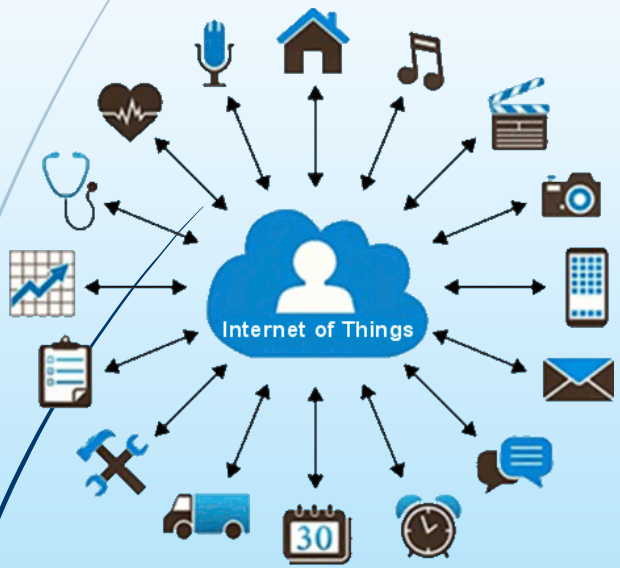
Sensor Networks - Project 1



- LoRaWAN connectivity implementation for the embedded plant monitoring IoT system platform using the B-L072Z-LRWAN1 ARM mbed-based platform

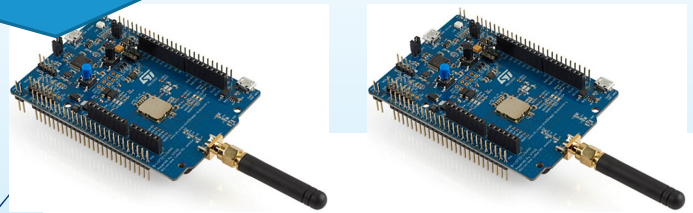
- Pedro J. Lobo
  Office A4202
  e-mail: pedro.lobo@upm.es
- Guillermo Azuara
  Office A4206
  e-mail: g.azuara@upm.es

Outline

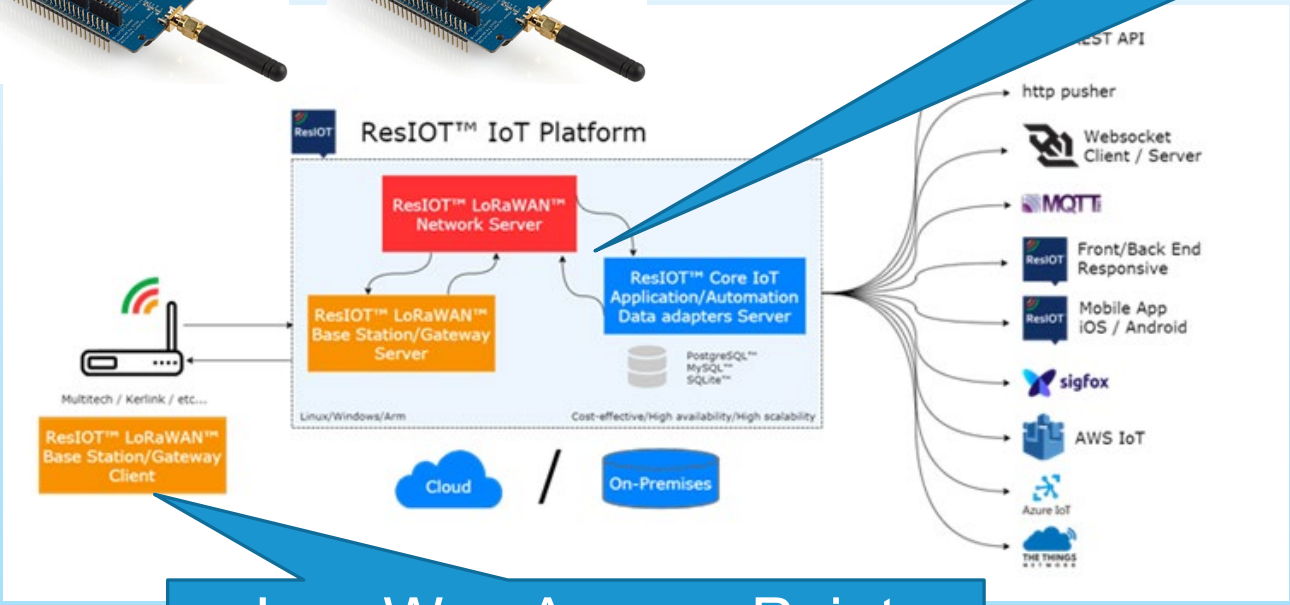
- Project main goal
  - Add LoRaWan connectivity to the project developed in the Embedded Systems course.
- Hardware, software and other resources to be used
  - HW: B-L072Z-LRWAN1, Multitech Conduit Access Point (Gateway)
  - SW: Mbed (node), Packet-Forwarder (Gateway) and ResIoT Server (software cloud applications)
- Project Specifications:
  - Work to be done by students
    - SW development using Mbed and Keil Microvision using a template provided by instructors.
    - Configuration of ResIoT software application, LUA script development
  - Report and demonstration
    - Implemented code (C++ and LUA)
    - Results

Main steps

Software development using LoRaWan protocol for B-L072Z-LRWAN1 device



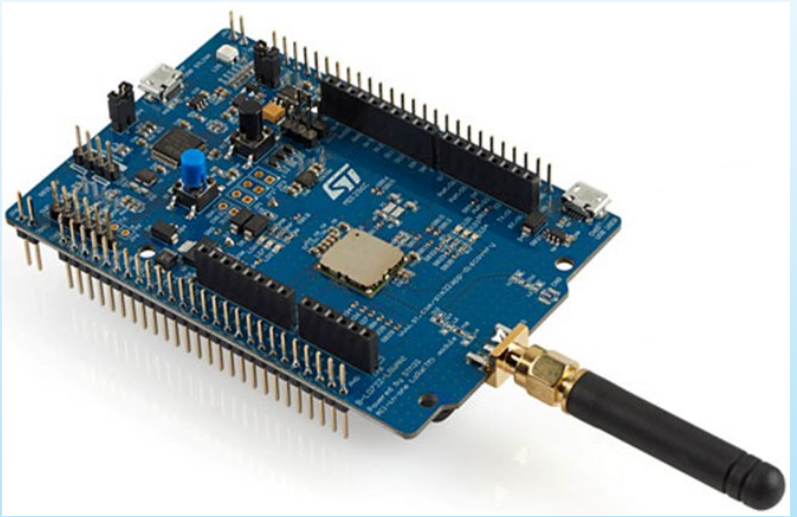
LoRaWan cloud Application LoRaWan server configuration



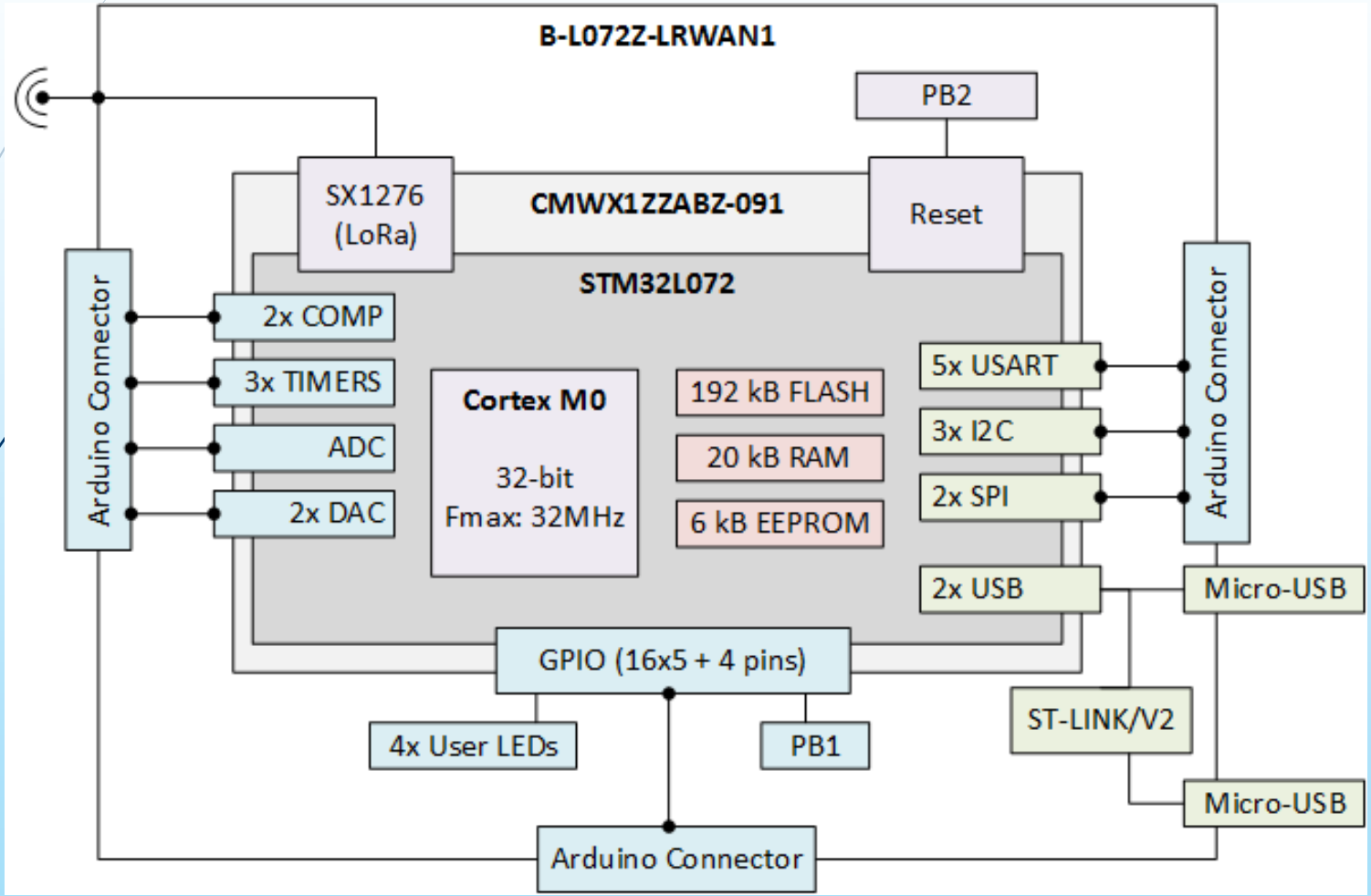
LoraWan Access Point (Gateway) Configuration

Recall: The B-L072Z-LRWAN1

- Our platform: B-L072Z-LRWAN1
  - Core: STM32L072CZ
    - ARM Cortex M0+
    - Ultra-low-power
  - Peripherals
    - ADC, DAC, timers
  - Serial connections
    - I2C, UART, SPI
  - Embedded LoRa module
  - User LEDs and buttons
  - Arduino compatible connectors
  - *Much more…*

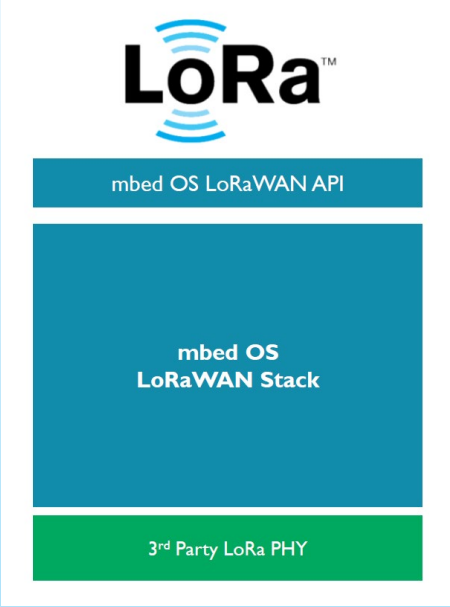


Recall: General diagram of the B-L072Z-LRWAN



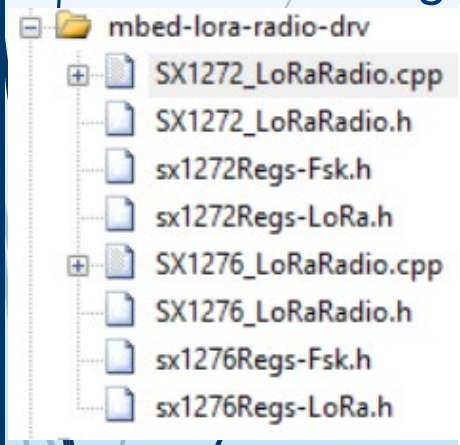
LoRa integration in Mbed OS

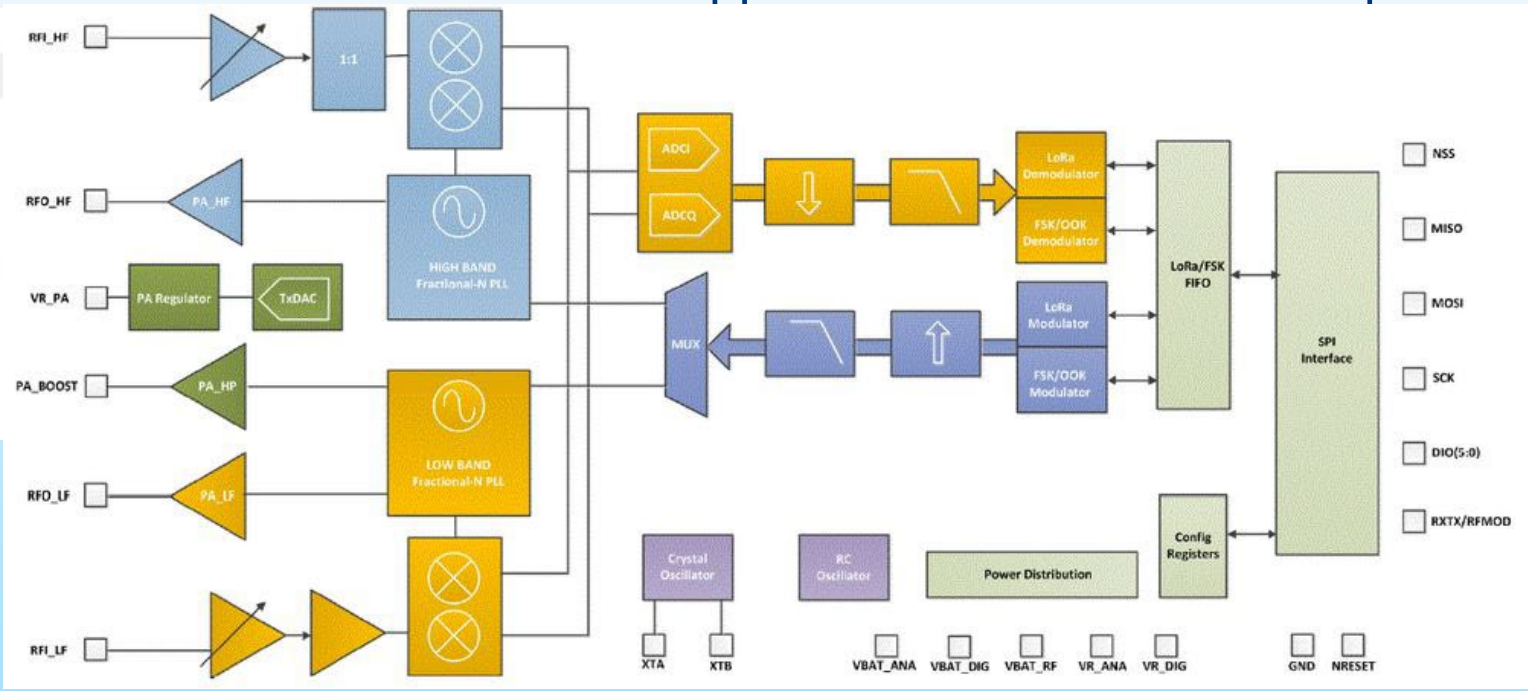
- Mbed OS already supports:
  - LoRa radio devices (PHY)
  - mbed LoRa radio drivers reside out of the mbed OS tree. Arm provides support for SX1272 and SX1276 LoRa radios, which are the most widely used LoRa end-device radio chipsets
  - LoRaWAN protocol (stack): LoRaWAN API
  - LoRaWAN Specification v1.0.2 and v1.1 (Some of the features in v1.1 are the improved security primitives and the support for roaming)
  - Two classes of devices: **A** and C
  - The LoRaWAN specification defines two methods for connecting to an access network
    - **Over the air activation (OTAA)**
    - Activation by personalization (ABP)



LoRa PHY Layer in Mbed

- Developed by SEMTECH and STACKFORCE for SX1272/6 chips
- Supported in mbed or in mbed-os
- Configuration of SPI interface and support for the SX127x interrupts





SX1276 radio features

- 168 dB maximum link budget (maximum attenuation between Device and Gateway)
- High sensitivity (RX) : down to −148 dBm
- +20 dBm – 100 mW constant RF output vs supply voltage (TX)
- +14 dBm high efficiency Power Amplifier
- Programmable bit rate up to 300 kbps
- Low RX current of 9.9 mA, 200 nA register retention
- Fully integrated synthesizer with a resolution of 61 Hz
- FSK (Frequency), GFSK (Gaussian Frequency Shift Keying) , MSK (Minimum), GMSK (Gaussian Minimum), LoRa and OOK (ASK ON/OFF) modulation
- 127 dB Dynamic Range RSSI
- Built-in temperature sensor and low battery indicator

Software configuration on LoRaWAN in Mbed (parameters, default values) I

- Name: lora.adr-on
  - Description: Turns Automatic Data Rate on/off
  - Defined by: library:lora
  - Value: 1 (set by library:lora)
- Name: lora.app-port
  - Description: Set the application port
  - Defined by: library:lora
  - Value: 15 (set by library:lora)
- **Name: lora.application-eui**
  - Description: Set AppEUI (application EUI needed for OTAA)
  - Defined by: library:lora
  - Value: {0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00} (set by library:lora)
- **Name: lora.application-key**
  - Description: Set AppKey (application key needed for OTAA)
  - Defined by: library:lora
  - Value: {0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00} (set by library:lora)

Software configuration on LoRaWAN in Mbed (parameters, default values) II

- Name: lora.appskey
  - Description: Set AppSkey (application session key needed for ABP)
  - Defined by: library:lora
  - Value: {0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00} (set by library:lora)
- Name: lora.device-address
  - Description: Set DevAddr (device address needed for ABP)
  - Defined by: library:lora
  - Value: 0x00000000 (set by library:lora)
- **Name: lora.device-eui**
  - Description: Set DevEUI (device EUI needed for OTAA)
  - Defined by: library:lora
  - Value: {0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00} (set by library:lora)
- Name: lora.duty-cycle-on
  - Description: Turns duty cycle on/off
  - Defined by: library:lora
  - Value: 1 (set by library:lora)

Software configuration on LoRaWAN in Mbed (parameters, default values) III

- Name: lora.lbt-on
  - Description: Turns LBT on/off
  - Defined by: library:lora
- Name: lora.nb-trials
  - Description: Set number of retries for a join request
  - Defined by: library:lora
  - Value: 12 (set by library:lora)
- Name: lora.nwkskey
  - Description: Set NwkSkey (network session key needed for ABP)
  - Defined by: library:lora
  - Value: {0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00} (set by library:lora)
- **Name: lora.over-the-air-activation**
  - Description: Enable or disable OTAA. **Value set to false would enable ABP**
  - Defined by: library:lora
  - Value: 1 (set by library:lora)

Software configuration on LoRaWAN in Mbed (parameters, default values) IV

- Name: lora.phy
  - Description: Set the region of operation for the device
  - Defined by: library:lora
  - Value: **EU868** (set by library:lora)
- Name: lora.public-network
  - Description: Set the public network parameter
  - Defined by: library:lora
  - Value: 1 (set by library:lora)
- Name: lora.tx-max-size
  - Description: Maximum outgoing buffer size
  - Defined by: library:lora
  - Value: 64 (set by library:lora)

Software configuration on LoRaWAN in Mbed (parameters, default values) V

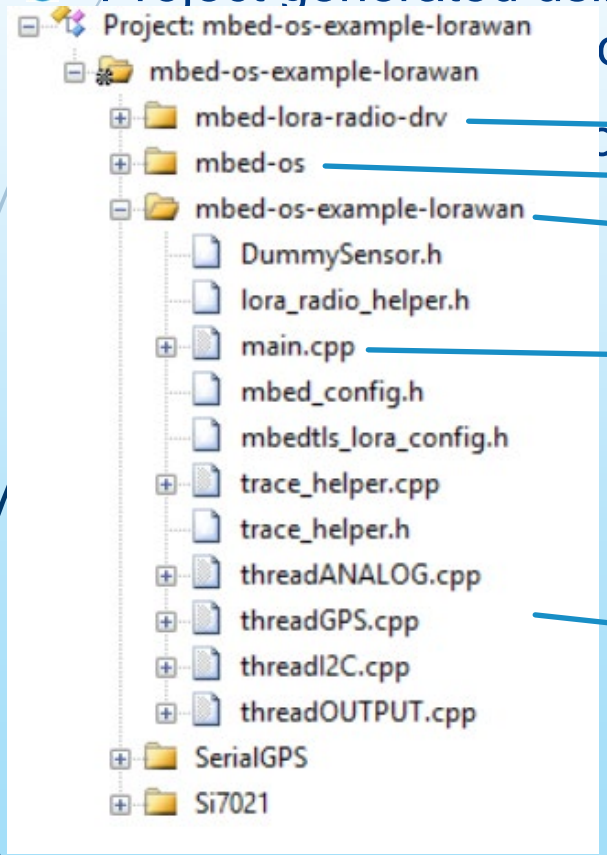
- If you want to change the default values (you have to, really):
  - You can edit the file mbed_app.json
    - mbed_config.h is generated from it
    - BUT if you modify it, the whole project (incl. Mbed) is recompiled
      - 10+ minutes
  - So, the parameters that you have to change are overridden in main.cpp

```
static uint8_t DEV_EUI[] = {0xNN, 0x39, 0x32, 0x35, 0x59, 0x37, 0x91, 0x94};
static uint8_t APP_EUI[] = {0x70, 0xb3, 0xd5, 0x7e, 0xd0, 0x00, 0xfc, 0x4d};
static uint8_t APP_KEY[] = {0xf3, 0x1c, 0x2e, 0x8b, 0xc6, 0x71, 0x28, 0x1d,
                            0x51, 0x16, 0xf0, 0x8f, 0xf0, 0xb7, 0x92, 0x8f};
```

- Get the DEV_EUI values from the user account information that is published in Moodle
- Do not change any other parameters!!!
- If the LoRaWAN layer is correctly configured, you can start using the LoRaWAN API

Mbed Studio project organization

- Download the example project from Moodle
  - Beware! 700+ MB ZIP file
- Project generated using mbed repositories, with mbed compiler and ovision



on of the project provided by the instructors

Lora PHY

Mbed OS

Application using LoRaWAN

Your main application

Other software modules

Example (I) Having a look at the main.cpp file

1. static EventQueue ev_queue (MAX_NUMBER_OF_EVENTS * EVENTS_EVENT_SIZE);
2. SX1276_LoRaRadio radio (MBED_CONF_APP_LORA_SPI_MOSI,
   MBED_CONF_APP_LORA_SPI_MISO,
   MBED_CONF_APP_LORA_SPI_SCLK,
   MBED_CONF_APP_LORA_CS,
   MBED_CONF_APP_LORA_RESET,.....);
3. static LoRaWANInterface lorawan (radio);
4. static lorawan_app_callbacks_t callbacks;
5. lorawan.initialize(&ev_queue); // initialization of LoRaWAN
6. callbacks.events = mbed::callback(lora_event_handler);
7. lorawan.add_app_callbacks(&callbacks); // setting a callback function

Example (II) Having a look at the main.cpp file

- Having a look at the main.cpp file!!!!
- retcode = lorawan.connect(); // see possible errors
- ev_queue.dispatch_forever(); Every time that an event occurs the callback is executed
- Return //main.cpp

Example (III) Having a look at the main.cpp file

- Callback code in main.cpp

```
static void lora_event_handler(lorawan_event_t event)
{
    switch (event) {
        case CONNECTED:
            printf("\r\n Connection - Successful \r\n");
            if (MBED_CONF_LORA_DUTY_CYCLE_ON) {
                send_message();
            } else {
                ev_queue.call_every(TX_TIMER, send_message);
            }

            break;
        case DISCONNECTED:
            ev_queue.break_dispatch();
            printf("\r\n Disconnected Successfully \r\n");
            break;
        case TX_DONE:
            printf("\r\n Message Sent to Network Server \r\n");
            if (MBED_CONF_LORA_DUTY_CYCLE_ON) {
                send_message();
            }
            break;
```

Example (III)

- Having a look at the callback code in main.cpp

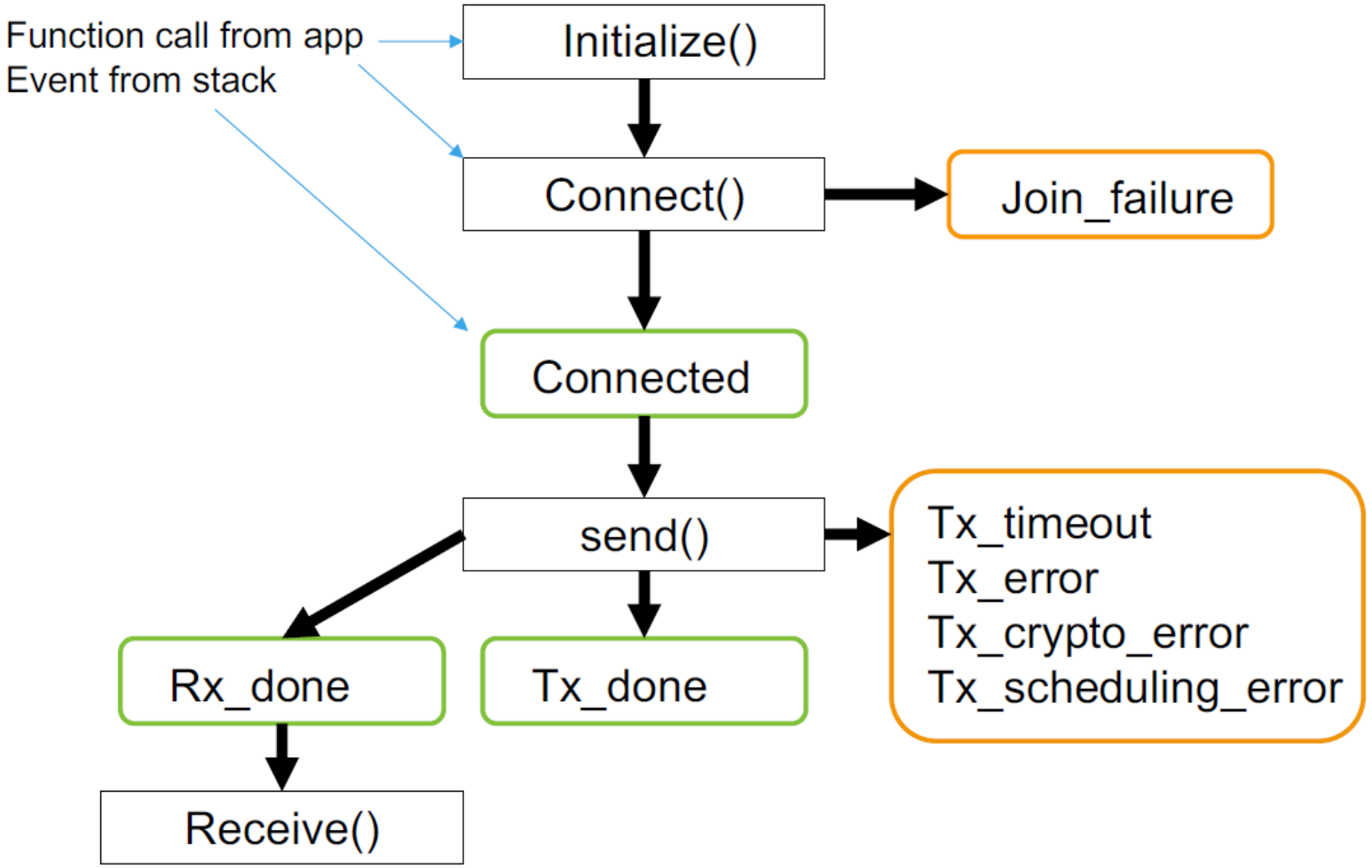
```
case TX_TIMEOUT:
    case TX_ERROR:
    case TX_CRYPTO_ERROR:
    case TX_SCHEDULING_ERROR:
        printf("\r\n Transmission Error - EventCode = %d \r\n", event);
        // try again
        if (MBED_CONF_LORA_DUTY_CYCLE_ON) {
            send_message();
        }
        break;
    case RX_DONE:
        printf("\r\n Received message from Network Server \r\n");
        receive_message();
        break;
    case RX_TIMEOUT:
    case RX_ERROR:
        printf("\r\n Error in reception - Code = %d \r\n", event);
        break;
    case JOIN_FAILURE:
        printf("\r\n OTAA Failed - Check Keys \r\n");
        break;
    case UPLINK_REQUIRED:
        printf("\r\n Uplink required by NS \r\n");
        if (MBED_CONF_LORA_DUTY_CYCLE_ON) {
            send_message();
        }
        break;
    default:
        MBED_ASSERT("Unknown Event");
}
```

Example (IV)

- Having a look at the callback code in main.cpp

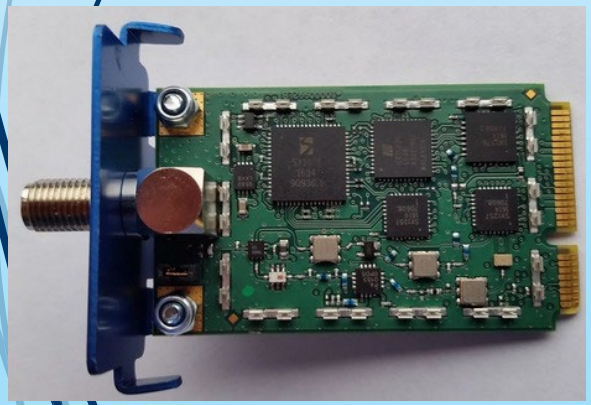
```
static void send_message()
{
    retcode = lorawan.send(MBED_CONF_LORA_APP_PORT, tx_buffer, packet_len,
                           MSG_UNCONFIRMED_FLAG);
……..
    printf("\r\n %d bytes scheduled for transmission \r\n", retcode);
    memset(tx_buffer, 0, sizeof(tx_buffer));
}


static void receive_message()
{
   int16_t retcode;
   retcode = lorawan.receive(rx_buffer, sizeof(rx_buffer), port, flags);

   if (retcode < 0) {
      printf("\r\n receive() - Error code %d \r\n", retcode);
      return;
   }

   printf(" Data:");

   for (uint8_t i = 0; i < retcode; i++) {
      printf("%x", rx_buffer[i]);
   }

   printf("\r\n Data Length: %d\r\n", retcode);

   memset(rx_buffer, 0, sizeof(rx_buffer));
}
```

LoRa in Mbed – Example stack events visible to application



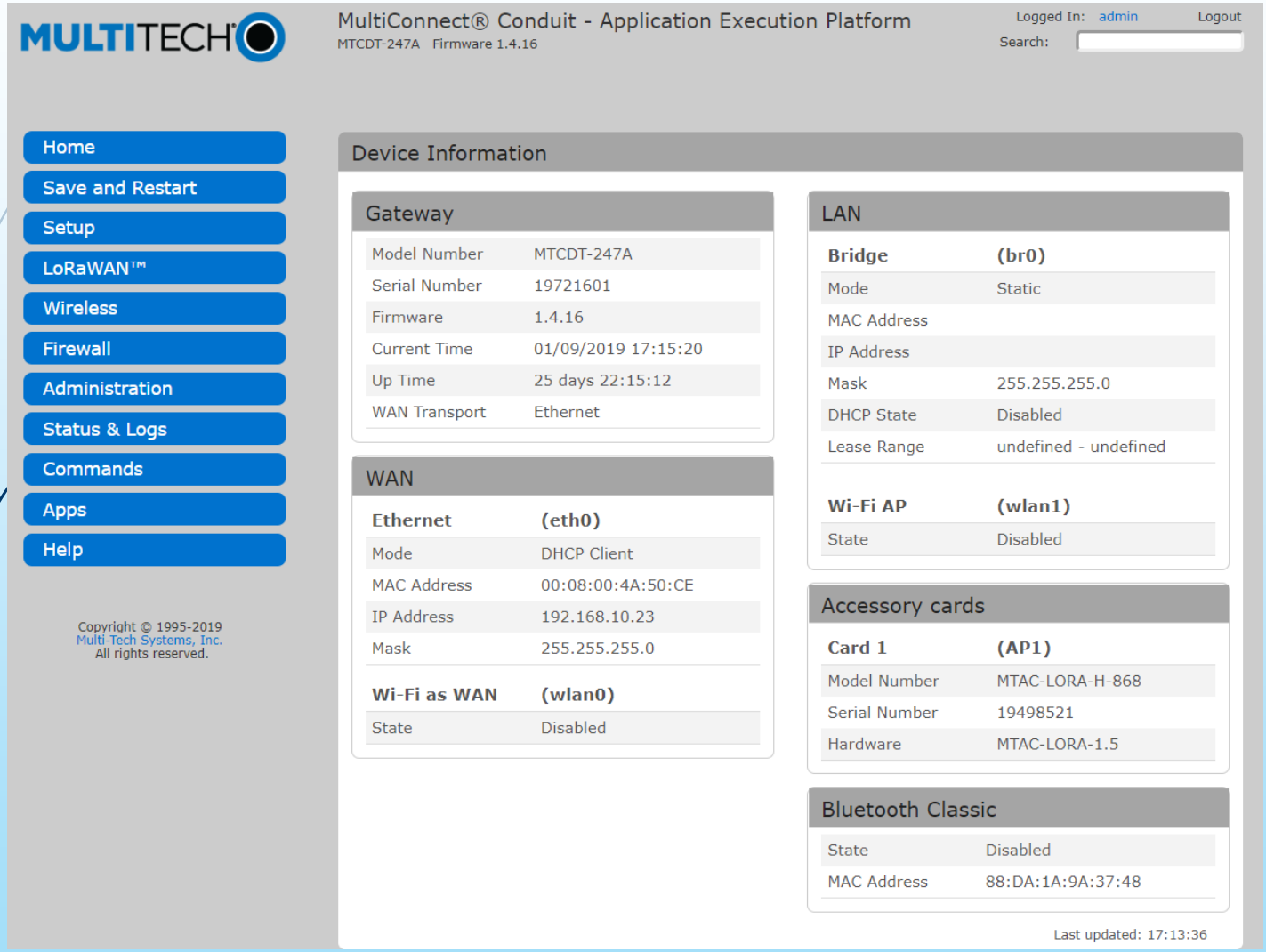
Demo

Gateway Mutitech Conduit: Hawdware

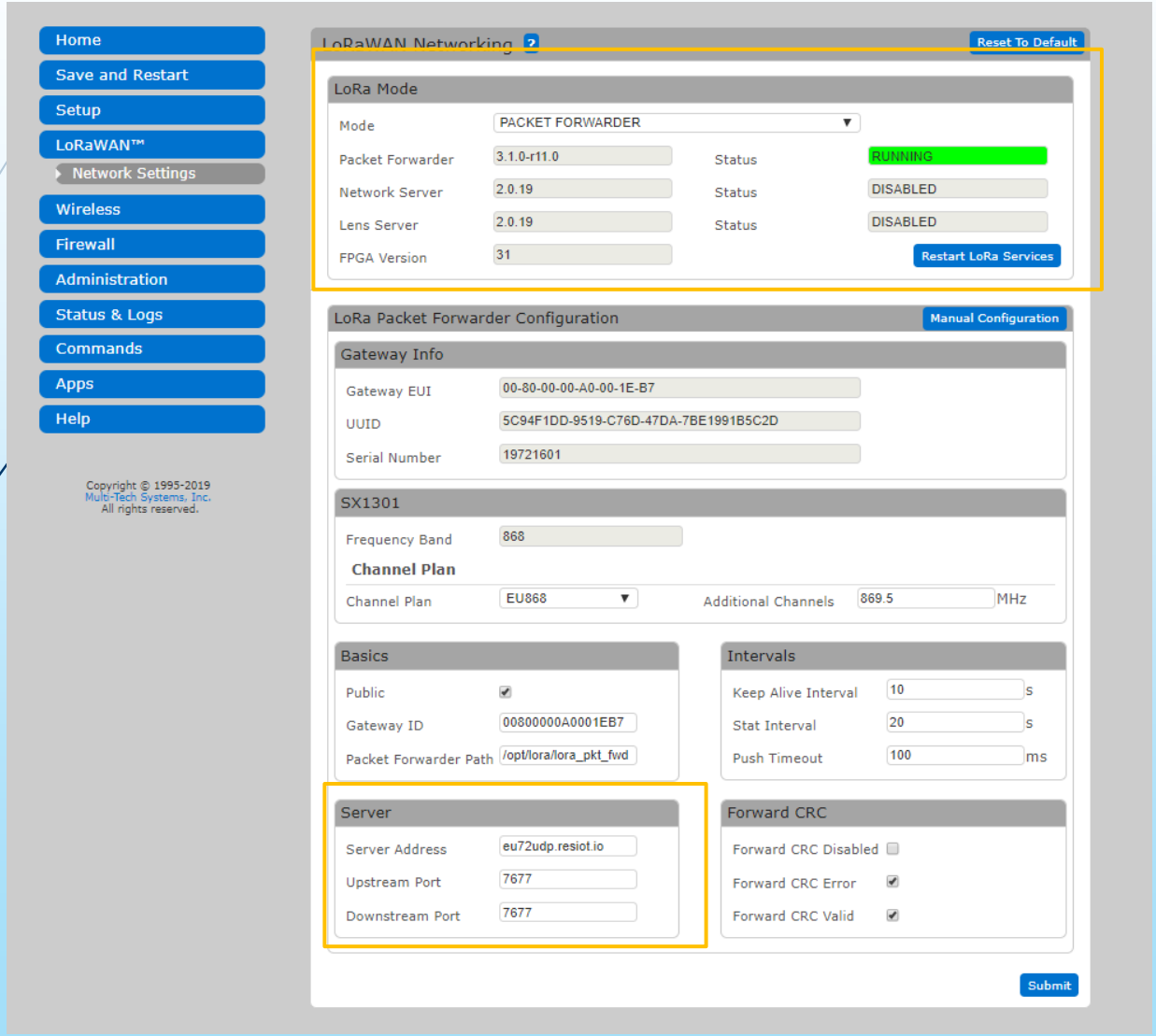


- Programmable Gateway for the Internet of Things (MTCDT-247A)
- Wi-Fi communication supporting 802.11 a/b/g/n 2.4 GHz and 5GHz with WPA2 personal transmission security. Wi-Fi Access Point and Client modes are supported simultaneously.
- BT Classic and BLE 4.1 communication supports local connectivity with automatic pairing with target devices utilizing 128 bit link key length security.
- GNSS module for LoRaWAN packet time-stamping and geo-location capability
- Backhaul options include 4G-LTE, 3G, 2G cellular or Ethernet for cost effective global deployment
- LORA FEATURES
  - Certified for Europe 868 MHz, North American and Australian 915 MHz ISM bands
  - 27 dBm (output power) support for European region
  - ISM band scanning for optimum LoRa® performance
  - Listen Before Talk LoRa operating protocol

Gateway Configuration for LoRawan



LoRaWan Setup



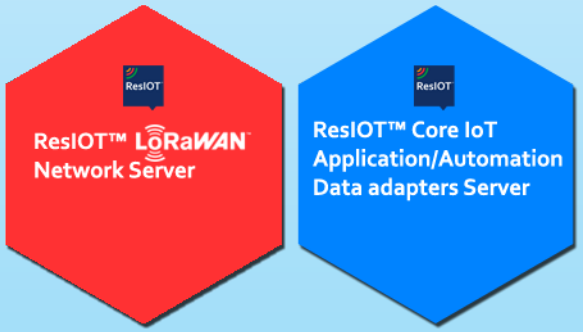
Where is the GW?

ResIoT: Main features

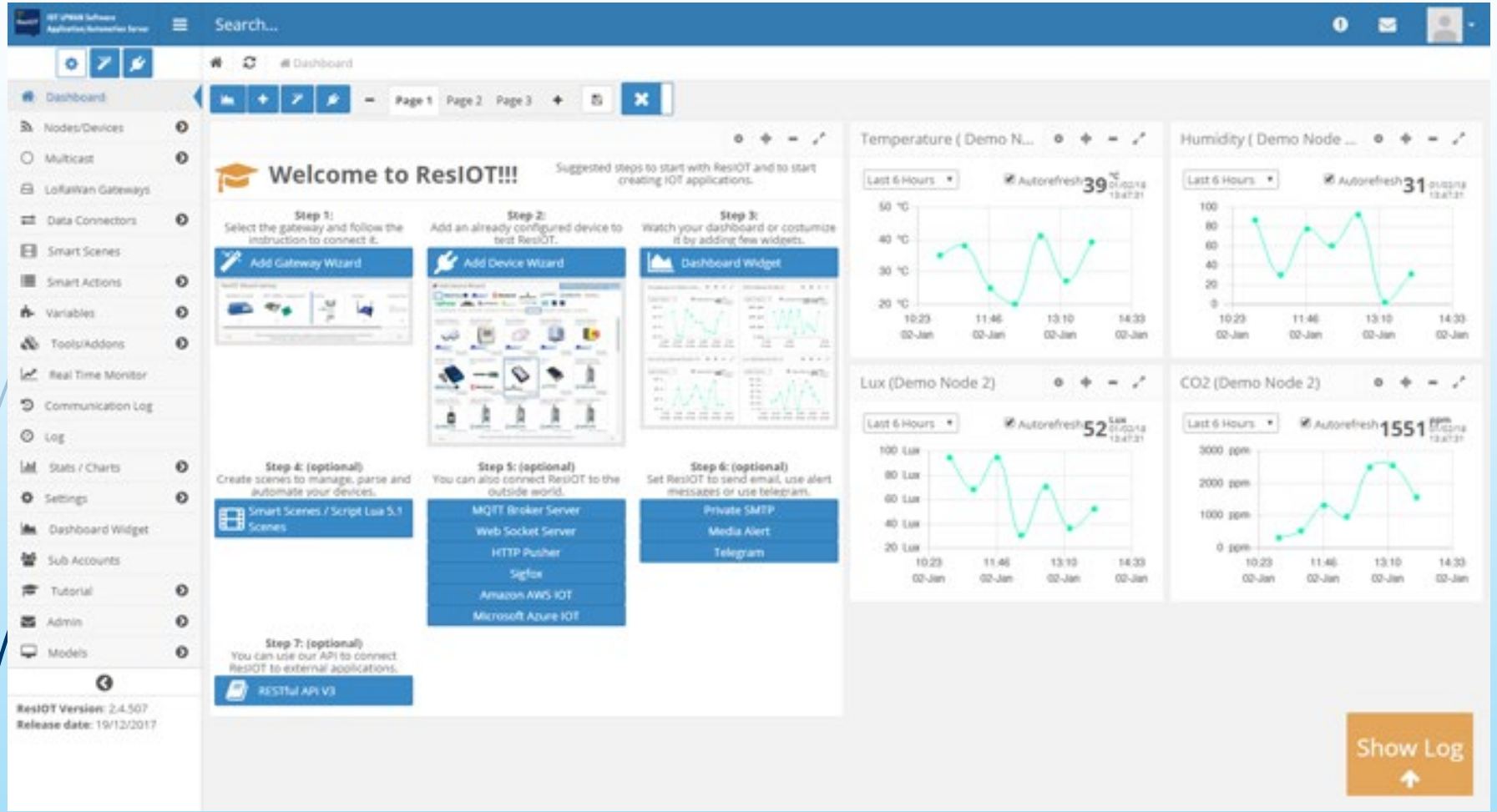
- IoT Platform and LoRaWAN™ Network Server (On-Premises or Cloud)
- ResIOT™ IoT Platform provides all the necessary software to manage networks with millions of devices but also to manage small private network
- The entire platform can be installed on Linux or Windows server clusters but also on Embedded Arm, x86, x64 systems, as an all-in-one gateway. Below you will find all the software features of the ResIOT™ IoT Platform
- ResIOT™ is suitable for all LPWAN and IoT projects for Smart City or Industry 4.0 and can be used free of charge up to 15 Devices and 1 LoRaWAN™ Gateway

ResIOT: Main software elements

- ResIOT Core IoT Application Automation Data adapters Server
  - Is the heart of the platform, manages all configured data, maintains the link with all ResIOT™ LoRaWan™ Network Servers and all other connectors, provides a web interface for configuration but also for the creation of IoT projects, manages the data automation, data store in the database and many other features
- ResIOT™ LoRaWAN™ Network Server
  - Manages all the features of the LoRaWAN™ network, such as node authentication, encryption key maintenance, uplink management and downlink transmission through all the base stations / gateways of the LPWAN network



Dashboard/Control Center



ResIOT™ LoRaWAN™ Base Station/Gateway Server

- It is the main component for connection to all the LoRaWAN™ Base Station / Gateway of the network. Allows the receipt and transmission of all data from the LoRaWAN™ devices and collects the monitoring data of each Base Station / Gateway. ResIOT™ LoRaWAN Base Station / Gateway Server communicates with ResIOT™ LoRaWan Network Server and ResIOT™ IoT Application / Automation / Data adapters Server with MQTT protocol

ResIOT™ LoRaWAN™ Base Station/Gateway Packet forwarder

- It is the component that is installed on the Base Station / Gateway. It allows the management of the LoRa radio cards, takes care of the transmission and reception of all data and communicates constantly with the ResIOT™ component LoRaWAN™ Base Station / Gateway Server. Available for Multitech Ftdi, Multitech SPI, Kerlink Ftdi, IMST + Raspberry SPI and other gateways. It also allows to remotely configure data such as gain antenna, frequencies, server connection parameters, etcc

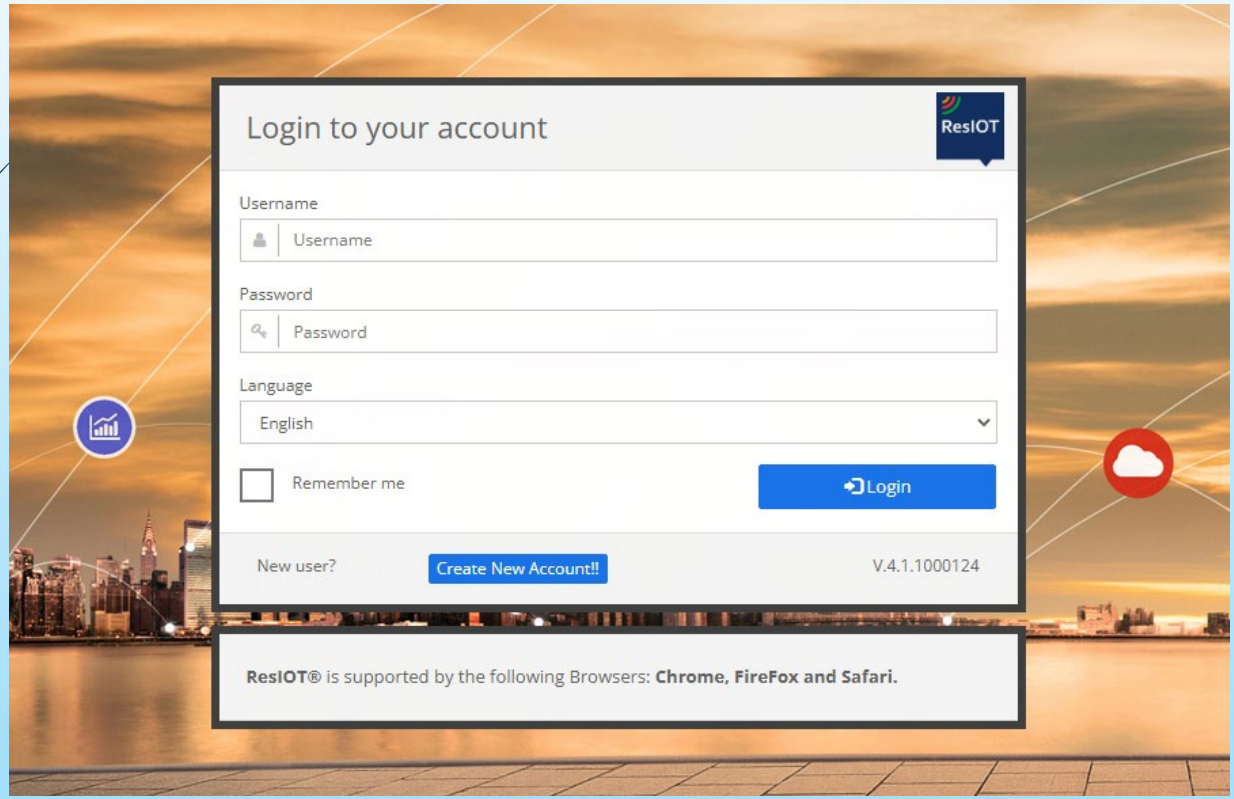
We are not using this option

Other components

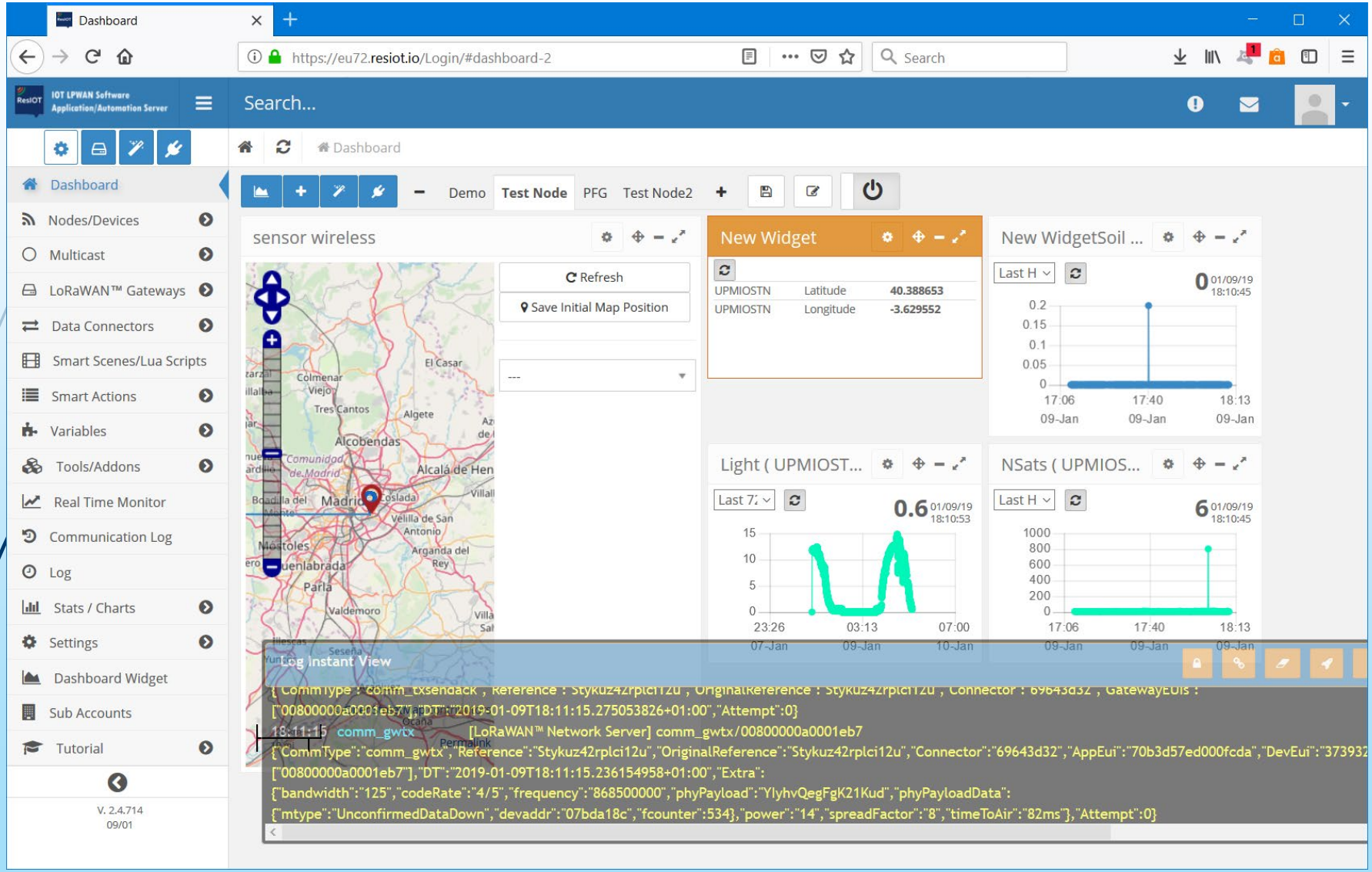
- Automatic update of the software installed in your local installation (not our case)
- Mobile APP for Android/IOS. Please install it in your mobiles!

ResIoT

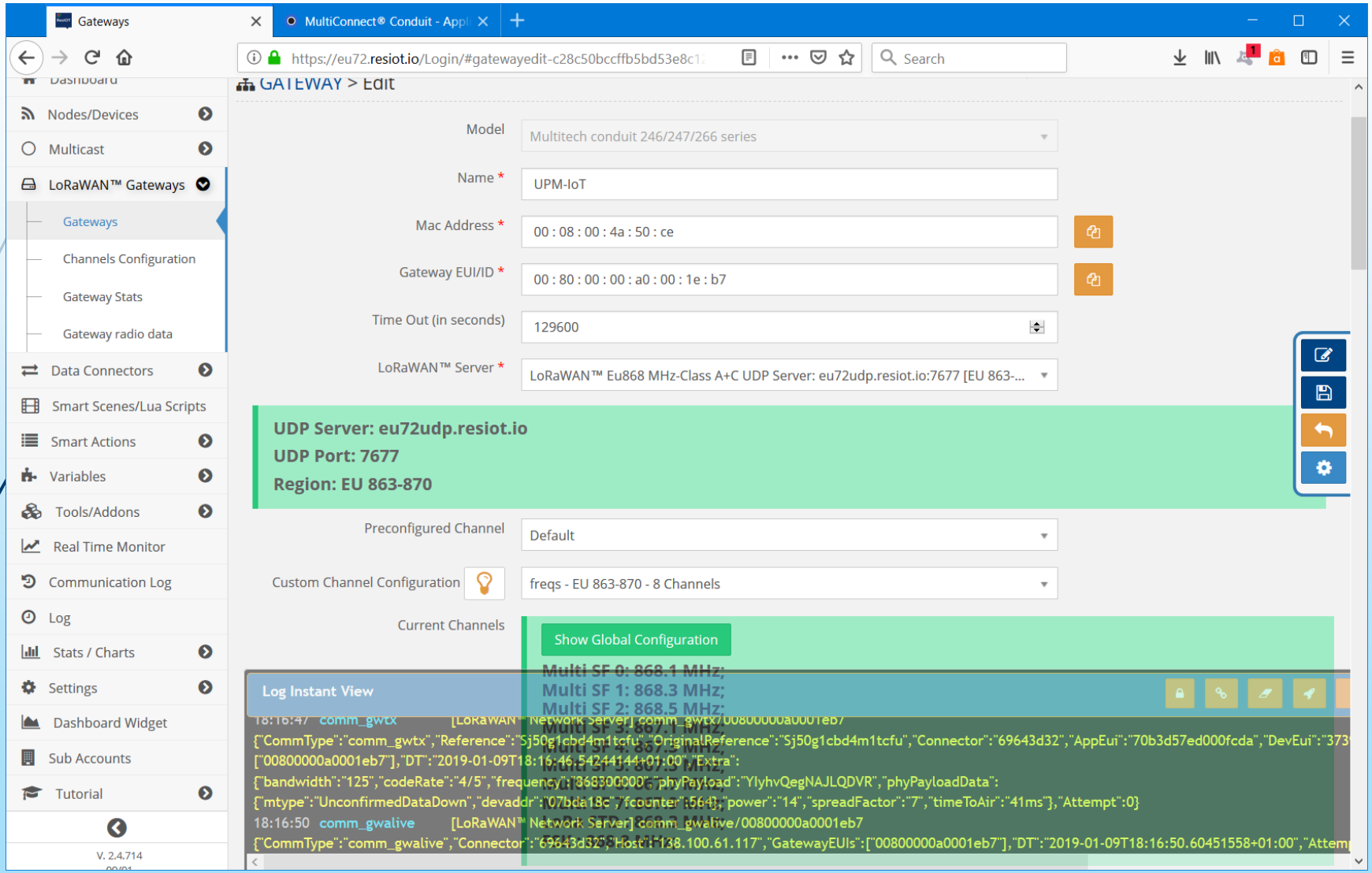
- URL: https://eu72.resiot.io/
- Every student has a login and a password to access to the cloud application



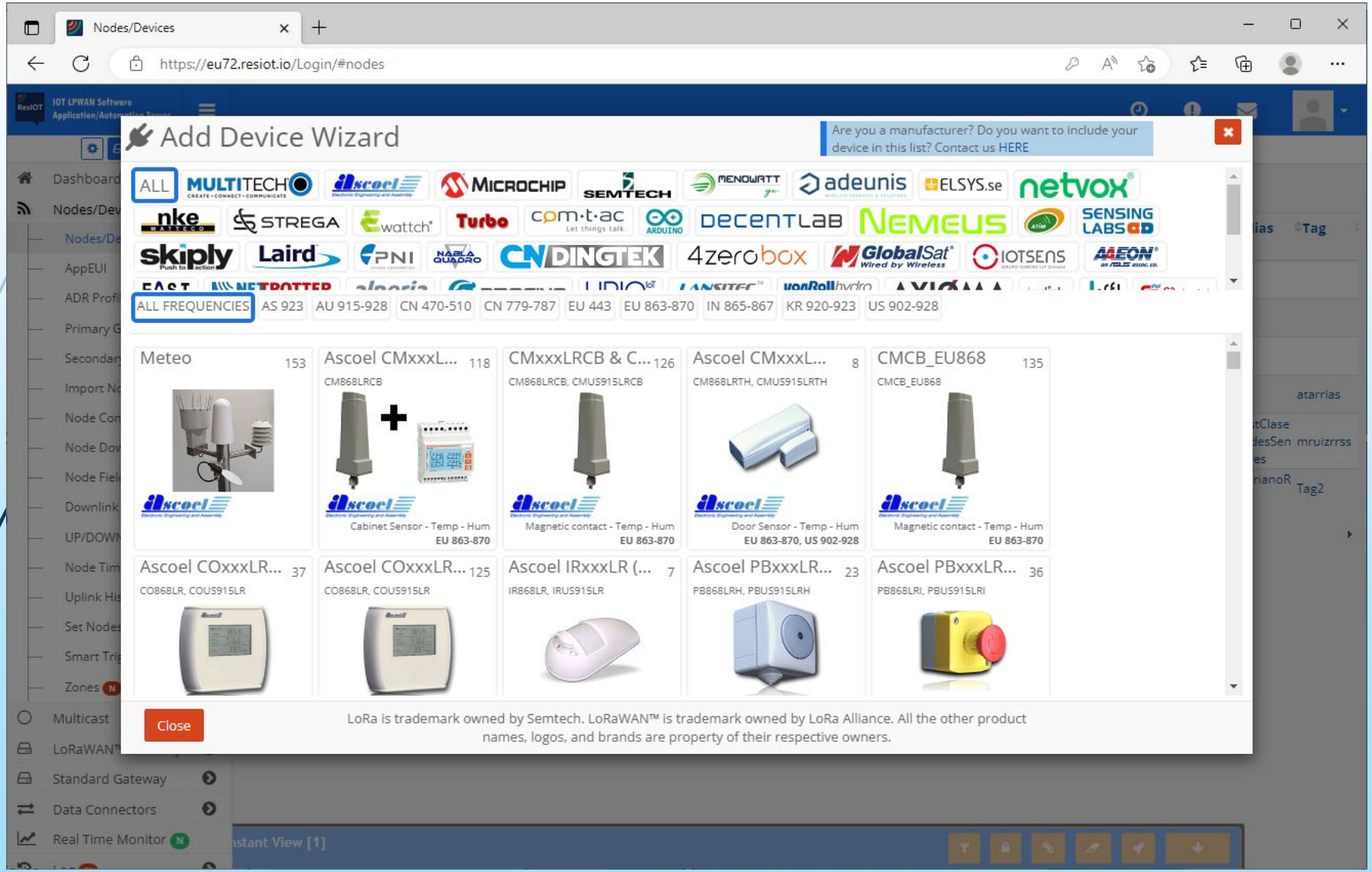
Dashboard

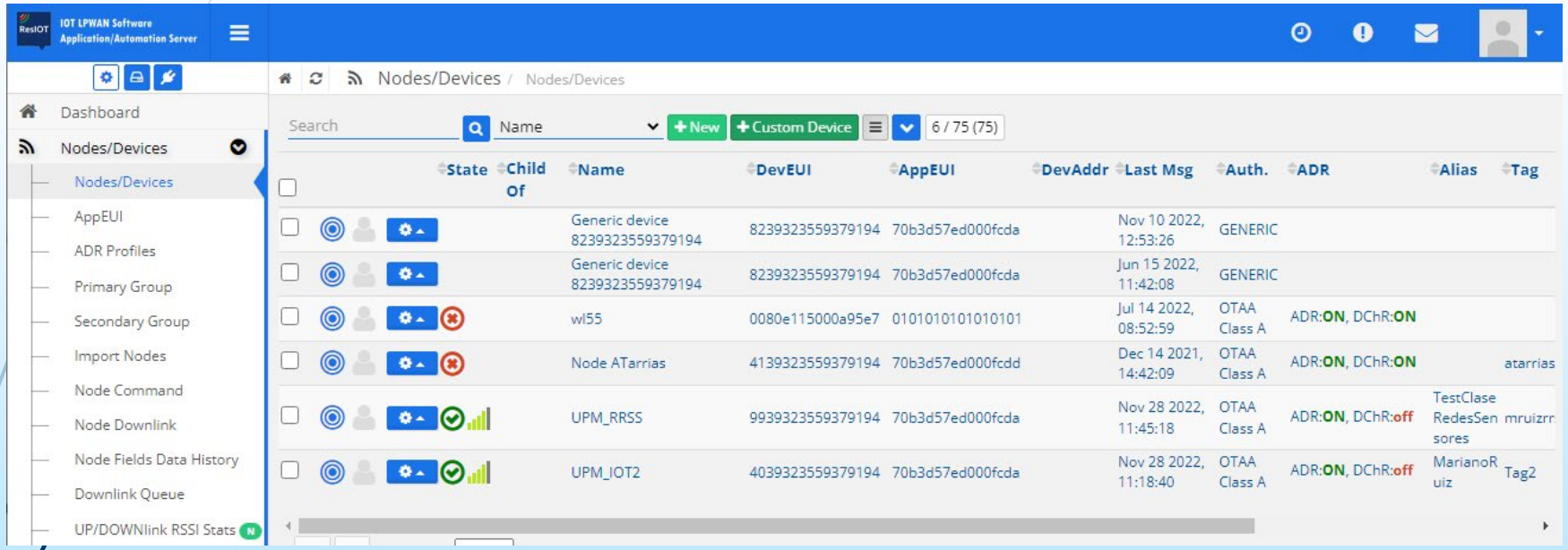

Dashboard
https://eu72.resiot.io/Login/#dashboard-2
Search...
IOT LPWAN Software Application/Automation Server
Dashboard
Nodes/Devices
Multicast
LoRaWAN™ Gateways
Data Connectors
Smart Scenes/Lua Scripts
Smart Actions
Variables
Tools/Addons
Real Time Monitor
Communication Log
Log
Stats / Charts
Settings
Dashboard Widget
Sub Accounts
Tutorial
V. 2.4.714
09/01
Demo
Test Node
PFG
Test Node2
sensor wireless
Refresh
Save Initial Map Position
New Widget
UPMIOSTN Latitude 40.388653
UPMIOSTN Longitude -3.629552
New WidgetSoil ...
Light (UPMIOST...
NSats (UPMIOS...
Log Instant View
[LoRaWAN™ Network Server] comm_gwtx/00800000a0001eb7

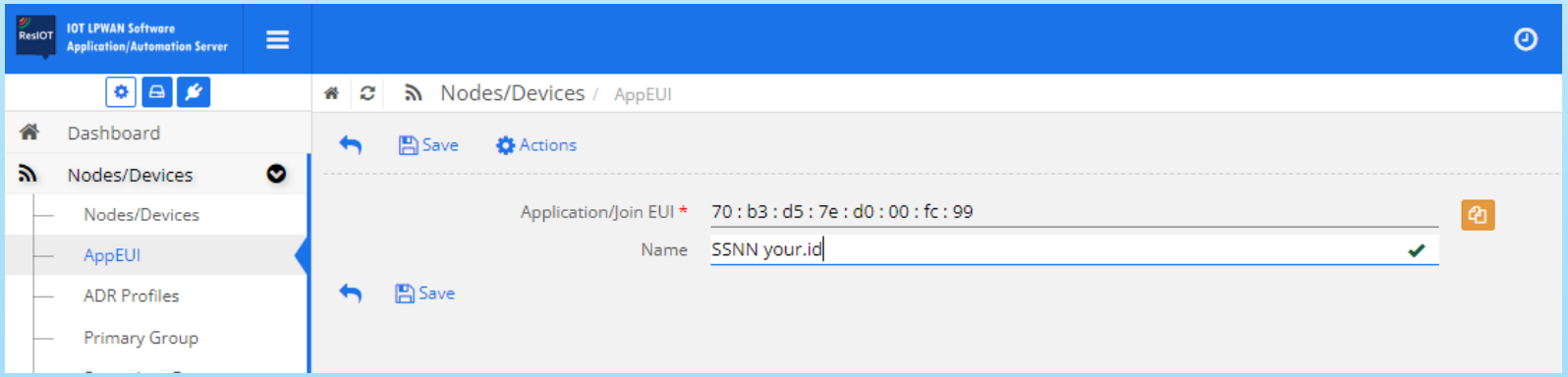
Adding a Gateway

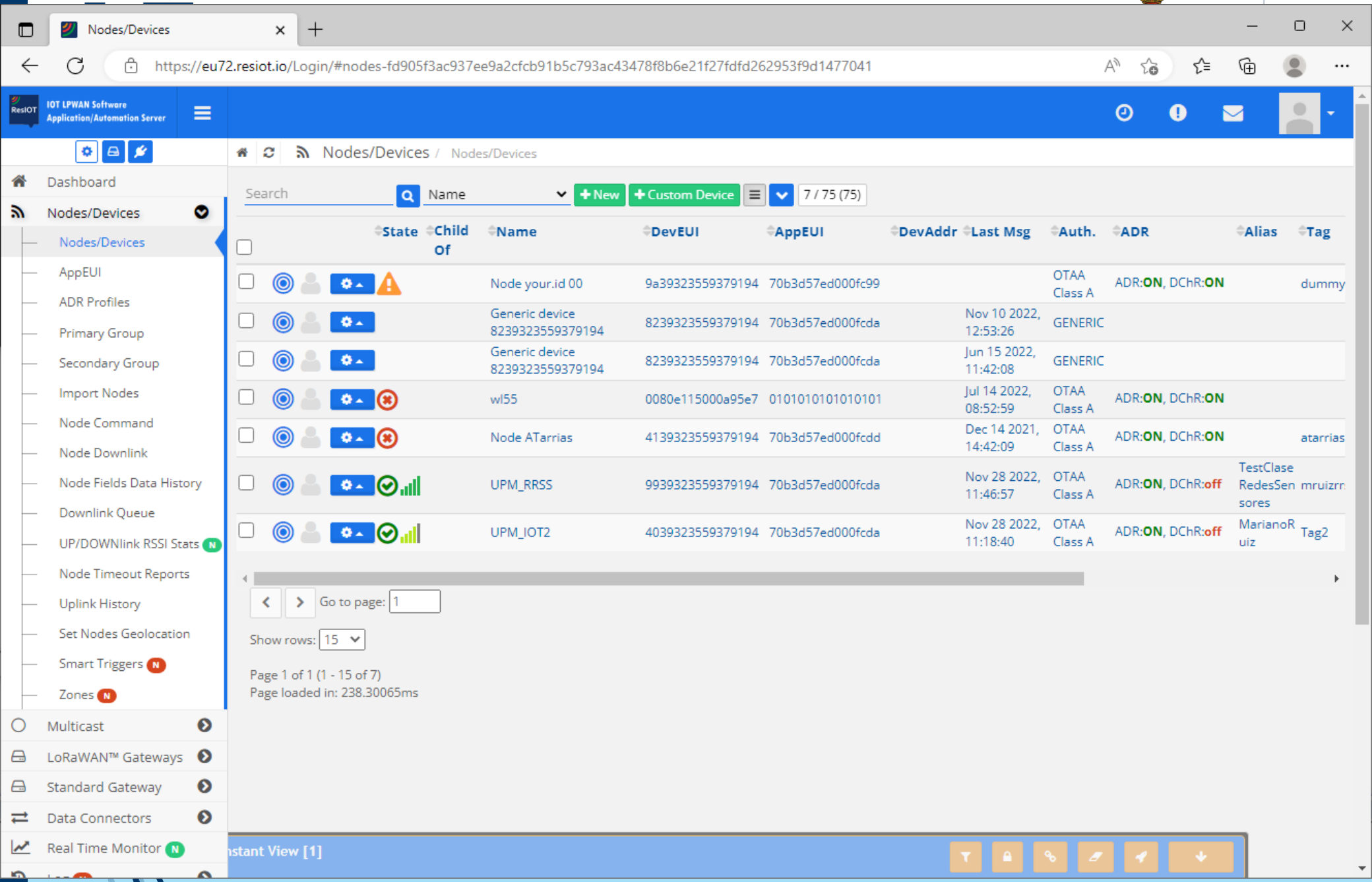


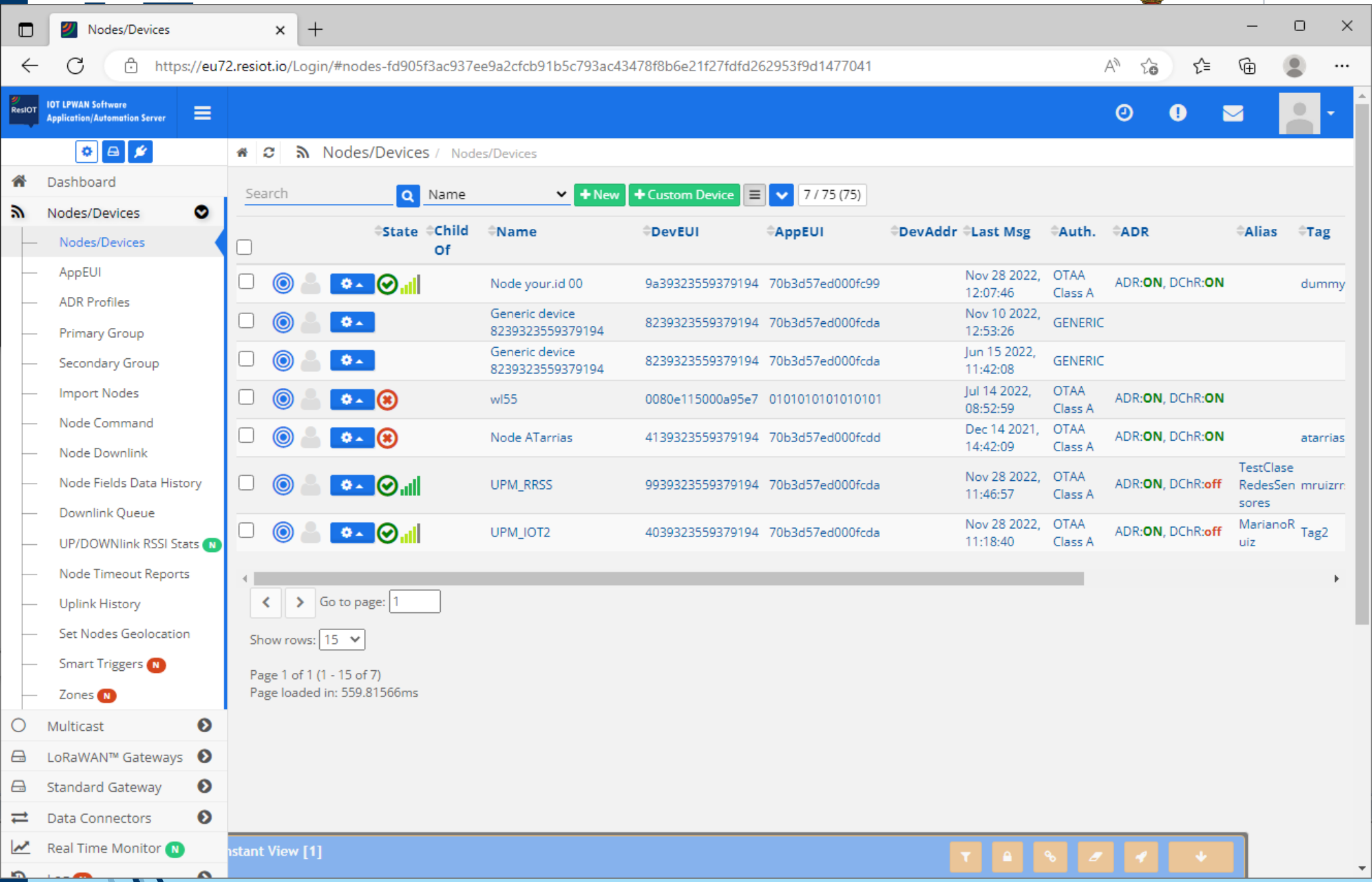
Adding a Node: Step 0, wizard



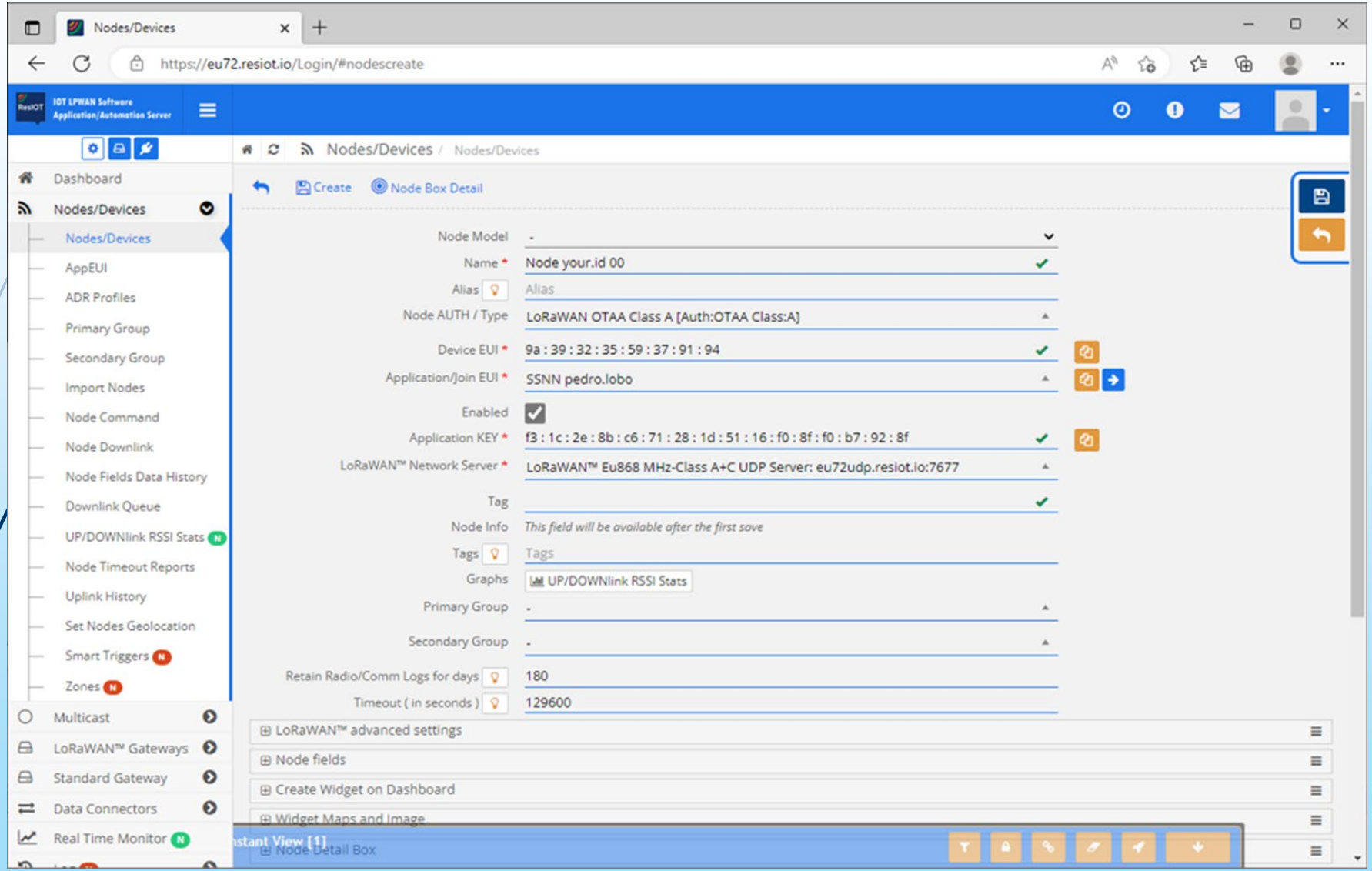

IOT LPWAN Software
Application/Automation Server
Nodes/Devices / Nodes/Devices
Dashboard
Nodes/Devices
Nodes/Devices
AppEUI
ADR Profiles
Primary Group
Secondary Group
Import Nodes
Node Command
Node Downlink
Node Fields Data History
Downlink Queue
UP/DOWNlink RSSI Stats
Search
Name
+ New
+ Custom Device
6 / 75 (75)
State
Child Of
Name
DevEUI
AppEUI
DevAddr
Last Msg
Auth.
ADR
Alias
Tag
Generic device 8239323559379194
8239323559379194
70b3d57ed000fcda
Nov 10 2022, 12:53:26
GENERIC
Generic device 8239323559379194
8239323559379194
70b3d57ed000fcda
Jun 15 2022, 11:42:08
GENERIC
wl55
0080e115000a95e7
0101010101010101
Jul 14 2022, 08:52:59
OTAA Class A
ADR:ON, DChR:ON
Node ATarrias
4139323559379194
70b3d57ed000fcdd
Dec 14 2021, 14:42:09
OTAA Class A
ADR:ON, DChR:ON
atarrias
UPM_RRSS
9939323559379194
70b3d57ed000fcda
Nov 28 2022, 11:45:18
OTAA Class A
ADR:ON, DChR:off
TestClase RedesSensores
mruizrr
UPM_IOT2
4039323559379194
70b3d57ed000fcda
Nov 28 2022, 11:18:40
OTAA Class A
ADR:ON, DChR:off
MarianoRuiz
Tag2


IOT LPWAN Software
Application/Automation Server
Nodes/Devices / AppEUI
Dashboard
Nodes/Devices
Nodes/Devices
AppEUI
ADR Profiles
Primary Group
Save
Actions
Application/Join EUI *
70 : b3 : d5 : 7e : d0 : 00 : fc : 99
Name
SSNN your.id
Save

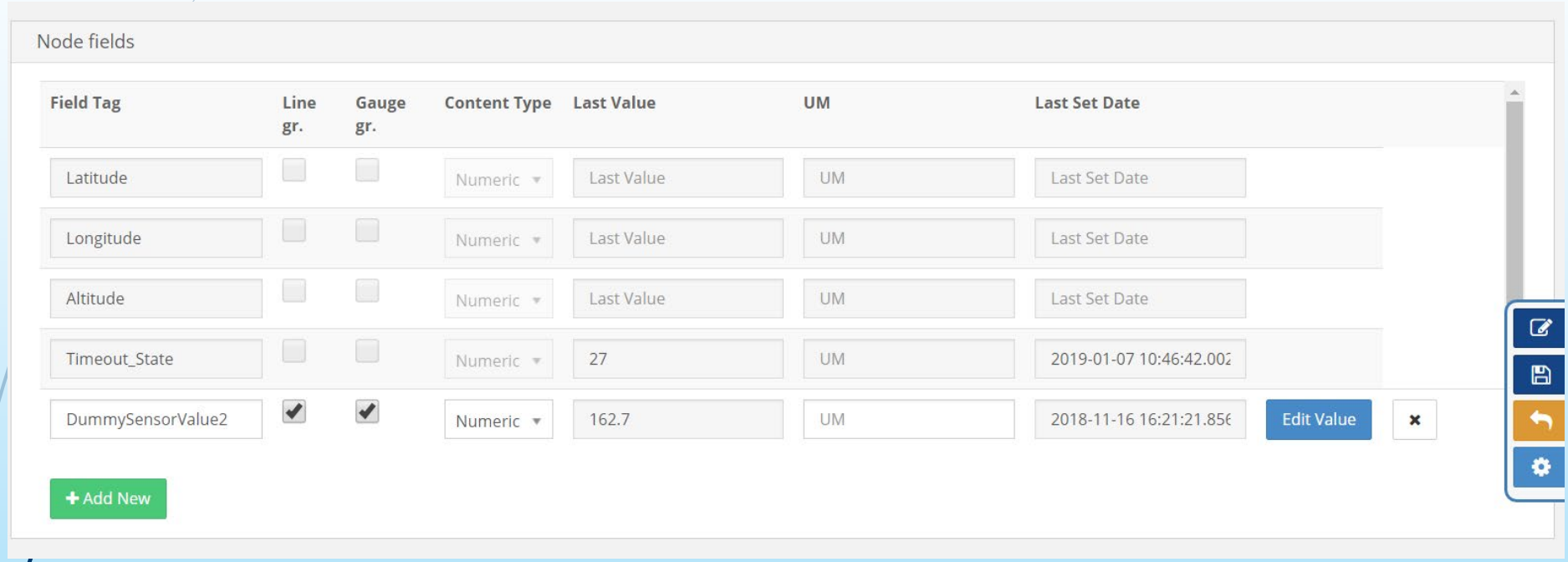

Nodes/Devices
https://eu72.resiot.io/Login/#nodes-fd905f3ac937ee9a2cfcb91b5c793ac43478f8b6e21f27fdfd262953f9d1477041
IOT LPWAN Software Application/Automation Server
Nodes/Devices / Nodes/Devices
Dashboard
Nodes/Devices
Nodes/Devices
AppEUI
ADR Profiles
Primary Group
Secondary Group
Import Nodes
Node Command
Node Downlink
Node Fields Data History
Downlink Queue
UP/DOWNlink RSSI Stats
Node Timeout Reports
Uplink History
Set Nodes Geolocation
Smart Triggers
Zones
Multicast
LoRaWAN™ Gateways
Standard Gateway
Data Connectors
Real Time Monitor
Search
Name
+ New
+ Custom Device
7 / 75 (75)
State
Child Of
Name
DevEUI
AppEUI
DevAddr
Last Msg
Auth.
ADR
Alias
Tag
Node your.id 00
9a39323559379194
70b3d57ed000fc99
OTAA Class A
ADR:ON, DChR:ON
dummy
Generic device 8239323559379194
8239323559379194
70b3d57ed000fcda
Nov 10 2022, 12:53:26
GENERIC
Generic device 8239323559379194
8239323559379194
70b3d57ed000fcda
Jun 15 2022, 11:42:08
GENERIC
wl55
0080e115000a95e7
0101010101010101
Jul 14 2022, 08:52:59
OTAA Class A
ADR:ON, DChR:ON
Node ATarrias
4139323559379194
70b3d57ed000fcdd
Dec 14 2021, 14:42:09
OTAA Class A
ADR:ON, DChR:ON
atarrias
UPM_RRSS
9939323559379194
70b3d57ed000fcda
Nov 28 2022, 11:46:57
OTAA Class A
ADR:ON, DChR:off
TestClaseRedesSensores
mruizrr
UPM_IOT2
4039323559379194
70b3d57ed000fcda
Nov 28 2022, 11:18:40
OTAA Class A
ADR:ON, DChR:off
MarianoRuiz
Tag2
Go to page: 1
Show rows: 15
Page 1 of 1 (1 - 15 of 7)
Page loaded in: 238.30065ms


Nodes/Devices
https://eu72.resiot.io/Login/#nodes-fd905f3ac937ee9a2cfcb91b5c793ac43478f8b6e21f27fdfd262953f9d1477041
IOT LPWAN Software Application/Automation Server
Nodes/Devices / Nodes/Devices
Dashboard
Nodes/Devices
Nodes/Devices
AppEUI
ADR Profiles
Primary Group
Secondary Group
Import Nodes
Node Command
Node Downlink
Node Fields Data History
Downlink Queue
UP/DOWNlink RSSI Stats
Node Timeout Reports
Uplink History
Set Nodes Geolocation
Smart Triggers
Zones
Multicast
LoRaWAN™ Gateways
Standard Gateway
Data Connectors
Real Time Monitor
Search
Name
+ New
+ Custom Device
7 / 75 (75)
State
Child Of
Name
DevEUI
AppEUI
DevAddr
Last Msg
Auth.
ADR
Alias
Tag
Node your.id 00
9a39323559379194
70b3d57ed000fc99
Nov 28 2022, 12:07:46
OTAA Class A
ADR:ON, DChR:ON
dummy
Generic device 8239323559379194
8239323559379194
70b3d57ed000fcda
Nov 10 2022, 12:53:26
GENERIC
Generic device 8239323559379194
8239323559379194
70b3d57ed000fcda
Jun 15 2022, 11:42:08
GENERIC
wl55
0080e115000a95e7
0101010101010101
Jul 14 2022, 08:52:59
OTAA Class A
ADR:ON, DChR:ON
Node ATarrias
4139323559379194
70b3d57ed000fcdd
Dec 14 2021, 14:42:09
OTAA Class A
ADR:ON, DChR:ON
atarrias
UPM_RRSS
9939323559379194
70b3d57ed000fcda
Nov 28 2022, 11:46:57
OTAA Class A
ADR:ON, DChR:off
TestClase RedesSensores
mruizrr
UPM_IOT2
4039323559379194
70b3d57ed000fcda
Nov 28 2022, 11:18:40
OTAA Class A
ADR:ON, DChR:off
MarianoRuiz
Tag2
Go to page: 1
Show rows: 15
Page 1 of 1 (1 - 15 of 7)
Page loaded in: 559.81566ms
stant View [1]

Adding a Node: Step 1, identification

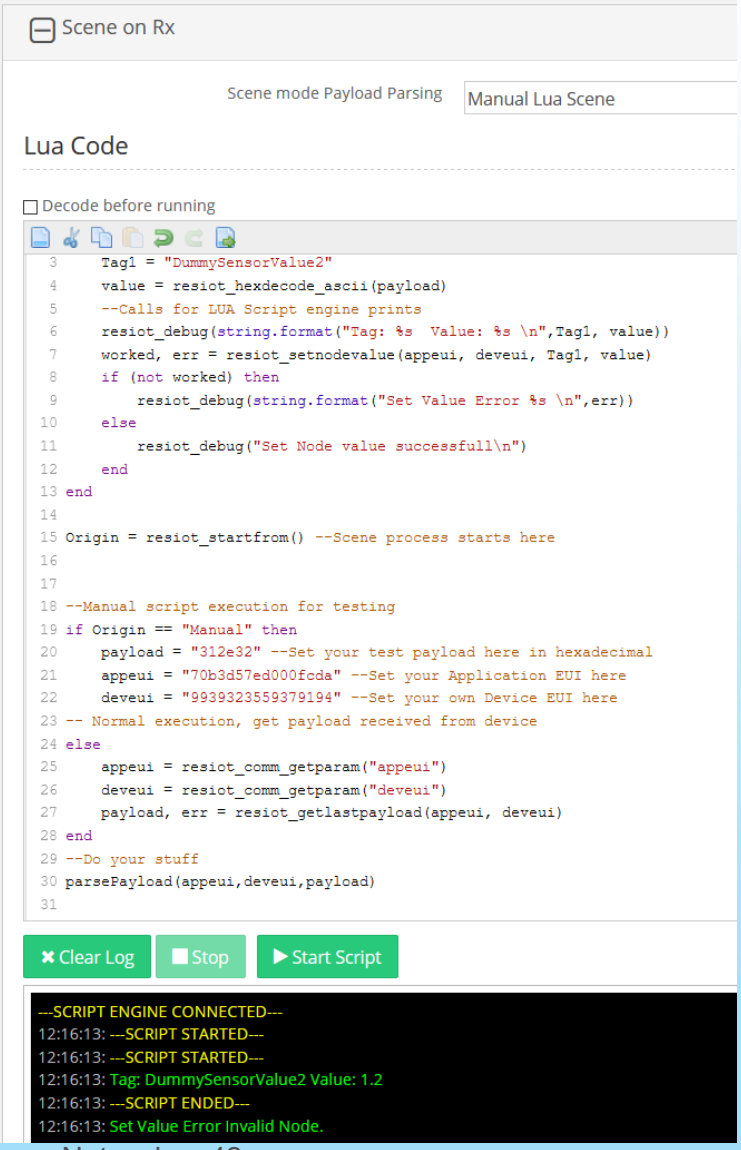


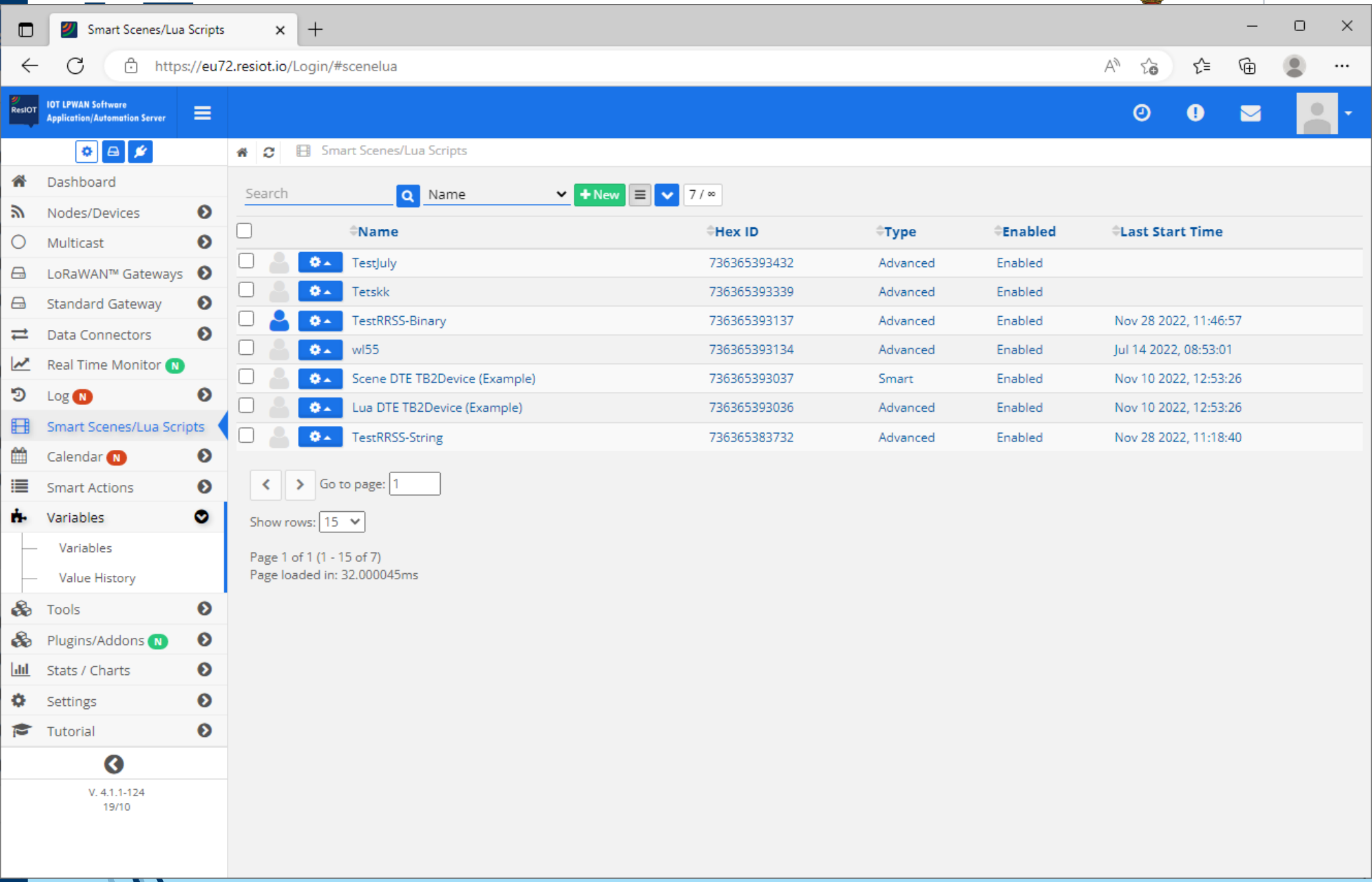
Adding a Node: Step 2, adding Node Fields

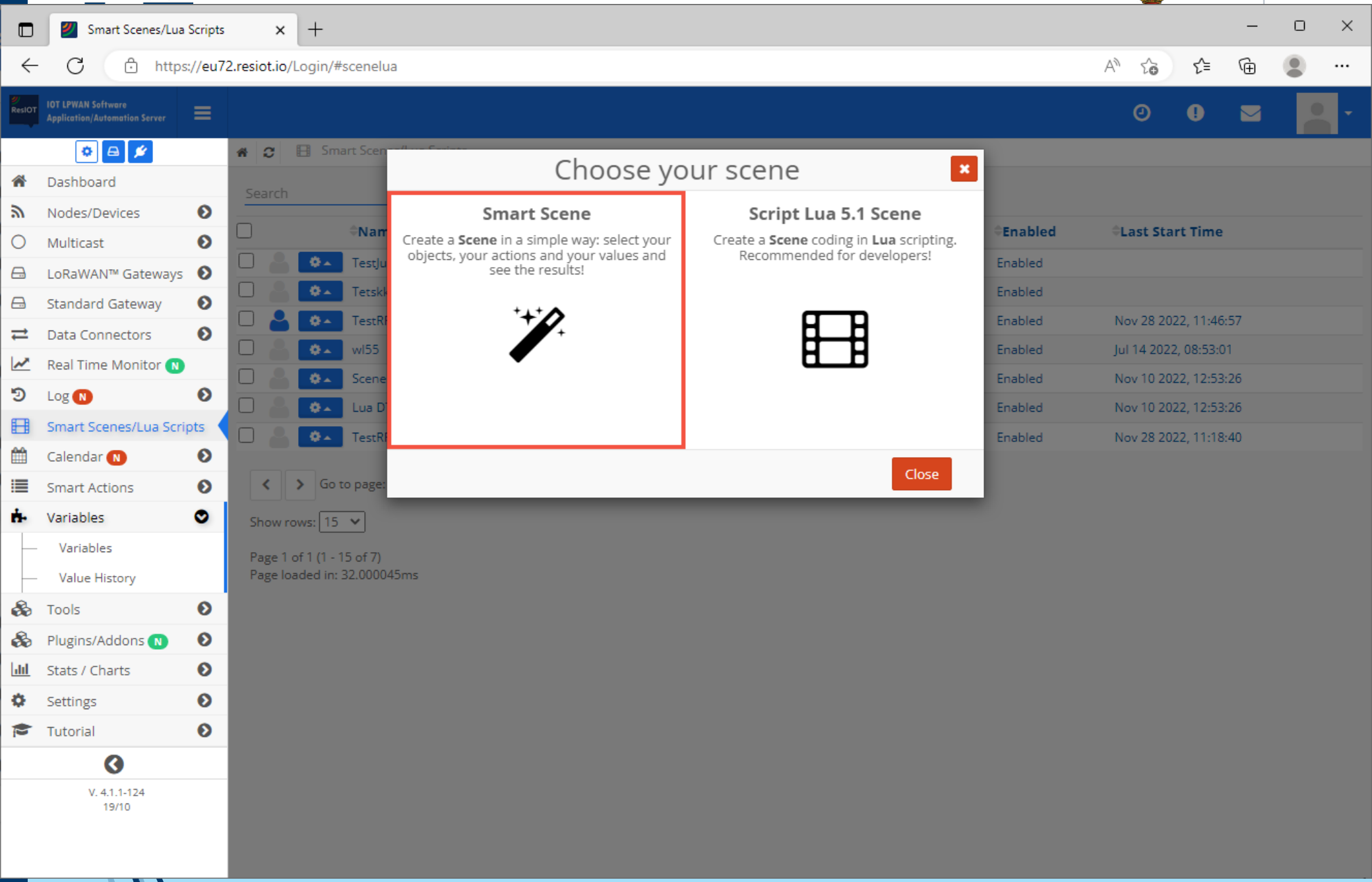


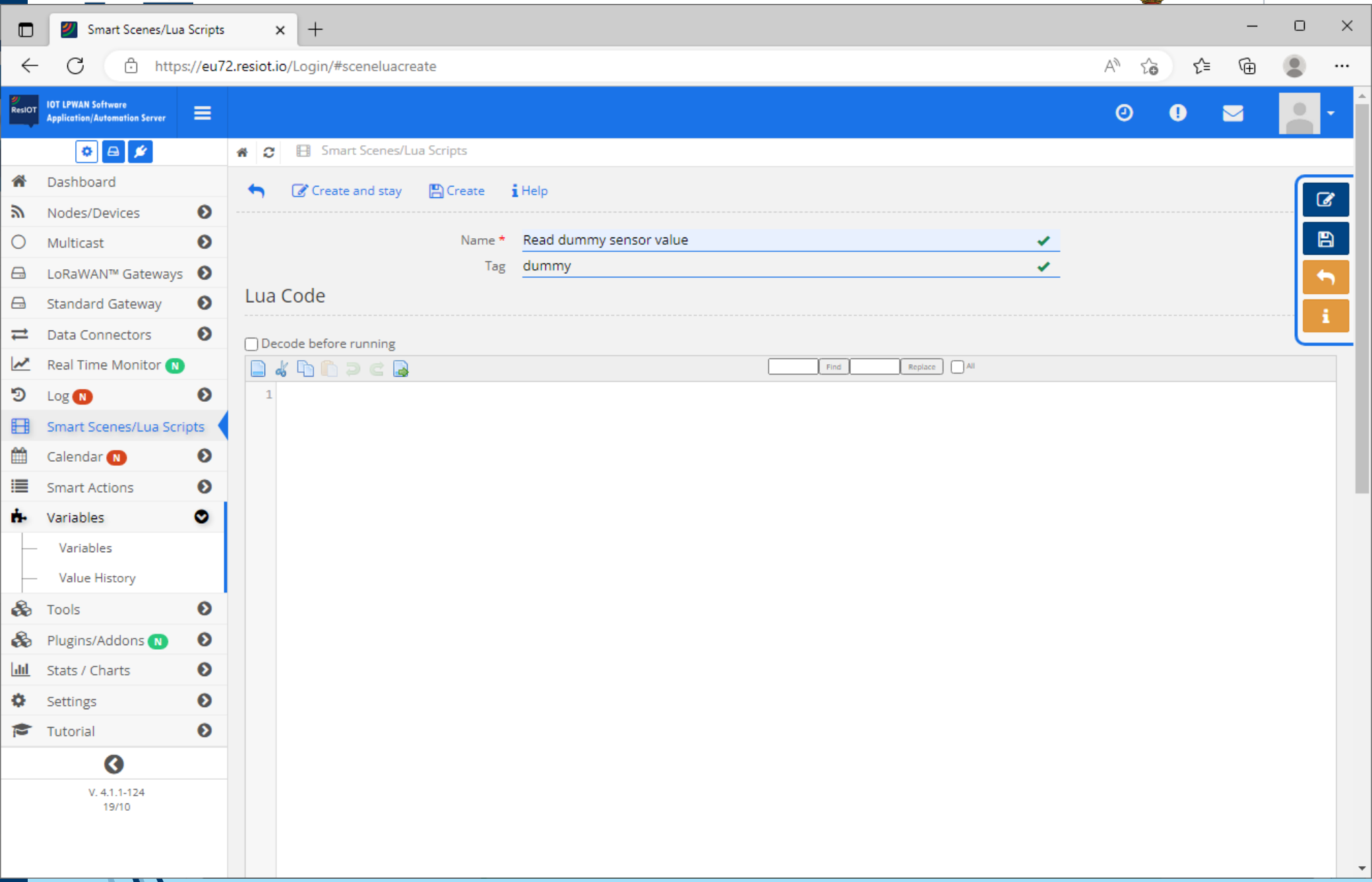
Adding a Node: Step 3, store payload data into Node Fields

- Is your responsibility to define the content of the payload and how to use it. ResIOT uses LUA scripting to “translate” the content of the payload to the TAGS
- In order to test the LUA script, ResIOT allows to execute the script in a test environment.
- Define your custom payload and use the terminal to debug your script




Smart Scenes/Lua Scripts
https://eu72.resiot.io/Login/#scenelua
IOT LPWAN Software Application/Automation Server
Smart Scenes/Lua Scripts
Dashboard
Nodes/Devices
Multicast
LoRaWAN™ Gateways
Standard Gateway
Data Connectors
Real Time Monitor N
Log N
Smart Scenes/Lua Scripts
Calendar N
Smart Actions
Variables
Variables
Value History
Tools
Plugins/Addons N
Stats / Charts
Settings
Tutorial
V. 4.1.1-124
19/10
Search
Name
New
7 / ∞
Name
Hex ID
Type
Enabled
Last Start Time
TestJuly 736365393432 Advanced Enabled
Tetskk 736365393339 Advanced Enabled
TestRRSS-Binary 736365393137 Advanced Enabled Nov 28 2022, 11:46:57
wl55 736365393134 Advanced Enabled Jul 14 2022, 08:53:01
Scene DTE TB2Device (Example) 736365393037 Smart Enabled Nov 10 2022, 12:53:26
Lua DTE TB2Device (Example) 736365393036 Advanced Enabled Nov 10 2022, 12:53:26
TestRRSS-String 736365383732 Advanced Enabled Nov 28 2022, 11:18:40
Go to page: 1
Show rows: 15
Page 1 of 1 (1 - 15 of 7)
Page loaded in: 32.000045ms


Smart Scenes/Lua Scripts
https://eu72.resiot.io/Login/#scenelua
IOT LPWAN Software
Application/Automation Server
Dashboard
Nodes/Devices
Multicast
LoRaWAN™ Gateways
Standard Gateway
Data Connectors
Real Time Monitor
Log
Smart Scenes/Lua Scripts
Calendar
Smart Actions
Variables
Variables
Value History
Tools
Plugins/Addons
Stats / Charts
Settings
Tutorial
V. 4.1.1-124
19/10
Search
Enabled
Last Start Time
Enabled
Nov 28 2022, 11:46:57
Jul 14 2022, 08:53:01
Nov 10 2022, 12:53:26
Nov 10 2022, 12:53:26
Nov 28 2022, 11:18:40
Go to page:
Show rows: 15
Page 1 of 1 (1 - 15 of 7)
Page loaded in: 32.000045ms
Choose your scene
Smart Scene
Create a Scene in a simple way: select your objects, your actions and your values and see the results!
Script Lua 5.1 Scene
Create a Scene coding in Lua scripting. Recommended for developers!
Close


Smart Scenes/Lua Scripts
https://eu72.resiot.io/Login/#sceneluacreate
IOT LPWAN Software
Application/Automation Server
Smart Scenes/Lua Scripts
Dashboard
Nodes/Devices
Multicast
LoRaWAN™ Gateways
Standard Gateway
Data Connectors
Real Time Monitor
Log
Smart Scenes/Lua Scripts
Calendar
Smart Actions
Variables
Variables
Value History
Tools
Plugins/Addons
Stats / Charts
Settings
Tutorial
V. 4.1.1-124
19/10
Create and stay
Create
Help
Name *
Read dummy sensor value
Tag
dummy
Lua Code
Decode before running
Find
Replace
All
1

Interesting tools in ResIoT

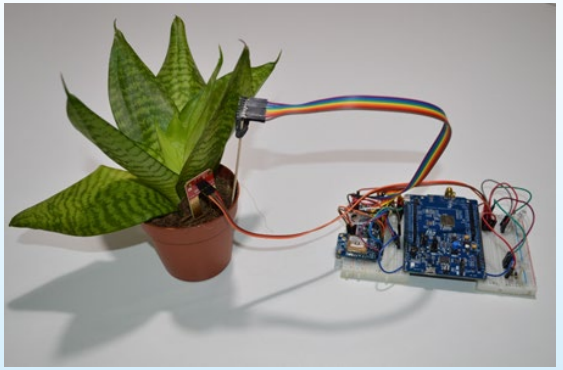
- Logging window

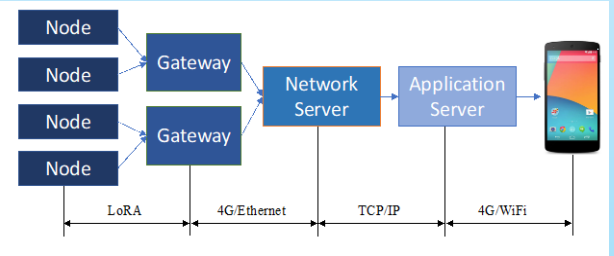
- Communication logs
- Statistics
- Creation of widgets
- Wizards to add gateways and nodes


```
21:34:55comm_rx[LoRaWAN™ Network Server] comm_rx/70b3d57ed000fcda/3639323559379194
{"CommType":"comm_rx","Connector":"69643d32","AppEui":"70b3d57ed000fcda","DevEui":"363
9323559379194","DevAddr":"074eebe0","Port":"15","GatewayEUIs":["00800000a0001eb7"],"Pa
yload":"302e302c302e302c302c302e3030303030302c302e303030303030","DT":"2019-01-
13T21:34:51.814948713+01:00",
"Extra":{"JSONRXINFO":[{"altitude":0,"latitude":40,"loRaSNR":9.5,"longitude":-3,
"mac":"00800000a0001eb7", "name":"00800000a0001eb7", "rssi":-
66}],"adr":"true","bandwidth":"125","fCnt":"14809","frequency":"868100000","spreadFact
or":"7"},"Attempt":0}
```

Project phases and basic specifications

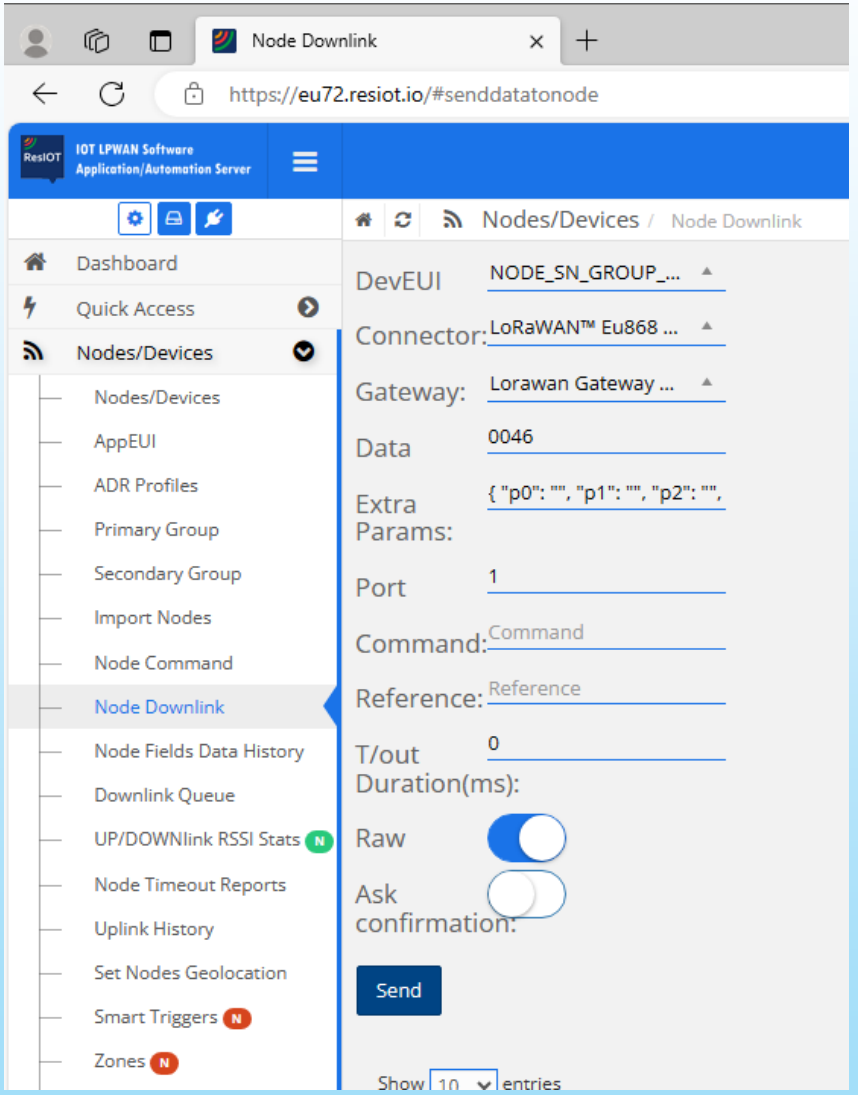
- The project has three different phases (see specifications document)
- Phase A: Use of LoraWAN Mbed example available at Moodle
  - Configuration
  - Setup of the node in ResIoT (add the basic LUA code)
  - Test (using Widgets). Use also your mobile
- Phase B: Adding your custom code developed in embedded hardware course
  - Adding code incrementally
  - Define the organization of your payload (maximum number of bytes to be used ….)
  - Add your LUA custom code and test
- Phase C: optional part! Improving the design





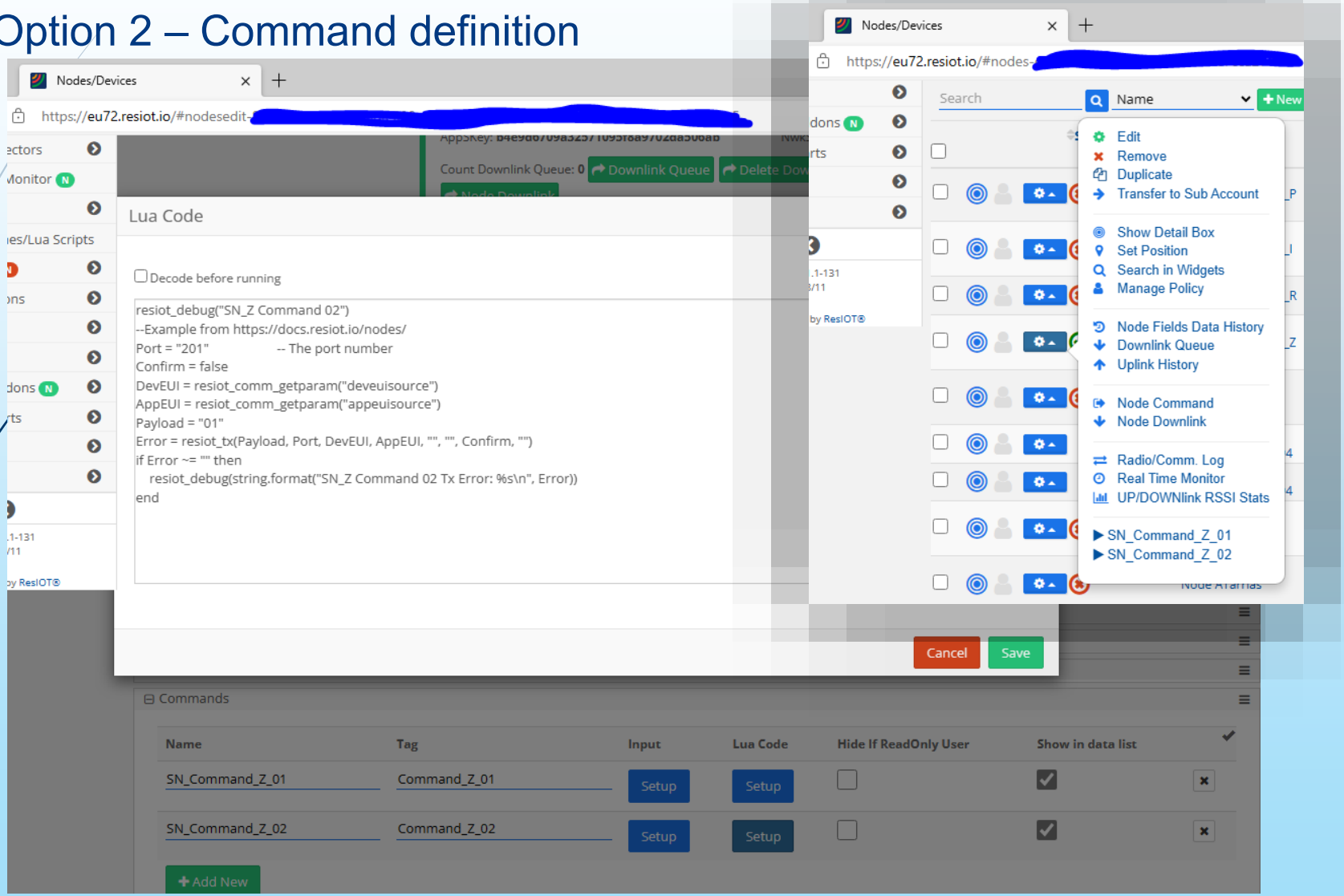
Sending commands to your node in ResIOT

- Option 1 – Direct interface
  - Raw (hex) / Text
  - Mind the confirmation



Sending commands to your node in ResIOT

Option 2 – Command definition



Q/A

- Questions
- Comments
- Next steps:
  - Read document with the Project specifications
  - Check the example provided and give us your feedback!
  - Check your solutions in the lab or at least close to the gateways!

Tested Coverage

- Around campus 200m (problems with walls and other metallic object)
- From a point with direct visibility (1.4km)

Attenuation

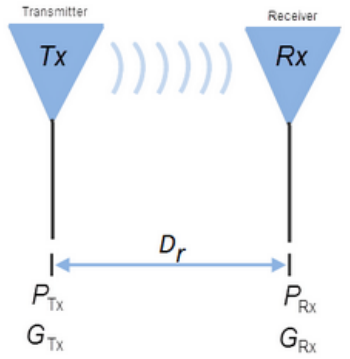
https://www.pasternack.com/t-calculator-friis.aspx

Transmitter Power:	+14	dBm
Transmitter Gain (dBi):	1	
Frequency:	868	MHz
Distance:	122	Meters
Receiver Gain (dBi):	6	

Calculate

Result:

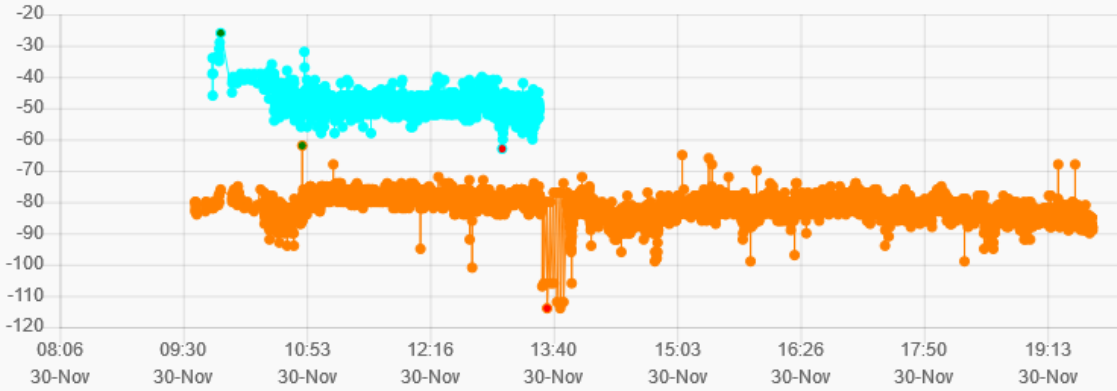
Received Power: -51.95 dBm



Statistics

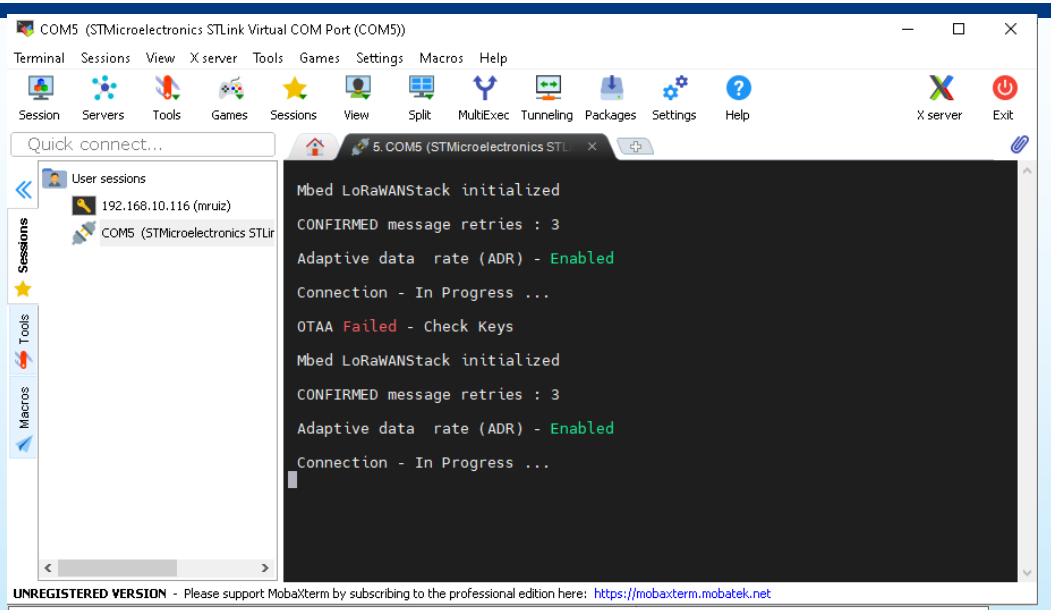
Rx/Uplink RSSI analysis

GwEui	Last RSSI|SNR	Min RSSI	Max RSSI	Average RSSI
1) 00800000a0003d71	-89 | 10	-114 | -14.8	-62 | 9	-81.67 | 8.22
2) 00800000a0001eb7	-50 | 10.5	-63 | 8.5	-26 | 10.8	-48.46 | 8.92

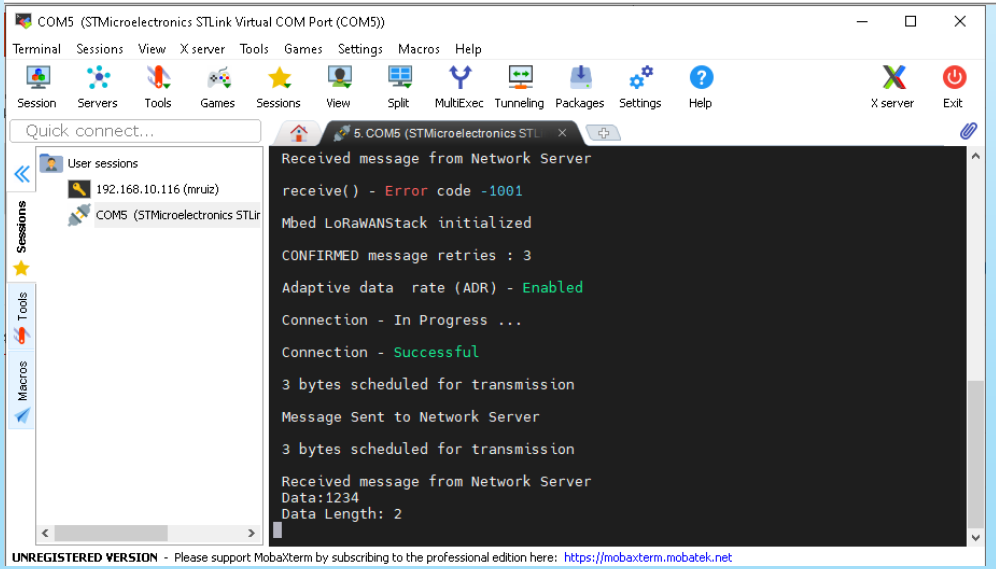


Typical screens

- Bad configuration or communication error in the initialization



- Successful TX and RX



Some guidelines

- Next class could be chaotic if all the nodes try to join simultaneously
  - Be patient! Joining and first connection takes some minutes
    - (yes, **minutes**)
  - Trial an error process!
- ResIoT:
  - We have an academic license with “low low low …. low, the lowest” priority. This implies that once the data is received it takes some minutes until widgets are updated
  - The platform has a lot of features for debugging, try to play with that and understand the internals of Lorawan