D5.4 Central cloud-based data storage and analysis platform deployed

Project Information

Grant Agreement Number	101016706
Project Full Title	Photonic system for Adaptable muLtiple-analyte monitoring of fOod-quality
Project Acronym	h-ALO
Funding scheme	RIA
Start date of the project	1st January 2021
Duration	36 months
Project Coordinator	Stefano Toffanin (CNR-ISMN)
Project Website	https://h-alo.eu/

Deliverable Information

Deliverable n°	D5.4
Deliverable title	Central cloud-based data storage and analysis platform deployed
WP no.	5
WP Leader	7bulls.com
Contributing Partners	RISE, CONF
Nature	Other
Authors	Andrzej Tomczyk (7bulls), Michał Kłosiński (7bulls)
Contributors	/
Reviewers	Pranav Narayanan (RISE), Stefano Toffanin (CNR)
Contractual Deadline	30-06-2022
Delivery date to EC	24-07-2022

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016706".

Dissemination Level

PU	Public	**X**
PP	Restricted to other programme participants (incl. Commission Services)	
RE	Restricted to a group specified by the consortium (incl. Commission Services)	
CO	Confidential, only for the members of the consortium (incl. Commission Services)	

Document Log

Version	Date	Description of Change
V0.1	28/06/2022	First version
V0.2	12/07/2022	Comments and suggestions from reviewers
V1.0	20/07/2022	Final version

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016706".

Table of Contents

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016706".

1 Executive Summary

This document outlines the contents of the description and protocol of use of the central cloud-based data storage and analysis platform that will be implemented in h-ALO project.
Every output of the deliverable is endowed with technical documentation in form of *.md files (README) in respective repositories. This document focuses on reporting the high-level contents of the data storage approach and the overall architecture of the proposed solution. It also points to more detailed, technical information maintained together with the codebase.

2 Outputs

The complete software that has been developed consists of following components/repositories:

1. Frontend - https://gitlab.com/h-alo-public/halo-web
2. Backend - https://gitlab.com/h-alo-public/halo-backoffice

3 Architecture

The complete software consists of few interacting components. Each one has a specific role in creating the whole functioning system. High-level architecture of the system is presented on the diagram below (Figure 1).

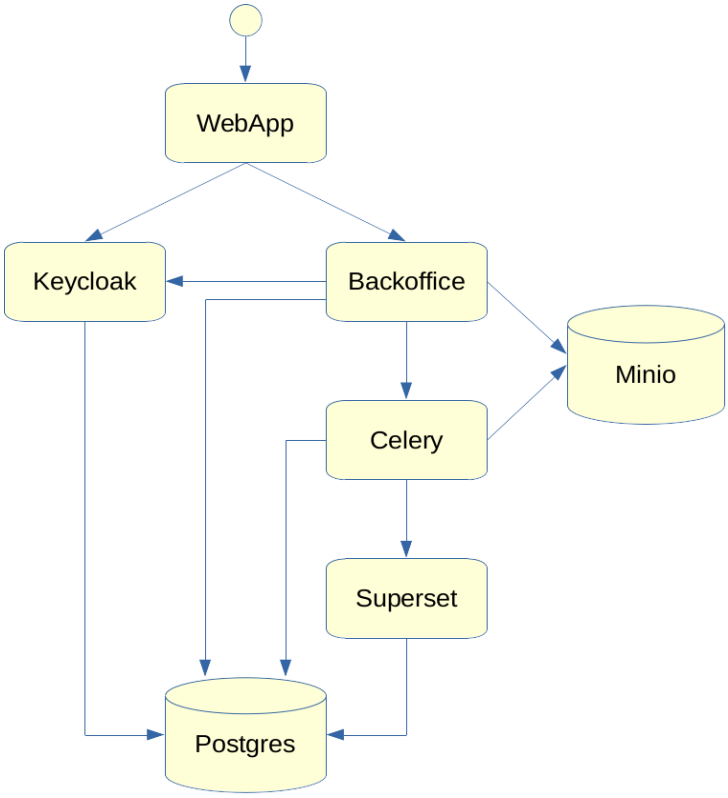

Figure 1 – System components

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016706".

Here below we report a short description of each single component:

- WebApp – React application presenting test/measurement results.
- Backoffice – REST Application Programming Interface (API) Java implementation providing access to data objects.
- Keycloak – Authentication provider.
- Celery – Task queue with Workers for data processing.
- Superset – Data visualisation application (charts/dashboards).
- Postgres – Database for common and processed measurement data.
- Minio– Database for raw measurement data.

More information about component interactions and workflows is in Chapter 5.

4 Web application (*Test Manager application*)

Web application (*Test Manager application*) allows logged users to search, view and modify test data from within their organizations. Test data are generated from raw sensor data during the "data processing" process described in Chapter 5.2.

The *Test Manager application* has been designed in accordance with end-users needs and expectations during Stakeholders Workshop. The Workshop was split into two sessions:

- The first session was held on May 4th, 2022 aiming at triggering an exchange of information and recommendations from stakeholders on safety and both risk-relevant and quality-relevant analytical needs in relation to chemical and microbiological hazards in the food value chains.

- The second session - a follow-up meeting was held online on May 27th, 2022 aiming at finalizing the discussion about the usability and operability of the sensor according to the direct experience of the end-users.

During the second session of the workshop, 7BULLS presented the *Test Manager application* to the end-users and inquired their opinion about:

- which the main goals of the *Test Manager application* should be;
- in which steps of the production process, it is more useful to register results of the testing through the h-ALO sensor;
- which specific features are needed by the end-users.

According to end-users' feedback, the *Test Manager application* should allow users to register results of tests for contaminants and to quickly identify if there is a problem. The testing frequency and procedure may vary depending on the food chain. As for aquaponics, results of testing should be registered routinely for each different fish tank, while in the case of milk testing could be useful after milking cows or where the milk is gathered in the tank. Honey could be tested during the harvest and in the tank of maturation. Beer could be tested during different steps of the process and most importantly when the final product is available. For each of these testing steps, the application should be able to record the data from the measurements.

The following diagram (Figure 2) shows numerous use cases provided by the *h-ALO Test Manager application*.

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016706".

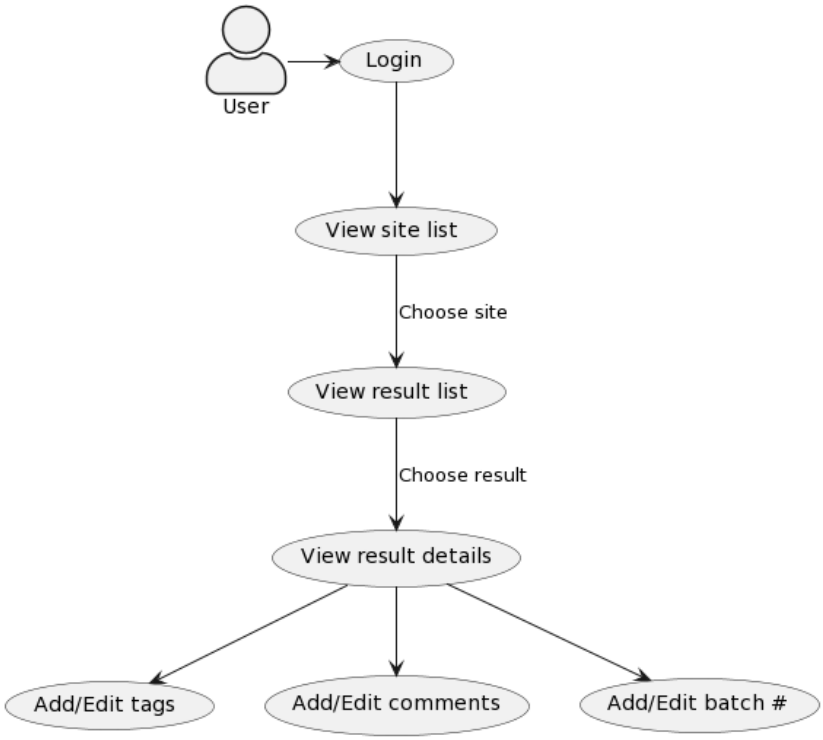

Figure 2 - WebApp use cases

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016706".

4. 1 Organization View

After login, the user is redirected to the Organization View screen. The user can see a list of sites in the form of white rectangular cards. Sites represent physical locations where test are performed (or where sensor data are gathered). Site cards are responsive - they wrap on smaller screens. They react on hover - change the background colour.

On each card user can see a small summary about each site (Figure 3):

- site name
- number of tests performed
- lowest result (i.e. analyte concentration value) achieved
- average result (i.e. analyte concentration value) achieved

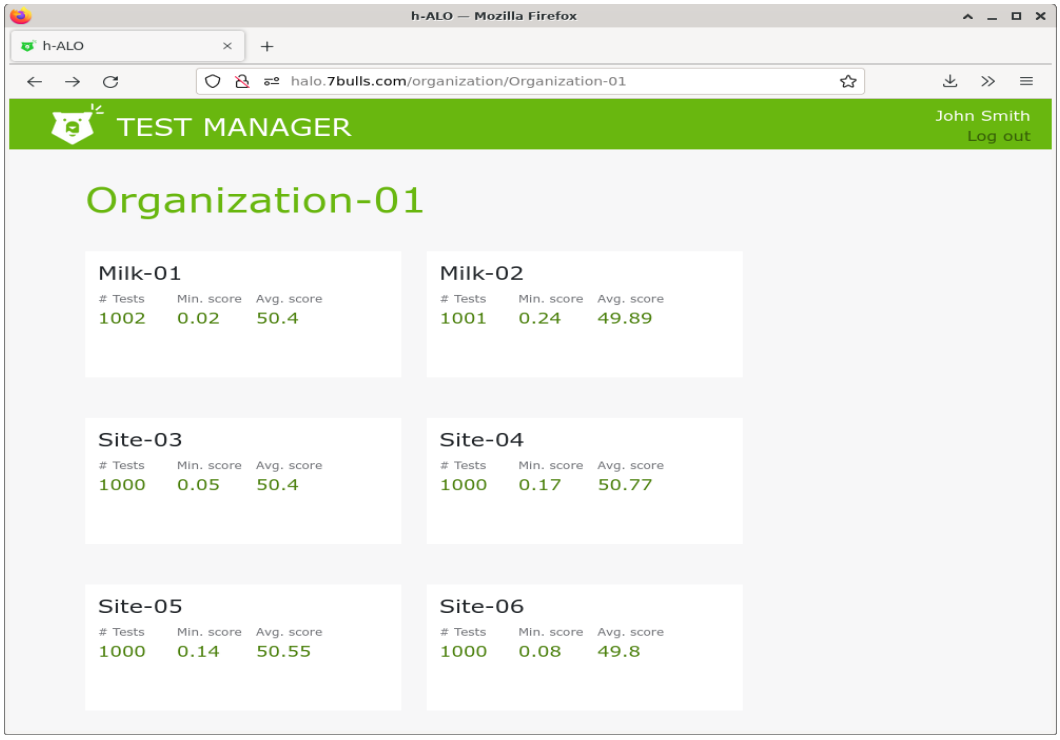

Figure 3 - Organization page, list of sites

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016706".

4.2 Site View

On clicking the site card on Organization View the user is directed to Site View. The URL points to this exact Site View. On screen, users can see the site name and the list of tests performed on this site (Figure 4). Tests are sorted by date and time descending (newest on top).

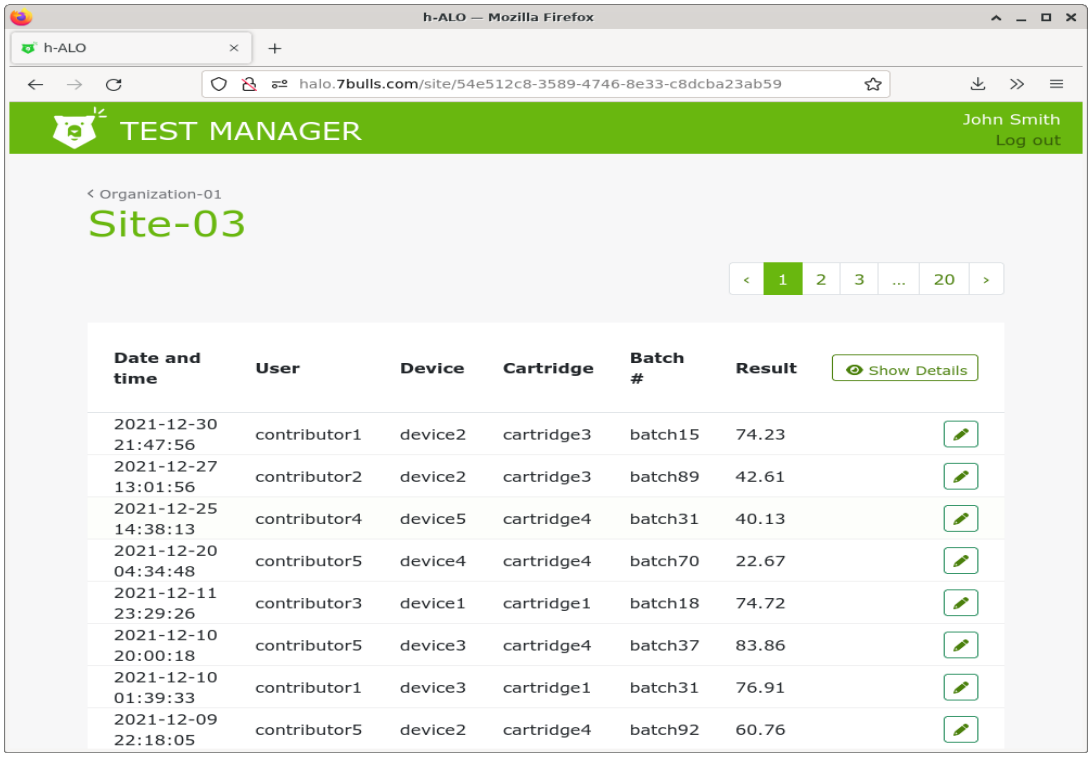

Figure 4 - Site page, result list

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016706".

4.3 Test Details

In order to be able to enrich the tests metadata and classify tests, user can edit comment and assign tags to the tests using (Figure 5):

1. On Site View, on each row user can click on the edit button and open the Test Details dialog.
2. On Test Details dialog user can see:
    a. all the data of the test
    b. can enter/edit comment
    c. can assign tags – system suggests tags from the set of all the tags already used on any of the tests in this organization.
3. On save, changes are saved and reflected in the table on Site View.
    a. with yellow fade effect on the table
4. On cancel, changes are forgotten.

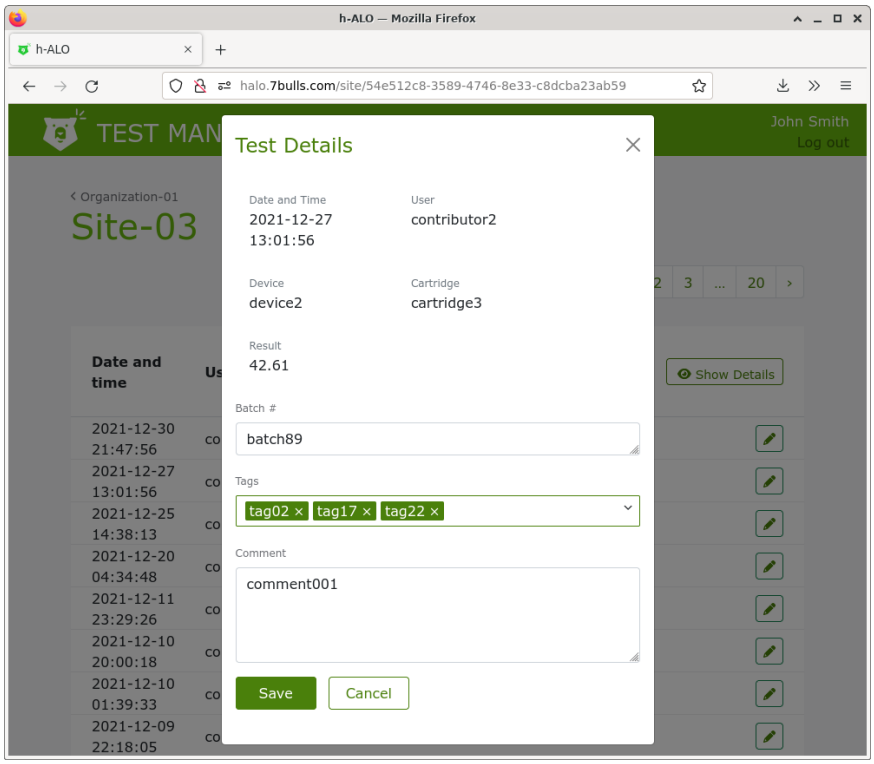

Figure 5 - Result details

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016706".

5 Backend components

5.1 REST API

Backend components provide specific functionalities but most of them are accessed by REST API (as an interface) implemented by Backoffice. Its main function is to provide data for WebApp with access control. Users and their credentials are managed by Keycloak. The typical API call (calling a specific function of the system) is presented below (Figure 6).

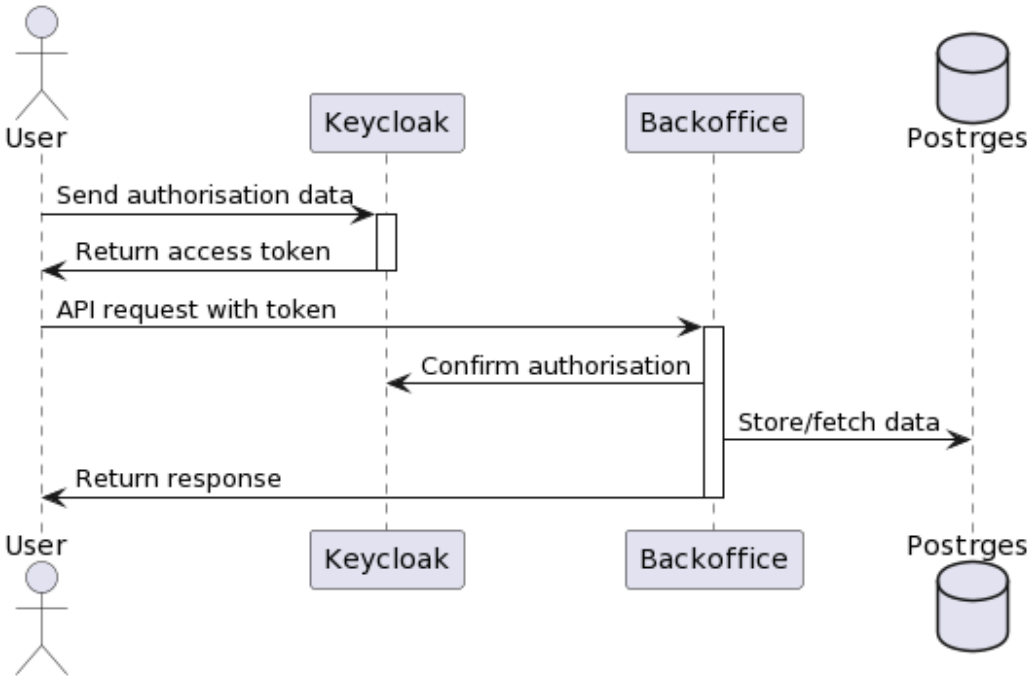

Figure 6 - Abstract REST API call sequence

Full specification of the API can be found in the repository: https://gitlab.com/h-alo-public/halo-backoffice/-/blob/main/backoffice/src/main/resources/backoffice_api.yaml

5.2 Data processing

Backend is also a processing framework for the data collected during measurement session by the sensor. The goal is to take data from a sensor and calculate human-readable results (i.e. values of the concentration of the analytes of interest).

It consists of following functionalities (in a cloud-based environment):

- Store raw data provided by sensors. Storage is provided by Minio.
- Process the data in a distributed manner. It is realised by Celery workers.
- Create data visualization in Superset.
- Create human-readable results to be presented by Webapp.

The sequence diagram in Figure 7 shows component interactions during the data processing.

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016706".

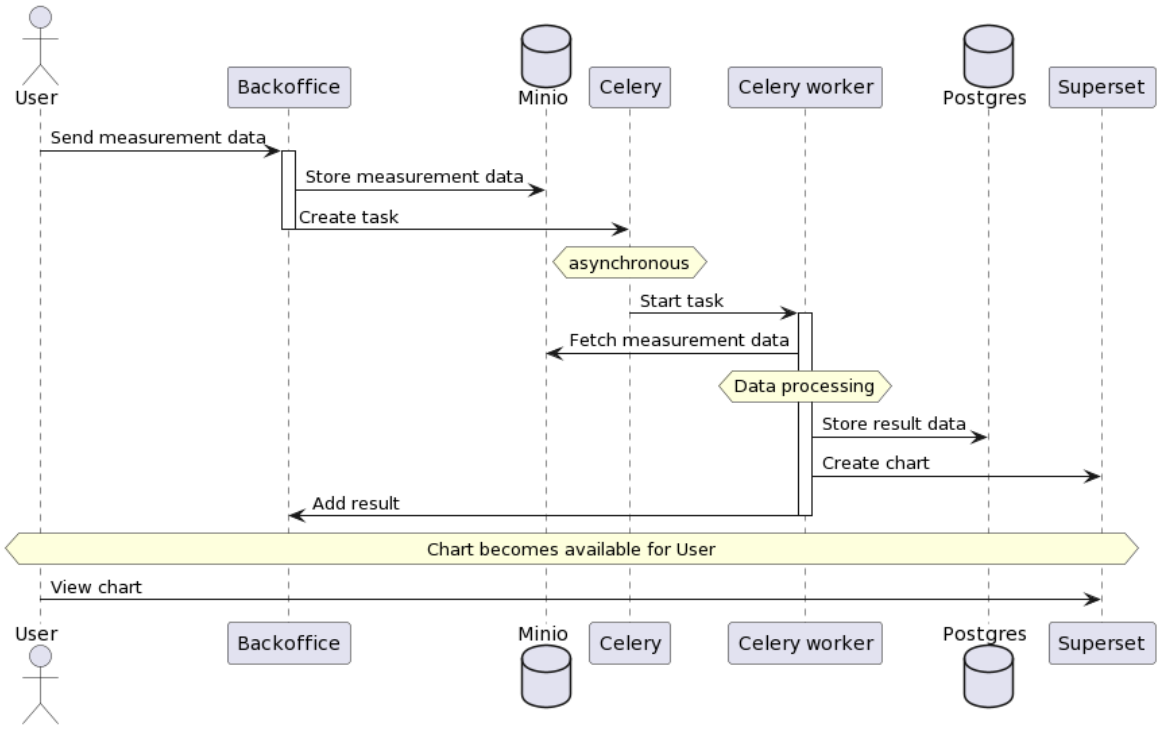

Figure 7 - Measurement data processing

Intention of such setup is to allow for any type of data processing. Starting from simple statistics ending with machine learning supported analysis. The appropriate and effective algorithm for performing data analysis will be defined within WP3 and WP4 by considering also the know-how from Plasmore, WFSR and CNR partners that has been developed in previous project (such as H2020 EU MOLOKO project).

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016706".

5.3 Superset

Superset is a reporting tool that allows us to represent data in a graphical way (charts, tables). Like the whole data processing subsystem, it is supposed to be flexible: it is not mandatory at this stage to define how the reporting dashboards will look exactly. However, we provided the tool that together with the data processing framework can address the needs of the future users.

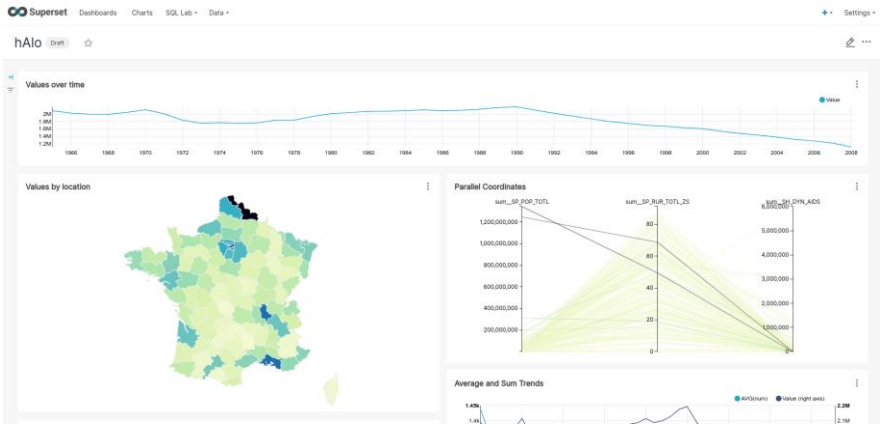

Figure 8 - example of Superset dashboard

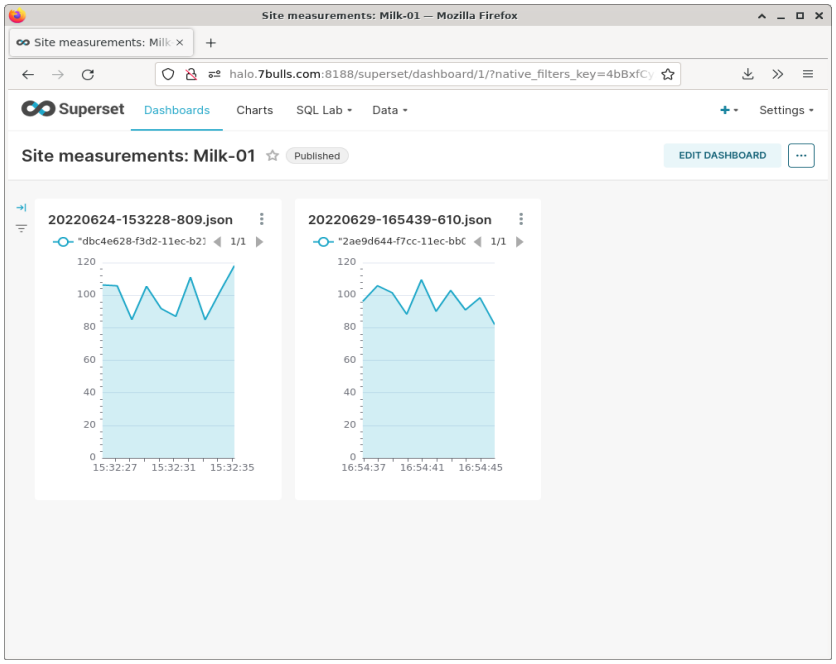

Figure 9 - example of Superset dashboard.

(NOTE: the data reported are not real, just random examples)

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016706".

6 Conclusions

The main goal of this task has been achieved. The complete software (cloud-based application) has been developed and delivered as a public repository in gitlab. Web application (*Test Manager application*) has been presented to the End Users. Feedback from End Users has been included in the newest version of the *Test Manager application*.

The work is continued in the next task (Task 5.5 – Mobile application development). The mobile application will be communicated with data collection device through Bluetooth. Collected data can be displayed in the application and sent to central data storage for further analysis. Also, some user data management can be possible from mobile application level.

7 References

REST API: https://en.wikipedia.org/wiki/Representational_state_transfer
Keycloak documentation: https://www.keycloak.org/documentation.html
Superset documentation: https://superset.apache.org/docs/intro/
Celery documentation: https://docs.celeryq.dev/en/stable/getting-started/introduction.html

"This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016706".