VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi - 590018

Mini Project Report

on

"BOOKMYPOOL APPLICATION"

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF ENGINEERING

in

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

By

SALWA IMTHIYAZ AHAMED 4MT20AI037

Under the Guidance of
Mr. Sunil Kumar S
Senior Assistant Professor
& HOD AIML

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

MANGALORE INSTITUTE OF TECHNOLOGY & ENGINEERING

Accredited by NAAC with A+ Grade, An ISO 9001: 2015 Certified Institution
(*A Unit of Rajalaxmi Education Trust®, Mangalore - 575001*)
Affiliated to VTU, Belagavi, Approved by AICTE, New Delhi
Badaga Mijar, Moodabidri-574225, Karnataka

2022-23

MANGALORE INSTITUTE OF TECHNOLOGY & ENGINEERING
Accredited by NAAC with A+ Grade, An ISO 9001: 2015 Certified Institution
(*A Unit of Rajalaxmi Education Trust®, Mangalore - 575001*)
Affiliated to VTU, Belagavi, Approved by AICTE, New Delhi
Badaga Mijar, Moodabidri-574225, Karnataka

Department of Artificial Intelligence and Machine Learning

CERTIFICATE

This is to certify that the mini project entitled "**BookMyPool Application**" is a bonafide work carried out by Ms. **SALWA IMTHIYAZ AHAMED (4MT20AI037)** in partial fulfillment for the requirement of 6th semester MAD Laboratory with mini project (18AIL68). It is certified that all the corrections/suggestions indicated for the Internal Assessment have been incorporated in the report.The mini project has been approved as it satisfies the academic requirement in respect of the 18AIL68 prescribedfor the 6th Semester B.E in Artificial Intelligence & Machine Learning Program by the **Visvesvaraya Technological University, Belagavi**, for the academic year 2022 – 2023.

…… ……..

Signature of the Guide
Mr. Sunil Kumar S

Signature of the HOD
Mr. Sunil Kumar S

Name of the Examiners	**Signature with Date**
1. ……	**……**
2. ……	**……**

ACKNOWLEDGEMENT

The successful completion of any significant task is the outcome of invaluable aggregate combination of different people in radial direction explicitly and implicitly. We would therefore take opportunity to thank and express our gratitude to all those without whom the completion of project would not be possible.

We express our thanks to **Mr. Sunil Kumar S, Senior Assistant Professor, Head of the Department of Artificial Intelligence & Machine Learning** for having provided all the facilities that helped us and also for hir support and guidance in timely completion of this report.

We would like to thank **Dr. Prashanth C M**, **Principal, Mangalore Institute of Technology and Engineering, Moodabidri** for his s support and encouragement.

I express my sincere gratitude to our institution and management for providing us with good infrastructure, laboratory facilities, qualified and inspiring staffs, and whose guidance was of immense help in completion of this seminar successfully.

SALWA IMTHIYAZ AHAMED

4MT20AI037

ABSTRACT

BookMyPool is a revolutionary Android application that revolutionizes the concept of carpooling, offering a simplified and convenient platform for individuals to share rides and reduce their carbon footprint. By connecting users who are traveling in the same direction, the application aims to streamline the carpooling process and make it more accessible to a wider audience.The primary goal of BookMyPool is to address common challenges associated with individual commuting, such as traffic congestion, limited parking spaces, and environmental pollution. By encouraging ride-sharing, the application seeks to optimize the utilization of available transportation resources, decrease the number of vehicles on the road, and contribute to a greener and more sustainable environment.Key features of the BookMyPool application include a user-friendly interface that enables users to quickly create profiles, specify their preferences as riders or passengers, and search for compatible ride-sharing options. The application utilizes advanced algorithms to match users based on their routes and other preferences, ensuring efficient and reliable connections between riders. Moreover, BookMyPool empowers passengers by providing them with multiple ride options. Passengers can compare the costs offered by different drivers and select the ride that best suits their needs. This feature adds flexibility and convenience to the carpooling experience, enhancing user satisfaction and participation.BookMyPool offers a simple and intuitive carpooling solution for Android users, bringing together like-minded individuals who want to share rides, reduce their environmental impact, and enhance their commuting experience. BookMyPool sets out to revolutionize the way people commute by providing a platform that makes carpooling more accessible, efficient, and appealing to a wider audience. By combining technology, convenience, and environmental consciousness, this innovative Android application aims to create a positive impact on transportation systems and contribute to a greener future.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 DEFINITION

The BookMyPool Android application is an innovative solution developed using Android Studio that aims to simplify the process of carpooling, providing users with a convenient platform to share rides and reduce their carbon footprint. This introduction will provide an overview of the project, its objectives, and the key features it offers to users. In today's world, traffic congestion, limited parking spaces, and environmental pollution are pressing challenges faced by commuters. To address these issues, BookMyPool offers a user-friendly and efficient carpooling solution that optimizes transportation resources and promotes sustainable commuting practices.

The primary objective of BookMyPool is to provide a seamless and convenient experience for users seeking to share rides with fellow commuters traveling in the same direction. By connecting individuals with similar routes, the application aims to reduce the number of vehicles on the road and minimize traffic congestion.

1.2 IMPORTANCE

Carpooling apps, like BookMyPool, offer several benefits and hold great importance in today's society which includes the following:

1. **Environmental Impact:** Carpooling reduces the number of vehicles on the road, leading to reduced traffic congestion and lower emissions of greenhouse gases. It contributes to sustainable transportation and helps combat air pollution and climate change.
2. **Cost Savings:** Carpooling allows individuals to share the costs of commuting, including fuel expenses, tolls, and parking fees. It helps individuals save money and reduce their transportation expenses.
3. **Community Building:** Carpooling apps foster social connections and promote community building by connecting individuals with similar travel routes. It encourages collaboration and reduces social isolation during commuting.
4. **Efficient Resource Utilization:** Carpooling optimizes the use of existing resources, such as vehicles and road infrastructure. It maximizes transportation efficiency and minimizes wasted resources.
5. **Reduced Traffic Congestion:** By promoting shared rides, carpooling apps help alleviate traffic congestion, especially during peak hours. This results in smoother traffic flow, shorter travel times, and improved overall transportation efficiency.

1.3 APPLICATIONS

The BookMyPool app has various applications that cater to the needs of different users:

1. **Daily Commuters:** The app allows daily commuters to find carpooling partners who share similar routes and schedules. It enables them to save time and money by sharing rides on a regular basis.
2. **Long-Distance Travel:** BookMyPool can also facilitate long-distance ridesharing for intercity or interstate travel. Users can find travel companions for sharing long journeys, making travel more affordable and enjoyable.
3. **Events and Activities:** The app can be used for carpooling to events, concerts, or other social gatherings. It helps attendees coordinate rides and reduce the number of vehicles parked at the venue.
4. **Airport Transfers:** Travelers can utilize BookMyPool for sharing rides to and from the airport. By connecting with other passengers heading in the same direction, they can split transportation costs and avoid the hassle of parking.
5. **Corporate Carpooling:** BookMyPool can be adopted by organizations to promote carpooling among their employees. It facilitates efficient transportation for staff members, reduces parking requirements, and aligns with corporate sustainability initiatives.

Overall, the BookMyPool app offers a convenient and sustainable solution for shared transportation, benefiting individuals, communities, and the environment.

Chapter 2

REQUIREMENTS

2.1. FUNCTIONAL REQUIREMENTS

Functional requirements describe what a system or software application should do or the specific functionalities it should provide. These requirements define the actions or services that the system must perform to meet the needs of its users. They focus on the system's behavior, features, and interactions with users or other systems. Examples of functional requirements for BookMyPool Application could include:

1. **User Registration:**

   - Users should be able to register a new account by providing a valid email address and password.
   - The registration process should include input validation and display appropriate error messages for invalid inputs.
   - Upon successful registration, the user's account information should be stored securely in Firebase Authentication.

2. **User Login:**

   - Registered users should be able to log in using their email and password.
   - Input validation should be performed, and appropriate error messages should be displayed for invalid login credentials.
   - Upon successful login, users should be granted access to the profile page and their account details should be retrieved from Firebase Authentication.

3. **User Logout:**

   - Logged-in users should have the ability to log out from the profile page.
   - Upon logout, users should be redirected to the login screen and their session should be terminated.

These functional requirements ensure the successful implementation of a Sign up page with SQL Authentication integration, providing users with registration, login, and logout capabilities, as well as displaying their profile information securely.

2.2 NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements specify the criteria that define the system's operation, performance, security, usability, and other qualities. They focus on the overall attributes of the system rather than its specific functionalities. Non-functional requirements help ensure that the system meets certain quality standards and user expectations. Examples of non-functional requirements for BookMyPool Application could include:

1. User Interface: The app should have an intuitive and user-friendly interface that is easy to navigate and visually appealing.

2. Performance: The app should have fast response times, ensuring minimal delays in searching, booking, and matching rides.

3. Security: The app should employ robust security measures to protect user data and ensure secure transactions.

4. Scalability: The app should be able to handle a growing user base and increasing ride requests without compromising performance.

5. Reliability: The app should be reliable and available for use without frequent downtime or disruptions.

6. Compatibility: The app should be compatible with a wide range of Android devices and operating system versions to reach a broader user base.

7. Localization: The app should support multiple languages and regional settings to cater to users from different locations.

8. Accessibility: The app should follow accessibility guidelines, ensuring that users with disabilities can access and use the app effectively.

9. Data Privacy: The app should adhere to data protection regulations and prioritize the privacy of user information.

10. Offline Capability: The app should provide some level of functionality even in offline mode, allowing users to access essential features and information when not connected to the internet.

These functional and non-functional requirements form the foundation for developing the BookMyPool app, ensuring that it meets the needs of users while providing a reliable and user-friendly experience.

2.3 HARDWARE REQUIREMENTS

1. **Android Device**: The app is designed specifically for Android devices, so users will need a compatible smartphone or tablet running on the Android operating system.
2. **Processor**: The device should have a capable processor to handle the app's functionalities smoothly. A processor with at least a dual-core configuration is recommended for optimal performance.

3. **RAM**: The device should have sufficient RAM to ensure smooth multitasking and efficient app performance. A minimum of 2GB RAM is recommended.
4. **Storage Space**: The app itself requires a certain amount of storage space on the device. Additionally, users may need ample space to store downloaded health articles, test results, and other related data.
5. **Internet Connectivity**: HealthCare Pro requires an active internet connection to access real-time information, synchronize data, and enable features like doctor search, online pharmacy, and article updates.

2.4 SOFTWARE REQUIREMENTS

1. **Operating System**: The app is developed for Android devices, so users need to have an Android operating system version compatible with the app. The minimum supported Android version should be specified by the app developer.
2. **Android Development Tools**: Developers working on the app will require Android Studio, the official Integrated Development Environment (IDE) for Android app development. It provides the necessary tools, libraries, and emulators for building, testing, and debugging Android applications.
3. **Programming Language**: The app is developed using Java or Kotlin programming languages, as these are the primary languages for Android app development. Developers should have a strong command of either language to work on the app's codebase.
4. **Backend Infrastructure**: The app may require a backend infrastructure, including servers, databases, and APIs, to support features like user authentication, data storage, and external service integrations. The specific backend technologies and infrastructure requirements depend on the app's architecture and implementation.
5. **Libraries and Frameworks**: The app may utilize various libraries and frameworks for specific functionalities, such as networking, database management, user interface design, and security. The specific libraries and frameworks used will depend on the app's technical requirements and the preferences of the development team.

Chapter 3

SYSTEM DESIGN

3.1. Architecture Diagram :

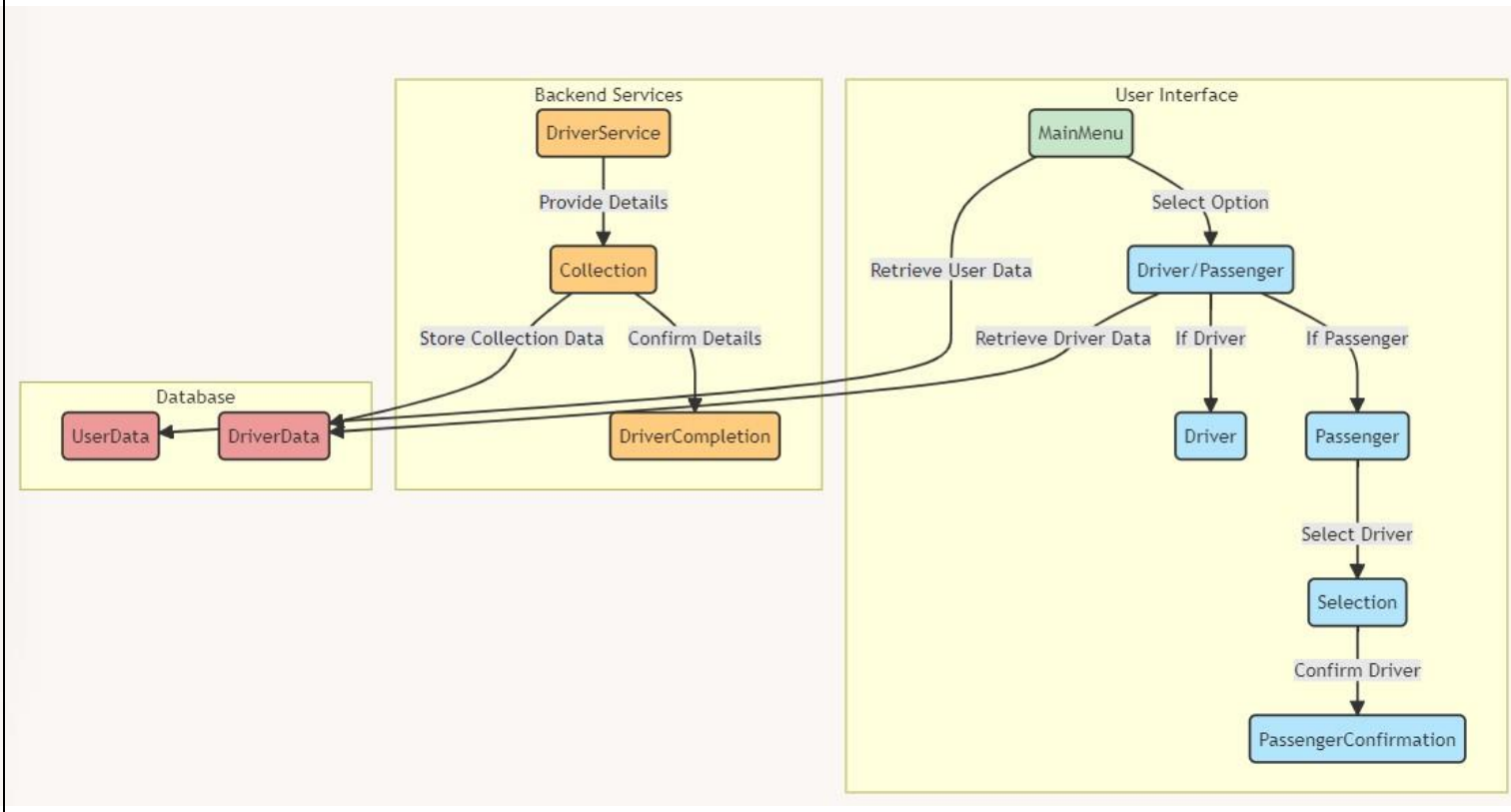

Fig 3.1 Architecture Diagram

Fig 3.1 represents the Architecture Diagram illustrating the interaction flow between various participants in the carpooling application. This diagram provides an overview of the major steps and interactions involved in the carpooling application's user journey. Representing the user of the carpooling application followed by the main menu screen where the user can choose between the driver or passenger option. Then comes the driver/passenger selection screen where the user can choose to be a driver or a passenger. If the user selects the driver option, they proceed to the driver screen.The driver enters their driver details and confirms them. If the user selects the passenger option, they proceed to the passenger screen. The passenger selects their preferred driver and confirms the selection.The diagram showcases the flow of actions starting from the user logging in, choosing between driver or passenger, and then further branching into the respective driver and passenger functionalities. The driver enters their details and confirms them, while the passenger selects their preferred driver and confirms the selection.

3.2. FlowChart :

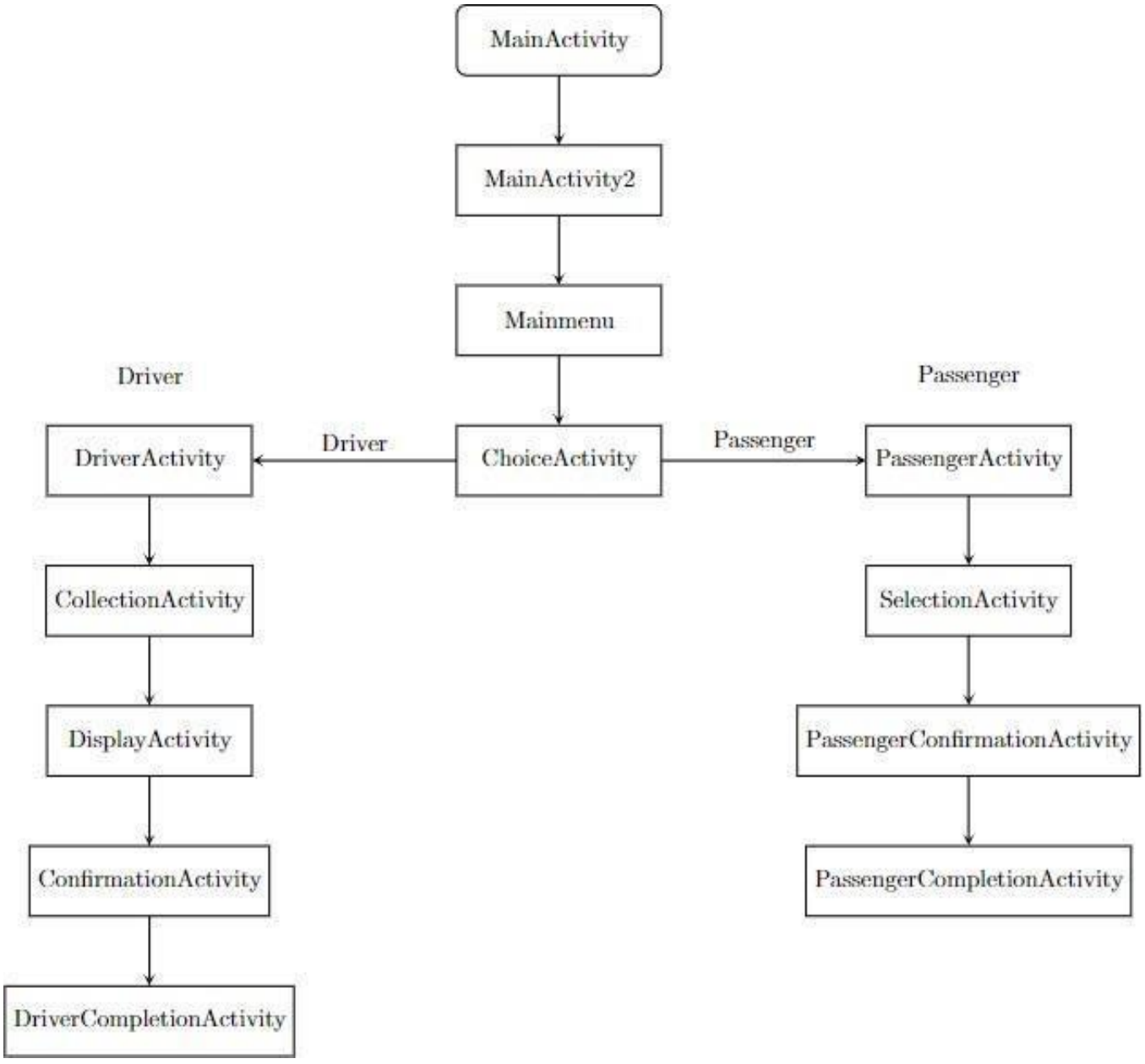

Fig 3.2 Flowchart

Fig 3.2 represents the flowchart diagram illustrating the functioning sequence of activities and the flow of the BookMyPool app.

3.3. User Diagram :

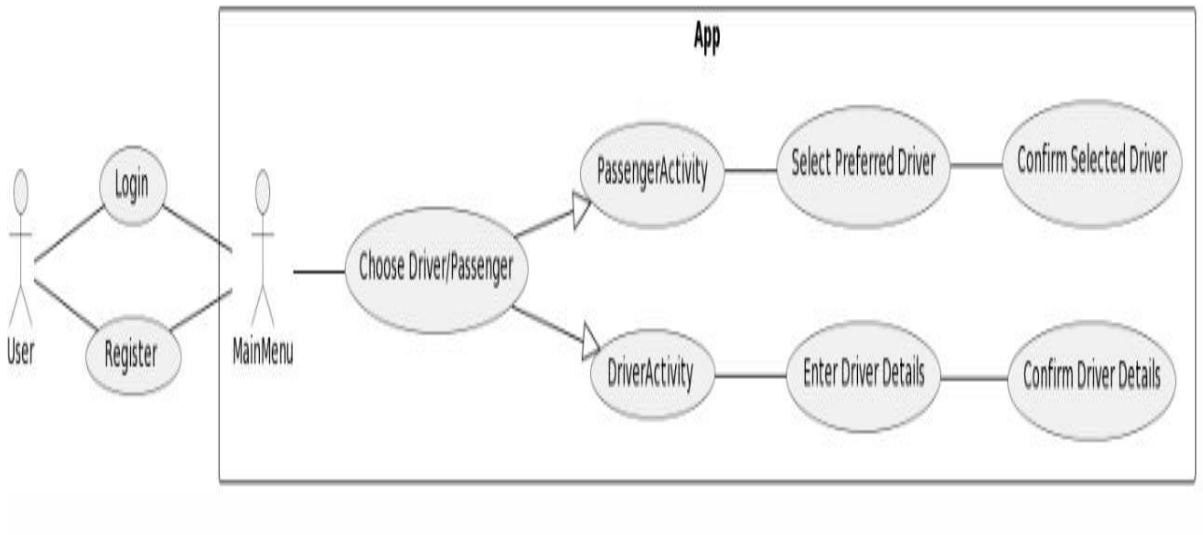

Fig 3.3 User Diagram

Fig 3.3 represents The User Diagram represents the entities involved in the BookMyPool app and their relationships. It includes actors such as the User, Driver, and Passenger. The User Diagram provides an overview of the entities interacting with the app and their roles in the system.

3.4. Sequence Diagram :

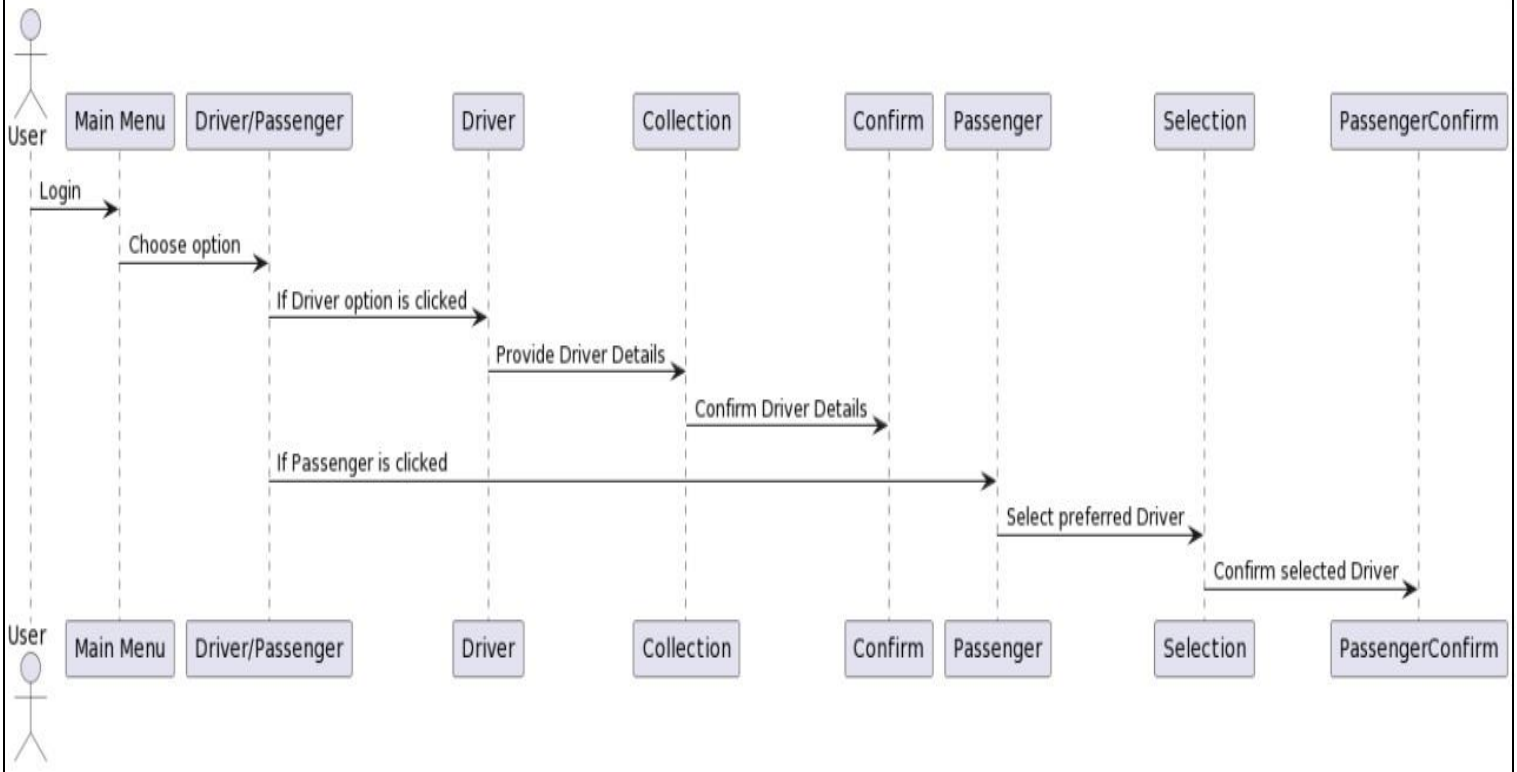

Fig 3.4 Sequence Diagram

Fig 3.4 represents The Sequence Diagram shows the interactions between different components or entities in the BookMyPool app over time. It depicts the chronological order of messages or events exchanged between actors, objects, or systems. The Sequence Diagram helps to visualize the flow of control and communication between entities, showcasing how they collaborate to achieve a specific task or functionality.

3.5. Activity Diagram :

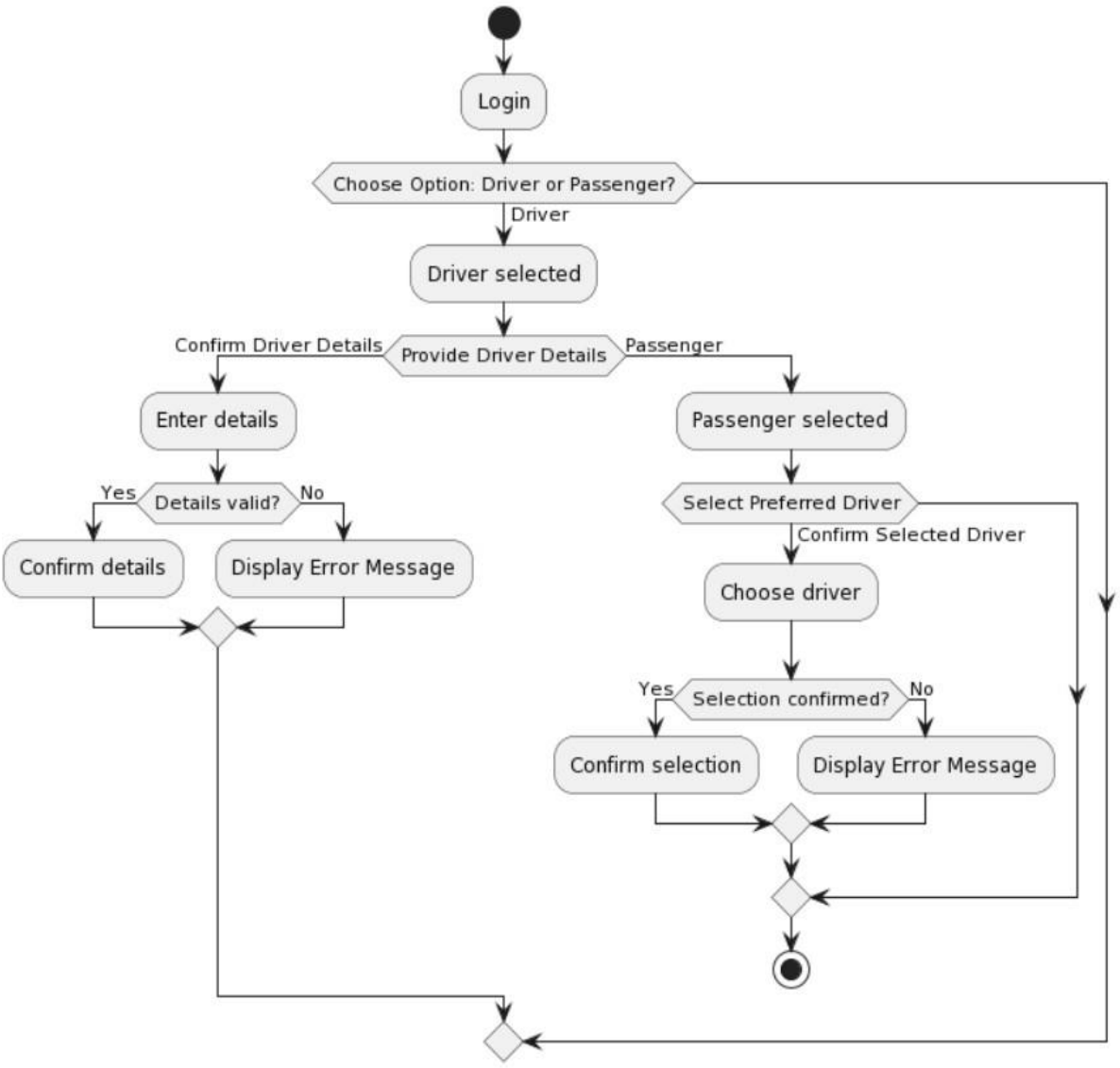

Fig 3.5 Activity Diagram

Fig 3.5 represents The Activity Diagram represents the workflow or flow of activities within the BookMyPool app. It shows the different states, actions, and decisions involved in performing a specific task or process. The Activity Diagram provides a visual representation of the app's behavior and the sequence of steps that need to be followed to accomplish a particular goal.

3.6. UseCase Diagram :

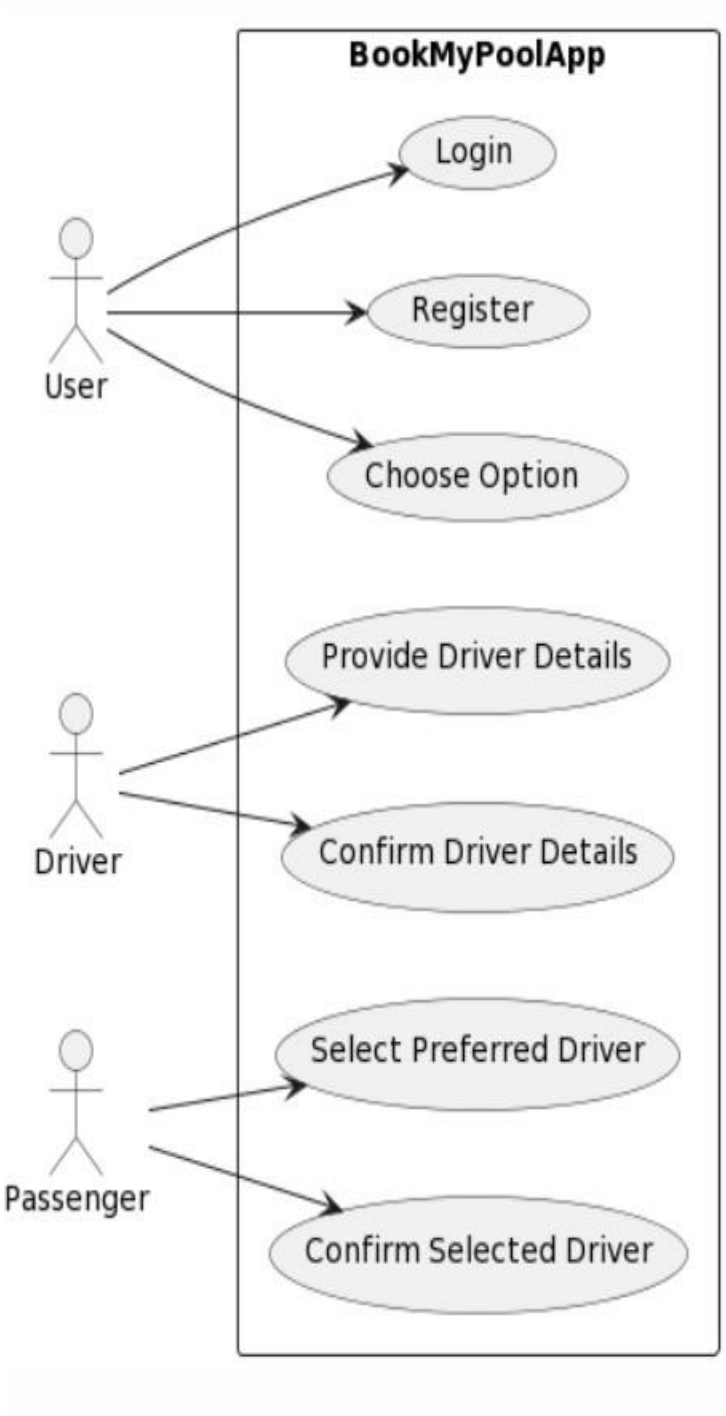

Fig 3.6 UseCase Diagram

Fig 3.6 represents The Use Case Diagram for the BookMyPool app illustrates the various use cases or functionalities provided by the system. It shows the interactions between actors (such as User, Driver, and Passenger) and the system itself. The Use Case Diagram highlights the actions or operations that users can perform within the app, including Login, Register, Choose Option, Provide Driver Details, Confirm Driver Details, Select Preferred Driver, and Confirm Selected Driver.

Chapter 4

IMPLEMENTATION

Pseudocode For Login:

1: **Input:** email, password
2: **Output:** success or failure message
3: **Procedure:**
4: Initialize SQLite through DBHelper
5: Authenticate the user with email and password
6: **Function:** authenticateUser(email, password)
7: **Input:** email, password
8: **Output:** success or failure message
9: Procedure:
10: Create an instance of the Sqlite
11: Call the signInWithEmailAndPassword() method with the provided email and password
12: **If** authentication is successful
13: Grant access to the app
14: Display success message
15: **Else**
16: Display failure message
17: **EndIf**
18: **EndFunction**
19: Call the authenticateUser() function with the input email and password

Pseudocode For Register:

1: **Input:** email, password,,repassword
2: **Output:** success or failure message
3: **Procedure:**
4: Initialize SQLite through DBHelper
5: Validate the input fields (email, password)
6: **if** input fields are valid then
7: Create a new user account in Sqlite with the given email and password and repassword.
8: **if** user account creation is successful **then**
9: Save user information (email,password,repassword) to the SQLite database
10: Display success message
11: **else**
12: Display failure message
13: **end if**
14: **else**
15: Display failure message
16: **end if**

CHAPTER 5

TESTING

Software testing is the process of evaluating a software system or application to determine if it satisfies the specified requirements and to identify any defects or issues that need to be addressed. It is an essential step in the software development process, as it helps to ensure that the software is of high quality and that it functions as intended. There are many different types of software testing, each with its own specific purpose and methodology.

5.1 Testing process

Best testing process is to test each subsystem separately, as we have done in project. Best done during implementation. Best done after small sub-steps of the implementation rather than large chunks. Once each lowest level unit has been tested, units are combined with related units and retested in combination. This proceeds hierarchically bottom-up until the entire system is tested as a whole.

5.2 Unit testing

Unit Testing is a type of software testing that is performed on individual units or components of the software system. It is typically done by developers, and it helps to ensure that each unit of the software functions as intended.

5.3 Integration testing

Integration Testing is a type of software testing that is performed on a group of integrated units or components of the software system. It is typically done after unit testing, and it helps to ensure that the different units of the software work together correctly.

5.4 System testing

System testing tests a completely integrated system to verify that it meets its requirements. After the completion of the entire module, they are combined together to test whether the entire project is working properly.

5.5 Performance testing

Evaluating the systems performance, scalability, and responsiveness under different load or test conditions.

5.6 Test Cases

A Test Case is a software testing document, which consists of events, actions, inputs, outputs, expected result and actual result. Technically a test case includes test description, procedure, expected result and remarks. Test cases should be based primarily on the software requirements and developed to verify correct functionally and to establish conditions that reveal potential errors.

Test Case	Test Description	Input	Expected Output	Actual Output	Pass
1	Valid Username and password	Username: "test@example.com", password: "Aa1@eeee"	Successful login	Successful login	Pass
2	Valid Username and invalid password	Username: "test@example.com", password: "11eee"	Error: Invalid password	Error: Invalid password	Pass
3	Empty Username or password field	Username: " ", password: "123"	Error: Username field cannot be empty	Error: Username field cannot be empty	Pass
4	Valid Username and password containing emoji	Username: "test@example.com" Password: "🔒😊"	Error: Invalid username or password	Error: Invalid username or password	Pass
5	Valid Username and password in Uppercase	Username: " TEST@EXAMPLE.COM", password: " AA1@EEEE"	Error: Invalid username or password	Error: Invalid username or password	Pass
6	Invalid email containing emoji and valid password	Username: "😊test@example.com", password: "Aa1@eeee"	Error: Invalid username or password	Error: Invalid username or password	Pass
7	Invalid email missing .com and valid password	Username: "test@examplem", password: "Aa1@eeee"	Error: Invalid username or password	Error: Invalid username or password	Pass
8	Username and password for sign in	Username:test@gmail.com Password:"Aa1@aaaa" Re-enter:" Aa1@aaaa"	Password is valid and sign in successful	Password is valid and sign in successful	Pass
9	Username and password for sign in	Username :test@gmail.com Password:"Aa1@aaaa" Re-enter:" Aa1@aa"	Password doesn't match	Password doesn't match	Pass
10	Username and password for login	Username :test@gmail.com Password:"1Xa!asde"	Login successful	Login successful	Pass
11	Username and password for login	Username :test@gmail.com Password:"Aa1@a"	Invalid password or username	Invalid password or username	Pass
12	Username and password for login	Username :@gmail.com Password:"Aa1@aaaa"	Invalid password or username	Invalid password or username	Pass

13	Invalid password or username login	Username :test@gmom Password:	Invalid password or username	Invalid password or username	Pass
14	Username and password.containing emoji characters	Username: "😊test@example.com" password: "🔒😀"	Invalid password or username	Invalid password or username	Pass
15	Username and password with whitespace as valid input	Username: " " , password: " "	Invalid password or username	Invalid password or username	Pass
16	Username and password with non-ASCII characters	Username: "….”@example.com", password: "…."	Invalid password or username	Invalid password or username	Pass
17	Invalid Username and password in uppercase	Username: "TEST@EXPLE.COM", password: "PASSWORD"	Invalid password or username	Invalid password or username	Pass
18	Username and password in lowercase	Username: "test@example.com", password: "password"	Invalid password or username	Invalid password or username	Pass
19	Username and password with leading/trailing whitespaces	Username : " test@example.com ", password: " 123 "	Invalid password or username	Invalid password or username	Pass
20	Username and password with SQL injection characters	Username: "test@example.com'; DROP TABLE users;", password: "123' OR '1'='1"	Invalid password or username	Invalid password or username	Pass

Chapter 6

RESULT AND DISCRIPTION:

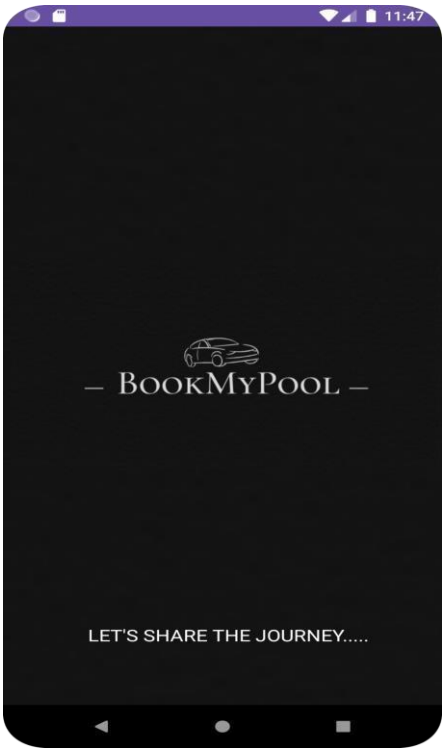

Fig 6.1 SplashScreen

Fig.6.1 shows the SplashScreen which is the initial screen that appears when the BookMyPool app is launched. It serves as a visual indication to the user that the app is loading and prepares the user for the upcoming content.

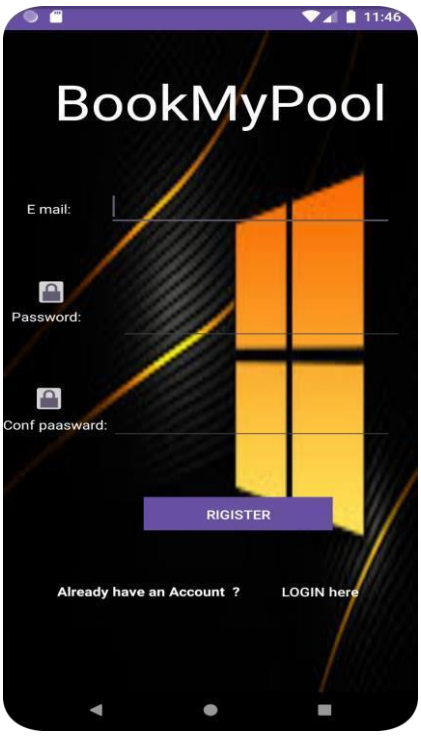

Fig 6.2 Register Page

Fig 6.3 Login Page

Fig.6.2 depicts the Register Page providing their email, password, and confirming the password.Once users have filled in the necessary information and passed the validation checks, they can proceed by clicking the registration button. The app then saves the user'sdetails in the database.

Fig.6.3 depicts the Login Page of the BookMyPool app. This page enables registered users to access their accounts by entering their email and password.Upon entering valid login credentials, users can initiate the login process by clicking the login button. The app then verifies the provided information against the stored user data in the database. If the login credentials match, the app grants the user access to their personalized account and proceeds to the main app interface.

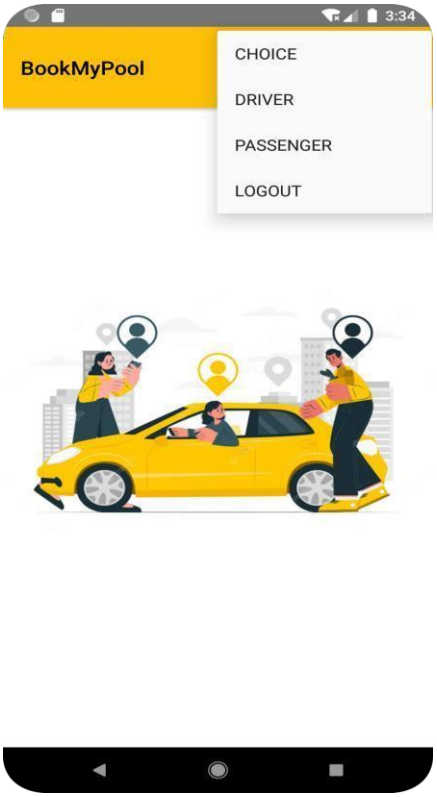

Fig 6.4 Mainmenu

Fig 6.4 illustrates the Mainmenu page of the BookMyPool app, which features a dropdown menu providing convenient access to various pages and functionalities within the app. The dropdown menu offers options to go to the ChoiceActivity, Driver, and Passenger pages.

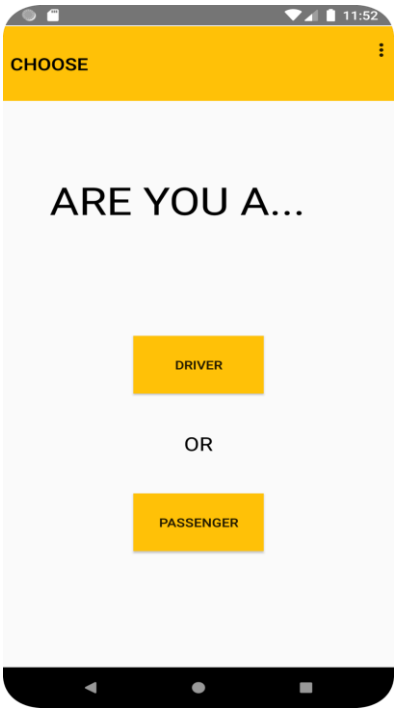

Fig 6.5 Choice Page

Fig 6.5 illustrates the Choice Activity page of the BookMyPool app, which asks the user to choose if he/she is a driver or passenger.

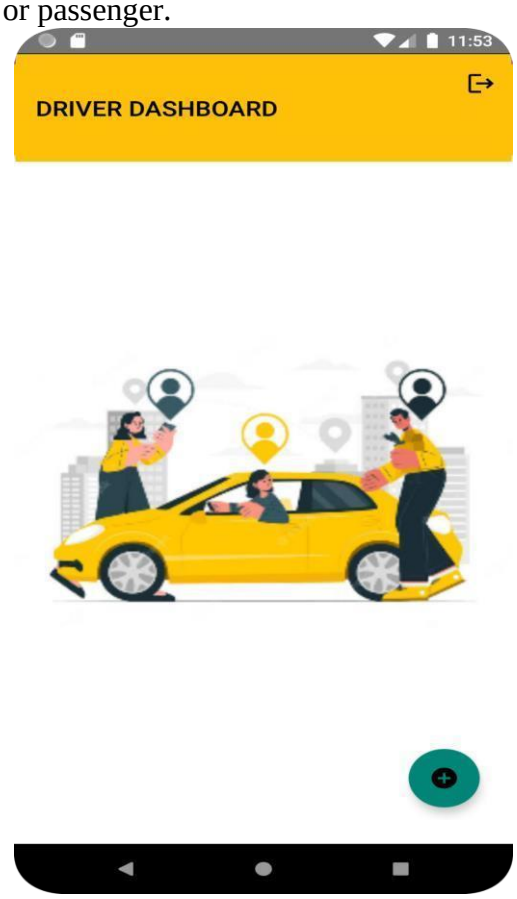

Fig 6.6 Driver Dashboard

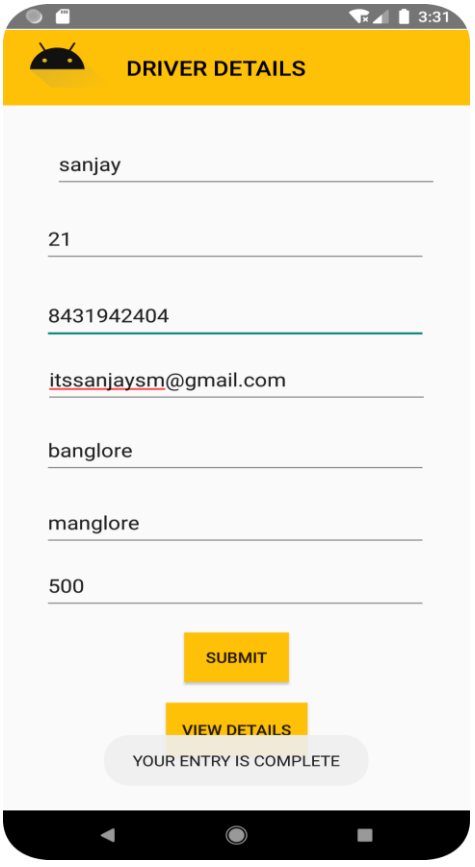

Fig 6.7 Driver Details

Fig 6.6 illustrates the Driver Dashboard of the BookMyPool app, which has a floating action button that is used to move to the driver details page.
Fig 6.7 illustrates the Driver Details Page of the BookMyPool app, which allows the driver to enter his details.

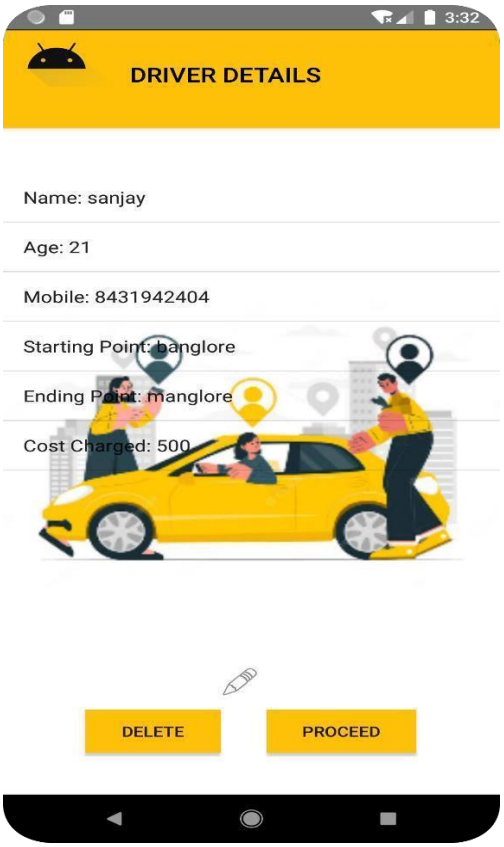

Fig 6.8 Displayed Details

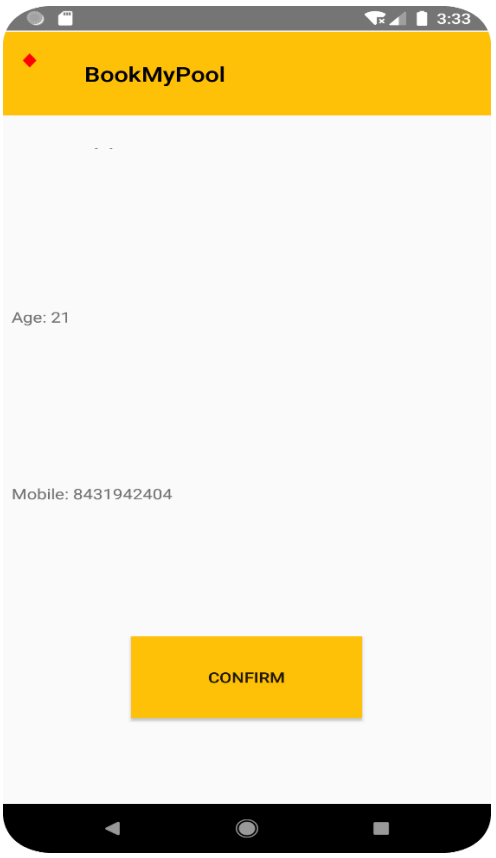

Fig 6.9 Confirm Details

Fig 6.8 displays the view of the driver details entered in the BookMyPool app, which allows to edit ,delete & proceed according to the user.
Fig 6.9 illustrates the Confirm Page of the BookMyPool app, which confirms the entered details.

Fig 6.10 Final Driver page

Fig 6.10 shows the final display page for the driver in the BookMyPool app, this is the end page for the driver which will redirect the page to logout or mainmenu

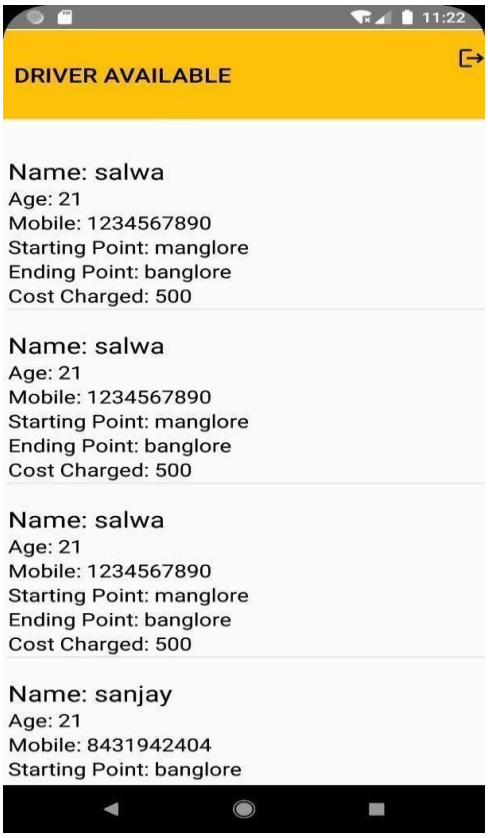

Fig 6.11 Drivers Available

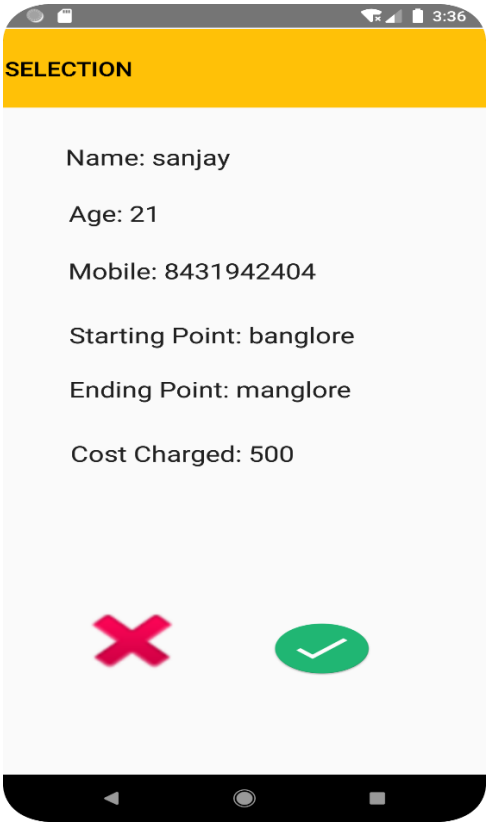

Fig 6.12 Confirm Selected Driver

Fig 6.11 represents the list of drivers available in the passengers dashboard of the BookMyPool app, which allows the passenger to choose their desired driver.
Fig 6.12 gives the confirm or reject option for the driver selected in the BookMyPool app.

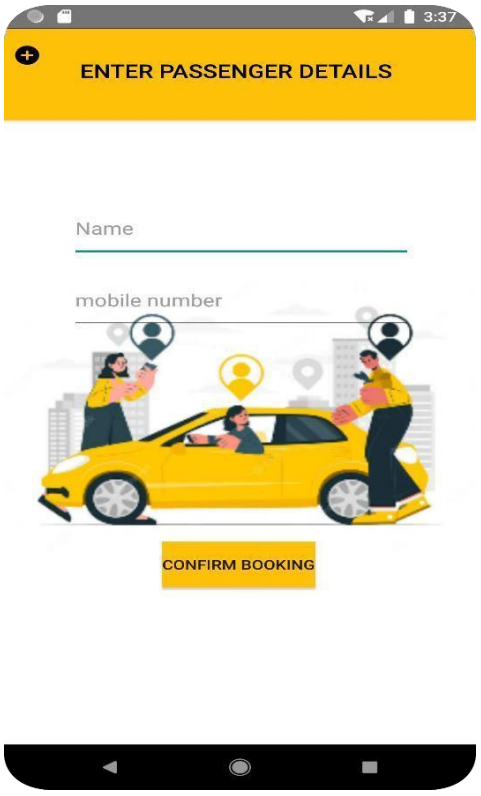

Fig 6.13 Passenger Details

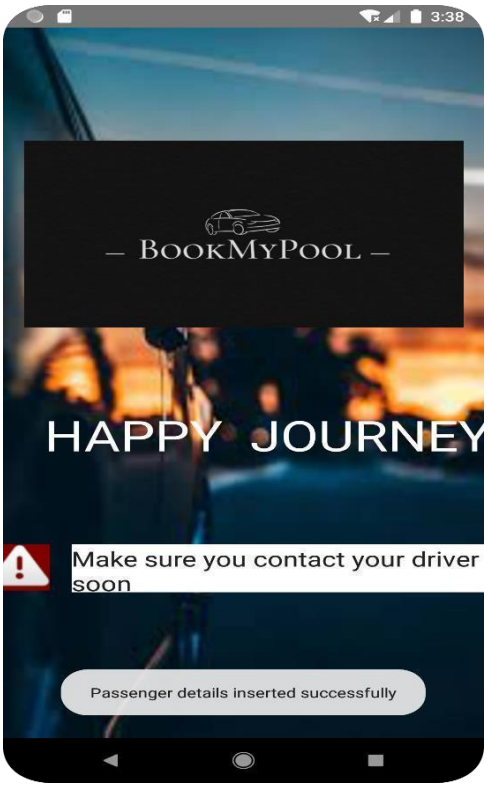

Fig 6.14 Final Passenger page

Fig 6.13 shows a window where passenger has to enter their name &mobile number details and confirm booking.
Fig 6.14 shows the final display page passenger in the which is the end page for the passenger .

CONCLUSION

In conclusion, the BookMyPool app is a comprehensive and user-friendly application designed to streamline the process of booking and managing pool services. The app provides a seamless experience for both drivers and passengers, offering a range of features and functionalities.

Throughout the development of the app, various activities were implemented to cater to different user interactions. Starting with the main activities like MainActivity and MainActivity2, users are guided through the app's functionalities, including accessing options, selecting choices, and interacting with specific features.

For drivers, the app offers activities such as DriverActivity, CollectionActivity, DisplayActivity, and ConfirmationActivity. These activities enable drivers to enter their details, collect items, view the collected items, and confirm their selections. The DriverCompletionActivity signifies the completion of driver tasks.

Passengers, on the other hand, have their own set of activities, including PassengerActivity, SelectionActivity, PassengerConfirmationActivity, and PassengerCompletionActivity. These activities allow passengers to view and select driver details, confirm their selections, and complete the passenger tasks.

The app's architecture follows a well-defined flow, ensuring a smooth user experience and efficient functionality. The activities are interconnected, facilitating the seamless progression from one step to another.

Overall, the BookMyPool app provides an intuitive and convenient platform for booking pool services. It enhances the interaction between drivers and passengers, simplifying the process and improving overall efficiency. With its user-friendly interface and robust functionality, the app offers a reliable solution for managing pool bookings effectively.

REFERENCES

1. Android Studio Documentation :

- https://developer.android.com/courses/android-basics-compose/unit-2
- https://developer.android.com/reference/androidx/cardview/widget/CardView
- https://developer.android.com/reference/android/window/SplashScreen

2. Beginning Android Programming with Android Studio, Fourth Edition:

- https://www.wowebook.org/beginning-android-programming-with-android-studio-fourth-edition/

3. Geeks for geeks :

- https://www.geeksforgeeks.org/how-to-install-android-virtual-deviceavd/

4. Papers :

- Holzer, A., & Ondrus, J. (2009, April). Trends in mobile application development. In *International Conference on Mobile Wireless Middleware, Operating Systems, and Applications* (pp. 55-64). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Albertengo, G., Debele, F. G., Hassan, W., & Stramandino, D. (2020). On the performance ofweb services, google cloud messaging and firebase cloud messaging. *Digital Communications and Networks*, *6*(1), 31-37.

5. Android Developers YouTube Channel:

- https://www.youtube.com/user/androiddevelopers

6. Stack Overflow:

- https://stackoverflow.com/

7. Android Arsenal:

- https://android-arsenal.com/

8. Install DB Browser for SQLite in Android & attach a Database in SQLite Browser

- Link: https://www.youtube.com/watch?v=N0cP415v5B0