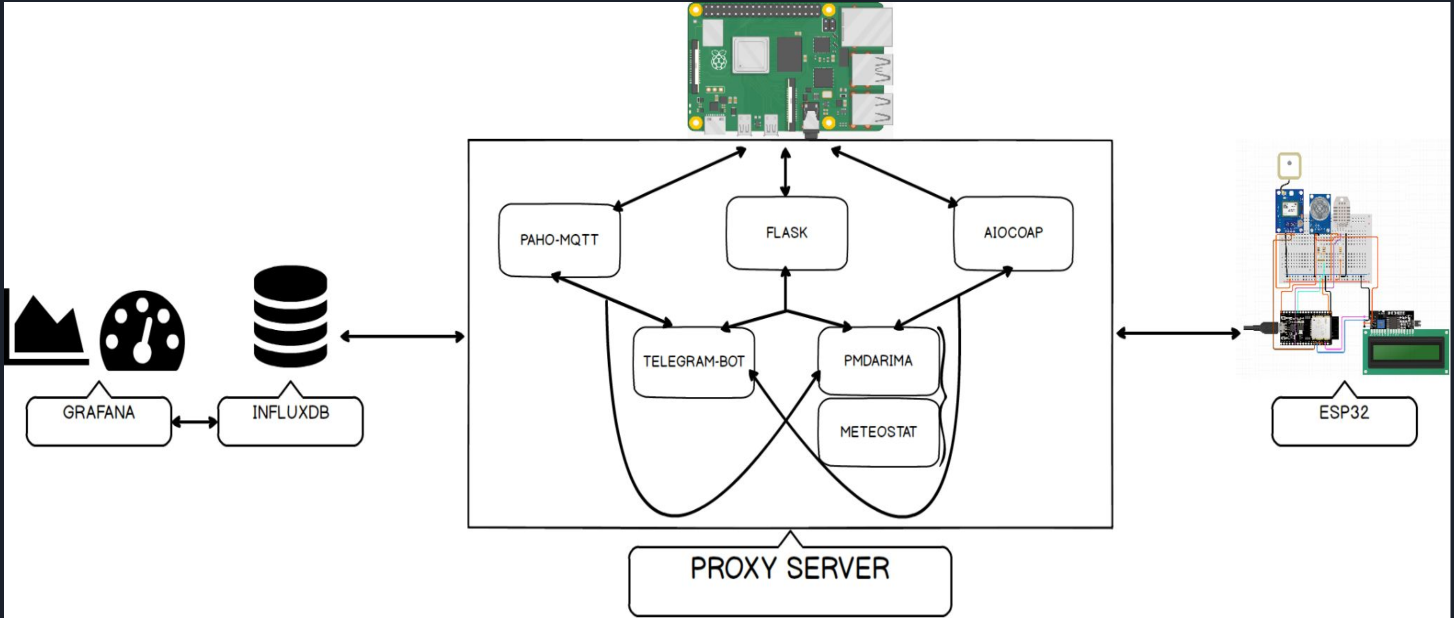
IoT Project 2022

Primiano Arminio Cristino: primiano.cristino@studio.unibo.it
Francesco Palmisano: francesco.palmisano2@studio.unibo.it

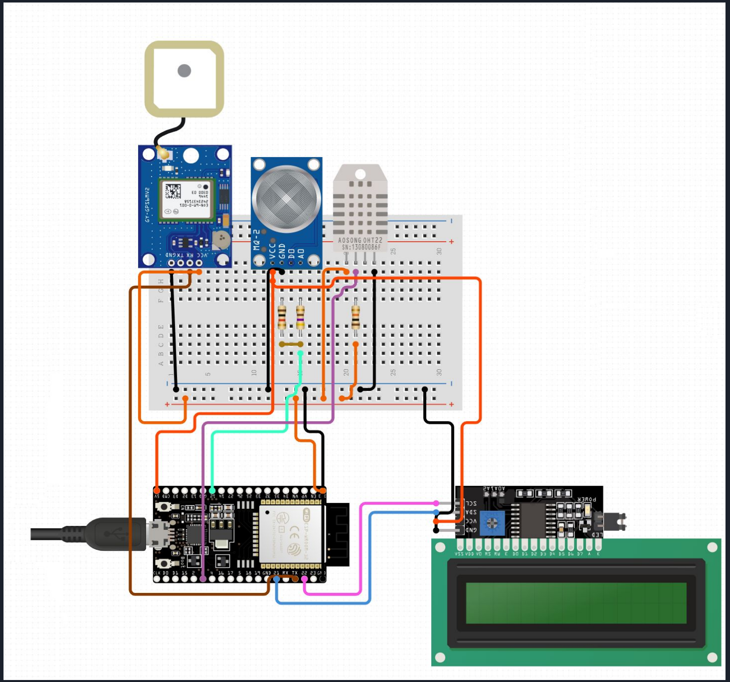
Outlines

Introduction

Sensory data from ESP32 devices are collected by a server proxy and then stored on a database and eventually visualized.



IoT Devices



Each ESP32 device uses several chips:

- DHT22 for temperature and humidity
- MQ-2 for raw gas concentration
- NEO-6M for GPS coordinates
- 16x2 LCD Screen

Communication design

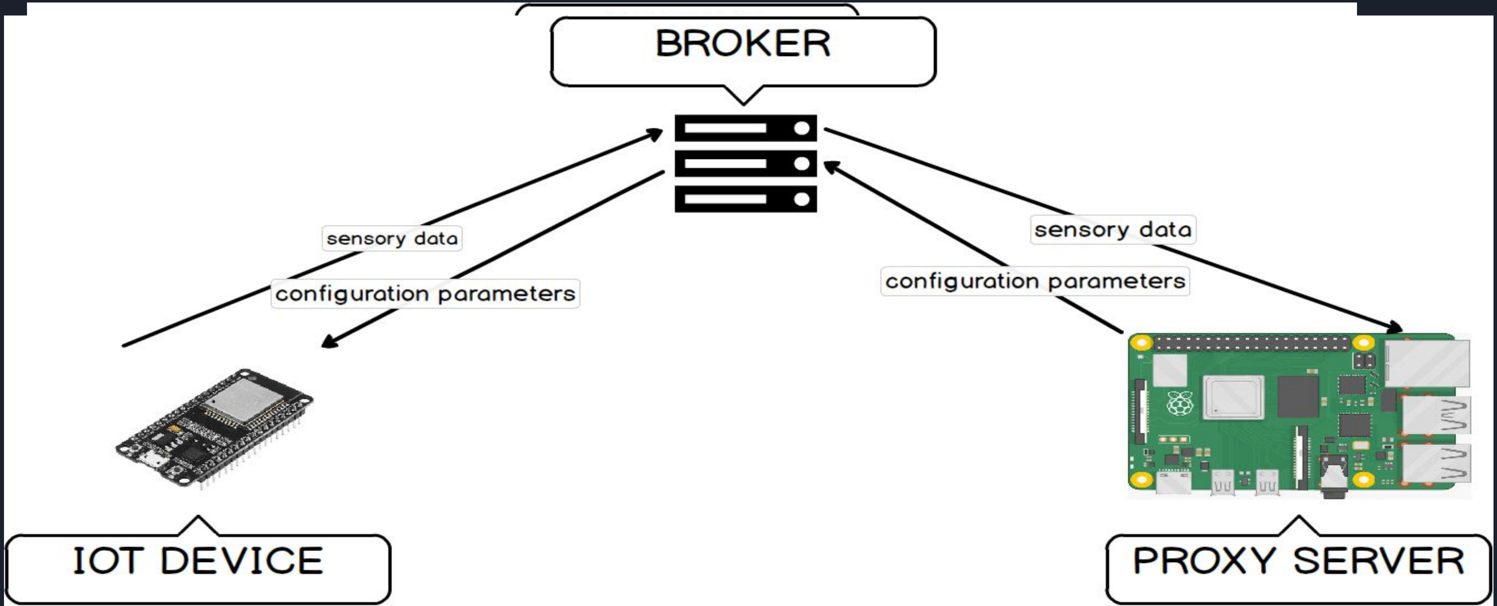
Each ESP32 device communicates via 3 protocols:

- MQTT
- ⅓ CoAP
- HTTP

Moreover, IoT devices send their sensory data but also receive configuration data such as sample frequency an min/max gas concentration value from the proxy server.

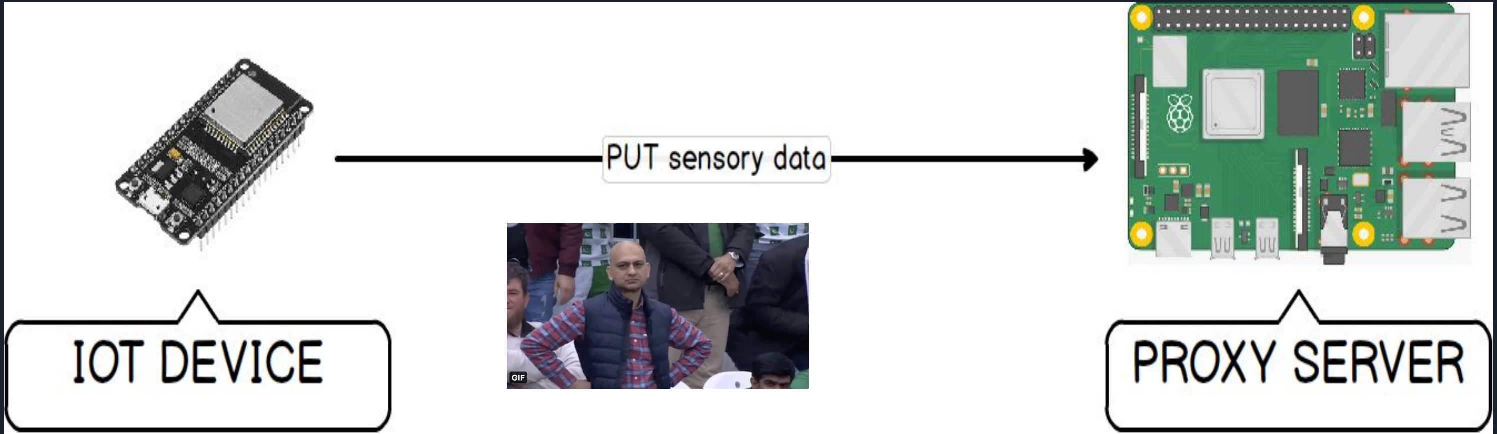
MQTT

IoT devices and the proxy server communicate with an external broker (broker.emqx.io).



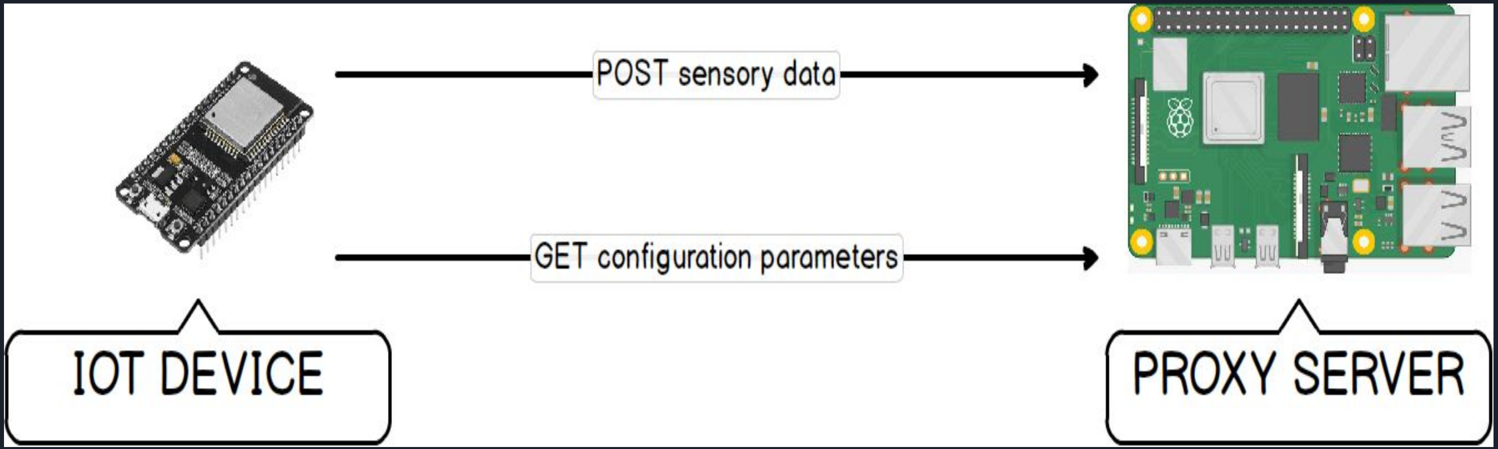
COAP

The CoAP arduino library is not completed, so ESP32 device uses only the PUT method to upload sensory data in the proxy server.



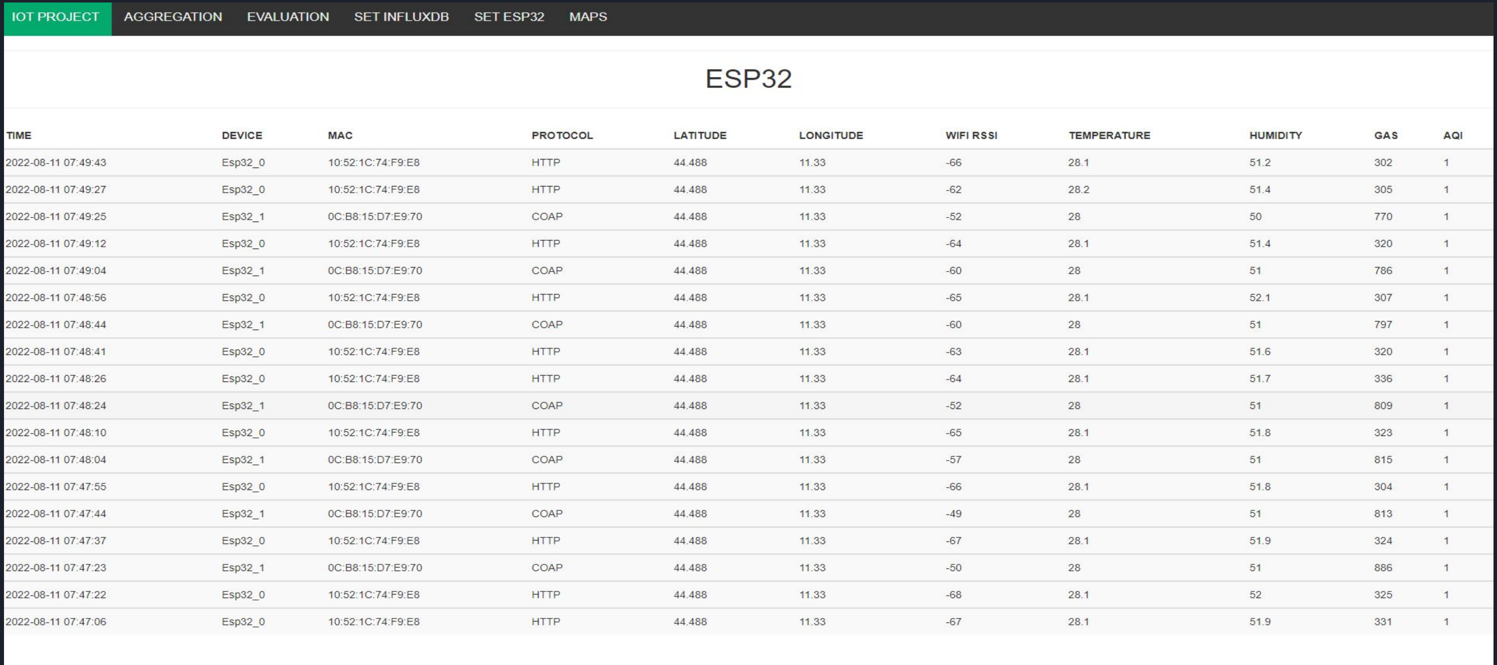
HTTP

The GET HTTP request works both on HTTP and CoAP.



Web application

The proxy server provides also a web application to monitor the IoT devices’ behaviour

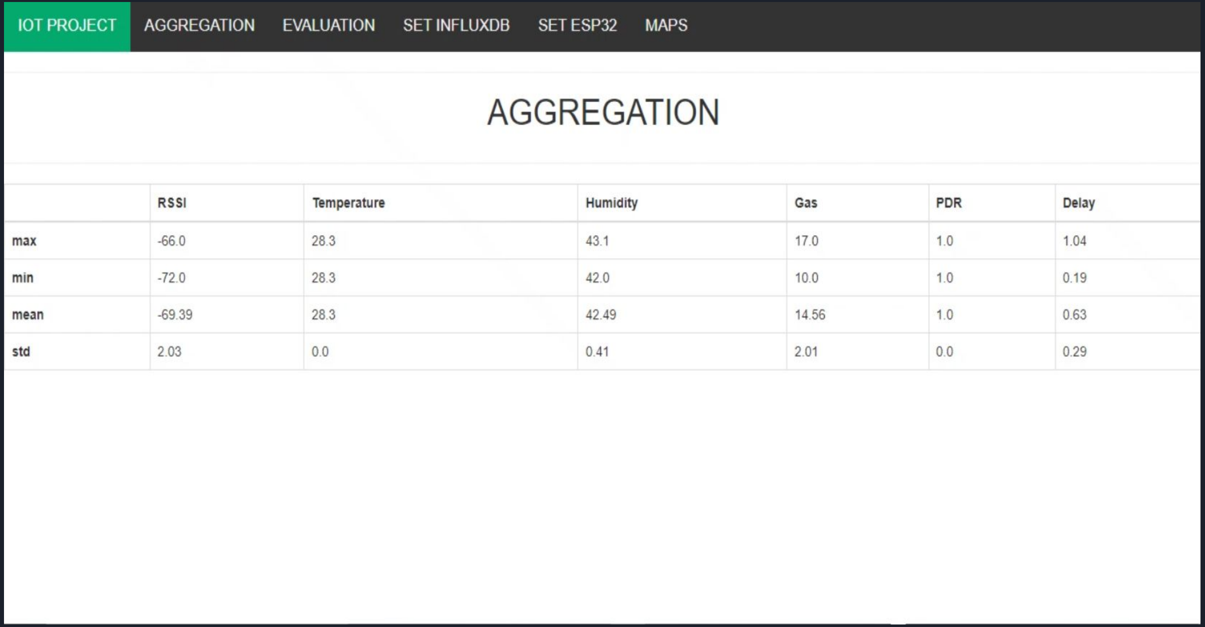


ESP32

TIME	DEVICE	MAC	PROTOCOL	LATITUDE	LONGITUDE	WIFI RSSI	TEMPERATURE	HUMIDITY	GAS	AQI
2022-08-11 07:49:43	Esp32_0	10:52:1C:74:F9:E8	HTTP	44.488	11.33	-66	28.1	51.2	302	1
2022-08-11 07:49:27	Esp32_0	10:52:1C:74:F9:E8	HTTP	44.488	11.33	-62	28.2	51.4	305	1
2022-08-11 07:49:25	Esp32_1	0C:B8:15:D7:E9:70	COAP	44.488	11.33	-52	28	50	770	1
2022-08-11 07:49:12	Esp32_0	10:52:1C:74:F9:E8	HTTP	44.488	11.33	-64	28.1	51.4	320	1
2022-08-11 07:49:04	Esp32_1	0C:B8:15:D7:E9:70	COAP	44.488	11.33	-60	28	51	786	1
2022-08-11 07:48:56	Esp32_0	10:52:1C:74:F9:E8	HTTP	44.488	11.33	-65	28.1	52.1	307	1
2022-08-11 07:48:44	Esp32_1	0C:B8:15:D7:E9:70	COAP	44.488	11.33	-60	28	51	797	1
2022-08-11 07:48:41	Esp32_0	10:52:1C:74:F9:E8	HTTP	44.488	11.33	-63	28.1	51.6	320	1
2022-08-11 07:48:26	Esp32_0	10:52:1C:74:F9:E8	HTTP	44.488	11.33	-64	28.1	51.7	336	1
2022-08-11 07:48:24	Esp32_1	0C:B8:15:D7:E9:70	COAP	44.488	11.33	-52	28	51	809	1
2022-08-11 07:48:10	Esp32_0	10:52:1C:74:F9:E8	HTTP	44.488	11.33	-65	28.1	51.8	323	1
2022-08-11 07:48:04	Esp32_1	0C:B8:15:D7:E9:70	COAP	44.488	11.33	-57	28	51	815	1
2022-08-11 07:47:55	Esp32_0	10:52:1C:74:F9:E8	HTTP	44.488	11.33	-66	28.1	51.8	304	1
2022-08-11 07:47:44	Esp32_1	0C:B8:15:D7:E9:70	COAP	44.488	11.33	-49	28	51	813	1
2022-08-11 07:47:37	Esp32_0	10:52:1C:74:F9:E8	HTTP	44.488	11.33	-67	28.1	51.9	324	1
2022-08-11 07:47:23	Esp32_1	0C:B8:15:D7:E9:70	COAP	44.488	11.33	-50	28	51	886	1
2022-08-11 07:47:22	Esp32_0	10:52:1C:74:F9:E8	HTTP	44.488	11.33	-68	28.1	52	325	1
2022-08-11 07:47:06	Esp32_0	10:52:1C:74:F9:E8	HTTP	44.488	11.33	-67	28.1	51.9	331	1

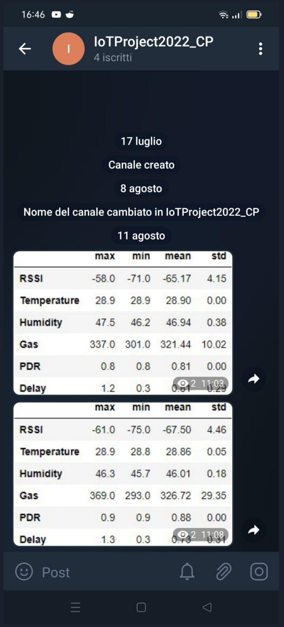
Data Aggregation

The proxy server computes some statistics over data and send them to a telegram channel.



AGGREGATION

	RSSI	Temperature	Humidity	Gas	PDR	Delay
max	-66.0	28.3	43.1	17.0	1.0	1.04
min	-72.0	28.3	42.0	10.0	1.0	0.19
mean	-69.39	28.3	42.49	14.56	1.0	0.63
std	2.03	0.0	0.41	2.01	0.0	0.29



IoTProject2022_CP
4 iscritti

17 luglio

Canale creato

8 agosto

Nome del canale cambiato in IoTProject2022_CP

11 agosto

	max	min	mean	std
RSSI	-58.0	-71.0	-65.17	4.15
Temperature	28.9	28.9	28.90	0.00
Humidity	47.5	46.2	46.94	0.38
Gas	337.0	301.0	321.44	10.02
PDR	0.8	0.8	0.81	0.00
Delay	1.2	0.3	0.81	0.29

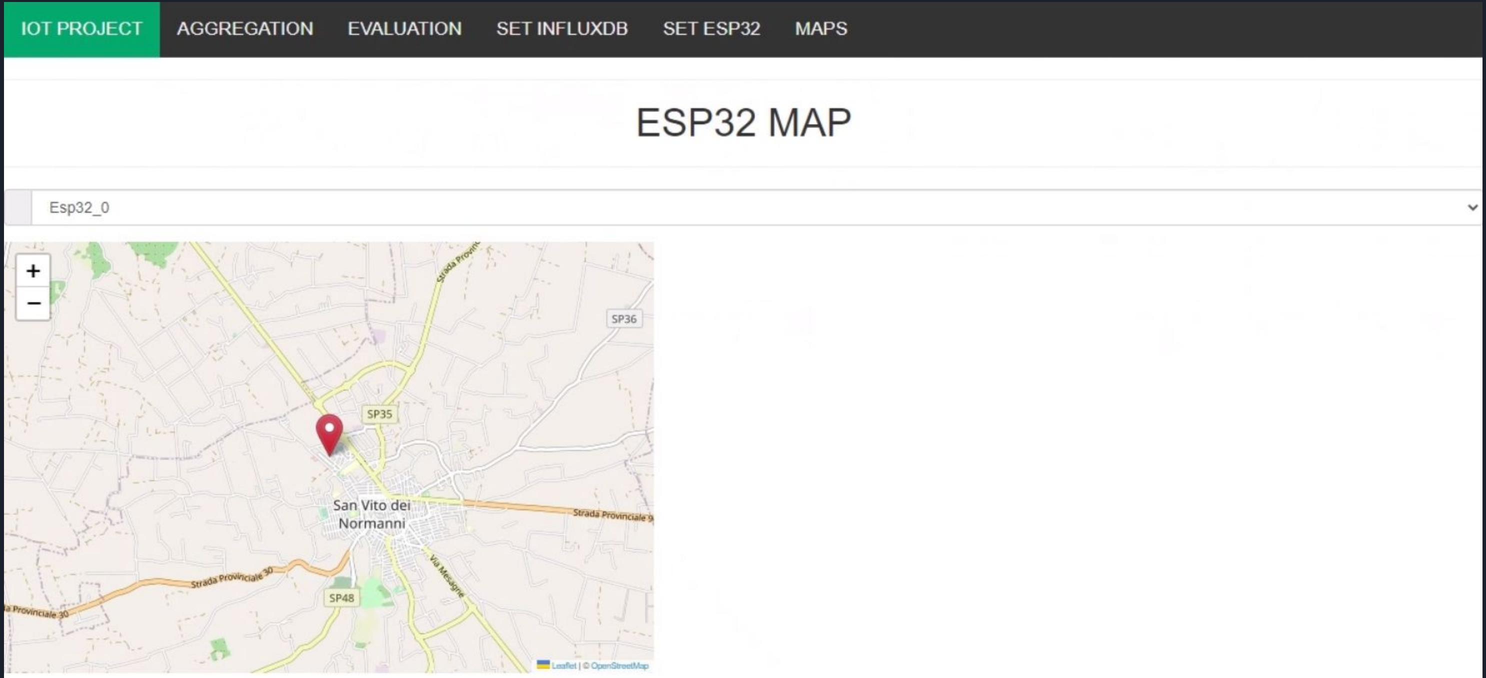
2 11:03

	max	min	mean	std
RSSI	-61.0	-75.0	-67.50	4.46
Temperature	28.9	28.8	28.86	0.05
Humidity	46.3	45.7	46.01	0.18
Gas	369.0	293.0	326.72	29.35
PDR	0.9	0.9	0.88	0.00
Delay	1.3	0.3	0.73	0.31

2 11:08

Open Street Map

The proxy server monitors also the GPS position of IoT devices.

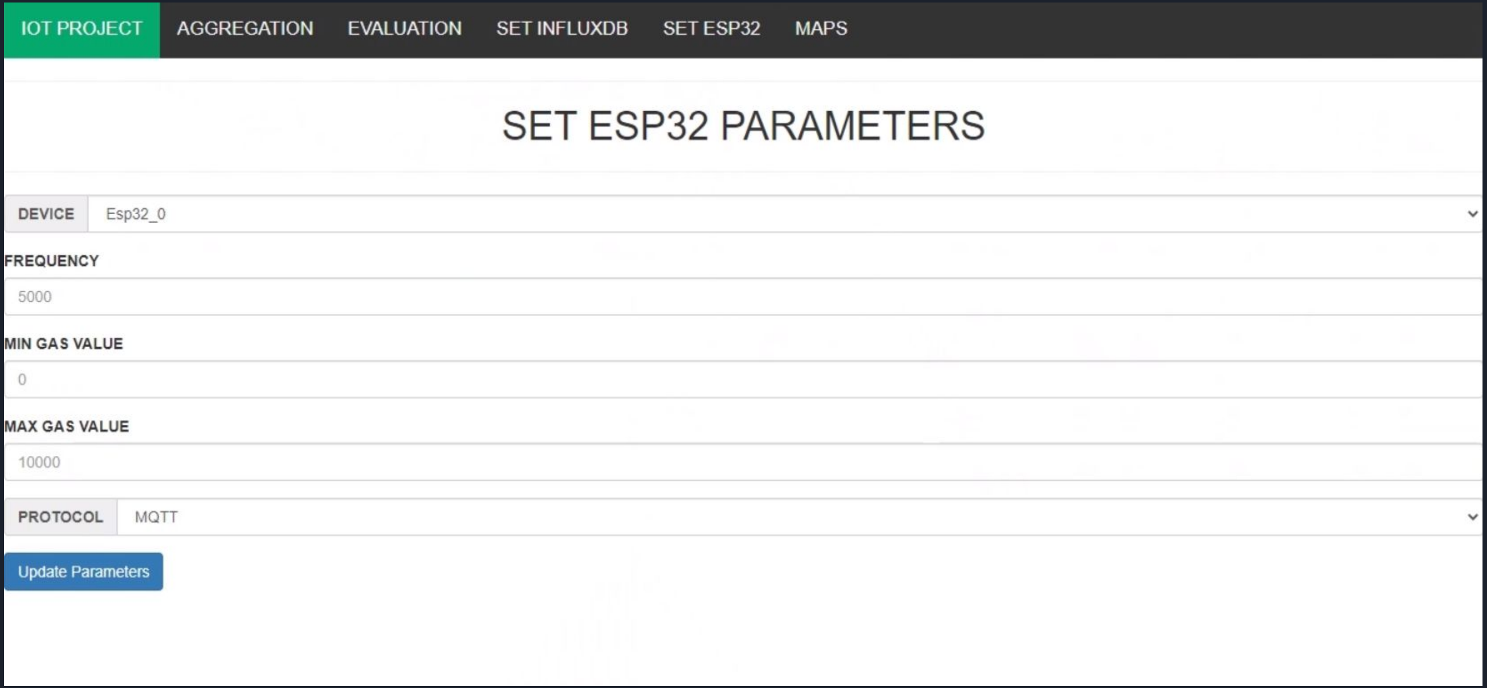


ESP32 MAP

Esp32_0

Parameters Configuration

Each IoT device has an own configuration that can be customized by web application.



SET ESP32 PARAMETERS

DEVICE: Esp32_0

FREQUENCY: 5000

MIN GAS VALUE: 0

MAX GAS VALUE: 10000

PROTOCOL: MQTT

Database

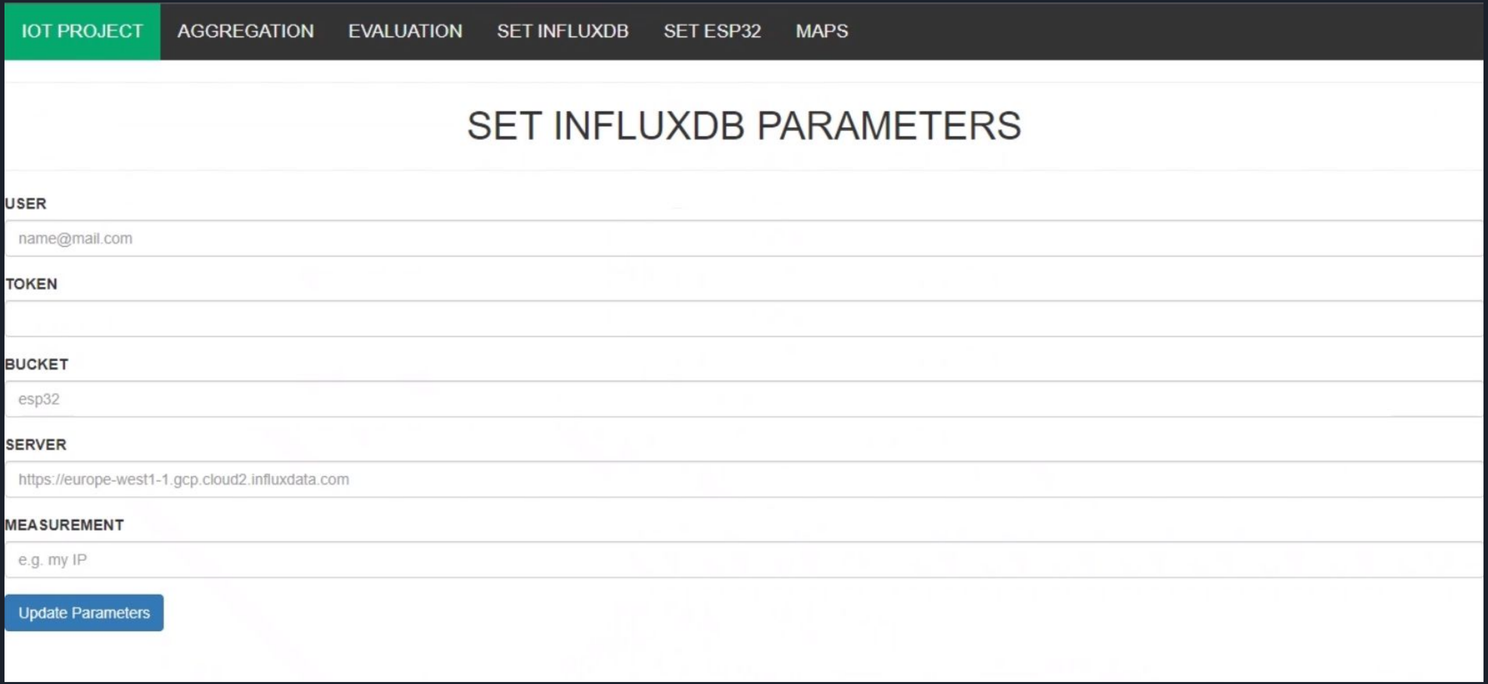
The data collected by the proxy server are stored into an InfluxDB bucket.

However we discovered some issues:

- InfluxDB is not consistent.
- InfluxDB is slow at writing data.

InfluxDB Configuration

Anyways, the Influx configuration can be changed at any time via web application.



SET INFLUXDB PARAMETERS

USER

name@mail.com

TOKEN

BUCKET

esp32

SERVER

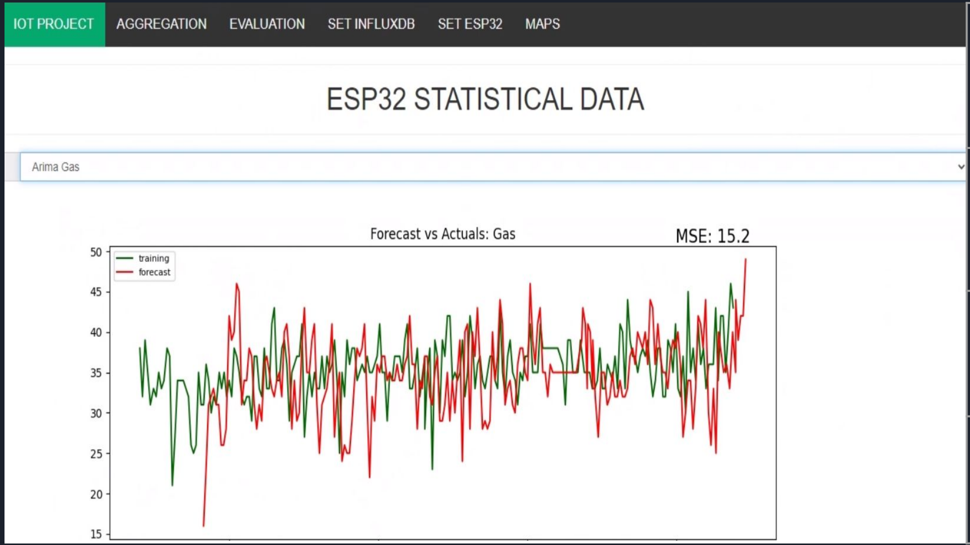
https://europe-west1-1.gcp.cloud2.influxdata.com

MEASUREMENT

e.g. my IP

Forecasting

- To predict data we first decided to use an ARIMA model to predict sensory data. However we discovered that it doesn't fit very well with sensory data, especially with Meteostat data, due to the presence of seasonality. So we decided to use a SARIMA-based model.
- Data prediction graphs are shown by web application, just as Grafana dashboard.



ESP32 STATISTICAL DATA

Arima Gas

Indoor value	MSE	Conf. int.
Temperature	0	[27.74, 27.92]
Humidity	0.016	[38.67, 39.73]
Gas	9.2	[103.52, 203.20]

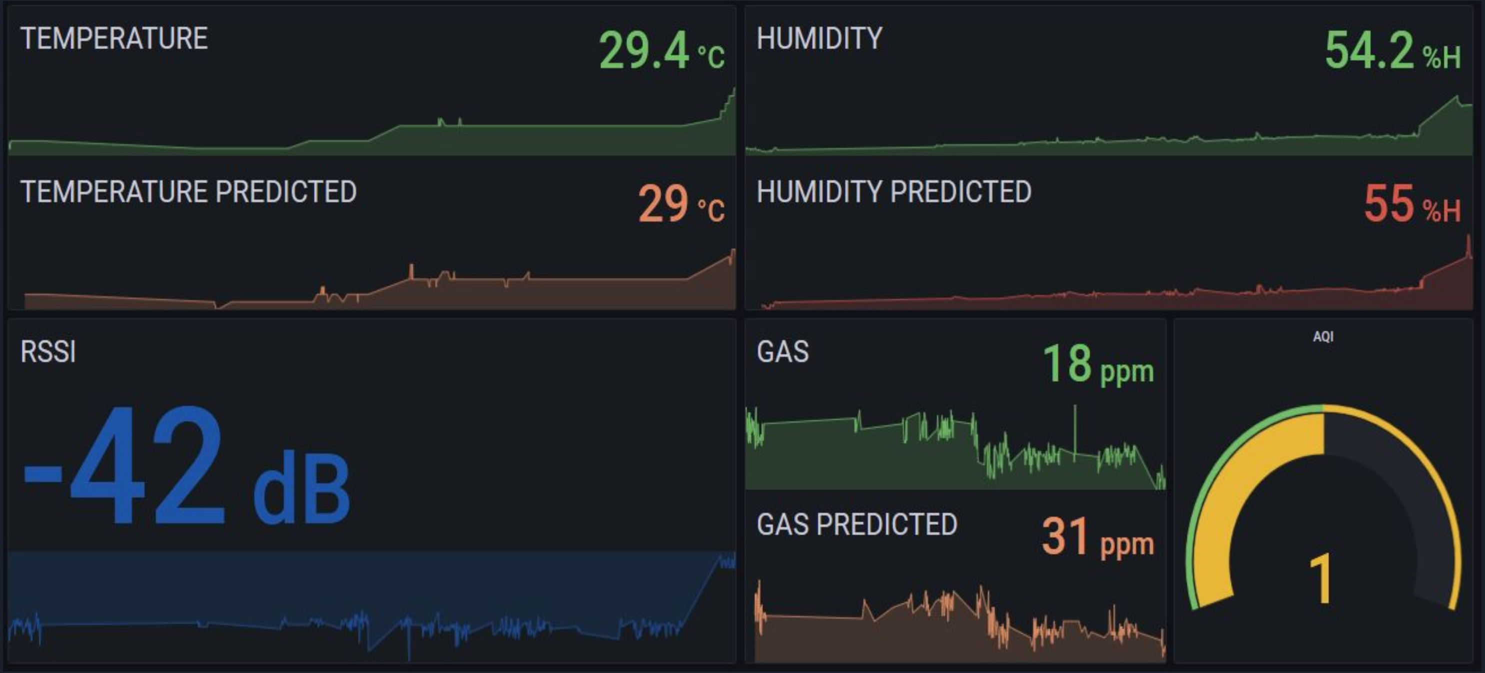
Results: data evaluation

SAS

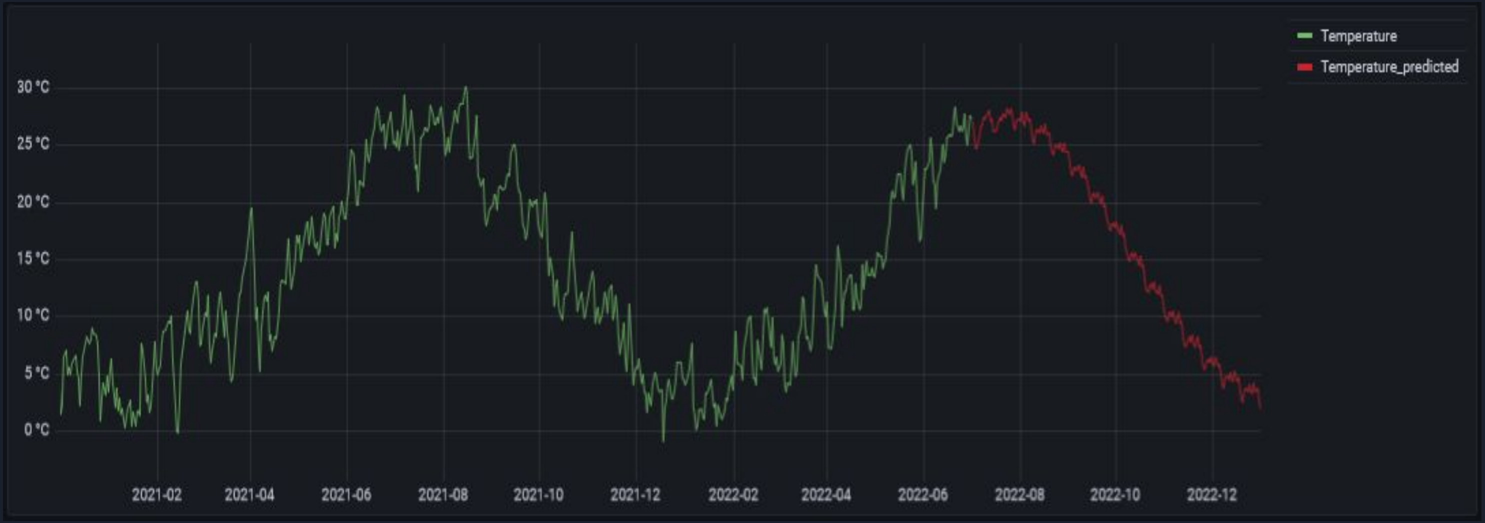
Indoor value	MSE	Confidence interval
Temperature	0	[27.74, 27.92]
Humidity	0.016	[38.67, 39.73]
Gas	9.2	[103.52, 203.20]

Grafana Dashboard

An alert is triggered when the AQI value is above 0.



Meteostat dashboard



Results: protocol evaluation

- HTTP and CoAP seem to have a similar behaviour. MQTT has a slightly higher delay than the others due to the communication with an external broker.
- Although we performed several tests on different range distance, the PDR didn’t change so much.

/	RSSI = [-89db, -79db]			RSSI = [-77db, -73db]			RSSI = [-42db, -26db]		
/	HTTP	CoAP	MQTT	HTTP	CoAP	MQTT	HTTP	CoAP	MQTT
Delay	0.64	0.56	1.13	0.55	0.54	0.84	0.57	0.59	0.82
PDR	0.95	1.0	1.0	0.95	1.0	1.0	1.0	0.99	1.0

DEMO