Mini-Project Report On

AutoInsight (A one-stop destination for all car lovers)

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

Bachelor of Technology

in

Computer Science & Engineering

By

Neil Sunny (U2003148)
Nevil Biju Varghese (U2003150)
Salman Sidhik (U2003182)
Siddhartha Goutaman (U2003201)

Under the guidance of
Ms. Anita John

RSET
RAJAGIRI SCHOOL OF ENGINEERING & TECHNOLOGY (AUTONOMOUS)

Department of Computer Science & Engineering
Rajagiri School of Engineering and Technology (Autonomous)
Rajagiri Valley, Kakkanad, Kochi, 682039

July 2023

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
RAJAGIRI SCHOOL OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
RAJAGIRI VALLEY, KAKKANAD, KOCHI, 682039

CERTIFICATE

This is to certify that the mini-project report entitled ***"AutoInsight (A one stop destination for all car lovers)"*** *is a bonafide work done by* ***Mr. Neil Sunny (U2003148), Nevil Biju Varghese (U2003150), Salman Sidhik (U2003182), Siddhartha Goutaman (U2003201)****, submitted to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B. Tech.) in Computer Science and Engineering during the academic year 2022-2023.*

Dr. Preetha K. G.	**Ms. Anita John**	**Ms. Anita John**
Head of Department	Mini-Project Coordinator	Project Guide
Professor	Asst. Professor	Asst. Professor
Dept. of CSE	Dept. of CSE	Dept. of CSE
RSET	RSET	RSET

ACKNOWLEDGEMENTS

We wish to express our sincere gratitude towards **Dr. P. S. Sreejith**, Principal of RSET, and **Dr. Preetha K. G.**, Professor and Head of Department of Computer Science and Engineering for providing us with the opportunity to undertake our mini-project, "AutoInsight App".

We are highly indebted to our mini-project coordinators, **Ms. Anita John**, Assistant Professor, Department of Computer Science and Engineering and **Mr. Sajanraj T. D.**, Assistant Professor, Department of Computer Science and Engineering for their valuable support.

It is indeed our pleasure and a moment of satisfaction for us to express our sincere gratitude to our mini-project guide **Ms. Anita John**, for her patience and all the priceless advice and wisdom she has shared with us.

Last but not the least, we would like to express our sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

Neil Sunny

Nevil Biju Varghese

Salman Sidhik

Siddhartha Goutaman

ABSTRACT

Our goal is to develop a car analysis app that provides comprehensive information and analysis on various aspects of automobiles. The app aims to assist users in making informed decisions when buying, selling, or maintaining a car. By offering a range of features and tools, the app will empower users with detailed insights into the performance, reliability, market value, and maintenance requirements of different car models.The app provides a vast database of car models, including their specifications, features, performance metrics, and historical data. Users will be able to analyse the performance of a specific car model by accessing data such as acceleration, top speed, fuel efficiency, and handling characteristics.

The app also offers information about the reliability ratings, common issues, and maintenance requirements for different car models. Users will be able to compare the maintenance costs of various models and estimate the overall cost of ownership.They can also determine the market value of a car based on factors such as the model, age, mileage, condition, and location. The app provides tools to calculate depreciation rates and estimate the potential resale value of a car.The app will have an intuitive and user-friendly interface that allows users to easily navigate through different sections, search for specific car models, and access relevant information and analysis.

Contents

List of Figures

Chapter 1

Introduction

1.1 General Background

AutoInsight is an innovative mobile application designed to provide automotive enthusiasts and potential car buyers with a centralized platform to access extensive information about a wide range of vehicles. The app not only offers detailed specifications and performance data for various car models but also equips users with a comprehensive maintenance guide and authentic user reviews. Additionally, AutoInsight streamlines the car-buying process by offering test drive bookings and the option to customize purchases by selecting interior trims and exterior colors. This report introduces the key features and functionalities of AutoInsight, highlighting its significance in revolutionizing the car research and purchasing landscape.

- The automotive industry is constantly evolving, with a multitude of car models introduced each year, making it challenging for consumers to make informed decisions about their ideal vehicles. AutoInsight addresses this concern by presenting an all-in-one platform that amalgamates essential car information, user experiences, and the convenience of personalized purchasing.

- Key Features:

  - Comprehensive Car Database: AutoInsight boasts an extensive and constantly updated database, encompassing a diverse array of cars, ranging from popular models to niche options. Users can access detailed specifications, performance metrics, and high-quality images for each listed vehicle, facilitating a comprehensive understanding of their favorite models.
  - Maintenance Guide: Proper car maintenance is paramount to ensuring longevity, performance, and safety. AutoInsight provides users with a comprehensive

maintenance guide, including routine service schedules and essential tips for upkeep. This feature empowers car owners to optimize the performance of their vehicles while minimizing potential repair costs.

  - Authentic User Reviews: Understanding the real-world experiences of other car owners is crucial for making informed decisions. AutoInsight's user review section enables individuals to share genuine feedback about their vehicles, highlighting both positive aspects and areas of improvement. This feature fosters a community-driven environment, enriching the car research process.
  - Test Drive Bookings: With AutoInsight, potential buyers can schedule test drives directly through the app. This streamlined feature allows users to experience their desired vehicles firsthand, helping them assess driving comfort, handling, and overall suitability.
  - Customized Purchases: AutoInsight revolutionizes the car-buying journey by offering users the ability to customize their purchases. By selecting interior trims, exterior colors, and optional features, buyers can design their dream cars seamlessly within the app. This personalized approach enhances the overall purchasing experience and encourages greater satisfaction with the final product.
  - Hidden Features Descriptions: We provide detailed descriptions of its hidden features. These could include unique functionalities, advanced settings, or lesser-known capabilities that are not commonly mentioned in the car's user manual. These could be crucial for making or breaking the bank or even save lives.

- AutoInsight's comprehensive approach to automotive research and purchase fosters a more informed and confident consumer base. By offering a user-friendly interface, up-to-date information, and a platform for user interactions, the app bridges the gap between automotive enthusiasts and car dealerships, empowering individuals to make well-informed decisions while simplifying the car-buying process.

- AutoInsight emerges as a transformative mobile application that revolutionizes the way automotive enthusiasts and potential car buyers approach their vehicle research

and purchasing endeavors. With its comprehensive car database, maintenance guide, user reviews, test drive bookings, and personalized purchase options, AutoInsight stands as a cutting-edge solution in the automotive industry, catering to the evolving needs of consumers and cultivating a more empowered and gratified car-buying community.

1.2 Objectives

The objective of our app is to provide users with a comprehensive car maintenance/-booking experience by offering detailed car data, performance analysis, reliability and maintenance information, market value insights, and a user-friendly interface. We aim to empower users to make well-informed decisions when buying or owning a car, ensuring they have access to all the essential information they need. By providing comprehensive car data, performance analysis, and reliability information, our app enables users to compare and evaluate different car models. Additionally, we help users assess the market value and resale potential of specific cars, contributing to their financial decision-making process. Our user-friendly interface ensures a seamless and intuitive experience, allowing users to navigate effortlessly and access the desired car information. Overall, our app aims to enhance the car buying and ownership experience, providing users with the tools they need to make informed decisions and optimize their car-related activities.

1.3 Motivation

Our motivation for developing this app is driven by a passion for cars and a commitment to enhancing the car buying and ownership experience. We recognize the lack of an all-encompassing solution that offers accurate and in-depth analysis and information about cars, hindering users from navigating the complex automotive landscape. Existing resources often fall short in terms of accuracy, accessibility, and personalization, leaving users without a reliable source for making informed choices. Our motivation lies in bridging this gap by developing a platform.

1.4 Summary of the Report

This report provides an overview of the development process involved in creating an Android application aimed at helping users find their ideal car and optimize their ownership journey. The report is divided into several chapters, each addressing specific aspects of the project. The first chapter introduces the project's background, objective, and motivation. The second chapter includes an analysis of relevant papers and research materials that were referenced throughout the project. The third chapter focuses on the problem definition, system architecture, scope, and implementation details, including the technologies used. The fourth chapter presents the final results, along with discussions and analysis of the project's outcomes. The report concludes with a summary of the entire project, its implications, and potential future developments. It also includes a reference section and appendixes containing supplementary information such as the project code and objectives.

Chapter 2

Literature Review

2.1 Car-Based Mobile Applications and User Experience

Authors: Rachel Harrison, Derek Flood, David Duce[2]

The usefulness of mobile devices has increased greatly in recent years allowing users to perform more tasks in a mobile context. This increase in usefulness has come at the expense of the usability of these devices in some contexts. We conducted a small review of mobile usability models and found that usability is usually measured in terms of three attributes; effectiveness, efficiency and satisfaction. Other attributes, such as cognitive load, tend to be overlooked in the usability models that are most prominent despite their likely impact on the success or failure of an application. To remedy this we introduces the PACMAD (People At the Centre of Mobile Application Development) usability model which was designed to address the limitations of existing usability models when applied to mobile devices. PACMAD brings together significant attributes from different usability models in order to create a more comprehensive model. None of the attributes that it includes are new, but the existing prominent usability models ignore one or more of them. This could lead to an incomplete usability evaluation. We performed a literature search to compile a collection of studies that evaluate mobile applications and then evaluated the studies using our model.

Existing models of usability Nielsen identified five attributes of usability:

- Efficiency: Resources expended in relation to the accuracy and completeness with which users achieve goals;
- Satisfaction: Freedom from discomfort, and positive attitudes towards the use of the product.
- Learnability: The system should be easy to learn so that the user can rapidly start

getting work done with the system;

- Memorability: The system should be easy to remember so that the casual user is able to return to the system after some period of not having used it without having to learn everything all over again;

- Errors: The system should have a low error rate, so that users make few errors during the use of the system and that if they do make errors they can easily recover from them. Further, catastrophic errors must not occur

2.2 Car test drive Slot Booking System

Authors: Chaganti Sandeep Reddy, Dr. Preeti Savant[1]

Current day market going digital market, Applications use to help you give more peace of mind to people. Electrical mobile and desktop applications are the most attractable and effective way and time consuming and also to reach out to customers today. Using these applications is the trending technology now. So, today developers, think in an innovative way and always try to provide the best apps and makes users' life more comfortable and easier. Day by day so many mobile applications are being launched every day with new technology and with smart work. Car cleaning is an android application that brings back life to users and more fast works and more time saves work. So, it provides car services to the customers by taking the customer wish to take as input. We develop the application so compulsory to use the Android Studio as a tool for building and developing the app. It's also the official IDE (Integrated Development Environment) for android. Android platform is more compatible with both mobile and digital computers so it is used to the motivation behind initiating this project is to make people's life more comfortable and smoother. Our application assists people to get their car services at their fingertips. The app It will store all the user details can store the data in the database and it builds the data store very safe and secures it concludes the overall the user data will be clear and higher technology. Piramal Abhishek, Deep Shrivastava, "Vehicle Service Management and Live Monitoring with Predictive Maintenance System". This paper explains vehicle service and monitors the vehicle's prediction system. The Car Service app helps solely customers to book the car wash service in which there are different options available to the users and its available for free of cost, easy to use and saves time. 'Car Service' is

a method to use to book the slot and also every method works under the process and is maintained. This system has mainly developed some different either online or offline. The booking is very easy for your customers and clients a straightforward. Today's car service method is increasing step by step with useful to humans. It is also one type of development the system in car cleaning. Early this process was just one-two way wise using only and using human power and servicing. Later it generated the different types of methods and types in the system. It was the latest and advanced car service system and either service twice or mechanical wise both are the same features and same design. In 2000 there was a small change in washing and but the method is also the same. Later by time giving it was the small method and growing firstly. Slowly this system depends on human strength and works together.

A. Advantage

- Customers can book the Car Service and just one click but used with internet only.
- Easy to use open the app, book the slot on which day just book it and free it.
- All the services are using the same method but based on the service's vice they just change in working format.
- This system is mainly useful for no need to carry cash or either paper or pen just in one-touch operation.
- Based on the veiled system it changes automatically and fasts forward.
- By using this feature multiple cars can wash by at a time on user demand.

B. Disadvantage

- Internet is compulsory to book the car service slot booking system.
- If you need to book the car service, we have Internet, an online booking application on our mobile.
- Weather all installed booking services major are uniform we mostly at least one supplier.
- Sometimes users book the service but didn't get the confirmation message due to server or internet issues.

2.3 Firebase (Backend as A Service) for Mobile Application Development

Authors: Prachi R. Saraf, Sakshi M. Jadhao, Saurabh J. Wanjari, Shital G. Kolwate, Prof. Ankush D. Patil[4]

A. Backend As A Service(BaaS) In the area of mobile application development, there are two major terms on which the developers has to work i.e. for frontend and backend. Backend is the most crucial part of mobile application development which responsible for storing the data, securing data, etc. The backend of the application is like a server for mobile apps, as it stores and sorts the data properly and the end user can only see the necessary information. BaaS i.e Backend-as-a-Service is a cloud service model in which developers outsource all the behind-the-scenes aspects of a web applications or mobile application so that they only have to write and maintain the frontend part. BaaS vendors provide pre-written software for activities that take place on servers, such as user authentication, database ,remote updating, and push notifications (for mobile apps), as well as cloud storage and hosting.

B. Firebase

1. Introduction to Firebase: Firebase is a real-time database and also acts as a Backend-as-a-Service(BaaS). It allows to store a list of objects. Google Firebase is Google-backed application development software which allows developers to develop applications for Android, iOS, and Web apps. Firebase is a grouping of Google's many services in the cloud, including instant messaging, user authentication, real-time database, storage, hosting, etc. Firebase offer real time database, authentication, cloud storage, cloud functions, etc. Firebase accomplishes real-time data in the database. Firebase makes easy to exchanges the data to and from the database. Firebase provides backend for- iOS, Android, and Web applications. Firebase applications can be arranged over a protected connection to the firebase server. Firebase offers a dashboard simple control. It provides number of useful services. It is highly secure and minimal setup. Cloud Functions feature is one of the new feature of firebase cloud which allows developers to write programs in JavaScript language and placed them on the Firebase cloud platform. Using Firebase, each unit can directly access the database.Firebase is a Google-owned multi-service cloud-computing solution for mobile and web developers. The feature grouping in Firebase rushes the cloud database integration automatically in both web and mobile app.

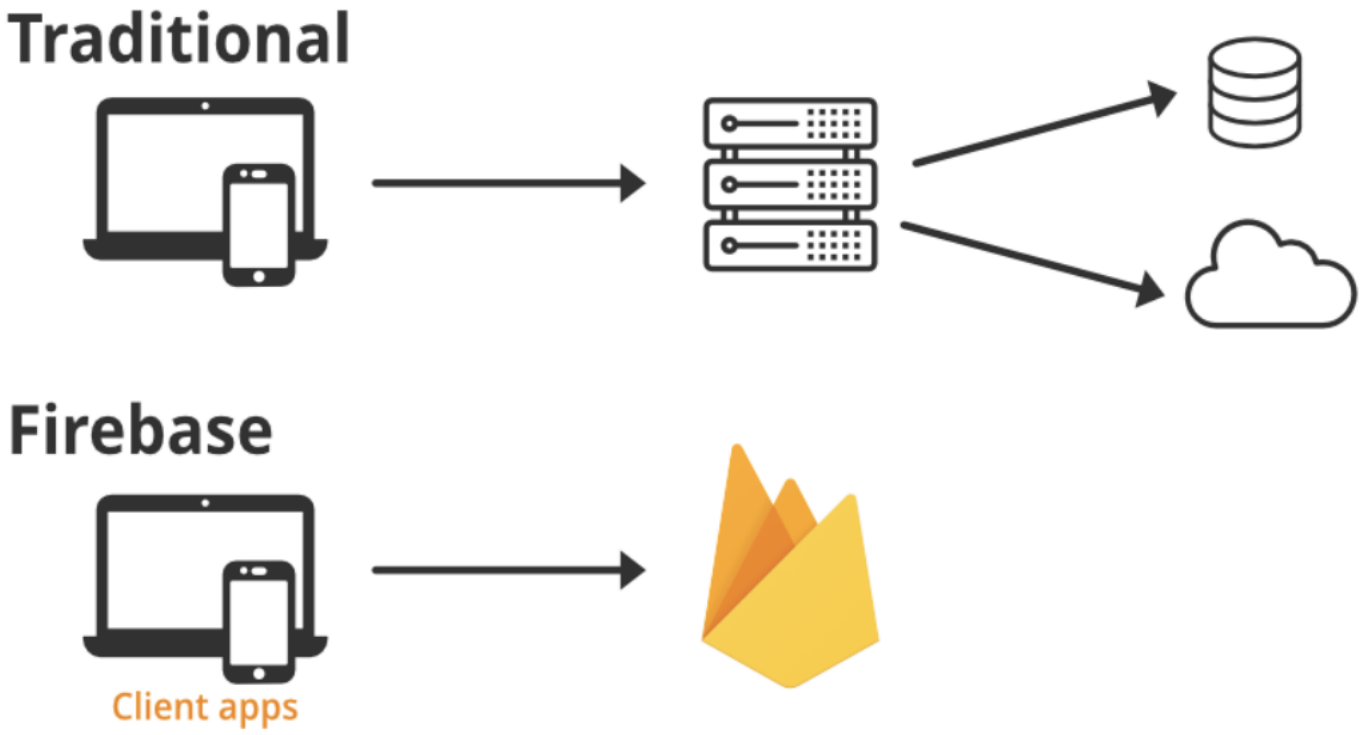

Figure 2.1: LS3 Flowchart

2. Ease of using Firebase: Firebase is a web application development platform created by Google. It lets you to developed the whole application on the front-end without any server-side code. At the same time, it does let you set up some server-side logic through Firebase Functions if you need to react to certain events (Creation of data or files, login, https requests) so that you can send emails or push notifications or process the data after it is written.It's easy to start a project with Firebase or add a Firebase to your project. It allows real-time database connection, which means multiple users can see the changes in the data when the data gets created or edited. Data transmission is handled with web sockets so you don't have to send requests to get new data, you only need to subscribe once.The same applies for file storage. Quick setup authentication through the major providers (Google, Twitter, Facebook, GitHub). https by default - secure http traffic without setting up certificates. Any static html/javascript content can be hosted.

3. Services of Firebase :Analytics, This feature are also enables the application developer to understand how users are using his application. The SDK capture events and properties on its own and also allows you to get custom data. The dashboard also provides

details like your most active user or what feature of your application is used most.

2.4 Mobile Application Development Based on Flutter Platform

Authors: Shreya A. Bhagat, Sakshi G. Dudhalkar, Prathmesh D. Kelapure, Aniket S. Kokare, Prof. Sudesh A. Bachwani[3]

Flutter as a framework is very promising and right now has a big dev community. Even now we can find complex apps in the market which are based on Flutter, like Alibaba, Google Ads, Reflect, Birch Finance, Hamilton Musical, Hookle (Skuza, 2019). In the Authors opinion, this technology is a good choice for small and medium-size applications or when content and basic features require constant iteration. The technology potential is also big as during Flutter interact conference Google introduces support for web applications (Sneath, 2019). Dart language is also the fastest-growing programming language nowadays. Its list features added during the last two years is also big and includes extension functions, null safety support.

Flutter itself is not a programming language. Rather, it's an SDK with prewritten code, consisting of ready-to-use and customizable widgets. The programming language that's used is Dart, is also developed by Google. By avoiding using a bridge to communicate with the native layer such as Android or iOS, Flutter minimizes performance issues and boosts app start-up time.

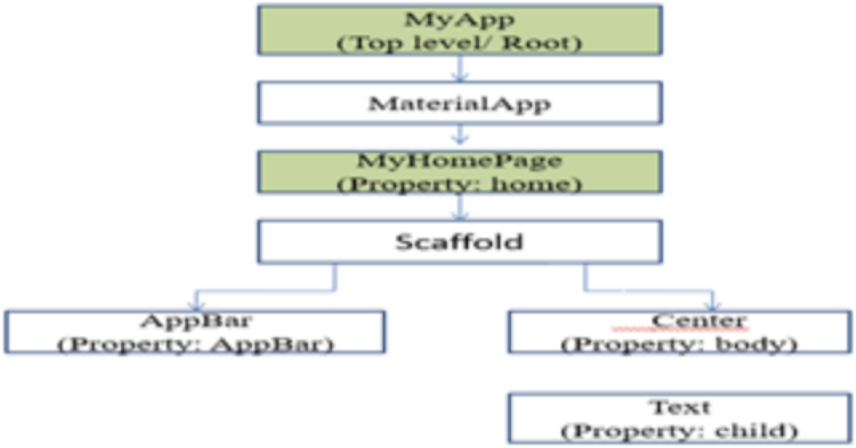

Figure 2.2: Layers of widgets

To develop an app using Flutter, you need developers to code in Dart. That shouldn't

be an issue, because it's similar to Kotlin, Java, Swift, and JavaScript. Also, it's easy to learn. According to Google, Dart is a client-optimized language for fast apps on any platform. Object-oriented like Java, C++, and Python, it compiles ahead of time to native ARM or x64 machine code, and to JavaScript bytecode for web. As a result, apps written in Dart are impossible to distinguish from native apps at the machine level.

Conclusion

This system is useful for those who need easily want to service their own vehicles. The method is very useful and it is a private and protected way. Everyone used this system in multiple ways. Each system has multiple options and generates one by one and given in the same system. Early introduction this system old way has generated but features it has changed and given all the way. The potential of giving the car service where from home or anywhere is fashionable more to materiality in the present world. The subsist companies' same way to the application was designed and current in the literature valuation. It was a freely used application in which the users can interchange directly with the car service. This changes people's life much easier in overripe schedules. The present application is rechanged in a similar way that it underwrites. Making on this stand deduction out Mobile Car Services, we come to the opinion that such computerization systems are quite favorable and save the time of the system. As we going the as regards the technology dominant growth and the business status of the person is also growing the day by day, the source of both is according to the people not only to notice but also to achieve on their own desire. The majority of cars in the world will upgrade in the near time ahead. Any ware always increases the number of cars can go to a mechanic shop to get the facility. This system plays the operation multiple times. By using this application. The new feature was generated by service we can use internet service and also an enriching of person that best service of the user. The whole feature was an instant overhaul and smart and very speed. Every system has not only controlled the real-time condition us maintain the best trump card also bring forth essential data and forecast and next service and the rough cost. Overall, the system adds to the check over cost. All this system saves flow and funds for customers.

Chapter 3

System Analysis

3.1 Expected System Requirements

The system of user which is a smart phone is expected to have the following features:

- Android platform with a version above 8.0(Oreo).
- Requirement of Internet connection to connect to database.
- A storage space of approximate 100 MB for the app.
- A minimum Ram size of 2GB is required in the device.

3.2 Feasibility Analysis

3.2.1 Technical Feasibility

This aspect assesses whether the technology and resources required to build and maintain the app are available and feasible. Considerations include:

- Expertise and skills: Evaluate whether the development team possesses the necessary skills in Flutter, Android Studio, Figma, and Firebase to create the app successfully.
- Integration: Analyze the compatibility and integration of the app with various devices, car models, and operating systems.
- Scalability: Determine whether the app can handle increased user traffic and data as it grows in popularity.

3.2.2 Operational Feasibility

Operational feasibility evaluates whether the app can be effectively integrated into the existing operations and processes. Consider the following:

- User adoption: Analyze the ease of use and acceptability of the app among target users (car owners, potential buyers, etc.).
- Customer support: Plan for providing adequate customer support and addressing user inquiries and issues.
- Data management: Ensure that the database (Firebase) can handle data efficiently and securely.

3.2.3 Economic Feasibility

The app can reduce the overhead of expense incurred by geriatric people in order to maintain physical assets essential for them to interact with society. The development of the app is also zero budget as it was built using free resources.

3.3 Hardware Requirements

The following are the system requirements to develop the AutoInsight App.

- Processor: Intel Core i5 / Apple Silicon M1
- Hard Disk: Minimum 100GB
- RAM: Minimum 8GB

3.4 Software Requirements

The following are the softwares used in the development of the app.

Operating System: Windows and MacOS

3.4.1 Android Studio for flutter app development

Android Studio is a popular Integrated Development Environment (IDE) for developing Flutter apps, as it provides a wide range of tools and features that can help you build high-quality apps faster. Some of the key features of Android Studio for Flutter development include:

A rich set of tools for debugging, testing, and profiling your app. A powerful code editor with support for code completion, refactoring, and more A flexible build system with support for building, testing, and deploying your app. Integration with popular version control systems like Git. A visual layout editor for building attractive user interfaces.

3.4.2 Firebase

Firebase is a Backend-as-a-Service (BaaS) app development platform that provides hosted backend services such as a realtime database, cloud storage, authentication, crash reporting, machine learning, remote configuration, and hosting for your static files.

Firebase supports Flutter. For more information, see: The Firebase plugins page Getting started with Firebase and Flutter Get to know Firebase for Flutter video workshop based on the codelab Get to know Firebase for Flutter codelab Use Firebase to host your Flutter app on the web

3.4.3 Flutter and flutter plugins

Plugins are the wrapper of the native code like android(java or kotlin) and iOS(swift or Objective C). Plugins are written in platform-specific code to access the platform-specific features. Flutter does support using packages and plugins contributed by other developers to its ecosystem

Flutter plugins are thin Dart wrappers on top of native (Java, Kotlin, ObjC, Swift) mobile APIs and services. For instance, if you wanted to access a sensor on the phone, the only way is to write a plugin (or use one that's already there). The API of the plugin is written in Dart.

A module is able to integrate the flutter with the help of the existing native application. These are the major differences between the flutter plugin and the flutter module.

3.4.4 Figma(UI/UX Design)

Figma is a collaborative web application for interface design, with additional offline features enabled by desktop applications for macOS and Windows. The feature set of Figma focuses on user interface and user experience design, with an emphasis on real-time collaboration, utilising a variety of vector graphics editor and prototyping tools. The Figma mobile app for Android and iOS allows viewing and interacting with Figma prototypes in real-time on mobile and tablet devices.

Chapter 4

System Implementation

4.1 Database Design and Firebase Integration

With Firebase chosen as the database platform, the team will proceed to design the database schema. Firebase's NoSQL real-time database structure will be leveraged to create collections and documents to store user data, including usernames, passwords, and previous test drive bookings. Security rules will be implemented to ensure data privacy and restrict unauthorized access.

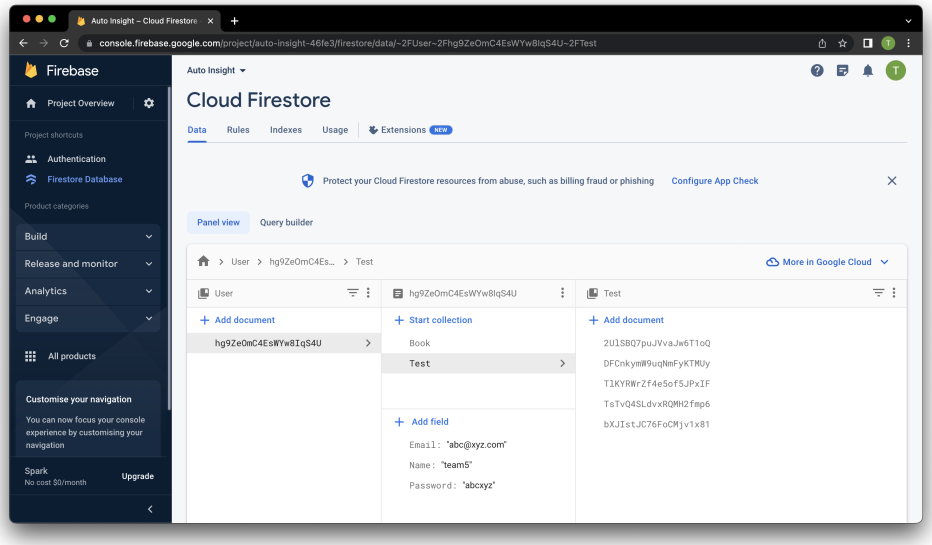

Figure 4.1: Test drive and book details

In figure[4.1] we can see how the booking details as well as the test drive details stored in the firebase database. From the database the data is also pulled to be displayed on the previous booking section in the app.

Here in figure[4.2] the authentication details of the user is stored in the firebase database such as the username, password, first name and last name. This ensures privacy

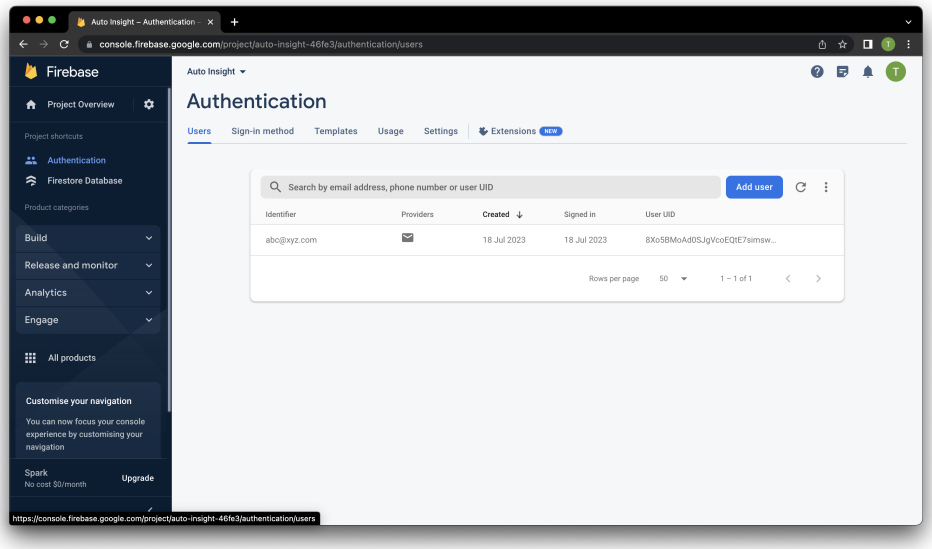

Figure 4.2: User authentication details

and protection of data of one user from the other that is only the user can view his/her previous booking details and test drive details.

4.2 UI Design Implementation

- Asset Export: Export all design assets (icons, images, etc.) from Figma in the required formats and resolutions for app development.

Figure 4.3: Travel icons

- UI Components: Break down the Figma design into reusable UI components, ensuring consistency and facilitating development.

- Responsive Design: Ensure that the UI design is responsive and adapts to various screen sizes and orientations, catering to both smartphones and tablets.

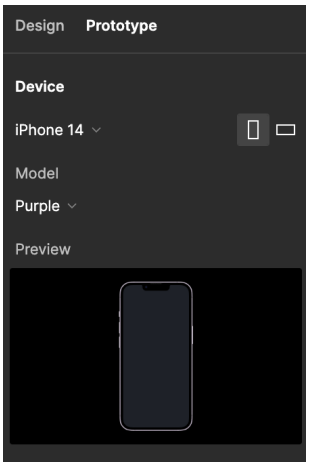

Figure 4.4: Prototyping in Figma

4.3 Development (Flutter with Android Studio)

- Firebase Project Setup: Create a new Firebase project, configure necessary services (Authentication, Firestore), and obtain API keys for integration with the Flutter app.

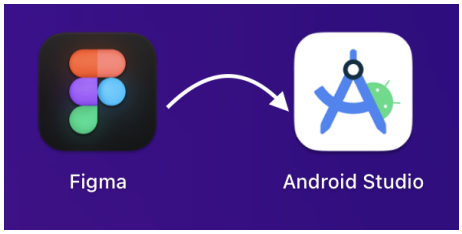

Figure 4.5: Figma to flutter

- Firebase Database: Integrate Firebase Authentication with the Flutter app to enable user registration, login, password and view functionalities. Set up the database structure in Firebase Firestore to store user data, car test drive bookings and car booking details.

4.4 Maintenance and Updates

- Monitoring and Bug Fixing: Monitor the app's performance, user feedback, and crash reports. Address and fix any reported bugs or issues promptly.
- Continuous Improvement: Continuously gather user feedback to identify areas for improvement and introduce new features or enhancements to enhance the app's functionality and user experience.

4.5 Conclusion

By following this detailed system implementation plan, the AutoInsight app will be developed with the Figma-designed UI, integrated with Firebase for user authentication, database storage, and real-time data updates using Flutter and Android Studio. The app will provide users with a seamless car research and buying experience, empowering them with comprehensive car information and personalized features for test drive bookings and car customizations.

Chapter 5

System Design

5.1 Architecture Diagram

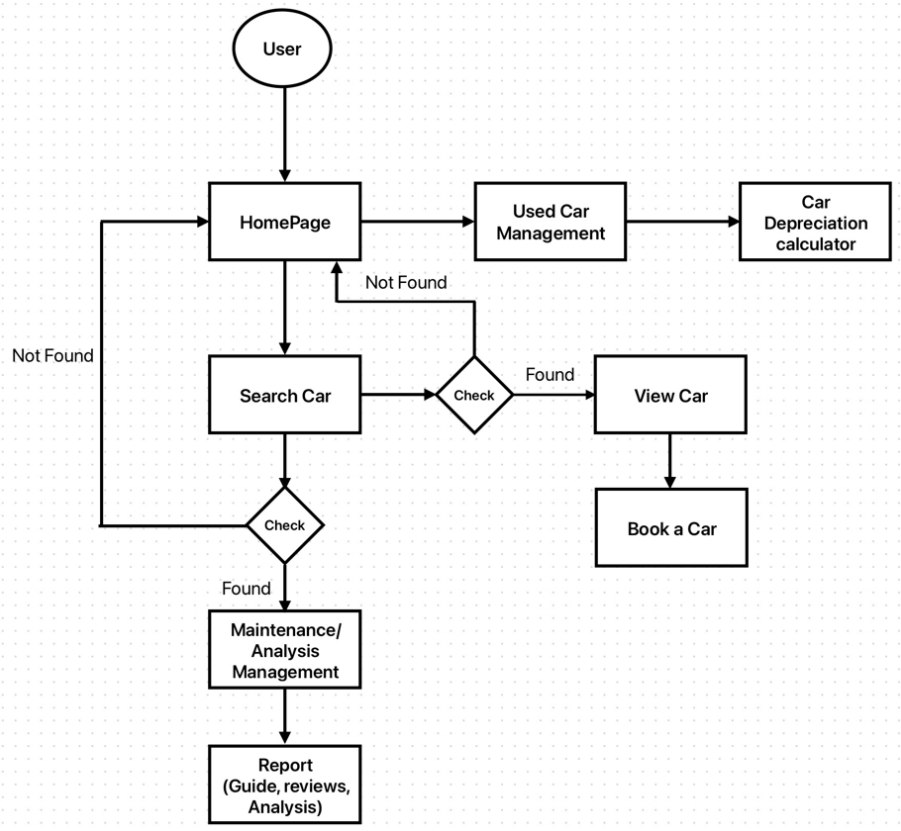

Figure 5.1: Architecture Diagram

5.2 Sequence Diagram

5.2.1 Overall

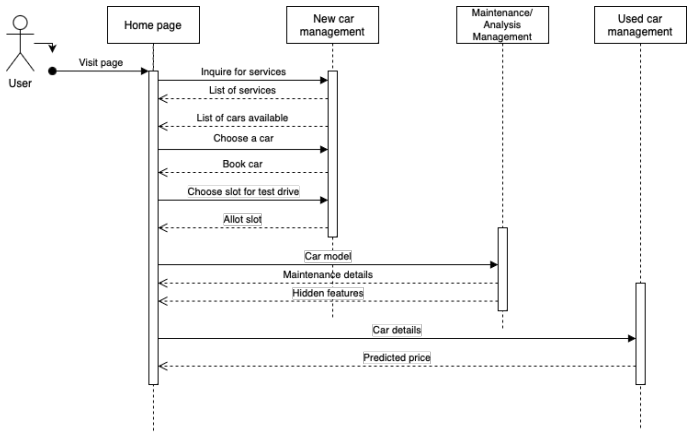

Figure 5.2: Sequence Diagram

5.2.2 Maintenance

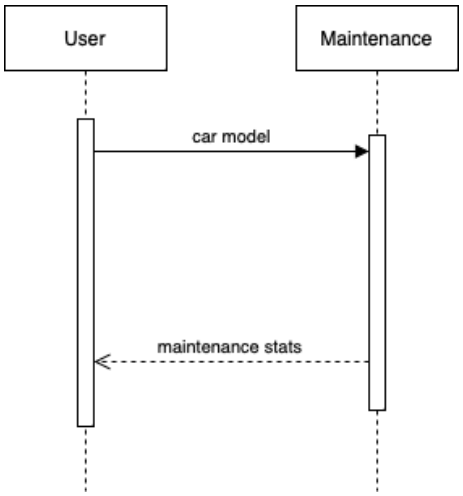

Figure 5.3: Car maintenance

Here we specify the model of the car that we need a maintenance guide on and then the app retrieves the data from the database and displays it. Here the input is the model and the output is the maintenance stats.

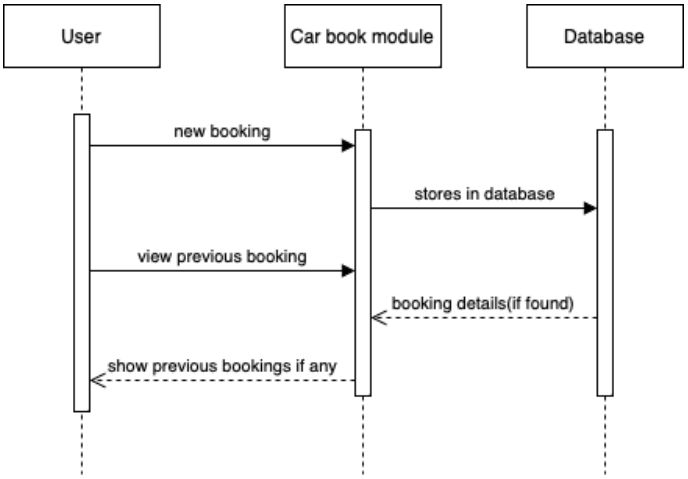

Figure 5.4: Booking sequence

5.2.3 Car booking

In the module of the app we have to options. The first one is to view the previous bookings made and the second one is to make a new booking. When previous booking option is selected the app retrieves the history from the database from the users profile and displays it. When new booking tab is selected the user is asked to select the car he/she wishes to have the car delivered and select a preferred date according to their convenience to carry out the delivery of the car. The data is then stored in the database.

5.2.4 Test drive

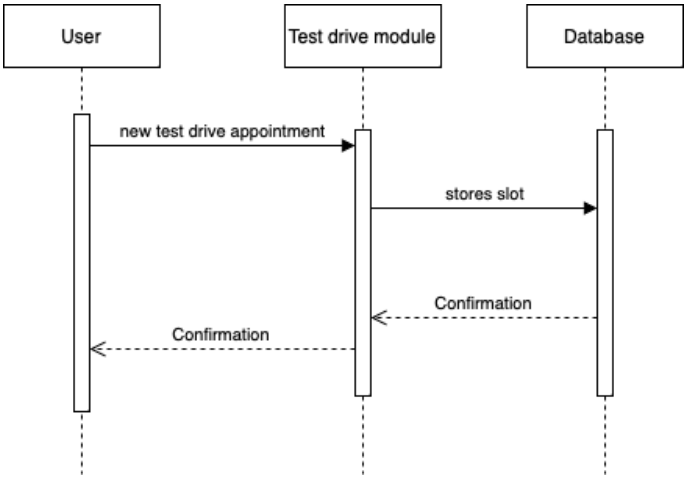

Figure 5.5: Test drive sequence

In the test drive module of the app the user is asked to enter the date and preferred

time as input for the test drive.This data is stored in the database and then the user is directed to a confirmation page.

5.2.5 Used car

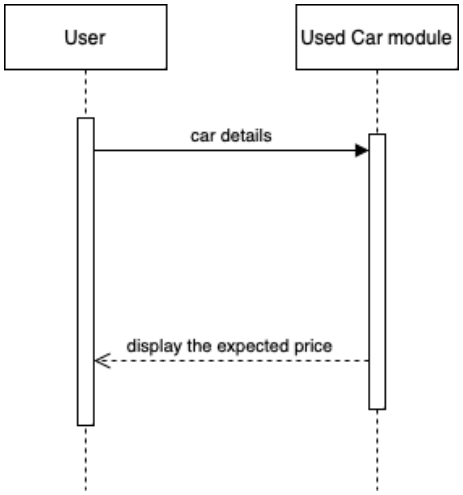

Figure 5.6: Used car sequence

In this module the car details are sent as the input. The input includes the car model, manufactured date, registeration year, odometer reading and then these data are inputed into the machine learning model to generate an expected price as output of the used car.

5.3 Modulewise Diagram

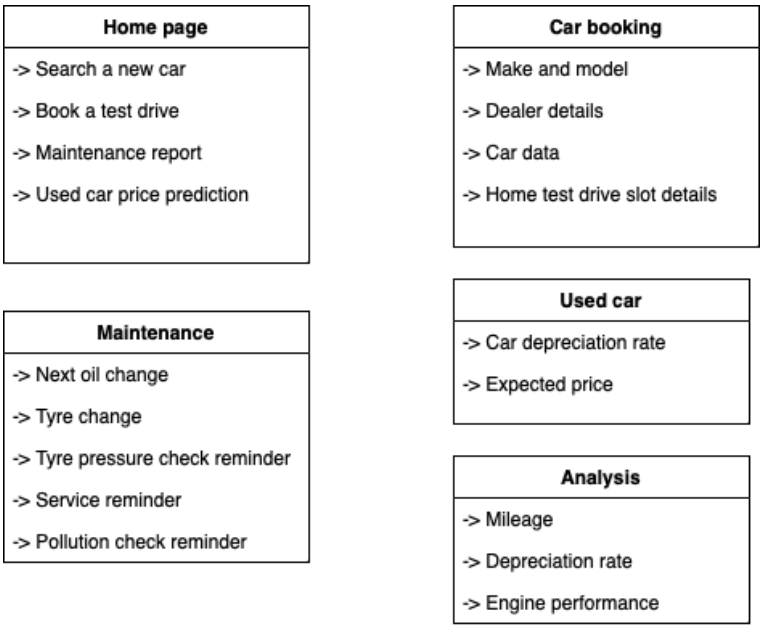

Figure 5.7: Modulewise Diagram

5.4 Design of the System

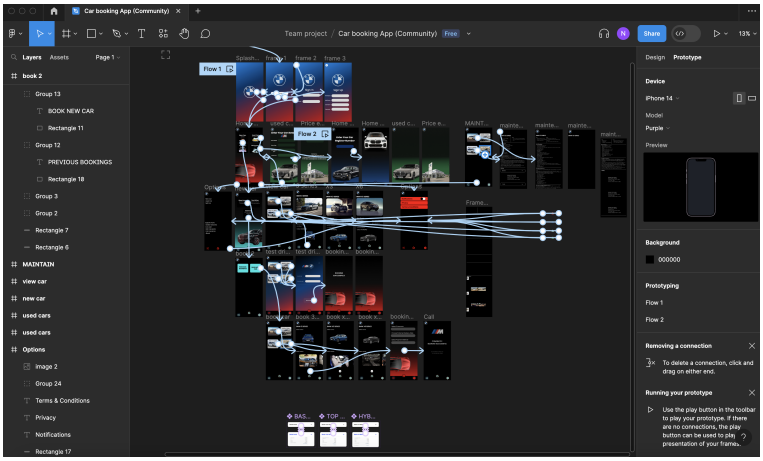

Figure 5.8: UI Design

Chapter 6

RESULTS AND DISCUSSIONS

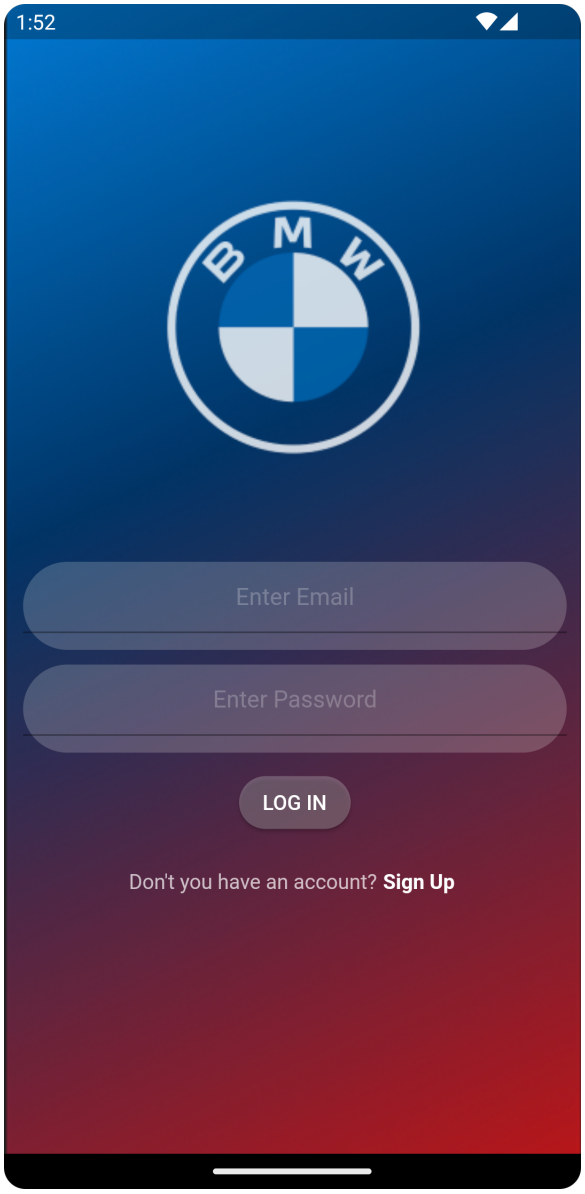

Figure 6.1: Login Page

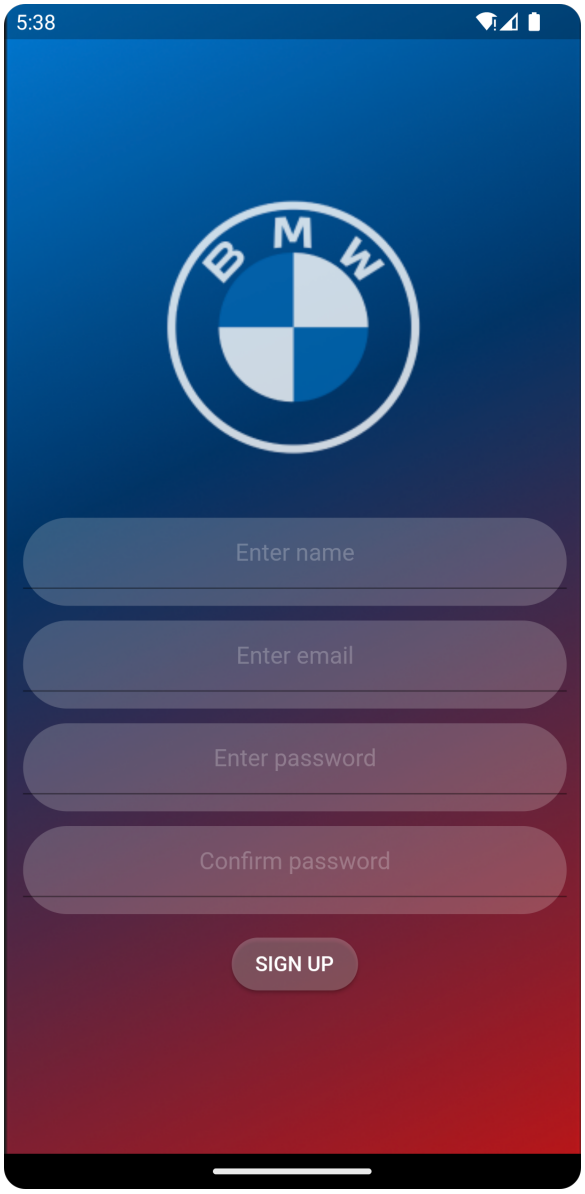

Figure 6.2: Sign-up page

Figure 6.3: Main menu

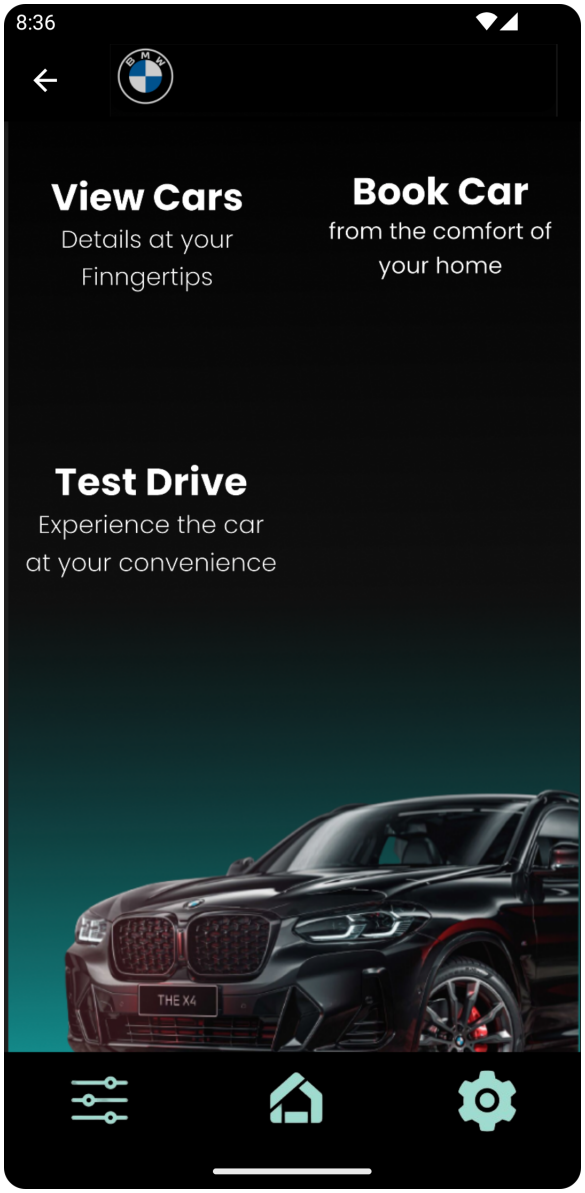

Figure 6.4: New car options

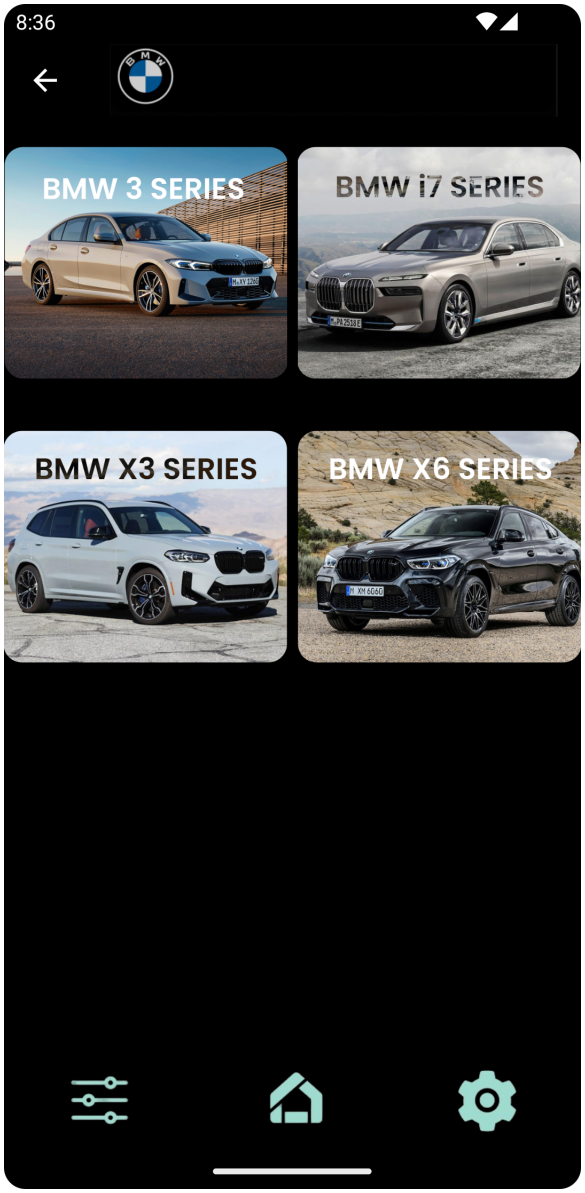

Figure 6.5: View Cars

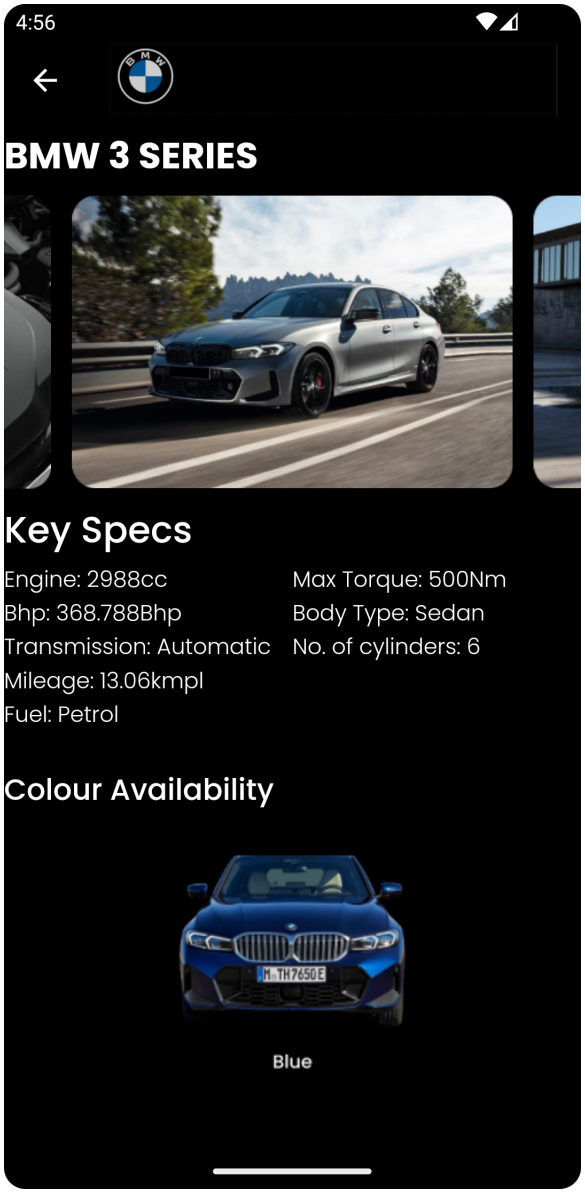

Figure 6.6: Car Details

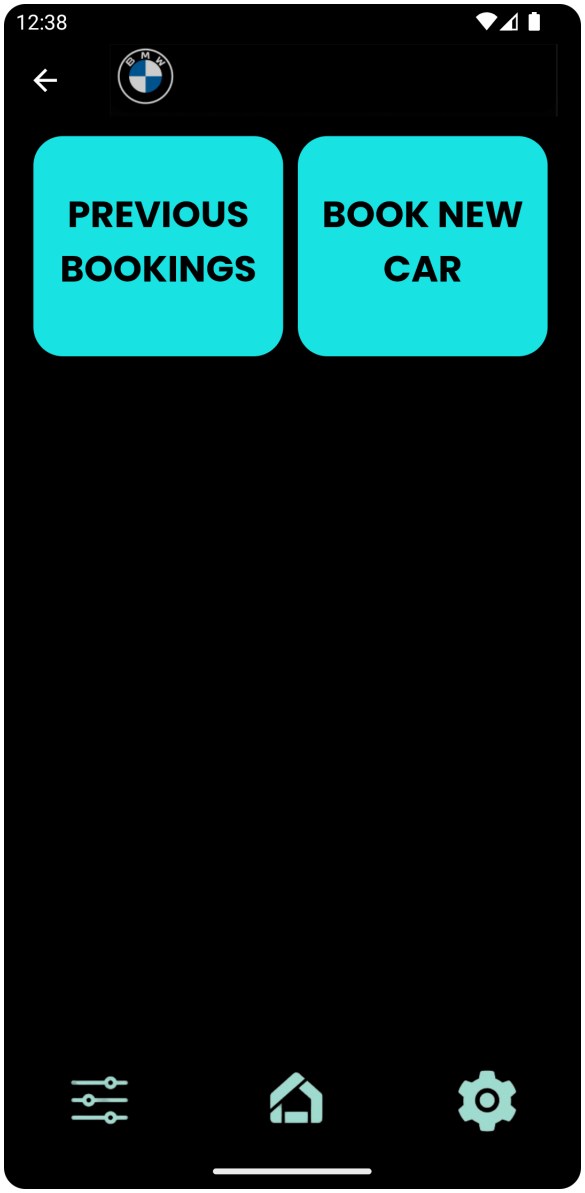

Figure 6.7: Car booking options

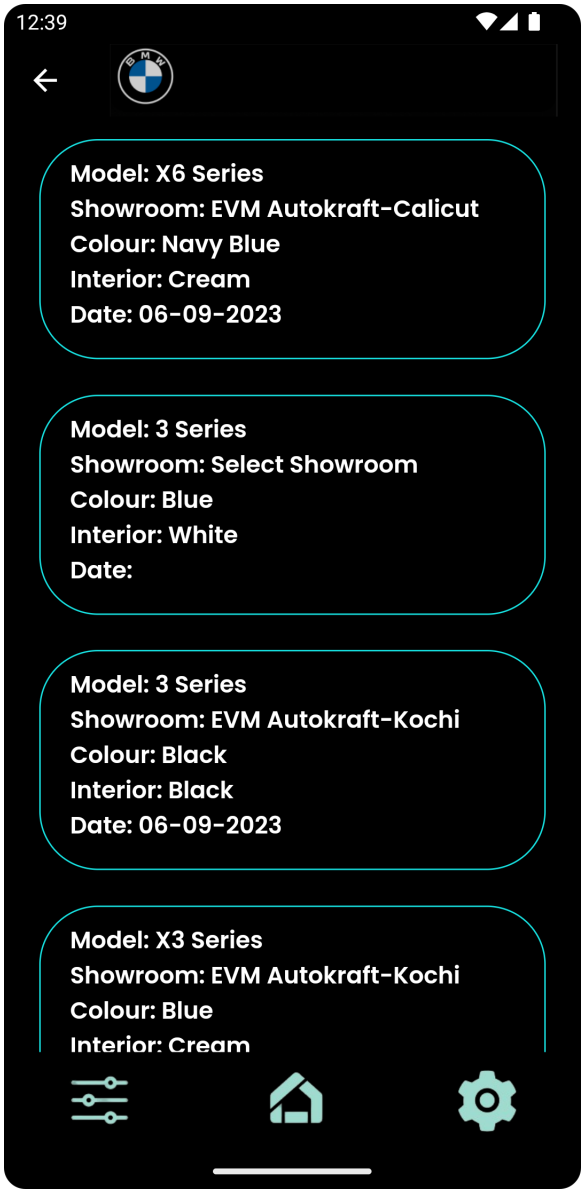

Figure 6.8: Previous Bookings

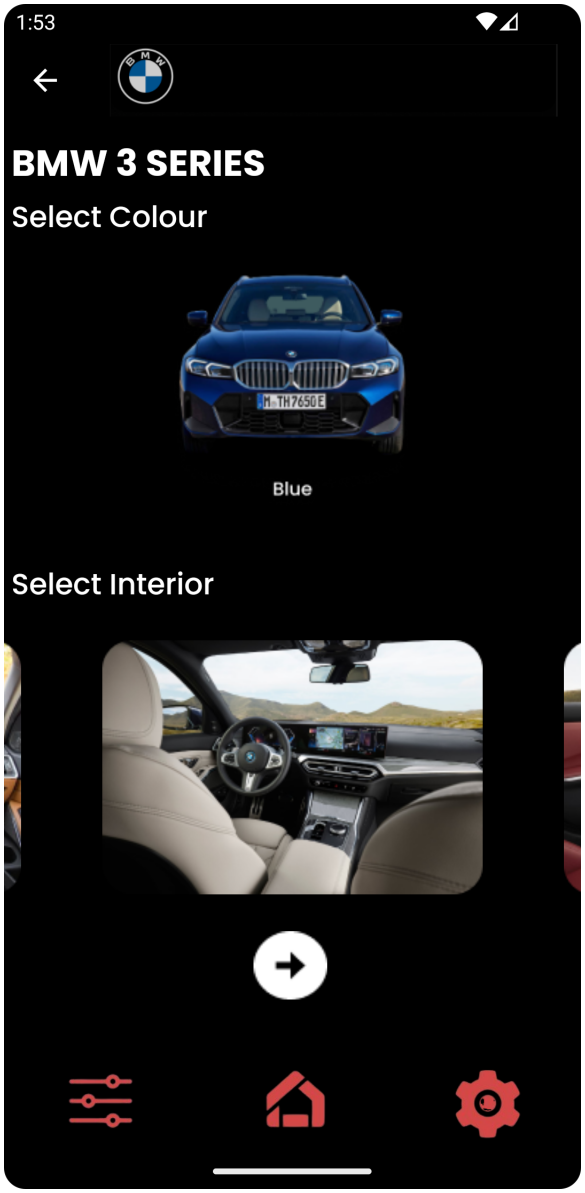

Figure 6.9: Book New Car (1/2)

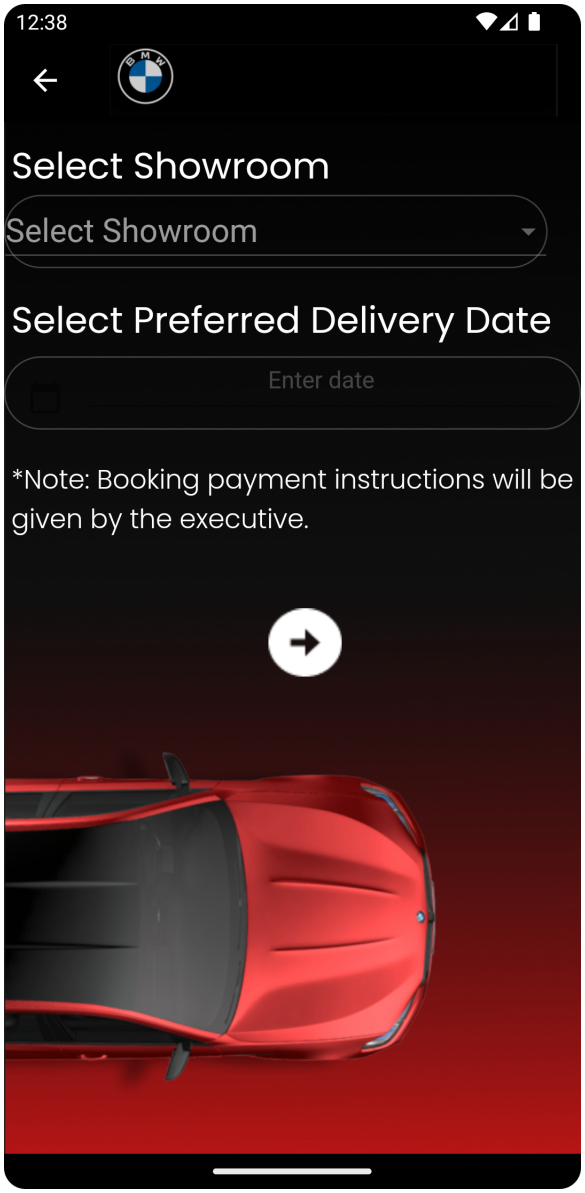

Figure 6.10: Book New Car (2/2)

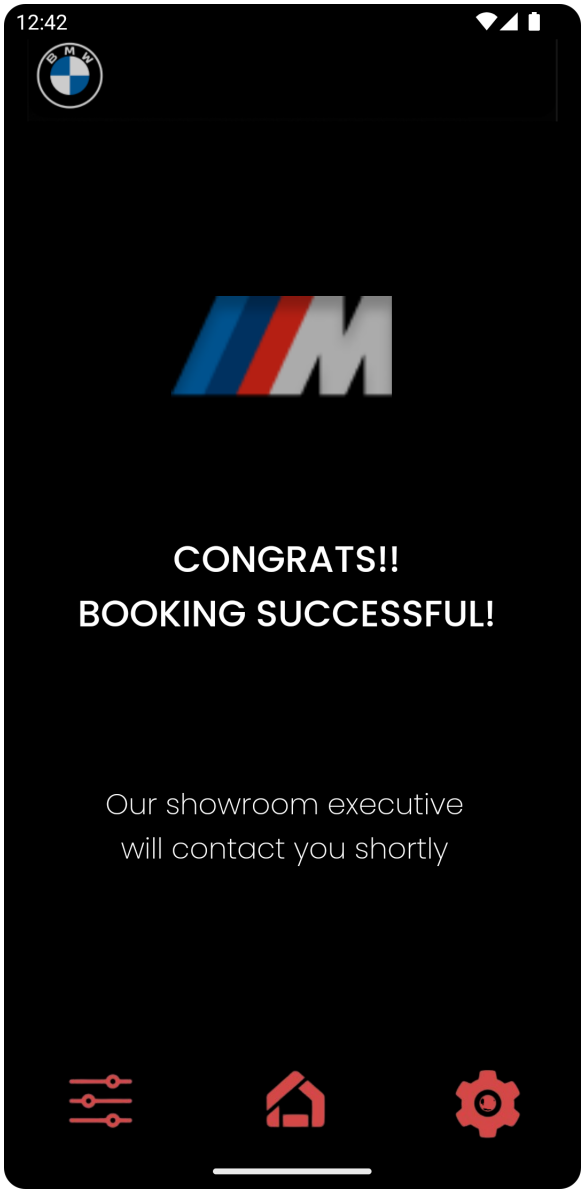

Figure 6.11: Booking Successful Page

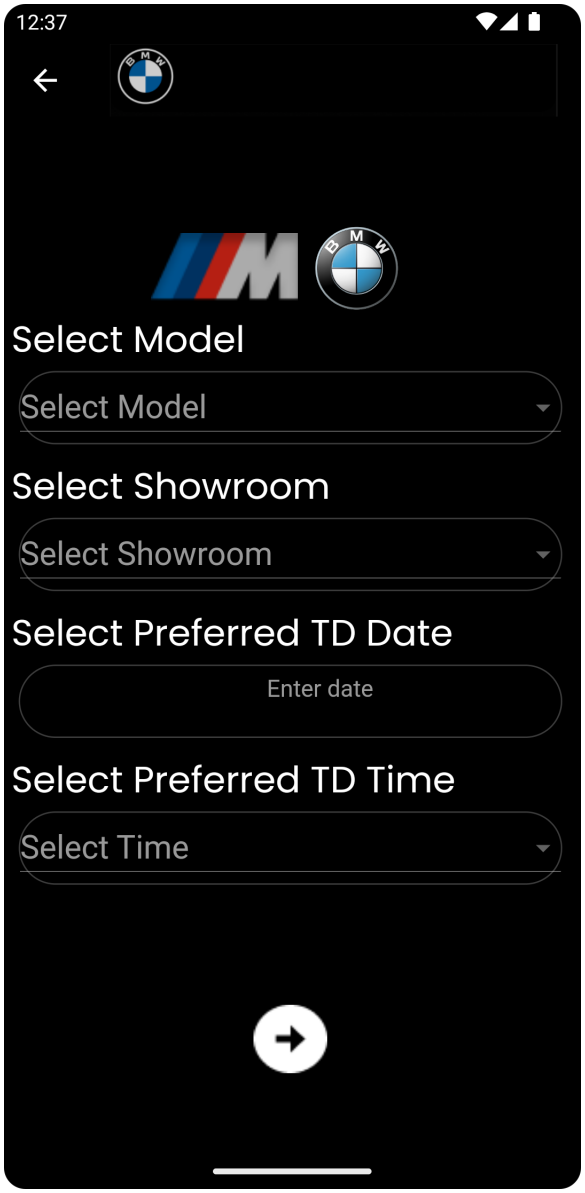

Figure 6.12: Test Drive options

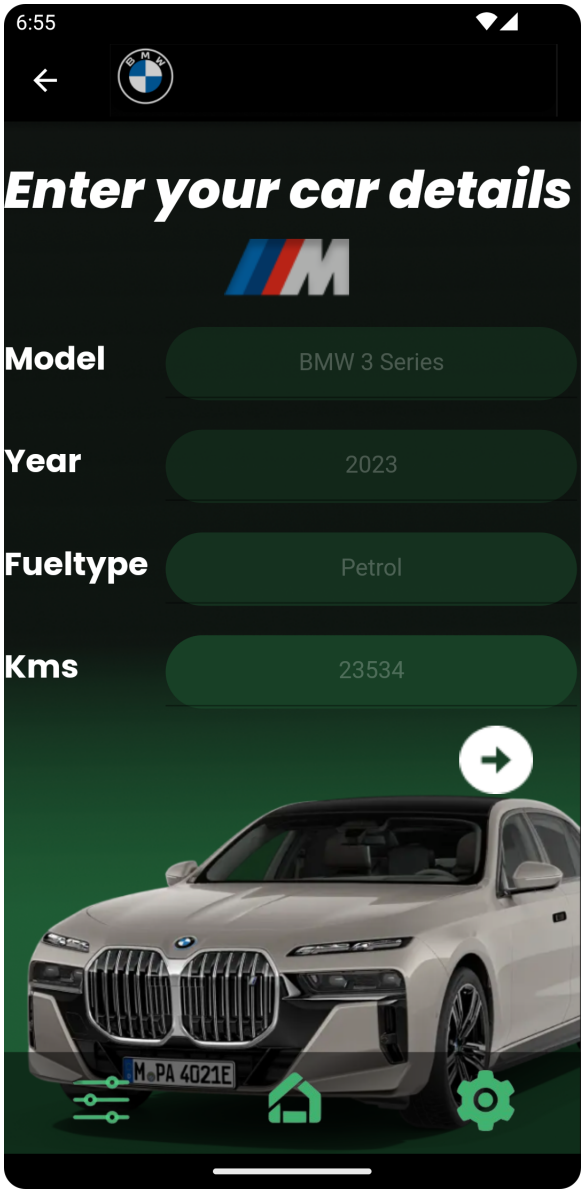

Figure 6.13: Used Car details

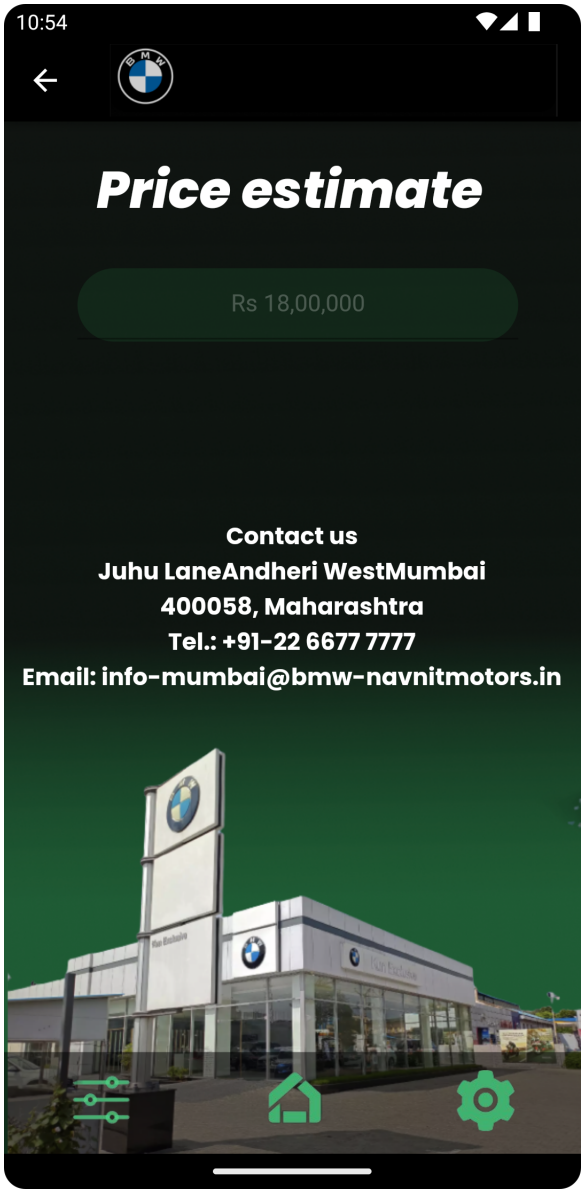

Figure 6.14: Price Predictor page

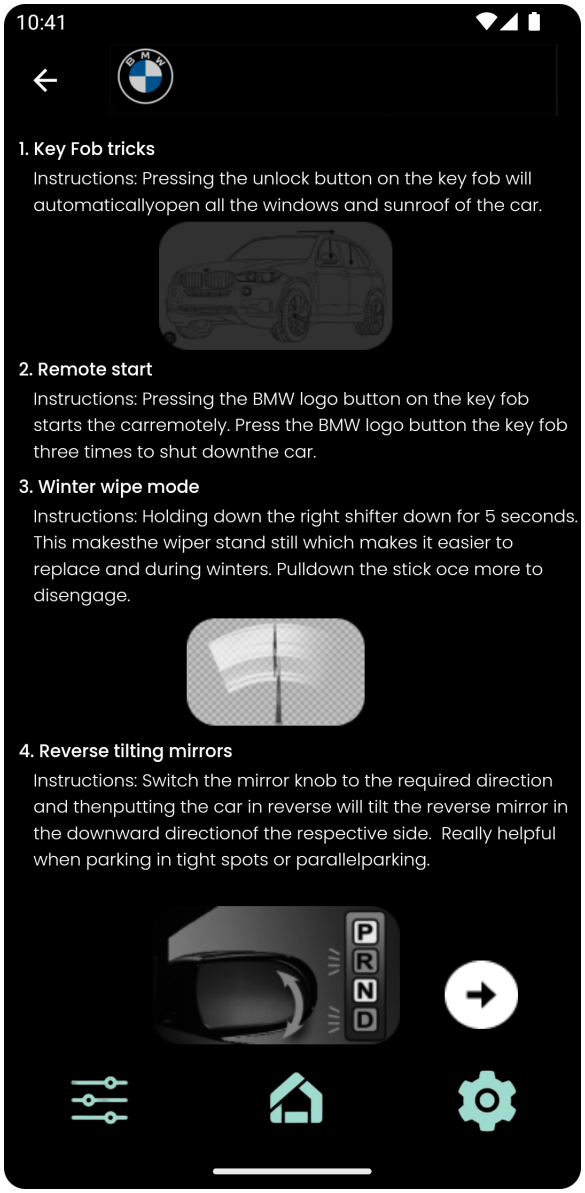

Figure 6.15: Hidden features (1/2)

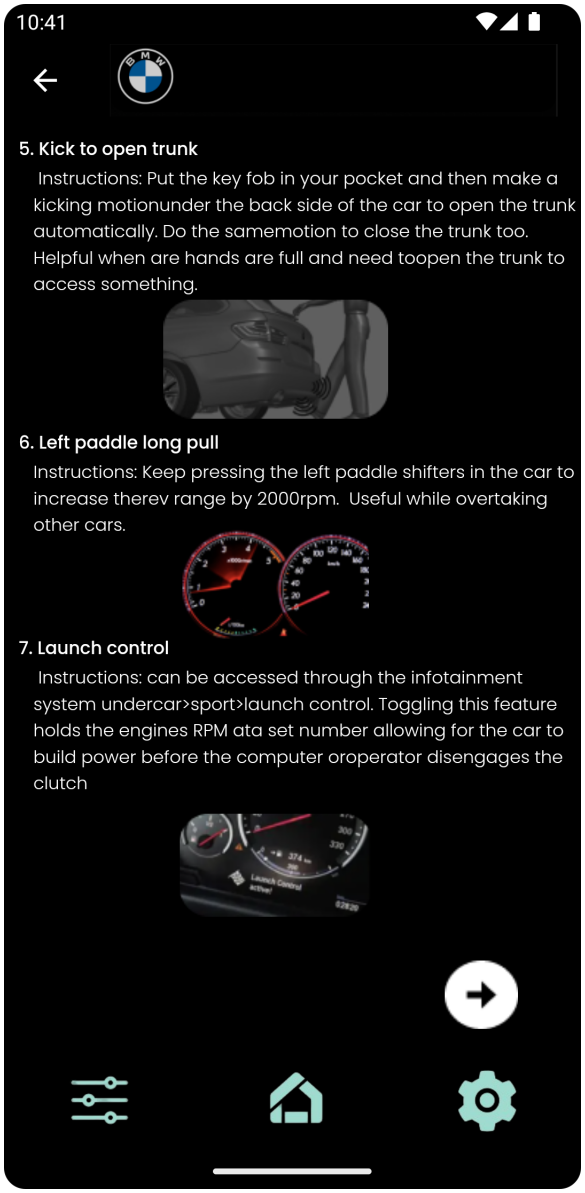

Figure 6.16: Hidden features (2/2)

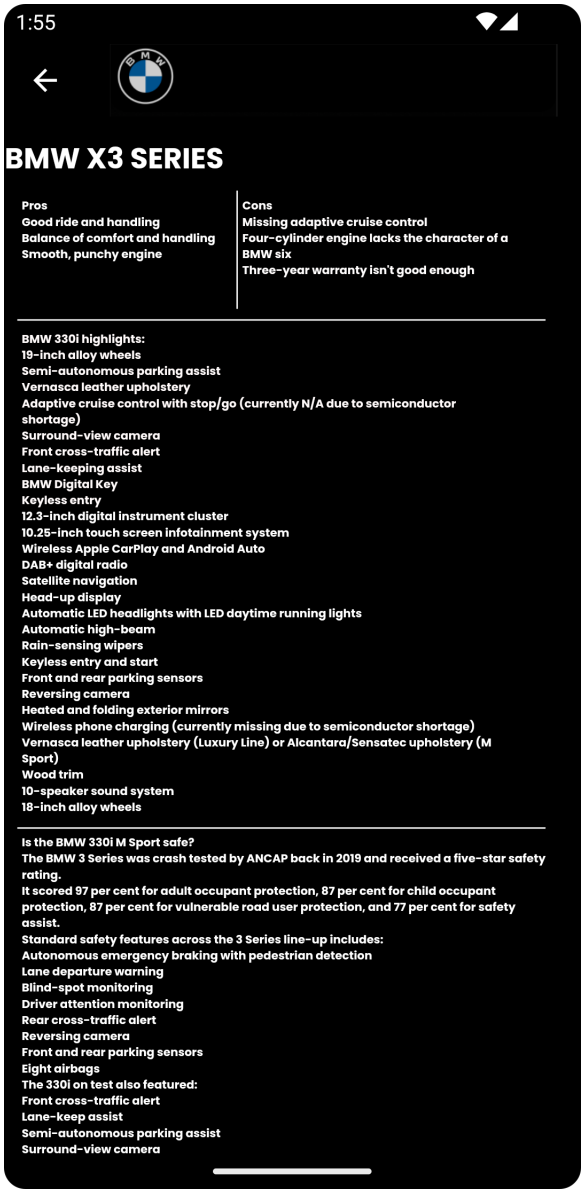

Figure 6.17: Maintenance guide (1/2)

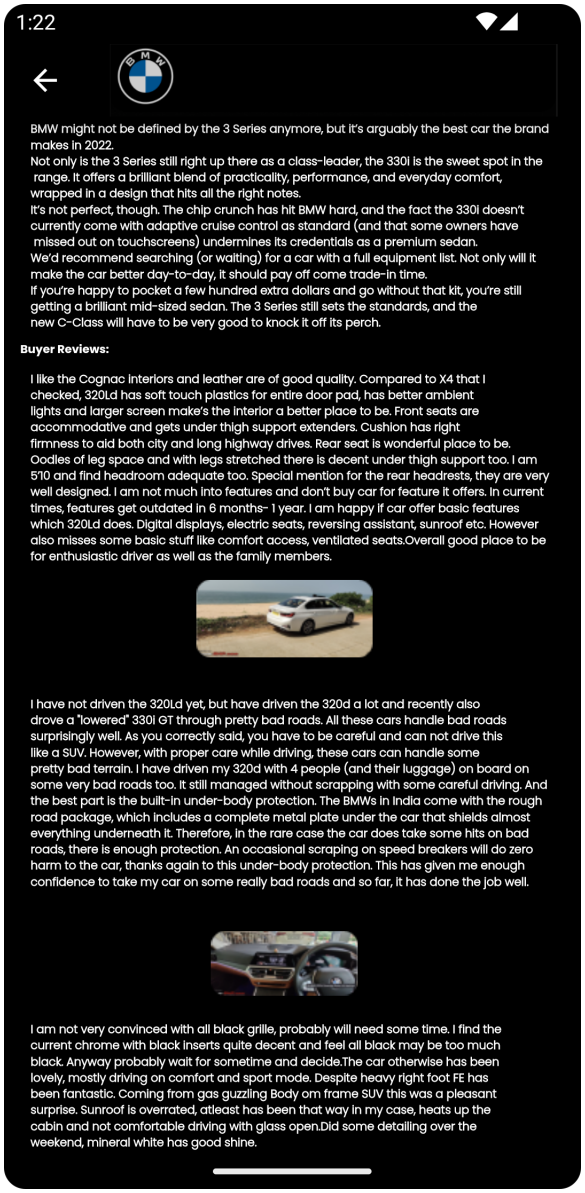

Figure 6.18: Maintenance guide (2/2)

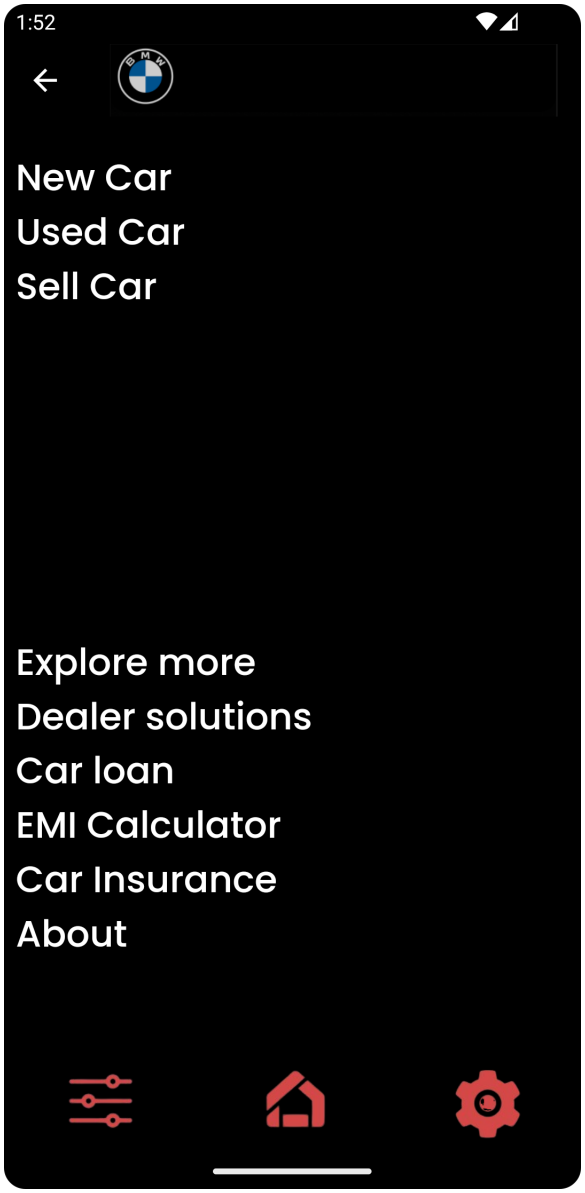

Figure 6.19: Contents Page

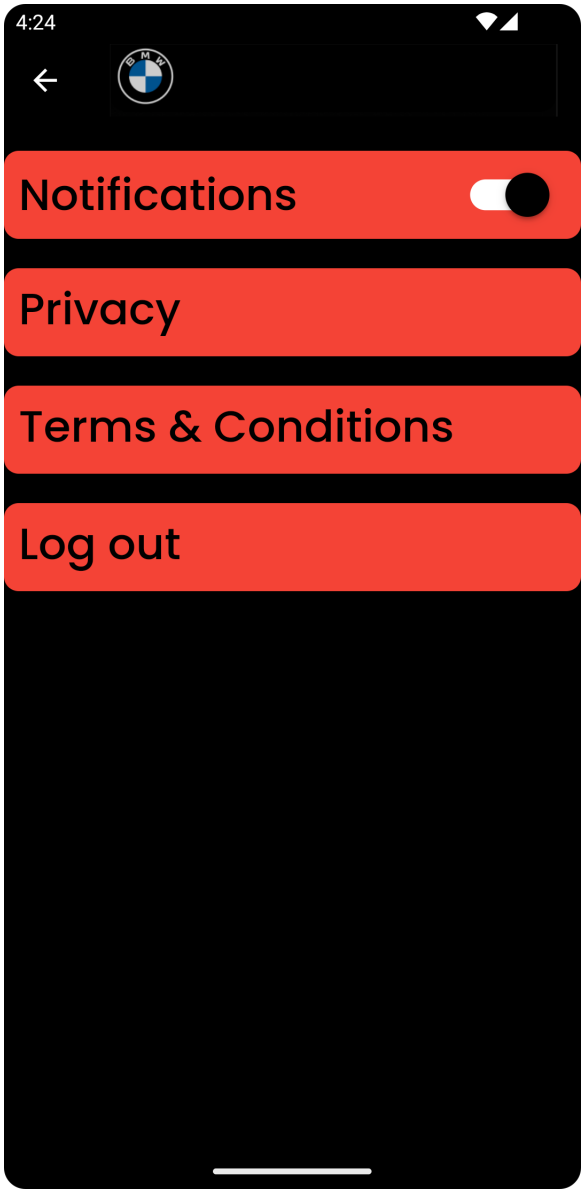

Figure 6.20: Settings Page

Chapter 7

Conclusion and Future Scope

7.1 Challenges

Implementing integration with vehicle systems, such as protocols and interfaces, can be complex. Copyright infringement should be avoided by obtaining necessary licenses and permissions. Adapting to a fast-paced market requires continuous innovation. Difficulty in data collection may arise due to limited access, privacy concerns, and technical limitations. Real-time data processing requires robust infrastructure and algorithms to handle the volume, velocity, and variety of data.

7.2 Conclusion

Based on our analysis, the car analysis app proves to be an invaluable tool for car enthusiasts and buyers alike. Its comprehensive features, such as providing detailed specifications, pricing information, and expert reviews, offer users a wealth of knowledge to make informed decisions. The app's intuitive user interface and interactive elements ensure a seamless user experience. In conclusion, the car analysis app revolutionises the way people research, evaluate, and purchase cars. Its comprehensive features, user-friendly interface, accurate predictions, and community-driven platform make it an essential tool for anyone in the market for a car. With this app, users can confidently make informed decisions and embark on their car-buying journey with ease.

7.3 Scope of Future Work

Future work for the car analysis app includes enhancing data visualization, connecting with IoT and connected cars, integrating with e-commerce platforms, implementing machine learning for predictive analysis.

References

[1] Chaganti Sandeep Reddy, Dr. Preeti Savant, A Survey on Car Service Slot Booking System, IJRASET Journal For Research in Applied Science and Engineering Technology, ijraset.2022.41448

[2] Rachel Harrison, Derek Flood and David Duce. Usability of mobile applications: literature review and rationale for a new usability model, Journel of Interaction Science, Article number: 1 (2013)

[3] Shreya A. Bhagat, Sakshi G. Dudhalkar, Prathmesh D. Kelapure, Aniket S. Kokare and Prof. Sudesh A. Bachwani, Review on Mobile Application Development Based on Flutter Platform, IJRASET Journal For Research in Applied Science and Engineering Technology, ijraset.2022.39920

[4] Prachi R. Saraf, Sakshi M. Jadhao, Saurabh J. Wanjari, Shital G. Kolwate, Prof. Ankush D. Patil, A Review on Firebase (Backend as A Service) for Mobile Application Development, IJRASET Journal For Research in Applied Science and Engineering Technology, ijraset.2022.39958

Appendix A: Sample Code

SIGNUP:

```
import 'package:flutter/material.dart';

import 'comp/signupBody.dart';

class signup extends StatelessWidget {
  const signup({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body:  signupbody(),
    );
  }
}

import 'package:autoinsight/screens/Signin/signin.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:email_validator/email_validator.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';

class signupbody extends StatefulWidget {
  const signupbody({super.key});

  @override
  State<signupbody> createState() => _signupbodyState();
}

class _signupbodyState extends State<signupbody> {

  final TextEditingController emailController = TextEditingController();
  final TextEditingController passwordController = TextEditingController();
  final TextEditingController nameController = TextEditingController();
  final TextEditingController confirmpwdController = TextEditingController();

  @override
  Widget build(BuildContext context) {
    return  Scaffold(
      body: Stack(
        children: [
          Container(
            constraints: const BoxConstraints.expand(),
            decoration: const BoxDecoration(
              image: DecorationImage(
                image: AssetImage("assets/multilogo.png"),
                fit: BoxFit.cover,
              ),
            ),
          ),
          Container(margin: EdgeInsets.only(top: 350,left: 13),
            height: 60,
            width: 370,
            child: TextField(
              controller: nameController,
              textAlign: TextAlign.center,
              decoration: InputDecoration(hintText: 'Enter name',hintStyle: TextStyle(color:
Colors.white24)),
              cursorColor: Colors.white,
            ),
```

```
          decoration: BoxDecoration(
            borderRadius: BorderRadius.circular(40),
            color: Color(0XFFFAF9F6).withOpacity(0.2),
          ),
        ),
        Container(margin: EdgeInsets.only(top: 420,left: 13),
          height: 60,
          width: 370,
          child: TextField(
            controller: emailController,
            keyboardType: TextInputType.emailAddress,
            textAlign: TextAlign.center,
            decoration: InputDecoration(hintText: 'Enter email',hintStyle: TextStyle(color:
Colors.white24)),
            cursorColor: Colors.white,
          ),
          decoration: BoxDecoration(
            borderRadius: BorderRadius.circular(40),
            color: Color(0XFFFAF9F6).withOpacity(0.2),
          ),
        ),
        Container(margin: EdgeInsets.only(top: 490,left: 13),
          height: 60,
          width: 370,
          child: TextField(
            controller: passwordController,
            textAlign: TextAlign.center,
            decoration: InputDecoration(hintText: 'Enter password',hintStyle: TextStyle(color:
Colors.white24)),
            cursorColor: Colors.white,
          ),
          decoration: BoxDecoration(
            borderRadius: BorderRadius.circular(40),
            color: Color(0XFFFAF9F6).withOpacity(0.2),
          ),
        ),
        Container(margin: EdgeInsets.only(top: 560,left: 13),
          height: 60,
          width: 370,
          child: TextField(
            controller: confirmpwdController,
            obscureText: true,
            textAlign: TextAlign.center,
            decoration: InputDecoration(hintText: 'Confirm password',hintStyle: TextStyle(color:
Colors.white24)),
            cursorColor: Colors.white,
          ),
          decoration: BoxDecoration(
            borderRadius: BorderRadius.circular(40),
            color: Color(0XFFFAF9F6).withOpacity(0.2),
          ),
        ),

        Positioned(top: 630, left: 155,
            child: ElevatedButton(onPressed: () async{

              if (passwordController.text == '') {
                ScaffoldMessenger.of(context).showSnackBar(SnackBar(
                  content: Text('Please enter a password.'), ));
                print("No password entered.");
```

```
            return;
          }
          else if (passwordController.text != confirmpwdController.text) {
            ScaffoldMessenger.of(context).showSnackBar(SnackBar(
                content: Text('Password does not match.'), ));
            print("Password do not match");
            return;
          }
          else if (emailController.text == null||!EmailValidator.validate(emailController.text)) {
            ScaffoldMessenger.of(context).showSnackBar(SnackBar(
              content: Text('Please enter a valid mail.'), ));
            print("Not a valid email.");
            return;
          }

          await FirebaseAuth.instance.createUserWithEmailAndPassword(
            email: emailController.text,
            password: passwordController.text,
          );
          CollectionReference collref = FirebaseFirestore.instance.collection('User');
          collref.add({
            'Name': nameController.text,
            'Email': emailController.text,
            'Password': passwordController.text,
          });
          Navigator.push(
            context,
            MaterialPageRoute(builder: (context) => const signin()),
          );
        },
            style: ElevatedButton.styleFrom(
                primary: Colors.white24,
                shape: RoundedRectangleBorder(
                    borderRadius: BorderRadius.circular(40)
                )
            ),
            child: const Text('SIGN UP'))
      ),
    ],
  ),
 );
 }
}
```

SIGNIN:

```
import 'package:autoinsight/screens/Signin/comp/signinBody.dart';
import 'package:flutter/material.dart';

class signin extends StatefulWidget {
  const signin({super.key});

  @override
  State<signin> createState() => _signinState();
}

class _signinState extends State<signin> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: signinbody(),
```

```
    );
  }
}

import 'package:autoinsight/screens/FirstScreen/FirstScreen.dart';
import 'package:autoinsight/screens/Signup/signup.dart';
import 'package:email_validator/email_validator.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';

class signinbody extends StatefulWidget {
  const signinbody({super.key});

  @override
  State<signinbody> createState() => _signinbodyState();
}


class _signinbodyState extends State<signinbody> {

  final _emailcontroller = TextEditingController();
  final  _passwordcontroller = TextEditingController();

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: Stack(
        children: [

      Container(
            constraints: const BoxConstraints.expand(),
            decoration: const BoxDecoration(
              image: DecorationImage(
                image: AssetImage("assets/multilogo.png"),
                fit: BoxFit.cover,
              ),
            ),
          ),
            Container(margin: EdgeInsets.only(top: 380,left: 13),
            height: 60,
            width: 370,
            child:TextFormField(
              textAlign: TextAlign.center,
              controller: _emailcontroller,
              keyboardType: TextInputType.emailAddress,
              validator: (value){
                if (value == null || value.isEmpty){
                  return 'Please enter your Email';}
                return null;
              },
              decoration: InputDecoration(hintText: 'Enter Email',hintStyle: TextStyle(color:
Colors.white24)),
              cursorColor: Colors.white,
            ),
            decoration: BoxDecoration(
              borderRadius: BorderRadius.circular(40),
              color: Color(0XFFFAF9F6).withOpacity(0.2),
            ),
          ),
```

```
        Container(margin: EdgeInsets.only(top: 450,left: 13),
          height: 60,
          width: 370,
          child: TextFormField(
            textAlign: TextAlign.center,
            controller: _passwordcontroller,
            obscureText: true,
            validator: (value){
              if (value == null || value.isEmpty){
                return 'Please enter your Password';}
              return null;
            },
            decoration: InputDecoration(hintText: 'Enter Password',hintStyle: TextStyle(color:
Colors.white24)),
            cursorColor: Colors.white,
          ),
          decoration: BoxDecoration(
            borderRadius: BorderRadius.circular(40),
            color: Color(0XFFFAF9F6).withOpacity(0.2),
          ),
        ),
        Positioned(top: 520, left: 160,
          child: ElevatedButton(onPressed: () async{
            if (_emailcontroller.text == null||!EmailValidator.validate(_emailcontroller.text)) {
              ScaffoldMessenger.of(context).showSnackBar(SnackBar(
                content: Text('Please enter a valid mail.'), ));
              print("Not a valid email.");
              return;
            }
            if (_passwordcontroller.text == '') {
              ScaffoldMessenger.of(context).showSnackBar(SnackBar(
                content: Text('Please enter a password.'), ));
              print("No password entered.");
              return;
            }

            await FirebaseAuth.instance
                .signInWithEmailAndPassword(
                email: _emailcontroller.text,
                password: _passwordcontroller.text)
                .then((value) async {
              final user = FirebaseAuth.instance.currentUser;
              if (user != null) {
                Navigator.pushReplacement(
                  context,
                  MaterialPageRoute(builder: (context) => FirstScreen()),
                );
              }
            }
            );
          },
          style: ElevatedButton.styleFrom(
            primary: Colors.white24,
            shape: RoundedRectangleBorder(
              borderRadius: BorderRadius.circular(40)
            )
          ),
              child: const Text('LOG IN'))
        ),
        Positioned(top: 590, left: 85,
```

```
            child: Row(mainAxisAlignment: MainAxisAlignment.center,
           children: [
            const Text("Don't you have an account? ",
            style: TextStyle(color: Colors.white70)),
            GestureDetector(
              onTap: () {
                Navigator.push(
                  context,
                  MaterialPageRoute(builder: (context) => const signup()),
                );
              },
              child: const Text("Sign Up",
              style: TextStyle(color: Colors.white, fontWeight: FontWeight.bold),
              ),
            )
          ],
        )
        ),
      ],
    ),
   );
  }
}
```

BOOK CAR:
CarModelChoice:

```
import 'package:autoinsight/screens/CarBook/ThreeSeriesBook/comp/
ThreeSeriesBookBody.dart';
import 'package:flutter/material.dart';

class ThreeSeriesBook extends StatelessWidget {
  const ThreeSeriesBook({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: ThreeSeriesBookBody(),
    );
  }
}

import 'package:autoinsight/screens/CarBook/BookingDetails/BookingDetails.dart';
import 'package:autoinsight/screens/CarBook/BookingDetails/Pass.dart';
import 'package:autoinsight/screens/Contents/contents.dart';
import 'package:autoinsight/screens/FirstScreen/FirstScreen.dart';
import 'package:carousel_slider/carousel_slider.dart';
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';

class ThreeSeriesBookBody extends StatefulWidget {
   ThreeSeriesBookBody({super.key});

  @override
  State<ThreeSeriesBookBody> createState() => _ThreeSeriesBookBodyState();
}

class _ThreeSeriesBookBodyState extends State<ThreeSeriesBookBody> {
  List<String> colour = ['Blue','White','Grey','Black'];
  List<String> interior = ['White','Red','Black','Brown'];
```

```
int _currentColour = 0;
int _currentInt = 0;

Pass obj = new Pass();
@override
Widget build(BuildContext context) {
  return Center(
      child: Scaffold(
      appBar: AppBar(
      title: Image.asset("assets/topbar.png", fit: BoxFit.cover),
            backgroundColor: Colors.black,

      ),
        body: Stack (
            children: [
        Container(
        constraints: const BoxConstraints.expand(),
        decoration: const BoxDecoration(
            color: Colors.black
        ),
          ),
            Container(margin: EdgeInsets.only(top: 10, left: 5),
              child: const Text("BMW 3 SERIES",
                style: TextStyle(
                    color: Colors.white,fontSize: 25,
                    fontFamily: 'PoppinsBold'
                ),
              ),
            ),
            //model text
            Container(margin: EdgeInsets.only(top: 50, left: 5),
              child: const Text("Select Colour",
                style: TextStyle(
                    color: Colors.white,fontSize: 20,
                    fontFamily: 'PoppinsMed'
                ),
              ),
            ),
            //text colour
            Container(margin: EdgeInsets.only(top: 80),
              child: CarouselSlider(
                options: CarouselOptions(height: 200.0,
                  onPageChanged: (index, reason) {
                    _currentColour = index;
                    setState(() {});
                  },
                ),
                items: [1,2,3,4].map((i) {
                  return Builder(
                    builder: (BuildContext context) {
                      return Container(
                        width: 300,//MediaQuery.of(context).size.width,
                        margin: const EdgeInsets.symmetric(horizontal: 5.0),
                        decoration: const BoxDecoration(
                            color: Colors.black
                        ),
                        child: Image.asset("assets/3sBook$i.png"),
                      );
                    },
                  );
```

```
    }).toList(),
  ),
),
//carousel colour
Container(margin: EdgeInsets.only(top: 300, left: 5),
  child: const Text("Select Interior",
    style: TextStyle(
        color: Colors.white,fontSize: 20,
        fontFamily: 'PoppinsMed'
    ),
  ),
),
//text interior
Container(margin: EdgeInsets.only(top: 340),
  child: CarouselSlider(
      options: CarouselOptions(height: 200.0,
        onPageChanged: (index, reason) {
          _currentInt = index;
          setState(() {});
        },
    ),
    items: [1,2,3,4].map((i) {
      return Builder(
        builder: (BuildContext context) {
          return Container(
            width: 300,//MediaQuery.of(context).size.width,
            margin: const EdgeInsets.symmetric(horizontal: 5.0),
            decoration: const BoxDecoration(
                color: Colors.black
            ),
            child: Image.asset("assets/3sInt$i.png"),
          );
        },
      );
    }).toList(),
  ),
),
//carousel interior
Positioned(top: 550, left: 170,
  child: InkWell(
    onTap: (){
      obj.recieveCI(colour[_currentColour], interior[_currentInt]);
      obj.receiveModel('3 Series');
      Navigator.push(
        context,
        MaterialPageRoute(builder: (context) => const BookingDetails()),
      );
    },
    child: Container(height: 50,width: 50,
      //margin: EdgeInsets.only(top: 420,left: 310),
      decoration: const BoxDecoration(
          image: DecorationImage(
              image: AssetImage("assets/nextarrow.png"),
              fit: BoxFit.contain)
      ),
    ),
  ),
),
//arrow
Container(margin: EdgeInsets.only(top: 634),
```

```
    height: 70,
    width: 410,
    color: Colors.black,
    child: Row(
      children: [
        Expanded(child: InkWell(
            onTap: () {
              Navigator.push(
                  context, MaterialPageRoute(builder: (context) => const Contents())
              );
            },
            child: Container(
              height: 60,
              width: 60,
              decoration: const BoxDecoration(
                  image: DecorationImage(
                      image: AssetImage("assets/contents_red.png")
                  )
              ),
            )
          //Icon(Icons.horizontal_split_rounded,color: Colors.white,size: 50)
        )
        ),
        Expanded(child: InkWell(
            onTap: () {
              Navigator.push(
                  context, MaterialPageRoute(builder: (context) => const FirstScreen())
              );
            },
            child: Container(
              height: 60,
              width: 60,
              decoration: const BoxDecoration(
                  image: DecorationImage(
                      image: AssetImage("assets/home_red.png"),fit: BoxFit.contain)
              ),
            )
          //Icon(Icons.home,color:Colors.white,size: 50)
        )
        ),
        Expanded(child: InkWell(
            onTap: () {
              Navigator.push(
                  context, MaterialPageRoute(builder: (context) => const Contents())
              );
            },
            child: Container(
              height: 60,
              width: 50,
              decoration: const BoxDecoration(
                  image: DecorationImage(
                      image: AssetImage("assets/settings_red.png"), fit: BoxFit.contain)
              ),
            )
          //Icon(Icons.settings,color:Colors.white,size: 50)
        )
        ),
      ],
    )
)
```

```
          ],
        ),

      ),
    );
  }
}
```

Slot Booking:

```
import 'package:autoinsight/screens/CarBook/BookingDetails/comp/BookingBodyFul.dart';
import 'package:flutter/material.dart';

class BookingDetails extends StatelessWidget {
  const BookingDetails({super.key});

  @override
  Widget build(BuildContext context) {
    return const Scaffold(
      body: BookingBodyFul(),
    );
  }
}

import 'package:autoinsight/screens/CarBook/BookingDetails/Pass.dart';
import 'package:autoinsight/screens/CarBook/Successful/BookingSuccess.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:flutter/material.dart';
import 'package:intl/intl.dart';

class BookingBodyFul extends StatefulWidget {
  const BookingBodyFul({super.key});

  @override
  State<BookingBodyFul> createState() => _BookingBodyFulState();
}

class _BookingBodyFulState extends State<BookingBodyFul> {
  TextEditingController dateController = TextEditingController();

  List<String> showrooms = ['Select Showroom','EVM Autokraft-Kochi','EVM Autokraft-Calicut'];
  String? selectedShowroom = 'Select Showroom';

  Pass obj = new Pass();

  @override
  void initState() {
    dateController.text = ""; //set the initial value of text field
    super.initState();
  }
  @override
  Widget build(BuildContext context) {
    return Center(
      child: Scaffold(
        appBar: AppBar(
          title: Image.asset("assets/topbar.png",fit: BoxFit.cover),
          backgroundColor: Colors.black,
        ),
        body: Stack(
          children: [
```

```
Container(
  constraints: const BoxConstraints.expand(),
  decoration: const BoxDecoration(
    image: DecorationImage(
      image: AssetImage("assets/bg.png"),
      fit: BoxFit.cover,
    ),
  ),
),
//bg
Positioned(top: 50,
  child: Container(
    width: 370,
    decoration: BoxDecoration(border: Border.all(color: Colors.white24,),
        borderRadius: BorderRadius.circular(40)),
    child: DropdownButton<String>(
      value: selectedShowroom,
      isExpanded: true,
      borderRadius: BorderRadius.circular(40),
      dropdownColor: Colors.black,
      items: showrooms.map((showroom) => DropdownMenuItem<String>(
        value: showroom,
        child: Text(showroom,
            style: TextStyle(
                fontSize: 22,
                color: Colors.white60
            )),
      ))
          .toList(),
      onChanged: (showroom) => setState(() => selectedShowroom = showroom),
    ),
  ),
),
//showroom textbox
Container(
  // color: Colors.white,
  margin: EdgeInsets.only(top: 160),
    padding:const EdgeInsets.all(15),
    height:50,
    decoration: BoxDecoration(
      border: Border.all(color: Colors.white.withOpacity(0.24)),
        borderRadius: BorderRadius.circular(40)),
    child:TextField(
    cursorColor: Colors.white,
    textAlign: TextAlign.center,
    style: TextStyle(
      color: Colors.white60
    ),
    controller: dateController, //editing controller of this TextField
    decoration: const InputDecoration(
      icon: Icon(Icons.calendar_today),
      hintText: 'Enter date',
      hintStyle: TextStyle(color: Colors.white24),
  ),
  readOnly: true,  // when true user cannot edit text
  onTap: () async {
    DateTime? pickedDate = await showDatePicker(
        context: context,
        initialDate: DateTime.now(), //get today's date
        firstDate: DateTime.now(),
```

```
                lastDate: DateTime(2101)
              );

              if(pickedDate != null ){
                print(pickedDate);  //get the picked date in the format => 2022-07-04 00:00:00.000
                String formattedDate = DateFormat('dd-MM-yyyy').format(pickedDate); // format date
in required form here we use yyyy-MM-dd that means time is removed
                print(formattedDate);
                setState(() {
                  dateController.text = formattedDate; //set foratted date to TextField value.
                });
              }else{
                print("Date is not selected");
              }
            },
          ),
        ),
          //date picker

          Container(margin: EdgeInsets.only(top: 12,left: 5),
            child: Text('Select Showroom',
              style: TextStyle(
                fontSize: 25,
                fontFamily: 'PoppinsReg',
                color: Colors.white,
              ),
            ),
          ),
          //showroom text
          Container(margin: EdgeInsets.only(top: 117,left: 5),
            child: Text('Select Preferred Delivery Date',
              style: TextStyle(
                fontSize: 25,
                fontFamily: 'PoppinsReg',
                color: Colors.white,
              ),
            ),
          ),
          // date text
          Container(margin: EdgeInsets.only(top: 230,left: 5),
            child: Text('*Note: Booking payment instructions will be given by the executive.',
              style: TextStyle(
                fontSize: 18,
                fontFamily: 'PoppinsLight',
                color: Colors.white,
              ),
            ),
          ),
          //payment text
          Positioned(top: 330, left: 180,
            child: InkWell(
              onTap: (){
                obj.receiveSD(selectedShowroom.toString(), dateController.text);
                obj.send();
                Navigator.push(
                  context,
                  MaterialPageRoute(builder: (context) => const BookingSucces()),
                );
              },
              child: Container(height: 50,width: 50,
```

```
                //margin: EdgeInsets.only(top: 420,left: 310),
                decoration: const BoxDecoration(
                    image: DecorationImage(
                        image: AssetImage("assets/nextarrow.png"),
                        fit: BoxFit.contain)
                ),
              ),
            ),
          ),
          //next arrow
        ],
      ),
    ),
  );
 }
}

import 'package:cloud_firestore/cloud_firestore.dart';
String colour="",interior="",showroom="",date="",model="";
class Pass {
 Future<void> recieveCI(String col,String intr)
 async {
  colour=col;
  interior=intr;
 }
 Future<void> receiveSD(String sr,String d)
 async {
  showroom = sr;
  date = d;
 }
 Future<void> receiveModel(String m)
  async {
   model=m;
  }
 Future<void> send()
 async{
  CollectionReference collref = FirebaseFirestore.instance.collection('/User/
hg9ZeOmC4EsWYw8IqS4U/Book');
  collref.add({
   'Showroom': showroom,
   'Pdate': date,
   'Colour':colour,
   'Interior':interior,
   'Model':model,
  });
 }
}
```

ViewPreviousBooking:

```
import 'package:autoinsight/screens/CarBook/PreviousBook/comp/PreviousBookBody.dart';
import 'package:flutter/material.dart';

class prevbook extends StatelessWidget {
 const prevbook({super.key});

 @override
 Widget build(BuildContext context) {
  return Scaffold(
   body: prevbookbody(),
  );
```

```
  }
}

import 'package:autoinsight/screens/CarBook/PreviousBook/booklist.dart';
import 'package:autoinsight/screens/Contents/contents.dart';
import 'package:autoinsight/screens/FirstScreen/FirstScreen.dart';
import 'package:autoinsight/screens/Settings/settings.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';

class prevbookbody extends StatefulWidget {
  const prevbookbody({super.key});

  @override
  State<prevbookbody> createState() => _prevbookbodyState();
}

class _prevbookbodyState extends State<prevbookbody> {

  final user = FirebaseAuth.instance.currentUser!;

  List<String> docIDs = [];

  Future getdocID() async {
    await FirebaseFirestore.instance.collection('/User/hg9ZeOmC4EsWYw8IqS4U/
Book').get().then(
          (snapshot) => snapshot.docs.forEach((document) {
            print(document.reference);
            docIDs.add(document.reference.id);
          }),
    );
  }

  @override
  Widget build(BuildContext context) {
    return Center(
      child: Scaffold(
        appBar: AppBar(
          title: Image.asset("assets/topbar.png", fit: BoxFit.cover),
          backgroundColor: Colors.black,
        ),
        body: Stack(
          children: [
            Container(
              constraints: const BoxConstraints.expand(),
              decoration: const BoxDecoration(color: Colors.black),
            ),
            //bg
            Expanded(
              child: FutureBuilder(
                  future: getdocID(),
                  builder: (context,snapshot) {
                return ListView.builder(
                  itemCount:docIDs.length,
                  itemBuilder: (context,index){
                    return Padding(
                      padding: const EdgeInsets.all(8.0),
                      child: ListTile(
                        tileColor: Colors.grey[200],
```

```
            title: BookList(documentId: docIDs[index]),
          ),
        );
    },
    );
  })
),
Container(
  margin: EdgeInsets.only(top: 634),
  height: 70,
  width: 410,
  color: Colors.black,
  child: Row(children: [
    Expanded(
        child: InkWell(
            onTap: () {
              Navigator.push(
                  context,
                  MaterialPageRoute(
                      builder: (context) => const Contents()));
            },
            child: Container(
              height: 60,
              width: 60,
              decoration: const BoxDecoration(
                  image: DecorationImage(
                      image:
                          AssetImage("assets/contents_cyan.png"))),
            )
            //Icon(Icons.horizontal_split_rounded,color: Colors.white,size: 50)
            )),
    Expanded(
        child: InkWell(
            onTap: () {
              Navigator.push(
                  context,
                  MaterialPageRoute(
                      builder: (context) => const FirstScreen()));
            },
            child: Container(
              height: 60,
              width: 60,
              decoration: const BoxDecoration(
                  image: DecorationImage(
                      image: AssetImage("assets/home_cyan.png"),
                      fit: BoxFit.contain)),
            )
            //Icon(Icons.home,color:Colors.white,size: 50)
            )),
    Expanded(
        child: InkWell(
            onTap: () {
              Navigator.push(
                  context,
                  MaterialPageRoute(
                      builder: (context) => const SettingsPage()));
            },
            child: Container(
              height: 60,
              width: 50,
```

```
                    decoration: const BoxDecoration(
                        image: DecorationImage(
                            image: AssetImage("assets/settings_cyan.png"),
                            fit: BoxFit.contain)),
                    )
                    //Icon(Icons.settings,color:Colors.white,size: 50)
                    )),
              ]),
            ),
          ],
        ),
      ),
    );
  }
}
```

ML MODEL FOR PRICE PREDICTION:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import matplotlib as mpl
%matplotlib inline
mpl.style.use('ggplot')
car=pd.read_csv('/content/bmw.csv')
car.head()
car.shape
car.info()
car.describe(include='all')
car['model'].unique()
import seaborn as sns
plt.subplots(figsize=(15,7))
ax=sns.boxplot(x='model',y='price',data=car)
ax.set_xticklabels(ax.get_xticklabels(),rotation=40,ha='right')
plt.show()
plt.subplots(figsize=(20,10))
ax=sns.swarmplot(x='year',y='price',data=car)
ax.set_xticklabels(ax.get_xticklabels(),rotation=40,ha='right')
plt.show()
sns.relplot(x='mileage',y='price',data=car,height=7,aspect=1.5)
plt.subplots(figsize=(14,7))
sns.boxplot(x='fuelType',y='price',data=car)
ax=sns.relplot(x='model',y='price',data=car,hue='fuelType',size='year',height=7,aspect=2)
ax.set_xticklabels(rotation=40,ha='right')
X=car[['model','year','mileage','fuelType','transmission']]
y=car['price']
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2)
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import OneHotEncoder
from sklearn.compose import make_column_transformer
from sklearn.pipeline import make_pipeline
from sklearn.metrics import r2_score
ohe=OneHotEncoder()
ohe.fit(X[['model','fuelType','transmission']])
column_trans=make_column_transformer((OneHotEncoder(categories=ohe.categories_),
['model','fuelType','transmission']),
                                     remainder='passthrough')
lr=LinearRegression()
pipe=make_pipeline(column_trans,lr)
```

```
pipe.fit(X_train,y_train)
y_pred=pipe.predict(X_test)
r2_score(y_test,y_pred)
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.1,random_state=np.argmax(scores))
lr=LinearRegression()
pipe=make_pipeline(column_trans,lr)
pipe.fit(X_train,y_train)
y_pred=pipe.predict(X_test)
r2_score(y_test,y_pred)
```

Appendix B: CO PO MAPPING

COURSE OUTCOMES:

After completion of the course the student will be able to

SL. NO	DESCRIPTION	Blooms' Taxonomy Level
CO1	Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO2	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO3	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO4	Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO5	Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)	Level 3: Apply

CO-PO AND CO-PSO MAPPING

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO3
CO1	3	3	3	3		2	2	3	2	2	2	3	2	2	2
CO2	3	3	3	3	3	2		3	2	3	2	3	2	2	2
CO3	3	3	3	3	3	2	2	3	2	2	2	3			2
CO4	2	3	2	2	2			3	3	3	2	3	2	2	2
CO5	3	3	3	2	2	2	2	3	2		2	3	2	2	2

3/2/1: high/medium/low

JUSTIFICATIONS FOR CO-PO MAPPING

MAPPING	LOW/ MEDIUM/ HIGH	JUSTIFICATION
100003/CS622T.1-PO1	**HIGH**	Identify technically and economically feasible problems by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
100003/CS622T.1-PO2	**HIGH**	Identify technically and economically feasible problems by analysing complex engineering problems reaching substantiated conclusions using first principles of mathematics.
100003/CS622T.1-PO3	**HIGH**	Design solutions for complex engineering problems by identifying technically and economically feasible problems.
100003/CS622T.1-PO4	**HIGH**	Identify technically and economically feasible problems by analysis and interpretation of data.
100003/CS622T.1-PO6	**MEDIUM**	Responsibilities relevant to the professional engineering practice by identifying the problem.
100003/CS622T.1-PO7	**MEDIUM**	Identify technically and economically feasible problems by understanding the impact of the professional engineering solutions.
100003/CS622T.1-PO8	**HIGH**	Apply ethical principles and commit to professional ethics to identify technically and economically feasible problems.
100003/CS622T.1-PO9	**MEDIUM**	Identify technically and economically feasible problems by working as a team.
100003/CS622T.1-PO10	**MEDIUM**	Communicate effectively with the engineering community by identifying technically and economically feasible problems.
100003/CS622T.1-P011	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles by selecting the technically and economically feasible problems.
100003/CS622T.1-PO12	**HIGH**	Identify technically and economically feasible problems for long term learning.
100003/CS622T.1-PSO1	**MEDIUM**	Ability to identify, analyze and design solutions to identify technically and economically feasible problems.
100003/CS622T.1-PSO2	**MEDIUM**	By designing algorithms and applying standard practices in software project development and Identifying technically and economically feasible problems.
100003/CS622T.1-PSO3	**MEDIUM**	Fundamentals of computer science in competitive research can be applied to Identify technically and economically feasible problems.
100003/CS622T.2-PO1	**HIGH**	Identify and survey the relevant by applying the knowledge of mathematics, science, engineering fundamentals.

100003/CS622T.2-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems get familiarized with software development processes.
100003/CS622T.2-PO3	**HIGH**	Design solutions for complex engineering problems and design based on the relevant literature.
100003/CS622T.2-PO4	**HIGH**	Use research-based knowledge including design of experiments based on relevant literature.
100003/CS622T.2-PO5	**HIGH**	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes by using modern tools.
100003/CS622T.2-PO6	**MEDIUM**	Create, select, and apply appropriate techniques, resources, by identifying and surveying the relevant literature.
100003/CS622T.2-PO8	**HIGH**	Apply ethical principles and commit to professional ethics based on the relevant literature.
100003/CS622T.2-PO9	**MEDIUM**	Identify and survey the relevant literature as a team.
100003/CS622T.2-PO10	**HIGH**	Identify and survey the relevant literature for a good communication to the engineering fraternity.
100003/CS622T.2-PO11	**MEDIUM**	Identify and survey the relevant literature to demonstrate knowledge and understanding of engineering and management principles.
100003/CS622T.2-PO12	**HIGH**	Identify and survey the relevant literature for independent and lifelong learning.
100003/CS622T.2-PSO1	**MEDIUM**	Design solutions for complex engineering problems by Identifying and survey the relevant literature.
100003/CS622T.2-PSO2	**MEDIUM**	Identify and survey the relevant literature for acquiring programming efficiency by designing algorithms and applying standard practices.
100003/CS622T.2-PSO3	**MEDIUM**	Identify and survey the relevant literature to apply the fundamentals of computer science in competitive research.
100003/CS622T.3-PO1	**HIGH**	Perform requirement analysis, identify design methodologies by using modern tools & advanced programming techniques and by applying the knowledge of mathematics, science, engineering fundamentals.
100003/CS622T.3-PO2	**HIGH**	Identify, formulate, review research literature for requirement analysis, identify design methodologies and develop adaptable & reusable solutions.

100003/CS622T.3-PO3	**HIGH**	Design solutions for complex engineering problems and perform requirement analysis, identify design methodologies.
100003/CS622T.3-PO4	**HIGH**	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
100003/CS622T.3-PO5	**HIGH**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools.
100003/CS622T.3-PO6	**MEDIUM**	Perform requirement analysis, identify design methodologies and assess societal, health, safety, legal, and cultural issues.
100003/CS622T.3-PO7	**MEDIUM**	Understand the impact of the professional engineering solutions in societal and environmental contexts and Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions.
100003/CS622T.3-PO8	**HIGH**	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions by applying ethical principles and commit to professional ethics.
100003/CS622T.3-PO9	**MEDIUM**	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
100003/CS622T.3-PO10	**MEDIUM**	Communicate effectively with the engineering community and with society at large to perform requirement analysis, identify design methodologies.
100003/CS622T.3-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering requirement analysis by identifying design methodologies.
100003/CS622T.3-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by analysis, identify design methodologies and develop adaptable & reusable solutions.
100003/CS622T.3-PSO3	**MEDIUM**	The ability to apply the fundamentals of computer science in competitive research and prior to that perform requirement analysis, identify design methodologies.
100003/CS622T.4-PO1	**MEDIUM**	Prepare technical report and deliver presentation by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
100003/CS622T.4-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems by preparing technical report and deliver presentation.

100003/CS622T.4-PO3	**MEDIUM**	Prepare Design solutions for complex engineering problems and create technical report and deliver presentation.
100003/CS622T.4-PO4	**MEDIUM**	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions and prepare technical report and deliver presentation.
100003/CS622T.4-PO5	**MEDIUM**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and Prepare technical report and deliver presentation.
100003/CS622T.4-PO8	**HIGH**	Prepare technical report and deliver presentation by applying ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
100003/CS622T.4-PO9	**HIGH**	Prepare technical report and deliver presentation effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
100003/CS622T.4-PO10	**HIGH**	Communicate effectively with the engineering community and with society at large by prepare technical report and deliver presentation.
100003/CS622T.4-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work by prepare technical report and deliver presentation.
100003/CS622T.4-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by prepare technical report and deliver presentation.
100003/CS622T.4-PSO1	**MEDIUM**	Prepare a technical report and deliver presentation to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas.
100003/CS622T.4-PSO2	**MEDIUM**	To acquire programming efficiency by designing algorithms and applying standard practices in software project development and to prepare technical report and deliver presentation.
100003/CS622T.4-PSO3	**MEDIUM**	To apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs by preparing technical report and deliver presentation.
100003/CS622T.5-PO1	**HIGH**	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
100003/CS622T.5-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems by applying engineering and management principles to achieve the goal of the project.

100003/CS622T.5-PO3	**HIGH**	Apply engineering and management principles to achieve the goal of the project and to design solutions for complex engineering problems and design system components or processes that meet the specified needs.
100003/CS622T.5-PO4	**MEDIUM**	Apply engineering and management principles to achieve the goal of the project and use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
100003/CS622T.5-PO5	**MEDIUM**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO6	**MEDIUM**	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities by applying engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO7	**MEDIUM**	Understand the impact of the professional engineering solutions in societal and environmental contexts, and apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO8	**HIGH**	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice and to use the engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO9	**MEDIUM**	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PSO1	**MEDIUM**	The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas. Apply engineering and management principles to achieve the goal of the project.

100003/CS622T.5-PSO2	**MEDIUM**	The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PSO3	**MEDIUM**	The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur and apply engineering and management principles to achieve the goal of the project.