

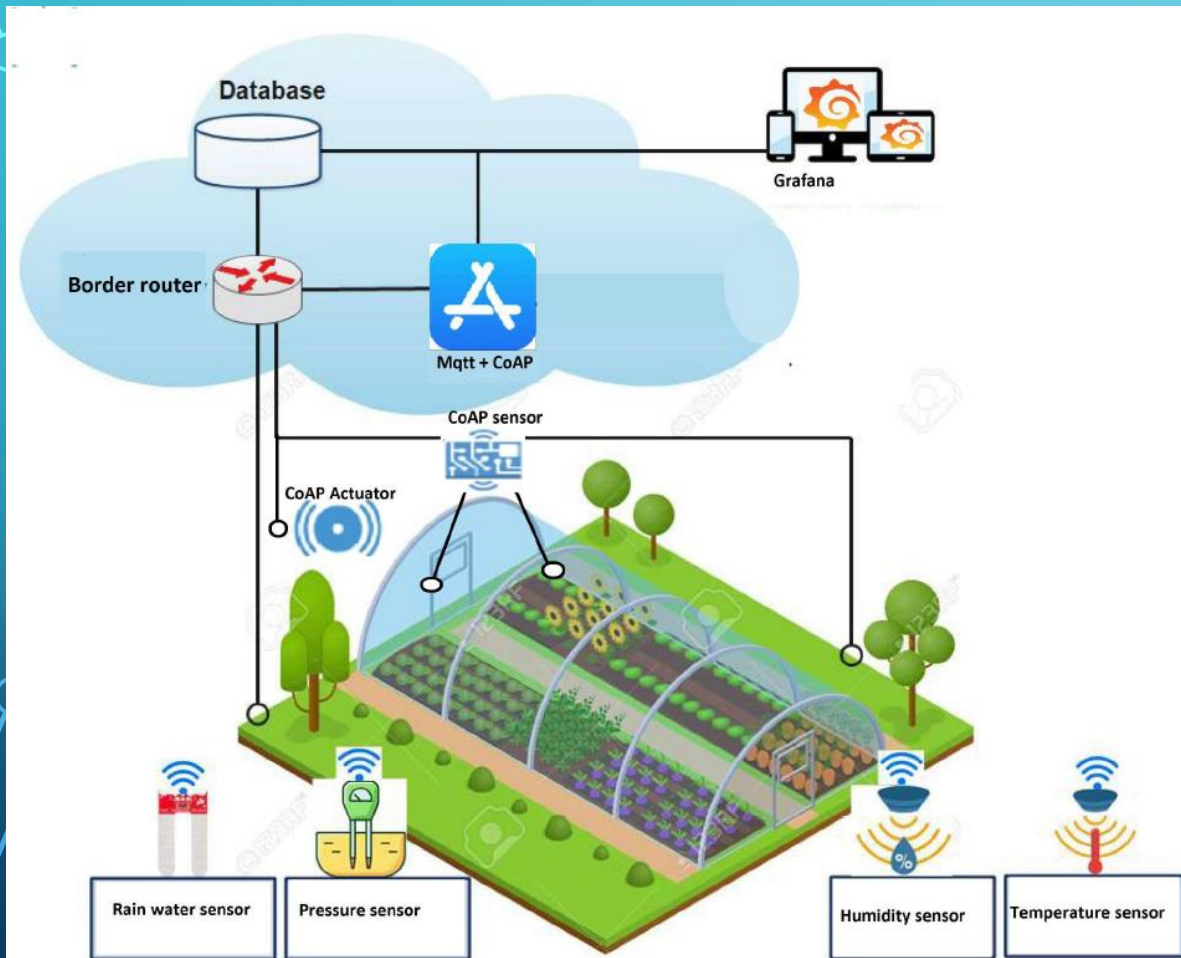
# INTERNET OF THINGS FOR SMART AGRICULTURE

IOT PROJECT 21/22

Giuseppe Aniello – 643032 – [g.aniello@studenti.unipi.it](mailto:g.aniello@studenti.unipi.it)

Leonard Bellizzi – 643019 – [l.bellizzi@studenti.unipi.it](mailto:l.bellizzi@studenti.unipi.it)

# PROJECT GOAL



- Build and set up an automatic system to protect grapes.
- System core is composed of two mechanical covers that are closed when bad weather comes in.
- The two covers are side-located to grape marquees and both have a 90° opening radius.

# PROJECT COMPONENTS

- MQTT
- COAP
- GRAFANA

# MQTT OVERVIEW

- MQTT sensors are used to monitor ground real time conditions measuring temperature, rain water, humidity, pressure and actual weather.
- After the connection, data are sent in a JSON format on the topic "Info".

```
if (state == STATE_SUBSCRIBED) {  
    // Publish something , specify tag of topic  
    sprintf(pub_topic,"%s", "info");  
    strcpy(currentforecast, forecast[rand() % 3]);  
    if (strcmp(currentforecast,"Sunny")){  
        temperature = (rand() % (32 + 1 - 25) + 25);  
        humidity = (rand() % (25 + 1 - 17) + 17);  
        pressure = (rand() % (1040 + 1 - 1015) + 1015);  
        whether = 1;  
        mm_water = 0;  
    }else if (strcmp(currentforecast,"Cloudly")){  
        temperature = (rand() % (25 + 1 - 18) + 18);  
        humidity = (rand() % (50 + 1 - 25) + 25);  
        mm_water = (rand() % (2 + 1 - 1) + 1);  
        pressure = (rand() % (1015 + 1 - 998) + 998);  
        whether = 2;  
    }else if (strcmp(currentforecast,"Heavy Rain")){  
        temperature = (rand() % (18 + 1 - 12) + 12);  
        humidity = (rand() % (90 + 1 - 50) + 50);  
        mm_water = (rand() % (5 + 1 - 2) + 2);  
        pressure = (rand() % (998 + 1 - 990) + 990);  
        whether = 3;  
    }else if (strcmp(currentforecast,"Icy")){  
        temperature = (rand() % (5 + 1 - (-3)) + (-3));  
        humidity = (rand() % (17 + 1 - 5) + 5);  
        mm_water = 0;  
        whether = 4;  
        pressure = (rand() % (990 + 1 - 882) + 882);  
    }  
}
```



# MQTT

- The message sent to Broker has the following structure

```
printf(app_buffer, "{ \"temperature\":%d, \"humidity\":%d, \"forecast\":%d, \"pressure\":%d, \"rain\":%d}", temperature, humidity, whether, pressure, mm_water)
```

- Data are inserted into “mqtsensors” table using MySQL

```
Topic: info
QoS: 0
Payload: b'{"temperature":19,"humidity":28,"forecast":2,"pressure":1011,"rain":2}'
```

temperature	humidity	pressure	forecast	water	timestamp
31	23	1034	GOOD ATM CONDITIONS	0	2022-12-03 10:30:06
26	24	1029	GOOD ATM CONDITIONS	0	2022-12-03 10:30:07
30	21	1016	GOOD ATM CONDITIONS	0	2022-12-03 10:30:07
32	17	1039	GOOD ATM CONDITIONS	0	2022-12-03 10:30:21
20	34	1002	BAD ATM CONDITIONS	2	2022-12-03 10:30:22

- While on Cooja there will be a window like the following one:

```
01:00.348 ID:4 Subscribing
01:00.348 ID:4 Message: {"temperature":23,"humidity":32,"forecast":2,"pressure":1010,"rain":1}
01:00.360 ID:2 Application is subscribed to topic successfully
01:00.411 ID:4 Application is subscribed to topic successfully

01:00.899 ID:3 Subscribing
01:00.899 ID:3 Message: {"temperature":18,"humidity":48,"forecast":2,"pressure":1015,"rain":2}
01:00.981 ID:3 Application is subscribed to topic successfully
```

# COAP OVERVIEW

- Motion resource generates data according to a prefixed logic since physical sensors are not available.
- Motion sensors are triggers of the physical cover that it will close with a variable degree radius and an alarm will be triggered.

# COAP

- LEDs are turned on for a short period of time to indicate “Success” if registration to BR has been successful
- If resources state change a query is done on MySQL and a POST is made to set opening degree radius
- If the system is active, cover is activated, alarm is triggered and covers start to close with different radius degrees.

```
01:00.251 ID:6 Event trigger
01:00.251 ID:6 MSG detection send : {"closed":"N", "active":"N", "opening":"76"}
01:00.716 ID:5 Event trigger
01:00.716 ID:5 MSG detection send : {"closed":"N", "active":"N", "opening":"55"}
```

closed	activation	opening	timestamp
0	N	0	2022-12-03 10:29:16
0	N	0	2022-12-03 10:29:17
1	T	0	2022-12-03 10:29:21
1	T	0	2022-12-03 10:29:21
0	N	73	2022-12-03 10:29:51

# GRAFANA

Grafana was used to plot and display what is stored in “collector” in MySQL DB.  
On VM, Chromium has been installed, to access Grafana dashboard.  
Dashboard auto update is set to 30 seconds.

