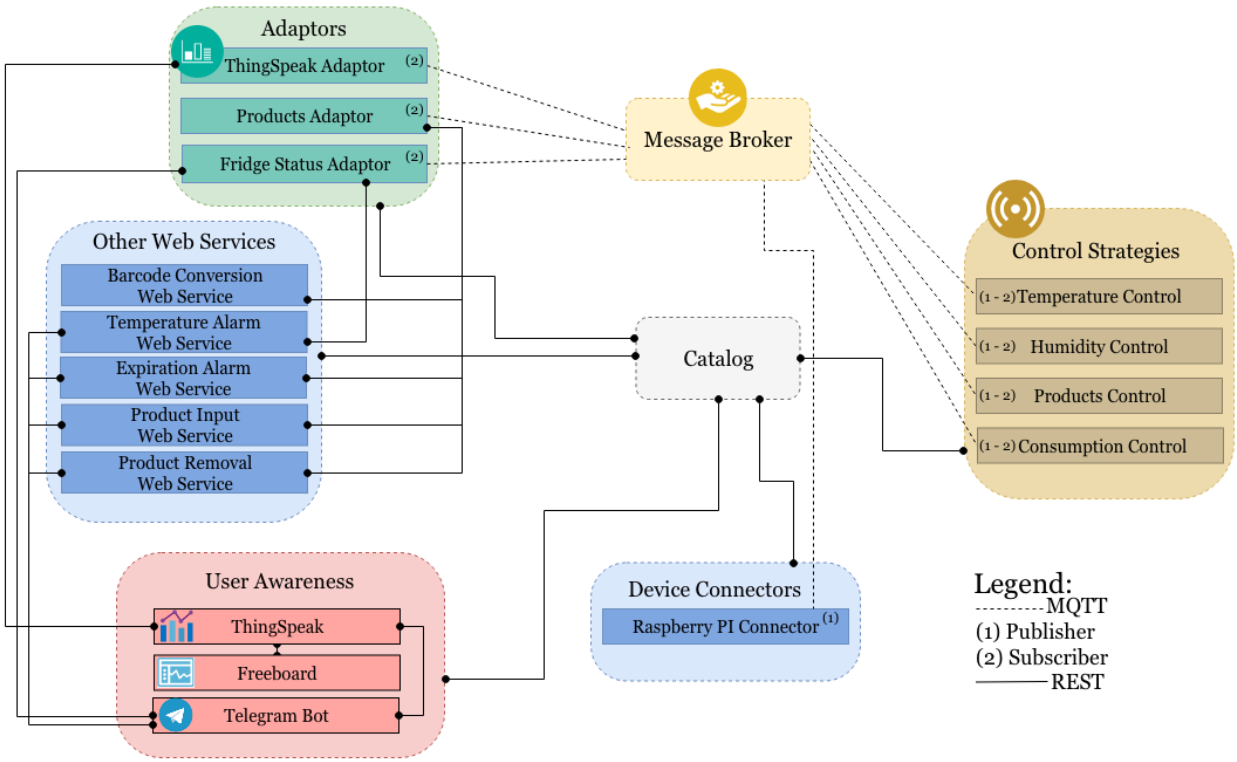
1 Name of Use Case

Name of the Use Case	**My Green Fridge**
Version No.	v2.0
Submission Date	10/01/2019
Team Members (with student ids)	Andrulli Martina (s255191), Bergamasco Letizia (s263328), Grasso Stefano (s242455), Riccio Piera (s262849)

2 Scope and Objectives of Function

Scope and Objectives of Use Case	
Scope	The proposed IoT platform aims at providing services for a smart fridge management.
Objective(s)	The objectives consist in providing the possibility for the users to control their own impact on the environment and in promoting their awareness around the issue of food waste.
Domain(s)	Environment, Smart Home
Stakeholder(s)	Home inhabitants
Short description	The proposed IoT platform aims at enhancing the standard functionalities of a fridge, thus making it smarter and greener, by means of unified interfaces (REST and MQTT). It offers control strategies to monitor the temperature and humidity of the fridge, as well as the products that are inserted in or removed from the fridge. In order to minimize food waste, a notification system is activated when the products are approaching their expiration date. The platform improves user-awareness around their consumption behaviour by keeping track of the wasted food and periodically providing statistics. In short, this platform provides: • monitoring of the temperature and humidity of the fridge; • monitoring of the products inserted in or removed from the fridge; • notification service for reducing food waste; • statistics on the user behaviour in food consumption.

3 Diagram of Use Case



Complete description of the system

The proposed IoT platform for My Green Fridge is designed following a micro-services approach and exploiting two communication paradigms: i) publish/subscribe based on MQTT protocol and ii) request/response based on REST Web Services. The system is composed by 18 actors that are described as follows:

- The **Message Broker** exploits the MQTT protocol to provide an asynchronous communication used by the Raspberry PI Connector, the Control Strategies and the Adaptors. It relies on the publish/subscribe approach.
- The **Catalog** provides a registry of all the available IoT devices and network services in the system, as well as the exposed resources. It also provides configuration settings that each actor must retrieve during its start-up, by means of the Catalog's REST Web Services. Among the settings, the Catalog contains information about end-points of each actor (i.e. REST Web Services and MQTT topics).
- The **Raspberry Pi Connector** is an implementation of a *Device Connector*, which allows the co-operation of heterogeneous devices; in fact, the Raspberry Pi Connector integrates into the platform Raspberry Pi boards. The Raspberry is equipped with temperature and humidity sensors to provide information about the status of the fridge and two cameras to monitor the products that are inserted in and removed from the fridge. The Raspberry Pi Connector works as an MQTT publisher, sending the information from the sensors to the Message Broker, and it also communicates through REST Web Services with the Catalog.
- The **Temperature Control** is a control strategy that manages the information about the temperature in the fridge, comparing it to a threshold that is chosen during the configuration process. It works both

as an MQTT subscriber, to receive real-time information on the temperature value, and as an MQTT publisher, to communicate with ThingSpeak Adaptor and Fridge Status Adaptor.

- The **Humidity Control** is a control strategy that manages the information about the humidity in the fridge. It works both as an MQTT subscriber, to receive real-time information on the humidity value, and as an MQTT publisher, to communicate with ThingSpeak Adaptor and Fridge Status Adaptor.
- The **Products Control** is a control strategy that manages the images of the products inserted in the fridge. It works both as an MQTT subscriber, to receive the images from the first webcam, and as an MQTT publisher, to return the corresponding EAN codes to the Products Adaptor.
- The **Consumption Control** is a control strategy that manages the images of the products removed from the fridge. It works both as an MQTT subscriber, to receive the images from the second webcam, and as an MQTT publisher, to return the corresponding EAN codes to the Products Adaptor.
- The **ThingSpeak Adaptor** is an MQTT subscriber that receives data about the status of the fridge from the control strategies. It communicates through REST Web Services both with the Catalog, in order to retrieve information about the wasted products, and with ThingSpeak software, in order to upload all the data it has received.
- The **Products Adaptor** is an MQTT subscriber that receives data from the control strategies. It communicates through REST Web Services with the Catalog, in order to update the information about the products present in the fridge and the wasted products, and with the other Web Services, in particular with Barcode Conversion Web Service, Expiration Alarm Web Service, Product Input Web Service and Product Removal Web Service.
- The **Fridge Status Adaptor** is an MQTT subscriber that receives data about the status of the fridge from the control strategies. It communicates with the Temperature Alarm Web Service using REST Web Services, indicating whether the temperature value is above the threshold. With the same kind of communication, it also allows the user to visualize the current status of the fridge on the available interfaces, in this case a Telegram Bot.
- The **Barcode Conversion Web Service** provides the mapping from EAN Code of a product to its description through REST Web Services.
- The **Temperature Alarm Web Service** sends a warning through REST Web Services to the available interfaces (in this case, Telegram Bot) when the temperature is above the threshold and the alarm has been enabled by the user. The communication about the status of the temperature is received from the Fridge Status Adaptor through REST Web Services.
- The **Expiration Alarm Web Service** sends a warning through REST Web Services to the available interfaces (in this case, Telegram Bot) when a product in the fridge is approaching its expiration date. The expiration dates of the products are retrieved from the Catalog through REST Web Services.
- The **Product Input Web Service** receives the description of each inserted product from the Product Adaptor and sends a message through REST Web Services to the available interfaces (in this case, Telegram Bot) asking the user to provide the expiration date of the inserted product.
- The **Product Removal Web Service** receives the description of each removed product from the Product Adaptor and sends a message through REST Web Services to the available interfaces (in this case, Telegram Bot) asking the user whether the product is going to be thrown away or consumed.
- **ThingSpeak** is a third-party software (https://thingspeak.com/) that provides REST Web Services. It is an open-data platform for the Internet of Things to store, post-process and visualize data (through plots).
- **Freeboard** is a dashboard to retrieve data from IoT devices and visualize them. It exploits the ThingSpeak Web Services to import plots about the trend over time of the status of the fridge and of the consumption behaviour of the user in terms of food waste.
- **Telegram Bot** is a service to integrate the proposed infrastructure into a Telegram platform, which is a cloud-based instant messaging system. The bot present in this IoT platform communicates with the Catalog through REST Web Services and serves the function of interfacing the user with the entire platform, by communicating with the other Web Services.

4 Desired Hardware

Device Name	Quantity	Needed for…
Raspberry Pi 3 model B	1	Collecting and transmitting data from the sensors (already provided in the Raspberry kit).
DHT11 Temperature and humidity sensor	1	Measuring the temperature and the humidity of the fridge (already provided in the Raspberry kit).
Webcam	2	Detecting the barcodes of the products during insertion and removal.