

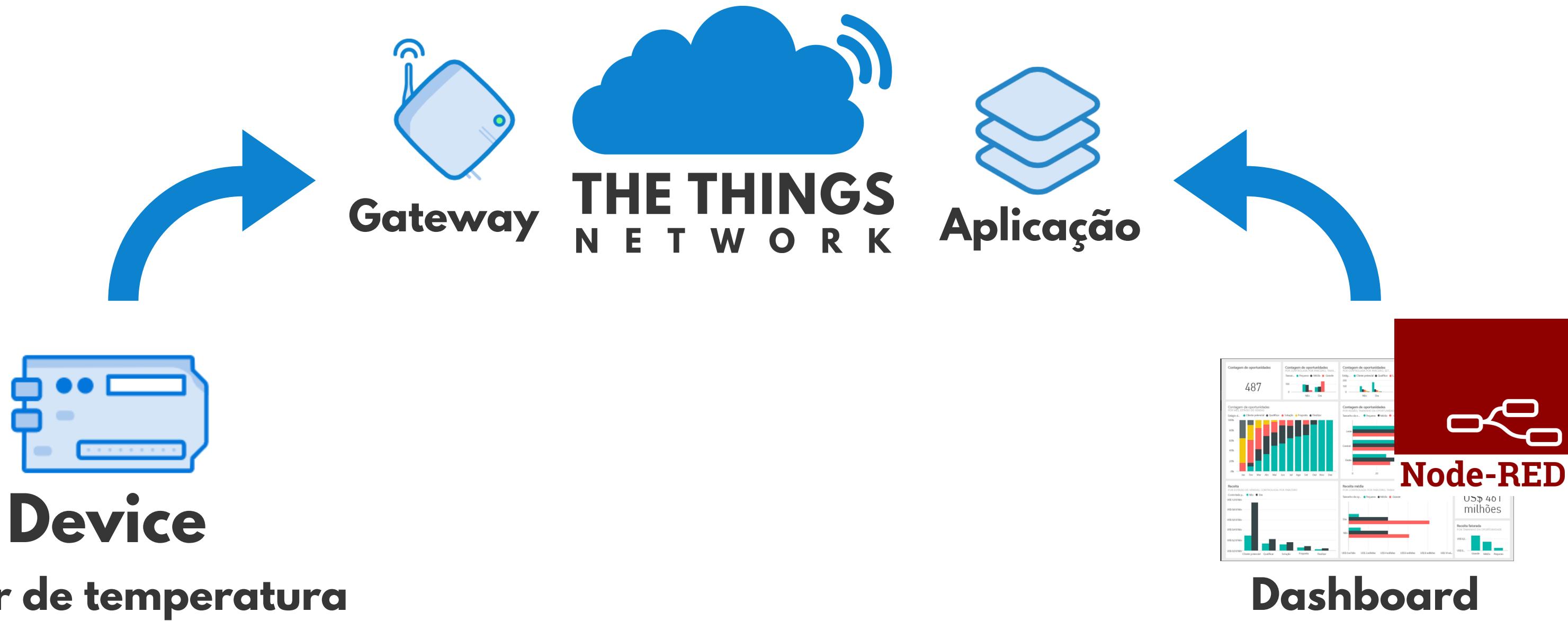
# **APLICAÇÃO LORAWAN THE THINGS NETWORK**

**Maria Fernanda Tutui**

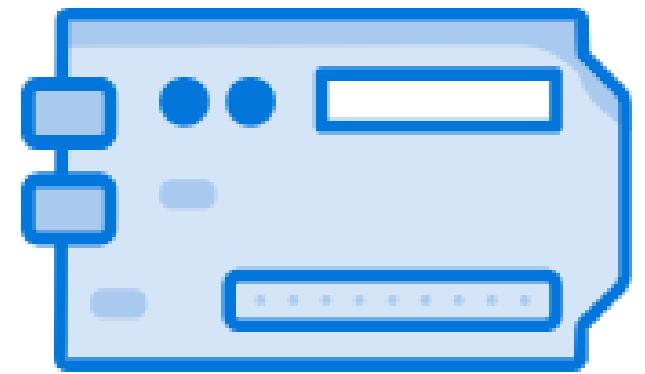


**[github.com/mftutui/ttn-lorawan-application](https://github.com/mftutui/ttn-lorawan-application)**

# Cenário



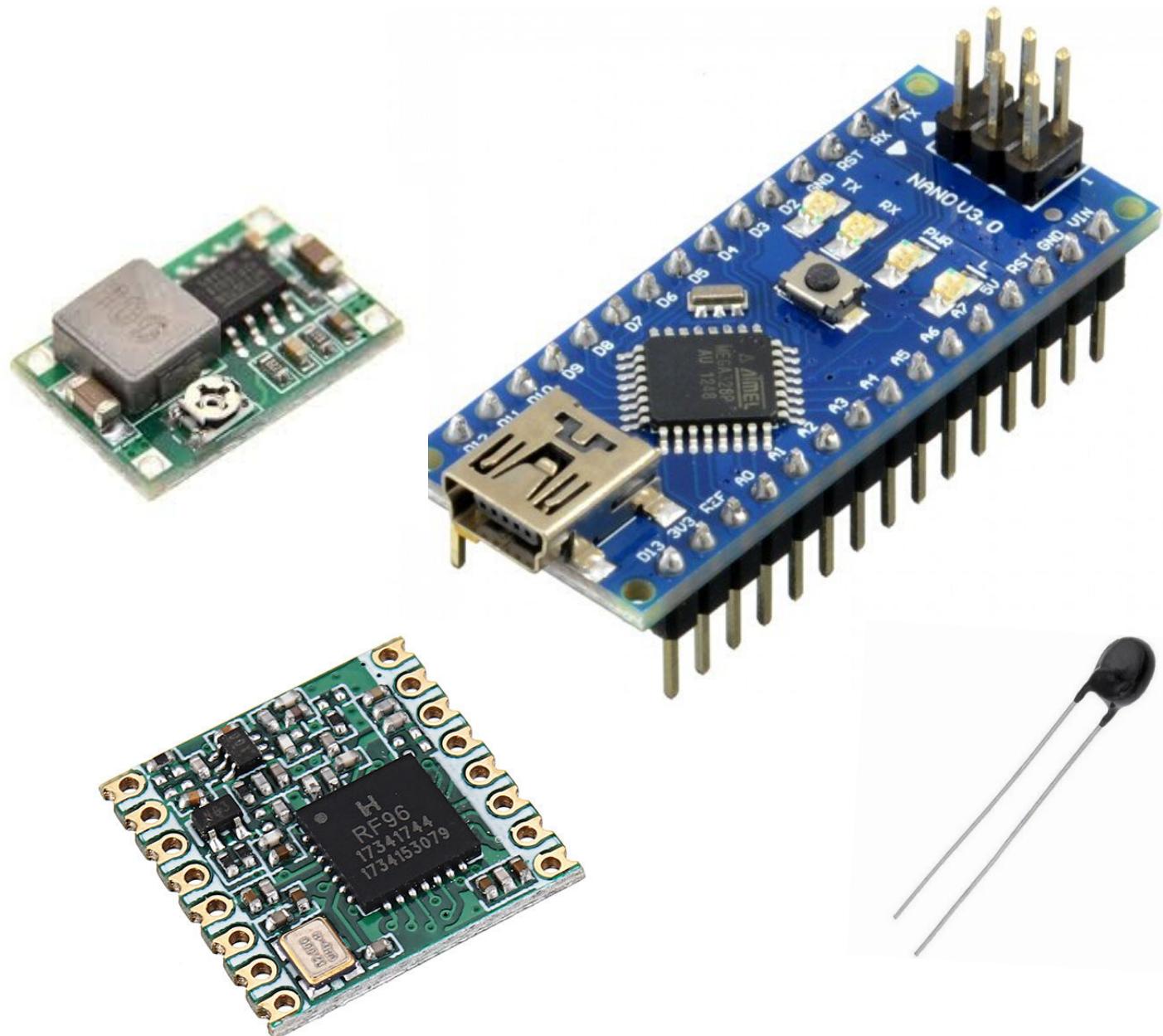
# Fase 1/5



## Device: montagem

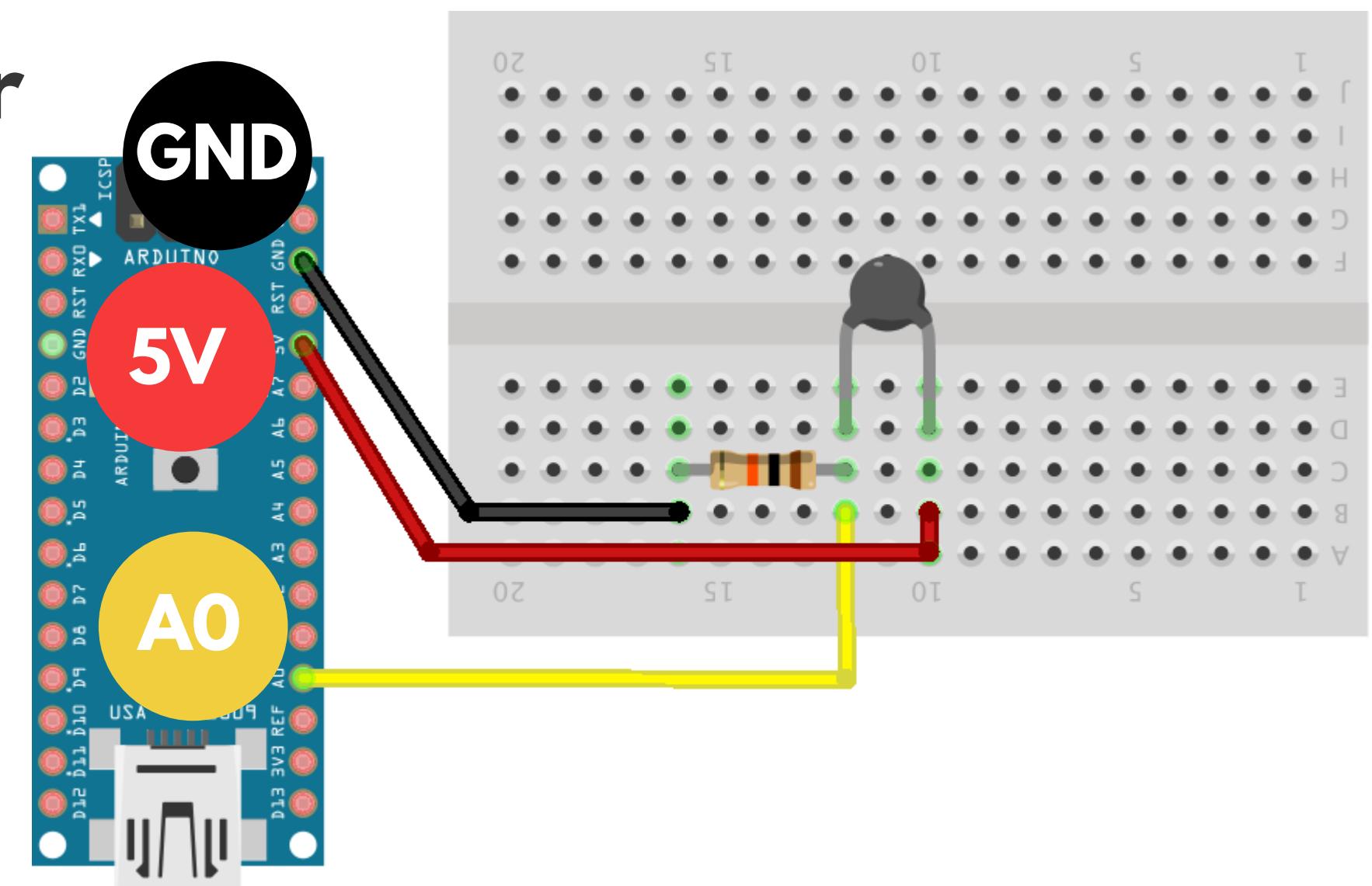
# Device: montagem

- **Arduino Nano**
- **Regulador de tensão**
- **Módulo Transceptor LoRa RF96**
- **Sensor de temperatura NTC**

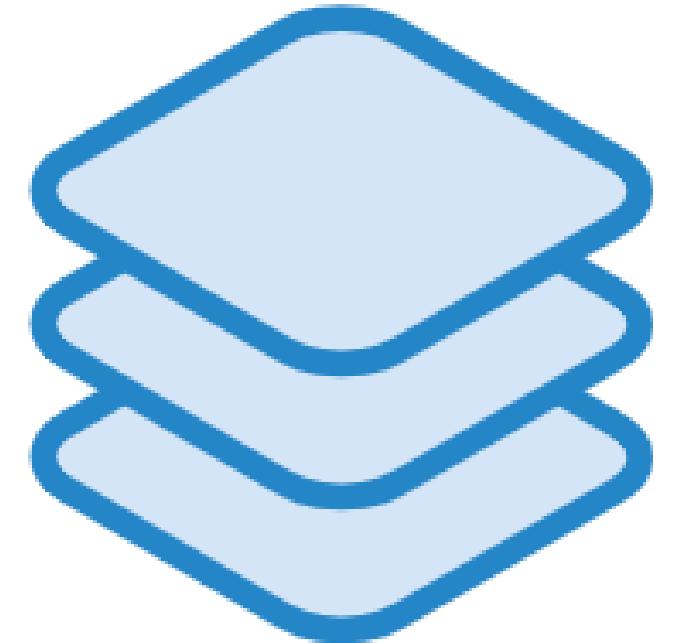


# Device: montagem

- Montagem com o sensor de temperatura



# Fase 2/5



## Aplicação: criação

# Aplicação: criação

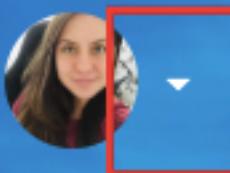
- 1. Criar uma aplicação**  
[thethingsnetwork.org](https://thethingsnetwork.org)
- 2. Associar um dispositivo**



**THE THINGS  
N E T W O R K**

[Communities](#)[Learn](#)[Support](#)[Forum](#)[Marketplace](#)

Hi Maria Fernanda

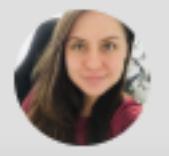


# Building a global open LoRaWAN™ network.



[Communities](#)[Learn](#)[Support](#)[Forum](#)[Marketplace](#)

Hi Maria Fernanda

[My Profile](#)[Console](#)[Log Out](#)

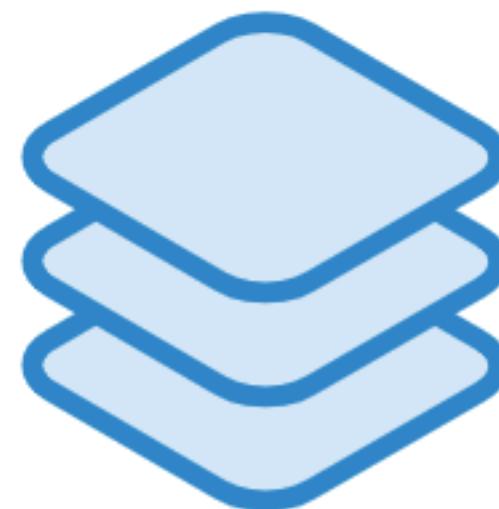
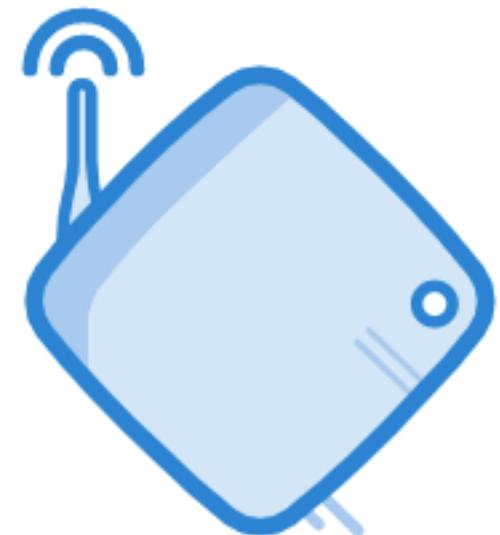
# Building a global open LoRaWAN™ network.



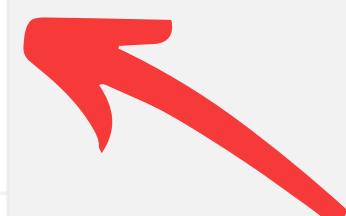
 **Hi, Maria Fernanda!**

Welcome to The Things Network Console.

This is where the magic happens. Here you can work with your data. Register applications, devices and gateways, manage your integrations, collaborators and settings.

**APPLICATIONS****GATEWAYS**

## Applications

**APPLICATIONS** add application

 <b>new example</b>	Aplicación para manejo de downlink e uplink (aprovechando PET)	 <b>new handle</b>	79 83 05 79 00 01 00 00
 <b>new handle</b>	Aplicación para traza de SF usando node-red	 <b>new handle</b>	79 83 05 79 00 01 00 03
 <b>new handle</b>	Sensores de temperatura, luminosidad, ultrasonidos e LCD	 <b>new handle</b>	79 83 05 79 00 01 00 08
 <b>new handle or new</b>	Sensores de temperatura, luminosidad, ultrasonidos e LCD (new+new)	 <b>new handle or new</b>	79 83 05 79 00 01 00 06
 <b>new</b>	Node-red (new)	 <b>new handle</b>	79 83 05 79 00 01 00 03
 <b>new device</b>	End Device 2 WFC SU	 <b>new handle</b>	79 83 05 79 00 01 00 04
 <b>new module</b>	RFID	 <b>new handle</b>	79 83 05 79 00 01 00 05

## ADD APPLICATION

### Application ID

The unique identifier of your application on the network



### Description

A human readable description of your new app

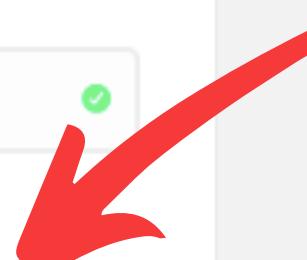


### Application EUI

An application EUI will be issued for The Things Network block for convenience, you can add your own in the application settings page.

### Handler registration

Select the handler you want to register this application to



Cancel

Add application

[Overview](#) [Devices](#) [Payload Formats](#) [Integrations](#) [Data](#) [Settings](#)

## APPLICATION OVERVIEW

[documentation](#)**Application ID** [hackathon-ttn-florianopolis](#)**Description** Sensor de temperatura**Created** 6 days ago**Handler** meshed-handler

## APPLICATION EUIS

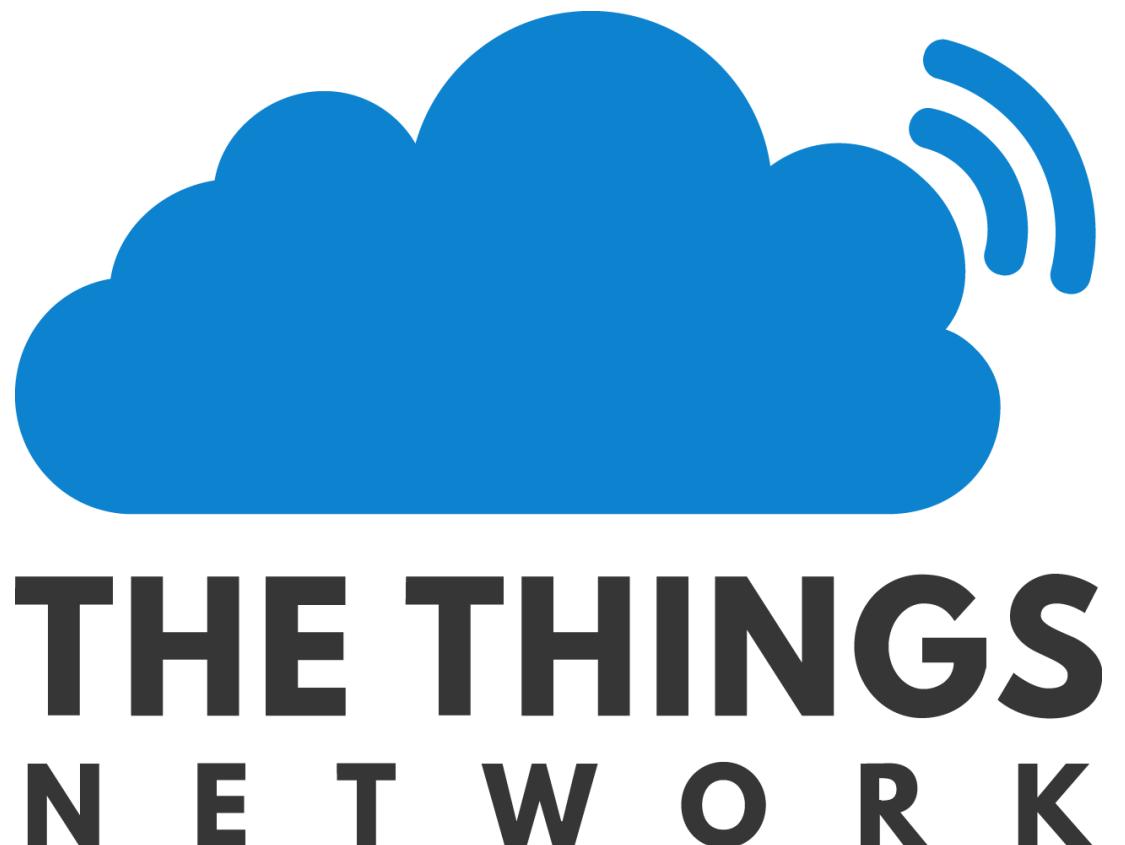
[manage euis](#)  70 B3 D5 7E D0 02 4E AE 

## DEVICES

[register device](#) [manage devices](#)

# Aplicação: criação

- 1. Criar uma aplicação**  
[thethingsnetwork.org](https://thethingsnetwork.org)
- 2. Associar um dispositivo**



[Overview](#) [Devices](#) [Payload Formats](#) [Integrations](#) [Data](#) [Settings](#)

## APPLICATION OVERVIEW

[documentation](#)**Application ID** **hackathon-ttn-florianopolis****Description** Sensor de temperatura**Created** 5 days ago**Handler** meshed-handler

## APPLICATION EUIS

[manage euis](#)  **70 B3 D5 7E D0 02 4E AE** 

## DEVICES

[register device](#) [manage devices](#)

## REGISTER DEVICE

[bulk import devices](#)

### Device ID

This is the unique identifier for the device in this app. The device ID will be immutable.



### Device EUI

The device EUI is the unique identifier for this device on the network. You can change the EUI later.



0 bytes

### App Key

The App Key will be used to secure the communication between your device and the network.



this field will be generated

### App EUI

70 B3 D5 7E D0 02 4E AE



Cancel

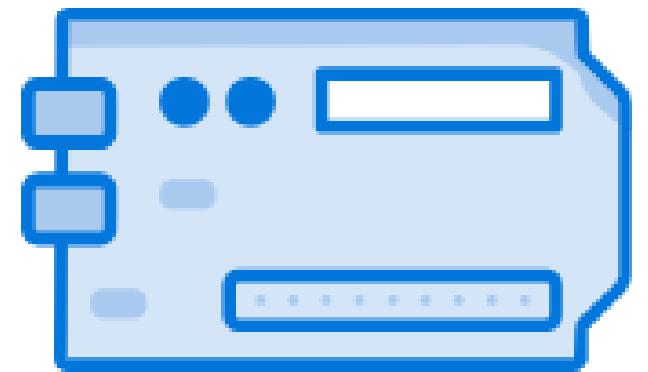
Register

[Overview](#) [Data](#) [Settings](#)

## DEVICE OVERVIEW

**Application ID** [hackathon-ttn-florianopolis](#)**Device ID** device-sensor-temperatura**Activation Method** [OTAA](#)**Device EUI** [!\[\]\(9c7a728b22e5d7455ab257bb0ec5eaf2\_img.jpg\)](#) 00 64 97 BC 89 41 38 5D [!\[\]\(7996c6d514dc3415f6eb7adfac834252\_img.jpg\)](#)**Application EUI** [!\[\]\(c19358fd94e0cf6da112c93f72051a9c\_img.jpg\)](#) 70 B3 D5 7E D0 02 4E AE [!\[\]\(e9cb59f1cab4c2a1a8a08ad26198f050\_img.jpg\)](#)**App Key** [!\[\]\(378038420973c474427b09381a3aac1a\_img.jpg\)](#) [!\[\]\(81f51d316c4207427fd9c1a75429d548\_img.jpg\)](#) ..... [!\[\]\(23ca8631d321341ab400f733110c9133\_img.jpg\)](#)**Status**  never seen**Frames up** 0 [reset frame counters](#)**Frames down** 0

# Fase 3/5



**Device: Arduino IDE**

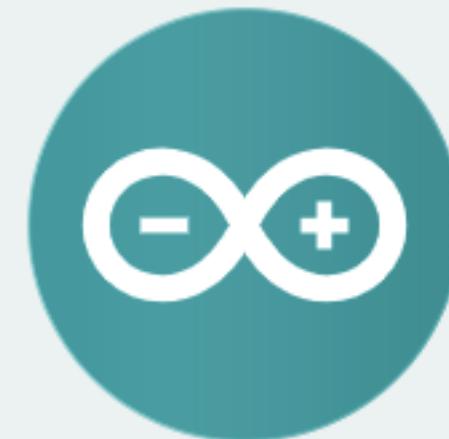
# Device: Arduino IDE

- 1. Baixar a IDE do Arduino**
- 2. Baixar driver para Arduino Nano V3.0**
- 3. Incluir bibliotecas**



# Device: Arduino IDE

## Arduino IDE

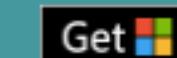
[HOME](#)[STORE](#)[SOFTWARE](#)[EDU](#)[RESOURCES](#)[COMMUNITY](#)[HELP](#)[SIGN IN](#)

### ARDUINO 1.8.10

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software. This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions.

**Windows** Installer, for Windows XP and up  
**Windows** ZIP file for non admin install

**Windows app** Requires Win 8.1 or 10



**Mac OS X** 10.8 Mountain Lion or newer

**Linux** 32 bits

**Linux** 64 bits

**Linux** ARM 32 bits

**Linux** ARM 64 bits

[Release Notes](#)

[Source Code](#)

[Checksums \(sha512\)](#)

# Device: Arduino IDE

- 1. Baixar a IDE do Arduino**
- 2. Baixar driver para Arduino Nano V3.0**
- 3. Incluir bibliotecas**



# Device: Arduino IDE

## Driver CH340/CH341USB



Entrar | Registrar



Centro de produtos

Plano de aplicação

Comunidade Yuheng

Suporte de serviço

Sobre Yuheng

Todos



Recursos de desenvolvimento

Drive & Ferramentas

Outros

Contate-nos

### Informações relacionadas

Nome dos dados

1. Introdução

[CH341SER.EXE](#)

CH340 / CH341USB para driver de WINDOWS da porta serial, suporte para Windows 10/ 8.1 / 8/7 / VISTA / XP de 32/64 bits, SERVIDOR 2016/2012/2008/2003, 2000 / ME / 98, autenticação de assinatura digital da Microsoft, suporte USB Transfira para portas seriais de 3 e 9 fios para distribuição aos usuários finais.

[CH341SER\\_LINUX.ZIP](#)

CH340 / CH341 USB para driver LINUX serial, suporta sistema de 32/64 bits.

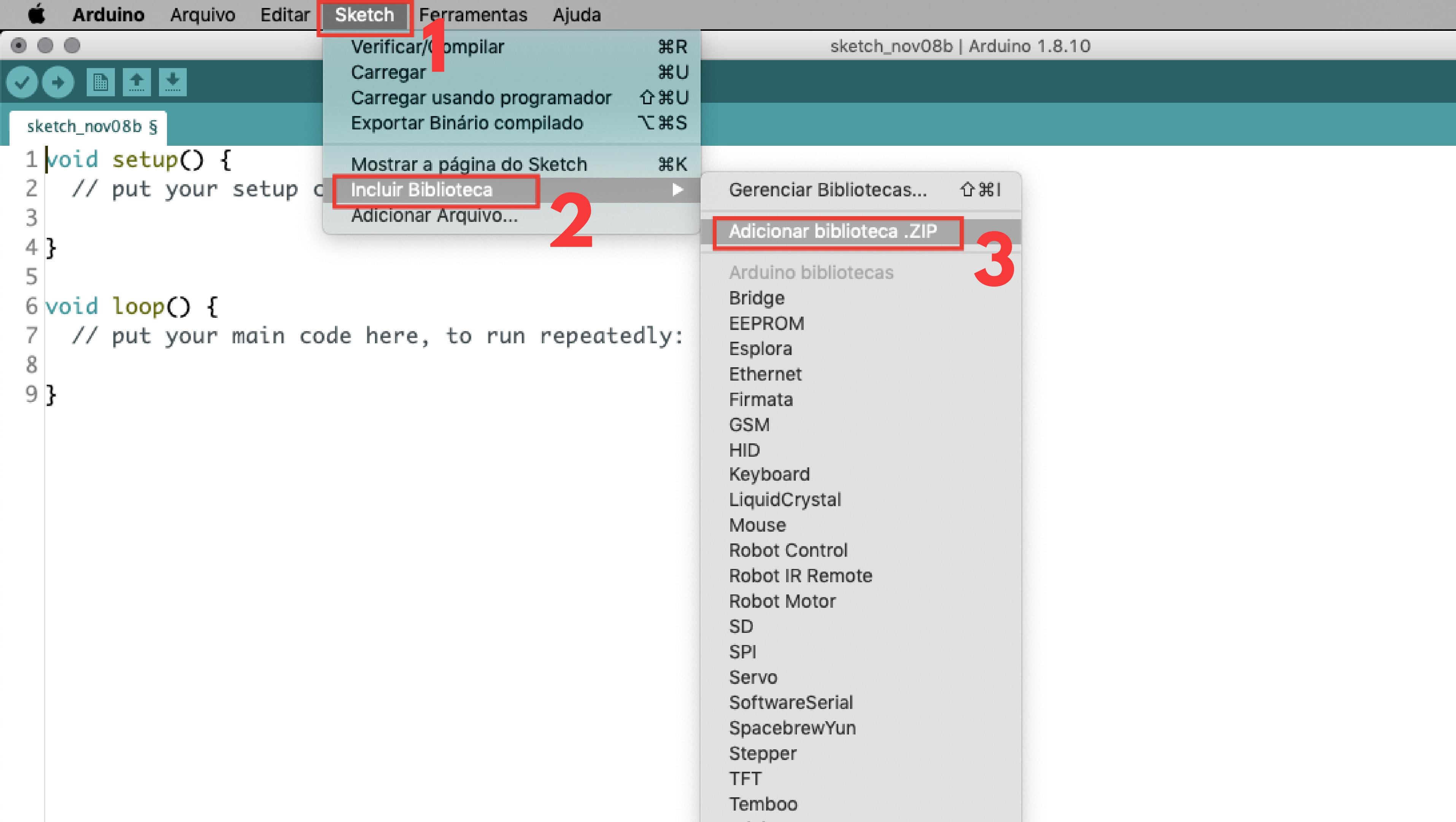
[CH341SER\\_MAC.ZIP](#)

CH340 / CH341 USB para driver MAC OS serial, suporta sistema de 32/64 bits, há instruções de uso.

# Device: Arduino IDE

- 1. Baixar a IDE do Arduino**
- 2. Baixar driver para Arduino Nano V3.0**
- 3. Incluir bibliotecas**





# Device: Arduino IDE

**4. Abrir código base nos exemplos**

**5. Completar com APPEUI, DEVEUI, APPKEY**

- APPEUI: linha 49
- DEVEUI: linha 53
- APPKEY: linha 59

**6. Configurar IDE**

Biblioteca adicionada às suas bibliotecas. Veja o menu "Incluir"

Novo ⌘N  
Abrir... ⌘O  
Abrir Recente  
Sketchbook  
**Exemplos** ►

1 void setup() {  
2 // put your setup code here, to  
3 // initialize variables used by your  
4 }  
5  
6 void loop() {  
7 // put your main code here, to  
8 }  
9 }

Exemplos embutidos  
01.Basics  
02.Digital  
03.Analog  
04.Communication  
05.Control  
06.Sensors  
07.Display  
08.Strings  
09.USB  
10.StarterKit\_BasicKit  
11.ArduinoISP

Exemplos para qualquer placa  
Adafruit Circuit Playground  
Bridge  
Esplora  
Ethernet  
Firmata  
GSM  
LiquidCrystal  
Robot Control  
Robot Motor  
SD  
Servo  
SpacebrewYun  
Stepper  
Temboo  
DESCONTINUADO

Exemplos para Arduino Nano  
EEPROM  
SoftwareSerial  
SPI  
Wire

Exemplos de Bibliotecas Personalizadas  
**MCCI LoRaWAN LMIC library** ►  
TheThingsNetwork  
TheThingsNode

compliance-otaa-halconfig  
**hackathon-ttn-temperatura** ►  
header\_test  
raw  
raw-feather  
raw-halconfig  
ttn-abp  
ttn-abp-feather-us915-dht22  
ttn-otaa  
ttn-otaa-feather-us915  
ttn-otaa-feather-us915-dht22  
ttn-otaa-halconfig-us915  
ttn-otaa-network-time

sketch\_nov08b | Arduino 1.8.10

# Device: Arduino IDE

**4. Abrir código base nos exemplos**

**5. Completar com APPEUI, DEVEUI, APPKEY**

- APPEUI: linha 49
- DEVEUI: linha 53
- APPKEY: linha 59

**6. Configurar IDE**



```
44
45 // This EUI must be in little-endian format, so least-significant-byte
46 // first. When copying an EUI from ttncctl output, this means to reverse
47 // the bytes. For TTN issued EUIs the last bytes should be 0xD5, 0xB3,
48 // 0x70.
49 static const u1_t PROGMEM APPEUI[8] = { FILLMEIN };
50 void os_getArtEui (u1_t* buf) { memcpy_P(buf, APPEUI, 8);}
51
52 // This should also be in little endian format, see above.
53 static const u1_t PROGMEM DEVEUI[8] = { FILLMEIN };
54 void os_getDevEui (u1_t* buf) { memcpy_P(buf, DEVEUI, 8);}
55
56 // This key should be in big endian format (or, since it is not really a
57 // number but a block of memory, endianness does not really apply). In
58 // practice, a key taken from ttncctl can be copied as-is.
59 static const u1_t PROGMEM APPKEY[16] = { FILLMEIN };
60 void os_getDevKey (u1_t* buf) { memcpy_P(buf, APPKEY, 16);}
61
62 static osjob_t sendjob;
63
64 // Schedule TX every this many seconds (might become longer due to duty
65 // cycle limitations).
66 const unsigned TX_INTERVAL = 60;
67
68 // Pin mapping
69 const lmic_pinmap lmic_pins = {
70     .nss = 10,
71     .rxtx = LMIC_UNUSED_PIN,
72     .rst = 5,
73     .dio_5 = 3 LMIC_UNUSED_BTNL
```

[Overview](#) [Data](#) [Settings](#)

## DEVICE OVERVIEW

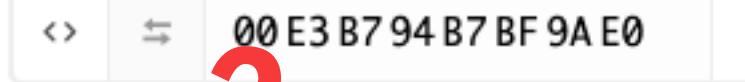
**Application ID** [hackathon-ttn-florianopolis](#)**Device ID** device-sensor-temperatura**Activation Method** OTAA**Device EUI**  00 E3 B7 94 B7 BF 9A E0 **Application EUI**  1  70 B3 D5 7E D0 02 4E AE **App Key**  ..... **Status**  never seen**Frames up** 0 [reset frame counters](#)**Frames down** 0

## DEVICE OVERVIEW

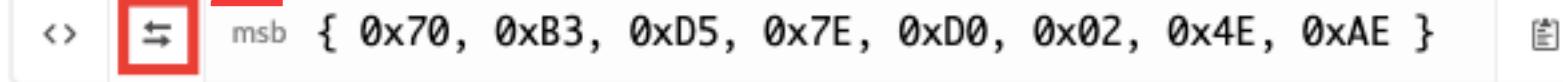
**Application ID**  hackathon-ttn-florianopolis

**Device ID** device-sensor-temperatura

**Activation Method**  OTAA

**Device EUI**  00E3B794B7BF9AE0

2

**Application EUI**  msb { 0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x02, 0x4E, 0xAE }

**App Key**  ..... 

**Status**  never seen

**Frames up** 0 [reset frame counters](#)

**Frames down** 0

[Overview](#) [Data](#) [Settings](#)

## DEVICE OVERVIEW

**Application ID** [hackathon-ttn-florianopolis](#)**Device ID** device-sensor-temperatura**Activation Method** [OTAA](#)**Device EUI** [!\[\]\(b9dd0a8b640efb5e99b498245af8b0e7\_img.jpg\)](#) 00E3B794B7BF9AE0 [!\[\]\(e705d0133a5444c18bef4c7e6697d1c1\_img.jpg\)](#)

3

**Application EUI** [!\[\]\(72c3240ee67ca6107f727634a17f171f\_img.jpg\)](#) [!\[\]\(5fbaf1ddedb1cc03be7fa2c4cf06b1f0\_img.jpg\)](#) { 0xAE, 0x4E, 0x02, 0xD0, 0x7E, 0xD5, 0xB3, 0x70 }**App Key** [!\[\]\(3271ad3f963d2e9d60af4e824e455014\_img.jpg\)](#) [!\[\]\(42aef8dbc015b88674dcfe1acd09d802\_img.jpg\)](#) [!\[\]\(cab9dcb6014d536fcc82537e13b3f7db\_img.jpg\)](#) ..... [!\[\]\(e60cb2d2f7af84ce6368cad31fcc37d0\_img.jpg\)](#)**Status**  never seen**Frames up** 0 [reset frame counters](#)**Frames down** 0

## DEVICE OVERVIEW

**Application ID**  hackathon-ttn-florianopolis

**Device ID** device-sensor-temperatura

**Activation Method**

OTAA  
**45**

**Device EUI**   lsb { 0xE0, 0x9A, 0xBF, 0xB7, 0x94, 0xB7, 0xE3, 0x00 } 

**Application EU**   lsb { 0xAE, 0x4E, 0x02, 0xD0, 0x7E, 0xD5, 0xB3, 0x70 } 

**App Key**   msb { 0x54, 0x20, 0xDE, 0xB5, 0xAC, 0x46, 0xA6, 0x16, 0xD7, 0xEE, 0x49, 0x93, 0x00 } 

**Status**  never seen

**Frames up** 0 [reset frame counters](#)

**Frames down** 0

**45** **6**  
**7** **8**

# **Obs: demais alterações**

## **1. Intervalo de envio**

Tempo entre os envios de uplink

## **2. Pin mapping**

Mapeamento dos pinos do Arduino para uso de SPI com o módulo RF95

## **3. Função "do\_send()"**

Função responsável por enviar os dados



```
hackathon-ttn-temperatura

63
64 // Schedule TX every this many seconds (might become longer due to duty
65 // cycle limitations).
66 const unsigned TX_INTERVAL = 60;
67
68 // Pin mapping
69 const lmic_pinmap lmic_pins = {
70     .nss = 10,
71     .rxtx = LMIC_UNUSED_PIN,
72     .rst = 5,
73     .dio = {2, 3, LMIC_UNUSED_PIN},
74 };
75
76 void onEvent (ev_t ev) {
77     Serial.print(os_getTime());
78     Serial.print(": ");
79     switch(ev) {
80         case EV_SCAN_TIMEOUT:
81             Serial.println(F("EV_SCAN_TIMEOUT"));
82             break;
83         case EV_BEACON_FOUND:
84             Serial.println(F("EV_BEACON_FOUND"));
85             break;
86         case EV_BEACON_MISSED:
87             Serial.println(F("EV_BEACON_MISSED"));
88             break;
89         case EV_BEACON_TRACKED:
90             Serial.println(F("EV_BEACON_TRACKED"));
91             break;
92         case EV_TTNMSG:
```



hackathon-ttn-temperatura

```
182
183 void do_send(osjob_t* j){
184
185     int ValorSensor = analogRead(A0);
186     int Temp = (ValorSensor*0.2027)-82;
187
188     Serial.println("-----");
189     Serial.print("Temperatura aproximada: ");
190     Serial.print(Temp);
191     Serial.println("°C");
192     Serial.println("-----");
193
194     u1_t mydata[1];
195     mydata[0] = Temp;
196
197     // Check if there is not a current TX/RX job running
198     if (LMIC.opmode & OP_TXRXPEND) {
199         Serial.println(F("OP_TXRXPEND, not sending"));
200     } else {
201         // Prepare upstream data transmission at the next possible time.
202         LMIC_setTxData2(1, mydata, sizeof(mydata), 0);
203         Serial.println(F("Packet queued"));
204     }
205     // Next TX is scheduled after TX_COMPLETE event.
206 }
207
208 void setup() {
209     Serial.begin(9600);
210
211     // LMTC init +
```

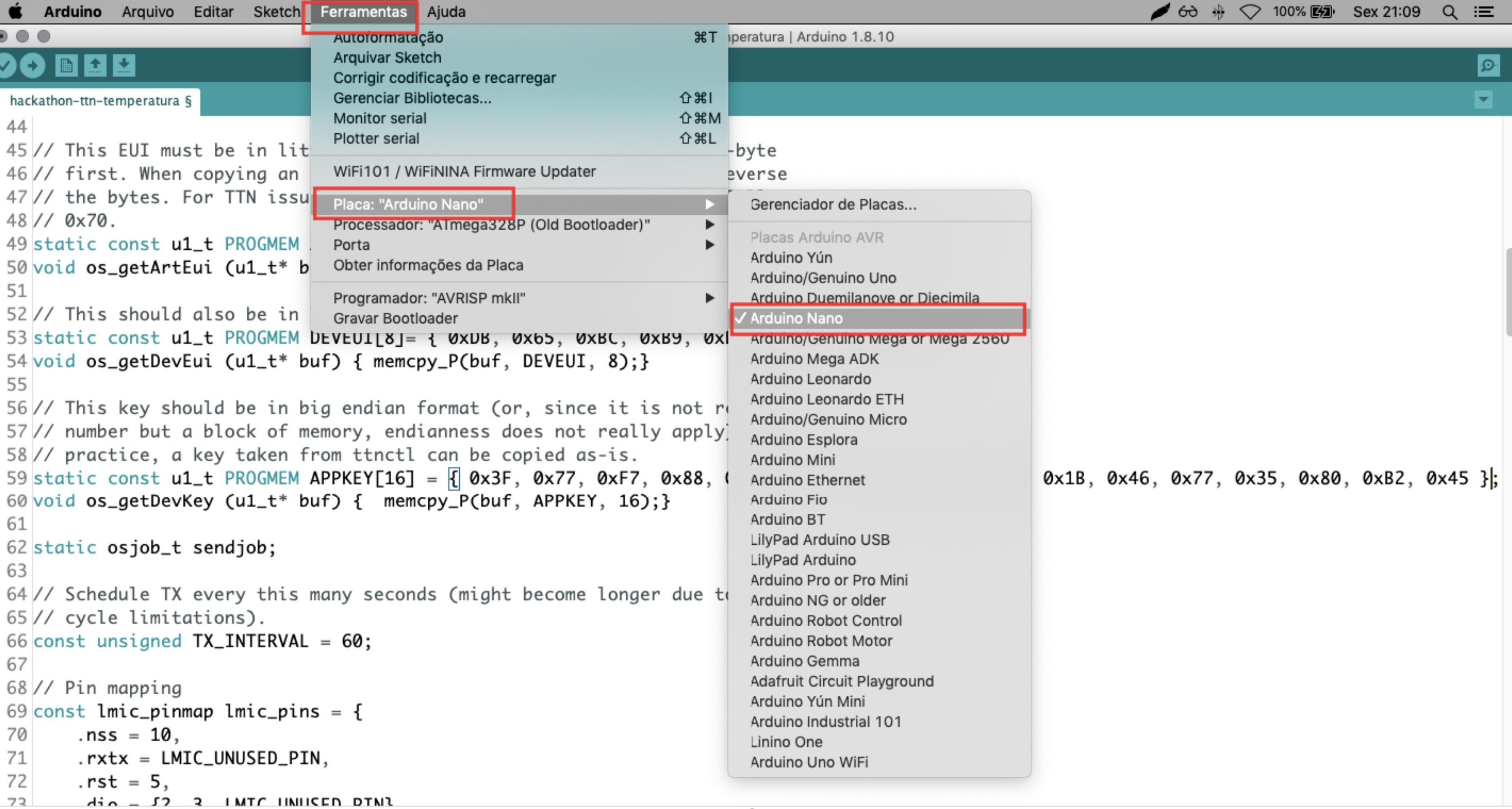
# Device: Arduino IDE

**4. Abrir código base nos exemplos**

**5. Completar com APPEUI, DEVEUI, APPKEY**

- APPEUI: linha 49
- DEVEUI: linha 53
- APPKEY: linha 59

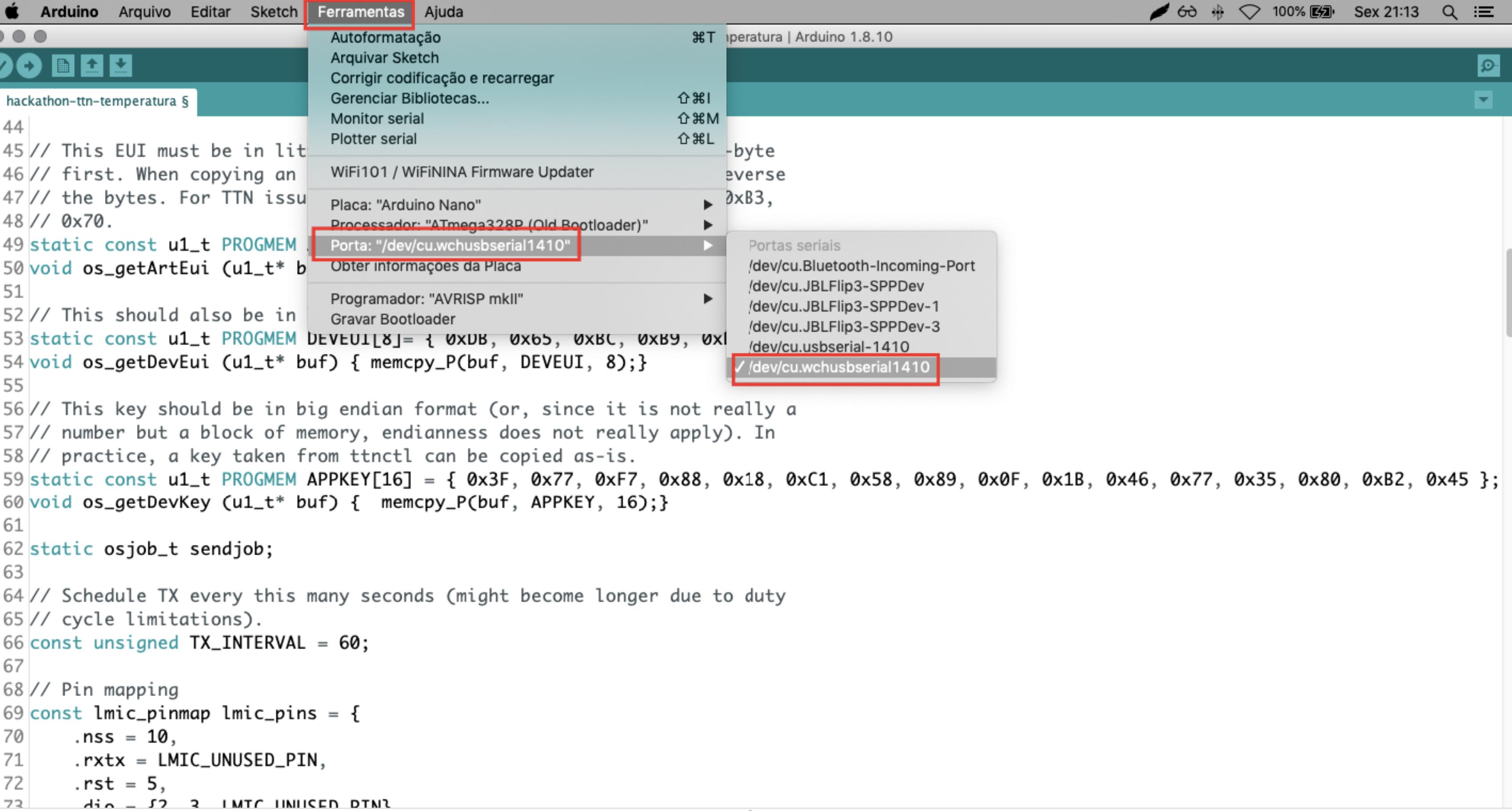
**6. Configurar IDE**



```
hackathon-ttn-temperatura §
44
45 // This EUI must be in little
46 // first. When copying an
47 // the bytes. For TTN issue
48 // 0x70.
49 static const u1_t PROGMEM
50 void os_getArtEui (u1_t* b
51
52 // This should also be in
53 static const u1_t PROGMEM DEVEUI[8] = { 0xDB, 0x65, 0xBC, 0xB9, 0xDB, 0xC2, 0x8D, 0x00 };
54 void os_getDevEui (u1_t* buf) { memcpy_P(buf, DEVEUI, 8);}
55
56 // This key should be in big endian format (or, since it is not really a
57 // number but a block of memory, endianness does not really apply). In
58 // practice, a key taken from ttncctl can be copied as-is.
59 static const u1_t PROGMEM APPKEY[16] = { 0x3F, 0x77, 0xF7, 0x88, 0x18, 0xC1, 0x58, 0x89, 0x0F, 0x1B, 0x46, 0x77, 0x35, 0x80, 0xB2, 0x45 };
60 void os_getDevKey (u1_t* buf) { memcpy_P(buf, APPKEY, 16);}
61
62 static osjob_t sendjob;
63
64 // Schedule TX every this many seconds (might become longer due to duty
65 // cycle limitations).
66 const unsigned TX_INTERVAL = 60;
67
68 // Pin mapping
69 const lmic_pinmap lmic_pins = {
70     .nss = 10,
71     .rxtx = LMIC_UNUSED_PIN,
72     .rst = 5,
73     .dio_52 = LMIC_UNUSED_DIO_52
```

Autoformatação  
Arquivar Sketch  
Corrigir codificação e recarregar  
Gerenciar Bibliotecas...  
Monitor serial  
Plotter serial  
WiFi101 / WiFiNINA Firmware Updater  
Placa: "Arduino Nano"  
Processador: "ATmega328P (Old Bootloader)"  
Porta  
Obter informações da Placa  
Programador: "AVRISP mkII"  
Gravar Bootloader

Byte  
reverse  
0xB3,  
ATmega328P  
✓ ATmega328P (Old Bootloader)  
ATmega168



hackathon-ttn-temperatura | Arduino 1.8.10

Verificar

```
44
45 // This EUI must be in little-endian format, so least-significant-byte
46 // first. When copying an EUI from ttncctl output, this means to reverse
47 // the bytes. For TTN issued EUIs the last bytes should be 0xD5, 0xB3,
48 // 0x70.
49 static const u1_t PROGMEM APPEUI[8] = { 0xAE, 0x4E, 0x02, 0xD0, 0x7E, 0xD5, 0xB3, 0x70 };
50 void os_getArtEui (u1_t* buf) { memcpy_P(buf, APPEUI, 8);}
51
52 // This should also be in little endian format, see above.
53 static const u1_t PROGMEM DEVEUI[8] = { 0xDB, 0x65, 0xBC, 0xB9, 0xDB, 0xC2, 0x8D, 0x00 };
54 void os_getDevEui (u1_t* buf) { memcpy_P(buf, DEVEUI, 8);}
55
56 // This key should be in big endian format (or, since it is not really a
57 // number but a block of memory, endianness does not really apply). In
58 // practice, a key taken from ttncctl can be copied as-is.
59 static const u1_t PROGMEM APPKEY[16] = { 0x3F, 0x77, 0xF7, 0x88, 0x18, 0xC1, 0x58, 0x89, 0x0F, 0x1B, 0x46, 0x77, 0x35, 0x80, 0xB2, 0x45 };
60 void os_getDevKey (u1_t* buf) { memcpy_P(buf, APPKEY, 16);}
61
62 static osjob_t sendjob;
63
64 // Schedule TX every this many seconds (might become longer due to duty
65 // cycle limitations).
66 const unsigned TX_INTERVAL = 60;
67
68 // Pin mapping
69 const lmic_pinmap lmic_pins = {
70     .nss = 10,
71     .rxtx = LMIC_UNUSED_PIN,
72     .rst = 5,
73     .dio_0 = 52, // LMIC_UNUSED_DTMR
```

ttn-otaa-temperatura | Arduino 1.8.10

```
44
45 // This EUI must be in little-endian format, so least-significant-byte
46 // first. When copying an EUI from ttncctl output, this means to reverse
47 // the bytes. For TTN issued EUIs the last bytes should be 0xD5, 0xB3,
48 // 0x70.
49 static const u1_t PROGMEM APPEUI[8] = { 0xAE, 0x4E, 0x02, 0xD0, 0x7E, 0xD5, 0xB3, 0x70 };
50 void os_getArtEui(u1_t* buf) { memcpy_P(buf, APPEUI, 8); }
51
52 // This should also be in little endian format, see above.
53 static const u1_t PROGMEM DEVEUI[8] = { 0xDB, 0x65, 0xBC, 0xB9, 0xDB, 0xC2, 0x8D, 0x00 };
54 void os_getDevEui(u1_t* buf) { memcpy_P(buf, DEVEUI, 8); }
55
56 // This key should be in big endian format (or, since it is not really a
57 // number but a block of memory, endianness does not really apply). In
58 // practice, a key taken from ttncctl can be copied as-is.
59 static const u1_t PROGMEM APPKEY[16] = { 0x3F, 0x77, 0xF7, 0x88, 0x18, 0xC1, 0x58, 0x89, 0x0F, 0x1B, 0x46, 0x77, 0x35, 0x80, 0xB2, 0x45 };
60 void os_getDevKey(u1_t* buf) { memcpy_P(buf, APPKEY, 16); }
61
62 static osjob_t sendjob;
63
64 // Schedule TX every this many seconds (might become longer due to duty
65 // cycle limitations).
66 const unsigned TX_INTERVAL = 60;
67
68 // Pin mapping
69 const lmic_pinmap lmic_pins = {
70     .nss = 10,
71     .rxtx = LMIC_UNUSED_PIN,
72     .rst = 5,
```

Compilando sketch...



Monitor serial



```
ttn-otaa-temperatura §
TJ TELMELI
44
45 // This EUI must be in little-endian format, so least-significant-byte
46 // first. When copying an EUI from ttncctl output, this means to reverse
47 // the bytes. For TTN issued EUIs the last bytes should be 0xD5, 0xB3,
48 // 0x70.
49 static const u1_t PROGMEM APPEUI[8] = { 0xAE, 0x4E, 0x02, 0xD0, 0x7E, 0xD5, 0xB3, 0x70 };
50 void os_getArtEui (u1_t* buf) { memcpy_P(buf, APPEUI, 8);}
51
52 // This should also be in little endian format, see above.
53 static const u1_t PROGMEM DEVEUI[8] = { 0xDB, 0x65, 0xBC, 0xB9, 0xDB, 0xC2, 0x8D, 0x00 };
54 void os_getDevEui (u1_t* buf) { memcpy_P(buf, DEVEUI, 8);}
55
56 // This key should be in big endian format (or, since it is not really a
57 // number but a block of memory, endianness does not really apply). In
58 // practice, a key taken from ttncctl can be copied as-is.
59 static const u1_t PROGMEM APPKEY[16] = { 0x3F, 0x77, 0xF7, 0x88, 0x18, 0xC1, 0x58, 0x89, 0x0F, 0x1B, 0x46, 0x77, 0x35, 0x80, 0xB2, 0x45 };
60 void os_getDevKey (u1_t* buf) { memcpy_P(buf, APPKEY, 16);}
61
62 static osjob_t sendjob;
63
64 // Schedule TX every this many seconds (might become longer due to duty
65 // cycle limitations).
66 const unsigned TX_INTERVAL = 60;
67
68 // Pin mapping
69 const lmic_pinmap lmic_pins = {
70     .nss = 10,
71     .rxtx = LMIC_UNUSED_PIN,
72     .rst = 5,
```

Carregado.

O sketch usa 24284 bytes (79%) de espaço de armazenamento para programas. O máximo são 30720 bytes.

Variáveis globais usam 1769 bytes (86%) de memória dinâmica, deixando 279 bytes para variáveis locais. O máximo são 2048 bytes.

Pouca memória disponível, problemas de estabilidade podem ocorrer.

Arduino 66 100% Sex 21:39

ttn-otaa-temperatura §

44

45 // This EUI must be in little-endian form  
46 // first. When copying an EUI from ttctrl  
47 // the bytes. For TTN issued EUIs the last  
48 // 0x70.

49 static const u1\_t PROGMEM APPEUI[8] = { 0x  
50 void os\_getArtEui (u1\_t\* buf) { memcpy\_P(

51

52 // This should also be in little endian fo  
53 static const u1\_t PROGMEM DEVEUI[8] = { 0x  
54 void os\_getDevEui (u1\_t\* buf) { memcpy\_P(

55

56 // This key should be in big endian forma  
57 // number but a block of memory, endianne  
58 // practice, a key taken from ttctrl can  
59 static const u1\_t PROGMEM APPKEY[16] = { 0  
60 void os\_getDevKey (u1\_t\* buf) { memcpy\_P(

61

62 static osjob\_t sendjob;

63

64 // Schedule TX every this many seconds (m  
65 // cycle limitations).

66 const unsigned TX\_INTERVAL = 60;

67

68 // Pin mapping

69 const lmic\_pinmap lmic\_pins = {  
70 .nss = 10,  
71 .rxtx = LMIC\_UNUSED\_PIN,  
72 .rst = 5,

Carregado.

O sketch usa 24284 bytes (79%) de espaço de ar  
Variáveis globais usam 1769 bytes (86%) de mem  
Pouca memória disponível, problemas de estabil

/dev/cu.wchusbserial1410

Enviar

Temperatura aproximada: 23°C

Packet queued

3155: EV\_JOINING

4582: EV\_TXSTART

Temperatura aproximada: 24°C

Packet queued

3180: EV\_JOINING

4607: EV\_TXSTART

347207: EV\_JOINED

netid: 19

devaddr: 260017AE

artKey: 4CFE7D4835FAD77CB24130C7FE64D834

nwkKey: BCA13D4B5BFAEDB7FBAFC82A7CD5396C

351992: EV\_TXSTART

499728: EV\_TXCOMPLETE (includes waiting for RX windows)

Auto-rolagem  Show timestamp

Nova-linha

9600 velocidade

Deleta a saída

## DEVICE OVERVIEW

**Application ID** hackathon-ttn-florianopolis

**Device ID** device-sensor-temperatura

**Activation Method** OTAA

**Device EUI** 00 E3 B7 94 B7 BF 9A E0 

**Application EUI** 70 B3 D5 7E D0 02 4E AE 

**App Key** ..... 

**Device Address** 26 01 25 42 

**Network Session Key** ..... 

**App Session Key** ..... 

**Status** ● 39 seconds ago

Frames up 7 [reset frame counters](#)

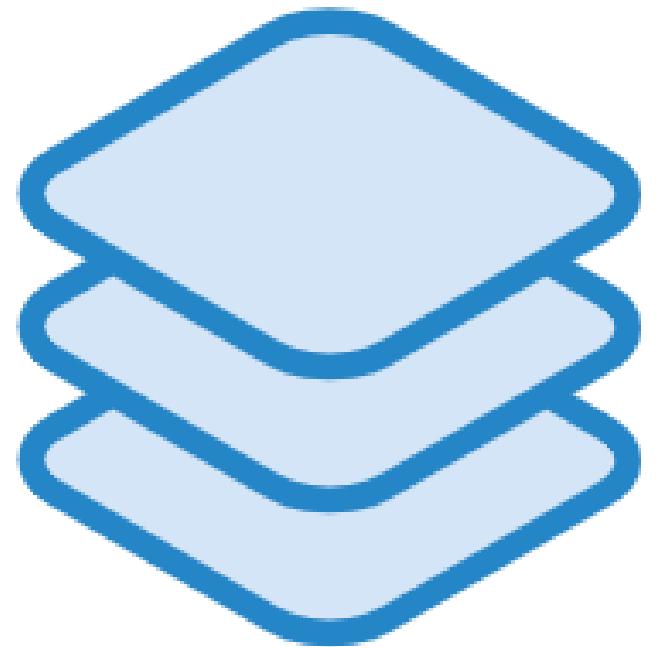
### APPLICATION DATA

II pause ⌛ clear

Filters: uplink downlink activation ack error

time	counter	port				
▲ 21:44:02	1	1	dev id:	<a href="#">device-sensor-temperatura</a>	payload:	18
▲ 21:43:00	0	1	retry	dev id:	<a href="#">device-sensor-temperatura</a>	payload: 17
⚡ 21:42:55				dev id:	<a href="#">device-sensor-temperatura</a>	dev addr: 26 06 2CCD app eui: 70 B3 D5 7E D0 02 4E AE dev eui: 00 8C

# Fase 4/5

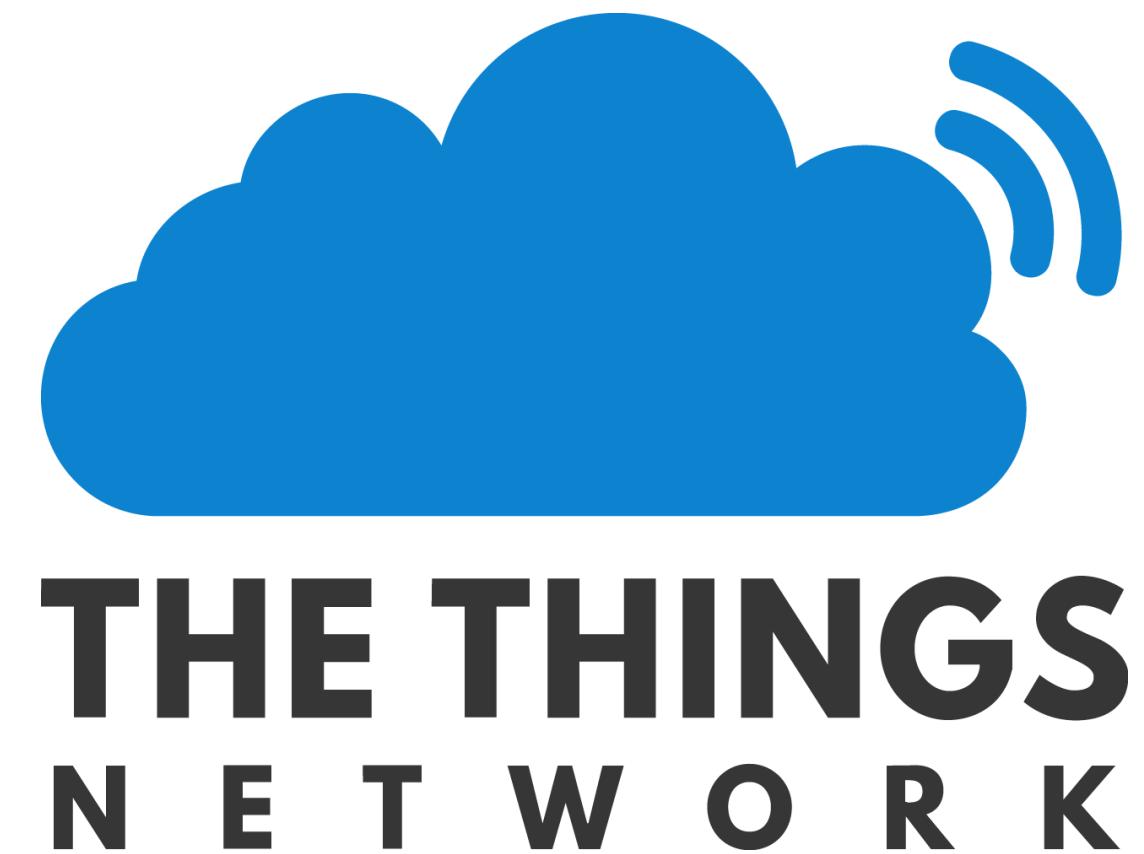


**Aplicação: payload**

# Aplicação: payload

## 1. Decodificar o payload

Tornar o payload legível transformando  
um valor hexadecimal em decimal



### APPLICATION DATA

II pause  clear

Filters [uplink](#) [downlink](#) [activation](#) [ack](#) [error](#)

time	counter	port	
▲ 00:58:13	25	1	payload: 16
▲ 00:57:11	24	1	payload: 16
▲ 00:56:08	23	1	payload: 16
▲ 00:55:06	22	1	payload: 16
▲ 00:54:04	21	1	payload: 16
▲ 00:53:01	20	1	payload: 16
▲ 00:49:54	17	1	payload: 16
▲ 00:48:52	16	1	payload: 16
▲ 00:45:44	13	1	payload: 16
▲ 00:43:39	11	1	payload: 16

[Overview](#) [Devices](#) [Payload Formats](#) [Integrations](#) [Data](#) [Settings](#)

## APPLICATION OVERVIEW

[documentation](#)**Application ID** **hackathon-ttn-floripa****Description** Sensor de temperatura**Created** 23 minutes ago**Handler** meshed-handler

## APPLICATION EUIS

[manage euis](#)   70 B3 D5 7E D0 02 54 43

## DEVICES

[register device](#) [manage devices](#)

[decoder](#) [converter](#) [validator](#) [encoder](#)[remove decoder](#)

```
1 function Decoder(bytes, port) {  
2     var value1 = bytes[0];  
3  
4     return {  
5         Temperatura: value1+"°C"  
6     };  
7 }
```

decoder has unsaved changes [undo changes](#)

## Payload

16

1 byte

1

[Test](#)

```
{  
    "Temperatura": "22°C"  
}
```

 Payload was valid

```
1 function Decoder(bytes, port) {  
2     var value1 = bytes[0];  
3  
4     return {  
5         Temperatura: value1+"°C"  
6     };  
7 }
```

decoder has unsaved changes [undo changes](#)

## Payload

16

1 byte

1

Test

```
{  
    "Temperatura": "22°C"  
}
```

■ Payload was valid

Cancel

save payload functions

## PAYLOAD FORMATS

### Payload Format

The payload format sent by your devices

Custom



[decoder](#)

converter

validator

encoder

[remove decoder](#)

```
1 function Decoder(bytes, port) {
2   var value1 = bytes[0];
3
4   return {
5     Temperatura: value1+"°C"
6   };
7 }
```

decoder has no changes

### Payload

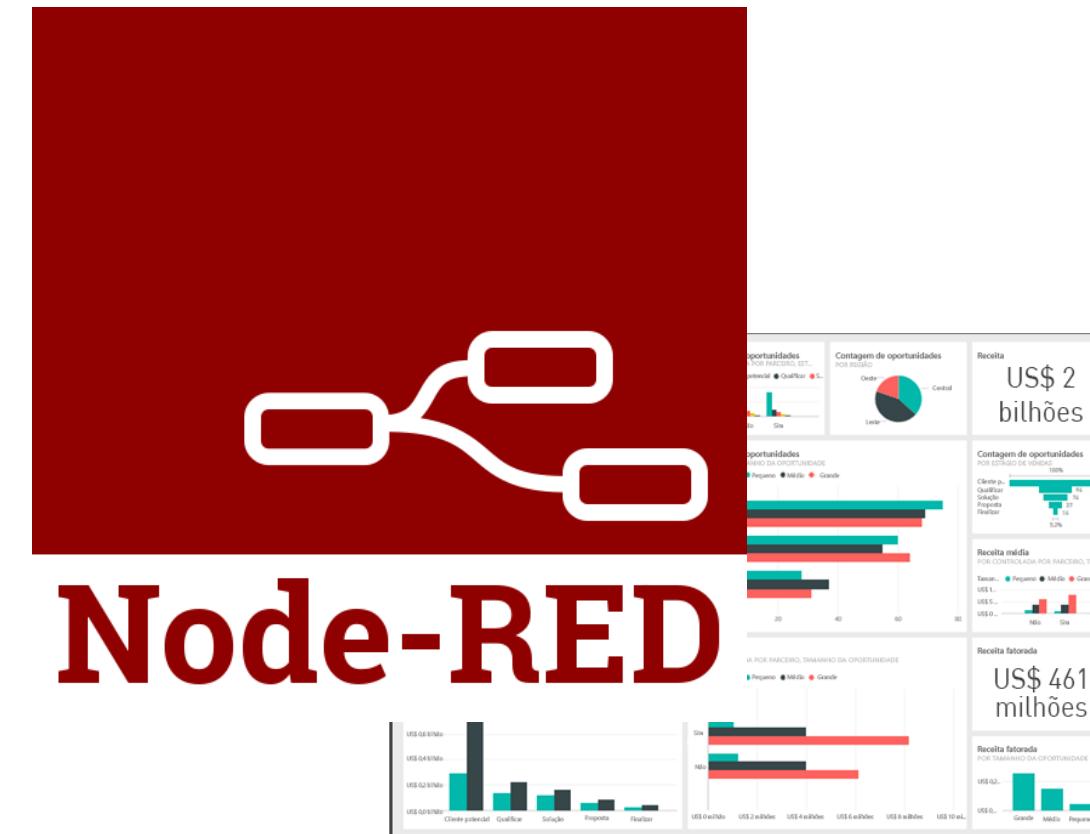
**APPLICATION DATA**

II pause  clear

Filters [uplink](#) [downlink](#) [activation](#) [ack](#) [error](#)

time	counter	port	
▲ 22:22:43	42	1	payload: 1C Temperatura: "28°C"
▲ 22:22:11	41	1	payload: 1E Temperatura: "30°C"
▲ 22:21:38	40	1	payload: 1F Temperatura: "31°C"
▲ 22:21:06	39	1	payload: 21 Temperatura: "33°C"
▲ 22:20:34	38	1	payload: 24 Temperatura: "36°C"
▲ 22:20:01	37	1	payload: 16 Temperatura: "22°C"
▲ 22:19:29	36	1	payload: 16
▲ 22:18:56	35	1	payload: 15
▲ 22:18:24	34	1	payload: 16
▲ 22:17:52	33	1	payload: 16

# Fase 5/5



**Node-RED**

## Node-RED: dashboard

# Node-RED: dashboard

**1. Instalar Node.js**

**2. Instalar Node-RED**

**Instalação**

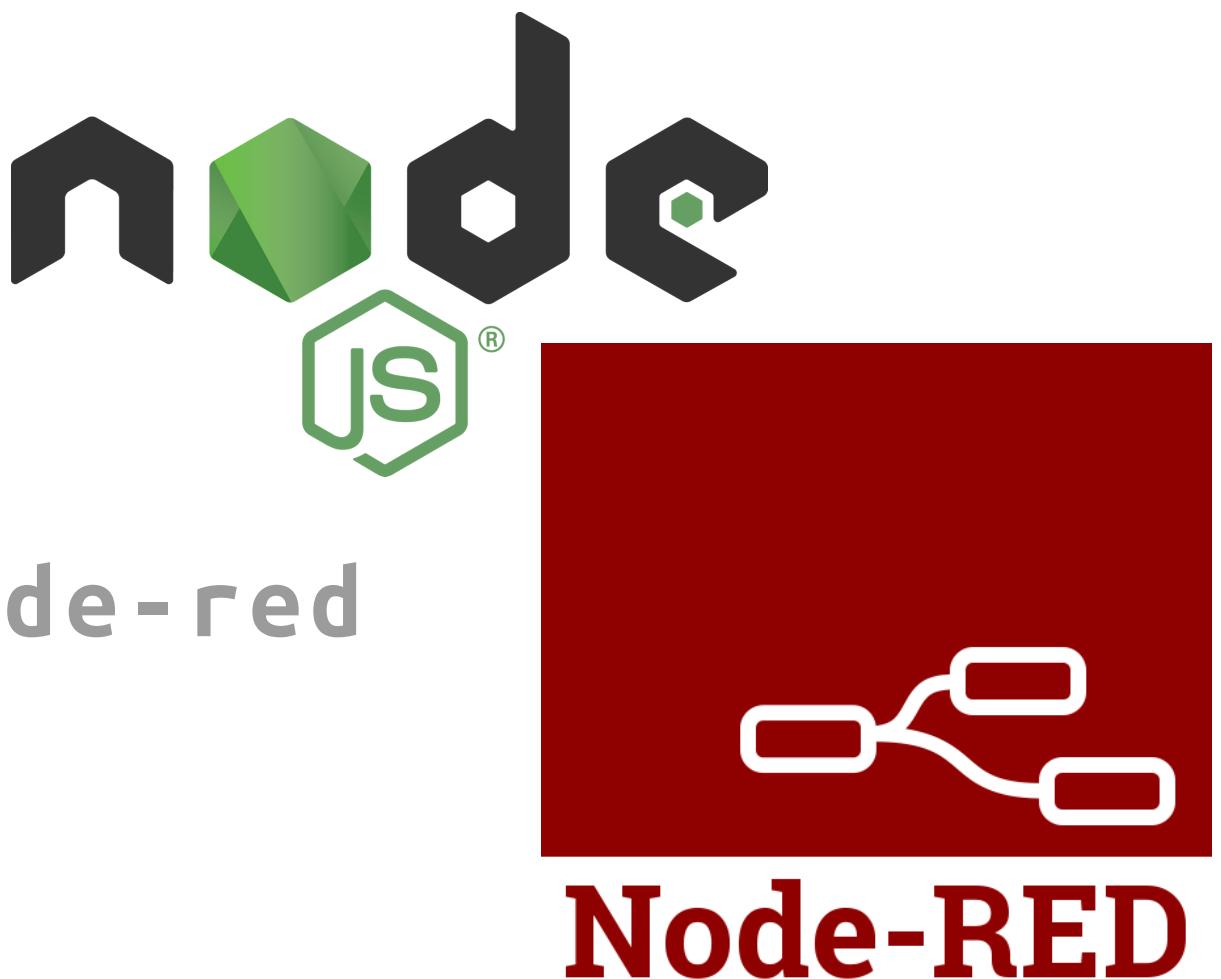
```
> npm install -g --unsafe-perm node-red
```

**Inicialização**

```
> node-red
```

**Acesso pelo navegador**

<http://127.0.0.1:1880/>



# Node-RED: dashboard

## Node.js

The screenshot shows the official Node.js website's "Downloads" page. At the top, there is a dark navigation bar with the Node.js logo and links for HOME, ABOUT, DOWNLOADS, DOCS, GET INVOLVED, SECURITY, NEWS, and FOUNDATION. The FOUNDATION link is highlighted with a green background. On the left, a large "Downloads" heading is followed by a note about the latest LTS version (12.13.0). Below this, a call to action encourages users to download source code or pre-built installers for their platform. Two main download sections are shown: "LTS" (Recommended For Most Users) and "Current" (Latest Features). Each section includes icons for Windows, macOS, and Source Code.

node

HOME | ABOUT | DOWNLOADS | DOCS | GET INVOLVED | SECURITY | NEWS | FOUNDATION

G X

## Downloads

Latest LTS Version: 12.13.0 (includes npm 6.12.0)

Download the Node.js source code or a pre-built installer for your platform, and start developing today.

### LTS

Recommended For Most Users

### Current

Latest Features

Windows Installer

macOS Installer

Source Code

# Node-RED

**1. Instalar Node.js**

**2. Instalar Node-RED**

**Instalação**

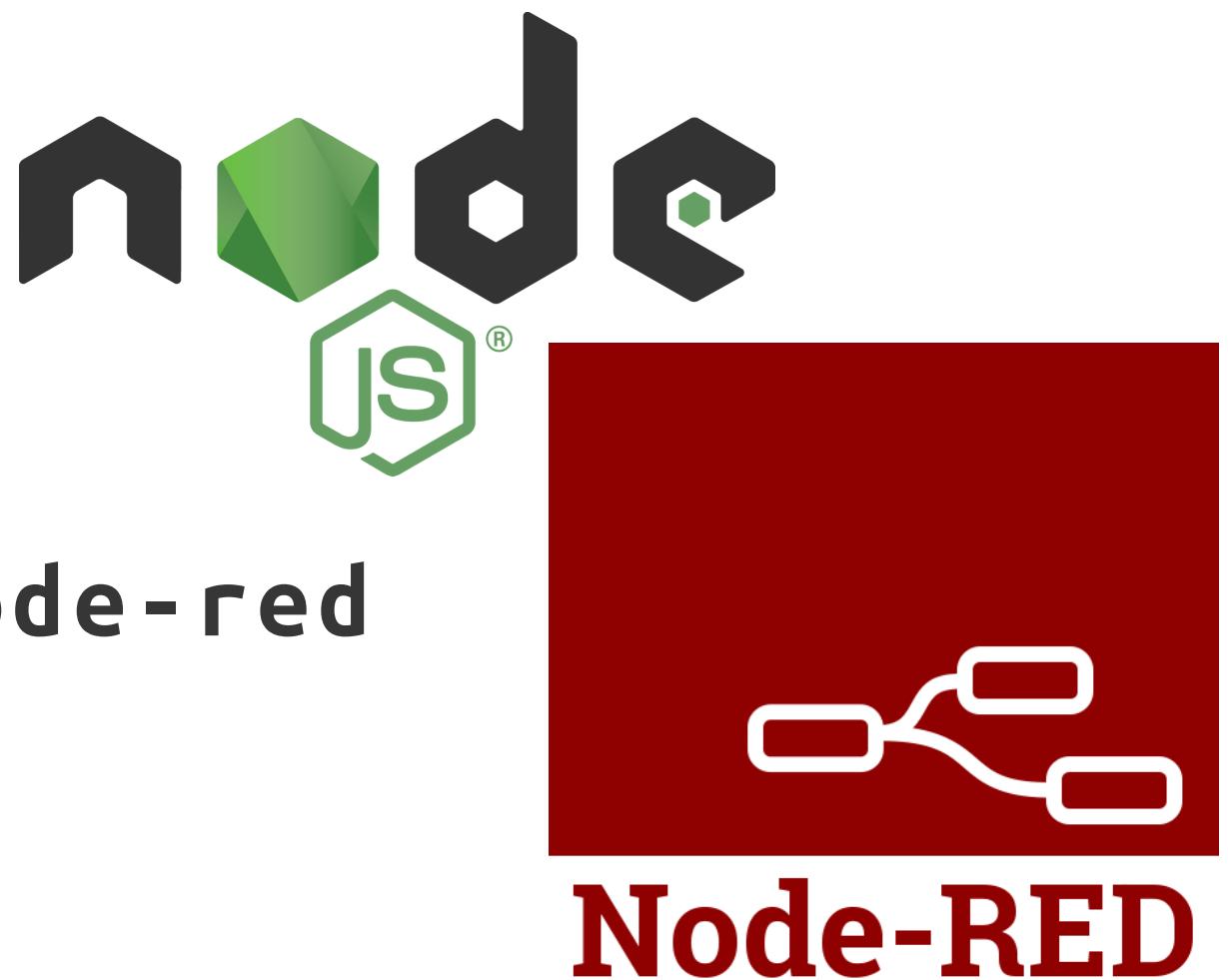
```
> npm install -g --unsafe-perm node-red
```

**Inicialização**

```
> node-red
```

**Acesso pelo navegador**

<http://127.0.0.1:1880/>



filter nodes

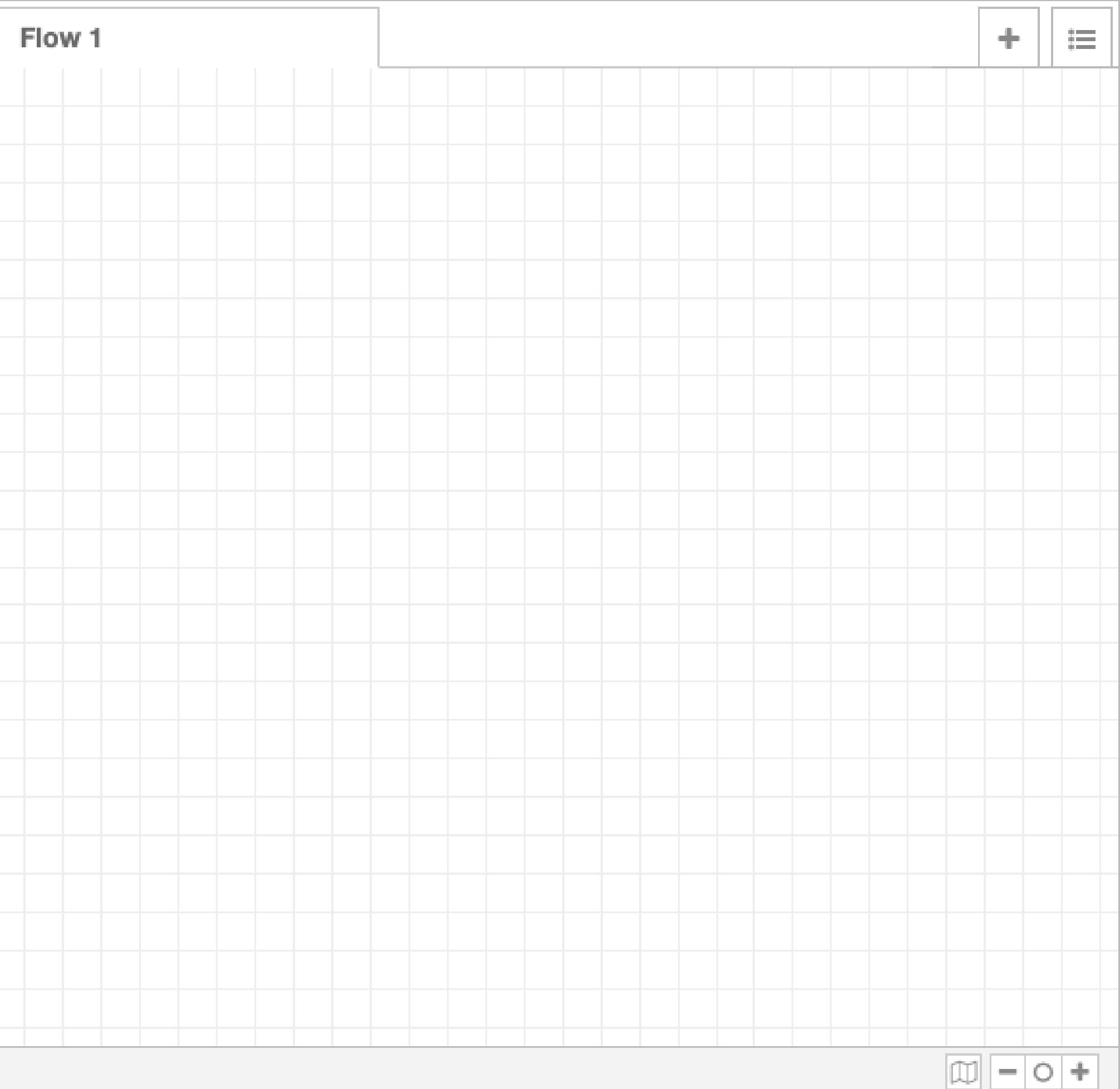
common

- inject
- debug
- complete
- catch
- status
- link in
- link out
- comment

function

- function
- switch

+   ≡



i info

View

Import

Export

Flow

Name

Status

Search flows

Configuration nodes

Flows

Subflows

Manage palette

Settings

Keyboard shortcuts

Node-RED website

v1.0.2

Hold down ⌘ when you click on  
a node port to enable quick-wiring

☰



User Settings

View      Nodes      **Install**

filter nodes

node-red  
1.0.2  
46 nodes

node-red-node-rbe  
0.2.5  
1 node

node-red-node-tail  
0.0.3  
1 node

Close

Palette



i info

Information

Flow "7e3545a4.a305dc"  
Name Flow 1  
Status Enabled

Description

You can remove the selected nodes or links with delete

Close X

User Settings

View Nodes Install

sort: ↓ a-z recent

5 / 2275 ×

node-red-dashboard

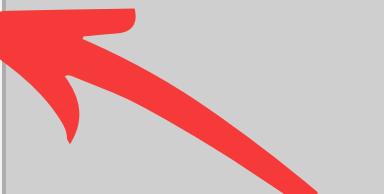
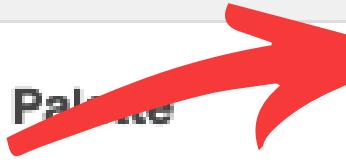
node-red-dashboard A set of dashboard nodes for Node-RED  
2.17.1 2 weeks ago install

node-red-node-ui-list Node-RED Dashboard UI widget node for simple list  
0.2.4 2 weeks ago install

node-red-node-ui-vega Node-RED UI widget node for Vega visualization grammar  
0.1.2 5 days ago install

node-red-node-ui-table Table UI widget node for Node-RED Dashboard  
0.1.5 1 month ago install

Palette



i info

Information

Flow "7e3545a4.a305dc"  
Name Flow 1  
Status Enabled

Description

Your flow configuration nodes are listed in the sidebar panel. It can be accessed from the menu or with **#g** **c**



filter node

User Settings

View Nodes

Keyboard

deb

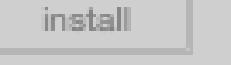
!

!

Palettes

node-red-dashboard

5 / 2275

node-red-dashboard  A set of dashboard nodes for Node-RED  
2.17.1 2 weeks ago 

node-red-node-ui-list  Node-RED Dashboard UI widget node for simple list  
0.2.4 2 weeks ago 

node-red-node-ui-vega  Node-RED UI widget node for Vega visualization grammar  
0.1.2 5 days ago 

node-red-node-ui-table  Table UI widget node for Node-RED Dashboard  
0.1.5 1 month ago 

Installing 'node-red-dashboard'

Before installing, please read the node's documentation. Some nodes have dependencies that cannot be automatically resolved and can require a restart of Node-RED.

Cancel Open node information Install

sort:  a-z recent 

Dragging a node onto a wire will splice it into the link

i info

Information

Flow "7e3545a4.a305dc"

Flow 1 Enabled

Description

C X

filter node

common

debounce

!

!

link

functions

f

share

### User Settings

View      Nodes      Install

sort:

1 / 2275

node-red-contrib-ttn

node-red-contrib-ttn

The Things Network Node-RED Application Nodes

2.0.5 10 months ago

install

A red arrow points from the 'Palette' button in the sidebar to the search bar.

i info

### Information

Flow	"7e3545a4.a305dc"
Name	Flow 1
Status	Enabled

### Description

Search for nodes using

filter nod

commo



## User Settings

deb



View

Nodes

!

Keyboard

!

Palette

!



link

!

function

f

!

## Installing 'node-red-contrib-ttn'

Before installing, please read the node's documentation. Some nodes have dependencies that cannot be automatically resolved and can require a restart of Node-RED.

Cancel

Open node information

Install

sort: ↓ a-z recent

1 / 2275

node-red-contrib-ttn

node-red-contrib-ttn

The Things Network Node-RED Application Nodes

2.0.5 10 months ago

install

Deploy

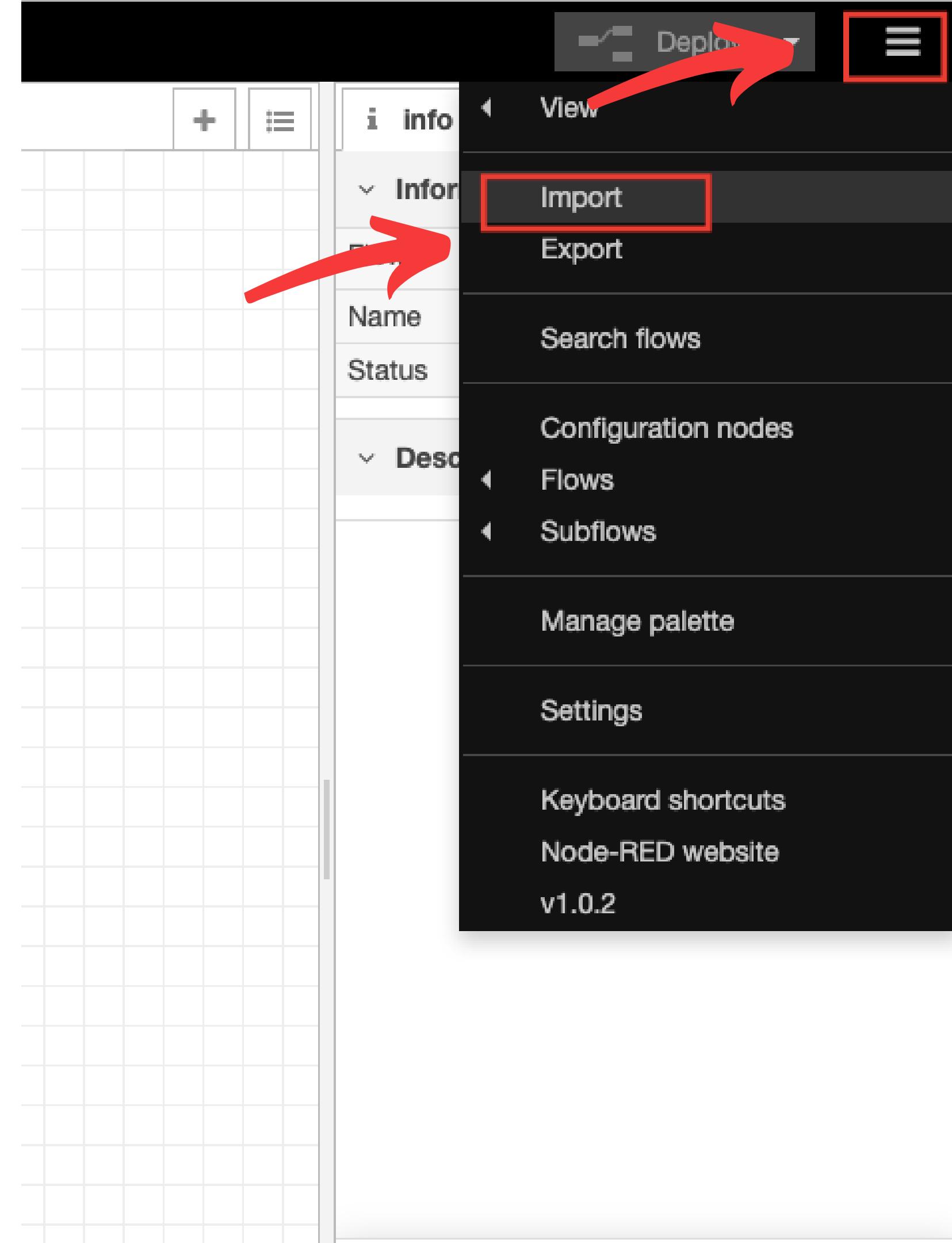
i info

Information

Flow	7e3545a4.a305dc
Flow 1	Enabled

Description

You can manage your palette of nodes with



## Import nodes

Clipboard

Library

Examples

Paste flow json or

 select a file to import

Import to

current flow

new flow

Cancel

Import

## Import nodes

### Clipboard

Paste flow json or

Library

```
[{"id":"8a0c0f41.3c76e","type":"tab","label":"Flow 1","disabled":false,"info":""}, {"id":"7624e586.0d1e7c","type":"ttn uplink","z":"8a0c0f41.3c76e","name":"Sensor","app":"","dev_id":"","field":"","x":70,"y":40,"wires": [[{"50790e92.db7b5","c2e0060e.d54eb8","806e2712.8c44c8","5a2c76ea.2b9a98","429fd1b7.9e776"}]], {"id":"50790e92.db7b5","type":"debug","z":"8a0c0f41.3c76e","name":"","active":false,"tosidebar":true,"console":false,"tostatus":false,"complete":true,"x":410,"y":40,"wires":[]}, {"id":"b00f492c.f552b8","type":"debug","z":"8a0c0f41.3c76e","name":"RSSI","active":false,"tosidebar":true,"console":false,"tostatus":false,"complete":true,"x":410,"y":240,"wires":[]}, {"id":"c42ffd9.5d4de","type":"debug","z":"8a0c0f41.3c76e","name":"SNR","active":false,"tosidebar":true,"console":false,"tostatus":false,"complete":true,"x":410,"y":200,"wires":[]}, {"id":"b453ba10.8851c8","type":"debug","z":"8a0c0f41.3c76e","name":"Device","active":false,"tosidebar":true,"console":false, "x":410,"y":160,"wires":[]}], [{"id": "8a0c0f41.3c76e", "label": "Flow 1", "x": 100, "y": 100}]]
```

Examples

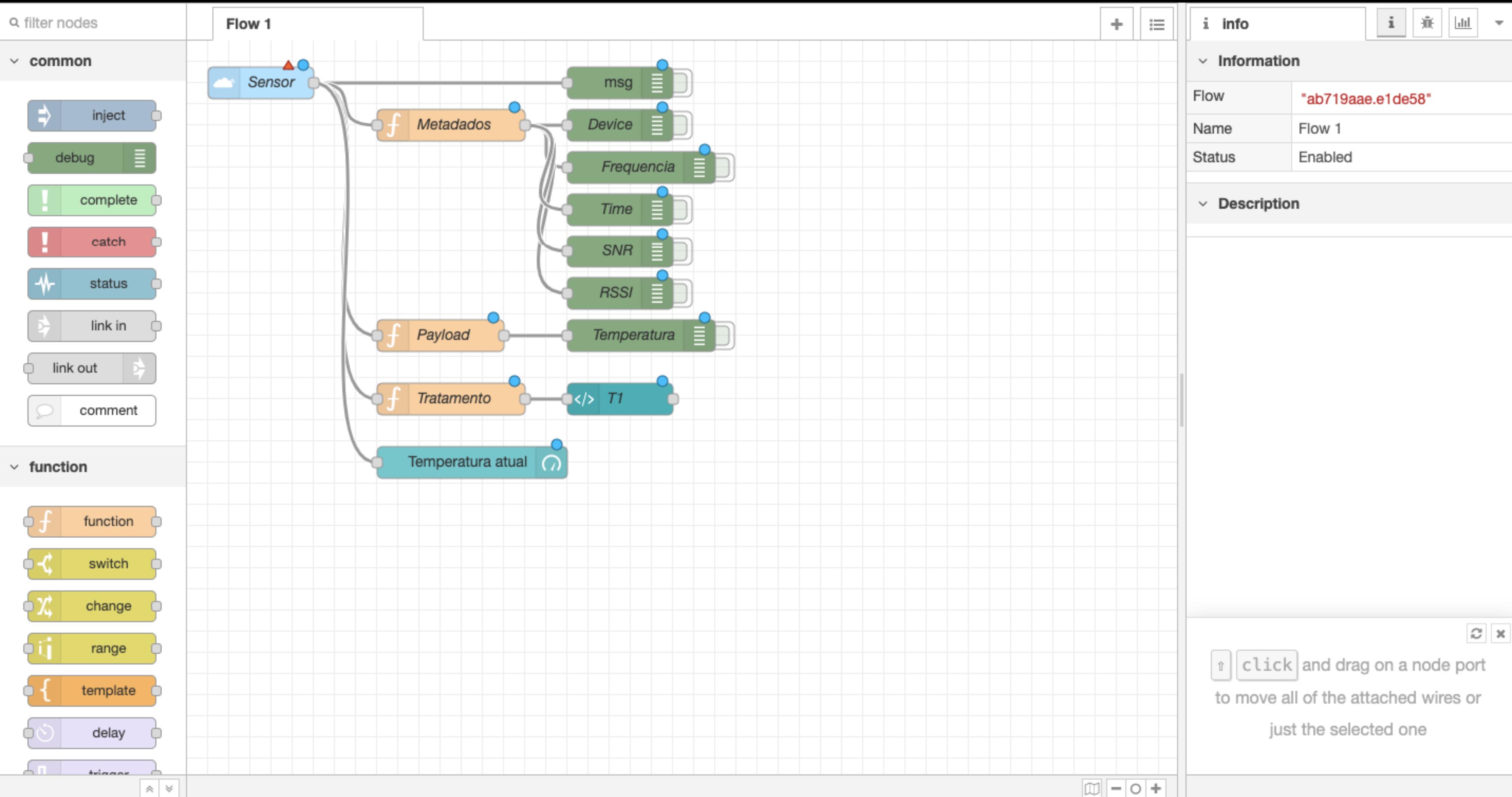
Import to

current flow

new flow

Cancel

Import



filter nodes

Flow 1

+

=

info

Information

Node: "bfb9c153.55bc5"  
Name: Sensor  
Type: ttn uplink

Description

Node Help

A node to receive uplink messages from devices on The Things Network.  
The application must be configured in the node.  
A device ID and field to filter on can also be configured.

The output message:

- **dev\_id**, the ID of the device that sent the message.
- **payload**, an object of fields or the specified field if the application is configured with a decoder payload function, or a **Buffer** if it is not.
- Unless a field was specified, the message is

Hold down ⌘ when you click on a node port to enable quick-wiring

inject

debug

complete

catch

status

link in

link out

comment

Sensor

Metadados

Device

Frequencia

Time

SNR

RSSI

Payload

Tratamento

T1

Temperatura

Temperatura atual

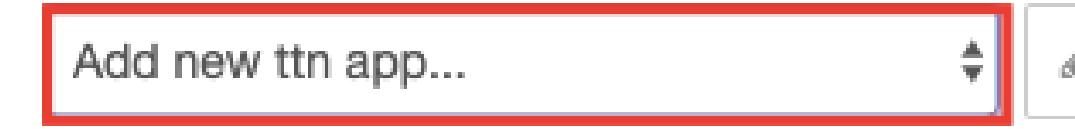
```
graph LR; Sensor --> msg[msg]; msg --> Device[Device]; msg --> Frequencia[Frequencia]; msg --> Time[Time]; msg --> SNR[SNR]; msg --> RSSI[RSSI]; msg --> Temperatura[Temperatura]; msg --> T1[T1]; msg --> TemperaturaAtual[Temperatura atual]; Metadados[Metadados] --> Device; Metadados --> Payload[Payload]; Payload --> Temperatura; Tratamento[Tratamento] --> T1;
```

### Edit ttu uplink node

[Delete](#) [Cancel](#) [Done](#)

**Properties**

**Name** Sensor 

**App** Add new ttu app... 

**Device ID**

**Field**

Edit ttu uplink node

Delete Cancel Done

**Properties**

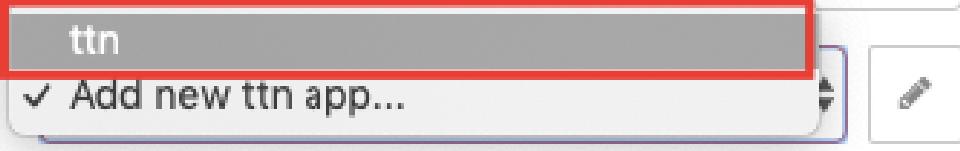
Name: Sensor

App:  

Add new ttu app... 

Device ID:

Field:



Edit ttu uplink node

Delete Cancel Done

**Properties**

Name: Sensor

App:  

Device ID:

Field:



Edit ttn uplink node > **Edit ttn app node**

**Delete** **Cancel** **Update**

**Properties**  

 App ID

 Access Key

 Discovery address



## APPLICATION OVERVIEW

[documentation](#)**Application ID** **hackathon-ttn-floripa****Description** Sensor de temperatura**Created** 2 hours ago**Handler** meshed-handler

## APPLICATION EUIS

[manage euis](#)  **70 B3 D5 7E D0 02 54 43** 

## DEVICES

 [register device](#)  [manage devices](#)**1** registered device

## DEVICES

 1 registered device

## COLLABORATORS

 tutui

[collaborators](#) [delete](#) [devices](#) [settings](#)

## ACCESS KEYS

**default key** [devices](#) [messages](#)

 ..... base64 



Edit ttn uplink node > **Edit ttn app node**

Delete

Cancel

Update

Properties



hackathon-ttn-floripa

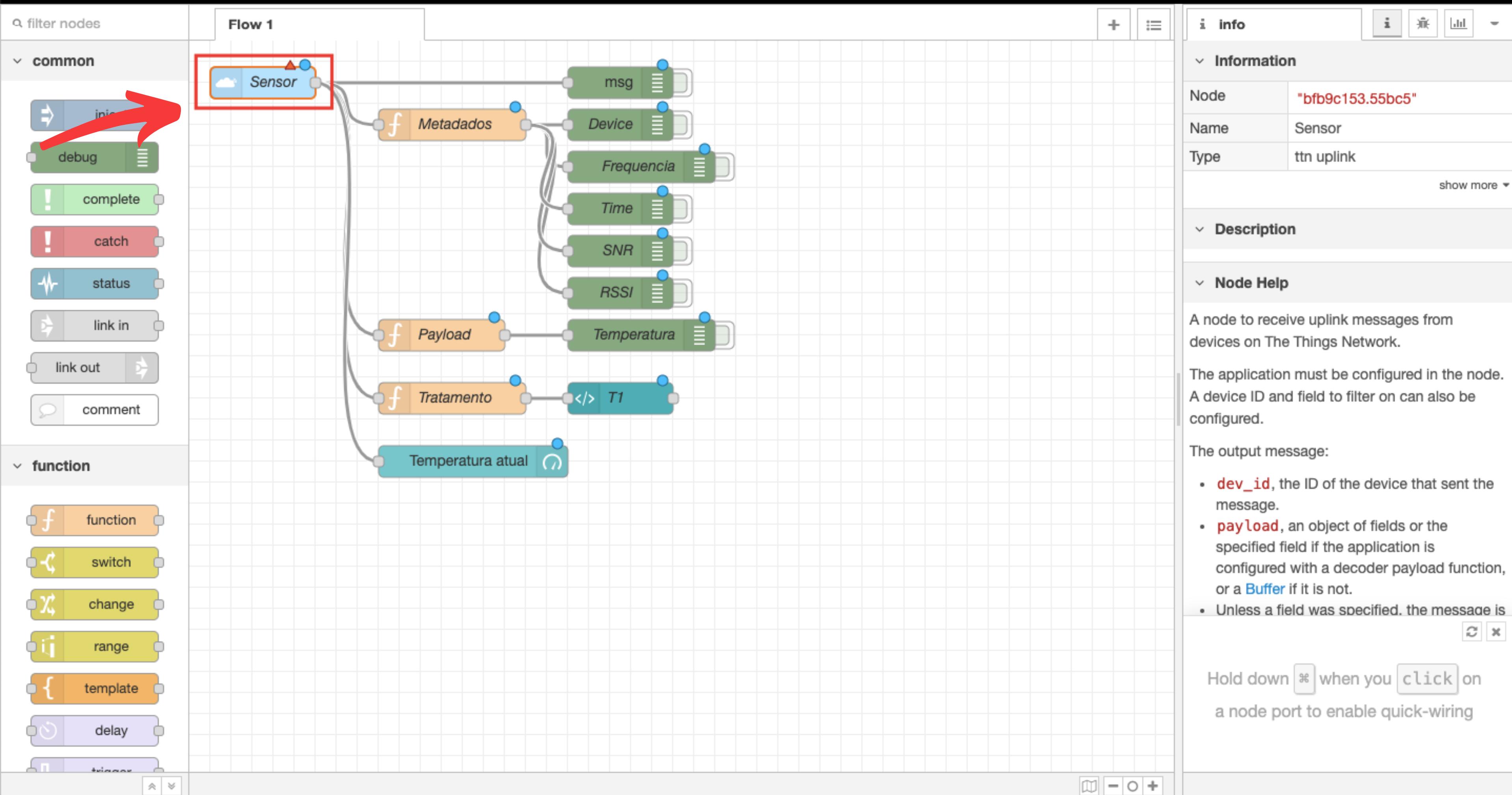


.....



Discovery  
address

discovery.thethingsnetwork.org:1900



## Edit ttn uplink node

Delete

Cancel

Done

### Properties



Name

Sensor

App

hackathon-ttn-floripa



Device ID

Field

[Overview](#) [Devices](#) [Payload Formats](#) [Integrations](#) [Data](#) [Settings](#)

## APPLICATION OVERVIEW

[documentation](#)**Application ID** **hackathon-ttn-floripa****Description** Sensor de temperatura**Created** 2 hours ago**Handler** meshed-handler

## APPLICATION EUIS

[manage euis](#)   70 B3 D5 7E D0 02 54 43 

## DEVICES

[register device](#) [manage devices](#)

[Overview](#) [Data](#) [Settings](#)

## DEVICE OVERVIEW

**Application ID** [hackathon-ttn-floripa](#)**Device ID** [device-sensor-temperatura](#)**Activation Method** [OTAA](#)**Device EUI** [!\[\]\(4924db801212cbdeeeb8d4ec3fb4cef8\_img.jpg\)](#) [!\[\]\(eb5122df28a218e573ec2334b47f78a7\_img.jpg\)](#) [00 BD 2C 77 26 E6 6F 65](#) [!\[\]\(4e3becdb4e0e1f27099188775c8f3ff3\_img.jpg\)](#)**Application EUI** [!\[\]\(b8bb5434880f645ebaa66bec084c8df9\_img.jpg\)](#) [!\[\]\(75ff033d5a40aabb090b8de8e10a88f4\_img.jpg\)](#) [70 B3 D5 7E D0 02 54 43](#) [!\[\]\(16527ff9239ca182504be64263a83a42\_img.jpg\)](#)**App Key** [!\[\]\(69bfc53a331255a776a72d846e0f36cc\_img.jpg\)](#) [!\[\]\(8028f948fa52b034fba7fcd49a33f874\_img.jpg\)](#) [!\[\]\(e1196398f603b4b13ba8090ac72eaab6\_img.jpg\)](#) [.....](#) [!\[\]\(50481ffbf1726b4a21db09391147d7dd\_img.jpg\)](#)**Device Address** [!\[\]\(46a68b268293e131dbc319d81200050a\_img.jpg\)](#) [!\[\]\(d3828e9a47e25bbc061a3549708fa930\_img.jpg\)](#) [26 06 1C 0B](#) [!\[\]\(38b54d25cb94086af727feb6c7d82c17\_img.jpg\)](#)**Network Session Key** [!\[\]\(7aa4257834ed13ee1a3c240c15d9b05a\_img.jpg\)](#) [!\[\]\(3e03b83d3d151c417ec16080608357c3\_img.jpg\)](#) [!\[\]\(b7758e36ec8e006f555c667071c578f8\_img.jpg\)](#) [.....](#) [!\[\]\(ec4558c99419dc732e3083aab109a865\_img.jpg\)](#)**App Session Key** [!\[\]\(093692436495d772cbac89c01c496b5f\_img.jpg\)](#) [!\[\]\(70c68ae3f8961769b9b30e49dd435c57\_img.jpg\)](#) [!\[\]\(4ee518fb38cdb3614613f66f69c0e6f8\_img.jpg\)](#) [.....](#) [!\[\]\(48086d637285dca8d19e73b5dafaa41e\_img.jpg\)](#)

## Edit ttn uplink node

Delete

Cancel

Done

### Properties



Name

Sensor

App

hackathon-ttn-floripa



Device ID

device-sensor-temperatura

Field

filter nodes

Flow 1

Information

Node	"bfb9c153.55bc5"
Name	Sensor
Type	ttn uplink

Description

A node to receive uplink messages from devices on The Things Network.

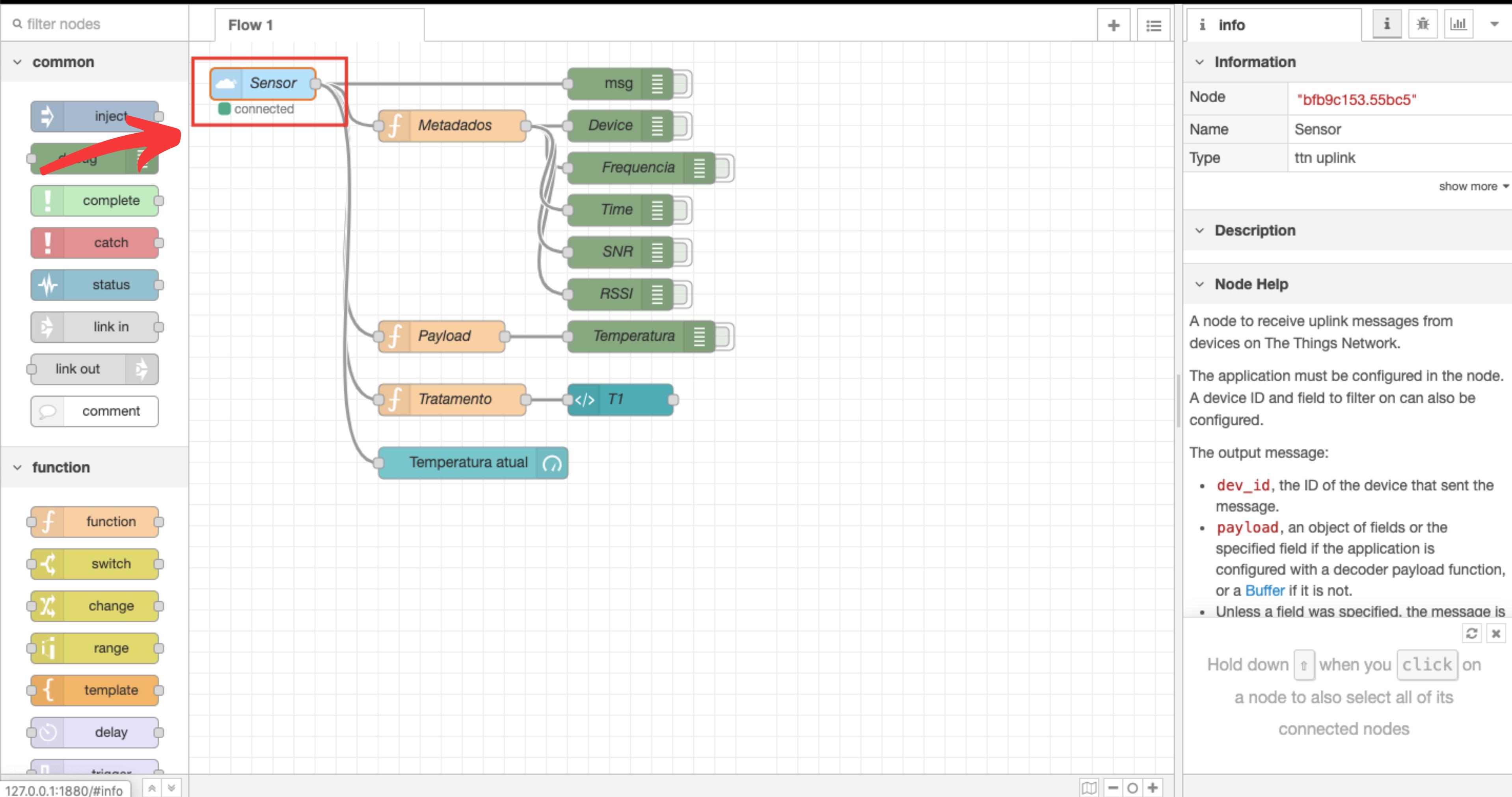
The application must be configured in the node. A device ID and field to filter on can also be configured.

The output message:

- **dev\_id**, the ID of the device that sent the message.
- **payload**, an object of fields or the specified field if the application is configured with a decoder payload function, or a **Buffer** if it is not.
- Unless a field was specified, the message is

Your flow configuration nodes are listed in the sidebar panel. It can be accessed from the menu or with **⌘g** **c**

127.0.0.1:1880/#



filter nodes

Flow 1

```
graph LR; Sensor[Sensor] --> msg1[msg]; msg1 --> Device[Device]; msg1 --> Frequencia[Frequencia]; msg1 --> Time[Time]; msg1 --> SNR[SNR]; msg1 --> RSSI[RSSI]; Sensor --> msg2[msg]; msg2 --> Payload[Payload]; msg2 --> Tratamento[Tratamento]; msg2 --> Temperatura[Temperatura]; msg2 --> T1[T1]; msg2 --> TemperaturaAtual[Temperatura atual];
```

The flow starts with a **Sensor** node connected to a **msg** node. From the **msg** node, six parallel paths branch out to **Device**, **Frequencia**, **Time**, **SNR**, **RSSI**, and another **msg** node. This second **msg** node then connects to three more parallel paths: **Payload**, **Tratamento**, and **Temperatura**. Finally, all three paths converge at a **T1** node, which then connects to a **Temperatura atual** node.

debug

08/11/2019 23:45:18 node: b4a710e0.2bb3d  
msg : Object  
▶ { app\_id: "hackathon-ttn-floripa", dev\_id: "device-sensor-temperatura", hardware\_serial: "00BD2C7726E66F65", port: 1, counter: 195 ... }

all nodes

common

- inject
- debug
- complete
- catch
- status
- link in
- link out
- comment

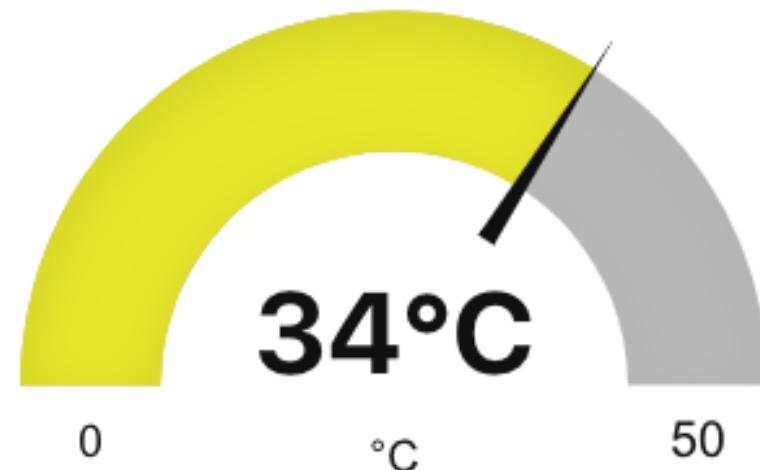
function

- function
- switch
- change
- range
- template
- delay
- trigger

## TTN - Controle de temperatura

Device	Temperatura	SNR	RSSI	Frequência
device-sensor-temperatura	34 °C	[-6.5]	[-96]	917.4Hz

Temperatura atual



# **OBRIGADA!**

**Maria Fernanda Tutui**  
**tutuimf@gmail.com**