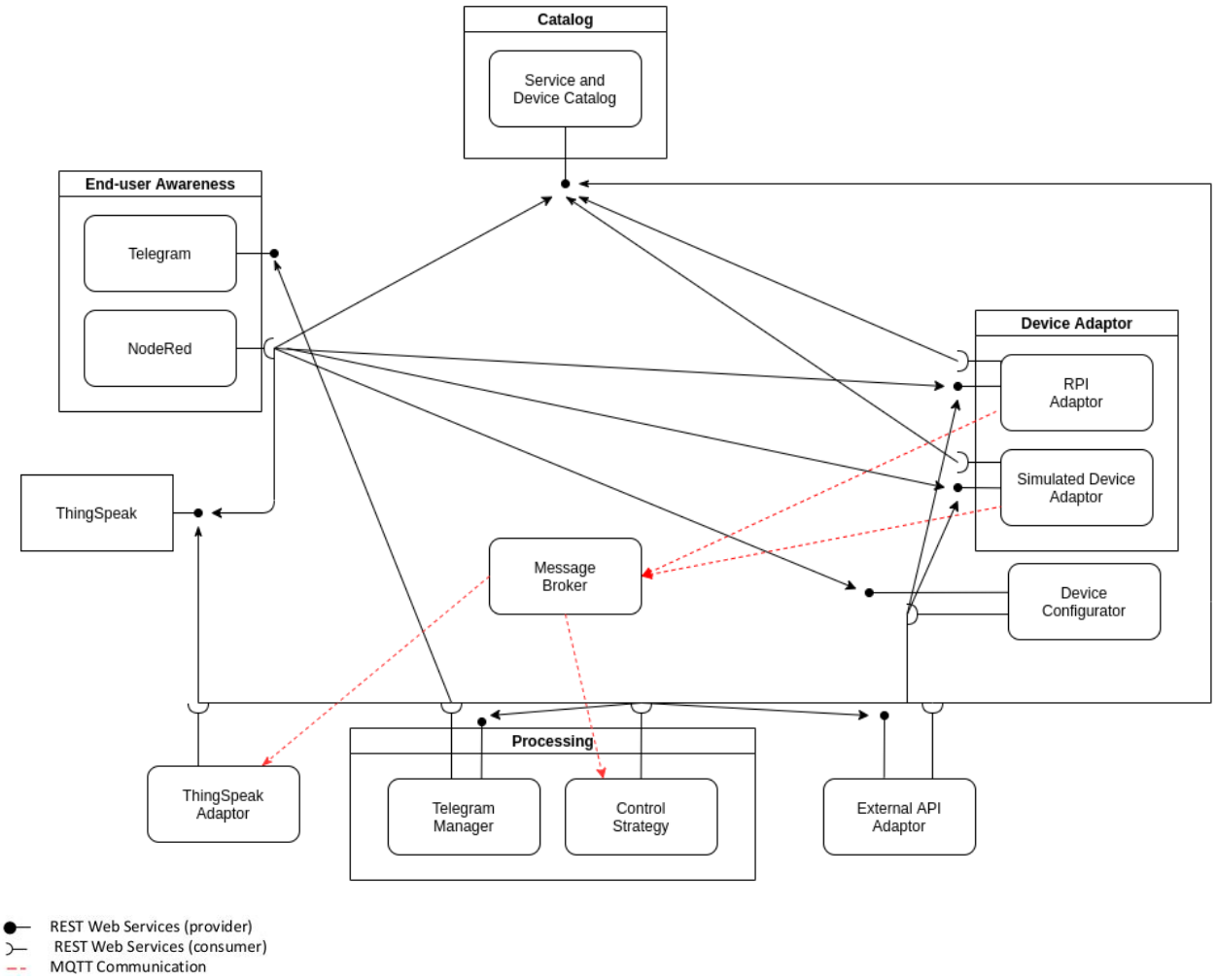
1 Name of Use Case

Name of the Use Case	IoT platform for user-involved air recirculation system
Version No.	v0.2
Submission Date	05/12/2020
Team Members (with student ids)	Matteo Battilana 281389 s281389@studenti.polito.it Jacopo Ferrero 223357 s223357@studenti.polito.it Michele Wysocki 269875 s269875@studenti.polito.it Andrea Zarri 280055 s280055@studenti.polito.it

2 Scope and Objectives of Function

Scope and Objectives of Use Case	
Scope	This IoT system aims to measure the air quality in indoor spaces.
Objective(s)	To monitor the air quality and notify the users in order to maintain a healthy environment.
Domain(s)	Health, Individual well-being with user self-awareness, Indoor environment monitoring.
Stakeholder(s)	Home inhabitants, Healthcare provider, City services companies, Public infrastructure.
Short description	Our IoT platform aims to solve the air recirculation problem in indoor environments. This need can easily occur in the presence of numerous individuals such as in a classroom or in a restaurant, especially during this pandemic, but also in every home when we are dealing with humidity and the risk of mold creation. A device is provided with a temperature and humidity sensor with the addition of a co2 sensor. The CO2 sensor provides us information on how dense the air is with carbon dioxide (ppm) inside a room and from this we can deduce when it is time to open the windows of this environment; for example we can reduce the spread of a virus among those present. At this point, the user is notified via Telegram and he can decide to open the windows. Optionally, in order to propose the best solution to the user, the system can also retrieve information (humidity, temperature, pollution) about the outside air, eventually suggesting the use of an air purifier or air conditioner according to the user configuration previously set via the Telegram Bot or via the web interface. The main features are: ● User awareness via Telegram Notification; ● User-end interface for expose data and configure the devices; ● MQTT protocol to exchange data in a scalable platform; ● RESTFul API exposed by the services in order to configure or retrieve information; ● Air monitoring system.

3 Diagram of Use Case



4 Complete description of the system

The IoT platform is based on multiple integrated devices that use sensors for monitoring the air quality of an indoor space. Multiple sensors can communicate over the MQTT and RESTFul protocols.

IoT Platform microservices:
Message Broker: The MQTT Broker is an external MQTT server that receives all messages from the clients and then routes the messages to the appropriate destination clients, exploiting the synchronous communication based on Publish/Subscribe MQTT protocol. Its address is hardcoded in the Catalog.

Catalog: It is an unique microservice that is composed by the Service Catalog and the Resource Catalog. The first one is used to register the services available in the network exposing JSON-based RESTful API. On the other hand the Resource Catalog registers and provides a registry of available IoT devices and the resources they expose. It exposes a simple JSON-based RESTful API. We decided to merge these two entities in order to avoid duplicated logic and reduce the communications. At the startup the devices will register to it and retrieve information of the other services and devices.

Device Adapter: We defined two types of device adapter, the RaspberryPi adapter is the one used by the a real RPI hardware with real sensors, such the temperature, humidity and C02 sensors. The Simulated

Device Adaptor is used to implement other devices with simulated hardware. The MQTT protocol is used only to publish data. Both the real and the simulated adaptors expose RESTFul API to allow their configuration and in order to obtain data. We decided to let the configuration part exposed with RESTFul API instead of MQTT.

User Awareness: The user awareness is based on two main parts, the first one is based on a Telegram Bot, that notifies the user when the quality of the air is not good enough. The user wants to have some statistics or simply access to the data the system has acquired; in order to add this functionality, the platform implements a user-friendly interface exposed via the Freeboard platform, exposed via a web server.

Telegram Manager: This service exposes only RESTFul API to the other processing service, to allow the user to send the notification and alert to the user. It can be used to configurare the devices, by calling the REST API in the Device Configurator.

NodeRed: This dashboard is an information management tool to visualize and analyze metrics and data to monitor (https://nodered.org). Moreover the platform allows to configure the device parameters. Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.

ThingSpeak: ThingSpeak is a third-party software that provides storage capabilities, exposing RESTFul API.

ThingSpeak Adaptor: It is a MQTT subscriber that listens to all the messages in the network, only the ones that contain data are uploaded to ThingSpeak via the RESTFul API. These components act as a middleware, in order to let our system to interface with the external service.

Device Configurator: This component lets the user to configure the devices, such as the sampling period, the silent mode time (in order to avoid sending notification in sensible times) and so.

External API Adaptor: Since our platform also needs to know the temperature outside, an external RESTFul API must be called to obtain the desired information such as temperature, humidity and pollution.

Control Strategy: It is the main service that performs statistics and computation over the data. It exploits the MQTT protocol to receive the messages containing the sensor values and also makes requests to the other services that are necessary for the computation. The service performs continuous evaluation of the current situation of the indoor air quality. This service, exploiting the MQTT protocol, receives periodically data from the devices (the sensor samples) in the platform and, in relation with the external conditions, provides the user suggestions on how to increase the quality of his environment. For example, if the system detects a high level of humidity in a room, it checks the external conditions via an external device or via the external API, to suggest the user to open a window or in case, turn on the dehumidifier if present. This logic can be applied to all other parameters such CO2 and temperature and used for different purposes, starting from avoiding the mold creation to the air recirculation (this is particularly useful during the COVID pandemic).

The sensor data can be elaborated in order to obtain the maximum and minimum values and moreover the average over the day, week or even month.

The control strategy can also use the statistics of the previous week/day, according to the weather condition about the following day retrieved by the external APIs, in order to give the user the information about the best time to start recirculating the air. This prediction will be more accurate on the current day.