Geetha Anne
008741057

CMPE 277-ANDROID APP-WALKINGTRACKER

GOAL OF THE APPLICATION:

Walking Mate is an Android application that can be handy when a person in need of knowing how far has he/she has walked. This helps in calculation of number of steps taken, distance travelled and calories burnt during the journey of walking. The Goal of the application to detect the steps taken by the user and the distance travelled by him. Body weight of the person needs to be entered in the application to allow the calculation of calories.

ANDROID TECHNIQUES:

The following android techniques are utilized in the application:

Services:

Bound Services: Binding and unbinding of services is done in the application which allows it to run the step counting operation in the background without interrupting the UI thread. bindSptepService() and unbindStepService() are used in here for the background processing.

Handlers:

A Handler object registers itself with the thread in which it is created. For example, if you create a new instance of the Handler class in the onCreate() method of your *activity*, the resulting Handler object can be used to post data to the main thread. In this WlakingMate application, handler is created in the WalkingMate.java class helps the Calories, Distance and Number of steps on the main activity which is the only activity.

Shared Preferences:

Android provides many ways of storing data of an application. One of this way is called Shared Preferences. Shared Preferences allow you to save and retrieve data in the form of key, value pair. Only primitive data will be able to stored using the shared preferences. The Walking mate application only shows few limited data like calories and distance and steps which are stored using shared preferences.

Android SENSORS:

The following sensors are utilized on this android application for various operations like step detection, distance calculation. TYPE_STEP_COUNTER and TYPE_STEP_DETECTOR are the new sensors that are made available in android APIs starting from Android 4.4 but only nexus. This application does not use the above two sensors. Instead various algorithms are formulated to do the calculations. The following sensors are used in the app:

1. SensorManager.MAGNETIC_FIELD_EARTH_MAX
2. SensorManager.STANDARD_GRAVITY
3. Sensor. ACCELEROMETER

HIGH LEVEL ARCHITECTURE:

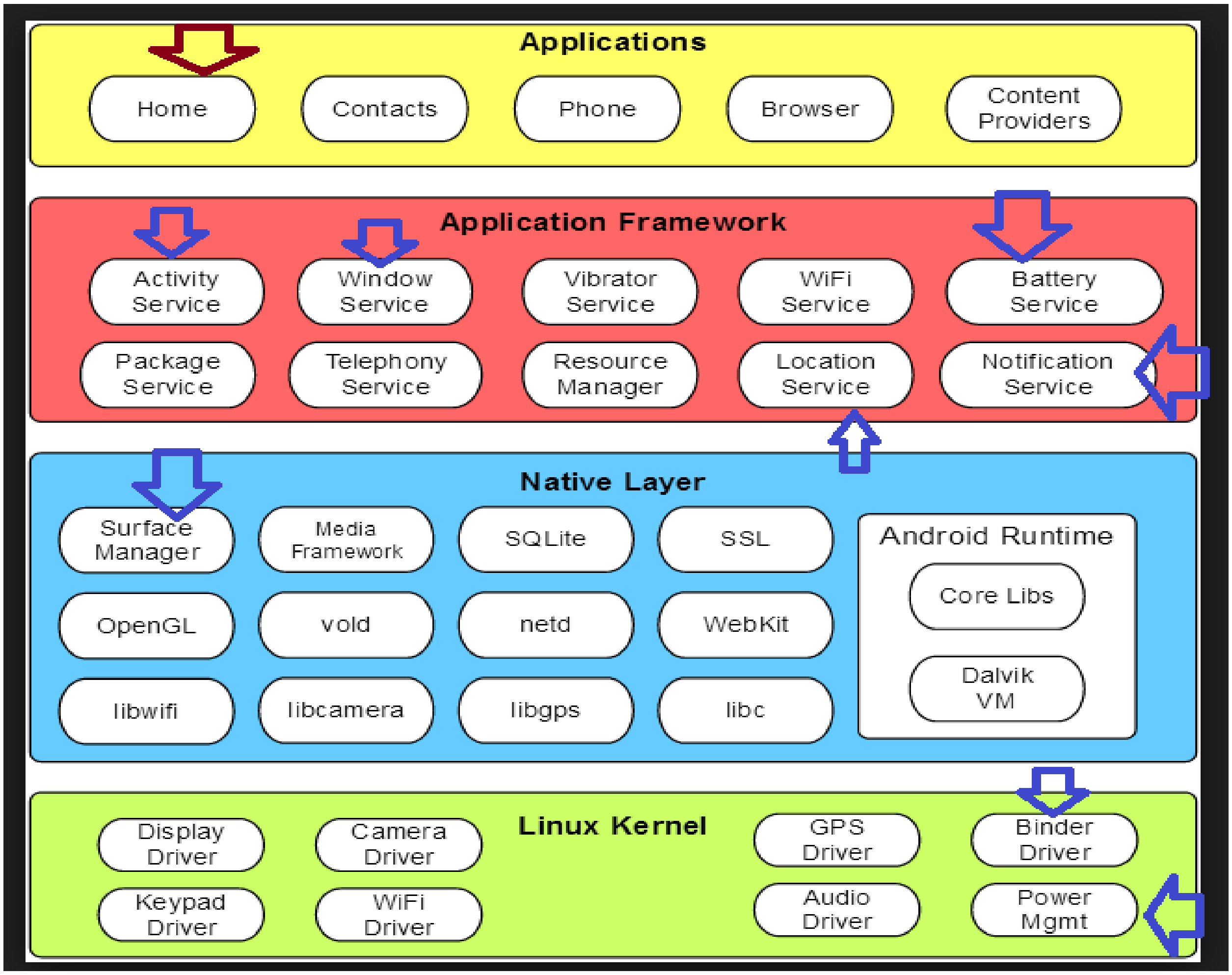


Figure: High level architecture of the android application shown.

In the above figure the various sections of the high level architecture of the android are shown. The areas that are highlighted are the ones that are specifically used in this particular application. Walking Mate application concentrated maninly on the areas where arrow marks are drawn.

Geetha Anne
008741057

SCREEN CAPTURES:

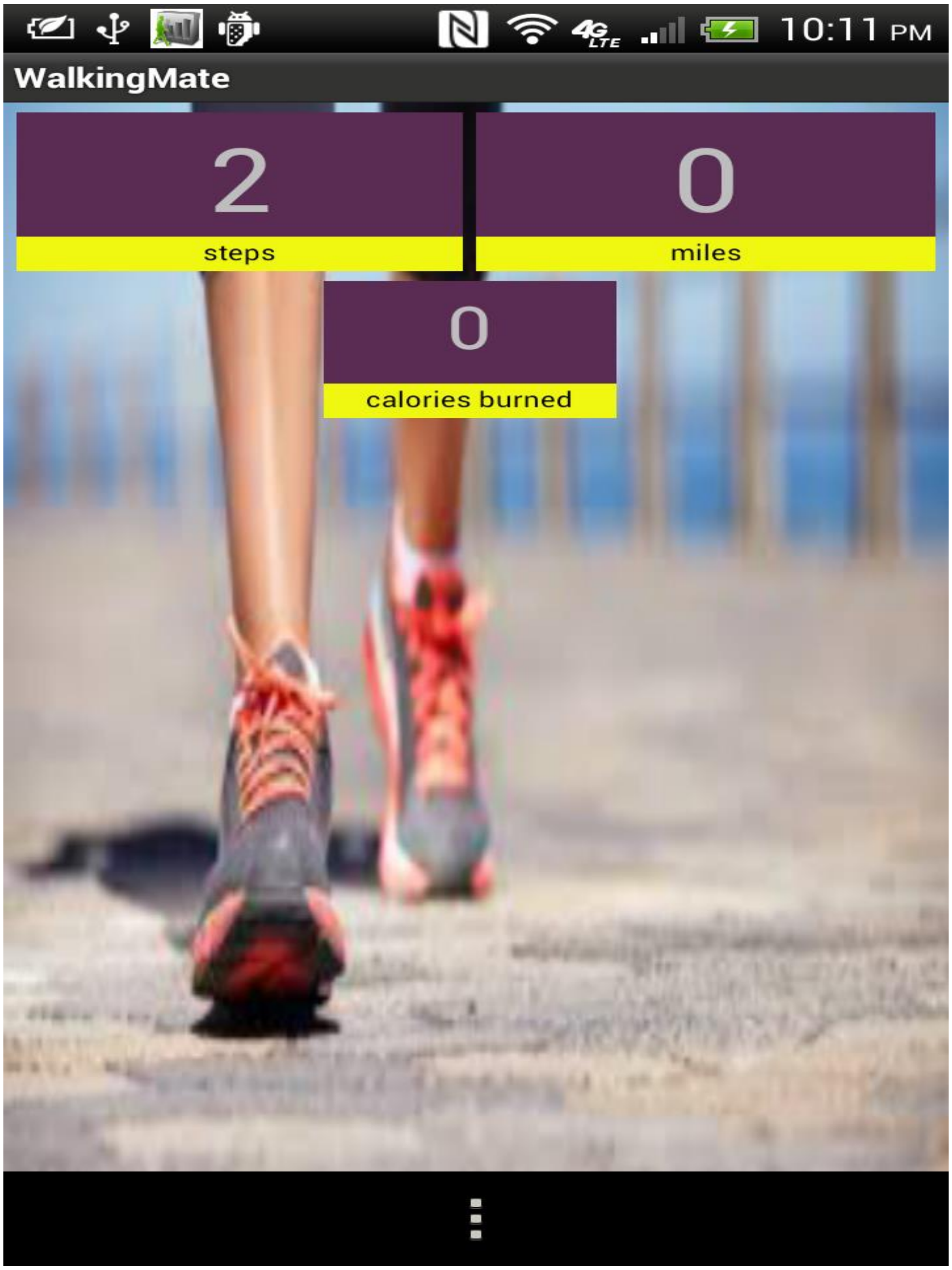
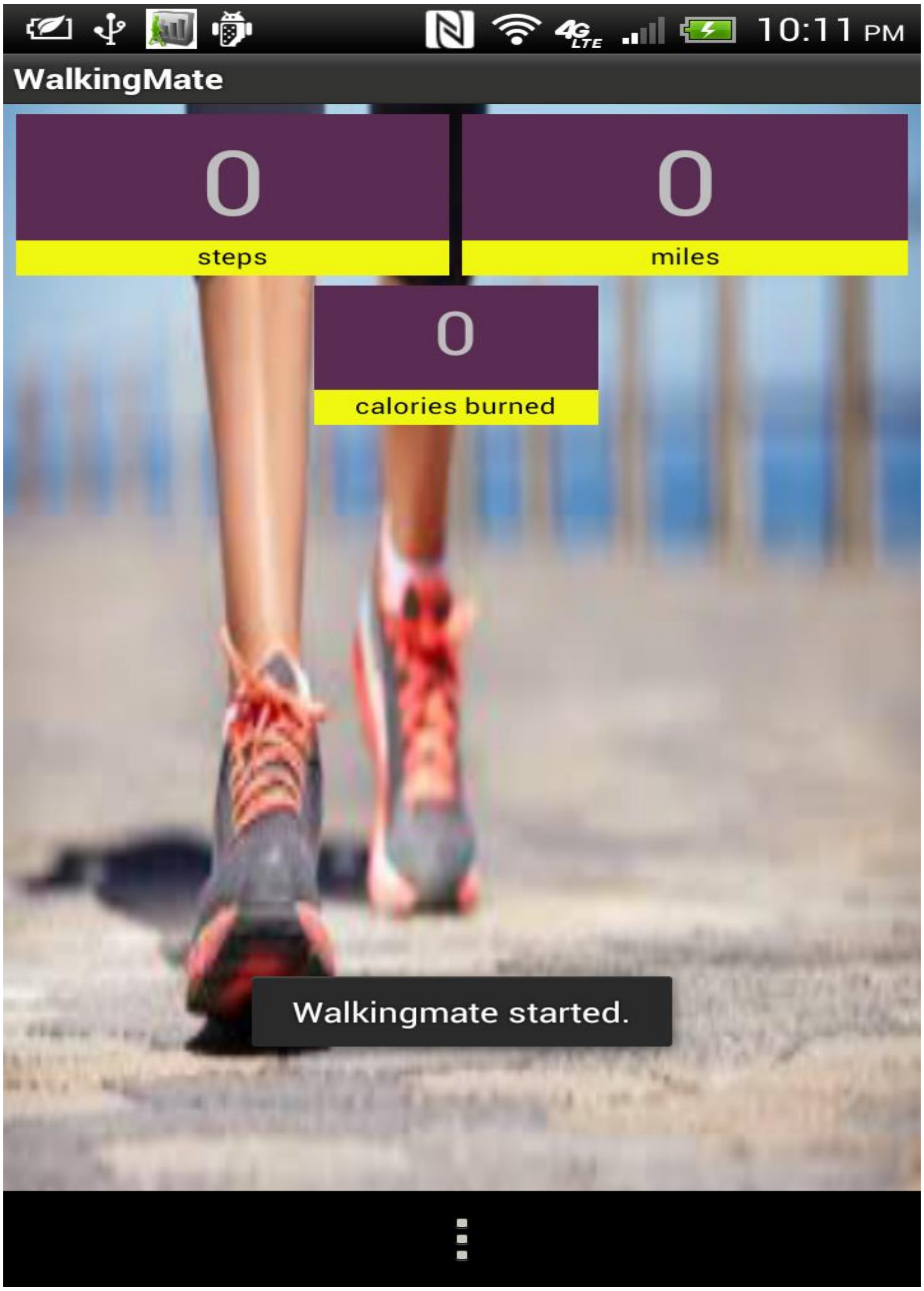


Figure: Home Screen of the Application

Geetha Anne
008741057



WalkingMate

0
steps

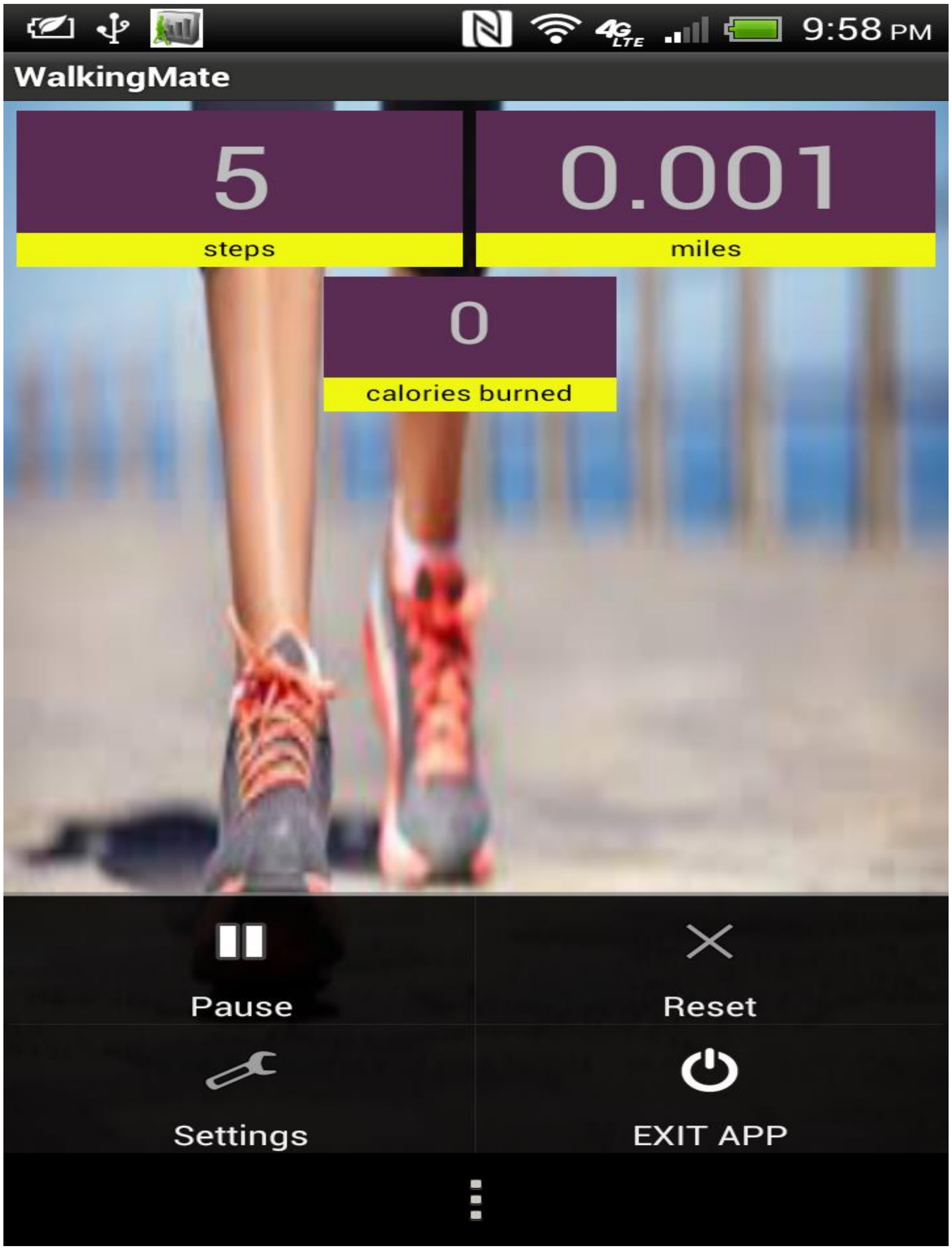
0
miles

0
calories burned

Walkingmate started.

Figure: Toast message shown when the app starts running

Geetha Anne
008741057



WalkingMate

5
steps

0.001
miles

0
calories burned

Figure: Three dots down at the screen brings the hidden menu

Geetha Anne
008741057

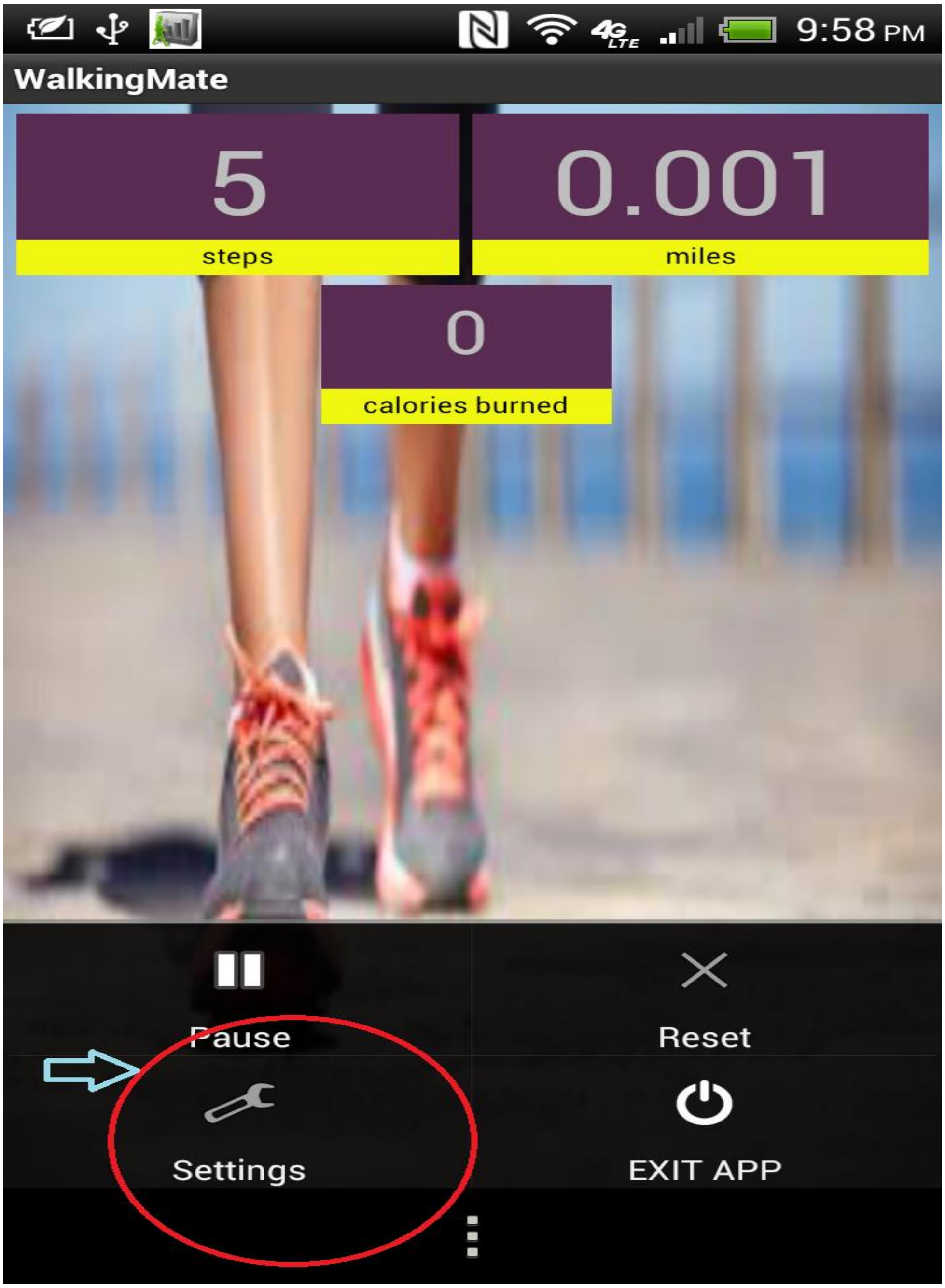
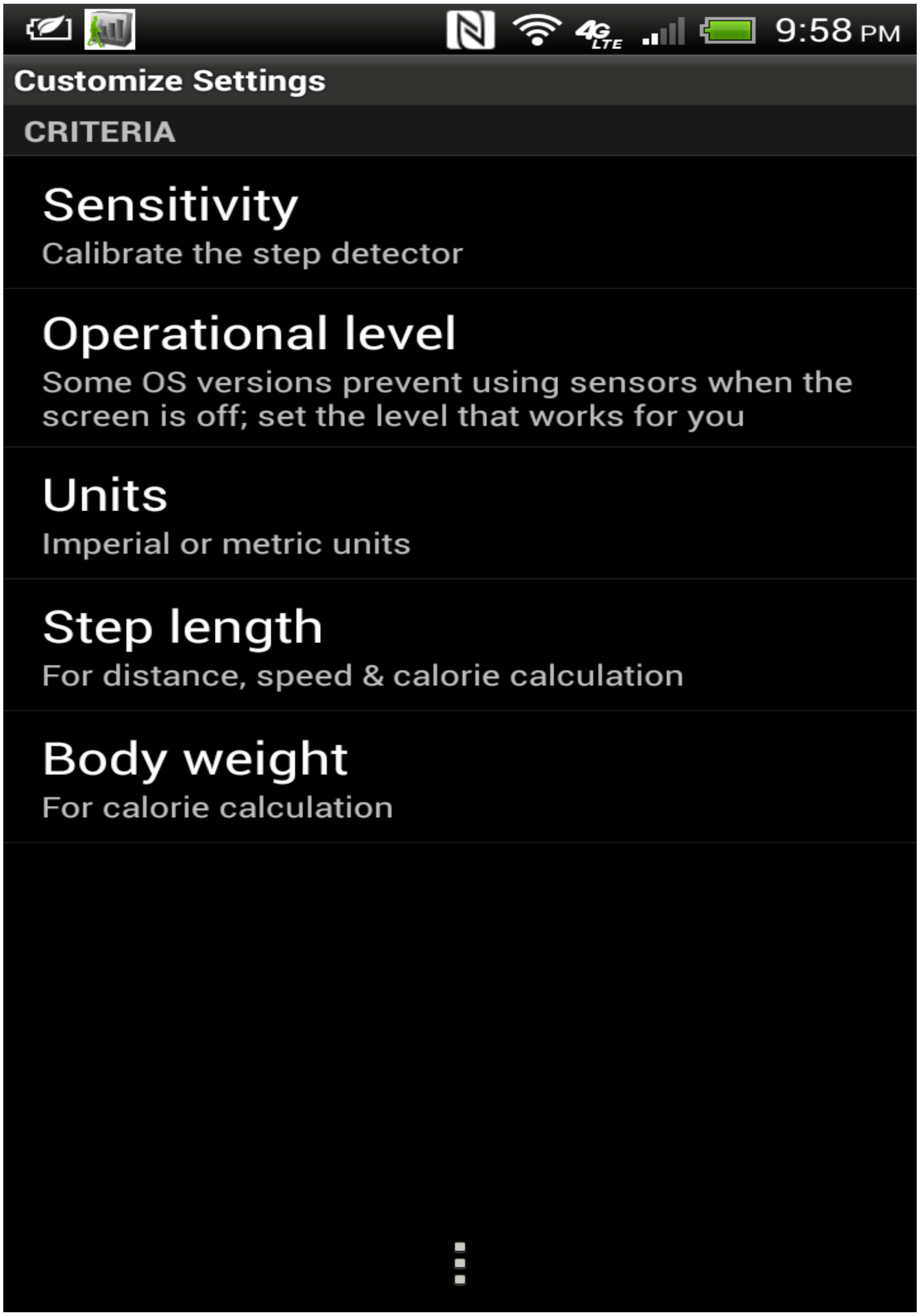


Figure: Settings tab needs to be set first before running the app.



Customize Settings

CRITERIA

Sensitivity
Calibrate the step detector

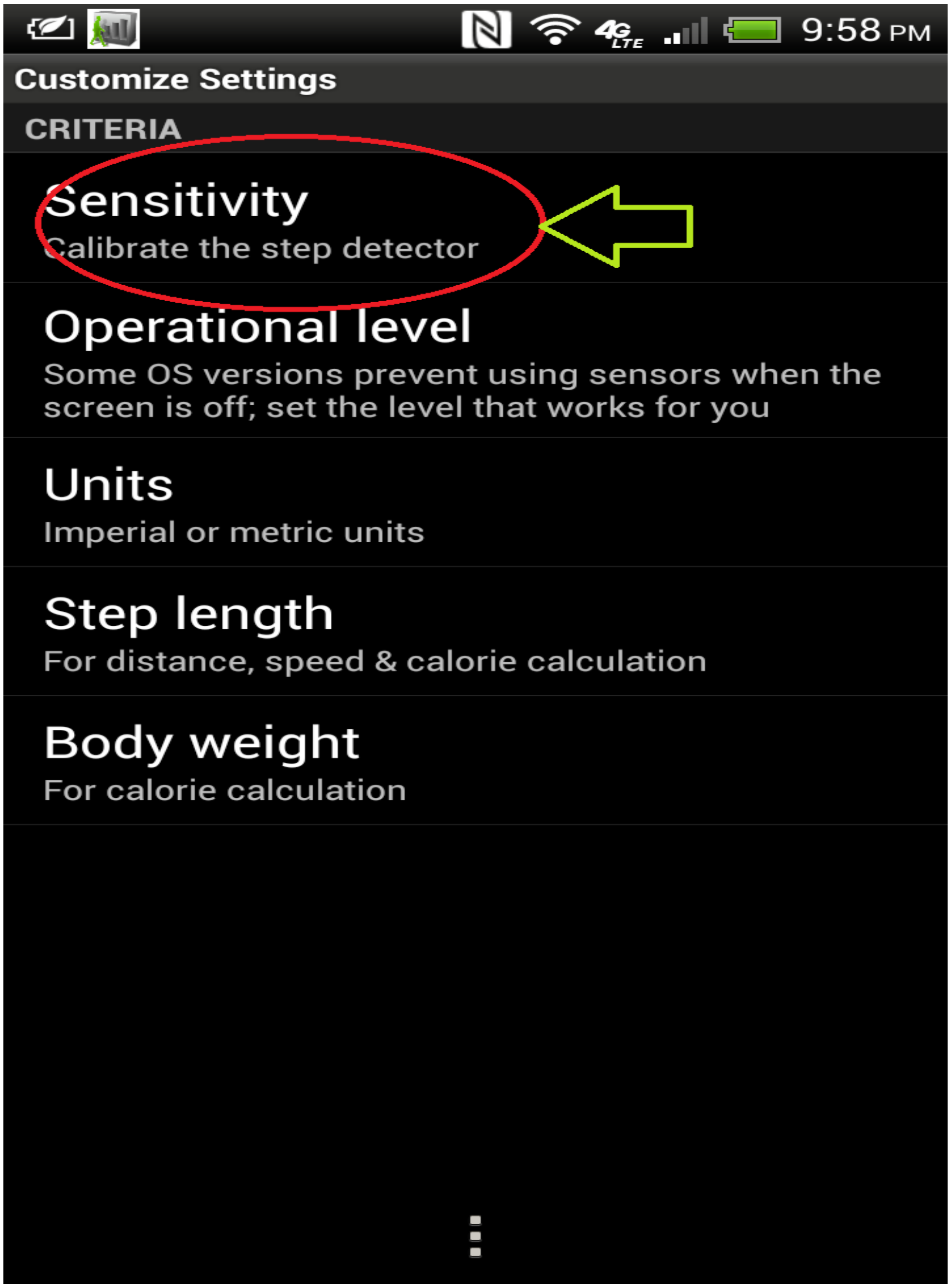
Operational level
Some OS versions prevent using sensors when the screen is off; set the level that works for you

Units
Imperial or metric units

Step length
For distance, speed & calorie calculation

Body weight
For calorie calculation

Figure: Various options menu is shown on clicking Settings tab



Customize Settings

CRITERIA

Sensitivity
Calibrate the step detector

Operational level
Some OS versions prevent using sensors when the screen is off; set the level that works for you

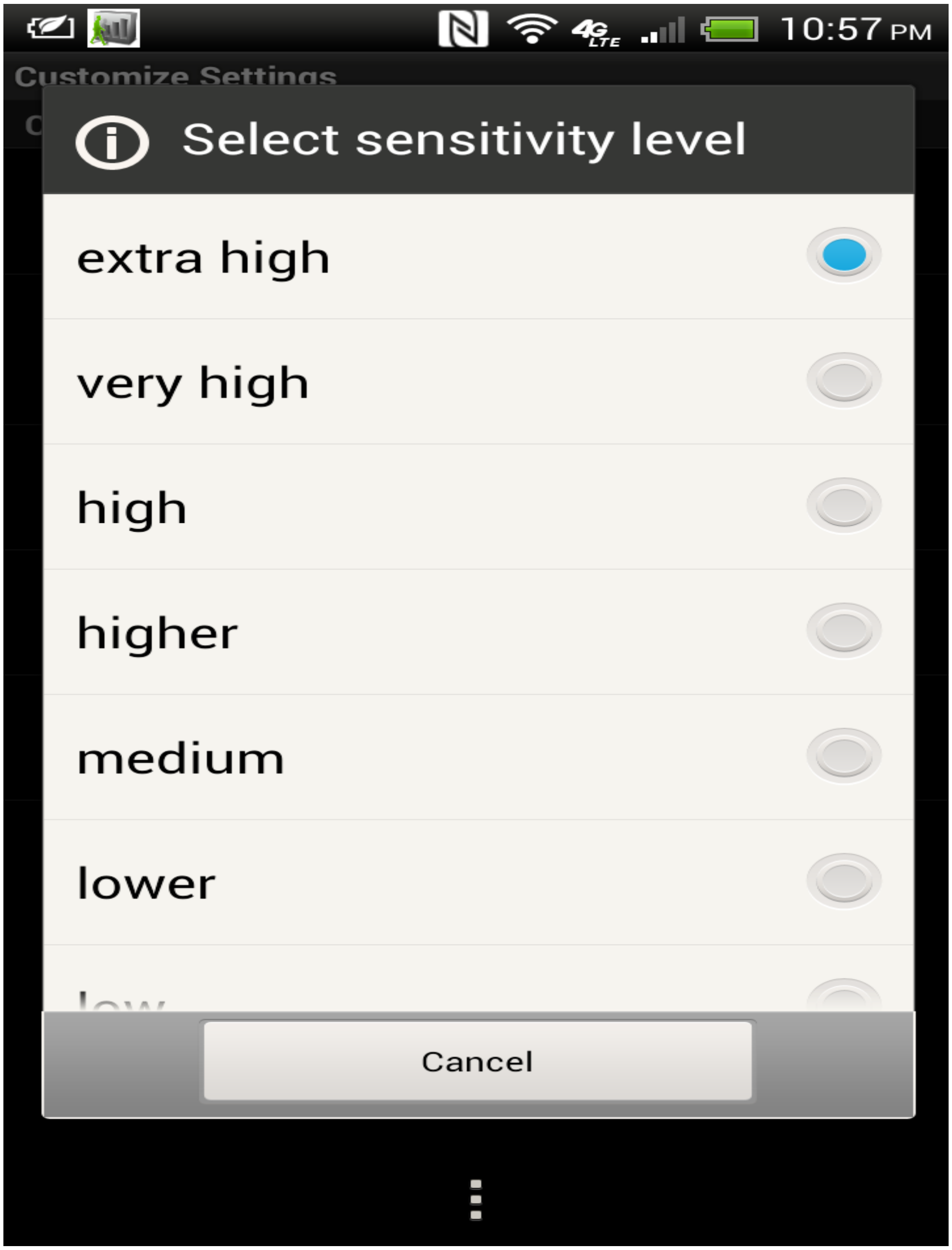
Units
Imperial or metric units

Step length
For distance, speed & calorie calculation

Body weight
For calorie calculation

Figure: Menu showing sensitivity of the sensor to detect the steps

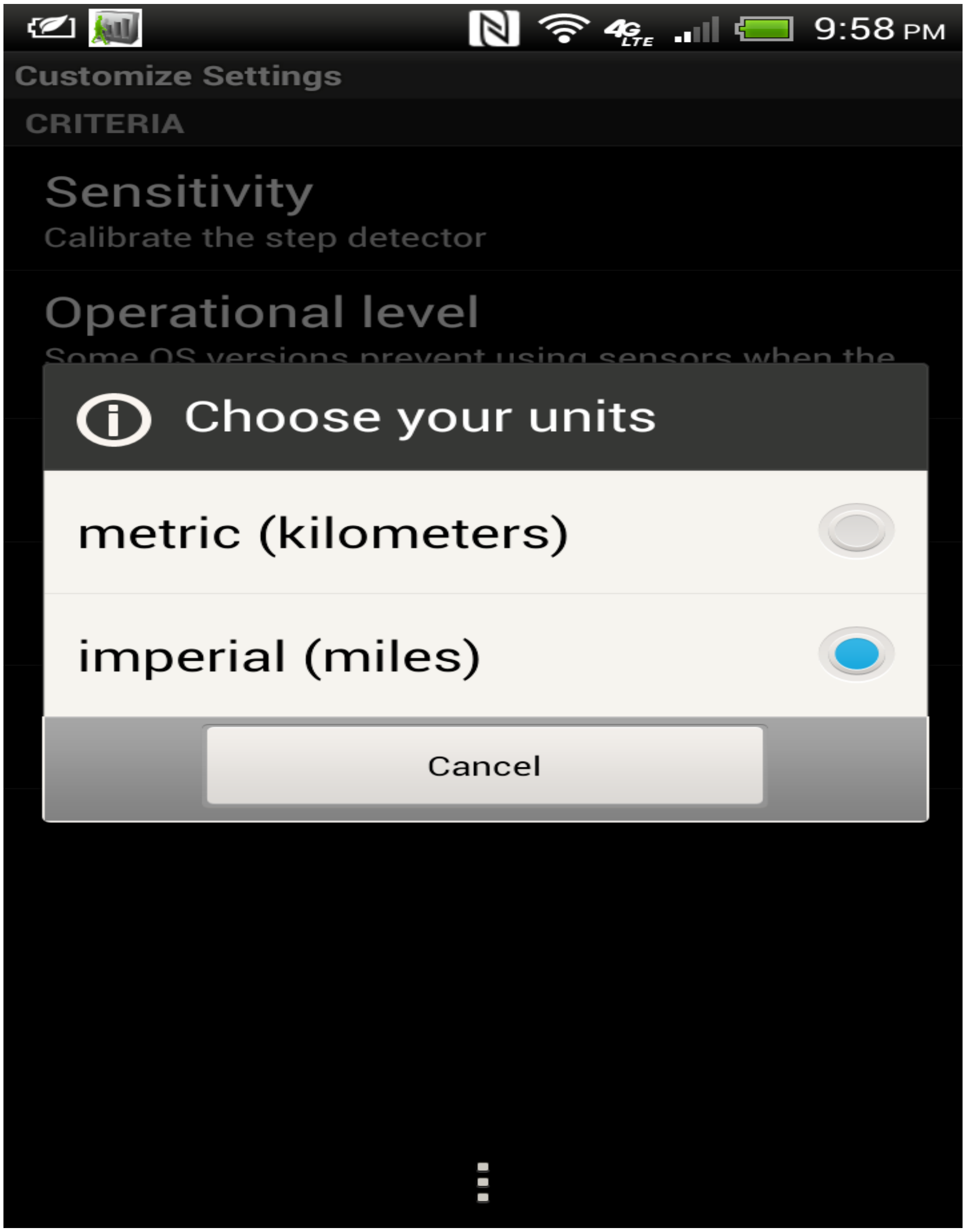
Geetha Anne
008741057



Select sensitivity level

- extra high ●
- very high ○
- high ○
- higher ○
- medium ○
- lower ○
- low ○

Figure: Various levels of sensitivity



Customize Settings

CRITERIA

Sensitivity
Calibrate the step detector

Operational level
Some OS versions prevent using sensors when the

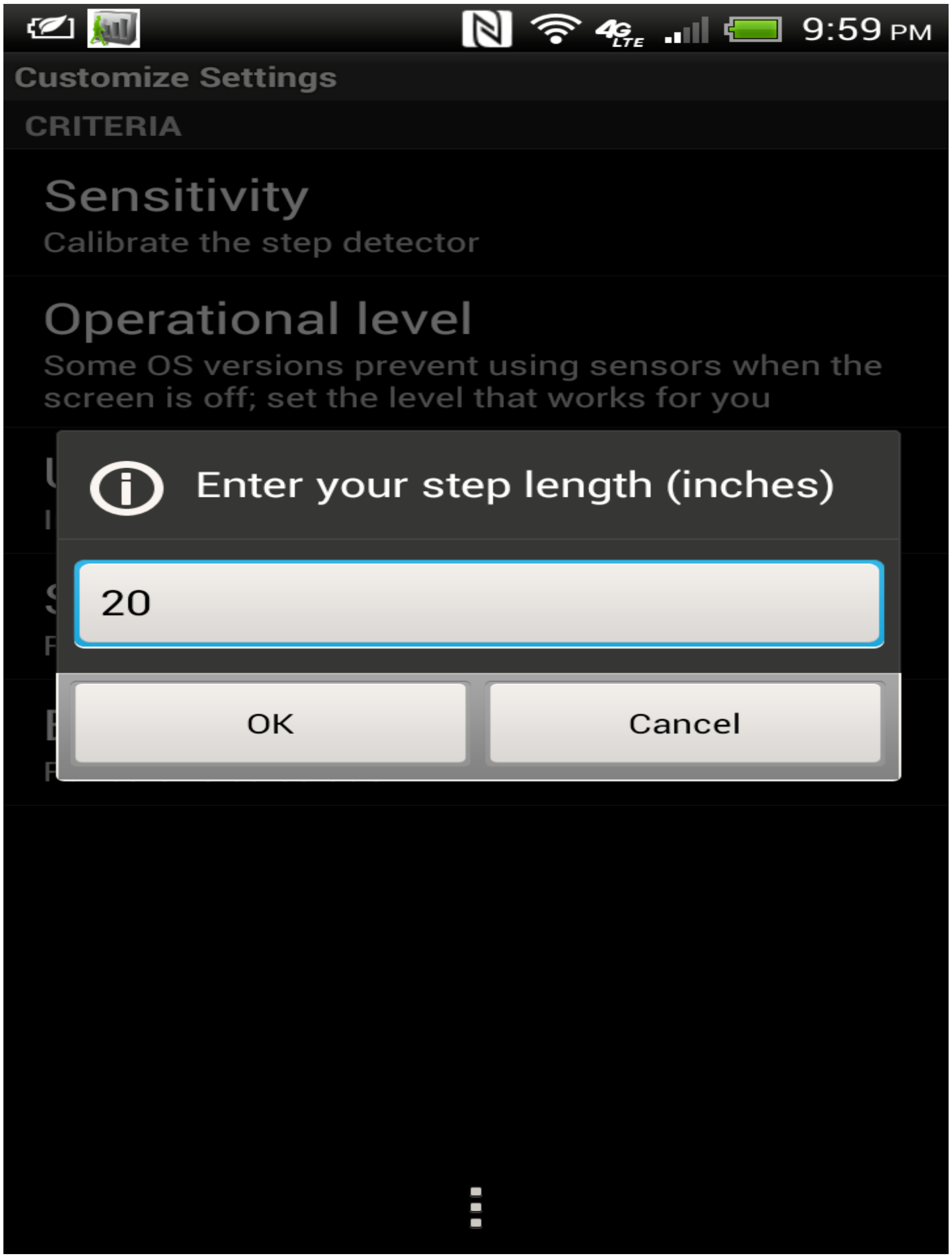
Choose your units

metric (kilometers) ○

imperial (miles) ●

Figure: Units of distance measurement(miles or kms)

Geetha Anne
008741057



9:59 PM

Customize Settings

CRITERIA

Sensitivity

Calibrate the step detector

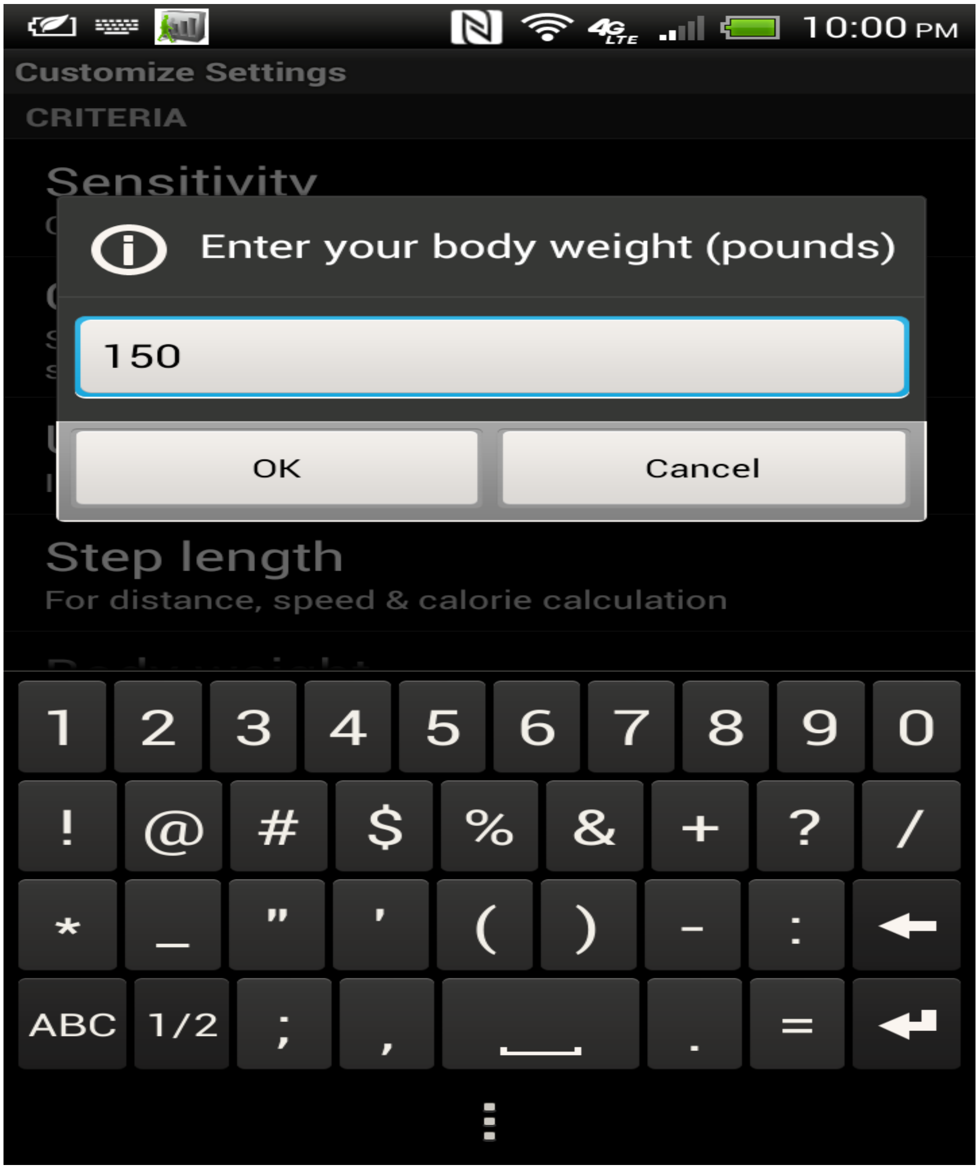
Operational level

Some OS versions prevent using sensors when the screen is off; set the level that works for you

Enter your step length (inches)

20

Figure: Length of the step to determine the distance



Customize Settings

CRITERIA

Sensitivity

Enter your body weight (pounds)

150

Step length

For distance, speed & calorie calculation

Figure: Body weight in pounds to be entered to calculate calories

Geetha Anne
008741057

TESTING:

1. **Back box testing** is done on the Walking mate application and following test cases are noticed and tested with successful execution of app

   Testcase1:

   Start app, press HOME.

   Test Result: Service or the application icon should be in the notification bar**.**

   Testcase2:

   Start the application, adjust the sensitivity and let the counting and distance be calculated .Then pause the application. Press home button. Launch the app again

   Test Result:

   On resuming the app from pause state, the previous values are retained and continuous to count the distance form then.

   Testcase3:

   Press the Reset option on the hidden bar

   Test Result:

   The values in the calories, distance and steps are set to 0.

   Test Case 4:

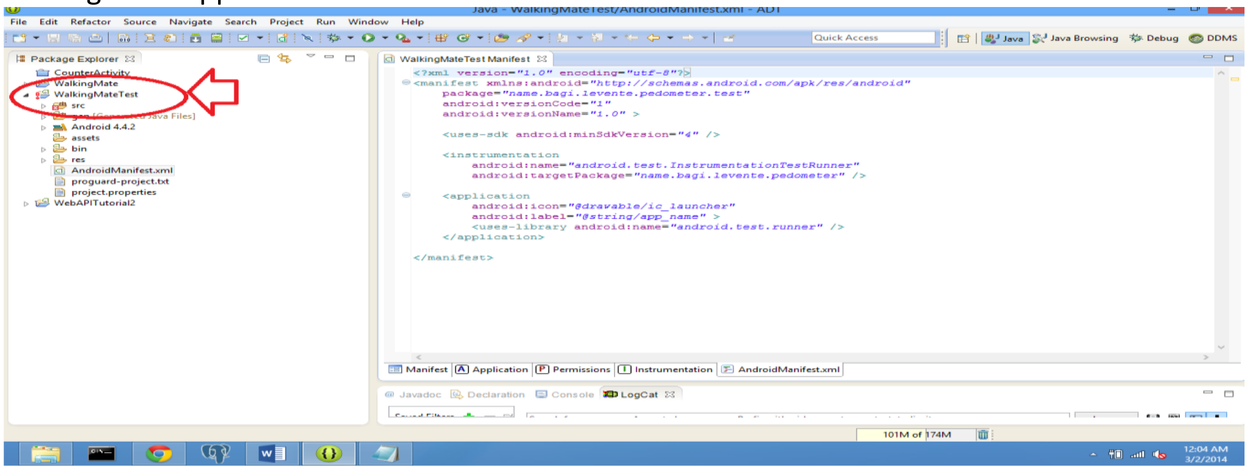
   Tried changing the metrics from miles to kilometers and vice versa.

   Test Result:

   The selected option of metrics is displayed and changed from previous unit of measurement.

   Test Case 5: Given the app is counting steps. When I start the app from the IDE (so it's killed without running any callbacks. And I press HOME. Then I should see the service icon in the notification bar.

2. JUnit testing is also done by creating a test project in the Eclipse for the each class of Walking mate application.



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="name.bagi.levente.pedometer.test"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk android:minSdkVersion="4" />

    <instrumentation
        android:name="android.test.InstrumentationTestRunner"
        android:targetPackage="name.bagi.levente.pedometer" />

    <application
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name" >
        <uses-library android:name="android.test.runner" />
    </application>

</manifest>
```

Geetha Anne
008741057