```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">

        <edu.illinois.cs498.dots.DotsView
            android:id="@+id/dots_view"
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:background="@color/oldPaper"
            android:layout_weight="1"
            android:layout_gravity="fill_vertical"/>

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

            <Button
                android:id="@+id/color_selection_button"
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:layout_gravity="fill_horizontal"
                android:text="Color"/>

            <Button
                android:id="@+id/width_selection_button"
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:layout_gravity="fill_horizontal"
                android:text="Width"/>

        </LinearLayout>

</LinearLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <group android:checkableBehavior="single">
        <item android:id="@+id/black_menu_item"
            android:title="Black"
            android:checked="true"/>
        <item android:id="@+id/red_menu_item"
            android:title="Red"
            android:checked="false"/>
        <item android:id="@+id/blue_menu_item"
            android:title="Blue"
            android:checked="false"/>
        <item android:id="@+id/green_menu_item"
            android:title="Green"
            android:checked="false"/>
    </group>
</menu>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <group android:checkableBehavior="single">
        <item android:id="@+id/small_menu_item"
            android:title="Small"
            android:checked="true"/>
        <item android:id="@+id/medium_menu_item"
            android:title="Medium"
            android:checked="false"/>
        <item android:id="@+id/large_menu_item"
            android:title="Large"
            android:checked="false"/>
        <item android:id="@+id/area_menu_item"
            android:title="Area"
            android:checked="false"/>
    </group>
</menu>
```

```java
package edu.illinois.cs498.dots;

import android.app.Activity;
import android.graphics.Color;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.PopupMenu;

/**
 * Created by Administrator on 2/16/2016.
 */
public class ColorSelectionMenu implements View.OnClickListener, PopupMenu.
OnMenuItemClickListener {
    private Activity activity;
    private Button cSelectionButton;
    private PopupMenu cSelectionMenu;

    public ColorSelectionMenu(Activity activity) {
        this.activity = activity;
        cSelectionButton = (Button) activity.findViewById(R.id.
color_selection_button);
        cSelectionButton.setOnClickListener(this);

        cSelectionMenu = new PopupMenu(activity, cSelectionButton);
        MenuInflater inflater = cSelectionMenu.getMenuInflater();
        inflater.inflate(R.menu.color_selection_menu, cSelectionMenu.getMenu
());
        cSelectionMenu.setOnMenuItemClickListener(this);

    }

    public void onClick(View view) {
        int id = view.getId();
        switch (id) {
            case R.id.color_selection_button:
                cSelectionMenu.show();
                break;
            default:
                break;
        }
    }

    public boolean onMenuItemClick(MenuItem item) {
        item.setChecked(true);
        DotsView dots = (DotsView) activity.findViewById(R.id.dots_view);
        switch (item.getItemId()) {
            case R.id.black_menu_item:
                dots.setColor(Color.BLACK);
                cSelectionButton.setText("BLACK");
                break;
            case R.id.red_menu_item:
                dots.setColor(Color.RED);
                cSelectionButton.setText("RED");
                break;
            case R.id.green_menu_item:
                dots.setColor(Color.GREEN);
```

```
                    cSelectionButton.setText("GREEN");
                    break;
                case R.id.blue_menu_item:
                    dots.setColor(Color.BLUE);
                    cSelectionButton.setText("BLUE");
                    break;
            }
            return true;
        }
    }
```

```java
package edu.illinois.cs498.dots;

import android.app.Activity;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.PopupMenu;

/**
 * Created by Administrator on 2/16/2016.
 */
public class WidthSelectionMenu implements View.OnClickListener, PopupMenu.
OnMenuItemClickListener {
    private Activity activity;
    private Button wSelectionButton;
    private PopupMenu wSelectionMenu;

    public WidthSelectionMenu(Activity activity) {
        this.activity = activity;
        wSelectionButton = (Button) activity.findViewById(R.id.
width_selection_button);
        wSelectionButton.setOnClickListener(this);

        wSelectionMenu = new PopupMenu(activity, wSelectionButton);
        MenuInflater inflater = wSelectionMenu.getMenuInflater();
        inflater.inflate(R.menu.width_selection_menu, wSelectionMenu.getMenu
());
        wSelectionMenu.setOnMenuItemClickListener(this);

    }

    public void onClick(View view) {
        int id = view.getId();
        switch (id) {
            case R.id.width_selection_button:
                wSelectionMenu.show();
                break;
        }
    }

    public boolean onMenuItemClick(MenuItem item) {
        item.setChecked(true);
        DotsView dots = (DotsView) activity.findViewById(R.id.dots_view);
        switch (item.getItemId()) {
            case R.id.small_menu_item:
                dots.setDotRadius(DotsView.SMALL_RADIUS);
                wSelectionButton.setText("SMALL");
                break;
            case R.id.medium_menu_item:
                dots.setDotRadius(DotsView.MEDIUM_RADIUS);
                wSelectionButton.setText("MEDIUM");
                break;
            case R.id.large_menu_item:
                dots.setDotRadius(DotsView.LARGE_RADIUS);
                wSelectionButton.setText("LARGE");
                break;
            case R.id.area_menu_item:
```

```java
                    dots.setDotRadius(DotsView.AREA_RADIUS);
                    wSelectionButton.setText("AREA");
                    break;
                default:
                    break;
            }
            return true;
        }
    }
```

```java
package edu.illinois.cs498.dots;

import android.content.Context;
import android.graphics.Bitmap;
import android.graphics.Canvas;
import android.graphics.Paint;
import android.graphics.Point;
import android.util.AttributeSet;
import android.util.Log;
import android.view.MotionEvent;
import android.view.View;
import java.util.HashMap;


/**
 * Created by Administrator on 2/12/2016.
 */
public class DotsView extends View implements View.OnTouchListener {
    public static int SMALL_RADIUS = 10;
    public static int MEDIUM_RADIUS = 20;
    public static int LARGE_RADIUS = 40;
    public static int AREA_RADIUS = 0;

    private Paint  mPaint;
    private Bitmap mBitmap;
    private Canvas mCanvas;
    private int dotRadius;
    private HashMap pointerMap;


    public DotsView(Context context) {
        super(context);
        initDotsView();
    }

    public DotsView(Context context, AttributeSet attrs) {
        super(context, attrs);
        initDotsView();
    }

    public DotsView(Context context, AttributeSet attrs, int defStyleAttr) {
        super(context, attrs, defStyleAttr);
        initDotsView();
    }

    public DotsView(Context context, AttributeSet attrs, int defStyleAttr,
int defStyleRes) {
        super(context, attrs, defStyleAttr, defStyleRes);
        initDotsView();
    }

    private void initDotsView() {
        mPaint = new Paint();
        mPaint.setStrokeCap(Paint.Cap.ROUND);
        dotRadius = SMALL_RADIUS;
        pointerMap = new HashMap();
        setOnTouchListener(this);
    }
```

```java
    @Override
    protected void onSizeChanged(int w, int h, int oldw, int oldh) {
        super.onSizeChanged(w, h, oldw, oldh);
        mBitmap = Bitmap.createBitmap(w, h, Bitmap.Config.ARGB_8888);
        mCanvas = new Canvas(mBitmap);
    }

    public void setDotRadius(int r) {
        dotRadius = r;
    }

    public void setColor(int c) {
        mPaint.setColor(c);
    }

    @Override
    protected void onDraw(Canvas canvas) {
        canvas.drawBitmap(mBitmap, 0, 0, mPaint);
    }

    public boolean onTouch(View v, MotionEvent event) {
        // Log.d("DEBUG", "Receiving touch event");
        int action = event.getActionMasked();
        int index = event.getActionIndex();
        int id = event.getPointerId(index);
        float x = event.getX(index);
        float y = event.getY(index);

        switch (action) {
            case MotionEvent.ACTION_DOWN:
            case MotionEvent.ACTION_POINTER_DOWN:
                Point p = new Point((int)x, (int)y);
                pointerMap.put(id, p);
                break;

            case MotionEvent.ACTION_MOVE:
                for (int i=0; i<event.getPointerCount(); ++i) {
                    id = event.getPointerId(i);
                    x = event.getX(i);
                    y = event.getY(i);
                    Point last = (Point) pointerMap.get(id);
                    if (last != null) {
                        if (dotRadius == AREA_RADIUS) {
                            mPaint.setStrokeWidth((float) event.getSize(i) *
1000);
                        } else {
                            mPaint.setStrokeWidth((float) dotRadius);
                        }
                        mCanvas.drawLine(last.x, last.y, x, y, mPaint);
                    }
                    pointerMap.put(id, new Point((int) x, (int) y));
                }
                invalidate();
                break;

            case MotionEvent.ACTION_UP:
```

```
            case MotionEvent.ACTION_POINTER_UP:
            case MotionEvent.ACTION_CANCEL:
                pointerMap.remove(id);
                break;
            default:
                break;

        }
        return true;
    }
}
```

```java
package edu.illinois.cs498.dots;


import android.app.Activity;
import android.graphics.Color;
import android.hardware.camera2.params.ColorSpaceTransform;
import android.os.Bundle;

public class MainActivity extends Activity {

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

        WidthSelectionMenu wMenu = new WidthSelectionMenu(this);
        ColorSelectionMenu cMenu = new ColorSelectionMenu(this);
    }
}
```