Project Summary

In this project we implemented a networked system consisting of 1 temperature sensor, 1 humidity sensor, 1 gateway and 1 server. The sensors connected to the gateway and sent information to the gateway. The gateway connected to the server and after monitoring the activity of the sensors, it sent the information to the server. On the server, the information received was printed on the console and displayed on 2 HTTP servers. We decided to use Python and the socket library for this project.

Our Solution Approach

We started with research about:

- Socket library in Python,
- TCP and UDP sensors,
- How to create a gateway and a server,
- What are the properties of the gateway and server?
- How can we send received data from server to http's?

After gathering some information, we decided to build a temperature sensor and a humidity sensor. The reason was that they were easier to implement than a gateway and a server. Once we were sure that the sensors were working correctly, we started to think 'how do we design a gateway'. After some research, we started to design our gateway step by step. First, we connected the temperature sensor to the gateway and when it was working correctly, we connected the humidity sensor to the gateway. Then we started researching and designing a server. When we connect the server to the gateway, we add some properties to the gateway and the server. The properties were the "TEMP OFF" signal when the temperature sensor does not send new messages after 3 seconds, "HUMIDITY OFF" signal when the humidity sensor does not send "ALIVE" message after 7 seconds, and HTTP to show the temperature and humidity sensors in a web page.

Encountered Problems and Solutions

We handle the "TEMP OFF" signal in a simple way. But when we tried the same method with the humidity sensor, it didn't work. The possible reason was that the humidity sensor sends 2 different messages to the gateway. When we check the elapsed time using the last message, the other message updates it and we can't get the time right. So we searched and found a new solution which was the settimeout(7) method. This method cancels the humidity sensor and at the end of the "HUMIDITY OFF" message sent to the gateway. Also, we firstly handle the HTTP server with http module in Python, but that was forbidden, so we tried to figure how can we handle that and found the solution by researching some resources.

Unresolved Issues

We couldn’t achieve the bonus part. We tried some methods but if we can’t store the last value on the sensor, our methods can’t send the value to the gateway.

Usage Explanation

In our project solution, there are 4 different python files which are “temp.py, humidity.py, gateway.py, server.py”.

The usage is as given:

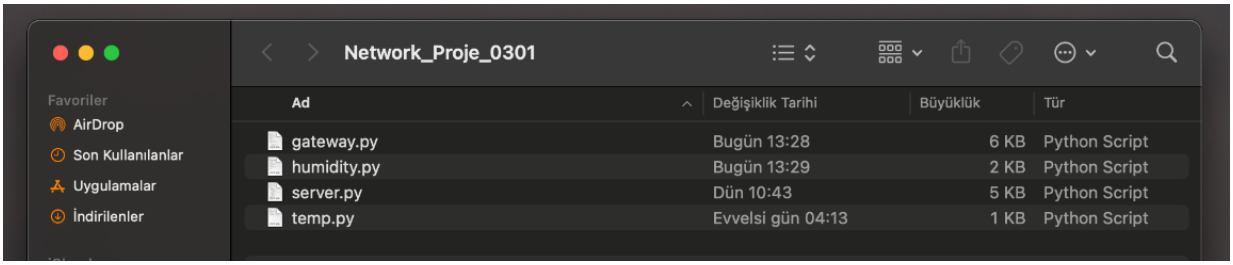
- Run the server.py
- Run the gateway.py
- Run the humidity.py
- Run the temp.py

When all the files are running go and see results on “http://localhost:8080/humidity” for humidity values and “http://localhost:8080/temperature” for the temperature values. Also, the values printed to the console. In the project file, there will be log files created for each python file. You can check it too.

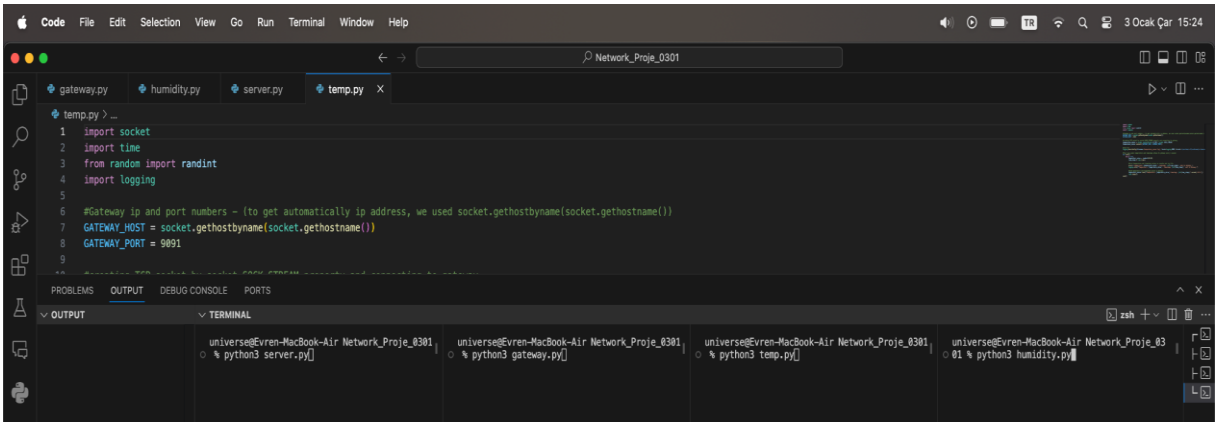
Details of Protocol (Between Server-Gateway)

On the server we receive the messages from the gateway. If the received message starts with "temperature" or "TEMP", it is stored in a global array named temperature_data, and if the received message starts with "humidity" or "HUM", it is stored in a global array named humidity_data,it is stored in a global array called humidity_data. After storing these received messages in arrays, we sent them to the HTTP according to “GET temperature” or “GET humidity” responses.

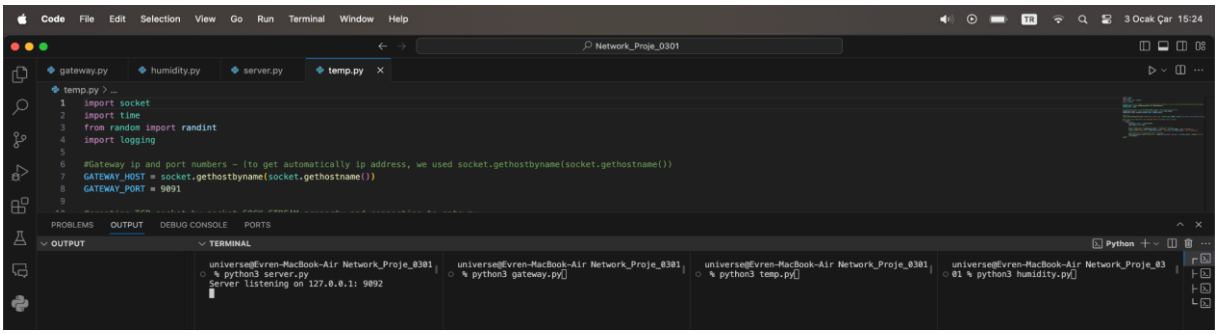
Screenshots



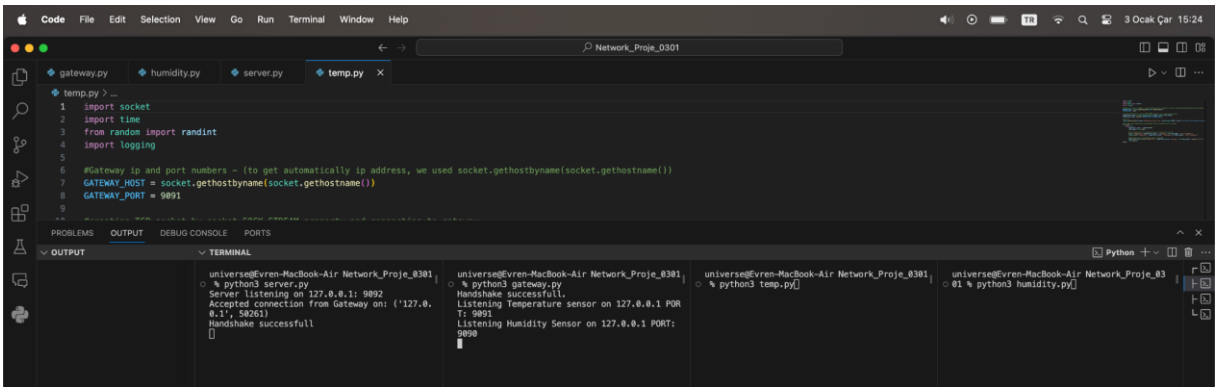
1- Initial Folder



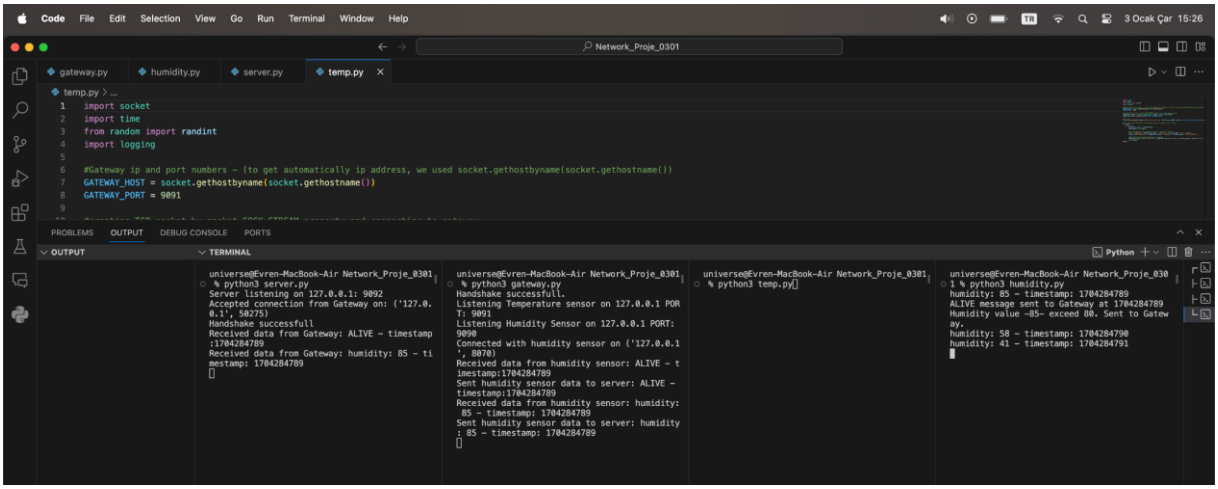
2- Console Commands



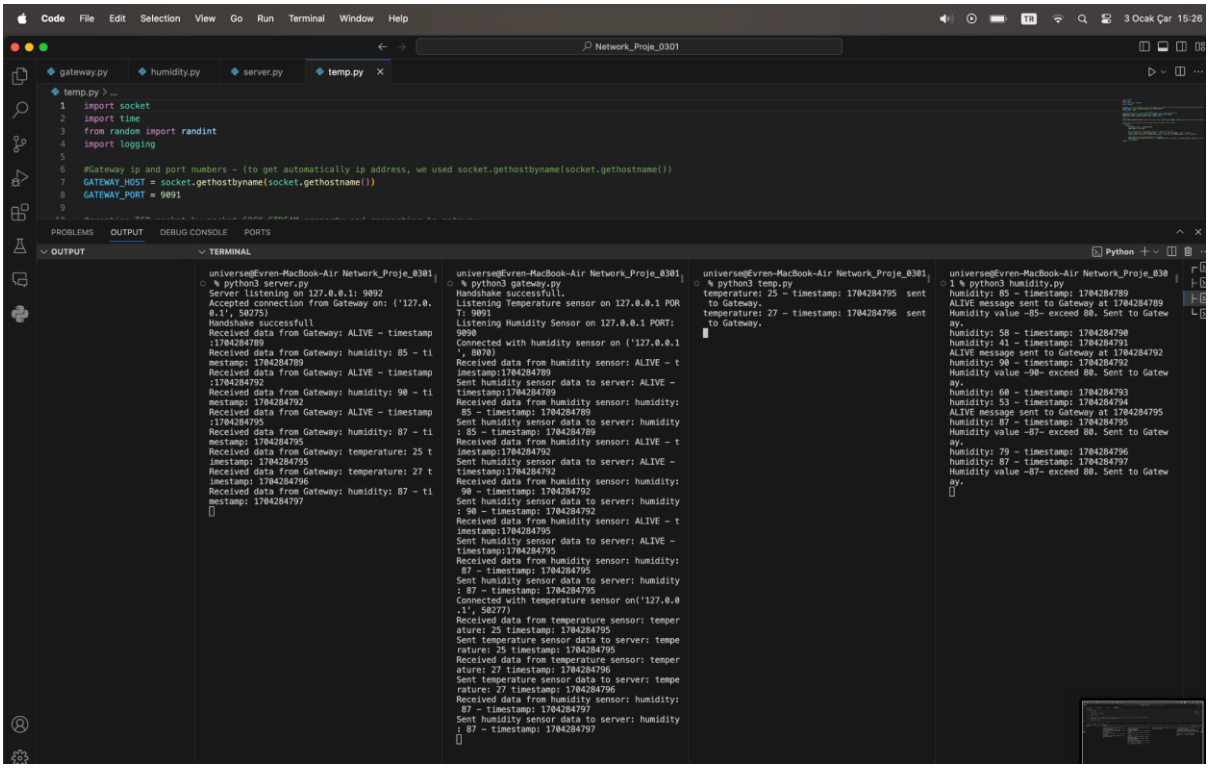
3- Run the Server



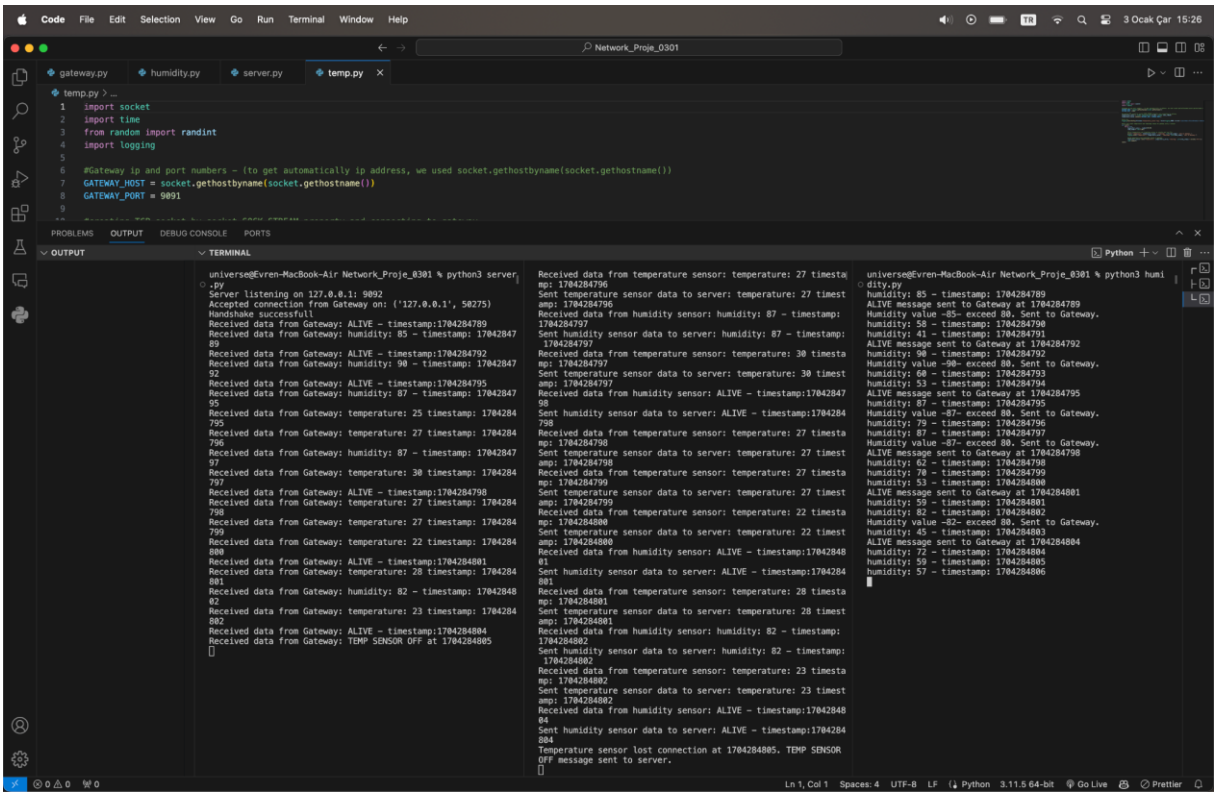
4- Run the Gateway



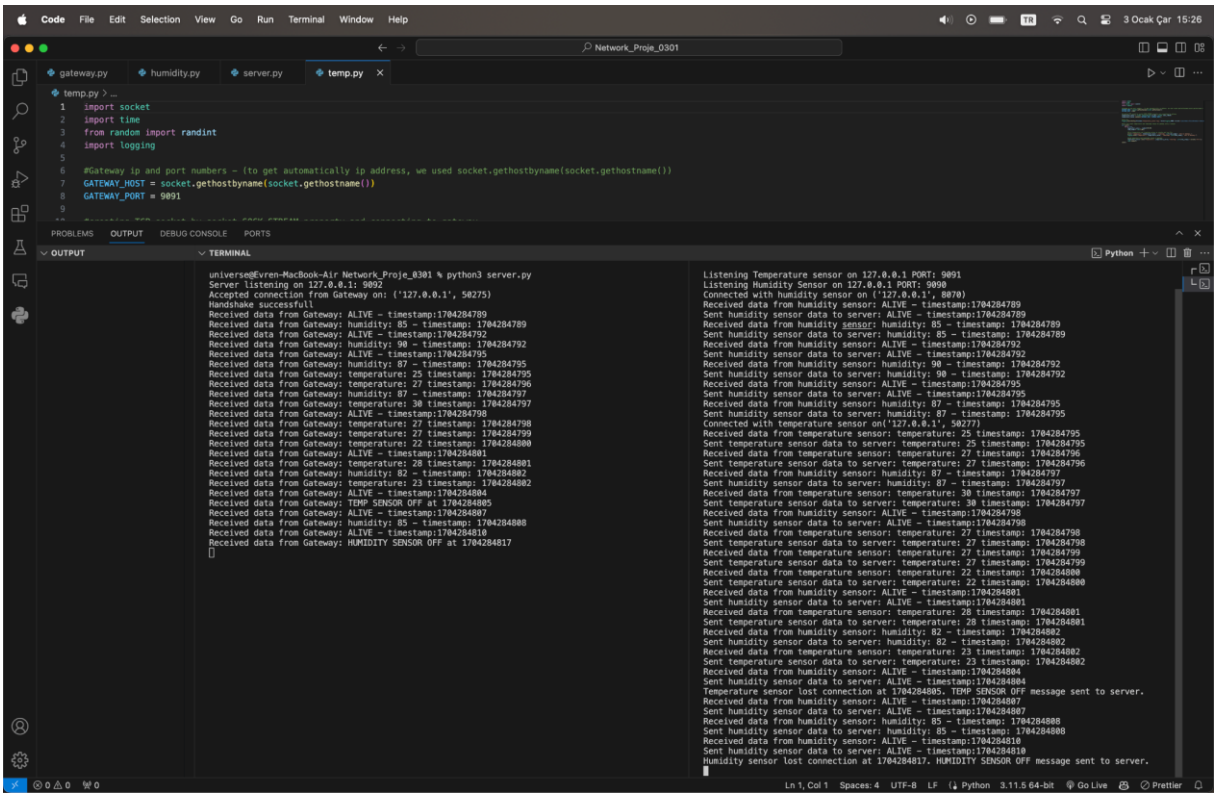
5- Run the Humidity Sensor



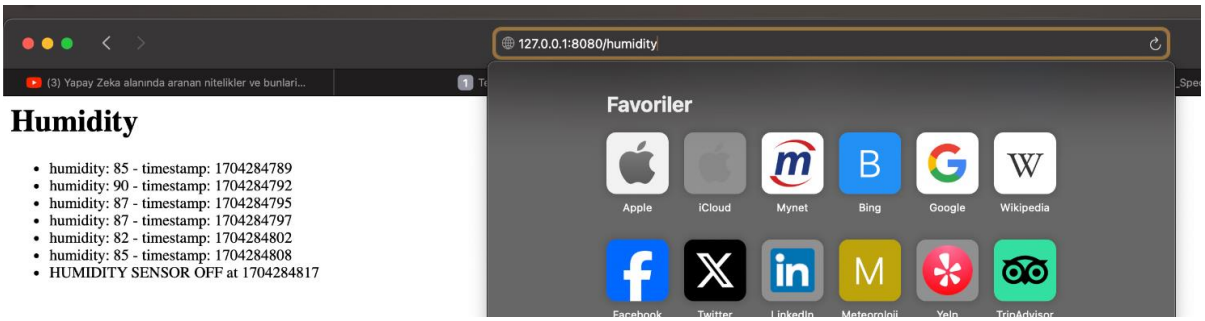
6- Run the Temperature Sensor



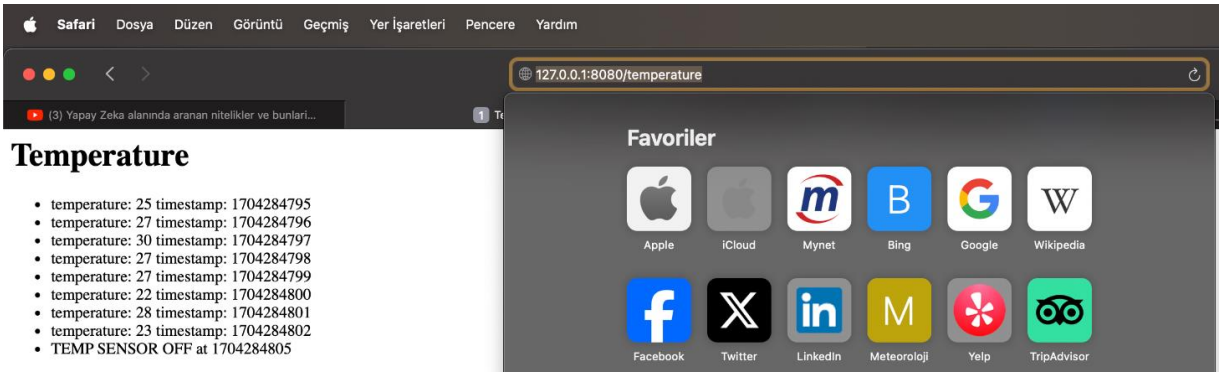
7- Close the Temperature Sensor (see TEMP SENSOR OFF msg on server console)



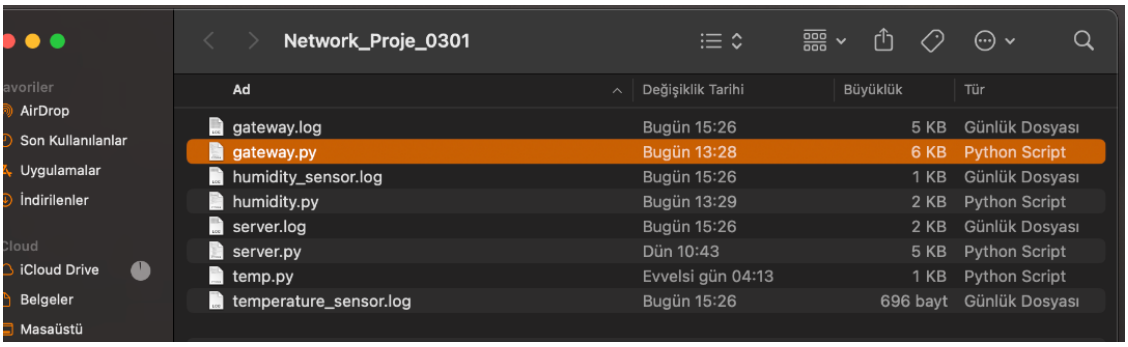
8- Close the Humidity Sensor (HUMIDITY SENSOR OFF msg on server console)



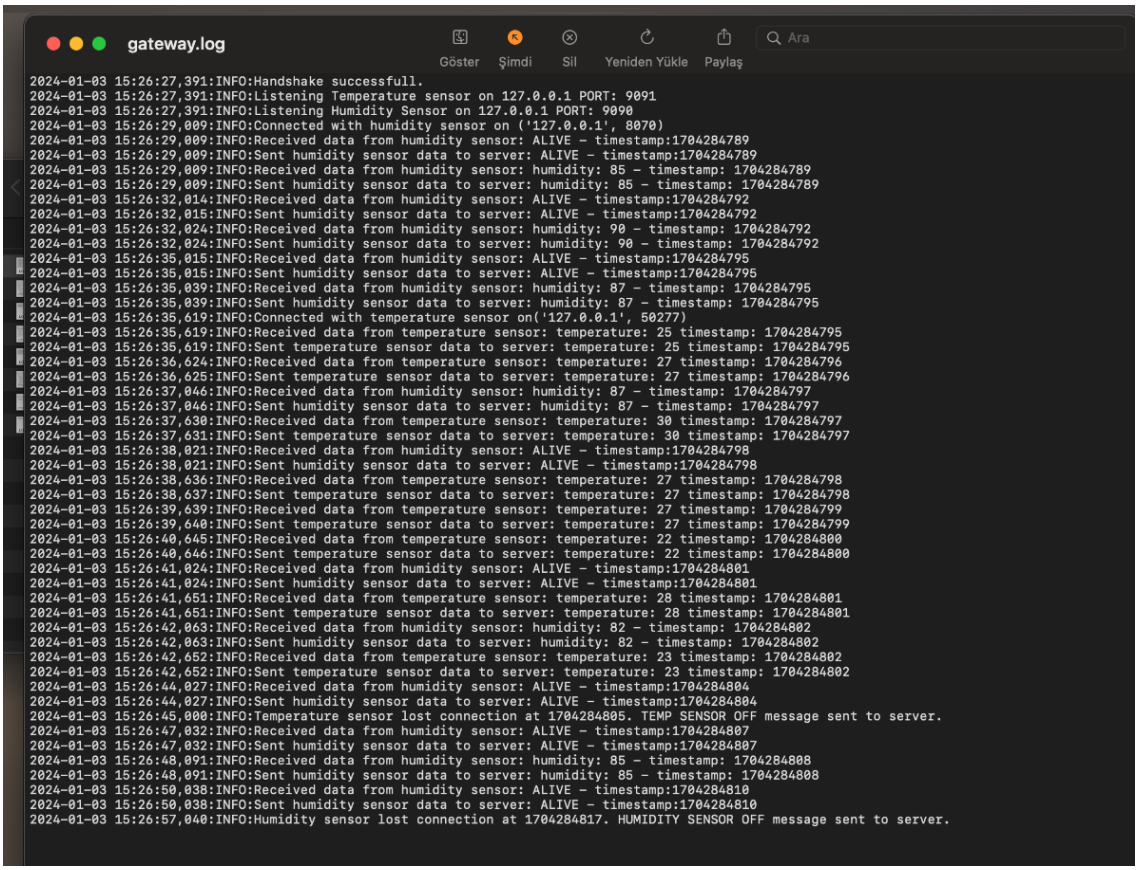
9- See the Results from localhost:8080/humidity



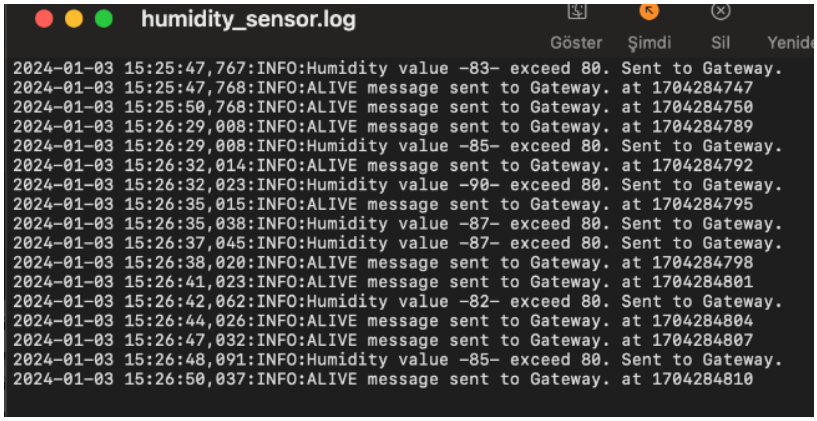
10- See the Results from localhost:8080/temperature



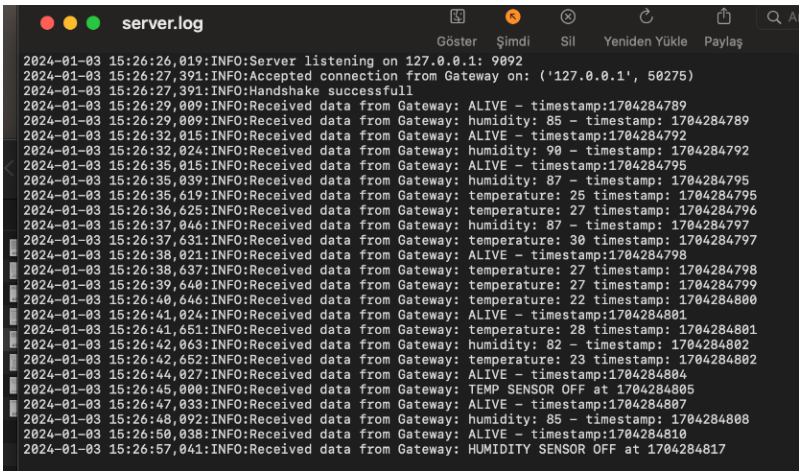
11- Log Files Created



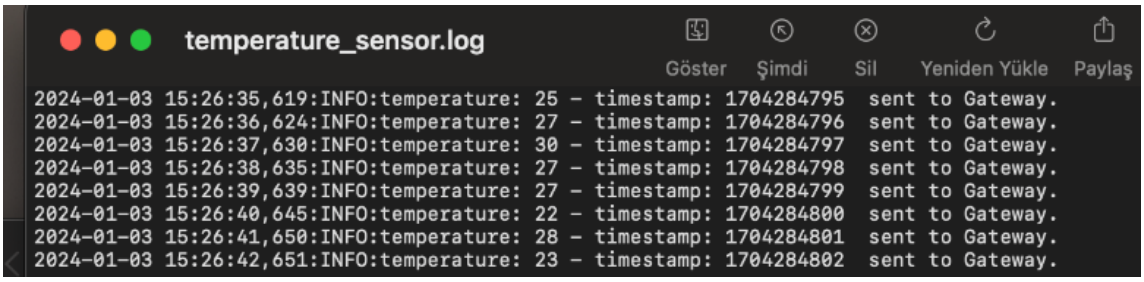
12- Log file of Gateway



13- Log file of Humidity Sensor



14- Log file of Server



15- Log File of Temperature Sensor