VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi – 590 018

Mobile Application Development

A Mini Project Work (18CSMP68) Report

on

"QUIZ 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

SANYAM SHARMA	**1KI20CS087**
VARSHA T S	**1KI20CS123**

Under the guidance of

SUPREETHA PATEL T. P. B.E., M.Tech., (Ph. D)

Assistant 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 – 572 201
Department of Computer Science and Engineering

CERTIFICATE

Certified that the Mini Project Work (18CSMP68) entitled **"Quiz Application"** is a bonafide work carried out by

SANYAM SHARMA	**1KI20CS087**
VARSHA T S	**1KI20CS123**

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**
Supreetha Patel T. P. Assistant Professor Dept. of CSE KIT, Tiptur	**Prof. Shashidhara M. S.** Associate Professor & HOD Dept. of CSE KIT, Tiptur

Name of Examiners	**Signature with Date**
1. _____	_____
2. _____	_____

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 **"QUIZ APPLICATION"** has been successfully completed under the guidance of **Supreetha Patel T. P.**, Assistant 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: 26/06/2023

Project Associates:

Sl. No.	*Student Name*	*USN*
1	SAYAM SHARMA	1KI20CS087 -
2	VARSHA T S	1KI20CS123 -

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 **Supreetha Patel T. P.,** Assistant Professor, Department of Computer Science and Engineering, Kalpataru Institute of Technology, Tiptur for her support, technical assistance and constructive suggestions and guidance for successful completion of our project. We are very much thankful to her 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.

Thank you,

Sl. No.	***Student Name***	***USN***
1	Sanyam Sharma	1KI20CS087
2	Varsha T S	1KI20CS123

ABSTRACT

The Quiz Application for Android is a user-friendly mobile application designed to provide a straightforward and enjoyable quiz experience for users. The project aims to create an easily accessible and interactive platform that enables users to test their knowledge and have fun while learning.

The application offers a simple and intuitive interface, allowing users to navigate through quiz questions with ease. Further development can be made which provides the users can choose from predefined quiz categories or create their own custom quizzes, tailoring the content to their interests and learning objectives.

The project utilizes modern Android development tools and follows industry best practices to ensure a smooth and reliable user experience. It prioritizes simplicity and efficiency, making it suitable for a wide range of users, including students, professionals, and casual learners.

TABLE OF CONTENTS

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 superceded 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

Date	Version	Nickname	API Level
Nov 2014	5.0	Lollipop	21
Oct 2015	6.0	Marshmallow	23
Aug 2016	7.0	Nougat	24
Mar 2017	O preview	*Android O Developer Preview*	

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.

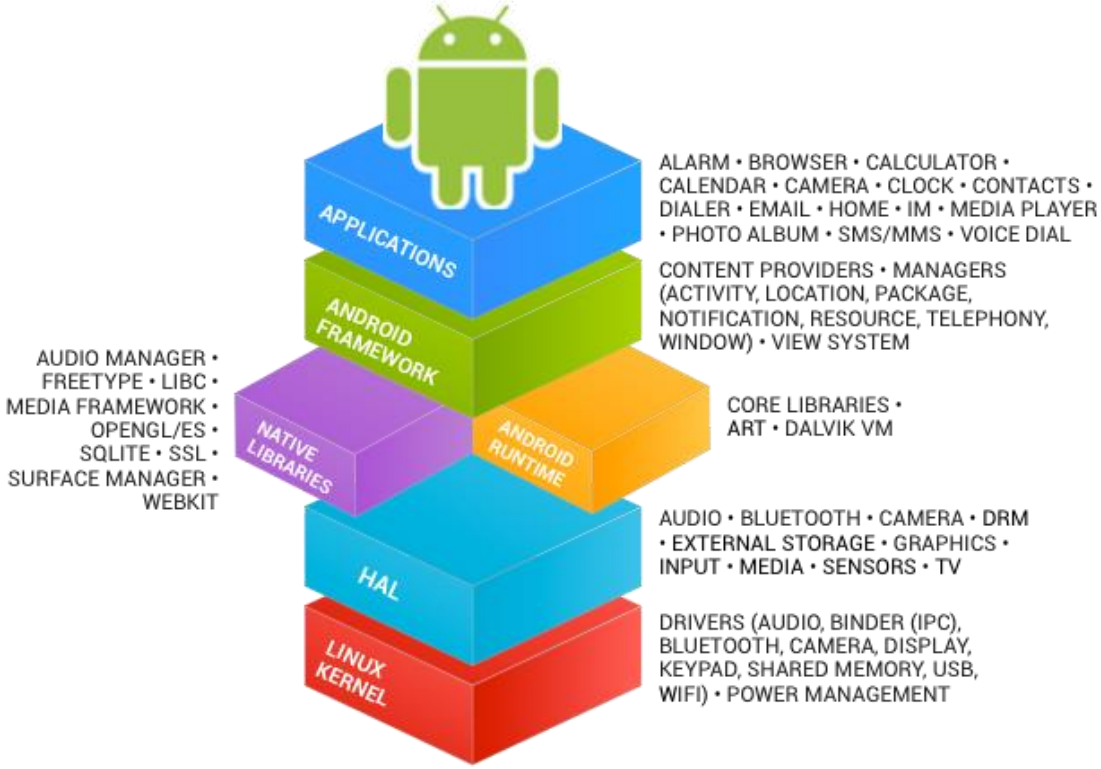


Fig 1: Android Architecture

The Android platform is built as a layered architecture:

- At it's 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 an 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.

1.2.1 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. 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 (E**X**tensible **M**arkup **L**anguage). 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. Android made up; so it's like defining web pages, except with a new set of elements.

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

- **System** : 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 | Java

2.2 Installation Procedure of Android Studio

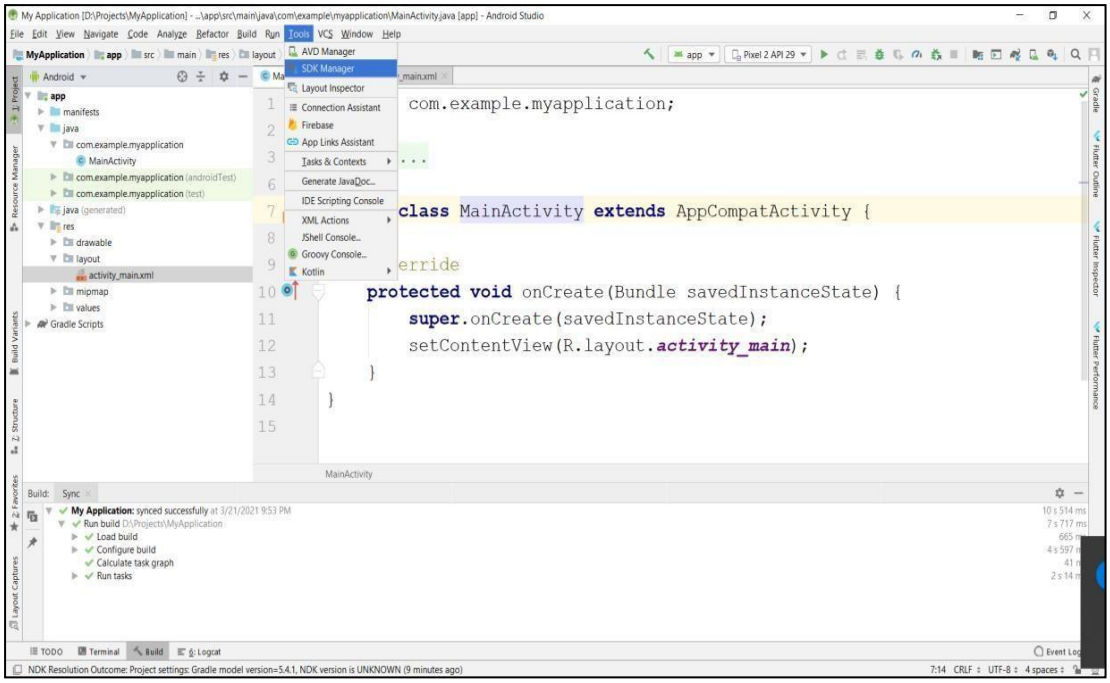
Install Android Studio and Packages:

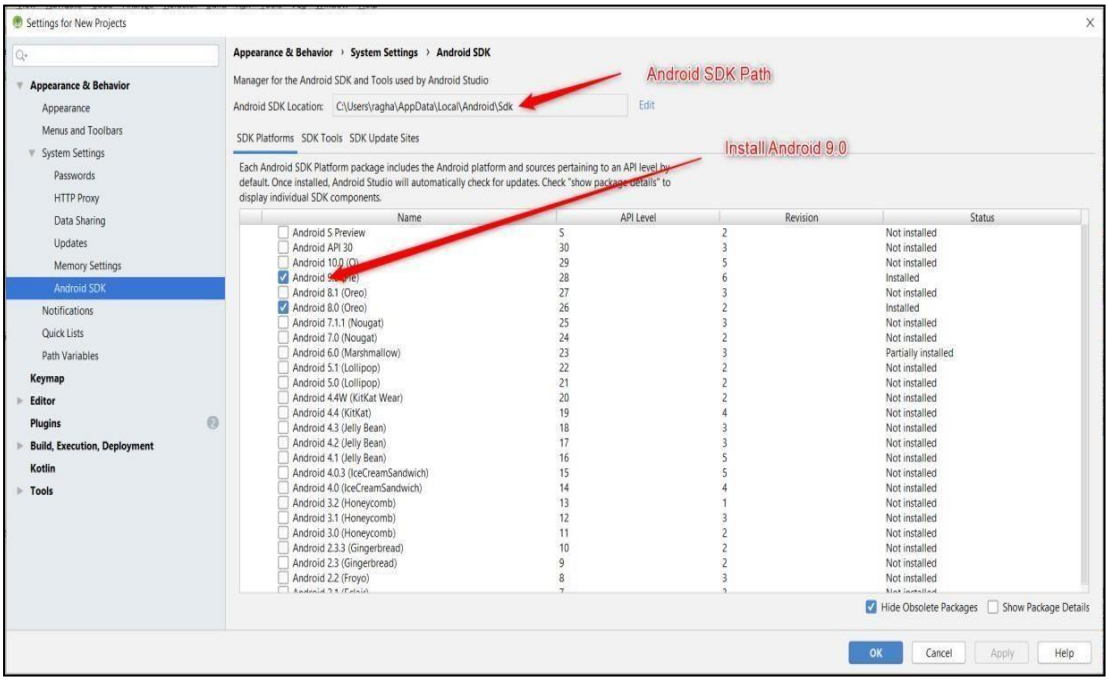
Download Android Studio from the below link :

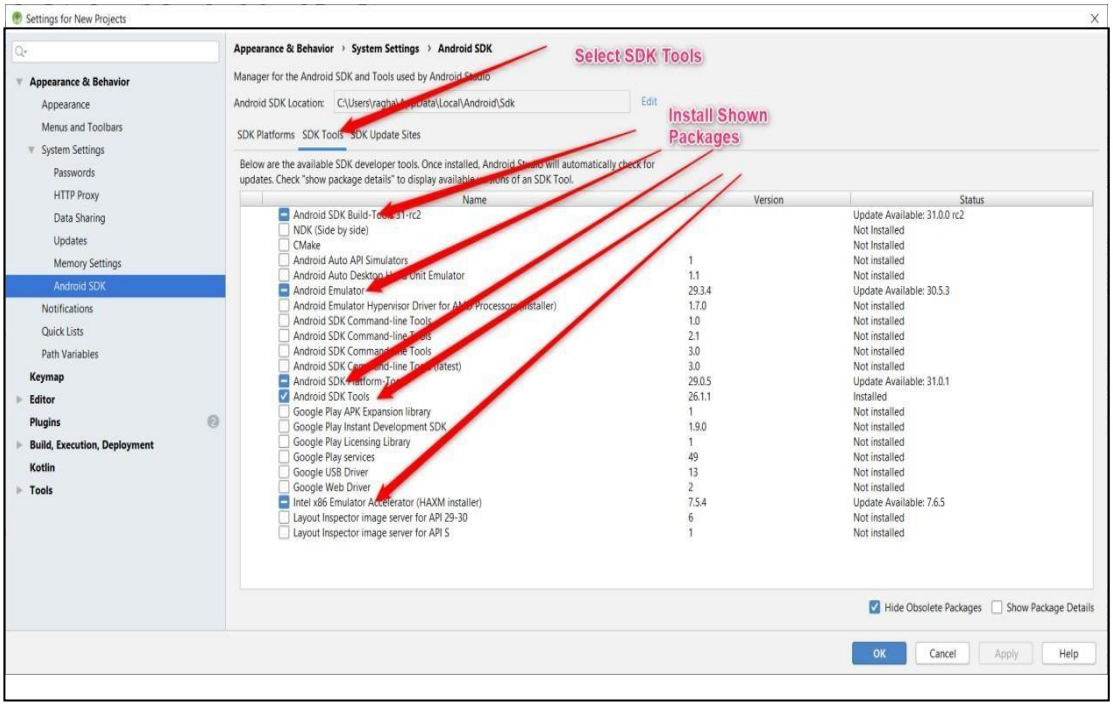
https://developer.android.com/studio

Configure Android SDK packages:

Go to Tools SDK Manager

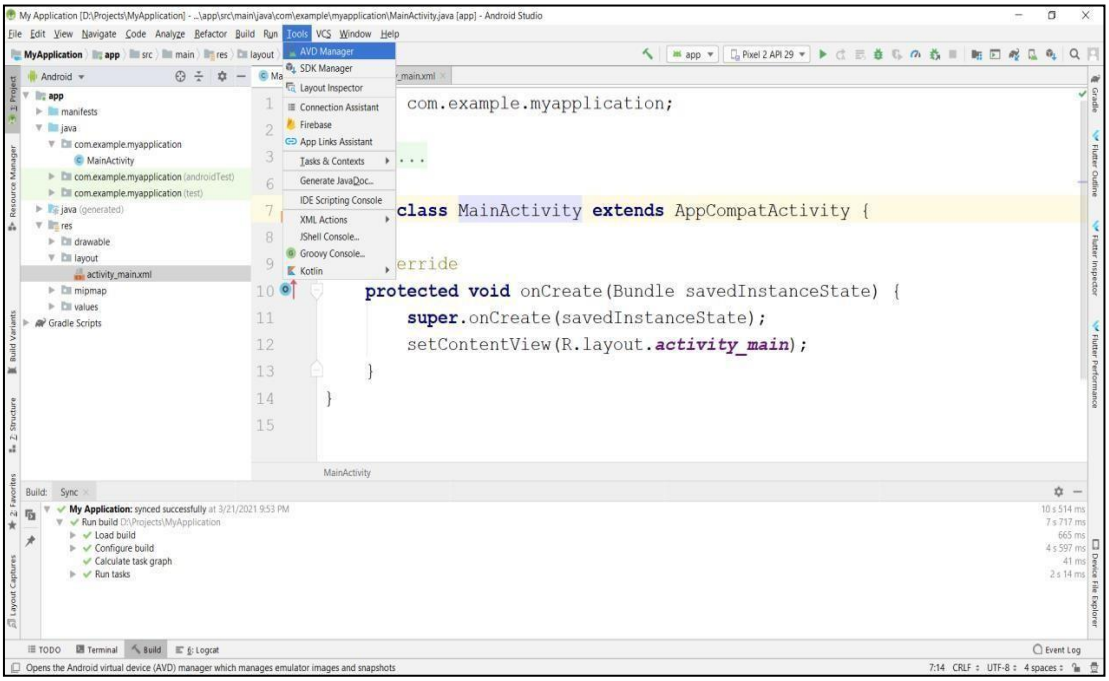


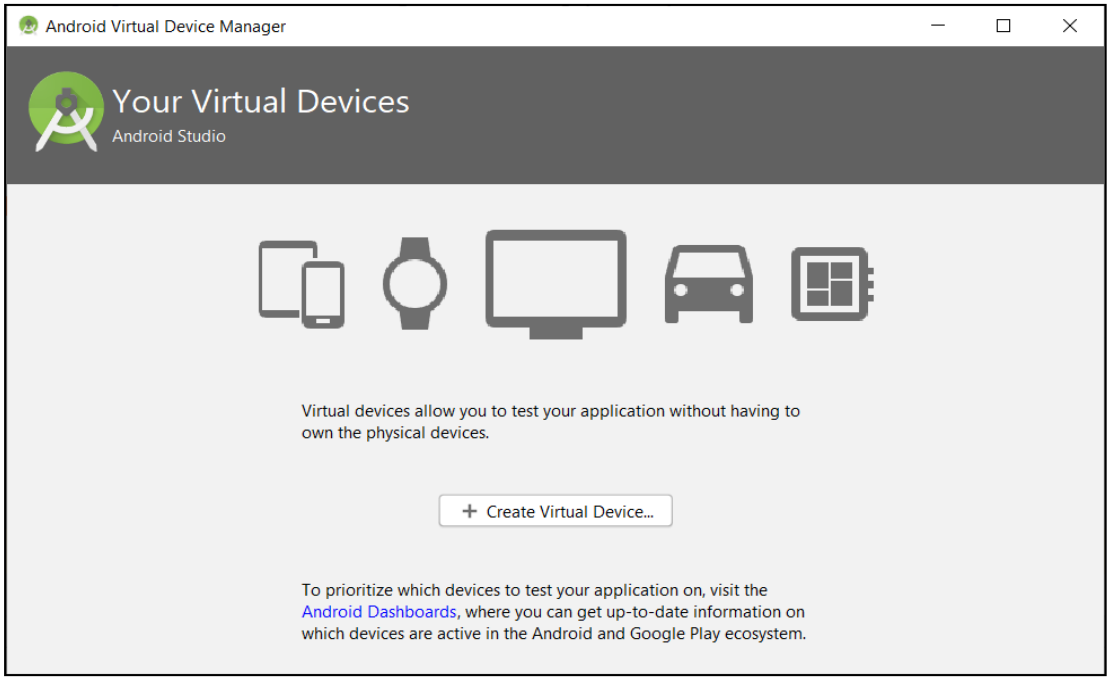




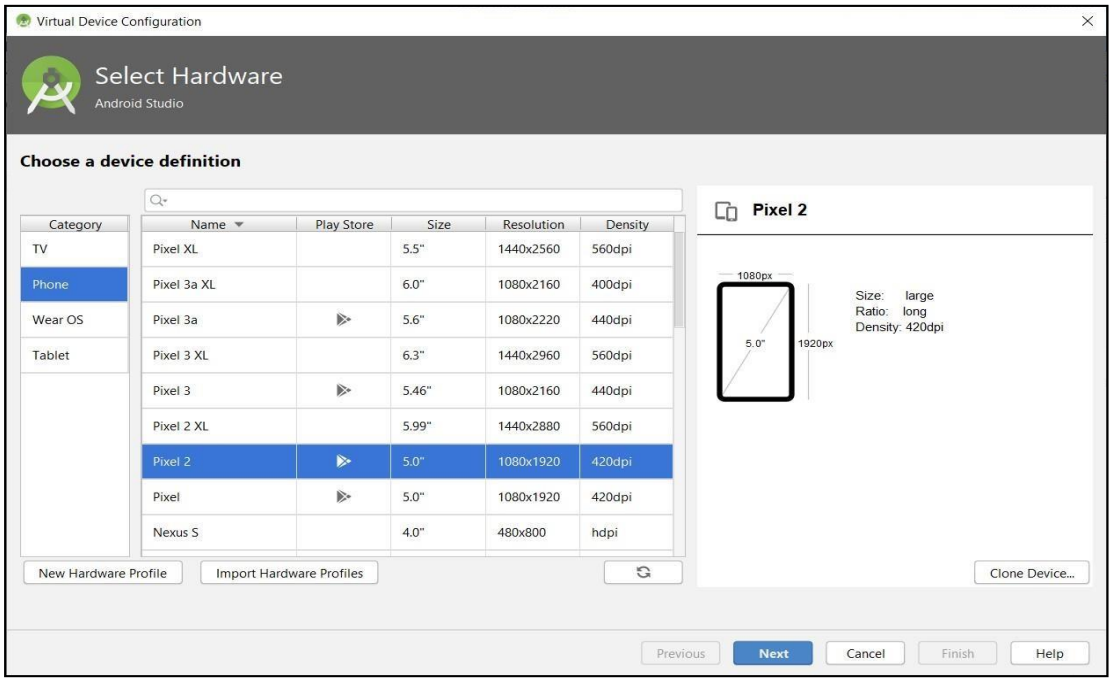
Creating Emulator

Go to Tools Select AVD Manager

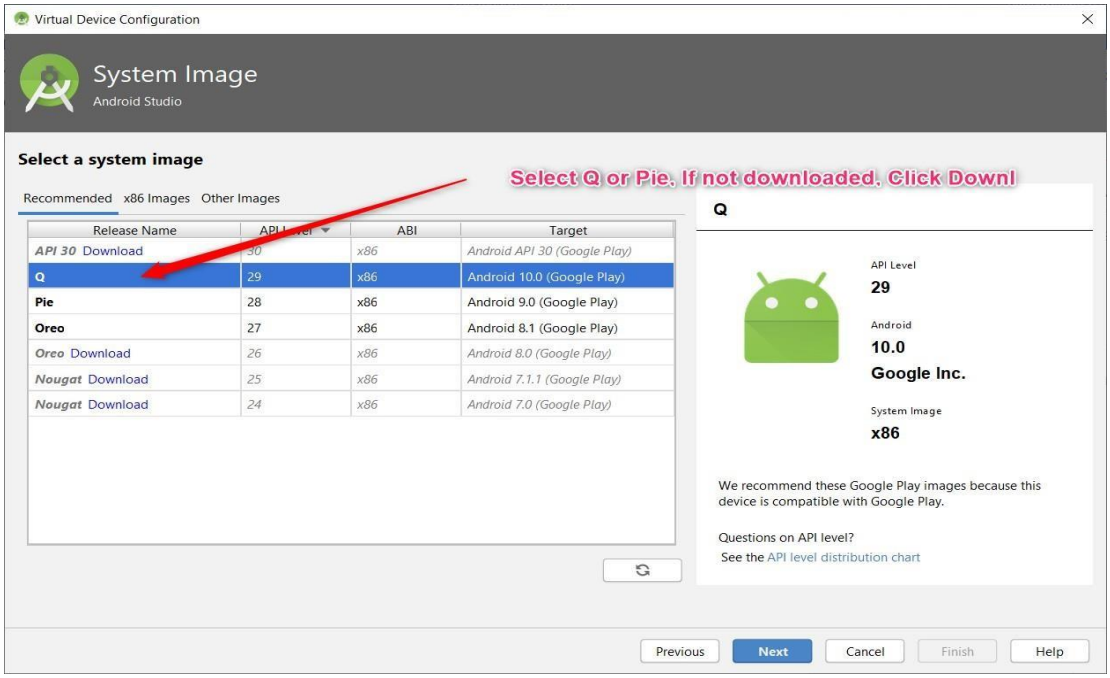




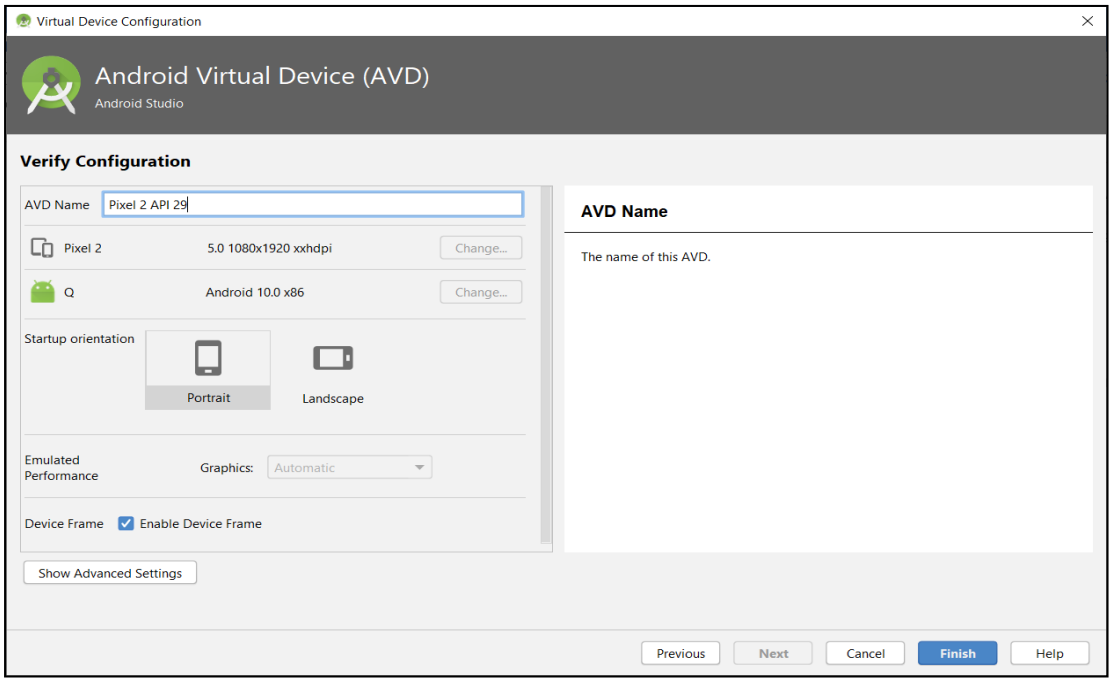
Select Create Virtual Device □Select Phone □Pixel 2 □ Press Next



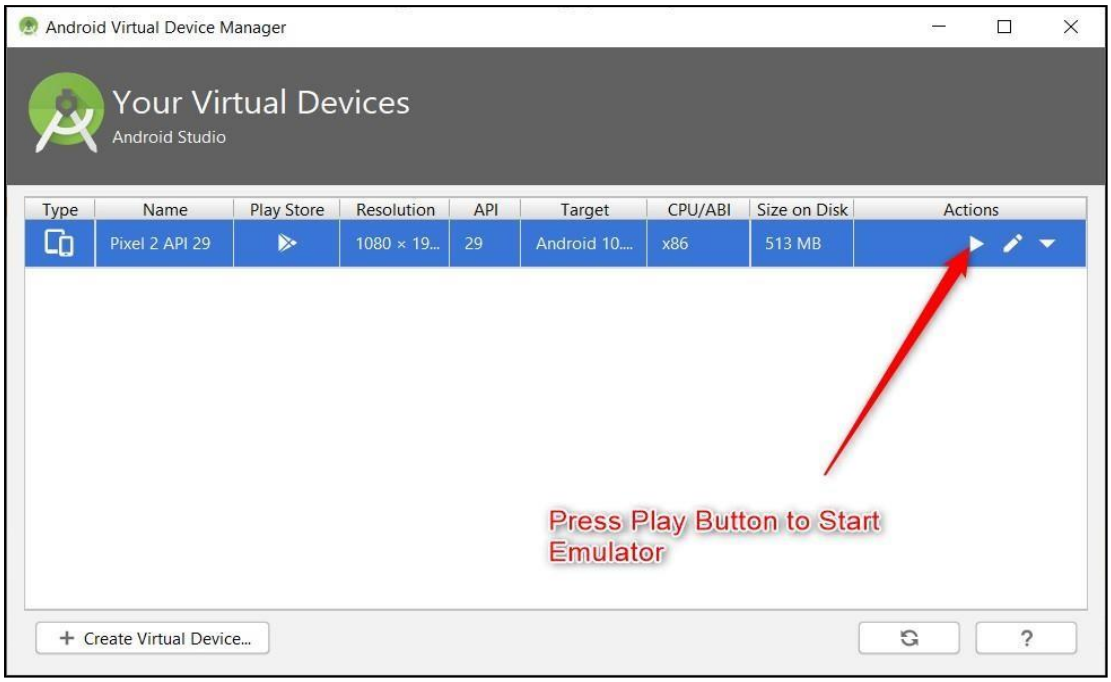
Select Android Q, if not already downloaded press download, After download completes SelectQ and Press Next Button.

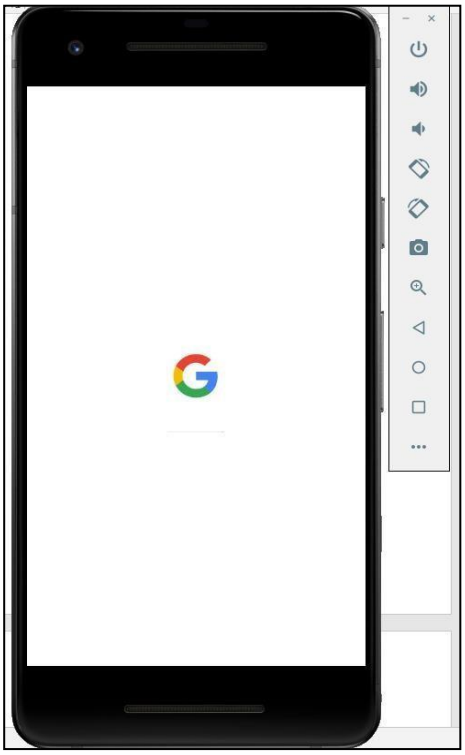


Enter AVD Name and Press Finish.



Press Play Button to Start Emulator





2.3 System Architecture

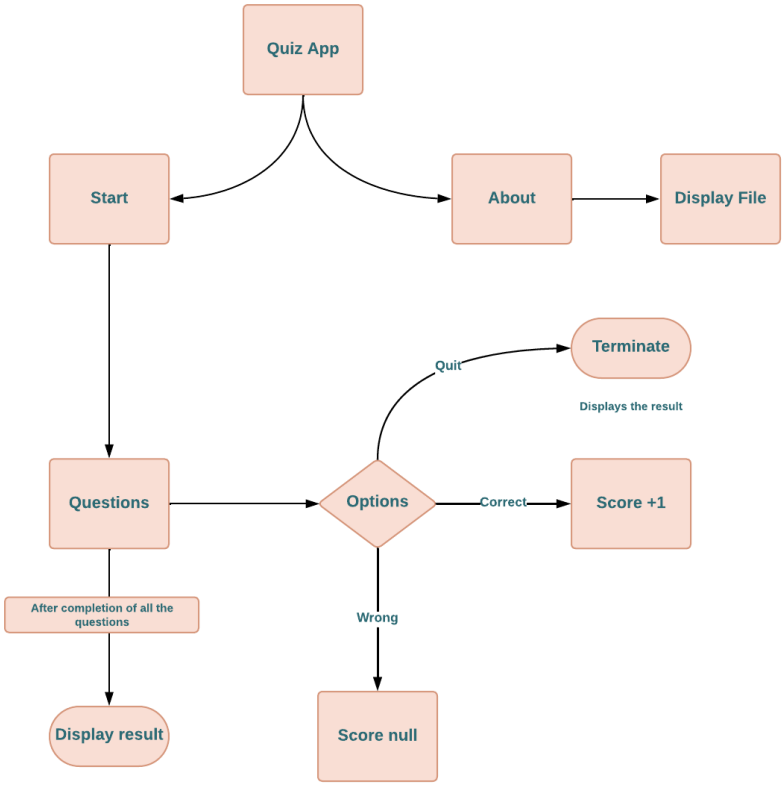


Fig – System Architecture of quiz application

The aforementioned flowchart shows how our Android quiz application is created as well as the alternatives that are offered to users. When the quiz app is launched, two buttons, labelled "start" and "about," are displayed. When the 'about' button is pressed, information about the developers is shown, and when the'start' button is clicked after the user name has been entered, it links to the quiz page where the questions are displayed one after the other.The following question is displayed if the selected option is correct and moves on. If the answer is wrong, it still advances to the following question but displays incorrect. There is also a 'quit' option, which takes you to the results page and displays the number of questions answered, the number of correct answers, and the number of incorrect answers. If all of the questions are answered, the results page will be shown.

Chapter 3

IMPLEMENTATION DETAILS

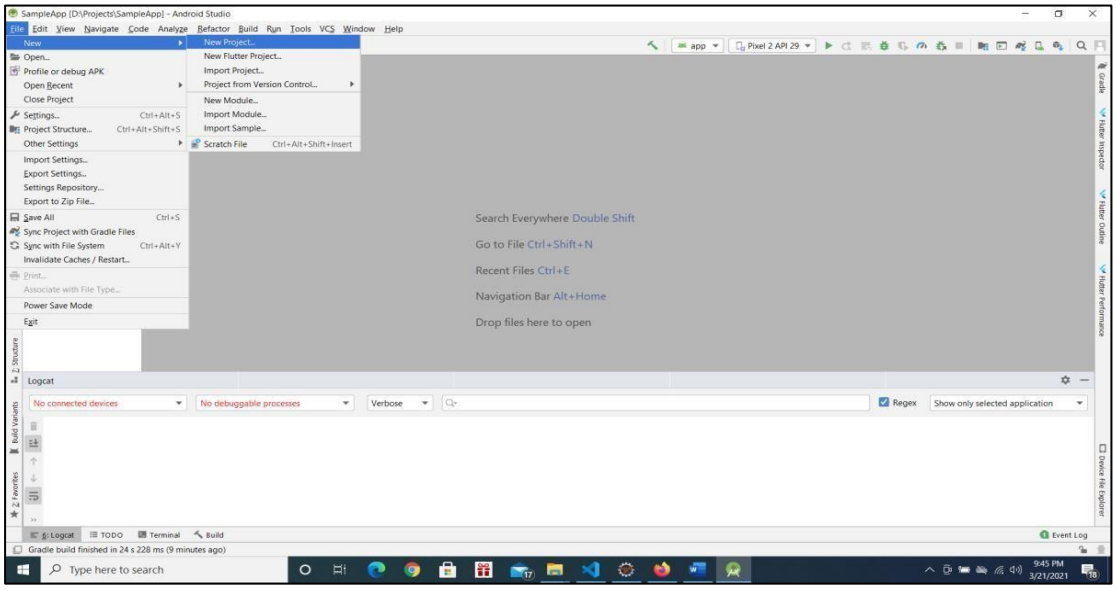
The implementation phase involves more than just writing code. Code also needs to be tested and debugged as well as compiled and built into a complete executable product. We usually need to utilize configuration management in order to keep track of different version of code. This is the stage of the 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.

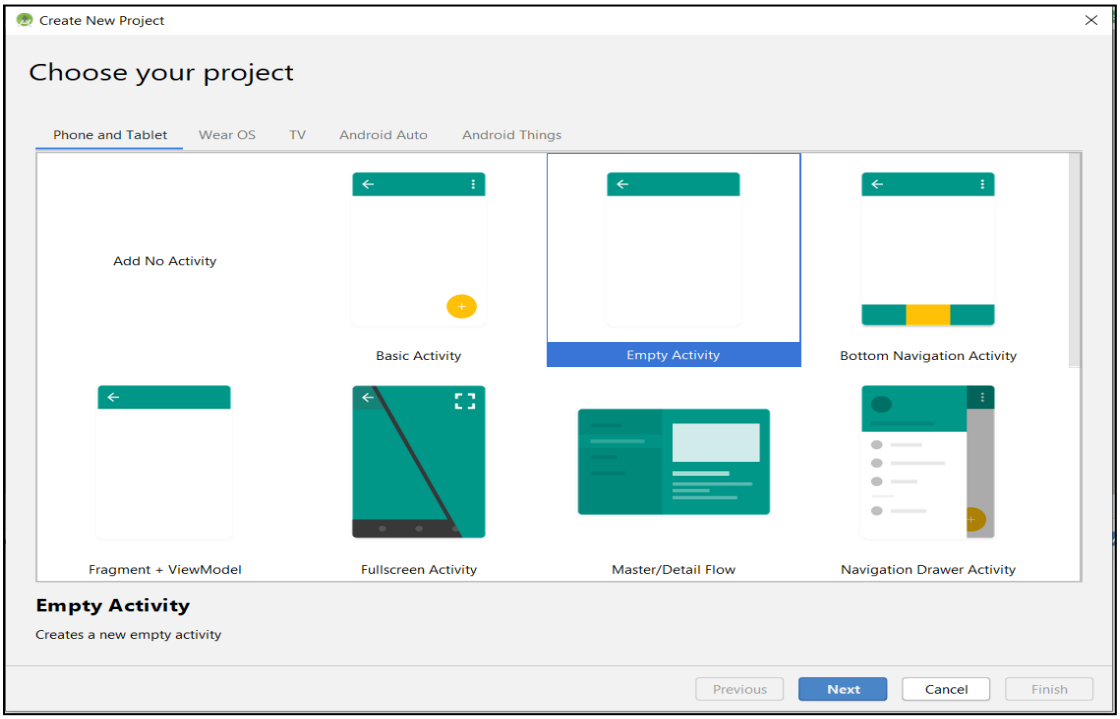
Creating a New Project in Android

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

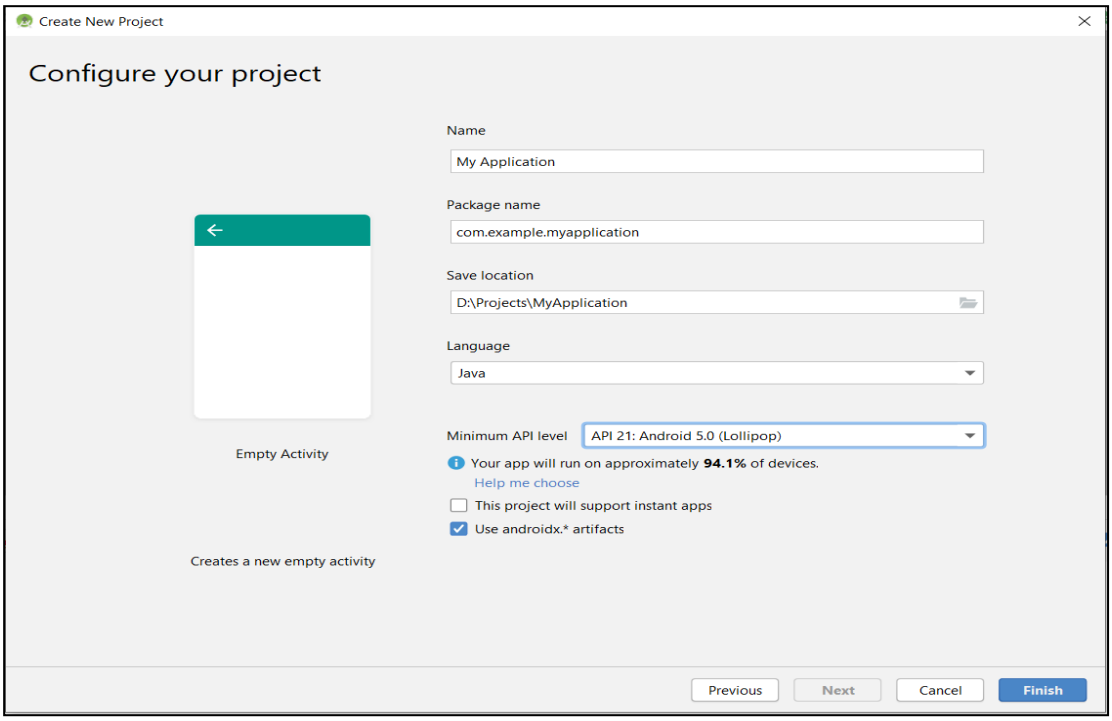
Go to File □New □New Project



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 HomeScreen Icon.

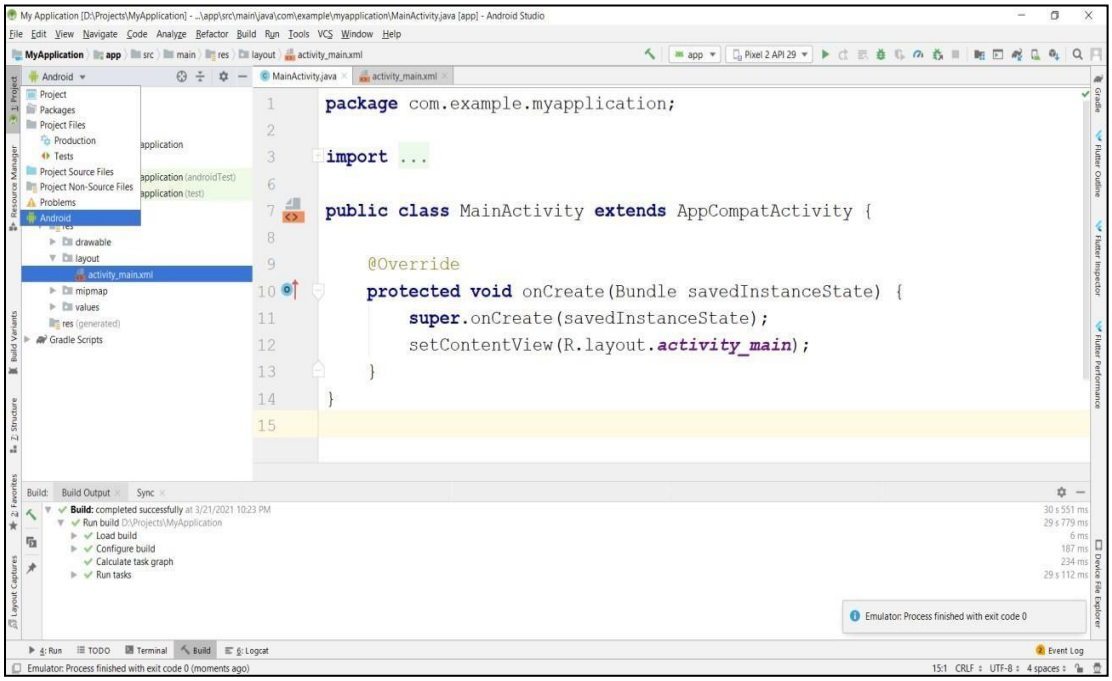
Package Name Enter package name atleast two identifier (Eg: com.example). Best Practice is3 or more identifier (Eg: com.example.firstapp).

Save Location Location where to save the ProjectLanguage Choose Java

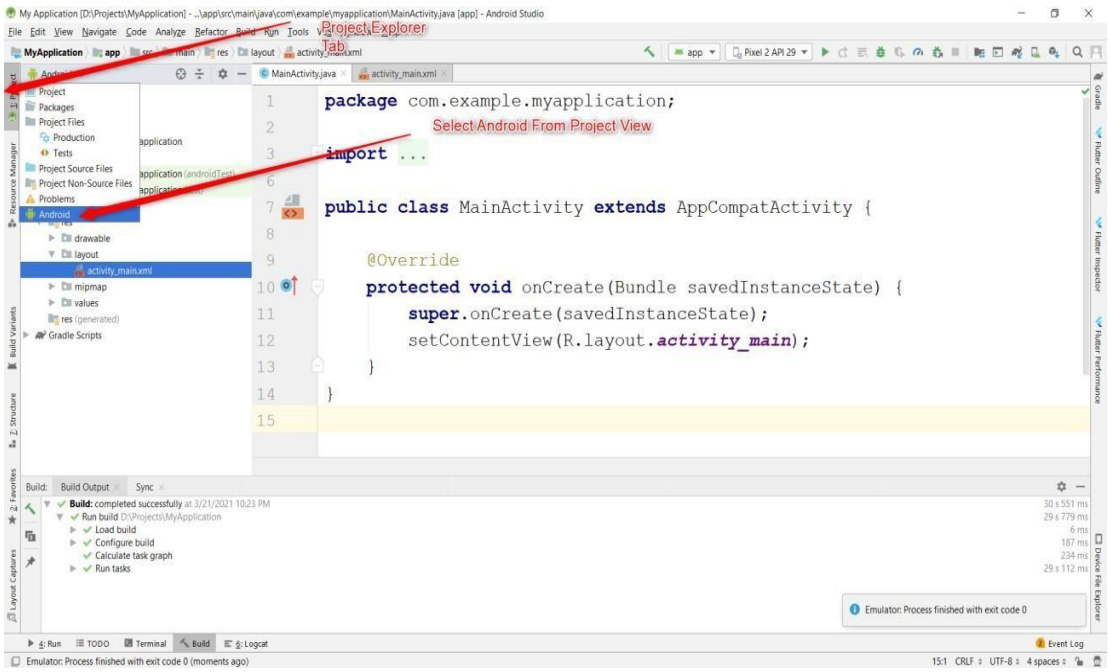
Minimum API Level Android 5.0

Select Checkbox Use androidx.artifacts folder as below screenshot.

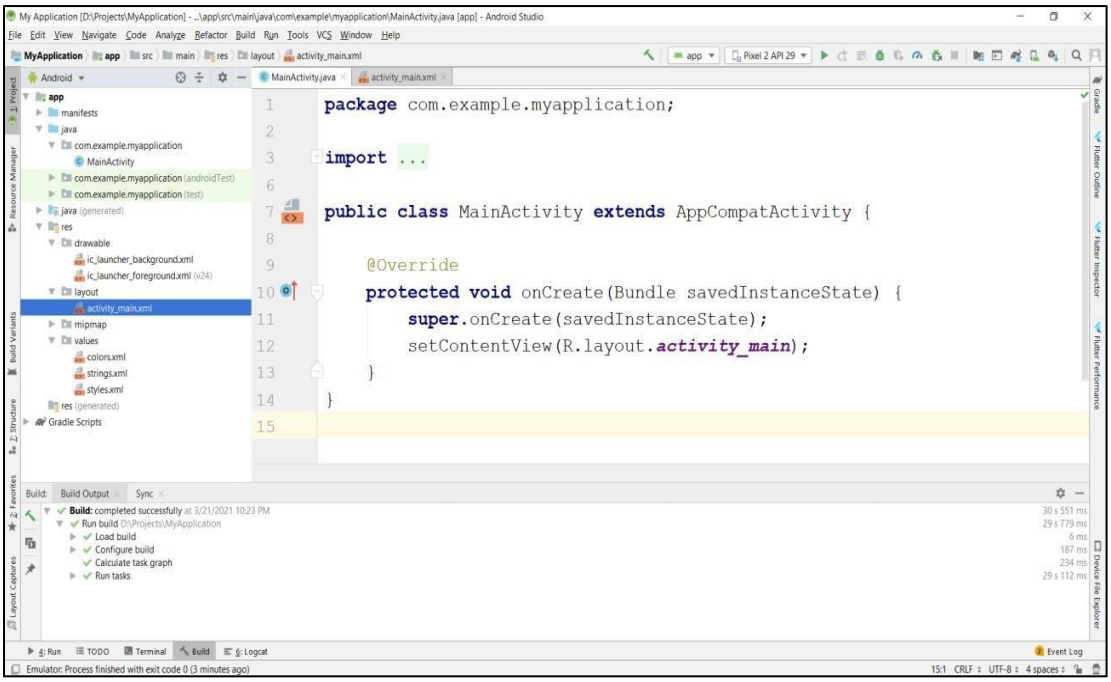
Android Project Structure:



Select Project Explorer and Select Android from Project View

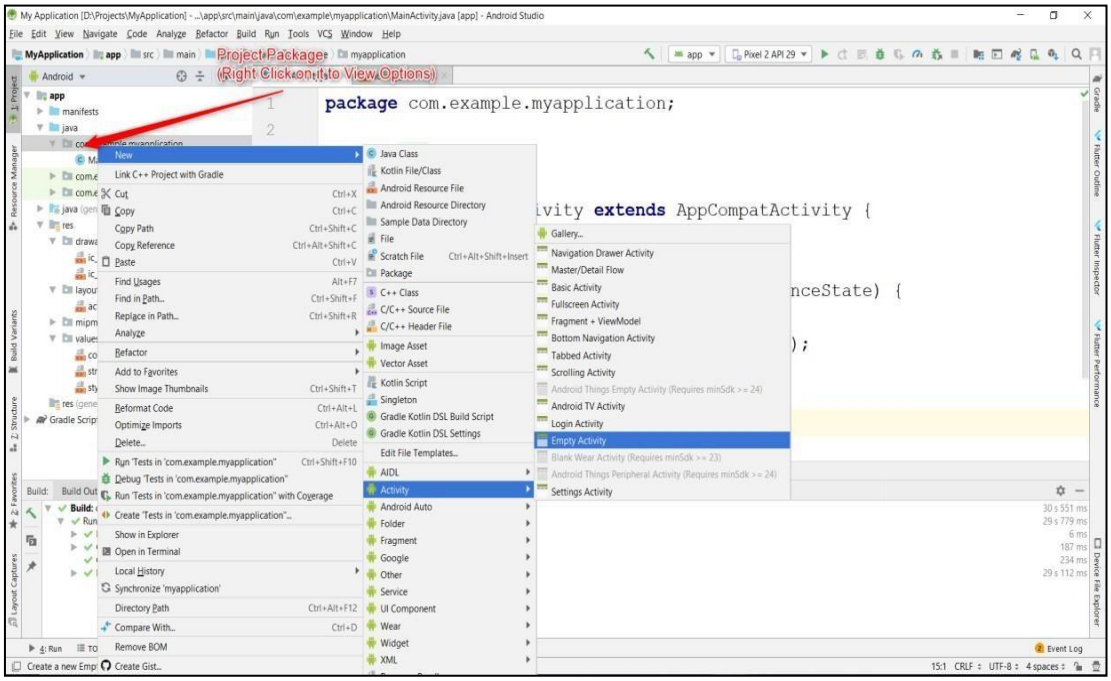


Basic View:

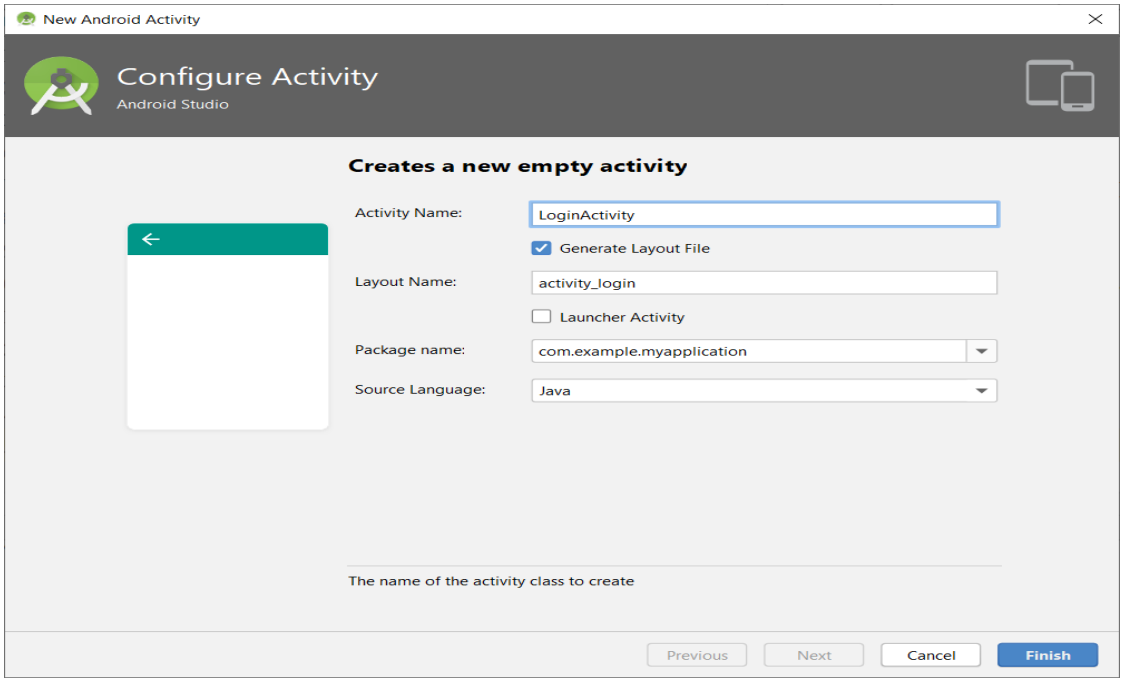


Creating an Activity in Android

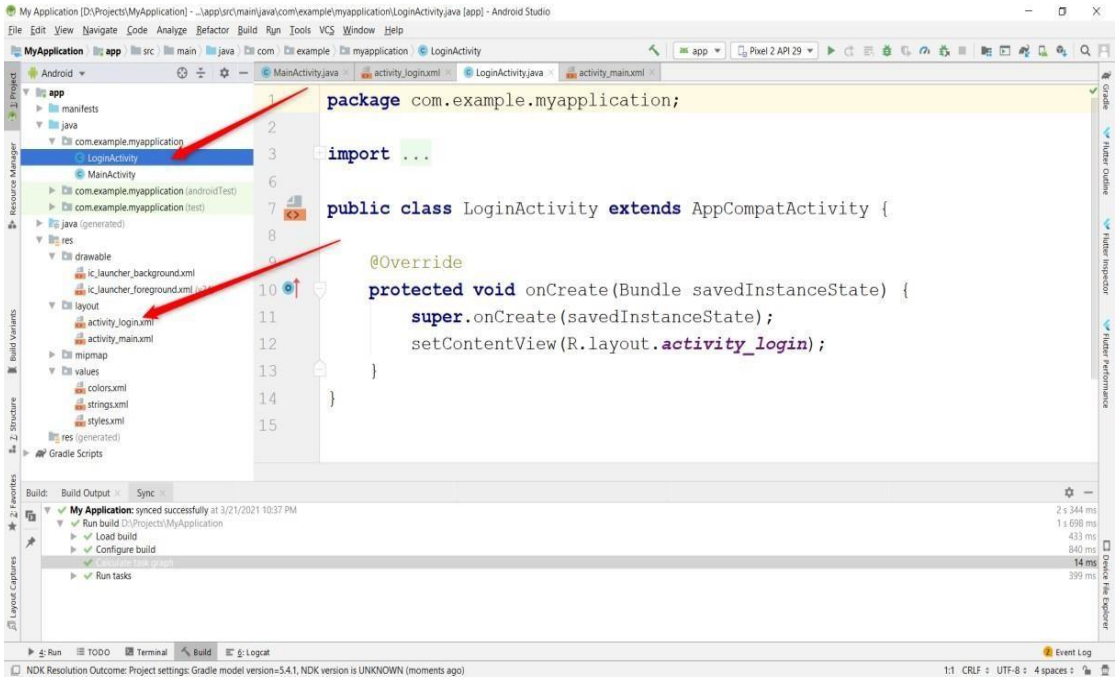
Right Click on Package New Activity Empty Activity



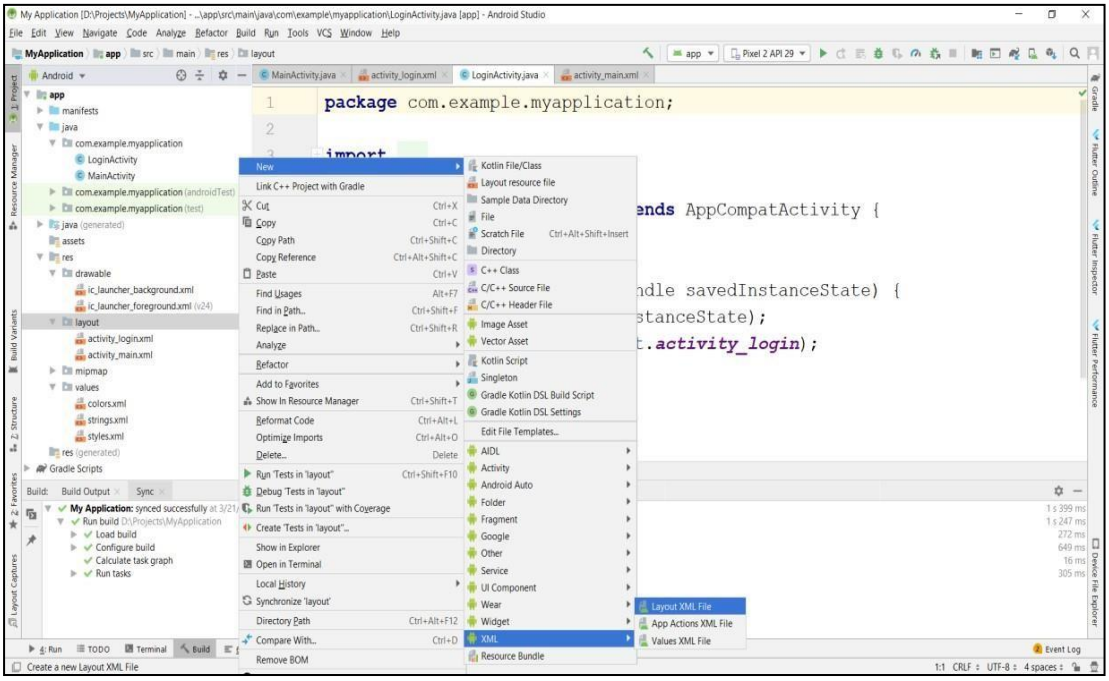
Enter Activity Name and Press Finish



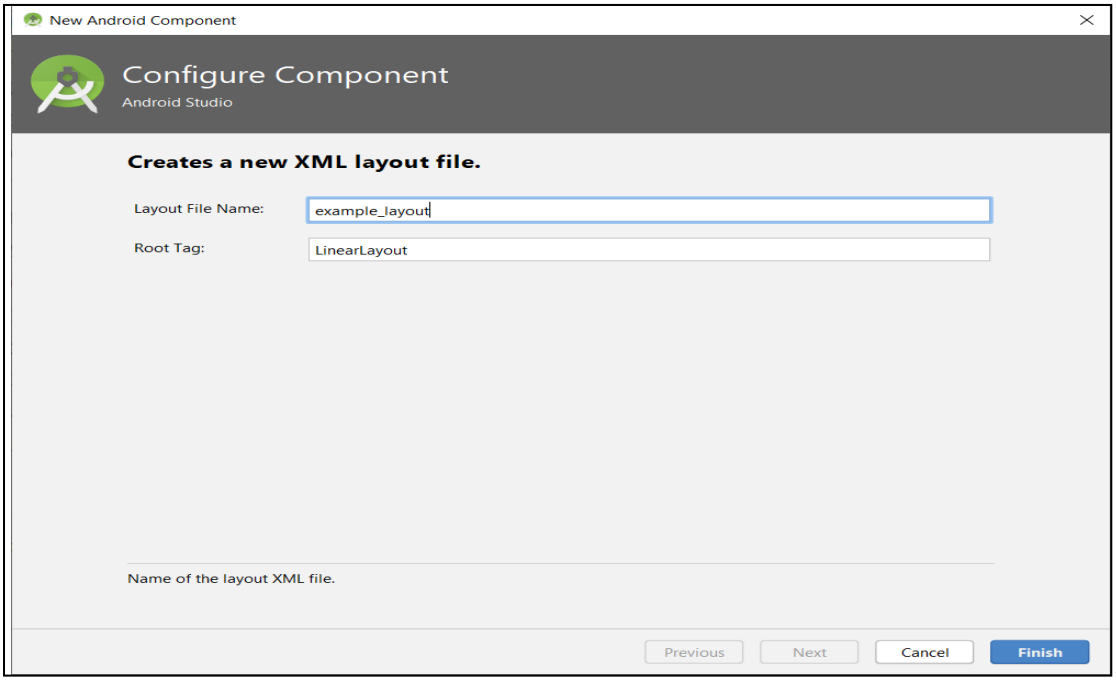
Creating a Layout in Android



Right Click on Layout Folder □ New □ XML□Layout XML File

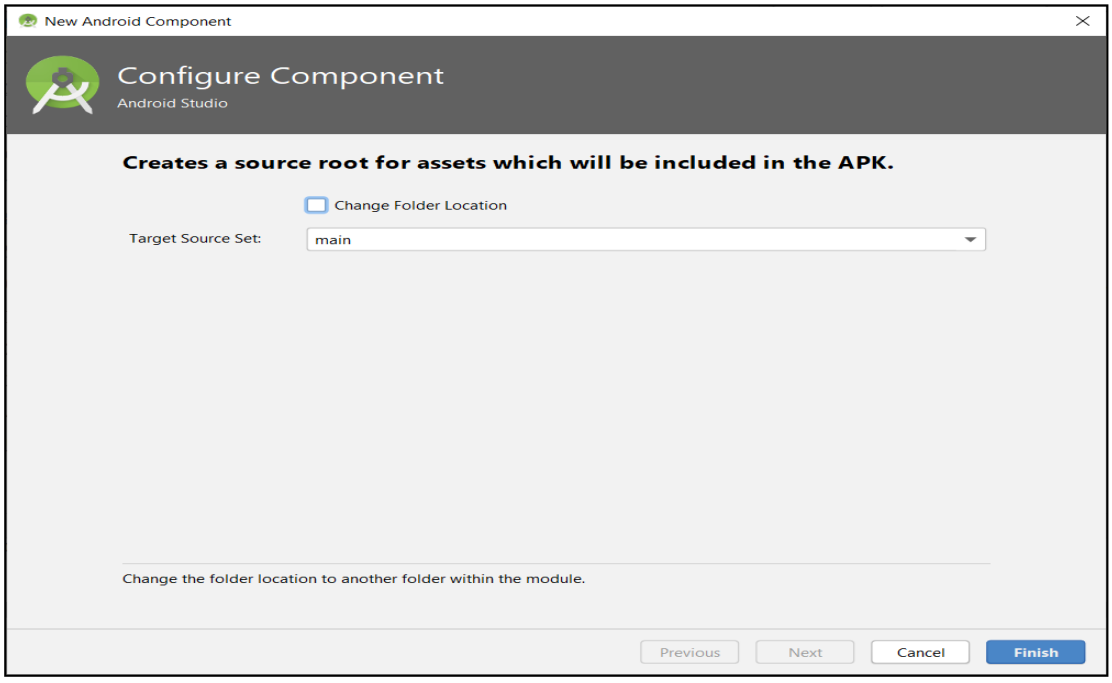


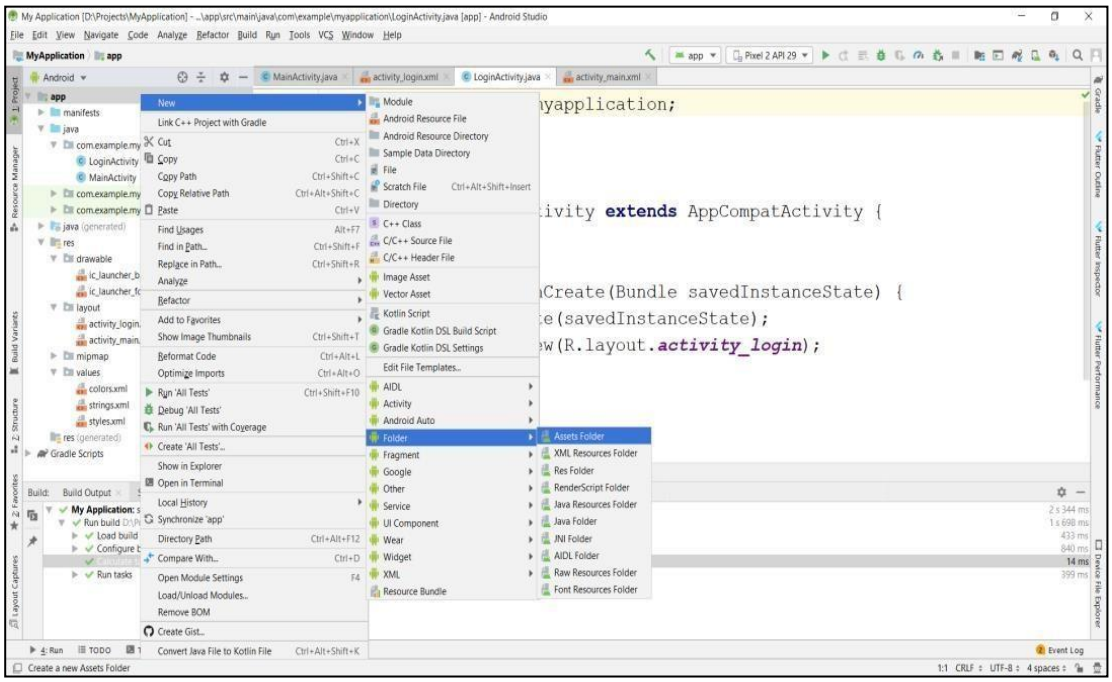
Enter xml file name and press Finish

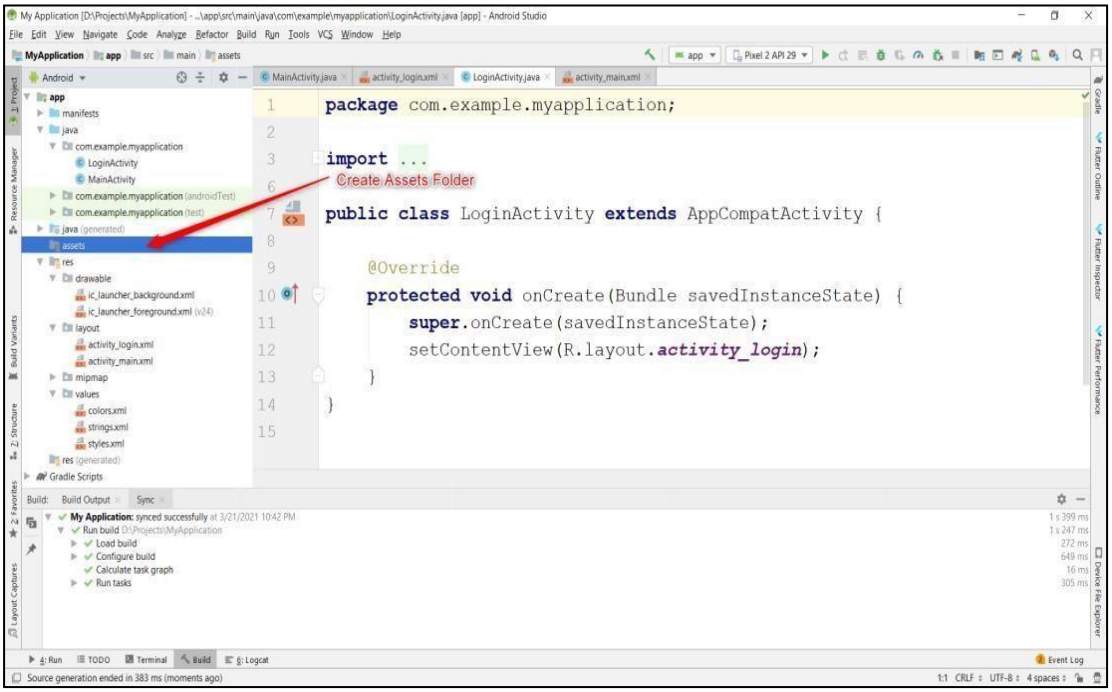


Creating Assets Folder in Android

Right Click on app folder □ New □ Folder □ Assets Folder □ Press Finish Button

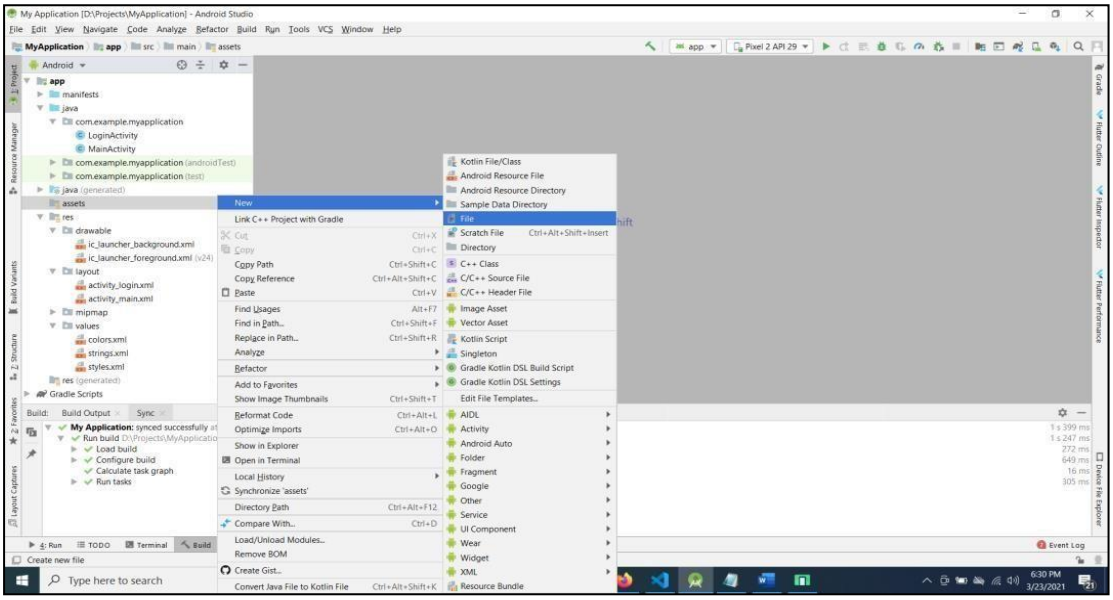



New
Module
Android Resource File
Android Resource Directory
Sample Data Directory
File
Scratch File
Directory
C++ Class
C/C++ Source File
C/C++ Header File
Image Asset
Vector Asset
Kotlin Script
Gradle Kotlin DSL Build Script
Gradle Kotlin DSL Settings
Edit File Templates...
AIDL
Activity
Android Auto
Folder
Fragment
Google
Other
Service
UI Component
Wear
Widget
XML
Resource Bundle
Assets Folder
XML Resources Folder
Res Folder
RenderScript Folder
Java Resources Folder
Java Folder
JNI Folder
AIDL Folder
Raw Resources Folder
Font Resources Folder


Create Assets Folder
package com.example.myapplication;
import ...
public class LoginActivity extends AppCompatActivity {
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_login);
}
}

Creating File in assets Folder:

Right Click on assets folder □ New □ File



Enter filename with extension (Eg: abc.xml)

3.2 Code Lines for the Application

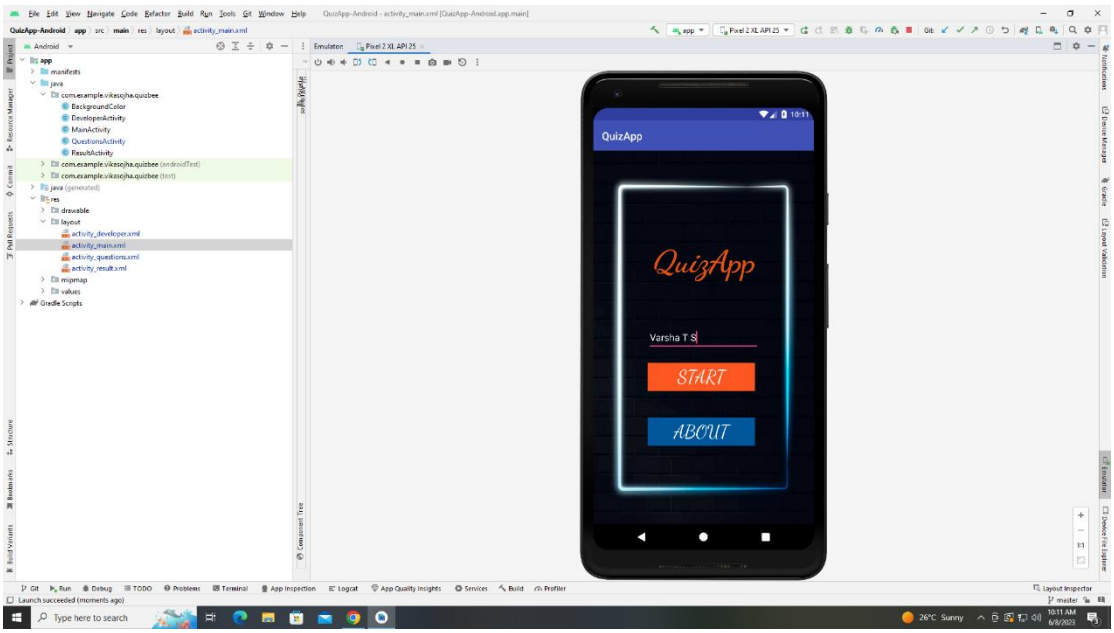
3.2.1 Activitymain.java File

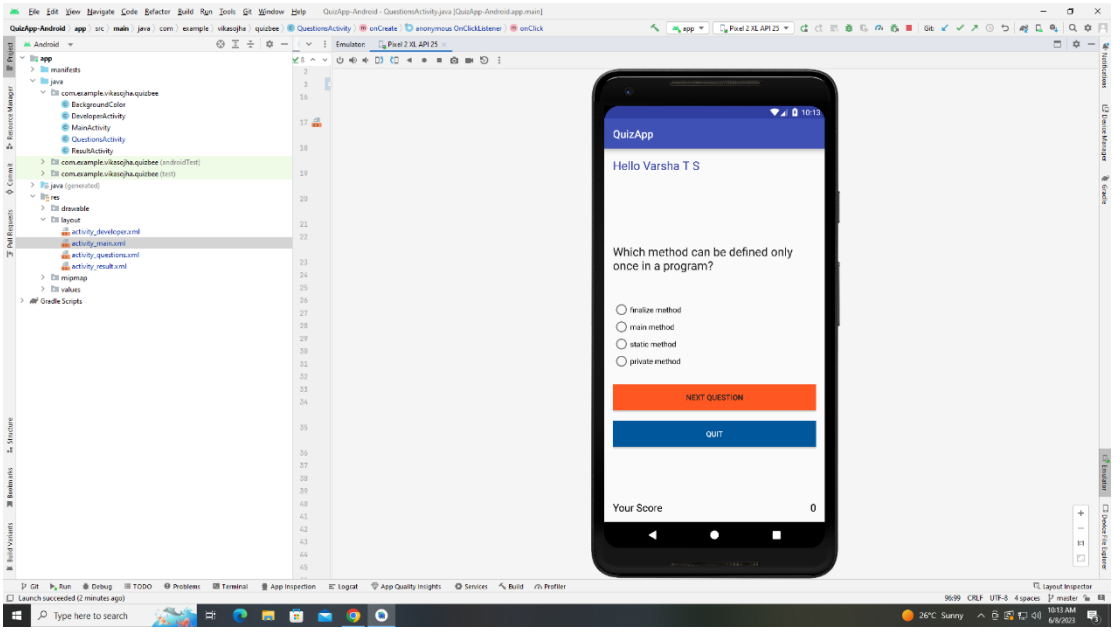
```
package com.example.varsha.quizbee;
import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
public class MainActivity extends AppCompatActivity {

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

        Button startbutton=(Button)findViewById(R.id.button);
        Button aboutbutton=(Button)findViewById(R.id.button2);
        final EditText nametext=(EditText)findViewById(R.id.editName);

            startbutton.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View v) {
                    String name=nametext.getText().toString();
                    Intent intent=new Intent(getApplicationContext(),QuestionsActivity.class);
                    intent.putExtra("myname",name);
                    startActivity(intent);
                }
            });

            aboutbutton.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View v) {
                Intent intent=new Intent(getApplicationContext(),DeveloperActivity.class);
                startActivity(intent);
            }
        });
    }
}
```

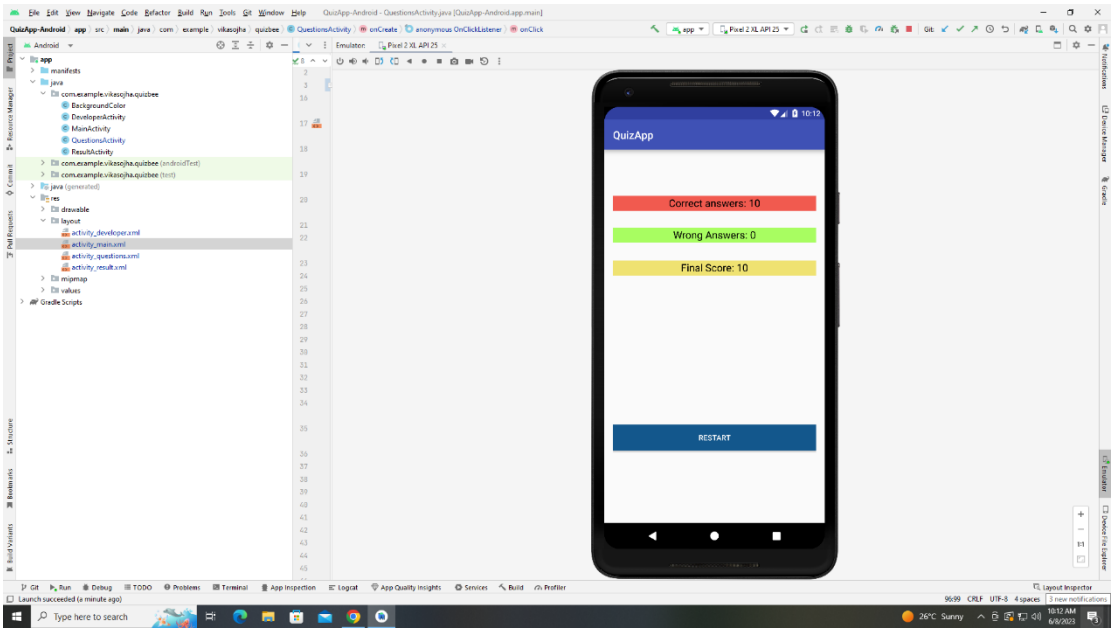
Chapter 4

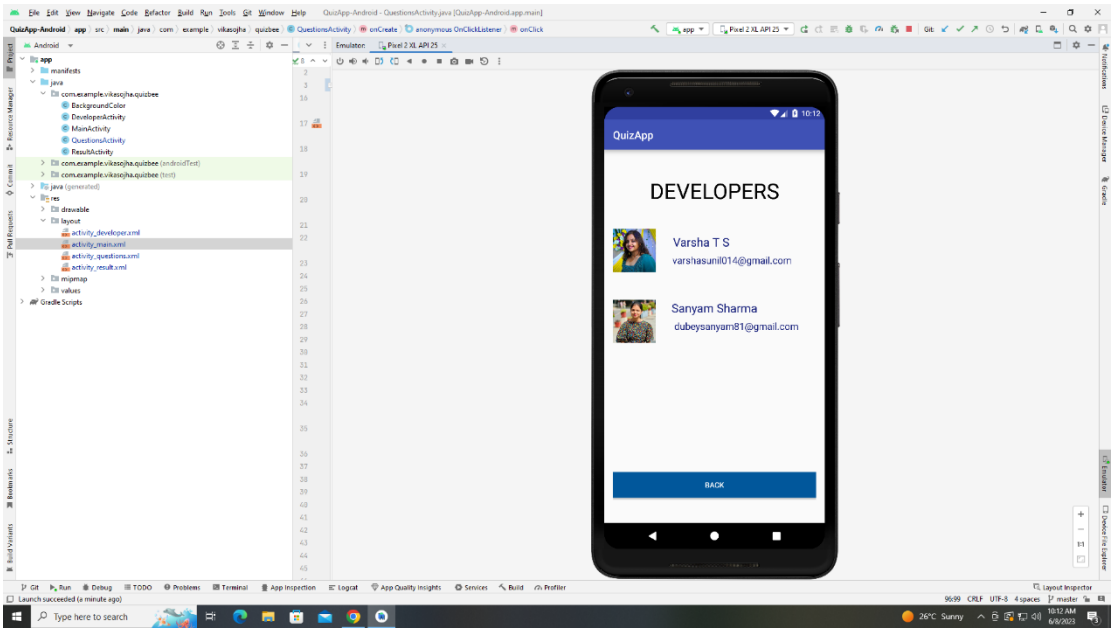
RESULTS

4.1 Sample Screenshots






QuizApp
Correct answers: 10
Wrong Answers: 0
Final Score: 10
RESTART


QuizApp
DEVELOPERS
Varsha T S
varshasunil014@gmail.com
Sanyam Sharma
dubeysanyam81@gmail.com
BACK

CONCLUSION

In conclusion, the development of a simple Android quiz application provides an engaging and interactive way for users to test their knowledge and learn new information. By incorporating various features and functionalities, such as multiple choice questions, scoring system, timed quizzes, and user-friendly interface, the application offers an enjoyable experience for both casual users and avid learners.

The quiz application serves as a valuable educational tool, enabling users to enhance their understanding in specific subject areas or broaden their general knowledge. It encourages active learning by presenting questions in a dynamic and interactive format, stimulating critical thinking and problem-solving skills.

Moreover, the Android platform offers a wide user base, ensuring that the quiz application reaches a large audience. With the increasing popularity of smartphones and the convenience of mobile applications, the Android quiz app becomes readily accessible to users anytime and anywhere.

Overall, the simple Android quiz application is an effective and engaging solution for users to challenge themselves, acquire new knowledge, and enjoy the process of learning. It combines the benefits of technology, education, and entertainment, making it a valuable addition to the world of mobile applications.

BIBLIOGRAPHY

[1] RESEARCH ARTICLE - Vaibhavi Balaji Kunale, International Journal of Engineering Research and Applications www.ijera.com

[2] REEARCH ARTICLE - Prof.P.P Bastawade, International Journal of Advanced Research in Computer and Communication Engineering www.ijarcce.com

[3] www.videolan.org

[4] www.wikipedia.com

[5] www.w3schools.com

[6] www.google.com

[7] www.youtube.com