1. Name of Use Case

Name of the Use Case	**IOT enhanced smart farming for plant monitoring**
Version No.	V1.0
Submission Date	06 December 2024
Team Members (with student ids)	Guillaume CLOUARD s335346 Kevin George KOSMA PATABENDIGE DALPATHADU s338635 Lorenzo CAVALLARO s346742 Negar Abdi s343747

2. Scope and Objectives of Function

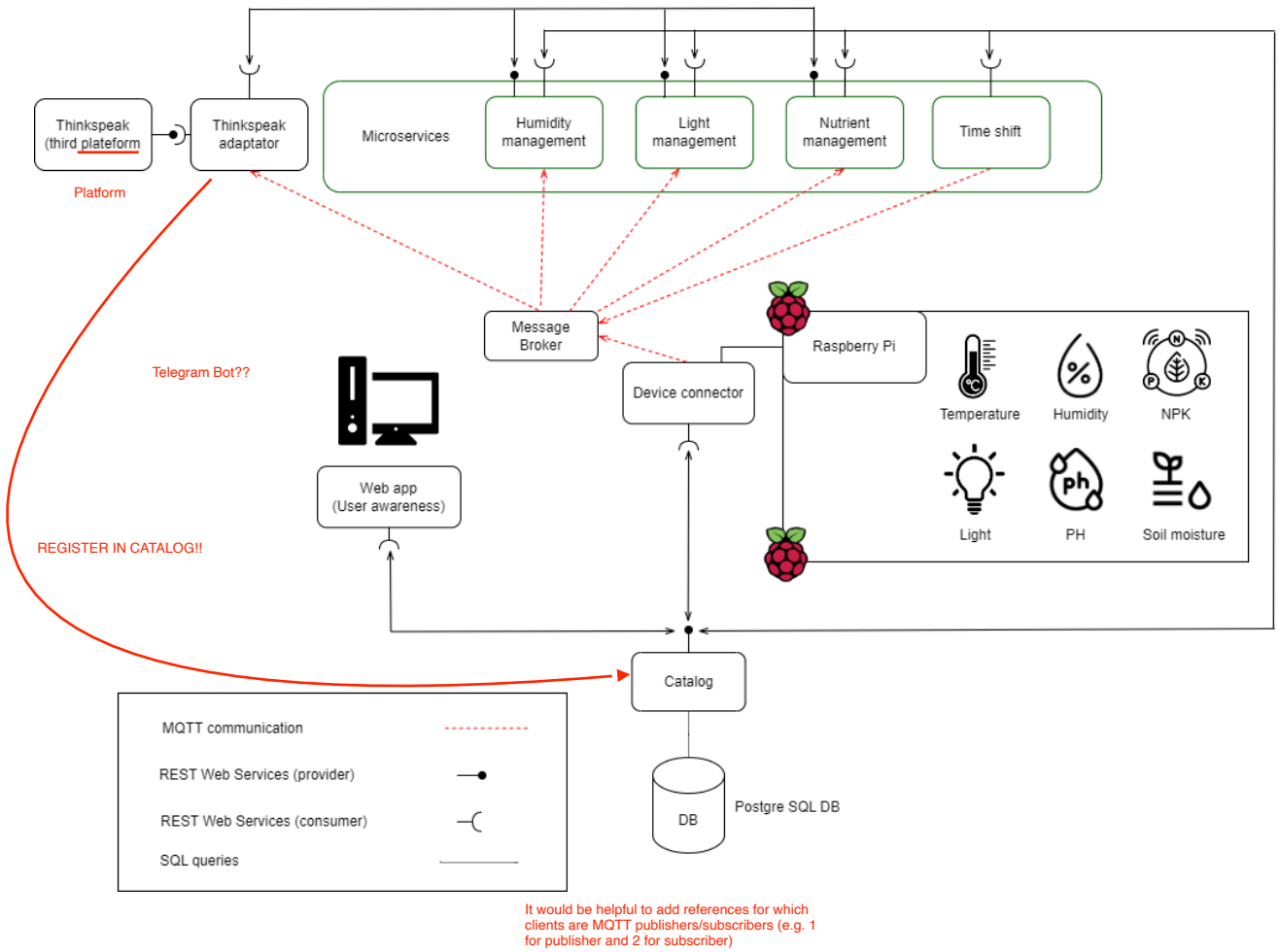
Scope and Objectives of Use Case	
Scope	The IOT-Enhanced smart farming aims to simplify plant care by automating the essential functions such as watering and monitoring soil health that send information to the owner. This process reduced the manual intervention and optimize plant growth.
Objective(s)	1. Monitor soil moisture, temperature, light and soil health level 2. Sends status and notifications on plant health via a Web application
Domain(s)	Smart Gardening, IOT agriculture
Stakeholder(s)	Home Gardeners, Urban Farming, IOT based greenhouse
Short description	The IOT based smart pot is equipped with sensors that can monitor the soil moisture and temperature plus the sufficiency of light. Besides that, the processes can be monitored by users via a web application.

- The data processing services are a bit too simple, try to add some additional services and maybe include some kind of statistics or more “complex” algorithm (not just if-else or for-loop). For each of the services remember to give and high level view of the algorithm and describe: (i)its inputs, (ii) its outputs

- Missing Telegram Bot

- Check because there are some functionalities described in the services, that are not matching (e.g. the possibility to send commands to device connector)

3. Diagram of Use Case



4. Complete description of the system

The proposed IoT platform for Smart Farming is designed using a microservices architecture and utilizes two communication paradigms:

1. **Publish/Subscribe** model based on the **MQTT protocol** to provide asynchronous communication.

2. **Request/Response** model using **REST Web Services** which provides the seamless interaction among system components.

In this setup, the following key actors are defined:

- The **Message Broker** is a server needed to apply the asynchronous publish/subscribe protocol
- The **Device Connector** receives environmental data collected by sensors and sends them to the microservices via MQTT as a subscriber. Gets configuration settings via REST web services by the Catalog. send data, working as MQTT publisher!
- The **Catalog** functions as a main repository for data, information and endpoints as well as configurations. It is accessed via Rest Web Services during the startup. It also operates as a service and device registry for all actors within the system. It also contains all kinds of data about the plant.
- The database stores the data accessible through the catalog.

Try to add some words, that the potential database is PostgreSQL, saying just what it is etc...

- Multiple **Microservices** are within the system, exploiting MQTT publish/subscribe and REST web services for communications :

- **Nutrient Management**, based on the collected data from the NPK sensors, we can help the user to monitor the soil status and nutrient level. Receives data with MQTT.

Steps :

- What is the data retrieved from thingspeak? (REST connection in the diagram)
- If notifications are sent using MQTT… missing mqtt connection in the UI service…

- Read Data from Sensors : Be aware this sensors are actually connected to the Device Connector!!
  - NPK Sensor: Monitors levels of nitrogen, phosphorus, and potassium in the soil.
  - Soil Humidity Sensor: Confirms soil has enough moisture for nutrient uptake. If the soil is too dry, the applied nutrients won't be accessible to plant roots, reducing their effectiveness.
  - PH sensor: Monitors the pH level of the soil with a probe. Most plants grow best in soils with a pH of 6.0 or 6.5 to 7.4. (7.0 is neutral, above is alkaline, under is acidic).

- Evaluate Conditions and notify the user : Notify the user how? MQTT? REST? Seems can be MQTT…

    - If NPK levels are below target thresholds, additional nutrients are required.
    - Ensure soil moisture is adequate before applying nutrients.
    - If the soil is too acidic or too alkaline, nutrients could be less available.

- **Humidity Management**, maintain optimal air and soil moisture levels. Receives data with MQTT.

  Steps:

  - What is the data retrieved from thingspeak? (REST connection in the diagram)
  - If notifications are sent using MQTT… missing mqtt connection in the UI service…

- Read Data from Sensors:
    - Humidity Sensor: Tracks air moisture levels.
    - Soil Humidity Sensor: Monitors soil water content.

- Evaluate Conditions and notify the user : Notify the user how? MQTT? REST? Seems can be MQTT…

    - If air humidity is too low, plants may experience increased transpiration and stress.
    - If soil humidity is too low, it indicates drought stress; if too high, it risks waterlogging and root rot.

- **Light Management**, ensure plants receive optimal light for photosynthesis and growth by notifying the user if detects any inconvenience. Receives and sends data with MQTT.

  - What is the data retrieved from thingspeak? (REST connection in the diagram)
  - If notifications are sent using MQTT… missing mqtt connection in the UI service…

  Steps:

- Read Data from Sensors :

    - Light Sensor: Measures light intensity and duration.
    - Temperature Sensor: Helps understand the interaction between heat and light.

- Evaluate Conditions, notify the user and act on the lights:

  - If light intensity is too low, artificial lighting may be required.
  - If light is excessive, reduce the light to prevent heat stress.
  - Regulate the temperature if not in the accepted range
  - Schedule the amount of hours of light per day

- The **Time Shift** is a control strategy to manage time-schedules provided by the catalog. It works as a MQTT publisher to send commands to IoT devices.

When described device Connector, the mqtt subscription was not mentioned to received commands....

- The **Web Application** is made to provide the user information about the state of the greenhouse. It communicates as an input with REST web services.

What about the notifications? No MQTT connection? Also you must include Telegram BOT!

- The **Thingspeak Adaptor** is an MQTT subscriber that receives environmental measurements and uploads them on **Thingspeak** through REST Web Services.

Thingspeak Adaptor also exposes REST APIs for other services to retrieve historical data...

- **Thingspeak** is a third-party software (https://thingspeak.com/) that provides REST Web Services. It is an open-data platform for the IoT to store and visualize data.

5. Desired Hardware components (only among those we can provide)

Device Name	Quantity	Needed for…
Temperature sensor	1+	Monitoring and regulating temperature level
Humidity sensor	1+	Monitoring humidity level
NPK sensor	1+	Monitoring nutrients present in the soil
Soil humidity sensor	1+	Monitoring soil moisture level
pH sensor	1+	Monitoring pH of the soil
Light sensor	1+	Monitor natural or artificial light
LEDs	1+	Regulate artificial lightning, ensuring optimal light exposure for growth.