1 Name of Use Case

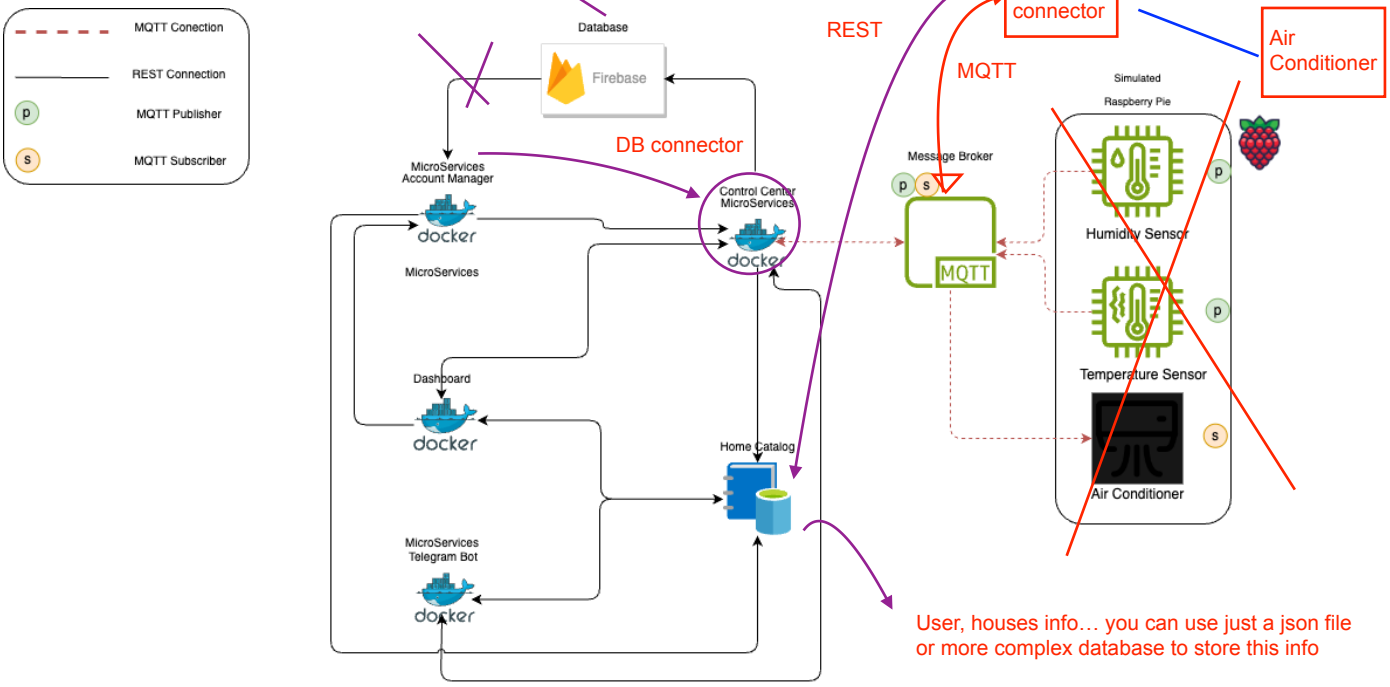
Name of the Use Case	**IoT platform for Smart Home**	
Group	24	
Version No.	V1.0	
Submission Date	07/01/2024	
Team Members (with student ids)	Ali Mohammad Alizadeh (308885) Mohammad Eftekhari Pour (307774) Mehdi Rajaee (308497) Hamid Shabanipour (314041)	s308885@studenti.polito.it s307774@studenti.polito.it s308497@studenti.polito.it s314041@studenti.polito.it

2 Scope and Objectives of Function

Scope and Objectives of Use Case	
Scope	The proposed IoT platform aims to provide additional services for Smart Homes for having a control over air conditioning of environment.
Objective(s)	The expected results include: • Enable precise control over environmental conditions (including temperature and humidity) for a consistently comfortable living space. • Gather Historical Data for in-depth Statistical Analysis and informed decision-making. • Monitor Home Environment Conditions, identifying patterns and fluctuations over the previous periods. • Implement both Manual and Automatic Commands for Air Conditioning Devices.
Domain(s)	Smart Home
Stakeholder(s)	Home Inhabitants, Air Conditioning System
Short description	The proposed IoT platform integrates various IoT devices to optimize air-conditioning systems, offering control strategies to minimize energy consumption and enhance overall living conditions. The platform provides unified interfaces (REST and MQTT) to seamlessly integrate the home into a Smart Building, allowing for flexible Demand/Response policies. End-users gain detailed insights into their environment conditions. Summarizing, the main features it offers are: • Monitoring of House temperature and humidity. • Control strategies for temperature, humidity, and air conditioning systems. • Unified interfaces (REST Web Services and MQTT queues) to support Demand/Response scenarios. • End-user applications for an enriched experience, offering detailed information on the current environment. • Collection of Historical Data for future Data Analytics and Decision-making. • Comprehensive Historical data and Data Analytics support.

In this case example I add to your diagram seems to follow your explanation, with the RPI as a device connector, and the sensors/actuators are connected to it…. If you intend to use the sensors or the air conditioner as an IoT device itself, the diagram will be different, as well as the explanation….

3 Diagram of Use Case



4 Complete description of the system

The proposed IoT platform for Smart Home follows the microservices designing pattern. It also exploits two communication paradigms:

i) publish/subscribe based on MQTT protocol
ii) Request/response based on REST Web Services.

In this context, some actors have been identified and introduced in the following:

Where is the RPI Connector in the diagram?

- **Raspberry Pi Connector** is used for connecting humidity and temperature sensor to provide information about the environmental conditions in order to monitor and maintain the appropriate conditions required for inhabitants. Another important task is based on the MQTT protocol: working as a MQTT publisher it is able to send information about environment data (from humidity and temperature sensors), progressively analysed and finally showed on the Web-Dashboard. Also, Manual or Automatic Commands will be sent to Air-Conditioning system to be set with needed information or be turned Off and On.

- **Environmental Control (Temperature and humidity control)** based on the information's taken from the sensors, is a control protocol created to maintain the ideal conditions to provide consistent appropriate environment. The tool and the sensor work with MQTT communication, like subscriber and publisher, comparing the data generated by the sensors to the specifications assigned by the home owner in order to manage the air condition of the area. Missing in the diagram? Maybe send MQTT alert when some parameters are out of limits?

- **Data Analysis,** one of the uses of historical data is having some analysis over timeseries and make predictions for future home environment, in this case we will use timeseries such as hourly temperature and humidity, hourly comfortable setting for Airconditioning system, also considering the manual command of the user to predict better

Missing in the diagram? Where does it retrieves data from? Time-series database? Which protocol does it use? What are its outputs, and "who" consumes them? Does it expose REST APIs? Send MQTT commands to actuator?

configurations for Air-Conditioning system, to implement this feature, Moving Average over the data or ML algorithms like Regression or ARIMA model can be used since we have timeseries.

- **The Home Catalogue** works as service and device registry system for all the actors in the system. It provides information about end-points (i.e., REST Web Services and MQTT topics) of all the devices, resources and services in the platform. It also provides configuration settings for applications and control strategies (e.g. historical Data, list of sensors and actuators). Each actor, during its start-up, must retrieve such information from the Home Catalogue exploiting its REST Web Services.

- **Sensors and Actuators,** two sensors namely Humidity and Temperature is used to get the environmental data which will be sent over MQTT microservices. Also, Air Conditioning devices are integrated to be turned off and on depending on the data provided by analytic service for the user and it could be done manually by the user.

Unclear with RPI device connector.. are the sensors and actuators "wirely" connected to RPI? Or are the sensors IoT enabled, and communicate over the internet?

- **Message Broker** provides an asynchronous communication based on the publish/subscribe approach. It exploits the MQTT protocol.

- **Control Centre,** is Responsible for communicating With database and other REST APIs to send MQTT messages and receives data provided by MQTT communication and send them to other microservices.

Sort of DB adaptor? Unclear….

- **Database,** Stores user-related data and information collected through sensors. First decision is to use Firebase but it might change based on the need.

- **Account Manager,** is Responsible for managing users and houses corresponding to them and associated device.

- **Web-Dashboard:** Enables users to access their profile and generated data by the system will be demonstrated to the user through graphics and plots by using Rest APIs. Using REST web services statistical information about the status of the House will be retrieved in order to show it on Web-Dashboard. For Realtime plots there might be a need to use Thing Speak but this needs further study. Check comments below…

- **Telegram Bot** latest house environment and weekly temperature average also has the ability to send a command to Air-Conditioning device. No….. Computation must be done on another service… the Bot is just for user interaction… the weekly average of temperature and other calculations must be done somewhere else…

Maybe receive some MQTT alerts when parameters are out of limits?

5 Desired Hardware

Device Name	Quantity	Needed for…
Raspberry Pi	1	Broker
DHT11	1	Collect temperature and humidity data
Relay	3	Turn on and turn off the air conditioner

- Check diagram…. some actors are missing!!! Also check connections (e.g. MQTT to Telegram Bot)

- Check the comments on the RPI connector… In the diagram you can find the example as using the RPI as a device connector….

- Environmental control missing in the diagram?

- Data Analysis missing in the diagram? Where does it retrieves data from? Time-series database? Which protocol does it use? What are its outputs, and "who" consumes them? Does it expose REST APIs to calculate on "request" this analysis? Does it calculates some values automatically and send MQTT commands?
Try to provide a bit more details, specially about the "output"

- Database -> I suggested you to use one database for storing time-series information (e.g. sensors measurements) and another database or just json file for storing user related data, and other catalog information…

  * For the time-series database, we will see Thingspeak during the course for storing time-series information. Another time-series database you can use is influx-db

- Control center is unclear… Sort of Database Connector? I suggested you to just have this service that communicates with the timeseries-database…

- Web-dashboard -> the plots and statistical information and calculations, must be "computed" somewhere else: the Web-dashboard must be just for "retrieving" and showing this information. For example, can retrieve this info from data analysis….

- Telegram Bot
  *. Check comment about computation that must be done somewhere else. Also the suggestion of maybe using this service to receive some MQTT alerts.

  * Try to provide more details of this service… Latest house environment measurements are received using MQTT? or REST? … Maybe calculations like weekly temperature retrieved from data analysis using REST… Provide this details…