UNIVERSITY MALAYSIA TERENGGANU

FACULTY OF OCEAN ENGINEERING TECHNOLOGY & INFORMATICS

FRAMEWORK-BASED MOBILE

APPLICATION DEVELOPMENT

CSM3114

Project 2

Title of the Project

Haulier Tracking App

Prepared by:

Muaz Bin Zainal (S62250)

Prepared for:

Dr Mohamad Nor Hassan

[Mobile Computing]

SEMESTER 5 2023/2024

TABLE OF CONTENT

Executive Summary of the Prototype

The Haulier Tracking App designed to provide vehicle movement and logistics management for admin and their drivers. The application helps to address the challenges of inefficient route planning, limited communication, and inaccurate delivery timing experienced within the haulage industry.

Key Features:
- Develop a user-friendly mobile application for admin and drivers to view truck schedule, utilization and movement.
- Enhance communication and efficiency through integrated messaging and route optimization features.
- Improve delivery accuracy and client satisfaction by providing estimated arrival times and automated notifications.

Key Findings:
- User testing revealed positive feedback on the app's intuitiveness and ease of use.
- Truck movement tracking significantly improved delivery timeliness and operational transparency.
- Integrated communication features streamlined the logistics process and reduced delays.

Overall Impact:
User testing The Haulier Tracking App has demonstrably improved operational efficiency, communication, and customer satisfaction within the haulage industry. Its continued development and expansion hold significant potential for further streamlining logistics processes and optimizing fleet performance.
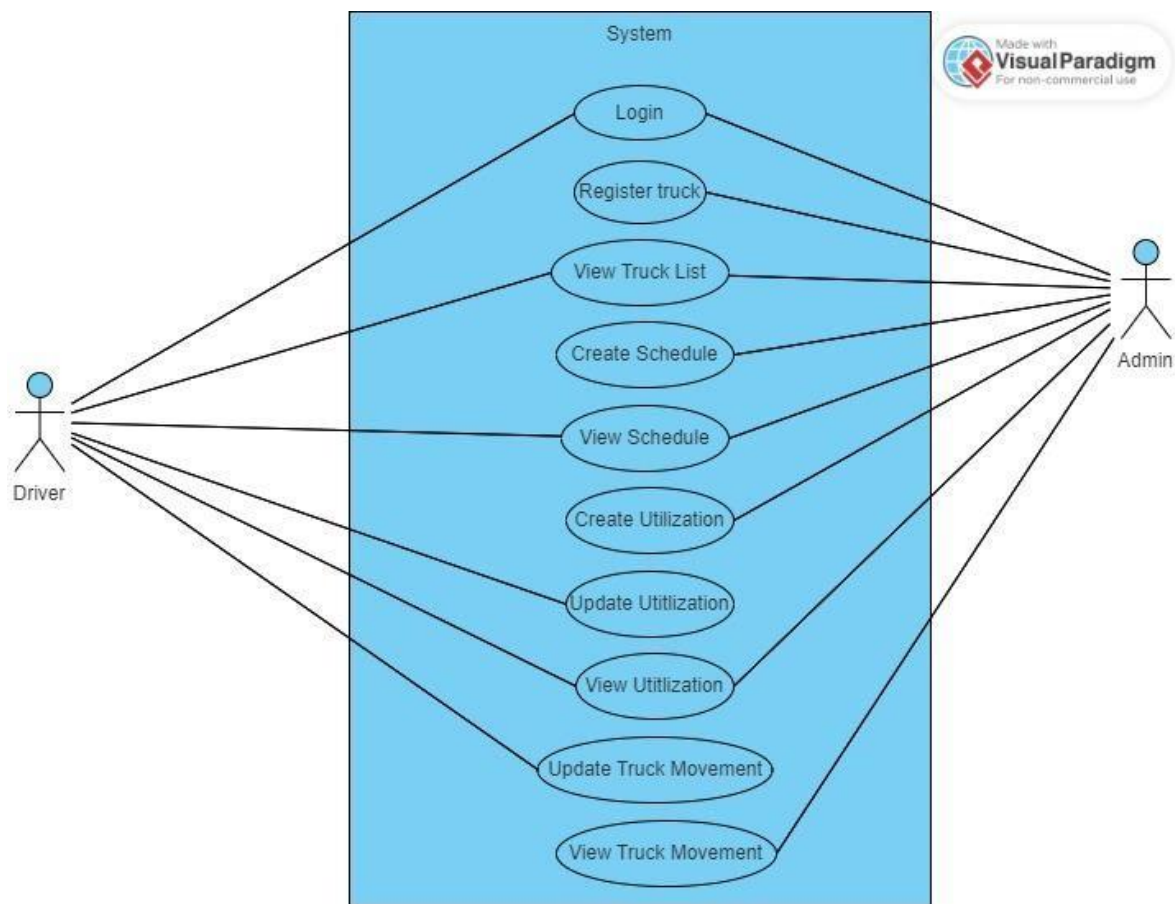
Use Case

Figure 1

Structure of tree widget

Admin tree widget
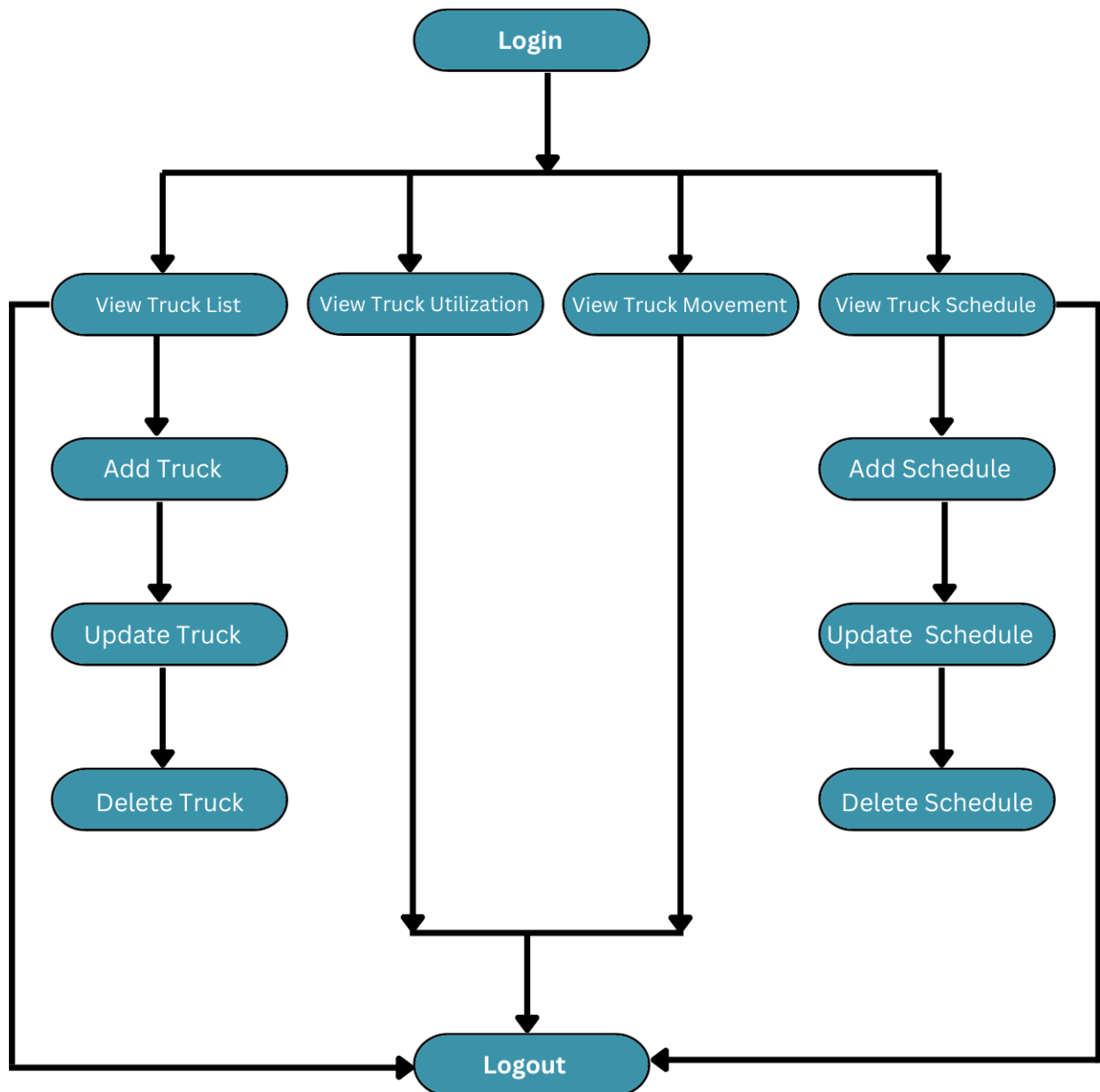

Figure 2.0

Driver tree widget
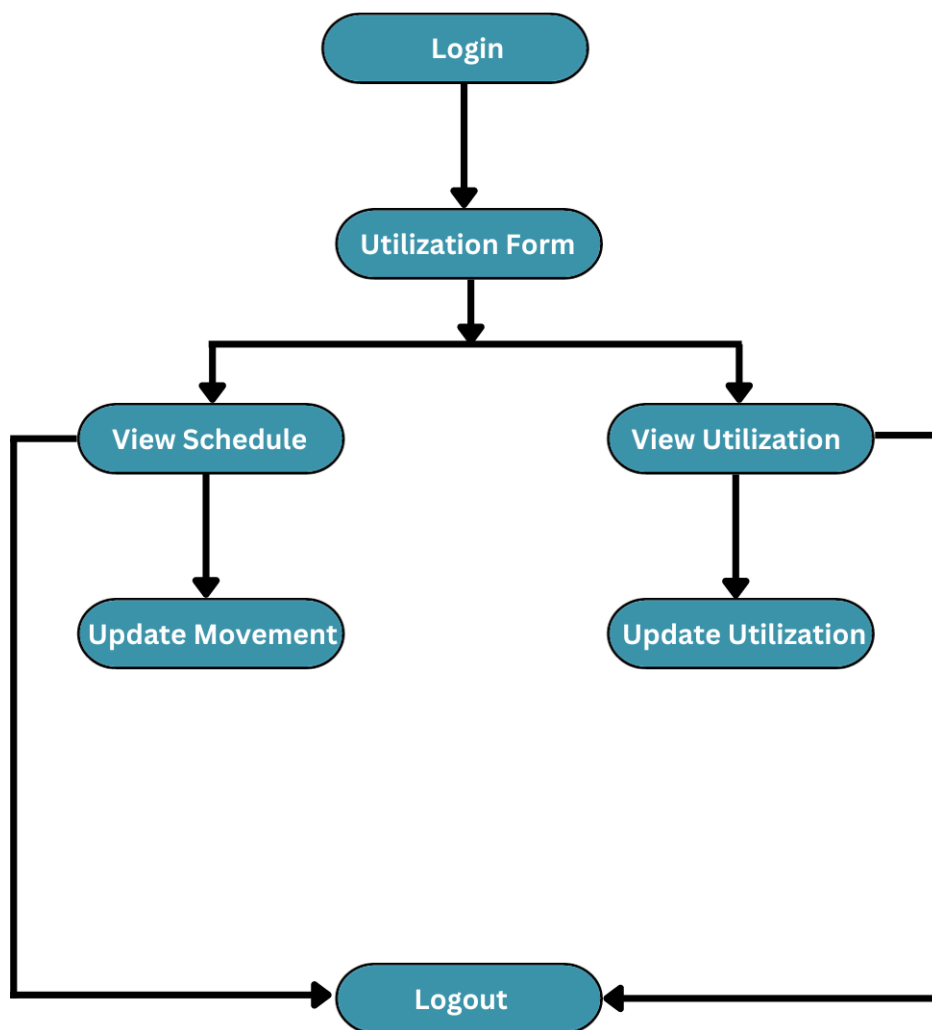

Figure 2.2

Flutter Widgets and Features:

Flutter Widgets

Login:

- **TextFormField:** Collects user credentials (username, password).
- **ElevatedButton:** Initiates login process.

CRUD for Firebase:

- **FutureBuilder**: Likely used to display the schedule data based on its loading state.
- **TextFields**: Edit data for updates.
- **ElevatedButtons**: Trigger create, read, update, and delete operations.

Validation for TextFormField:

- Enforces input rules (e.g., required fields, email format).

Navigation:

- Manages screen transitions and app flow.

BottomNavigationBar:

- Provides persistent access to primary app sections.

Drawer:

- Offers hidden navigation menu for additional options.

DropdownButton:

- Presents a list of options for user selection.

Card:

- Frames content visually, often for displaying tracking details.

ListView:

- Renders scrollable lists of items, such as tracking events or hauliers.

DateFormat:

- Formats dates consistently for display (e.g., "YYYY-MM-DD").

TimeFormat:

- Formats times consistently for display (e.g., "HH:MM:SS").

Icons:

- Visually represent actions or information (e.g., truck icons, location markers).

AssetImage:

- Displaying local images stored in app's assets folder.

RadioButton:

- Enables users to select a single option from a group of choices, potentially used for status updates or preferences.

Flutter Features:

Login: Secure user authentication for accessing app features.

CRUD for Firebase:

- **Create:** Add new truck records or schedule.

- **Read:** Retrieve tracking details truck information.

- **Update:** Modify existing truck data.

- **Delete:** Remove unnecessary records.

Validation for TextFormField: Ensures data integrity and prevents errors.

Navigation: Smooth screen transitions and app flow management.

BottomNavigationBar: Easy access to core app sections.

Drawer: Additional navigation options without cluttering the main UI.

DropdownButton: User-friendly selection from predefined choices.

Card: Visually appealing presentation of key information.

ListView: Efficient display of dynamic content lists.

DateFormat/TimeFormat: Consistent and readable date and time presentation.

Icons: Enhanced visual communication and user guidance.
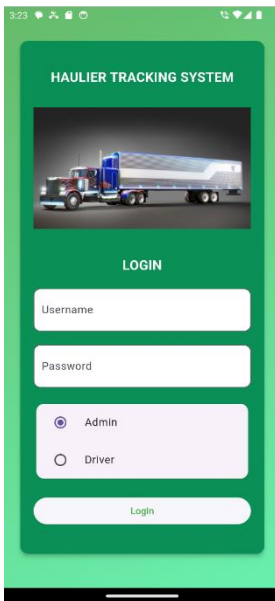
AssetImage: Load of images from internal sources.
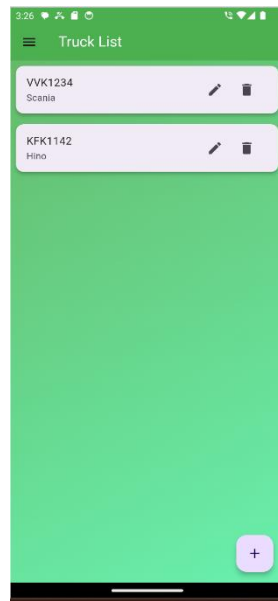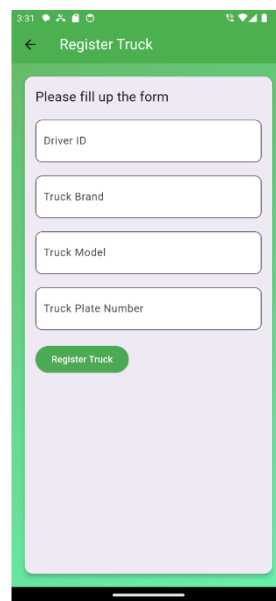
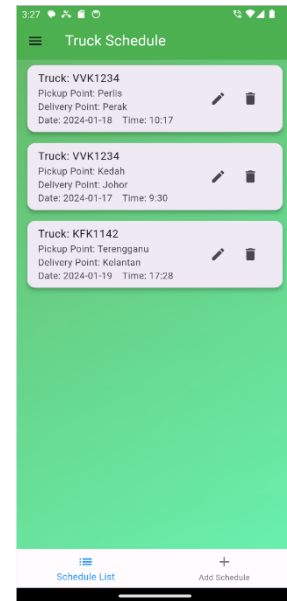**Sample of interface**

Figure 3.0
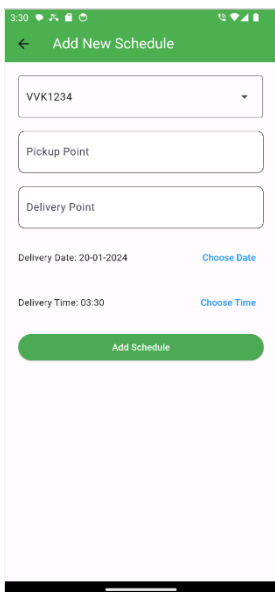

Figure 3.1

Figure 3.2
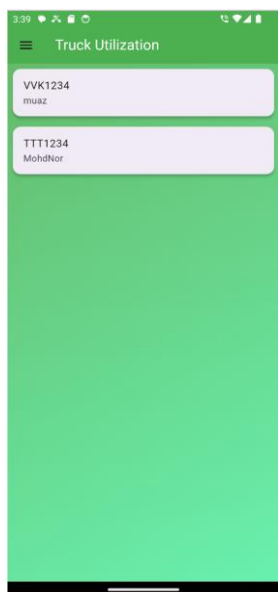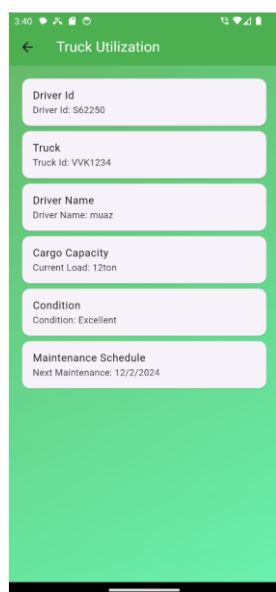

Figure 3.3

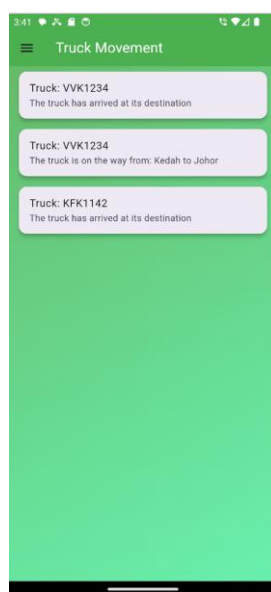

Figure 3.4

Figure 3.5

Figure 3.6

Figure 3.7

UI Explaination:

Figure 3.0: Login page where user need to select admin or driver

Figure 3.1: List of all truck will display
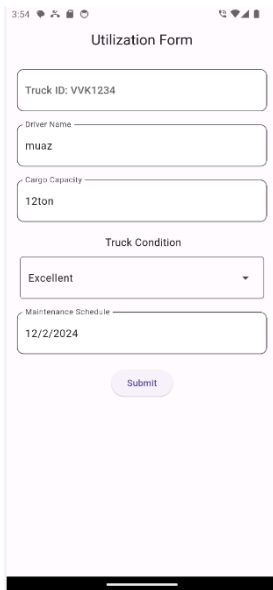
Figure 3.2: Form to register a new truck

Figure 3.3: List of all schedule will display

Figure 3.4: Form to add new schedule

Figure 3.5: List of all truck utilization

Figure 3.6: Truck utilization detail

Figure 3.7: Drawer for admin user
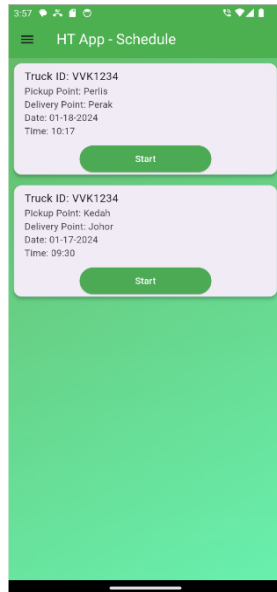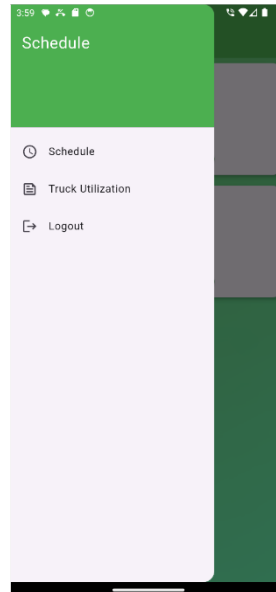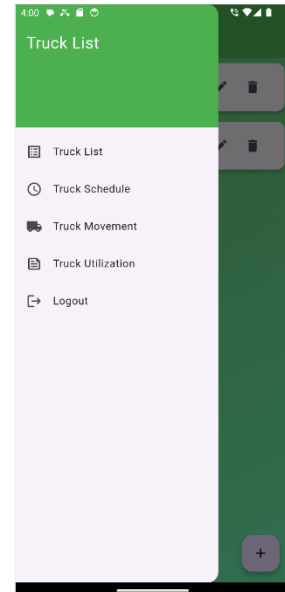
| Figure 3.8 | Figure 3.9 | Figure 3.10 | Figure 3.11 |

UI Explaination:

Figure 3.8: Utilization form to update the truck utilization

Figure 3.9: Schedule with start button to update truck movement

Figure 3.10: Drawer for driver

Figure 3.11: Drawer for admin

Conclusion

In conclusion, the development and implementation of the Haulier Tracking App, powered by Flutter widgets and features, have proven to be a pivotal solution in addressing the longstanding challenges faced by the haulage industry. The app's user-friendly interface, coupled with its robust set of features, has resulted in significant improvements in operational efficiency, communication, and overall customer satisfaction.

Finally, the Haulier Tracking App stands as a testament to the effectiveness of modern mobile technology in revolutionizing traditional industries. Its successful implementation and positive outcomes position it as a valuable tool for fleet operators and clients alike, paving the way for increased efficiency and satisfaction within the haulage sector.

References

1. "Cookbook." Flutter, docs.flutter.dev/cookbook. Accessed 5 Jan. 2024.

2. "Display Images from the Internet." Flutter, docs.flutter.dev/cookbook/images/network-image. Accessed 6 Jan. 2024.

3. "Listtile Class." ListTile Class - Material Library - Dart API, api.flutter.dev/flutter/material/ListTile-class.html. Accessed 6 Jan. 2024.

4. Yuto. "Flutter Convert TimeOfDay to 24 Hours Format." Technical Feeder, Technical Feeder, 28 Oct. 2022, www.technicalfeeder.com/2022/02/flutter-convert-timeofday-to-24-hours-format/.

5. Goodman, A. "Working with Time Picker in Flutter (Updated)." KindaCode, 2023, www.kindacode.com/article/working-with-time-picker-in-flutter/.

6. Hitanshu GogoiHitanshu Gogoi 96733 gold badges1515 silver badges2424 bronze badges, et al. "Bottom Overflow by 30px." Stack Overflow, 1 Sept. 1964, stackoverflow.com/questions/51972371/bottom-overflow-by-30px.

7. Add Firebase to your Flutter app. (n.d.). Firebase. https://firebase.google.com/docs/flutter/setup

8. U, S. A. (2023, February 28). HTTP CRUD methods in Flutter - Santhosh Adiga U - Medium. Medium. https://santhosh-adiga-u.medium.com/http-crud-methods-in-flutter-f679ecbd0671