VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belagavi-590018

A Mobile Application Development Mini-Project Report

On

FITNESS APPLICATION

SUBMITTED IN PARTIAL FULFILLMENT FOR THE AWARD OF DEGREE

BACHELOR OF ENGINEERING

IN

INFORMATION SCIENCE AND ENGINEERING

SUBMITTED BY

GAURI R(1JB19IS028)
KARTHIK S(1JB19IS041)

Under the Guidance of
Mr. Jeevaraj R
Assistant Professor
Dept. of ISE, SJBIT

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

SJB INSTITUTE OF TECHNOLOGY

BGS HEALTH AND EDUCATION CITY,
KENGERI, BENGALURU-560060, KARNATAKA, INDIA.

2021 – 2022

|| Jai Sri Gurudev ||
Sri Adichunchanagiri Shikshana Trust ®

SJB INSTITUTE OF TECHNOLOGY

BGS Health & Education City, Kengeri, Bengaluru – 560 060

Department of Information Science & Engineering

CERTIFICATE

Certified that the Mini-project work entitled "**FITNESS APPLICATION",** is a bonafide work carried out by **GAURI R(1JB19IS028)** and **KARTHIK S(1JB19IS041),** students of **SJB Institute of Technology,** in partial fulfilment for 6th semester **Mobile Application Development Laboratory with mini project** in **INFORMATION SCIENCE AND ENGINEERING** of the **VISVESVARAYA TECHNOLOGICAL UNIVERSITY**, **BELAGAVI** during the academic year **2021-22.** It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The mini project report has been approved as it satisfies the academic requirements in respect of mini project prescribed for the said degree.

Mr. JEEVARAJ R **Asst. Professor** **Dept. of ISE, SJBIT**	**Dr. REKHA B** **Professor & Head** **Dept. of ISE, SJBIT**

EXTERNAL VIVA

	NAME OF THE EXAMINERS	SIGNATURE WITH DATE
1.		
2.		



ACKNOWLEDGEMENT

We would like to express our profound gratitude to His Divine Soul **Padmabhushan Sri Sri Sri Dr. Balagangadharanatha MahaSwamiji** and His Holiness **Jagadguru Sri Sri Sri Dr. Nirmalanandanatha MahaSwamiji** for providing us an opportunity to complete our academics in this esteemed institution.

We would also like to express our profound thanks to **Revered Sri Sri Dr. Prakashnath Swamiji**, **Managing Director**, SJB Institute of Technology, for his continuous support in providing amenities to carry out this project in this admired institution.

We express our gratitude to **Dr. K V Mahendra Prashanth**, **Principal**, SJB Institute of Technology, for providing us an excellent facilities and academic ambience; which have helped us in satisfactory completion of project work.

We extend our sincere thanks to **Dr. Rekha B**, **Professor & Head**, Department of Information Science and Engineering; for providing us an invaluable support throughout the period of our project work.

We wish to express our heartfelt gratitude to the **mini-project coordinator, Mr. Jeevaraj R, Asst. Prof,** Department of Information Science and Engineering for his valuable guidance, suggestions and cheerful encouragement during the entire period of our project work.

Finally, we take this opportunity to extend our earnest gratitude and respect to our parents, Teaching & Non-teaching staff of the department, and all our friends, who have directly or indirectly supported us during the period of our project work.

GAURI R(1JB19IS028)
KARTHIK S(1JB19IS041)

ABSTRACT

Fitness application is an android application where the user can get access to various yoga asanas, on clicking any asana, a detailed description of that particular asana with its benefits and the steps to perform the asana is displayed. This application has many asanas on home page which the user can access based on his requirements. The amalgamation of yoga and technology has led to a yoga mobile app development. This app presents an opportunity for everyone out there to learn yoga without stepping out of their homes. With a collection of various yoga asanas squeezed into one app, this mobile app is seen as a medium that can change your life.

This app is a useful solution for people who want to practice yoga but cannot or do not want to attend an in-person class. Using this app, it makes it easy to practice yoga anywhere and anytime.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

Android is a Linux-based operating system designed primarily for touchscreen mobile devices such as smartphones and tablet computers.Initially developed by Android, Inc., which Google backed financially and later bought in 2005. Android is open source and Google releases the code under the Apache License. This open source code and permissive licensing allows the software to be freely modified and distributed by device manufacturers, wireless carriers and enthusiast developers. Additionally, Android has a large community of developers writing applications ("apps") that extend the functionality of devices, written primarily in a customized version of the Java programming language.

TABLE 1.1 : Android Versions and Specifications

Version	Code Name	Release Date	API Level	Distribution
1.5	Cupcake	April 30,2009	3	0%
1.6	Donut	September 15,2009	4	0.1%
2.0-2.1	Éclair	October 26,2009	7	1.5%
2.2	Froyo	May 20,2010	8	3.2%
2.3.3-2.3.7	Gingerbread	February 9,2010	10	36.4%
2.3-2.3.2	Gingerbread	December 6,2010	9	0.1%
3.2	Honeycomb	May 10,2011	12	0%
3.2	Honeycomb	July 15,2011	13	0.1%
4.0.x	Ice Cream Sandwich	December 16,2011	15	25.6%
4.1.x	Jelly Bean	July 9,2012	16	29.0%
4.2.x	Jelly Bean	November 13,2012	17	4%

Android software development is the process by which new applications are created for the Android operating system. Applications are usually developed in the Java programming language using the Android Software Development Kit.

ADT (Android Development Tools) is the software used to develop android apps. It basically encases Eclipse IDE,which is a multi-language Integrated development environment (IDE) comprising a base workspace and an extensible plug-in system for customizing the environment.. The latest version comes with ADT plugin preinstalled and bundled to the IDE.

This is how the IDE looks like with the important elements marked

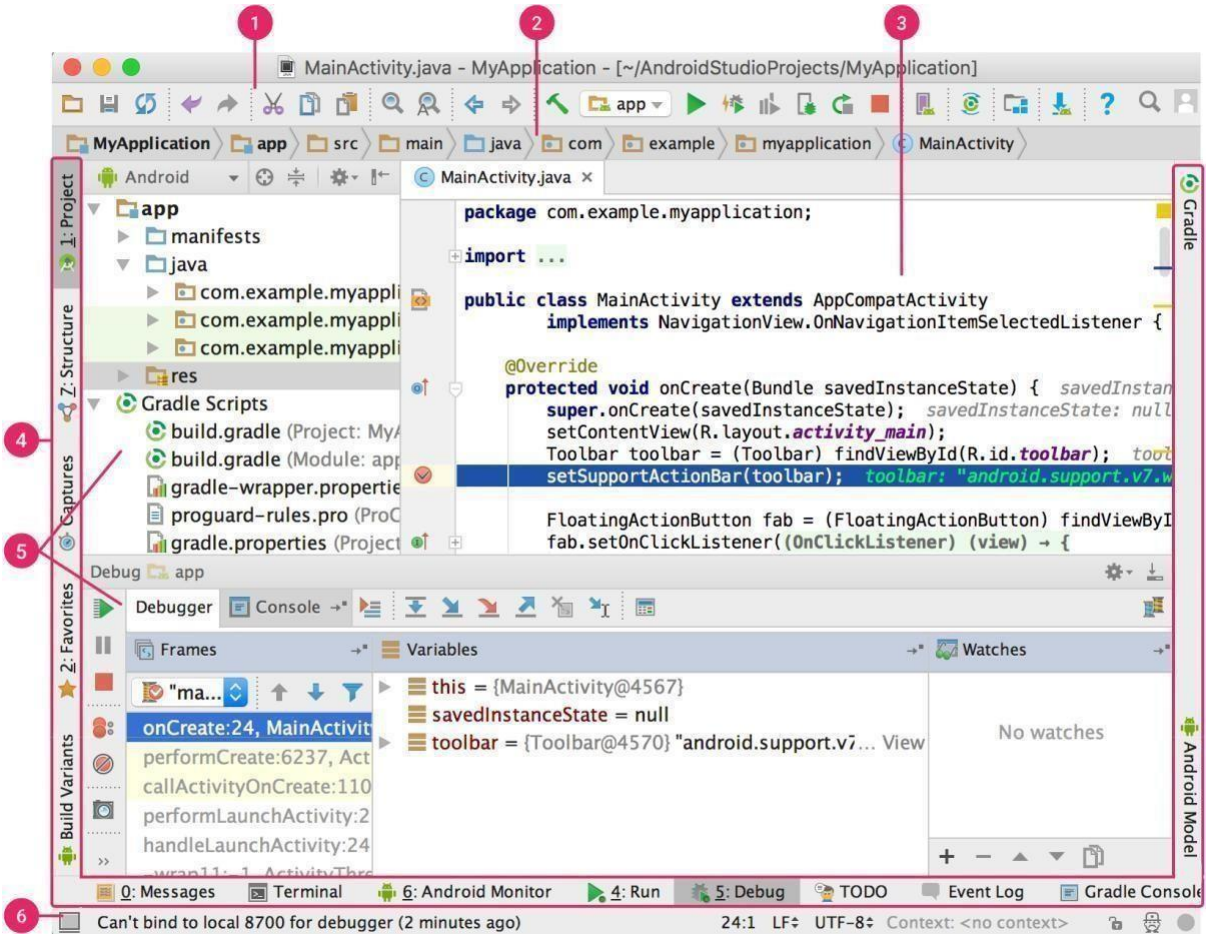

Figure 1.1 Android Studio main window

The **toolbar** lets you carry out a wide range of actions, including running your app and launching Android tools.

The **navigation bar** helps you navigate through your project and open files for editing. It provides a more compact view of the structure visible in the **Project** window.

The **editor window** is where you create and modify code. Depending on the current file type, the editor can change. For example, when viewing a layout file, the editor displays the Layout Editor.

The **tool window bar** runs around the outside of the IDE window and contains thebuttons that allow you to expand or collapse individual tool windows. The **tool windows** give you access to specific tasks like project management,earch, version control, and more. You can expand them and collapse them.

The **status bar** displays the status of your project and the IDE itself, as well as any warnings or messages.

Application programming interface (API) specifies how some software components should interact with each other. In practice in most of the cases an API is a library that usually includes specification for routines, data structures, object classes, and variables. An API specification can take many forms, including an International Standard such as POSIX, vendor documentation such as the Microsoft Windows API, the libraries of a programming language, e.g., Standard Template Library in C++ or Java API.

Google APIs can be downloaded from Google Code, Google's site for developer tools, APIs and technical resources. The Google Data API] allow programmers to create applications that read and write data from Google services. Currently, these include APIs for Google Apps, Google Analytics, Blogger, Google Base, Google Book Search, Google Calendar, Google Code Search, Google Earth, Google Spreadsheets, Google Notebook, and Picasa Web Albums.

SDK (Software Development Kit or "devkit") is typically a set of software development tools that allows for the creation of applications for a certain software package,software framework, hardware platform, computer system, video game console, operating system, or similar development platform. It may be something as simple as an application programming interface (API) in the form of some files to interface to a particular programming language or include sophisticated hardware to communicate with a certain embedded system. Common tools include debugging aids and other utilities often presented in an integrated development environment (IDE).

In the latest version of ADT, the android SDK adds on to the IDE automatically as soon as you unzip and load the IDE.

SDK Manager enables us to download Google APIs and use them in our code

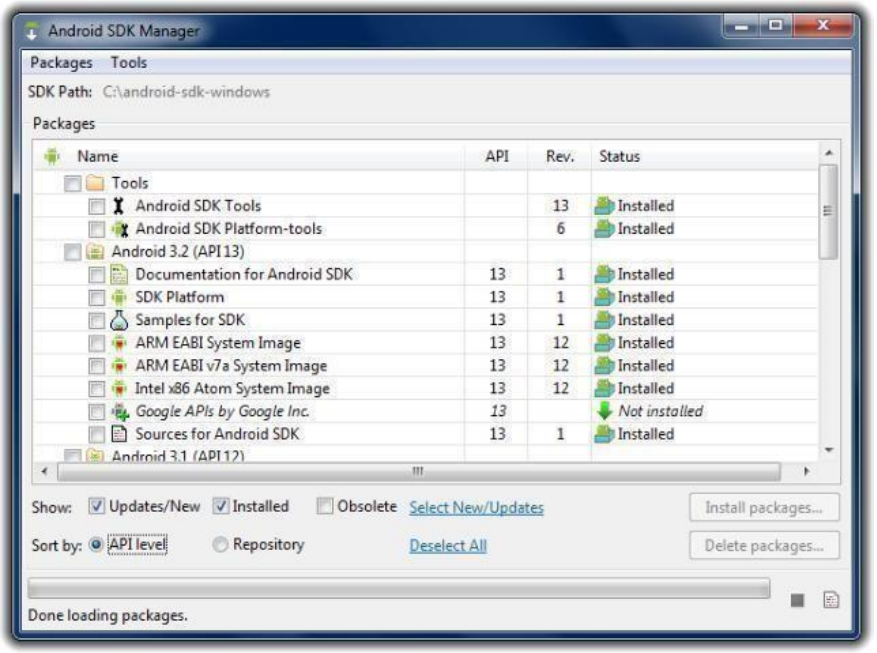

Figure 1.2 SDK Manager

Android Virtual Device (AVD) manager enables us to launch virtual android devices/ emulators in our PC and run the app in the emulator, and at the same time we can track and debug each app activity from the Logcat in our IDE.

ADVANTAGES:

- Improves Efficiency.
- Offers High Scalability.
- Secures the App Data.
- Easy to Maintain.
- Improves Customer Relationship
- Facilitates New Client Data Retrieval.
- Provides Real-time Project Access.

FEATURES:

- A flexible Gradle-based build system.
- A unified environment where one can develop for all Android devices.
- Apply Changes to push code and resource changes to the running app without restarting the app.
- 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
- Cloud Messaging and App Engine.

Chapter 2

SYSTEM REQUIREMENTS SPECIFICATION

2.1 HARDWRE REQUIREMENTS

- RAM: 2GB recommended, 512 MB minimum.
- HARD DISK: 110 MB of hard disk space required, 40 MB additional hard disk space required for installation (150 MB total).
- MONITOR: 15 VGA Color Monitor
- SYSTEM: Intel Core 2 Duo or Above

2.2 SOFTWARE REQUIREMENTS

Data Requirements

The set of data that is involved in any project is defined using data requirements. For this project, the main data required is the product information to register the application and the item's information. Without this information the application cannot process the transaction.

Functional Requirements

Functional requirements are properties that must exist in the final system. For any mobile application, we need to download the application from the play store. The application could be either free or paid depending upon the store or merchant. To use the application, the user needs to register and login to the application after installing by providing login information. Once, he or she logins into the application, they can use all the features.

Performance Requirements

Response time, scalability, platform dependencies, tolerance are the performance requirements that should be considered when developing any system. It should be able to deliver the information about any of those issues to the user when the system is no longer able to provide results when the user wants scalable enough to accept new features when we want to expand the application complexity.

System Requirements

The application should be installed into a device, system or any machine in such a way that it should have basic requirements like supporting software and hardware of the device, accessing in-built software, say camera for mobile device, internet permissions, and potential security issues such as virus or any malware detection.[7]

- DEVELOPMENT PLATFORM : WINDOWS 7
- LANGUAGE TOOL : JAVA
- *SOFTWARE USED: Android Studio SDK

Chapter 3

SYSTEM DESIGN

The purpose of the design phase is to develop a clear understanding of what the developer wants people to gain from his/her project. As the developer works on the project, the test for every design decision should be "Does this feature fulfil the ultimate purpose of the project?" A purpose statement affects the design process by explaining what the developer wants the project to do, rather than describing the project itself. The Design Document will verify that the current design meets all of the explicit requirements contained in the system model as well as the implicit requirements desired by the customer.

3.1 SEQUENCE DIAGRAM

A sequence diagram is a type of interaction diagram because it describes how and in what order a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are also known as event diagrams or event scenarios.

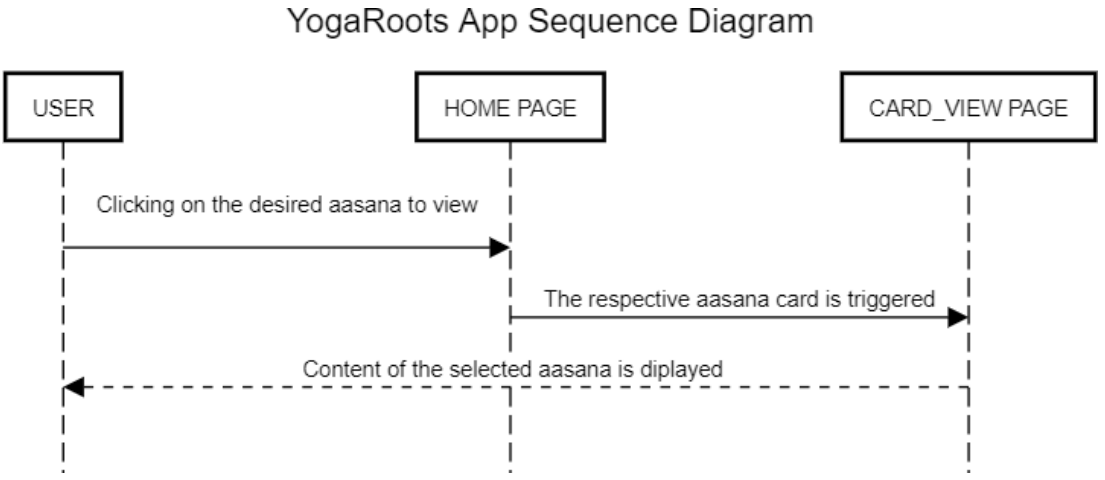

Figure 3.1 Sequence Diagram

3.2 DATA FLOW DIAGRAM

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. They can be used to analyze an existing system or model a new one. Like all the best diagrams and charts, a DFD can often visually "say" things that would be hard to explain in words. Data flow diagrams were popularized in the late 1970s, arising from the book Structured Design, by computing pioneers Ed Yourdon and Larry Constantine. They based it on the "data flow graph" computation models by David Martin and Gerald Estrin. The structured design concept took off in the software engineering field, and the DFD method took off with it. It became more popular in business circles, as it was applied to business analysis, than in academic circles.

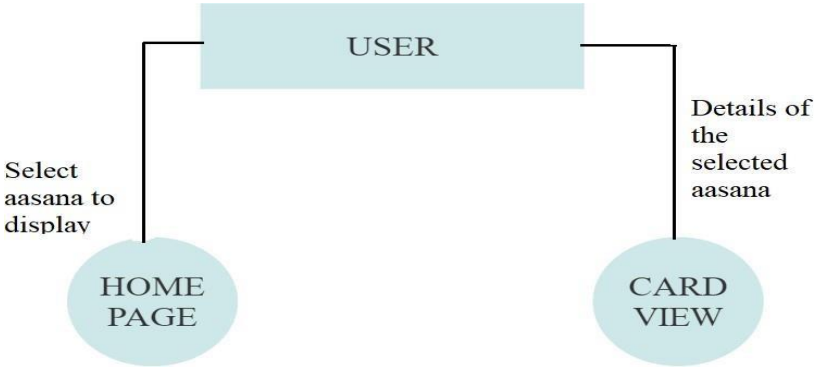

Figure 3.2 Dataflow Diagram

Chapter 4

IMPLEMENTATION

4.1 Introduction to Programming Languages, IDE'S, Tools:

4.1.1 Java

As the project is developing an Android Application, the default programming language is Java. All Android applications are built using Java in Android Studio or Eclipse or both. Java is a popular and widely used language throughout the world. As mentioned in, Java is one of the powerful programming languages like C, C++. developed by Sun Microsystems which has many powerful features as described below. The language is also easy to learn, understand and implement. Java is used in various kinds of applications like Web, Desktop, Mobile, and Big Data. Many powerful features are supported by Java including various libraries, application services, graphics library for 2D/3D applications. The language is flexible enough to maintain code complexity, test, implementation, integration and support. Apart from these, there are other key features which make Java more special. It is object oriented programming language, one of the important hierarchies in the programming languages which is used to implement real time applications, it provides for code reusability, it has a platform independence feature including any virtual machines(Write Once Read Everywhere), as in no need to write the 20 code for different OS as the Java Compliers convert the java source files to bytecode and this could be interpreted by any machine and the actual code is compiled irrespective of any machine, OS. It is more secured as the compilers are designed efficiently to figure out any kind of errors.

4.1.2 IDE's, Tools and Technologies:

Android Studio Android Studio is exclusively designed for developing Android applications. Say, suppose we declared few variables or methods that starts with an 'S', it automatically senses everything that starts with an 'S' and makes suggestions. It also supports Git as a version control system to maintain the app changes and push them into github. All java files, layout files (for design) are integrated into a single project easily. After the completion of project, the whole application could be put as an .APK (Android Package) file, in which we can run that APK file in any device and use the application. Other main tools include Android SDK, ADB,and Gradle Build.

Android Software Development Kit (SDK):

One of the main tools used in developing android applications, as it packages many core features into one SDK and it can be used in the application easily. This helps us to avoid writing lot of code, and building applications faster.

Android Debug Bridge (ADB):

Android SDK uses ADB tool as a connection device which allows us to connect the Android Devices or Emulator with the machine via USB. After developing or while developing applications, we can connect with the device to check how the application runs. Later, we can debug and run the applications.

Gradle Build:

Gradle Scripts are the recent feature that is added to Android Studio. It is basically an automated build system which is used to automate the various phases involved in designing an application that includes design, development, test, debug, and publish. We need to configure the project and modules by mentioning all the supported jar files, SDK's, version name, level, compiled SDK version, build tools version. to ensure that the developed app is compatible with the testing device/emulator. Gradle is also similar to Ant and Maven which helps in maintaining java projects (repositories).

Android Device Monitor:

If we want to access all the hidden files that are generated when we run the application, we can use the monitor. We can select any project and explore the files that are related to that project. But, as they are hidden files, we need root permissions to access them. Suppose, if we run the app in device, we need to root the device and run commands in adb shell to get permissions.

SDK Manager:

It is one of the main tools to maintain the updates of all the installed components required to run the project. It also notifies us when the project is not compatible with device or any other compatibility issues and to download any component that is required.

AVD Manager:

It is used to create virtual devices of any desired API level to support higher level SDK's incase our device does not support. Using emulators to test the application is difficult as it might be little slower when compared to real device.

Create an Entity

Room allows you to create tables via an Entity. Let's do this now.

Create a new Kotlin class file called Word containing the Word data class. This class will describe the Entity (which represents the SQLite table) for your words. Each property in the class represents a column in the table. Room will ultimately use these properties to both create the table and instantiate objects from rows in the database.

4.2 XML AND .java CODES OF THE PROJECT

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    tools:context=".MainActivity"
    android:orientation="vertical"
     >

    <android.support.v7.widget.RecyclerView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/recyclerView_id"
        android:clipChildren="false"
        android:clipToPadding="true"
```

```
    android:scrollbars="vertical"
    >
    </android.support.v7.widget.RecyclerView>
    </LinearLayout>
```

activity_yoga.xml

```
  <?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
  tools:context=".YogaActivity">
  <android.support.v4.widget.NestedScrollView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginTop="10dp"
    android:padding="10dp"
    >
    <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:orientation="vertical"
      >
      <TextView
        android:id="@+id/text_recipe"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:text="Aasanas"
        android:textSize="21sp"
        android:textStyle="bold"/>
```

```
    <TextView
        android:id="@+id/ingredients"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:layout_margin="18dp"
        android:textSize="20sp"
        android:text="Benefits"
        />
    <TextView
        android:id="@+id/method"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:text="Aasana"
        android:textSize="21sp"
        android:textStyle="bold"
        />
    <TextView
        android:id="@+id/recipe"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:layout_margin="18dp"
        android:textSize="20sp"
        android:text="Benefits"
        />
  </LinearLayout>
</android.support.v4.widget.NestedScrollView>
```

cardview_yoga.xml

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.v7.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="190dp"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_margin="5dp"
    android:clickable="true"
    app:cardCornerRadius="4dp"
    android:foreground="?android:attr/selectableItemBackground"
    android:id="@+id/cardview_id"
    >
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        >

        <ImageView
            android:id="@+id/aasana_img_id"
            android:layout_width="match_parent"
            android:layout_height="160dp"
            android:layout_marginTop="10dp"
            android:background="@color/colorPrimaryDark"
            android:scaleType="centerCrop"
            />
        <TextView
            android:id="@+id/aasana_text"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:gravity="center"
            android:text="Aasana"
```

```
        android:textSize="13sp"
        android:textColor="#2d2d2d"
        />
    </LinearLayout>

    </android.support.v7.widget.CardView>
```

MainActivity.java

```
package com.example.yoga;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.GridLayoutManager;
import android.support.v7.widget.RecyclerView;

import java.util.ArrayList;
import java.util.List;

public class MainActivity extends AppCompatActivity {

    RecyclerView myrecyclerView;
    RecyclerViewAdapter myAdapter;

    List<Yoga> yoga1;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        yoga1 = new ArrayList<>();
        yoga1.add(new Yoga("Virabhadrasana","1. This asana is known to strengthen and tone the lower
        back, the arms, and the legs.\n" +
```

"2. It helps to stabilize and balance the body as it increases the stamina.\n" +
"3. It is also a great asana for those with desk or sedentary jobs. It stimulates the metabolismas well as restores the spine.\n" +

"4. This asana helps ease out frozen shoulders.\n" +

"5. It also helps release stress from the shoulders almost immediately.\n" +

"6. This asana relaxes the mind and the body, spreading the notion of peace, courage, grace, and a sense of auspiciousness.","STEPS","1.Stand erect and spread your legs about three to four feet apart. Your right foot should be in the front and the left foot behind.\n" +
"2.Now, turn your right foot outwards by 90 degrees and the left by 15 degrees, making sure the heel of the right foot is perfectly aligned with the center of the left foot.\n" +
"3.Lift your arms sideways until they reach the height of your shoulders. Your arms must be parallel to the ground, and your palms should be facing upwards.\n" +
"4.Exhale and bend your right knee, such that your knee and ankle form a straight line. Make sure that your knee does not go ahead of your ankle.\n" +
"5.Now turn your gaze to your right.\n" +
"6.As you move into the pose, stretch your arms further and join your palms above your head.
Look at your palms. Gently push your pelvis down.\n" +
"7.Hold the pose with the same determination as a warrior, and wear a smile on your face.
Breathe normally and keep going down.\n" +
"8.Inhale and come up.\n" +
"9.Exhale and gently bring your hands down from the sides.\n" +
"10.Repeat this pose on the left side, with your left leg in the front and the right one at the back.\n",R.drawable.virabhadrasana));

yoga1.add(new Yoga("Ardha-Chakrasana","1.Opens up the chest.\n" +
"2.Improves stamina.\n" +
"3.Increases confidence.\n" + "4.Relaxes
respiratory muscles.\n" + "5.Clears
narrowing of the bronchi.\n" +"6.Opens up
airway obstruction.\n" +
"7.Improves lung capacity.","STEPS","1.Stand straight with having some distance betweenyour feet and balance your weight equally on both feet.\n" +"2.Keep arms alongside the body.\n" +

```
"3.Breathe in, extend your arms overhead, palms facing each other. Exhale\n"
"4.You can also put your palm at the waist to support the back.\n" +
"5.While inhaling, gently bend backward from the lumbar region and pushing the pelvis
forward.\n" +
"6.Bend the neck backward to stretch the neck muscles, keep the arms in line with the ears,elbows,
and knees straight.\n" +
"7.This is the final position of Ardha Chakrasana.\n" +
"8.Hold the position for 5 – 8 seconds and breathe normally.\n" +"9.Come
back to the straight position. Relax\n" +
"10.Repeat this cycle for 2 – 3 rounds. Take 5 to 10 seconds rest between each round. Add 10seconds
every 2 weeks.\n",R.drawable.ardhachakrasana));


    yoga1.add(new Yoga("Bhujangasana","1.Relieves backache\n" +
            "2.Keeps spine healthy\n" +
            "3.Beneficial for abdominal organs like liver and kidneys\n" +
            "4.Helps in menstrual and gynecological disorders\n" +
            "5.Eases constipation\n" +
            "6.Good for stimulating the appetite\n" +
            "7.Opens of your chest as well as heart and is helpful for people suffering from
depression\n","STEPS","1.Lie down flat on your stomach with your head on the floor, legs straight and
feet together.\n" +
            "2.Keep your palms on the floor under your shoulders and your elbows close to the body.\n" +
            "3.Relax your body and lower back.\n" +
            "4.With inhalation slowly raise your head and start lifting your chest up, keeping your chest
open, use your back muscles for the lift and put less pressure on the hands.\n" +
            "Keep your pubic bone in contact with the floor, the naval can be raised a little bit.\n" +
            "5.You can keep your elbows a little bent and once you reach a comfortable height, look
diagonally.\n" +
            "6.Stay for 5-8 breaths\n" +
            "7.To return, exhale and slowly bend the arms, lower the naval, chest, shoulders and forehead
to the ground.\n" +
            "8.Relax!\n",R.drawable.bhujangasana));
```

```
yoga1.add(new Yoga("Trikonasana","1.Helps in Stretches hips, back muscles, chest and shoulders.\n" +
        "2.Stretches the spine.\n" +
        "3.Give Strength to the thighs, calves and buttocks.\n" +
        "4.Stimulates the spinal nerves.\n" +
        "5.It improves the flexibility of the spine, correct alignment of shoulders\n" +
        "6.It relieves from backache, gastritis, indigestion, acidity, flatulence\n" +
        "7.Assists treatment of neck sprains, reduces stiffness in the neck, shoulders and knees, strengthens the ankles and tones the ligaments of the arms and legs\n","STEPS","1.Stand and keep a minimum distance of 3 feet between your legs.\n" +
        "2.Extend both your arms sideways and keep them level with the shoulders.\n" +
        "3.While inhaling slowly, raise the left arm and bend the body towards the right, with the right arm pointing downwards, with fingers pointed at your toes.\n" +
        "4.Your eyes should face the ceiling and most importantly, be open for keeping optimum body balance.\n" +
        "5.Make sure to inhale deeply and relax the body on exhalation during the final position.\n" +
        "6.Stay in the position for a minimum of 1 minute.\n" +
        "7.Repeat the asana for the other side as well\n",R.drawable.trikonasana));


    yoga1.add(new Yoga("Badhakonasana","1.Stimulates abdominal organs, ovaries, bladder and kidneys.\n" +
        "2.Improves blood circulation.\n" +
        "3.Stretches the inner thighs.\n" +
        "4.Helps in relieve the mild depression, anxiety and fatigue.\n" +
        "5.Helps relieve the symptoms of menopause.\n","STEPS","1.sit with legs straight out in front of you. Exhale and bend your knees, and pull your heels towards the pelvis.\n" +
        "2.Drop your knees out to the sides and press the soles of feet together.\n" +
        "3.Bring your heels as close to pelvis as you can.\n" +
        "4.Grab the big toe to each foot with first two fingers and thumb. Try to keep the outer edges of feet on the floor.\n" +
        "5.Sit with tailbone in back and pubis equidistant from the floor. Lengthen the front torso through the top of the sternum.\n" +
```

"6.Don’t force knees down. Instead release the heads of thigh bones towards the floor.\n" +

"7.Stay in the pose for 1-5 minutes. Inhale and lift your knees away from the floor and extend the legs back to original position.\n",R.drawable.badhakonasana));

yoga1.add(new Yoga("Dhanurasana","1.It opens up your chest, abdomen, throat, ankles and groins.\n" +

"2.Gives you good posture and toned body.\n" +

"3.Stretches the entire body.\n" +

"4.It strengthens the lower back, abdominal muscles and back.\n" +

"5.Dhanurasana helps people with slip-discs.\n","STEPS","1.Lie down in prone position; facing your chest downwards and your back upwards.\n" +

"2.Now relax your body and exhale deeply.\n" +

"3.While you’re exhaling, bend your knees and bring your both heel as close to your hips as possible.\n" +

"4.Then raise your chin and bend your head and neck backward. Your chest should still be touching the ground.\n" +

"5.Now inhale slowly and pull your legs upwards.\n" +

"6.Keep raising your head, neck, chin, chest, thighs, and knees backward and keep only the navel region touching the ground. Balance your body in the navel region.\n" +

"7.Stay in this pose from 20 to 30 seconds.\n" +

"8.Now, release your body as you exhale, and lay down to take some breath.\n",R.drawable.dhanurasana));

yoga1.add(new Yoga("Sarvangasana","1.Favourable changes in vasomotor ability (causing or relating to the constriction or dilatation of blood vessels) due to the increased interchange of blood in the upper part of the body, especially the thorax, the neck and the head.\n" +

"2.Temporary replacement of the abdominal and pelvic viscera.\n" +

"3.Relief in the case of constipation, indigestion, headache, giddiness, neurasthenia, functional disorders of the eye, the ear, the nose and the throat as well as general and sexual debility.\n" +

"4.Wholesome effects of gravity-pressure on the various organs of the body above the waist including the vital endocrine glands.\n" +

"5.Relief in the case of constipation, indigestion, headache, giddiness, neurasthenia.\n","STEPS","1.Exhaling, raise high the legs together enough to make a right angle with the body. Keep the knees straight and the body above the hip-joint on the ground undisturbed.\n" +

"2.At this stage, still exhaling, raise the arms and hold the waist and push the body up as far as possible. Put all the body weight on the arms and rest on the elbows, with the legs thrown upwards.\n" +

"3.When this position is firmly secured, by careful manipulation, make an attempt to shift the hands slowly towards the waist, with the fingers extended to the back of the hip-bones and the thumbs pressed lightly on both sides of the navel.\n" +

"4.Set the chin in the jugular notch and place the full weight upon the shoulders, the neck and the back of the head (final position). Complete the above steps in 4 seconds, while exhaling.\n" +

"5.Maintain this pose as long as convenient, but not longer than two minutes, breathe normally slow, rhythmic and natural.\n" +

"6.Return to starting position: slowly bend the knees and then gently lower the hips towards the mat, supported by the hands in 4 seconds, while inhaling.\n" +

"7.Release the hands from the back and assume the starting position.\n" +

"8.Take a few deep breaths and then rest a while, breath normally.\n",R.drawable.sarvangasana));

yoga1.add(new Yoga("Shavasana","1. Savasana improves blood circulation\n" +

"2. It helps in reducing stress.\n" +

"3. Savasana boosts energy.\n" +

"4. Relaxes your muscles.\n" +

"5. Savasana helps reduce headaches.\n" +

"6. Beneficial for those suffering from diabetes & indigestion\n","TITLE","1.Lie flat on your back,Legs should be separated.\n" +

"2.Keep your arms at your side and your palms facing up. Just relax.\n" +

"3.Close your eyes and breathe deeply and slowly through the nostrils.\n" +

"4.Start concentrating from your head to your feet. Don’t move ahead without relaxing particular part of the body.\n" +

"5.On each inhaling and exhaling (breathing) think that your body is totally relaxed. Let your tension, stress, depression and worry run away on each exhaling. You can practice this asana for about

```
3–5 minutes.\n" +
            "\n",R.drawable.shavasana));


    myrecyclerView = (RecyclerView)findViewById(R.id.recyclerView_id);

    myAdapter = new RecyclerViewAdapter(this, yoga1);

    myrecyclerView.setLayoutManager(new GridLayoutManager(this,1));

    myrecyclerView.setAdapter(myAdapter);



  }

}
```

RecyclerViewAdapter.java

```
package com.example.yoga;



import android.content.Context;
import android.content.Intent;
import android.support.annotation.NonNull;
import android.support.v7.widget.CardView;
import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
```

```
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import java.util.List;

public class RecyclerViewAdapter extends RecyclerView.Adapter<RecyclerViewAdapter.MyHolder> {

    private Context mContext;
    private List<Yoga> mData;

    public RecyclerViewAdapter(Context mContext, List<Yoga> mData){
        this.mContext = mContext;
        this.mData = mData;
    }
    @NonNull
    @Override
    public MyHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int i) {

        View view ;
        LayoutInflater layoutInflater = LayoutInflater.from(mContext);
        view = layoutInflater.inflate(R.layout.cardview_yoga,viewGroup,false);
        return new MyHolder(view);
    }
```

```
@Override
    public void onBindViewHolder(@NonNull final MyHolder myHolder, final int i) {

        myHolder.aasanaTitle.setText(mData.get(i).getAasana());
        myHolder.img_recipe_thumbnail.setImageResource(mData.get(i).getThumbnail());
        myHolder.cardView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(mContext, YogaActivity.class);

                intent.putExtra("Aasana",mData.get(i).getAasana());
                intent.putExtra("Benefits",mData.get(i).getBenefits());
                intent.putExtra("MethodTitle",mData.get(i).getMethodTitle());
                intent.putExtra("Method",mData.get(i).getMethod());
                intent.putExtra("Thumbnail",mData.get(i).getThumbnail());

                mContext.startActivity(intent);
            }
        });
    }

    @Override
    public int getItemCount() {
        return mData.size();
```

```
    }
    public class MyHolder extends RecyclerView.ViewHolder {

        TextView aasanaTitle;
        CardView cardView;
        ImageView img_recipe_thumbnail;

        public MyHolder(@NonNull View itemView) {
            super(itemView);

            aasanaTitle = (TextView)itemView.findViewById(R.id.aasana_text);
            img_recipe_thumbnail = (ImageView)itemView.findViewById(R.id.aasana_img_id);
            cardView = (CardView)itemView.findViewById(R.id.cardview_id);


        }
    }
}
```

YogaActivity.java

```
package com.example.yoga;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.TextView;
```

```
public class YogaActivity extends AppCompatActivity {

 private TextView mAasana;
 private TextView mBenefits;
 private TextView mMethodTitle;
 private TextView mMethod;

 @Override
 protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_yoga);

    mAasana = (TextView)findViewById(R.id.text_recipe);
    mBenefits = (TextView)findViewById(R.id.ingredients);
    mMethodTitle = (TextView)findViewById(R.id.method);
    mMethod = (TextView)findViewById(R.id.recipe);

    Intent intent = getIntent();
    String Title = intent.getExtras().getString("Aasana");
    String Benefits =intent.getExtras().getString("Benefits");
    String MethodTitle = intent.getExtras().getString("MethodTitle");
    String Method = intent.getExtras().getString("Method");

    mAasana.setText(Title);
    mBenefits.setText(Benefits);
    mMethodTitle.setText(MethodTitle);
    mMethod.setText(Method);

 }
}
```

Yoga.java

```
package com.example.yoga;

public class Yoga {

    private String Aasana;
    private String Benefits;
    private String MethodTitle;
    private String Method;
    private int Thumbnail;


    public Yoga(String name, String benefits, String methodtitle, String method, int thumbnail){

        Aasana = name;
        Benefits = benefits;
        MethodTitle = methodtitle;
        Method = method;
        Thumbnail = thumbnail;

    }


    public String getAasana(){

        return Aasana;
    }
```

```
 public String getBenefits(){
        return Benefits;
    }

    public String getMethodTitle(){
        return MethodTitle;
    }

    public String getMethod(){
        return Method;
    }

    public int getThumbnail(){
        return Thumbnail;
    }
}
```

Chapter 5

RESULT AND SNAPSHOT

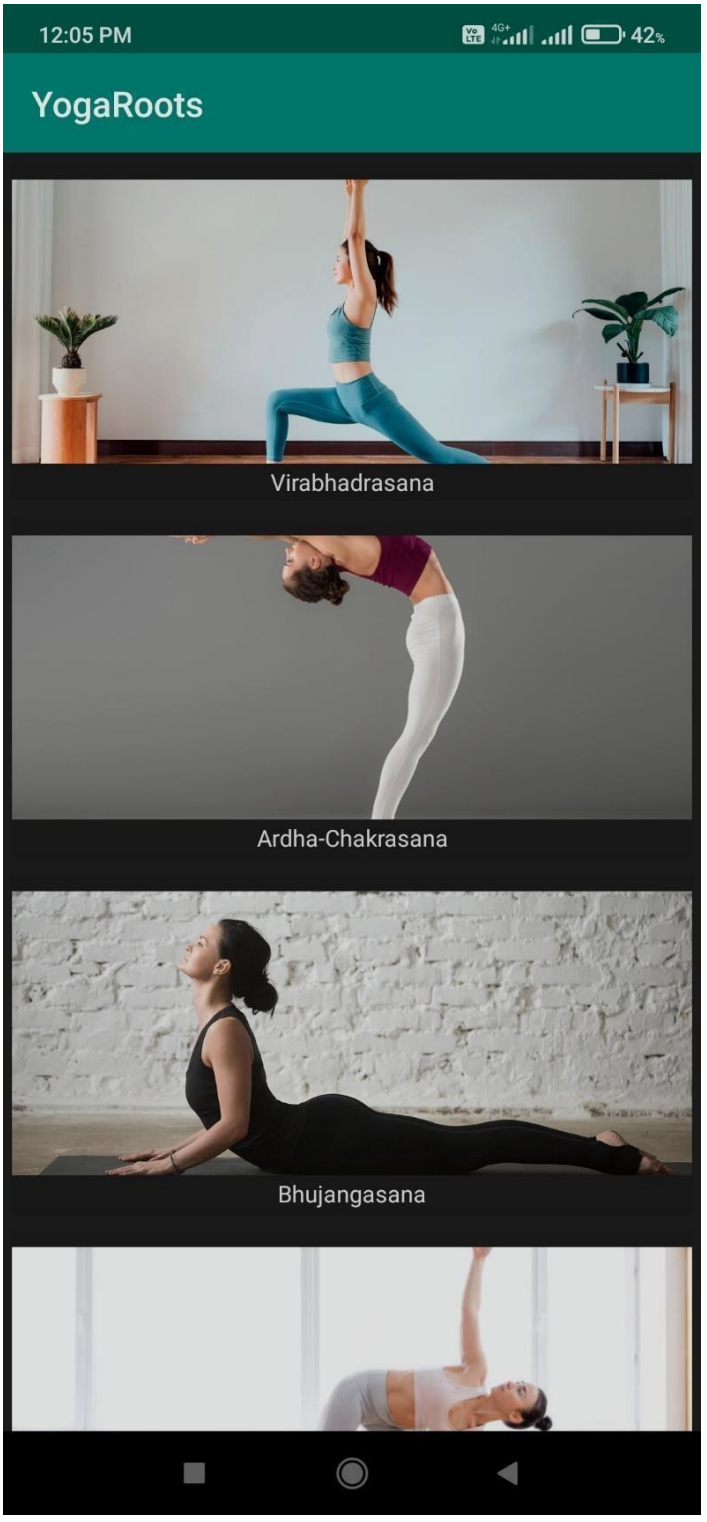

Figure 5.1 home page

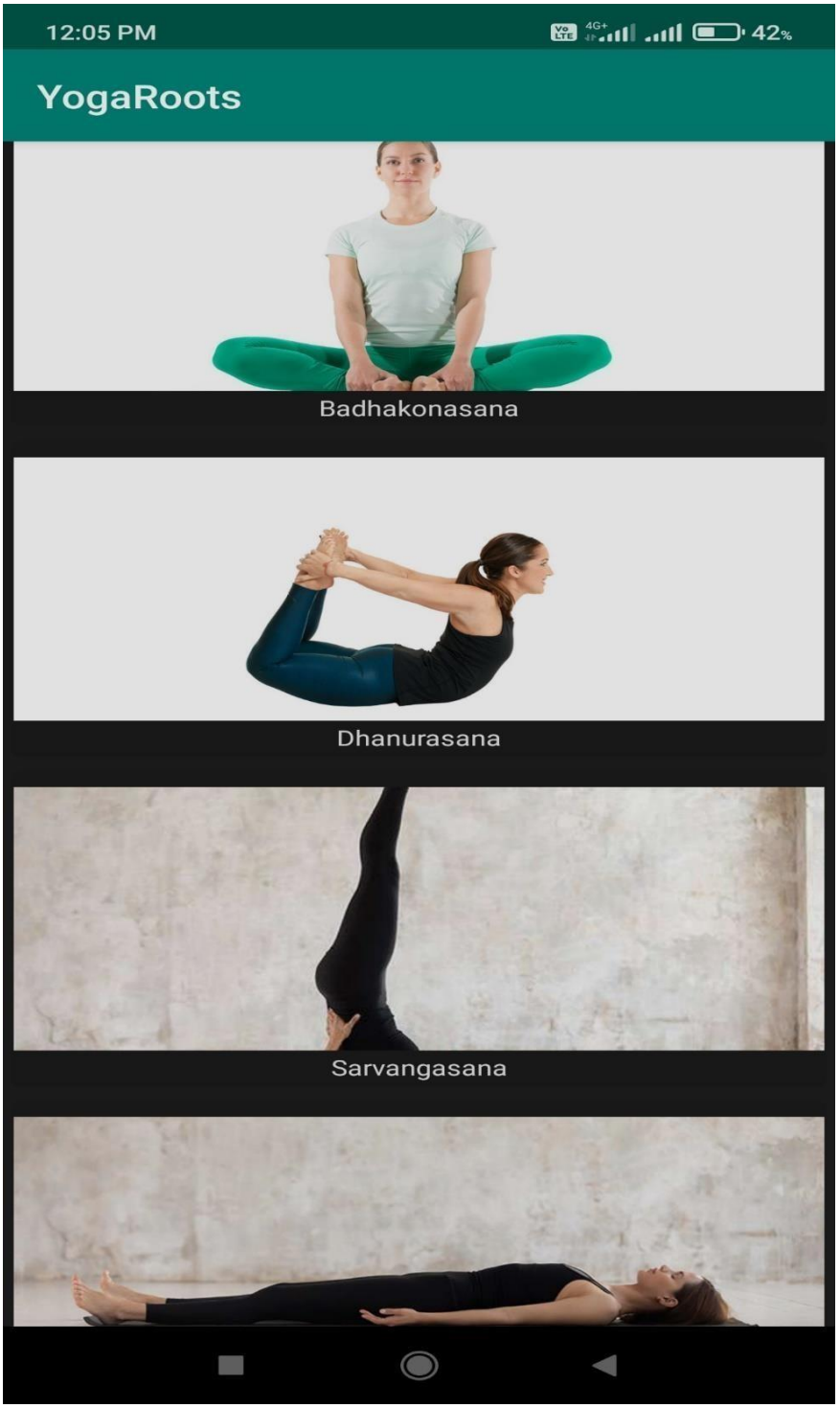

Fig 5.2 Scrollable Home Page

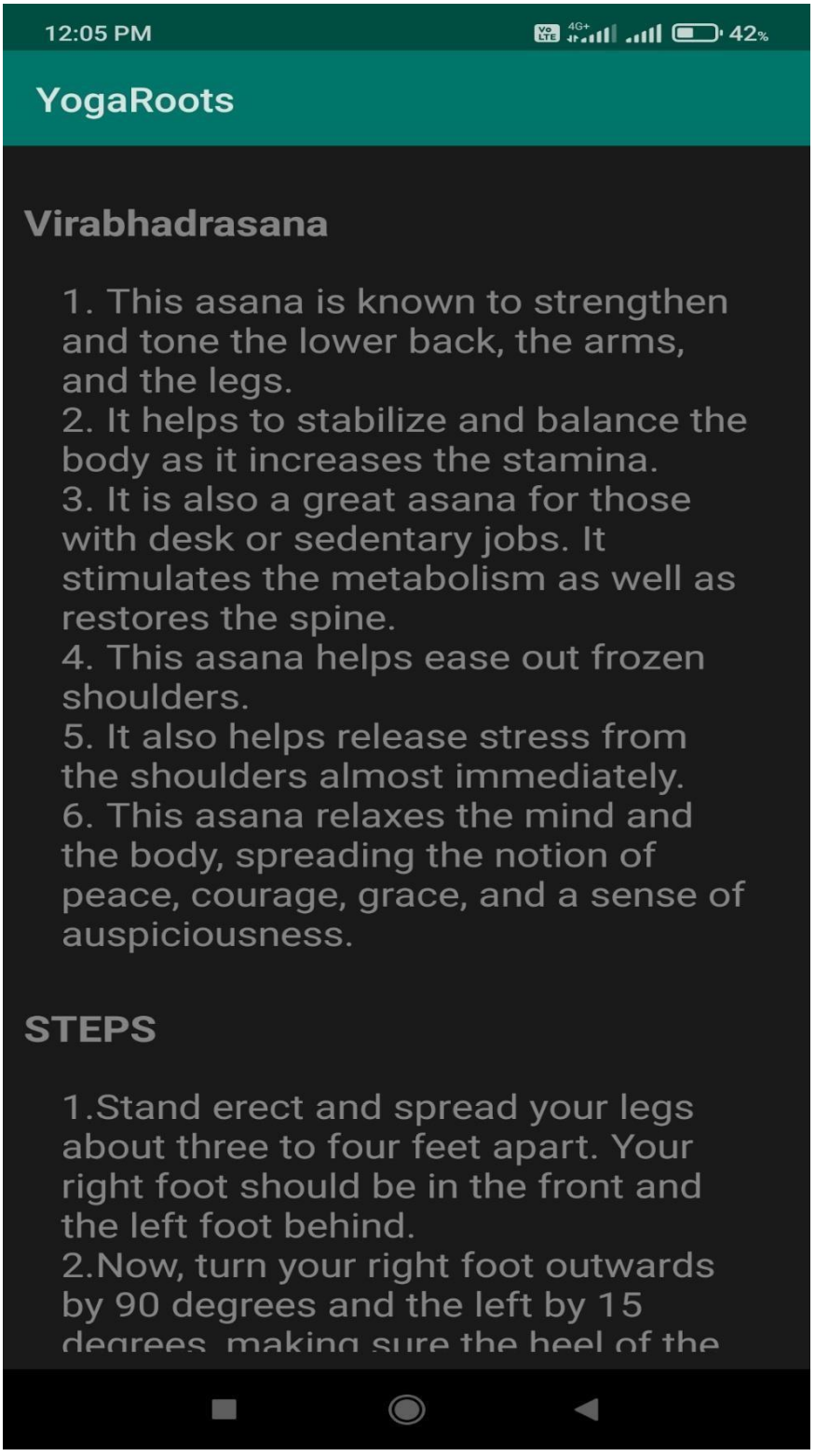

Fig. 5.3 Asana Benefits

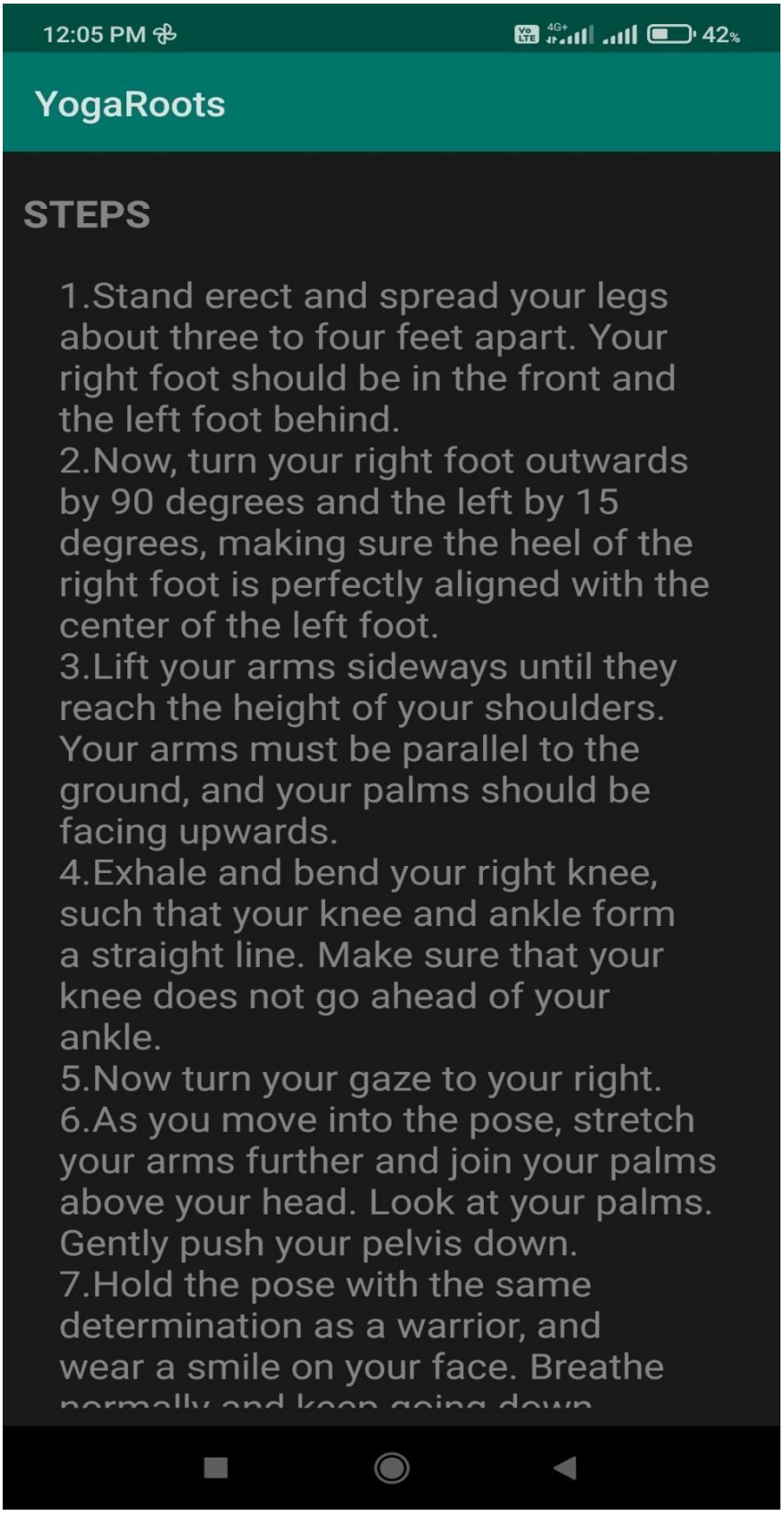

Fig. 5.4 Asana Steps

CONCLUSION AND FUTURE ENHANCEMENT

Conclusion

Fitness Application has been successfully developed using JAVA under Windows platform in Android Studio IDE (Integrated Development Environment). The User can choose the asana of his choice in the home page. The main purpose of this application is to provide the guidance for performing various yoga asanas without attending any in-person classes. The application provides us with various yoga asanas along with its benefits and steps to perform in detail.

Advantages:

• Fast retrieval of information.

• Easy access.

• The user can easily opt for any asana.

Future Enhancement:

For any system, present satisfaction is important, but it is also necessary to see and visualize the future scope. Future enhancement is necessary for any system as the limitations that cannot be denied by anybody. These limitations can be overcome by

• Better technologies.

• Collecting more data.

• Adding new way of collecting the data.

The main goals of this mini-project are

• New Asanas can be efficiently inserted by simple appending the details to the already existing list of asanas.

• It helps to bring N number of asanas under single platform.

REFERENCES

- Android Studio Documentation - https://developer.android.com/docs/
- Java Documentation - https://docs.oracle.com/en/java/
- YouTube - https://www.youtube.com/
- https://stackoverflow.com/
- http://www.oracle.com/technetwork/java/javase/downloads/index-jsp-138363.html.
- B. Khan, "Android Paypal Integration Tutorial," 1 May 2016. [Online]. Available: https://www.simplifiedcoding.net/android-paypal-integration-tutorial/
- "Android Pie Chart using MpAndroidChart Library Tutorial," Numetric Technologies, 29 January 2016. [Online]. Available: Android Pie Chart using AchartEngine Library Tutorial.
- Jackson, Wallace. Learn Android App Development. Apress, 2013.
- https://developer.android.com/guide
- Alubady-Lecture, Raaid. "Understanding mobile programming and applications."
- https://sequencediagram.org/