

EasyMech

(On Road Vehicle Breakdown Assistance)

ANDROID APPLICATION

PROJECT SUBMITTED TO THE

GOA UNIVERSITY

BY

Mr. MANSOOR NIJATULLAH

Mr. BAKHTIAR ABDUL RAQIB

Mr. MOHAMOD JUMA KHAN

OF

THIRD YEAR B.Sc. COMPUTER SCIENCE

2019-2020

UNDER GUIDANCE OF

MRS. AVANI KHARDE

(DEPARTMENT OF COMPUTER SCIENCE)

ST.XAVIER’S COLLEGE OF ARTS, SCIENCE &

COMMERCE

MAPUSA-GOA

CERTIFICATE

Certified that this project is a record of work by the candidates themselves during the period

of study under my guidance, to the best of my knowledge and that it has not previously

formed the basis of the award of any Degree or Diploma in the Goa University or elsewhere.

Mrs. AVANI KHARDE Mr. EDWIN D’SOUZA

(Project Guide) (H.O.D, Dept. of Computer Science)

St. Xavier’s College St. Xavier’s College

DECLARATION

We declare that this project report has been wholly composed and compiled by us and has not been the basis for award of any degree or diploma in GOA University or elsewhere .

Third Year B.Sc.(Computer Science)

Mr. MANSOOR NIJATULLAH Mr. BAKHTIAR ABDUL RAQIB

( ) ( )

Mr. MOHAMOD JUMA KHAN

( )

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all effort with success.

We are extremely grateful to our Project Guide, Mrs. Avani Kharde for being a source of inspiration and for her constant support in the design and implementation of this project. We are also thankful to the Head of the Computer Science Department, Mr. Edwin D’Souza for his constant support throughout the project, which benefited us a lot while developing the project.

The have been a constant source of inspiration and motivation for hard work. Through this column, it would be our utmost pleasure to express our warm thanks to them for their encouragement, co-operation and consent without which we would not be able to accomplish this project.

We also express our gratitude to our Principal (Dr. Blanche Mascarenhas) for providing us the infrastructure to carry out the project and all staff members who were directly and indirectly instrumental in enabling us to stay committed for this project.

Last but not the least, we would like to thank Mr. Ashley Fernandes for his dedication and help in the Computer Laboratory.

CONTENTS

|  |  |  |
| --- | --- | --- |
| Sr. No. | Topic | Page No. |
| 1 | Introduction | 1-2 |
| 2 | Basics of Android | 3-6 |
| 3 | Technologies Used | 7-13 |
| 4 | Development Environment | 14-17 |
| 5 | Use Case Diagram | 18 |
| 6 | Use Case Description | 19-24 |
| 7 | Firebase Realtime Database | 25-28 |
| 8 | Software Requirement | 29 |
| 9 | User Manual | 30-44 |
| 10 | Further Enhancement | 45 |
| 11 | Conclusion | 46 |
| 12 | References | 47 |

**INTRODUCTION**

In recent years, with the rapid growth in technology, mobile phones have become a vital part in human life. Even though mobile phones’ core function is being telecommunication but now mobile apps have made many things possible. The services that mobile apps and smartphone provide are plenty. For example, online shopping, food delivery and taxi service apps, banking apps, e-tickets. By using these apps, the people would simply fulfil their craving with few clicks over the phone.

Our project which is a mobile app called “EasyMech”, can provide vehicle repairing services. As we know today the number of vehicles increases day by day and everyone want to have their own vehicle. It is important to keep your vehicle in good driving condition and therefore regular servicing is very important. Servicing of a vehicle is very important, especially if you are use your vehicle extensively. By servicing your vehicle on a regular basis, you will avoid costly repairs and will save lot of money. So you need to find a mechanic to do this job for you. For this you need to go out and you have to search for a master mechanic which you can trust to service your vehicle, it is costly and time consuming and not sure you will find a good and trusted mechanics or not.

It would be much easier, if you are simply sitting at home or from your work place and find a master mechanic which you can trust. Through our EasyMech app, these services can be provided to repair your vehicle. For this purpose, the app developed by us, is user-friendly and it helps save time, cost and the most importantly, you can find the best mechanic simply with few clicks over the phone.

Our application “EasyMech”, you can find mechanic and you can get rid of your vehicle problems and service it with a trusted mechanic. The app consists of three users- Admin, mechanic and user (vehicle owner).

The admin does the system management work, for example adding a new service or adding mechanic, removing mechanic and other tasks related to him.

The user should login by entering the respective identifications. The user will get the menu of all the services available in the system after logging in. After clicking a specific service, the user can see list of all the mechanics. The user can search for a specific mechanic. After finding, the user and mechanic can connect with each other by the details provided.

The mechanic will be added by the admin and the mechanic can accept an appointment or reject the appointment.

# EasyMech app’s goal is to offer clients and customers with safe and suitable vehicle repair services. This app can reduce your work and can easily find the service centers from various areas at the same time reducing the user time and cost of repairing.

**BASICS OF ANDROID**

**What is Android?**

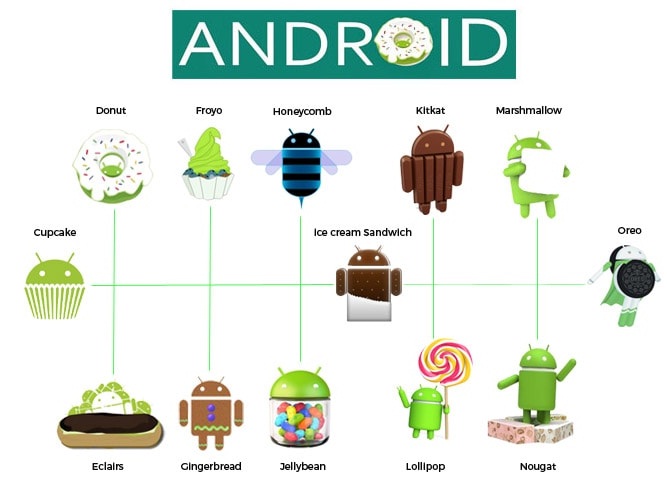
Android is a Linux based operating system it is designed primarily for touch screen mobile devices such as smart phones and tablet computers. The operating system have developed a lot in last 15 years starting from black and white phones to recent smart phones or mini computers. One of the most widely used mobile OS   these days is android.  The android is software that was founded in Palo Alto of California in 2003

The android is a powerful operating system and it supports large number of applications in Smartphones. These applications are more comfortable and advanced for the users. The android is an open source operating system means that it’s free and any one can use it. The android has got millions of apps available that can help you managing your life one or other way and it is available low cost in market at that reasons android is very popular.

[](https://www.elprocus.com/wp-content/uploads/2013/10/Android-Logo.png)

The android development supports with the full java programming language. Even other packages that are API and JSE are not supported. The first version 1.0 of android development kit (SDK) was released in 2008 and latest updated version is jelly bean.

Google's Android named all of its version codenames after desserts (just as Intel names all of its CPUs after rivers). To celebrate a new version, a giant mock-up of the dessert that matches the codename is usually delivered to the Google Campus and put on display.



The Latest Version of Android is 10.0 (Q). Its initial release was on September 3, 2019 on Google's Pixel devices. Unlike previous versions of Android, this version doesn't have a cute.

## **API Levels of Android**

API Level is an integer value that uniquely identifies the framework API revision offered by a version of the Android platform.

|  |  |  |  |
| --- | --- | --- | --- |
| **Code name** | **Version number(s)** | **Initial release date** | **API level** |
| No codename | 1.0 | September 23, 2008 | 1 |
| 1.1 | February 9, 2009 | 2 |
| [Cupcake](https://en.wikipedia.org/wiki/Android_Cupcake) | 1.5 | April 27, 2009 | 3 |
| [Donut](https://en.wikipedia.org/wiki/Android_Donut) | 1.6 | September 15, 2009 | 4 |
| [Eclair](https://en.wikipedia.org/wiki/Android_Eclair) | 2.0 – 2.1 | October 26, 2009 | 5 – 7 |
| [Froyo](https://en.wikipedia.org/wiki/Android_Froyo) | 2.2 – 2.2.3 | May 20, 2010 | 8 |
| [Gingerbread](https://en.wikipedia.org/wiki/Android_Gingerbread) | 2.3 – 2.3.7 | December 6, 2010 | 9 – 10 |
| [Honeycomb](https://en.wikipedia.org/wiki/Android_Honeycomb) | 3.0 – 3.2.6 | February 22, 2011 | 11 – 13 |
| [Ice Cream Sandwich](https://en.wikipedia.org/wiki/Android_Ice_Cream_Sandwich) | 4.0 – 4.0.4 | October 18, 2011 | 14 – 15 |
| [Jelly Bean](https://en.wikipedia.org/wiki/Android_Jelly_Bean) | 4.1 – 4.3.1 | July 9, 2012 | 16 – 18 |
| [KitKat](https://en.wikipedia.org/wiki/Android_KitKat) | 4.4 – 4.4.4 | October 31, 2013 | 19 – 20 |
| [Lollipop](https://en.wikipedia.org/wiki/Android_Lollipop) | 5.0 – 5.1.1 | November 12, 2014 | 21 – 22 |
| [Marshmallow](https://en.wikipedia.org/wiki/Android_Marshmallow) | 6.0 – 6.0.1 | October 5, 2015 | 23 |
| [Nougat](https://en.wikipedia.org/wiki/Android_Nougat) | 7.0 – 7.1.2 | August 22, 2016 | 24 – 25 |
| [Oreo](https://en.wikipedia.org/wiki/Android_Oreo) | 8.0 – 8.1 | August 21, 2017 | 26 – 27 |
| [Pie](https://en.wikipedia.org/wiki/Android_Pie) | 9.0 | August 6, 2018 | 28 |
| [Android 10](https://en.wikipedia.org/wiki/Android_10) | 10.0 | September 3, 2019 | 29 |

**TECHNOLOGIES USED**

**JAVA**

Java is a programming language and computing platform first released by Sun Microsystems in 1995. There are lots of applications and websites that will not work unless you have Java installed, and more are created every day. Java is fast, secure, and reliable. From laptops to data centres, game consoles to scientific supercomputers, cell phones to the Internet, Java is everywhere.

**Java** is a general-purpose, concurrent, object-oriented, class-based, and the runtime environment(JRE) which consists of **JVM** which is the cornerstone of the Java platform. It is owned by Oracle.

The Android platform allows developers to write managed code using Java to manage and control the Android device. Android applications can be developed by using the Java programming language and the Android SDK. So, familiarity with the basics of the Java programming language is a prerequisite for programming on the Android platform.

Java is used for:

* Mobile applications (specially Android apps)
* Desktop applications
* Web applications
* Web servers and application servers
* Games
* Database connection
* And much, much more!

**Picasso**

[Picasso](http://square.github.io/picasso/) is an image library for Android. It's created and maintained by [Square](http://square.github.io/), and caters to image loading and processing. It simplifies the process of displaying images from external locations. In many cases only a few lines of code is required to implement this neat library.

Picasso shines for displaying remote images. The library handles every stage of the process, from the initial HTTP request to the caching of the image. This can be quite verbose when writing code to perform these actions yourself.

[**Firebase**](http://firebase.google.com/official/site%E2%80%8E)

Firebase is a mobile and web app development platform that provides developers with a plethora of tools and services to help them develop high-quality apps, grow their user base, and earn more profit.

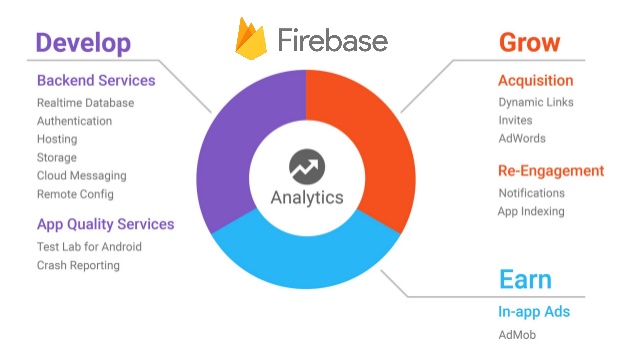
With a variety of server-side technologies that are on the market today, developers have a tough job of deciding what kind of backend is most suitable for their app.



In April 2012, Firebase was created as a separate company that provided Backend-as-a-Service with real-time functionality.

After it was acquired by Google in 2014, Firebase rapidly evolved into the multifunctional behemoth of a mobile and web platform that it is today.

**Firebase Services**

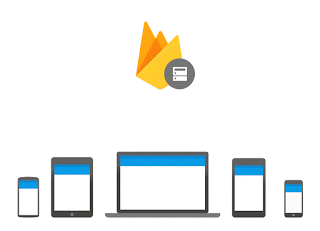
****

#### Firebase Services can be divided into two groups:

|  |  |
| --- | --- |
| Develop & test your app | Grow & Engage your audience |
| * [Realtime Database](https://firebase.google.com/docs/database/) * [Auth](https://firebase.google.com/docs/auth/) * [Test Lab](https://firebase.google.com/docs/test-lab/) * [Cloud Functions](https://firebase.google.com/docs/functions/) * [Firestore](https://firebase.google.com/docs/firestore/) * [Cloud Storage](https://firebase.google.com/docs/storage/) * [Performance Monitoring](https://firebase.google.com/docs/perf-mon/) * [Crash Reporting](https://firebase.google.com/docs/crash/) * [Hosting](https://firebase.google.com/docs/hosting/) | * [Firebase Analytics](https://firebase.google.com/docs/analytics/) * [Invites](https://firebase.google.com/docs/invites/) * [Cloud Messaging](https://firebase.google.com/docs/cloud-messaging/) * [Predictions](https://firebase.google.com/docs/predictions/) * [AdMob](https://firebase.google.com/docs/admob/) * [Dynamic Links](https://firebase.google.com/docs/dynamic-links/) * [Adwords](https://firebase.google.com/docs/adwords/) * [Remote Config](https://firebase.google.com/docs/remote-config/) * [App Indexing](https://firebase.google.com/docs/app-indexing/) |

### **Firebase Realtime Database**

Store and sync data with our NoSQL cloud database. Data is synced across all clients in realtime, and remains available when your app goes offline. The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in realtime to every connected client. When you build cross-platform apps with our iOS, Android, and JavaScript SDKs, all of your clients share one Realtime Database instance and automatically receive updates with the newest data.



Realtime syncing makes it easy for your users to access their data from any device, be it web or mobile. Realtime Database also helps your users collaborate with one another. Another amazing benefit of Realtime Database is that it ships with mobile and web SDKs, allowing you to build your apps without the need for servers. When your users go offline, the Realtime Database SDKs use local cache on the device to serve and store changes. When the device comes online, the local data is automatically synchronized. The Realtime Database can also integrate with Firebase Authentication to provide a simple and intuitive authentication process. Firebase apps remain responsive even when offline because the Firebase Realtime Database SDK persists your data to disk. Once connectivity is reestablished, the client device receives any changes it missed, synchronizing it with the current server state.

### **Firebase Authentication**

Most apps need to know the identity of a user. Knowing a user's identity allows an app to securely save user data in the cloud and provide the same personalized experience across all of the user's devices.

Firebase Authentication provides backend services, easy-to-use SDKs, and ready-made UI libraries to authenticate users to your app. It supports authentication using passwords, phone numbers, popular federated identity providers like Google, Facebook and Twitter, and more.



Firebase Authentication provides backend services, easy-to-use SDKs, and ready-made UI libraries to authenticate users to your app. Normally, it would take you months to set up your own authentication system. And even after that, you would need to keep a dedicated team to maintain that system. But if you use Firebase, you can set up the entire system in under 10 lines of code that will handle everything for you, including complex operations like account merging. You can authenticate your app’s users through the following methods:

* Email & Password
* Phone numbers
* Google
* Facebook
* Twitter

**Firebase Cloud Storage**

Cloud Storage is built for app developers who need to store and serve user-generated content, such as photos or videos. Cloud Storage for Firebase is a powerful, simple, and cost-effective object storage service built for Google scale. The Firebase SDKs for Cloud Storage add Google security to file uploads and downloads for your Firebase apps, regardless of network quality. You can use our SDKs to store images, audio, video, or other user-generated content. On the server, you can use [Google Cloud Storage](https://cloud.google.com/storage), to access the same files.

**Firebase Hosting**

Firebase Hosting provides fast and secure hosting for your web app, static and dynamic content, and microservices. Firebase Hosting is production-grade web content hosting for developers. With a single command, you can quickly deploy web apps and serve both static and dynamic content to a global CDN (content delivery network). You can also [pair Firebase Hosting with Cloud Functions or Cloud Run](https://firebase.google.com/docs/hosting/serverless-overview) to build and host microservices on Firebase.

# Cloud Functions for Firebase

Cloud Functions for Firebase let you automatically run backend code in response to events triggered by Firebase features and HTTPS requests. Your code is stored in Google's cloud and runs in a managed environment. There's no need to manage and scale your own servers.

### **Firebase Cloud Messaging (FCM)**



Firebase Cloud Messaging (FCM) provides a reliable and battery-efficient connection between your server and devices that allows you to deliver and receive messages and notifications on iOS, Android, and the web at no cost. You can send notification messages (2KB limit) and data messages (4KB limit).Using FCM, you can easily target messages using predefined segments or create your own, using demographics and behavior. You can send messages to a group of devices that are subscribed to specific topics, or you can get as granular as a single device.FCM can deliver messages instantly, or at a future time in the user’s local time zone. You can send custom app data like setting priorities, sounds, and expiration dates, and also track custom conversion events. The best thing about FCM is that there is hardly any coding involved! FCM is completely integrated with Firebase Analytics, giving you detailed engagement and conversion tracking.

**DEVELOPMENT ENVIROMENT**

**Android software development kit**



A software development kit that enables developers to create application for the Android platform. The Android SDK includes sample projects with source code, development tools, an emulator, and required libraries to build Android application. Application are written using the java programming language and run on Dalvik, a custom virtual machine designed for embedded use which runs on top of a Linux kernel.

**Android Studio**



Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems.

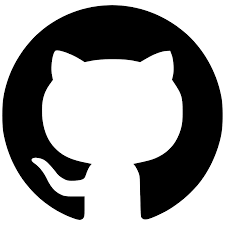
**ADOBE PHOTOSHOP**



Adobe Photoshop is the predominant photo editing and manipulation software on the market. Its uses range from the full-featured editing of large batches of photos to creating intricate digital paintings and drawings that mimic those done by hand.

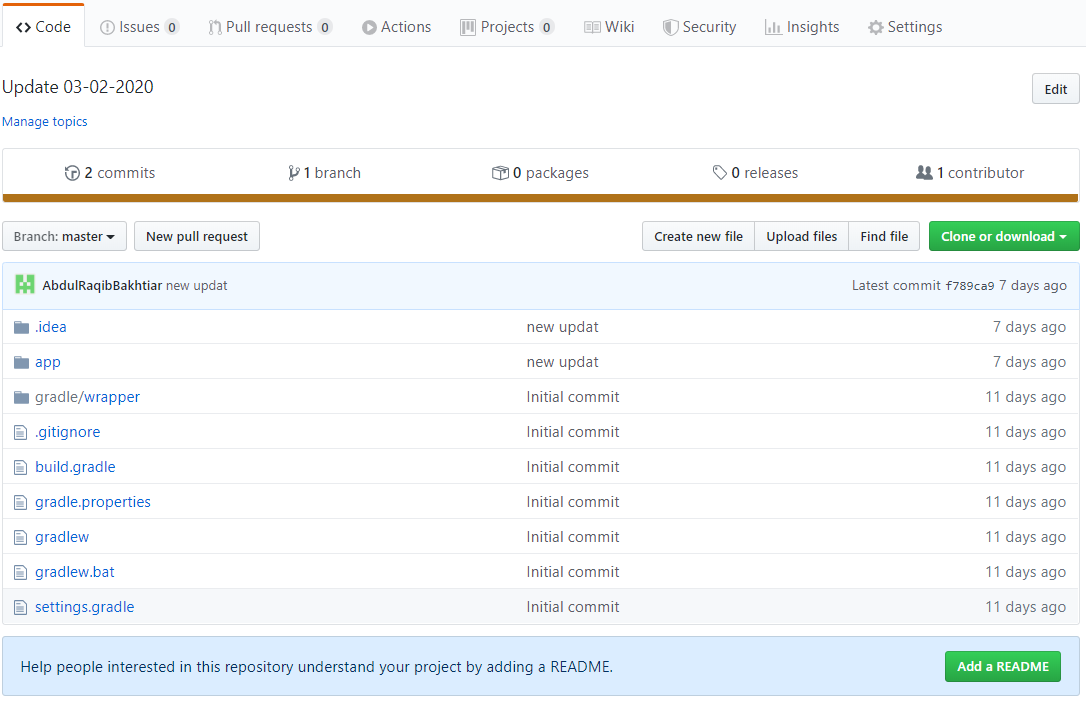
Adobe Photoshop is a critical tool for designers, web developers, graphic artists, photographers, and creative professionals. It is widely used for image editing, logo and icon design, retouching, creating image compositions, website mockups, and adding affects. Digital or scanned images can be edited for use online or in-print

**Git and GitHub**

Git is a distributed version-control system for tracking changes in source code during software development. It is designed for coordinating work among programmers, but it can be used to track changes in any set of files. Its goals include speed, data integrity, and support for distributed, non-linear workflows.

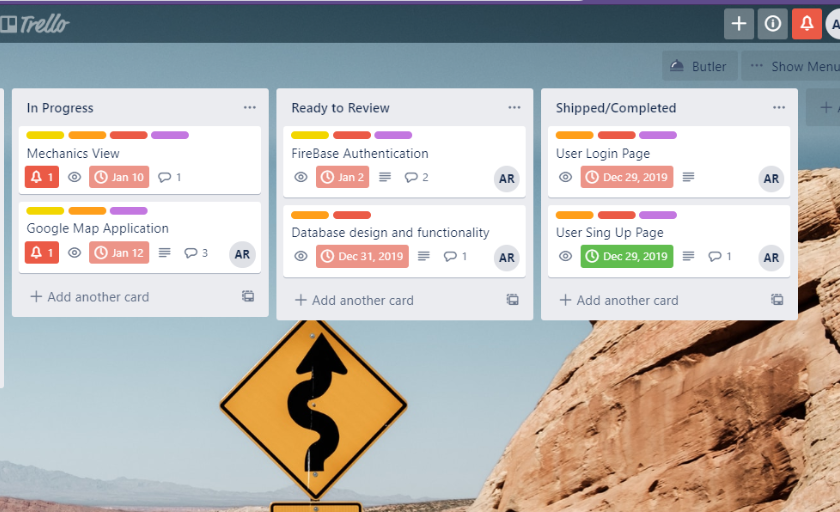
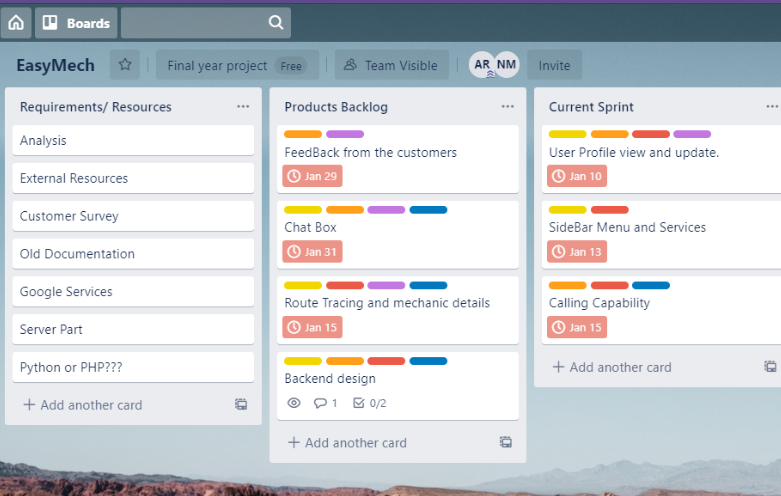
GitHub is a development platform inspired by the your work. From open source to business, you can host and review code, manage projects, and build software alongside millions of developers.



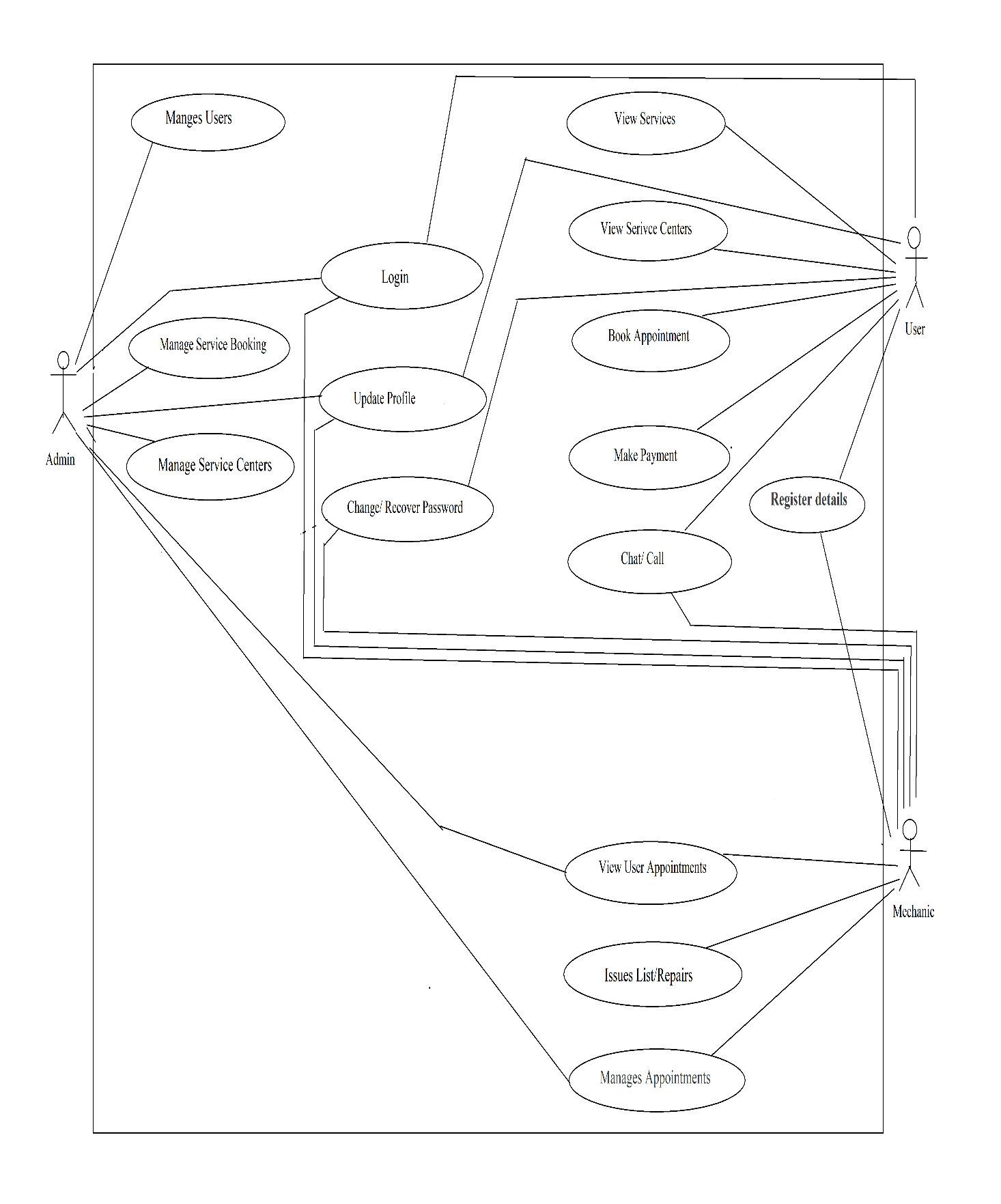
**Trello**



Trello is a collaboration tool that organizes your projects into boards. In one glance, Trello tells you what's being worked on, who's working on what, and where something is in a process. Imagine a white board, filled with lists of sticky notes, with each note as a task for you and your team. Now imagine that each of those sticky notes has photos, attachments from other data sources like Bit Bucket or Salesforce, documents, and a place to comment and collaborate with your teammates. Now imagine that you can take that whiteboard anywhere you go on your smartphone, and can access it from any computer through the web. That's Trello!

****

**USE CASE DIAGRAM ( EasyMech)**



**USE CASE DESCRIPTION**

**Use case description : Login**

* **Use case name**: Login
* **Summary** : Mechanic and user can login.
* **Actor :** Mechanic , User
* **Pre-Condition :** The user should have an account.
* **Use case description :** The system requests the user to enter their credentials to login in this case the identification is password and Email. The app validates the Password and the Email and logs the particular user into the app.
* **Alternative** : If the respective identifications is invalid, then the user is notified.
* **Post Condition** : The user is logged in successfully into the system.

**Use case description : Register**

* **Use case name: Register**
* **Summary :** User successfully registers to the system
* **Actor :** user, mechanic
* **Pre-Condition :** The system sign up page is displayed.
* **Description :** The system will allow a user to become a member.
* **Alternatives :** If the details are incorrect or incomplete, an error message will be displayed.
* **Post Condition:** The user has successfully registered into the system.

**Use case Description: Update Profile**

* **Use Case Description name**: Update Profile
* **Summary:** user can update their profile details
* **Actor:** User, mechanic, admin
* **Pre-Condition:** The member should be logged in into their account.
* **Description:** The member can update their details and the system saves the details successfully.
* **Alternative:** If the details are incorrect or incomplete, an error message will be displayed.
* **Post-Condition:** The member profile is updated successfully into the system.

**Use case description: Change/ Recover Passwords**

* **Use case name**: Change/Recover Password
* **Summary:** The users can recover or change their passwords
* **Actors:** user, mechanic, admin
* **Pre-Condition**: The users must have an account in case of recovering password, or they should be logged in case of changing password.
* **Description:** In case of changing password, system displays a message to update to new password. User enters new password and the system validates the entered details. The system updates the password successfully.
* In case of recovery of password, system displays alternative method to recover password. User enters required details and waits for system notification for password update.
* **Alternative:** The system displays an error message for wrong password details.
* **Post-Condition**: Their passwords are updated successfully in the system.

**Use case Description: View Services**

* **Use case name: View Services**
* **Actor: User**
* **Summary:** An authenticated user can view the list of services.
* **Pre-Condition: The user is successfully logged in to the system.**
* **Description: The user can view the list services and can choose a service which can solve the problem.**
* **Post-Condition: The system displays the list of services selected by user.**

**Use case description: View Service Centres**

* **User case name:** View Service Centres
* **Summary:** The system displays the list of service centres. Users can choose the facilities such as mechanic selection, booking an appointment, call/chat, locate the service centre etc.
* **Actor:** Users, mechanics
* **Pre-condition:** The user has to be logged in into the system.
* **Description:** The system displays list of service centres. User views list of services centres. user can choose any service such as mechanic selection, booking an appointment, call/chat, locate the service centre etc.
* **Alternative:** system displays appropriate error messages in case of service not being available.
* **Post-Condition:** System performs the required action for the user.

**Use case description: Book Appointment**

* **Use case name**: Book Appointment
* **Summary:** Book an Appointment
* **Actor** : User
* **Pre-condition** : The user is successfully logged in to the system.
* **Description** : the user must fill the required fields for booking an appointment. System records the booking appointment of the user and sends a notification to the user.
* **Alternative** : Appropriate error messages are displayed in case of incomplete/ invalid details entered by the user.
* **Post-condition** : System sends notification of confirmed booking appointment.

**Use case description : Chat and Call**

* **Use case name**: Chat and Call
* **Summary**: Mechanics and Users can communicate via chat/ call.
* **Actor:** Mechanics, Users
* **Pre-Condition**: The user is successfully logged in to the system.
* **Use case description**: User chooses chat/ call option to communicate to a mechanic. System establishes a connection to the mechanic.
* **Post-Condition**: The user and mechanic have successfully communicated.

**Use case description : Manages Appointments**

* **Use case name**: Manage Appointments
* **Actor**: Mechanic
* **Summary:** The Service centres (mechanics) are able to view list of all users who made appointment with them
* **Pre-Condition**: The user is successfully logged in to the system.
* **Use case description**: The system displays the appointment details of the users.
* **Post-Condition**: Mechanics take appropriate actions on the booked appointments of the users.

**Use case description : Manage Users**

* **Use case name**: Manage Users
* **Actor**: Admin
* **Summary**: The admin manages the system users.
* **Pre-Condition**: The admin is successfully logged in to the system.
* **Use case description**: The system displays the user details to admin. Admin does the needful changes with respect to the user.
* **Post-Condition:** The changes made by admin, are saved successfully into the system.

**Use case description : Manage Service Centres**

* **Use case name**: Manage Service Centres
* **Actor:** Admin
* **Summary**: The admin manages the Service Centres registered in the system.
* **Pre-Condition:** The admin is successfully logged in to the system.
* **Use case description:** The system displays the service centres details to the admin. Admin does the needful changes with respect to the service centres.
* **Post-Condition:** The changes made by admin, are saved successfully into the system.

**Use case description : Manage Service Booking**

**Use case name**: Manage Service Booking

**Actor**: Admin

**Summary**: The admin manages the service booking and appointments.

**Pre-Condition**: The admin is successfully logged in to the system.

**Use case description**: The system displays the booked services and appointments made by the users. Admin can see and manage all the booked services and payments.

**Post-Condition**: If any change made by admin with respect to appointments and payments, they are saved successfully into the system.

## **FIREBASE REALTIME DATABASE**

**What does Firebase Realtime mean ?**

Instead of typical HTTP requests, the Firebase Realtime Database uses data synchronization—every time data changes, any connected device receives that update within milliseconds. Provide collaborative and immersive experiences without thinking about networking code.

**How the data is stored in Firebase Realtime Database?**

All Firebase Realtime Database data is stored as JSON objects. You can think of the database as a cloud-hosted JSON tree.

Unlike a SQL database, there are no tables or records. When you add data to the JSON tree, it becomes a node in the existing JSON structure with an associated key. You can provide your own keys, such as user IDs or semantic names, or they can be provided for you using  [push()](https://firebase.google.com/docs/reference/js/firebase.database.Reference#push). Data is stored in firebase as a large JSON document. The data is stored as a large objects which can hold key value pairs where value can be a string, number or another object.

**MAIN ROOT (JSON TREE)**

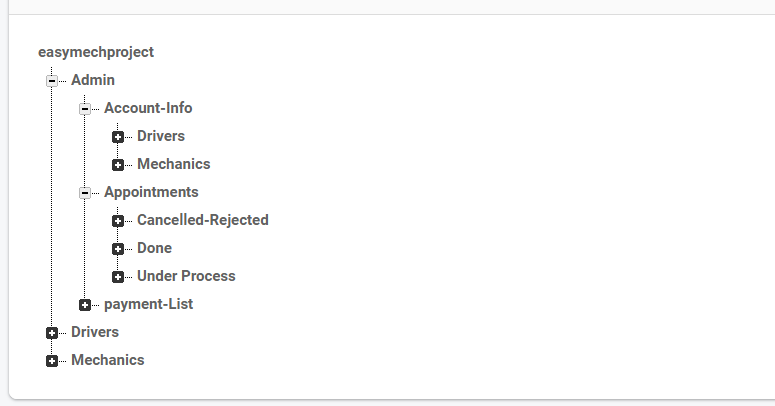
The root of the JSON tree is easymechproject, it has three branches- Admin, Drivers (user) and mechanics.

Branch name: Admin

Admin has a number of sub branches.

Branches:

1. Account info: Information about all the accounts of users and mechanics
2. Appointments: list of all appointments made by users whether those are under process, completed and rejected or cancelled.
3. Payment-List: Data about the payments (future enhancement)

****

**Branch name: Drivers(Users)**

Keys:

1. Driver key: Each driver has a unique key which acts as an id, for example the

03MiAN23ExXY1s1kUCE9DpPmrDD3 is the key of our first user.

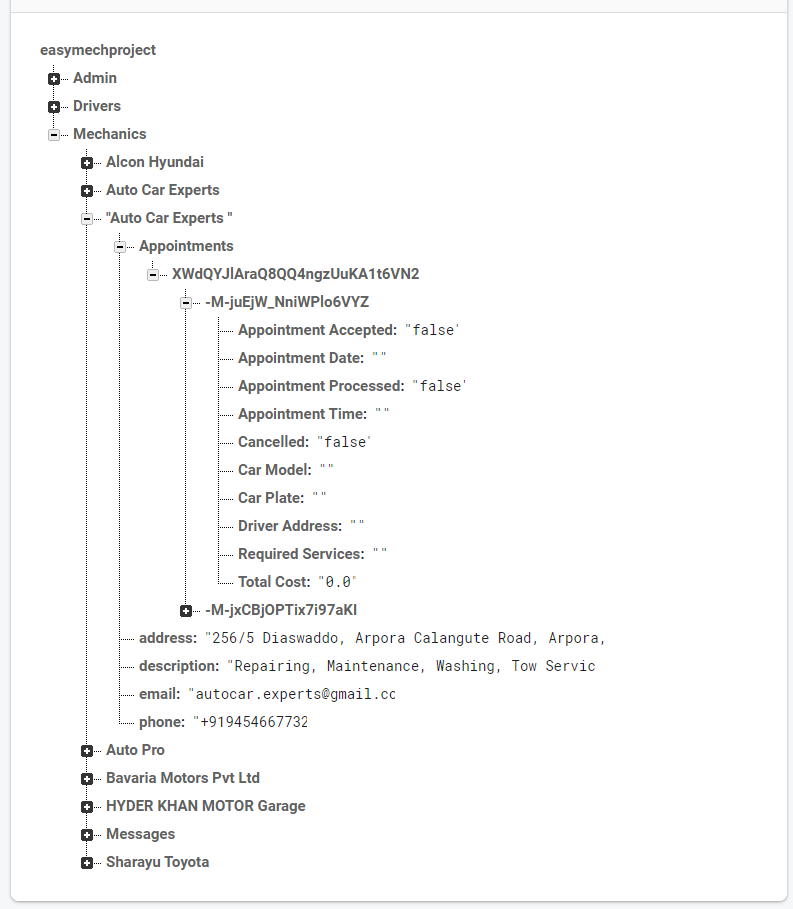
1. Appointment key: Each appointment made by the user has a unique key similar to driver key; which acts as an identifier to these appointments.

****

**Branch name: Mechanics**

Keys:

1. Mechanic id: their service centre name acts as an identifier.
2. Appointment key: Each appointment made by the user has a unique key; which acts as an identifier to these appointments in mechanic branch.

****

**SOFTWARE-REQUIREMENTS**

▪ OS: Android 8.0 (Oreo) or above.

▪ ROM: 8GB or above.

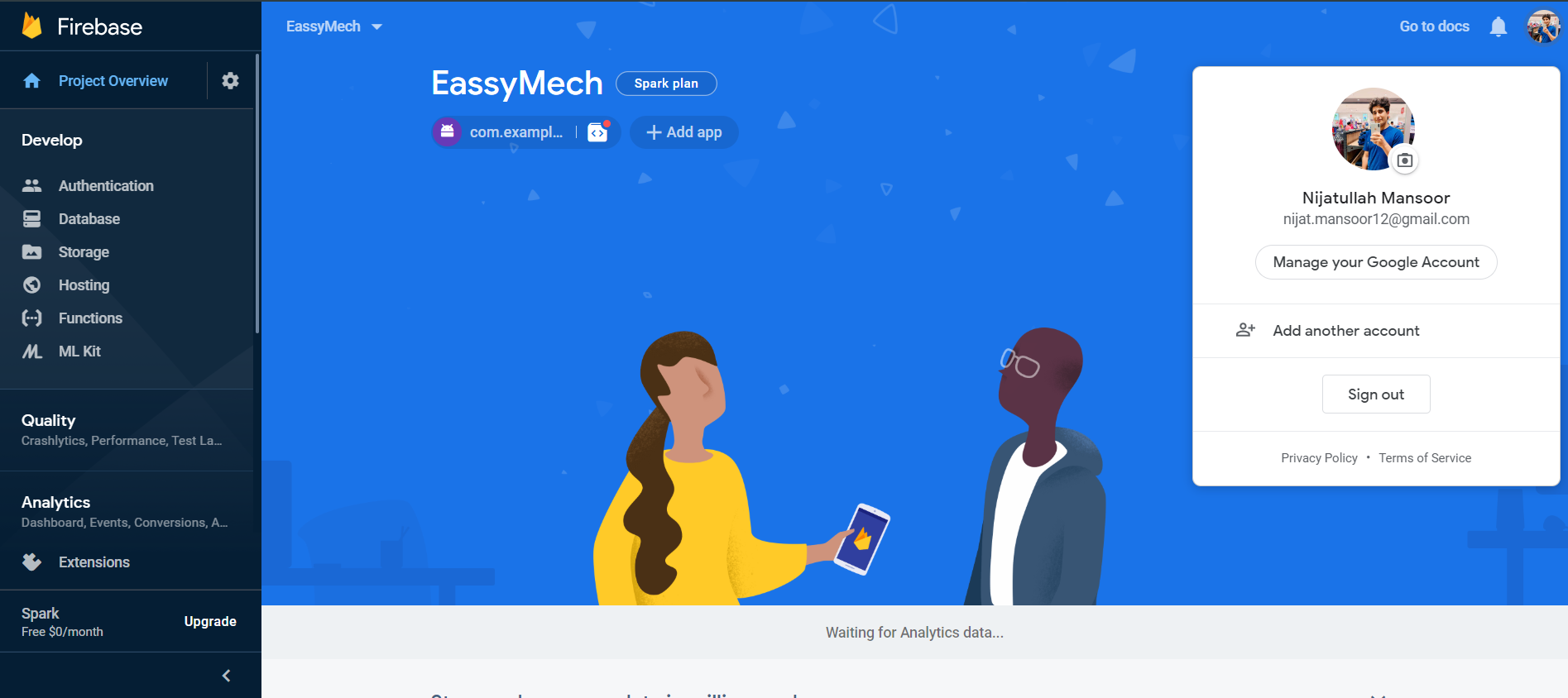
▪ RAM: 1GB or above.

▪ Internet connectivity

**USER MANUAL**

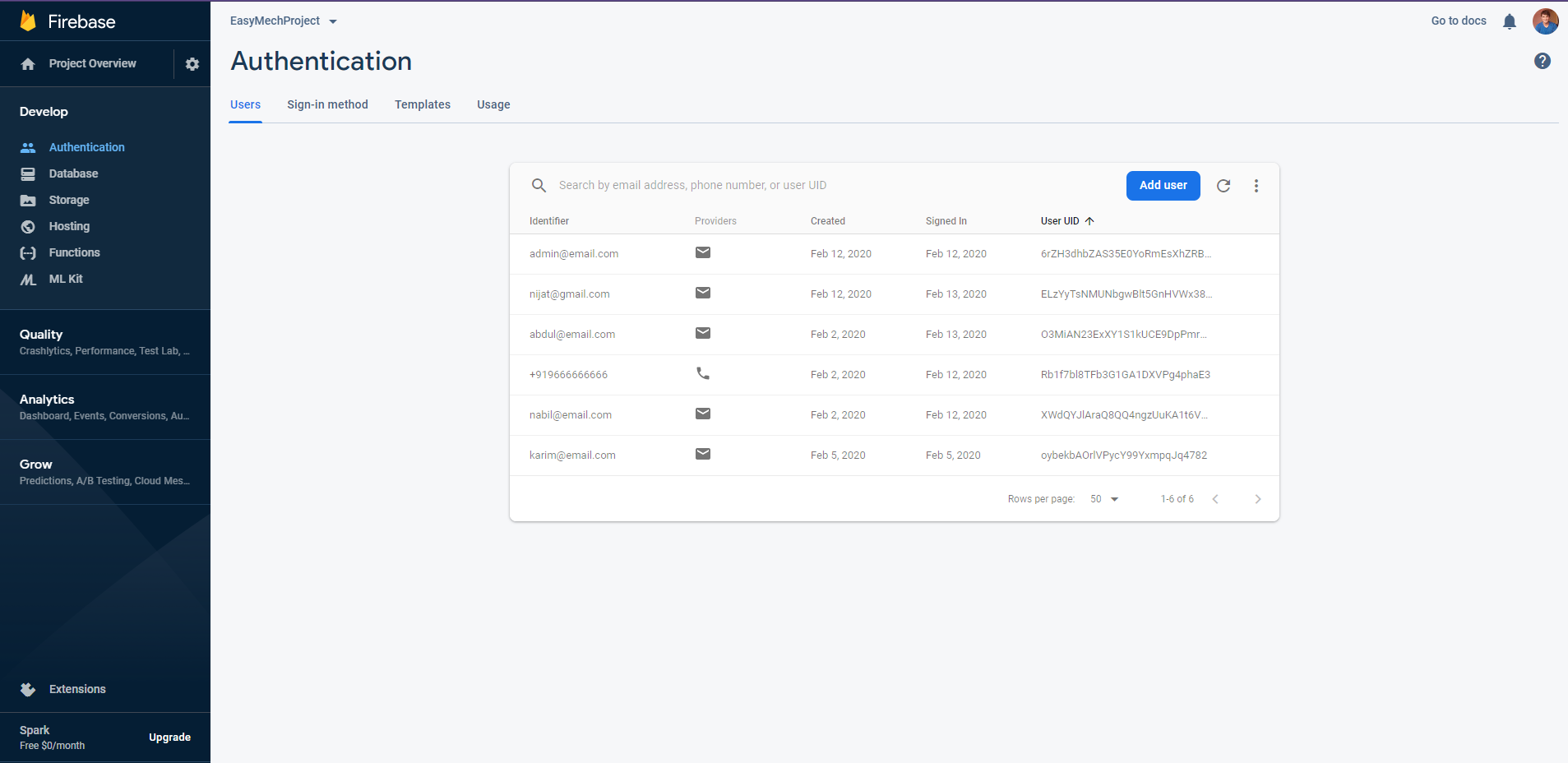
**ADMIN**

Admin can login through their account and can do the System management. For example admin can add the user and remove the user. Admin can reset the password, disable the account, and can delete the account of the user. Similarly admin can add and remove the services, as well as the mechanics form the app.



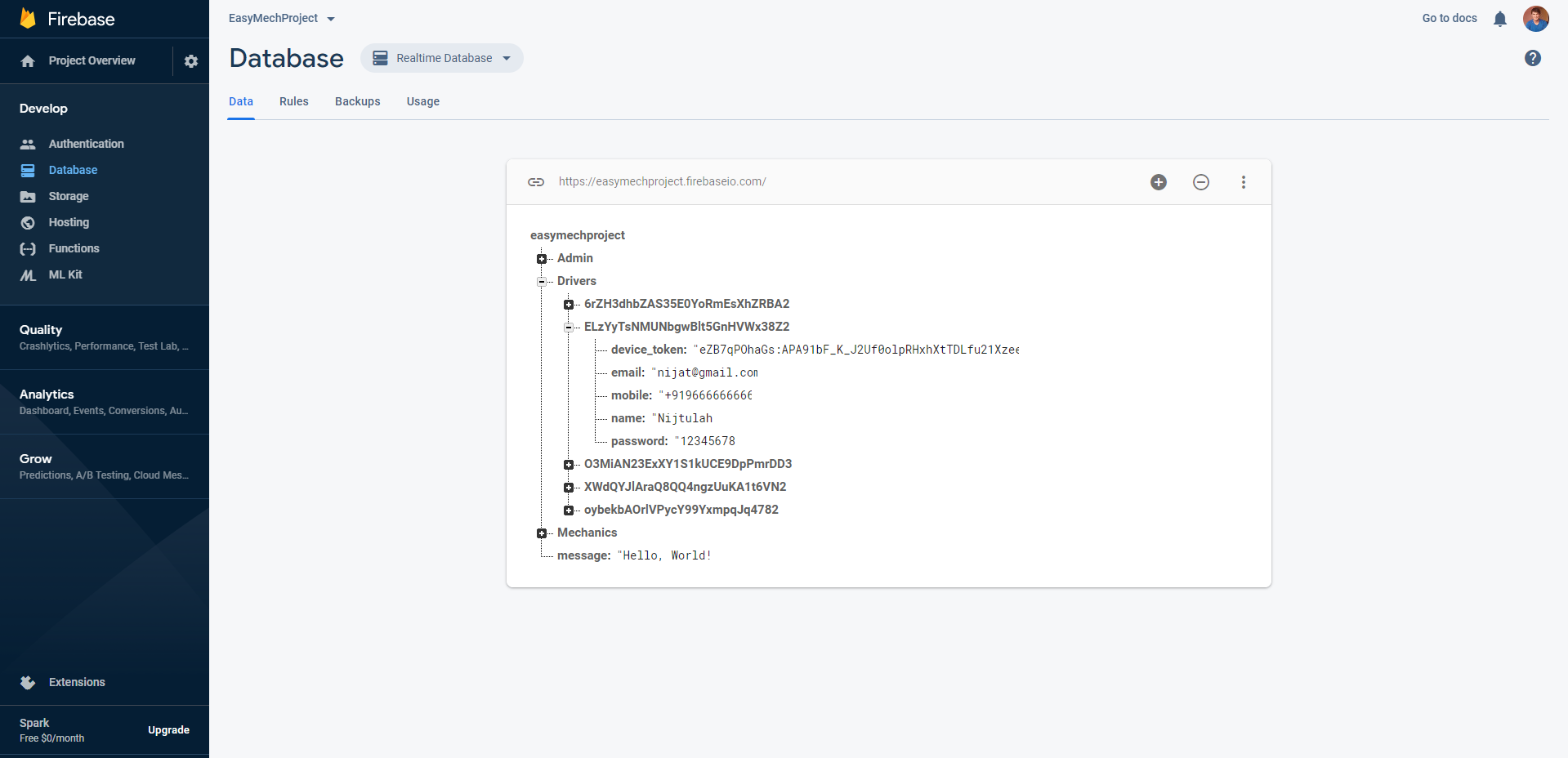
**Authentication :**

After registration of the user the firebase store the email address and password here and give a unique id to each user.



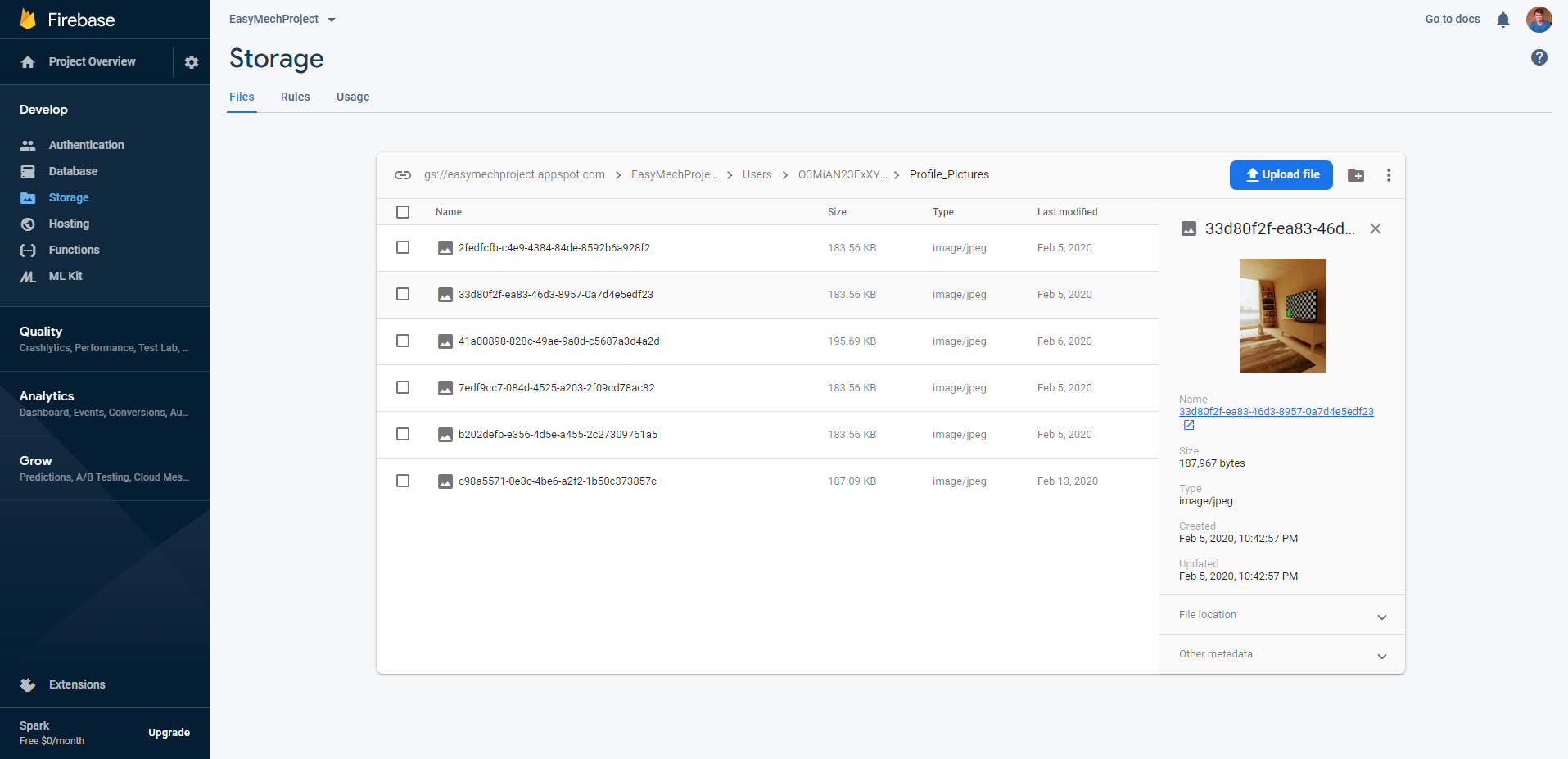
**Database**

The overall view of database.

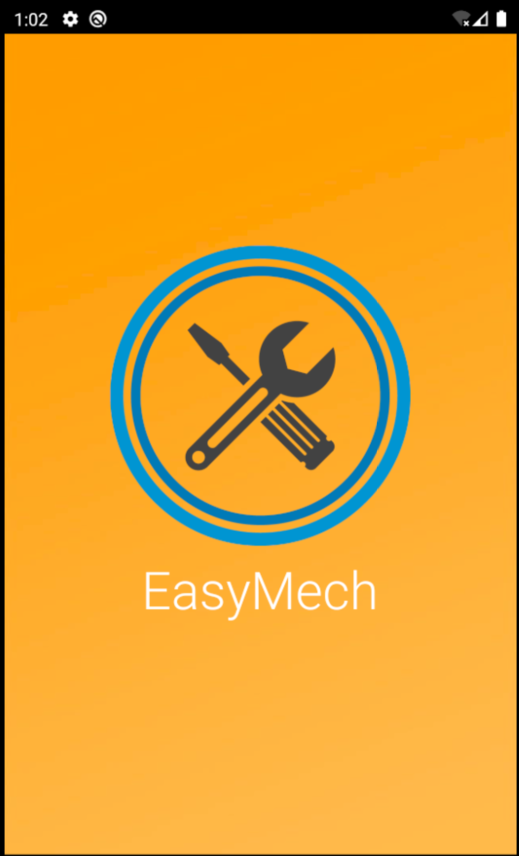


**Storage**

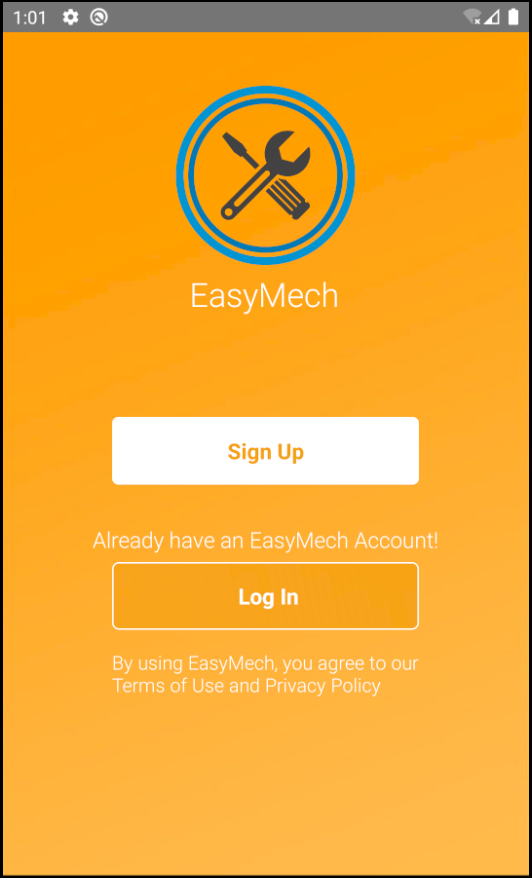
All the files and images are store in firebase storage.



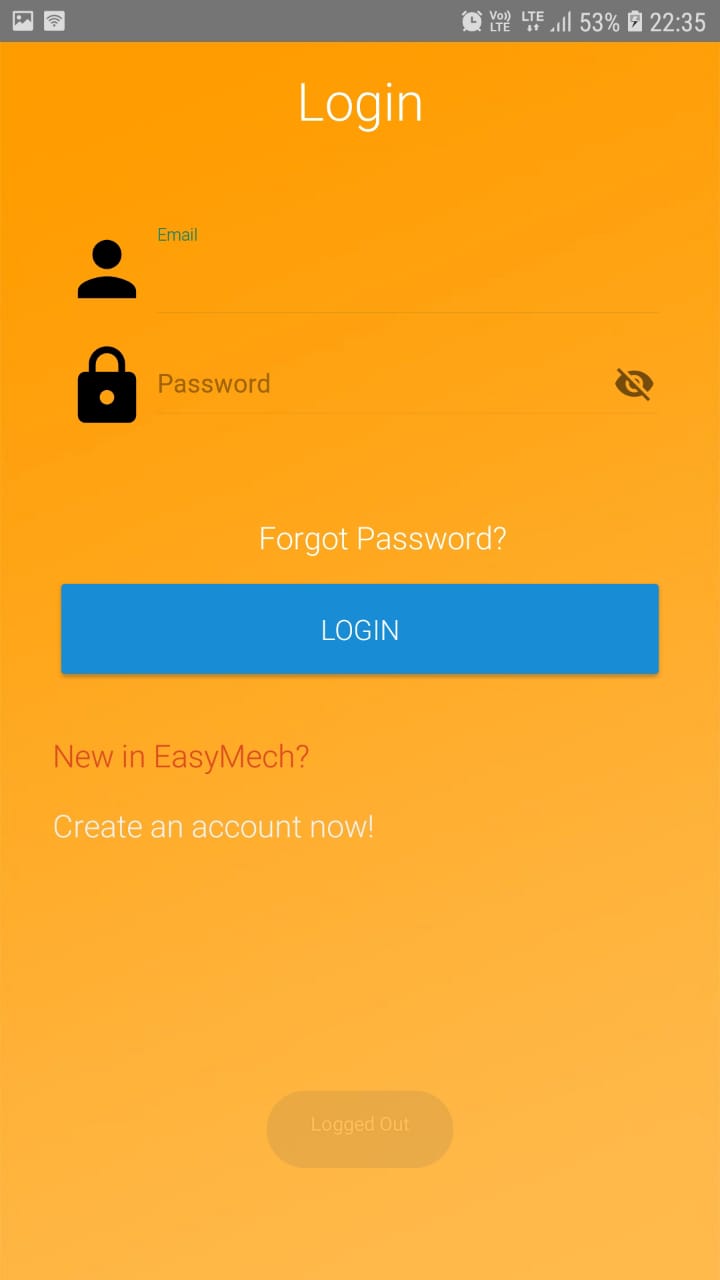
**Start-up Page**

****

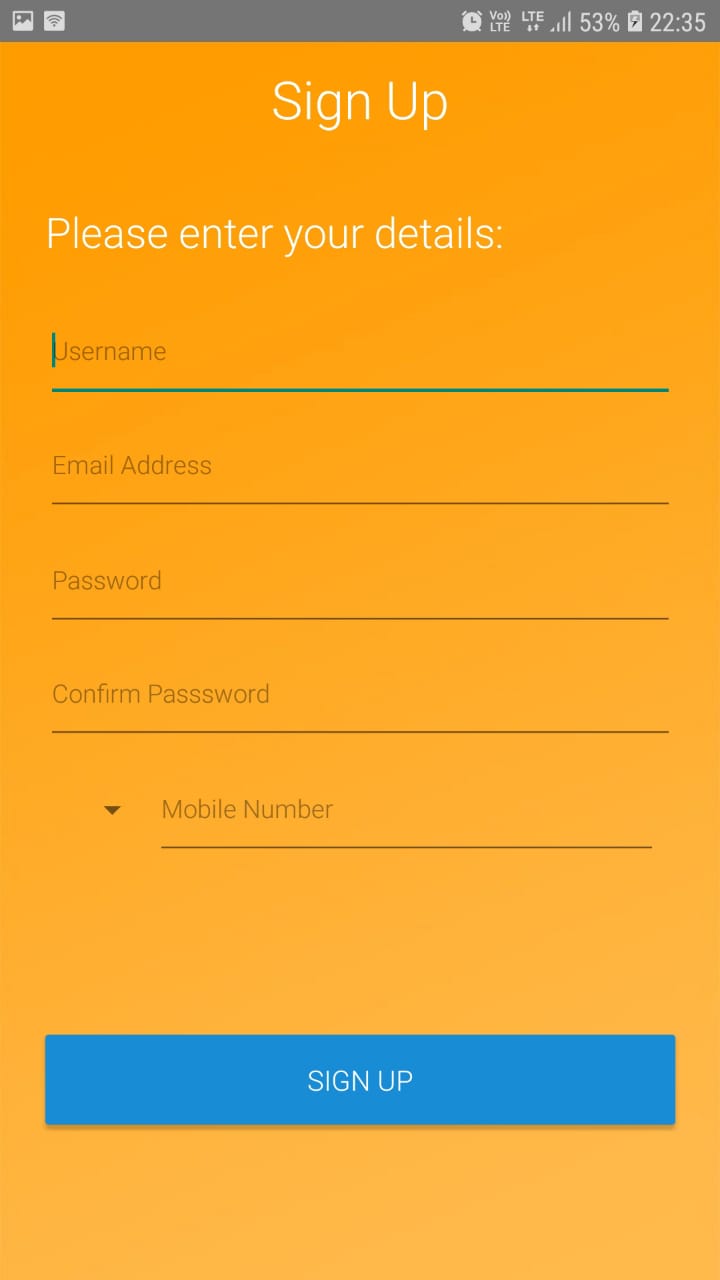
**Home Page Activity**

****

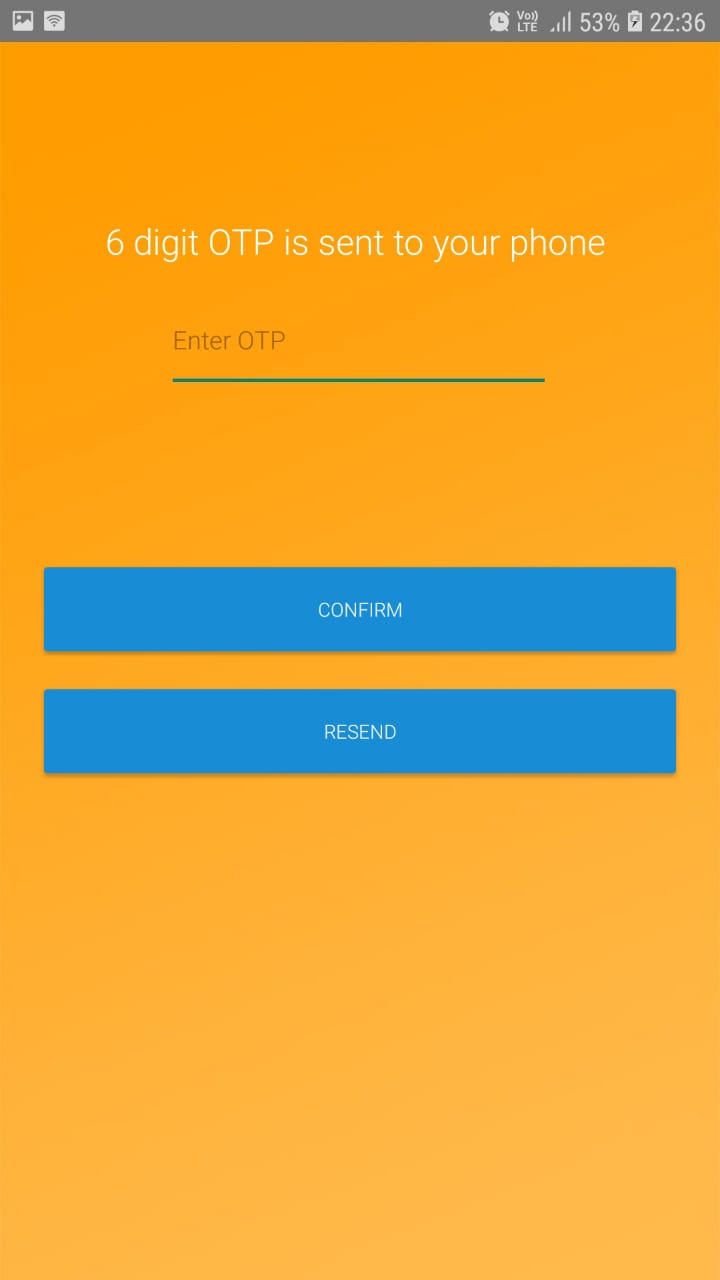
**Login Activity**



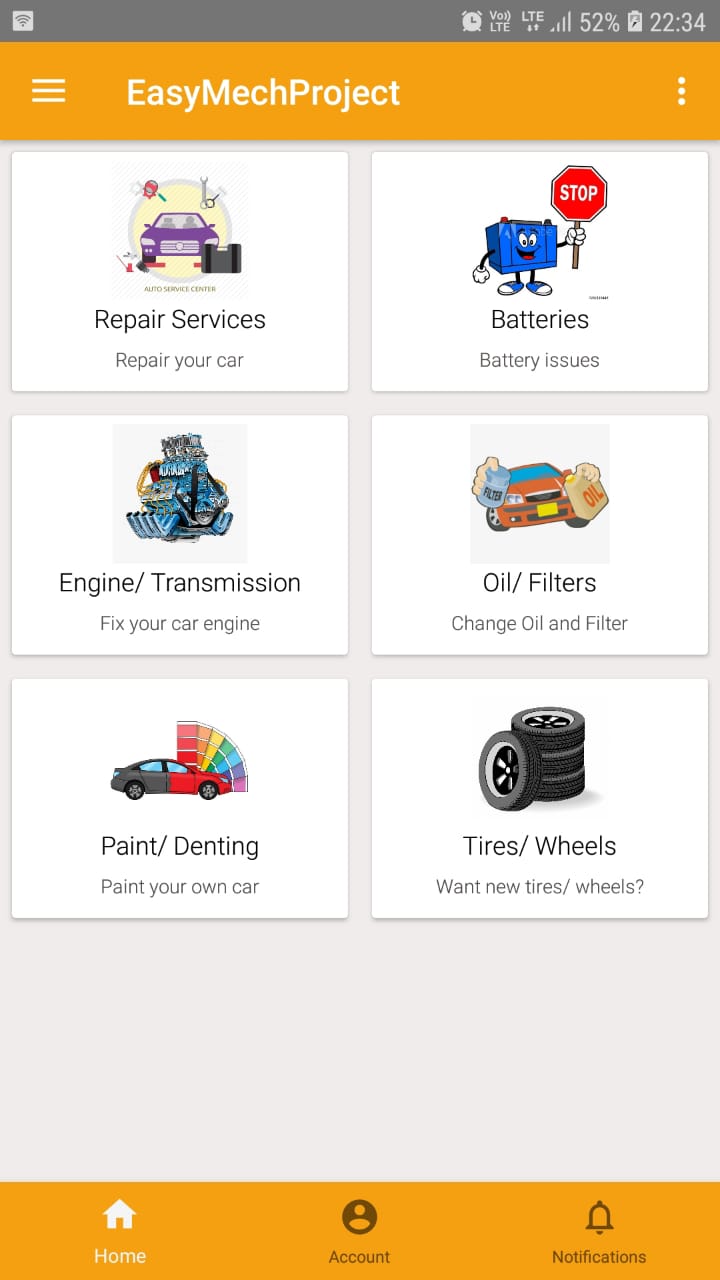
**Sign Up Activity**



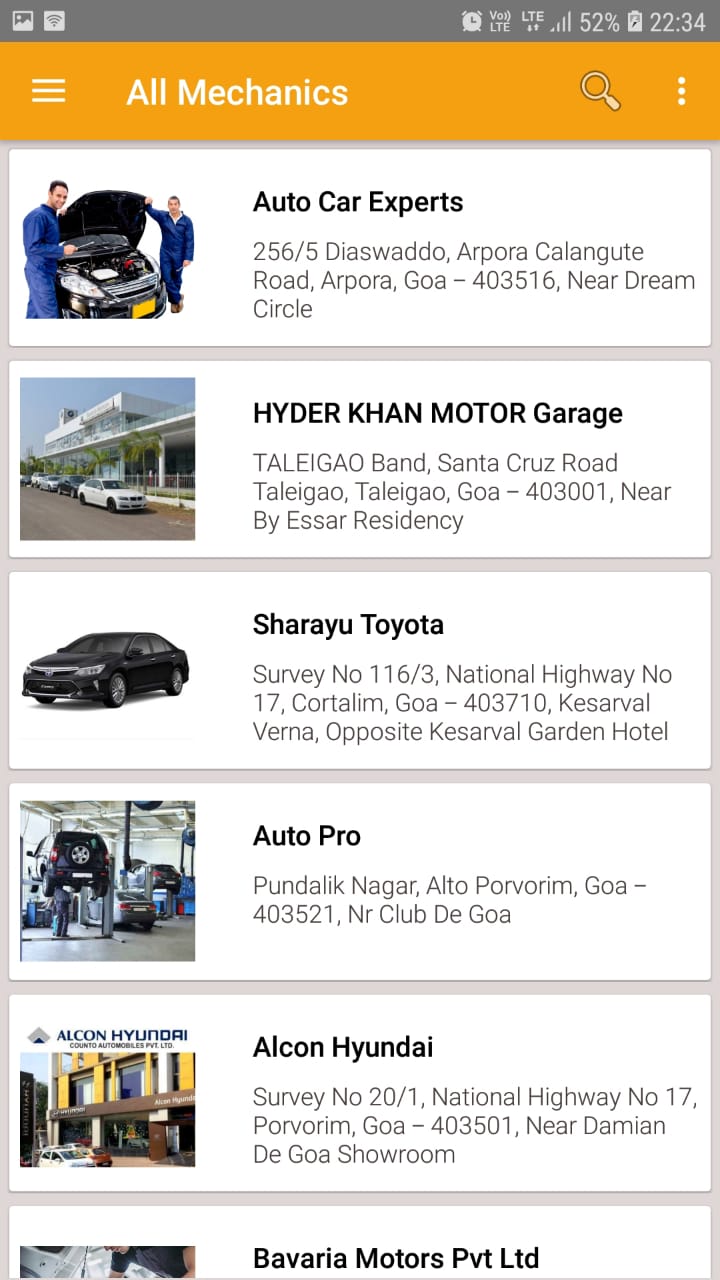
**Mobile Verification Activity**



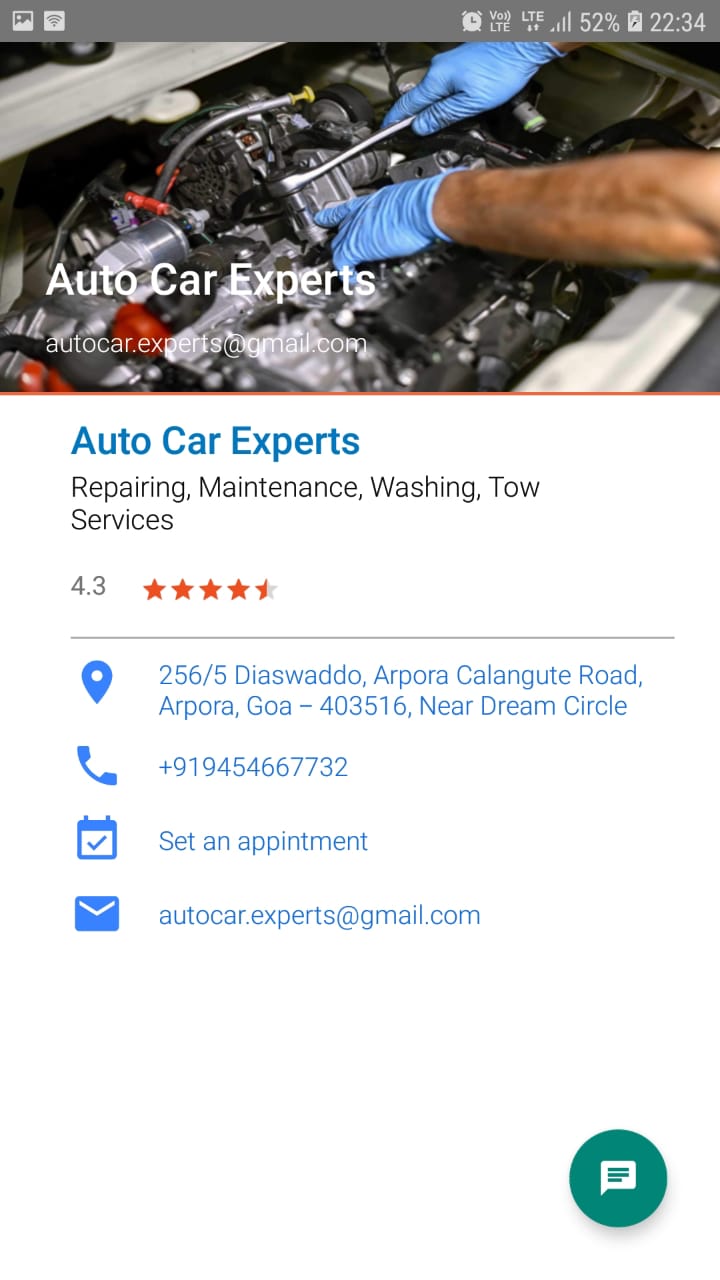
**Services Activity**



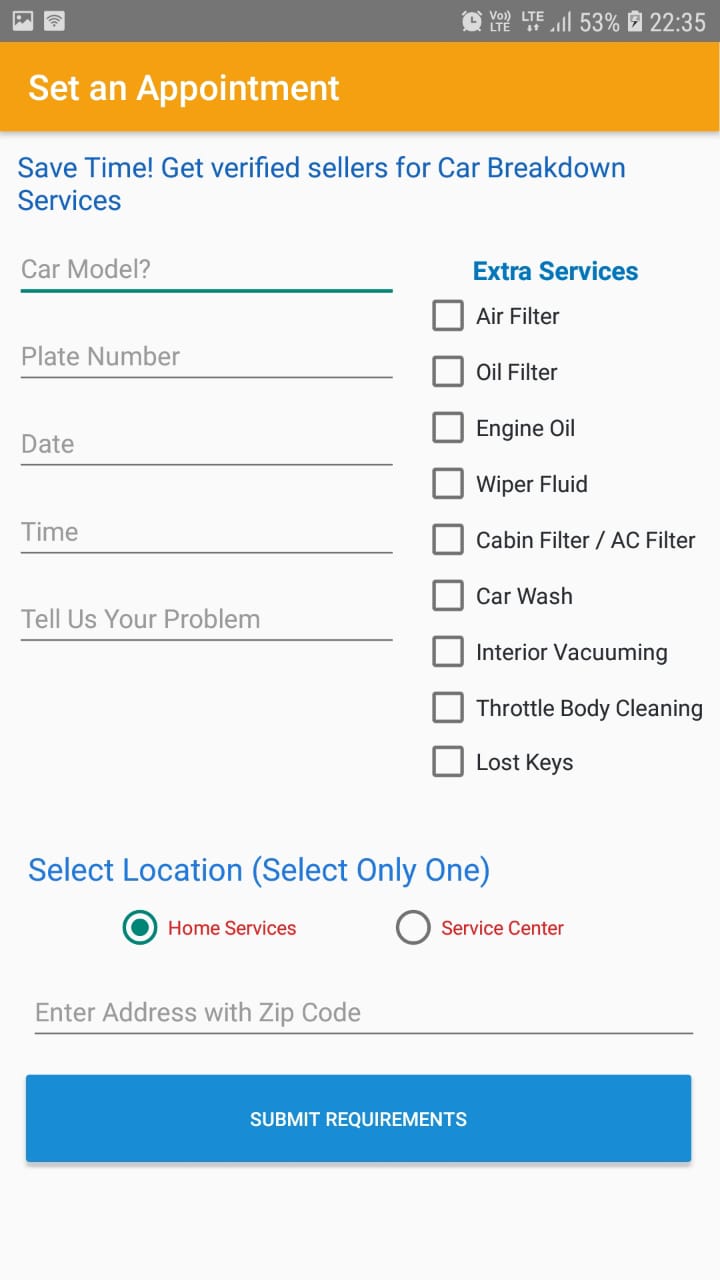
**Service Centres Activity**



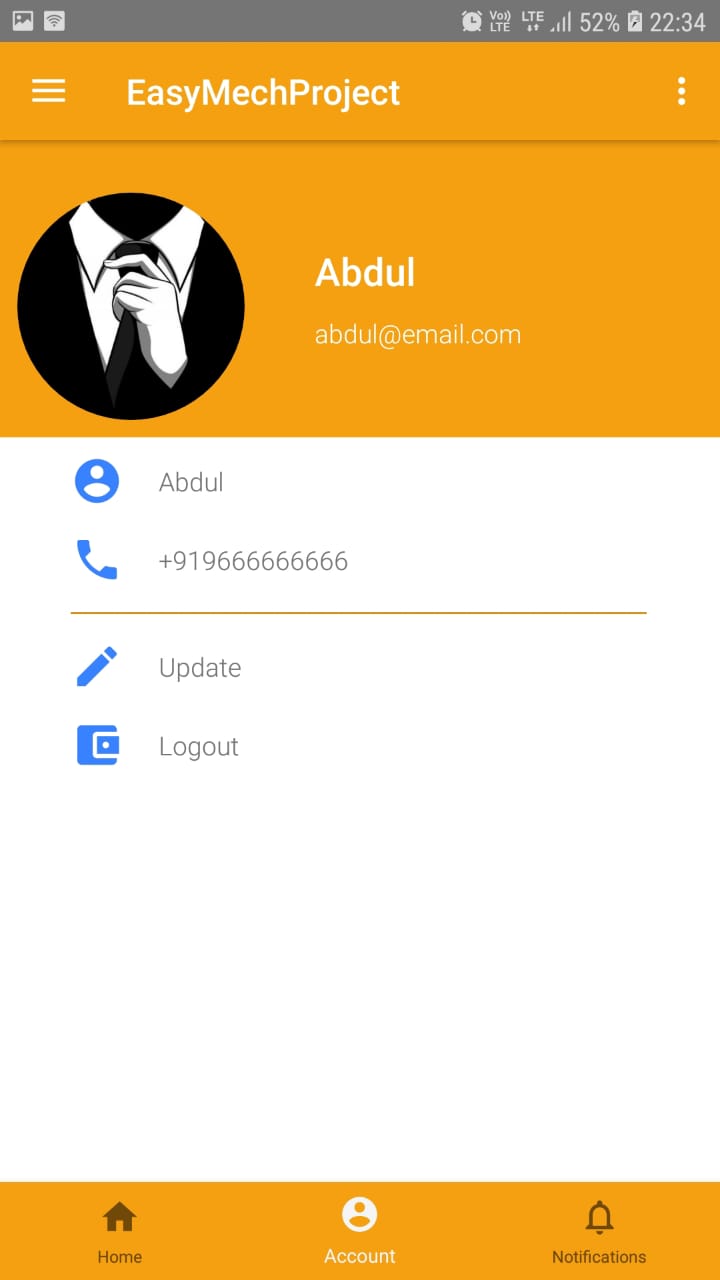
**Mechanics activity**



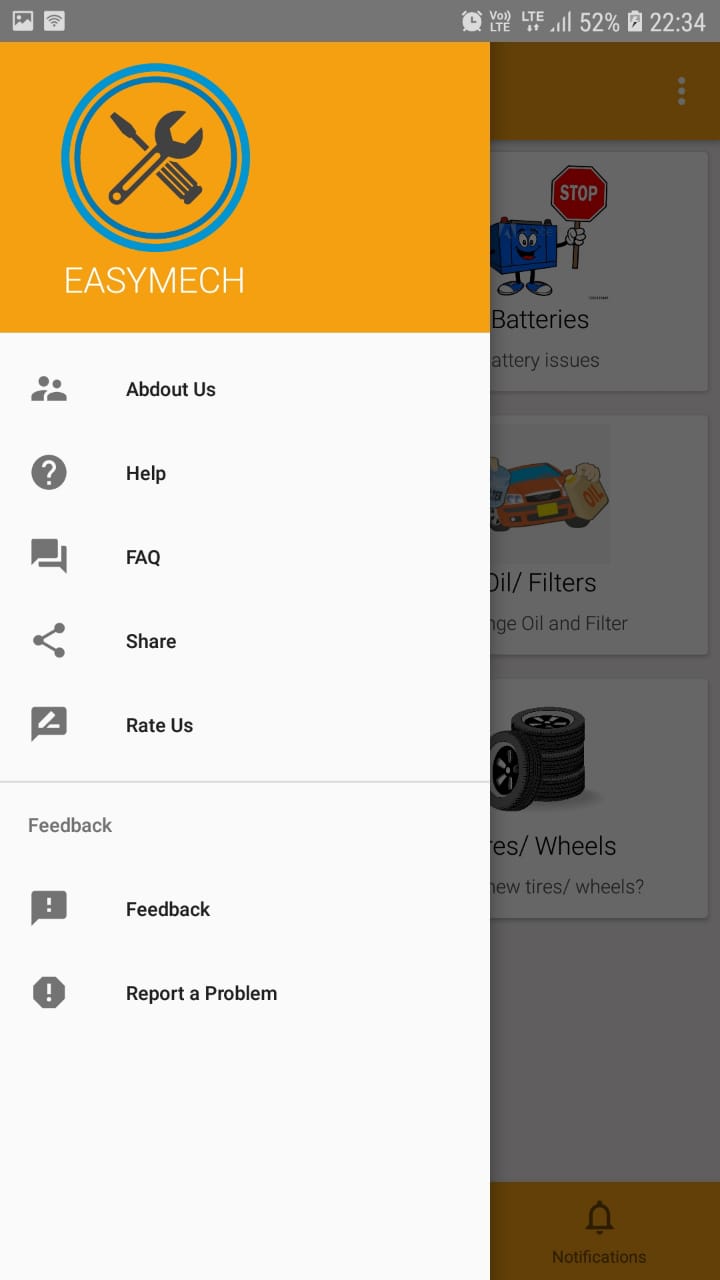
**Appointment activity**



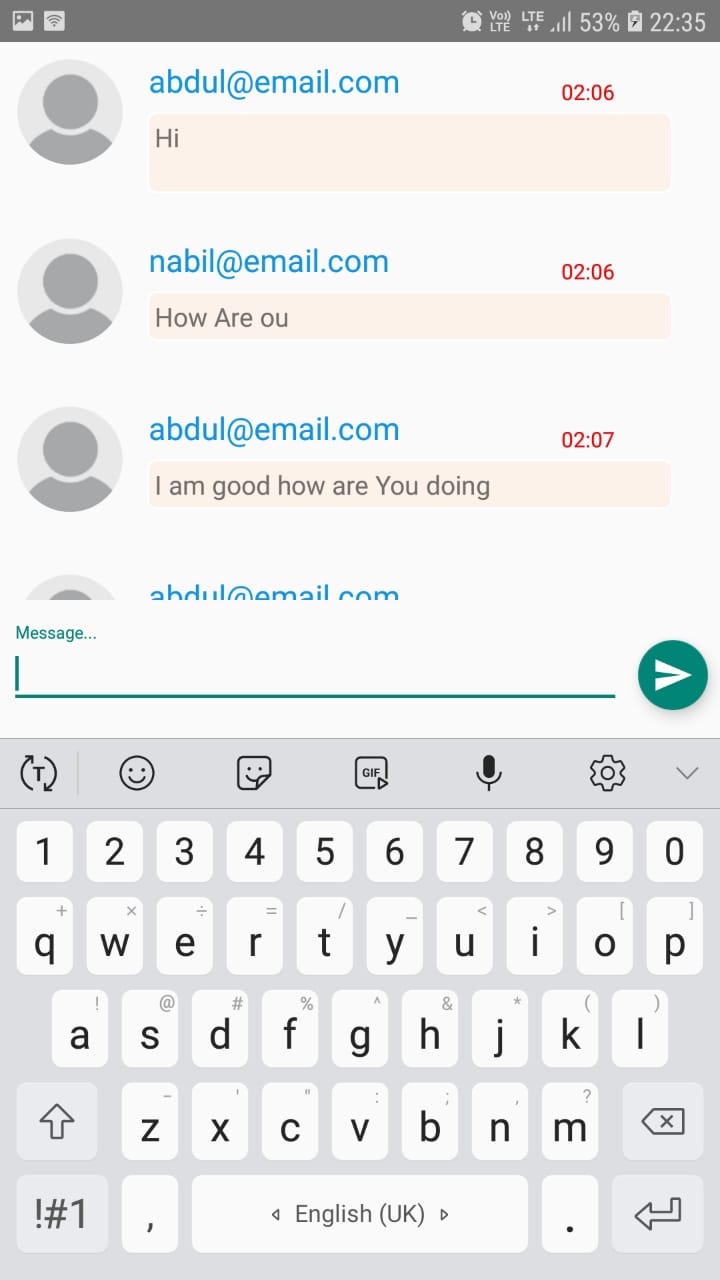
**Profile Activity**



**Navigation Drawer**



**Chat Activity**



**FUTURE ENHANCEMENT**

* Adding live location of the mechanic and the user where the user can see all the mechanics in the nearby areas. Similarly, the mechanic can see the user location, which the mechanic can reach easily
* Adding more services for the user such as buying or selling the vehicle etc.
* Adding online payment for the transaction between user and the service centre.

**CONCLUSION**

The primary goal of the developing “EasyMech” app is to provide best services to the users for vehicles. It can also maximize the number of customers for the mechanics. Because mechanics is not limited to his working hours, the customer can make an appointment any time and the appointment can be confirmed according to the availability of the mechanic during a particular time.

We would like to conclude that we have been quite successful in achieving the goals that we had set initially while developing “EasyMech”.

We also state that the future enhancements can be made to the application to further enhance the overall performance of the app.

REFERENCES

<https://www.developer.android.google.com>

<https://www.youtube.com>

<https://www.firebase.com/docs>

<https://www.stackoverflow.com>

<https://www.github.com>

<https://www.materialdesign.com>

<https://www.google.com>

<https://www.wikipedia.org/>