Orientation Challenge: Climate monitoring system.

By: Johnson Domacasse (#4471709)

09 March 2024

Contents

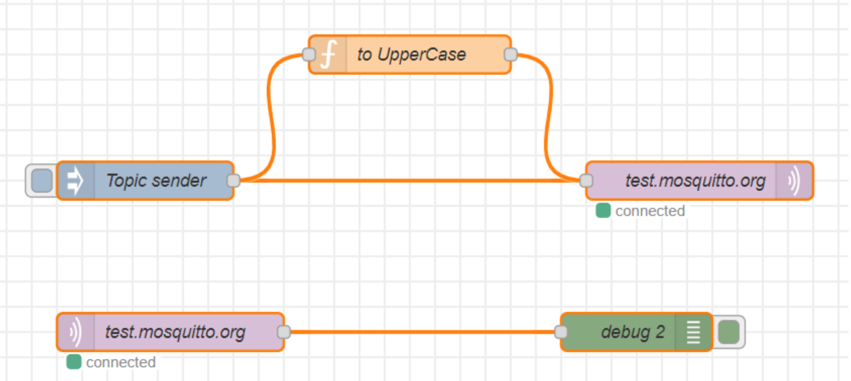
1. Introduction:

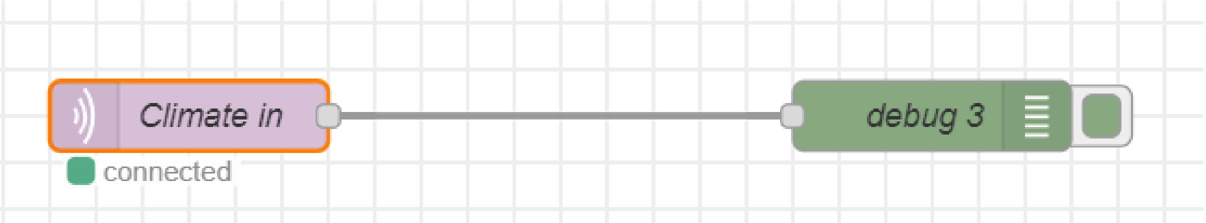
IoT has always been important in smart industry. It offers data to be sent over to other devices using powered devices. This challenge offers insight into Node-red, I software that can be used to monitor your IoT devices. It also introduces the concepts of MQTT and sensors for data collection.

NOTE: *Since I have some knowledge on all of these topics from semester 3, I will not go too deep into understanding these concepts but rather what I did to get the desired results.*

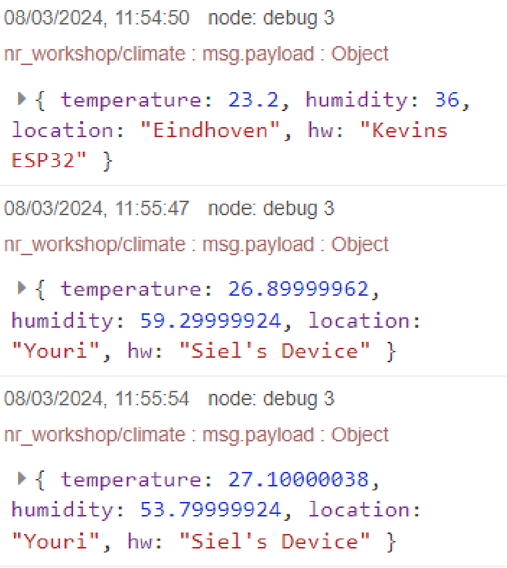
2. Steps Taken:

First, I wanted to confirm if the MQTT broker worked. I have a previously made example to test the mosquito broker and this worked fine still. In the image on the right here you can see a simple example. I have a send block to inject a simple "hello into the block" and then read topic to see if it works. In this case it did. A similar approach was taken to first test if the brokers was working properly. Since there was data being posted we only needed to read this data.



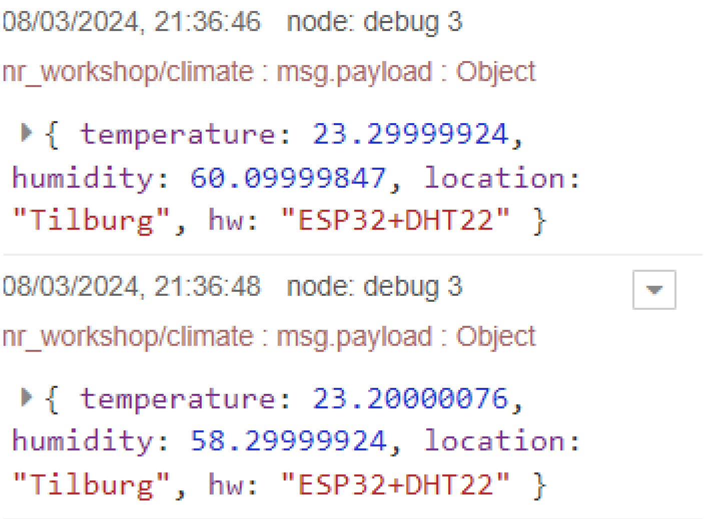


This is what was done and immediately we started receiving data. Now that we know our broker works we can begin sending our own data to it.

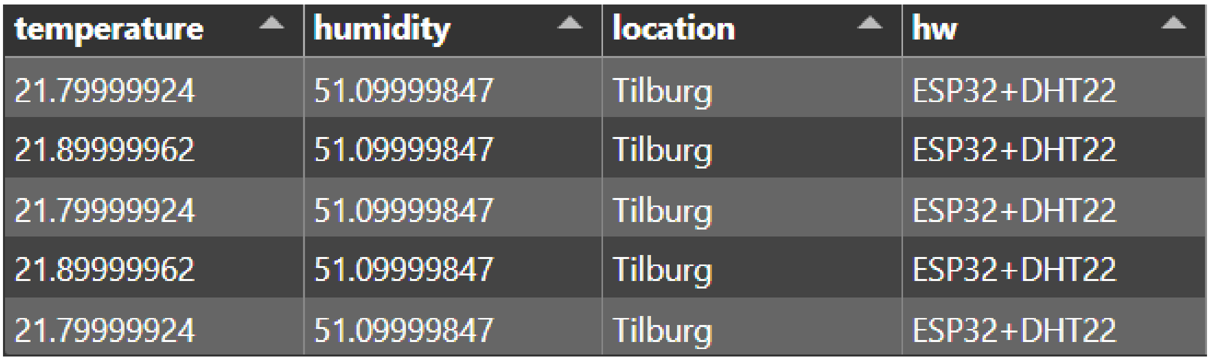


```
08/03/2024, 11:54:50   node: debug 3
nr_workshop/climate : msg.payload : Object
▶{ temperature: 23.2, humidity: 36,
location: "Eindhoven", hw: "Kevins
ESP32" }

08/03/2024, 11:55:47   node: debug 3
nr_workshop/climate : msg.payload : Object
▶{ temperature: 26.89999962,
humidity: 59.29999924, location:
"Youri", hw: "Siel's Device" }

08/03/2024, 11:55:54   node: debug 3
nr_workshop/climate : msg.payload : Object
▶{ temperature: 27.10000038,
humidity: 53.79999924, location:
"Youri", hw: "Siel's Device" }
```

We started by just reading data from the sensor using the already provided library to us**[1].** After it was confirmed that we are receiving data correctly, we need to connect it to our WIFI to send the data via our broker to the topic. We use te same topic as the other students are posting already. We take an old code that can send data to the MQTT topic and connect it with our program. This was not a difficult task. For me the most difficult task was founding out how I can transform this data into JSON format so I can send it to the topic. I used an additional library for this that was found here**[2].** This gave me insight on how to add data into a JSON object and then send this in the form of a string to the broker. After which we use the same blocks as above to see if we are getting the data and sure enough, we started receiving data. We confirmed that as well by following the table dashboard link provided.

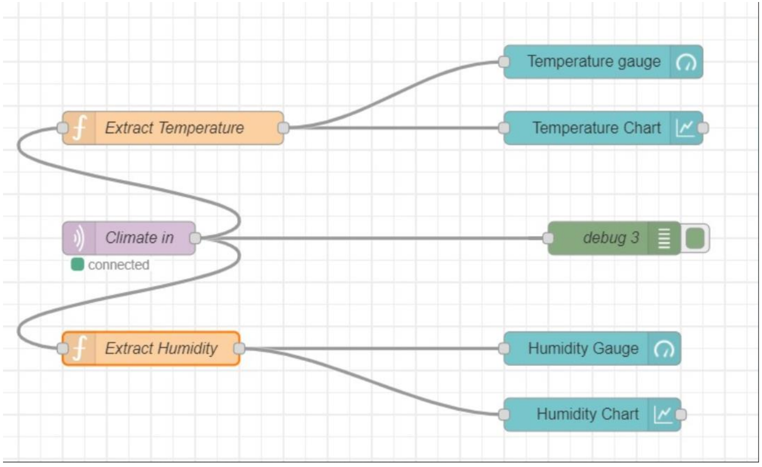


```
08/03/2024, 21:36:46   node: debug 3
nr_workshop/climate : msg.payload : Object
▶{ temperature: 23.29999924,
humidity: 60.09999847, location:
"Tilburg", hw: "ESP32+DHT22" }

08/03/2024, 21:36:48   node: debug 3
nr_workshop/climate : msg.payload : Object
▶{ temperature: 23.20000076,
humidity: 58.29999924, location:
"Tilburg", hw: "ESP32+DHT22" }
```

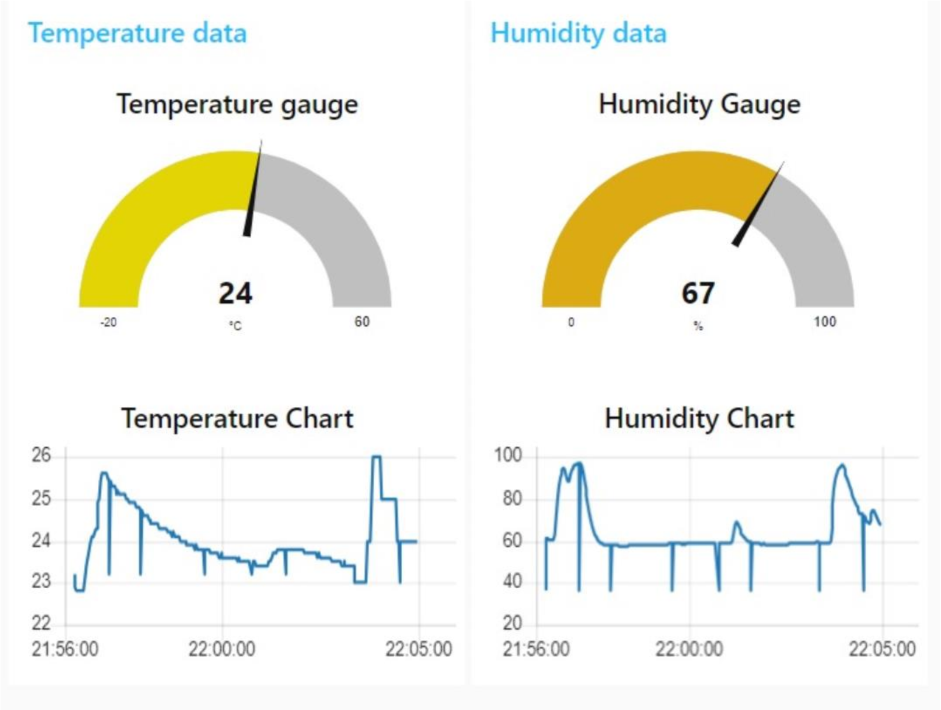


temperature ▲	humidity ▲	location ▲	hw ▲
21.79999924	51.09999847	Tilburg	ESP32+DHT22
21.89999962	51.09999847	Tilburg	ESP32+DHT22
21.79999924	51.09999847	Tilburg	ESP32+DHT22
21.89999962	51.09999847	Tilburg	ESP32+DHT22
21.79999924	51.09999847	Tilburg	ESP32+DHT22

The next step to showcase this data within a dashboard. The new flows looks something like this:



I added two function blocks to extract both the temperature and the humidity from the read MQTT topic and the showcase this is both a chart and a gauge. The purpose of the chart was to keep track of how the temperature changes over time and the gauge is to show the current reading of the two values. I group them in two different dashboard classes so they are neatly next to each other. Which then looks something like this.



I will include a link of the flows in JSON format with my submission to this challenge.

The primary thing I learned in this challenge was learning to work further with the JSON formatting system. I feel like this will be important in the coming weeks when we dive deeper into the technology. I also learned something about dashboarding and how they are grouped together to give the user a more friendly overlay to work with. Finally I learned a bit more about JavaScript and how simple functions that are already built in can help you achieve your goal.

3. Bibliography:

[1] - Thakur, M. R. (2019, January 25). *ESP32: DHT11/22 humidity temperature sensor interfacing example*. Circuits4you.com. https://circuits4you.com/2019/01/25/esp32-dht11-22-humidity-temperature-sensor-interfacing-example/

[2] - BenoitBlanchon. (n.d.). *Efficient JSON serialization for embedded C++*. ArduinoJson. https://arduinojson.org/