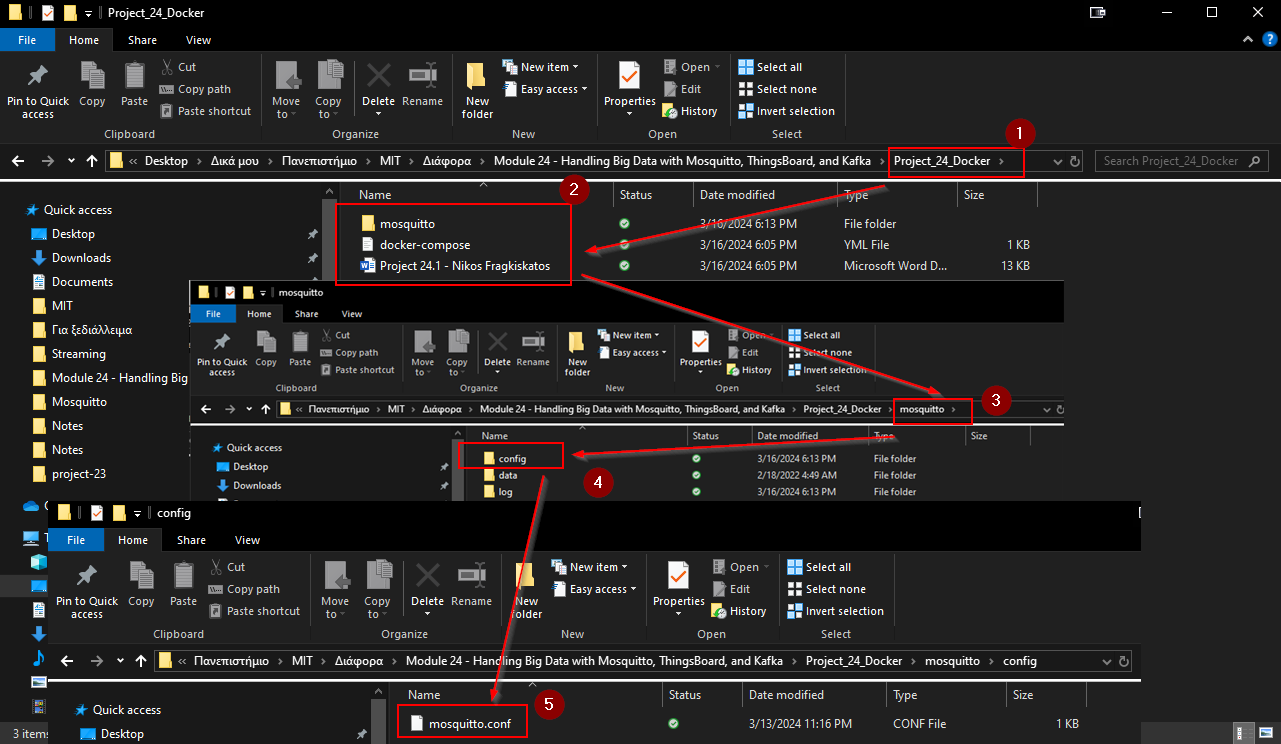
Screenshot 1: Folder structure



Screenshot 2: Ran the command to create the container. Validated that the container is running

A screenshot of a computer

Description automatically generated

Screenshot 3: Installed already the paho-mqtt library through a terminal

A computer screen with text

Description automatically generated with medium confidence

Screenshot 4: Made the required folders in the root directory

A screen shot of a computer

Description automatically generated

Screenshot 5: Created the folders for the MQTT ThingsBoard

A screenshot of a computer

Description automatically generated

Screenshot 6: Ran the docker compose up command in a terminal to start the ThingsBoard server on docker. Note: I had to stop the airflow container (it was a project from a different file), because its webserver was occupying port 8080:8080 as you can see from the picture.

A screenshot of a computer

Description automatically generated

Screenshot 7: Added the humidity sensor. Also, showing a summary of the project setup so far

A computer screen shot of a computer

Description automatically generated

Screenshot 8: Running the code, successfully publishing temperature and humidity readings

A screen shot of a computer

Description automatically generated

Screenshot 9: Logged into the ThingsBoard server

A screenshot of a computer

Description automatically generated

Screenshot 10: Seeing the readings on the DHT11 Demo Device. Those readings change every three seconds

A screenshot of a computer

Description automatically generated

Screenshot 11: Created a Firebase project named module24Project. Created a Realtime Database in it and added a temperature field

A screenshot of a computer

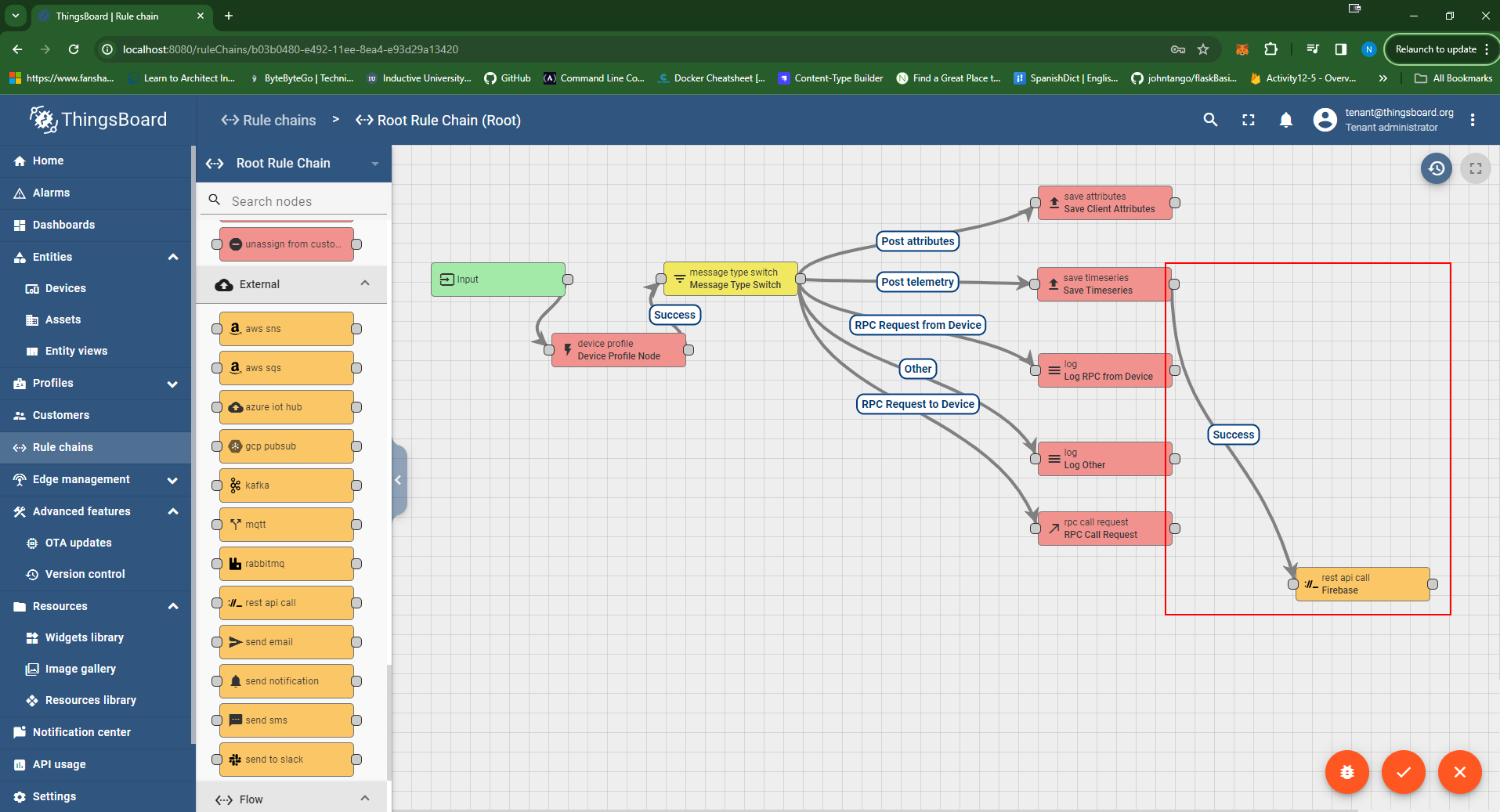
Description automatically generated

Screenshot 12: Configured the Root Rule Chain node

A screenshot of a computer

Description automatically generated

Screenshot 13: Connected the Firebase through a rest api call, by creating a success link from Save Timeseries node



Screenshot 14: Capturing the entries successfully on the Firebase side (showing the content of a couple of the values also)

