Lab 4 - Programming for IoT applications

Matteo Orlando

Intrduction

In this lab we will try to develop different kind of sensor simulator that use the MQTT protocol

Remember to always use SenML as data format:

```
{
    "bn": "http://example.org/sensor1/",
    "e": [
            {
                "n": "temperature",
                "u": "Cel",
                "t": 1234,
                "v":22.5
            }
        ]
}
```

Exercise 1

Develop an MQTT publisher to emulate a temperature sensor that publish random values in the range [–10,39] every 5 seconds for 2 minutes. Develop also an MQTT subscriber that receives these values, prints these on screen and save these on a json file called *temp_log.json*. To generate the values, you can use one of the functions of the library random listed below:

- random.randint(a,b)
- random.random()
- random.uniform(a,b)

Exercise 2

Develop an MQTT publisher to emulate an heartrate sensor that publish random values in the range [55,180] every 5 seconds for 2 minutes. Develop also an MQTT subscriber that receives these values, prints these on screen and save

these on a json file called *hr_log.json*. To generate the values, you can use one of the functions of the library numpy listed at **this link**, try with different functions to evaluate which is the one that has the most realistic result. You can find a visualization of some of this generator **here** (you can even make your test there before writing your code)

Exercise 3

Using the file *hr_log.json* created in the exercise before, develop an MQTT publisher to emulate a heartrate sensor that publish on by one the data in that file

Exercise 4

Develop an MQTT publisher to emulate a sensor of your choice that publish random values in the 3 possible ranges. For example for an heartrate sensor we could define 3 ranges as [resting,sport,danger]. Then create a simple terminal client for this MQTT publisher to select the range to be used to send the data. Develop also an MQTT subscriber to receive those data and in case of data in a warning range provide a visual feedback.