Assignment: SensorManager

Jason Kremhelmer

Southern New Hampshire University

CS 360: Mobile Architect & Programing

Austin Krauss

8/11/2024

According to the Android Developer’s site, the SensorManager allows you access to a device’s sensor. This is accomplished with the use of the SENSOR\_SERVICE argument. This app makes use of the available sensor suite on the device. The number of services available is dependent on the device, but could be any of the following; barometer, motion sensors, thermometer, geomagnetic, position sensor, etc.

To get a list of available sensors on the device you would use the *getSensorList* method.

*List<Sensor> deviceSensors = mSensorManager.getSensorList(Sensor.TYPE\_ALL);*

*for (Sensor sensor : deviceSensors) {*

*Log.d(TAG, "Sensor: " + sensor.getName() + " - " + sensor.getType());*

*}*

After the sensors available are identified, for example, the orientation sensor, the app can provide useful content to the user. For example, the compass uses magnetic North and the phones orientation to return a directional image or *TextView*. This uses *onSensorChanged* to return the angle from the data provided.

If you wanted to check your elevation, the barometric pressure sensor could be used. You would use the pressure sensor, and convert the pressure to a height measurement.

The *SensorManager* also allows for the developer to adjust the callback frequency. There are a number of levels that can be used to either delay or increases the timing of the callback. The levels are normal, game, UI, fastest. If you are using the frequency for screen orientation, you might want ot use normal, as there is no need to make it exceedingly fast. However, you may want to use a faster setting for games, hence the game setting. Or if you really want it to be sped up, you will trigger the fastest setting.