AgroSense_Soil & Air EC-TH Pro

LoRaWAN® Manual

V1.0

Author: Yuki

Time: 2025.10.23

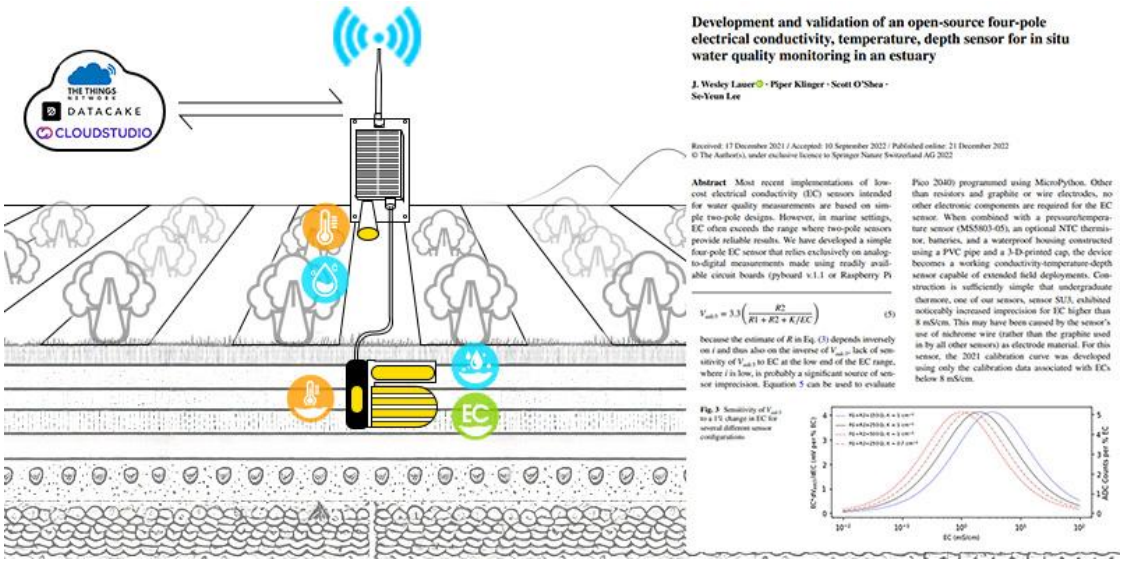
Contents

1 Product Description

1.1 Introduction

In modern agricultural production, monitoring soil and air conditions is crucial. Soil moisture directly affects irrigation efficiency, while soil electrical conductivity (EC) reflects salinity and fertility levels that are essential for crop growth. At the same time, air temperature and humidity are key factors influencing plant health and yield. The AgroSense_Soil & Air EC-TH Pro LoRaWAN® provides an all-in-one solution, enabling farmers and researchers to track these critical parameters accurately and remotely for smarter agricultural management.

The AgroSense Soil & Air EC-TH Pro sensor combines soil Temperature / Humidity / EC Sensor with Air Temperature & Humidity Sensor, measures soil humidity at the range of 12-bit ADC, -40℃ to 80℃, temperature and humidity in the atmosphere at the range of -40℃ to 85℃ and 0 to 100 %RH .



Soil EC Sensor adopts a four-electrode (four-pole) measurement design, measures at the range of 0-20000μS/cm， with 1μS/cm resolution which applies current through two outer electrodes and senses voltage via two inner electrodes. Compared with traditional two-pole probes, this method is far less affected by electrode polarization or fouling, ensuring higher accuracy and stability in long-term field deployments. Reference Environ Monit Assess (2023) Development and validation of an open-source four-pole electrical conductivity, temperature, depth sensor by J. Wesley Lauer from SEATTLE UNIVERSITY.

Development and validation of an open-source four-pole electrical conductivity, temperature, depth sensor for in situ water quality monitoring in an estuary

J. Wesley Lauer · Piper Klinger · Scott O'Shea · Se-Yeun Lee

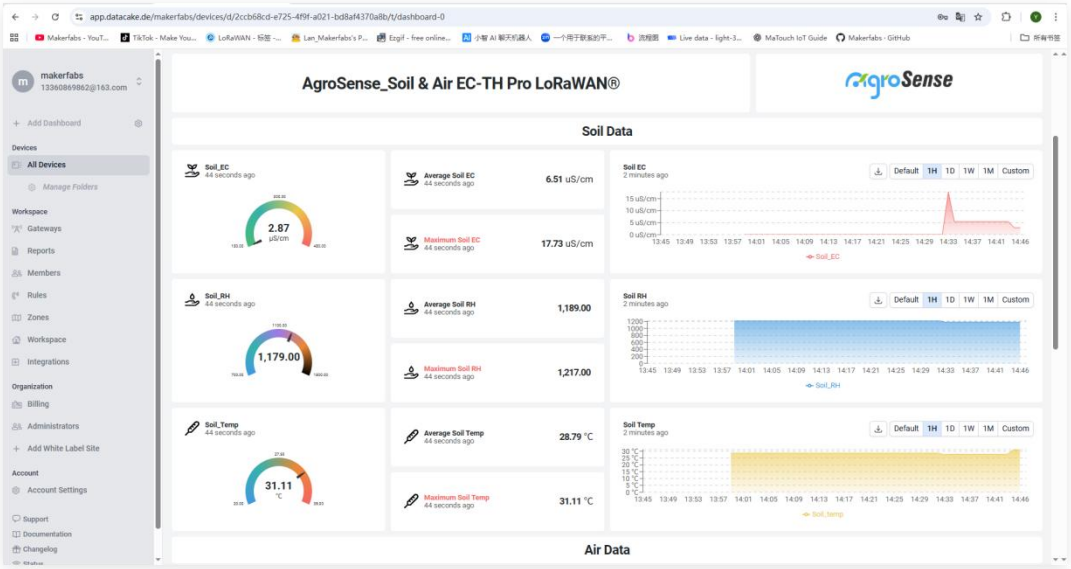
Received: 17 December 2021 / Accepted: 10 September 2022 / Published online: 21 December 2022

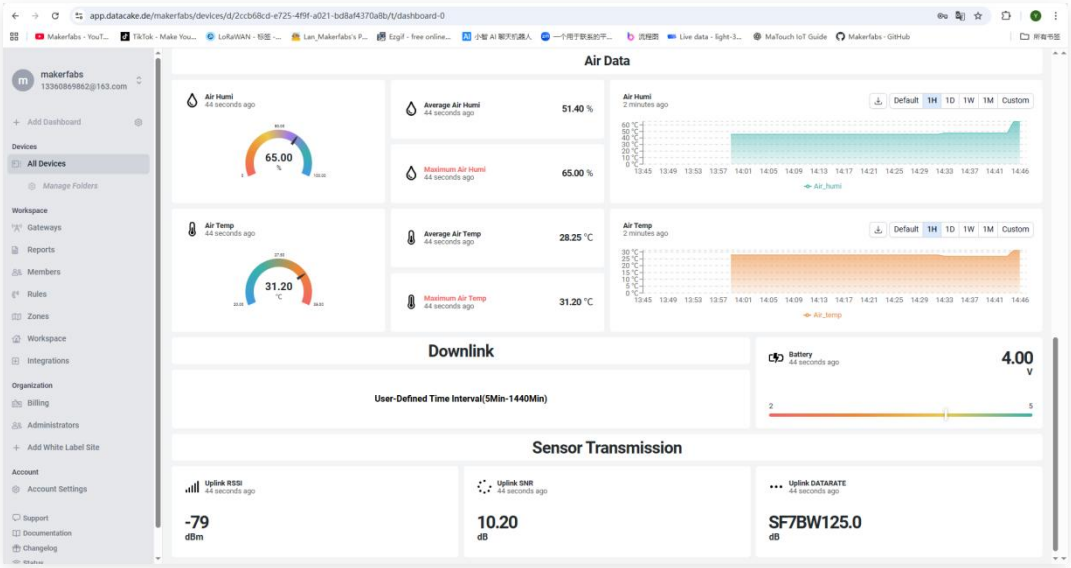
Abstract Most recent implementations of low-cost electrical conductivity (EC) sensors intended for water quality measurements are based on simple two-pole designs. However, in marine settings, EC often exceeds the range where two-pole sensors provide reliable results. We have developed a simple four-pole EC sensor that relies exclusively on analog-to-digital measurements made using readily available circuit boards (pyboard v.1.1 or Raspberry Pi Pico 2040) programmed using MicroPython. Other than resistors and graphite or wire electrodes, no other electronic components are required for the EC sensor. When combined with a pressure/temperature sensor (MS5803-05), an optional NTC thermistor, batteries, and a waterproof housing constructed using a PVC pipe and a 3-D-printed cap, the device becomes a working conductivity-temperature-depth sensor capable of extended field deployments. Construction is sufficiently simple that undergraduate science students can construct one during three 3-h lab periods. Lab calibrations performed on several prototypes at ECs between 0.18 and 45 mS/cm show that confidence limits as good as about ±3% of EC are possible. Re-calibration of several prototypes 1 year after initial calibration shows that long-term calibration drift is modest. Data collected by the pro-

J. W. Lauer (✉)
Department of Civil & Environmental Engineering, Seattle University, 901 12th Avenue, Seattle, WA 98122, USA
e-mail: lauerj@seattleu.edu

P. Klinger · S.-Y. Lee
Department of Civil & Environmental Engineering, Seattle University, Seattle, WA 98122, USA

This product is based on LoRaWAN. We have cooperated with cloud service vendors such as ThingSpeak and Datacake, allowing users to remotely monitor and control the device via the cloud.





Also, this product cased with IP67 case, solar panel powered, can be used long-term in filed application.

Benefits from LoRaWAN®, which ensures stability and reliability. It is capable of covering a long transmission range while maintaining low power consumption. Unlike wireline devices, it is by dual lithium batteries and a solar panel, the sensor achieves ultra-low power consumption, enabling more than 10 years of continuous operation, reducing the workload and complexity of deployment, design and development for end-users that can work via powering it, and setting the configuration in the cloud server.

1.2 Feature

- Includes a **high precision** sensor.
- LoRaWAN version: LoRaWAN Specification 1.0.3. OTAA **Class A.**
- Monitor data and upload **real-time** data regularly.
- Modify the product parameters through **AT commands.**
- Support **downlink** to modify the time interval.
- Compatible with Worldwide **LoRaWAN® Networks: Support** the universal frequency bands EU868/ US915.
- **Long Range:** Up to 2 kilometers in the city, up to 10 kilometers in the wilderness, receive sensitivity -137dBm , transmit power up to 22dBm.
- **Data encryption:** Provide end-to-end secure communication, including device authentication and network data encryption, to ensure the security of data transmission and prevent data theft and malicious attacks.
- **High stability and reliability:** good stability in noisy environments, able to penetrate

buildings and obstacles, so it can maintain good communication quality in urban and suburban environments.

1.3 Parameter

1. General Parameters

Product Model	AGLWSAEC
Air Temperature Measurement Range	-40°C ~85°C
Air Temperature Measurement Accuracy	±0.3°C
Air Temperature Resolution	0.01°C
Air Humidity Measurement Range	0%-100% RH
Air Humidity Measurement Accuracy	±2%
Air Humidity Resolution	0.024% RH
Soil Temperature Measurement Range	-40°C ~80°C
Soil Temperature Measurement Accuracy	±0.5°C
Soil Temperature Resolution	0.1°C
Soil Humidity Measurement Range	0%-100% RH
Soil Humidity Measurement Accuracy	±2%
Soil Humidity Resolution	0.1% RH
EC Measurement Range	0-20000μS/cm
EC Resolution	1μS/cm

2.Wireless Parameters

Communication Protocol	Standard LoRaWAN® protocol V1.0.3
Network Access/Operating Mode	OTAA **Class A**
MAX Transmit Power	22dBm
Receiver Sensitivity	-137dBm/125kHz SF=12
Frequency Band	EU868/US915

3.Physical Parameters

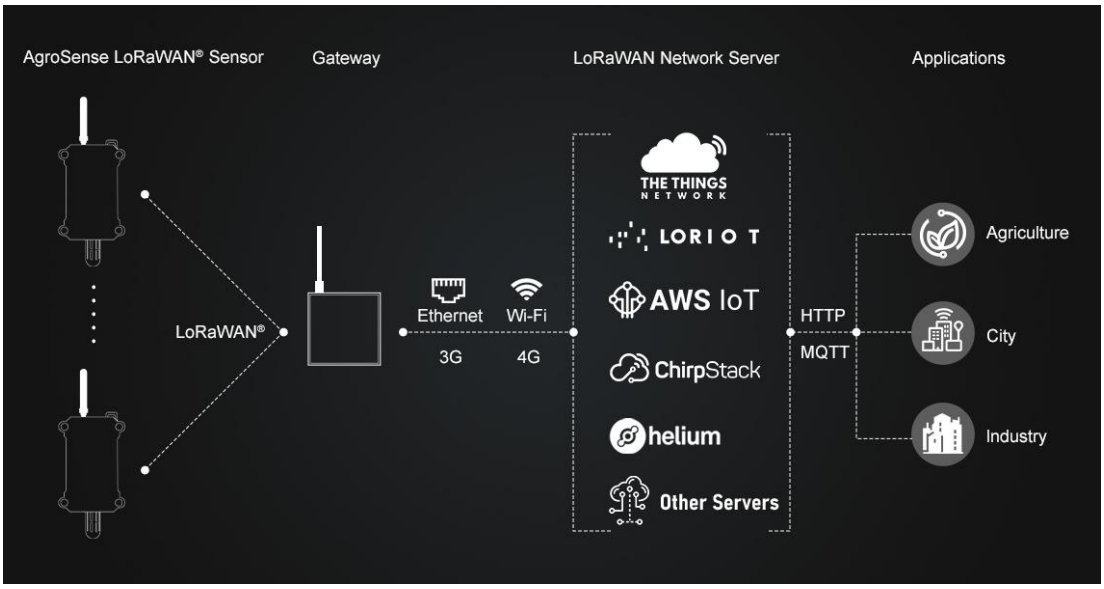
Batteries Power Supply	2 x 18650 3.7V Lion batteries
Solar Power Supply	5V2W
Operating Temperature	-40°C ~85°C
Protection Class	IP67
Dimensions	115 × 85 × 35 mm
Mounting	Wall Mounting

2 Technical route

2.1 System Framework

AgroSense_Soil & Air EC-TH Pro uses LoRAWAN technology, and it network architecture includes four parts: End Nodes, Concentrator/Gateway, Network Server and Application Server.

End Nodes	It is responsible for collecting sensing data and then transmitting it to Gateway via the LoRaMAC protocol.
Concentrator/Gateway	It is mainly responsible for transmitting node data to the server.
Network Server	Organize the data into JSON packets and decode them.
Application Server	Display the data.



Uplink:

1.Data Collection & Transmission

Sensor data and transmits it to the Gateway via LoRaWAN® protocol.

2.Gateway Forwarding

The Gateway packages the raw data and forwards to the Network Server.

3.Data Decoding & Routing

The Network Server decodes the payload and forwards it to the designated Application Server.

4.User Monitoring

The Application Server processes the data and updates the user interface (APP), allowing real-time monitoring of data.

Downlink:

1.Command Generation

A downlink commands generated in the Network Server or Application Server through a predefined API/interface. (Example: Set sampling interval to 10 minutes; Control Valve ON/OFF.)

2.Gateway Transmission

The command is encapsulated into a downlink packet and sent to the Gateway via the network.

3.End Node Execution

The Gateway transmits the downlink command to the target End Node using the wireless protocol. The End Node parses the command and performs the corresponding action (e.g., activate valve, modify configuration).

2.2 Regional frequency band

At the present moment, our product solely accommodates compatibility with the US915 and EU868.

area	frequency band	center frequency
China	470-510MHz	CN486MHz
America	902-928MHz	US915MHz
Europe	863-870MHz	EU868MHz
Korea	920-923MHz	KR922MHz
Australia	915-928MHz	AU923MHz
New Zealand	921-928MHz	NZ922MHz
Asia	920-923MHz	AS923MHz

3 Usage

In the phase, We use The Things Network(TTN) as data server, and Thingspeak as console to display data& control the valve.

we need to configuration the country/area frequency, inputting DEV EUI/ APP EUI/ APP Key, decodes, and connect to ThingSpeak.

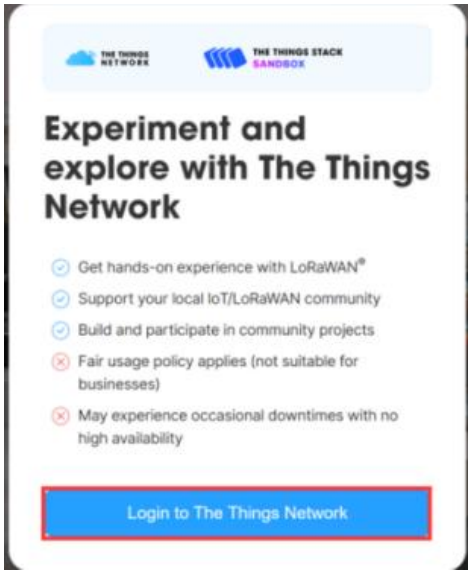
DEV EUI	Unique identification of device, authorized by IEEE
APP EUI	Unique identification of application
APP Key	One of the join network parameters on OTAA mode, calculated by DE EUI

- End Nodes and Gateway:AgroSense_Soil & Air EC-TH Pro(The AgroSense series is applicable)
- Network Server: The Things Network. (Datacake, Loriot, AWS IoT, ChirpStack, ect)
- Application Server: ThingSpeak.(Datacake, Blockbax, akenza, ect)

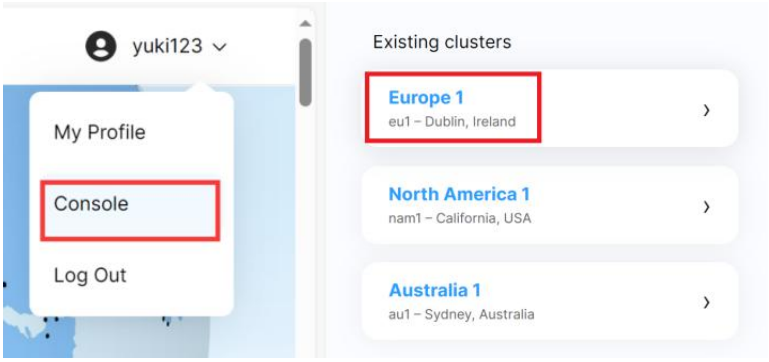
3.1 Usage with TTN &ThingSpeak

3.1.1 Network Server configuration

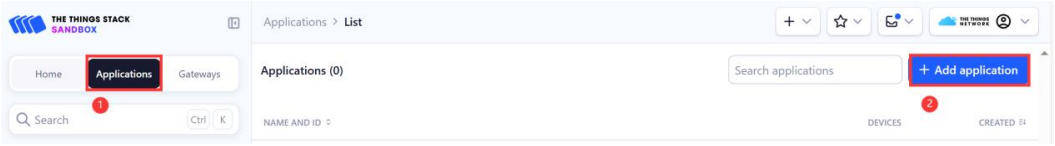
- Open The Things Network in your browser and login it. (Or register an account)



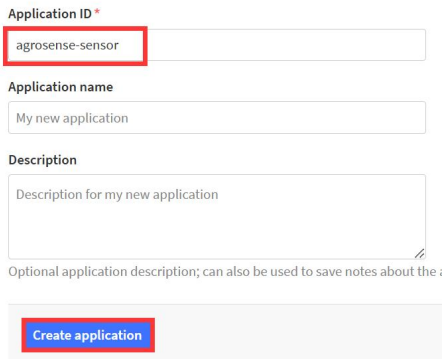
- Click “Console” and select clusters. (we take the European region for example.)



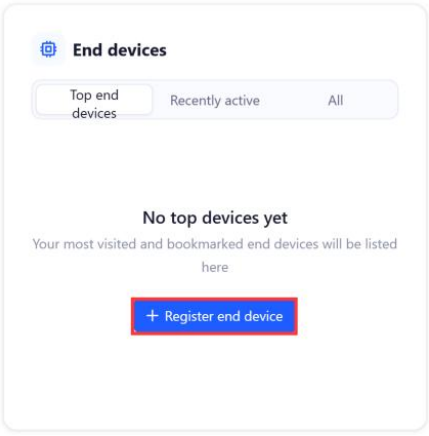
- Click “Go to applications” --> “+ Create application”.



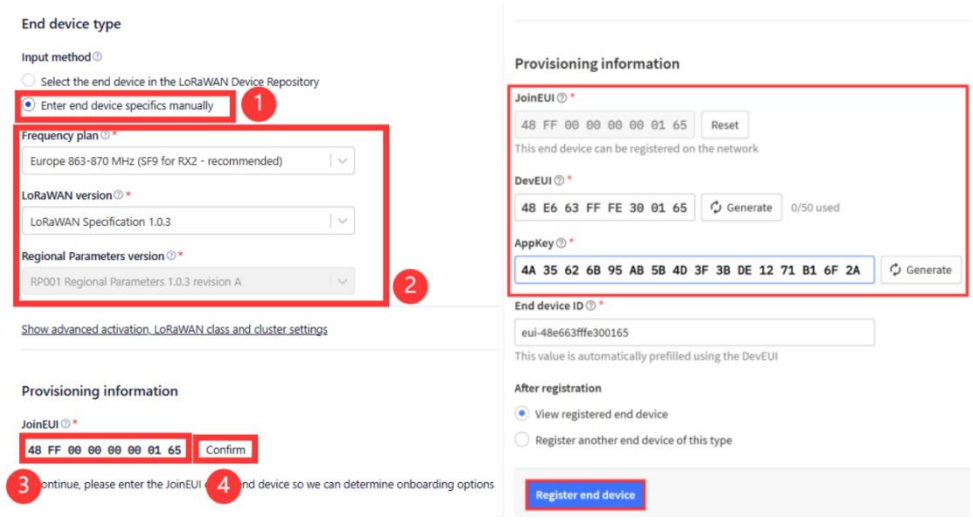
- Write the Application ID and click “Create application”.



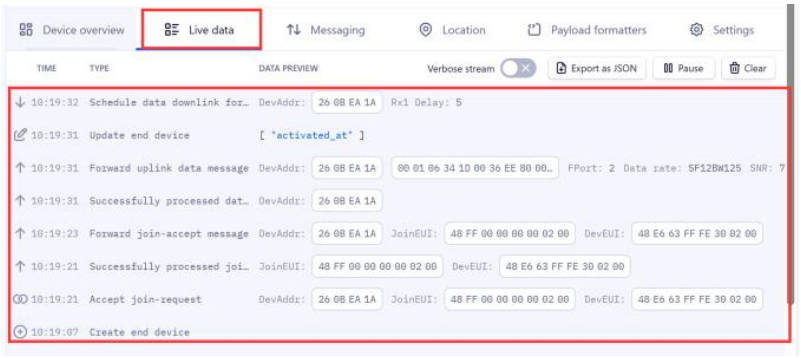
- Click “+ Register and device”.



- Fllowing the steps, and input the DEV EUI/ APP EUI/ APP Key (notice: JoinEUI=APP EUI) and subsequently click on "Register end device" to complete the registration process.

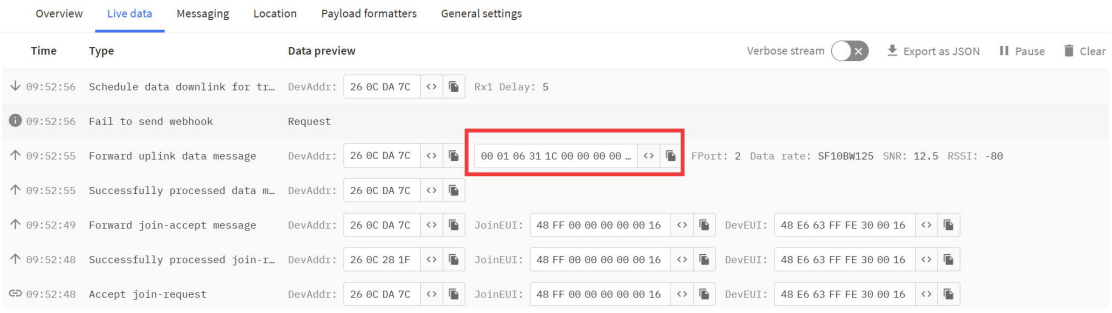


- Plug the battery and press RES button, you can see the device is connected successfully in the TTN.



3.1.2 Decoder

- Now, we need to decoder the data.



Data length	Data description	Value range	Explanation
byte 0	Data packet sequence number high 8 bits	0-0xFFFF	Counting starts from 0 and increments, resetting back to 0 after reaching 65535
byte 1	Data packet		

	sequence number low 8 bits		
byte 2	Battery voltage		The value is amplified by a factor of 10. To get the actual value, divide it by 10. For example, if the value is 0x21 (33), the actual voltage is 3.3 V
byte 3	Soil temp bits 8 to 15		The value is amplified by a factor of 100. To get the actual value, divide it by 100. For example, if the value is 0x08FC = 2300, then the value is 23.
byte 4	Soil temp bits 0 to 7		
byte 5	Soil RH bits 8 to 15		Soil RH ADC value, For example, if the value is 0x0379 = 889, then the value is 889.
byte 6	Soil RH bits 0 to 7		
byte 7	Soil EC bits 24 to 31		The value is amplified by a factor of 100. To get the actual value, divide it by 100. For example, if the value is 0x000008FC = 2300, then the value is 23.
byte 8	Soil EC bits 16 to 23		
byte 9	Soil EC bits 8 to 15		
byte 10	Soil EC bits 0 to 7		
byte 11	Air temp bits 8 to 15		The value is amplified by a factor of 10. To get the actual value, divide it by 10. For example, if the value is 0x0113 (275), the actual value is 27.5
byte 12	Air temp bits 0 to 7		
byte 13	Air Humi bits 8 to 15		The value is amplified by a factor of 10. To get the actual value, divide it by 10. For example, if the value is 0x01D2 (466), the actual value is 46.6
byte 14	Air Humi bits 0 to 7		
byte 15	data transmission interval bits 24 to 31		The time interval for data transmission has been increased by a factor of 1000. The unit is seconds.
byte 16	data transmission interval bits 16 to 23		
byte 17	data transmission interval bits 8 to 15		
byte 18	data transmission interval bits 0 to 7		
Downlink			
Fport 1	Change the data sending interval		5min-1440min

Example:0x00 0x01 0x24 0x08 0xFC 0x03 0x79 0x00 0x00 0x08 0xFC 0x01 0x13 0x01 0xD2 0x00 0x36 0xEE 0x80

Data parsing：

Battery voltage is 3.6V.

Soil temp is 23℃.

Soil RH is 889.

Soil EC is 23 μS/cm.

Air temp is 27.5℃.

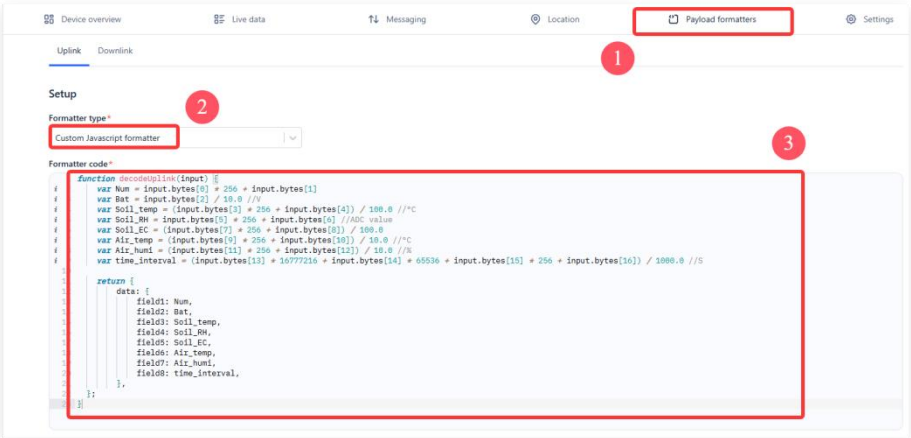
Air Humi is 46.6%.

Data transmission interval value is 3600s.

- Know how to decode it after, we need to write it in code. (You can check it out on Github)

```
function decodeUplink(input) {
    //var Num = input.bytes[0] * 256 + input.bytes[1]
    var Bat = input.bytes[2] / 10.0 //V
    var Soil_temp = (input.bytes[3] * 256 + input.bytes[4]) / 100.0 //°C
    var Soil_RH = input.bytes[5] * 256 + input.bytes[6] //ADC value
    // 1270 corresponds to the ADC value in air, and 815 corresponds to the ADC value in water.
    // Based on this, the ADC can be converted into a percentage.
    // Since water quality varies from place to place, customers need to modify these values themselves.
    var Soil_RH_Percentage=(1270-Soil_RH)*100/(1270-815) //%
    var Soil_EC = (input.bytes[7] * 16777216 + input.bytes[8] * 65536 + input.bytes[9] * 256 + input.bytes[10]) / 100.0 //µS/cm
    var Air_temp = (input.bytes[11] * 256 + input.bytes[12]) / 10.0 //°C
    var Air_humi = (input.bytes[13] * 256 + input.bytes[14]) / 10.0 //%
    var interval = (input.bytes[15] * 16777216 + input.bytes[16] * 65536 + input.bytes[17] * 256 + input.bytes[18]) / 1000.0 //S
    return {
        data: {
            //field1: Num,
            field1: Bat,
            field2: Soil_temp,
            //field3: Soil_RH,
            field3: Soil_RH_Percentage,
            field4: Soil_EC,
            field5: Air_temp,
            field6: Air_humi,
            field7: time_interval,
        },
  };
}
```

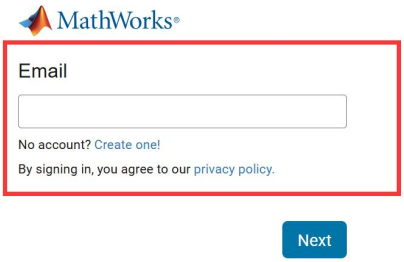
- Select “Payload formatters” and follow the steps.



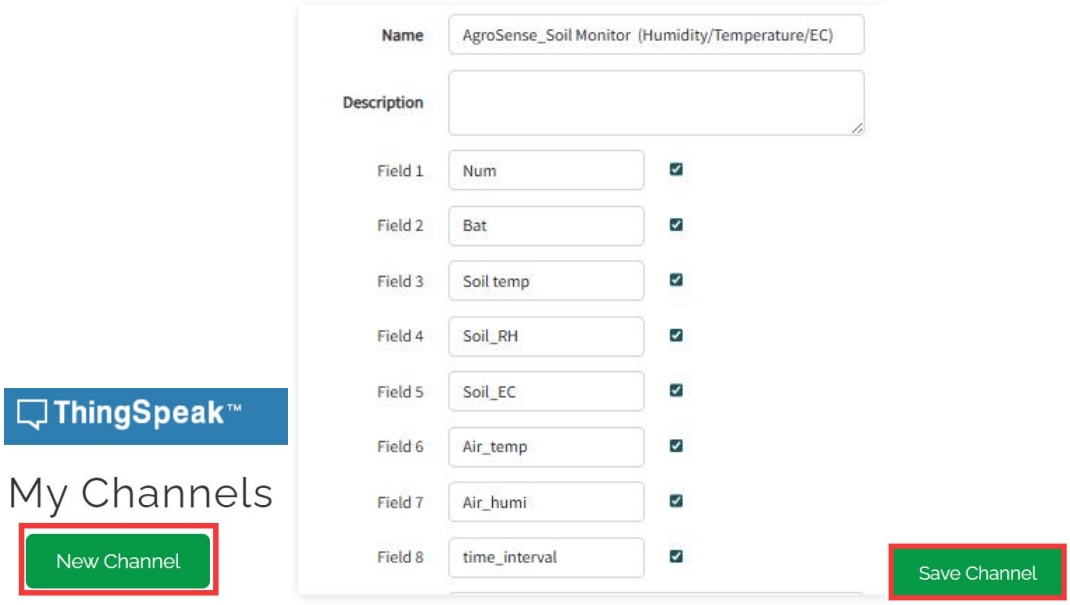
3.1.3 Application Server configuration

In the Application Server configuration, we need to create ThingSpeak channel and get Channel ID and API Key, this is the key to our connection to TTN.

- Login to the ThingSpeak. (Or register an account)



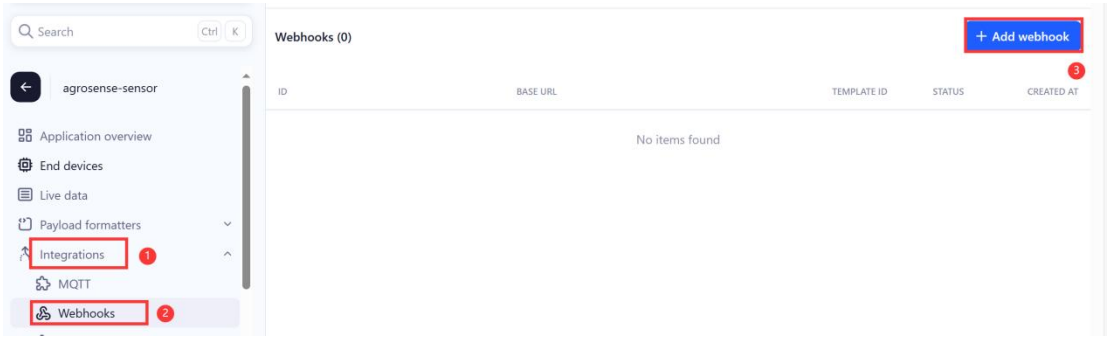
- Click “New Channel”, fill in the Channel name and field names and click “Save Channel”.



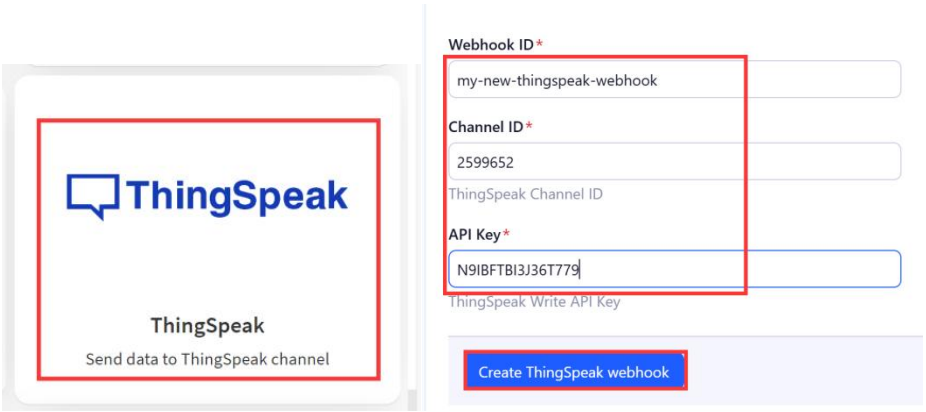
- After successful creation, copy the Channel ID and API Key.

3.1.4 Connect the Network Server and Application Server

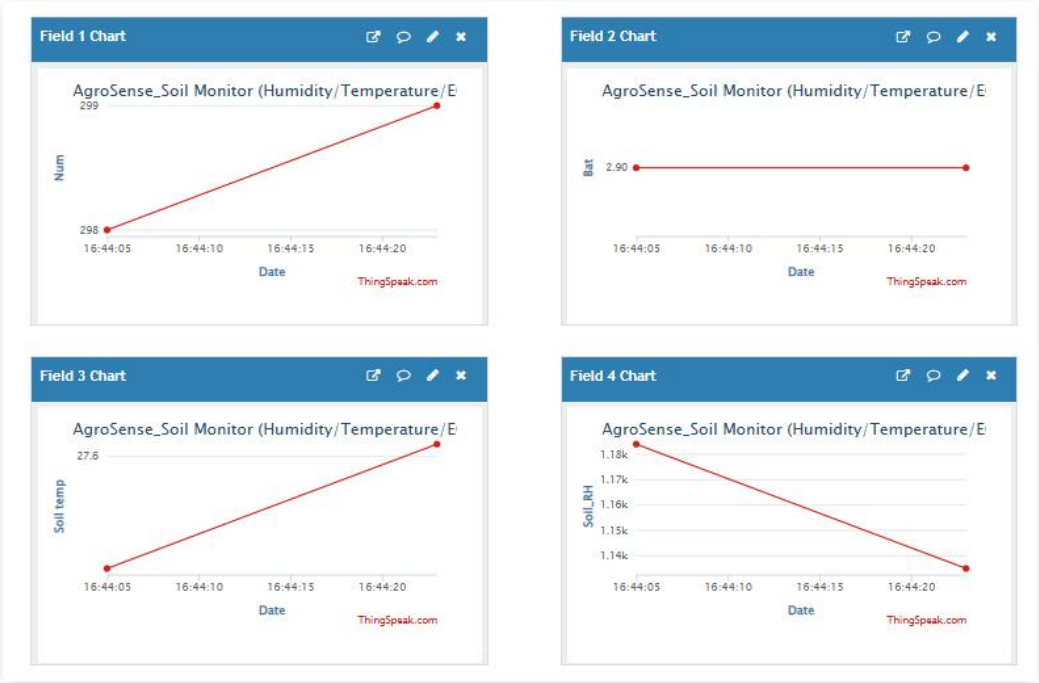
- In the TTN, click “integrations” --> “Webhooks” --> “+ Add webhook”.

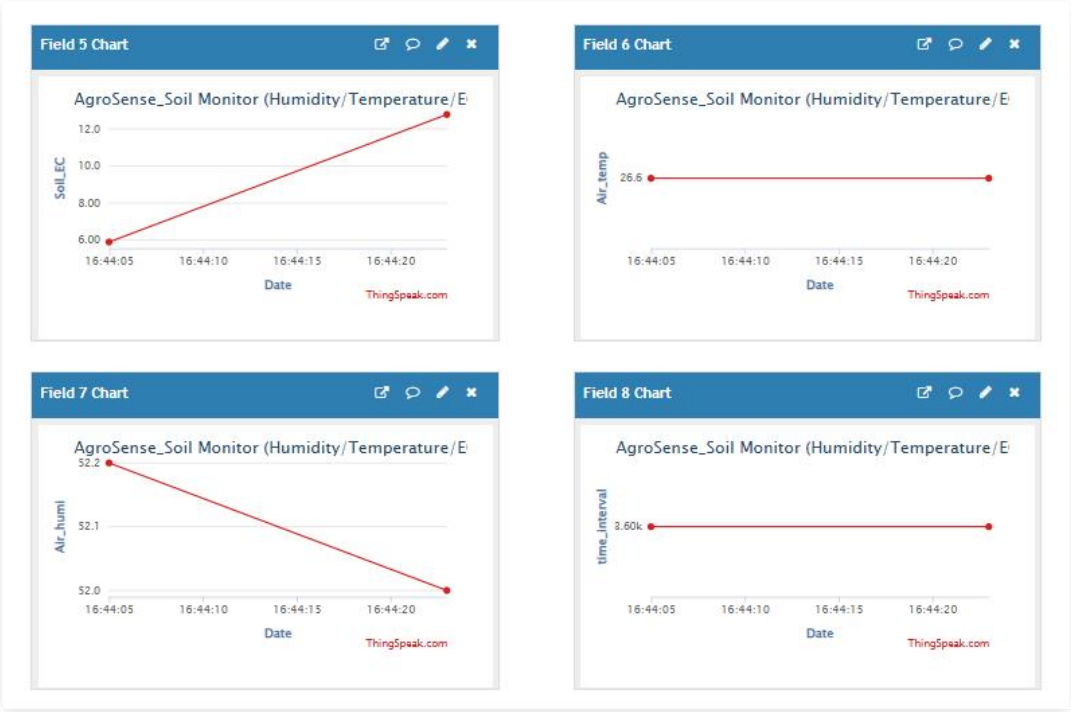


- Select “ThingSpeak”, Fill in the Webhook ID and paste the Channel ID and API Key, click “Create ThingSpeak Webhook”.



- Press RST button, wait about a minute, you will successfully see the data in ThingSpeak.(You will recive the data every hour.)





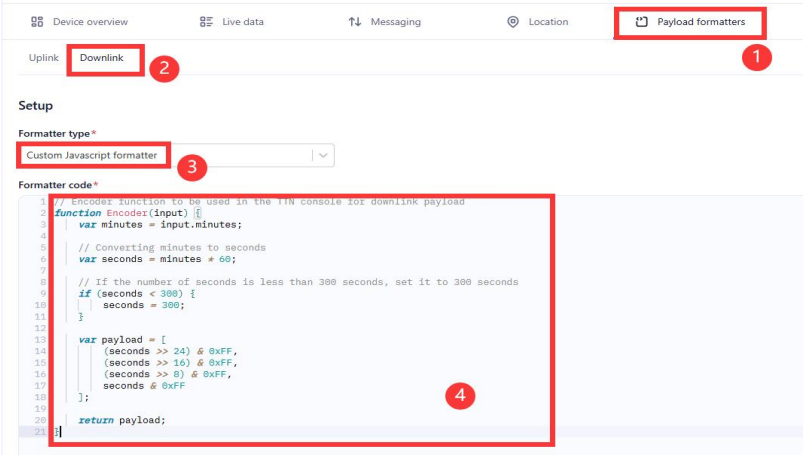
3.1.5 Downlink

- Modification time interva (Fport1)

Modify the time interval for uploading data, the default is one hour.

1 、 If you need to change time Interval (Default 60 minutes), you can click “Payload formatters-->Downlink” and follow the steps.

Formatter code you can find in Github.



2、Click “Save changes”.

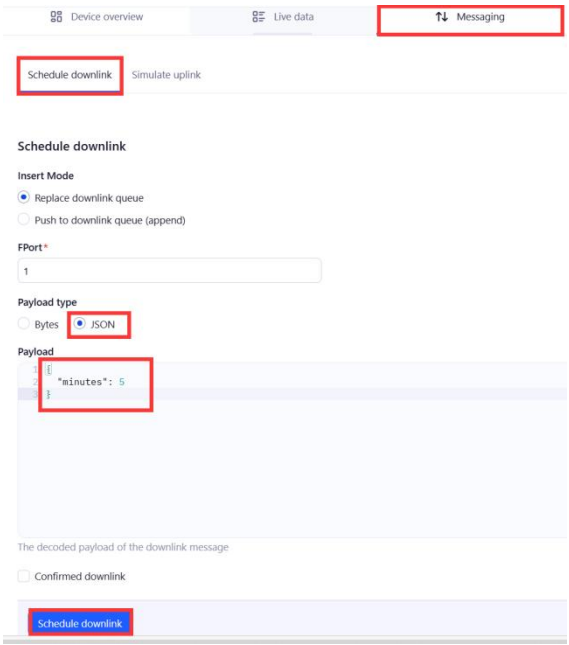
3、Click “Messaging-->Schedule downlink”.

Note: you must use this format:

{

"minutes": 5

}

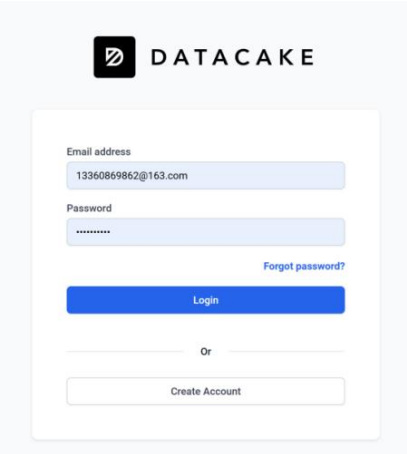


4、The modified interval will be updated after the next data upload.

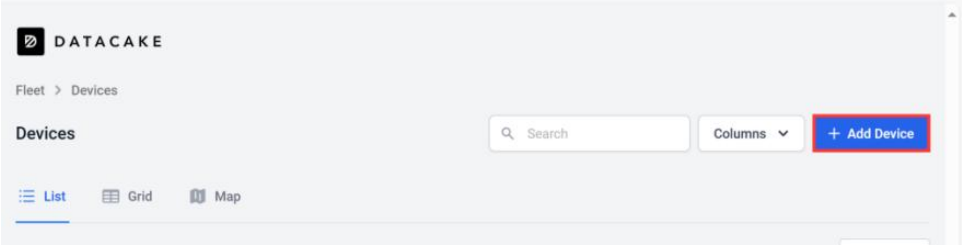
3.2 Usage with Datacake

In this phase, we use DataCake(https://datacake.co/) as the data server & console.

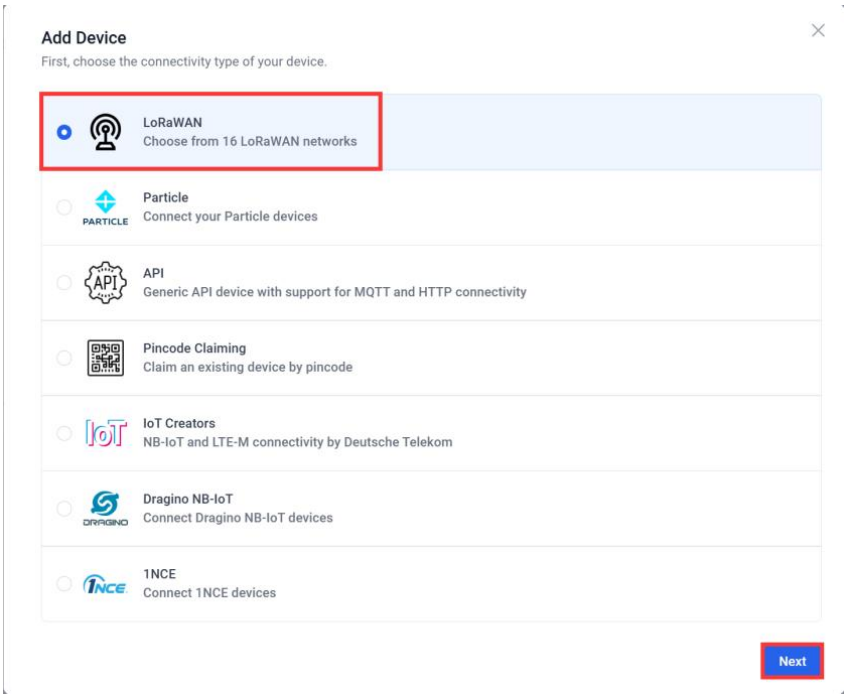
1、Login datacake or Create Account



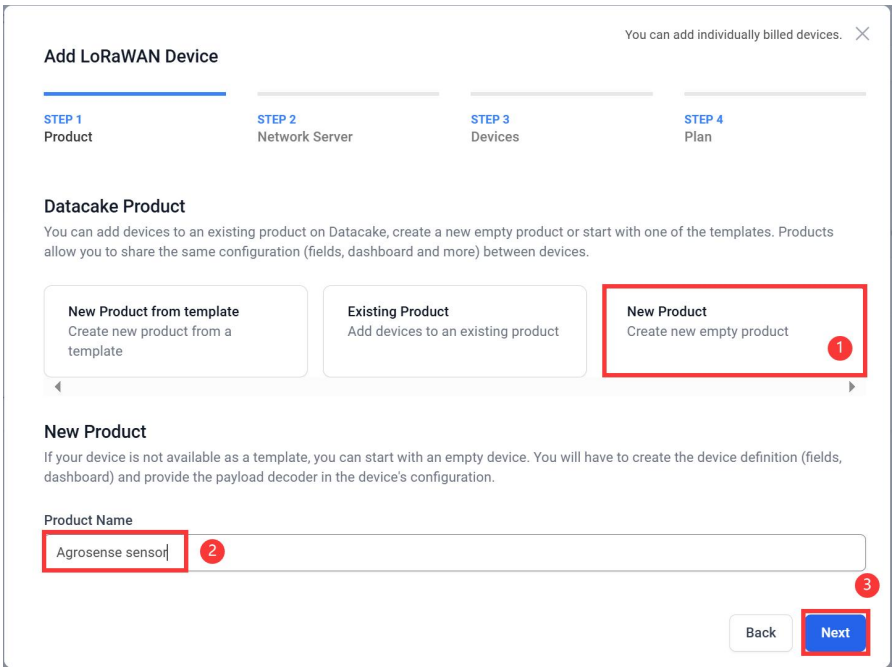
2、Click “Add Device”



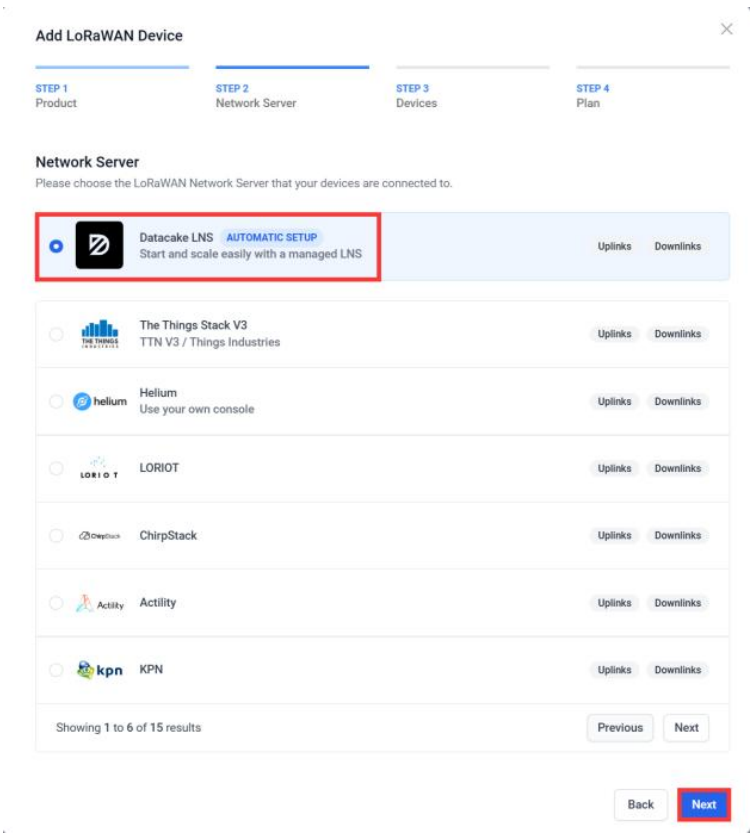
3、Select LoRaWAN and click “Next”



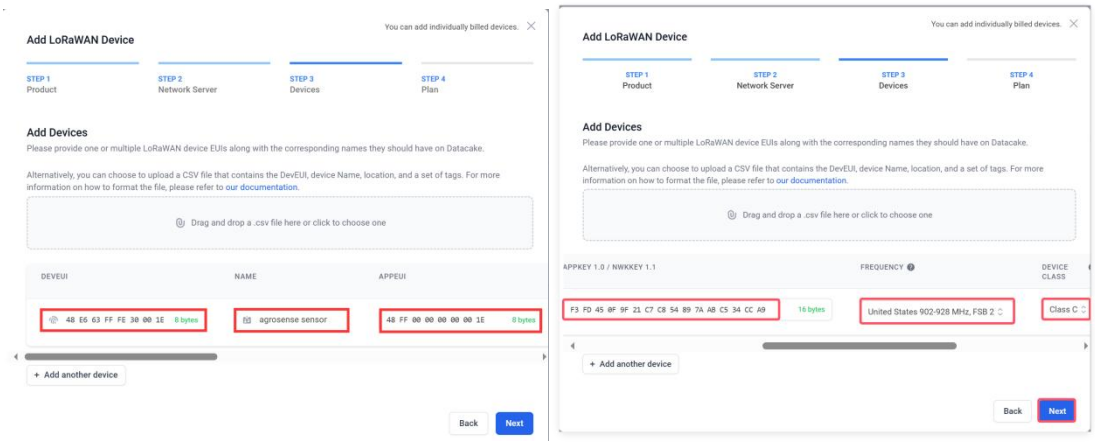
4、Select a Product based on your needs, take "Create new empty product" as an example.



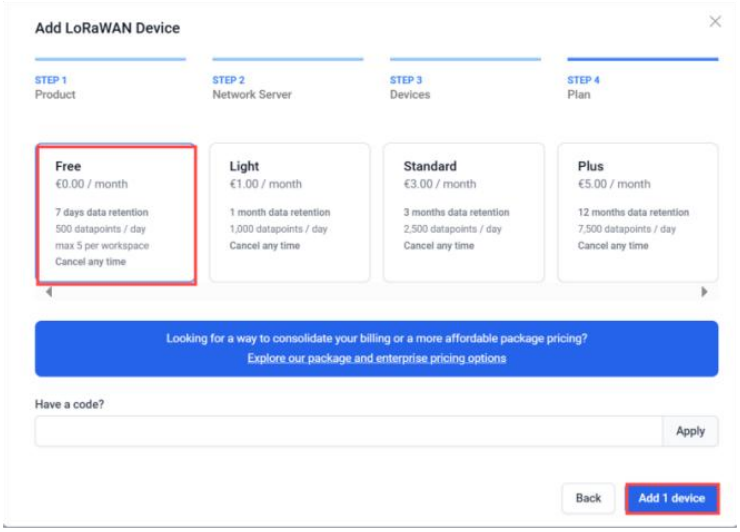
5、Select “Datacake LNS”



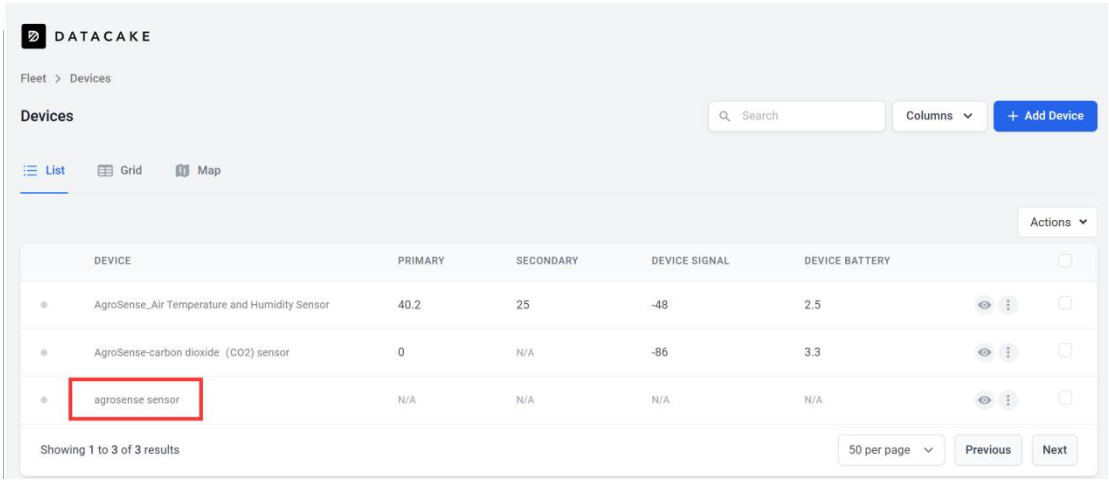
6、Enter DEVEUI、APPEUI、APPKEY、FREQUENCY(take 915 for example) and DEVICE CLASS.



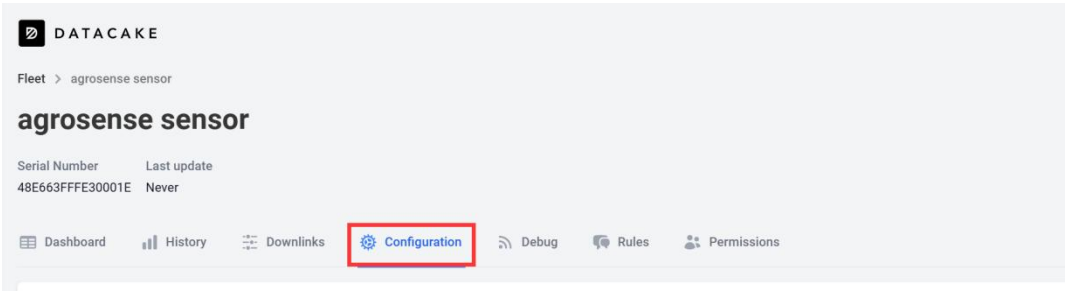
7、Choose the type according to your needs, and click “Add 1 device”.

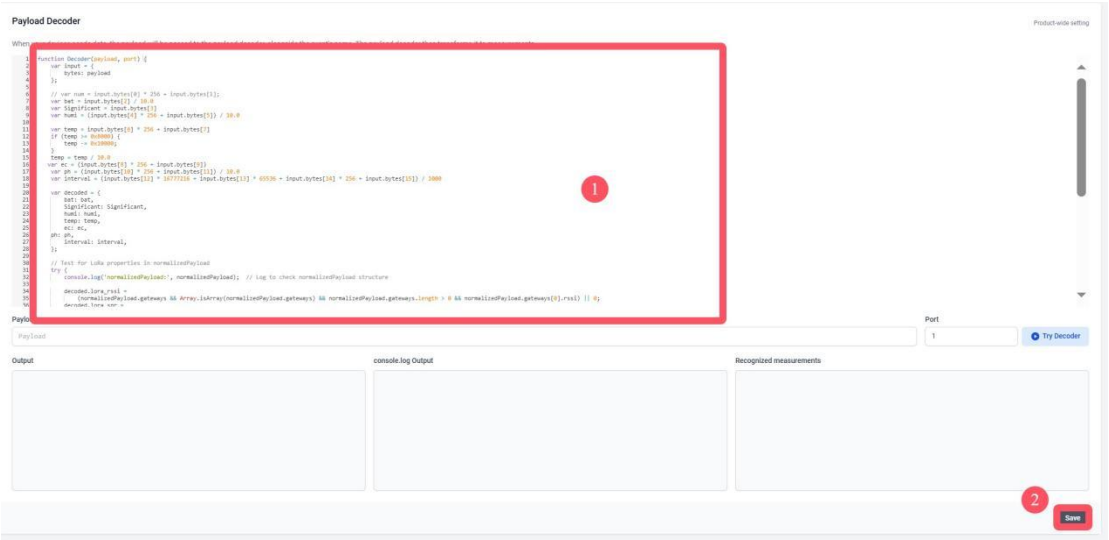


8、Click to go to the device you just added.



9、Click “Configuration”, enter Decoder and click “Save”.(You can check it out on Guihub)



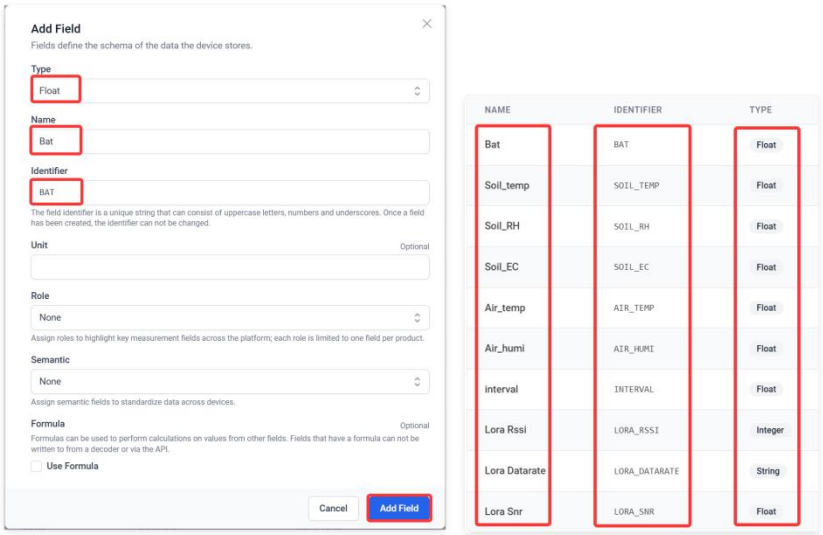



```
function Decoder(payload, port) {
    var input = {
        bytes: payload
    };
    var Num = input.bytes[0] * 256 + input.bytes[1]
    var Bat = input.bytes[2] / 10.0 //V
    var Soil_temp = (input.bytes[3] * 256 + input.bytes[4]) / 100.0 //° C
    var Soil_RH = input.bytes[5] * 256 + input.bytes[6] //ADC value
    // 1270 corresponds to the ADC value in air, and 815 corresponds to the ADC value in water.
    // Based on this, the ADC can be converted into a percentage.
    // Since water quality varies from place to place, customers need to modify these values themselves.
    var Soil_RH_Percentage=(1270-Soil_RH)*100/(1270-815) //%
    var Soil_EC = (input.bytes[7] * 16777216 + input.bytes[8] * 65536 + input.bytes[9] * 256 + input.bytes[10]) / 100.0 //μS/cm
    var Air_temp = (input.bytes[11] * 256 + input.bytes[12]) / 10.0 //° C
    var Air_humi = (input.bytes[13] * 256 + input.bytes[14]) / 10.0 //%
    var interval = (input.bytes[15] * 16777216 + input.bytes[16] * 65536 + input.bytes[17] * 256 + input.bytes[18]) / 1000.0 //S

    var decoded =
    {
        //NUM:Num
```

```
            BAT:Bat
            SOIL_TEMP:Soil_temp
            //SOIL_RH:Soil_RH
            SOIL_RH_PERCENTAGE:Soil_RH_Percentage
            SOIL_EC:Soil_EC
            AIR_TEMP:Air_temp
            AIR_HUMI:Air_humi
            INTERVAL:interval
        };
        // Test for LoRa properties in normalizedPayload
  try {
    if (normalizedPayload.gateways && normalizedPayload.gateways.length > 0) {
        decoded.LORA_RSSI = normalizedPayload.gateways[0].rssi || 0;
        decoded.LORA_SNR = normalizedPayload.gateways[0].snr || 0;
    } else {
        decoded.LORA_RSSI = 0;
        decoded.LORA_SNR = 0;
    }
    decoded.LORA_DATARATE = normalizedPayload.spreading_factor
                            || normalizedPayload.data_rate
                            || (normalizedPayload.networks && normalizedPayload.networks.lora &&
normalizedPayload.networks.lora.dr)
                            || "unknown";
} catch (error) {
    console.log('LoRa property parsing error:', error);
    decoded.LORA_RSSI = 0;
    decoded.LORA_SNR = 0;
    decoded.LORA_DATARATE = "unknown";
}
return decoded;
}
```

10、Follow the steps to add a field. (Every fields is the same way)



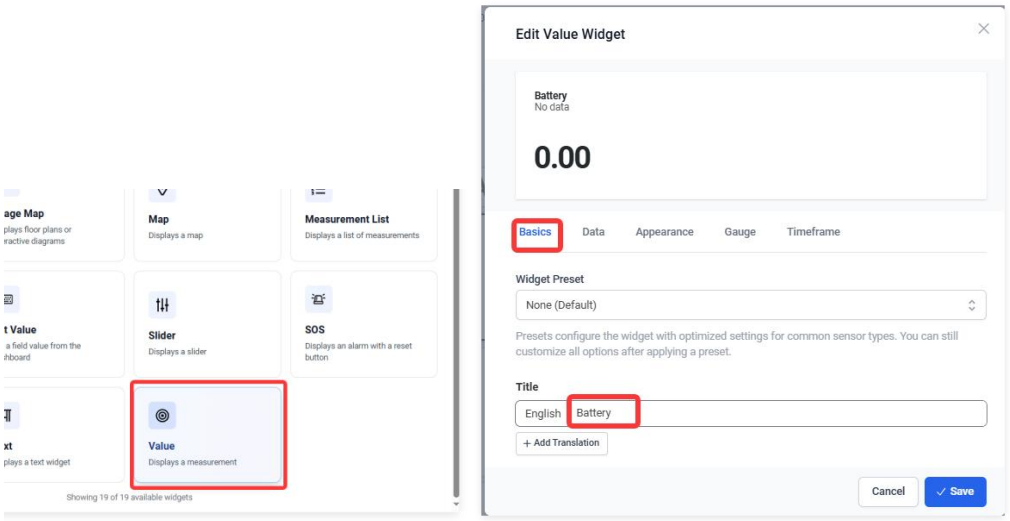
11、Press RST button, wait until the sensor connects to the gateway successfully, you will see the data the sensor is currently reading.

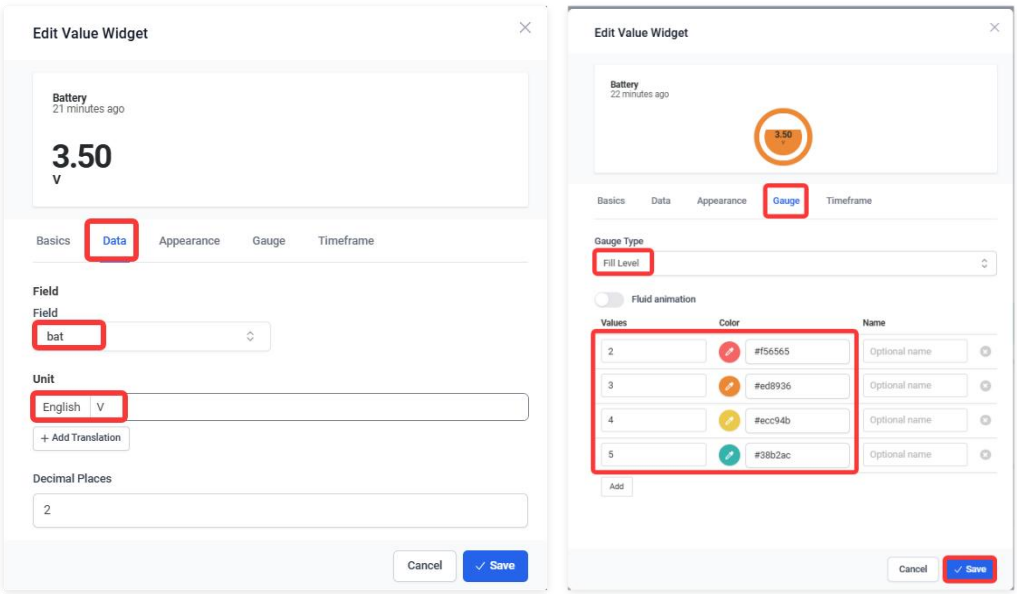
NAME	IDENTIFIER	TYPE	ROLE / SEMANTIC	CURRENT VALUE	LAST UPDATE
Bat	BAT	Float	Battery	2.9	0 seconds ago
Soil_temp	SOIL_TEMP	Float		27.83	0 seconds ago
Soil_RH	SOIL_RH	Float		1,051	0 seconds ago
Soil_EC	SOIL_EC	Float		185.55	0 seconds ago
Air_temp	AIR_TEMP	Float		26	0 seconds ago
Air_humi	AIR_HUMI	Float		63	0 seconds ago
interval	INTERVAL	Float		3,600	0 seconds ago
Lora Rssi	LORA_RSSI	Integer		-87	0 seconds ago
Lora Datarate	LORA_DATARATE	String		SF7BW125.0	0 seconds ago
Lora Snr	LORA_SNR	Float		11.2	0 seconds ago

12、To get a better look at the data, we can add widget.

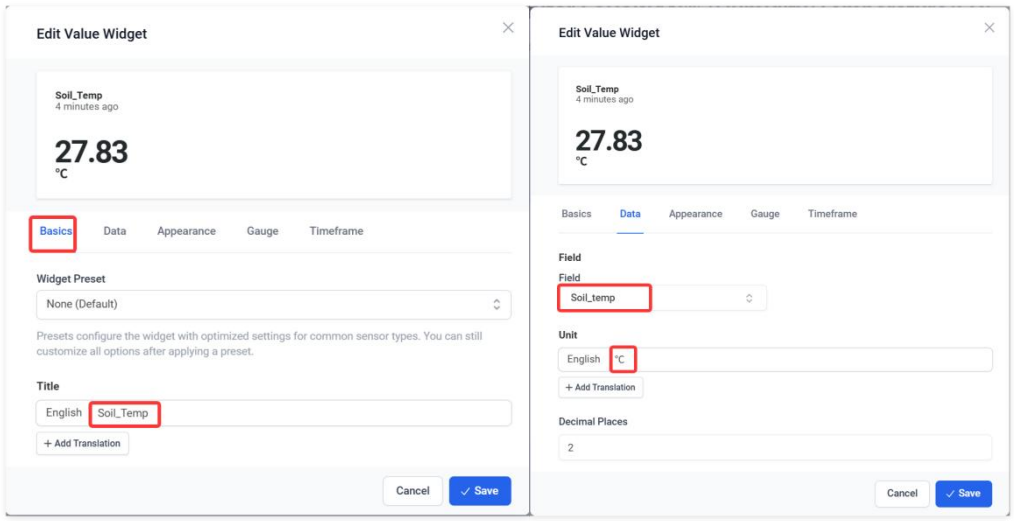
Click “Dashboard-->switch-->+ Add Widget”.

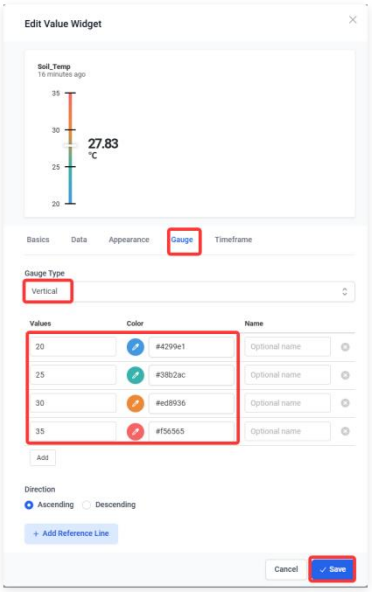
13、Add “Battery” field, Select “Value” and follow the steps below to complete the setup.



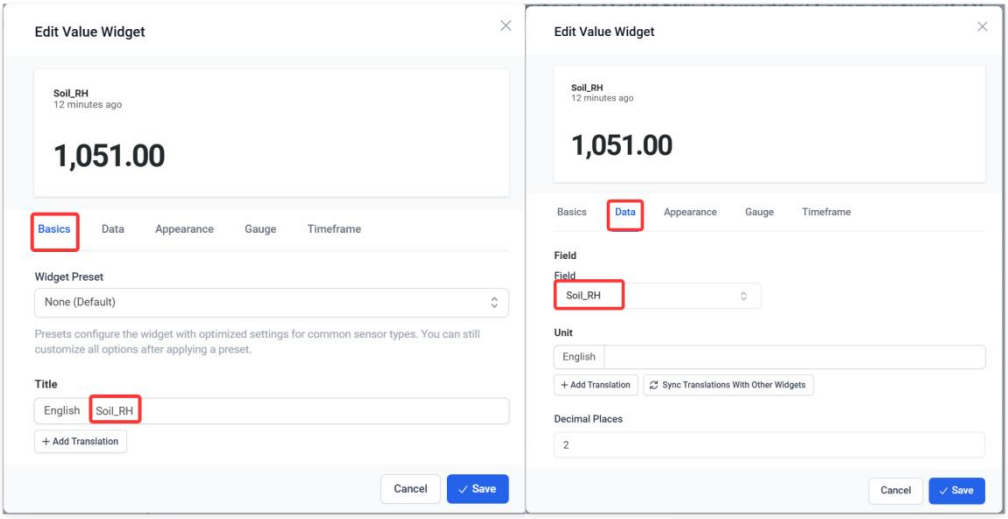


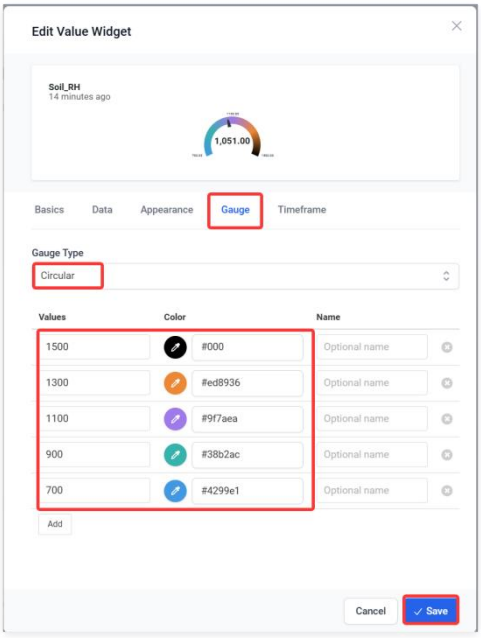
14、Add “Soil_Temp” field, Select “Value” and follow the steps below to complete the setup.



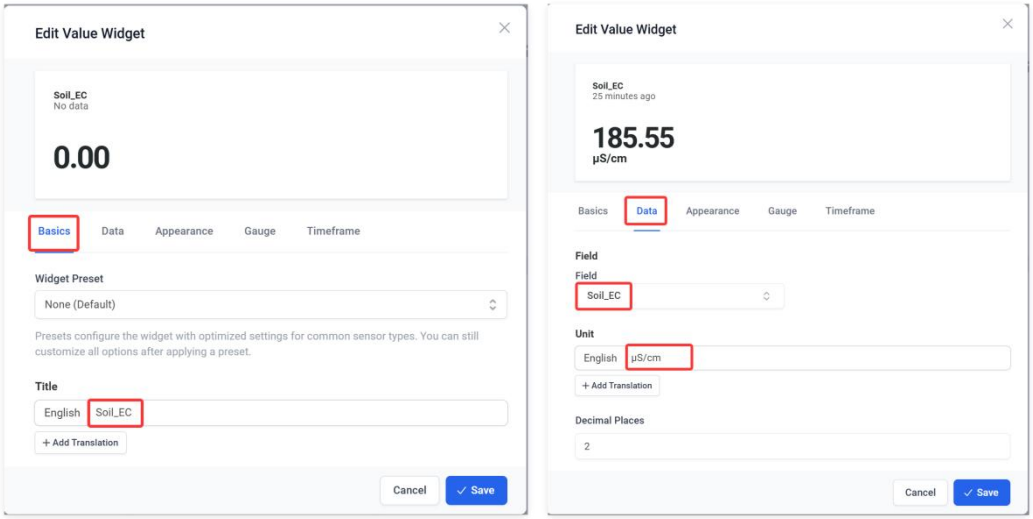


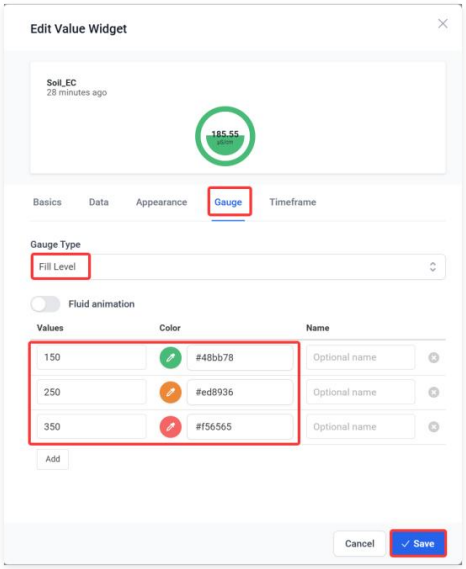
15、Add “Soil_RH” field, Select “Value” and follow the steps below to complete the setup.



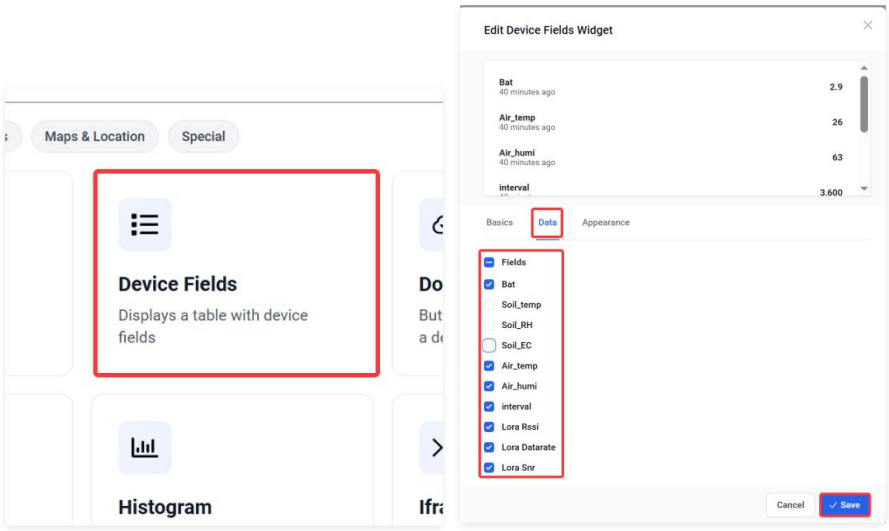


16、Add “Soil_EC” field, Select “Value” and follow the steps below to complete the setup.

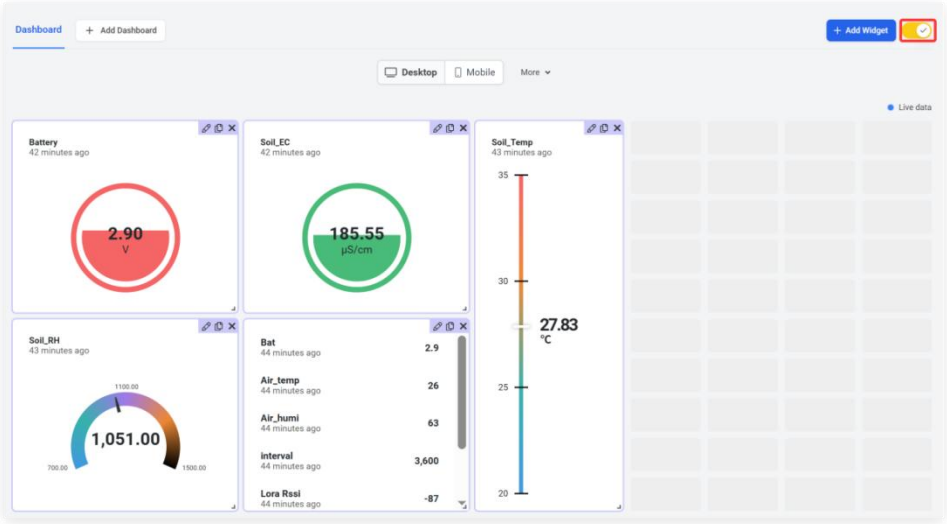




17、Select Device Fields, check “Fields” and click “Save”.



18、Click the switch to save, and you can see the data visually.

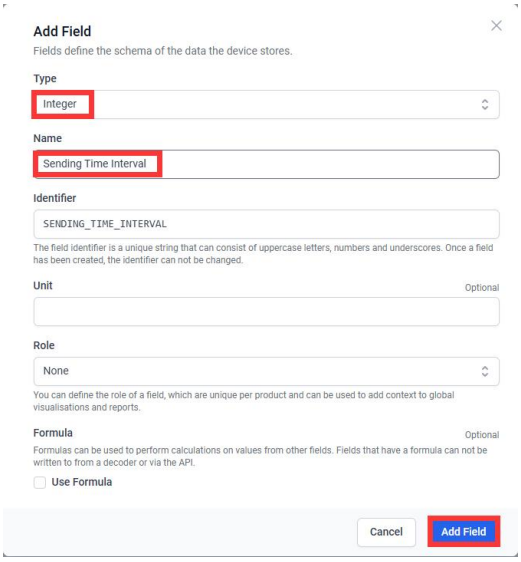


3.2.1 Downlink

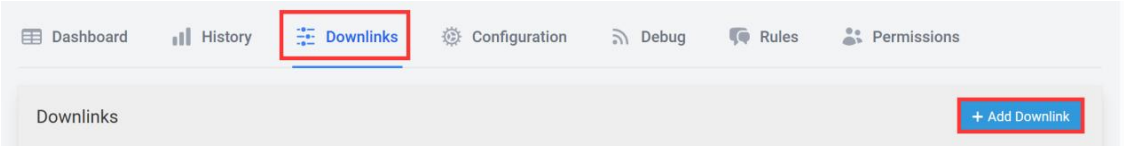
- Modification time interva (Fport1)

Modify the time interval for uploading data, the default is one hour.

1 、 If you need to change time Interval (Default 60 minutes), you can click “Configuration-->Fields-->+Add Field”



2、Click ”Downlink-->Add Downlink”.

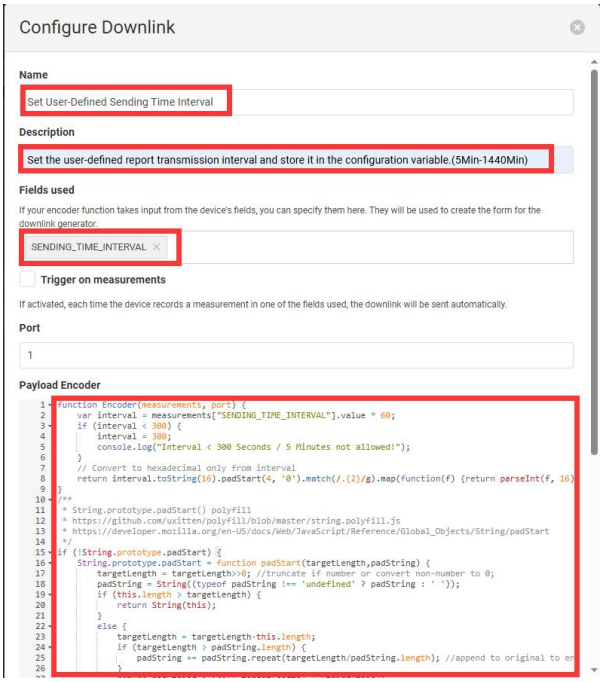


Enter name、description、fields used and payload encoder respectively.

Name: Set User-Defined Sending Time Interval

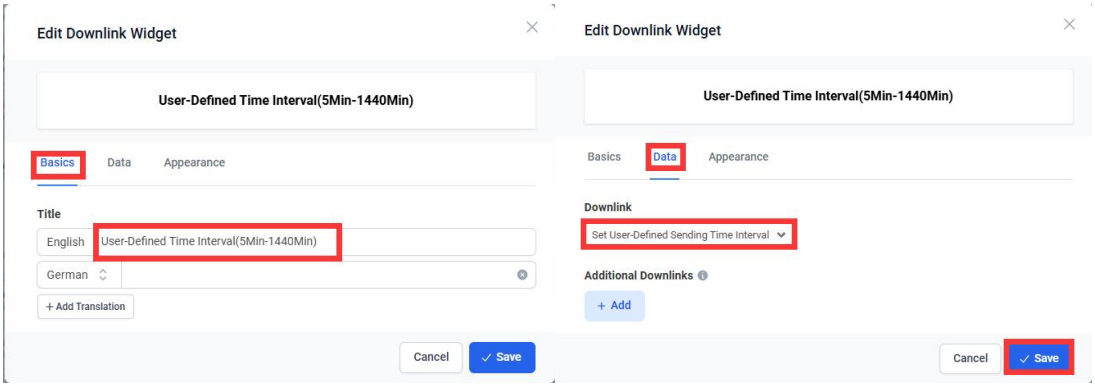
Description: Set the user-defined report transmission interval and store it in the configuration variable.(5Min-1440Min)

Payload Encoder: copy in Github.



3、Click “Dashboard-->switch-->+ Add Widget”.

Select “Downlink” and setting as follow image.



4、Click the switch to save, and you can click to change your time Interval.