***ASSIGNMENT NO.4***

* **Activity Class for Event Demonstration:**

1. ***Tools & Technologies Used***

* ***Android Studio***
* ***Java***

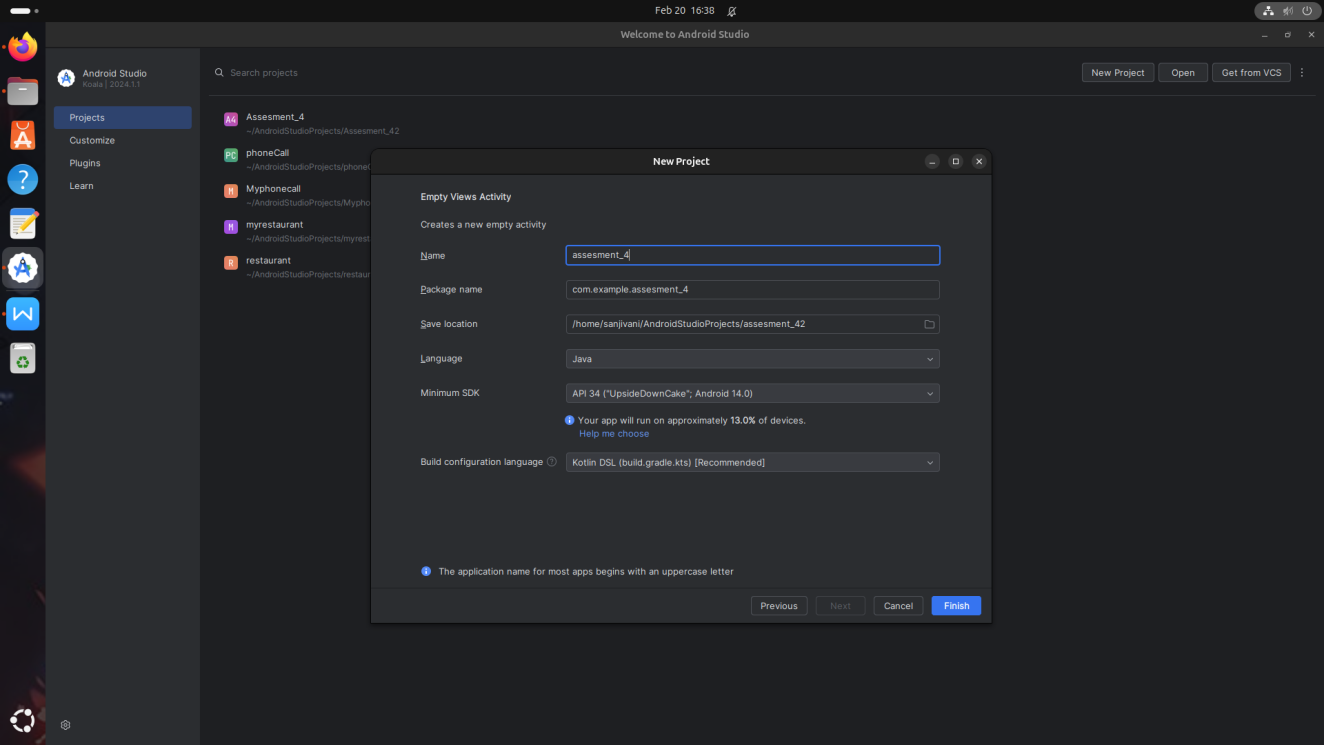
# Procedure & Steps

## Step 1: Create a New Project (AP) ⚫

Open Android Studio and create a new project.

* Set the project name and package name.
* Select the programming language (Java/Kotlin).

**Screenshot:**



## Step 2: Designing the UI

* Open activity\_main.xml and design the layout using XML.
* Add UI components such as TextView, EditText, Button, etc
* And Some Code Also.
* CODE FOR activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<!-- Button for Click Event -->

<!-- Button for Long Click Event -->

<!-- Button for Swipe Event -->

<!-- TextView to display event message -->

<Button

android:id="@+id/buttonClick"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Click Me"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.477"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.25" />

<Button

android:id="@+id/buttonLongClick"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Long Click Me"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.498"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.462" />

<Button

android:id="@+id/buttonSwipe"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Swipe Me"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.488"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.361" />

<TextView

android:id="@+id/textViewMessage"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Event Output"

android:textSize="20sp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.498"

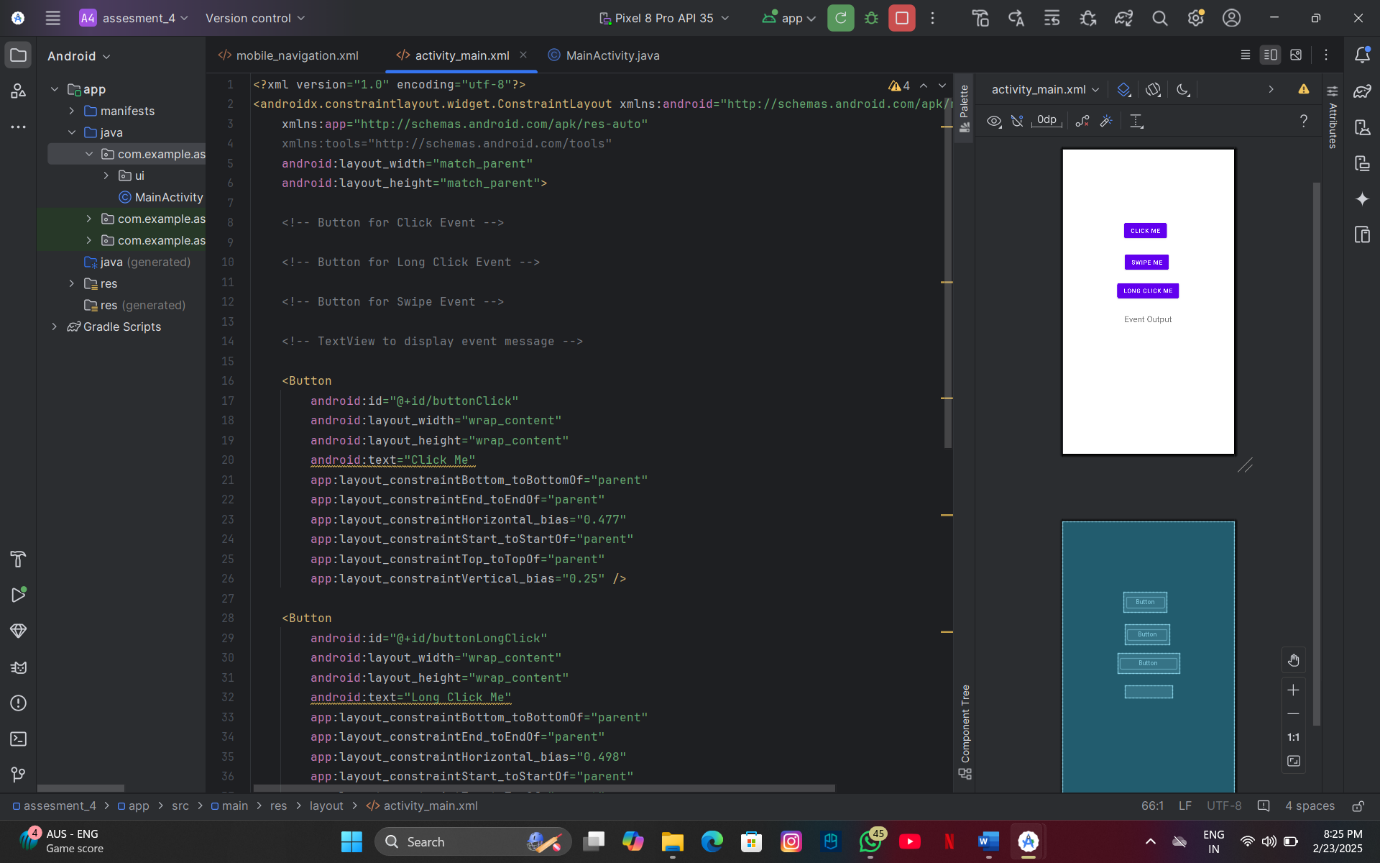
app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.559" />

</androidx.constraintlayout.widget.ConstraintLayout>

**Screenshot:**



## Step 3: Writing the Code

* Open MainActivity.java.
* Implement functionality such as button clicks, form validation, etc.
* Use necessary Android components like Intents, RecyclerView, Fragments, etc.

package com.example.assesment\_4;

import android.os.Bundle;

import android.view.GestureDetector;

import android.view.MotionEvent;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;

import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import com.example.assesment\_4.R;

public class MainActivity extends AppCompatActivity {

private TextView textViewMessage;

private Button buttonClick, buttonLongClick, buttonSwipe;

// Gesture Detector for swipe event

private GestureDetector gestureDetector;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

// Initialize views

textViewMessage = findViewById(R.id.textViewMessage);

buttonClick = findViewById(R.id.buttonClick);

buttonLongClick = findViewById(R.id.buttonLongClick);

buttonSwipe = findViewById(R.id.buttonSwipe);

// Button Click Event

buttonClick.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

textViewMessage.setText("Button Clicked!");

}

});

// Button Long Click Event

buttonLongClick.setOnLongClickListener(new View.OnLongClickListener() {

@Override

public boolean onLongClick(View v) {

textViewMessage.setText("Button Long Pressed!");

return true; // Returning true indicates the event is handled

}

});

// Initialize gesture detector for swipe event

gestureDetector = new GestureDetector(this, new GestureDetector.SimpleOnGestureListener() {

@Override

public boolean onFling(MotionEvent e1, MotionEvent e2, float velocityX, float velocityY) {

// Check if the swipe is left or right

if (e1.getX() < e2.getX()) {

textViewMessage.setText("Swipe Right Detected!");

} else if (e1.getX() > e2.getX()) {

textViewMessage.setText("Swipe Left Detected!");

}

return true;

}

});

// Set up touch listener for the Swipe button

buttonSwipe.setOnTouchListener(new View.OnTouchListener() {

@Override

public boolean onTouch(View v, MotionEvent event) {

return gestureDetector.onTouchEvent(event); // Pass the touch event to gesture detector

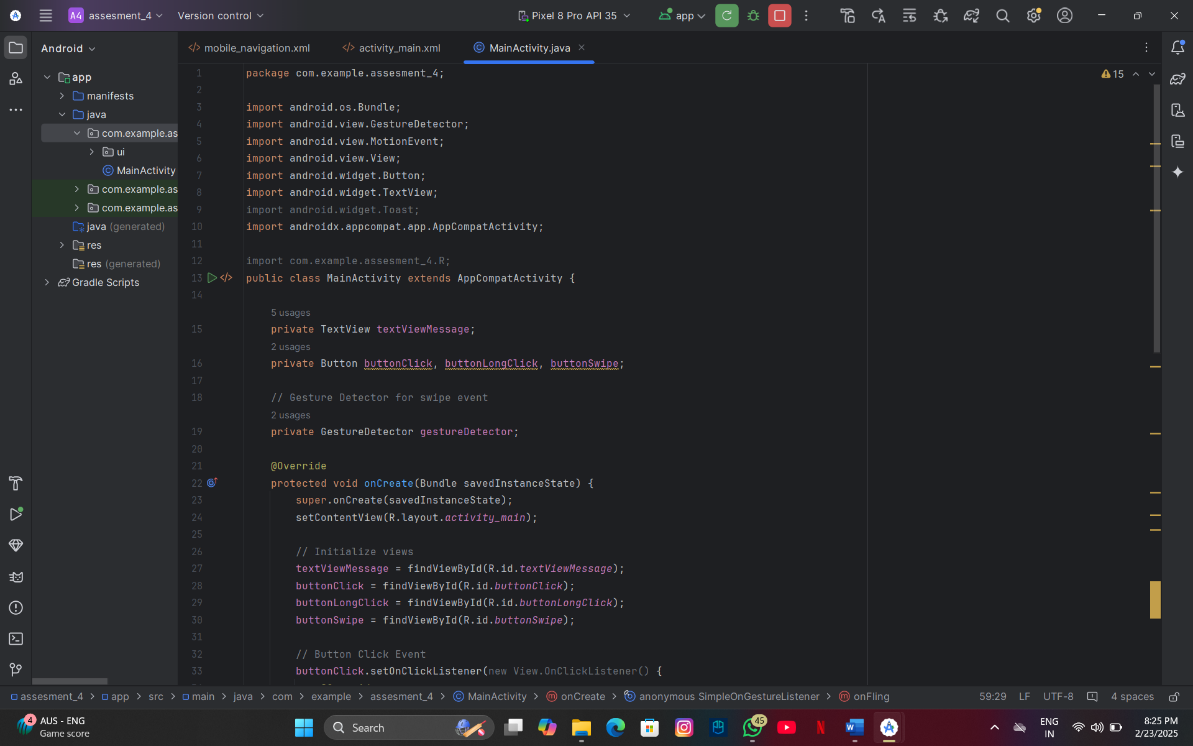
}

});

}

}

**Screenshot:**



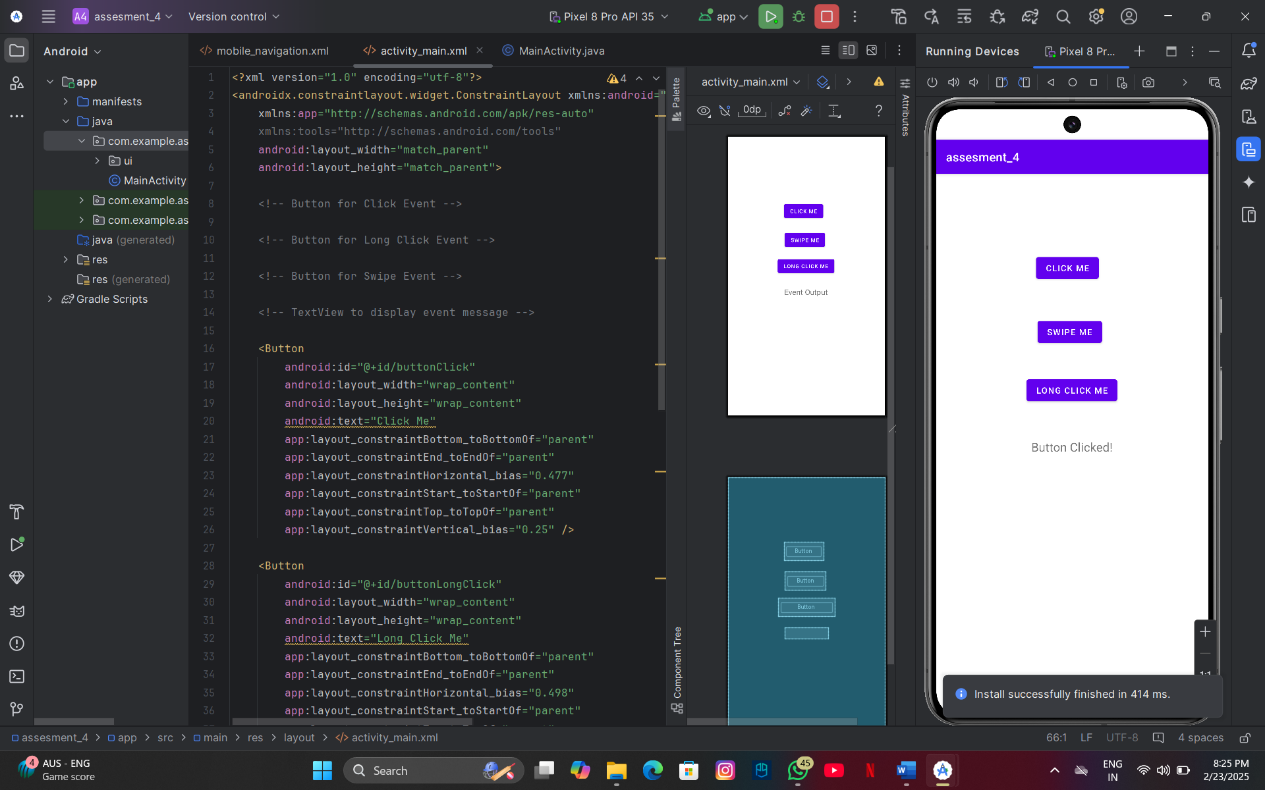
## Step 4: Running the Application on Emulator

* Click on the **Run** button in Android Studio.
* Select the emulator and launch the app.

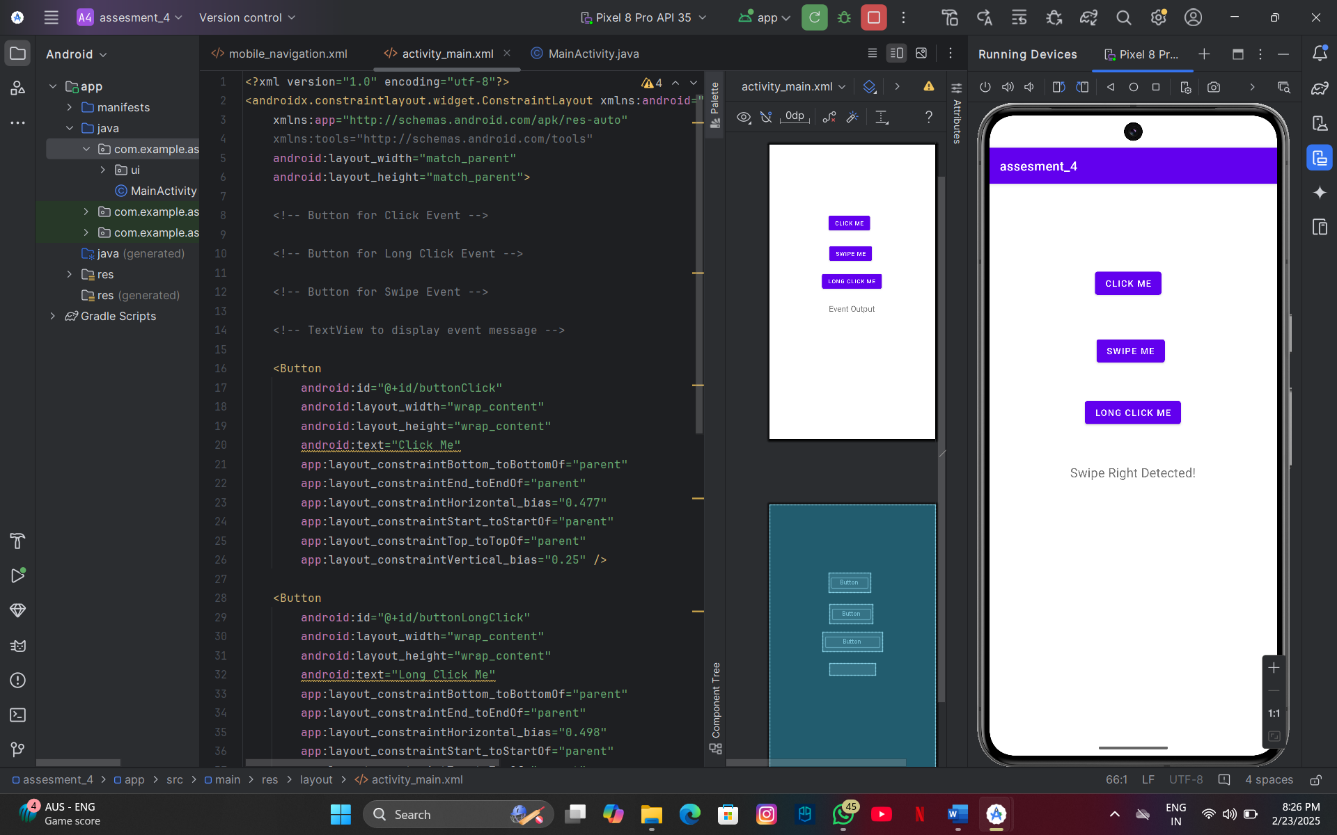
**Output**:

**Screenshot:**

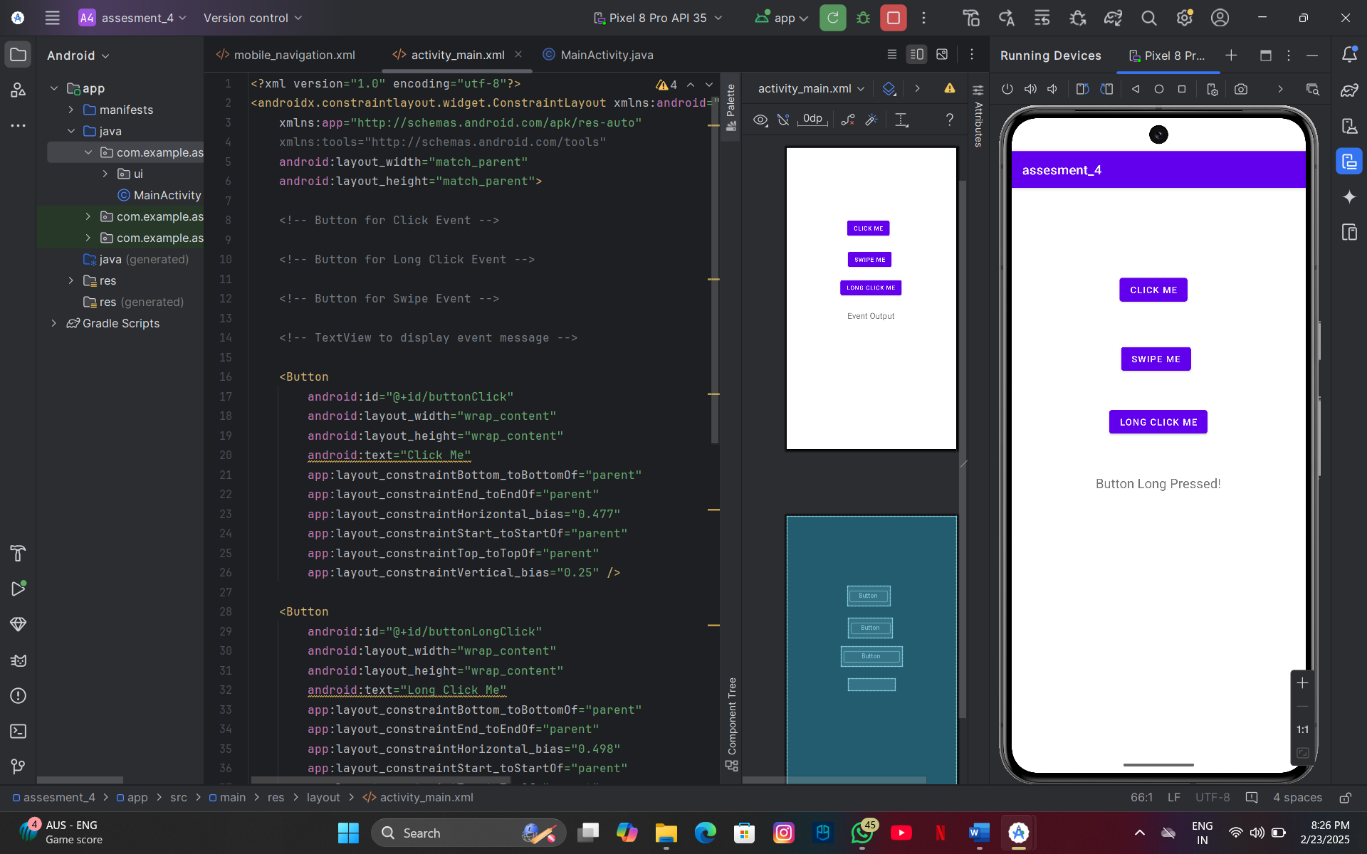
* **For button click event :**



* **For button swipe event:**



* **For button long press event:**



## step 5: Testing t& Output

* Test different functionalities of the app.
* Capture the output results.
  + The emulators works Correctly.
  + And all Works Good.

# Conclusion

This Android Studio project demonstrates how to use an activity class to handle user events like button clicks, long presses, and swipes. Each event triggers a specific action, such as displaying a message, showcasing basic event handling and interactivity in mobile apps. It serves as a foundation for creating more complex and responsive applications.

**Prasad Gaikwad**

**2124UCSM1024**

**CYBER SECURITY**