

```

package com.ece420.lab1;

import android.content.pm.PackageManager;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.Manifest;
import android.os.Build;
import android.support.v4.app.ActivityCompat;
import android.support.v4.content.ContextCompat;
import android.support.v7.app.AppCompatActivity;
import android.content.pm.ActivityInfo;
import android.os.Bundle;
import android.view.View;
import android.view.WindowManager;
import android.widget.Button;
import android.widget.TextView;

import com.jjoe64.graphview.GraphView;
import com.jjoe64.graphview.series.DataPoint;
import com.jjoe64.graphview.series.DataPointInterface;
import com.jjoe64.graphview.series.LineGraphSeries;
import com.jjoe64.graphview.series.PointsGraphSeries;

public class PedometerSimple extends AppCompatActivity {

    // Sensor Variables
    private SensorReader mSensorReader;
    private boolean sensorsOn;

    // UI Plotting Variables
    public LineGraphSeries<DataPoint> accelGraphData;
    public PointsGraphSeries<DataPoint> accelGraphSteps;

    // UI Text Variables
    public TextView textStatus;

    // Declare the private variable buttonStart
    private Button buttonStart;
    private Button buttonStop;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        getWindow().addFlags(WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);
        setContentView(R.layout.activity_pedometer_simple);
        super.setRequestedOrientation (ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);

        sensorsOn = false;
    }

```

```

mSensorReader = new SensorReader(this);

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

// Link the private variable buttonStart to the button in layout .xml by id
buttonStart = (Button) findViewById(R.id.buttonStart);

// Declare buttonStart event listener, you will have something similar for
buttonStop
buttonStart.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if (!sensorsOn) {
            sensorsOn = mSensorReader.startCollection();
            if (sensorsOn) {
                textStatus.setText("Started!");
            }
        }
    }
});

// Link the private variable buttonStop to the button in layout .xml by id
buttonStop = (Button) findViewById(R.id.buttonStop);

// Declare buttonStop even listener, you will have something similar for
buttonStop
buttonStop.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if (sensorsOn) {
            sensorsOn = false;
            mSensorReader.stopCollection();
            textStatus.setText("Stopped!");
        }
    }
});

GraphView graph = (GraphView) findViewById(R.id.graph);
accelGraphData = new LineGraphSeries<>();
graph.addSeries(accelGraphData);
graph.getViewPort().setXAxisBoundsManual(true);
graph.getViewPort().setMinX(0);
graph.getViewPort().setMaxX(300); // N * 100 (N = number of seconds to show
on graph)

accelGraphSteps = new PointsGraphSeries<>();
graph.addSeries(accelGraphSteps);
accelGraphSteps.setColor(Color.GREEN);
accelGraphSteps.setCustomShape(new PointsGraphSeries.CustomShape() {
    @Override

```

```

        public void draw(Canvas canvas, Paint paint, float x, float y,
DataPointInterface dataPoint) {
            paint.setStrokeWidth(8);
            canvas.drawLine(x-15, y-15, x+15, y+15, paint);
            canvas.drawLine(x+15, y-15, x-15, y+15, paint);
        }
    });

    if (Build.VERSION.SDK_INT >= 23) {
        if (ContextCompat.checkSelfPermission(this,
Manifest.permission.WRITE_EXTERNAL_STORAGE)
            != PackageManager.PERMISSION_GRANTED ||
ContextCompat.checkSelfPermission(this, Manifest.permission.READ_EXTERNAL_STORAGE)
            != PackageManager.PERMISSION_GRANTED) {
            ActivityCompat.requestPermissions(this,
                new String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE,
Manifest.permission.READ_EXTERNAL_STORAGE},
                1);
        }
    }

    @Override
    protected void onResume() {
        super.onResume();

        if (sensorsOn) {
            mSensorReader.register();
        }
    }

    @Override
    protected void onPause() {
        super.onPause();

        if (sensorsOn) {
            mSensorReader.unregister();
        }
    }
}

```