Intelligent Irrigation System for Low-cost Autonomous Water Control in Small-scale Agriculture

Deliverable D2.1b

Final report on specifications & functionalities of the edge-enabled sensor-gateway framework for smart irrigation system

Responsible Editor:	WAZIUP
Contributors:	UPPA
Document Reference:	INTEL-IRRIS D2.1b
Distribution:	Public
Version:	1.1
Date:	September 2023

CONTRIBUTORS TABLE

DOCUMENT SECTION	AUTHOR(S)
SECTION 1	C. Pham (UPPA) & A. Rahim (WAZIUP)
SECTION 2	F. Markwordt & J. Forster (WAZIUP)

DOCUMENT REVISION HISTORY

Version	Date	Changes
V1.1	Sep 14th, 2023	PUBLIC RELEASE
V1.0	Aug 31st, 2023	FIRST DRAFT VERSION FOR INTERNAL APPROVAL
V0.1	Aug 16th, 2023	FIRST RELEASE FOR REVIEW

Executive Summary

Deliverable D2.1b describes the edge-enabled IoT gateway framework that will be used in the INTEL-IRRIS project. It updates D2.1a and presents the results of Task 2.1 in WP2.

Table of Contents

1. INTRODUCTION & REVIEW

1.1. Edge-enabled sensor-gateway framework for smart irrigation system

The objective at the Edge-enabled IoT gateway is to have an autonomous gateway capable of collecting data from the soil devices and capable of displaying these data in a user-friendly manner. When targeting the smallholder farmers, the gateway should be able to simply display irrigation notifications on a small embedded device/screen, in addition to the more advanced and embedded web interface that would be accessible through a smartphone or a tablet. An example is provided below in the form of an illustration where the edge-enabled IoT gateway is the control part.

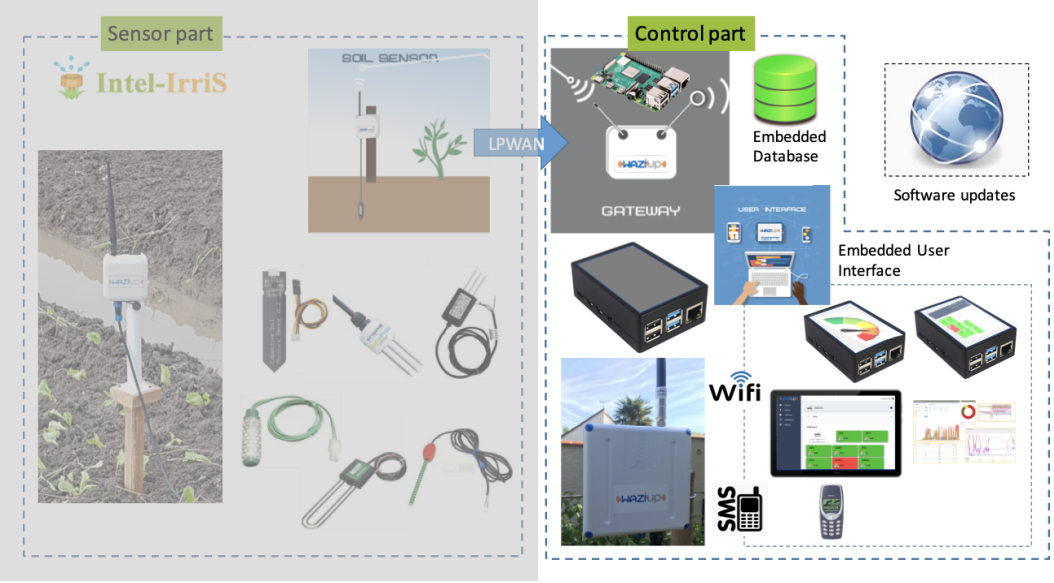

Figure 1 – The control part represented by the edge-enabled gateway

1.2. Specification & Functionalities

The INTEL-IRRIS's edge-enabled IoT gateway will be built on the low-cost, versatile, embedded and open IoT gateway expertise from both UPPA and WAZIUP. The main desirable features of the IoT gateway in INTEL-IRRIS, besides receiving sensor data from deployed soil devices, are:

- Implement the "Intelligent Irrigation in-the-box" with "plug-&-sense" approach
  - take into account the complex water-soil-plant interaction
  - embed Decision Support System (DSS), disruptive Artificial Intelligence (AI)
  - integrate various knowledge streams to present relevant indication and recommendations
  - Integrate of irrigation-specific software application modules including adaptable User Interface depending on smallholder farmers profile to provide all the necessary "control" components of a smart irrigation system
- Provide display capability of irrigation data for the end-user
- Integration of software modules for gateway maintenance and administration

2. The WaziGate V3

In this section, we'll present the new version of the WaziGate under development, version 3. This version will have the following features:

- New UI
- Better LoRaWAN connectivity
- Better apps integration
- Improved maintenance

This version has a completely new UI. Some screenshots will be presented in the following sections. It also improves the LoRaWAN connectivity. Finally, we also improved the apps integration and several maintenance aspects.

2.1. Overview

Designed to streamline data collection and management, the WaziGate offers a central hub for sensor integration, data transmission, and remote access. Its user-friendly interface and compatibility with various sensors make it an invaluable tool for monitoring and optimising agricultural processes.

Figure 2 – WaziGate indoor version

WaziGate is a LoRaWAN Gateway, ideal for remote IoT applications. The Gateway can cover up to 100 IoT sensing nodes using LoRaWAN radio network: Weather stations, soil monitoring, GPS applications... The Gateway can also control your actuators, such as electro-valves. You can host your own applications directly in the gateway, and connect to it through WiFi. The gateway can easily work without Internet connectivity and still provides data to end-users through its embedded database and web-based visualization module. The WaziGate features:

- Edge capacity to host your applications
- LoRa communication up to 10-12 Km
- Permanent Wifi hotspot
- Wifi/3G/Ethernet internet connection
- Low power consumption
- Automation
- Remote management

The WaziGate is connected and can be controlled from the WaziCloud, when the Internet is available. The following screen shows the WaziGate V3 main dashboard. It shows the general status of the gateway, including health, Cloud synchronisation, network connectivity, connected devices status and apps status.

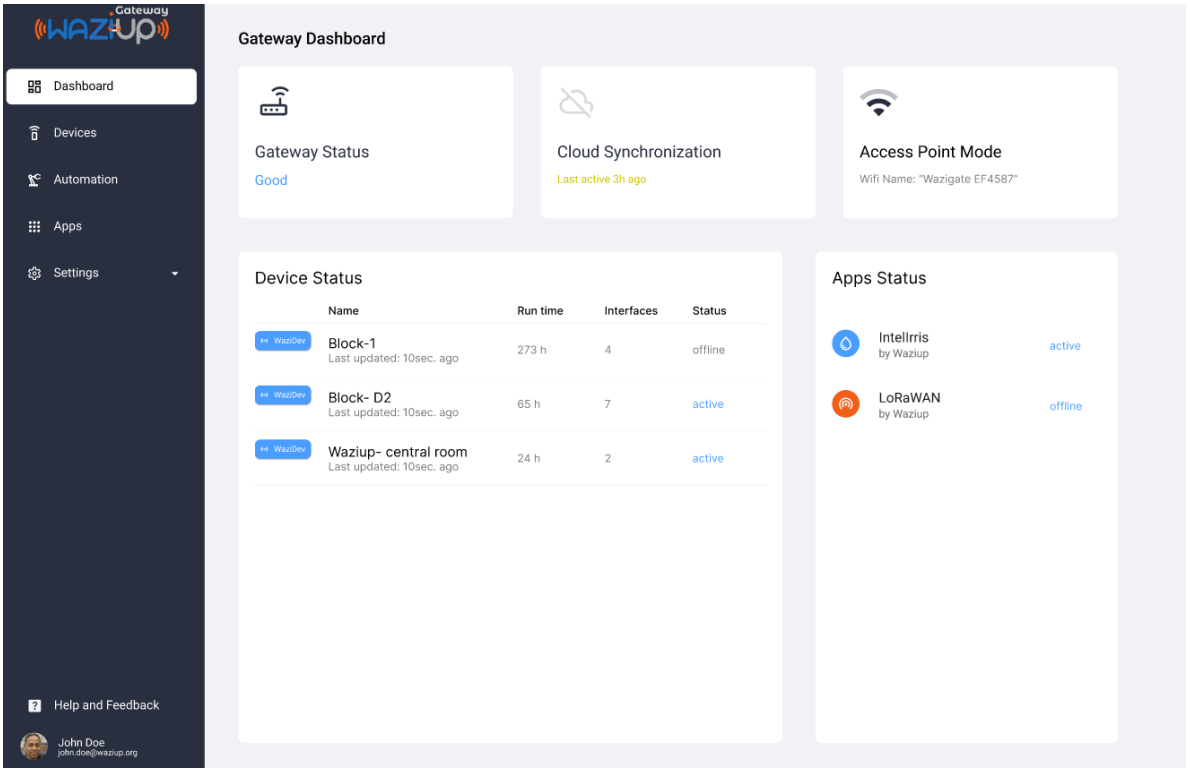

Figure 3 – WaziGate V3 Dashboard

2.2. Devices management

One of the main purposes of the WaziGate is to manage remote devices, connected via the LoRaWAN radio network. The following Figure shows the devices view, with devices status, and latest sensor readings. This screen also allows you to add additional devices and configure them (including LoRaWAN settings).

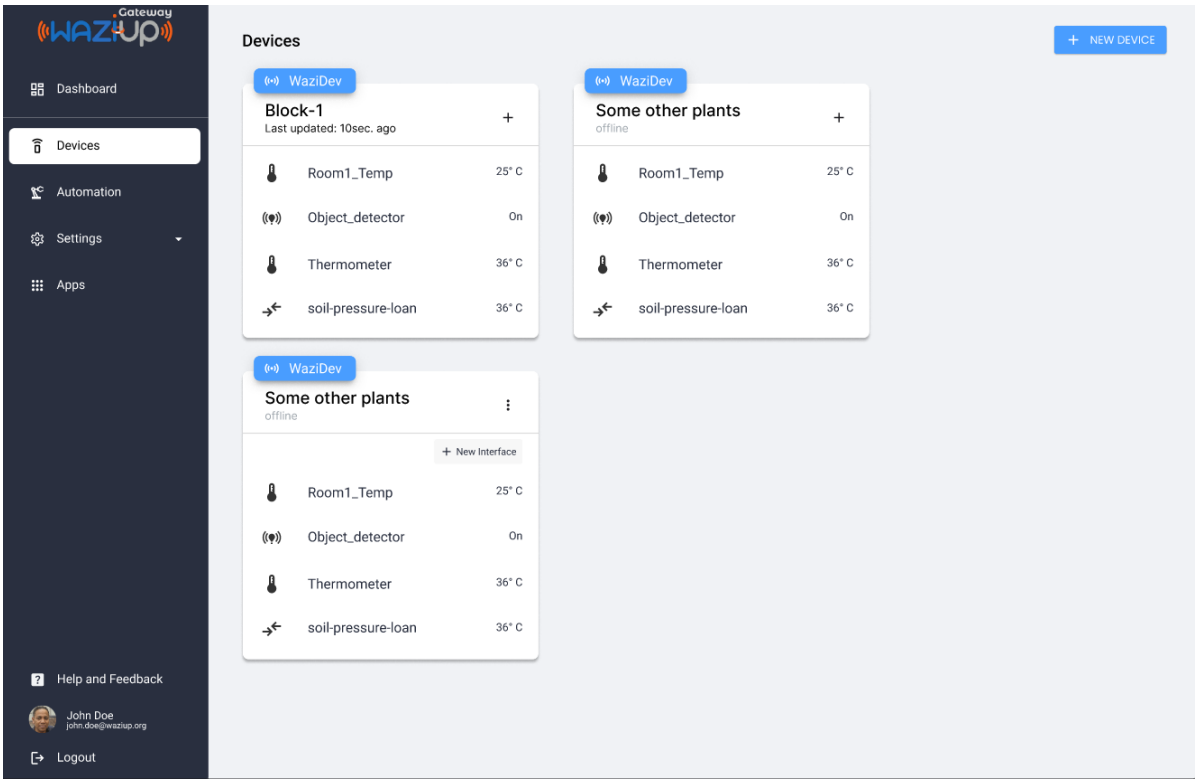

Figure 4 – WaziGate V3 Devices screen

The following Figure shows a specific sensor view, with sensor measurements graph and historical values. The settings of the sensor can also be accessed, allowing to configure parameters such as sensor type and synchronization interval.

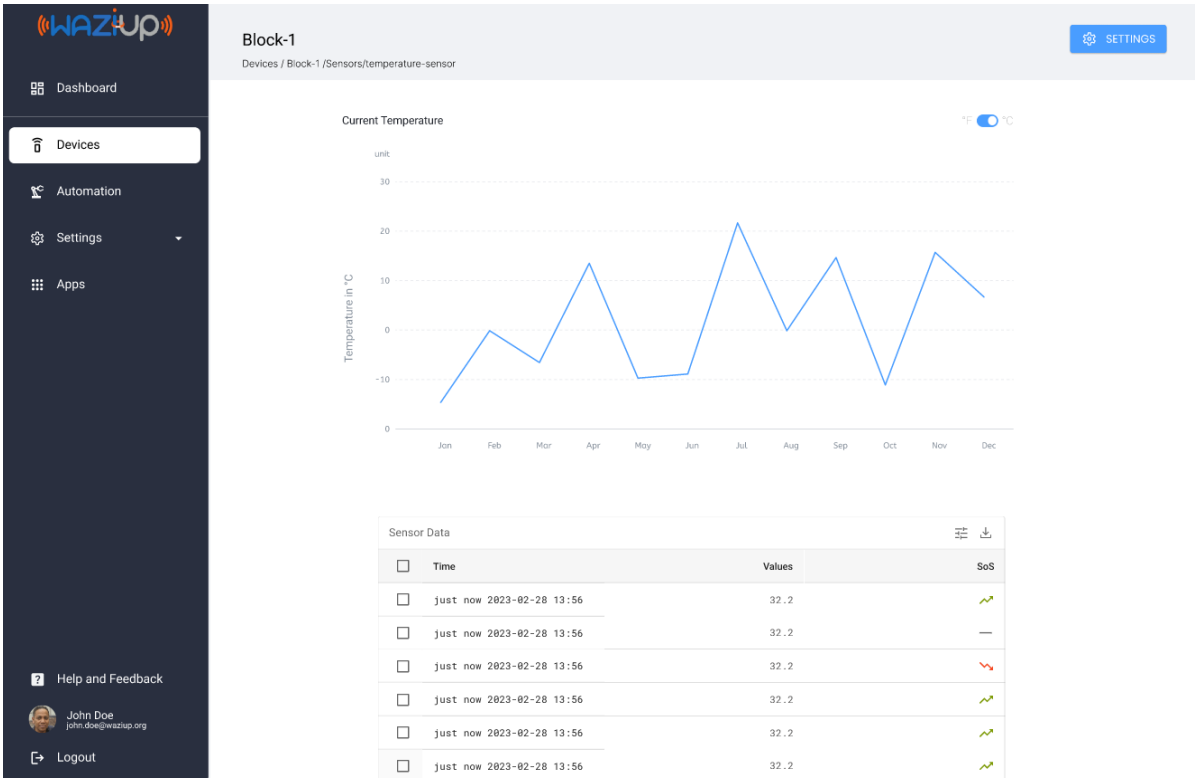

Figure 5 – WaziGate V3 Sensor view

2.3. Automation

The following screen shows the automation view, allowing the user to configure some simple automation. This automation allows to bind a sensor to an actuator with some simple logic. It should be noted that for more complex automations, a WaziApp should be developed (see next screen).

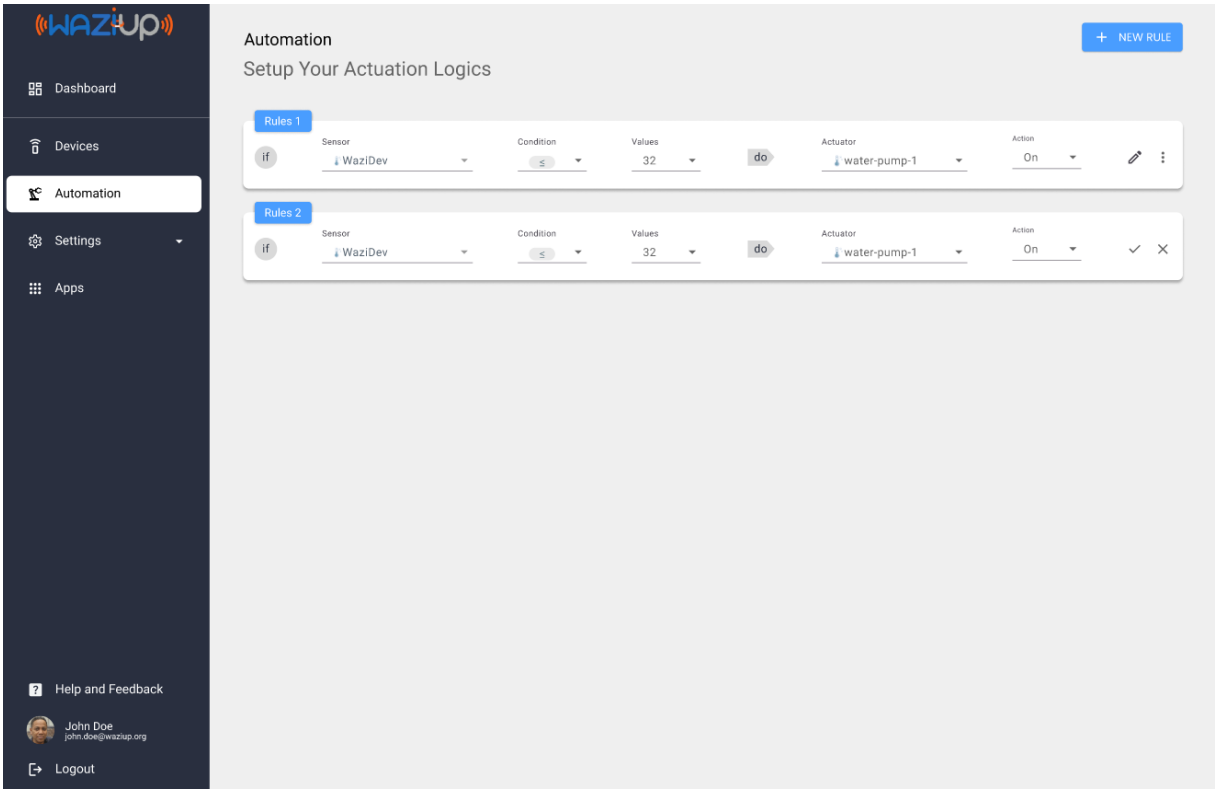

Figure 6 – WaziGate V3 Automation screen

2.4. Edge capacity

The WaziGate provides you with the ability to host your applications directly within the gateway itself. This offers a localised and responsive environment for your applications. Furthermore, Internet connectivity is not needed which is especially important for very remote applications. The following Figure shows the “WaziApps” screen. With this screen, you can see the WaziApps currently running, configure them and add new ones.

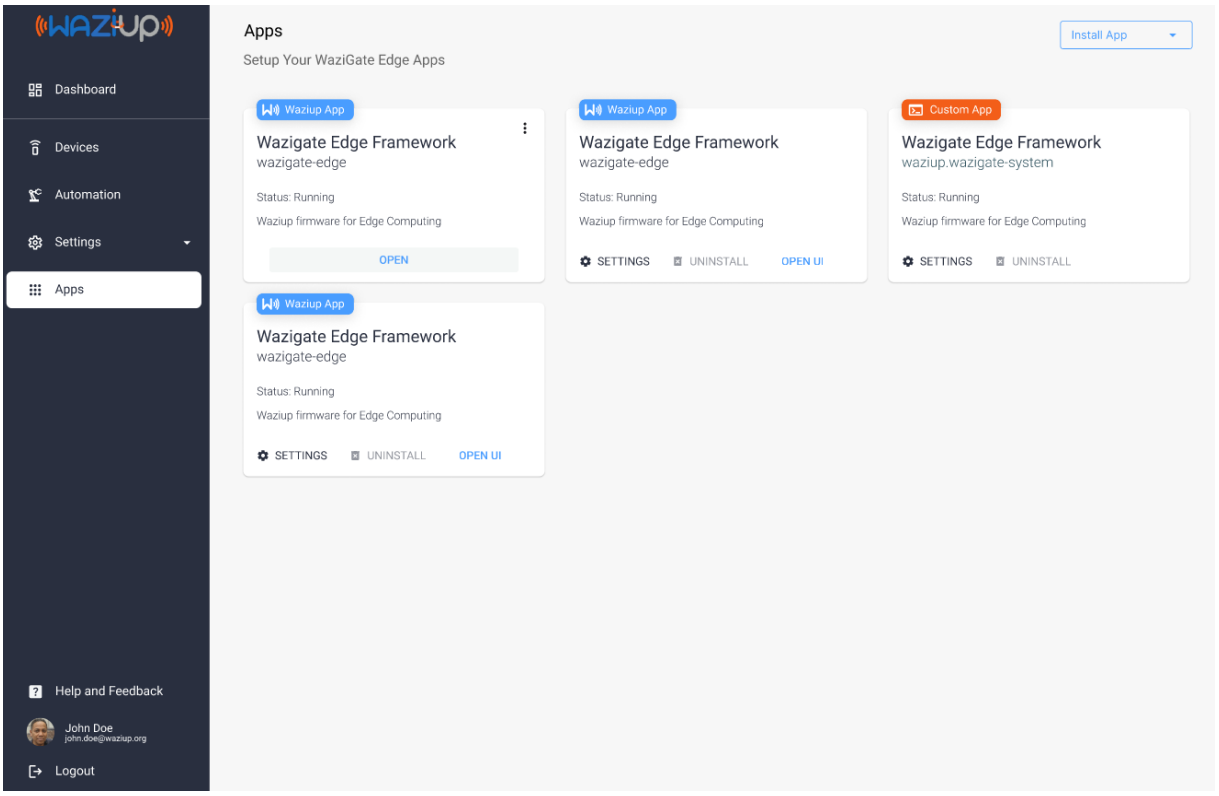

Figure 7 – WaziGate V3 Apps screen

WaziGate extends its reach with LoRa communication spanning up to 10-12 kilometres, providing expansive coverage for your IoT applications. Additionally, the gateway establishes a permanent WiFi hotspot, facilitating seamless device connectivity.

WaziGate also showcases low power consumption, optimizing its performance while minimizing energy usage. This efficient approach contributes to sustainable and prolonged operational periods.

2.5. LoRaWAN

2.5.1. Over the Air Activation

This new version will include better LoRaWAN support. In particular, it will support “Over the Air Activation” (OTAA). More and more commercial devices support OTAA. Devices need to perform a join-procedure with the network, during which a dynamic DevAddr is assigned and security keys are negotiated with the device. In the case of OTAA with LoRaWAN V1.0, the following fields are necessary on the UI: DevEUI and AppEUI and AppKey.

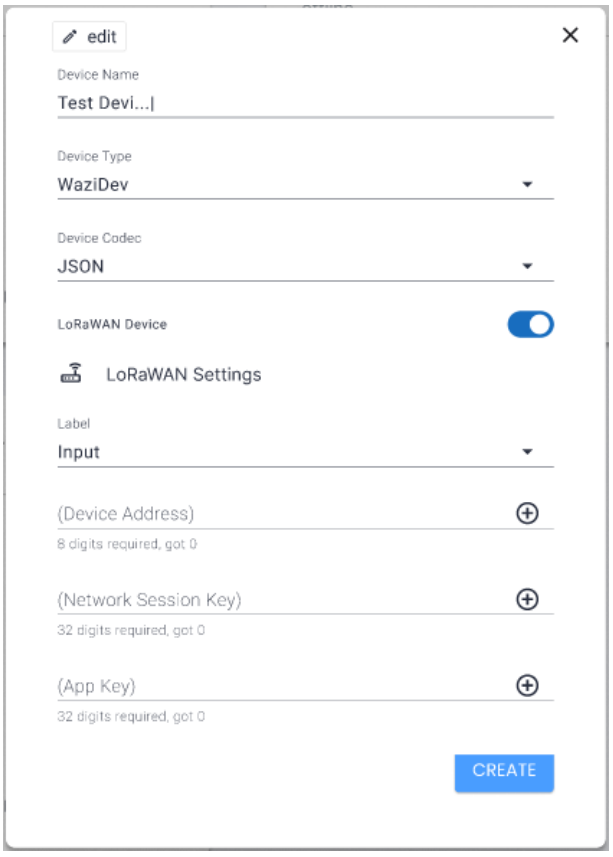

Figure 8 – LoRaWAN ABP settings

2.5.2. Multi-channel concentrators

The WaziGate will include support for other multi-channel LoRA concentrators, such as RAK HATs. This requires adding the new packet forwarder, updating the RESET pin in the startup script and selecting the correct configuration file for either single or multi channel.

On the UI we will have a configuration section where a user selects a HAT on their WaziGate. This information will be communicated to wazigate-lora and the concentrator will be started. The selected HAT should also be saved locally so that the WaziGate starts automatically after booting up.

2.6. Maintenance

2.6.1. SSH terminal

We integrated a new graphical terminal into the UI of the WaziGate. This terminal allows access to the WaziGate through a command line interface, using the SSH protocol. This is very important for diagnosis and maintenance.

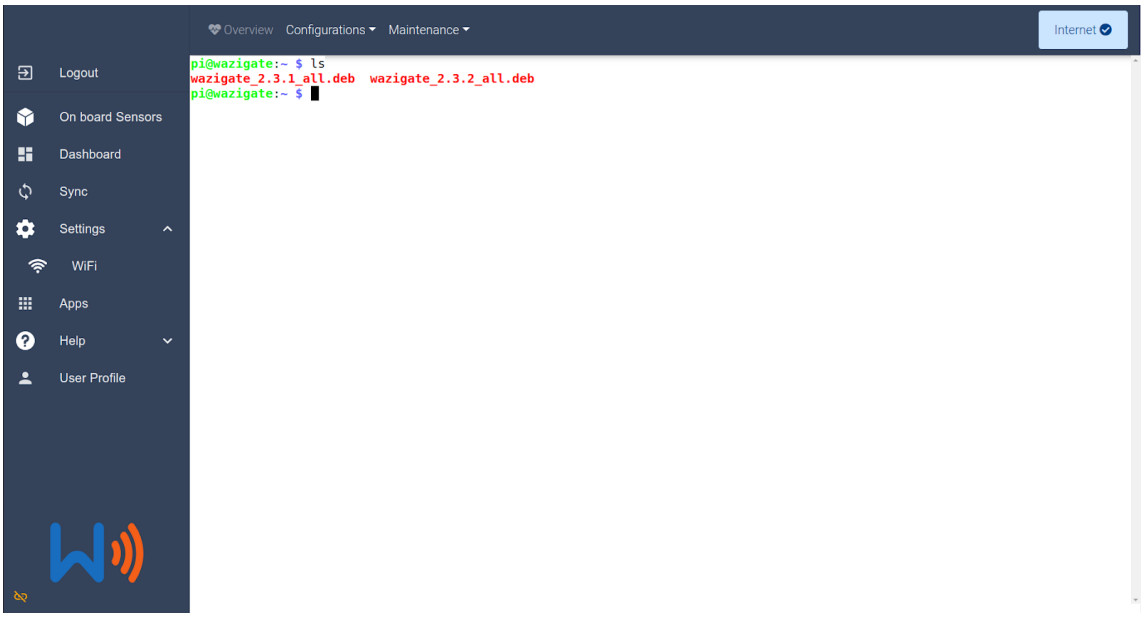

Figure 9 – SSH login in the WaziGate UI

This feature is meant to be accessed through the "tunnel" from WaziCloud. In this way, maintenance can be done on a WaziGate from remote. This is particularly useful in situations where the WaziGate is in the private network, inaccessible from the Internet.

2.6.2. Data retrieval

To enhance the utilisation of accumulated data from the gateway for machine learning tasks, we have streamlined the process of data retrieval. With just one click, you can now download all the necessary data from a single gateway. We offer three convenient options to cater to your specific requirements:

1. Tree of CSV Files: Export the data of all sensors and actuators into a well-organised tree structure of CSV files. This allows for easy access and management of individual sensor and actuator data.
2. Single CSV File: Export the data of all sensors and actuators into a single comprehensive CSV file. This consolidated format simplifies data handling and analysis.
3. Custom Time Spans and Time Bins: In addition to the above options, we provide the ability to export data with custom time spans. Furthermore, the data can be summarised into time bins, making it particularly valuable for machine learning applications. This feature enables efficient data preprocessing and facilitates seamless integration into ML workflows.

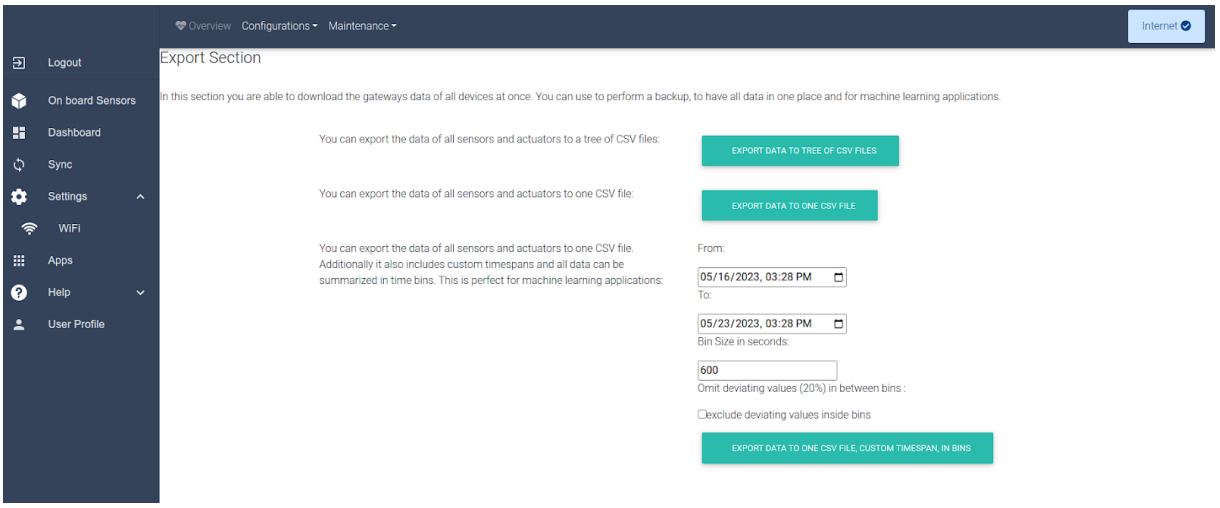

Figure 10 – Data export options

This functionality proves especially valuable when dealing with numerous sensors connected to the gateway. By eliminating the need for manual API calls or individual sensor downloads through the user interface, it saves time and effort. This unlocks the power to use accumulated data effortlessly with our comprehensive data management solution.

Another feature added is the export function for data in the WaziGate. It is now possible to export the data as images (SVG and PNG) and tabulated data (CSV). Bulk data export has also been implemented (see next section).

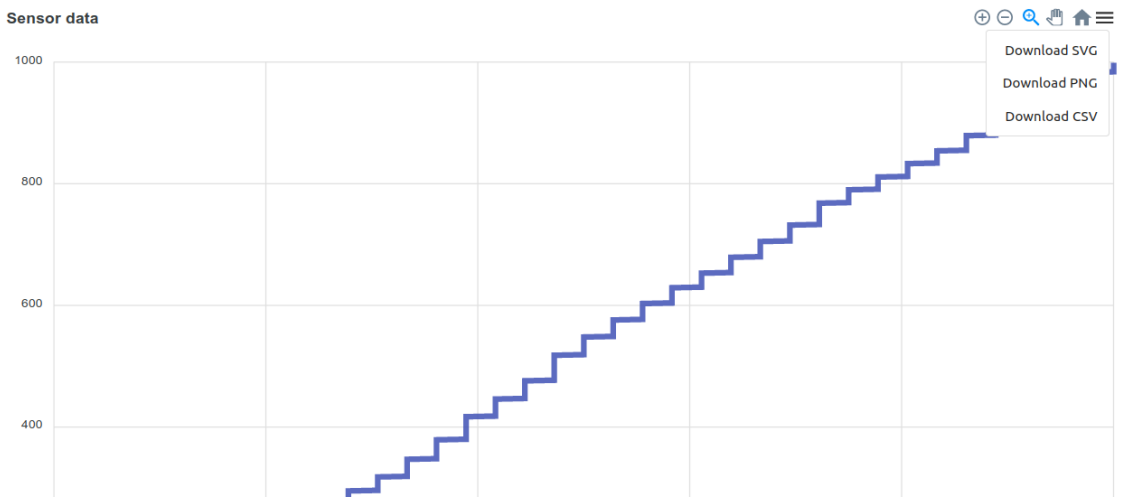

Figure 11 – Exporting graphs

2.6.3. Clock and timezone management

Another important feature that was added is to set the clock and timezone automatically. This is done using the Network Time Protocol (NTP) during boot time. This is an important feature, because all data incoming from sensors is time stamped by the gateway upon arrival. If the timestamp is wrong, the data collected might be unusable.

2.6.4. Performance improvements

The previous release of the Gateway (V2.3.0) included technical and performance improvements:

1. Memory performance improvement
2. CPU Performance improvement
3. Better Wifi support
4. Transition to 64 bits architecture

Those improvements greatly improved the user experience: faster boot time, less SD card wear, less heat generated, more stable internet connection. This version also introduced the display of sensor historical values.

2.7. Embedded AI framework

For development purposes, we have implemented AI models directly on the gateway to facilitate seamless experimentation. To simplify this process, we have integrated an application that runs JupyterLab in a docker container, equipped with essential machine learning packages pre-installed. Additionally, we provide convenient scripts for effortless installation of additional packages. These scripts serve as valuable examples, demonstrating how to utilise various functionalities such as leveraging linear regression to solve problems or dynamically accessing sensor and actuator values through the gateway's API.

Through our comprehensive AI-driven solutions, we empower users to explore the full potential of their data, enhancing efficiency, and facilitating informed decision-making.

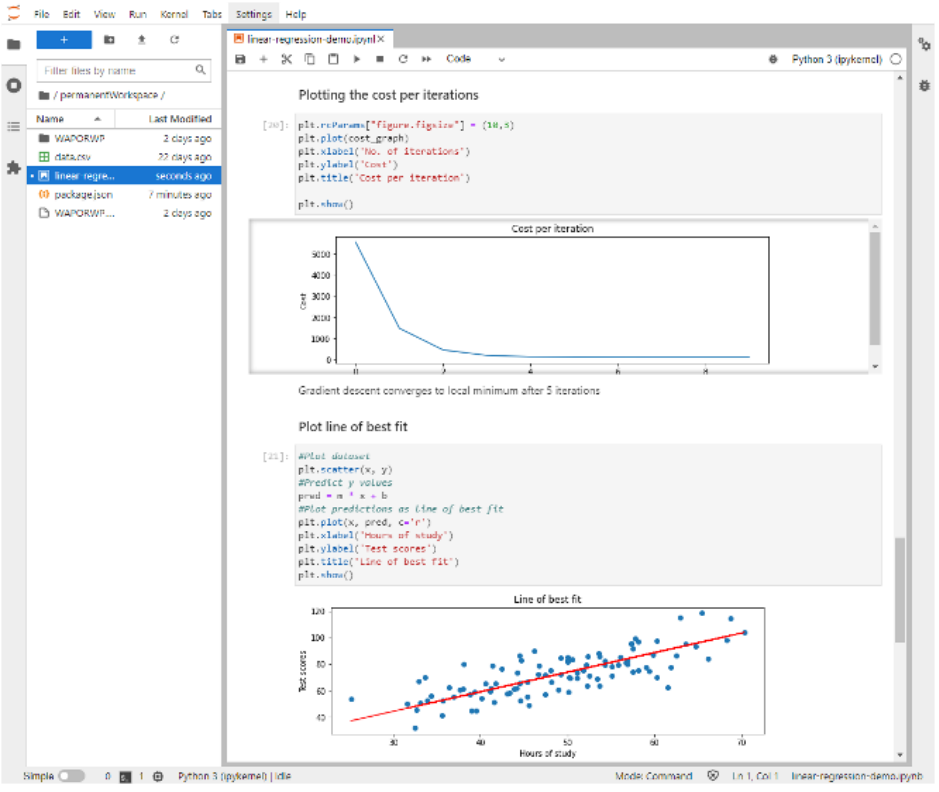

Figure 12 – Embedded JupyterLab environment

Acronyms List

Acronym	Explanation
AI	Artificial Intelligence
API	Application Programing Interface
CSV	Comma-Separated Values
DSS	Decision Support System
GPS	Global Positioning System
ML	Machine Learning
NTP	Network Time Protocol
RTSP	Real Time Streaming Protocol
SSH	Secure Shell
UI	User Interface

Project Co-ordinator Contact

Pr. Congduc Pham

University of Pau

Avenue de l'Université

64000 PAU

FRANCE

Email: Congduc.Pham@univ-pau.fr

ACKNOWLEDGEMENT

This document has been produced in the context of the PRIMA INTEL-IRRIS project. The INTEL-IRRIS project consortium would like to acknowledge that the research leading to these results has received funding from the European Union through the PRIMA program.