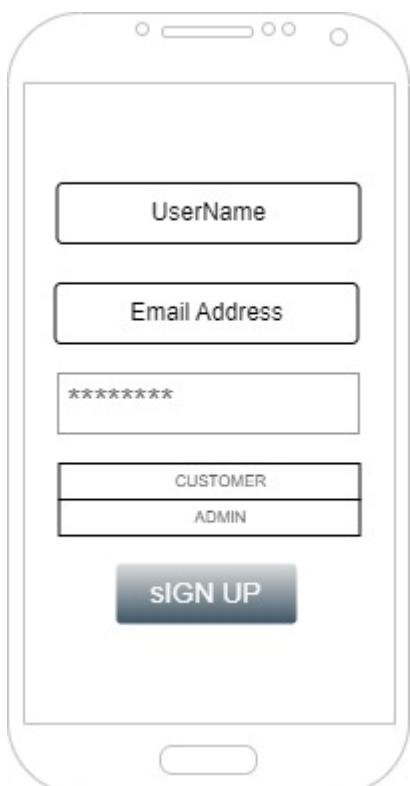


For, QnA page Admin can answer questions from customers. Admin can also view the question and answer that they already answers. They also can edit update or delete the answer.

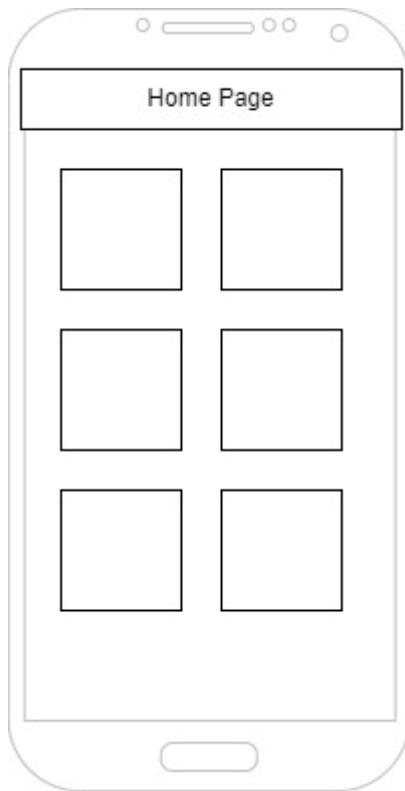
Customer



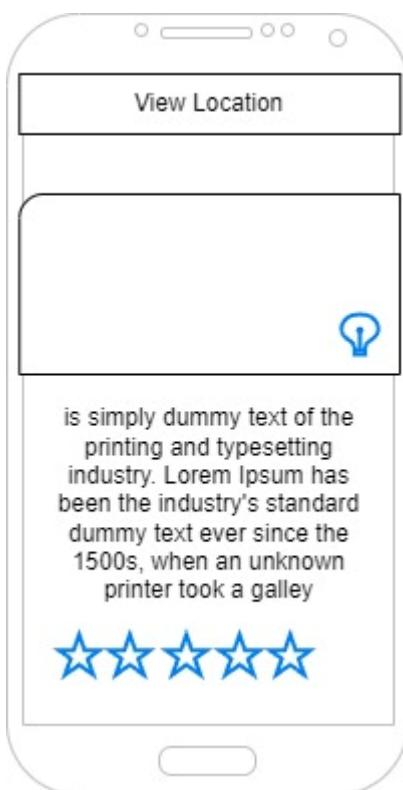
Customer can login here after successfully registering in the sign up page. Customer can insert their email address and password and hit the login button to login.



If customer did not have an account just yet they can go to registration page first to create an account. After that they can login. Customers need to insert their details like username, email address, password and the most important things their role which is customer.



After that they will redirect to Home page. Here they can see a list of places that admin already add.



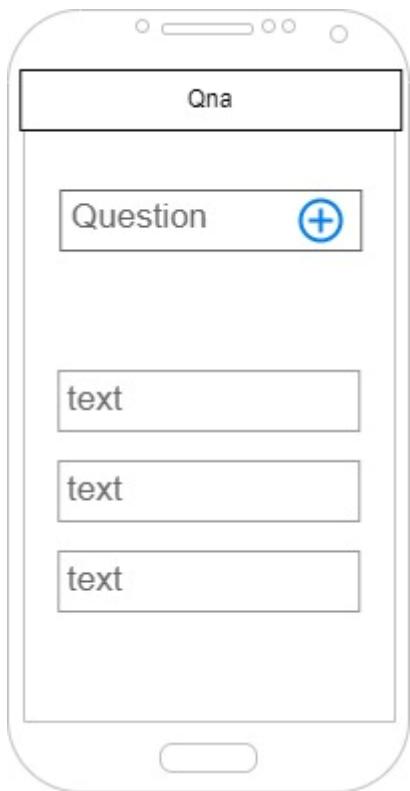
They can view the place and have the see the details that admin already add. They can also give rating to the place as a feedback to admin.

is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley





At profile page they can add another details like their name, phone no, picture and many more.



At Qna page they can submit question about what they want to know more about the place. And they can view the question that admin already submitted.

Conclusion

In conclusion, "TravelMate" emerges as a comprehensive and user-centric Flutter project aimed at improving the travel experience for both customers and administrators. By seamlessly combining functionalities specific to each user role, the platform excels at offering a complete travel environment.

Customers enjoy an immersive journey with TravelMate's easy design, which allows them to easily explore, rate, and engage with places. The opportunity to ask administrators questions and receive timely responses improves the user experience by creating a sense of community and assistance. Personalisation options, such as profile management, help to a more personalised and pleasurable experience with the platform.

On the administrative side, TravelMate provides administrators with extensive tools to further improve the platform. The introduction of new locations and the handling of client feedback result in a dynamic and responsive ecosystem. Direct connection with clients by answering questions demonstrates a dedication to effective communication and customer happiness.

Overall, TravelMate demonstrates Flutter's ability to create diverse applications. It not only promotes travel exploration but also builds a sense of community and responsiveness, making it an invaluable resource for both travellers and administrators. With its smart design and feature-rich environment, TravelMate is ready to change the way users interact with travel-related content, delivering a seamless and engaging experience for everybody.

Reference

1. Afgprogrammer. (2019, October 7). *Flutter UI Tutorial / Login Page UI Design and Animation - day 14* [Video]. YouTube.
<https://www.youtube.com/watch?v=txvyAO894DY>
2. Dear Programmer. (2022, October 27). *Travel app UI design in Flutter - Flutter Traveling App UI design - Speed Code* [Video]. YouTube.
https://www.youtube.com/watch?v=YlgzSH_DPD0
3. Dear Programmer. (2023, February 22). *Travel Mobile App UI design in Flutter - Flutter UI design tutorial* [Video]. YouTube.
<https://www.youtube.com/watch?v=3ipHn2tcbqE>
4. Net Ninja. (2019, October 21). *Flutter & Firebase App Tutorial #1 - Introduction* [Video]. YouTube. <https://www.youtube.com/watch?v=sfA3NWDBPZ4>
5. Mitch Koko. (2022, April 21).  *Firebase Setup for IOS • Flutter Backend Tutorial* [Video]. YouTube. <https://www.youtube.com/watch?v=hy0NtR0NW4Q>
6. *ChatGPT*. (n.d.). <https://chat.openai.com/>
7. *image_picker / Flutter Package*. (n.d.). Dart Packages.
https://pub.dev/packages/image_picker
8. *path_provider / Flutter Package*. (n.d.). Dart Packages.
https://pub.dev/packages/path_provider

Link Github

<https://github.com/DanialSolehin/PROJECT2-FRAMEWORK.git>