20CYS404 – Android Application Development  
End Sem

Aim:

Create a Free Hand Drawing App exactly with color palette

Code:

**Activity\_main.xml**  
<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical">

<!-- TextView for displaying name and roll number -->

<TextView

android:id="@+id/name\_rollno\_text"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Saravanan R [21074]"

android:textSize="18sp"

android:textStyle="bold"

android:gravity="center"

android:padding="16dp"

android:textColor="#000000" />

<FrameLayout

android:id="@+id/drawing\_frame"

android:layout\_width="match\_parent"

android:layout\_height="0dp"

android:layout\_weight="1"

android:background="#FFFFFF" />

<Spinner

android:id="@+id/brush\_size\_spinner"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:entries="@array/brush\_sizes" />

<!-- Horizontal layout for buttons -->

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal">

<!-- GridLayout for the color palette at the bottom -->

<GridLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:background="#cccccc"

android:columnCount="6"

android:gravity="center"

android:padding="8dp"

android:rowCount="2">

<!-- Color icons in grid format -->

<ImageView

android:id="@+id/colorBrown"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_margin="4dp"

android:background="#8B4513"

android:contentDescription="Brown Color" />

<ImageView

android:id="@+id/colorRed"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_margin="4dp"

android:background="#ff0000"

android:contentDescription="Red Color" />

<ImageView

android:id="@+id/colorOrange"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_margin="4dp"

android:background="#ffa500"

android:contentDescription="Orange Color" />

<ImageView

android:id="@+id/colorYellow"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_margin="4dp"

android:background="#ffff00"

android:contentDescription="Yellow Color" />

<ImageView

android:id="@+id/colorGreen"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_margin="4dp"

android:background="#008000"

android:contentDescription="Green Color" />

<ImageView

android:id="@+id/colorTeal"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_margin="4dp"

android:background="#008080"

android:contentDescription="Teal Color" />

<ImageView

android:id="@+id/colorBlue"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_margin="4dp"

android:background="#0000ff"

android:contentDescription="Blue Color" />

<ImageView

android:id="@+id/colorPurple"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_margin="4dp"

android:background="#800080"

android:contentDescription="Purple Color" />

<ImageView

android:id="@+id/colorPink"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_margin="4dp"

android:background="#FF69B4"

android:contentDescription="Pink Color" />

<ImageView

android:id="@+id/colorWhite"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_margin="4dp"

android:background="#ffffff"

android:contentDescription="White Color" />

<ImageView

android:id="@+id/colorGray"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_margin="4dp"

android:background="#808080"

android:contentDescription="Gray Color" />

<ImageView

android:id="@+id/colorBlack"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_margin="4dp"

android:background="#000000"

android:contentDescription="Black Color" />

</GridLayout>

<!-- Clear button -->

<Button

android:id="@+id/clear\_button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Clear" />

<!-- Color picker button -->

<Button

android:id="@+id/color\_picker\_button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Color Picker" />

</LinearLayout>

</LinearLayout>

**MainActivity.java**

package com.example.drawing;

import static com.example.drawing.R.id.clear\_button;

import static com.example.drawing.R.id.color\_picker\_button;

import android.annotation.SuppressLint;

import android.graphics.Bitmap;

import android.graphics.Canvas;

import android.graphics.Color;

import android.graphics.Paint;

import android.graphics.Path;

import android.graphics.PorterDuff;

import android.os.Bundle;

import android.util.AttributeSet;

import android.view.MotionEvent;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.Button;

import android.widget.FrameLayout;

import android.widget.Spinner;

import androidx.appcompat.app.AppCompatActivity;

import yuku.ambilwarna.AmbilWarnaDialog;

public class MainActivity extends AppCompatActivity {

private DrawingView drawingView;

private Spinner brushSizeSpinner;

private Button colorPickerButton, clearButton;

private int currentColor = Color.BLACK;

@SuppressLint("MissingInflatedId")

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

FrameLayout drawingFrame = findViewById(R.id.drawing\_frame);

drawingView = new DrawingView(this, null);

drawingFrame.addView(drawingView);

brushSizeSpinner = findViewById(R.id.brush\_size\_spinner);

colorPickerButton = findViewById(color\_picker\_button);

clearButton = findViewById(clear\_button);

// Set up the spinner for brush size selection

ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource(this,

R.array.brush\_sizes, android.R.layout.simple\_spinner\_item);

adapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

brushSizeSpinner.setAdapter(adapter);

// Set the brush size based on the selected item

brushSizeSpinner.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {

@Override

public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {

String selectedSize = parent.getItemAtPosition(position).toString();

switch (selectedSize) {

case "Small":

drawingView.setBrushSize(10);

break;

case "Medium":

drawingView.setBrushSize(30);

break;

case "Large":

drawingView.setBrushSize(50);

break;

}

}

@Override

public void onNothingSelected(AdapterView<?> parent) {

}

});

colorPickerButton.setOnClickListener(v -> openColorPicker());

clearButton.setOnClickListener(v -> drawingView.clearCanvas());

setUpColorPalette();

}

private void openColorPicker() {

AmbilWarnaDialog colorPicker = new AmbilWarnaDialog(this, currentColor, new AmbilWarnaDialog.OnAmbilWarnaListener() {

@Override

public void onOk(AmbilWarnaDialog dialog, int color) {

currentColor = color;

drawingView.setColor(currentColor);

}

@Override

public void onCancel(AmbilWarnaDialog dialog) {

}

});

colorPicker.show();

}

private void setUpColorPalette() {

findViewById(R.id.colorBrown).setOnClickListener(v -> drawingView.setColor(Color.parseColor("#8B4513")));

findViewById(R.id.colorRed).setOnClickListener(v -> drawingView.setColor(Color.RED));

findViewById(R.id.colorOrange).setOnClickListener(v -> drawingView.setColor(Color.parseColor("#FFA500")));

findViewById(R.id.colorYellow).setOnClickListener(v -> drawingView.setColor(Color.YELLOW));

findViewById(R.id.colorGreen).setOnClickListener(v -> drawingView.setColor(Color.parseColor("#008000")));

findViewById(R.id.colorTeal).setOnClickListener(v -> drawingView.setColor(Color.parseColor("#008080")));

findViewById(R.id.colorBlue).setOnClickListener(v -> drawingView.setColor(Color.BLUE));

findViewById(R.id.colorPurple).setOnClickListener(v -> drawingView.setColor(Color.parseColor("#800080")));

findViewById(R.id.colorPink).setOnClickListener(v -> drawingView.setColor(Color.parseColor("#FF69B4")));

findViewById(R.id.colorWhite).setOnClickListener(v -> drawingView.setColor(Color.WHITE));

findViewById(R.id.colorGray).setOnClickListener(v -> drawingView.setColor(Color.GRAY));

findViewById(R.id.colorBlack).setOnClickListener(v -> drawingView.setColor(Color.BLACK));

}

public class DrawingView extends View {

private Path drawPath;

private Paint drawPaint, canvasPaint;

private Canvas drawCanvas;

private Bitmap canvasBitmap;

public DrawingView(MainActivity context, AttributeSet attrs) {

super(context, attrs);

setupDrawing();

}

private void setupDrawing() {

drawPath = new Path();

drawPaint = new Paint();

drawPaint.setColor(currentColor);

drawPaint.setAntiAlias(true);

drawPaint.setStrokeWidth(10);

drawPaint.setStyle(Paint.Style.STROKE);

drawPaint.setStrokeJoin(Paint.Join.ROUND);

drawPaint.setStrokeCap(Paint.Cap.ROUND);

canvasPaint = new Paint(Paint.DITHER\_FLAG);

}

@Override

protected void onSizeChanged(int w, int h, int oldW, int oldH) {

super.onSizeChanged(w, h, oldW, oldH);

canvasBitmap = Bitmap.createBitmap(w, h, Bitmap.Config.ARGB\_8888);

drawCanvas = new Canvas(canvasBitmap);

}

@Override

protected void onDraw(Canvas canvas) {

canvas.drawBitmap(canvasBitmap, 0, 0, canvasPaint);

canvas.drawPath(drawPath, drawPaint);

}

@Override

public boolean onTouchEvent(MotionEvent event) {

float touchX = event.getX();

float touchY = event.getY();

switch (event.getAction()) {

case MotionEvent.ACTION\_DOWN:

drawPath.moveTo(touchX, touchY);

break;

case MotionEvent.ACTION\_MOVE:

drawPath.lineTo(touchX, touchY);

break;

case MotionEvent.ACTION\_UP:

drawCanvas.drawPath(drawPath, drawPaint);

drawPath.reset();

break;

default:

return false;

}

invalidate();

return true;

}

public void setColor(int newColor) {

drawPaint.setColor(newColor);

}

public void setBrushSize(float newSize) {

drawPaint.setStrokeWidth(newSize);

}

public void clearCanvas() {

drawCanvas.drawColor(Color.TRANSPARENT, PorterDuff.Mode.CLEAR);

invalidate();

}

}

}

AndriodManifest.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">  
  
 <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.Drawfull"  
 tools:targetApi="31">  
 <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>  
  
 <!-- Permission to write to external storage -->  
 <uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />  
</manifest>

Output:



