

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi – 590018



Mobile Application Development
A Mini Project Work (18CSMP68) Report
on

“FOOD RECIPES APPLICATION”

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING

Submitted by

SINCHANA S

1KI20CS100

SPOORTHY N

1KI20CS104

Under the guidance of

Dr. Manoj Kumar D. P. B.E., M.E., Ph.D.

Associate Professor

Department of Computer Science & Engineering
Kalpataru Institute of Technology, Tiptur – 572201



Department of Computer Science and Engineering
Kalpataru Institute of Technology, Tiptur – 572201

2022-23

Kalpataru Institute of Technology, Tiptur – 572201
Department of Computer Science and Engineering



CERTIFICATE

Certified that the Mini Project Work (18CSMP68) entitled **“FOOD RECIPES APPLICATION”** is a bonafide work carried out by

SINCHANA S

1KI20CS100

SPOORTHY N

1KI20CS104

in partial fulfillment for the award of **Bachelor of Engineering** in **Computer Science and Engineering** of the **Visvesvaraya Technological University, Belagavi** during the year 2022-23. 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 work prescribed for the said degree.

Guide

HOD

Dr. Manoj Kumar D. P.

Associate Professor
Dept. of CSE
KIT, Tiptur

Name of Examiners

1. _____

2. _____

Prof. Shashidhara M. S

Associate Professor &
HOD Dept. of CSE
KIT, Tiptur

Signature with Date

Kalpataru Institute of Technology, Tiptur - 572201
Department of Computer Science and Engineering



DECLARATION

We, the students of Sixth Semester of Computer Science and Engineering, Kalpataru Institute of Technology, Tiptur - 572201, declare that the work entitled **“FOOD RECIPES APPLICATION”** has been successfully completed under the guidance of **Dr. Manoj Kumar D.P.**, Associate Professor, Department of Computer Science and Engineering. This project work is submitted to Visvesvaraya Technological University in partial fulfillment of the requirements for the award of Degree of Bachelor of Engineering in Computer Science and Engineering during the academic year 2022-2023. Further, the matter embodied in the mini project report has not been submitted previously by anybody for the award of any degree or diploma to any university.

Place: Tiptur

Date: 07/07/2023

Project Associates:

Sl. No.	Student Name	USN	
1	SINCHANA S	1KI20CS100	-
2	SPOORTHY N	1KI20CS104	-

ACKNOWLEDGEMENT

The satisfaction and great happiness that accompany the successful completion of any task would be incomplete without mentioning about the people who made it possible. Here we make an honest attempt to express our deepest gratitude to all those who have been helpful and responsible for the successful completion of our project.

We would like to thank **Dr. G.D. Gurumurthy**, Principal, Kalpataru Institute of Technology, Tiptur for his continuous support and encouragement throughout the course of this project.

We would like to thank **Prof. Shashidhara M.S.**, Head of the Department, Department of Computer Science and Engineering, Kalpataru Institute of Technology, Tiptur for his continuous support and encouragement throughout the course of this project.

We are immensely indebted to our guide **Dr. Manoj Kumar D.P.**, Associate Professor, Department of Computer Science and Engineering, Kalpataru Institute of Technology, Tiptur for his support, technical assistance and constructive suggestions and guidance for successful completion of our project. We are very much thankful to him for the encouragement that has infused at most real into us to complete the project work.

We would like to thank all the faculty members of Department of Computer Science and Engineering, KIT, Tiptur, our family members and to our beloved friends who are directly or indirectly responsible for our success.

Thanking you,

Sl. No.	Student Name	USN
1	SINCHANA S	1KI20CS100
2	SPOORTHY N	1KI20CS104

ABSTRACT

An Android application that will give users recipes based on the ingredients they have on hand is the suggested project. The user must choose one or more ingredients from a list before the application displays a selection of recipes that call for that particular combination of ingredients. A step-by-step guide to North Indian and South Indian recipes may be found in the Food Recipes mobile application. Detailed instructions with cooking temperature, flame intensity, and cooking time are provided in a food recipe application. Each food dish includes information on hints, variants, serving suggestions, and flavor to make cooking enjoyable for food fans. The goal of this project is to create a user-friendly Mobile Application name **“Food Recipes”** that will allow users to explore delectable meals while on the go. The app will feature a diverse collection of carefully curated recipes, spanning various cuisines, dietary preferences and skill levels. Users will have the ability to search for recipes based on ingredients, cooking time, and difficulty level, ensuring they can find the perfect dish for any occasion. To enhance the cooking experience, **“Food Recipes”** will provide step-by-step instructions, visually appealing images that guide users through each recipe. In further development of this application users will have the ability to adjust serving sizes, view nutritional information, and utilize handy built-in tools such as timers and measurement converters, making the cooking process more convenient and efficient.

TABLE OF CONTENTS

LIST OF FIGURES----- (i)

LIST OF TABLES----- (ii)

1. INTRODUCTION TO MOBILE

APPLICATION DEVELOPMENT----- 01 – 04

1.1 Android History

1.1.1 Android Versions

1.2 Android Architecture

1.3 Programming Languages

2. SYSTEM REQUIREMENTS----- 05 – 11

2.1 System Requirements

2.1.1 Hardware Requirements

2.1.2 Software Requirements

2.2 Installation Procedure of Android Studio

2.3 System Architecture

3. IMPLEMENTATION DETAILS----- 12 – 22

3.1 Creation of New Project

3.2 Code Lines for the Application

3.2.1 mainactivity.java

4. RESULTS----- 23–24

4.1 Sample Screenshots

CONCLUSION----- 25

BIBLIOGRAPHY----- 26

LIST OF FIGURES

<i>FIGURE NO.</i>	<i>DESCRIPTION</i>	<i>PAGE NO.</i>
1.1	Android Architecture	3
2.1	Configuring Android SDK Project	7
2.2	Creation of Virtual Device	8
2.3	Emulator	10
2.4	System Architecture	11
3.1	Creation of new project	12
3.2	Configuring the Project	14
3.3	Android Project Structure	14
3.4	Basic View of IDE	15
3.5	Creation of Android Activity	16
3.6	Creation of Layout	17
3.7	Creation of assets folder	18
3.8	Creating File in assets folder	20
4.1	Sign Up Page	23
4.2	Login Page	23
4.3	Recipes List	24
4.4	Description of Recipe	24

LIST OF TABLES

<i>TABLE NO.</i>	<i>DESCRIPTION</i>	<i>PAGE NO.</i>
1.1	Android Versions	2

Chapter 1

INTRODUCTION TO MOBILE APPLICATION DEVELOPMENT

Android is an operating system. That is, it's software that connects hardware to software and provides general services. But more than that, it's a *mobile specific* operating system: an OS designed to work on *mobile* (read: handheld, wearable, carry-able) devices.

Note that the term “Android” also is used to refer to the “platform” (e.g., devices that use the OS) as well as the ecosystem that surrounds it. This includes the device manufacturers who use the platform, and the applications that can be built and run on this platform. So “Android Development” technically means developing applications that run on the specific OS, it also gets generalized to refer to developing any kind of software that interacts with the platform.

1.1 Android History

- **2003:** The platform was originally founded by a start-up “Android Inc.” which aimed to build a mobile OS operating system (similar to what Nokia’s Symbian was doing at the time)
- **2005:** Android was acquired by Google, who was looking to get into mobile
- **2007:** Google announces the Open Handset Alliance, a group of tech companies working together to develop “open standards” for mobile platforms. Members included phone manufacturers like HTC, Samsung, and Sony; mobile carriers like T-Mobile, Sprint, and NTT DoCoMo; hardware manufacturers like Broadcom and Nvidia; and others. The Open Handset Alliance now (2017) includes 86 companies.
- **2008:** First Android device is released: the HTC Dream (a.k.a. T-Mobile G1)
 - Specs: 528Mhz ARM chip; 256MB memory; 320x480 resolution capacitive touch; slide-out keyboard! Author’s opinion: a fun little device.

- **2010:** First Nexus device is released: the Nexus One. These are Google-developed “flagship” devices, intended to show off the capabilities of the platform.
 - Specs: 1Ghz Scorpion; 512MB memory; .37" at 480x800 AMOLED capacitive touch.
- For comparison, the iPhone 7 Plus (2016) has: 2.34Ghz dual core A10 64bit Fusion; 3GB RAM; 5.5" at 1920x1080 display.
 - As of 2016, this program has been superseded by the Pixel range of devices.
- **2014:** Android Wear, a version of Android for wearable devices (watches) is announced.
- **2016:** Daydream, a virtual reality (VR) platform for Android is announced

1.1.1 Android Versions

Date	Version	Nickname	API Level
Sep 2008	1.0	Android	1
Apr 2009	1.5	Cupcake	3
Sep 2009	1.6	Donut	4
Oct 2009	2.0	Éclair	5
May 2010	2.2	Froyo	8
Dec 2010	2.3	Gingerbread	9
Feb 2011	3.0	Honeycomb	11
Oct 2011	4.0	Ice Cream Sandwich	14
July 2012	4.1	Jelly Bean	16
Oct 2013	4.4	KitKat	19
Nov 2014	5.0	Lollipop	21
Oct 2015	6.0	Marshmallow	23
Aug 2016	7.0	Nougat	24
Mar 2017	O preview	<i>Android O Developer Preview</i>	

Table 1.1 – Android Versions

1.2 Android Architecture and Code

Developing Android applications involves interfacing with the Android platform and framework. Thus you need a high level understanding of the architecture of the Android platform.

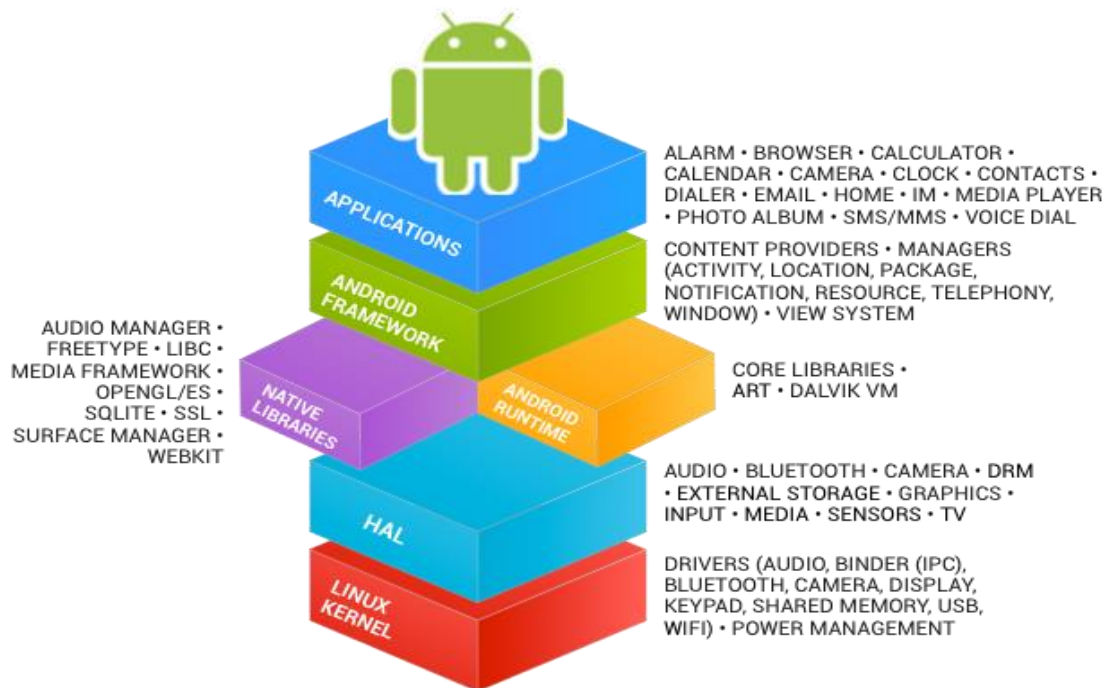


Fig 1.1 - Android Architecture

The Android platform is built as a layered architecture:

- At its base, Android runs on a Linux kernel for interacting with the device's processor, memory, etc. Thus an Android device can be seen as a Linux computer.
- On top of that kernel is the Hardware Abstraction Layer: an interface to drivers that can programmatically access hardware elements, such as the camera, disk storage, Wifi antenna, etc.
 - These drivers are generally written in C; we won't interact with them directly in this course.
- On top of the HAL is the Runtime and Android Framework, which provides a set of abstraction in the Java language which we all know and love. For this course, Android Development will involve writing Java applications that interact with the Android Framework layer, which handles the task of interacting with the device hardware for us.

1.3 Programming Languages

There are two programming languages we will be working with in this course:

1. **Java:** Android code (program control and logic, as well as data storage and manipulation) is written in Java.

Writing Android code will feel a lot writing any other Java program: you create classes, define methods, instantiate objects, and call methods on those objects. But because you're working within a framework, there is a set of code that already exists to call specific methods. As a developer, your task will be to fill in what these methods do in order to run your specific application.

- In web terms, this is closer to working with Angular (a framework) than jQuery (a library).
- Importantly: this course expects you to have “journeyman”-level skills in Java (apprenticeship done, not yet master). We'll be using a number of intermediate concepts (like generics and inheritance) without much fanfare or explanation.

2. **XML** : Android user interfaces and resources are specified in XML (EXtensible Markup Language). To compare to web programming: the XML contains what would normally go in the HTML/CSS, while the Java code will contain what would normally go in the JavaScript. XML is just like HTML, but you get to make up your own tags. Except we'll be using the ones that Android made up; so it's like defining web pages, except with a new set of elements. This course expects you to have some familiarity with HTML or XML, but if not you should be able to infer what you need from the examples.

Chapter 2

SYSTEM REQUIREMENTS

Software Requirement Specification (SRS) is a fundamental document, which forms the foundation of the software development process. SRS not only lists the requirements of a system but also has a description of its major features. These recommendations extend the IEEE standards. The recommendations would form the basis for providing clear visibility of the product to be developed serving as baseline for execution of a contract between client and the developer. SRS constitutes the agreement between clients and developers regarding the contents of the software product that is going to be developed. SRS should accurately and completely represent the system requirements as it makes a huge contribution to the overall project plan. The software being developed may be a part of the overall larger system or may be a complete standalone system in its own right.

2.1 System Requirements

2.1.1 Hardware Requirements

- **Processor** : Pentium(Any) or higher | AMD Athlon
- **Hard Disk** : 2 GB or higher
- **Monitor** : 14.1 inch or higher
- **RAM** : 1 GB or higher

2.1.2 Software Requirements

- **Operating System** : Windows 7 or higher
- **Coding Language** : XML and JAVA

2.2 Installation Procedure of Android Studio

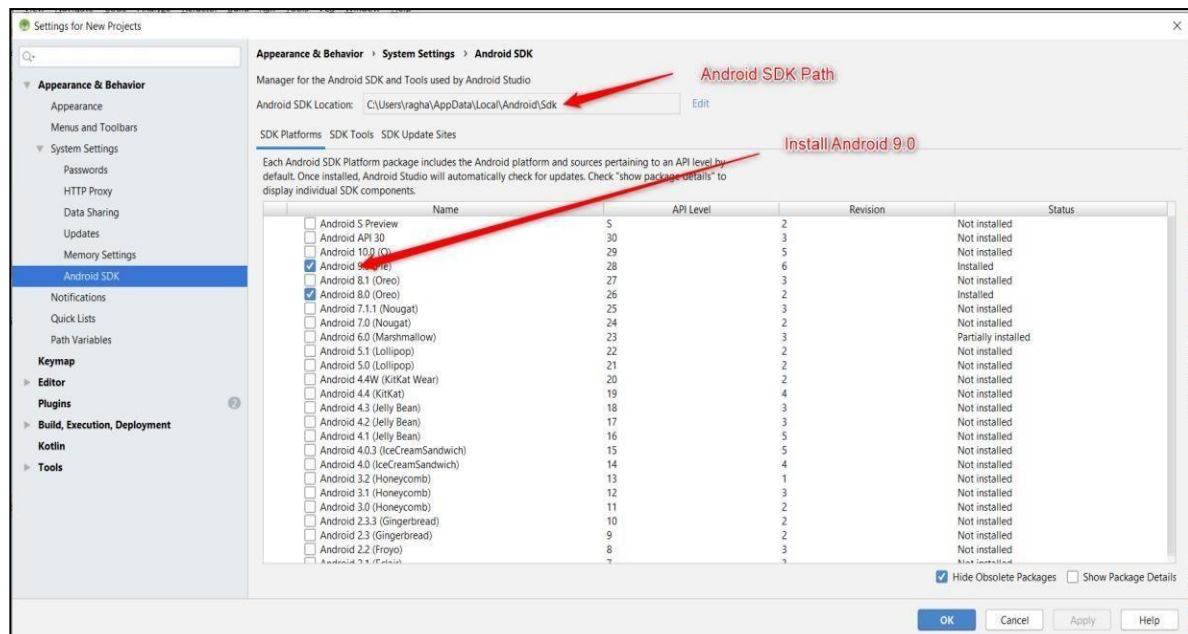
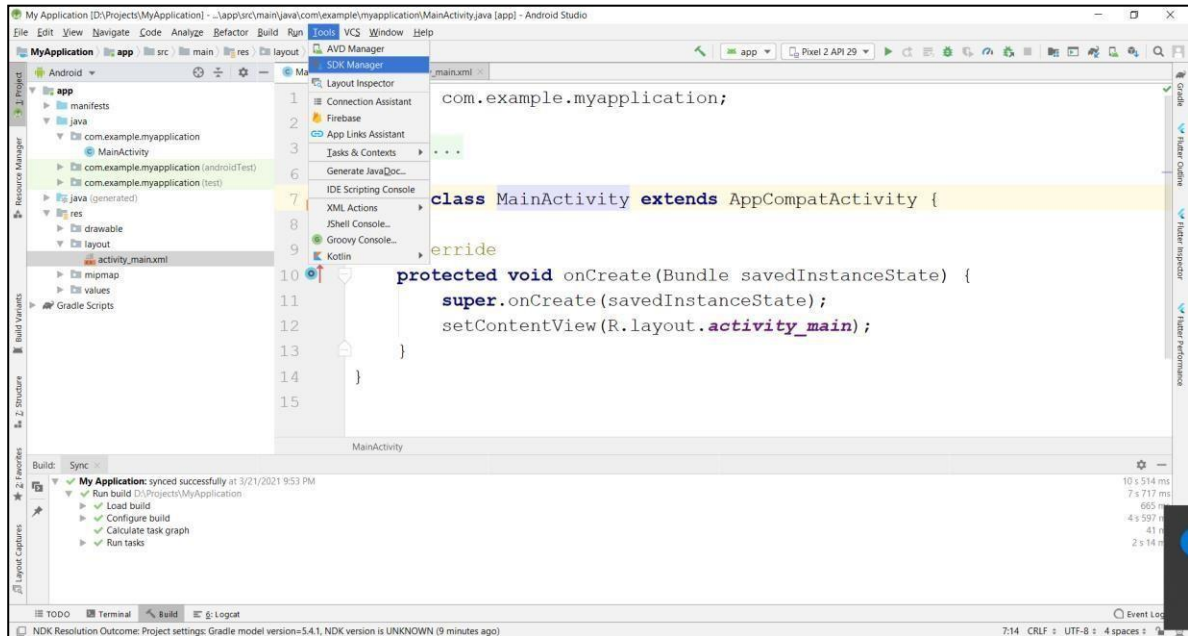
Install Android Studio and Packages:

Download Android Studio from the below link:

<https://developer.android.com/studio>

Configure Android SDK Packages:

Go to Tools -> SDK Manager



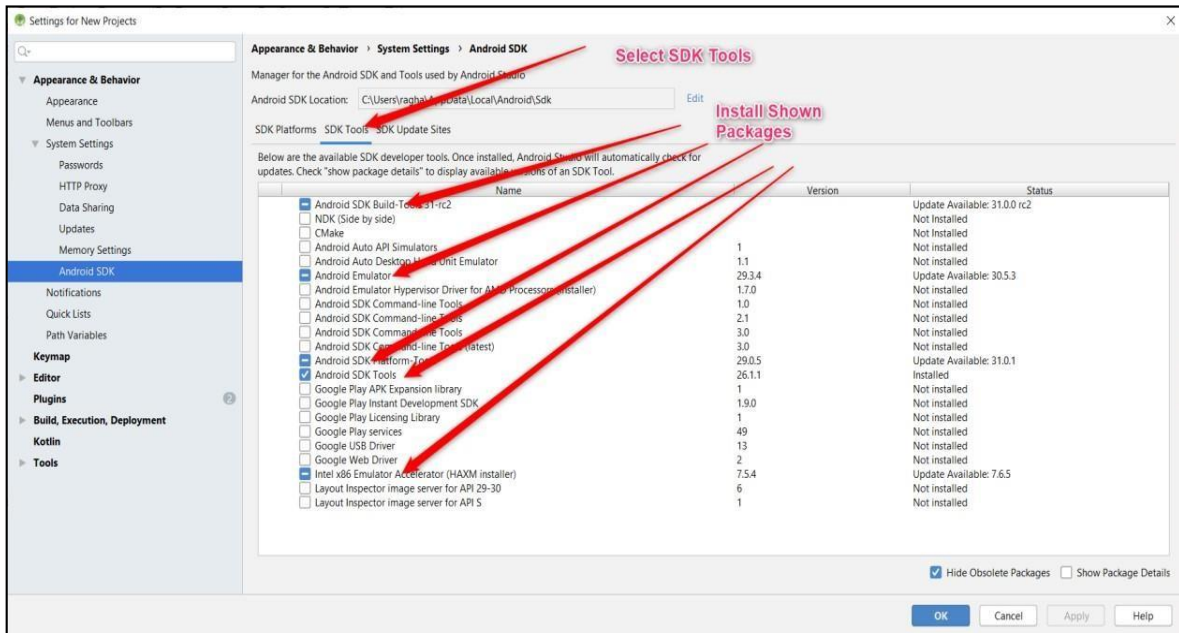
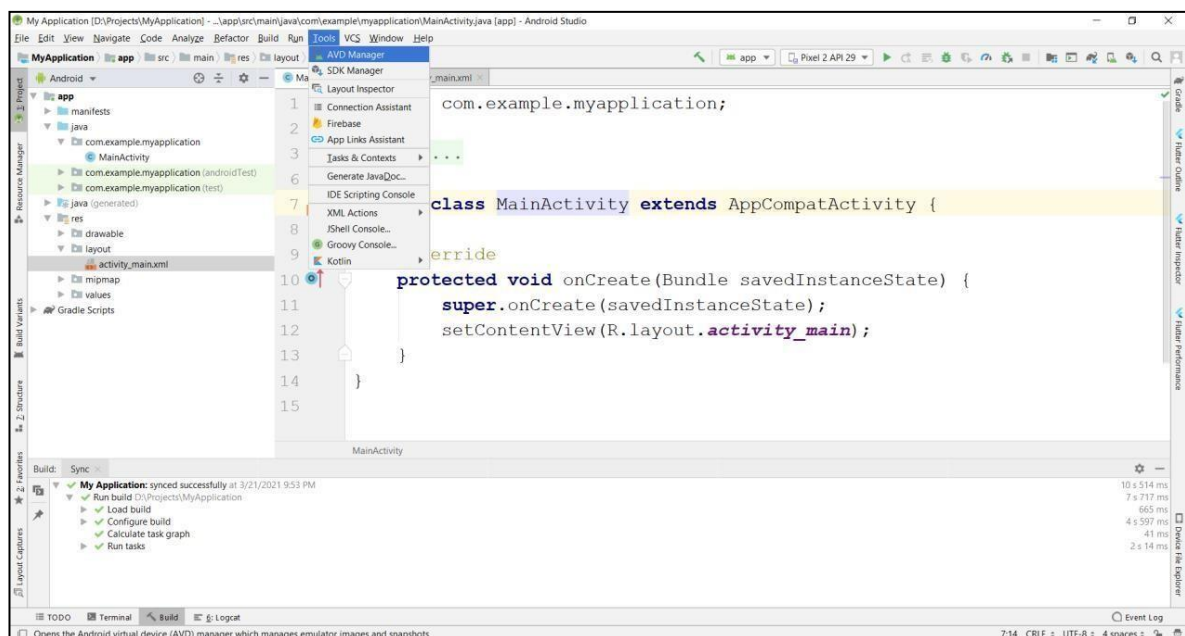
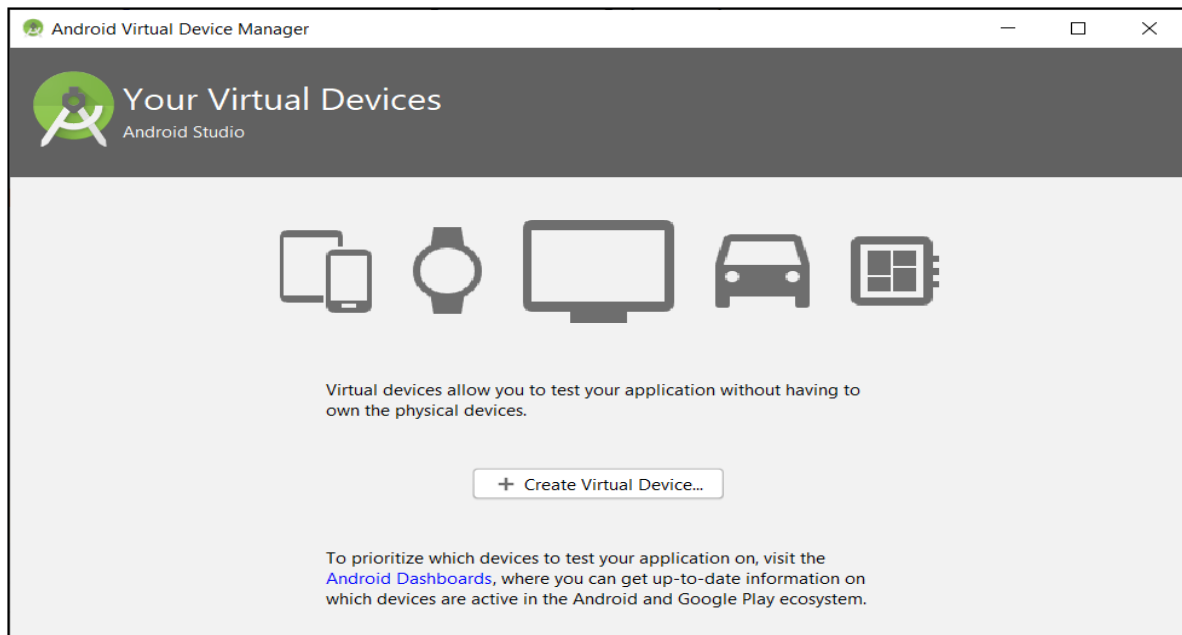


Fig : 2.1 – Configuring Android SDK Project

Creating Emulator:

Go to Tools -> Select AVD Manager





Select Create virtual Device -> Select Phone -> Pixel2 -> Press Next

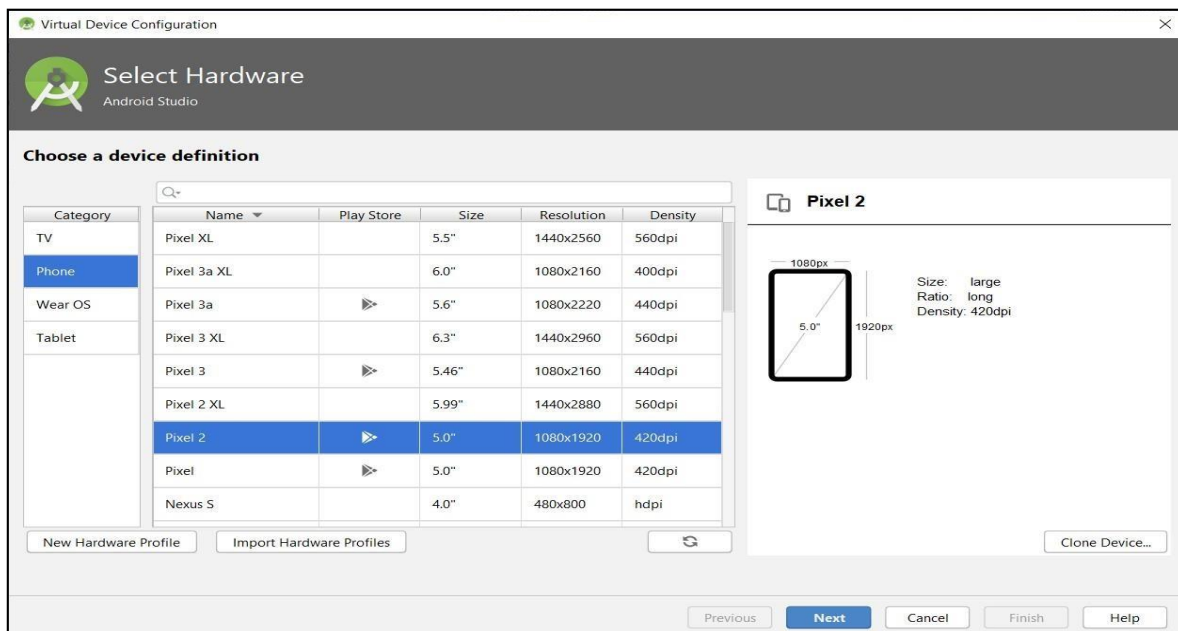
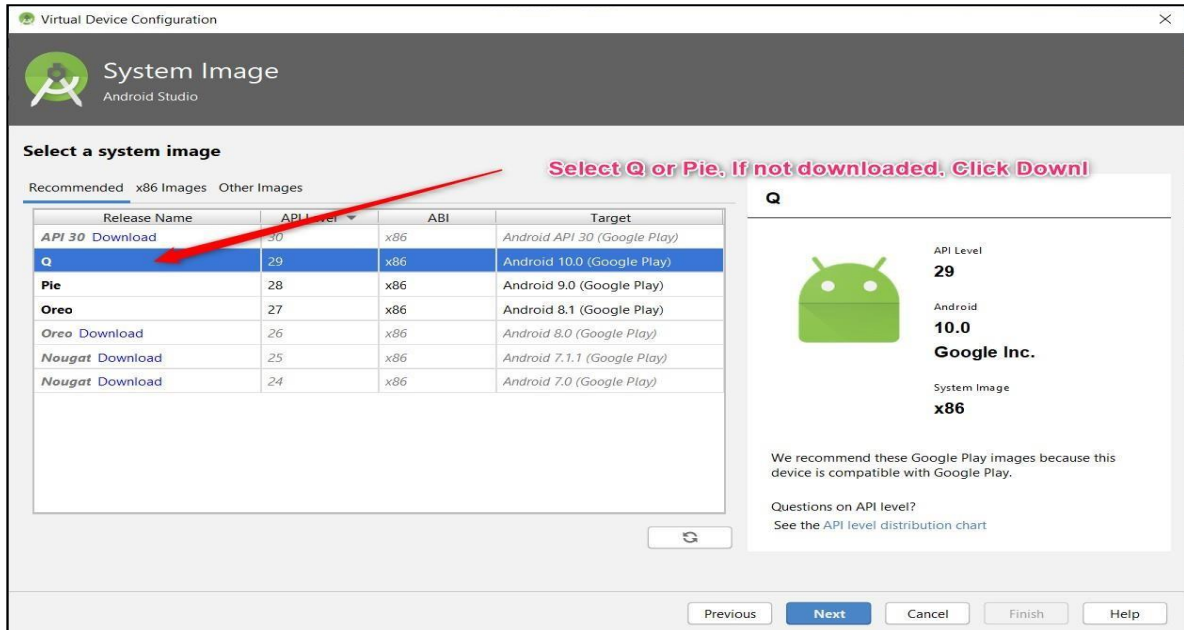
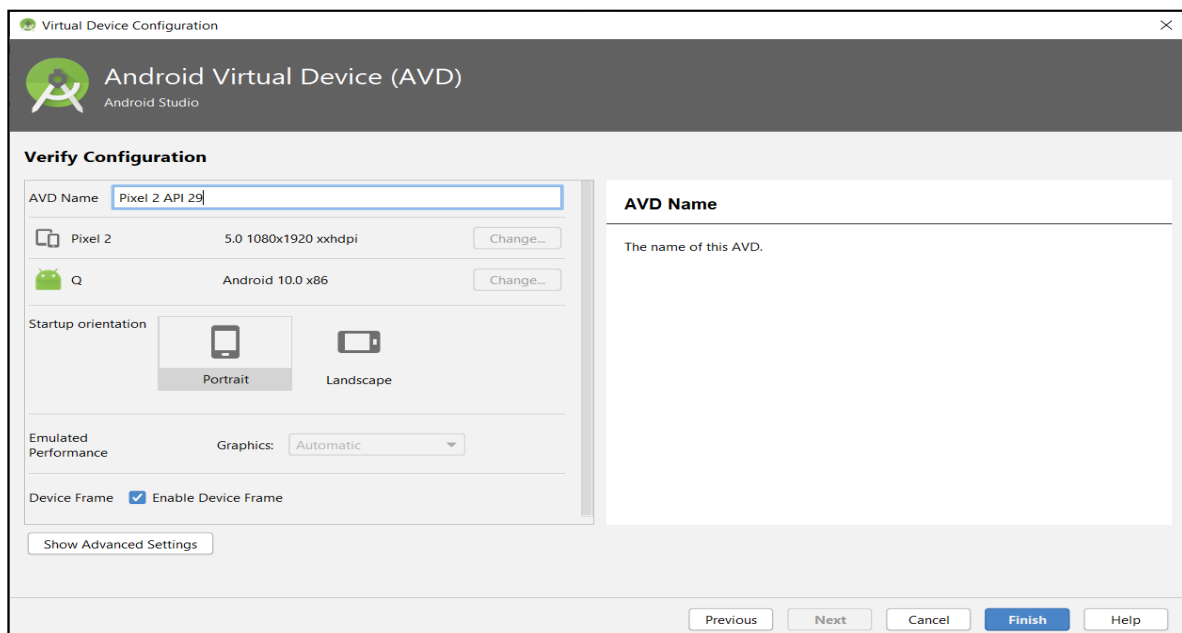


Fig:2.2 – Creation of Virtual Device

Select **Android Q**, if not already downloaded press download, After download completes Select **Q** and Press **Next** Button:



Enter AVD Name and Press Finish:



Press Play Button to Start Emulator

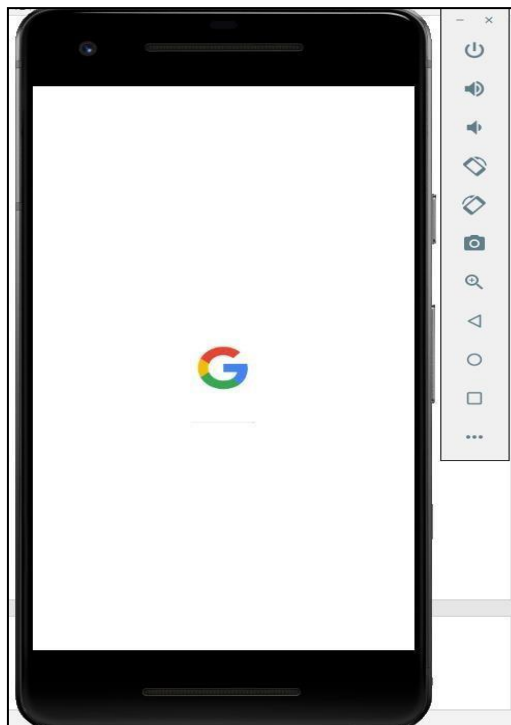
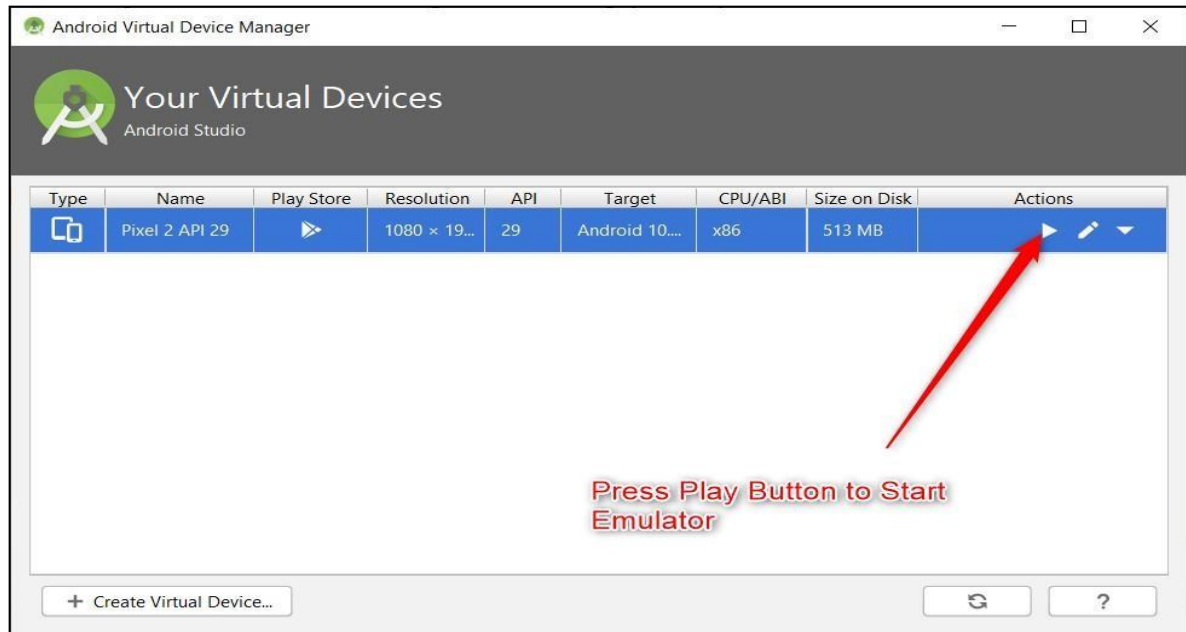


Fig : 2.3 - Emulator

2.3 System Architecture:

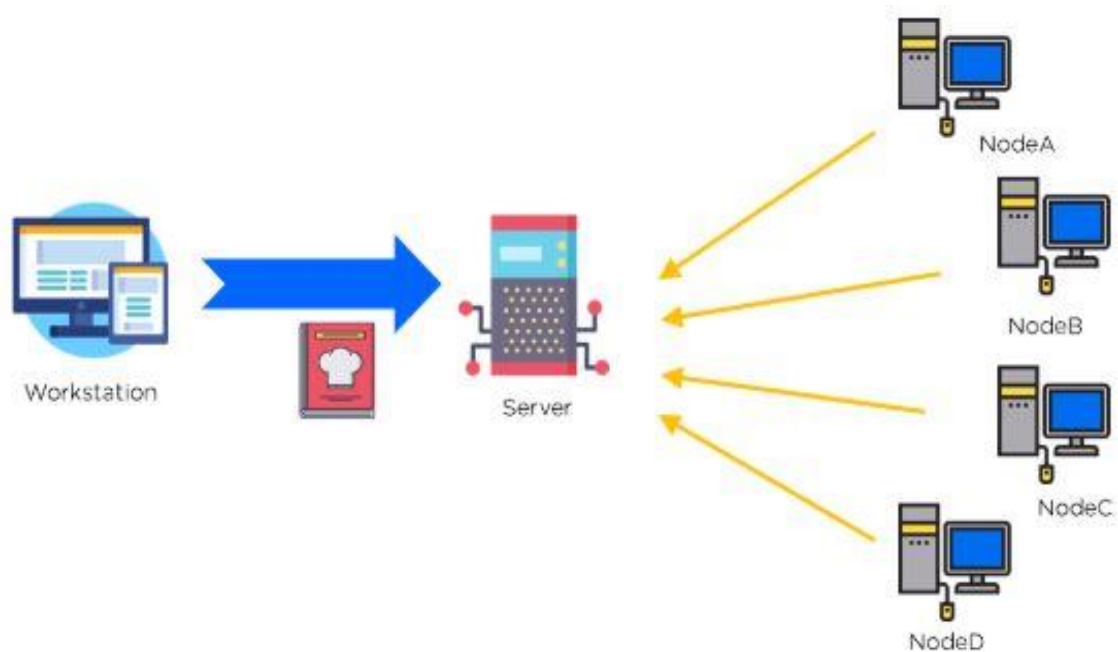
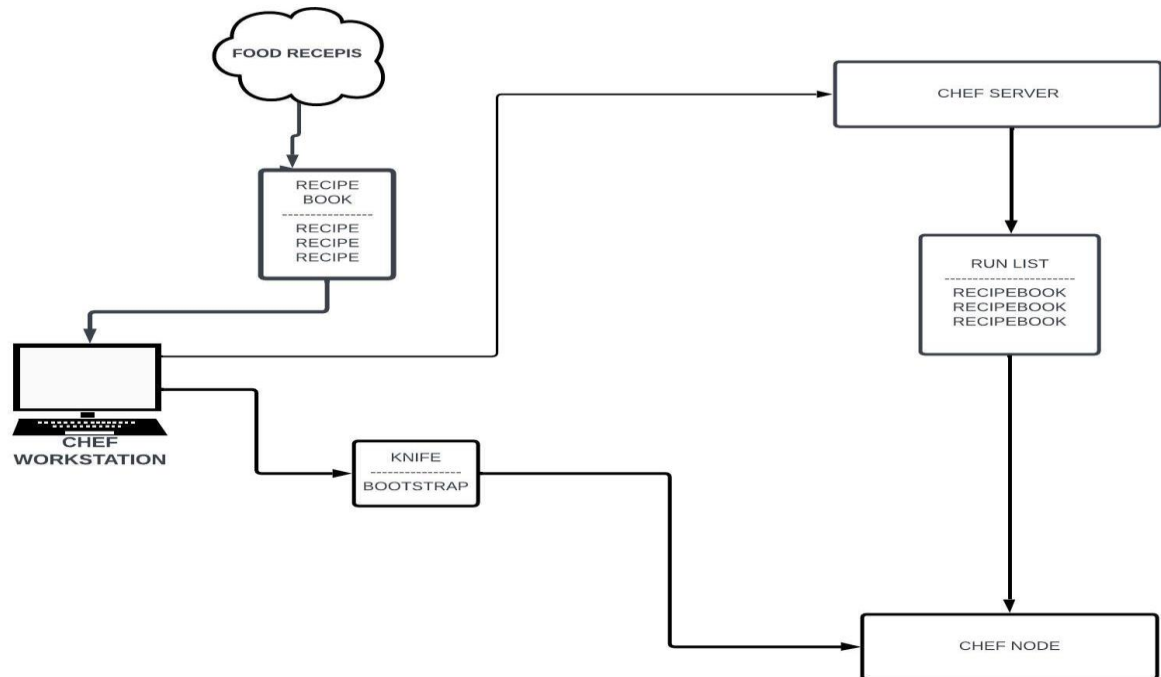


Fig : 2.4 – System Architecture

Chapter 3

IMPLEMENTATION DETAILS

The implementation phase involves more than just writing code. Code also needs to be tested and debugged as well as compiled and built into a complete executable product. We usually need to utilize configuration management in order to keep track of different version of code. This is the stage of the project where the theoretical design is turned into a working system. If the implementation is not carefully planned and controlled, it can cause chaos and confusions. It is always a good idea to keep in mind that some characteristics that should be found in a good implementation like Readability- our code is written in MVC Architecture ,JAVA to achieve the objective of the project that is to introduce a novel scheme of mechanism design for balancing the resource consumptions .

3.1 Creation of New Project.

Creating a New Project in Android:

While creating a New Project for First Time, make sure Android Studio is connected to internet. It downloads the required packages from internet.

Go to File -> New -> New Project

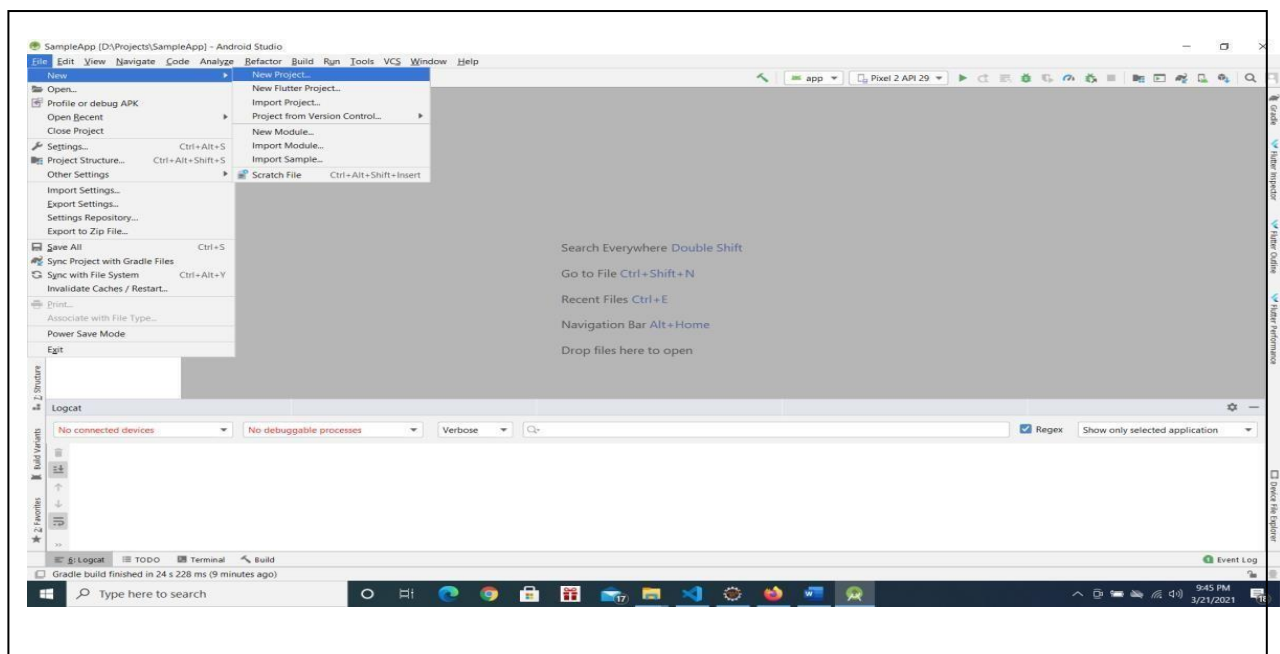
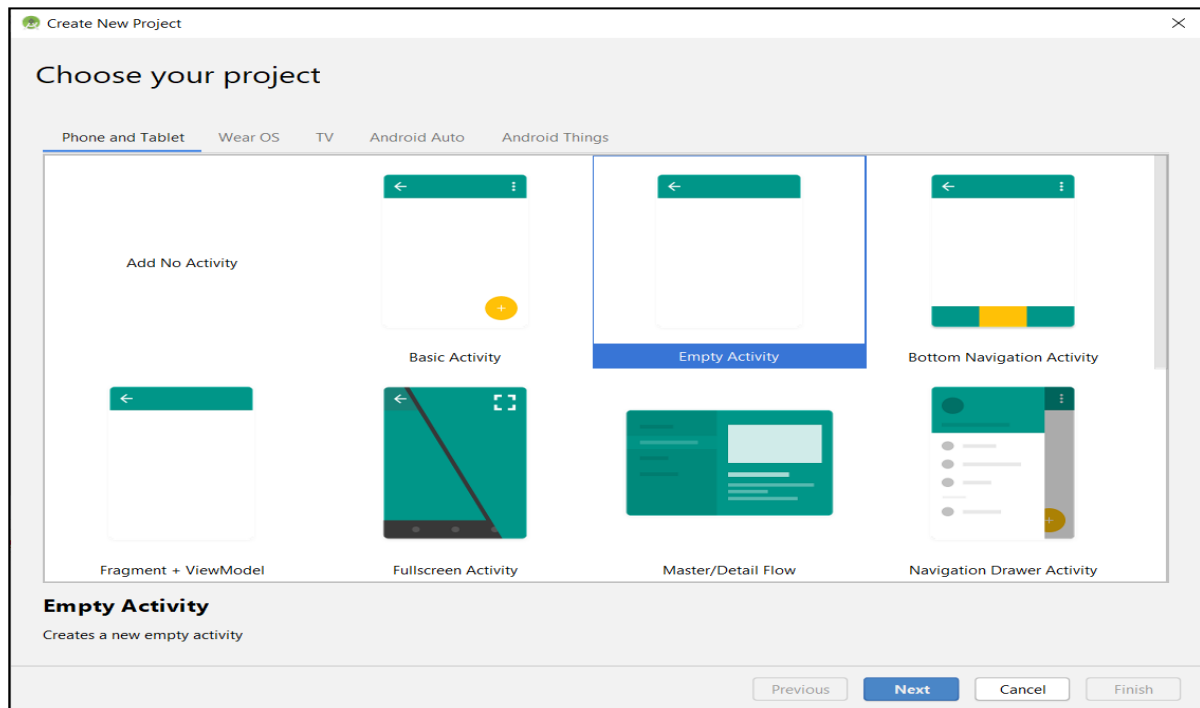


Fig : 3.1 - Creation of new project

Choose Phone and Tablet -> Empty Activity -> Press Next



- In Configure your Project Screen, Enter below details and Press Finish Button.
- Enter Name of the Application -> This will be application name this will be visible with Home Screen Icon.
- Package Name -> Enter package name atleast two identifier (Eg: com.example). Best Practice is 3 or more identifier (Eg:com.example.firstapp).
- Save Location -> Location
Where to save the Project
Language -> Choose Java
- Minimum API Level -> Android 5.0
- Select Checkbox Use androidx.artifacts folder as below screenshot.

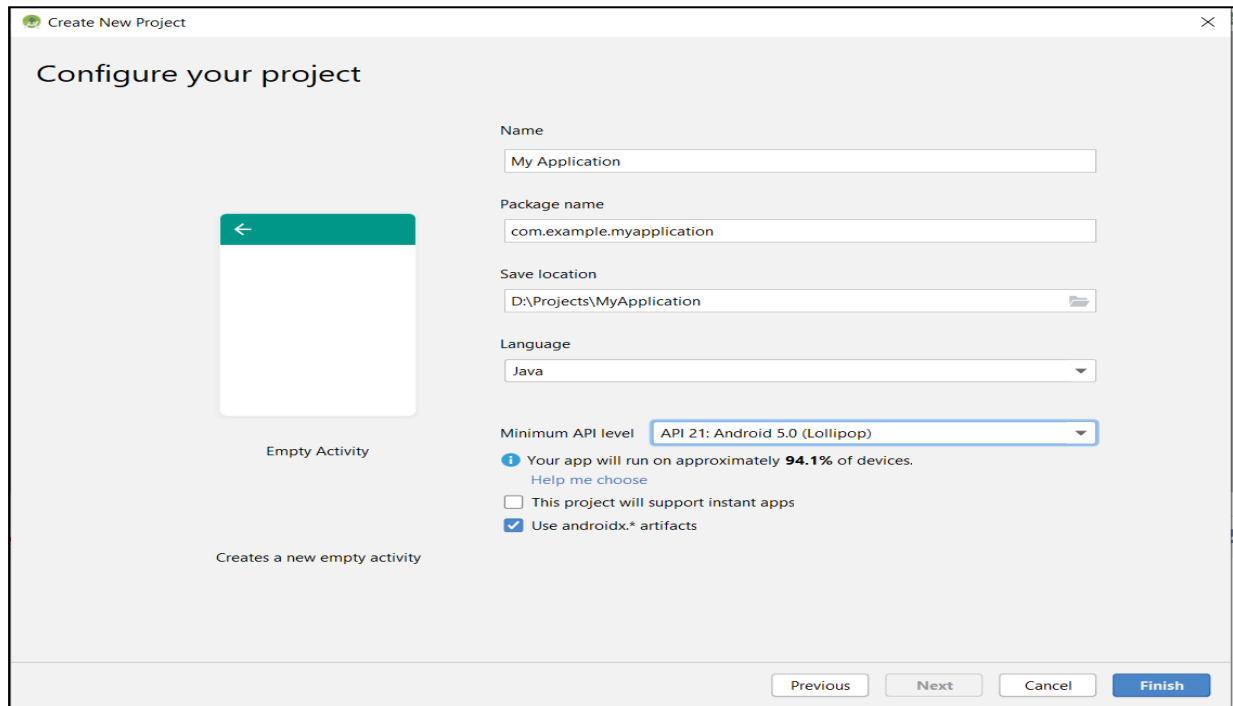


Fig : 3.2 – Configuring the Project

Android Project Structure:

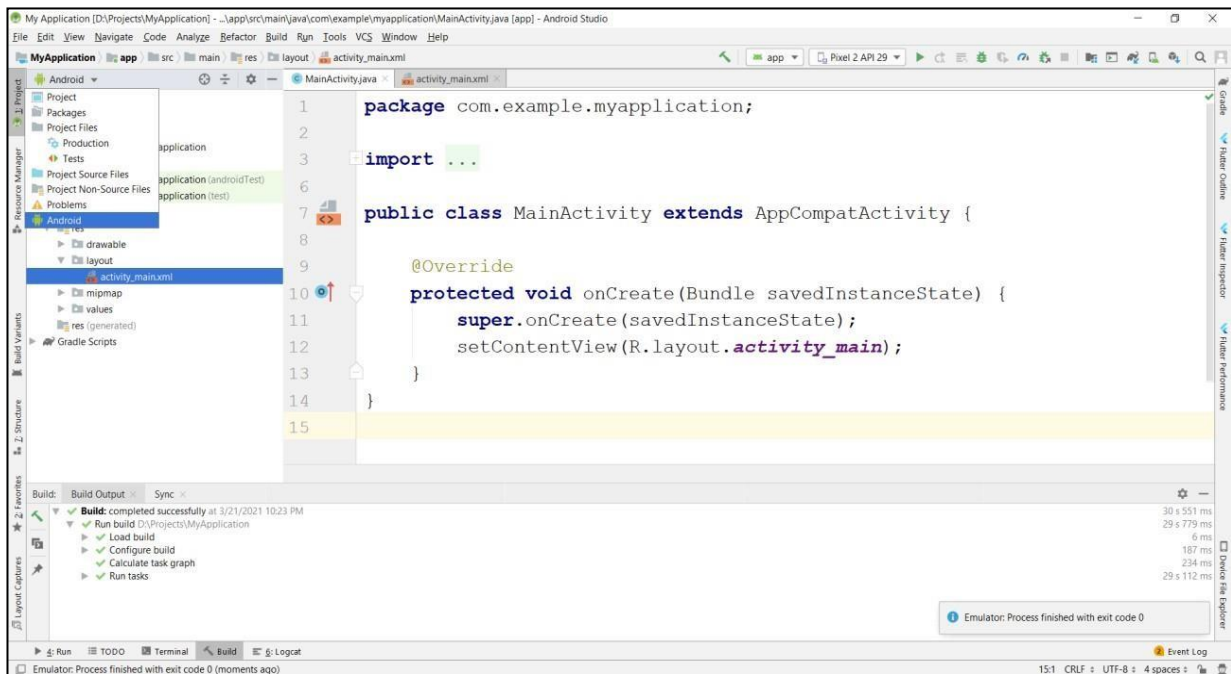
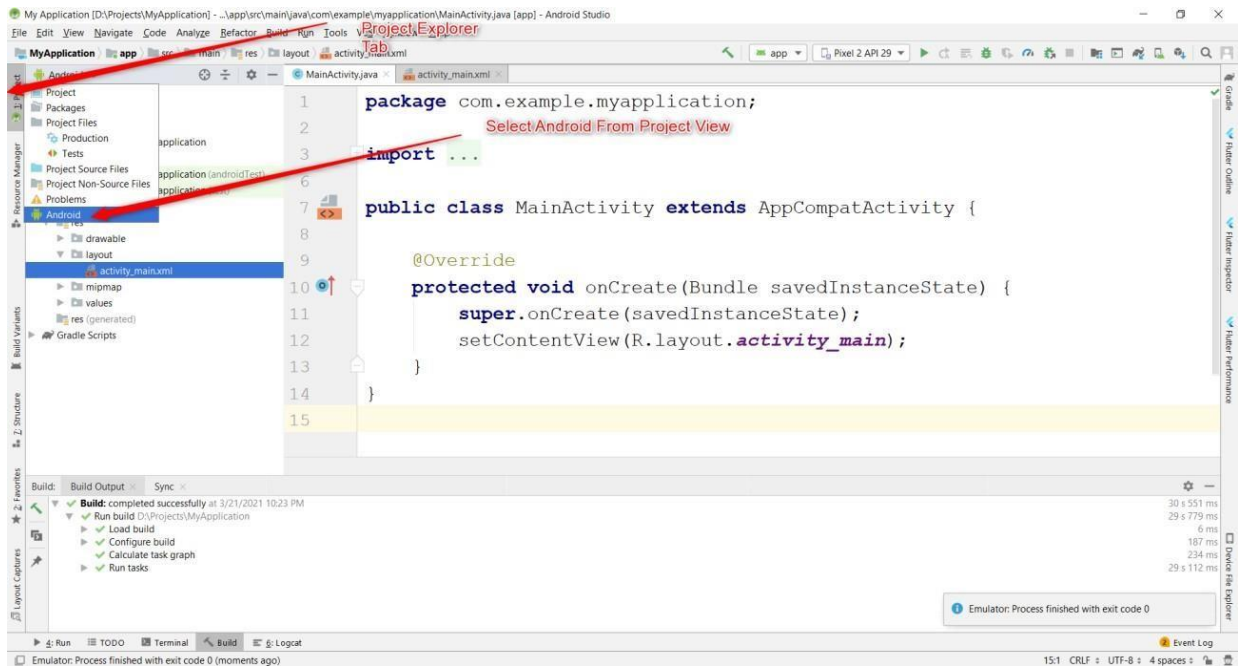


Fig : 3.3 – Android Project Structure

Select Project Explorer and Select Android from Project View



Basic View:

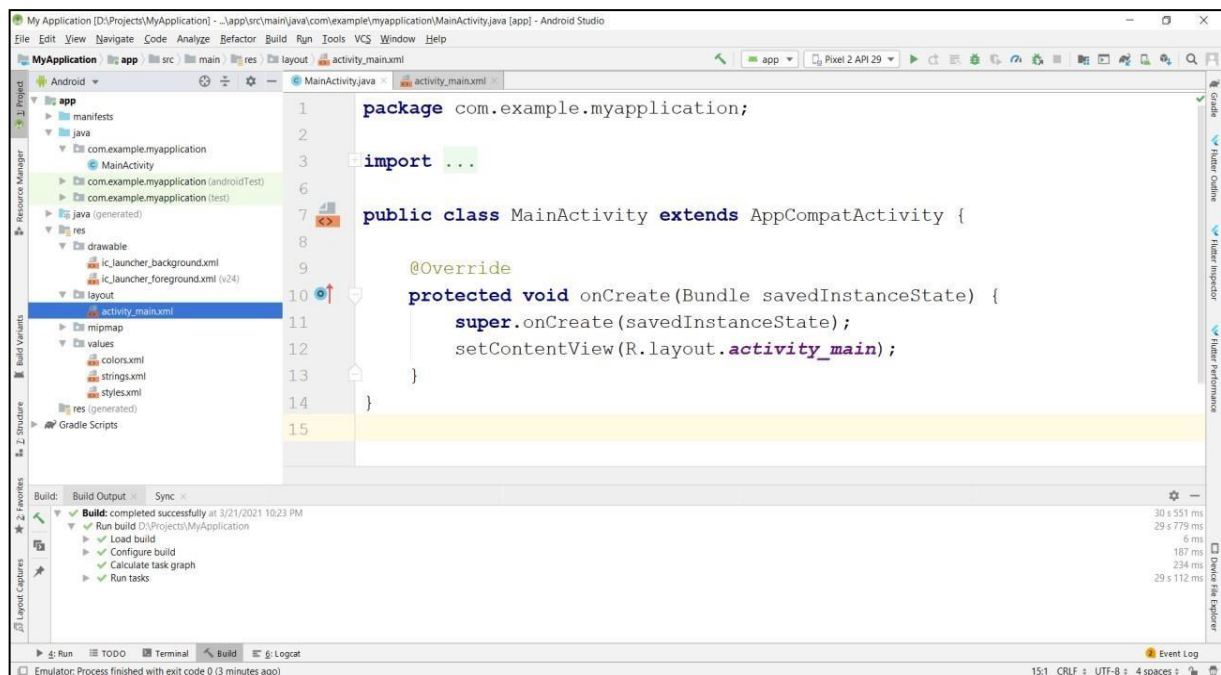


Fig : 3.4 – Basic View of IDE

Creating an Activity in Android:

Right Click on Package -> New -> Activity -> Empty Activity

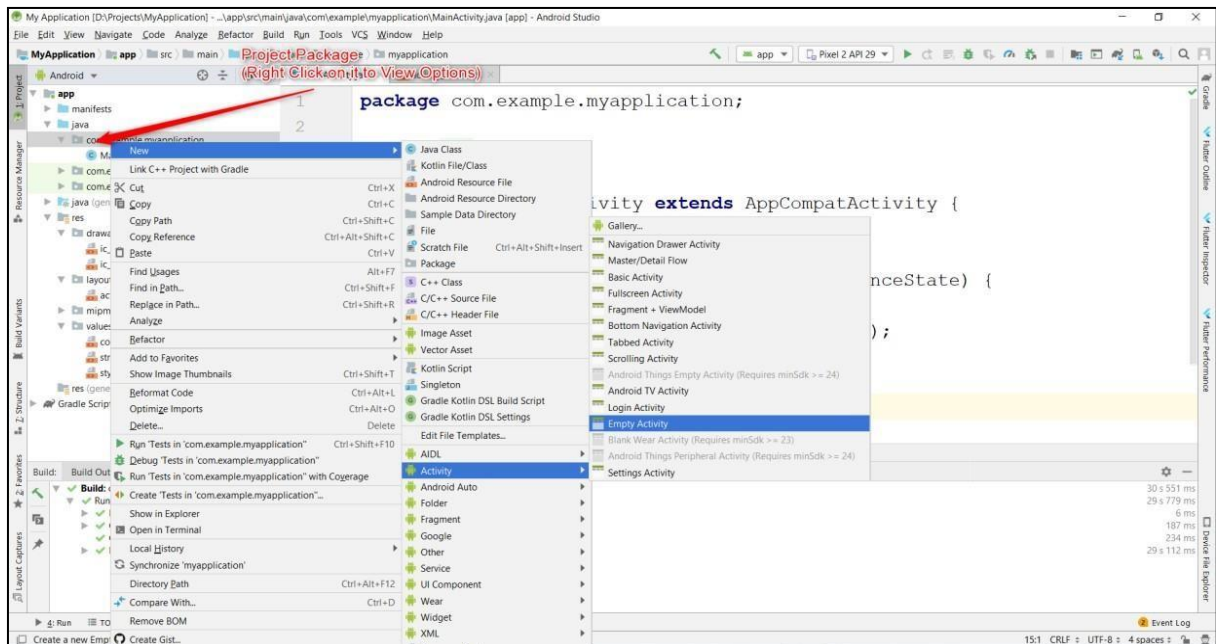
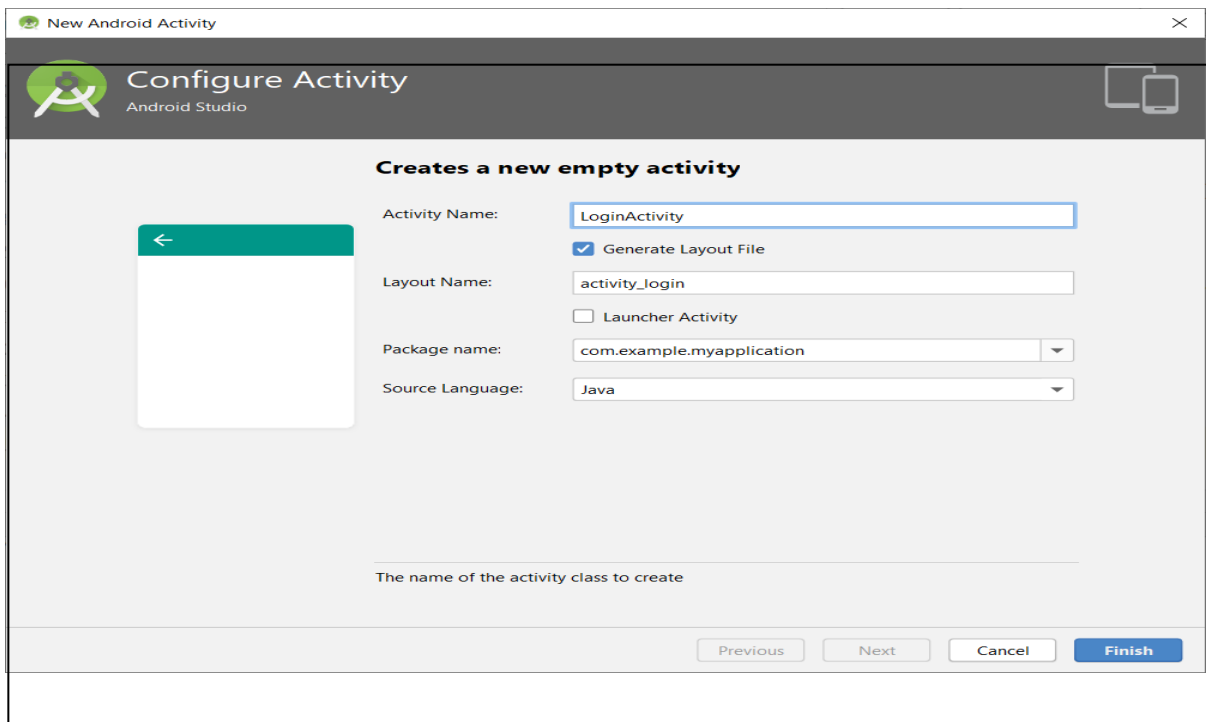
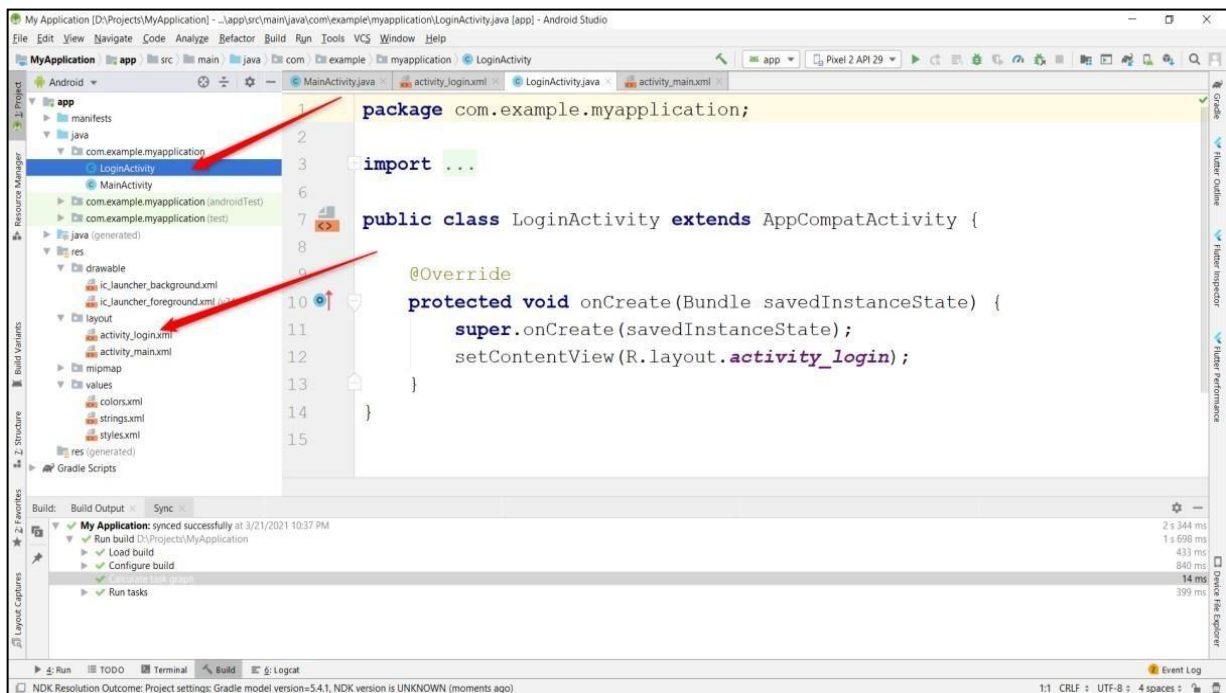


Fig : 3.5 – Creation of Android Activity

Enter Activity Name and Press Finish:





Creating a Layout in Android :

Right Click on Layout Folder -> New ->XML -> Layout XML File

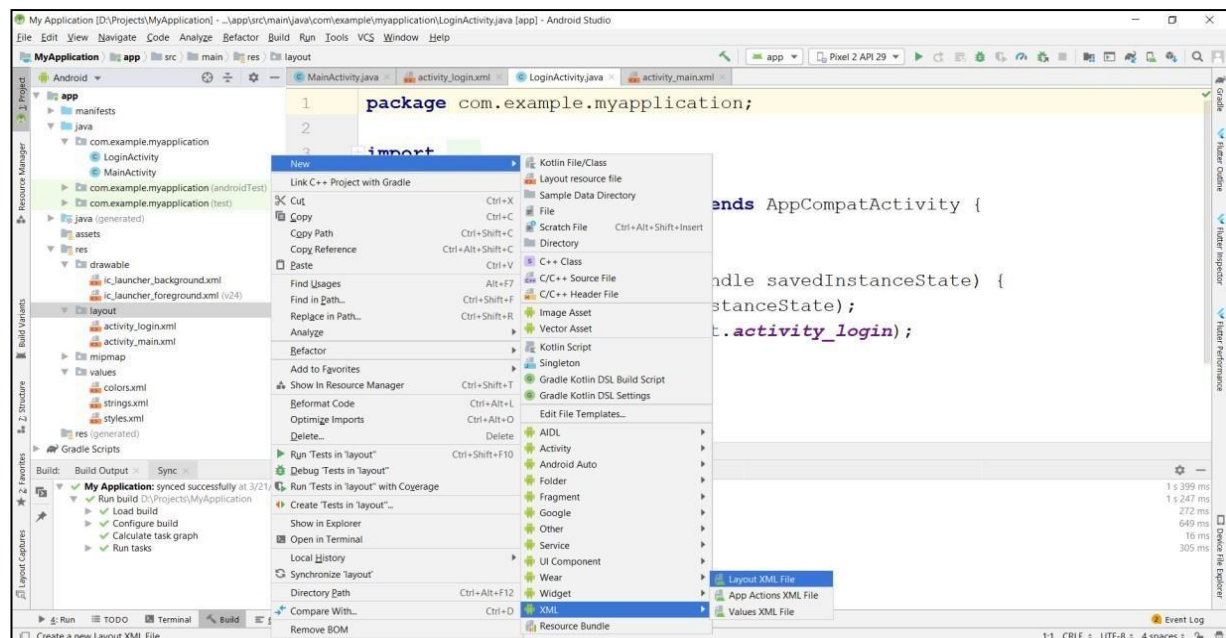
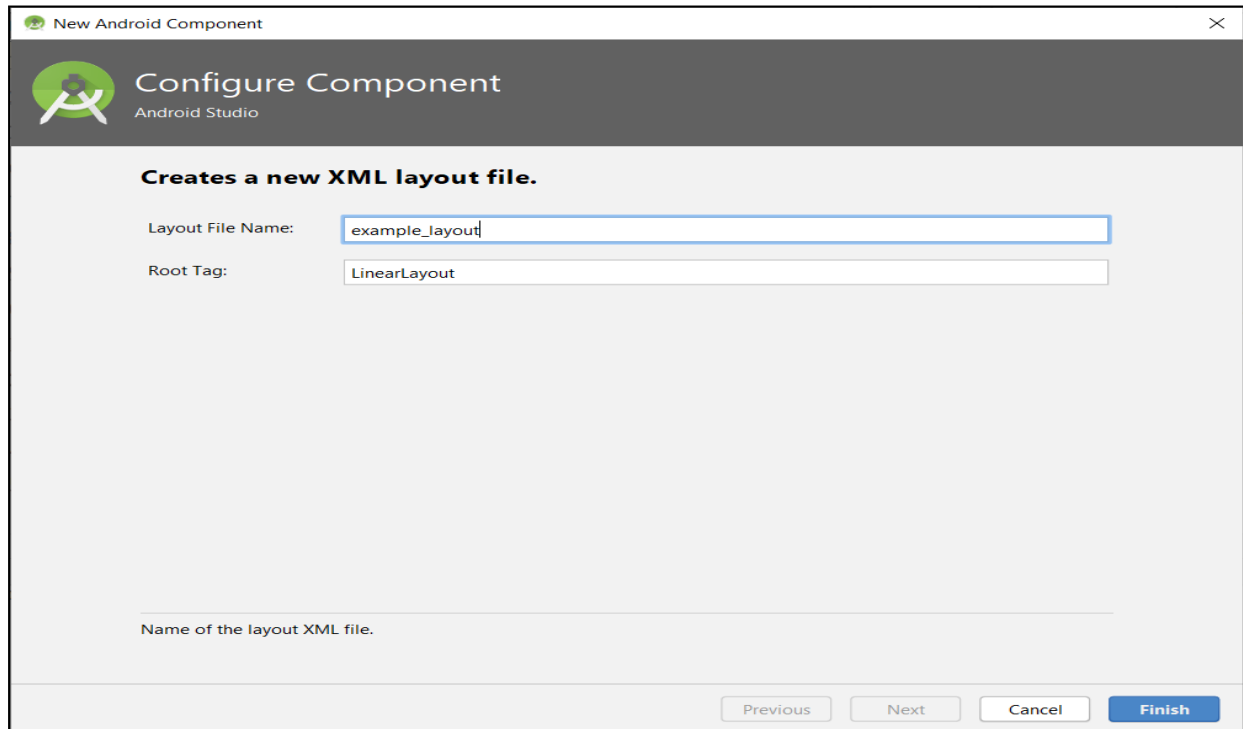


Fig : 3.6 – Creation of Layout in Android

Enter xml filename and press Finish:



Creating Assets Folder in Android:

Right Click on app folder -> New -> Folder -> Assets Folder ->
Press Finish Button

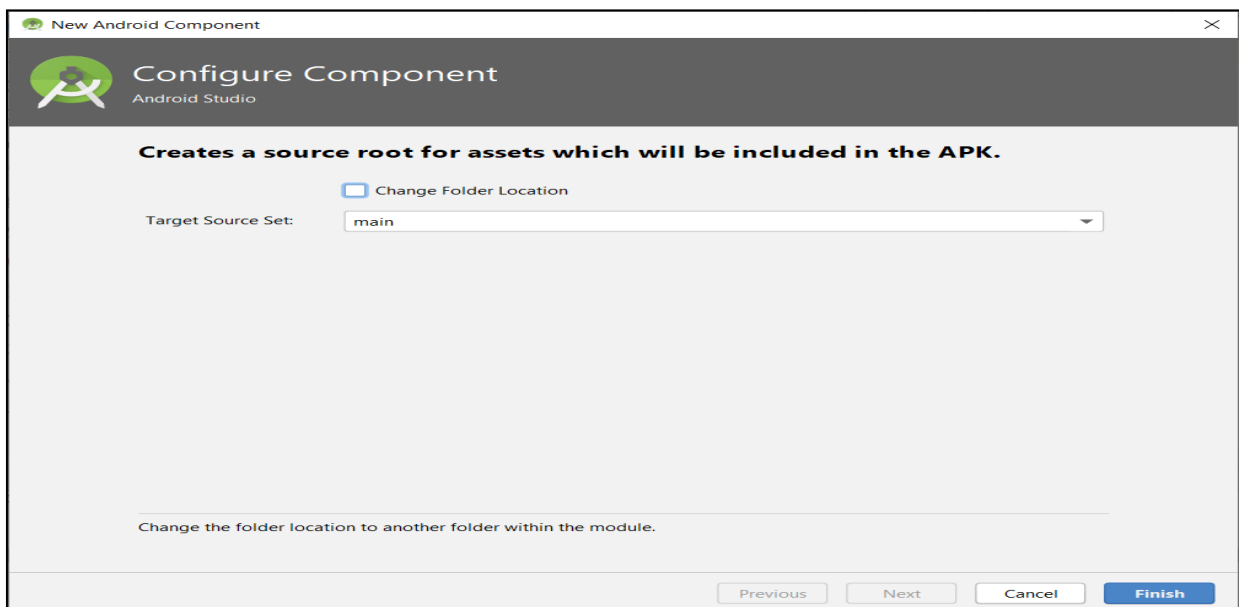
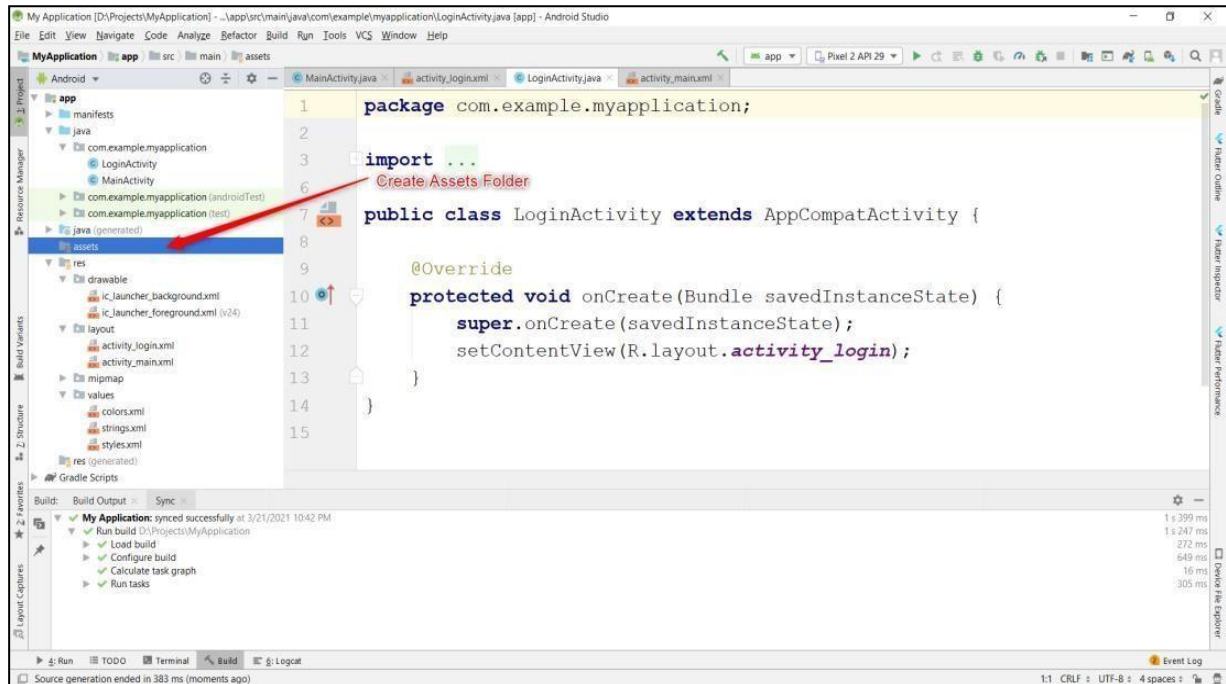
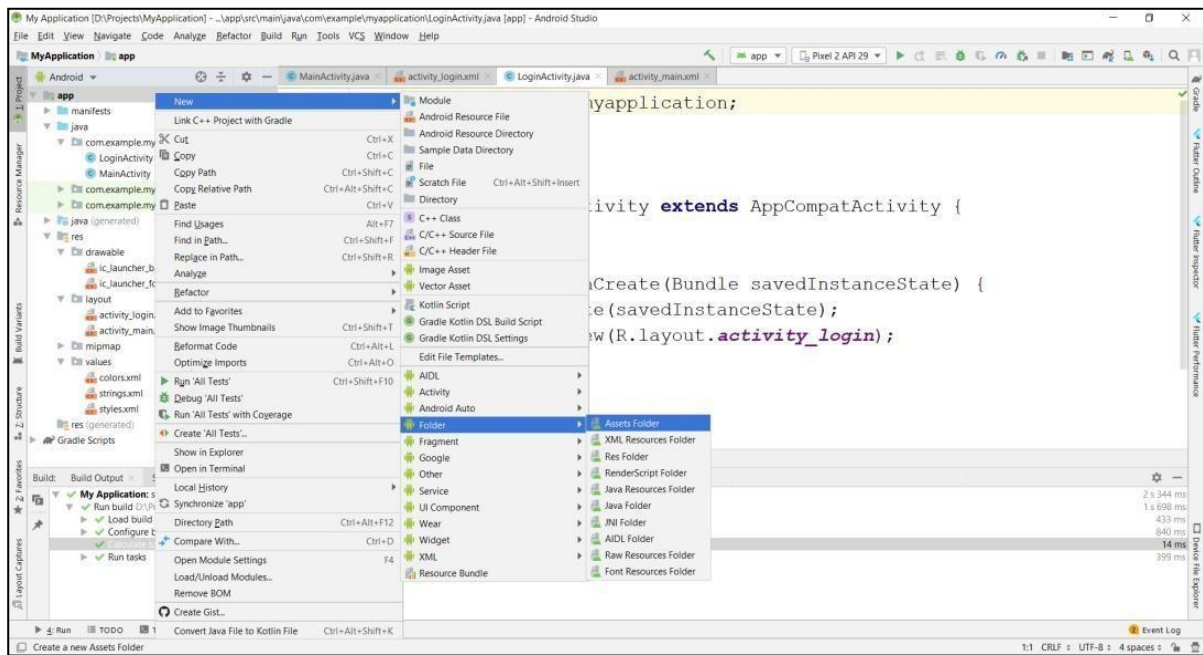


Fig : 3.7 – Creation of Assets Folder in Android



Creating File in assets Folder:

Right Click on assets folder -> New -> File

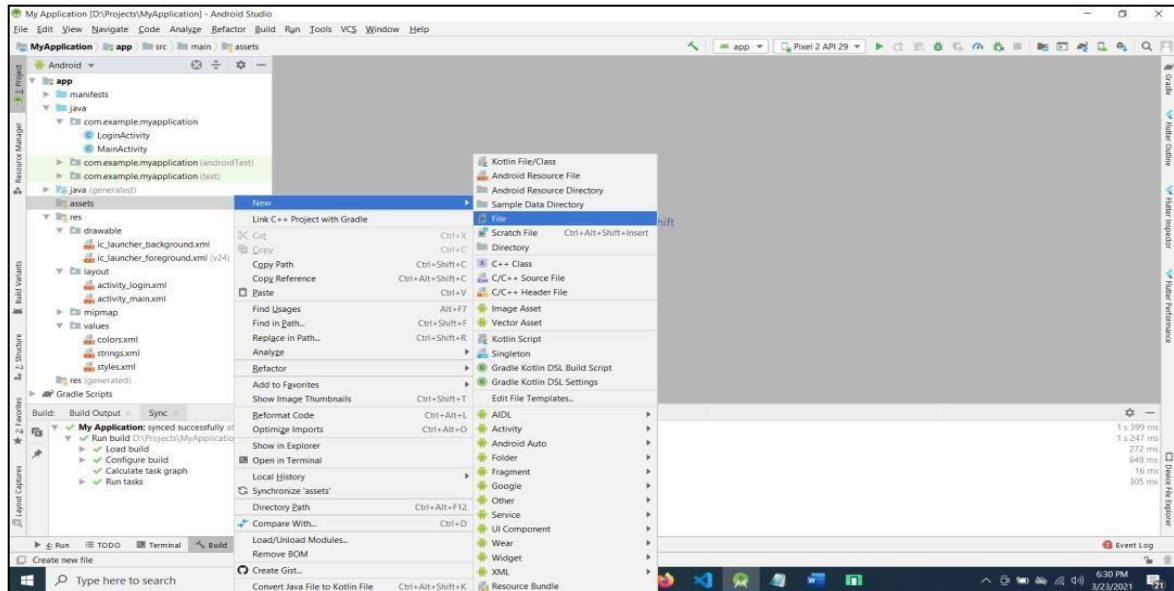
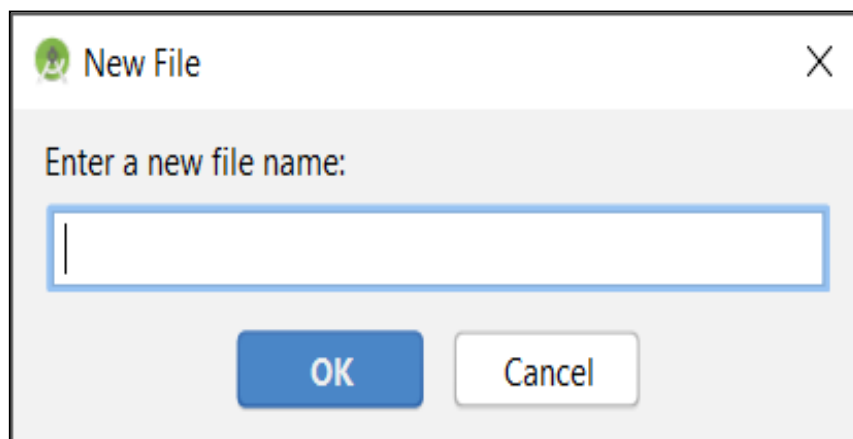


Fig : 3.8 – Creating File in assets folder

Enter filename with extension (Eg:abc.xml)



3.2 Code Lines for the Application

3.2.1 Java File

```
package com.example.recipes;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.os.PatternMatcher;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class MainActivity extends AppCompatActivity {

    EditText username,password;
    Button signup;

    String regexexp = "^(?=.*?[A-Z])(?=.*?[a-z])(?=.*?[0-9])(?=.*?[#?!@$%^&*~]).{8,}$";

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

        username=findViewById(R.id.username);
        password=findViewById(R.id.password);
        signup=findViewById(R.id.signup);
```

```
signup.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        String uname=username.getText().toString();  
        String pwd=password.getText().toString();  
  
        if(validatePassword(pwd)) {  
            Bundle bundle=new Bundle();  
            bundle.putString("username",uname);  
            bundle.putString("password",pwd);  
  
            Intent intent= new Intent(MainActivity.this,Loginmainact.class);  
            intent.putExtras(bundle);  
            startActivity(intent);  
  
        }  
  
        else {  
            Toast.makeText(MainActivity.this, "Invalid password",  
Toast.LENGTH_LONG).show();  
        }  
  
    }  
});  
}  
  
public boolean validatePassword(String pwd)  
{  
    Pattern pattern = Pattern.compile(regexexp);  
    Matcher matcher = pattern.matcher(pwd);  
    return matcher.matches();  
}  
}
```

Chapter 4

RESULTS

4.1 Sample Screenshots :

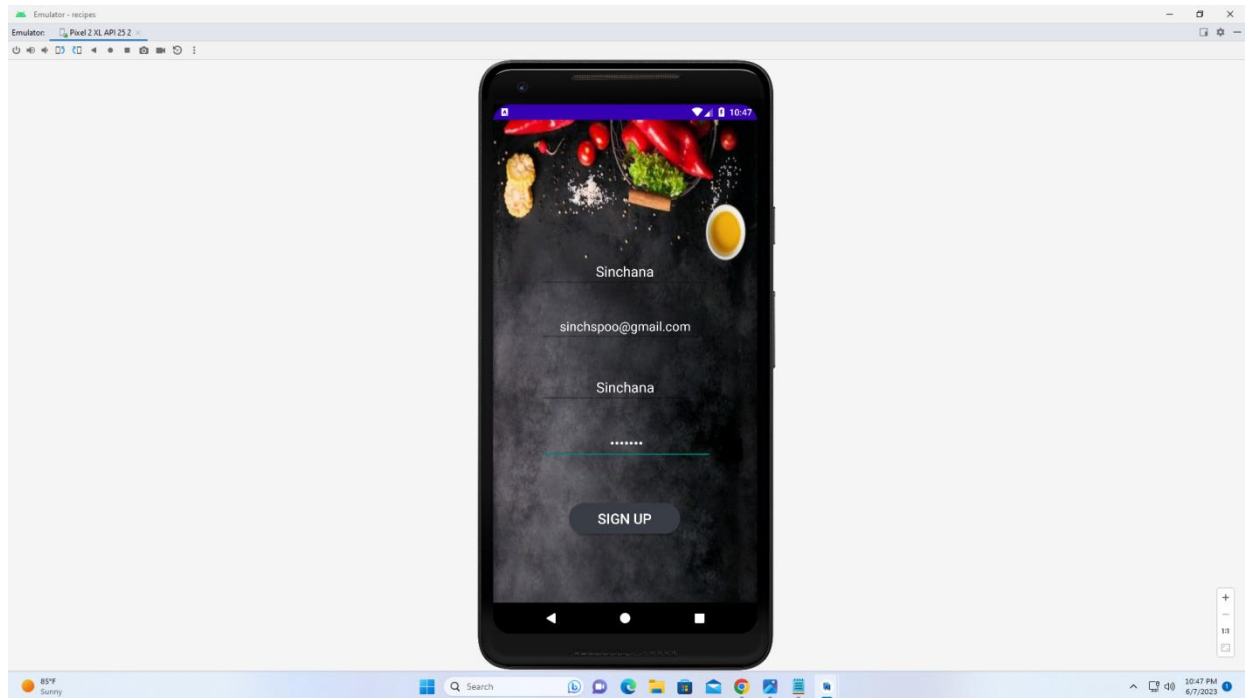


Fig : 4.1 – Sign Up Page

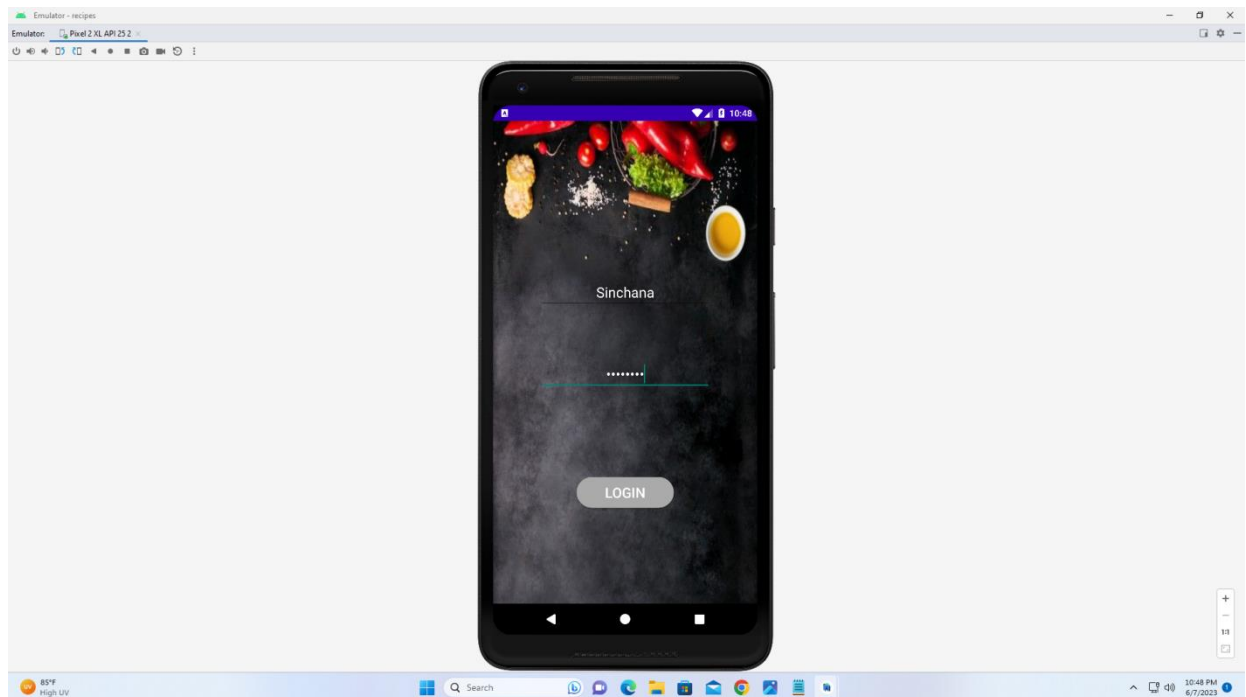


Fig : 4.2 – Login Page

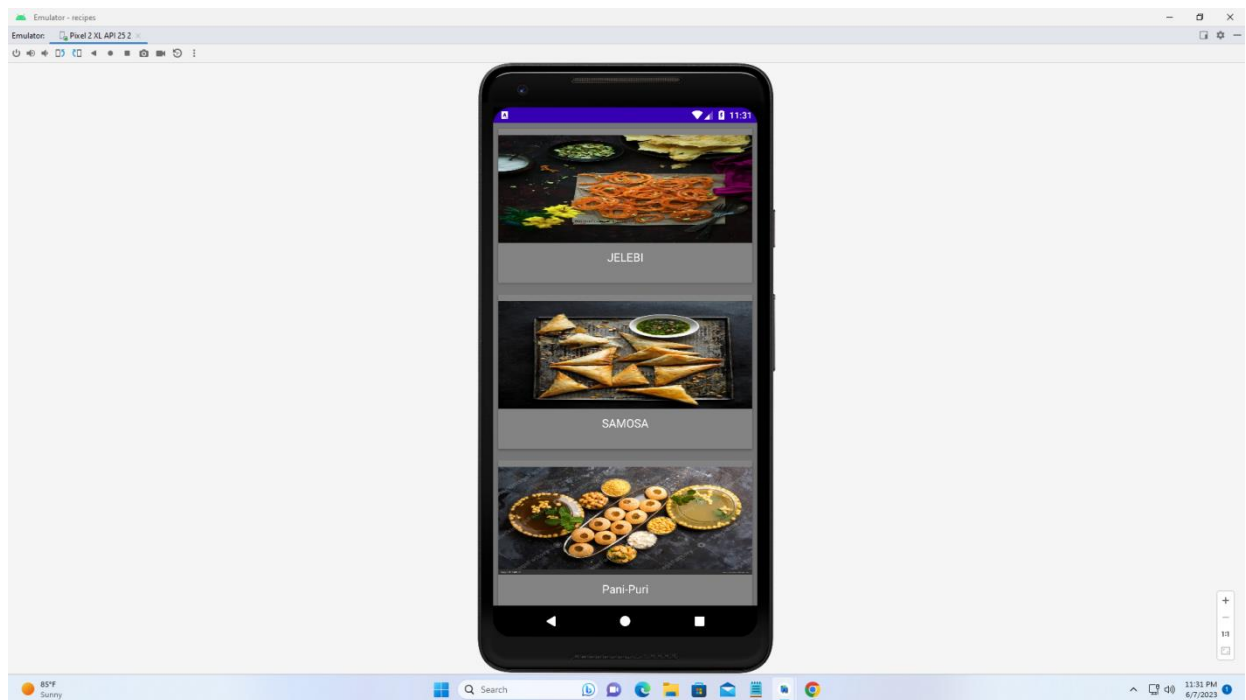


Fig : 4.3 – Recipes List

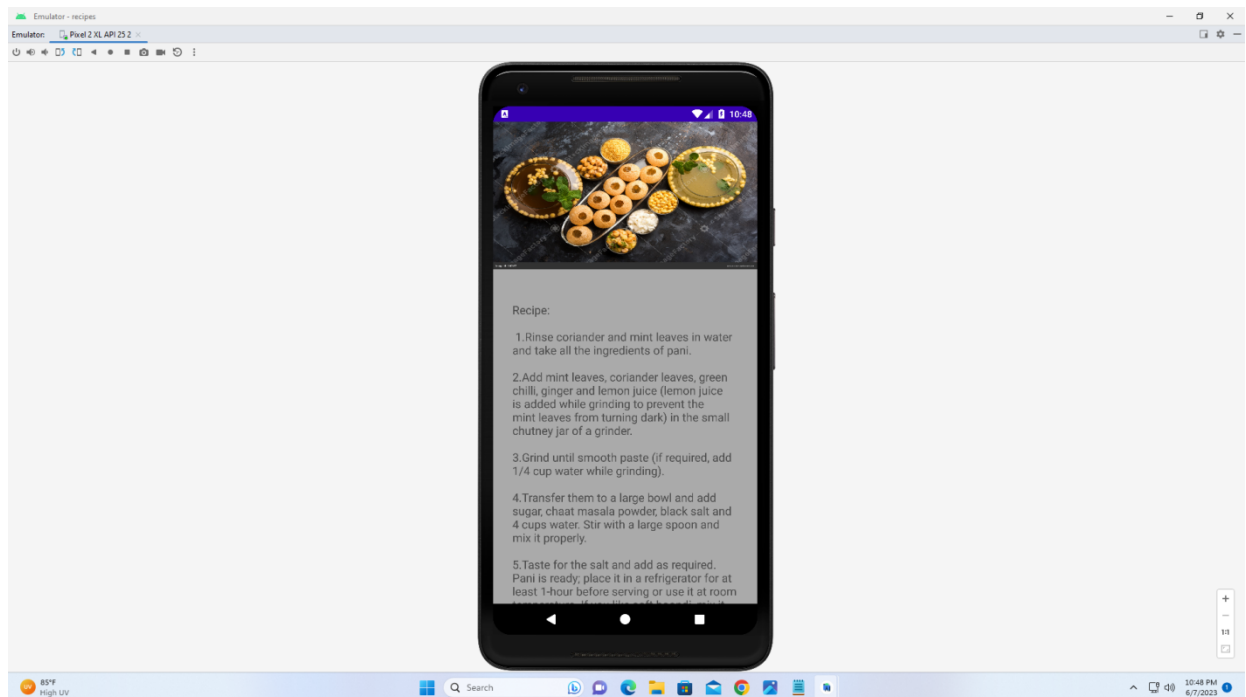


Fig : 4.4 – Description of Recipe

CONCLUSION

Cooking is a skill that requires work to perfect. Even while we try to provide as much information as we can about each recipe, presuming that all of our readers are beginners, it is occasionally necessary to have some prior cooking experience or fundamental cooking knowledge. We advise all food fans to thoroughly read our recipes, comprehend all of the instructions, suggestions, and serving suggestions before beginning the cooking process.

It is much easier to succeed the first time around if you are already familiar with the instructions in the recipe. As every application has some limitations so our project is not exceptional, but we will try to sort out them very shortly and deliver a defective less product to client. We are confident that this software package can be readily used by non-programming personal avoiding human handled chance of error.

BIBLIOGRAPHY

- [1] <https://www.vtuloop android-projects/.com/free/>
- [2] <https://foodviva.com/snacks-recipes/>
- [3] Punamiya Hanish et.al; International Journal of Advance Research, Ideas and Innovations in Technology – “Food recipe application”
- [4] www.videolan.org
- [5] www.wikipedia.com
- [6] ww.google.com
- [7] www.youtube.com