UNIVERSITÀ DI PISA

ENGINEERING DEPARTMENT

MASTER'S DEGREE IN
ARTIFICIAL INTELLIGENCE AND DATA ENGINEERING

INTERNET OF THINGS

Roasting Machine:
Intelligent Coffee Roasting Control System

Fabiani Martina – Sacco Rossana Antonella

ACADEMIC YEAR 2023/2024

Summary

Introduction

This project is part of the Industry 4.0 framework, characterised by the use of advanced technologies such as the Internet of Things (IoT), Artificial Intelligence (AI), advanced robotics, augmented reality (AR) and cloud computing. These technologies create a highly interconnected and digitalized production environment, enabling more efficient and intelligent management of manufactoring processes.

Our goal is to develop a smart solution for monitoring and controlling the coffee roasting process, ensuring consistent product quality without additional costs and with greater operational efficiency. To achieve this, we have integrated three sensors inside the roaster to monitor critical parameters during the process: humidity sensor, temperature sensor and CO2 sensor.

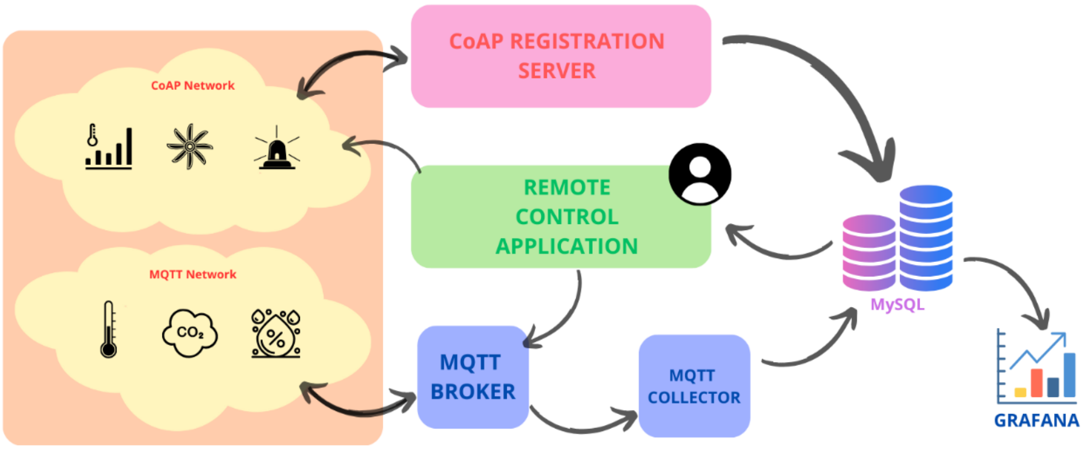
The real-time data collected by these sensors is used to automatically activate three actuators that regulate the internal environment of the machine: a temperature regulator, a fan and an alert system. This automatic control system maintains optimal roasting conditions, improving the quality of the coffee produced.

In addition, thanks to our Remote Control application, the user can remotely monitor and modify temperature, humidity and CO2 parameters, thus ensuring that the desired degree of roasting is achieved according to specific needs. This solution represents a step forward in the digitalization of the coffe roasting process, offering precise and customised control that meets the challenges of Industry 4.0.

Figure 1: Industrial roasting machine

Architecture

The system architecture is shown in the image below.



As we can see, the system is composed of two main IoT device networks: the MQTT network, consisting of sensors that use the MQTT protocol to collect data, and the CoAP network, which includes the actuators. Both the MQTT and CoAP networks are deployed using real sensors nRF52840 Dongle. These networks are connected to a border router that enables external access. Each element of the network communicates with the outside via the border router, which has been omitted for greater clarity regarding the interactions between the parts.

MQTT network

The sensors are a fundamental part of the system. In our project, we have considered one sensor for each type, but in practice, to obtain more accurate data, it would be necessary to install multiple sensors of each type at different points within the roasting chamber.

In the MQTT network, we have an MQTT broker responsible for receiving data from the sensors and forwarding it to the Remote Control Application and the MQTT Controller through topic subscriptions. In this project, we use Mosquitto, an open-source message broker that implements the MQTT protocol.

In the MQTT Collector, there is an MQTT client whose role is to collect the data transmitted by the sensors within the MQTT network. To achieve this, the application subscribes to all the topics where the sensors publish their messages. When a new publication is received, the Collector interprets it and inserts the new data into the database.

Sensors

- *Temperature sensor*: this sensor measures the temperature in real-time within the roasting chamber to prevent temperature fluctuations that could damage the product.
- *CO2 sensor*: this sensor measures the concentration of CO2 gas inside the roaster. Monitoring CO2 levels is crucial to avoid the accumulation of excessive gas, which could indicate incomplete combustion or other issues that may affect the quality and flavor of the roasted coffee.

- Humidity sensor: this sensor monitors the percentage of humidity within the roaster to ensure that the roasting process occurs in a controlled and uniform manner. Proper humidity levels are essential for achieving consistent roasting results and preventing uneven roasting or over-roasting of the coffee beans.

CoAP network

The CoAP network consists of actuators that communicate using the CoAP protocol. When an actuator joins the network, it sends a registration message to the CoAP server, containing essential information such as its IPv6 address and exposed resources. The CoAP server updates the database with this information, maintaining an up-to-date list of all available devices in the network and their resources.

Thanks to this mechanism, the Remote Control Application (RCA) can read the database to obtain the address and resources of the desired device. Once this information is retrieved, the RCA can effectively send commands to the actuators using the CoAP protocol.

The use of the CoAP network with a device discovery and registration mechanism offers significant advantages in terms of efficiency, as it is a lightweight and low-latency protocol, as well as in terms of ease of management, scalability, and reliability, making the IoT network more robust and responsive.

Similar to the sensors, for the purpose of this presentation, we have opted to showcase a single instance of each actuator type.

Actuators

- *Temperature Regulator:* this actuator adjusts the temperature inside the roaster based on the data collected by the temperature sensor. It can increase or decrease the temperature to maintain the desired roasting conditions, ensuring consistent product quality.
- *Fan:* the ventilation system helps to regulate air circulation within the roaster. By controlling the airflow, the fan aids in maintaining even temperature distribution and preventing overheating, which can result in more uniform roasting and better quality control.
- *Alarm System:* this system alerts workers if CO2 levels in the air exceed a threshold that is harmful to the environment. It provides warnings to ensure that any potentially dangerous conditions are promptly addressed, helping to maintain a safe working environment.

Database

It is essential to store the data collected with the sensors, also to be able to analyze them through Grafana.
The database "RoastingMachine" is particularly simple, we have defined different tables which will illustrate below.
There are no dependencies between the tables.

Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	auto_increment
timestamp	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
value	int	YES		NULL	
type	varchar(256)	YES		NULL	
interval	int	YES		NULL	

Figura 2: dataSensed table

The '*dataSensed*' table is populated by the MQTT Collector with the data measured by the sensors. The *'type'* field indicates the type of sensor that recorded the data and can be *'temperature,' 'humidity,'* or *'co2'*. Additionally, there is the *'interval'* field, which is only different from NULL for humidity data, as it is necessary to know the interval of the roasting process to better manage the roasting environment. This table is used by Grafana to produce real-time graphs of the roaster's status and by the Remote Control Application to implement application logic and activate the actuators when necessary.

Field	Type	Null	Key	Default	Extra
ip	varchar(256)	NO	PRI	NULL	
resource	varchar(256)	NO		NULL	
status	varchar(256)	YES		NULL	

Figura 3: actuators table

The *'actuators'* table is populated by the CoAP Server to store actuators information, which will be retrieved by the Remote Control Application to communicate with them.

Data encoding

In resource-constrained IoT scenarios, the efficiency of the chosen protocol is crucial. In our project, given the use of resource-limited devices such as sensors and actuators, JSON was preferred over XML. This is because JSON has a lighter and simpler structure, making it easier to process and reducing the computational load on devices. It introduces less overhead due to its less verbose syntax and is more suitable for efficient and fast data transmission.

Deployment and execution

Sensors

The data collected by the sensors are encapsulated in JSON format and sent through the MQTT protocol to the Collector. Each sensor publishes the collected data to a specific topic, identified as sensor/type, where type indicates the sensor type:

- *sensor/humidity*
- *sensor/temp_co2*

For temperature and CO2 data, we chose to use a single topic (sensor/temp_co2) because the two sensors are implemented on a single node, and the data are published together every 2 seconds. This decision was made to reduce energy consumption by sending a single, slightly longer message instead of two separate ones.

Additionally, since the user can modify the maximum and minimum thresholds for humidity, temperature, and CO2 values during the roasting process through the Remote Control Application (RCA), it is necessary for the sensors to subscribe to specific topics to receive these updates. The topics used for this purpose are:

- *param/temp*
- *param/co2*
- *param/humidity*

These topics allow the system to send updates about operational thresholds to the sensors, ensuring that the sensors are always updated and can promptly send data to the Collector in case of anomalous values.

Additional clarifications:

- The roasting process is divided into three distinct phases, each requiring different humidity thresholds. This subdivision is essential for ensuring optimal roasting of the product, as each phase of the process requires a specific humidity level. In reality, the entire roasting process lasts 15 minutes, with each phase extending for 5 minutes. However, in our implementation, we have reduced the times: the entire process lasts 1 minute, with each phase lasting 20 seconds.
- CO2 and temperature values are measured every 2 seconds but are sent to the MQTT Collector only if the values are out of range or every 30 seconds. This approach aims to reduce data traffic and save energy.
- Humidity values are measured every 5 seconds. Similar to CO2 and temperature, data are sent to the MQTT Collector only if the values are out of range or every 30 seconds. However, if anomalous values are detected, the humidity measurement frequency increases to every 3 seconds until the values return within the established limits, allowing for more rigorous process control.

To simulate values out of range, we use the button on our Nordic Semiconductor nRF52840 dongle:

- <u>*Humidity Sensor Node:*</u>
    1. First press: Values increase beyond the maximum threshold.
    2. Second press: Values decrease below the minimum threshold.
    3. Third press: All values generated are within range.
- <u>*Temp_CO2 Node:*</u>
    1. First press: Only the temperature value generated is out of range.
    2. Second press: Only the CO2 value generated is out of range.

3. Third press: All values generated are within range.

Actuators

When an actuator is started, it must register with the CoAP server by sending a JSON message. This message includes the actuator's name and its initial status. If registration fails, the actuator will attempt to re-register up to three times. After three failed attempts, the actuator will wait for 10 seconds before trying to register again.

Once registration and resource activation are successfully completed, the actuator enters a waiting state. During this phase, the actuator waits for events, specifically requests from the Remote Control Application. These requests include commands to turn the actuator on or off. Each actuator will continue to monitor events and respond appropriately based on the received requests.

The following are the commands accepted by each actuator.

- **Temperature Regulator:**
  - Up (Green LED): increases the temperature to counteract high humidity or a sudden drop in temperature.
  - Down (Blue LED): decreases the temperature in case of excessive heating.
  - Off (Red LED): turns off the regulator when temperature and humidity values are within the preset limits.
- **Fan**:
  - On (Green LED): turns on the ventilation to improve air circulation and reduce humidity.
  - Off (Red LED): turns off the ventilation when it is no longer needed.
- **Alarm System**:
  - On (Red LED): indicates that CO2 levels are too high and requires immediate action.
  - Off (No LED): indicates that the alarm system is deactivated and CO2 levels are under control.

At each command received the actuator sends back a response to the Remote Control Application that can contain the following CODEs:

- *CHANGED,* if the required action was performed correctly
- *BAD_OPTION,* if the request contain an unknown action
- *BAD_REQUEST, if the request does not contain any action*

Remote Control Application

The Remote Control Application is the central hub of our system, orchestrating several essential functions for the optimal management of the roasting machine. This central component is responsible for continuously monitoring data from the sensors. Every 2 seconds, it queries the database to retrieve the most recent data published by the sensors. Once the data is obtained, the application analyzes it and compares it against user-defined thresholds. If the values exceed these limits, the application takes immediate action by sending commands to the actuators via CoAP to bring the parameters back to normal levels.

Furthermore, the application is designed to facilitate smooth user interaction. Users can modify system parameters, such as temperature and humidity limits, through specific commands. These updates are transmitted to the sensors using MQTT, ensuring efficient and reliable communication. This process allows the system to quickly adapt to new settings and changing conditions.

The Remote Control Application also enables direct commands to the actuators. For example, users can turn the temperature regulator on or off, activate or deactivate the fan, and manage the alarm system. This direct control capability provides immediate flexibility, allowing operators to make real-time adjustments and respond swiftly to any changes in operating conditions.

Finally, the application integrates all system components, ensuring smooth communication between sensors, actuators, and user inputs. It provides a comprehensive overview of the system's status.

The Remote Control Application not only manages and optimizes the operation of the toaster but also facilitates effective user interaction and maintains precise control over operating conditions.

This application perform the following operations.

```
******* ROASTING MACHINE CONTROL CENTER *******

|                                             |

|  Type the keyword of the command you desire |

|  to run. Remember to type '/' first!        |

|                                             |

|  1. /SHOW_ACTUATORS_STATUS                  |

|  2. /SHOW_PARAMETERS                        |

|  3. /CHANGE_PARAMETERS                      |

|  4. /CHANGE_ACTUATORS_STATUS                |

|*********************************************|
```

/show_actuators_status: show the current status of the available actuators
/show_parameters: show the current parameters of each sensors
/change_parameters: open the menu to change sensors parameters

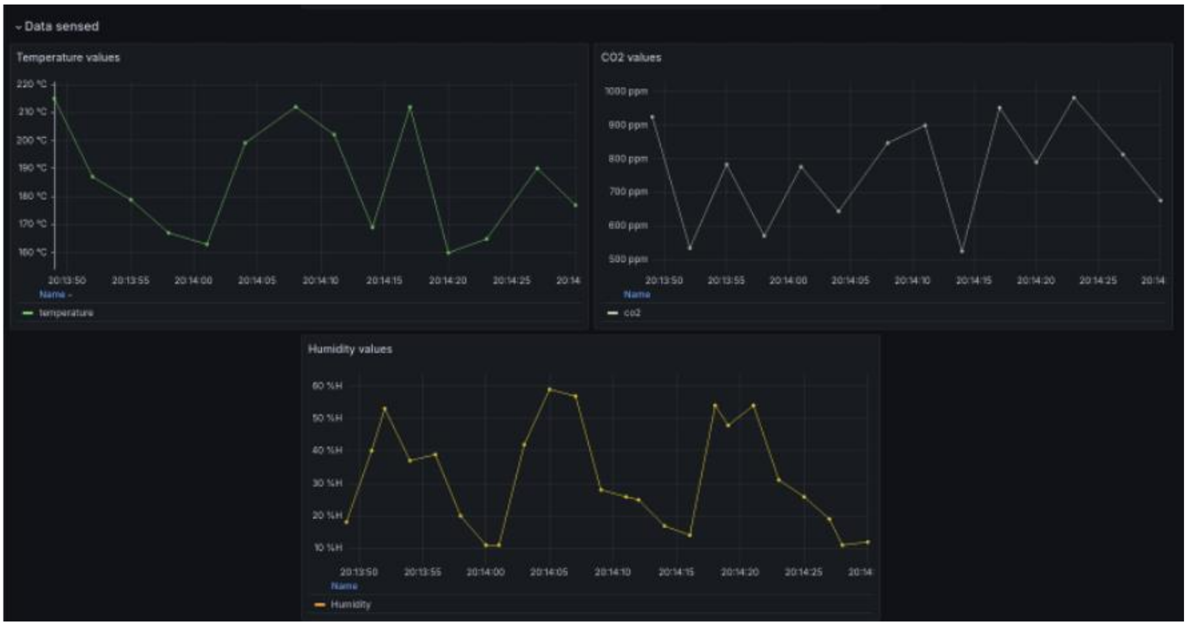
- ***/min_humidity_parameter_FIRST***: change the minimum value of the humidity in the first interval of the process
- ***/max_humidity_parameter_FIRST:*** change the maximum value of the humidity in the first interval of the process
- ***/min_humidity_parameter_SECOND:*** change the minimum value of the humidity in the second interval of the process
- ***/max_humidity_parameter_SECOND:*** change the maximum value of the humidity in the second interval of the process
- /***min_humidity_parameter_THIRD***: change the minimum value of the humidity in the third interval of the process
- /***max_humidity_parameter_THIRD:*** change the maximum value of the humidity in the third interval of the process
- ***/max_co2_parameter:*** change the maximum value of the co2 inside the roasting chamber
- **/min_temp_parameter:** change the minimum value of the temperature inside the roasting chamber
- **/max_temp_parameter:** change the maximum value of the temperature inside the roasting chamber

/change_actuators_status: open the menu to send a command to an actuator

- ***/reg_temp_up:*** increase the temperature
- ***/reg_temp_off:*** turn off the temperature regulator
- ***/reg_temp_down:*** decrease the temperature
- ***/alert_on:*** activate the alert
- ***/alert_off:*** deactivate the alert
- ***/fan_on:*** turn on the fan
- ***/fan_off:*** turn off the fan

Grafana

Grafana dashboard is used to compose some analytics based on the values that are stored on the database.



In this section, the most recent values measured by the sensors are displayed, including the minimum, maximum, and average values over the last 24 hours. Values that are outside the set limits are highlighted in red.

Use case

A coffee roaster is responsible for ensuring consistent quality across different batches of coffee beans. In a traditional setup, this requires manual monitoring and adjustments, often leading to variability in roast quality. To address this, the roaster implements a smart roasting system with integrated sensors (temperature, humidity, CO2) and actuators (temperature regulator, fan, alert system), along with remote control capabilities.

Actors:

- Coffee Roaster (Operator)
- Coffee Roasting Machine (with integrated sensors and actuators)

Process:

1. **Initial Setup:**
    - The operator selects the desired roast profile using the Remote Control Application, specifying parameters such as target temperature, humidity, and CO2 levels based on the type of beans and the desired roast.
2. **Real-Time Monitoring:**
    - During the roasting process, the temperature sensor continuously monitors the internal temperature of the roasting chambre to ensure it remains within the specified range.
    - The humidity sensor tracks moisture levels, helping to ensure beans are dried evenly without overheating.
    - The CO2 sensor monitors gas buildup, which can provide insights into the stage of roasting and ensure safety.
3. **Automatic Adjustments:**
    - When the temperature sensor detects a deviation from the target range, the temperature regulator automatically adjusts the heat source to bring it back in line, preventing under- or over-roasting.
    - The fan is automatically activated if the humidity level or CO2 level exceeds the set threshold, helping to maintain the optimal environment inside the drum.
    - If CO2 levels rise above safe or preset limits, an alert system is triggered, warning the operator remotely and in-person, and the machine can adjust airflow or pause roasting to prevent safety risks.
4. **Remote Intervention:**
    - The operator can monitor the status of each actuators through the RCA . If needed, adjustments to temperature, fan speed, or humidity thresholds can be made on-the-fly, ensuring flexibility for different coffee profiles.

Outcome:
This automated system ensures optimal and consistent roasting, reducing manual intervention, improving efficiency, and delivering consistent quality, while also offering the flexibility to adapt roast profiles through remote control based on real-time data.