HCIM Summer Workshop

Android Sensors

Eduardo Matos
ematos@di.fc.ul.pt

---

1. *Create a "hello world" app named AndroidSensors*

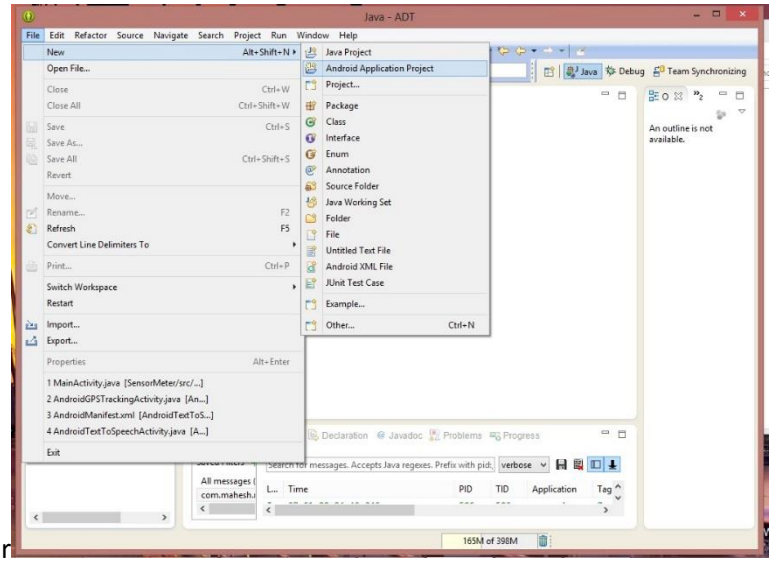

Figure 1 – Create a new app

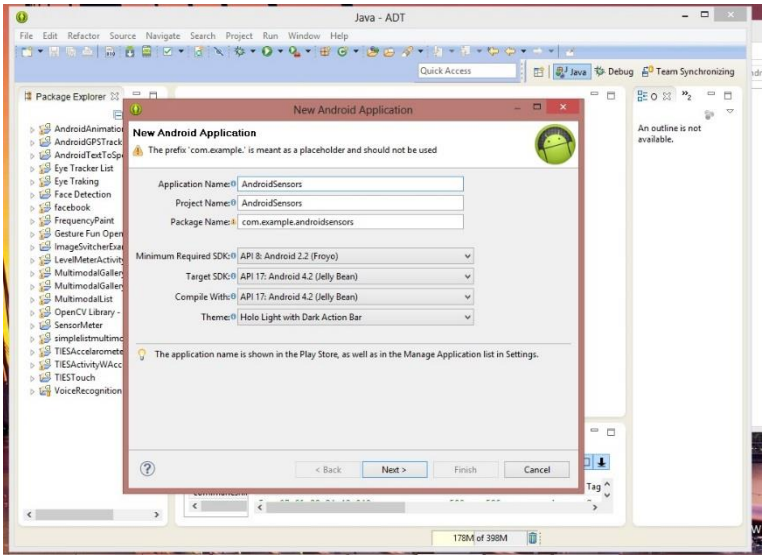

Figure 2 - Create app

- *Run as Android Application*

2. Proximity Sensor

2.1. Add TextView on Layout

package explorer>res>layout>activity_main.xml>add the following textview

```
<TextView
   android:id="@+id/prox"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_below="@+id/textView1"
  android:text="PROX: 0.0" />
```

2.2. Add these 3 parameters in your main class

```
TextView prox;
SensorManager sensorManager;
Sensor proxSensor;
```

2.3. Initialize parameters on the method onCreate

```
prox = (TextView) findViewById(R.id.prox);
sensorManager = (SensorManager) getSystemService(SENSOR_SERVICE);
proxSensor = sensorManager.getDefaultSensor(Sensor.TYPE_PROXIMITY);
```

2.4. Register/unregister Listener

```
sensorManager.registerListener(this, proxSensor,
                        SensorManager.SENSOR_DELAY_NORMAL);
```

Note: register/unregister sensors when not in use

>source>override/implemented methods>onStop/onPause/onResume

```
sensorManager.unregisterListener(this);
```

2.5. Implements SensorEventListener

```
MainActivity extends Activity implements SensorEventListener{
```

- *Add unimplemented methods*
  - *onSensorChanged*
  - *onAccuracyChanged*

2.6. Receive data from Sensors

```
public void onSensorChanged(SensorEvent arg0) {
      if (arg0.sensor.getType() == Sensor.TYPE_PROXIMITY) {
      prox.setText("PROX: " + String.valueOf(arg0.values[0]));
      }
```

```
}
```

- *Run as Android Application*

3. Accelerometer Sensor

3.1. Layout

```
<TextView
    android:id="@+id/x"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/prox"
    android:text="X: 0.0" />

<TextView
    android:id="@+id/y"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/x"
    android:text="Y: 0.0" />

<TextView
    android:id="@+id/z"
  …
```

3.2. TextView in MainActivity

Parameters: `TextView x, y, z;`

onCreate method:

```
x = (TextView) findViewById(R.id.x);
y = (TextView) findViewById(R.id.y);
z = (TextView) findViewById(R.id.z);
```

3.3. Sensors

Parameters: `Sensor accSensor;`

onCreate method:

```
accSensor = sensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER);
```

register/unregister listener:

```
sensorManager.registerListener(this, accSensor,
                        SensorManager.SENSOR_DELAY_NORMAL);
```

onSensorChanged method:

```
if (arg0.sensor.getType() == Sensor.TYPE_ACCELEROMETER) {
            x.setText("X: " + String.valueOf(arg0.values[0]));
            y.setText("Y: " + String.valueOf(arg0.values[1]));
```

```
            z.setText("Z: " + String.valueOf(arg0.values[2]));
}
```

- *Run as Android Application*

4. GPS

4.1. AndroidManifest.xml

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

4.2. You need to download the class named GPSTracker.java and add it to your current package.

4.3. Initialise componentes in MainActivity

Parameters: `GPSTracker gps;`

onCreate method:

```
gps = new GPSTracker(this);

      // check if GPS enabled
      if (gps.canGetLocation()) {

            double latitude = gps.getLatitude();
            double longitude = gps.getLongitude();

            // \n is for new line
            Toast.makeText(
                  getApplicationContext(),
                  "Your Location is - \nLat: " + latitude + "\nLong: "
                              + longitude, Toast.LENGTH_LONG).show();
      } else {
                  // can't get location
                  // GPS or Network is not enabled
                  // Ask user to enable GPS/network in settings
                  gps.showSettingsAlert();
            }
```