VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belgaum-590018, Karnataka

A MOBILE APPLICATION DEVELOPMENT MINI PROJECT REPORT ON

"WALLPAPER APP"

Submitted in partial fulfillment for the requirement for sixth semester of

BACHELOR OF ENGINEERING

In

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

By

SIDDARTHA R(1RR20AI029)

Under the Guidance of

Dr K RAJESH
Professor & HOD,
Dept. of AIML,
RRCE

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

RAJARAJESHWARI COLLEGE OF ENGINEERING
MYSORE ROAD, BANGALORE -560074

RAJARAJESHWARI COLLEGE OF ENGINEERING
MYSORE ROAD, BANGALORE – 560 074.

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

CERTIFICATE

This is to certify that the Mini project synopsis entitled **“WALLPAPER APP”** is a bonafide work carried out by **SIDDARTHA R(1RR20AI029)** in partial fulfilment of award of Degree of **Bachelor of Engineering in Artificial Intelligence and Machine Learning** of Visvesvaraya Technological University, Belagavi, during the academic year 2022-2023. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated. The Mini project has been approved as it satisfies the academic requirements in respect of mini project work for sixth semester.

Signature of HOD

DR.K.RAJESH.
Professor &HOD,
Dept. of AI&ML,
RRCE, Bengaluru.

Signature of Principal

Dr. R.BALAKRISHNA
Principal,
Dept. of AI&ML,
RRCE, Bengaluru.

RAJARAJESWARI COLLEGE OF ENGINEERING
MYSORE ROAD, BANGALORE– 560 074.

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

DECLARATION

I **SIDDARTHA R (1RR20AI029),** student of Sixth semester, **DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**, Rajarajeswari College of Engineering, Mysore Road Bangalore, declare that the project work entitled **"WALLPAPER APP"** has been carried out by us and submitted in partial fulfilment of the course requirements for the award of degree in Bachelor of Engineering in Artificial Intelligence And Machine Learning of Visvesvaraya Technological University, Belagavi during the academic year **2022-2023**. The matter embodied in this report has not been submitted to any other university or institution for the award of any other degree or diploma.

SIDDARTHA R(1RR20AI029)

ACKNOWLEDGEMENT

Here by we are submitting the Mini project report on **“WALLPAPER APP”**, as per the scheme of Visvesvaraya Technological University, Belagavi.

In this connection, we would like to express our deep sense of gratitude to our beloved institution Rajarajeshwari College of Engineering and also, we like to express our sincere gratitude and indebtedness to **Dr. R Balakrishna,** Principal, RRCE, Bengaluru.

We would like to express our sincere gratitude to **Dr. K Rajesh, Prof. & HOD, Department of Artificial Intelligence and Machine Learning**, RRCE, for providing a congenial environment to work in and carry out our project.

Finally, we are very much thankful to all the teaching and non-teaching members of the **Department of Artificial Intelligence and Machine Learning** for their constant encouragement, support and help throughout the completion of report.

SIDDARTHA R (1RR20AI029)

ABSTRACT

Android is open source code mobile phone operating system that comes out by Google in November 2007. Its appearance has broken the traditional closed mobile phone operating system. Anyone can modify the mobile phone operating system as well as function according to personal preference, which is also the most attractive merit of Android. Wallpaper App is an application software based on Google Android. This Project is on “WALLPAPER APP”. In order to solve the problem of complex functions and large required memory of mobile phone to change the wallpaper of the home screen which makes it simple, convenient, and requires less memory as well as it is user friendly is developed. Based on the Android technology, using the Java language led to design and coding of Wallpaper App. The new design mainly realizes core functions mainly since the app is connected to the Google pixel interface. The app on its own loads the images by the google pixel and presents it to the user making the user having extreme options to select from the app. This app has merits of high performance, simple operation, and run independently on the Android mobile devices making it much more user friendly.

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1 ANDROID:

Android is a mobile operating system (OS) currently developed by Google, based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. Android's user interface is mainly based on direct manipulation, using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with a virtual keyboard for text input. In addition to touchscreen devices, Google has further developed Android TV for televisions, Android Auto for cars, and Android Wear for wrist watches, each with a specialized user interface. Variants of Android are also used on notebooks, game consoles, digital cameras, and other electronics. Initially developed by Android, Inc., which Google bought in 2005, Android was unveiled in 2007, along with the founding of the Open Handset Alliance – a consortium of hardware, software, and telecommunication companies devoted to advancing open standards for mobile devices. As of July 2013, the Google Play store has had over one million Android applications ("apps") published, and over 50 billion applications downloaded. An April–May 2013 survey of mobile application developers found that 71% of developers create applications for Android, and a 2015 survey found that 40% of full-time professional developers see Android as their priority target platform.

Figure 1.1: Android Image

1.2 SOFTWARE DEVELOPMENT KIT:

A software development kit (SDK or "devkit") is typically a set of software development tools. that allows the creation of applications for a certain software package, software framework, hardware platform, computer system, video game console, operating system, or similar development platform.

1.3 ANDROID STUDIO:

Android Studio is an integrated development environment (IDE) for developing the Android platform. It was announced on May 16, 2013, at the Google I/O conference. Android Studio is freely available under the Apache License 2.0. Android Studio was in early access preview stage starting from version 0.1 in May 2013, then entered the beta stage starting from version 0.8 which was released in June 2014.

It is available for download on Windows, Mac OS X and Linux, and replaced Eclipse Android Development Tools (ADT) as Google's primary IDE for native Android application development.

Figure1.3: Android Studio

1.4 THE REQUIRED SOFTWARE:

1. Operation system: Windows 10, Linux
2. Software: Android SDK (Software Development Kit), ADT (Android Development Tool)
3. JDK: Java Runtime Environment Virtual Machine, Java Development Kit(JDK).

1.5 INSTALLATION STEPS FOR DEVELOPING ENVIRONMENT

1. Step 1: install the Java virtual machine JDK version – 7
2. Step 2: install the Android SDK: first download the Android SDK
3. Download address: http://developer-android-com/sdk/index-html
4. Input SDK tools path in the SDK location: D: \ android \ software \ android SDK– Windows and click OK.
5. The Android environment is set up successfully.

CHAPTER 2
SYSTEM SPECIFICATIONS

2.1 Hardware Requirements:

a. This APP meets minimum Hardware Requirements:

b. 1 GHz Processor

c. 512MB of RAM

d. 50MB of Internal Storage.

2.2 Software Requirements:

a. This APP meets minimum Software Requirements

b. Android Studio (any version)

CHAPTER 3

DESIGN

3.1 The Design Principle of Android Application

The principle of software design mainly includes the following points:

1. **Reliability**
   The reliability of the software design must be determined. The reliability of the software system refers to the ability to avoid fault occurred in the process of system running, as well as the ability to remedy troubles once the fault occurs.

2. **Reusability**
   Look for commonness of similar codes, and come out new method abstractly and reasonably. Pay attention to the generic design.

3. **Understandability**
   The understand ability of software not only require clear and readable document, but the simplified structure of software itself, which requires the designer possess keen insight and creativity, and know well about the design objects.

4. **Simple program**
   To keep the program simple and clear, good programmers can use simple program to solve complex problems.

5. **Testability**
   Testability means that the created system has a proper data collection to conduct a comprehensive test of the entire system.

6. **The Open-Closed Principal**
   Module is extensible but cannot be modified. That is to say, extension is open to the existing code in order to adapt to the new requirements.

3.2 Function and Structure Design of Android System

This system adopts the modularized program design, and system function is correspondingly divided into function modules, the main modules include:

1. **function module design of mobile terminal**: The home screen, **UI** settings screen page are realized.
2. **Backstage function module design of mobile terminal**: The specific function, Database function and other function are implemented.

3.3 Requirement Analysis of System

The feasibility analysis:

This section verified that it is feasible to add music player on the Android system from the aspects of economic, technical and social feasibility.

1. **Economic feasibility:**
   To design Android mobile phone music player as long as a computer has the Android development and the application development of Android is free. In addition, mobile phone radio music player is basic needs for public. And a lot of research is eliminated, thus saved the spending. Therefore, the whole process of development doesn't need to spend any money that is economic feasibility.

2. **Technical feasibility:**
   To design a radio music player which meets the basic requirements, a deep understand of JAVA language, the Android system architecture, application of framework and other technical knowledge are needed. (framework is the core of the application, and rules that all the programmers participating in the development must abide by).

3. **Social Feasibility:**
   With the rapid development of the mobile phone market, all kinds of news resources are widely circulated on the Internet. These resources

seem ordinary, but have gradually become an indispensable part of people life, which derived the development of all kinds of mobile phone player. But a lot of news apps devoted to fancy appearance, strong function causing a lot of wasted resources to the user's mobile phone and bringing a lot of inconvenience to the user as multitasking operation is needed. Some functions are useless to ordinary people.

4. **Saturation Overview:**
   This section describes requirements of the system based on basic control functions of types, and system setup function of the app according to research results of the project demand. According to the research results of project demand, the basic requirements of project system and its function structure are presented.

3.4 System Design

Any App Starting needs Android Manifest. XML file to start. And any new project content will automatically generate an Android Manifest. XML file. Configuration files are the core of the whole program, which contains the Android SDK version, and the default Activity in program running. The systems will automatically look for a logo in Android Manifest to react the corresponding operation when any component of the program triggers events.

Part of the function design:

The main home interface design. Convenience and practical should be fully considered in the design of the main interface. Every Android interface is a visual interface, which has its unique layout configuration files. We can configure various layout and resources files according to the requirements, such as images, text and colour reference, which can form different visual interface and glaring effect. Interface design works with SWIPE ONLY.

There are many corresponding news available for the first time login entering the program; users need to just swipe appropriately as shown below:

Function design is to just swipe with many usable Gestures:

1. When need to see the next news just swipe up.
2. When need to see the previous news just swipe down.
3. When need to toggle the menu just swipe right.

CHAPTER 4
PROPOSED MODEL

Modules:
The proposed software will be having the following modules:

1. **RSS List**:
   RSS stands for Really Simple Syndication. This list will provide the user with a way to get all the updates and content available on the website in the android application so that the user does not need to visit the different websites daily.

2. **Picaso**:
   A picaso link to imbedded into the application through the gradle build.

3. **API Key**:
   API Wallpaper Is JSON API for downloading and changing the wallpaper.

4. **Options**:
   In this module, the user can decide whether to download the image or to apply the image as a wallpaper.

5. **Activities**:
   This app will contain different activities like Scrolling activity and the previous and next moving activity. The scrolling activity will contain all the images from the google pixel in a scrolling format.

CHAPTER 5
IMPLEMENTATION

1. **Step 1:** Install and Configure your Android Studio with the new version, then afterward create a new android project, application name “Wallpaper App” with minimum android and tablets API 4.0 jelly Bean configuration with configuring activity “main activity”.
2. **Step 2:** Now configure Gradle Scripts in build. gradle for customizing dependencies then sync it, now in android manifest give the user permission for INTERNET.
3. **Step 3:** Now the main design process starts from res session accompany with drawable, layout, mipmap, and values (colour). Change it according to your desire.
4. **Step 4:** Now create the package modules in java, afterward create the in that module package as mention in News API.
5. **Step 5**: Generate your Wallpaper API key from www.wallpaper.org, API Wallpaper. Is JSON API for infinite options of wallpaper images.
6. **Step 6:** Create another package for API in the same java folder then, add a new API client class and API interface in it.
7. **Step 7:** API Interface uses retrofit2 as HTTP Client. Retrofit is a REST Client for Java and Android. Its use is very easy.
8. **Step 8:** Now the overall design process started by adding the new resource file name item in the layout folder.
9. **Step 9:** Configure your desire layout then, extend all the classes code in the Adaptor class and Util class.
10. **Step 10:** Then add your Wallpaper API key in the MainActivity class. Run the program

CHAPTER 6
SOURCE CODE

MainActivity.java

```
public class MainActivity extends AppCompatActivity implements
OnRecyclerClickListner {
    RecyclerView recyclerView_home;
    CuratedAdapter adapter;
    ProgressDialog dialog;
    RequestManager manager;
    FloatingActionButton fab_next,fab_prev,fab_top;
    int page;

    @SuppressLint("MissingInflatedId")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        fab_next=findViewById(R.id.fab_next);
        fab_prev=findViewById(R.id.fab_prev);
        fab_top = findViewById(R.id.fab_top);
         dialog= new ProgressDialog(this );
         dialog.setTitle("Loading..");
         manager = new RequestManager(this);
         manager.GetCuratedWallpaper(listeners,"1");

         fab_next.setOnClickListener(new View.OnClickListener() {
             @Override
             public void onClick(View view) {
                 String next_page=String.valueOf(page+1);
                 manager.GetCuratedWallpaper(listeners,next_page);
                 dialog.show();

             }
         });
```

WallpaperActivity.java

```
public class wallpaperActivity extends AppCompatActivity {

    ImageView Imageview_Wallpaper;
    FloatingActionButton fab_download,fab_Wallpaper;
    Photo photo;


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_wallpaper);
        Imageview_Wallpaper=findViewById(R.id.Imageview_Wallpaper);
        fab_download=findViewById(R.id.fab_download);
        fab_Wallpaper=findViewById(R.id.fab_Wallpaper);
        photo= (Photo) getIntent().getSerializableExtra("Photo");

            public void onClick(View view) {
                DownloadManager downloadManager= null;
                downloadManager=(DownloadManager)
getSystemService(Context.DOWNLOAD_SERVICE);

                Uri uri= Uri.parse(photo.getSrc().getLarge());
                DownloadManager.Request request= new
DownloadManager.Request(uri);


request.setAllowedNetworkTypes(DownloadManager.Request.NETWORK_W
IFI |
DownloadManager.Request.NETWORK_MOBILE).setAllowedOverRoaming(
false)
                    .setTitle("Wallpaper"+photo.getPhotographer()).setMimeType("i
mage/jpeg").setNotificationVisibility(DownloadManager.Request.VISIBILITY
_VISIBLE_NOTIFY_COMPLETED)
                    .setDestinationInExternalPublicDir(Environment.DIRECTORY_
PICTURES,"Wallpaper"+photo.getPhotographer()+".jpg");
```

```
                downloadManager.enqueue(request);

                Toast.makeText(wallpaperActivity.this, "Download Completed!",
Toast.LENGTH_SHORT).show();


                fab_Wallpaper.setOnClickListener(new View.OnClickListener() {
                    @Override
                    public void onClick(View view) {
                        WallpaperManager wallpaperManager=
WallpaperManager.getInstance(wallpaperActivity.this);
                        Bitmap bitmap =((BitmapDrawable)
Imageview_Wallpaper.getDrawable()).getBitmap();

                        try {
                            wallpaperManager.setBitmap(bitmap);
                            Toast.makeText(wallpaperActivity.this, "Wallpaper Applied",
Toast.LENGTH_SHORT).show();
                        }
                        catch (Exception e){
                            e.printStackTrace();
                            Toast.makeText(wallpaperActivity.this, "Couldn't Add
Wallpaper", Toast.LENGTH_SHORT).show();
                        }
                    }
                });
            }
        });

    }
}
```

AndroidManifest.xml

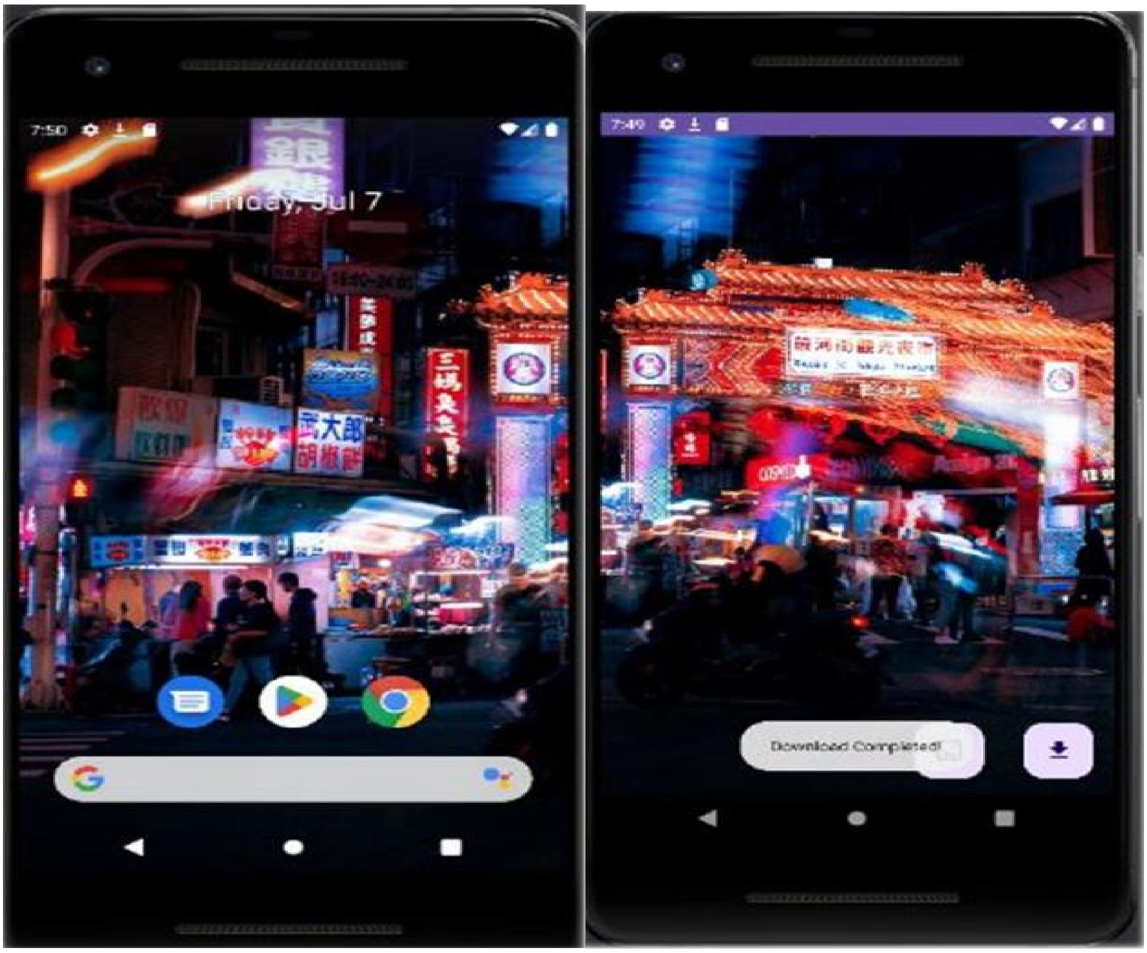
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission
android:name="android.permission.SET_WALLPAPER"/>

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Wallpaper"
        tools:targetApi="31">
        <activity
            android:name=".wallpaperActivity"
            android:exported="false" />
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

CHAPTER 7
RESULTS AND SNAPSHOTS



Download Completed!

CHAPTER 8
CONCLUSION

In conclusion, developing a wallpaper app linked with Google Pixel using Android Studio offers numerous benefits and opportunities. By creating a wallpaper app specifically designed for Google Pixel devices, you can provide a tailored and optimized user experience for Pixel users who are passionate about personalizing their device's appearance. Android Studio, as the official integrated development environment (IDE) for Android app development, offers a wide range of tools and resources to streamline the development process. Its robust features, such as the Android Emulator, layout editor, and debugging tools, enable developers to create high-quality and visually appealing wallpaper apps.

We can’t think a single moment without technology. From morning to night, we need help of the technology. This is the revolutionary time of computer technology. Most of the works depends on web application. For this reason, anytime, anywhere, anyone can access a website by internet at low cost and we can find our expectable and most update information from website. At present information is one the most valuable resource of the current world. We have developed our project so that we can aware the people. This is the combination management of Java language in the open source mobile platform based on Linux system configuration file. The system realized the radio music player programming. By adopting ANDROID STUDIO 3.1.2 + Java language as technical support of this system, with the Android plug-in tools, and combination of Latest Android SDK version lead to the comprehensive and smoothly design and development of the mobile terminal.

CHAPTER 9
REFERENCES

1. Ribeiro and A. R. D. Silva, “Survey on Cross-Platforms and Languages for Mobile Apps,” Eighth International Conference on the Quality of Information and Communications Technology, 2012.
2. Pohares, V. C. Kulloli, T. Bhattacharyya, and S. Bhure, “Cross-Platform Mobile Application Development,” International Journal of Computer Trends and Technology,vol. 4, no. 5, pp. 1095– 1100, 2013.
3. S. S. Jagtap and D. B. Hanchate, “Development of Android Based Mobile App forPrestaShop eCommerce Shopping Cart (ALC),” International Research Journal of Engineering and Technology (IRJET), vol. 4, no. 7, pp. 2248–2254, Jul. 2017.
4. M. Q. Huynh and P. Ghimire, “Browser App Approach: Can It Be an Answer to theChallenges in Cross-Platform App Development?,” Journal of Information Technology Education: Innovations in Practice, vol. 16, pp. 047–068,2017.
5. N. Litayem, B. Dhupia, and S. Rubab, “Review of Cross-Platforms for Mobile Learning Application Development,” International Journal of Advanced ComputerScience and Applications, vol. 6, no. 1, pp. 31–39, 2015. 6. Kathuria and A. Gupta, “Challenges in Android Application Development: A CaseStudy”International Journal of Computer Science and Mobile Computing, vol. 4, no. 5,pp. 294–299, May 2015.