MINI PROJECT

(2020-21)

“DrFit - Fitness Mobile Application”

Mid-Term Project Report

Institute of Engineering & Technology

Submitted By -

Harsh Tripathi (191500320)

Harshil Gupta (191500321)

Ishika Agarwal (191500357)

Under the Supervision Of

Md. Farmanul Haque

Technical Trainer

Department of Computer Engineering & Applications

Department of Computer Engineering and Applications

GLA University, 17 km. Stone NH#2, Mathura-Delhi Road,

Chaumuha, Mathura – 281406 U.P (India)

Declaration

I/we hereby declare that the work which is being presented in the Bachelor of technology. Project **"DR.FIT App"**, in partial fulfillment of the requirements for the award of the ***Bachelor of Technology*** in Computer Science and Engineering and submitted to the Department of Computer Engineering and Applications of GLA University, Mathura, is an authentic record of my/our own work carried under the supervision of **Md. Farmanul Haque, Technical Trainer, Dept. of CEA,GLA University.**

The contents of this project report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree.

Sign: *Harsh Tripathi* **Name of Candidate**: Harsh Tripathi
University Roll No.: 191500320

Sign: *Harshil Gupta* **Name of Candidate**: Harshil Gupta
University Roll No.:191500321

Sign: *Ishika Agarwal*
Name of Candidate: Ishika Agarwal
University Roll No.:191500357

Department of Computer Engineering and Applications

GLA University, 17 km. Stone NH#2, Mathura-Delhi Road,

Chaumuha, Mathura – 281406 U.P (India)

Certificate

This is to certify that the project entitled "**DrFit App**", carried out in Mini Project – I Lab, is a bonafide work by Harsh Tripathi, Harshil Gupta, and Ishika Agarwal and is submitted in partial fulfillment of the requirements for the award of the degree Bachelor of Technology (Computer Science & Engineering).

Signature of Supervisor:

Name of Supervisor: Md. Farmanul Haque

Date:

Training Certificates

- **Harsh Tripathi**

udemy

Certificate no: UC-a8d61649-9f98-4aa5-a5a2-63a01e356411
Certificate url: ude.my/UC-a8d61649-9f98-4aa5-a5a2-63a01e356411
Reference Number: 0004

CERTIFICATE OF COMPLETION

Flutter & Dart - The Complete Guide [2021 Edition]

Instructors **Academind by Maximilian Schwarzmüller, Maximilian Schwarzmüller**

Harsh Tripathi

Date **Oct. 28, 2021**
Length **42.5 total hours**

- **Harshil Gupta**

udemy

Certificate no: UC-16807157-4159-4008-9b2a-a4125f0e2a2d
Certificate url: ude.my/UC-16807157-4159-4008-9b2a-a4125f0e2a2d
Reference Number: 0004

CERTIFICATE OF COMPLETION

The Complete 2021 Flutter Development Bootcamp with Dart

Instructors **Dr. Angela Yu**

Harshil Gupta

Date **Nov. 21, 2021**
Length **29 total hours**

Department of Computer Engineering and Applications

GLA University, 17 km. Stone NH#2, Mathura-Delhi Road,

Chaumuha, Mathura – 281406 U.P (India)

ACKNOWLEDGEMENT

Presenting the ascribed project paper report in this very simple and official form, we would like to place my deep gratitude to GLA University for providing us the instructor Md Farmanul Haque, our technical trainer and supervisor.

He has been helping us since Day 1 in this project. He provided us with the roadmap, the basic guidelines explaining how to work on the project. He has been conducting regular meetings to check the progress of the project and providing us with the resources related to the project. Without his help, we wouldn't have been able to complete this project.

And last but not least we would like to thank our dear parents for helping us to grab this opportunity to get trained and also my colleagues who helped me find resources during the training.

Thanking You

Sign: *Harsh Tripathi* **Name of Candidate**: Harsh Tripathi
University Roll No.: 191500320

Sign: *Harshil Gupta* **Name of Candidate**: Harshil Gupta
University Roll No.:191500321

Sign: *Ishika Agarwal*
Name of Candidate: Ishika Agarwal
University Roll No.:191500357

ABSTRACT

In this project, we are creating a mobile application, basically, a Fitness App which we have named DrFit. This application will provide us a platform to access anything related to fitness at the ease of our fingertips. All the users will be having their separate accounts on this app which will be connected to their email id. **DR.FIT** is a mobile application that helps users to plan exercise according to days, chat with the instructor, and also they can buy fitness products according to their needs. We will also be adding yoga and meditation features by which users can plan their meditation and yoga according to desired days. Exercises, yoga, and meditation are well planned in this application. It goes from beginner to advanced levels and the user will not feel this transition gap and can do it smoothly.

The Android App ecosystem is diverse and is changing people's lives all over the world. Android users are expected to increase because of the advanced changes of the operating system and the way it deals with issues and compatibility with other mobile devices. Furthermore, designing solutions for the problems that we may face in the future is essential. Like this application definitely stands the need of students at any time at their fingertips without any barrier of place**.**

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER-1 INTRODUCTION

1.1 CONTEXT

This Android Application "**Dr.Fit**" has been submitted in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering at GLA University, Mathura supervised by Md. Farmanul Haque. This project has been completed approximately three months and has been executed in modules, meetings have been organized to check the progress of the work and for instructions and guidelines.

1.2 MOTIVATION

In recent years, we have realized the importance of virtual fitness training and how important it is for us to have our resources online.

In the century we are living in, the world is progressing at a really great pace, a lot of technologies come up every single day. To keep up with technology is also important to survive in this world of digitalization and learning. Along with this, we need to have a place to keep the resources for areas of our interest so we thought of developing an app that could provide us with virtual fitness training as well as a platform where we could keep the exercises we like.

The fitness app you choose shouldn't be a standalone program – it should work with your other phone apps to give you as large a view of your health as possible. "We know that fitness and health is a combination of diet, exercise, sleep and more," Higgins says. "Having apps that work together and talk to each other to track all of those factors is imperative.

1.3 OBJECTIVE

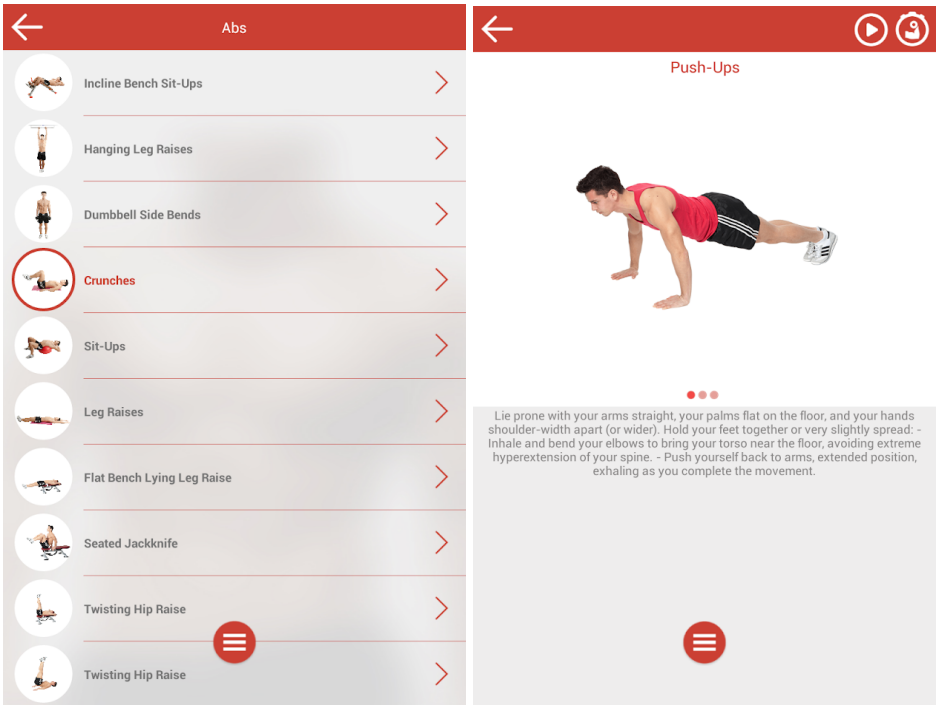
The main objective of this application is to create a Book Finder app named “Bookopedia” which will have a lot of e-books and a space to keep up the books one wants to read. There will be a facility to search any book one wishes to read by the use of any keyword like the author

name, book name, the name of the subject. After the search there will be a list of related books and one can view and read more about the details of the book and can further purchase it.

This application can be used at a variety of places, at education hubs and have its significance. The goal of the app was to provide a way to the learners and users to get all the books they desire to read at a particular location rather than randomly surfing the Internet.

1.4 EXISTING SYSTEM

In present there are many existing fitness apps which are in use.



(a) (b)

Figure-1: Existing System

1.5 SOURCES

The source of our project (including all the project work, documentations and presentations) will is available at the following link

https://github.com/harshilgupta-dev/DR.Fit.

CHAPTER -2

SOFTWARE REQUIREMENT ANALYSIS

2.1 IMPACT OF FITNESS ON DAILY LIFE

Exercise strengthens your heart and improves your circulation. The increased blood flow raises the oxygen levels in your body. This helps lower your risk of heart diseases such as high cholesterol, coronary artery disease, and heart attack. Regular exercise can also lower your blood pressure and triglyceride levels.

Meditation can give you a sense of calm, peace and balance that can benefit both your emotional well-being and your overall health.

And these benefits don't end when your meditation session ends. Meditation can help carry you more calmly through your day and may help you manage symptoms of certain medical conditions.
Meditation might also be useful if you have a medical condition, especially one that may be worsened by stress.

While a growing body of scientific research supports the health benefits of meditation, some researchers believe it's not yet possible to draw conclusions about the possible benefits of meditation.

2.2 PROBLEM STATEMENT

The Fitness App “DR.Fit” is an Android Application which will provide the user with instructions and examples of one or more types of exercise, physical activity, nutritional programs, or some other fitness topic.

Along the side, for the users a chat space is being provided for them to chat with the desired instructor they like in the “chat section”. As a help to the users, there will be a side bar containing the basic details and personal information about the user. One another feature that our app holds is calculating BMI Index each time the user updates his information that will be displayed and tell them if he/she is fit or not.

This app is an exercising app with all the facilities a user desires and with the proper User Interface as well.

2.3 HARDWARE AND SOFTWARE REQUIREMENTS

Hardware Requirement

- Processor :intel i5
- Operating System :Any Operating System
- RAM : 8 GB (or higher)
- Hard disk : 256GB

Software Requirement

- Software used: Android Studio / VS Code
- Language used : Dart, Flutter (Framework)
- Database: Firebase
- User Interface Design : Android Application

2.4 MODULES AND FUNCTIONALITIES

▪ **Splash Screen**: The first screen with which the user interacts will be this screen containing the logo and the app name .This will disappear within 3 seconds after the app is displayed.

▪ **Login Page**: This page is for the user to login in to the application using google sign-in.

▪ **Home Page:** This page is the root of the whole application. This entire app revolves around this screen. This screen would allow users to choose various types of fitness activities and will take them to that desired page or screen. This page will also contain some personal information about the user. It connects the various activities together like a sidebar on which the profile, the dashboard, the chat screen, the exercise screen are linked and on clicking on each you can visit the pages.

▪ **Exercise Page:** This page will contain all the exercises category, like, Abs, triceps, etc. Also for every category there is a list of exercises and their description.

▪ **Chat Page:** This page comes into picture when the user wants to chat with the particular instructor provided by the app itself. In this case this page will connect the user to the instructor via text.

▪ **Profile**: This page will contain all the user details that the user entered while creating the account on the app. The user can update and make changes to all this information as desired.

▪ **FAQ Pages:** This page contains some of the questions that might arise in the mind of the users while using the app and to answer those, these answers are pre-written.

▪ **Logout page**: Then is this last panel for the users to sign out from the account. As soon as the users sign out they are brought back to the login page.

2.5 DR.Fit ON ANDROID APPLICATION

DR.fit is actually a fitness based hub. Fitness has become one of the most important basic needs of a human being, but nowadays people have less time for their fitness activities. Usually they ignore spending time on their fitness. These days Home workout has become one of the trendy things as it saves time of the user as well as provides an ambience environment so that people can concentrate.

This app will give users new ways to show their love for exercise, check them online, keep track of what they do, share their experience with instructors, and, of course, discover new exercises and yoga to do.

CHAPTER- 3 SOFTWARE DESIGN

3.1 USE-CASE DIAGRAM:

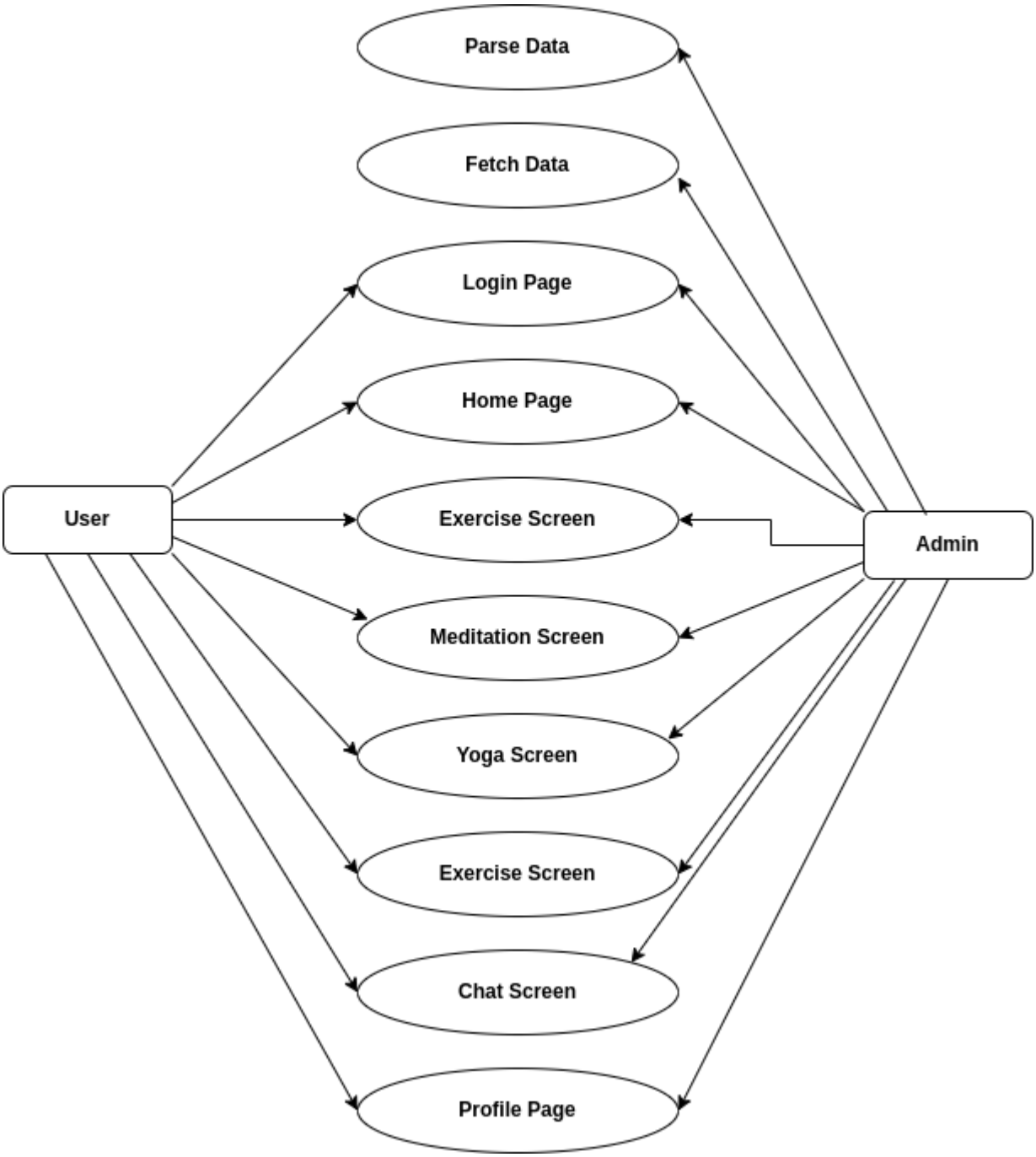


Figure-2: Use–Case Diagram

So the above diagram represents the point of view of the user and the developer and the arrows to each module show the interactivity of the person.

The user will first be required to login so will interact with the “login module” and sign in with

his google account. The next user will land into the homepage where there will be a sidedrawer , bottom navigation bar through which users can jump into the profile, chat, exercise, meditation, yoga, products pages.

For the developer he can connect with each and every module mentioned in the use case diagram. Apart from the modules mentioned in the use case diagram there are modules like profile, sign out, FAQ and about us section that every registered user can access.

CHAPTER-4
TECHNOLOGY USED

4.1 Flutter

We are using Flutter to develop “Dr.Fit”. The core concept of the Flutter framework is In Flutter, Everything is a widget. Widgets are basically user interface components used to create the user interface of the application.

In Flutter, the application is itself a widget. The application is the top- level widget and its UI is built using one or more children (widgets), which again build using its children widgets. This composability feature helps us to create a user interface of any complexity. Flutter is an open-source UI software development kit created by Google. It is used to develop cross platform applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia, and the web from a single codebase.
First described in 2015, Flutter was released in May 2017.

The core concept of the Flutter framework is In Flutter, Everything is a widget. Widgets are basically user interface components used to create the user interface of the application.

In Flutter, the application is itself a widget. The application is the top- level widget and its UI is built using one or more children (widgets), which again build using its children widgets. This

composability feature helps us to create a user interface of any complexity.

4.2 Types of Application

• **Native Apps:** An executable program coded in the machine language of the hardware platform it is running in. **Native applications** are compiled into the machine language of that CPU. For example, **Windows** and Mac executable **apps** are in x86 machine language, while **mobile apps** are ARM based. Native apps are the most common. They're coded in a specific language like Swift for **iOS** or Java for Android. A popular example is WhatsApp.

• **Web Apps:** are accessed via the internet browser and will adapt to whichever device you're viewing them on. They are not native to a particular system, and don't need to be downloaded or installed. Due to their responsive nature, they do indeed look and function a lot like mobile apps — and this is where the confusion arises.

• **Hybrid Apps:** Hybrid apps are deployed in a native container that uses a mobile Web View object. When the app is used, this object displays web content thanks to the use of web technologies (CSS, JavaScript, HTML, HTML5).It is in fact displaying web pages from a desktop website that are adapted to a Web View display. The web content can either be displayed as soon as the app is opened or for certain parts of the app only i.e. for the purchase funnel. In order to access a device's hardware features (accelerometer, camera, contacts…) for which the native apps are installed, it is possible to include native elements of each platform's user interfaces (iOS, Android): native code will be used to access the specific features in order to create a seamless user experience. Hybrid apps can also rely on platforms that offer JavaScript APIs if those functionalities are called within a Web View

4.3 VERSION OF ANDROID

Codename	Version	API level/NDK release
Android12	12	API level 31

Android11	11	API level 30
Android10	10	API level 29
Pie	9	API level 28
Oreo	8.1.0	API level 27
Oreo	8.0.0	API level 26
Nougat	7.1	API level 25
Nougat	7.0	API level 24
Marshmallow	6.0	API level 23
Lollipop	5.1	API level 22
Lollipop	5.0	API level 21
KitKat	4.4 - 4.4.4	API level 19
Jelly Bean	4.3.x	API level 18
Jelly Bean	4.2.x	API level 17
Jelly Bean	4.1.x	API level 16

Ice Cream Sandwich	4.0.3 - 4.0.4	API level 15, NDK 8
Ice Cream Sandwich	4.0.1 - 4.0.2	API level 14, NDK 7
Honeycomb	3.2.x	API level 13
Honeycomb	3.1	API level 12, NDK 6
Honeycomb	3.0	API level 11
Gingerbread	2.3.3 - 2.3.7	API level 10
Gingerbread	2.3 - 2.3.2	API level 9, NDK 5
Froyo	2.2.x	API level 8, NDK 4
Eclair	2.1	API level 7, NDK 3
Eclair	2.0.1	API level 6
Eclair	2.0	API level 5
Donut	1.6	API level 4, NDK 2
Cupcake	1.5	API level 3, NDK 1
(no codename)	1.1	API level 2

(no codename)	1.0	API level 1

Each year Android releases a new version with better features, better security and better User Interface experience and a new symbol. Here is the table of list of versions.

Figure-5: Android Kitkat

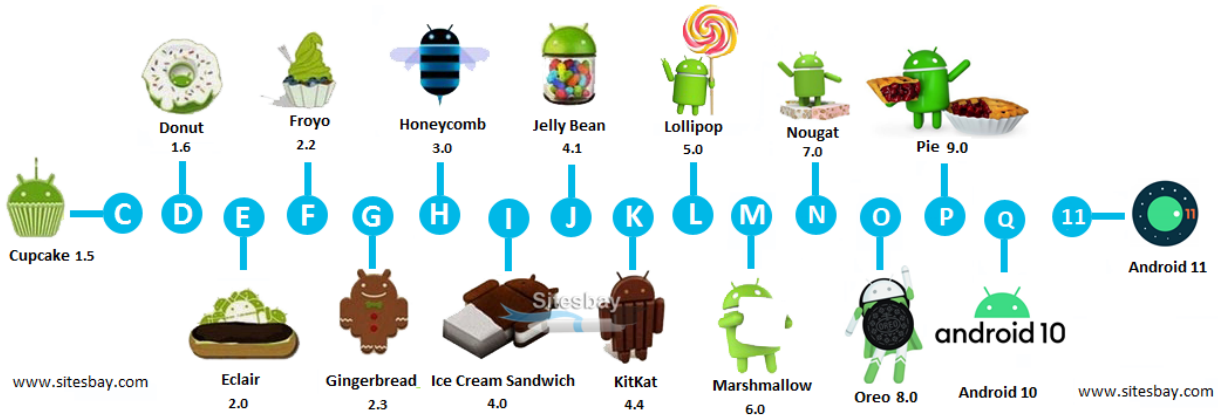


Table -1: Versions of Android

4.4 TOOLS AND LANGUAGES

Tools used to build the Android App are:-

• **Android Studio**: Android Studio is an environment that help us create and edit Android applications. It is the official IDE for Android App Development. It has intelliJ's powerful code editor and developer tools and various features that enhance productivity while developing apps.

• **Visual Studio Code**: Visual Studio Code is a streamlined code editor with support for development operations like debugging, task running, and version control. It aims to provide just the tools a developer needs for a quick code-build-debug cycle and leaves more complex workflows to fuller featured IDEs, such as Visual Studio IDE.

• **Software Development Kit (SDK)**: Android Studio requires a collection of libraries and data therefore SDK is mandatory.

Languages used in building an Android Application are classified as per the Front End and Back End. For designing the Front End of an application we have used XML and for designing the Back End we have used Kotlin.

• **Dart**: The Dart language is type safe; it uses static type checking to ensure that a variable's value *always* matches the variable's static type. Sometimes, this is referred to as sound typing. Although types are mandatory, type annotations are optional because of type inference. The Dart typing system is also flexible, allowing the use of a dynamic type combined with runtime checks, which can be useful during experimentation or for code that needs to be especially dynamic.
Dart offers sound null safety, meaning that values can't be null unless you say they can be. With sound null safety, Dart can protect you from null exceptions at runtime through static code analysis. Unlike many other null-safe languages, when Dart determines that a variable is non-nullable, that variable is *always* non-nullable. If you inspect your running code in the debugger, you'll see that non-nullability is retained at runtime (hence *sound* null safety).

4.5 BASIC TERMINOLOGY

• **Widget:** Widgets are the fundamental UI unit in Flutter. They represent an interactive element of a user interface such as a button or text field. Widget classes have their own build function to create a tree structure for drawing into the parent's Build Context, which is used to manage layout calculations and coordinate with native platform elements at runtime.

• **StatelessWidget**: Basically a stateless widget is one that doesn't store any internal data about the user's interaction with it. It may read its information from props passed in by parent widgets or other sources like local or session storage etc., but does not maintain its own separate set of values for things such as text content entered into an input field.

• **StatefulWidget**: A stateful widget is one that stores data between invocations of its build method, and updates the UI in response to those changes.

• **Emulator**: An emulator is an Android virtual device through which you can select the target Android version or platform to run and test your developed application.

• **Manifest file**: Manifest file acts as a metadata for every application. This file contains all the essential information about the application like app icon, app name, launcher activity, and required permissions etc.

• **MaterialApp:** Material apps are a set of predefined widgets which implement Google's material design guidelines on both mobile and desktop platforms. The Material App provides prebuilt implementations for common user interface patterns such as lists, grids, menus etc., so you can focus more time on building your app rather than reinventing the wheel!

• **Scaffold:** Scaffolds let you quickly add screens with basic content into your application without writing any code at all by using a templating engine called Stencils. You will need this when developing multiple screen applications (commonly referred to as multi-page applications).

APK: Short for "Android application package." The extension used in Android app installation files (e.g., app.apk). Similar in nature to an EXE file on Windows.

• **SDK:** Short for "Software Development Kit." As it pertains to Android, the SDK is a set of tools such as code libraries, a debugger, and a handset emulator that can be run on Windows, Mac, or Linux to facilitate the creation of Android apps by developers. While the SDK is generally intended for use by developers, end users can install the software on their home computer to execute ADB and Fast boot commands.

• **App Bar**: The app bar is an important design element, usually at the top of each screen in an app that provides a consistent familiar look between Android apps. It is used to provide better user interaction and experience by supporting easy navigation through tabs and drop-down lists.

• **Navigation bar**: Navigation Drawer is a sliding left menu that is used to display the important links in the application. Navigation drawer makes it easy to navigate to and fro between those links. It's not visible by default and it needs to open either by sliding from left or clicking its icon in the App Bar.

• **BuiltContext**: Build Context refers to an object that provides information to the widget tree as it renders. For example, layout calculations and coordinates with native platform elements at runtime are done based on Build Context objects.

• **Firebase** is a Backend-as-a-Service (Baas). It provides developers with a variety of tools and services to help them develop quality apps, grow their user base, and earn profit. It is built on Google's infrastructure. Firebase is categorized as a NoSQL database program,

which stores data in JSON-like documents. Firebase has three core services: a real-time database, user authentication and hosting. With the Firebase iOS SDK, you can use these services to create apps without writing any server code.

JSON stands for JavaScript Object Notation. It is an independent data exchange format and is the best alternative for XML. JSON is used for data interchange (posting and retrieving) from the server. Hence knowing the syntax and it's usability is important. JSON is the best alternative for XML and its more readable by human

CHAPTER -5

IMPLEMENTATION AND USER INTERFACE

Creating an app concept design with screen sketches and functional flow diagrams is the best way to communicate your vision to the mobile app developer. Making the concept clear to the developer is probably the most important factor in successful mobile app development. Yet it is one of the most common problems or obstacles in a mobile app development outsourcing project.

No matter what the marketing and profit goals are or if you are outsourcing an app for your personal use, you need to fully design and document the app concept if you expect a programmer to make your vision a reality. Developers are not mind readers and even descriptions given during conversations can be very fleeting or interpreted differently. Fully documenting your concept, therefore, leaves little to chance. The two most important things to do are: A) make a comprehensive description of how the app works and what it does (functionality) and B) create a comprehensive description of what the user sees and does (look and feel).

5.1 Implementation of the Dr.Fit:

Implementation of Bookopedia is taking place in various phases. Firstly we build the login interface then login page and then make various layouts for the supporting features.

5.1.1 Step to be followed to develop the app:

1. Firstly we created the splash screen with animated text and images.

2. After that we create a login phase which consists of various phases that allows users to login into the app.•

• **For authenticating the user we have used firebase authentication**.

3. Now, we are going to create the homepage of our application which consists of various widgets.

4. Now we have created various activities like Exercise List, Exercise Description and many more.

5. Now we add data to the Exercise screen, meditation screen and many more.

6. Now we implemented the chat app where the user and instructor can chat with each other. It uses the firebase firestore database.

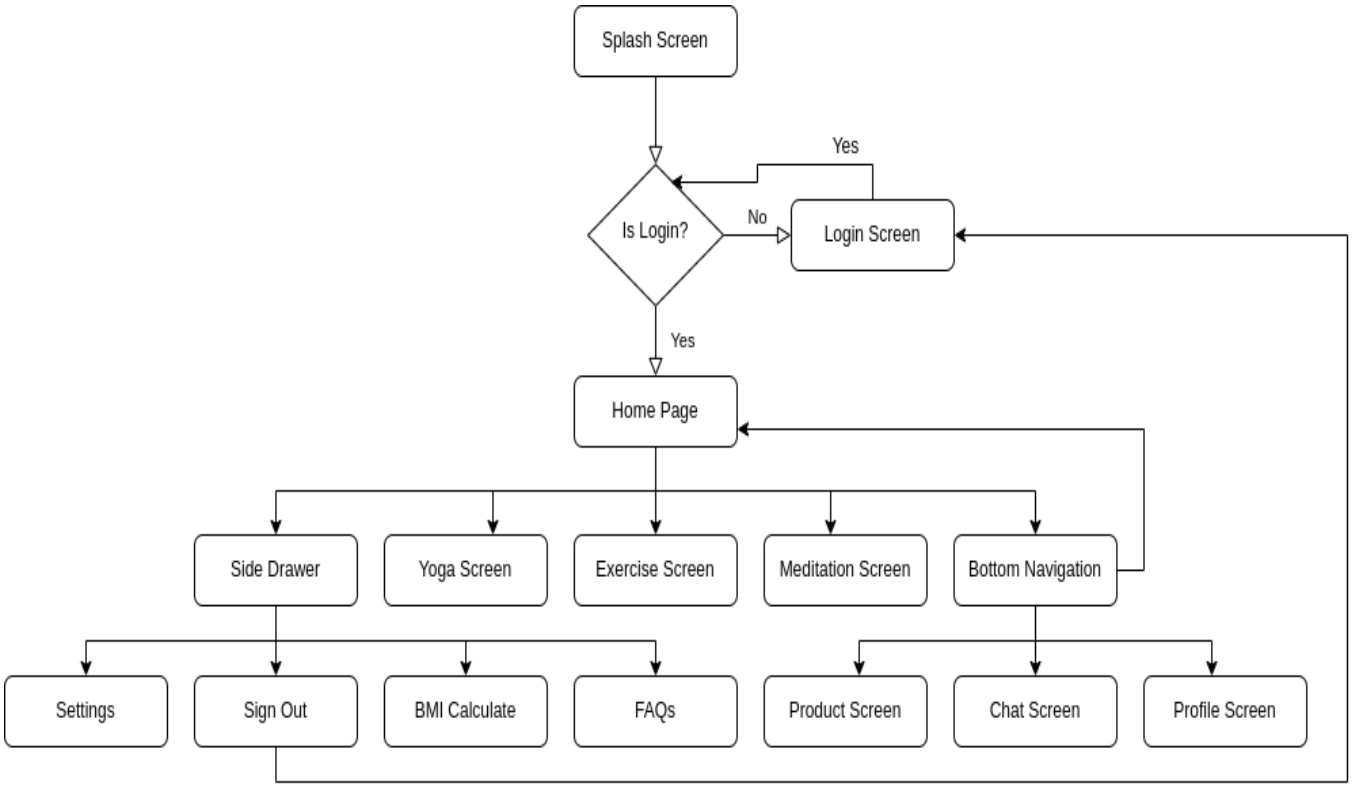


Figure-6: Flowchart for User

5.1.2 Step to be followed by the user

1. Firstly, we have built a splash activity to start the application.

2. Then, we have the Login activity using google. We first checks whether the user is already logged in or not, if the user is already logged in then we navigate to the Homepage otherwise Login Page.

3. We authenticate and store the user information from the Firebase authentication.

4. After that, we made a Homepage where we can go to any screen like drawer, exercise, meditation, profile, chat, etc .

5. We have a side drawer, it has Settings, Sign Out, BMI calculator, and FAQs sections.

6. Also we have a Bottom Navigation bar from there users can navigate to Chat screen, Products screen and Profile section.

7. Lastly we can sign out the profile from the side drawer on the homepage of the app and then it navigates to the login page.

5.2 User Interface

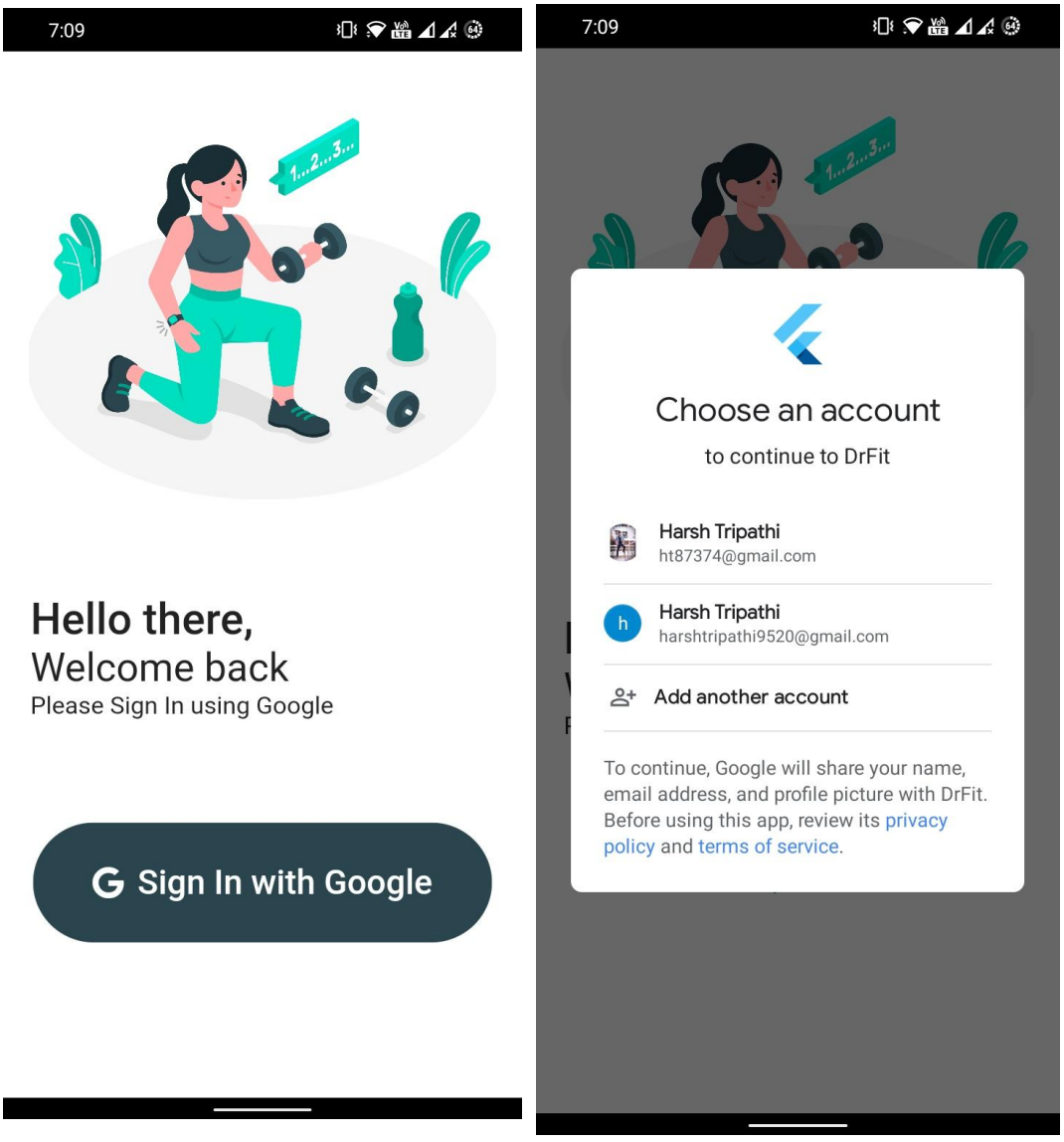


Figure 7 Login Screen (a) (b)

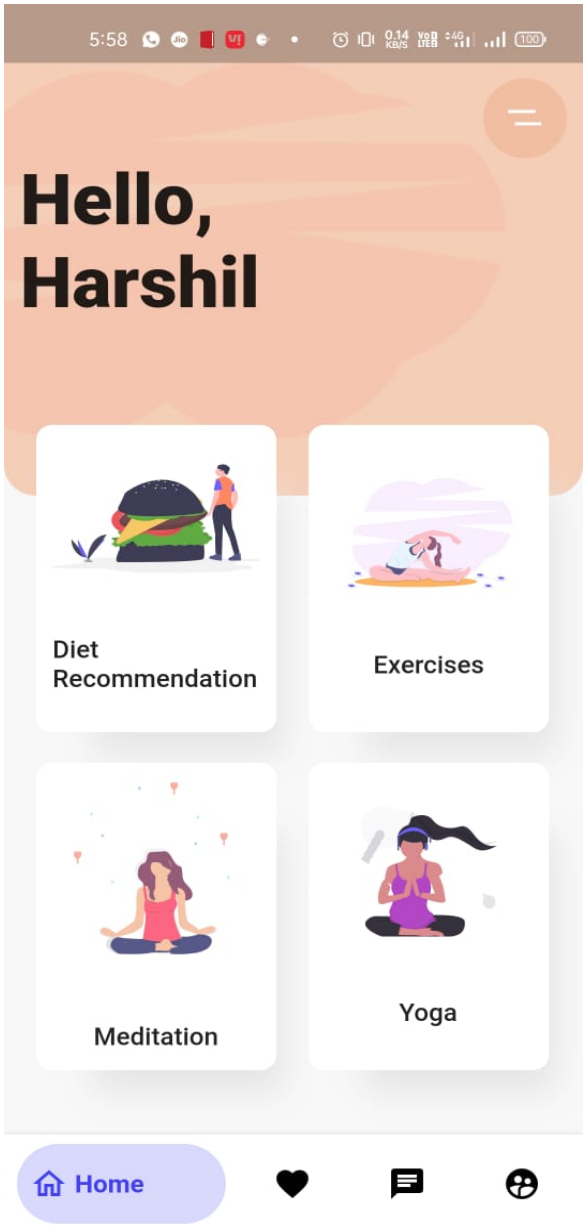
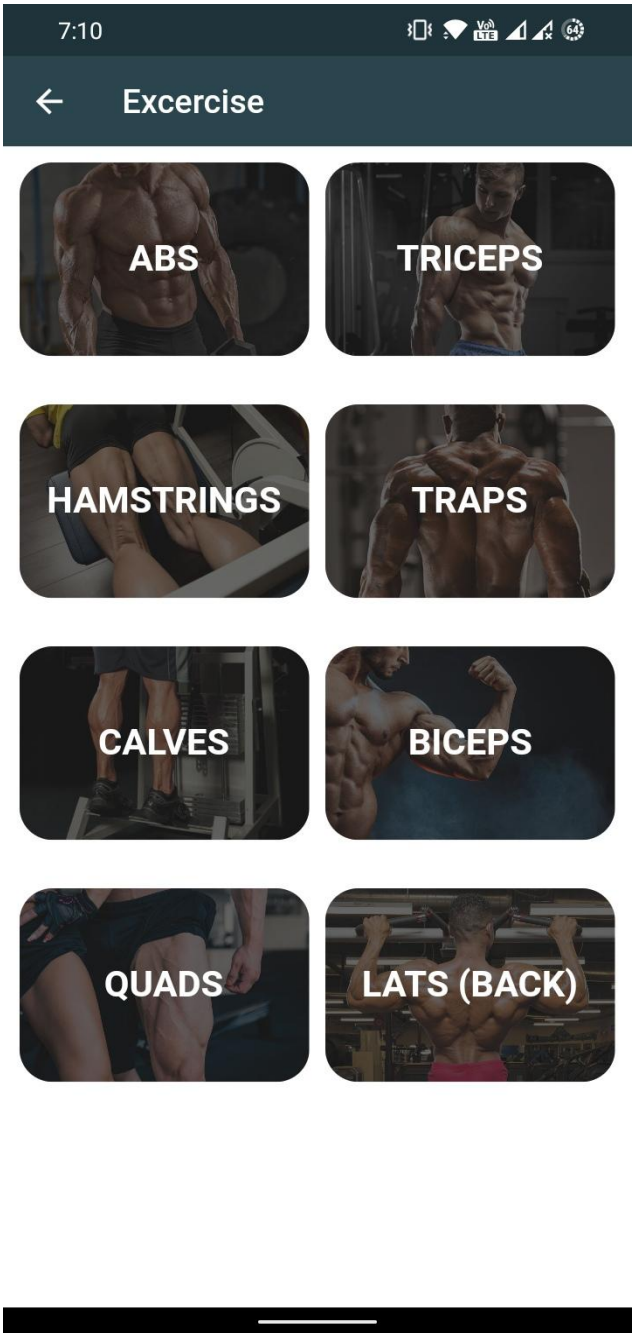
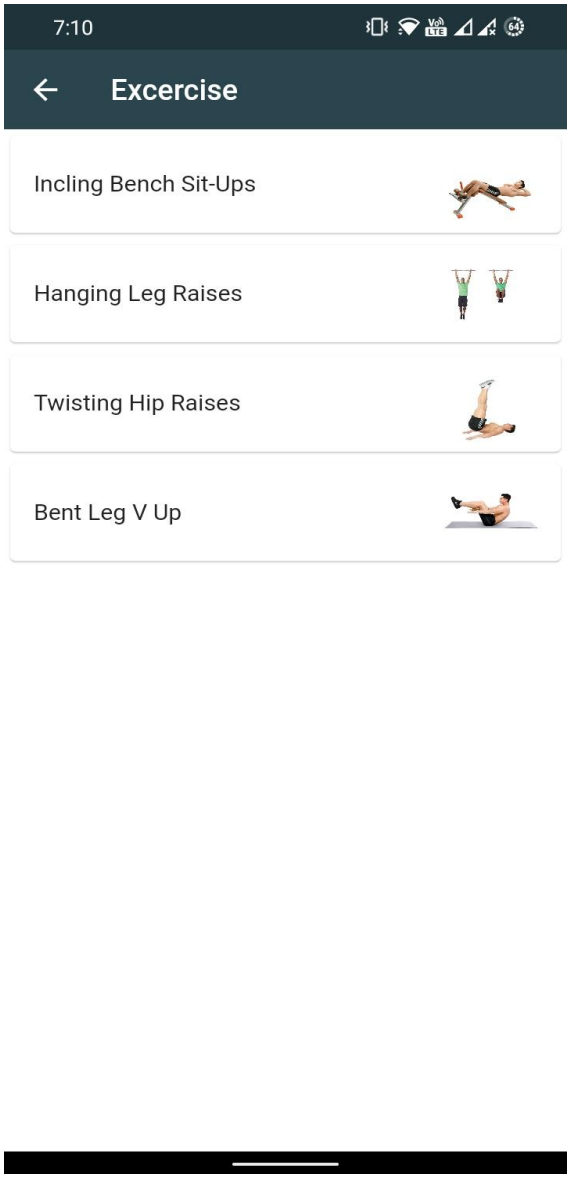


Figure-7 Home Page


7:10
Excercise
ABS
TRICEPS
HAMSTRINGS
TRAPS
CALVES
BICEPS
QUADS
LATS (BACK)



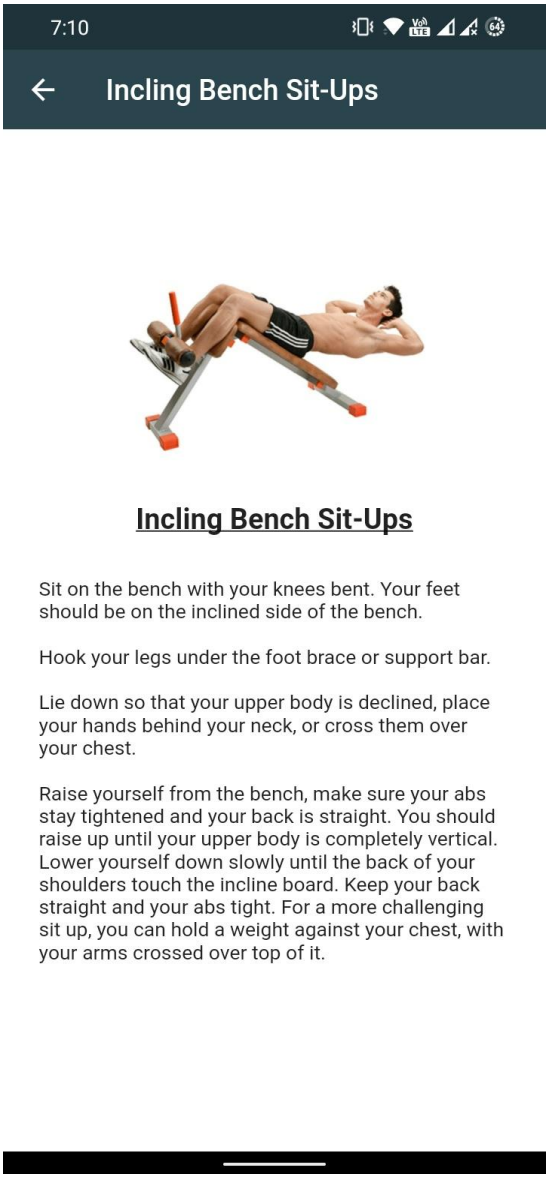


Figure 8 Exercise Screen (a) (b) (c)

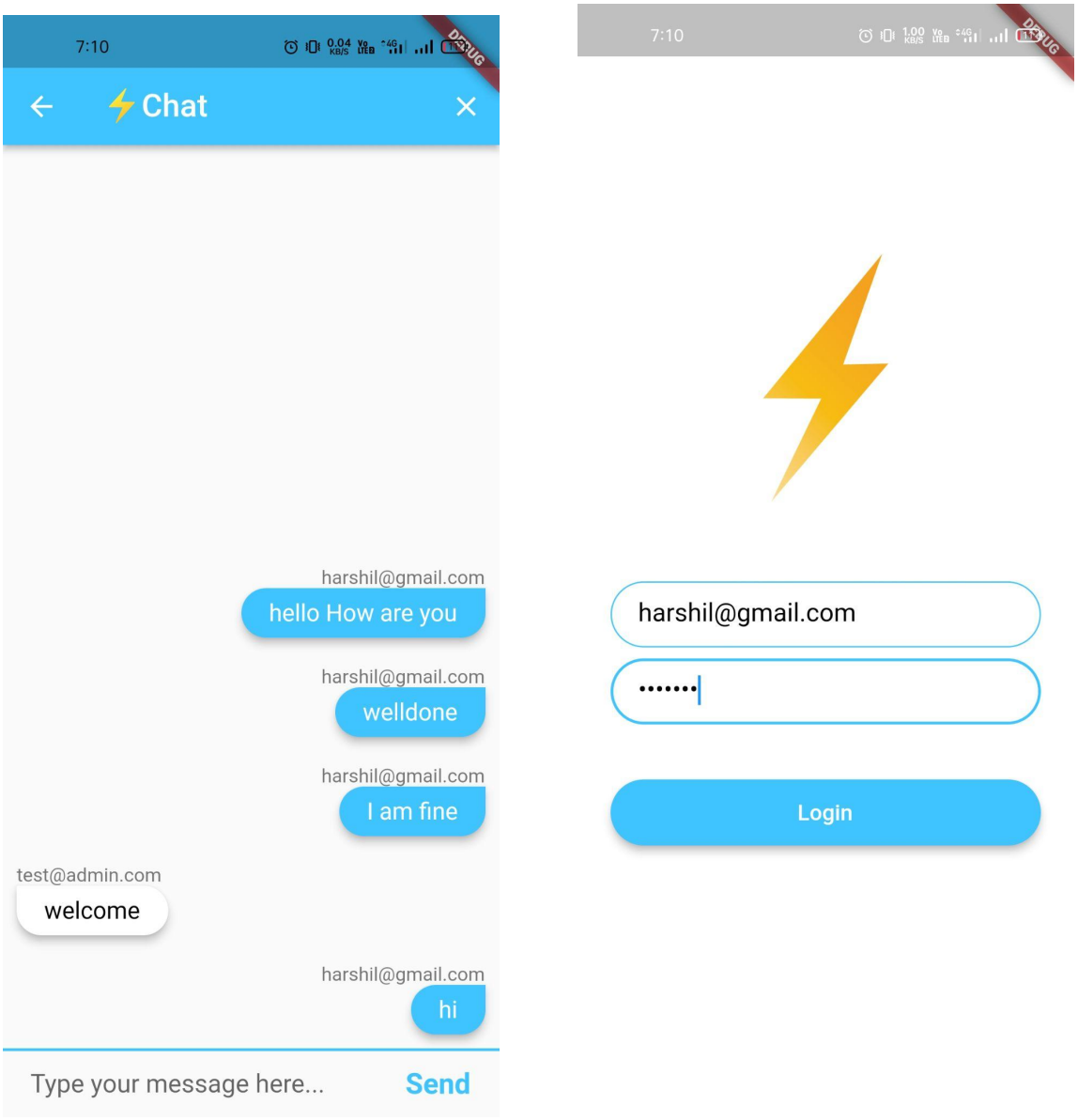


Figure 9 Chat Screen

REFERENCES

1. Introduction to Flutter:

https://docs.flutter.dev/

2 Training:

https://www.udemy.com/course/learn-flutter-dart-to-build-ios-android-apps/

https://www.udemy.com/course/flutter-bootcamp-with-dart/

3. For rectifying the error :

https://stackoverflow.com/

https://www.youtube.com/