Chapter 1

INTRODUCTION

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps, such as:

- A flexible Gradle-based build system
- A fast and feature-rich emulator
- A unified environment where you can develop for all Android devices
- Apply Changes to push code and resource changes to your running app without restarting your app
- Code templates and GitHub integration to help you build common app features and import sample code
- Extensive testing tools and frameworks
- Lint tools to catch performance, usability, version compatibility, and other problems
- C++ and NDK support
- Built-in support for Google Cloud Platform, making it easy to integrate Google loud Messaging and App Engine.

1.1Overview of Mobile Application Development

Mobile application development is the process of creating software applications specifically designed to run on mobile devices such as smartphones and tablets. In recent years, mobile applications have become an integral part of our daily lives, providing us with convenient access to information, services, and entertainment on the go. The field of mobile application development has witnessed significant growth and innovation, driven by advancements in mobile technology and the increasing demand for mobile apps across various industries. Mobile apps are developed for different platforms, including iOS (Apple's operating system) and Android (Google's operating system), each requiring specialized development approaches and tools.

Mobile application development involves multiple stages, starting from conceptualizing and designing the app's user interface to coding and testing its functionality. Developers utilize programming languages, frameworks, and software development kits (SDKs) specific to the targeted platform to create robust and user-friendly mobile applications. The scope of mobile application development is vast and diverse. Applications can be developed for various purposes, such as social networking, e-commerce, gaming, productivity, education, healthcare, and more. Developers need to consider factors like user experience, performance optimization, security, and compatibility across different devices and screen sizes while creating mobile apps. Mobile application development offers immense opportunities for businesses and entrepreneurs to reach their target audience, enhance customer engagement, and drive revenue growth. With the rising popularity of mobile devices and the increasing reliance on apps for everyday tasks, the demand for skilled mobile app developers continues to grow. In this dynamic and ever-evolving field, staying updated with the latest trends, emerging technologies, and user preferences is crucial for successful mobile application development.

The following points highlights the need and importance of Mobile Application Development in today’s world:

- High accessibility with and without internet
- Social Media Integration as a marketing tool
- Provides good security and performance for its applications.
- Useful in interpreting tasks to handle multiple activities.
- Provides efficient user interfaces for interacting with apps and triggering actions that works across a wide range of devices.
- Easy to implement adaptive methods for storing, sharing and retrieving data in android apps.

1.2 Problem Statement

In today's fast-paced and sedentary lifestyle, people are struggling to maintain a healthy level of physical fitness due to various challenges such as lack of motivation, limited access to personalized workout plans, inadequate knowledge about proper nutrition, and difficulty in tracking their progress effectively. This results in a significant portion of the population experiencing health issues, obesity, and a decline in overall well-being. Therefore, there is a pressing need for a comprehensive fitness application that addresses these challenges and empowers individuals to lead healthier lifestyles by providing personalized workout routines, nutritional guidance, and effective progress tracking, all in one convenient platform.Despite the growing popularity of fitness apps, many people still struggle to stay motivated and on track with their fitness goals. This is due to a number of factors, including:

- Lack of variety in exercises
- Lack of information about each exercise
- Poor user interface

1.3 Objective

The objectives of the fitness application mini project are as follows:

- Develop a user-friendly mobile application that provides seamless access to a wide range of gym workouts and yoga exercises.
- Offer clear and concise instructions for each exercise, including a built-in timer and a YouTube link for detailed guidance.
- Implement a BMI calculator to assist users in monitoring their body mass index and understanding their fitness levels.
- Create a collection of carefully selected workout songs to create an engaging and motivating workout environment.
- Provide a section with fitness tips and motivational quotes to inspire and support users in their fitness journey.

1.4 Motivation

The motivation behind the development of the fitness application mini project stems from the growing need for accessible and convenient fitness solutions. In today's fast-paced world, individuals often struggle to find the time and resources to engage in regular exercise. By providing a comprehensive platform that combines gym workouts, yoga exercises, a BMI calculator, workout songs, and inspirational content, the application aims to empower users to take control of their physical well-being. The project seeks to provide users with the necessary tools, guidance, and motivation to achieve their fitness goals, leading to improved overall health and well-being.

Chapter 2

METHODOLOGY

Our program was developed using the Java language. It is employed in the creation of Web applications, game - development, database association, and the production of mobile applications, particularly Android apps. The most popular method for creating high-performance mobile applications is Java app development.

2.1 Android Studio

Android Studio is the official integrated development environment (IDE) for Google's Android OS, 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 or as a subscription-based service in 2020. It is a replacement for the Eclipse Android Development Tools (E-ADT) as the primary IDE for native Android application development.

Android Studio was announced on May 16, 2013 at the Google I/O conference. It was in early access preview stage starting from version 0.1 in May 2013, then entered beta stage starting from version 0.8 which was released in June 2014. The first stable build was released in December 2014, starting from version 1.0. On May 7, 2019, Kotlin replaced Java as Google's preferred language for Android appdevelopment. Java is still supported, as is C++.

A specific feature of the Android Studio is an absence of the possibility to switchautosave feature off. The following features are provided in the current stable version:

- Gradle-based build support
- Android-specific refactoring and quick fixes
- Lint tools to catch performance, usability, version compatibility and other problems
- ProGuard integration and app-signing capabilities
- Template-based wizards to create common Android designs and components
- A rich layout editor that allows users to drag-and-drop UI components, option to preview layouts on multiple screen configurations
- Support for building Android Wear apps

2.1.1 Project Structure

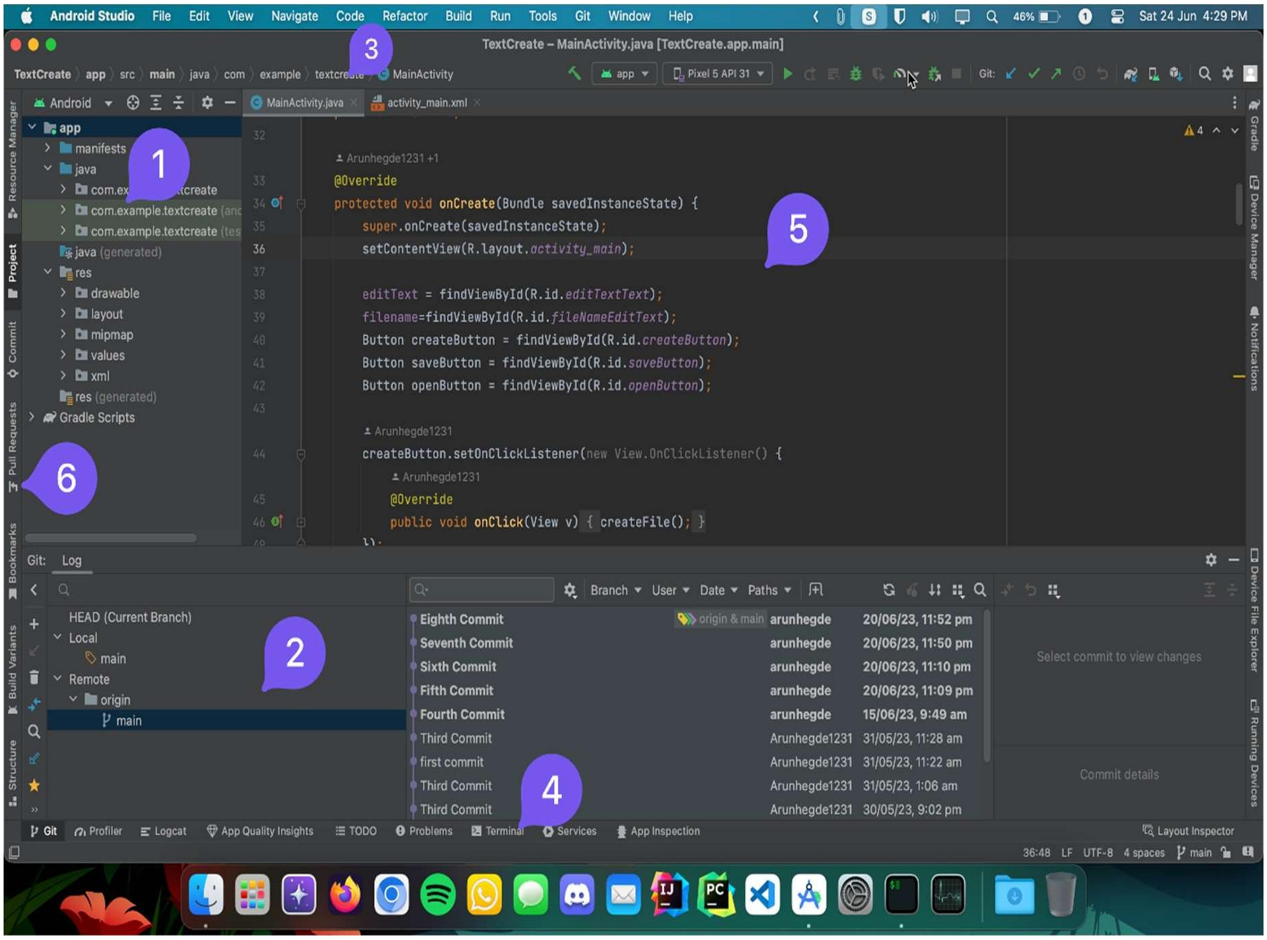
Each project in Android Studio contains one or more modules with source code files andresource files. Types of modules include:

- Android app modules
- Library modules
- Google App Engine modules

Each app module contains the following folders:

- **Manifests**: Contains the AndroidManifest.xml file.
- **Java**: Contains the Java source code files, including JUnit test code.
- **Res**: Contains all non-code resources, such as XML layouts, UI strings, and bitmap images.

2.1.2 User Interface



```
    Arunhegde1231 +1
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        editText = findViewById(R.id.editTextText);
        filename=findViewById(R.id.fileNameEditText);
        Button createButton = findViewById(R.id.createButton);
        Button saveButton = findViewById(R.id.saveButton);
        Button openButton = findViewById(R.id.openButton);

        Arunhegde1231
        createButton.setOnClickListener(new View.OnClickListener() {
            Arunhegde1231
            @Override
            public void onClick(View v) { createFile(); }
        });
```

```
Eighth Commit        origin & main  arunhegde      20/06/23, 11:52 pm
Seventh Commit                      arunhegde      20/06/23, 11:50 pm
Sixth Commit                        arunhegde      20/06/23, 11:10 pm
Fifth Commit                        arunhegde      20/06/23, 11:09 pm
Fourth Commit                       arunhegde      15/06/23, 9:49 am
Third Commit                        Arunhegde1231  31/05/23, 11:28 am
first commit                        Arunhegde1231  31/05/23, 11:22 am
Third Commit                        Arunhegde1231  31/05/23, 1:06 am
Third Commit                        Arunhegde1231  30/05/23, 9:02 pm
```

Fig 2.1 - User Interface

- **Project Structure Tab:** This sidebar lets you view the various files and assets thatmake up your project you can remove or add new files in the various directories as per your requirements to make the app have more functionality.
- **Tool Window:** This Window on the bottom lets you pick from various functionalities to be displayed at the bottom.
- **Navigation Bar:** This lets you see which directory you are currently in. And lets you see the structure of the ap.
- **Tools:** They are the various tools that can be displayed in the tool windows.
- **Editor Windows:** This lets us make changes to the code, it displays all the text andthe code and lets us edit them.
- **Side Tools:** Very similar to the bottom tools this lets us choose the various views for the structure tab.

2.1.3 Gradle build system

Android Studio uses Gradle as the foundation of the build system, with more Android- specific capabilities provided by the Android plugin for Gradle. This build system runs asan integrated tool from the Android Studio menu, and independently from the command line. You can use the features of the build system to do the following:

- Customize, configure, and extend the build process.
- Create multiple APKs for your app, with different features using the same project and modules.
- Reuse code and resources across source sets.

By employing the flexibility of Gradle, you can achieve all of this without modifying your app's core source files. Android Studio build files are named build.gradle. They areplain text files that use Groovy syntax to configure the build with elements provided by the Android plugin for Gradle. Each project has one top-level build file for the entire project and separate module-level build files for each module. When you import an existing project, Android Studio automatically generates the necessary build files.

2.2 Programming Languages

2.2.1 JAVA

Java is a programming language and computing platform first released by Sun Microsystems in 1995. It has evolved from humble beginnings to power a large share of today's digital world, by providing the reliable platform upon which many services and applications are built. Java is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. It is a general- purpose programming language intended to let application developers write once, run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java withoutthe need for recompilation. Java applications are typically compiled to byte- code that can runon any Java virtual machine (JVM) regardless of the underlying computer architecture. The syntax of Java is similar to C and C++, but has fewer low-level facilities thaneither of them. The Java runtime provides dynamic capabilities (such as reflection and runtimecode modification) that are typically not available in traditional compiled languages. It is owned by Oracle, and more than **3 billion** devices run Java.

It is used for:

- Mobile applications (specially Android apps)
- Desktop applications
- Web applications
- Web servers and application servers
- Games
- Database connection

2.2.2 XML

Extensible Markup Language (XML) lets you define and store data in a shareable manner. XML supports information exchange between computer systems such as websites, databases, and third-party applications. Predefined rules make it easy to transmit data as XML files over any network because the recipient can use those rules to read the data accurately and efficiently. It is a markup language that provides rules to define any data. Unlike other programming languages, XML cannot perform computing operations by itself.Instead, any programming language or software can be implemented for structured datamanagement.

Chapter 3

SYSTEM REQUIREMENT

To be used efficiently, all computer software needs certain hardware components orother software resources to be present on a computer. These prerequisites are known as (computer) system requirements and are often used as a guideline as opposed to an absoluterule. System requirements are a statement that identifies the functionality that is needed by a system in order to satisfy the customer's requirements.

3.1 User Requirements

When the app opens the user should be able to see their notes on the main page. They should be able to add/remove/archive their notes. An option is provided to add text/video/audioto their notes. The user should be able to categorize their notes and tag them and view the notes by category. The user can also make To-Do Lists and set timers and reminders for them. The user should also be able to share their notes with others. The user should be able to sort their notes according to date of creation, title of the note, etc.

3.2 Software Requirements

	Microsoft Windows	**Mac**	**Linux**
Operating System Version	Microsoft® Windows® 7/8/10 (32- or 64-bit) *The Android Emulator only supports 64-bit Windows.*	Mac® OS X® 10.10 (Yosemite) or higher, up to 10.14 (macOS Mojave)	GNOME or KDE desktop *Tested on gLinux based on Debian (4.19.67-2rodete2).*
Minimum required JDKversion	Java Development Kit 8		
Android Studio	Latest Version Available		

3.3Hardware Requirements

Random Access Memory (RAM)	4 GB RAM minimum; 8 GB RAM recommended.
Processor	Intel® Core™ i5-8265U CPU @ 1.60GHz 1.80 GHz
Free digital storage	2 GB of available digital storage minimum, 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image).
Minimum screen resolution	1280 x 800
System Type	64-bit Operating System, x64-bases processor

3.3 Functional Requirements

- When a user opens the app, the app requests the user to set their company name with their password.
- The user is able to view the content of generated invoice and PDF.
- The user can categorize bills which are paid and those which are not paid which are displayed separately in different pages.
- The user is able to download PDF of the invoice.

3.4Non-Functional Requirements

- The application is be user friendly.
- The application doesn't crash.
- The system loads meta data of the application on start up

Chapter 4

SYSTEM DESIGN

4.1 Work Flow Model

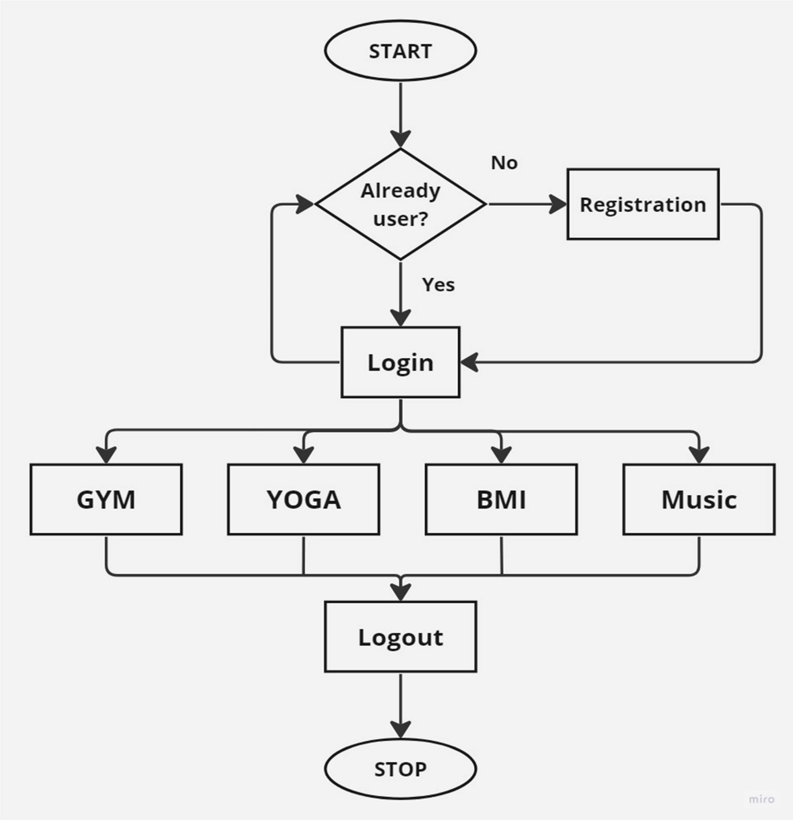


Fig 4.1 - Work Flow Model

The following figure shows the features of the Fitness application. They include:

1. Home Page:

- Gym Category: Clicking on this category will take the user to a list of exercises related to gym workouts.
- Yoga Category: Clicking on this category will take the user to a list of yoga exercises.
- Each exercise in the list will have a title and a brief description.
- Clicking on an exercise will display a timer for that exercise and provide a YouTube link for further information on the exercise steps.

2. Gym Exercises Page:

- This page will list various gym exercises.
- Each exercise will have a title, description, and thumbnail image.
- Clicking on an exercise will display a timer and a YouTube link for detailed instructions.

3. Yoga Exercises Page:

- This page will list various yoga exercises.
- Each exercise will have a title, description, and thumbnail image.
- Clicking on an exercise will display a timer and a YouTube link for detailed instructions.

4. BMI Calculator Page:

- This page will allow users to calculate their Body Mass Index (BMI).
- It will include input fields for height and weight, and a button to calculate the BMI.
- The calculated BMI will be displayed along with a corresponding BMI category (e.g., underweight, normal weight, overweight, etc.).

5. Songs Page:

- This page will provide a selection of workout songs.
- Users can browse through the list and play songs while exercising.

6. Tips and Quotes Page:

- This page will display fitness tips and motivational quotes to inspire and encourage users.

7. Login and Sign Up Page:

- Users will be able to create an account or log in to their existing account.
- The login page will include input fields for email/username and password.
- The sign-up page will include fields for name, email/username, password, and optional additional information.

Chapter 5

IMPLEMENTATION

5.1 XML Code

fragment_home.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout android:gravity="center"
    android:id="@+id/home"
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_gravity="center"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/homebg"
    android:orientation="vertical"
    xmlns:app="http://schemas.android.com/apk/res-auto">
    <androidx.cardview.widget.CardView
        android:id="@+id/cardView"
        android:layout_width="350dp"
        android:layout_height="100dp"
        android:layout_marginTop="40dp"
        app:cardCornerRadius="20dp"
        app:cardElevation="20dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/cardView2">
        <androidx.constraintlayout.widget.ConstraintLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:background="@color/black">
            <ImageView
        android:id="@+id/imageView2"
        android:layout_width="80dp"
        android:layout_height="match_parent"
        android:layout_marginStart="252dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.0"
        app:srcCompat="@drawable/yoga_icon" />
      <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:fontFamily="@font/aclonica"
        android:text="YOGA"
        android:textColor="@color/white"
        android:textSize="28sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.15"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
  </androidx.cardview.widget.CardView>
  <androidx.cardview.widget.CardView
    android:id="@+id/cardView2"
    android:layout_width="350dp"
    android:layout_height="100dp"
    android:layout_marginTop="300dp"
```

```
app:cardCornerRadius="20dp"
app:cardElevation="20dp"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent">
<androidx.constraintlayout.widget.ConstraintLayout
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:background="@color/black">
  <ImageView
    android:id="@+id/imageView"
    android:layout_width="80dp"
    android:layout_height="match_parent"
    android:layout_marginStart="252dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.0"
    app:srcCompat="@drawable/weightlifter_icon" />
  <TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:fontFamily="@font/aclonica"
    android:text="GYM WORKOUT"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toStartOf="@+id/imageView"
```

```
            app:layout_constraintHorizontal_bias="0.0"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent" />
    </androidx.constraintlayout.widget.ConstraintLayout>
</androidx.cardview.widget.CardView>
<!--androidx.cardview.widget.CardView
    android:id="@+id/cardView3"
    android:layout_width="350dp"
    android:layout_height="100dp"
    android:layout_marginTop="16dp"
    android:clickable="true"
    app:cardCornerRadius="20dp"
    app:cardElevation="20dp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/cardView">
    <androidx.constraintlayout.widget.ConstraintLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="@color/black">
        <ImageView
            android:id="@+id/imageView3"
            android:layout_width="80dp"
            android:layout_height="match_parent"
            android:layout_marginStart="252dp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.0"
            app:srcCompat="@drawable/calisthenics_icon" />
        <TextView
```

```
            android:id="@+id/textView3"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:fontFamily="@font/aclonica"
            android:text="Calisthenics"
            android:textAllCaps="true"
            android:textColor="@color/white"
            android:textSize="24sp"
            android:textStyle="bold"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toStartOf="@+id/imageView3"
            app:layout_constraintHorizontal_bias="0.153"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent" />
      </androidx.constraintlayout.widget.ConstraintLayout>
   </androidx.cardview.widget.CardView-->
   <ImageView
      android:id="@+id/imageView4"
      android:layout_width="344dp"
      android:layout_height="196dp"
      android:layout_marginTop="4dp"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.462"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:srcCompat="@drawable/bullhead" />
   <TextView
      android:id="@+id/tvmarq"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
```

```
app:cardElevation="20dp"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/cardView">
<androidx.constraintlayout.widget.ConstraintLayout
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:background="@color/black">
  <ImageView
    android:id="@+id/imageView3"
    android:layout_width="80dp"
    android:layout_height="match_parent"
    android:layout_marginStart="252dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.0"
    app:srcCompat="@drawable/calisthenics_icon" />
  <TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:fontFamily="@font/aclonica"
    android:text="Calisthenics"
    android:textAllCaps="true"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toStartOf="@+id/imageView3"
```

```
            app:layout_constraintHorizontal_bias="0.153"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent" />
    </androidx.constraintlayout.widget.ConstraintLayout>
</androidx.cardview.widget.CardView-->
<ImageView
    android:id="@+id/imageView4"
    android:layout_width="344dp"
    android:layout_height="196dp"
    android:layout_marginTop="4dp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.462"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:srcCompat="@drawable/bullhead" />
<TextView
    android:id="@+id/tvmarq"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="24dp"
    android:ellipsize="marquee"
    android:marqueeRepeatLimit="marquee_forever"
    android:scrollHorizontally="true"
    android:singleLine="true"
    android:text="@string/tvmar"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/cardView" />
<TextView
    android:id="@+id/textView3"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:fontFamily="@font/aclonica"
        android:text="BE FIT"
        android:textColor="#FDF65307"
        android:textSize="34sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/imageView4"
        app:layout_constraintVertical_bias="0.22" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

activity_timer.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#AED581"
    android:backgroundTintMode="multiply"
    android:orientation="vertical">
    <ImageView
        android:id="@+id/imageView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
    <Button
        android:id="@+id/startButton"
        android:layout_width="161dp"
        android:layout_height="72dp"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="192dp"
        android:layout_marginBottom="420dp"
        android:background="#F8FBF8"
        android:backgroundTint="#FFFFFF"
        android:text="Start"
        android:textColor="#000000"
        android:textSize="24sp"
        android:textStyle="bold" />
    <Button
        android:id="@+id/stopButton"
        android:layout_width="161dp"
        android:layout_height="72dp"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="14dp"
        android:layout_marginBottom="423dp"
        android:background="#FFFF00"
        android:text="Stop"
        android:textColor="#000000"
        android:textSize="24sp"
        android:textStyle="bold" />
    <TextView
        android:id="@+id/timerTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="91dp"
```

```
        android:layout_marginBottom="523dp"
        android:text="00:00:00"
        android:textColor="#000000"
        android:textSize="60sp" />
    <TextView
        android:id="@+id/textView10"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="135dp"
        android:layout_marginBottom="638dp"
        android:text="TIMER"
        android:textColor="#1C1C1B"
        android:textSize="48sp"
        android:textStyle="bold" />
</RelativeLayout>
```

5.1.2 JAVA Code

home_fragment.java

```
package com.zephgv.mad.beastfitness;
import android.content.Intent;
import android.graphics.drawable.Drawable;
import android.media.MediaPlayer;
import android.net.Uri;
import android.os.Bundle;
import androidx.cardview.widget.CardView;
import androidx.fragment.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;
import android.widget.VideoView;
public class HomeFragment extends Fragment {
    private static final String ARG_PARAM1 = "param1";
    private static final String ARG_PARAM2 = "param2";
    CardView cd1,cd2,cd3;
    TextView tvm;
    private String mParam1;
    private String mParam2;
    public HomeFragment() {
    }
    public static HomeFragment newInstance(String param1, String param2) {
        HomeFragment fragment = new HomeFragment();
        Bundle args = new Bundle();
        args.putString(ARG_PARAM1, param1);
        args.putString(ARG_PARAM2, param2);
        fragment.setArguments(args);
        return fragment;
    }
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);
            mParam2 = getArguments().getString(ARG_PARAM2);
        }
    }
@Override
```

```
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    View view = inflater.inflate(R.layout.fragment_home, container, false);
    cd1 = view.findViewById(R.id.cardView);
    cd2 = view.findViewById(R.id.cardView2);
    //cd3 = view.findViewById(R.id.cardView3);
    tvm = view.findViewById(R.id.tvmarq);
    tvm.setSelected(true);
    cd1.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent = new Intent(getActivity(),YogaActivity.class);
            startActivity(intent);
        }
    });
    cd2.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent = new Intent(getActivity(),WorkoutActivity.class);
            startActivity(intent);
        }
    });
    /*cd3.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent = new Intent(getActivity(),CalActivity.class);
            startActivity(intent);
        }
    });*/
    return view;
    }
}
```

timer.java

```
package com.zephgv.mad.beastfitness;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.content.Intent;
import android.os.Bundle;
import android.os.CountDownTimer;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import android.content.Intent;
import android.os.Bundle;
import android.os.CountDownTimer;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
public class timer extends AppCompatActivity {
        private ImageView imageView;
        private Button startButton;
        private Button stopButton;
        private TextView timerTextView;
        private CountDownTimer countDownTimer;
        private String content;
        @Override
        protected void onCreate(Bundle savedInstanceState) {
            super.onCreate(savedInstanceState);
            setContentView(R.layout.activity_timer);
            imageView = (ImageView) findViewById(R.id.imageView);
            startButton = (Button) findViewById(R.id.startButton);
            stopButton = (Button) findViewById(R.id.stopButton);
            timerTextView = (TextView) findViewById(R.id.timerTextView);
            content = getIntent().getStringExtra("content");
            startButton.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    countDownTimer = new CountDownTimer(60000, 1000) {
                        @Override
                        public void onTick(long millisUntilFinished) {
                            timerTextView.setText(String.format("%02d:%02d:%02d", millisUntilFinished
/ 3600000, (millisUntilFinished / 60000) % 60, (millisUntilFinished / 1000) % 60));
                        }
                        @Override
                        public void onFinish() {
                            timerTextView.setText("00:00:00");
                        }
                    };
                    countDownTimer.start();
                }
            });
           stopButton.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    countDownTimer.cancel();
                }
            });
        }
}
```

5.1.3 Modules

1. Home Page with Gym and Yoga Categories

- Category Selection Module: This module handles the display of the home page with the gym and yoga categories. It allows users to click on a category to navigate to the corresponding exercise list page.
- Gym Exercises Module: This module displays a list of gym exercises within the gym category. It retrieves exercise data from a database or API and presents it in a user-friendly manner.
- Yoga Exercises Module: Similar to the gym exercises module, this module displays a list of yoga exercises within the yoga category.

2. Exercise Details with Timer and YouTube Link

- Exercise Details Module: This module retrieves detailed information about a specific exercise, including its title, description, and other relevant details. It is responsible for fetching exercise data from the database or API.
- Timer Module: This module provides a countdown timer for each exercise, allowing users to track their workout duration. It handles the logic for starting, pausing, and resetting the timer.
- YouTube Integration Module: This module integrates the YouTube API to retrieve and display a YouTube video link for each exercise. It retrieves the video ID or URL from the database or API and embeds the video player within the application.

3. BMI Calculator, Songs, Tips, and Quotes

- BMI Calculator Module: This module includes the necessary logic to calculate the user's Body Mass Index (BMI) based on their input of height and weight. It validates the input, performs the calculation, and displays the calculated BMI along with the corresponding BMI category.
- Songs Module: This module manages the collection of workout songs. It retrieves and displays a list of songs or playlists that users can choose from. It may include features like song search, play/pause controls, and playlist creation.

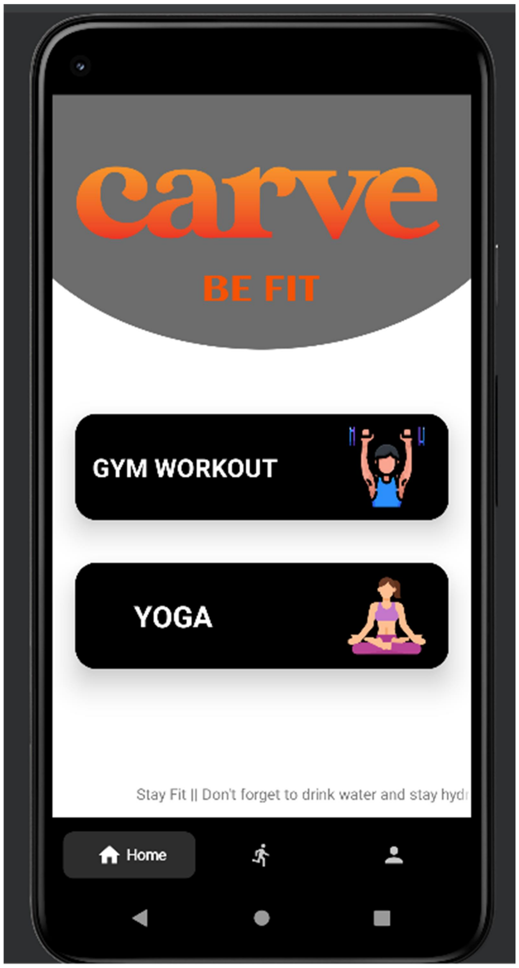
- Tips and Quotes Module: This module retrieves fitness tips and motivational quotes from a database or API and displays them to users. It may have features like random tip or quote generation and the ability to share them on social media.

4. Login and Sign Up Page

- Authentication Module: This module handles user authentication, including login and sign-up functionality. It verifies user credentials, securely stores user data, and provides session management.
- User Management Module: This module manages user profiles, allowing users to update their information, reset passwords, and view their workout history. It interfaces with the authentication module to ensure secure user data handling.

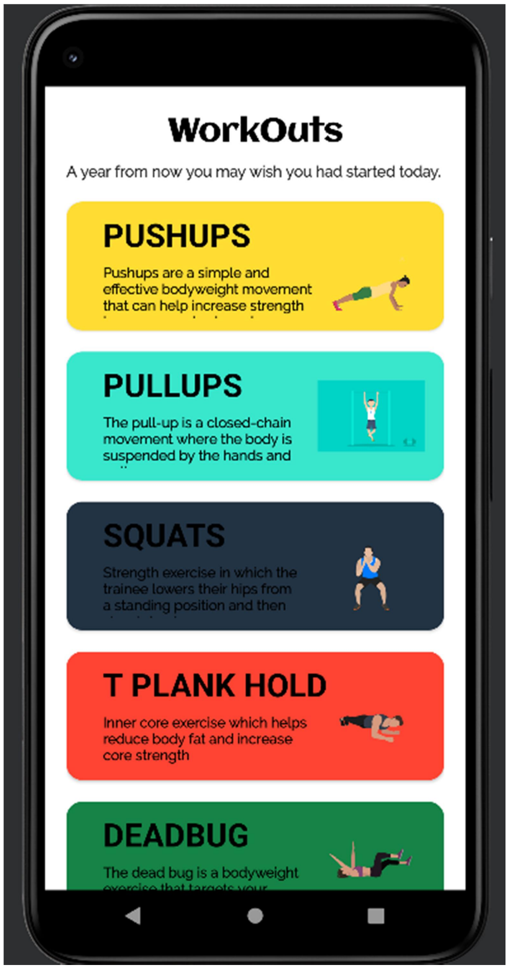
Chapter 6

RESULTS



Stay Fit || Don't forget to drink water and stay hyd

Fig 6.1 - Home Page



WorkOuts

A year from now you may wish you had started today.

PUSHUPS

Pushups are a simple and effective bodyweight movement that can help increase strength

PULLUPS

The pull-up is a closed-chain movement where the body is suspended by the hands and

SQUATS

Strength exercise in which the trainee lowers their hips from a standing position and then

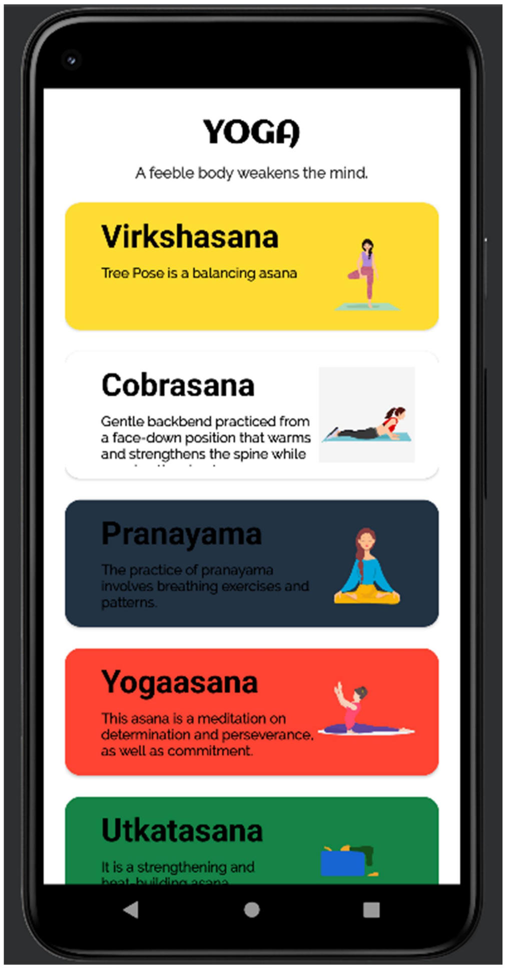
T PLANK HOLD

Inner core exercise which helps reduce body fat and increase core strength

DEADBUG

The dead bug is a bodyweight

Fig 6.2 - Workout Page



YOGA

A feeble body weakens the mind.

Virkshasana

Tree Pose is a balancing asana

Cobrasana

Gentle backbend practiced from a face-down position that warms and strengthens the spine while

Pranayama

The practice of pranayama involves breathing exercises and patterns.

Yogaasana

This asana is a meditation on determination and perseverance, as well as commitment.

Utkatasana

It is a strengthening and heat-building asana

Fig 6.3 - Yoga Page

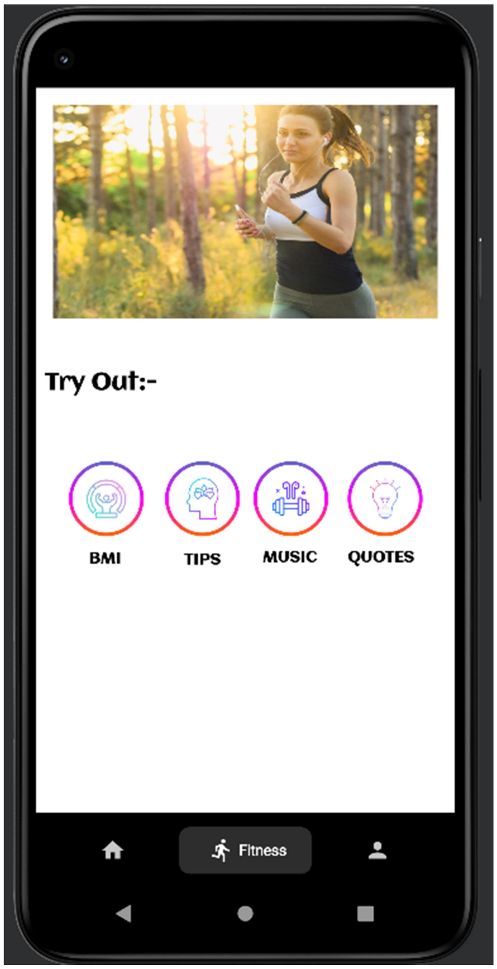
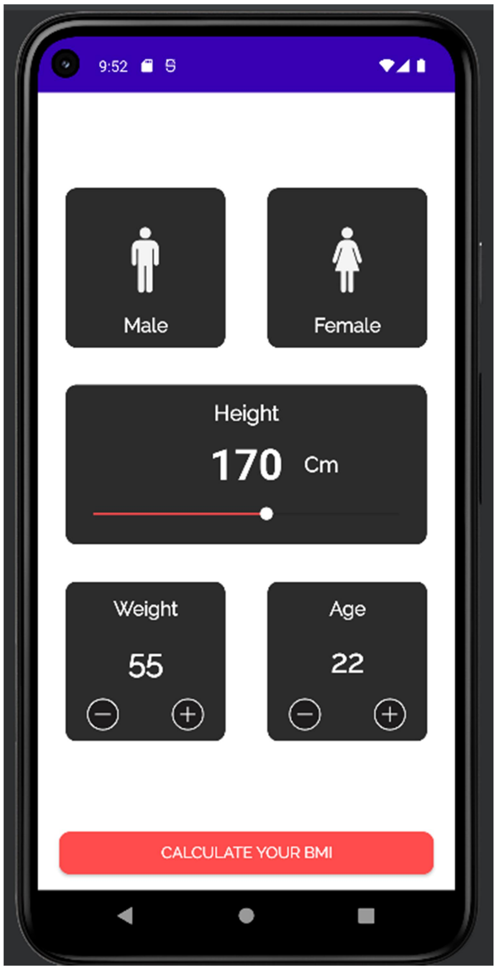


Fig 6.4 -. Fitness Page

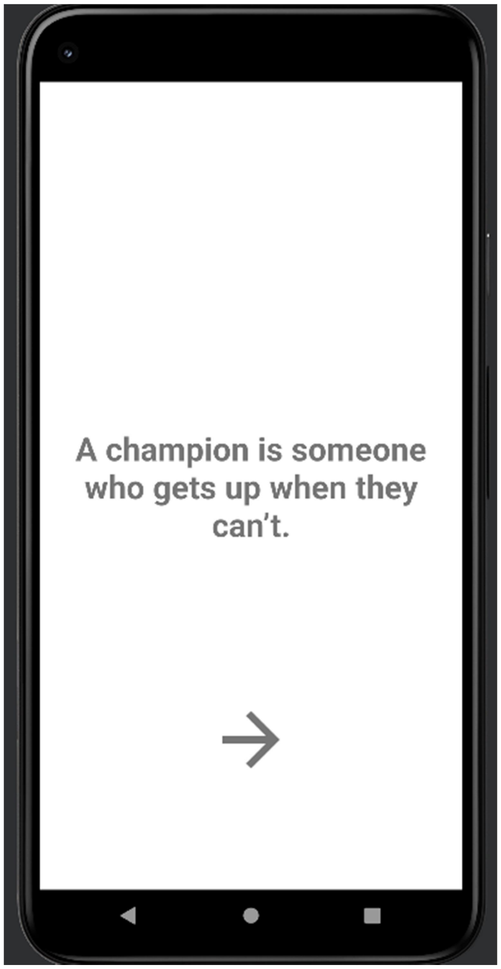


Height
170 Cm

Weight
55

Age
22

Fig 6.5 -. BMI Page



A champion is someone who gets up when they can't.

Fig 6.6 - Quote Page

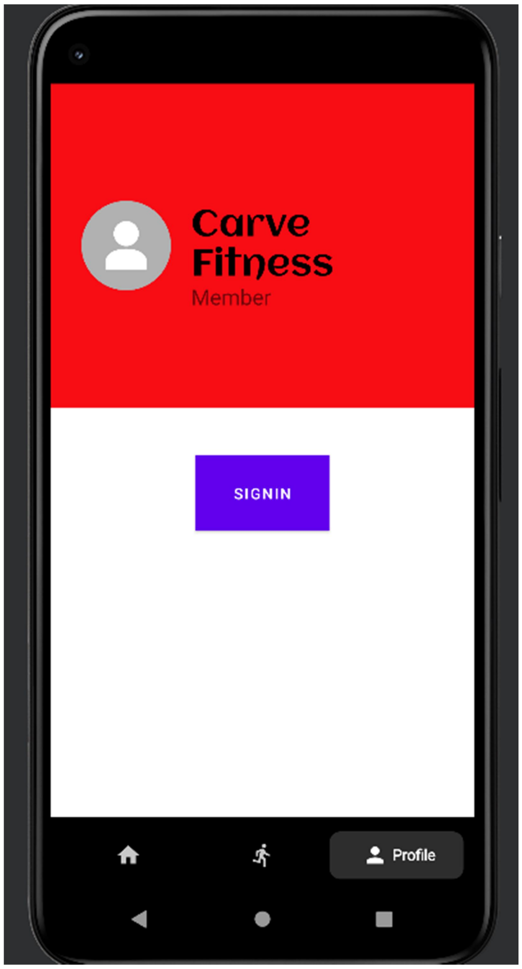


Fig 6.7 - Profile Page

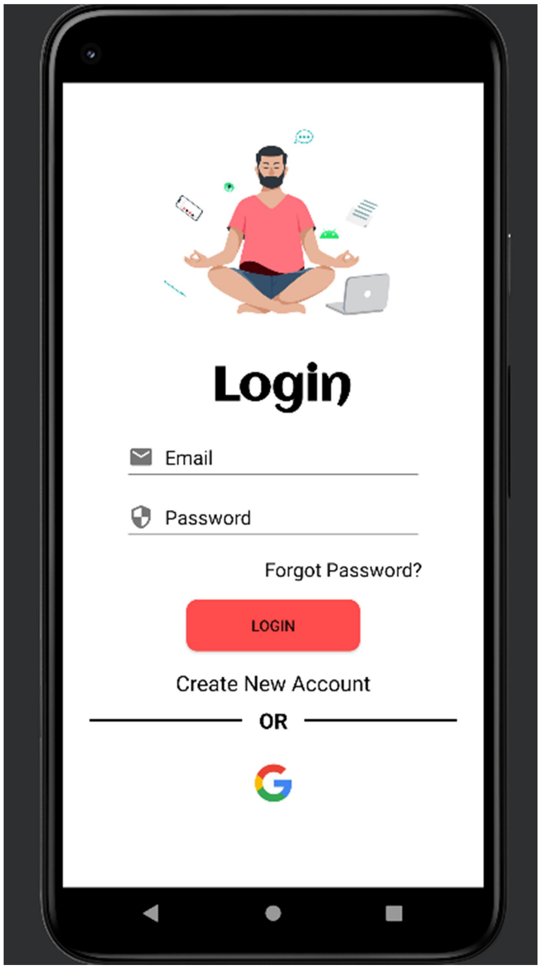


Fig 6.8 - Login Page

Chapter 7

CONCLUSION AND FUTURE ENHANCEMENT

The fitness app that we have developed is a comprehensive and user-friendly solution for individuals seeking to improve their fitness levels and overall well-being. The app offers a wide range of features and functionalities designed to cater to users of all fitness levels and preferences.In terms of user experience, the app boasts an intuitive and visually appealing interface, making it easy for users to navigate through different sections and access the desired features. The app is available on multiple platforms, ensuring accessibility to a wide range of users.Overall, the fitness app combines functionality, personalization, and community engagement to deliver a comprehensive and enjoyable fitness experience. With its user-friendly interface and array of features, the app has the potential to empower individuals on their fitness journeys, helping them achieve their goals and lead healthier, happier lives.

Future Enhancement Points for the Fitness Application Mini Project:

- Personalized Workout Plans: Implement a feature that allows users to create personalized workout plans based on their fitness goals, preferences, and available time. The application can provide recommendations and suggest exercises tailored to individual needs, taking into account factors such as fitness level, equipment availability, and target muscle groups.
- Progress Tracking and Analytics: Incorporate a comprehensive progress tracking system that allows users to monitor their exercise performance and track their progress over time. This can include features such as workout history, calorie tracking, step tracking, and the ability to set goals and receive performance analytics.
- Social Integration: Enable social sharing and integration with popular social media platforms, allowing users to share their workout achievements, progress, and challenges with their friends and followers. This can create a sense of community and foster motivation through healthy competition and support from peers.
- Nutrition and Meal Planning: Expand the application to include a nutrition and meal planning component. Users can access meal plans, recipes, and nutritional information to complement their exercise routines and promote a holistic approach to health and wellness.