Queens University Belfast

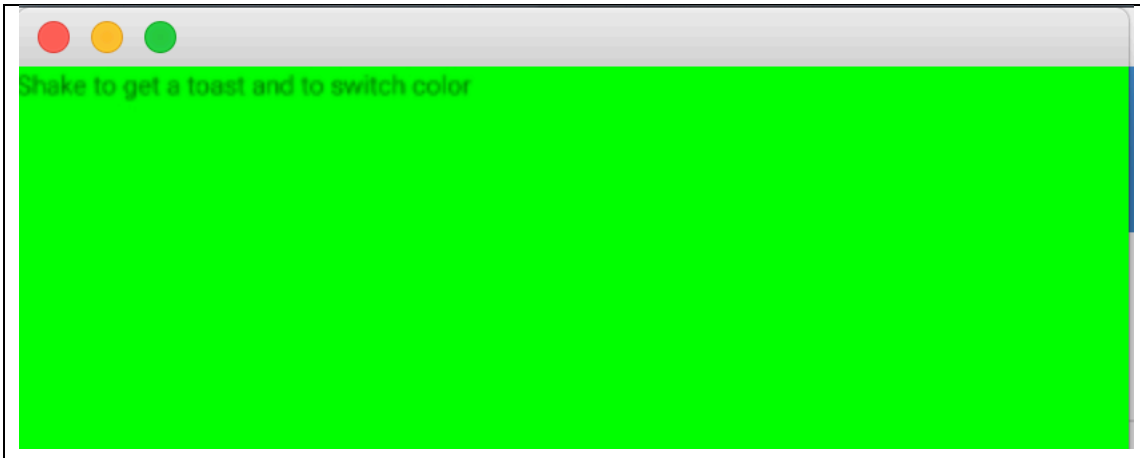
Android Studio

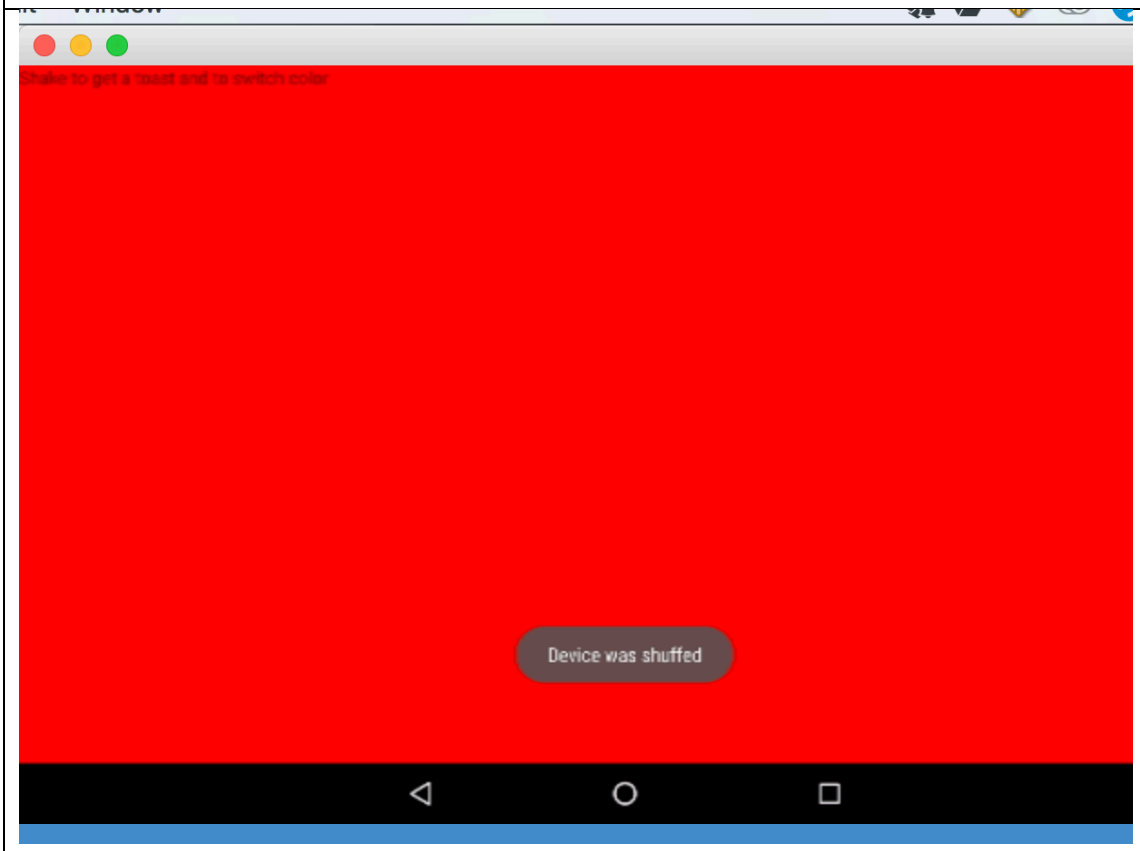
CSC3054 / CSC7054

Android Sensors Examples

Example 1

This app will change the colour of the screen if the device is shuffled.





Step 1 Create the Layout

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
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="Shake to get a toast and to switch color" />

</LinearLayout>
```

Step 2 Create the java class

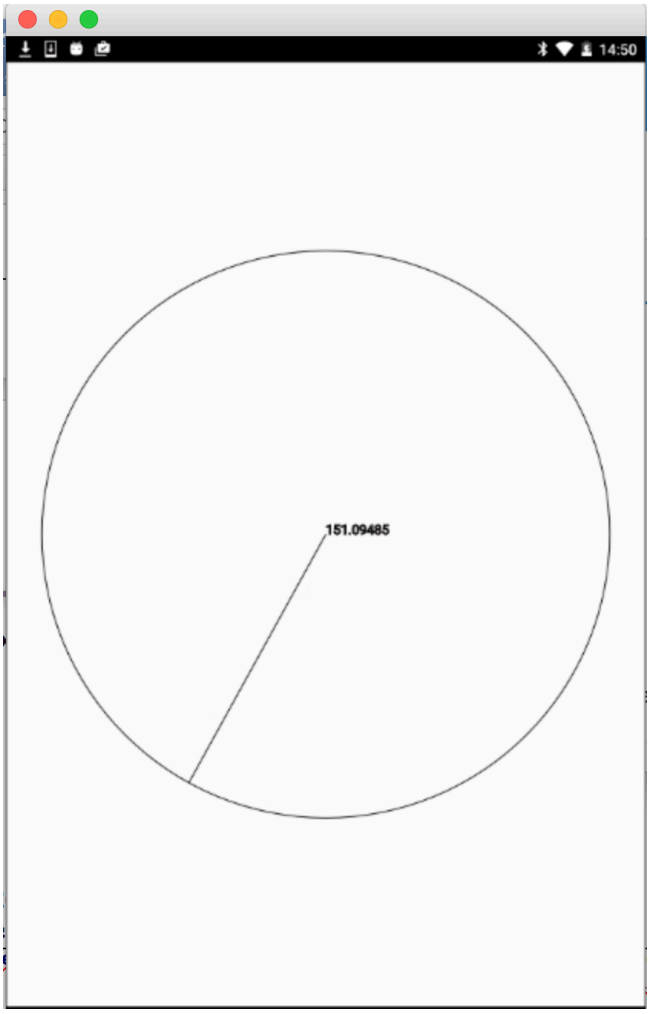
MainActivity.java

```
//change the background colour when the device is shuffled.
import android.app.Activity;
import android.graphics.Color;
import android.hardware.Sensor;
import android.hardware.SensorEvent;
import android.hardware.SensorEventListener;
import android.hardware.SensorManager;
import android.os.Bundle;
import android.view.View;
import android.view.Window;
import android.view.WindowManager;
import android.widget.Toast;

public class MainActivity extends Activity implements SensorEventListener {
    private SensorManager sensorManager;
    private boolean color = false;
    private View view;
    private long lastUpdate;

    /** Called when the activity is first created. */

    @Override
    public void onCreate(Bundle savedInstanceState) {
        requestWindowFeature(Window.FEATURE_NO_TITLE);
        getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
                WindowManager.LayoutParams.FLAG_FULLSCREEN);

        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        view = findViewById(R.id.textView);
        view.setBackgroundColor(Color.GREEN);

        sensorManager = (SensorManager) getSystemService(SENSOR_SERVICE);
        lastUpdate = System.currentTimeMillis();
```

```
    }

    @Override
    public void onSensorChanged(SensorEvent event) {
        if (event.sensor.getType() == Sensor.TYPE_ACCELEROMETER) {
            getAccelerometer(event);
        }
    }

    private void getAccelerometer(SensorEvent event) {
        float[] values = event.values;
        // Movement
        float x = values[0];
        float y = values[1];
        float z = values[2];

        float accelationSquareRoot = (x * x + y * y + z * z)
                / (SensorManager.GRAVITY_EARTH * SensorManager.GRAVITY_EARTH);
        long actualTime = event.timestamp;
        if (accelationSquareRoot >= 2) //
        {
            if (actualTime - lastUpdate < 200) {
                return;
            }
            lastUpdate = actualTime;
            Toast.makeText(this, "Device was shuffed", Toast.LENGTH_SHORT)
                    .show();
            if (color) {
                view.setBackgroundColor(Color.GREEN);
            } else {
                view.setBackgroundColor(Color.RED);
            }
            color = !color;
        }
    }

    @Override
    public void onAccuracyChanged(Sensor sensor, int accuracy) {
    }

    @Override
    protected void onResume() {
        super.onResume();
        // register this class as a listener for the orientation and
        // accelerometer sensors
        sensorManager.registerListener(this,
                sensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER),
                SensorManager.SENSOR_DELAY_NORMAL);
    }

    @Override
    protected void onPause() {
        // unregister listener
        super.onPause();
        sensorManager.unregisterListener(this);
    }
}
```

Example 2

This app will use sensors to display a compass.



Step 1 Create the layout

activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"
    android:layout_height="match_parent"
android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
tools:context=".MainActivity">

    <TextView android:text="@string/hello_world" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

</RelativeLayout>
```

Step 2 Create two Java classes

MainActivity.java

```
import android.app.Activity;
import android.content.Context;
import android.hardware.Sensor;
import android.hardware.SensorEvent;
import android.hardware.SensorEventListener;
import android.hardware.SensorManager;
import android.os.Bundle;
import android.util.Log;
import android.widget.Toast;

public class MainActivity extends Activity {

    private static SensorManager sensorService;
    private MyCompassView compassView;
    private Sensor sensor;

    /** Called when the activity is first created. */

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        compassView = new MyCompassView(this);
        setContentView(compassView);

        sensorService = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
        sensor = sensorService.getDefaultSensor(Sensor.TYPE_ORIENTATION);
        if (sensor != null) {
            sensorService.registerListener(mySensorEventListener, sensor,
                    SensorManager.SENSOR_DELAY_NORMAL);
            Log.i("Compass MainActivity", "Registerered for ORIENTATION Sensor");
        } else {
            Log.e("Compass MainActivity", "Registerered for ORIENTATION Sensor");
            Toast.makeText(this, "ORIENTATION Sensor not found",
                    Toast.LENGTH_LONG).show();
            finish();
        }
    }

    private SensorEventListener mySensorEventListener = new SensorEventListener() {

        @Override
        public void onAccuracyChanged(Sensor sensor, int accuracy) {
        }

        @Override
        public void onSensorChanged(SensorEvent event) {
            // angle between the magnetic north direction
            // 0=North, 90=East, 180=South, 270=West
            float azimuth = event.values[0];
            compassView.updateData(azimuth);
        }
    };
```

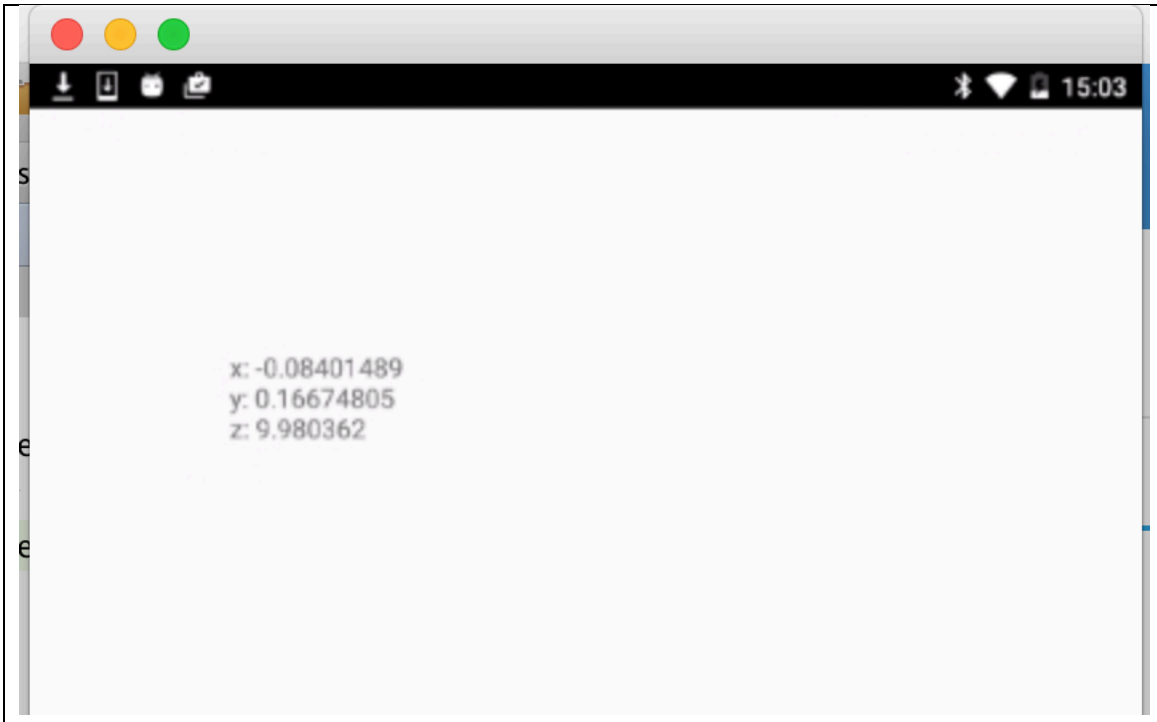
```
    @Override
    protected void onDestroy() {
        super.onDestroy();
        if (sensor != null) {
            sensorService.unregisterListener(mySensorEventListener);
        }
    }
}
```

MyCompassView.java

```
import android.content.Context;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.view.View;

public class MyCompassView extends View {

    private Paint paint;
    private float position = 0;
    public MyCompassView(Context context) {
        super(context);
        init();
    }
    private void init() {
        paint = new Paint();
        paint.setAntiAlias(true);
        paint.setStrokeWidth(2);
        paint.setTextSize(25);
        paint.setStyle(Paint.Style.STROKE);
        paint.setColor(Color.BLACK);
    }
    @Override
    protected void onDraw(Canvas canvas) {
        int xPoint = getMeasuredWidth() / 2;
        int yPoint = getMeasuredHeight() / 2;
        float radius = (float) (Math.max(xPoint, yPoint) * 0.6);
        canvas.drawCircle(xPoint, yPoint, radius, paint);
        canvas.drawRect(0, 0, getMeasuredWidth(), getMeasuredHeight(), paint);

        // 3.143 is a good approximation for the circle
        canvas.drawLine(xPoint,
                yPoint,
                (float) (xPoint + radius
                        * Math.sin((double) (-position) / 180 * 3.143)),
                (float) (yPoint - radius
                        * Math.cos((double) (-position) / 180 * 3.143)), paint);
        canvas.drawText(String.valueOf(position), xPoint, yPoint, paint);
    }


    public void updateData(float position) {
        this.position = position;
        invalidate();
    }
}
```

Example 3

This app will return the X, Y and Z coordinates of the position of the device.



Step 1 Create the Layout

activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:layout_marginLeft="92dp"
        android:layout_marginTop="114dp"
        android:text="TextView" />

</RelativeLayout>
```

Step 2 Create the Java file

MainActivity.java

```
import android.app.Activity;
import android.os.Bundle;
import android.widget.TextView;
import android.widget.Toast;
import android.hardware.SensorManager;
import android.hardware.SensorEventListener;
import android.hardware.SensorEvent;
import android.hardware.Sensor;
import java.util.List;

public class MainActivity extends Activity {
    SensorManager sm = null;
    TextView textView1 = null;
    List list;

    SensorEventListener sel = new SensorEventListener(){
        public void onAccuracyChanged(Sensor sensor, int accuracy) {}
        public void onSensorChanged(SensorEvent event) {
            float[] values = event.values;
            textView1.setText("x: "+values[0]+"\ny: "+values[1]+"\nz: "+values[2]);
        }
    };

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

        /* Get a SensorManager instance */
        sm = (SensorManager)getSystemService(SENSOR_SERVICE);

        textView1 = (TextView)findViewById(R.id.textView1);

        list = sm.getSensorList(Sensor.TYPE_ACCELEROMETER);
        if(list.size()>0){
            sm.registerListener(sel, (Sensor) list.get(0),
SensorManager.SENSOR_DELAY_NORMAL);
        }else{
            Toast.makeText(getBaseContext(), "Error: No Accelerometer.",
Toast.LENGTH_LONG).show();
        }
    }

    @Override
    protected void onStop() {
        if(list.size()>0){
            sm.unregisterListener(sel);
        }
        super.onStop();
    }
}
```