PDIoT Lab 1.1 Instructions

INFR11150 Principles and Design of IoT Systems
School of Informatics, University of Edinburgh
Course Organizer: Professor DK Arvind

Contents:

Introduction

Welcome to the PDIoT course! You will experience the different facets of designing and implementing a complex IoT system, from data collection (Coursework 1) and system specification to demonstration of a prototype implement, over the course of 10 weeks (Coursework 3). The practical work will be complemented by knowledge gained through personal research on foundational topics in Internet of Things to be distilled in one 3000-word essays (Coursework 2).

Students will be provided the following for Coursework 1 and 3:

- A wearable Respeck sensor
- Respeck accessories: plastic bags and MeFix tape
- A Nordic Cube (Thingy) – a compact multi-sensor prototyping platform based around the NRF525 SoC

You will need a smartphone running Android 6.0 or higher to run the apps needed for the course. If no one from your teams owns an Android phone we can provide you with one upon a special request.

This year, your task will be to implement an activity recognition system for a range of common activities by analysing data from a wearable sensor using machine learning techniques and displaying the results in real-time in an Android application.

You will first collect data using two sensors:

- The Respeck sensor, worn on the lower left ribcage- just below the ribs, sampling accelerometer and gyroscope data at 25Hz.
- The Thingy sensor, worn in the front right pocket of your trousers, sampling accelerometer and gyroscope data and magnetometer data at 25 Hz.

The data collection part comprises Coursework 1 and will be graded according to the quality of the data you collect. For a more detailed marking criteria and instructions on what data to collect, please refer to the Coursework 1 document listed on Learn. This data will be stored on a common repository where everyone in this class will have access to it for training their models. The data collection will mostly take place during Labs 1 and 2 where the Lab Demonstrator will make sure everyone is performing the activities

correctly. You will then develop data analysis and machine learning methods for identifying the different types of activities.

Data Analysis

1. Install miniconda

2. Create an environment for PDIoT

a. Update conda: conda update conda
b. Create the environment for the course. Call it pdiot and install python 3 (you can use any version you are comfortable with): conda create -n pdiot python=3.7

3. Setting up you environment

An environment is a collection of packages of specific versions. You can have multiple environments and switch between them for different projects. Conda is a tool for managing both environments *and* the packages within each environment. Here's a quick intro:

1. Show a list of your environments: conda env list
2. Print $PATH, one of your system's environment variables, in the terminal: echo $PATH
    - $PATH is the list of directories your terminal can search to find anything you execute:
3. Print a list of python installations on your $PATH (the top one is the one that will get executed if you type python in the terminal): which python -a
4. Activate the new environment: source activate pdiot
5. Show list of python installations on your system *now*: which python -a
6. Show your system $PATH again: echo $PATH
7. Deactivate the new environment: source deactivate
8. Observer how your $PATH has changed again: echo $PATH
9. Make an empty environment: conda create --name empty
10. You can clone environments; this is useful for backing up: conda create --name empty_bkp --clone empty
11. Make another python 3 environment with numpy already installed: conda create --name py3 python=3.7 numpy
12. conda env list
13. Activate py3: source activate py3
14. Show the installed packages: conda list
15. Switch environments: source deactivate; source activate empty
16. conda list to show packages (note that python and, crucially, pip are not installed)
17. Q: What python would get used now? which python A: the conda root environment installation of python i.e. *not* this environment's python.
18. Install numpy: conda install numpy
19. Q: What python would get used *now*? which python A: You may have clocked that conda installed a dependency of numpy (a python package)...python!
20. Let's delete these test environments:
    - source deactivate

    - conda env list
    - conda remove --name empty --all
    - conda remove --name empty_bkp --all
    - conda remove --name py3 --all
    - conda env list

4. Recommended setup

- Conda environment with python 3
- Jupyter notebooks + Numpy + Pandas + Matplotlib + Tensorflow2

Android Setup

1. If you don't already have it – Install Android Studio

It is recommended that you use Android Studio. The IDE can be downloaded from here.

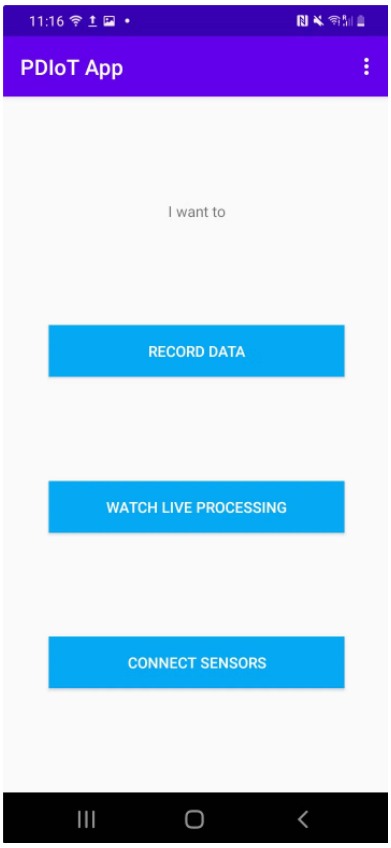
2. Phone

You need an Android phone running Android 6.0 or higher to complete this course. Teams will be formed so that at least one teammate has an Android phone. If there are still teams without a phone, we can provide you with one.

3. Test-building the PDIoT App

In order to test that the environment has been set up properly, we will build the app from Android Studio directly onto your smartphone.

1. Open Android Studio
2. Clone the Git repository of the PDIoT app.
3. Connect the phone to the computer using a USB cable.
4. Check that you can see the phone being connected in the top right corner of Android Studio.
5. Press on the "Run App" button, which can be found in the top right-hand part of the Android Studio interface. This will compile the code and install the app on the phone.
6. Unlock the phone and open the app.
7. If the app builds successfully and you can see the welcome screen of the app, your environment has been set up correctly.



4. Connecting to the Thingy through the Thingy:52 app

Obtaining the App

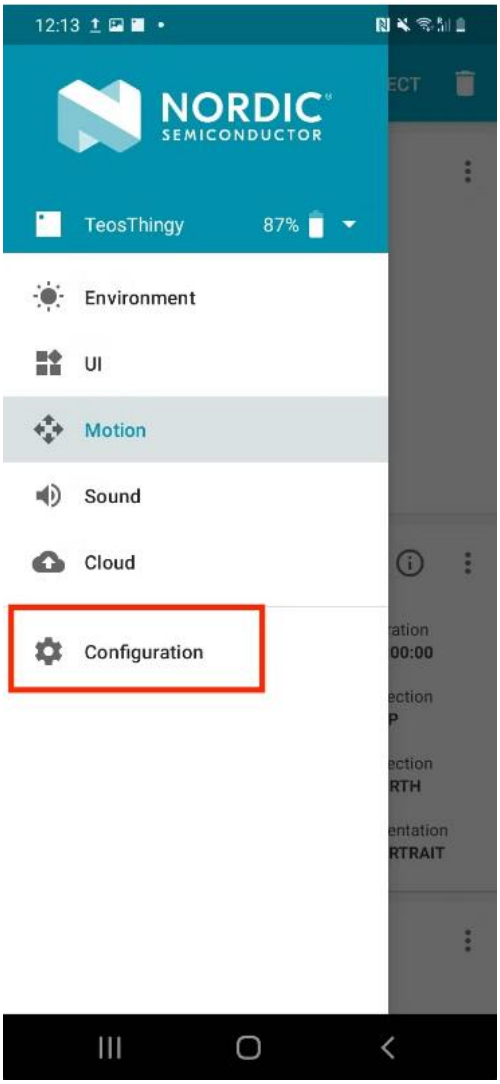
We first need to connect to the Thingy through the official app in order to set its sampling rate to 25Hz. You can read more about the Thingy:52 here. It contains multiple types of sensors but we will be mainly working with its motion package.
Download the Thingy:52 app from Google Play Store. Alternatively, you can download the code from GitHub and build the app in Android Studio.

Turning the Thingy on

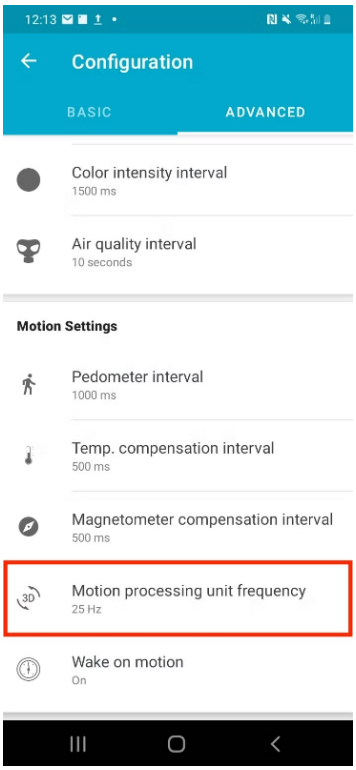
Remove the black rubber case of the Thingy. You will find a switch on its bottom side. When the Thingy is on, but not yet connected to any app, it will blink *blue*. When it's connected, it will blink *green*.

Setting the motion processing unit frequency to 25Hz

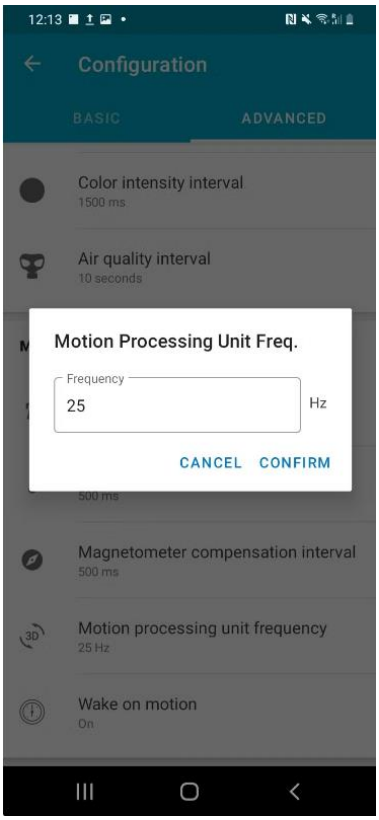
- Connecting your Thingy to the app by following the instructions
- Then, navigate to “Configuration” in the sidebar menu



- Switch to the “Advanced” tab and find the “Motion processing unit frequency” under “Motion settings”

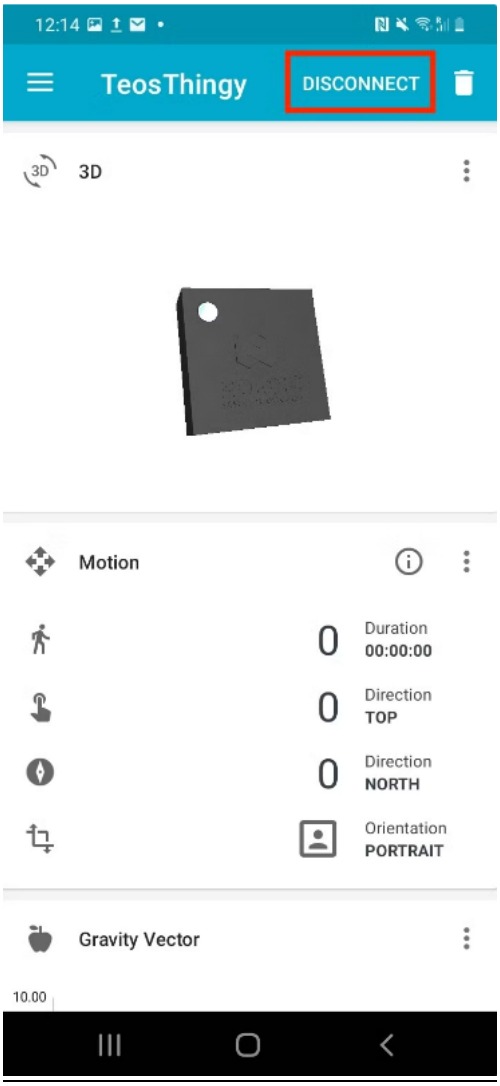


- Set this to 25Hz and click “Confirm” to apply the setting



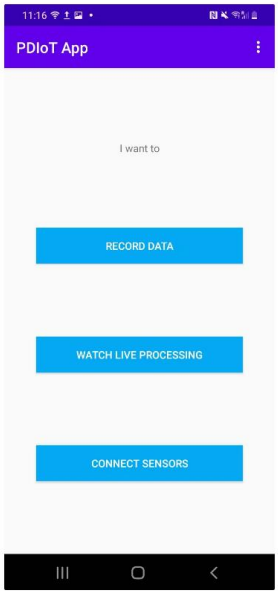
Disconnecting from the Thingy:52 app

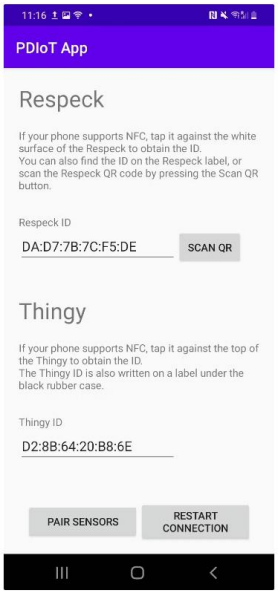
It's important to make sure that your sensors are not connected to any other apps or phones when you are trying to pair with them. Make sure you disconnect your Thingy from the Thingy:52 app by pressing the *Disconnect* button in the top right corner of the main screen.



5. Connecting to the sensors through the PDIoT app

When you first start the application, you will need to connect it to the Respeck and the Thingy. Do so by navigating to the *Connect Sensors* activity. Here, you will see two fields where you need to input the Respeck ID and the Thingy ID, respectively.





You have multiple choices for connecting the sensors:

- NFC pairing if your phone supports NFC
- Scanning the QR code of the Respeck
- Manually input the IDs into the fields

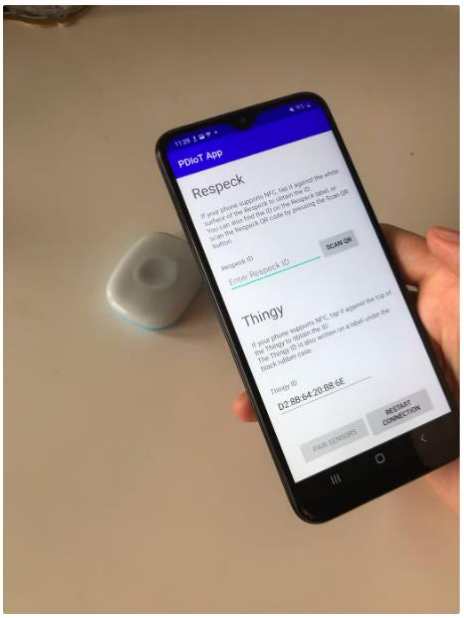
You should only need to pair these sensors once. Their IDs will be remembered by the app whenever you start it again.

Make sure the sensors are both on:

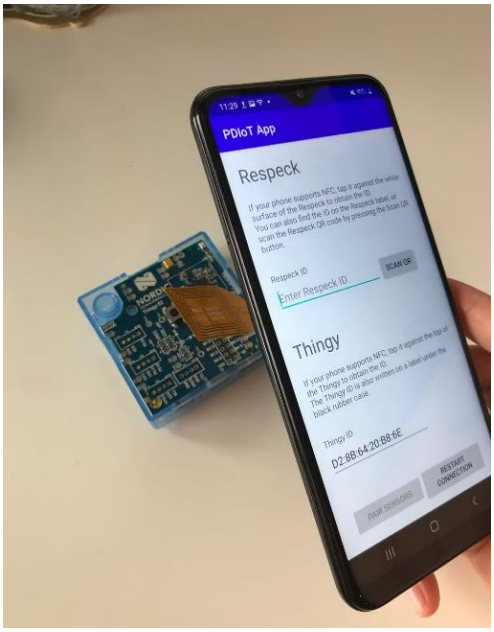
- Respeck – move the sensor around and it should blink *green* when it wakes up
- Thingy – turn on using the switch on the bottom left edge and it should blink *blue* when it is on

NFC Pairing

If your phone supports NFC you can simply tap it against the white surface of the Respeck to get the Respeck ID autocompleted in the corresponding field.



Similarly, you can tap the phone on the front side of the Thingy (here shown without the rubber case) to get the Thingy ID autocompleted in the corresponding field.

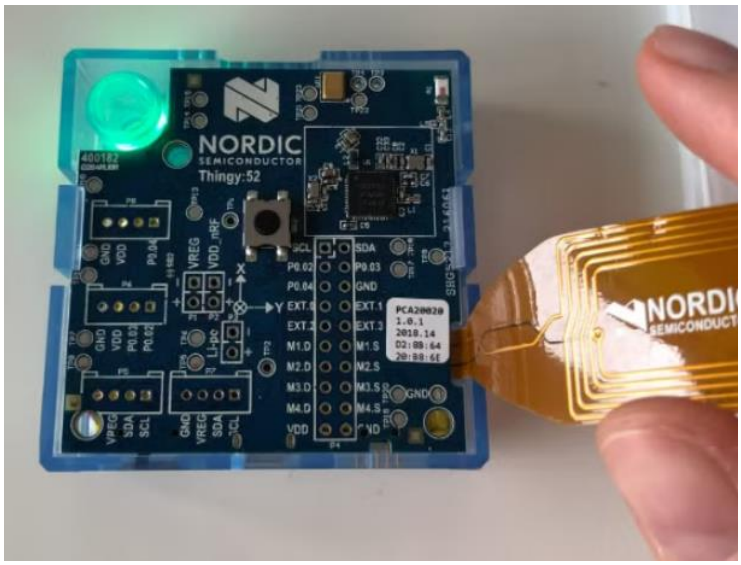


Scanning the Respeck QR code

If your phone does not support NFC, you can scan the QR code of the Respeck instead.
Every Respeck should have a QR code printed on its back. By pressing the Scan QR button, a camera view will pop up and you will be able to scan the Respeck QR code to pair it to your app. Only one Respeck can be paired with an app at one time.

Finding the Thingy ID

Unfortunately the Nordic Cubes do not have a QR code attached to them but you can find their ID (MAC address) on a label under the rubber case and NFC tag, as shown in the picture below. You need to manually input this code under the “Thingy ID” field.



Establishing the connection

Once you have entered both sensors’ IDs you will be able to click the button *Pair sensors* to start the Bluetooth service and connect to the sensors.

If you ever need to change the sensors you can scan the IDs of the new sensors and click on *Pair sensors* again. This will restart the service with the new IDs. If you have any connection issues, you can click on *Restart connection* to restart the Bluetooth service forcefully.

The sensors have differently coloured LEDs that change with connection states.

Thingy:

- *Blue* light -> sensor ON and NOT CONNECTED
- *Green* light -> sensor ON and CONNECTED
- No light -> sensor OFF

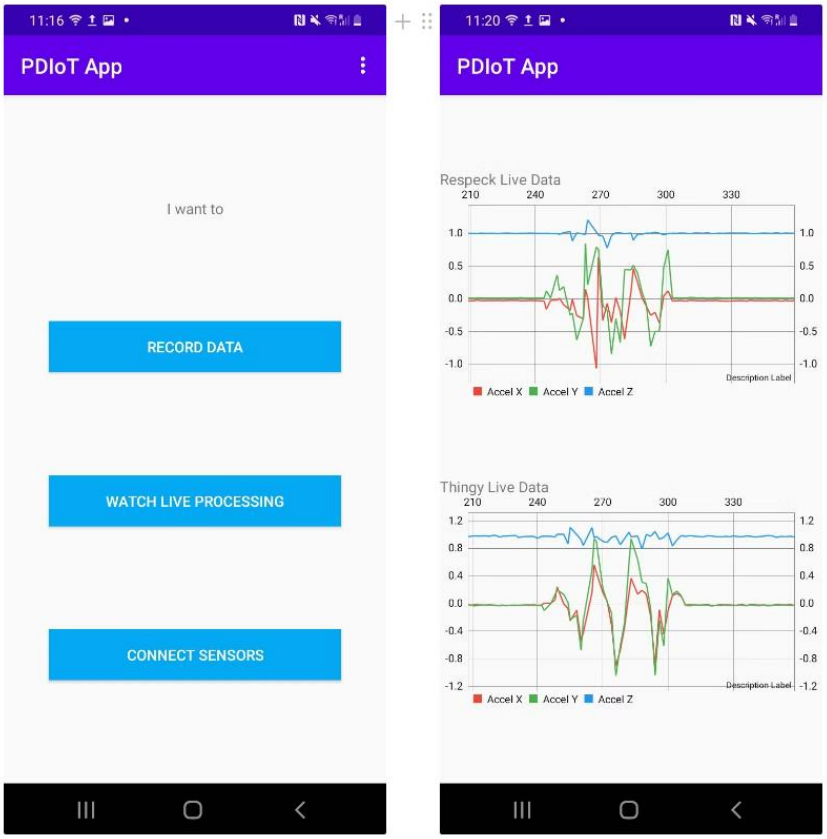
Respeck:

- *Green* light blink -> sensor ON and NOT CONNECTED
- *Blue* light -> sensor ON and CONNECTED
- *Red* light -> sensor ON but just DISCONNECTED

So, when you connect to them, you have to watch out for the Thingy to blink *green* and the Respeck to blink *blue*.

6. Viewing live data

You can view incoming data from both sensors in the “Watch live processing” activity. This will show you two live graphs of the accelerometer data from the Respeck (top) and Thingy (bottom). Both sensors should run at 25Hz.



Next Up

You have reached the end of Lab 1.1. Head over to the Lab 1.2 document to start collecting data!