



UNIVERSITI MALAYSIA TERENGGANU
FACULTY OF OCEAN ENGINEERING TECHNOLOGY &
INFORMATICS

CSM3114
FRAMEWORK-BASED MOBILE APPLICATION DEVELOPMENT

INDIVIDUAL PROJECT 2

PREPARED BY:
MUHAMMAD HISYAMUDDIN BIN SAZALI (S63651)

PREPARED FOR:
DR. MOHAMAD NOR BIN HASSAN

COMPUTER SCIENCE (MOBILE COMPUTING) WITH HONOURS
SEMESTER I 2023/2024

Contents

1. EXECUTIVE SUMMARY	3
2. USE CASE.....	4
3. COMMON STRUCTURE OF WIDGET TREE	5
4. FLUTTER WIDGET AND FEATURES ADOPTED IN THE APPLICATION	6
Widgets:	6
Features:	7
5. SAMPLE OF INTERFACE WITH EXPLANATION	8
6. CONCLUSION	11
7. REFERENCES.....	12

Figure 1 Use Case	4
Figure 2 Widget Tree.....	5
Figure 3 Login Page	8
Figure 4 Booking Manager	9
Figure 5 Booking Confirmation	9
Figure 6 Booked Room	10
Figure 7 Administrator : Add Room.....	10

1. EXECUTIVE SUMMARY

In the rapidly evolving hospitality industry, effective management is essential for success. The Hotel Management App serves as a comprehensive solution to streamline and elevate hotel operations, offering a seamless experience for both staff and guests.

Key features of the app include a secure and user-friendly login system with role-based access, ensuring authorized personnel have appropriate permissions. The intuitive room management feature allows administrators to effortlessly update and synchronize room information in real-time, minimizing the risk of overbooking. The booking process is made seamless for guests through an intuitive interface, with instant confirmation notifications providing assurance. A centralized dashboard empowers hotel staff to efficiently manage and monitor all bookings, with filters and sorting options facilitating easy identification of reservations and room availability. Detailed booking information, including guest details and special requests, allows for a personalized guest experience.

The app's benefits extend to time efficiency, enabling staff to focus on delivering exceptional service. Real-time updates and synchronization enhance accuracy in room availability and booking information. The app contributes to an enhanced guest experience by facilitating a smooth and hassle-free booking process. Additionally, its data-driven approach empowers hotel management with valuable insights into booking trends for informed decision-making.

In summary, the Hotel Management App is a powerful tool designed to enhance operational efficiency, elevate guest satisfaction, and keep hotels competitive in the dynamic hospitality landscape. Tailored to meet the diverse needs of modern hoteliers, whether managing a boutique hotel or a large resort, this app is a key asset for successful hotel operations.

2. USE CASE

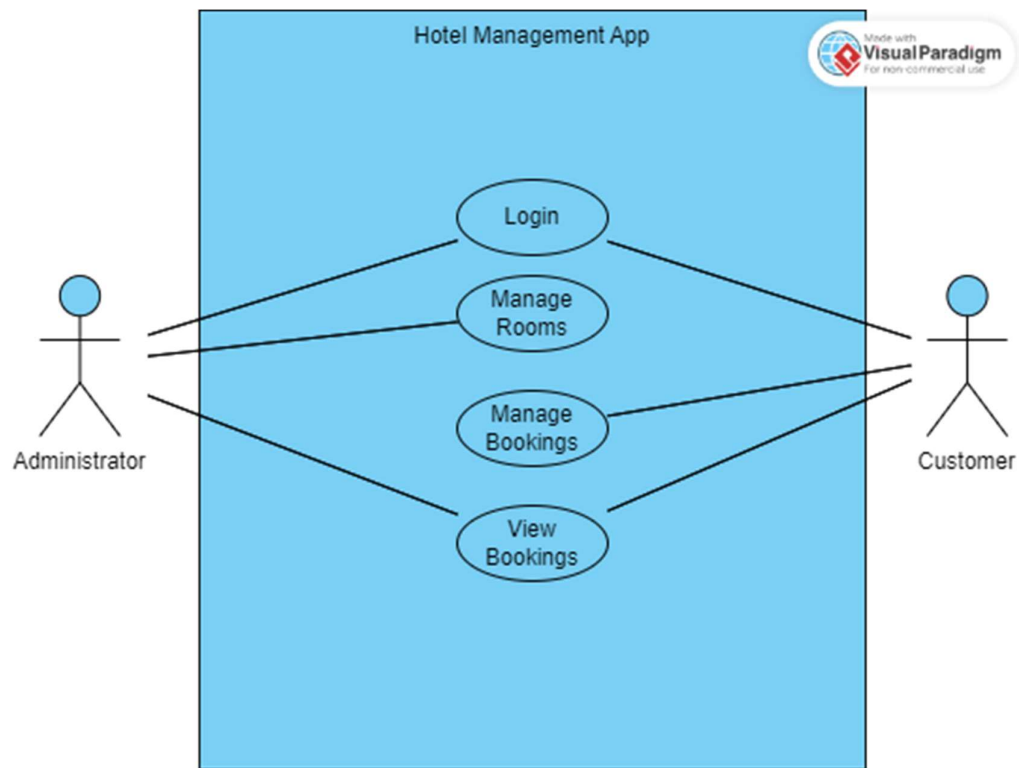


Figure 1 Use Case

3. COMMON STRUCTURE OF WIDGET TREE

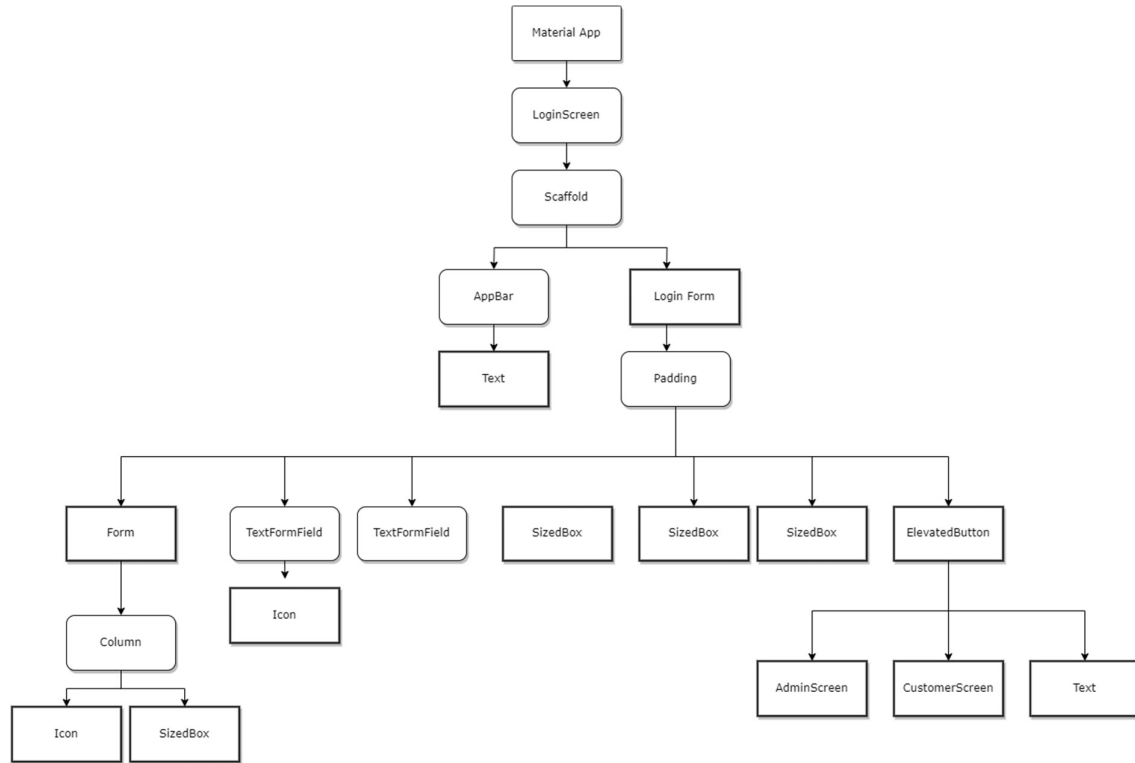


Figure 2 Widget Tree

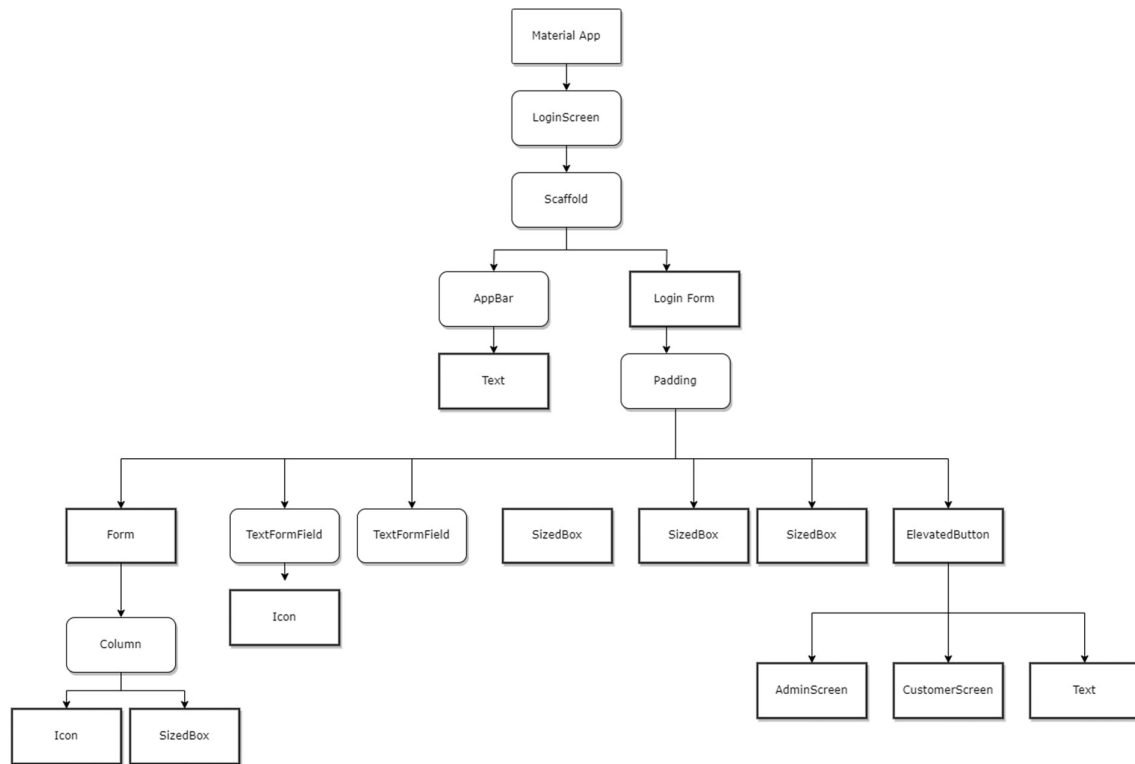
4. FLUTTER WIDGET AND FEATURES ADOPTED IN THE APPLICATION

Widgets:

1. Scaffold: Acts as the primary container, organizing the structure of each screen with components like app bars, bodies, and bottom navigation.
2. AppBar: Utilized on each screen to serve as a consistent navigation header, displaying screen titles and potential actions or icons like the logout button.
3. TextFormField: Employed for gathering user input in the login form and commonly used for form-related tasks, offering features like label text, validation, and secure text entry.
4. ElevatedButton: Used across the app for actions such as logging in, form submissions, and other critical functions. The button's appearance is customized for a more visually appealing look.
5. Icon: Implemented to showcase icons, enhancing the visual interface and ensuring a recognizable and intuitive user experience.
6. ListView.builder: Dynamically generates lists for rooms and booked rooms, efficiently handling large datasets by loading items as they become visible.
7. Card: Creates material design cards with elevation, providing visually appealing containers for room and booked room information.
8. ModalBottomSheet: Displayed as a bottom sheet for adding and editing rooms, serving as a temporary modal surface for additional functionality.
9. Column and Row: Used for organizing widgets vertically and horizontally, respectively, contributing to a clean and organized user interface.
10. SizedBox: Introduces spacing between widgets to enhance the overall layout and visual presentation.
11. Form and TextFormField: Applied for capturing and validating user input, with the Form widget managing the form state and TextFormField offering customizable text input fields.
12. PageRouteBuilder: Employed for navigating between screens with custom transition animations, providing more control over the navigation experience.
13. Container: Utilized for styling purposes and to add padding around specific elements, contributing to an improved visual appeal.

Features:

1. State Management (StatefulWidget and setState): Utilized StatefulWidget and setState to dynamically manage and update the app's state.
2. Navigation and Routing: Employed Navigator and MaterialPageRoute for seamless navigation between screens, ensuring a smooth flow within the app.
3. Theming (ThemeData): Incorporated basic theming using ThemeData to maintain a consistent look and feel throughout the app.
4. Dynamic Data Handling: Managed dynamic data for rooms and booked rooms, providing flexibility and scalability.
5. Bottom Sheet Usage: Utilized modal bottom sheets for adding and editing rooms, offering a smooth and context-aware user interaction.
6. List Building: Implemented dynamic lists for displaying rooms and booked rooms, ensuring scalability for handling various data.
7. Form Validation: Implemented form validation using the Form widget, ensuring user inputs meet specific criteria.
8. Text Editing Controllers: Used TextEditingController for managing text input and controlling the content of text fields.
9. Dialogs: Employed dialogs for booking confirmation, providing user feedback on the success of their actions.
10. AppBar Actions: Added actions like logout in the app bar, making essential functionalities directly accessible to the user.



5. SAMPLE OF INTERFACE WITH EXPLANATION

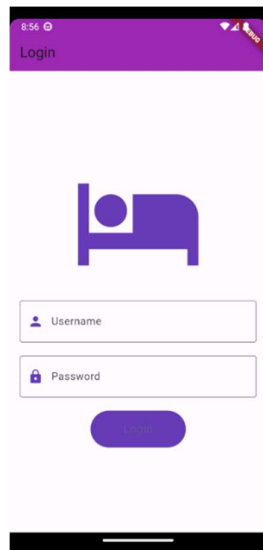


Figure 3 Login Page

Figure 3 shows the Login Screen where both admin and customer will login. The admin id already in the system and it will redirect to the administrator account. If else, it will redirect to the customer account.

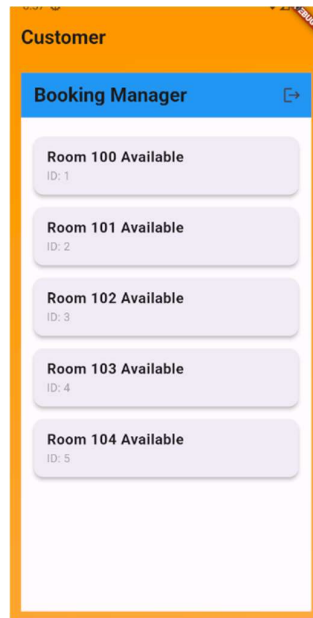


Figure 4 Booking Manager

Figure 4 shows the Booking Manager for the customer to choose their Room available for the hotel. When clicked, customers can book the room. The right top side button or icon will logout of the account and redirect to the Login Screen.

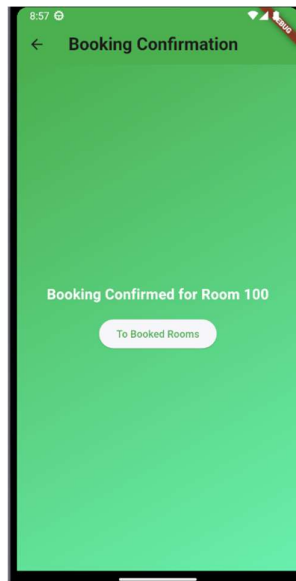


Figure 5 Booking Confirmation

Figure 5 refers to the Booking Confirmation Page stating that the customer already manage to book the room.

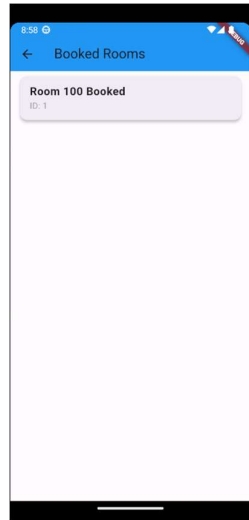


Figure 6 Booked Room

Figure 6 refers to the Booked Room Page to view booking made by the customer.

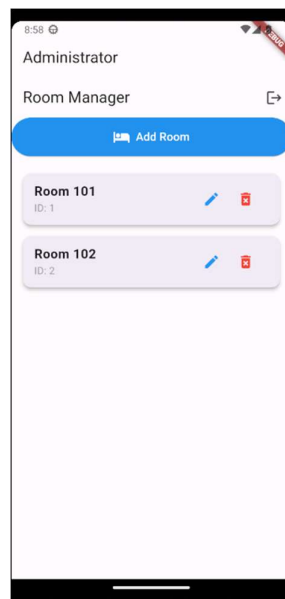


Figure 7 Administrator : Add Room

Figure 7 shows the Administrator module where he can create, update and delete rooms available at the hotel for the moment.

6. CONCLUSION

In conclusion, the Hotel Management App serves as a foundational framework for a comprehensive system catering to both administrators and customers. The app employs a clear Object-Oriented Programming (OOP) structure, ensuring modularity and maintainability. The functionalities are categorized into two main modules: Administrator and Customer, each offering distinct features to streamline hotel operations.

For administrators, the app provides a secure login mechanism, offering access to the Room Manager module. This module facilitates the management of rooms, allowing administrators to perform essential CRUD operations such as adding, updating, and deleting rooms. The Room Manager interface is designed with an intuitive user experience, featuring functionalities like room details display, editing, and deletion through interactive buttons.

On the customer side, the app caters to the user journey from login to room booking. The Booking Manager module is a placeholder for the comprehensive functionality required for customer reservations. It acknowledges the critical steps of booking a room, confirming the reservation, and viewing existing bookings. Although this module currently stands as a basic structure, it provides a foundation that can be extended to incorporate a robust booking system with features such as date selection, payment integration, and reservation status tracking.

7. REFERENCES

1. Course: FRAMEWORK-BASED MOBILE APPLICATION DEVELOPMENT (2023/2024).
umt.edu.my
In Youtube. <https://www.youtube.com/watch?v=622vOT8A1J0>
2. material library – Dart API (2023). Flutter.dev
<https://api.flutter.dev/>
3. Flutter Mapp. (2022) Flutter ExpansionTile Widget – Expand Tile [Youtube Video]
In Youtube. <https://www.youtube.com/watch?v=i34zFGTofCE>
4. Github link: <https://github.com/HisyamuddinS63651/S63651-PROJECT2>
5. Open AI ChatGPT (2022).