Paint

Software Documentation

Author: matiwa

Table of contents

Introduction

This software documentation includes: description of the application's operation, what is needed for use, algorithms used, interface description and source code description. This application serves as a drawing program.

Describing of the application’s operation

Drawing 1: The beginning of the application's operation [own study]

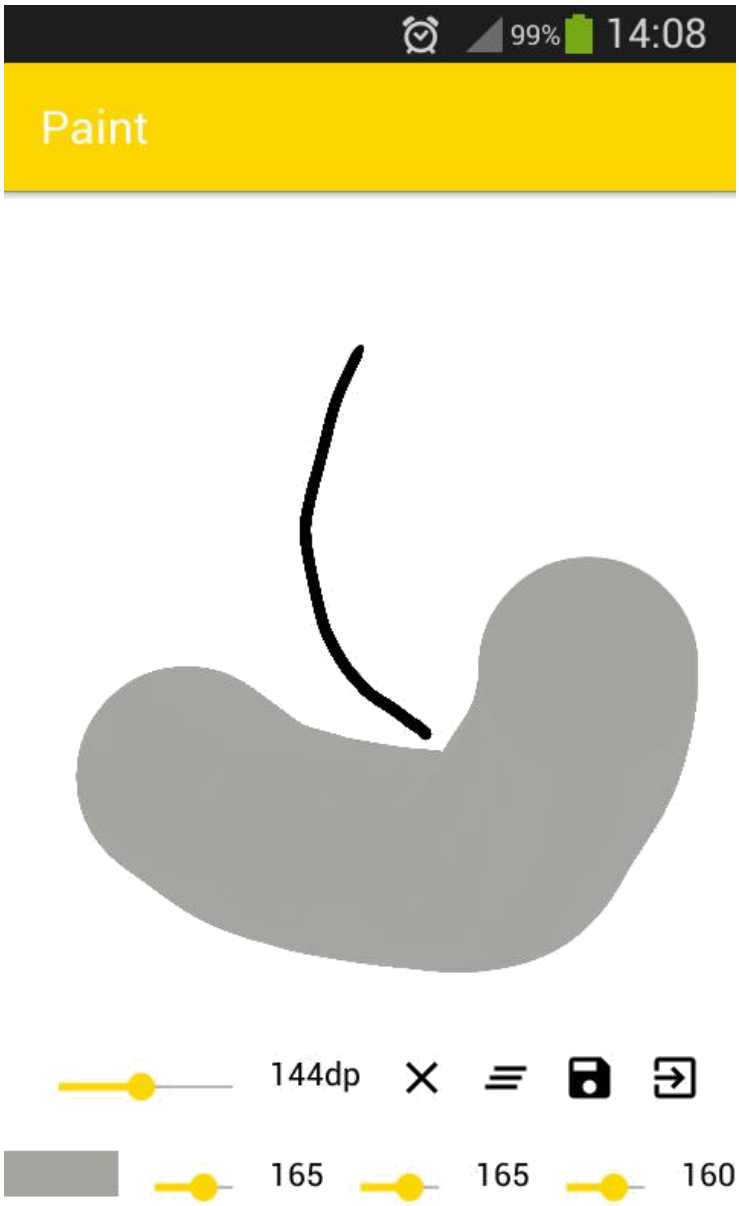
When the application starts, it is ready to draw. The default pen size is 5 dp. The color of the pen is determined by the RGB parameters, that is, the proportions of red, green and blue in the range 0 to 255. The default is black (R: 0, G: 0, B: 0).

Drawing 2: First steps in the application [own study]

It is enough for the user to touch the screen with a finger and thus create a drawing.

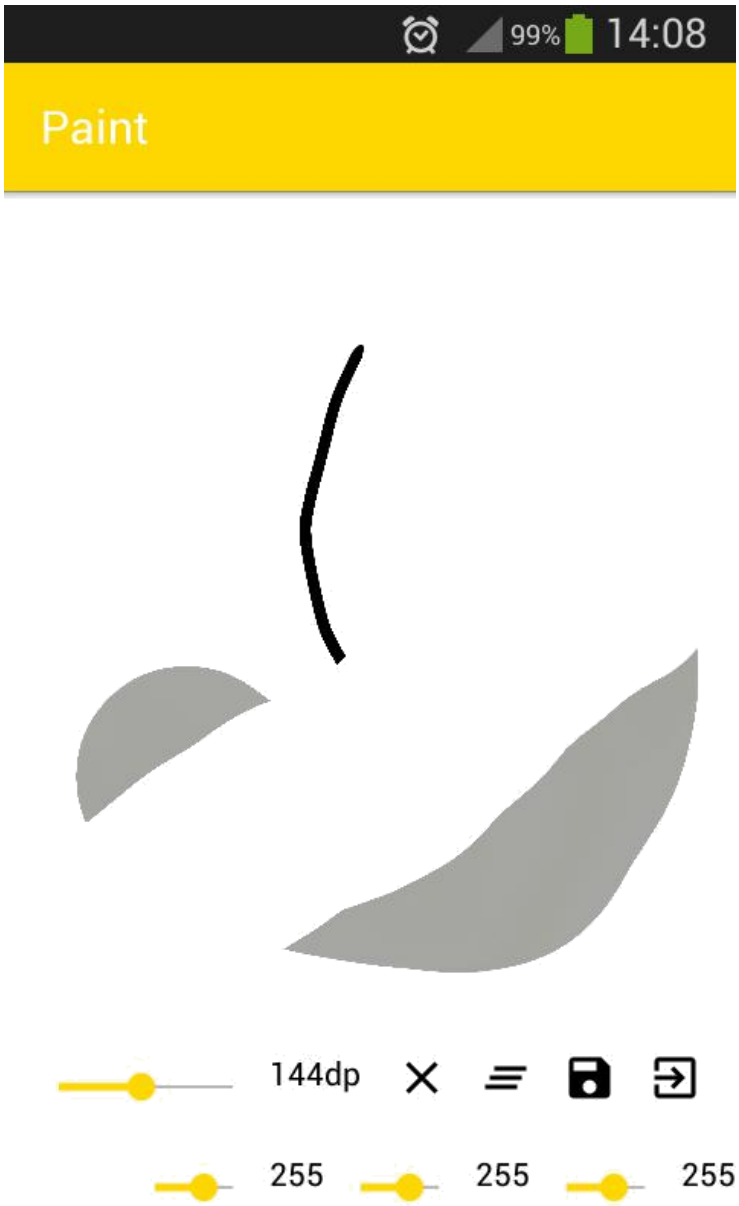
Drawing 3: Change the size and color of the pen [own study]

The user changes the color and size of the pen with sliders.



Drawing 4: The result of the changes [own study]

It is possible to use an eraser. The application selects a white color of the same size as the pen. The user can change it. The X button is used for this function.

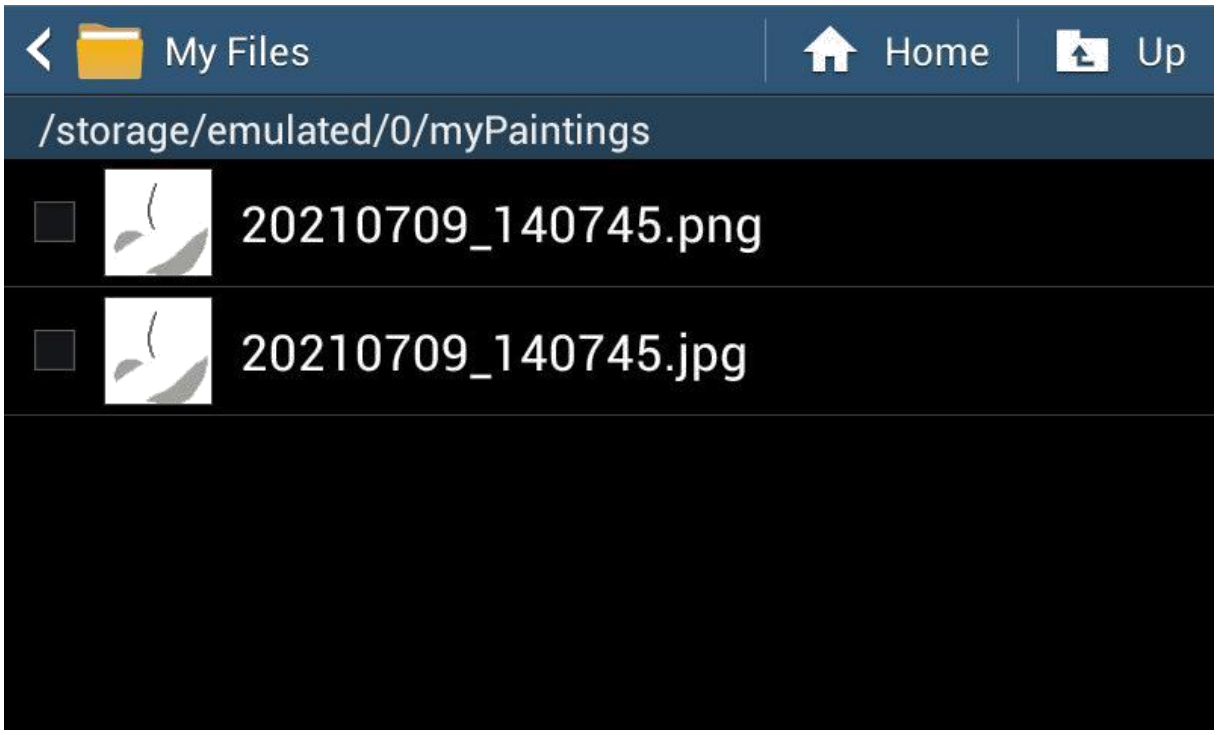


Drawing 5: Using the rubber [own study]

Another interesting option is the one that the user can use when he wants to start over without saving it. He has the ability to easily clean the entire working field by pressing the button with three lines. Pen settings do not reset!

Drawing 6: Cleared workspace [own study]

The last feature is work saving which is very useful. When starting the application, the date and time are remembered by the application. The name of the thesis file depends on it. It is characterized by the following year, month, day, hour, minute and second. The work is saved in two files in * .png and * .jpg format. After saving, a message about the success of the operation appears. The files are saved to the myPaintings folder. If it is missing, the application creates it automatically.



Drawing 7: Saved files in the myPaintings folder [own study]

What is needed for use?

The application requires installation on the Android operating system with a minimum API level Android 4.0 (IceCreamSandwich), i.e. on all mobile devices with this system.

Algorithm used

A repository was used to create the application, the links of which can be found under the youtube video: https://www.youtube.com/watch?v=xGrOHLk60q8&list=WL&index=4 . There are: https://github.com/zahid-ali-shah/SignatureView and https://github.com/Karumi/Dexter .

Interface description

Drawing 8: Graphical interface [own study]

The interface has the basic components available in the Android Studio development environment: ImageButton, View, SeekBar and TextView. SignatureView is an additional component here, possible to implement thanks to the repository https://github.com/zahid-ali-shah/SignatureView.

Source code description

The project was made in the Java programming language, in the Android Studio programming environment. All work was done on the Windows 10 operating system. The application's source code looks like this.

```
package com.example.paint;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;

import android.Manifest;
import android.graphics.Color;
import android.os.Bundle;

import com.karumi.dexter.Dexter;
import com.karumi.dexter.MultiplePermissionsReport;
import com.karumi.dexter.PermissionToken;
import com.karumi.dexter.listener.PermissionRequest;
import com.karumi.dexter.listener.multi.MultiplePermissionsListener;
import com.kyanogen.signatureview.SignatureView;

import android.os.Environment;
import android.view.View;
import android.widget.ImageButton;
import android.widget.SeekBar;
import android.widget.TextView;
import android.widget.Toast;
import android.graphics.Bitmap;

import java.io.ByteArrayOutputStream;
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.List;
import java.util.Locale;

public class MainActivity extends AppCompatActivity {
    int defaultColor;
    SignatureView signatureView;
    ImageButton imgClear,imgClearAll,imgSave,imgExit;
    View colorView;
    SeekBar seekBar,sred,sgreen,sblue;
    TextView txtPenSize,txtred,txtgreen,txtblue;

    private static String fileName,filename;
    File path=new
File(Environment.getExternalStorageDirectory().getAbsolutePath()+"/m
yPaintings");

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);

        initializeComponents();
        askPermission();
        SimpleDateFormat format=new
                SimpleDateFormat("yyyyMMdd_HHmmss",
Locale.getDefault());
        String date=format.format(new Date());
        fileName=path+"/"+date+".png";
        filename=path+"/"+date+".jpg";
        if(!path.exists()){
            path.mkdirs();
        }


defaultColor=ContextCompat.getColor(getApplicationContext(),R.color.
                colorPrimary);
    }

    private void openColorPicker() {


signatureView.setPenColor(Color.rgb(Integer.parseInt(txtred.getText(
                ).toString()),
                Integer.parseInt(txtgreen.getText().toString()),
                Integer.parseInt(txtblue.getText().toString())));


colorView.setBackgroundColor(Color.rgb(Integer.parseInt(txtred.getTe
xt().toString()),
                Integer.parseInt(txtgreen.getText().toString()),
                Integer.parseInt(txtblue.getText().toString())));
    }

    private void initializeComponents(){
        signatureView=findViewById(R.id.signature_view);
        seekBar=findViewById(R.id.penSize);
        txtPenSize=findViewById(R.id.txtPenSize);
        sred=findViewById(R.id.sred);
        txtred=findViewById(R.id.txtred);
        sgreen=findViewById(R.id.sgreen);
        txtgreen=findViewById(R.id.txtgreen);
        sblue=findViewById(R.id.sblue);
        txtblue=findViewById(R.id.txtblue);
        colorView=findViewById(R.id.colorView);
        openColorPicker();
        imgClear=findViewById(R.id.btnClear);
        imgClearAll=findViewById(R.id.btnClearAll);
        imgSave=findViewById(R.id.btnSave);
```

```
        imgExit=findViewById(R.id.btnExit);
        seekBar.setOnSeekBarChangeListener(new
        SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SeekBar seekBar, int
progress, boolean fromUser) {
                txtPenSize.setText(progress+"dp");
                signatureView.setPenSize(progress);
                seekBar.setMax(300);
            }

            @Override
            public void onStartTrackingTouch(SeekBar seekBar) {

            }

            @Override
            public void onStopTrackingTouch(SeekBar seekBar) {

            }
        });
        sred.setOnSeekBarChangeListener(new
SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SeekBar seekBar, int
progress, boolean fromUser) {
                txtred.setText(progress+"");
                openColorPicker();
                seekBar.setMax(255);
            }

            @Override
            public void onStartTrackingTouch(SeekBar seekBar) {

            }

            @Override
            public void onStopTrackingTouch(SeekBar seekBar) {

            }
        });
        sgreen.setOnSeekBarChangeListener(new
SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SeekBar seekBar, int
progress, boolean fromUser) {
                txtgreen.setText(progress+"");
                openColorPicker();
                seekBar.setMax(255);
```

```
            }

            @Override
            public void onStartTrackingTouch(SeekBar seekBar) {

            }

            @Override
            public void onStopTrackingTouch(SeekBar seekBar) {

            }
        });
        sblue.setOnSeekBarChangeListener(new
SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SeekBar seekBar, int
progress, boolean fromUser) {
                txtblue.setText(progress+"");
                openColorPicker();
                seekBar.setMax(255);
            }

            @Override
            public void onStartTrackingTouch(SeekBar seekBar) {

            }

            @Override
            public void onStopTrackingTouch(SeekBar seekBar) {

            }
        });
        imgClear.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick(View v) {
                txtred.setText("255");
                txtgreen.setText("255");
                txtblue.setText("255");


signatureView.setPenColor(Color.rgb(Integer.parseInt(txtred.getText(
).toString()),

Integer.parseInt(txtgreen.getText().toString()),

Integer.parseInt(txtblue.getText().toString())));
```

```
colorView.setBackgroundColor(Color.rgb(Integer.parseInt(txtred.getTe
xt().toString()),

Integer.parseInt(txtgreen.getText().toString()),

Integer.parseInt(txtblue.getText().toString())));
            }
        });
        imgClearAll.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                signatureView.clearCanvas();
            }
        });
        imgSave.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if(!signatureView.isBitmapEmpty()){
                    try {
                        saveImage();
                    } catch (IOException e) {
                        e.printStackTrace();


Toast.makeText(getApplicationContext(),"Couldn't
Save!",Toast.LENGTH_LONG).show();
                    }
                }
            }
        });
        imgExit.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                finish();
            }
        });
    }

    private void saveImage() throws IOException {
        File file=new File(fileName);

        Bitmap bitmap=signatureView.getSignatureBitmap();

        ByteArrayOutputStream bos=new ByteArrayOutputStream();
        bitmap.compress(Bitmap.CompressFormat.PNG,0,bos);
        byte[]bitmapData=bos.toByteArray();

        FileOutputStream fos=new FileOutputStream(file);
```

```
        fos.write(bitmapData);
        fos.flush();
        fos.close();

        //jpg
        File file2=new File(filename);
        Bitmap bitmap2=signatureView.getSignatureBitmap();
        bitmap2.compress(Bitmap.CompressFormat.JPEG,0,bos);
        FileOutputStream fos2=new FileOutputStream(file2);
        fos2.write(bitmapData);
        fos2.flush();
        fos2.close();

        Toast.makeText(getApplicationContext(),"Painting
Saved!",Toast.LENGTH_LONG).show();
    }

    private void askPermission(){


Dexter.withContext(this).withPermissions(Manifest.permission.READ_EX
TERNAL_STORAGE,

Manifest.permission.WRITE_EXTERNAL_STORAGE).withListener(new
MultiplePermissionsListener() {
                    @Override
                    public void
onPermissionsChecked(MultiplePermissionsReport
multiplePermissionsReport) {

Toast.makeText(getApplicationContext(),"Granted!",Toast.LENGTH_LONG)
.show();
                    }

                    @Override
                    public void
onPermissionRationaleShouldBeShown(List<PermissionRequest> list,

PermissionToken permissionToken) {
                        permissionToken.continuePermissionRequest();
                    }
                }).check();
    }
}
```

Listing 1: The source code of MainActivity.java file [own study]

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

```
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"
tools:context=".MainActivity">

<LinearLayout
    android:id="@+id/optiondHolder"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:gravity="center"
    android:orientation="vertical">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="4dp"
        android:layout_weight="1"
        android:orientation="horizontal">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="start"
            android:layout_margin="4dp"
            android:layout_weight="1">

            <SeekBar
                android:id="@+id/penSize"
                android:layout_width="150dp"
                android:layout_height="wrap_content"
                android:layout_weight="1"></SeekBar>

            <TextView
                android:id="@+id/txtPenSize"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="5dp"
                android:textColor="#000000"></TextView>

        </LinearLayout>

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="end"
            android:layout_margin="4dp"
```

```
        android:layout_weight="1">

        <ImageButton
            android:id="@+id/btnClear"
            android:layout_width="8dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@color/white"
            android:scaleType="center"
            android:src="@drawable/ic_clear" />

        <ImageButton
            android:id="@+id/btnClearAll"
            android:layout_width="8dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@color/white"
            android:scaleType="center"
            android:src="@drawable/ic_clear_all" />

        <ImageButton
            android:id="@+id/btnSave"
            android:layout_width="8dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@color/white"
            android:scaleType="center"
            android:src="@drawable/ic_save" />

        <ImageButton
            android:id="@+id/btnExit"
            android:layout_width="8dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@color/white"
            android:scaleType="center"
            android:src="@drawable/ic_exit" />
    </LinearLayout>
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal">

    <View
        android:id="@+id/colorView"
        android:layout_width="50dp"
        android:layout_height="20dp"></View>
```

```
<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <LinearLayout
        android:layout_width="90dp"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <SeekBar
            android:id="@+id/sred"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="5"></SeekBar>

        <TextView
            android:id="@+id/txtred"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="0"
            android:textColor="#000000"></TextView>
    </LinearLayout>

    <LinearLayout
        android:layout_width="90dp"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <SeekBar
            android:id="@+id/sgreen"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="5"></SeekBar>

        <TextView
            android:id="@+id/txtgreen"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="0"
            android:textColor="#000000"></TextView>
    </LinearLayout>

    <LinearLayout
        android:layout_width="90dp"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
```

```
                    <SeekBar
                        android:id="@+id/sblue"
                        android:layout_width="0dp"
                        android:layout_height="wrap_content"
                        android:layout_weight="5"></SeekBar>

                    <TextView
                        android:id="@+id/txtblue"
                        android:layout_width="wrap_content"
                        android:layout_height="wrap_content"
                        android:text="0"
                        android:textColor="#000000"></TextView>
                </LinearLayout>
            </LinearLayout>
        </LinearLayout>
    </LinearLayout>

    <com.kyanogen.signatureview.SignatureView
        android:id="@+id/signature_view"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_above="@+id/optiondHolder"
        app:backgroundColor="#ffffff"
        app:enableSignature="true"
        app:penColor="#000000"
        app:penSize="5dp" />

</RelativeLayout>
```

Listing 2: The source code of activity_main.xml file [own study]

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

<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.paint">

    <uses-permission

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

android:name="android.permission.WRITE_EXTERNAL_STORAGE" />

    <application

        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
```

```
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">

        <activity android:name=".MainActivity">
            <intent-filter>

                <action android:name="android.intent.action.MAIN" />

                <category

android:name="android.intent.category.LAUNCHER" />
            </intent-filter>

        </activity>
    </application>
</manifest>
```

Listing 3: The source code of AndroidMainfest.xml file [own study]

```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 31
    buildToolsVersion "31.0.0"
    defaultConfig {
        applicationId "com.example.paint"
        minSdkVersion 14
        targetSdkVersion 31
        versionCode 1
        versionName "1.0"
        testInstrumentationRunner
"androidx.test.runner.AndroidJUnitRunner"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android-
optimize.txt'), 'proguard-rules.pro'
        }
    }
}

dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'androidx.appcompat:appcompat:1.0.2'
    implementation'androidx.constraintlayout:constraintlayout:1.1.3'

    implementation 'com.kyanogen.signatureview:signature-view:1.2'
```

```
    implementation 'com.karumi:dexter:6.2.3'

    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'androidx.test.ext:junit:1.1.0'
    androidTestImplementation 'androidx.test.espresso:espresso-
core:3.1.1'
}
```

Listing 4: The source code of build.gradle file [own study]

List of drawings

List of listings