NEW HORIZON COLLEGE OF ENGINEERING

Autonomous College Permanently Affiliated to VTU, Approved by AICTE & UGC
Accredited by NAAC with 'A' Grade, Accredited by NBA

A MINI-PROJECT REPORT

for

Mini Project in Mobile Application Development (20CSE77A)

on

EASY EATS

Submitted by

Shabrish B Hegde

USN:1NH20CS198, Sem-Sec:7-D

In partial fulfillment of the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

AcademicYear:2023-24(ODD SEM)

NEW HORIZON
COLLEGE OF ENGINEERING
Autonomous College Permanently Affiliated to VTU, Approved by AICTE & UGC
Accredited by NAAC with 'A' Grade, Accredited by NBA

CERTIFICATE

This is to certify that the mini-project work titled

EASY EATS

submitted in partial fulfillment of the degree of Bachelor of Engineering in Computer Science and Engineering by

Shabrish B Hegde
USN:1NH20CS198

DURING

ODD SEMESTER 2023-2024

for

Course: Mini Project in Mobile Application Development -20CSE77A

Signature of Reviewer

Signature of HOD

SEMESTER END EXAMINATION

Name of the Examiner	*Signature with date*
1.	
2.	

PLAF REPORT TO ADD

ABSTRACT

Easy Eats is a revolutionary mobile application designed to transform the culinary journey for users of all expertise levels. This app stands out with its vast collection of thoughtfully curated recipes representing a wide array of global cuisines. Users benefit from personalized recipe suggestions that adapt to individual preferences, ensuring a unique and tailored culinary experience. The app's interactive platform offers seamless, step-by-step guidance, incorporating features such as links to tutorial videos to facilitate an enjoyable cooking process. For those prioritizing health, the app provides comprehensive nutritional insights, and its user-friendly meal planning and scheduling features support a well-rounded approach to culinary exploration. Easy Eats emerges as an indispensable pocket-sized companion, promising a flavorful and personalized adventure in the world of cooking.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be impossible without the mention of the people who made it possible, who's constant guidance and encouragement crowned our efforts with success.

I have great pleasure in expressing gratitude to **Dr. Mohan Manghnani**, Chairman, New Horizon Educational Institutions, for providing necessary infrastructure and creating good environment.

I take this opportunity to express my profound gratitude to **Dr. Manjunatha,** Principal, New Horizon College of Engineering, for the constant support and encouragement.

I would like to thank **Dr. Anandhi R J**, Professor and Dean-Academics, NHCE, for her valuable guidance.

I would also like to thank **Dr.B.Rajalakshmi,** Professor and HOD, Department of Computer Science and Engineering, for the constant support.

I also express my gratitude to **Ms. Shree Nagamanjularani**, Senior Assistant Professor, Department of Computer Science and Engineering, my mini project reviewer, for constantly monitoring the development of the project and setting up precise deadlines. Her valuable suggestions were the motivating factors in completing the work.

Shabrish B Hegde

1NH20CS198

CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 PROBLEM DEFINITION

IT bachelor students frequently encounter difficulties in maintaining a healthy and convenient diet due to their rigorous coursework, limited culinary skills, and busy lifestyles. These challenges lead to suboptimal eating habits, reliance on fast food, and inadequate nutrition, ultimately affecting their well-being and academic performance. To address these issues, there is a compelling need for a specialized food recipe app designed specifically for IT bachelor students.

1.2 OBJECTIVES

Following objectives define the main functionality of the app:

- The app will serve as an educational resource, equipping students with enhanced cooking skills through step-by-step guides and instructional videos.
- This will lead to a significant shift towards healthier eating habits, reducing the reliance on fast food and promoting the consumption of nutritious meals.

CHAPTER 2

ABOUT ANDROID

2.1 ANDROID

Android is a software that is open source, which means that anyone can take the software and use it and it is also based on the operating system called Linux. It is used to make applications that are supposed to run on mobile gadgets that have screens of various sizesand the application adjusts accordingly to the varied sizes of the screens. Android helps indevelopment of applications and the applications so developed can be uploaded on Google Play Store and the everyone will be able to use the application.

2.2 Android OS features

The UI of Android licenses wellsprings of data like tapping, swiping and crushing to beginexercises. Response to exercises can integrate vibrations, takes note. Application madein Android is open in Google Play and can be downloaded. Android runs on comprehensively conveyed standards like GSM/HSDPA and CDMA/EV-DO standards. It maintains Bluetooth, Edge, 3G/4G/5G correspondence shows, Wi-Fi, SMS, GPS, etc.

2.3 Android Versions

Adaptations of Android are given express name and supports unequivocal devices.Two or three examples of Android variations are as given under: -
Android 1.0 Conveyed in 2008 with applications like Gmail, Guides, Timetable and YouTube. Android 4.4 was called KitKat; Android 5 was called Treats. Android 9 was called Pie. Continuous conveyed of Android can't avoid being Android 12 (Shaved ice) and Latest version is Android 13(Tiramisu)

2.4 ACTIVITY LIFE CYCLE

Android Action life Cycle is constrained by techniques for Android.app.Activity Class. Theexistence pattern of a movement are as per the following: -

- OnCreate: This technique is called when the Movement is first made.
- OnStart: This technique is called when the Movement is noticeable to the client.
- OnResume: Called when movement is associating with client.
- OnPause: Called when movement isn't noticeable to the client.
- OnStop: Called when movement is as of now not accessible to the client.
- OnRestart: Called when Movement is halted before start.
- OnDestroy: Called before Movement is annihilated

2.5 Android User Interface Layouts.

UI is characterized as View Gatherings and View. View Gathering is a compartment that characterizes the design for a View.

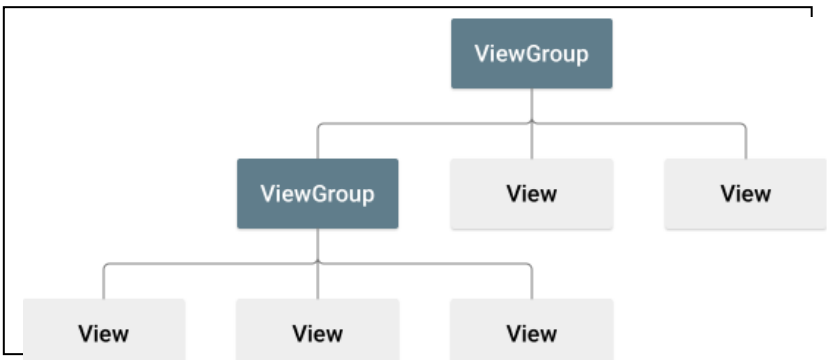

Fig 2.5 Android user Layouts

View Group examples are: Linear Layout, Relative Layout, Constraint Layout etc. Viewexamples are: TextView, ImageView, Button, etc.

2.6 JAVA

Java is a popular and flexible programming language for developing mobile apps, providing a stable framework for making apps that function well across a range of platforms. Java is well known for being portable and platform agnostic, allowing programmers to create code once and have it run on several different systems. Through the Android SDK, one of the most widely used mobile operating systems, Android, significantly depends on Java for app development. Because Java is object-oriented, it makes development simpler by allowing for reusable code components and making maintenance simpler.

2.7 KOTLIN

It has become clear that Kotlin is a strong and flexible programming language for creating mobile applications, especially for the Android operating system. Kotlin, acknowledged by Google as the official language for Android, has a clear and expressive syntax that minimizes boilerplate code and increases developer productivity. Developers with current Java codebases may easily convert to it thanks to its compatibility with Java. Modern Kotlin features like extension functions and null safety help to write more dependable and readable code. Because of its smooth interaction with well-known frameworks and development tools, Kotlin is a great option for creating reliable and effective mobile apps. It gives developers the tools they need to efficiently and clearly construct high-quality Android apps

2.8 XML

The structure and content of user interfaces are defined by the flexible markup language known as XML (eXtensible Markup Language), which is an essential part of the development of mobile apps. XML is frequently used in the context of mobile development to create layouts for Android applications using XML-based layout files. These files make it easier for developers and designers to collaborate by clearly separating the presentation from the business logic. Because of its ease of use and readability, XML is a great option for defining views, expressing user interface elements, and defining attributes like styles and layout parameters. Developers can maintain a modular and maintainable codebase while producing aesthetically pleasing and responsive mobile app interfaces by utilising XML.

CHAPTER 3

REQUIREMENT SPECIFICATION

3.1 HARDWARE REQUIREMENTS

- Android device (smartphone or tablet) running Android OS 8.0 or above.
- RAM 4GB or Higher.
- Hard Disk: 25MB or Higher.

3.2 SOFTWARE REQUIREMENTS

- Windows 8 or above.
- Android Studio: The code is written in Java and uses the Android framework, so you need Android Studio IDE to build and run the application.
- Android SDK: The project requires Android SDK tools and platform components to compile and run
- AndroidX libraries: The code uses AndroidX libraries for data binding and compatibility.

CHAPTER 4

DESIGN

4.1 FlowChart

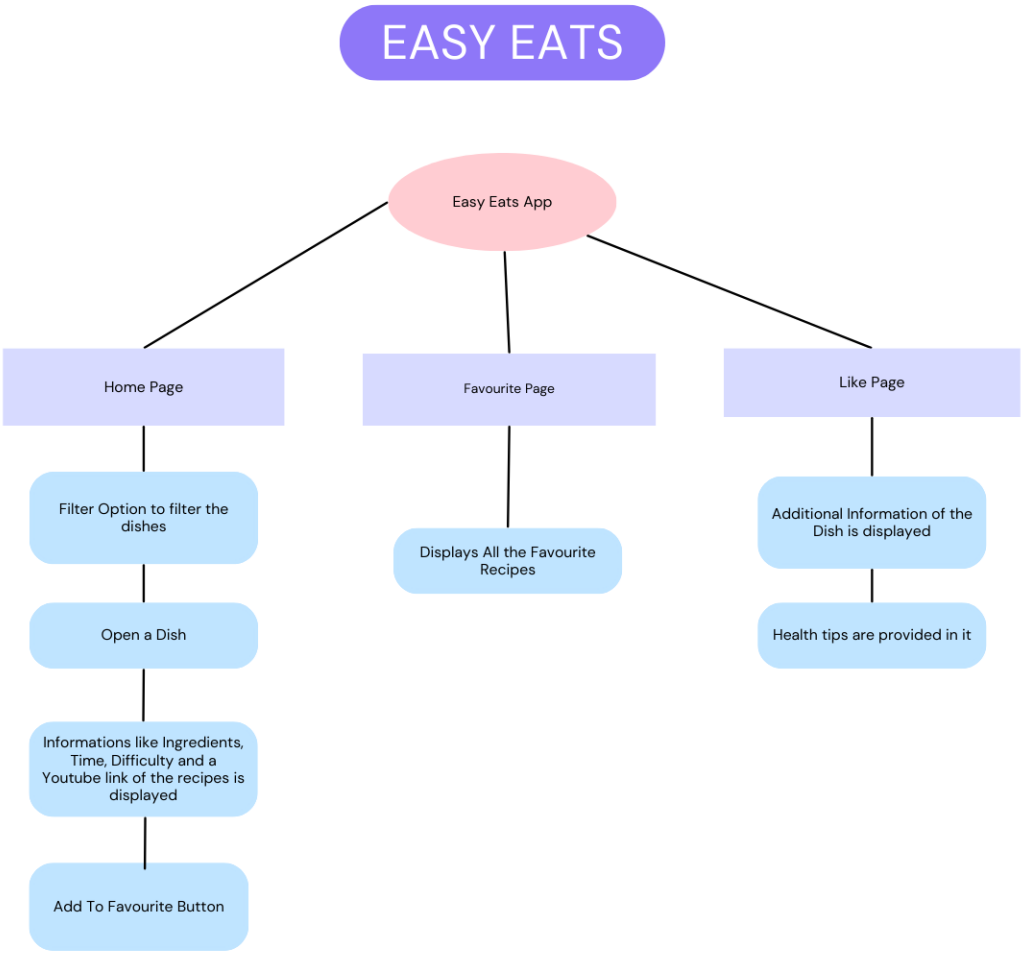

Fig 4.1 Flowchart of the Application

CHAPTER 5

IMPLEMENTATION

5.1 Main Activity:

This activity consists of all the Recipes for the users to access. Additionally, the users can filter the recipes based on their interests.

```
public class MainActivity extends AppCompatActivity {
    ImageView imageviewclick;
    RecyclerView categoryRecycler, courseRecycler;
    CategoryAdapter categoryAdapter;
    static CourseAdapter courseAdapter;
    static List<Course> courseList = new ArrayList<>();
    static List<Course> fullcoursesList = new ArrayList<>();
    @SuppressLint("MissingInflatedId")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        imageviewclick  = findViewById(R.id.imageclick);
        imageviewclick.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                courseList.clear();
                courseList.addAll(fullcoursesList);
                courseAdapter.notifyDataSetChanged();
            } });
        List<Category> categoryList = new ArrayList<>();
        categoryList.add(new Category(1, "Soups"));
```

5.2 CoursePage Activity:

This activity is for course recipe page . It provides all the required Information.

```
public class CoursePage extends AppCompatActivity {

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

    ConstraintLayout courseBg = findViewById(R.id.coursePageBg);
    ImageView courseImage = findViewById(R.id.coursePageImage);
    TextView courseTitle = findViewById(R.id.coursePageTitle);
    TextView courseDate = findViewById(R.id.coursePageDate);
    TextView courseLevel = findViewById(R.id.coursePageLevel);
    TextView courseText = findViewById(R.id.coursePageText);
    TextView link = findViewById(R.id.link);

    String url = "http://" + getIntent().getStringExtra("Link");

    courseBg.setBackgroundColor(getIntent().getIntExtra("courseBg",0));
    courseImage.setImageResource(getIntent().getIntExtra("courseImage",0));
    courseTitle.setText(getIntent().getStringExtra("courseTitle"));
    courseDate.setText(getIntent().getStringExtra("courseDate"));
    courseLevel.setText(getIntent().getStringExtra("courseLevel"));
    courseText.setText(getIntent().getStringExtra("courseText"));
    link.setText(getIntent().getStringExtra("Link"));
```

5.3 OrderPage Activity:

This activity is for sending the intent to the RecommendationPage and MainActivity Page.

```
public class OrderPage extends AppCompatActivity {
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_order_page);
      ListView orders_list = findViewById(R.id.recom_list);
      Set<Integer> addedCourseIds = new HashSet<>();
      List<String> coursesTitle = new ArrayList<>();
      for(Course c : MainActivity.fullcoursesList){
         if(Order.items_id.contains(c.getId()) && !addedCourseIds.contains(c.getId())) {
            coursesTitle.add(c.getTitle());
            addedCourseIds.add(c.getId());} }
      orders_list.setAdapter(new ArrayAdapter<>(this,
android.R.layout.simple_list_item_1, coursesTitle));}
public void fromLiketoRec(View view){
      Intent intent = new Intent(this, RecomendationPage.class);
      MainActivity.courseList.clear();
      MainActivity.courseAdapter.notifyDataSetChanged();
      startActivity(intent);}
   public void fromLiketoMain(View view){
      Intent intent = new Intent(this, MainActivity.class);
      MainActivity.courseList.clear();
      MainActivity.courseAdapter.notifyDataSetChanged();
      startActivity(intent);
   }}
```

5.4 RecomendationPage Activity:

This activity is for sending the intent to MainAcitvity and OrderPage.

```
public class RecomendationPage extends AppCompatActivity {
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_recom);

   }
      public void fromRectoMain (View view){
         Intent intent = new Intent(this, MainActivity.class);
         startActivity(intent);
      }
      public void fromRectoLike (View view){
         Intent intent = new Intent(this, OrderPage.class);
         startActivity(intent);
      }

}
```

5.5 RecPage Activity:

This activity is used for the final page in the application where there is a health benefit tips given for a dish.

```
public class recpage extends AppCompatActivity {
   private TextView newText;
   @SuppressLint("MissingInflatedId")
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_recpage);
      newText = findViewById(R.id.RecPageText);
      newText.setText("We’ve all been there.\n" +
            "\n" +
            "“I think I caught what’s going around the office,” \n or “I just have a tickle in my throat.”\n" +
            "\n" +
            "As the leaves fall and temperatures drop, people tend \n to spend more time indoors with others, which potentially increases the exposure to cold-causing viruses. At some point this season, most people will catch the common cold, a viral infection in the upper respiratory system characterized by coughing, sneezing, sore throat, and other symptoms.\n" +
            "\n" +
            "While medicine and getting a good night’s sleep can help \n , it’s no wives tale that the simple remedy of a bowl of soup can also help. The prescription predates even the oldest secret family recipe for chicken noodle soup; in the 12th century, Egyptian Jewish physician Moshe ben Maimonides prescribed chicken soup as a treatment for respiratory tract issues.\n" +
            "\n" +}}
```

5.6 CourseAdapter Activity:

```
public static final class CourseViewHolder extends RecyclerView.ViewHolder {

    CardView courseBg;
    ImageView courseImage;
    TextView courseTitle, courseDate, courseLevel;

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

      courseBg = itemView.findViewById(R.id.courseBg);
      courseImage = itemView.findViewById(R.id.courseImage);
      courseTitle = itemView.findViewById(R.id.courseTitle);
      courseDate = itemView.findViewById(R.id.courseDate);
      courseLevel = itemView.findViewById(R.id.courseLevel);
    }
  }
}
```

5.7 CategoryAdapter :

```
public class CategoryAdapter extends
RecyclerView.Adapter<CategoryAdapter.CategoryViewHolder> {

        Context context;
        List<Category> categories;

        public CategoryAdapter(Context context, List<Category> categories) {
          this.context = context;
          this.categories = categories;
        }

        @NonNull
        @Override
        public CategoryViewHolder onCreateViewHolder(@NonNull ViewGroup parent,
int viewType) {
          View categoryItems =
LayoutInflater.from(context).inflate(R.layout.category_item, parent, false);
          return new CategoryViewHolder(categoryItems);
        }

      @Override
        public void onBindViewHolder(@NonNull CategoryViewHolder holder,
@SuppressLint("RecyclerView") int position) {
          holder.categoryTitle.setText(categories.get(position).getTitle());

          holder.itemView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
              MainActivity.showCoursesByCategory(categories.get(position).getId());
            }
          });

        }
```

5.8 Gradle Script:

This gradle includes all the dependencies required to run this application.

```
android {
   namespace 'com.example.mainproject'
   compileSdk 32
   defaultConfig {
      applicationId "com.example.mainproject"
      minSdk 21
      targetSdk 32
      versionCode 1
      versionName "1.0"
      testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
      vectorDrawables.useSupportLibrary = true}
   buildTypes {
      release {
         minifyEnabled false
         proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-
rules.pro' }}
   compileOptions {
      sourceCompatibility JavaVersion.VERSION_1_8
      targetCompatibility JavaVersion.VERSION_1_8}}
dependencies {
   implementation 'androidx.appcompat:appcompat:1.5.1'
   implementation 'com.google.android.material:material:1.6.1'
   implementation 'androidx.constraintlayout:constraintlayout:2.1.4'
   testImplementation 'junit:junit:4.13.2'
   androidTestImplementation 'androidx.test.ext:junit:1.1.3'
   androidTestImplementation 'androidx.test.espresso:espresso-core:3.4.0'}
```

CHAPTER 6

OUTPUT

6.1 Home Page:

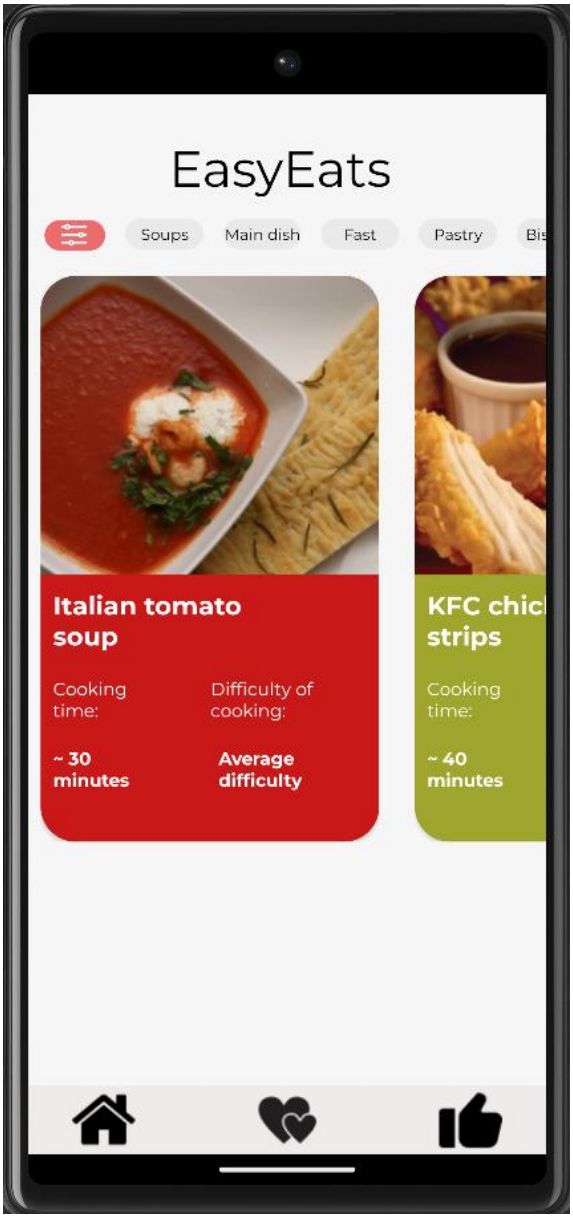

Fig 6.1 Home Page

6.2 Filter Option for the courses:

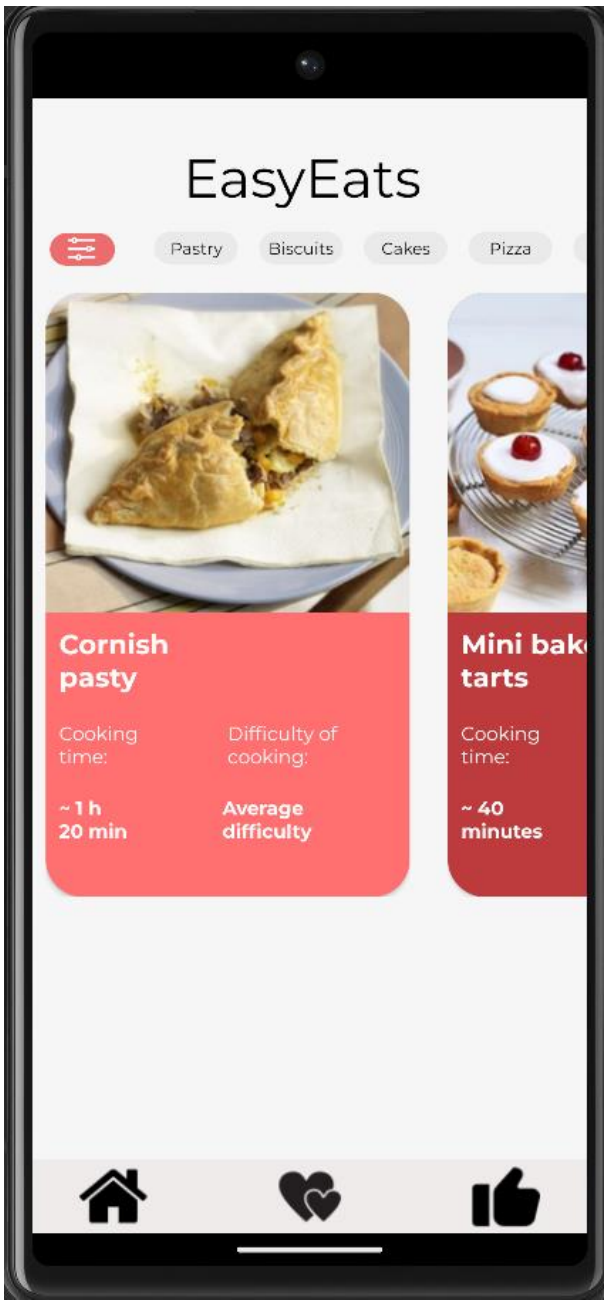

Fig 6.2 Filter Function

6.3 Course Recipe Page:

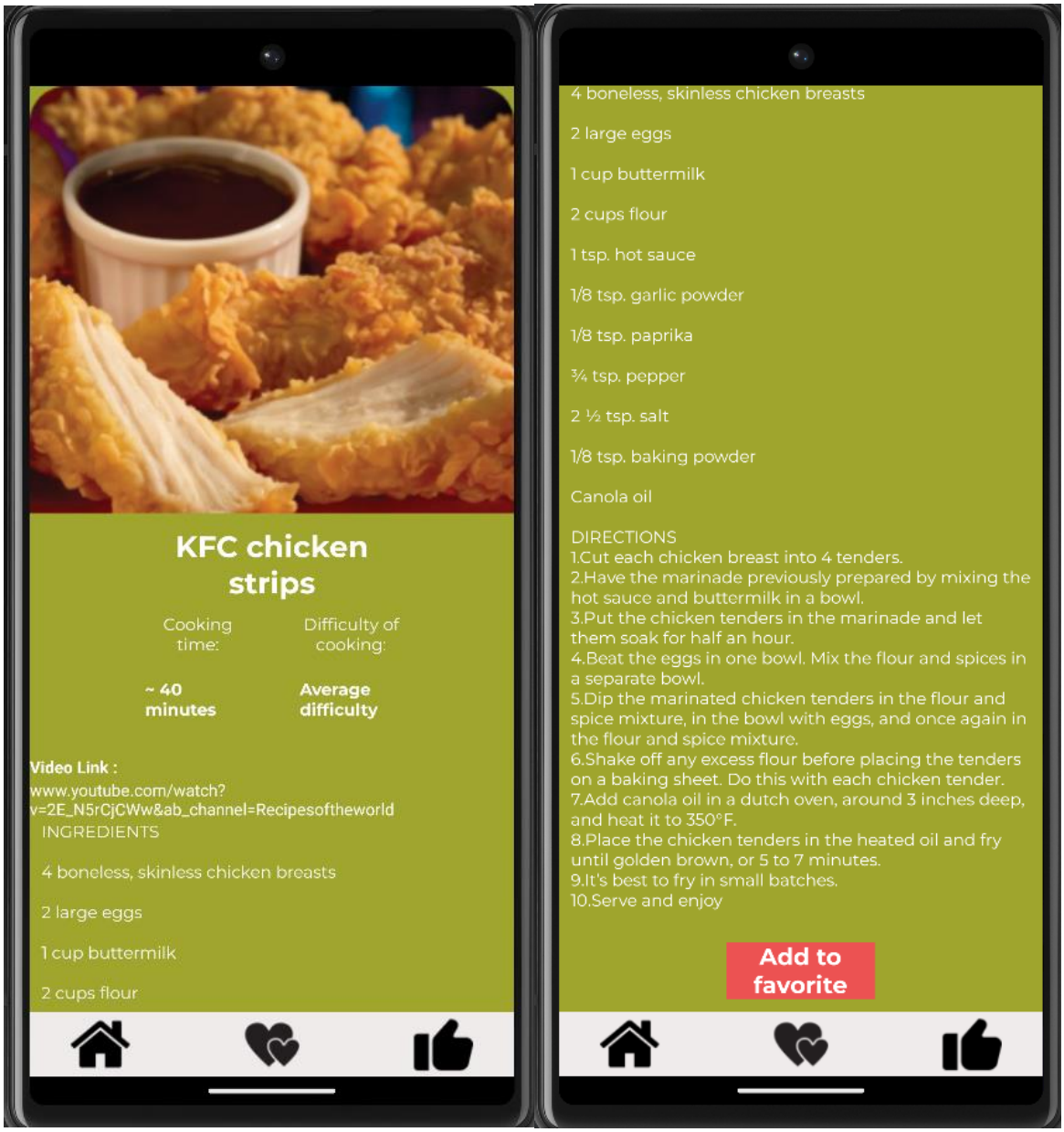

Fig 6.3 Recipe Page

6.4 Youtube Link For The Recipe :

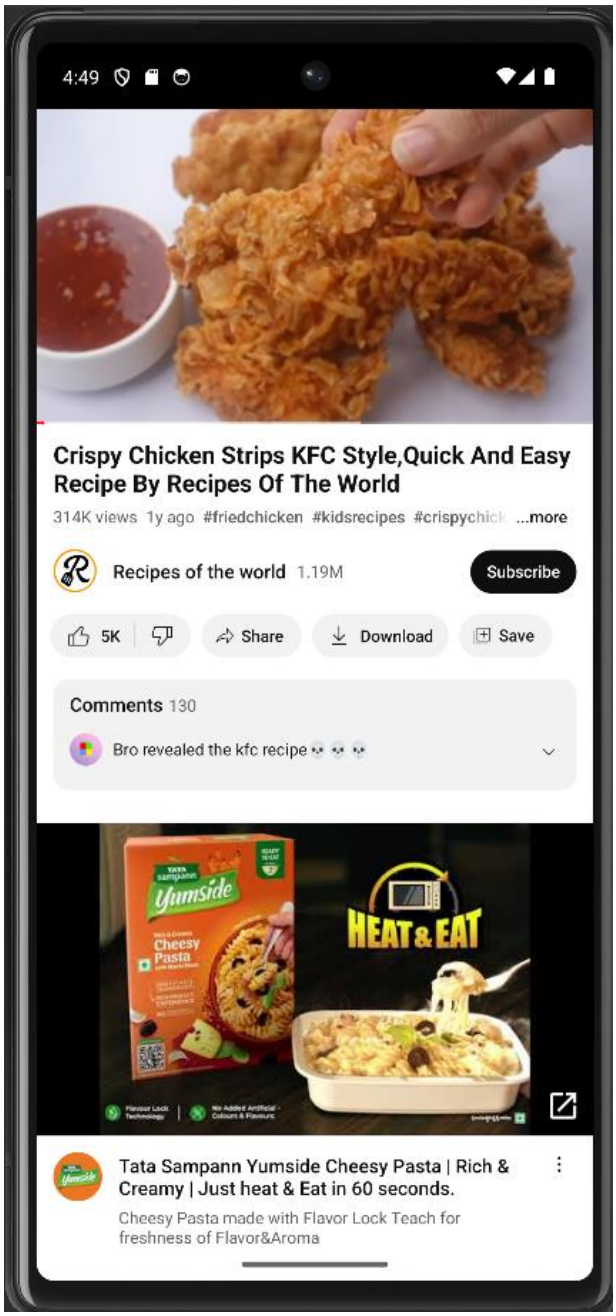

Fig 6.4 Youtube Link for Recipe

6.5 Favourite Recipe Page:

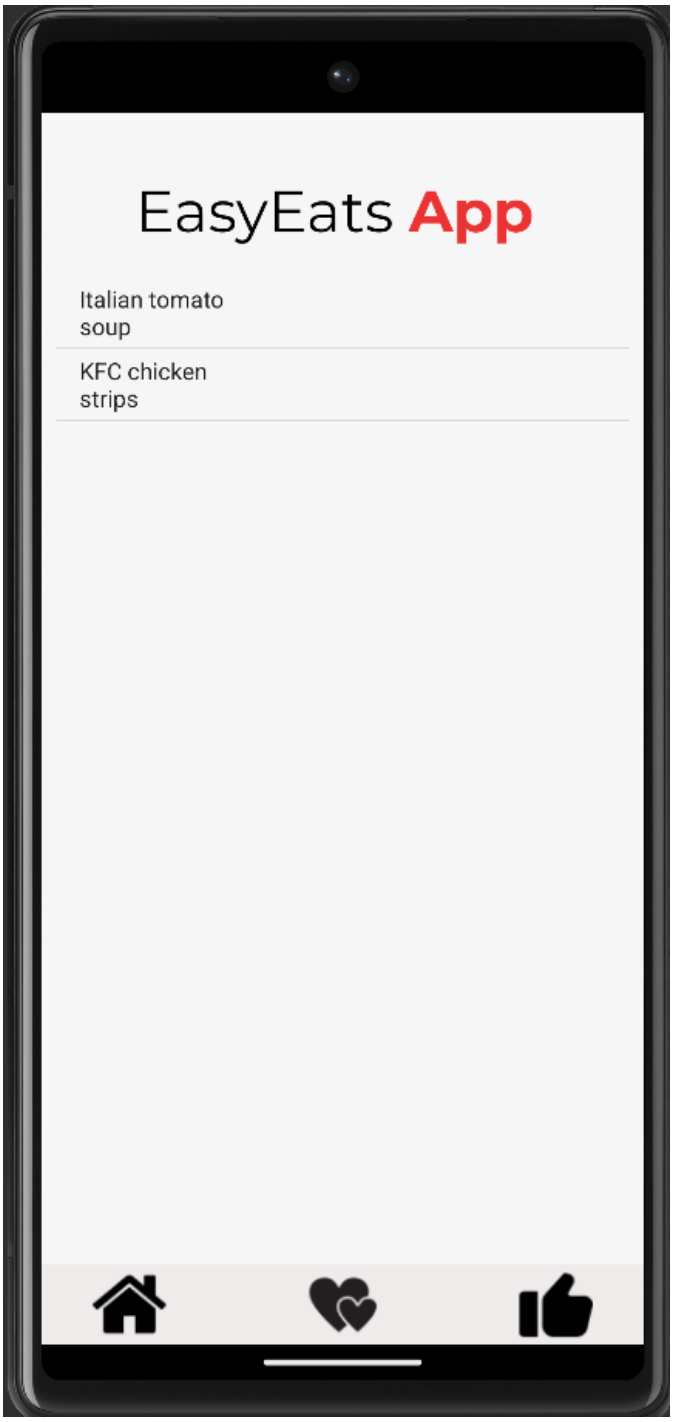

Fig 6.5 Favourite Recipe Page

6.6 Food Tips Page:

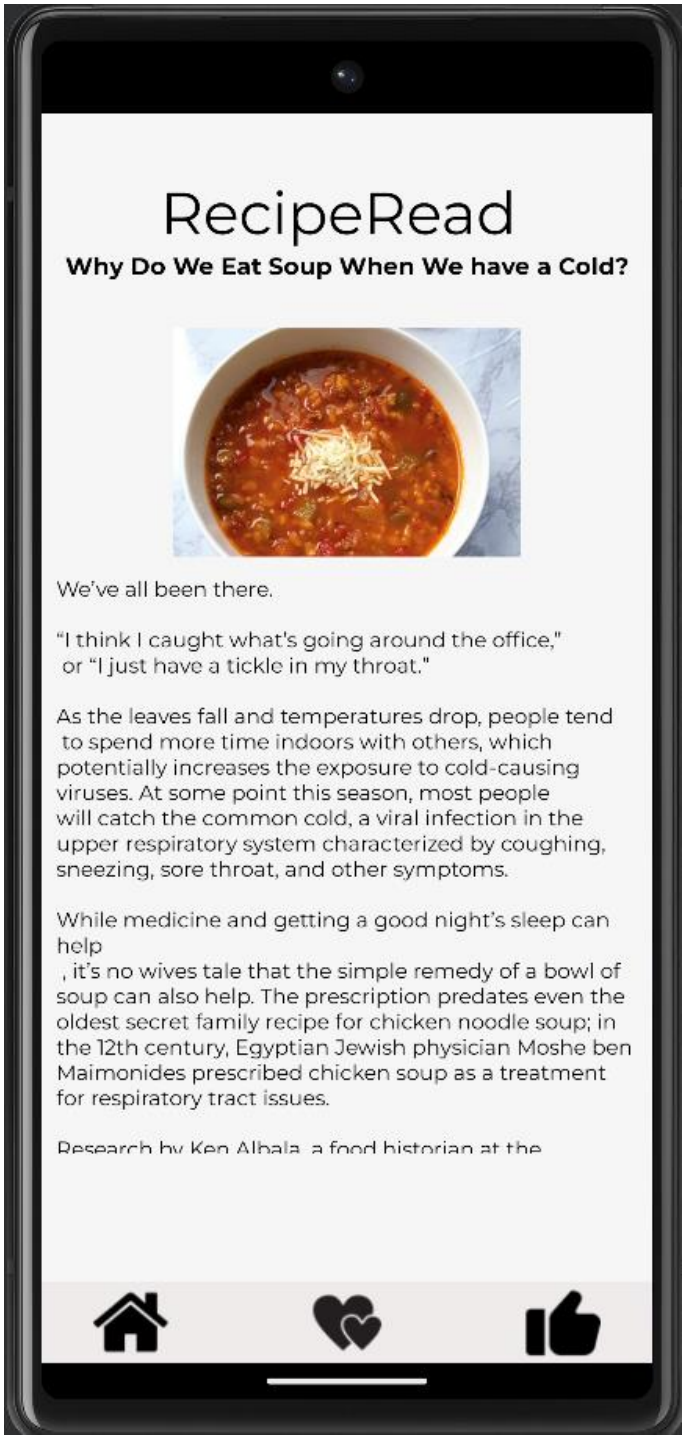

Fig 6.6 Food Tips Page

CHAPTER 7

CONCLUSION

In conclusion, our recipe application strives to redefine your culinary experience by seamlessly integrating two key components: direct links to YouTube recipe videos and a personalized favorites feature.

With our unique approach of embedding YouTube links alongside traditional written instructions, we aim to provide users with a comprehensive and engaging cooking experience. This multimedia feature allows users to benefit from both detailed written guidance and visual demonstrations, catering to diverse learning preferences. Whether you are a novice or an experienced chef, this approach ensures a dynamic and enriching cooking journey.

Additionally, the favorites option adds a layer of personalization to the user experience. This feature enables users to curate a collection of their preferred recipes, streamlining the meal planning process. Whether it's a quick weeknight dinner or a special occasion dish, the favorites section serves as a virtual cookbook, facilitating easy access to chosen recipes with just a few clicks.

In essence, our recipe application not only offers a diverse range of culinary inspiration through YouTube links but also empowers users to build and organize their culinary repertoire with a straightforward and intuitive favorites feature. Enhance your cooking adventure with our innovative and user-friendly recipe application.

REFERENCES

[1] Android Studio Documentation: https://developer.android.com/docs. Accessed on: June-2023.

[2] https://developer.android.com/develop/ui/views/layout/recyclerview#:~:text=RecyclerView%20is%20the%20ViewGroup%20that,by%20a%20view%20holder%20object.

[3] https://developer.android.com/reference/android/content/Intent

[4] Erik Hellman, "Android Programming – Pushing the Limits", 1st Edition, Wiley IndiaPvt Ltd, 2014. ISBN-13: 978-81265471