

Assignment 1

Introduction

The goal of this assignment is to collect temperature and humidity data from DHT11 sensor and save it to log file using Raspberry Pi (RPi) as an IoT gateway.

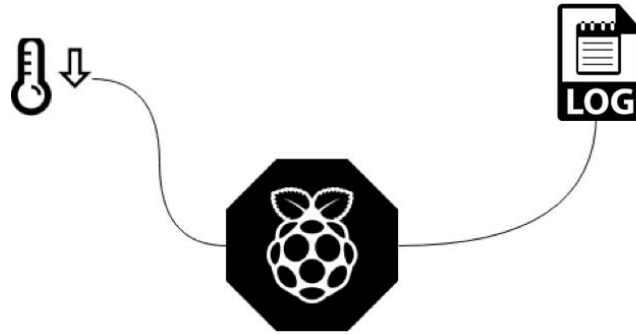


Figure 1 System architecture

At the end of this assignment the IoT system built by each student should:

1. Read temperature/humidity data accurately using Raspberry Pi
2. Collect data and save it to a log file located on RPi
3. Design multiple alerts

Steps

1. Connect DHT11 sensor using power, ground and data PINS on RPi GPIO board.
2. Write Python script to gather data from the DHT11 sensor to RPi.
3. Extend Python script to send sensor data to a dedicated log file.
4. Extend Python script to send sensor data to a dedicated log file continuously exactly every 5 seconds.
5. Design alerts according to instructions in alerts section below.

Note: Students should implement this assignment using single Python script

Log file

Collected temperature and humidity data should be saved to a log file (in your python script directory). Each entry should be prepended with a timestamp formatted as *Year-Month-Day Hour-Minutes-Seconds.Milliseconds* for example:

```
2012-12-15 01:21:05.838374    Temperature=20, Humidity=30%
2012-12-15 01:21:10.423322    Temperature=22, Humidity=30%
```



2012-12-15 01:21:15.838374 Temperature=21, Humidity=30%

Alerts

Alerts are notifications informing the user about an unusual event taking place in the system. Alerts usually take form a single line message written in plain English and should be self-explanatory.

Refer to this video to learn more about tags and alerts used by Logentries. Please note that you are not required to use Logentries in this assignment:

<https://logentries.com/resources/how-to-videos/tags-alerts/>

Alerts

You are asked to design multiple alerts. It is expected that students are going to be working in various physical environments, hence there no strict rules for the temperature or humidity thresholds imposed by this assignment, however each student should establish an average value for both temperature and humidity in their own environment.

*For example, being in one of the classrooms we have established that an average temperature is within 18° – 22°. We have therefore decided that any reading above 22° should trigger a tag labeled “It is very hot”, any reading below 18° should trigger a tag labeled “It is cold” and any reading within 18° - 22° should not trigger any alerts. The same rule applies for humidity readings. **Students can change those thresholds based on the physical environment they’re working in. Make sure to include your own table in the report.***

Alert messages should be also written to the same log file as above.

Temperature	Alert message	Humidity	Alert message
>22	“It is very hot”	0% - 30%	No message
18 - 22	No message	30% - 50%	“Humidity is high”
<18	“It is cold”	50% - 100%	“Humidity is very high”

Table 1 Sample table used to define tags

Sample alert:

```
2012-12-15 01:21:05.423322      It is very hot
2012-12-15 01:21:10.838374      It is very hot
2012-12-15 01:21:15.373628      Normal temperature
```

Note: Each entry should be prepended with a timestamp formatted as *Year-Month-Day Hour-Minutes-Seconds*



Sample log file

```
2012-12-15 01:21:00.423322    Temperature=20, Humidity=30%
2012-12-15 01:21:10.363782    Temperature=23, Humidity=30%
2012-12-15 01:21:15.242343    It is very hot
2012-12-15 01:21:20.654456    Temperature=20, Humidity=30%
2012-12-15 01:21:25.344346    Temperature=22, Humidity=30%
2012-12-15 01:21:30.654453    Temperature=21, Humidity=30%
```

Report

The purpose of the report is to present steps and methodology used when building the IoT system. Students are free to decide on the structure of the report; however, the following elements need to be included in order to obtain full marks:

- 10 to 100 lines from a generated log file to demonstrate the correct format of log entries and alerts
- Your own Temperature/Humidity thresholds table for reference (see table 1 above)
- Short discussion on how your system could be improved in the future

Submission

Please upload **2 files only** to CS Moodle only:

- Python script (.py)
- Report (.doc, .pdf)

